CFD EXPERTS

Simulate the Future

WWW.CFDEXPERTS.NET

Ansys

2021 R2

Engineering What's Ahead.

Ansys Meshing User's Guide

ANSYS, Inc.
Southpointe
2600 Ansys Drive
Canonsburg, PA 15317
ansysinfo@ansys.com
http://www.ansys.com
(T) 724-746-3304
(F) 724-514-9494

Release 2021 R2
July 2021

Published in the U.S.A.

Table of Contents

List of Figures

List of Tables

Meshing: Capabilities in Ansys Workbench

The following topics are discussed in this section.

Meshing Overview

Philosophy

The goal of meshing in Ansys Workbench is to provide robust, easy to use meshing tools that will simplify the mesh generation process. These tools have the benefit of being highly automated along with having a moderate to high degree of user control.

Physics Based Meshing

When the Ansys Meshing application is launched (that is, edited) from the Ansys Workbench Project Schematic, the physics preference will be set based on the type of system being edited. For analysis systems, the appropriate physics is used (p. 39). For a Mechanical Model system, the **Mechanical** physics preference is used. For a Mesh system, the physics preference defined in **Tools> Options> Meshing> Default Physics Preference** (p. 317) is used.

Meshing Implementation in Ansys Workbench

The meshing capabilities are available within the following Ansys Workbench applications. Access to a particular application is determined by your license level.

- The Ansys Mechanical application - Recommended if you plan to stay within the Ansys Mechanical application to continue your work (preparing and solving a simulation). Also, if you are planning to perform a Fluid-Structure Interaction problem, and desire to use a single project to manage your Ansys Workbench data, you can use the Mechanical application to perform your fluid meshing. This is most conveniently done in a separate model branch from the structural meshing and structural simulation.

- The Ansys Meshing application (p. 27) - Recommended if you plan to use the mesh to perform physics simulations in Ansys CFX or Ansys Fluent. If you wish to use a mesh created in the Meshing application for a solver supported in the Mechanical application, you can replace the Mesh system with a Mechanical Model system. See Replacing a Mesh System with a Mechanical Model System (p. 36).

Note:

In the 2021 R2 release, Ansys Autodyn runs inside the Mechanical application. The recommendation is to use an Explicit Dynamics analysis system, in which meshing comes as part of that system. As an alternative, you can also use this system to prepare a model for the traditional Ansys Autodyn application (AUTODYN component system). For simple Ansys Autodyn models, you can use the meshing tools within the traditional Ansys Autodyn application (AUTODYN component system).

Types of Meshing

The following types of meshing are discussed in this section.

Assembly Level Meshing vs. Part/Body Level Meshing

"Assembly meshing" (p. 367) refers to meshing an entire model as a single mesh process, as compared to part-based or body-based meshing, in which meshing occurs at the part or body level respectively.

In part-based meshing, parts are meshed individually and have no connections (other than mesh connections (p. 444) or node merge (p. 467)). Assembly meshing performs mesh-based Boolean operations that you would otherwise have to perform manually in Ansys Discovery SpaceClaim or Ansys DesignModeler, or a CAD system. These operations include a mixture of volume filling, volume intersection, and volume combination operations that create a conformal mesh between all solids, fluids, and virtual bodies in the analysis.

Assemblies can also be meshed using part-based meshing methods, but in such cases the mesher operates one part at a time, and therefore cannot mesh virtual bodies or evaluate parts that occupy the same space.

Mesh Methods

This section describes types of meshing in terms of element shapes. Applicable mesh control options are presented for each element shape shown below, and operate at the part level unless otherwise noted. See the Method Control (p. 196) section for further details.

Tet Meshing

- Patch Conforming Tetrahedron Mesher (p. 200)
- Patch Independent Tetrahedron Mesher (p. 200)
- Tetrahedrons (p. 367) algorithm (assembly level)
- Layered Tetrahedrons Mesher (p. 240)

Hex Meshing

- Swept Mesher (p. 223)
- Hex Dominant Mesher (p. 222)
- Thin Solid Mesher (p. 224)
- MultiZone Mesher (p. 228)

Cartesian Meshing

- CutCell (p. 367) algorithm (assembly level)
- Body-Fitted Cartesian (p. 236) (part level)

Quad Meshing

- Quad Dominant (p. 245)
- MultiZone Quad/Tri (p. 246)

Triangle Meshing

- All Triangles (p. 246)

Conformal and Non-Conformal Meshing

This section describes meshing options for Conformal and Non-conformal meshing.

Conformal Meshing Within a Part

A conformal mesh can be generated in a variety of ways when meshing in Ansys Workbench. In general, a part can be used to prescribe a region where you want a conformal mesh. Unless using Assembly Meshing (p. 367), the mesh generation operates at the part level. When a part is meshed, only the geometry in that part is sent to the mesher so geometry in other parts will generally not influence the mesh of a given part (there are some exceptions to this; for example, contact sizing controls). To create a conformal mesh consisting of multiple different regions or bodies, a multibody part should be used.

The following notes apply to multibody parts:

- Multibody parts need to be formed in DesignModeler or SpaceClaim.
- There are different types of multibody parts. The types of multibody parts prescribe how the interface between the different bodies should be handled. Options include:

- **Shared Topology**: Merges the pair(s) of common faces between bodies into a single set of common faces (2 faces become 1 face). The shared faces belong to both of the neighboring bodies. When meshing these types of multibody parts, the mesh at the interface will be conformal.

- No Shared Topology: Leaves the pair(s) of common faces between bodies separate. The separate faces are meshed separately so the mesh would be non-conformal. Contact detection (p. 25) should pick up these duplicate faces so that you can easily see the non-conformal interfaces.

  → Often it is helpful to imprint the bodies with each other so that there are pairs of common faces with the face boundaries being the same. This could help in creating more similar mesh at the non-conformal interface between the bodies. DesignModeler has **Imprint** as a method for **Shared Topology**. It is similar to **None**, but it will imprint all bodies in the multibody part with each other.

  → In some cases this option is used to help organize bodies into parts, but it is important to note that this has ramifications. A multibody part will have all bodies meshed using only 1 meshing process. If instead the bodies are in separate parts, parallel part by part meshing will be utilized and could significantly reduce the meshing time.

  → **Patch Independent Tetrahedrons** has an option **Match Mesh Where Possible** that will try to make the mesh conformal across bodies even without shared topology.

  Note:

  The **Patch Independent Tetrahedrons** method is being deprecated and will be removed in future releases.

- It is common to use multibody parts in hex meshing. The approach is to slice the model into sweepable bodies and use **Shared Topology** to get conformal mesh between those sweepable bodies.

- Generally multibody parts are formed from all solid bodies or all sheet bodies or all line bodies.

  - Multibody parts of solids, sheets, and line bodies are not allowed.

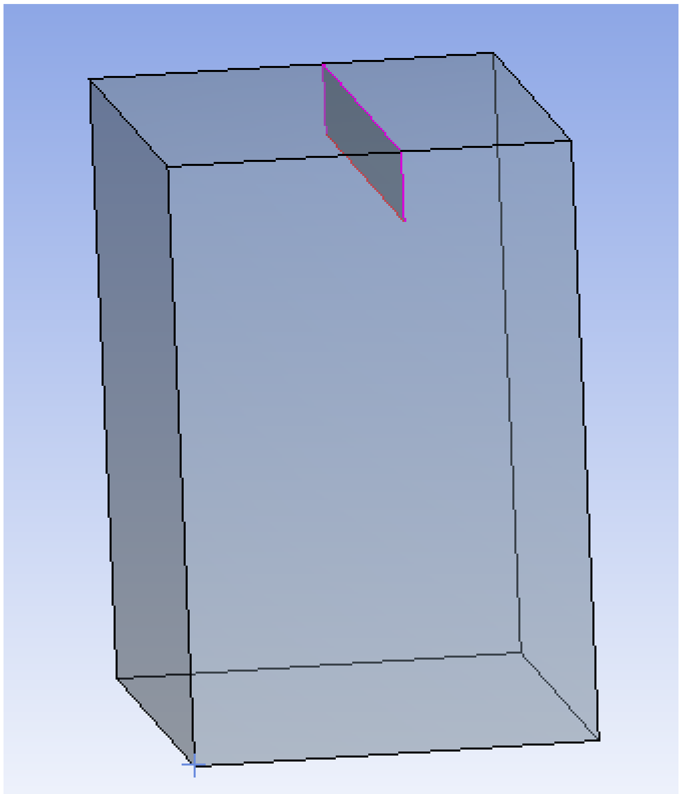
  - Multibody parts of solids and sheets are allowed, but note the following:

    → Solids are meshed first by default. You can use selective meshing (p. 404) to obtain a different behavior.

    → In SpaceClaim, the **Shared Topology** options: **Share** and **Merge**, both create shared topology. Use **Merge** when trying to embed sheets within a solid to construct a zero-thickness or baffle model. Otherwise, use the **Share** option.

    Figure 1: Baffle Model Constructed in SpaceClaim (p. 23) shows a model with one solid body with a baffle face embedded in it. This model was constructed from one solid body and one sheet body, using the **Merge** option in SpaceClaim.

Figure 1: Baffle Model Constructed in SpaceClaim

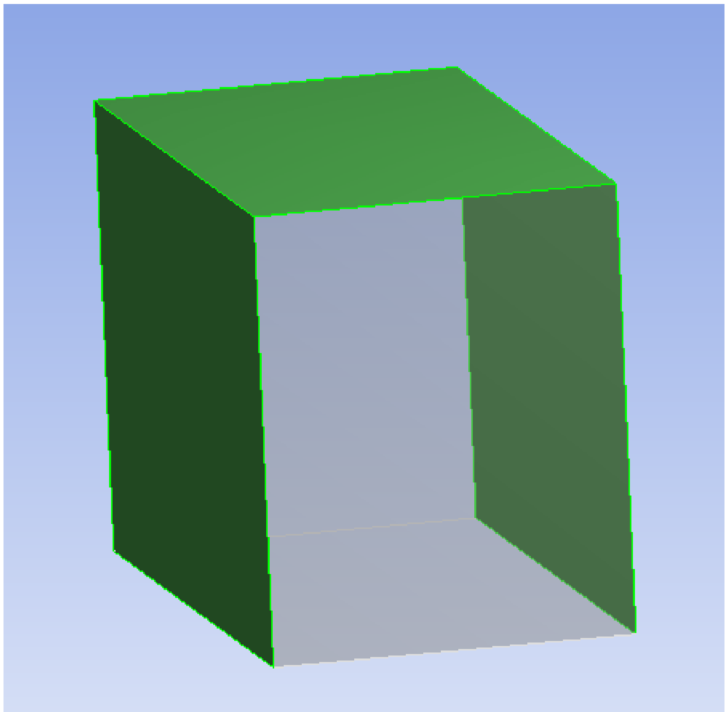


To construct the same model in Ansys DesignModeler, use the **Automatic** option for **Shared Topology**.

→ Solid/skin models can be constructed using the **Share** option in SpaceClaim. Like with other multibody parts, the common faces will become shared.

Solid/skin models where the sheets are coincident to the solid's faces are supported (Figure 2: Solid/Skin Model With Coincident Faces (p. 24)).

Figure 2: Solid/Skin Model With Coincident Faces



- Multibody parts of sheets and beams (line bodies) (p. 423) are supported.

- For related information, refer to Geometry Introduction in the Mechanical help.

Conformal Meshing Across Parts

In some cases, you may want to get conformal mesh without having to use **Shared Topology**.

For example, maybe instances are being used to create repetitive copies of the mesh, and you want to get conformal mesh between the copies. **Shared Topology** could be used within the part that is being copied, but each copy would have to be a unique part. Or, maybe meshing speed is paramount so you want to use parallel part by part meshing and later connect the meshes. There are several mesh-based ways to make the mesh conformal across parts. These include:

- Assembly meshing (p. 367): The assembly meshing approach takes all parts into the mesher at one time. With this approach everything is conformal if the parts are close enough.

- Mesh connections (p. 444): This approach starts from an existing shell mesh and imprints and connects the mesh to make it conformal.

- Node merge (p. 467): After meshing, you can insert a node merge operation to make coincident mesh (nodes close together within a tolerance) conformal. Merge nodes can be used in conjunction with match mesh (p. 280) or contact match mesh (p. 455) controls to make the nodes coincident prior to merging.

Connections/Interface Handling

When coming into the Meshing/Mechanical application, by default, connections are found between parts that have faces in proximity of each other. This automatic detection of connections or interfaces can be disabled if it is not desired. The **Auto Detect Contact On Attach** can be changed by selecting **Tools**> **Options** from the Ansys Workbench main menu, and then selecting either the Mechanical or Meshing category as appropriate. The option is enabled by default in both applications.

These contact regions can be used for mesh sizing (p. 263), and/or are used to define the behavior on how the contact/interface is treated when running the simulation.

For structural solvers see the description of connections in the Ansys Mechanical help.

For CFD solvers, contact regions are used differently by the Ansys CFX and Ansys Fluent solvers:

- Contact regions are used in Ansys CFX as General Grid Interface (GGI) definitions. For details, refer to the documentation available under the Help menu within CFX.

- Contact regions are used in Ansys Fluent as mesh interfaces. See Special Cases (p. 54) for details.

Meshing: Usage in Workbench

The Ansys Meshing application is a separate Ansys Workbench application. The Meshing application is data-integrated with Ansys Workbench, meaning that although the interface remains separate, the data from the application communicates with the native Ansys Workbench data. The following topics are addressed in this section:

Basic Meshing Application Workflows

The following sections describe several basic workflows for using the Meshing application in Ansys Workbench:

Overview of the Meshing Process in Ansys Workbench

The following steps provide the basic workflow for using the Meshing application as part of an Ansys Workbench analysis (non-Fluid Flow). Refer to the Ansys Workbench help for detailed information about working in Ansys Workbench.

1. Select the appropriate template in the Toolbox, such as *Static Structural*. Double-click the template in the Toolbox, or drag it onto the Project Schematic.

2. If necessary, define appropriate engineering data for your analysis. Right-click the Engineering Data cell, and select **Edit**, or double-click the Engineering Data cell. The Engineering Data workspace appears, where you can add or edit material data as necessary.

3. Attach geometry to your system or build new geometry. Right-click the Geometry cell and select **Import Geometry...** to attach an existing model or select **New SpaceClaim Geometry...** or **New DesignModeler Geometry...** to launch the Ansys Discovery SpaceClaim or Ansys DesignModeler application, respectively.

4. Access the Meshing application functionality. Right-click the Model cell and choose **Edit**. This step will launch the Ansys Mechanical application.

5. Once you are in the Mechanical application, you can move between its components by highlighting the corresponding object in the Tree as needed. Select the **Mesh** object in the Tree to access Meshing application functionality and apply mesh controls.

6. Define loads and boundary conditions. Right-click the Setup cell and select **Edit**. The appropriate application for your selected analysis type will open (such as the Mechanical application). Set up your analysis using that application's tools and features.

7. You can solve your analysis by issuing an Update, either from the data-integrated application you're using to set up your analysis, or from the Ansys Workbench GUI.

8. Review your analysis results.

Note:

You should save your data periodically (**File> Save Project**). The data will be saved as a `.wbpj` file. Refer to the Ansys Workbench help for more information about project file management in Workbench.

For more information:

- For information on identifying and correcting mesh failures, refer to Meshing: Troubleshooting (p. 535).

- For information about using the Ansys Meshing application to import or export mesh files, refer to Working with Legacy Mesh Data (p. 40) and Exporting Meshes or Faceted Geometry (p. 42).

- Fluids users of the Ansys DesignModeler, Ansys Meshing, and Ansys CFX applications should refer to Named Selections and Regions for Ansys CFX (p. 77) for important information about region definitions.

- Fluids users of the Ansys DesignModeler, Ansys Meshing, and Ansys Fluent applications should refer to Fluent Mesh Export (p. 43) for important information about Named Selection support.

Overview of the Meshing Process for CFD/Fluids Analyses

This section describes the basic process for using the Ansys Meshing application to create a mesh as part of an Ansys Workbench CFD/fluids analysis. Refer to Strategies for CFD/Fluids Meshing in Ansys Workbench (p. 33) for information about different CFD/Fluids meshing strategies. Refer to the Ansys Workbench help for detailed information about working in Ansys Workbench. There are four basic steps to creating a mesh:

Create Geometry

You can create geometry for the Meshing application in the Ansys Discovery SpaceClaim or Ansys DesignModeler application. You can also import the geometry from an external CAD file.

The Meshing application requires you to construct solid bodies (not surface bodies) to define the region for the 3D mesh (for 2D simulations a sheet body can be used). A separate body must be created for each region of interest in the fluids simulation. For example, a region in which you want the fluids solver to solve only heat transfer must be created as a separate body. Multiple bodies are created in the DesignModeler application by using the **Freeze** command, see **Freeze** in the DesignModeler help for details.

It is best practice to explicitly identify any fluid regions in the model as fluids rather than solids.

For new users or new models it is often useful to first generate a default mesh, evaluate it, and then apply the controls described in Define Mesh Attributes (p. 29) as appropriate to improve various mesh characteristics.

Define Named Selections

During the fluids simulation setup, you will need to define boundary conditions where you can apply specific physics. For example, you may need to define where the fluid enters the geometry or where it exits. Although it may be possible to select the faces that correspond to a particular boundary condition inside the solver application, it is rather easier to make this selection ahead of time in either the CAD connection, the Ansys DesignModeler application, or the Meshing application. In addition, it is much better to define the location of periodic boundaries before the mesh is generated to allow the nodes of the surface mesh to match on the two sides of the periodic boundary, which in turn allows for a more accurate fluids solution.

Creating a Named Selection will affect how the mesher treats that topology. For details, see Protecting Topology Defined Prior to Meshing (p. 180).

You can define the locations of boundaries by defining Named Selections, which can assist you in the following ways:

- You can use Named Selections to easily hide the outside boundary in an external flow problem.
- You can assign Named Selections to all faces in a model except walls, and **Program Controlled** (p. 148) inflation will automatically select all walls in the model to be inflation boundaries.

For more information:

- Fluids users of the Ansys DesignModeler, Ansys Meshing, and Ansys CFX applications should refer to Named Selections and Regions for Ansys CFX (p. 77).
- Fluids users of the Ansys DesignModeler, Ansys Meshing, and Ansys Fluent applications should refer to Fluent Mesh Export (p. 43).

Define Mesh Attributes

The mesh generation process in the Meshing application is fully automatic. However, you have considerable control over how the mesh elements are distributed. To ensure that you get the best fluids solution possible with your available computing resources, you can dictate the background element size, type of mesh to generate, and where and how the mesh should be refined. In general, setting up the length scale field for your mesh is a three-step process, as outlined below:

- Assign a suitable set of global mesh controls (p. 93).
- Override the default mesh type by inserting a different mesh method (p. 196).
- Override the global sizing or other controls locally on bodies, faces, edges, or vertices and the regions close to them by scoping local mesh controls (p. 195).

Generate Mesh

When you are ready to compute the mesh, you can do so by using either the **Update** feature or the **Generate Mesh** feature. Either feature computes the entire mesh. The surface mesh and the volume mesh are generated at one time. The mesh for all parts/bodies is also generated at one time. For help in understanding the difference between the **Update** and **Generate Mesh** features, see Updating the Mesh Cell State (p. 485).

For information on how to generate the mesh for selected parts/bodies only, refer to Generating Mesh (p. 486). The Previewing Surface Mesh (p. 489) and Previewing Inflation (p. 492) features are also available if you do not want to generate the entire mesh at one time.

Once the mesh is generated, you can view it by selecting the **Mesh** object in the Tree Outline. You can define Section Planes to visualize the mesh characteristics, and you can use the Mesh Metric (p. 123) feature to view the worst quality element based on the quality criterion for a selected mesh metric.

Note:

Fluids users should refer to Generation of Contact Elements (p. 441) for recommendations for defining contact for fluids analyses.

Overview of the Meshing Process for Hydrodynamics Analysis

This section describes the basic process for using the Ansys Meshing application to create a mesh as part of a hydrodynamics analysis. Refer to the *Aqwa User's Manual* for more information about hydrodynamics analysis. There are three basic steps to creating a mesh:

Create Geometry

You can create geometry for the Meshing application in the Ansys Discovery SpaceClaim or Ansys DesignModeler applications, or import a geometry from an external file. The model should consist of surface and line bodies only (no solid bodies). See General Modeling Requirements in the *Aqwa User's Manual* for full details of what is required for creating geometry for a hydrodynamics analysis.

Select Physics Preference

Before performing meshing, select **Hydrodynamics** as your Physics Preference (p. 93).

Note:

Hydrodynamics can be set as the default preference by selecting **Tools** > **Options**, clicking the **Meshing** heading, and changing the **Default Physics Preference** to **Hydrodynamics**. For more information, see Meshing: Options (p. 317).

Once defined, you can define the following mesh controls:

- Element Size (p. 98)
- Defeature Size (p. 106)

Define Mesh Attributes

The mesh generation process in the Meshing application is fully automatic. The Hydrodynamics preference corresponds to the uniform sizing with quad dominant method. However, you have considerable control over how the mesh elements are distributed. To ensure that you get the best solution possible with your available computing resources, you can dictate the background element size, type of mesh to generate, and where and how the mesh should be refined. In general, setting up your mesh is a two-step process, as outlined below:

- Assign suitable global mesh sizes within the Hydrodynamics Physics Preference
- Override the global sizing locally on bodies, faces, edges, or vertices and the regions close to them by scoping local mesh controls (p. 195)

Generate Mesh

When you are ready to compute the mesh, you can do so by using either the **Update** feature or the **Generate Mesh** feature. The mesh for all parts/bodies is also generated at one time. For help in understanding the difference between the **Update** and **Generate Mesh** features, see Updating the Mesh Cell State (p. 485).

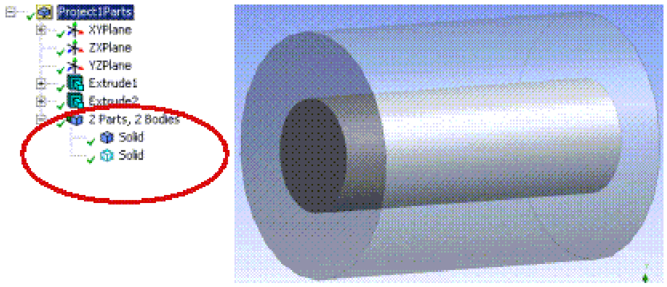
Once the mesh is generated, you can view it by selecting the **Mesh** object in the Tree Outline. You can use the Mesh Metric (p. 123) feature to view the worst quality element based on the quality criterion for a selected mesh metric.

Combining CFD/Fluids Meshing and Structural Meshing

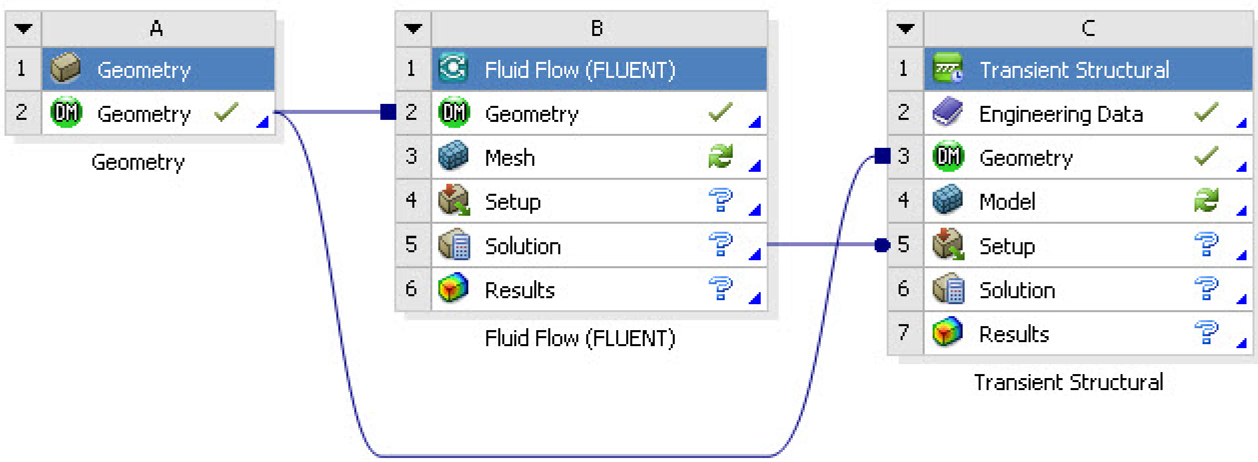
In some applications, a CFD/fluids mesh *and* a structural mesh are required within the same workflow. For these one-way coupling applications, the loading, solving, and postprocessing of the fluids meshed part(s) later occur in a Fluid Flow analysis system, while the loading, solving, and postprocessing of the structurally meshed part(s) later occur in a Structural analysis system. The best approach for this kind of application is to break the model into separate parts rather than use a continuous multibody part.

The following approach is recommended for these applications:

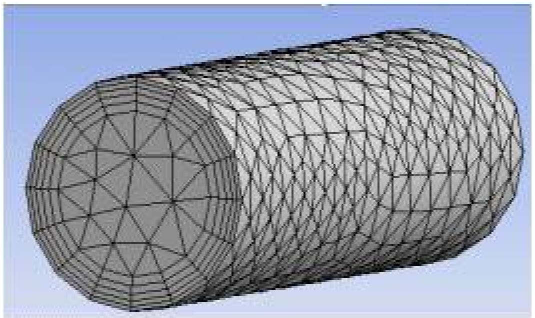
1. Attach the model to a Geometry (DesignModeler) system and use the Explode Part feature to create independent parts within the model.

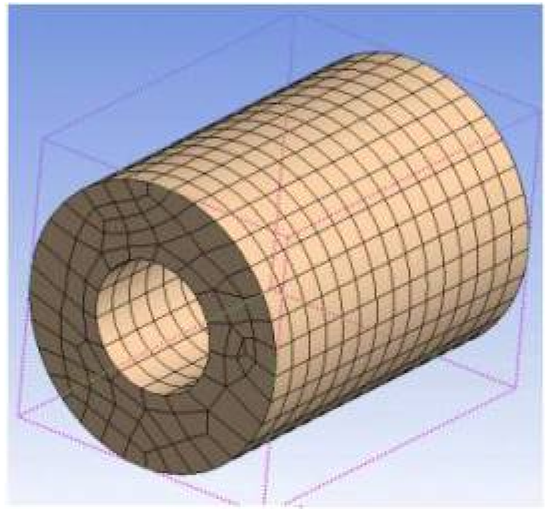


2. Link a Fluid Flow analysis system and a Structural analysis system to the Geometry system. The geometries may be shared or not, depending on whether defeaturing needs to be done to one or the other system. Dedicate the Fluid Flow analysis system to meshing the appropriate fluid domain for the fluids application. Suppress the structural part(s) in the model. Dedicate the Structural analysis system to meshing the appropriate structural part(s). Suppress the fluid part(s) in this model.



In this case, only the respective parts are meshed. The mesh of the Fluid Flow analysis system is shown below on the left, and the mesh of the Structural analysis system is shown on the right.





Note:

- You can set up the workflow schematic in different ways depending on various factors, including variations in the fluid/structural model, persistence, the desired multiphysics simulation, and so on.
- The coupling of the solvers is also handled from the Project Schematic. For details, refer to the discussion about creating and linking a second system in the Ansys Workbench help.
- For geometry persistence, both models will require updating when changing CAD parameters.

Strategies for CFD/Fluids Meshing in Ansys Workbench

Ansys Workbench offers various strategies for CFD/Fluids meshing. For each strategy, certain defaults are in place to target the particular needs of an analysis. The strategies and circumstances in which each of them are appropriate are described below. In all cases, your first decision is to determine whether you want to use assembly meshing (p. 20) or part-based meshing.

Tetra Dominant Meshing - Patch Conforming Tetra/Prism Meshing

The first strategy is to use conformal tetra/prism meshing plus the default Sweep (p. 223) method. This strategy is recommended for models involving moderately clean CAD (for example, native CAD, Parasolid, ACIS, and so on) for which you desire a tetra/hybrid dominant mesh.

Although the Patch Conforming Tetra (p. 200) mesh method is fully automated, it interacts with additional mesh controls and capabilities as necessary, including:

- Advanced tetra and inflation layer technology
- Pinch (p. 182) controls for removing small features at the mesh level (offered as an alternative to Virtual Topologies (p. 501), which work at the geometry level)

- Sizing (p. 100) controls for providing greater control over mesh distribution
- Conformal swept regions (p. 419)
- Body of influence (p. 257) (BOI) for setting one body as a size source for another body

Tetra Dominant Meshing - Patch Independent Tetra/Prism Meshing

An alternative for those desiring a tetra dominant mesh is Patch Independent Tetra (p. 200)/Prism meshing. This approach is best for "dirty CAD"—CAD models with many surface patches (for example, IGES, CATIA V4, and so on) and in cases with large numbers of slivers/small edges/sharp corners. It includes support for post inflation (p. 158), as well as CAD simplification built-in to the tetra mesher.

Note:

The **Patch Independent Tetrahedrons** method is being deprecated and will be removed in future releases.

Mapped Hex Meshing - All Hex Swept Meshing

This mapped hex approach (which includes both general Sweep (p. 323) and thin Sweep (p. 330)) is recommended for clean CAD. It supports single source to single target volumes, and may require you to perform manual geometry decomposition.

Benefits of this approach include:

- Support for Sizing (p. 100) controls
- Compatibility with Patch Conforming Tetra (p. 200) meshing
- Support for swept inflation (p. 414)

Mapped and Free Meshing - MultiZone Meshing

Best for moderately clean CAD, the MultiZone (p. 228) strategy for meshing provides multi-level sweep with automatic decomposition of geometry into mapped (structured) and free (unstructured) regions. When defining the MultiZone mesh method, you can specify a **Mapped Mesh Type** and a **Free Mesh Type** that will be used to fill structured and unstructured regions respectively. Depending on your settings and specific model, the mesh may contain a mixture of hex/prism/tetra elements.

The MultiZone mesh method and the Sweep (p. 223) mesh method described above operate similarly; however, MultiZone has capabilities that make it more suitable for a class of problems for which the Sweep method would not work without extensive geometry decomposition.

Additional benefits of this approach include:

- Support for 3D inflation
- Ability to selectively ignore small features

Assembly Meshing - CutCell and Tetrahedrons Algorithms

CutCell Cartesian meshing is a general purpose assembly meshing (p. 367) algorithm designed for Ansys Fluent.

CutCell is in principal a top-down mesher, and it is mostly suitable for moderately clean CAD models. It results in a mesh of 80-95% hex cells, which often leads to very accurate solutions, providing the physics can handle the relatively rapid size changes due to hanging-node configurations.

Additional benefits of this approach include:

- Support for Sizing (p. 100) controls
- Support for 3D inflation, although very thick inflation should be avoided when using the CutCell algorithm
- Ability to selectively ignore features

The Tetrahedrons assembly meshing algorithm is a derivative of the CutCell algorithm, with strengths and weaknesses similar to those of CutCell. The Tetrahedrons method starts from the CutCell mesh and through various mesh manipulations creates a high quality unstructured tet mesh.

Note:

Assembly Meshing is being deprecated and will be removed in future releases.

Accessing Meshing Functionality

You can access Meshing application functionality from the Model/Mesh cell in an analysis system, or from the Mesh cell in a Mesh component system. Before using the steps provided in this section, you should be familiar with the concepts of analysis systems and component systems in Ansys Workbench.

Accessing Meshing Functionality from an Analysis System

The Model cell (Mesh cell in Fluid Flow analysis systems) allows you to access a meshing application or share a mesh with another system. Model corresponds to the contents of the Model branch within the Ansys Mechanical application and allows you to perform physics-based meshing capabilities, such as spot welds, contact, etc. Mesh contains just node coordinates and mesh connectivity.

To launch the Meshing application from a Model cell in an analysis system (non-Fluid Flow):

1. From the Analysis Systems group of the Ansys Workbench Toolbox, either double-click or drag an analysis system onto the Project Schematic. As a result, a template for that type of analysis system appears in the Project Schematic.
2. In the analysis system, right-click the Geometry cell and choose **New SpaceClaim Geometry...** or **New DesignModeler Geometry...** to create geometry within the Ansys Discovery SpaceClaim or Ansys DesignModeler application, respectively; or choose **Import Geometry** to attach existing geometry.

3. Right-click the Model cell and choose **Edit**. This step will launch the Mechanical application. From the Mechanical application, you can access the Meshing application controls by clicking on the **Mesh** object in the Tree Outline.

To access meshing from a Mesh cell in a Fluid Flow analysis system:

1. From the Analysis Systems group of the Ansys Workbench Toolbox, either double-click or drag a Fluid Flow analysis system onto the Project Schematic. As a result, a template for that type of analysis system appears in the Project Schematic.

2. In the analysis system, right-click the Geometry cell and choose **New SpaceClaim Geometry...** or **New DesignModeler Geometry...** to create geometry within the Ansys Discovery SpaceClaim or Ansys DesignModeler application, respectively; or choose **Import Geometry** to attach existing geometry.

3. Right-click the Mesh cell and choose **Edit**. This step will launch the appropriate mesh application (for example, the Meshing application, etc.).

Accessing Meshing Functionality from a Mesh Component System

To launch the Meshing application from a Mesh component system:

1. From the Component Systems group of the Ansys Workbench Toolbox, either double-click or drag a Mesh component system onto the Project Schematic. As a result, a template of a Mesh system appears in the Project Schematic.

2. In the Mesh system, right-click the Geometry cell and choose **New SpaceClaim Geometry...** or **New DesignModeler Geometry...** to create geometry within the Ansys Discovery SpaceClaim or Ansys DesignModeler application, respectively; or choose **Import Geometry** to attach existing geometry.

3. Right-click the Mesh cell and choose **Edit**. This step will launch the appropriate mesh application (for example, the Meshing application, etc.).

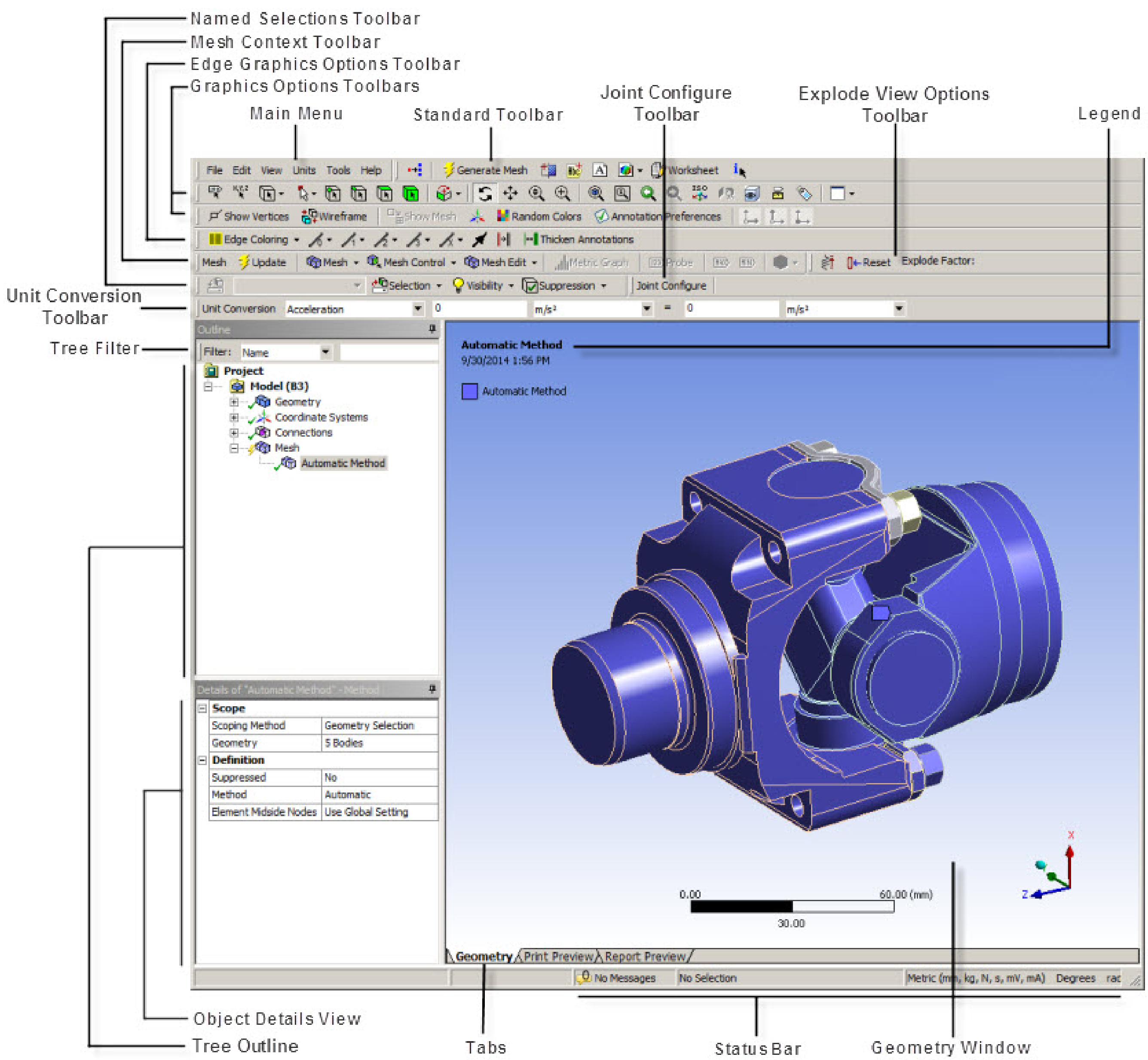
Replacing a Mesh System with a Mechanical Model System

You can replace a Mesh component system with a Mechanical Model component system. This system can then be shared with any analysis system. For details, refer to the description of Mechanical Model systems in the Ansys Workbench help.

Overview of the Meshing Application Interface

The intuitive Meshing application interface, which is shown in Figure 3: Meshing Application Interface (p. 37), facilitates your use of all meshing controls and settings (p. 89).

Figure 3: Meshing Application Interface



The functional elements of the interface are described in the following table.

Note:

The links in the table redirect you to topics in the Ansys Mechanical help that contain supplemental information. Not all of the items described in the Mechanical help are available in the Meshing application interface.

Window Component	Description
Main Menu	This menu includes the basic menus such as **File** and **Edit**.
Standard Toolbar	This toolbar contains commonly used application commands.
Graphics Toolbar	This toolbar contains commands that control pointer mode or cause an action in the graphics browser.
Context Options Toolbar	This toolbar contains task-specific commands that change depending on where you are in the Tree Outline.
Unit Conversion Toolbar	Not visible by default. This toolbar allows you to convert units for various properties.
Named Selections Toolbar	Not visible by default. This toolbar contains options to manage Named Selections similar to how they are managed in the Mechanical application.
Edge Graphics Options Toolbar	This toolbar provides quick access to features that are intended to improve your ability to distinguish edge and mesh connectivity in a surface body model.
Explode View Options Toolbar	This toolbar is a graphical display feature used to create imaginary distance between geometry bodies (only) of your model for viewing purposes.
Tree Outline	Outline view of the project. Always visible. Location in the outline sets the context for other controls. Provides access to object's context menus. Allows renaming of objects. Establishes what details display in the Details View.
Details View	The Details View corresponds to the Outline selection. Displays a details window on the lower left panel which contains details about each object in the Outline.
Geometry Window (also sometimes called the Graphics window)	Displays and manipulates the visual representation of the object selected in the Outline. This window may display: • 3D Geometry • 2D/3D Graph • Spreadsheet • HTML Pages • Scale ruler

<table>
<tr><th>Window Component</th><th>Description</th></tr>
<tr><td></td><td>
Triad control
Legend
Note:
Hover your mouse over the time stamp in the legend and RMB click to toggle the time stamp off and on.
The Geometry window may include splitter bars for dividing views.
See Ansys.com > Support > Platform Support for a complete list of 3Dconnexion products certified with the current release of Ansys applications.
</td></tr>
<tr><td>Reference Help</td><td>Opens an objects reference help page for the highlighted object. These are the same pages that are available in the Mechanical application.</td></tr>
<tr><td>Tabs</td><td>The document tabs that are visible on the lower right portion of the window.</td></tr>
<tr><td>Status Bar</td><td>Brief in-context tip. Selection feedback.</td></tr>
<tr><td>Splitter Bar</td><td>Application window has up to 3 splitter bars.</td></tr>
</table>

Determination of Physics, Analysis, and Solver Settings

Most systems in Ansys Workbench are defined by three primary attributes: physics type, analysis type, and solver type. The method you use to launch the Meshing application functionality determines how default physics, analysis, and/or solver settings are defined:

- Mesh systems, which are a type of component system, are unfiltered (physics, analysis, and solver). If you launch the Meshing application from a Mesh component system (p. 35), your preferences will be set to the defaults you previously defined within the Meshing application. See Meshing Overview (p. 19) for more information.
- If you launch the Meshing application from an analysis system (p. 35) (whether it be via the Model cell in a non-Fluid Flow analysis system or the Mesh cell in a Fluid Flow analysis system),

your Physics, Analysis, and Solver settings will be set according to the selected type of analysis system. To change the **Physics Preference**, you must use the Details View of the **Mesh** folder.

Note:

- To view the physics, analysis, and solver types that are defined for an analysis system, right-click the Model cell (non-Fluid Flow analyses) or Mesh cell (Fluid Flow analyses) and select Properties. This step will open the Properties window, where you can view the attributes. For example, for an *Electric* system, the Properties window will show that Physics is Electric, Analysis is Steady-State Electric Conduction, and Solver is Mechanical APDL.
- Mechanical Model systems, which are a type of component system, are unfiltered (physics and solver). For details, refer to the discussion of Mechanical Model systems in the Ansys Workbench help.

For more information:

- For a list of analysis systems available in Ansys Workbench and basic steps for building each type of system, refer to the discussion of analysis systems in the Ansys Workbench help.
- For details about the various types of non-Fluid Flow analyses and how to perform them, refer to the discussion of analysis types in the Ansys Mechanical help.
- For details about Fluid Flow analyses and how to perform them, refer to the documentation available under the Help menu within Ansys CFX or Ansys Fluent.

Working with Legacy Mesh Data

You can import legacy mesh files using the following methods. The method that is best for you depends on the type of file that you want to import and how you intend to use it:

- Choose **File> Import** from the Ansys Workbench Menu bar or click the **Import** button on the Ansys Workbench Toolbar to read legacy Ansys Workbench mesh data
- Right-click the Mesh cell and choose **Import Mesh File** to import a read-only mesh for downstream use
- Use the **External Model** system to read third-party mesh formats for other Ansys Workbench systems. You can use the **External Model** system for importing the following file formats:
  - NASTRAN Bulk Data (.bdf, .dat, .nas)
  - Abaqus Input (.inp)
  - Fluent Input (.msh, .cas)
  - ICEM CFD Input (.uns)
  - LS-DYNA Input (.k)

Importing Using File> Import or the Import Button

If you choose **File> Import** or click the **Import** button from Ansys Workbench, you can import mesh files that have an extension of .cmdb or .meshdat. Doing so creates a Mesh system in the Ansys Workbench Project Schematic. When the Mesh cell is edited, the mesh will open in the Meshing application where you can edit it.

For more information about reading a simulation/mesh database (.dsdb/.cmdb) from previous Ansys versions, refer to the discussion of importing legacy databases in the Ansys Workbench help.

Importing Read-only Meshes for Downstream Application Use

You can right-click a Mesh cell and choose **Import Mesh File** to browse to a mesh file that you want to import, provided the file is of one of the following types:

- Ansys CFX mesh file with extension .gtm or .cfx
- Ansys ICEM CFD mesh file with extension .cfx, cfx5, or .msh
- Ansys Fluent mesh file with extension .cas, .msh, or .gz
- Ansys Polyflow mesh file with extension .poly, .neu, or .msh

---

Note:

When you use this method, in the strictest sense you are not "importing" the mesh file. That is, you will not be able to edit the file in the Meshing application using this method. Rather, you are making the mesh available for downstream systems.

To be able to edit these types of files in the Meshing application, you must import the mesh into the **External Model** application, and then into another system.

The **Import Mesh File** method is enabled when:

- No geometry is associated with the Geometry cell.
- No generated mesh is associated with the Mesh cell. (Imported meshes do not disable the **Import Mesh File** menu item.)
- No incoming connections are associated with the Geometry cell or Mesh cell.
- No outgoing connections are associated with the Geometry cell.
- No outgoing connections from the Mesh cell are connected to the Mechanical APDL or Ansys Autodyn applications.

When you import the mesh to the Mesh cell:

- The Geometry cell is deleted.
- The title of the cell changes from "Mesh" to "Imported Mesh."
- The state of the Mesh cell is "Up to Date."

- No incoming connections are allowed.
- Outgoing connections can be established with a Mechanical APDL, Ansys Autodyn, Ansys CFX, Ansys Fluent, or Ansys Polyflow system.
- The Geometry cell in the target system is deleted.
- Using the reset command (right-clicking on the Imported Mesh cell and choosing **Reset**) deletes the imported mesh.

Exporting Meshes or Faceted Geometry

A mesh generated by the Meshing application can be exported into the following file formats:

- Meshing File format (*.meshdat), suitable for import into Ansys Workbench (p. 43)
- Ansys Fluent mesh format (*.msh), suitable for import into Ansys Fluent
- Ansys Polyflow format (*.poly), suitable for import into Ansys Polyflow
- CGNS format (*.cgns), suitable for import into a CGNS-compatible application
- Ansys ICEM CFD format (*.prj), suitable for import into Ansys ICEM CFD
- TGrid Faceted Geometry format (*.tgf), suitable for import into Fluent Meshing (formerly TGrid)

To export a mesh:

1. Generate the mesh.
2. If necessary, select an **Export Unit** (p. 96) or **Export Format** (p. 96) from the **Defaults** group in the Details View.
3. Select **File> Export** from the main menu.
4. In the **Save As** dialog box, choose a directory and specify a file name for the file. Then choose a file type from the **Save as type** drop-down menu and click **Save**.

Note:

- You can also use the Meshing application to export faceted geometry for use in Ansys Fluent Meshing. In such cases you can skip step 1 above. A file with the extension .tgf is created, suitable for import into Ansys Fluent Meshing.
- When the same entity is a member of more than one Named Selection, those Named Selections are said to be "overlapping". If you are exporting a mesh into the Ansys Fluent, Ansys Polyflow, CGNS, or Ansys ICEM CFD format (or faceted geometry into the Ansys Fluent Meshing format), and overlapping Named Selections are detected, the export will fail and you must resolve the overlapping Named Selections before proceeding. Any Named Selection whose **Send to Solver** (p. 78) option is set to **No** is skipped during the check for overlapping entities in Named Selections. For this reason, an easy way to avoid over-

lapping Named Selections is to set all values of **Send to Solver** to **No**. For details, see Repairing Geometry in Overlapping Named Selections (p. 79).

For details, refer to:

Mesh Application File Export

When you export a mesh file to Meshing File format (**File> Export** from the Meshing application main menu, then **Save as type Meshing File**), a file with the extension .meshdat is created. The exported file can be imported as a legacy file into Ansys Workbench by either selecting **File >Import** (p. 40) from the Menu bar or clicking the **Import** button on the Toolbar, and then selecting an **Importable Mesh File**. This will create a Mesh System in the Ansys Workbench Project Schematic.

Fluent Mesh Export

When you export a mesh file to Ansys Fluent mesh format (**File> Export** from the Meshing application main menu, then **Save as type Fluent Input Files**), a mesh file with the extension .msh is created. The exported mesh file is suitable for import into another product such as Ansys Fluent, or into Ansys Fluent Meshing outside of Ansys Workbench. For more control over the input file, refer to Meshing Options on the Options Dialog Box (p. 317).

CutCell meshes exported from the Meshing application are always in polyhedral format. When read into Ansys Fluent Meshing, it will skip the polyhedral cells.

If the mesh file you export contains quadratic elements, all midside nodes will be dropped during export; that is, all element types will be exported as linear element types for Ansys Fluent.

An orientation check/correction will be performed for 3D geometry models exported as 2D mesh such that all 2D cells will have the same orientation. You do not need to manually correct the orientation of the geometry face(s).

When the mesh file is exported to Ansys Fluent mesh format, the material properties of the bodies/parts in the model must be translated to proper continuum zone types (p. 46) for use in Ansys Fluent. To provide this information to Ansys Fluent, the following logic is used:

1. If Physics Preference (p. 93) is set to **CFD** and you do not use either of the methods described in steps 2 or 3 below to explicitly assign a body/part to be either solid or fluid, all zones are exported to Ansys Fluent mesh format as *FLUID* zones by default.

   Note:

   An exception to the above involves models that include an enclosure. If you used the Enclosure feature in the Ansys DesignModeler application, the enclosure body will be assigned a continuum zone type of *FLUID* by default.

   ---

2. For models created/edited in the DesignModeler application, a **Fluid/Solid** material property can be assigned to a solid body or a multibody part (if the multibody part contains at least one solid body). This material assignment appears under **Details of Body** in the Details View of the DesignModeler application.

   For multibody parts, you can change the material property for all bodies in one operation in the DesignModeler application by modifying the **Fluid/Solid** property for the multibody part and the modification will propagate to any solid bodies in the part. Similarly, you can use the DesignModeler application to modify the **Fluid/Solid** property for a solid body that belongs to a multibody part, and the **Fluid/Solid** property for the multibody part will be modified accordingly.

   When exported to Fluent Meshing format, a body/part with a material property of **Solid** will be assigned a continuum zone type of *SOLID* and a body/part with a material property of **Fluid** will be assigned a continuum zone type of *FLUID*. This setting in the DesignModeler application overrides the default behavior described in step 1.

   Note:

   Refer to Figure 5: Multibody Part Containing All Fluid Bodies in the DesignModeler Application (p. 56) for an example that illustrates where to set the **Fluid/Solid** material property in the DesignModeler application.

   ---

3. Finally, the **Fluid/Solid** material property setting in the Meshing application is considered. This material assignment appears in the Details View of the Meshing application when a prototype (**Body** object) is selected in the Tree Outline. Similar to the DesignModeler application feature described in step 2, the Meshing application lets you change the **Fluid/Solid** material property for a body.

   When exported to Ansys Fluent mesh format, a body with a material property of **Solid** will be assigned a continuum zone type of *SOLID* and a body with a material property of **Fluid** will be assigned a continuum zone type of *FLUID*. This setting in the Meshing application overrides any assignments that were made based on the default behavior described in step

1 or the **Fluid/Solid** setting described in step 2. Refer to Changing Fluid/Solid Material Property Settings (p. 379) for more information.

Note:

If there are multiple continuum zones or boundary zones of the same type in the DesignModeler application or the Meshing application, each zone name in the exported Ansys Fluent mesh file will contain the necessary prefix and an arbitrary number will be appended to the name to make it unique. Refer to Examples of Fluent Mesh Export: An Alternative to Using a Fluid Flow (Fluent) Analysis System (p. 55) for an example that illustrates multiple zones of the same type.

For related information, refer to:

Classes of Zone Types in Ansys Fluent

Standard Naming Conventions for Naming Named Selections

Zone Type Assignment

Examples of Fluent Mesh Export: An Alternative to Using a Fluid Flow (Fluent) Analysis System

---

Note:

- Contact regions are said to be "overlapping" when the same entity (face or edge) is a member of more than one contact region or when multiple contact regions share the same geometry. Ansys Fluent does not support overlapping contact regions. For part-based meshing only, if you are exporting a mesh into the Ansys Fluent format and overlapping contact regions are detected, the software will attempt to combine the regions. If it is unable to combine them, the export will fail and you must resolve the overlapping contact regions manually before proceeding. You can use the **Check Overlapping Contact Regions** option to identify the problematic contact regions. For details, see Resolving Overlapping Contact Regions (p. 79).

- If you are performing a 2D analysis and intend to export to Ansys Fluent, you should disable the **Auto Detect Contact On Attach** option to avoid problems that may otherwise occur upon export. You can access this option by selecting **Tools> Options** from the Ansys Workbench main menu, and then selecting either the Mechanical or Meshing category as appropriate. The option is enabled by default in both applications.

- Assembly meshing (p. 367) or Fluent mesh export may fail if you are using shared licensing, no licenses are available, and Ansys Fluent is running already. In such cases, the error is due to shared licensing restrictions, but the error message that is issued does not identify licensing as the cause.

- For additional information about importing files into Ansys Fluent or Ansys Fluent Meshing, refer to the documentation available under the Help menu within the respective product.

Classes of Zone Types in Ansys Fluent

Zone type specifications in Ansys Fluent define the physical and operational characteristics of the model at its boundaries and within specific regions of its domain. There are two classes of zone type specifications:

- Boundary zone types (sometimes called "face zones")
- Continuum zone types (sometimes called "cell zones")

Boundary zone type specifications, such as *WALL* or *INLET_VENT*, define the characteristics of the model at its external or internal boundaries. Boundary zones are collections of faces in 3D, and collections of edges in 2D. By default, a boundary zone of type *WALL* is created for the boundary of each body in the geometry during export to Ansys Fluent mesh format. Continuum zone type specifications, such as *FLUID* or *SOLID*, define the characteristics of the model within specified regions of its domain. By default, a continuum zone is created for each body in the geometry during export to Ansys Fluent mesh format. (By default, continuum zone types will be assigned as described in Fluent Mesh Export (p. 43).)

If you do not want the default zone type to be assigned to a boundary zone, you can set the Auto Zone Type Assignment (p. 317) option to **Off**. You can then override default type of *WALL* by defining Named Selections and naming them according to the conventions (p. 48) provided. All faces belonging to the same body in a Named Selection are placed into a single boundary zone.

Note:

For 2D models, you can group sheet surface bodies into a Named Selection, and the underlying faces contained in the sheet surface bodies will be placed into a single continuum zone (as long as the faces themselves are not contained in a Named Selection).

The following sections further describe boundary zone type and continuum zone type specifications and illustrate their purposes in the definition of an example computational model involving simple geometry.

For details on how zones are named and how zone types are assigned during export, refer to Zone Type Assignment (p. 49).

For an example that illustrates the basic workflow for using Ansys Workbench to create a model in the Ansys DesignModeler application, mesh it in the Meshing application, and export the mesh to Ansys Fluent, refer to Examples of Fluent Mesh Export: An Alternative to Using a Fluid Flow (Fluent) Analysis System (p. 55). In the example, Named Selections are defined in the Meshing application and the correct Ansys Fluent zone names/types are assigned in the exported Fluent mesh file based on those definitions.

Comparing and Contrasting Boundary Zone Types and Continuum Zone Types

Boundary zone type specifications define the physical and operational characteristics of the model at those topological entities that represent model boundaries. For example, if an *INFLOW* boundary zone type is assigned to a face entity that is part of three-dimensional model, the model is defined such that material flows into the model domain through the specified face. Likewise, if a *SYMMETRY* boundary zone type is assigned to an edge entity that is part of a two-dimensional model, the model is defined such that flow, temperature, and pressure gradients are identically zero along the specified edge. As a result, physical conditions in the regions immediately adjacent to either side of the edge are identical to each other.

Continuum zone type specifications define the physical characteristics of the model within specified regions of its domain. For example, if a *FLUID* continuum zone type is assigned to a body entity, the model is defined such that equations of momentum, continuity, and species transport apply at mesh nodes or cells that exist within the body. Conversely, if a *SOLID* continuum zone type is assigned to a body entity, only the energy and species transport equations (without convection) apply at the mesh nodes or cells that exist within the body.

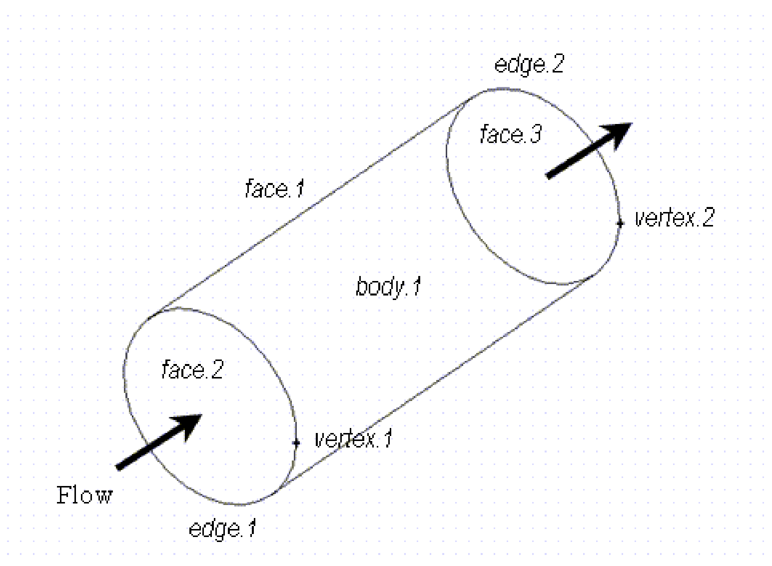
Generally speaking, entities are assigned to zone type classes in Ansys Fluent as shown in the following table:

Model Dimension	Entity	Zone Type Class
3D	Body	Continuum
	Face	Boundary
2D	Face	Continuum
	Edge	Boundary

The Effect of Zone Type Specifications

As an example of the effect of zone type specifications on the specification of a computational model, consider the geometry shown in Figure 4: Boundary Zone Type and Continuum Zone Type Specifications in Ansys Fluent (p. 47), which consists of a single volume in the shape of a straight, elliptical cylinder. The geometry includes one body, three faces, two edges, and two vertices.

Figure 4: Boundary Zone Type and Continuum Zone Type Specifications in Ansys Fluent



The geometry shown in Figure 4: Boundary Zone Type and Continuum Zone Type Specifications in Ansys Fluent (p. 47) can be used to model many different types of transport problems, including fluid flow through a straight, elliptical pipe and heat conduction through a solid, elliptical rod. The following table shows the zone type specifications associated with the fluid flow problem using the geometry shown in Figure 4: Boundary Zone Type and Continuum Zone Type Specifications in Ansys Fluent (p. 47).

Entity	Zone Class	Zone Type
face.1	Boundary	*WALL*
face.2	Boundary	*INFLOW*
face.3	Boundary	*OUTFLOW*
body.1	Continuum	*FLUID*

The following table shows the zone type specifications associated with the heat conduction problem using the geometry shown in Figure 4: Boundary Zone Type and Continuum Zone Type Specifications in Ansys Fluent (p. 47).

Entity	Zone Class	Zone Type
face.1	Boundary	*WALL*
body.1	Continuum	*SOLID*

Note:

For additional information about boundary (face) zones and continuum (cell) zones in Ansys Fluent, refer to the documentation available under the Help menu within Ansys Fluent.

Standard Naming Conventions for Naming Named Selections

This section is applicable to boundary zones only.

If you want to override the default boundary zone type assignments by using Named Selections, naming conventions have been provided for you to follow. Use these conventions when defining Named Selections in either the Ansys DesignModeler application or the Meshing application. By following these naming conventions consistently, you can ensure that Ansys Fluent boundary zone types will be assigned correctly and predictably in the exported Ansys Fluent file.

When naming Named Selections, it is best practice to specify the appropriate name from the list below exactly as shown (case-insensitive). In cases where the name shown below contains an underscore character, a hyphen is also acceptable (for example, both *EXHAUST_FAN* and *EXHAUST-FAN* will work). (Note, however, that although the Meshing application allows both hyphens and underscore characters to be used when defining Named Selections, the DesignModeler application allows only underscore characters.)

The name of each Named Selection is filtered upon export such that only allowable characters remain. Allowable characters include all alphanumeric characters as well as the following special characters:

` ! $ % ^ & * _ + - = : < > . ? /

All other characters, including spaces, are invalid. If an invalid character is used, it is replaced by an underscore (_) upon export.

In addition, the export process does allow for partial matches and special abbreviations, which are described in Zone Type Assignment (p. 49).

AXIS

EXHAUST_FAN

FAN

INLET_VENT

INTAKE_FAN

INTERFACE

INTERIOR

MASS_FLOW_INLET

OUTFLOW

OUTLET_VENT

PERIODIC

POROUS_JUMP

PRESSURE_FAR_FIELD

PRESSURE_INLET

PRESSURE_OUTLET

RADIATOR

RECIRCULATION_INLET

RECIRCULATION_OUTLET

SYMMETRY

THIN

VELOCITY_INLET

WALL

Note:

For details about the boundary (face) zone and continuum (cell) zone types in Ansys Fluent, refer to the documentation available under the Help menu within Ansys Fluent.

Zone Type Assignment

This section describes zone naming and the zone type assignment process.

Zone Naming

By default, zones are named after the part and body from which they are derived. For example, part "part" and body "solid body" will result in a zone name of "part-solid_body." When the zone name is created:

- Any invalid characters (such as the space in "solid body") are replaced by an underscore character ("solid_body").
- Names that begin with a digit are prefixed by "zone."

- If the part name and the body name are identical, only the body name is used to create the zone name. The same rule applies to single body parts.

Note:

If you set the **Auto Zone Type Assignment** option to **Off** (as described in Meshing Options on the Options Dialog Box (p. 317)), all boundary zones are set to the default zone type *WALL*.

If a boundary zone was created for a Named Selection (as described in Classes of Zone Types in Ansys Fluent (p. 46)), the name of the boundary zone is set to the name of the Named Selection.

In cases where the zone naming process could lead to conflicting zone names (for example, in a situation where the potential exists for a zone name that is already in use to be used to name a new zone), one of the following approaches is used:

- If the zone type *is not similar to* the zone name in question, the zone type will be prefixed to the zone name to make it unique. For example, an existing continuum zone named "fluid" and a new boundary zone named "fluid" (with zone type *WALL*) will result in the boundary zone being renamed "wall-fluid."

  A unique integer will be suffixed to the zone name, preceded by a dot character (.). For example, an existing continuum zone named "solid" (with zone type *SOLID*) and another continuum zone named "solid" (with zone type *FLUID*) will result in the continuum zone being renamed "solid.1."

- If the zone type *is similar to* the zone name in question, the approach is dependent on whether the zone is a continuum zone or a boundary zone:
  - If multiple continuum zones have the same type and same zone name, they are merged into a single continuum zone.
  - If multiple boundary zones have the same type and same zone name, a unique integer will be suffixed to the zone name, preceded by a dot character (.). For example, an existing boundary zone named "wall" and a second boundary zone named "wall" will result in the second boundary zone being renamed "wall.1." Subsequent boundary zones named "wall" will be renamed "wall.2," "wall.3," and so on.

Because part and body names influence the creation of continuum zones in Ansys Fluent, it is best practice to assign distinct part and body names that follow these rules:

- Names should include only allowable characters. Allowable characters include all alphanumeric characters as well as the following special characters:

  ` ! $ % ^ & * _ + - = : < > . ? /

  All other characters, including spaces, are invalid. If an invalid character is used, it is replaced by an underscore (_) upon export.

- It is not sufficient for part and body names to differ by case only. This approach may lead to name conflicts for zones in Ansys Fluent, because names are converted to all lowercase characters upon export.

- Using "INTER" as zone name or part of a zone name should be avoided. Because the mesher does not allow named selections with identical names and interface definitions require two named selections, using "INTER" can cause zone type conflicts.

Zone Type Assignment Process

The section is applicable to boundary zones only.

The zone type is derived from the zone name. To assign zone types, the string comparison operations detailed below are performed during the export process. These string comparison operations, which correspond to the naming conventions described in Standard Naming Conventions for Naming Named Selections (p. 48), are applied in the order in which they are listed below (that is, at first an exact match is tested, after that a partial match is tested, etc.) and are always *case-insensitive*. For example, fan, Fan, FAn, and faN are all exact matches for the 'FAN' string comparison and result in a zone type of *FAN* being assigned.

When the search operation begins, it will start by searching the first portion (or sub-string) of the string and if no match is found, it will search for a match anywhere in the string. For example, if a Named Selection with the name wall_inlet_flange is defined, it will be exported as zone type *WALL*. The 'inlet' portion of the name will have no effect on zone type assignment.

In some cases a name may match multiple patterns or rules, and the expected zone type may not be assigned. For this reason it is strongly advisable to use only unambiguous names that exactly match one of the rules presented below.

Once they are exported, names are *all* lowercase. The single quotation marks that are shown enclosing the strings below are not considered during the string comparison operations.

1. Exact matches are checked:

```
'AXIS'
'EXHAUST_FAN'
'FAN'
'INLET_VENT'
'INTAKE_FAN'
'INTERFACE'
'INTERIOR'
'INTERNAL'
'MASS_FLOW_INLET'
'OUTFLOW'
'OUTLET_VENT'
'POROUS_JUMP'
'PRESSURE_FAR_FIELD'
'PRESSURE_INLET'
'PRESSURE_OUTLET'
'RADIATOR'
'RECIRCULATION_INLET'
'RECIRCULATION_OUTLET'
'SYMMETRY'
'THIN'
'VELOCITY_INLET'
'WALL'
```

2. Partial matches are considered only if an exact match was not found in step 1:

```
'AXIS'
{'EXHAUST' && 'FAN'}
'FAN'
{'INLET' && 'VENT'}
{'INTAKE' && 'FAN'}
```

```
'INTERFACE'
'INTERIOR'
'INTERNAL'
{'MASS' && 'FLOW' && 'INLET'}
'OUTFLOW'
{'OUTLET' && 'VENT'}
{'POROUS' && 'JUMP'}
{'PRESSURE' && 'FAR' && 'FIELD'}
{'PRESSURE' && 'INLET'}
{'PRESSURE' && 'OUTLET'}
'RADIATOR'
{'RECIRCULATION' && 'INLET'}
{'RECIRCULATION' && 'OUTLET'}
'SYMMETRY'
{'VELOCITY' && 'INLET'}
```

3. String comparisons to the special abbreviations listed in the table below are performed if no match was found in step 1 or step 2. If an exact match to one of the strings listed in the table is found, the corresponding zone type is assigned:

When a match for this string is found...	This zone type is assigned...
`'CNDBY'`	*INTERFACE*
`'EXFAN'`	*EXHAUST FAN*
`'IFACE'`	*INTERFACE*
`'IN'`	*PRESSURE INLET*
`'INFAN'`	*INTAKE FAN*
`'INTERFACE'`	*INTERFACE*
`'IVENT'`	*INLET VENT*
`'MASFI'`	*MASS FLOW INLET*
`'OUT'`	*PRESSURE OUTLET*
`'OVENT'`	*OUTLET VENT*
`'PFAR'`	*PRESSURE FAR FIELD*
`'PORJ'`	*POROUS JUMP*
`'PRESF'`	*PRESSURE FAR FIELD*
`'PRESI'`	*PRESSURE INLET*
`'PRESO'`	*PRESSURE OUTLET*
`'PRESS'`	*PRESSURE FAR FIELD*
`'RAD'`	*RADIATOR*
`'RINLT'`	*RECIRCULATION INLET*
`'ROUT'`	*RECIRCULATION OUTLET*
`'SLIDE'`	*INTERFACE*

When a match for this string is found...	This zone type is assigned...
'SYM'	*SYMMETRY*
'SYMET'	*SYMMETRY*
'SYMM'	*SYMMETRY*
'VELF'	*VELOCITY INLET*
'VELI'	*VELOCITY INLET*

4. Partial matches are considered if no match was found in steps 1, 2, or 3. If a partial match to one of the strings listed in the following table is found, the corresponding zone type is assigned:

When a match for this string is found...	This zone type is assigned...
'EXHAUST'	*EXHAUST FAN*
'INLET'	*VELOCITY INLET*
'OUTLET'	*PRESSURE OUTLET*
'SYM'	*SYMMETRY*
'THIN'	*THIN*
'WALL'	*WALL*

5. If none of the string comparisons described in steps 1, 2, 3, or 4 result in a match, the boundary zone is assigned zone type *WALL* by default.

Note:

- If you are using **CutCell** and are defining Named Selections for 3D bodies, it is best practice to avoid the use of Ansys Fluent keywords. For example, do not include the string "inlet" in the Named Selection name.
- When Named Selections defined in the Ansys DesignModeler application are required in the Meshing application, you must set the appropriate geometry import options to ensure the Named Selections will be transferred properly. Refer to Importing DesignModeler Named Selections into the Meshing Application (p. 77) for details.
- For assembly (p. 367) meshing algorithms, the names of parts, bodies, and Named Selections should be limited to 64 characters.

Handling Periodic Regions

To export matching face meshes as *PERIODIC* boundary zone types, you must perform the following steps:

1. Define matching face meshes (for 3D) or edge meshes (for 2D only) as **Periodic Regions** or **Cyclic Regions** in a **Symmetry** folder (p. 423) or as **Match Controls** or **Arbitrary Match Controls**.

2. Define two **Named Selections** for each periodic boundary condition, one Named Selection per side, ensuring that "`periodic`" is used in the name (for example, `periodic1A`, `periodic1B`...). The same Named Selections could be used to define the periodic or cyclic region.

Note:

- Periodic regions must belong to the same Fluent mesh cell zone to ensure a successful export.
- When match controls on faces are used, only one periodic or cyclic transformation is supported (for example, you can export multiple match controls as long as they use the same coordinate system and have the same angle/translation).

Special Cases

Be aware of the following special cases related to boundary zone type assignment:

- If Physics Preference (p. 93) is set to **CFD** and no other zone assignment has been explicitly defined, all zones are exported as *FLUID* zones. See Fluent Mesh Export (p. 43) for more information.

- A boundary zone type of *INTERFACE* is assigned automatically to the contact source and contact target entities that compose contact regions at the time of mesh export. For this reason, you are not required to explicitly define *INTERFACE* zones to resolve contact regions. The zone type of *INTERFACE*, along with the contact information, is written to the mesh file.

  When reading the mesh file, Ansys Fluent creates a mesh interface for each contact region automatically. Each interface can consist of any number of *INTERFACE* zones for the source and any number of *INTERFACE* zones for the target.

  For 3D meshes, only face zones referenced in the contact region are considered during export. For 2D meshes, only face zones consisting of edges are considered. Each contact region can contain one or more face zones as sources and one or more face zones as targets. If any entity referenced in a contact region (either as source or target) is also contained in a Named Selection, that contact region is not considered during export.

  For more accurate results in cases in which one face zone in a pair has a higher element count than the other, the zone with the lower element count should be defined as **Interface Zone 1** of the mesh interface and the zone with the higher element count should be defined as **Interface Zone 2**. In the transfer of the contact region source and target regions it is ensured that the source regions are defined as interface zone 1 and the target regions as interface zone 2.

- A boundary zone type of *INTERNAL* is assigned automatically between two *FLUID* zones (sharing a common boundary) at the time of mesh export. For this reason, you are not required to explicitly define an *INTERNAL* zone in such cases if the name of this zone does not contain any special name derived from the part/body or Named Selection name.

- A boundary zone type of *WALL* is assigned automatically to a baffle, unless the baffle is part of a Named Selection that was defined in the Ansys DesignModeler application or the Meshing application, and the name of the Named Selection results in a different zone type assignment.

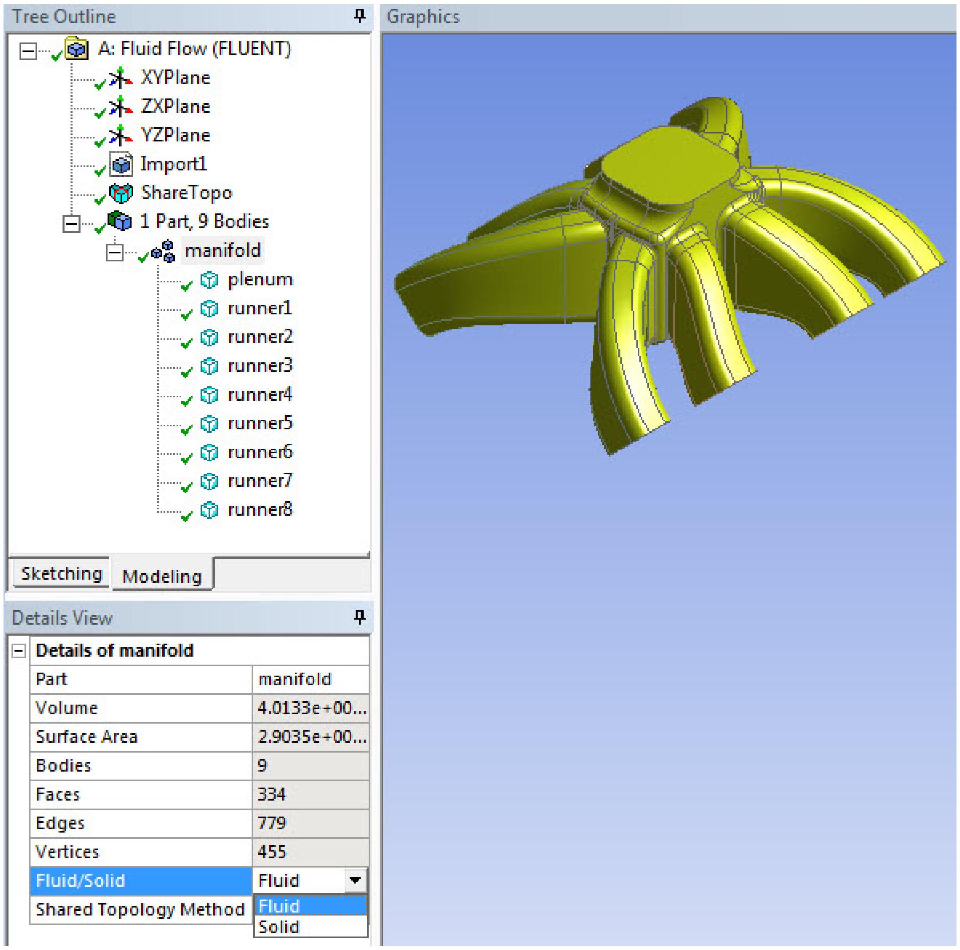
- A boundary zone type of *WALL* is assigned automatically between a *FLUID* zone and a *SOLID* zone at the time of mesh export. For this reason, you are not required to explicitly define a *WALL* zone in such cases. When reading the mesh file, Ansys Fluent will generate an additional *WALL SHADOW* zone automatically.

Examples of Fluent Mesh Export: An Alternative to Using a Fluid Flow (Fluent) Analysis System

The first example illustrates the basic workflow you can follow to create a multibody part in the Ansys DesignModeler application, mesh the model in the Meshing application, and export the mesh to Ansys Fluent. In the example, the bodies are renamed in the DesignModeler application, and Named Selections are defined in the Meshing application. Based on these definitions, Ansys Fluent zone names/types are assigned correctly and predictably (for both continuum and boundary zones) in the exported Fluent mesh file.

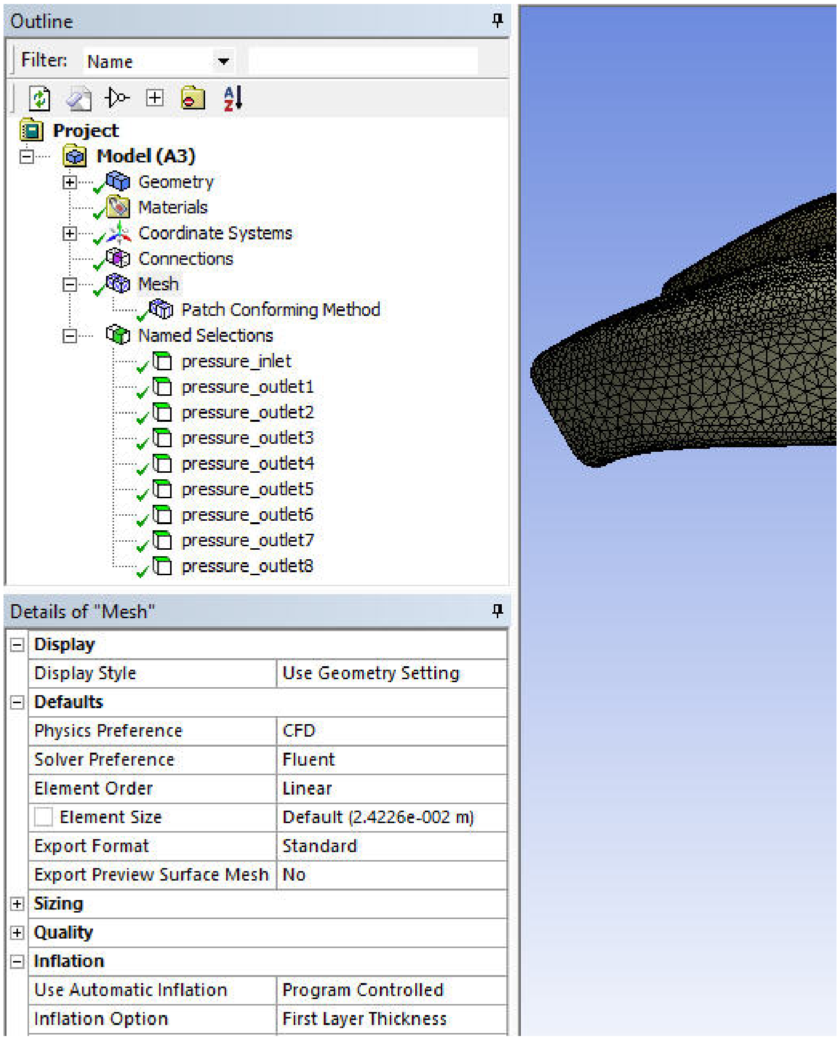
First, the model is imported into the DesignModeler application. The model consists of nine solid bodies after import. In the DesignModeler application, a multibody part is formed, the bodies are renamed, and all bodies are assigned a material property of fluid. (See Fluent Mesh Export (p. 43) for more information about the **Fluid/Solid** material property in the DesignModeler application.) Shared Topology is also used in this example. Refer to Figure 5: Multibody Part Containing All Fluid Bodies in the DesignModeler Application (p. 56).

Figure 5: Multibody Part Containing All Fluid Bodies in the DesignModeler Application



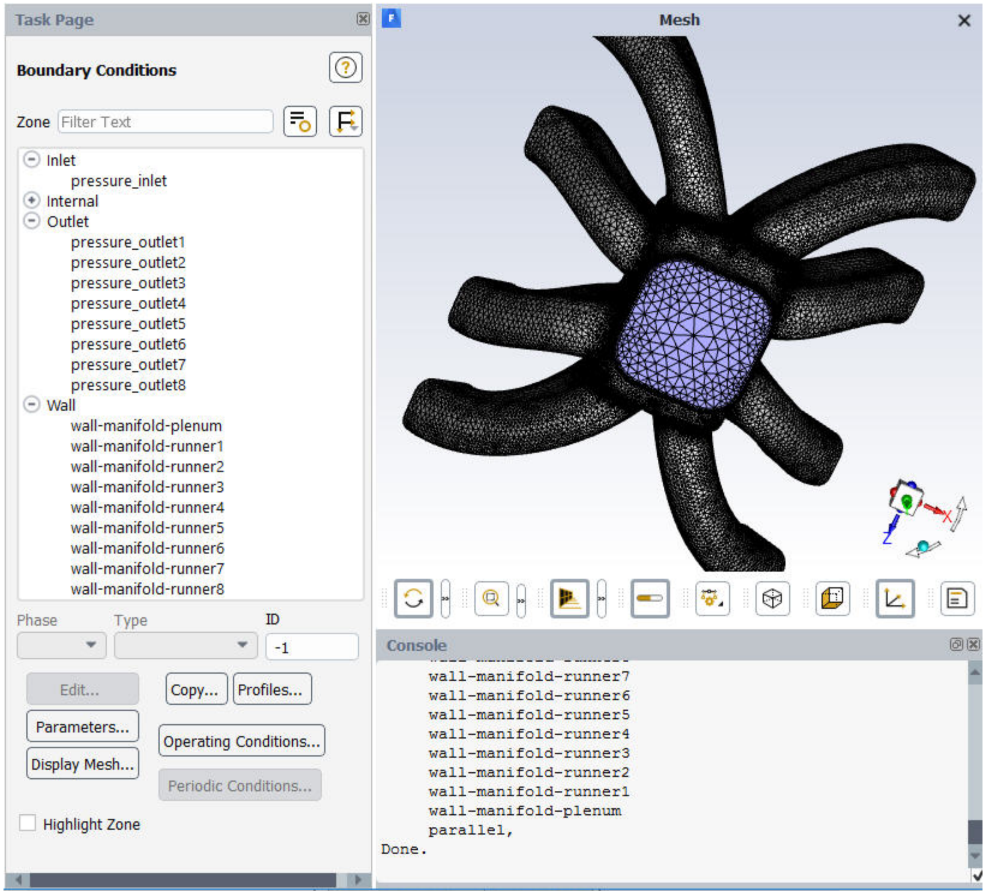
Next, the model is edited in the Meshing application. The patch conforming mesh method is applied with inflation, and Named Selections are defined for boundary zones. Virtual Topology is also used in this example to provide geometry cleanup. Refer to Figure 6: Named Selections Defined in Meshing Application (p. 57).

Figure 6: Named Selections Defined in Meshing Application



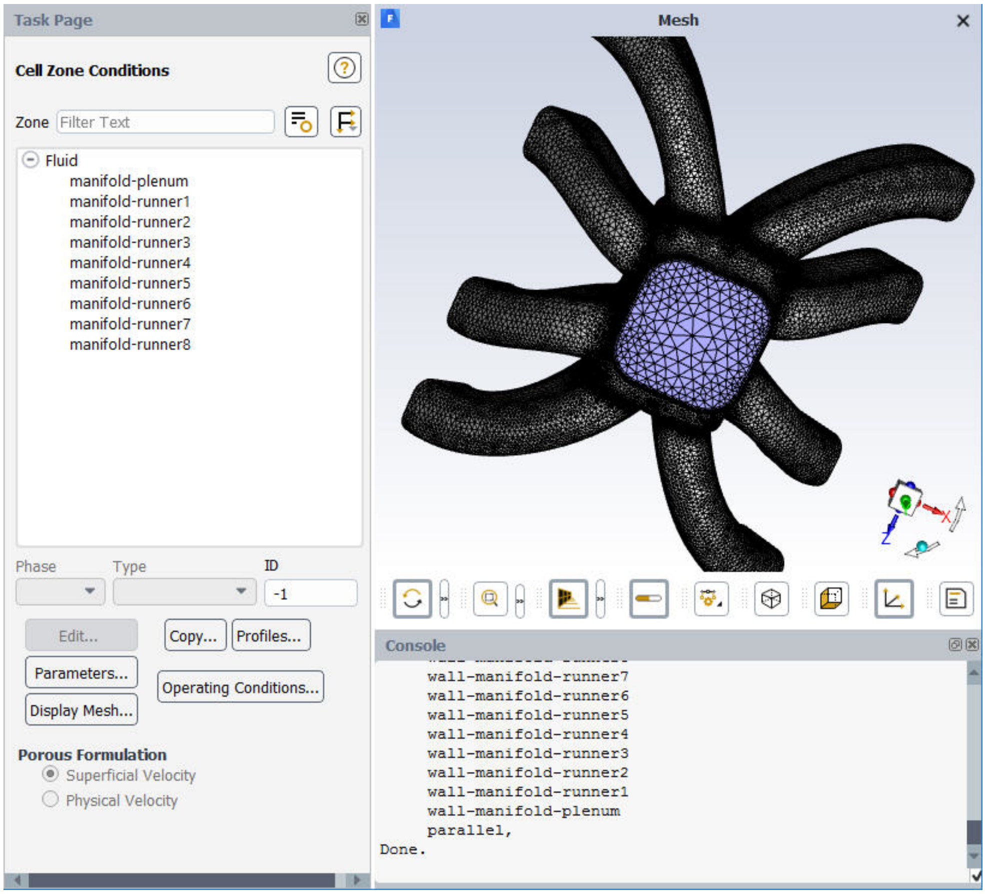
After meshing, the mesh is exported to Ansys Fluent format and read into Ansys Fluent. As shown in Figure 7: Boundary Zone Names and Types Transferred to Ansys Fluent (p. 58), the boundary zone names and types are transferred as expected.

Figure 7: Boundary Zone Names and Types Transferred to Ansys Fluent



Similarly, continuum (or cell) zone names and types (in this case, all fluid) are transferred as expected. Refer to Figure 8: Continuum Zone Names and Types Transferred to Ansys Fluent (p. 59).

Figure 8: Continuum Zone Names and Types Transferred to Ansys Fluent

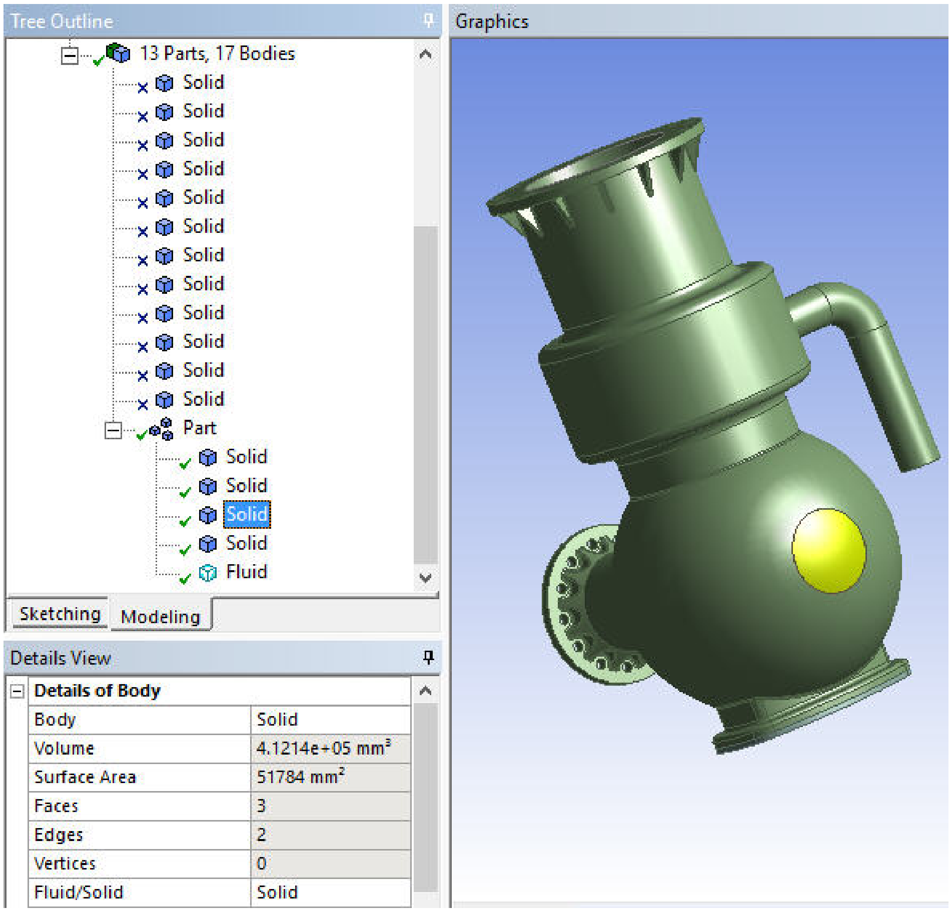


The second example also illustrates a workflow involving the DesignModeler application, the Meshing application, and Ansys Fluent. However, in this example the **Fluid/Solid** material property of a body is changed while the model is being edited in the Meshing application.

First, the model is imported into the DesignModeler application. The model consists of 13 parts and 17 bodies, but 12 of the bodies are not needed for this example and are suppressed. This leaves a multibody part consisting of five bodies. The **Fluid/Solid** material property is set to **Solid** for four of the bodies and to **Fluid** for the remaining body. Notice that one body is selected in the Tree Outline and is highlighted in the **Geometry** window. Refer to Figure 9: Multibody Part Containing Mix of Solid and Fluid Bodies in the DesignModeler Application (p. 60).

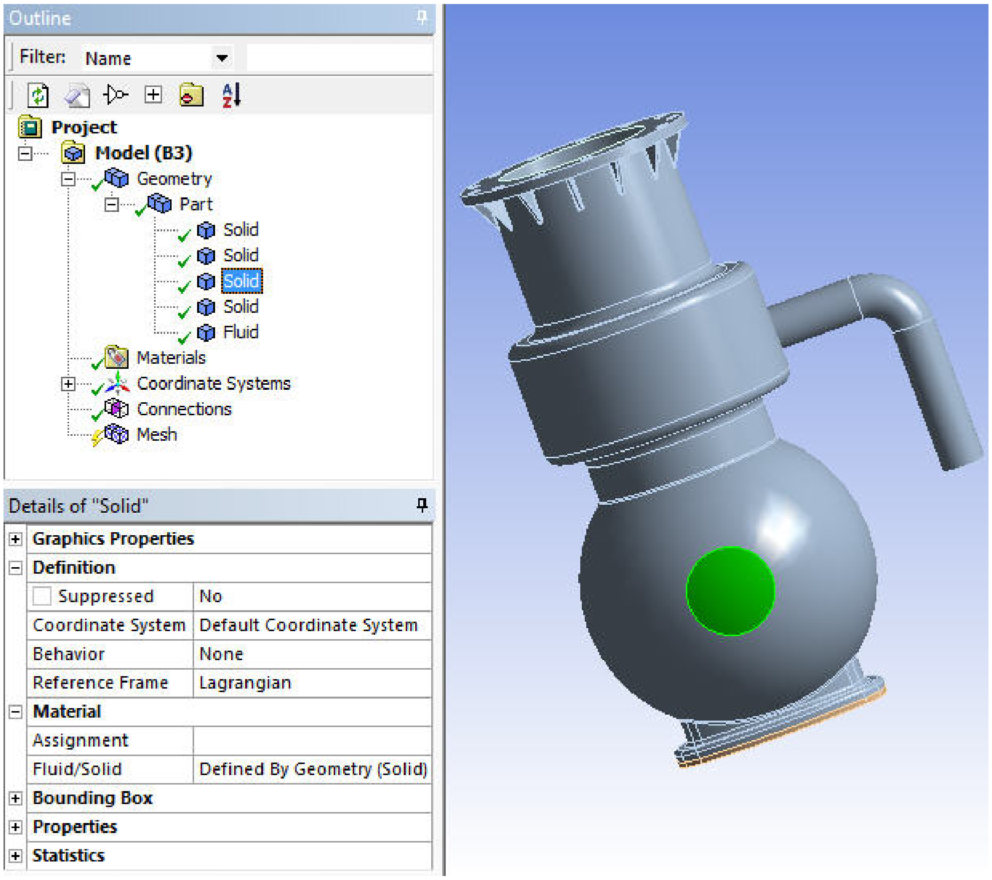
See Fluent Mesh Export (p. 43) for more information about the **Fluid/Solid** material property in the DesignModeler application.

Figure 9: Multibody Part Containing Mix of Solid and Fluid Bodies in the DesignModeler Application



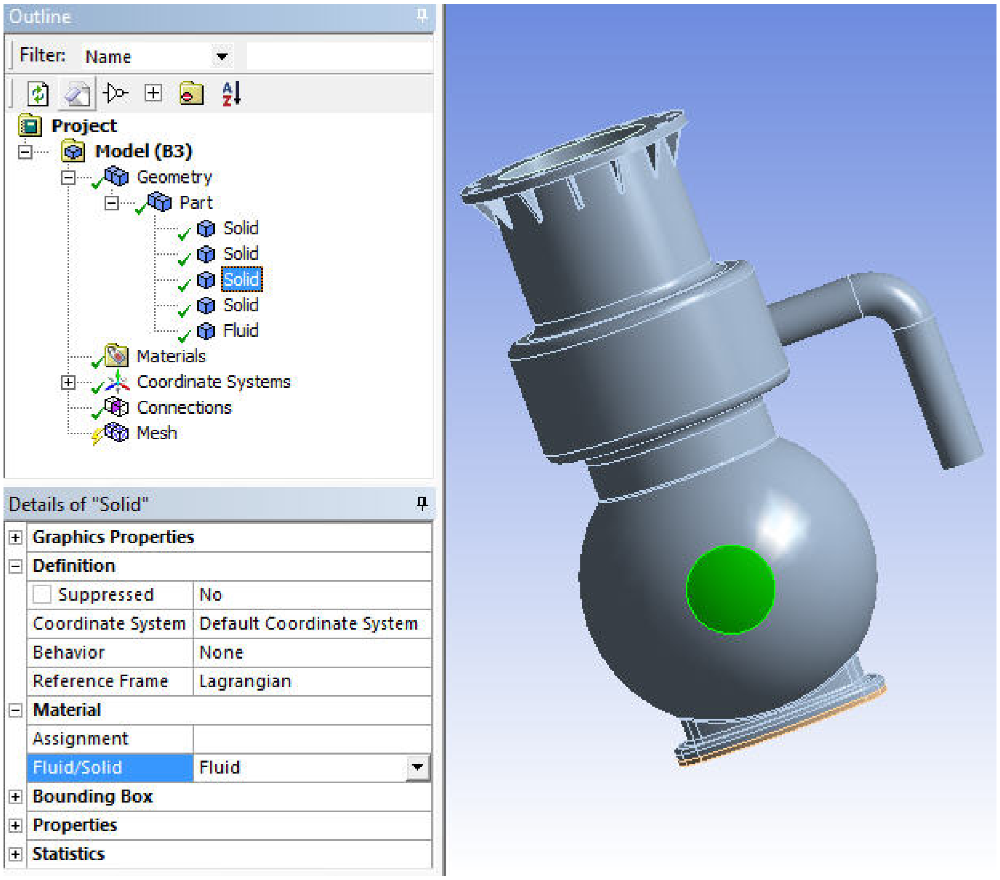
Next, the model is opened in the Meshing application for editing. Notice that the same body has been selected in the Tree Outline and is highlighted in the **Geometry** window. Also notice that the **Fluid/Solid** setting for the body is set to **Defined By Geometry (Solid)** in the Details View. When set to **Defined By Geometry**, the value is based on the **Fluid/Solid** material property that was assigned to the body in the DesignModeler application. Refer to Figure 10: Multibody Part Being Edited in the Meshing Application (p. 61).

Figure 10: Multibody Part Being Edited in the Meshing Application



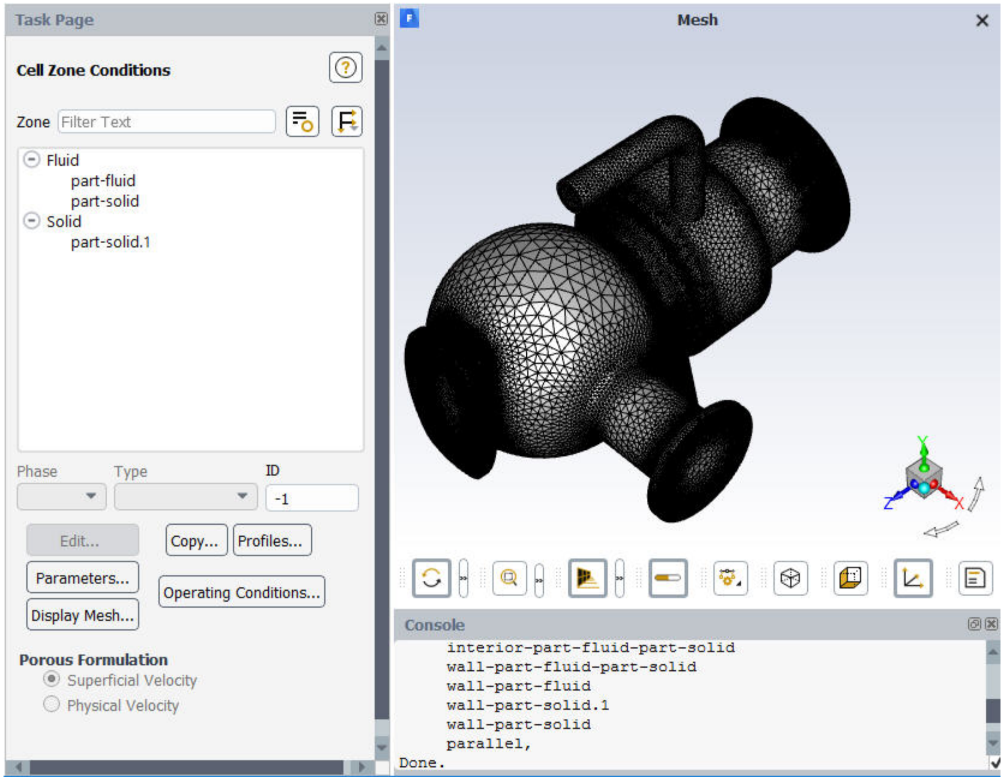
Next, the **Fluid/Solid** material property of the highlighted body is changed to **Fluid**. For all other bodies, the **Fluid/Solid** material property that was assigned in the DesignModeler application will be retained for this example. Refer to Figure 11: Changing the Fluid/Solid Material Property of a Body (p. 62).

Figure 11: Changing the Fluid/Solid Material Property of a Body



After meshing, the mesh is exported to Ansys Fluent format and read into Ansys Fluent. As shown in Figure 12: Continuum Zone Names and Types Transferred to Ansys Fluent (p. 63), the continuum (or cell) zone names and types are transferred as expected. Notice that the zone named "part-solid," which is highlighted in the panel on the left, has a zone type of *FLUID*.

Figure 12: Continuum Zone Names and Types Transferred to Ansys Fluent



Polyflow Export

When you export a mesh file to Polyflow format (**File> Export** from the Meshing application main menu, then **Save as type Polyflow Input Files**), a mesh file with the extension .poly is created. The exported mesh file is suitable for import into Ansys Polyflow and supports the following features:

- Named Selections - Named Selections that are present in the Meshing application will appear in the exported mesh file.
- PMeshes - You can create Named Selections to specify specialized modeling conditions on edges for 2-D or shell geometry; and edges and faces for 3-D geometry. The exported mesh file will contain the mesh nodes and elements associated with those Named Selections in PMesh format. For more information, refer to Generating Meshes in Ansys Meshing for Polyflow.
- Element types - Those that are supported in the exported mesh file are listed in the table below. Only linear meshes are supported for Polyflow export.

Dimension	Supported Element Type
3D	8–node hexahedral
	4–node tetrahedral
	5–node pyramid
	6–node wedge
2D	3–node triangle

Dimension	Supported Element Type
	4–node quadrilateral

Note:

- If you are using the **CutCell** assembly meshing algorithm (p. 367), the exported mesh file is in Fluent Meshing format. For all other mesh methods, including the **Tetrahedrons** assembly meshing algorithm, the exported mesh file is in Patran format. The Fluent Meshing format is used for **CutCell** because the Patran format does not support hanging nodes.

- As an alternative to the export process described here, you can transfer a mesh from a Mesh system into a downstream Polyflow system.

- If you change your mesh settings after generating the mesh and do not perform an **Update** (p. 485), the mesh that is exported is the currently existing mesh (that is, a mesh that does not reflect your new settings). In such cases, a warning message is issued that indicates that the mesh you are exporting to Polyflow format has not been updated. If you want the latest settings to affect the mesh, perform an **Update** and export the mesh again.

CGNS Export

When you export a mesh file to CGNS format (**File> Export** from the Meshing application main menu, then **Save as type CGNS Input Files**), a CGNS mesh file with the extension .cgns is created. The exported mesh file is suitable for import into a CGNS-compatible application. For more control over the input file, refer to Meshing Options on the Options Dialog Box (p. 317).

Named Selections are supported in the CGNS file.

Element types that are supported in the exported CGNS mesh are listed in the table below. Only linear meshes are supported for CGNS export.

Dimension	Supported Element Type
3D	8–node hexahedral
	4–node tetrahedral
	5–node pyramid
	6–node wedge
2D	3–node triangle
	4–node quadrilateral

The maximum number of elements in a part multiplied by the number of nodes per element cannot exceed 2 billion. For linear HEXA meshes, the maximum number of elements in a part (or body) should not exceed 250 million elements. For linear TETRA meshes, the maximum number of elements in a part (or body) should not exceed 500 million elements.

To export larger meshes, split them into several smaller parts (or Named Selections).

If you exceed the CGNS export limits, you will get the error message:

```
Error: CGNS section CGNS_section_name exceeds maximum mesh size limit, please split the mesh into smaller sectio
```

Ansys ICEM CFD Export

When you export from the Meshing application to Ansys ICEM CFD format, an Ansys ICEM CFD project file with the extension .prj, along with a geometry file (*.tin) and/or mesh file (*.uns) are written. The files that are created are the same as those that are created if you begin in Ansys ICEM CFD and use its Import Model option to import a file from the Meshing application.

Anytime you plan to export from the Meshing application to Ansys ICEM CFD format, it is best practice to define the desired part and body names for your model in the DesignModeler or SpaceClaim application prior to meshing the model in the Meshing application. This is recommended because the Ansys ICEM CFD part names will be derived from the part and body names that are defined for the model when you *initially* open the model in the Meshing application; the export process will ignore any renaming or Named Selections created in the Meshing application.

There are three ways to export meshing data to ICEM CFD format:

1. Choose **File> Export** from the Meshing application main menu, then **Save as type ICEM CFD Input Files.**

   Ansys ICEM CFD part names that appear in the exported files are derived from the Ansys Workbench geometry part and body names. In the case of a single body part, only the body name is used.

   > **Note:**
   >
   > The concept of a *part* in Ansys Workbench and a *part* in Ansys ICEM CFD is not the same. For information about parts in Ansys Workbench, refer to Conformal and Non-Conformal Meshing (p. 21) in the Meshing application help and Geometry Introduction in the Mechanical help. For information about parts in Ansys ICEM CFD, refer to the documentation available under the Help menu within Ansys ICEM CFD.

2. Save your Ansys Workbench files (*.mechdat or *.meshdat) and use the Ansys ICEM CFD **File > Import Model** option to import the files into Ansys ICEM CFD (as long as Ansys Workbench and Ansys ICEM CFD are installed on the same machine). Legacy formats such as *.dsdb and *.cmdb are also supported.

   Any defined Named Selections will be imported into Ansys ICEM CFD as subsets because subsets support non-exclusive sets (overlapping Named Selections). However, each entity can only be in a single part (exclusive sets). In cases where you want overlapping Named Selections to be converted to Ansys ICEM CFD parts, the overlapping subsets can be cleaned up in Ansys ICEM CFD and then converted into parts. For details about handling imported Ansys Workbench files in Ansys ICEM CFD, refer to the documentation available under the Help menu within Ansys ICEM CFD.

3. Use the Ansys ICEM CFD Workbench Add-In. In Ansys Workbench, right-click on a **Geometry** or **Mesh** cell and choose **Transfer Data Into New > ICEM CFD**. The advantage of the Add-

In connection is that it maintains the connectivity once the geometry is modified so, unlike the previous two methods, you can easily refresh the geometry in ICEM CFD and then update the mesh and the solver input. For more information, see Component Systems in the [Ansys Workbench User's Guide].

Rules Followed By the Export Process

When exporting to Ansys ICEM CFD format, these rules are followed:

Note:

The series of examples that follows this list illustrates many of the rules listed here.

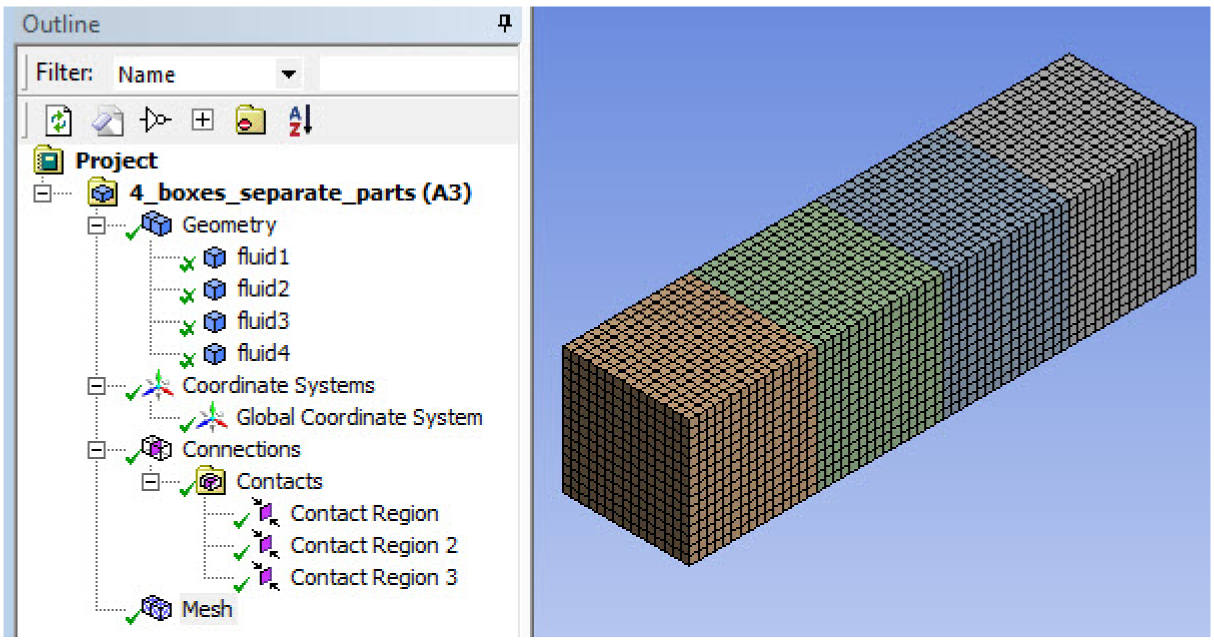
- To achieve unique Ansys ICEM CFD part names in the Ansys ICEM CFD format files, a unique integer is suffixed to all Ansys Workbench part/body names.
- A single body part in Ansys Workbench will appear as *<part_name>_<part_index>* in the Ansys ICEM CFD format files.
- A multibody part in Ansys Workbench will appear as *<part_name>_<part_index>/<body_name>_<body_index>* in the Ansys ICEM CFD format files. The / character denotes hierarchy.
- Bodies that are in a multibody part in Ansys Workbench are put into an Ansys ICEM CFD assembly. The structuring in the Ansys ICEM CFD format files reflects the part/body structure present in Ansys Workbench.
- As long as they are *not* contained in Named Selections, faces that are shared between bodies in the same multibody part in Ansys Workbench are put into separate Ansys ICEM CFD parts. This type of shared face is named according to the bodies having the face in common, with the body names separated by the # character.
- Entities that are contained in a Named Selection are put into a separate Ansys ICEM CFD part.
- For each body, an Ansys ICEM CFD Material Point is created and put into the corresponding Ansys ICEM CFD part. The names of Material Points have the suffix _MATPOINT.
- If a mesh has been generated, it is exported along with the geometry. In such cases, these additional rules are followed:
  - As long as they are *not* contained in a Named Selection, node/line/surface mesh cells are associated with the corresponding geometry part/body in Ansys ICEM CFD.
  - As long as they are *not* contained in a Named Selection, volume mesh cells are associated with the Material Point part.
  - Mesh cells that are associated with geometry entities that are contained in a Named Selection are associated with the Ansys ICEM CFD part that corresponds to that Named Selection.

The first example is a model consisting of four separate single body parts in Ansys Workbench. The single body parts are named fluid1, fluid2, fluid3, and fluid4. The table below shows the geometry in Ansys Workbench and the corresponding part names that will appear in Ansys ICEM CFD:

This geometry in Ansys Workbench...	Results in these part names in Ansys ICEM CFD...
A model consisting of four separate single body parts named:	
fluid1	FLUID1_1
fluid2	FLUID2_2
fluid3	FLUID3_3
fluid4	FLUID4_4

The figure below shows the model after it was meshed in the Meshing application:

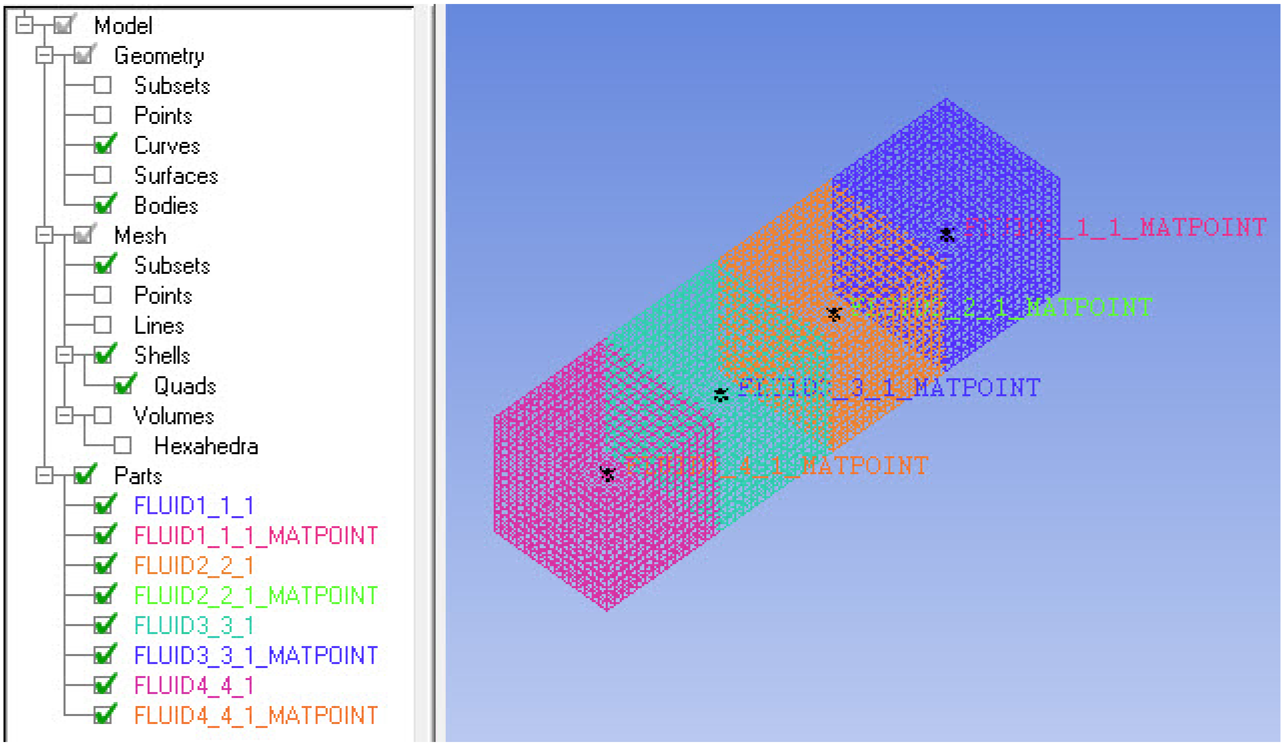
Figure 13: Meshed Model (Four Separate Workbench Parts) Ready for Export to Ansys ICEM CFD



Next, the model was exported from the Meshing application to Ansys ICEM CFD format. In the figure below, the corresponding .prj file has been opened in Ansys ICEM CFD. Notice the names that are assigned to the various entities in the Ansys ICEM CFD format file:

- Each body/part name has been suffixed with a unique integer to distinguish it from similarly named bodies/parts. (Note that in this example, *part_name* is equal to *body_name*.)
- Each single body part in Ansys Workbench appears as <*part_name*>_<*part_index*> in the Ansys ICEM CFD format files. For example, the part named fluid1 in Ansys Workbench has a part name of FLUID1_1 in Ansys ICEM CFD, which appears as FLUID1_1_1 in the Ansys ICEM CFD format files after the *part_index* is added.
- For each body in the Ansys Workbench file (fluid1, fluid2, fluid3, fluid4), a Material Point has been assigned (FLUID1_1_1_MATPOINT, FLUID2_2_1_MATPOINT, FLUID3_3_1_MATPOINT, FLUID4_4_1_MATPOINT).

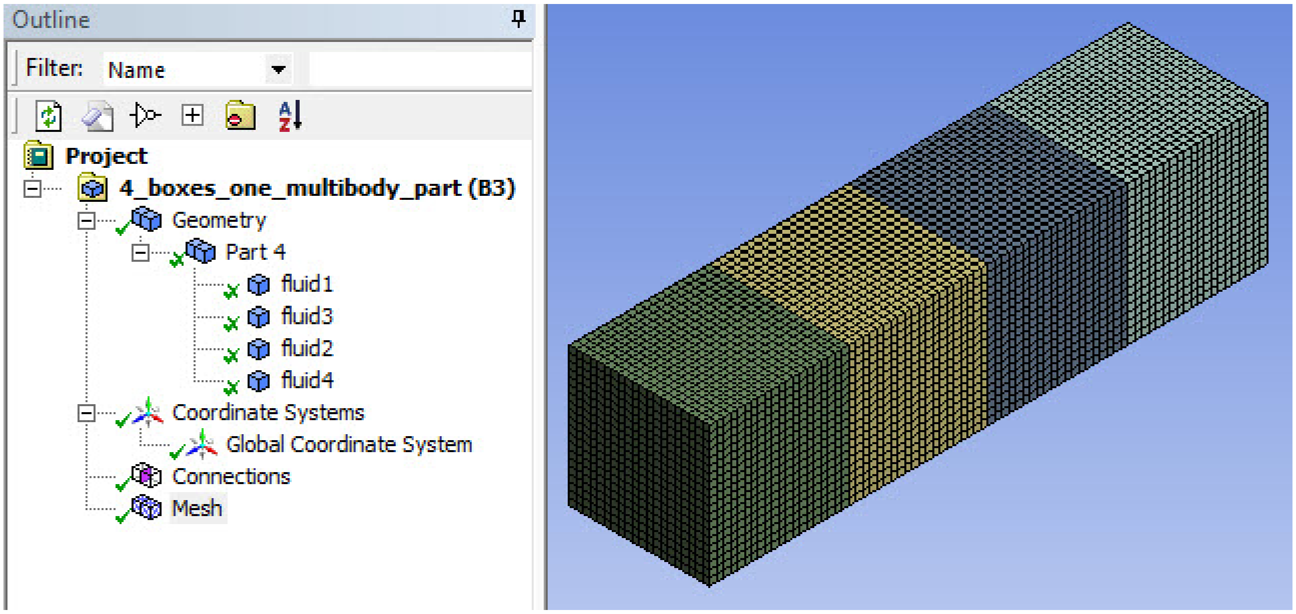
Figure 14: Opening the .prj File (Four Separate Workbench Parts) in Ansys ICEM CFD



The second example is a model consisting of one multibody part in Ansys Workbench. The multibody part, which is named Part 4, contains four bodies named fluid1, fluid2, fluid3, and fluid4. The table below shows the geometry in Ansys Workbench and the corresponding part names that will appear in Ansys ICEM CFD:

This geometry in Ansys Workbench...	**Results in these part names in Ansys ICEM CFD (the / character denotes hierarchy)...**
A model consisting of one multibody part named Part 4, containing four bodies named:	
fluid1	PART_4_1/FLUID1_3
fluid2	PART_4_1/FLUID2_2
fluid3	PART_4_1/FLUID3_1
fluid4	PART_4_1/FLUID4_4

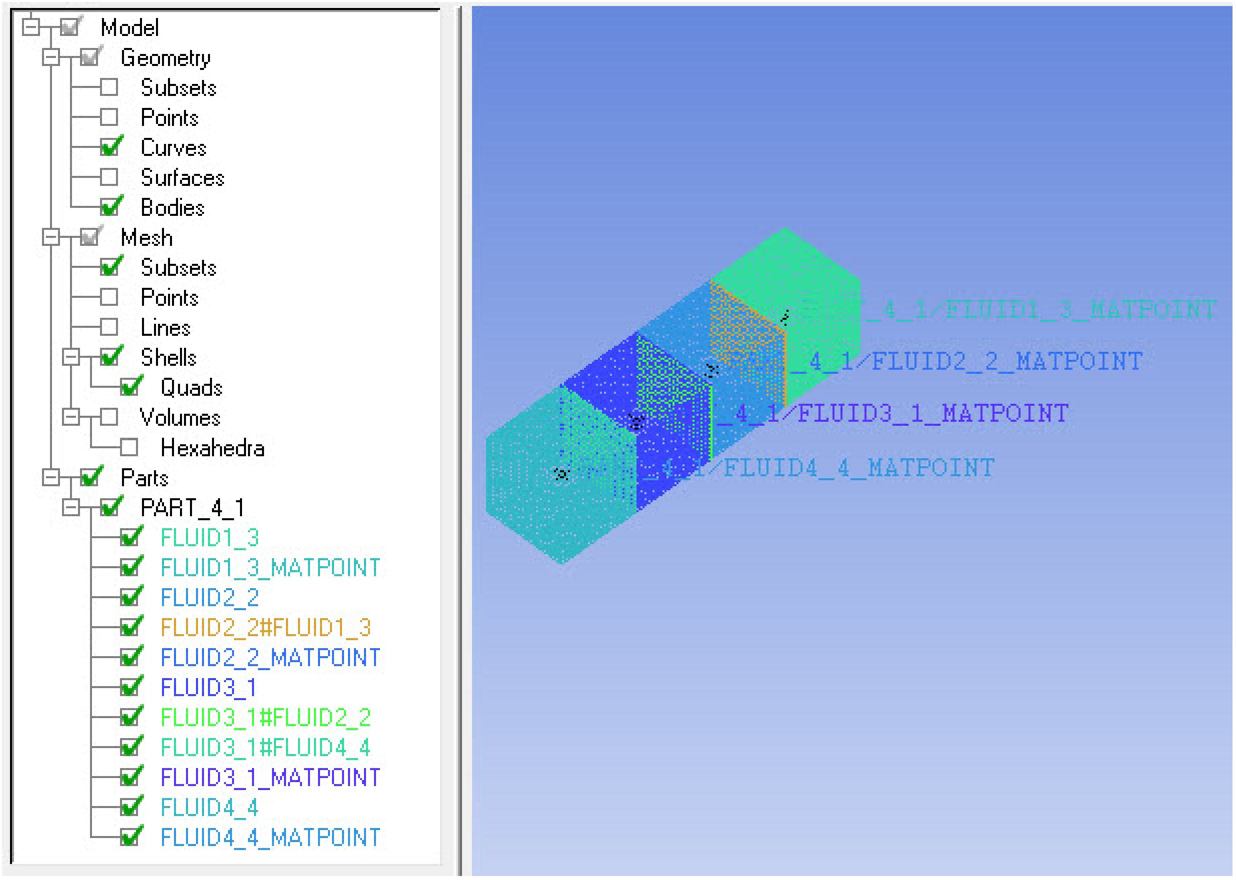
Figure 15: Meshed Model (One Multibody Workbench Part) Ready for Export to Ansys ICEM CFD



Next, the model was exported from the Meshing application to Ansys ICEM CFD format. In the figure below, the corresponding .prj file has been opened in Ansys ICEM CFD. Notice the names that are assigned to the various entities in the Ansys ICEM CFD format file:

- Each body/part name has been suffixed with a unique integer to distinguish it from similarly named bodies/parts.
- Each multibody part in Ansys Workbench appears as *<part_name>_<part_index>/<body_name>_<body_index>* in the Ansys ICEM CFD format files. For example, the fluid1 body in Part 4 in Ansys Workbench has a part name of PART_4_1/FLUID1_3 in the Ansys ICEM CFD format files.
- The bodies that are in the multibody part in the Ansys Workbench file (fluid1, fluid2, fluid3, and fluid4) have been put into an Ansys ICEM CFD assembly named Part_4.
- The faces that are shared between the various pairs of bodies have been named FLUID2_2#FLUID1_3, FLUID3_1#FLUID2_2, and FLUID3_1#FLUID4_4.
- For each body in the Ansys Workbench file (fluid1, fluid2, fluid3, fluid4), a Material Point has been assigned (FLUID1_3_MATPOINT, FLUID2_2_MATPOINT, FLUID3_1_MATPOINT, FLUID4_4_MATPOINT).

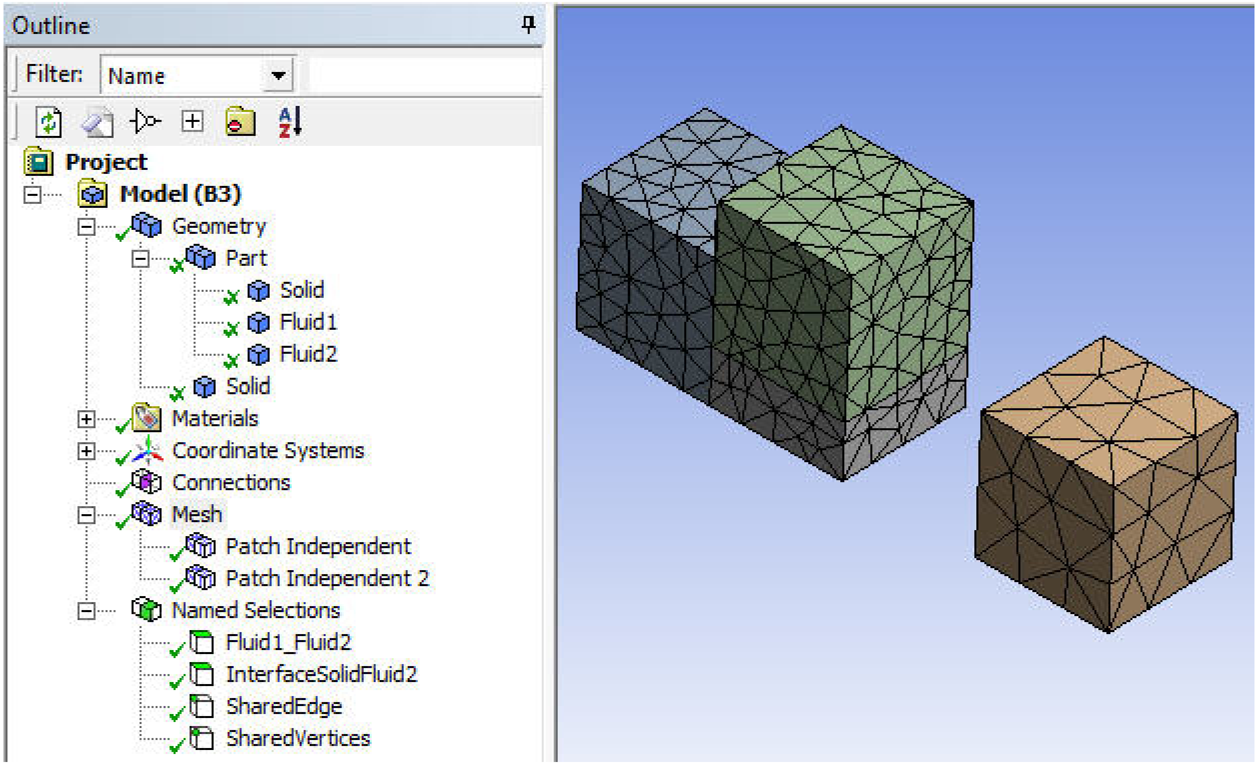
Figure 16: Opening the .prj File (One Multibody Workbench Part) in Ansys ICEM CFD



The third (and final) example involves a model for which four Named Selections are defined in the DesignModeler application. The model is meshed in the Meshing application, exported to Ansys ICEM CFD format, and opened in Ansys ICEM CFD.

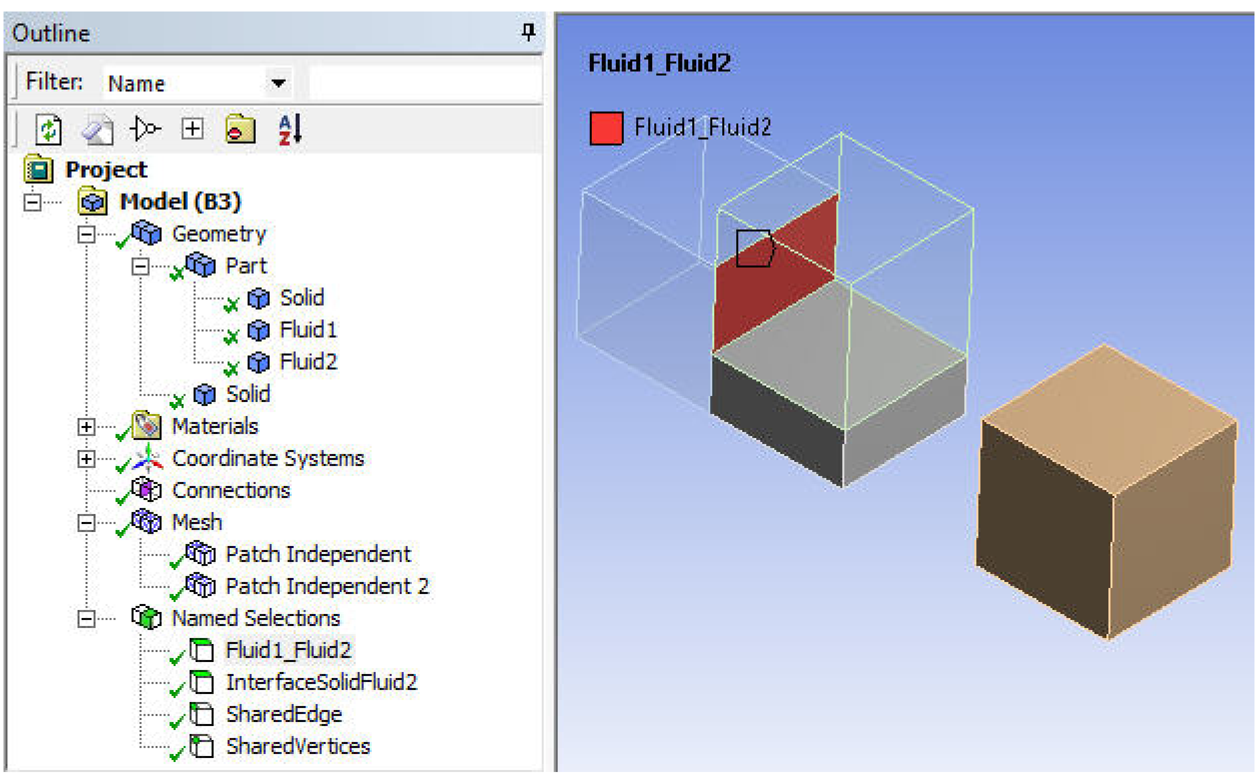
The first figure shows the model after it was meshed in the Meshing application.

Figure 17: Meshed Model (with Named Selections) Ready for Export to Ansys ICEM CFD



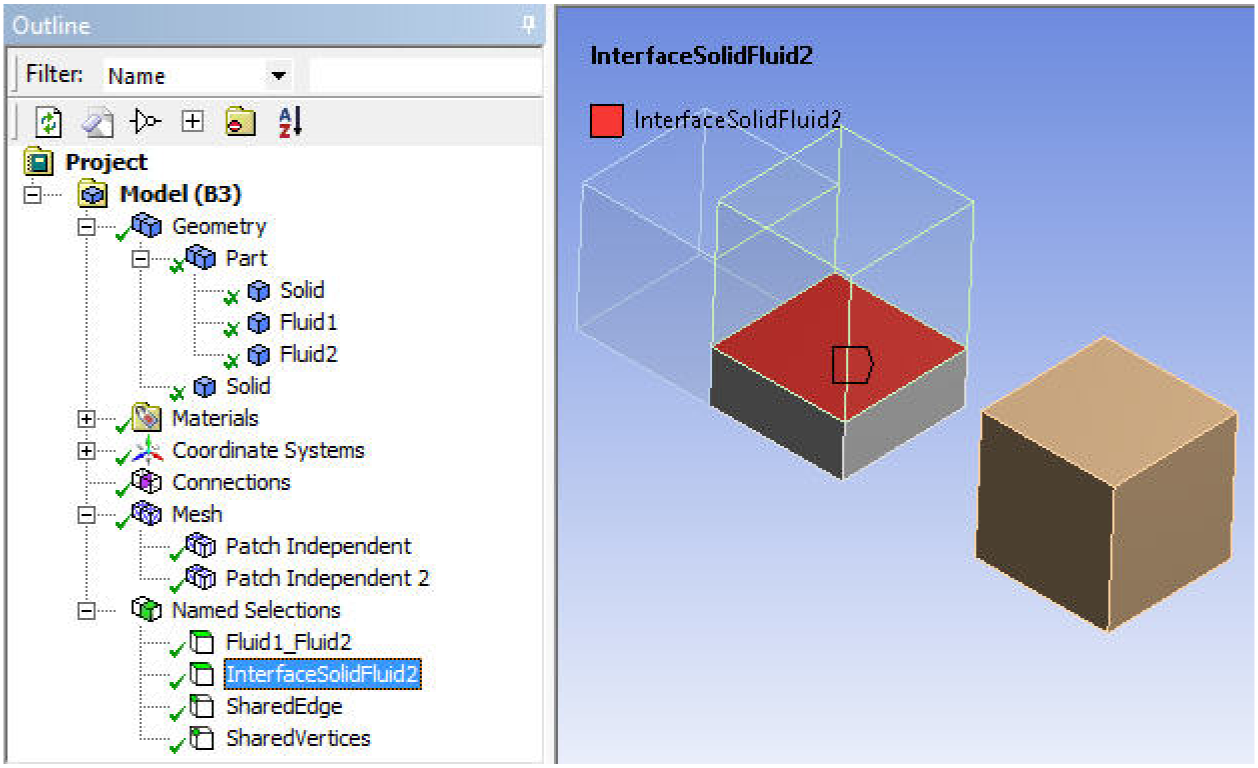
The next four figures show the entit(ies) in the model that are contained in each of the four Named Selections. In the figure below, the Fluid1_Fluid2 Named Selection is highlighted.

Figure 18: Fluid1_Fluid2 Named Selection



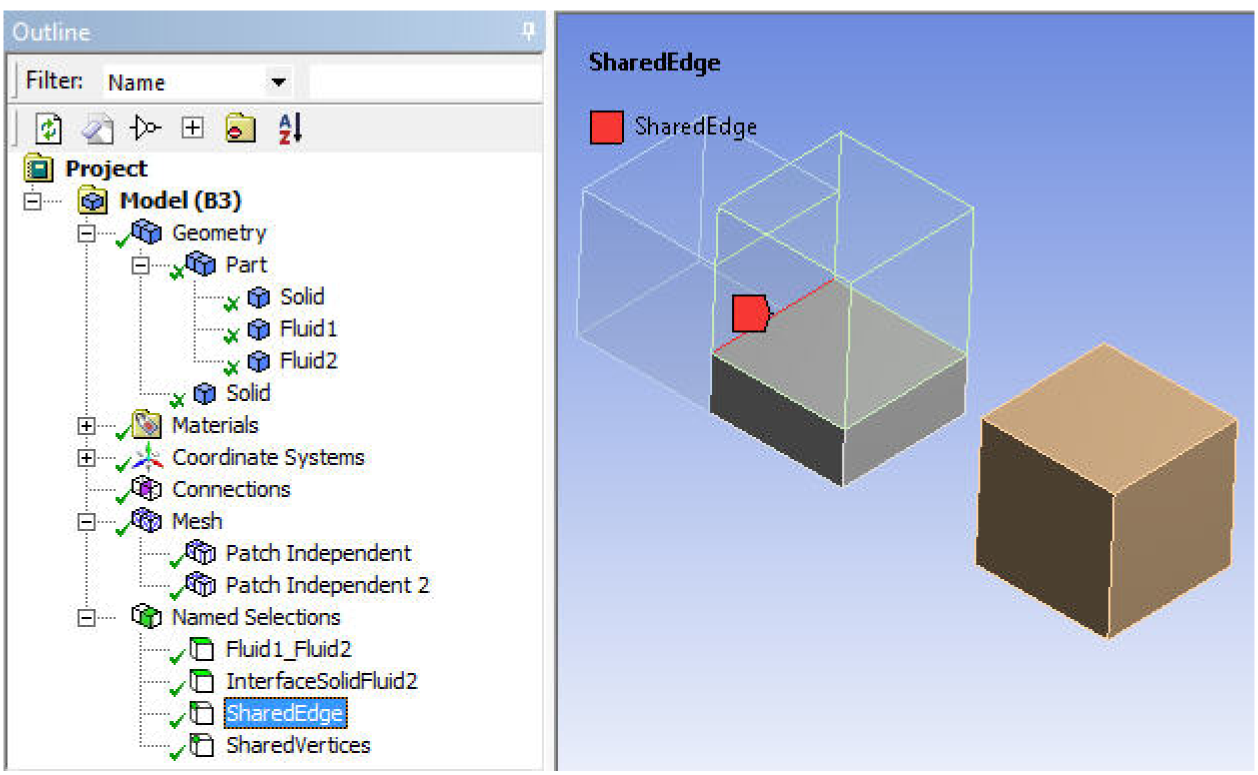
In the figure below, the InterfaceSolidFluid2 Named Selection is highlighted.

Figure 19: InterfaceSolidFluid2 Named Selection



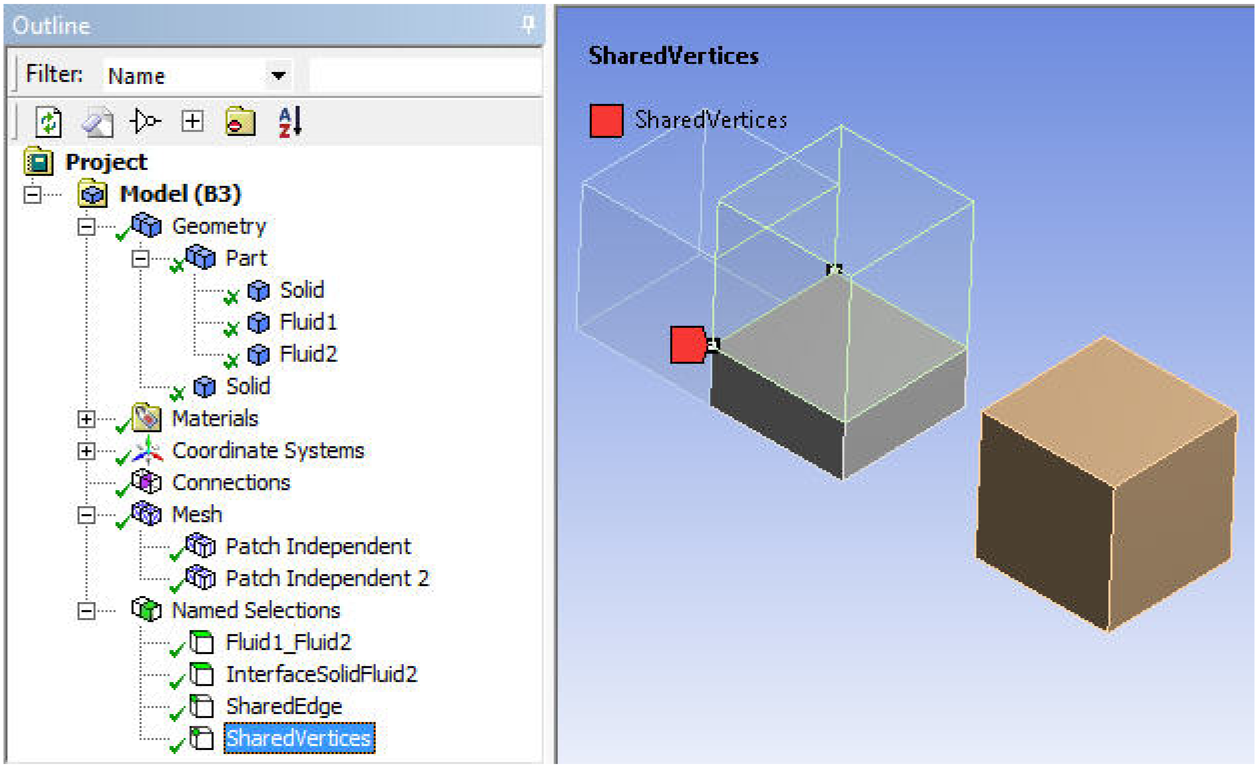
In the figure below, the SharedEdge Named Selection is highlighted.

Figure 20: SharedEdge Named Selection



In the figure below, the SharedVertices Named Selection is highlighted.

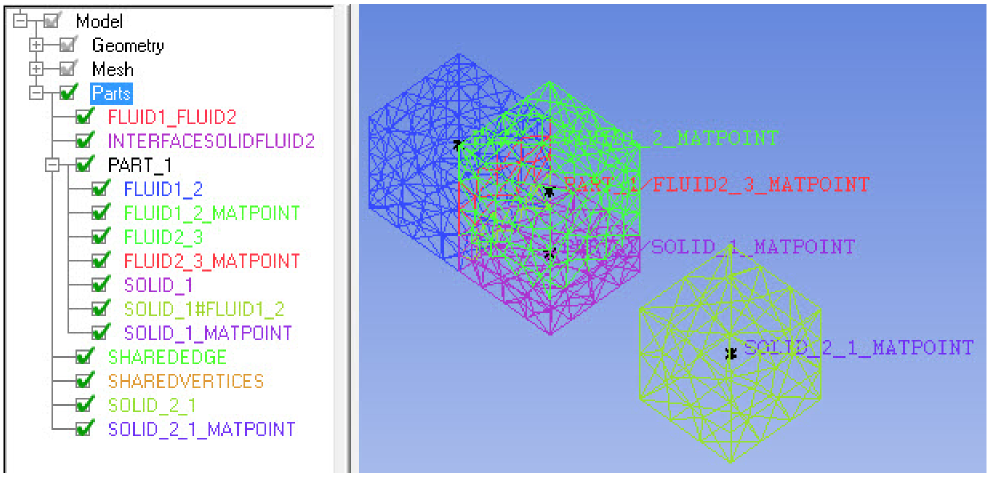
Figure 21: SharedVertices Named Selection



Next, the model was exported from the Meshing application to Ansys ICEM CFD format. In the figure below, the corresponding .prj file has been opened in Ansys ICEM CFD. Notice the names that are assigned to the various entities in the Ansys ICEM CFD format file:

- Each body/part name has been suffixed with a unique integer to distinguish it from similarly named bodies/parts.
- The bodies that are in the multibody part in the Ansys Workbench file (Solid, Fluid1, and Fluid2) have been put into an Ansys ICEM CFD assembly named Part_1.
- The face that is shared between SOLID_1 and FLUID1_2 has been named SOLID_1#FLUID1_2.
- Because Fluid1_Fluid2, InterfaceSolidFluid2, SharedEdge, and SharedVertices are all Named Selections in the Ansys Workbench file, each of them has been put into a separate Ansys ICEM CFD part.
- For each body in the Ansys Workbench file (Solid, Fluid1, Fluid2, Solid), a Material Point has been assigned (SOLID_1_MATPOINT, FLUID1_2_MATPOINT, FLUID2_3_MATPOINT, and SOL-ID_2_1_MATPOINT).

Figure 22: Opening the .prj File (with Named Selections) in Ansys ICEM CFD



Note:

For additional information, refer to the documentation available under the Help menu within Ansys ICEM CFD.

Exporting Faceted Geometry to Ansys Fluent Meshing

You can use the Meshing application to export faceted geometry for use in Ansys Fluent Meshing (formerly TGrid):

1. Select **File**> **Export** from the main menu.

2. In the **Save As** dialog box, choose a directory and specify a file name for the file. Then choose **TGrid Faceted Geometry File** from the **Save as type** drop-down menu and click **Save**.

As a result, a file with the extension .tgf is created. The exported file can be imported into Ansys Fluent Meshing, where you can use such features as the Ansys Fluent Meshing wrapper utility.

Note:

The .tgf file has the same format as a .msh file and will be recognized as a "Mesh File" when read into Ansys Fluent Meshing (File/Read/Mesh... menu item).

Upon export, Ansys Fluent Meshing objects and zones are created according to geometry bodies and Named Selections. Part and body names are used in the Ansys Fluent Meshing object/zone names to identify the parts and bodies from which they originated.

Remember the following information when exporting to Ansys Fluent Meshing:

- The quality of the exported facets is derived from the CAD system. You can use the **Facet Quality** option (**Tools** > **Options** > **DesignModeler** > **Graphics** > **Facet Quality**) to control the quality of the exported facets (the higher the setting, the higher the quality). However, be aware that higher settings create large numbers of facets, which can slow down the Meshing application.

- The part, body, and Named Selection names that were present in the Meshing application are exported in all lowercase characters for use in the corresponding Ansys Fluent Meshing zone names.
- Only part and body names that were imported into the Meshing application are used in the exported zone name. For example, names that were initially defined in the Ansys DesignModeler application or initially appeared in the Tree Outline when a CAD file was loaded directly into the Meshing application will be used. Any subsequent renaming of parts and bodies that occurs in the Meshing application will not be taken into account in the exported zone names.
- Vertices (regardless of whether they are contained in a Named Selection) are ignored.
- The name of each Named Selection is filtered upon export such that only allowable characters remain in the name of the Ansys Fluent Meshing zone. Allowable characters include all alphanumeric characters as well as the following special characters:

  _ + - : .

  All other characters, including spaces, are invalid. If an invalid character is used, it is replaced by a hyphen (-) upon export.
- When the same entity is a member of more than one Named Selection, those Named Selections are said to be "overlapping." If you are exporting faceted geometry into the Ansys Fluent Meshing format (or a mesh into the Ansys Polyflow, CGNS, or Ansys ICEM CFD format), and overlapping Named Selections are detected, the export will fail and you must resolve the overlapping Named Selections before proceeding. For details, see Repairing Geometry in Overlapping Named Selections (p. 79).
- For assembly (p. 367) meshing algorithms, the names of parts, bodies, and Named Selections should be limited to 64 characters.

The figures below illustrate the process of exporting geometry in faceted representation from the Meshing application to Ansys Fluent Meshing. Figure 23: Part, Body, and Named Selection Names in the Meshing Application (p. 76) shows the model in the Meshing application. The geometry consists of a multibody part named AeroValve, and the three bodies that AeroValve contains are named Outletbody, Valve, and Inletbody. Notice that three Named Selections have been defined and are highlighted in the **Geometry** window: Inlet, Outlet, and Valve_opening.

Figure 23: Part, Body, and Named Selection Names in the Meshing Application

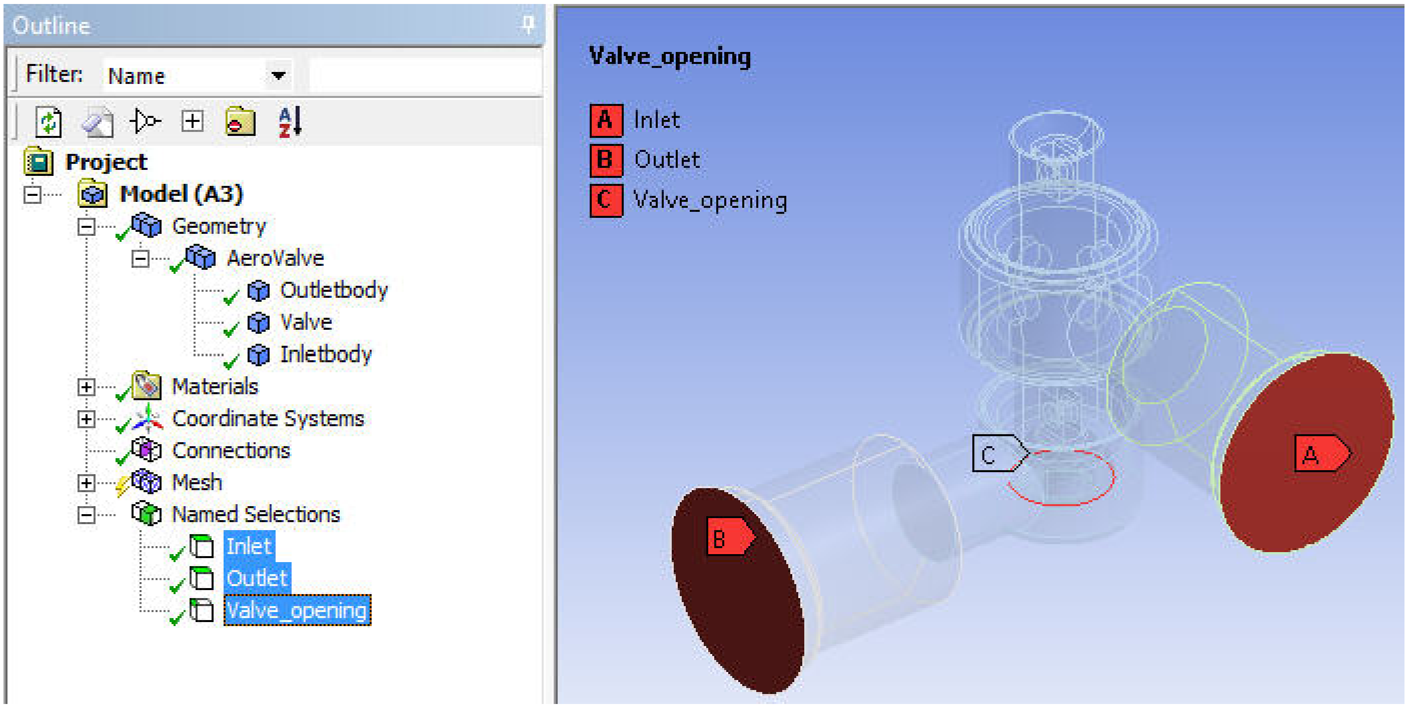
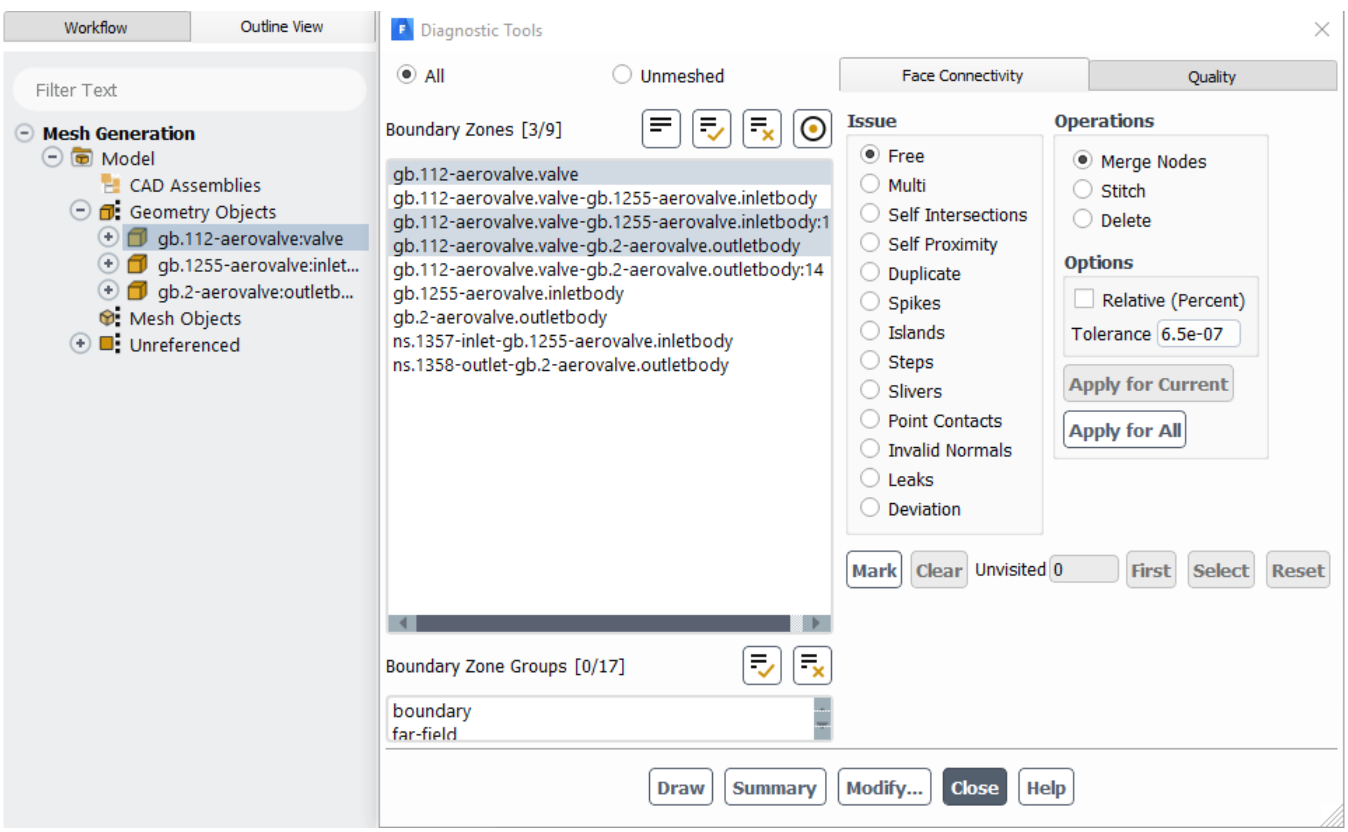


Figure 24: Objects/Zone Names Transferred to Ansys Fluent Meshing (p. 76) shows the Surface Retriangulation panel after the exported .tgf file is imported into Ansys Fluent Meshing.

Figure 24: Objects/Zone Names Transferred to Ansys Fluent Meshing



Named Selections and Regions for Ansys CFX

There are a number of places in the Ansys Workbench and CFX applications where geometric faces and regions of mesh can be grouped together and assigned names. While this provides a large amount of flexibility, it can lead to confusion. To have more control over which groups of names are loaded into CFX to help simplify this issue, this section describes the 'best practices' for region definitions and is targeted mainly at Ansys CFD users of the DesignModeler, Meshing, and CFX applications.

Note:

Refer to Passing Named Selections to the Solver (p. 78) for related information.

Defining Names in the DesignModeler Application or the Meshing Application

When creating or importing geometry in the DesignModeler application or editing a mesh in the Meshing application, Named Selections can be defined in terms of one or more CAD faces. If it is desirable for these Named Selections to be available in CFX-Pre, then they must adhere to these simple rules: Named Selections should not overlap or be defined as multi-dimensional and all Named Selections must have unique names.

Importing DesignModeler Named Selections into the Meshing Application

If the Named Selections defined in the DesignModeler application are required in the Meshing application, you must set the appropriate geometry import options to ensure the Named Selections will be transferred properly:

1. From the Ansys Workbench Menu Bar, select **Tools> Options**.
2. In the left pane of the **Options** dialog box, select **Geometry Import**.
3. In the right pane of the **Options** dialog box, select the **Named Selections** check box and either clear the **Filtering Prefixes** field (to import all names) or set it to the appropriate filter if required.

The next time that you attach geometry containing Named Selections and launch the Meshing application, the application will start, load the geometry, and load any Named Selections previously defined for that geometry. The preferences you set on the **Options** dialog box are local settings and affect only you.

If a mesh is generated, the Named Selections defined in the DesignModeler application will be available when the mesh is imported into CFX-Pre.

Note:

For detailed descriptions of the geometry import options, see the CAD Integration section in the Ansys Workbench help.

Using Multiple Mesh Methods

Named Selections can be defined in the Meshing application, the DesignModeler application, or supported CAD systems. When the resulting mesh is loaded into CFX-Pre, all of these Named Selections are available. Control over which names are imported with the mesh is available in the Mesh Import options.

It is possible to define Named Selections that conflict with some virtual topology, or do not conform to the CFX Topology Model in some other way. Such Named Selections will not be imported into CFX-Pre.

If a Named Selection is created that contains characters not supported in CFX-Pre names, these names will be modified on import to remove the illegal characters. Note that non-ASCII characters are illegal in CFX-Pre names.

Invalid Named Selections

The use of Virtual Topology within the Meshing application can invalidate Named Selection definitions. Consider the case where a Named Selection 'Region1' is defined in terms of 4 CAD faces (1, 2, 3 and 4). If this geometry is then loaded into the Meshing application and Virtual Topology is used to merge 2 of these faces (3 and 4), the original Named Selection can no longer be resolved. In CFX-Pre, the Named Selection will be modified so that it only refers to the 'resolvable' faces. Hence, in CFX-Pre, the Named Selection 'Region1' would be defined in terms of faces 1 and 2 only.

Named Selections Involving Interior Faces

If an interior face (for example, a face that is shared between two bodies in the same part in DesignModeler) is included in a Named Selection in the Meshing application, then the corresponding location in CFX-Pre will be a Composite region that includes two Primitive regions: one for each side of the interior face. Care should be taken when using the resulting Composite region for evaluating expressions and performing calculations: for example, areas evaluated on such a Composite region would be likely to give double the expected result (since the area is evaluated over both of the primitive regions that form the two sides of the interior face) and mass flows will be approximately zero (since the mass flow is evaluated on both of the primitive regions and will have the opposite sign on each). You can optionally edit the Composite region in CFX-Pre to include only one of the primitive regions.

Passing Named Selections to the Solver

When defining Named Selections, you can use the **Send to Solver** option to control whether a Named Selection is passed to the solver. **Send to Solver** is available in both the Mechanical and Meshing applications.

The default is **Yes** for Named Selections that you create, and **No** for Named Selections that are generated automatically by the **Mesh** worksheet (p. 409). When set to **No**, the selected Named Selection is not passed to Ansys Fluent, Ansys CFX, Polyflow, CGNS, Ansys ICEM CFD, Ansys Fluent Meshing, Mechanical APDL, Ansys Autodyn, or any other downstream system that may be connected to the Mechanical or Meshing application.

Repairing Geometry in Overlapping Named Selections

When you export (p. 42) a mesh into the Ansys Fluent, Polyflow, CGNS, or Ansys ICEM CFD format, or as a faceted geometry (p. 74) for use in Ansys Fluent Meshing, each Named Selection must be unique. In other words, the entities (bodies, faces, or edges) in one Named Selection cannot also exist in a second Named Selection. Any entity that is in two or more Named Selections is considered overlapping. The overlapping entities need to be resolved before the mesh can be exported.

Note:

Any Named Selection whose **Send to Solver** (p. 78) option is set to **No** is skipped during the processing described above. For this reason, an easy way to avoid overlapping Named Selections is to set all values of **Send to Solver** to **No**.

If a mesh fails to export and an error about overlapping Named Selections occurs, you can display the overlapping entities by right-clicking on the **Mesh** object in the Tree Outline and choosing **Show**>**Geometry in Overlapping Named Selections**. You can then inspect the Named Selections to remove the duplicate (overlapping) entities and proceed with exporting the mesh.

Alternatively, you can repair the overlapping Named Selections automatically by right-clicking on the **Named Selections** folder and choosing **Repair Overlapping Named Selections**. This option checks all Named Selections (for which **Send to Solver** is set to **Yes**) in the order in which they appear in the Tree (top to bottom). When it encounters an overlapping Named Selection, it moves that Named Selection to a new folder called **Overlapping Named Selections** and sets its **Send to Solver** option to **No**. You can delete the **Overlapping Named Selections** folder.

The repaired Named Selections are placed at the bottom of the Tree and their names are prefixed with **Repaired**. To repair a Named Selection, the Meshing application removes any duplicate (overlapping) entities from the Named Selection and sets its **Send to Solver** option to **Yes**.

Note:

- In cases where you want overlapping Named Selections to be converted to Ansys ICEM CFD parts, the overlapping subsets can be cleaned in Ansys ICEM CFD and then converted into parts.
- If you are using Named Selections for mesh settings (such as sizing), set the **Send to Solver** (p. 78) option for these Named Selections to **No** so that they are skipped when the check for overlapping Named Selections is performed. Otherwise, the check may cause the mesh state to go out-of-date.

Resolving Overlapping Contact Regions

Contact regions are said to be "overlapping" when the same entity (face or edge) is a member of more than one contact region; or when multiple contact regions share the same geometry. When you export a mesh to Ansys Fluent or Ansys CFX, contact regions become mesh interfaces, and interfaces must be unique. In other words, if there are overlapping contact regions, the overlaps need to be resolved into exclusive groups for the export to be successful.

Overlapping entities can come from a variety of sources. In the example shown in the following series of images, the first contact region has one face as a contact and one as a target; this is a good connection for Ansys Fluent or Ansys CFX. The other three contact regions have two faces as targets and only one face as a contact; the extra target face in each of these contact regions overlaps, resulting in overlapping contact regions that may need to be addressed for the export to succeed. In this particular case, the default tolerance that was used to detect contacts was too large. This problem could be fixed by deleting all the contacts and using a smaller Face Overlap Tolerance to recreate them.

Figure 25: First Contact Region: One Contact and One Target

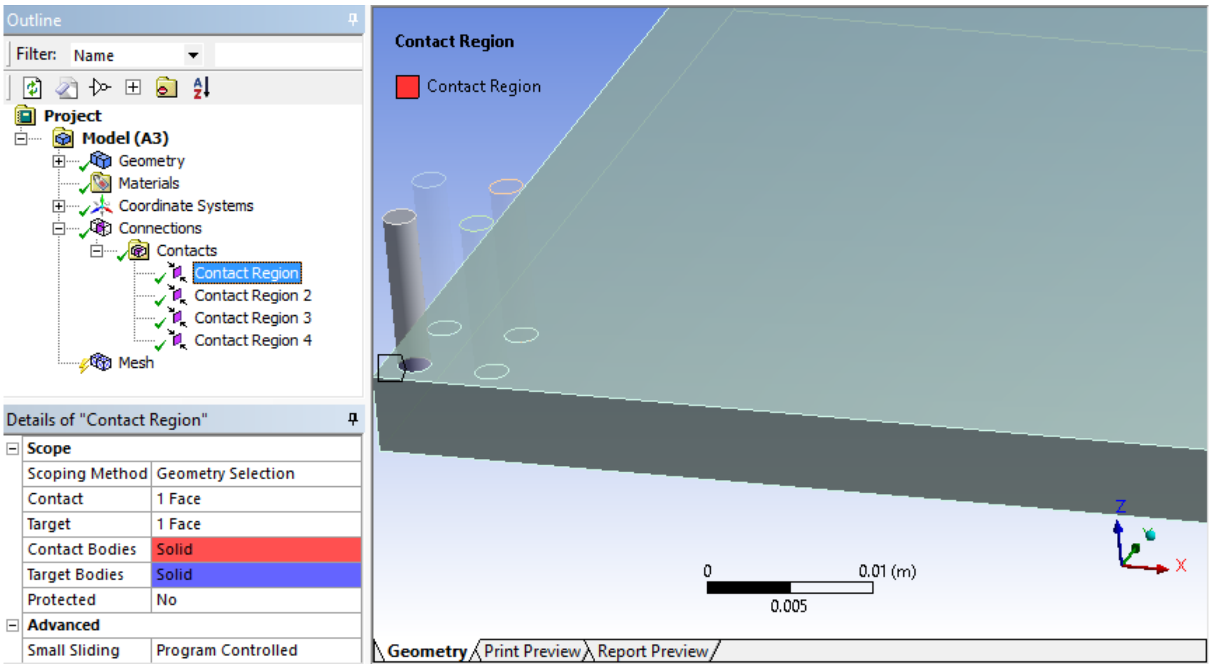


Figure 26: Second Contact Region: One Contact, Two Targets

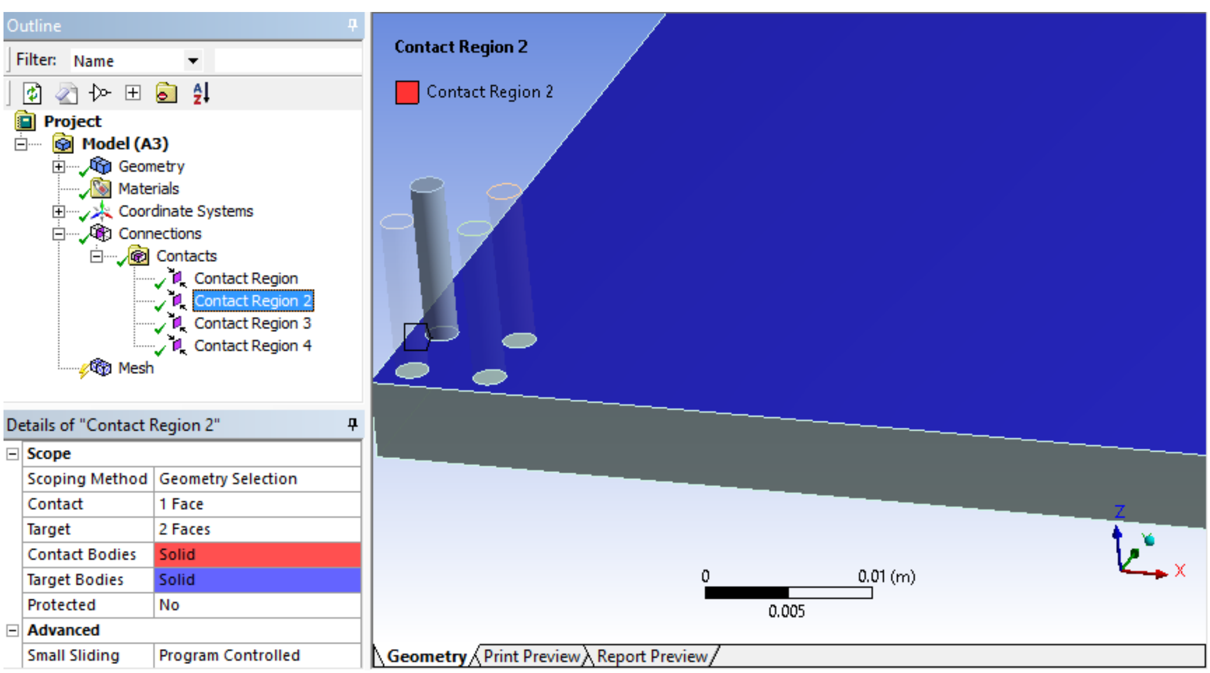


Figure 27: Third Contact Region: One Contact, Two Targets

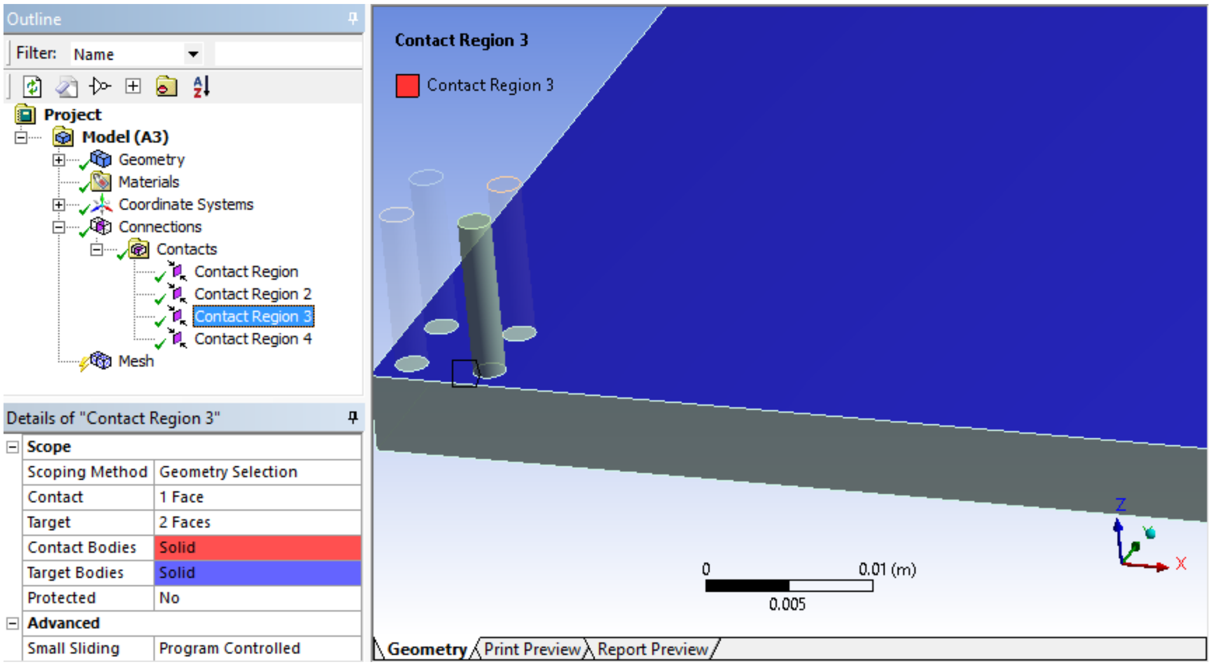
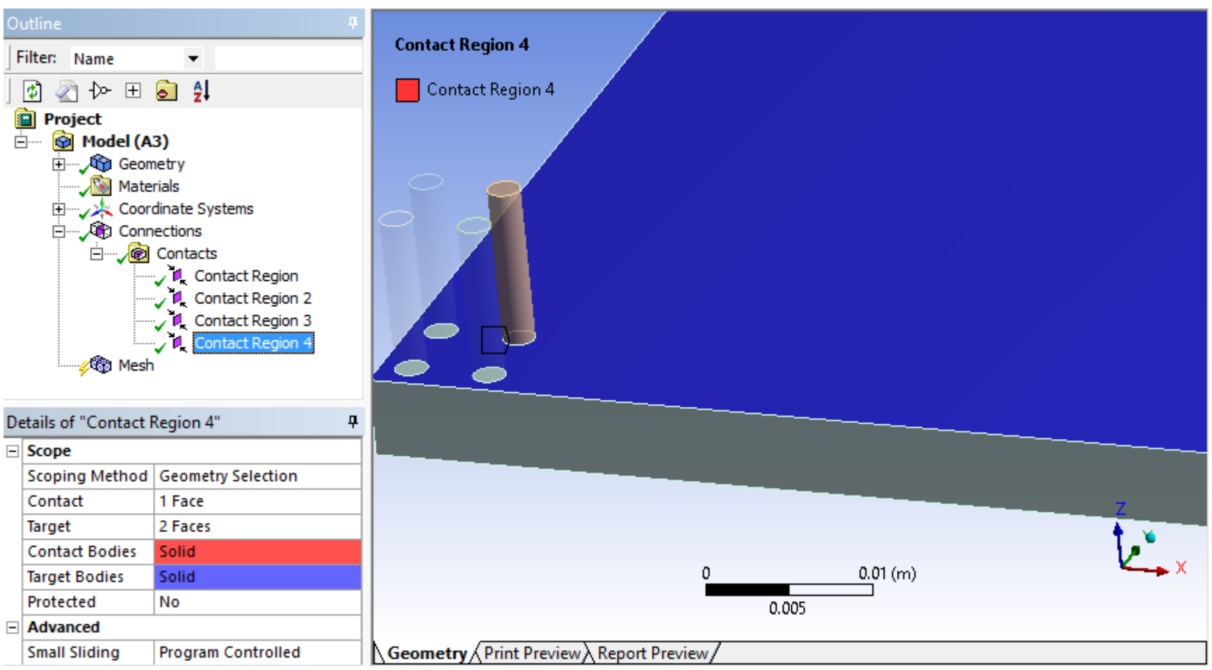


Figure 28: Fourth Contact Region: One Contact, Two Targets



Other cases of overlap may have different causes and solutions. To identify overlapping contact regions so that you can resolve them and export the mesh successfully, follow these steps:

1. Right-click the **Connections**, **Connection Group**, or **Contacts** folder in the Tree and choose **Check Overlapping Contact Regions**.

   If any overlapping contact regions are found, an informational message appears in the **Messages** window.

2. Right-click the message and select **Go To Object**.

   The overlapping contact regions that are responsible for the message are highlighted in the Tree.

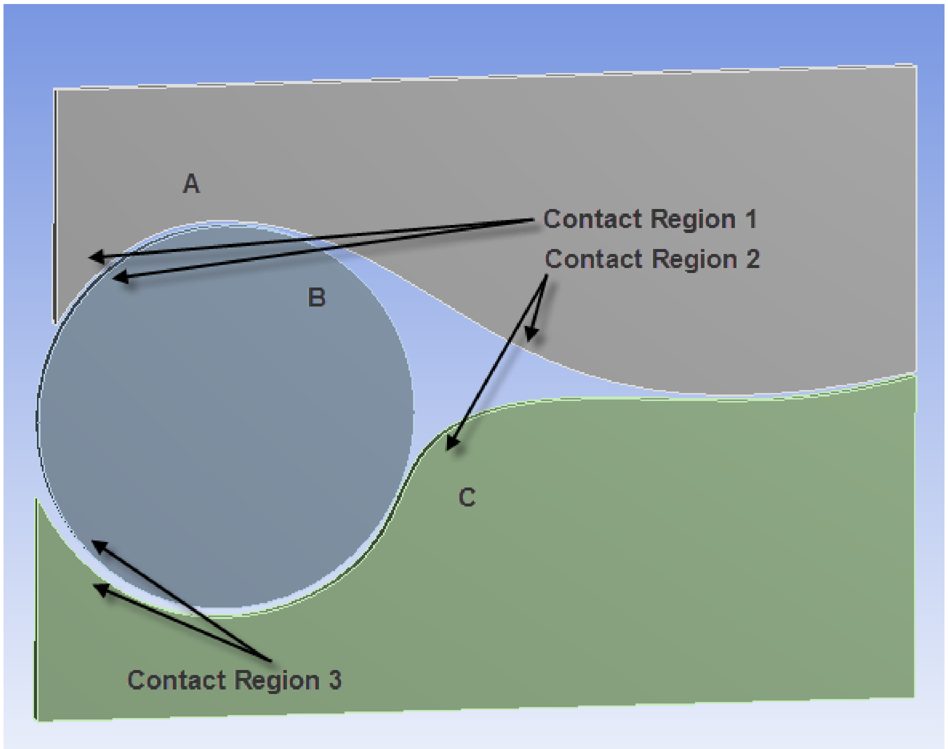
3. With all of the overlapping contact regions still highlighted in the Tree, right-click one of them and select **Group**.

   All of the overlapping contact regions are grouped into a new folder to isolate them for diagnosis.

To resolve the overlapping contact regions, select each contact region that was placed into the new folder one at a time and examine its properties. Verify that all associated properties are defined appropriately. The following approaches are recommended:

- Often tolerances are set too large, as described in the example shown above. Try deleting all the contacts and using smaller tolerance values to recreate them. It may help to set **Tolerance Type** to **Value** rather than **Slider**, so that you can enter an exact **Tolerance Value**. Make sure the Tolerance Value and Face Overlap Tolerance are set to reasonable values before regenerating the contacts.
- Depending on the case, setting Group By to **Faces** may help to pinpoint problematic faces. In other situations, setting **Group By** to **Bodies** and promoting the overlapping contact regions to Named Selections may be a better approach.
- If the tolerances are different for different bodies, run contact on selected bodies with different tolerances.
- If the faceting of the geometry is too coarse, increase the facet tolerance in DesignModeler.
- If some of the overlapping contact regions are not needed in Ansys Fluent or Ansys CFX, you can suppress them so they will not affect the success of the export.
- If the case involves cyclic constraints, use imprinting in DesignModeler or SpaceClaim. Imprinting results in exact pairs and ensures faces have common boundaries between parts. Refer to Figure 29: Geometry with Cyclic Redundancies (p. 83), which illustrates a complex case of cyclic redundancy. Here the overlap is A->B->C->A, where A is the face contact of the top body, B is the cylindrical face of the middle body, and C is the face in contact of the bottom body.

Figure 29: Geometry with Cyclic Redundancies



After you have resolved all of the overlapping contact regions, you can retry the export.

Note:

This feature is applicable to part-based meshing only. The check for overlapping contact regions does not occur when assembly (p. 367) meshing algorithms are being used.

Extended Ansys ICEM CFD Meshing

The features described here extend Meshing application functionality through integration of the Meshing application with Ansys ICEM CFD, and enable you to use Ansys Workbench to drive automation of Ansys ICEM CFD.

With this approach you can leverage advantages of Ansys Workbench such as:

- Its capabilities for running design optimization and what-if scenarios using parameters
- The strength of its core product solvers

The following topics are discussed in this section:

Writing Ansys ICEM CFD Files

The **Write ICEM CFD Files** control determines whether Ansys ICEM CFD files are written, and includes options for running Ansys ICEM CFD interactively or in batch mode from an Ansys ICEM CFD Replay file.

Note:

The **Write ICEM CFD Files** control is available when you use any Ansys ICEM CFD mesh method that is available in the Meshing application (Patch Independent Tetra (p. 200), MultiZone (p. 228), or MultiZone Quad/Tri (p. 246)). However, the **Interactive** and **Batch** options are available only if Ansys ICEM CFD is installed. A warning is issued if you try to use one of these options and Ansys ICEM CFD cannot be found.

Note:

Another method for writing meshing data to ICEM CFD files is to select the Mesh cell in Workbench and select **Transfer Data to New>ICEM CFD**. While the **Write ICEM CFD** control writes the geometry of a part, using the Ansys ICEM CFD Add-in component links to the entire Geometry assembly. For more information, see Component Systems in the Ansys Workbench User's Guide.

Options for **Write ICEM CFD Files** include:

No

No files are written. This is the default.

Yes

Writes Ansys ICEM CFD files. Useful when you are working in Ansys Workbench but you want to export your project files for further mesh editing in Ansys ICEM CFD. If this control is set to **Yes**, your Ansys ICEM CFD project (.prj), geometry (.tin), unstructured domain (.uns), and blocking (.blk) files will be saved during mesh generation.

- *If your Ansys Workbench project file has been saved,* your Ansys ICEM CFD files will be written to the Ansys Workbench project directory. Ansys Workbench creates a project folder as the top level folder for each project, at the same level as the project file. The project folder will be named <*filename*>_files, where <*filename*> is a name you provide. The project file will be named <*filename*>.wbpj. Under the project folder is a design point subdirectory for each design point in the project. The active design point is always design point 0 (dp0) and corresponds to the design point labeled **Current** in the Ansys Workbench GUI. Under each design point folder are system folders for each system type in the project (Mechanical, Fluent, etc.). Each system folder contains solver-specific files and folders, such as input files, model directories, engineering data, resources, etc. Following this structure, the Ansys ICEM CFD files will be written to <*filename*>_files\dp0\global\MECH\SYS.

- *If your Ansys Workbench project file has not been saved,* your Ansys ICEM CFD files will be written to the temporary Ansys Workbench folder, as follows: `%TEMP%\WB_<computer>\unsaved_project_files\dp0\global\MECH\SYS`.

Note:

- Only appropriate files are written, based on the selected mesh method.
- Refer to the Ansys Workbench help for more information about project file management in Ansys Workbench.

Interactive

Applicable only when Ansys ICEM CFD is installed.

When the **Interactive** option is set and you select **Generate Mesh**, the Meshing application launches Ansys ICEM CFD in interactive mode. When you specify the **Interactive** option, you must also select an option for **ICEM CFD Behavior** (p. 86) to determine whether the geometry and/or mesh is transferred to Ansys ICEM CFD.

Typically you will run in interactive mode to set up an Ansys ICEM CFD Replay script file (*.rpl) that can be run later in either batch or interactive mode. You will begin by loading a generic Replay file. It is important to use this default Replay file because the batch process requires the pre and post steps that are defined within it. To load the default Replay file:

1. From within Ansys ICEM CFD, select **File** > **Replay Scripts** > **Replay Control** and the **Replay Control** window appears.
2. Click **Load** on the **Replay Control** window and the **Open Script File** window appears.
3. Click **Open** on the **Open Script File** window to load the default Replay file.

You can incorporate your custom commands into the Replay file by using the Replay Control feature or a text editor. The Replay file will be associated with your Ansys ICEM CFD project when you save the project and exit Ansys ICEM CFD.

After the mesh is returned to the Meshing application and the Ansys Workbench project is saved, the Replay file will be written to the Ansys Workbench project directory. Later you can set **Write ICEM CFD Files** to **Batch** and the mesh will be updated automatically in batch. You can change parameters on the Ansys Workbench project page and **Update** the mesh in batch from either the project page or from within the Meshing application.

Refer to Rules for Interactive Editing (p. 86) and Limitations of Ansys ICEM CFD Interactive (p. 86) for related information.

For more information about Replay Control, refer to the documentation available under the Help menu within Ansys ICEM CFD.

Batch

Applicable only when Ansys ICEM CFD is installed.

Runs Ansys ICEM CFD in batch mode from an existing Replay file. If you specify the **Batch** option, you must also select an option for **ICEM CFD Behavior** (p. 86) to determine whether the geometry and/or mesh is transferred to Ansys ICEM CFD.

ICEM CFD Behavior

Determines Ansys ICEM CFD behavior when running Ansys ICEM CFD in **Interactive** or **Batch** mode.

For Patch Independent Tetra, the available options under **Advanced** are:

- **Generate Mesh** - After the meshing operation completes, transfers both the geometry and mesh to Ansys ICEM CFD for editing.

- **Skip Meshing** - Bypasses the meshing operation and transfers only the geometry to Ansys ICEM CFD for meshing and editing.

For MultiZone and MultiZone 2D, the available options under **Advanced** are:

- **Generate Blocking and Mesh** - After the meshing operation completes, transfers the geometry, blocking, and mesh to Ansys ICEM CFD for editing.

- **Generate Blocking** - Bypasses the meshing operation and transfers only the geometry and blocking to Ansys ICEM CFD for meshing and editing.

- **Update pre-existing Blocking** - Bypasses the blocking operation, updates pre-existing blocking and meshes, and transfers the geometry and mesh to Ansys ICEM CFD for meshing and editing.

Rules for Interactive Editing

The final mesh must pass Ansys Workbench shape and topology checks in order for the mesh to be returned to Ansys Workbench. This requirement imposes the following rules and guidelines for editing in Ansys ICEM CFD:

- Do not modify the geometry.

- Pay attention to face and volume part naming. Use the same naming conventions and do not adjust the part naming.

- Pay attention to mesh quality. If the mesh quality becomes degraded, the mesh will not be returned to Ansys Workbench unless Check Mesh Quality (p. 118) is set to **No**.

- The final mesh must pass Ansys Workbench shape and topology checks in order for the mesh to be returned to Ansys Workbench without a warning or error message.

- Follow file naming conventions for proper archiving and batch interaction. Retain the default names. Changing the names will break the association between the Replay file and the Ansys Workbench project.

Limitations of Ansys ICEM CFD Interactive

Be aware of the following limitations when using Ansys ICEM CFD Interactive:

- Ansys ICEM CFD Interactive is designed to work at the part level. If you have assigned multiple Ansys ICEM CFD mesh methods (Patch Independent Tetra (p. 200), MultiZone (p. 228), or MultiZone Quad/Tri (p. 246)) to different bodies of a multibody part, the Ansys ICEM CFD Interactive options specified for the Ansys ICEM CFD method control that appears lowest in the Tree will be honored. In other words, those Ansys ICEM CFD Interactive options will affect all bodies being meshed with Ansys ICEM CFD methods regardless of whether a particular option is turned on for a particular body.

Working with Meshing Application Parameters

The term *parameters* in the Meshing application includes mesh input parameters (such as element size, height of the initial inflation layer, and growth rate) as well as mesh output parameters (number of elements, number of nodes, and mesh metric values).

A check box appears to the left of each field in the Details View that can be treated as a parameter. Clicking the check box causes a **P** to appear in the box, which indicates that the field has been exposed as a parameter for use in the Parameter Workspace. Fields that cannot be parameterized do not contain a check box and are left-aligned to save space. The Parameter Workspace collects all specified parameters and lists them in the Parameter Workspace grids for later use and/or modification.

Note:

- If an object has a parameterized field, and that object definition is changed in a way that makes that parameterization non-meaningful, the non-meaningful parameterization will *not* be removed by the program. For example, if there is a parameterized **Number of Divisions** sizing control defined in the Meshing application and you switch to the **Element Size** sizing control and parameterize it, the **Number of Divisions** parameterization will not be removed and will continue to appear in the Parameter Workspace grids. The presence of a non-meaningful parameter in the grids has no harmful effect, but you can delete the parameter manually if you do not want it to appear. To do so, return to the Meshing application and uncheck the corresponding check box in the Details View.
- If a mesh control is suppressed, the parameter associated with the control will be deleted.
- If you are using a parameterized field in a Design Point study, you must specify a real value for the field rather than using its default value.
- Refer to the Ansys Workbench help for detailed information about the Parameter Workspace and Design Points.

Ansys Workbench and Mechanical APDL Application Meshing Differences

While the meshing algorithms used by Ansys Workbench Mechanical originated from the meshing capabilities present in the Mechanical APDL (MAPDL) application, over time these algorithms have diverged. The divergence was due to the different focus of early versions of the Workbench Mechanical application (then known as DesignSpace). As Workbench Mechanical and its meshing and solving capabilities evolved, new technology was added and existing technology was enhanced, making Workbench Mechanical a full, general-purpose, finite element code that supports all levels of multiphysics disciplines.

To accommodate ease of use in Workbench Mechanical, as well as the need for different default meshes based on simulation type, physics preferences (p. 93) are available. These physics preferences automate default mesh settings related to element size, element quality, and so on. The **Mechanical** physics preference has two choices for **Error Limits** (p. 118), or shape check values:

- **Standard Mechanical**, which uses quality error limits that are less strict than those used by MAPDL.

- **Aggressive Mechanical**, which uses quality error limits that are similar to those used by MAPDL.

In addition to letting you set a value for **Error Limits**, Workbench Mechanical lets you set values for **Target Quality** and **Check Mesh Quality** (p. 118). The mesher uses the **Target Quality** as a goal for mesh quality. You can think of the **Target Quality** as a warning limit (in MAPDL terminology), but you can set the **Target Quality** however you deem appropriate. **Check Mesh Quality** is very similar to MAPDL's **Level of shape checking**. Depending on the setting of **Check Mesh Quality**, you can check for errors, errors and warnings, or neither (turn off checks altogether).

One of the major differences between **Standard Mechanical** and **Aggressive Mechanical** error limits is in the computation of the Jacobian ratio. The Jacobian Ratio (p. 132) is a metric that compares a given element's shape to that of an ideal element. Jacobian ratio can be computed at the corner nodes (**Aggressive Mechanical**) or at Gauss points (**Standard Mechanical**). Depending on the type of simulation, the calculation at the Gauss points may be sufficient and thus **Standard Mechanical** is the default as it makes meshing more robust. If you are interested in higher accuracy or your problem is nonlinear, you may want to either set **Error Limits** to **Aggressive Mechanical** or set **Physics Preference** to **Nonlinear Mechanical**. In both cases, the Jacobian ratio that is used is computed at the corner nodes (as it is in MAPDL).

Meshing: Mesh Controls Overview

When in the Mechanical application, your part or multibody part is automatically meshed at solve time. The default element size is determined by the software (for details, see default sizing (p. 99)).

You can also create the mesh before solving. To do this, select **Mesh** from the Tree Outline and right-click your mouse. Mesh controls are available to assist you in fine tuning the mesh to your analysis. These controls are described in the sections that follow.

Mesh control overview topics include:

Global and Local Mesh Controls

If you want more control over the mesh, you can get it by using mesh control tools. Both global mesh controls (p. 93) and local mesh controls (p. 195) are available. Global mesh controls are located in the Details View when the **Mesh** object is selected in the Tree Outline. To access the local mesh control tools, highlight the **Mesh** object, then either click the right mouse button and choose **Insert> {<choose a mesh control tool>}**, or choose a mesh control from the **Mesh Control** context toolbar.

In general, in cases where a numeric value can be specified for an option but a value of 0 would not make sense within the context of the option, specifying 0 resets the option to its default. For example, the **Curvature Normal Angle** (p. 109) option accepts a value from 0 to 180, but since 0 does not make sense within its context, specifying 0 resets **Curvature Normal Angle** to its default.

Understanding the Influence of the Sizing Options

An important aspect of meshing in Workbench is the size function. Depending on the physics preference, the sizing options allow you to control whether you want a uniform mesh or whether the mesh should be influenced by the model's curvature and/or proximity of faces next to each other (thickness of model).

The following chart recommends how to get the appropriate mesh:

	Uniform mesh	Mesh with Curvature Refinement	Mesh with Proximity Refinement	Mesh with 2D Curvature and Proximity Refinement
Physics Preference = Nonlinear Mechanical	Set **Capture Curvature** and **Capture Proximity** to **No**	Set **Capture Curvature** to **Yes**	Set **Capture Proximity** to **Yes**	Not recommended

If using any other physics preference	Set **Use Adaptive Sizing**, **Capture Curvature**, and **Capture Proximity** to **No**	Set **Capture Curvature** to **Yes**	Set **Capture Proximity** to **Yes**	Set both **Capture Curvature** and **Capture Proximity** to **Yes**

Note:

When **Physics Preference** is **Hydrodynamics**, the only properties you can set are Element Size (p. 98) and Defeature Size (p. 106).

Adaptive Sizing

Adaptive Sizing can be thought of as a 2D curvature and proximity-based refinement approach which refines edges based on curvature and/or proximity but does not propagate the refined mesh along the face. It works with **Resolution**, **Span Angle Center** and **Transition**, and has historically been used as a way to reduce the total element count while capturing every edge of the model. This approach can significantly stretch the mesh leading to poor mesh quality, so it is generally recommended to use one of the other sizing options and use defeaturing to reduce the total element count when needed.

When **Use Adaptive Sizing** is set to **Yes**, the mesher uses the value of the element size property to determine a starting point for the mesh size. The value of the element size property can be set by the user or automatically computed using Defaults (p. 99). When meshing begins, edges are meshed with this size initially, and then they are refined for curvature and 2D proximity. Next, mesh based defeaturing and pinch control execution occurs. The final edge mesh is then passed into a least-squares fit size function, which guides face and volume meshing.

Uniform, Curvature, Proximity, or Proximity and Curvature Sizing

A size function is computed when meshing begins. The mesher examines the global and local sizes and uses the smallest, most local size to seed the mesh. It uses the growth rate to transition the mesh from a small size to a larger size.

The following factors contribute to the final mesh distribution obtained by the mesher:

- The value of the Element Size (p. 98) option
- The value of the Curvature Min Size (p. 108)/Proximity Min Size (p. 110) option
- The value of the Max Size (p. 105) option
- The value of the Growth Rate (p. 105) option
- Features of the geometry, which can be any of the following:
  - Edge and face curvature (p. 102), based on the normal angle variation between adjacent mesh elements in 2D (edge) or 3D (face)
  - Edge and face proximity (p. 102), based on the number of element layers created between a gap between edges in 2D or between faces in 3D

- Local element sizing (p. 248) on selected edges, faces, or bodies
- Sphere of influence (p. 256) scoped to a selected body, face, edge, or vertex
- Body of influence (p. 257) scoped to a selected body
- Influence of an existing mesh or a swept (p. 323) body

Note:

- The mesher uses the sizes defined by the user or computes its own Defaults (p. 99).
- The size function works within parts, but not across parts.
- When **Capture Curvature** and/or **Capture Proximity** are set to **Yes**, the default pinch tolerance (p. 188) is 90% of the value of **Curvature Min Size**/**Proximity Min Size** (whichever is smaller). This differs from the tolerance used by the default mesh based defeaturing; refer to Mesh Defeaturing (p. 106) for details.
- The size function computes a background grid prior to meshing and re-uses that background grid during meshing. This background grid could differ for solid parts and sheet parts.
- The size function may over-refine bodies being meshed with sweep or MultiZone as side faces can influence source faces and source faces can influence side faces to try to make the mesh more uniform. Use hard edge sizing controls or set the number of divisions along the sweep path to override such behavior.
- In some situations, the mesh will exceed the applied **Element Size** setting to create a better quality mesh. To set a strict upper limit for each element edge, set the **Element Size** to a size slightly smaller than the desired size.

Overriding Sizing Minimum and Maximum Sizes

One of the most important values related to sizing is minimum size (Curvature Min Size (p. 108) and Proximity Min Size (p. 110) controls). Setting a value for minimum size that is too small may lead to an unnecessarily fine mesh and longer meshing time or robustness issues due to the mesh tolerances being smaller than the geometry tolerances (mesh falling into holes/gaps in the model). On the other hand, setting a value that is too large may mean that important features are not captured by the mesh.

The following notes may help:

- Use local sizing controls to override global controls to either increase or decrease the mesh size as necessary.
- Use the **Mechanical Min Size Factor** and/or **CFD Min Size Factor** to change the relationship between **Element Size** and the default min size. See Sizing (p. 319) for details.

Meshing: Global Mesh Controls

Global mesh controls are located in the Details View when the **Mesh** object is selected in the Tree Outline.

Global settings are categorized into the following groups and are available through drop-down menus:

Defaults Group

Sizing Group

Quality Group

Inflation Group

Assembly Meshing Group of Controls

Batch Connections

Advanced Group

Statistics Group

Defaults Group

The **Defaults** group allows you to control these options:

Physics Preference

Solver Preference

Export Format

Export Unit

Export Preview Surface Mesh

Element Order

Element Size

Physics Preference

The **Physics Preference** option allows you to establish how Workbench will perform meshing based on the physics of the analysis type that you specify. Available options are: **Mechanical**, **Electromagnetics**, **CFD**, **Explicit**, **Nonlinear Mechanical**, and **Hydrodynamics**. The value of the **Physics Preference** option sets the default for various meshing controls, as detailed in the following table.

Note:

- When **Physics Preference** is **CFD**, **Solver Preference** is **Fluent** or **Polyflow**, and an assembly meshing algorithm is being used, some defaults will differ from those that appear in the table. Refer to Assembly Meshing (p. 367) for details.

- The solver options for the **Mechanical** physics preference appear within the Mechanical application only, and only under certain circumstances. See Solver Preference (p. 95) for more information.

- When **Physics Preference** is **Hydrodynamics**, the only properties you can set are Element Size (p. 98) and Defeature Size (p. 106). The meshing controls that appear in the table below are not applicable to the **Hydrodynamics** physics preference; therefore, **Hydrodynamics** is not included in the table. Refer to Overview of the Meshing Process for Hydrodynamics Analysis (p. 30) for related information.

- Refer to the Meshing Overview (p. 19) section for further details about physics based meshing.

Meshing Control	Physics Preference							
	Mechanical		Nonlinear Mechanical	Electromagnetics	CFD			Explicit
	Mechanical APDL Solver (p. 95)	Rigid Body Dynamics Solver (p. 95)			CFX-Solver (p. 95)	Fluent Solver (p. 95)	Polyflow Solver (p. 95)	
Element Order (p. 96)	**Program Controlled**	**Linear (Read-only)**	**Program Controlled**	**Quadratic**	**Linear**	**Linear**	**Linear**	**Linear**
Straight Sided Elements (p. 176)	**No**	**N/A**	**No**	**Yes**	**N/A**	**N/A**	**N/A**	**N/A**
Element Size (p. 98)	**Default**	**Default**	**Default**	**Default**	**Default**	**Default**	**Default**	**Default**
Sizing Options (p. 100)	**Capture Curvature** set to **Yes** for shell models; otherwise **Use Adaptive Sizing** set to **Yes**	**Capture Curvature** set to **Yes** for shell models; otherwise **Use Adaptive Sizing** set to **Yes**	**Capture Curvature** set to **Yes**	**Use Adaptive Sizing** set to **Yes**	**Capture Curvature** set to **Yes**	**Capture Curvature** set to **Yes**	**Capture Curvature** set to **Yes**	**Capture Curvature** set to **Yes** for shell models; otherwise **Use Adaptive Sizing** set to **Yes**
Transition (p. 107) / Growth Rate (p. 105)	**Fast** / 1.85	**Fast** / 1.85	**N/A** / 1.5	**Fast** / 1.85	**N/A** / 1.2	**N/A** / 1.2	**N/A** / 1.2	**Slow** / 1.2
Span Angle Center (p. 107) /	**Coarse** / 70.395°	**Coarse** / 70.395°	**N/A** / 60°	**Coarse** / 70.395°	**N/A** / 18°	**N/A** / 18°	**N/A** / 18°	**Coarse** / 70.395°

Meshing Control	Physics Preference							
	Mechanical		Nonlinear Mechanical	Electromagnetics	CFD			Explicit
	Mechanical APDL Solver (p. 95)	Rigid Body Dynamics Solver (p. 95)			CFX-Solver (p. 95)	Fluent Solver (p. 95)	Polyflow Solver (p. 95)	
Curvature Normal Angle (p. 109)								
Error Limits (p. 118)	Standard Mechanical	Standard Mechanical	Nonlinear Mechanical	Electromagnetics	CFD	CFD	CFD	Explicit
Smoothing (p. 123)	Medium	Medium	N/A	Medium	Medium	Medium	Medium	High
Inflation Algorithm (p. 154)	Pre	Pre	Pre	Pre	Pre	Pre	Pre	Pre
Collision Avoidance (p. 158)	Stair Stepping	Stair Stepping	Stair Stepping	Stair Stepping	Stair Stepping	Layer Compression	Stair Stepping	Stair Stepping
Transition Ratio (p. 152)	0.272	0.272	0.272	0.272	0.77	0.272	0.272	0.272
Rigid Body Behavior (p. 177)	Dimensionally Reduced	Dimensionally Reduced	Dimensionally Reduced	Dimensionally Reduced	Dimensionally Reduced	Dimensionally Reduced	Dimensionally Reduced	Full Mesh

Solver Preference

Choosing **CFD** as your **Physics Preference** causes a **Solver Preference** option to appear in the Details View of the **Mesh** folder. The value of **Solver Preference** can be **CFX**, **Fluent**, or **Polyflow**. Based on the value, the Meshing application sets certain defaults that will result in a mesh that is more favorable to the **CFX**, **Fluent**, or **Polyflow** solver, respectively. Inflation controls with defaults that are affected by the **Solver Preference** setting include:

- **Aspect Ratio (Base/Height)** (p. 154)
- **Collision Avoidance** (p. 158)
- **Transition Ratio** (p. 152)

Within the Mechanical application only, choosing **Mechanical** as your **Physics Preference** also causes a **Solver Preference** option to appear, providing the analysis system is a Transient Structural or Rigid Dynamics system during the initial geometry attach. In such cases, **Solver Preference** can be set to either **Mechanical APDL** or **Rigid Body Dynamics**. Controls with defaults that are affected by the **Solver Preference** setting include **Element Order** (p. 96).

Other differences (such as what happens when you insert a **Method** control) are dependent on the behavior of the geometry bodies and the **Rigid Body Behavior** setting. See Rigid Body Meshing (p. 424) for more information.

Refer to Physics Preference (p. 93) for more information about defaults that are set based on **Physics Preference** and **Solver Preference**.

Export Format

Available only when **Physics Preference** is set to **CFD** and the **Solver Preference** is set to **Fluent**, the **Export Format** option defines the format for the mesh when exported to Ansys Fluent. The default is **Standard**. You can change this to **Large Model Support** to export the mesh as a cell-based Fluent mesh.

Note:

The **Large Model Support** option is not available for the following types of meshes:

- 2D
- Assembly (using the CutCell method)

For more information about exporting meshes to Ansys Fluent, see Fluent Mesh Export (p. 43).

Export Unit

Available only when **Physics Preference** is set to **CFD** and the **Solver Preference** is set to **Polyflow**, the **Export Unit** option defines the unit of measurement for the mesh when exported to Ansys Polyflow. The default is **Use Project Unit**, which means the mesh is not scaled. If you change this to another value (centimeter, millimeter, micrometer, inch, or foot), the mesh is scaled according to the export unit you select.

Export Preview Surface Mesh

Available only when **Physics Preference** is set to **CFD**, the **Solver Preference** is set to **Fluent**, and the **Export Format** is set to **Standard**, the **Export Preview Surface Mesh** option controls the export of the preview surface mesh elements. This option can be used when the bodies have been meshed only partially, that is, not all volumes have been filled with elements and only previewing of surface meshes was done. In such cases, you can choose to export the previewed surface meshes and continue meshing in Fluent Meshing.

The default is **No**, which results in export of only volume mesh elements to the Fluent mesh file. You can change this to **Yes** to export both the volume mesh and the preview surface meshes to the Fluent mesh file.

Element Order

The global **Element Order** option allows you to control whether meshes are to be created with midside nodes (quadratic elements) or without midside nodes (linear elements). Reducing the number of midside nodes reduces the number of degrees of freedom. Choices for the global **Element Order** option include **Program Controlled**, **Linear**, and **Quadratic**.

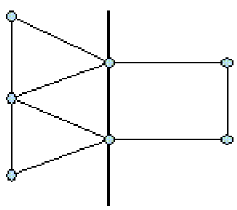
Examples are presented below. The heavy vertical line in each graphic represents the body boundary.

Program Controlled

Program Controlled is the default. For surface bodies and beam models, **Program Controlled** is identical to the **Linear** option described below. For solid bodies and 2-D models, **Program Controlled** is identical to the **Quadratic** option described below.

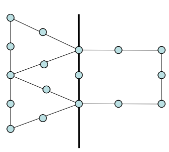
Linear

The **Linear** option removes midside nodes on all elements. Examples shown below are for a solid body.



Quadratic

The **Quadratic** option retains midside nodes on elements created in the part or body. All elements will have midside nodes.



If **Element Order** is set to **Quadratic**, and if **Straight Sided Elements** is set to **No**, the midside nodes will be placed on the geometry so that the mesh elements properly capture the shape of the geometry. However, if the location of a midside node might affect the mesh quality, the midside node may be relaxed to improve the element shape. Therefore, some midside nodes might not follow the shape of the geometry precisely.

For more information about how the **Straight Sided Elements** control affects midside nodes, see Straight Sided Elements (p. 176).

Note:

- Both linear and quadratic elements are supported for 2D axisymmetric models. This includes support for 2D linear and quadratic inflation layers on axisymmetric parts. You must set **Element Order** to **Linear** to obtain linear elements in such cases.

- Both linear and quadratic elements are supported for 2D shell and line models. This includes support for 2D linear and quadratic inflation layers on shell models.

- When setting the Method control (p. 196) to a scoped body, you can change the setting of the **Element Order** option to control whether meshes are to be created on the scoped body with midside nodes or without midside nodes. You do this by using the **Element Order** setting under **Definition** in the Details View. This setting will override the setting of the global **Element Order** option described here. See Method Controls and Element Order Settings (p. 196) for additional information.

Element Size

Element Size allows you to specify the element size used for the entire model. This size will be used for all edge, face, and body meshing.

- For solids, the **Element Size** is calculated as Bounding Box Diagonal (p. 116) * Bounding Box Factor (p. 319).
- For sheets or solids/sheets, the **Element Size** is calculated as Average Surface Area (p. 117) * Surface Area Factor (p. 319).

Sizing Group

Mesh Sizing Defaults (p. 99) describes how default sizes are computed and how the values respond when you modify other sizing values.

The **Sizing** group contents depend on the **Physics Preference** being used.

- When the **Physics Preference** is **Mechanical**, **Electromagnetics**, or **Explicit**, **Use Adaptive Sizing** is set to **Yes** by default.
- When the **Physics Preference** is **Nonlinear Mechanical**, or **CFD**, **Capture Curvature** is set to **Yes** by default.
- When the **Physics Preference** is **Hydrodynamics**, the only properties you can set are Element Size (p. 98) and Defeature Size (p. 106).

With **Use Adaptive Sizing** set to **Yes**, it includes:

- Resolution (p. 104)
- Mesh Defeaturing (p. 106) (Defeature size)
- Transition (p. 107)
- Span Angle Center (p. 107)
- Initial Size Seed (p. 108)
- Enable Washers (p. 113)
- Bounding Box Diagonal (p. 116)
- Average Surface Area (p. 117)

- Minimum Edge Length (p. 117)

With **Use Adaptive Sizing** set to **No**, it includes:

- Growth Rate (p. 105)
- Max Size (p. 105)
- Mesh Defeaturing (p. 106) (Defeature size)
- Capture Curvature (Curvature Min Size (p. 108) and Curvature Normal Angle (p. 109))
- Capture Proximity (Proximity Min Size (p. 110), Num Cells Across Gap (p. 110), and Proximity Size Function Sources (p. 110))
- Enable Washers (p. 113)
- Bounding Box Diagonal (p. 116)
- Average Surface Area (p. 117)
- Minimum Edge Length (p. 117)

Mesh Sizing Defaults

When a model is loaded, the default **Element Size** is automatically set by the software using the physics preference and characteristics about the model.

- If the model only has solid parts, the **Element Size** is set as a factor of the Bounding Box Diagonal (p. 116). When **Use Adaptive Sizing** is set to **Yes**, the factor is determined by using a combination of the Physics Preference (p. 93) and Initial Size Seed (p. 108). For other sizing options, the default **Bounding Box Factor** is set to 0.05, but you can modify this in the **Options** dialog box (**Tools-> Options-> Meshing-> Bounding Box Factor** (p. 319)).
- If the model has sheet parts, the **Element Size** is set as a factor of the average surface area. When **Use Adaptive Sizing** is set to **Yes**, the factor is determined by using a combination of the Physics Preference (p. 93), and Initial Size Seed (p. 108). For other sizing options, the default **Surface Area Factor** is set to 0.125, but you can modify this in the **Options** dialog box (**Tools-> Options-> Meshing-> Surface Area Factor** (p. 319)).

Other default mesh size settings, such as defeature size, curvature minimum size, and proximity minimum size, are set in relation to the **Element Size**. If **Use Adaptive Sizing** is set to **No**, a simple factor is used to scale the defeature size, curvature minimum size, and proximity minimum size.

Beginning in Release 18.2, you can rely on dynamic defaults to scale other sizes off of the element size. When you modify the element size, the other default sizes update dynamically in response, thereby providing a more direct scaling of sizing values.

Dynamic defaults are controlled by the **Mechanical Min Size Factor**, **CFD Min Size Factor**, **Mechanical Defeature Size Factor** and **CFD Defeature Size Factor** options. These options are available in the Options (p. 317) dialog box. You use these options to set your preferences for the scale factors (p. 319) that will be used to calculate the corresponding default sizes. Essentially, the scale factors control the default values for global minimum size and global defeature size, as well as the default sizes used by local mesh sizing controls when **Type** is set to Factor of Global Size (p. 258).

Using Dynamic Mesh Sizing Defaults

To use dynamic defaults:

1. Retain the default settings of **Defeature Size**, **Curvature Min Size**, **Proximity Min Size**, and/or **Max Size**.

   Note:

   If you modify any of these default sizes, their values will not update dynamically. This is because user-defined sizes are always retained.

2. Adjust the **Element Size**.

   In response, the other default global mesh sizes will update dynamically.

3. (Optional) For more control over global size scaling, modify the values of the **Mechanical Min Size Factor**, **CFD Min Size Factor**, **Mechanical Defeature Size Factor** and/or **CFD Defeature Size Factor** options (p. 319) to obtain the desired scaling.

4. (Optional) To control scaling locally, choose Factor of Global Size (p. 258) as the **Type** when you define a local sizing control. Then, depending on the entities and other options you select for the control, use the **Defeature Size Scale**, **Curvature Min Size Scale**, and **Proximity Min Size Scale** local sizing control options (p. 254) to obtain local sizing based on the global sizing.

   Note:

   By setting **Type** to **Factor of Global Size**, you ensure the local sizes will update dynamically if you change the global element size.

Sizing Options

The **Sizing** options provide greater control over the following properties:

- Mesh growth (transition) between small and large sizes based on a specified growth rate
- Curvature based refinement and angles between normals for adjacent mesh elements (curvature (p. 102)-based sizing)
- Number of mesh elements employed in the gaps between two geometric entities (proximity (p. 102)-based sizing)

By default, **Use Adaptive Sizing** is set to **Yes**, unless Physics Preference (p. 93) is set to **CFD** or **Nonlinear Mechanical** (in which case the default is **Capture Curvature** set to **Yes**), or **Hydrodynamics** (in which case the sizing is **Uniform**). The option you choose determines which refinement mechanisms are activated, as described in the following sections.

Controls for sizing include **Element Size** (p. 98), **Growth Rate** (p. 105), **Max Size** (p. 105), and **Curvature Min Size** (p. 108)/ **Proximity Min Size** (p. 110). The **Curvature Min Size**/**Proximity Min Size** and **Max Size** specifications represent, respectively, the global minimum and global maximum allowable element size. The **Element Size** specification represents the global maximum allowable size of the elements created by the free surface meshers of the supported methods. The **Growth Rate** represents the increase in element edge length with each succeeding layer of elements from the edge or face. For example, a growth rate of 1.2 results in a 20% increase in element edge length with each succeeding layer of elements.

Sizing options are applicable to the following mesh methods:

Volume Meshing:

- General Sweep (p. 323) (Surface mesh only)
- Thin Sweep (p. 330) (Surface mesh only)
- Hex Dominant (p. 222) (Surface mesh only)
- Patch Conforming Tetrahedron (p. 200)
- Patch Independent Tetrahedron (p. 200)
- MultiZone (p. 228)
- Assembly meshing algorithms (CutCell and Tetrahedrons) (p. 367)

Note:

- The sizing controls are passed to the Patch Independent Tetrahedron method. That is, the Patch Independent Tetrahedron method does not *use* the same sizing algorithms; rather, this method *interprets* the controls and uses internal algorithms to respect the user-defined settings.
- The General Sweep, Thin Sweep, and Hex Dominant mesh methods use the sizing controls in the creation of the surface mesh, but the volume mesh does not use them.

Surface Meshing:

- Quad Dominant (p. 245)
- All Triangles (p. 246)

- MultiZone Quad/Tri (p. 246)

Note:

For an overview of how using the sizing influences mesh distribution, refer to Understanding the Influence of the Sizing Options (p. 89).

Sizing topics include:

- Curvature-Based Sizing (p. 102)
- Proximity-Based Sizing (p. 102)
- Uniform Sizing (p. 103)

Curvature-Based Sizing

The mesher examines curvature on edges and faces and computes element sizes on these entities such that the size will not violate the maximum size or the curvature normal angle, which are either automatically computed by the mesher or defined by the user.

Curvature-based sizing is defined by the following properties:

- **Curvature Normal Angle** (p. 109)
- **Curvature Min Size** (p. 108)
- **Element Size** (p. 98)
- **Max Size** (p. 105)
- **Growth Rate** (p. 105)

The **Curvature Normal Angle** (p. 109) is the maximum allowable angle that one element edge is allowed to span given a certain geometry curvature.

Proximity-Based Sizing

You can specify the minimum number of element layers created in regions that constitute "gaps" in the model for proximity-based sizing. For the purposes of specifying proximity-based sizing, a "gap" is defined in one of two ways:

- The internal volumetric region between two faces
- The area between two opposing boundary edges of a face

Proximity-based sizing is defined by the following properties:

- **Proximity Min Size** (p. 110)
- **Num Cells Across Gap** (p. 110)
- **Proximity Size Function Sources** (p. 110)

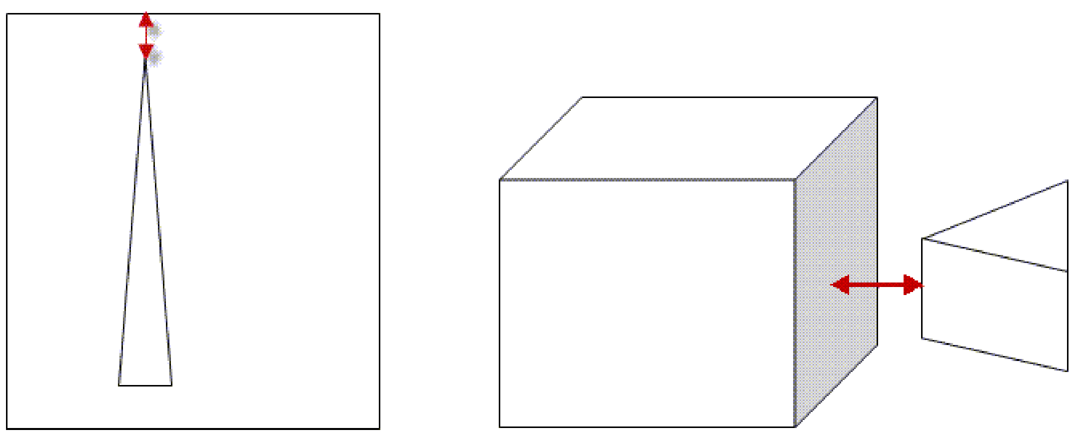
- **Element Size** (p. 98)
- **Max Size** (p. 105)
- **Growth Rate** (p. 105)

The **Num Cells Across Gap** (p. 110) is the number of layers of elements to be generated in the gaps. **Proximity Size Function Sources** (p. 110) determines which regions of proximity are considered in calculations.

Note:

The proximity sizing does not recognize the proximity between certain entities in cases involving voids in a model. For example, the proximity between a vertex and an edge on a face in 2D (below left) is ignored if the triangle is a void in the rectangle. Similarly, the proximity between a vertex or edge and a face in 3D (below right) is ignored if the prism and the block are voids in a larger domain. The two-headed arrows in the figure indicate the areas where proximity is ignored.

Figure 30: Proximity Sizing Limitation



Uniform Sizing

Uniform sizing does not refine the mesh based on curvature or proximity. Rather, you specify defeature and maximum sizes and gradation is provided between sizes based on a specified growth rate.

Uniform sizing is defined by the following properties:

- **Defeature Size** (p. 106)
- **Element Size** (p. 98)
- **Max Size** (p. 105)
- **Growth Rate** (p. 105)

With uniform sizing, you must use size controls to control mesh sizes locally, as opposed to the curvature and proximity sizing, which refine the sizes locally based on curvature and proximity of features in the geometry. Even if the specified local sizes are **Soft** (p. 262) sizes, they may override the global sizes when the uniform sizing is being used.

> **Note:**
>
> The **Use Uniform Size Function for Sheets** (p. 113) enables you to use uniform sizing while you are using other sizing options to refine the mesh for the rest of the model. This setting is available only when the model contains mixed body types (for example, sheet and solid bodies), and when **Capture Curvature** and/or **Capture Proximity** is set to **Yes**.

Setting Sizing Options

By default, **Use Adaptive Sizing** is set to **Yes**, unless Physics Preference (p. 93) is set to **CFD** or **Nonlinear Mechanical** (in which case the default is **Capture Curvature** set to **Yes**), or **Hydrodynamics** (in which case the sizing is uniform). The option you choose determines which refinement mechanisms are activated.

The following options are available:

Resolution

Available when **Use Adaptive Sizing** is set to **Yes**, the **Resolution** option controls the mesh distribution. The default setting is **Program Controlled** (see **Adaptive Resolution** in the **Options** dialog box (p. 319)). The range of values that can be set is 0 to 7, with the mesh resolution changing from coarse (**0**) to fine (**7**).

A value of **-1** will set the **Resolution** to the default value set for **Adaptive Resolution** in the **Options** dialog box (p. 319).

- If **Adaptive Resolution** is set to **Program Controlled**, the default value is **4** when the Physics Preference (p. 93) is **Electromagnetics** or **Explicit**. For all other Physics Preference (p. 93), the default value is **2**.
- If **Adaptive Resolution** is explicitly set in the **Options** dialog box (p. 319) (**0**-**7**), the value set will be used as default.

Note:

- If you set the **Resolution** explicitly, the set value will be retained.
- For databases from previous releases (prior to Release 2021 R2), the **Resolution** will be explicitly set to a value which is **not** the default.
- For new databases, if **Resolution** is set to **Default** and the **Adaptive Resolution** in the **Options** dialog box (p. 319) is different than it was when the database was saved, the new value will be used instead.

Growth Rate

Growth Rate represents the increase in element edge length with each succeeding layer of elements. For example, a growth rate of 1.2 results in a 20% increase in element edge length with each succeeding layer of elements. This option is available when **Use Adaptive Sizing** is set to **No**. Specify a value from 1 to 5 or accept the default.

The default is calculated based on the Physics Preference (p. 93) option and the presence of sheet bodies.

Note:

For sheet models, the Details View does not display the default value of **Growth Rate**.

Max Size

Max Size is the maximum size that the sizing controls will return to the mesher. This option is available when **Use Adaptive Sizing** is set to **No**. Specify a value greater than 0 or accept the default.

Note:

- The **Max Size** option is hidden if there are no solids in your model.
- For information about overriding sizes, refer to Overriding Sizing Minimum and Maximum Sizes (p. 91).
- For information about the relationship between the **Element Size** control and the **Min Size**, and **Max Size** controls, refer to Understanding the Influence of the Sizing Options (p. 89).

Mesh Defeaturing

The Meshing application automatically defeatures small features and dirty geometry according to the **Defeature Size** you specify here.

Mesh defeaturing is supported for the following mesh methods:

Solid Meshing:

- Patch Conforming Tetrahedron (p. 200)
- Patch Independent Tetrahedron (p. 200)
- MultiZone (p. 228)
- Thin Sweep (p. 330)
- Hex Dominant (p. 222)

Surface Meshing:

- Quad Dominant (p. 245)
- All Triangles (p. 246)
- MultiZone Quad/Tri (p. 246)

For the Patch Independent Tetrahedron (p. 200), MultiZone (p. 228), and MultiZone Quad/Tri (p. 246) methods, the **Defeature Size** you set here will be populated to the local (scoped) method controls. If you subsequently make changes to the local settings, the local settings will override the global **Defeature Size** set here. See the descriptions of the individual methods for more information.

See Protecting Topology Defined Prior to Meshing (p. 180) for details on protecting defeatured topology.

The options for defining mesh defeaturing are described below.

Mesh Defeaturing

Use the **Mesh Defeaturing** control to enable and disable defeaturing. When **Mesh Defeaturing** is **Yes** (default), features smaller than or equal to the value of **Defeature Size** are removed automatically.

Defeature Size

Defeature Size is available only when **Mesh Defeaturing** is set to **Yes**. Specify a positive value to set the global tolerance for defeaturing, or accept the default. Specifying a value of **0.0** resets the **Defeature Size** to its default, which is determined by the **Mechanical Defeature Size Factor** or **CFD Defeature Size Factor** preference (p. 319) specified in the Options dialog box. If you retain the default **Defeature Size** and subsequently modify the **Element Size** (p. 98), the default **Defeature Size** is recalculated to reflect the size factor preferences.

Note:

- If a user-defined value has been specified for **Defeature Size**, that value will be used for everything—sheets and solids. In addition, the value will not update dynamically if you modify the **Element Size** because user-defined values are always retained.
- If you allow the **Defeature Size** to default and you modify the mesh size by applying local sizing controls (for example, local face or edge sizing), the tolerance may be modified automatically. In such cases, a warning message is issued to notify you that the global defeature size has been modified. To prevent this behavior, manually specify a value for **Defeature Size**.
- Some bodies or parts might be ignored if the **Defeature Size** is too high. If this happens, the body will be considered to be meshed but will have no elements. These bodies or parts can be suppressed if desired. To capture these bodies or parts in the mesh, regenerate the mesh with a reduced **Defeature Size**.
- If the geometry contains close vertices, setting the **Defeature Size** can resolve them.

  On the **Graphics Options** toolbar, you can click the **Close Vertices** button to determine if there are any close vertices in the model, and then set the **Defeature Size** to a value that will sufficiently defeature any extra vertices.
- You can also set the **Defeature Size** on a body, face, or edge by inserting a local size control (p. 252).

Transition

When **Use Adaptive Sizing** is set to **Yes**, **Transition** affects the rate at which adjacent elements will grow. **Slow** produces smooth transitions while **Fast** produces more abrupt transitions.

Span Angle Center

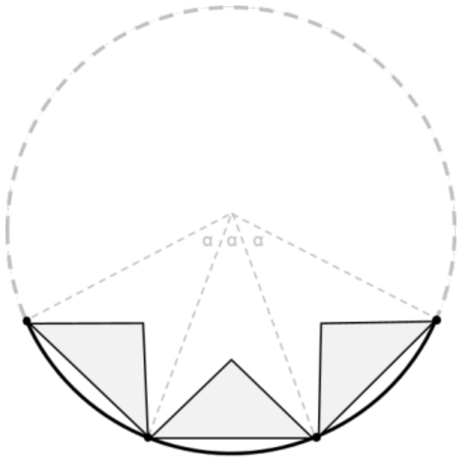
When **Use Adaptive Sizing** is set to **Yes**, **Span Angle Center** sets the goal for curvature-based refinement. For a curved region, the mesh will subdivide along the curvature until the individual elements span this angle.

You can set the span angle by choosing one of the following options:

- **Coarse** - 91^{o} to 60^{o}
- **Medium** - 75^{o} to 24^{o}
- **Fine** - 36^{o} to 12^{o}

Example 1: How the Span Angle Center Controls Element Placement

The following figure shows a curved region. If the curvature were extended to 360^o to create a circle, the Curvature Normal Angle is the arc span from the center of the circle to the curved region. The Curvature Normal Angle (α = 45^o) is computed based on the Span Angle Center setting.



Initial Size Seed

When **Use Adaptive Sizing** is set to **Yes**, **Initial Size Seed** allows you to control the initial seeding of the mesh size for each part.

- **Assembly** (default) bases the initial seeding on the diagonal of the bounding box that encloses all assembly parts regardless of the number of suppressed parts. As a result, the mesh never changes due to part suppression.
- **Part** bases the initial seeding on the diagonal of the bounding box that encloses each particular individual part as it is meshed. The mesh never changes due to part suppression. This option typically leads to a finer mesh and is recommended for situations where the fineness of an individual part mesh is important relative to the overall size of the part in the assembly.

Note:

When **Use Adaptive Sizing** is set to **No**, **Assembly** is set internally and cannot be modified.

Curvature Min Size

Available when **Capture Curvature** (p. 102) is set to **Yes**, **Curvature Min Size** is the minimum size returned to the mesher. Some element sizes may be smaller than this based on local feature sizes or other geometric anomalies.

Specify a positive value or accept the default, which is determined by the size factor preferences (p. 319) (**Mechanical Min Size Factor** and/or **CFD Min Size Factor**) specified in the Options dialog box. If you retain the default **Curvature Min Size** and subsequently modify the **Element Size** (p. 98), the default **Curvature Min Size** is recalculated to reflect the size factor preferences.

Note:

- The appropriate size factor is based on the physics preference. The **Mechanical Min Size Factor** applies when physics preference is **Mechanical**, **Electromagnetics**, or **Explicit**. The **CFD Min Size Factor** applies when physics preference is **CFD**.
- When **Capture Proximity** (p. 102) is also set to **Yes**, you can specify a global **Proximity Min Size** to be used in proximity calculations in addition to the global **Curvature Min Size**. When only **Capture Proximity** is selected, only **Proximity Min Size** is available. Refer to **Proximity Min Size** (p. 110) for details.
- For information about overriding sizes, refer to Overriding Sizing Minimum and Maximum Sizes (p. 91).
- For information about the relationship between the **Element Size** control and the **Max Size** controls, refer to Understanding the Influence of the Sizing Options (p. 89).

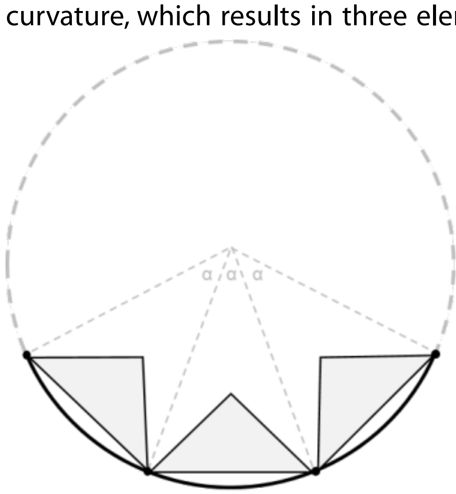
Curvature Normal Angle

Curvature Normal Angle is the maximum allowable angle that one element edge is allowed to span given a particular geometry curvature. This option is available only when **Capture Curvature** is set to **Yes**. You can specify a value from 0 to 180 degrees or accept the default. (A value of 0 resets the option to its default.) The default value depends on the Physics Preference (p. 93) selected.

You can use the **Curvature Normal Angle** to limit the number of elements that are generated along a curve or surface if the Curvature Min Size (p. 108) is too small for that particular curve.

Example 2: How the Curvature Normal Angle Controls Element Placement

The following figure shows a curved region with a Curvature Normal Angle of $\alpha = 45^{\circ}$. If the curvature were extended to 360° to create a circle, the Curvature Normal Angle is the arc span from the center of the circle to the curved region. Three arc spans are needed to span the entire curvature, which results in three elements placed along the curvature: one in each arc span.



Proximity Min Size

Available when **Capture Proximity** (p. 102) is set to **Yes**, **Proximity Min Size** enables you to specify a global minimum size to be used in proximity sizing calculations. Specify a positive value or accept the default, which is set equal to the default of **Curvature Min Size**.

Any feature that operates based on minimum element size will be based on the smaller of the two minimum size values. Affected features include: Tessellation Refinement (p. 166), Pinch Tolerance (p. 188), Defeature Size (p. 106), Finding Thin Sections (p. 393), Finding Contacts (p. 395), and contact sizing (p. 398) when used with assembly meshing.

When only **Capture Proximity** is selected, only **Proximity Min Size** is available.

Some element sizes may be smaller than this based on local feature sizes or other geometric anomalies.

Num Cells Across Gap

Num Cells Across Gap is the minimum number of layers of elements to be generated in the gaps. This option is available only when **Capture Proximity** is set to **Yes**. You can specify a value from 1 to 100, or accept the default (3).

Remember the following information:

- The value of **Num Cells Across Gap** is an estimate; it may not be exactly satisfied in every gap. When using mapped Face Meshing controls (p. 265) or sweeping (p. 323), interval assignment may change the number of divisions (elements or cells) in a gap.

- The value of **Num Cells Across Gap** is approximate for assembly (p. 367) meshing algorithms. For example, if you specify 3 cells per gap on a narrow face, the final mesh may contain anything between 2-4 cells across the gap, depending on the orientation in relation to the global X, Y, Z axis.

- In cases involving Patch Conforming tetra (p. 200) meshing and Swept (p. 323) meshing, the proximity sizing drives the surface mesh size distribution as follows. The value of **Num Cells Across Gap** is applicable to both 3D proximity (the number of 3D elements/cells between two faces in a body) and 2D proximity (the number of 2D elements/cells between two edges on a face), and the global **Growth Rate** (p. 105) value is automatically taken into account in the gap. However, the 3D proximity sizing affects only the surface mesh in the gap, and assumes the volume mesh will use the global settings. Hence, if you define local mesh sizing on a body and specify local **Element Size** (p. 255) or local **Growth Rate** (p. 261) settings that differ drastically from the global sizing settings (or if inflation is specified), the final number of cells across a 3D gap may deviate from the specified **Num Cells Across Gap** value.

- When **Method** (p. 196) is set to Automatic (p. 199), the proximity calculation in swept bodies may result in an under-refined mesh. You can use inflation (p. 291) layers or a scoped fixed size at such locations to produce a sufficient number of elements/cells.

Proximity Size Function Sources

Proximity Size Function Sources determines whether regions of proximity between faces and/or edges are considered when proximity size function calculations are performed. This option is available only when **Capture Proximity** is set to **Yes**. You can specify **Edges**, **Faces**, or **Faces and Edges**:

- **Edges** - This is the default when an assembly meshing algorithm (p. 367) is selected. Considers edge-edge proximity. Face-face and face-edge proximity are not considered.
- **Faces** - Considers face-face proximity between faces. Face-edge and edge-edge proximity are not considered (that is, the trailing edge of fluid around wings will not be captured with this setting).
- **Faces and Edges** - This is the default for part/body-based meshing methods. Considers face-face and edge-edge proximity. Face-edge proximity is not considered.

Note:

- In cases involving face-face proximity, the face normal orientation is ignored during the proximity calculation.
- In cases involving edge-edge proximity, edges across voids in a model are refined with assembly meshing because the volume exists at the time the refinement occurs. With part/body-based meshing methods, these edges are not refined.
- It is important to resolve all edges as much as possible for better feature capturing and for minimizing the occurrence of non-manifold nodes. For this reason, you should specify a setting of either **Edges** or **Faces and Edges**. For many models, the **Edges** setting may be sufficient to resolve all proximity situations. For large complex models, using either the **Faces and Edges** or **Faces** setting may result in longer computation time.

The figures below illustrate the effect of each **Proximity Size Function Sources** setting. For all three meshes, the **Tetrahedrons** assembly meshing algorithm was used.

Figure 31: Proximity Size Function Sources = Edges

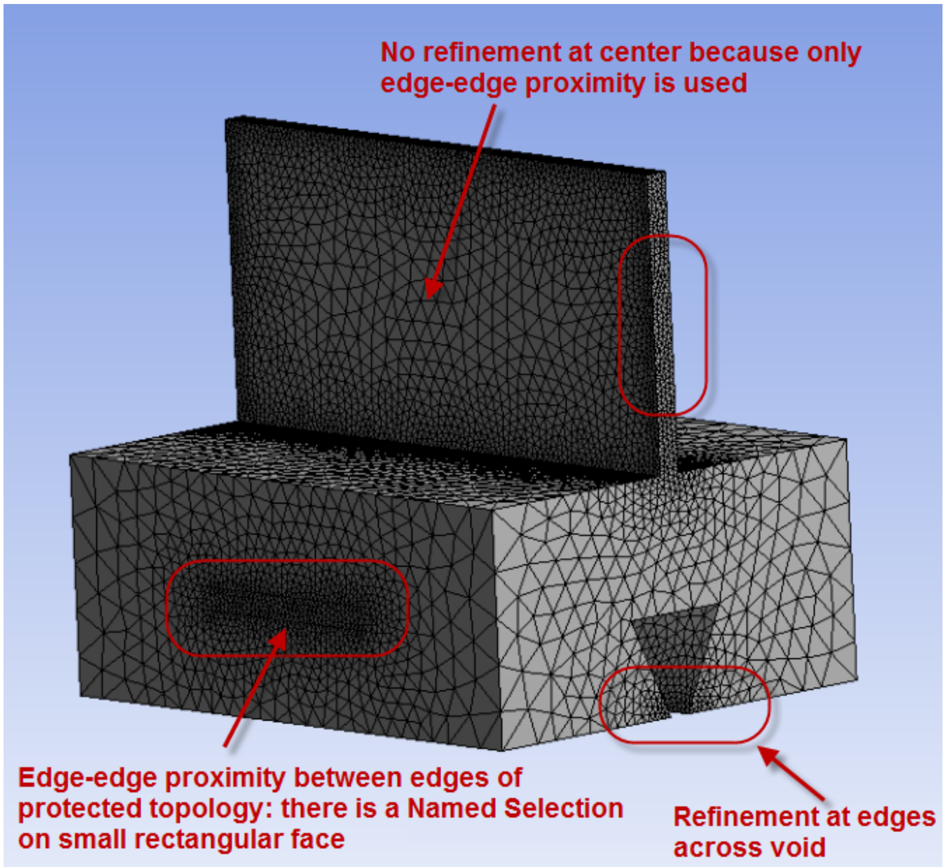


Figure 32: Proximity Size Function Sources = Faces

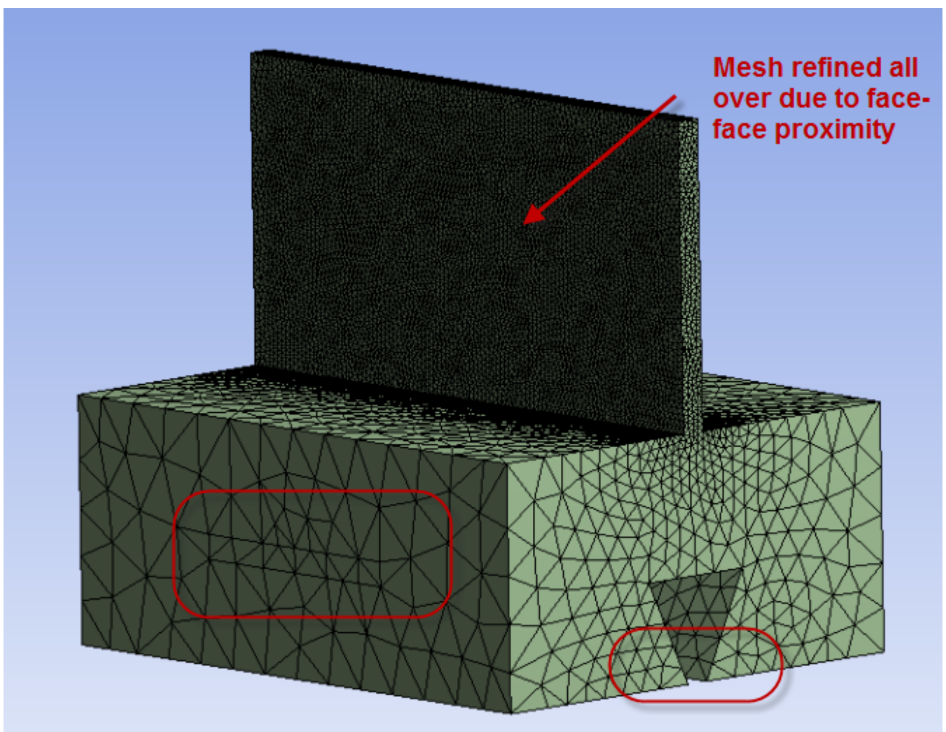
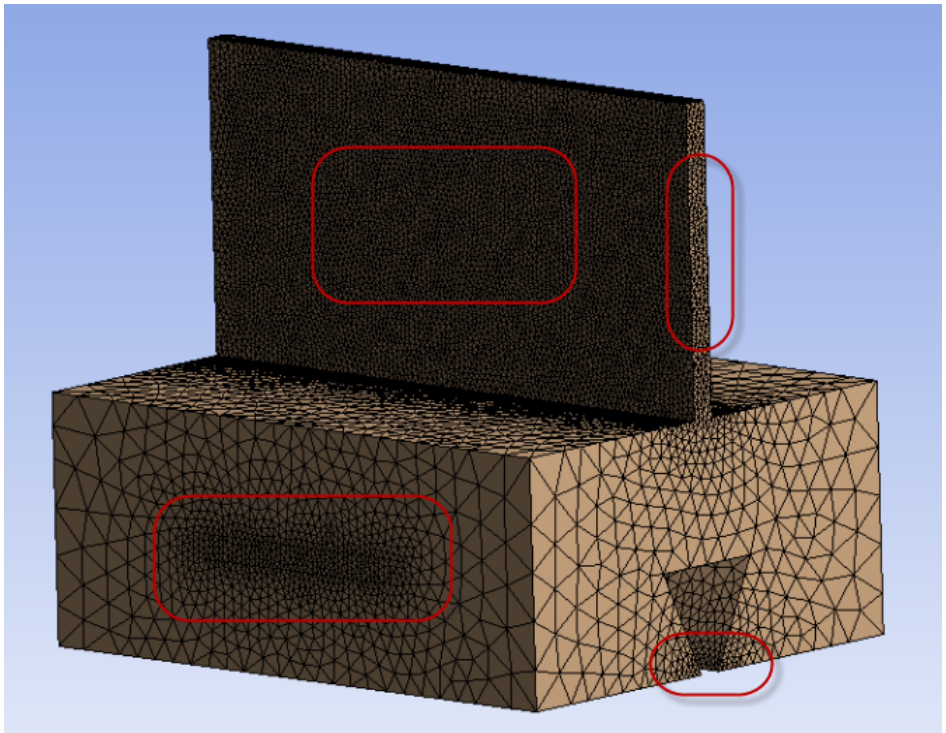


Figure 33: Proximity Size Function Sources = Faces and Edges



Use Uniform Size Function for Sheets

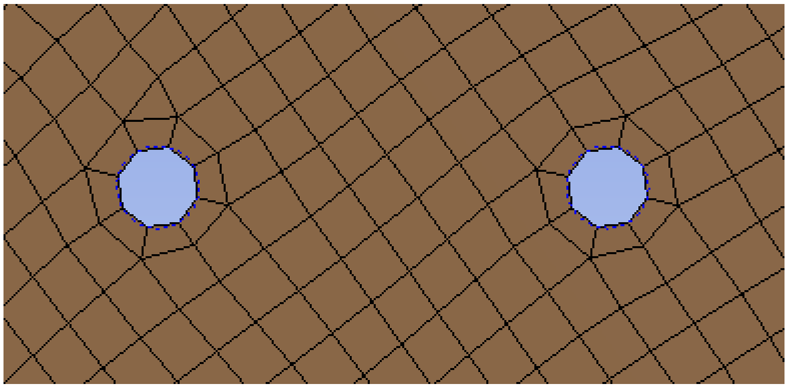
Use Uniform Size Function for Sheets enables you to use the **Uniform** sizing while you are using other options to refine the mesh for the rest of the model. This setting is available only when the model contains mixed body types (for example, sheet and solid bodies), and when **Capture Curvature** and/or **Capture Proximity** is set to **Yes**. The default for this setting is **No**.

Enable Washers

Enable Washers controls the pattern of the mesh that is generated around any holes in a sheet body. If you set this option to **Yes**, the mesh will be generated with a layer of equally-spaced quadratic elements—called washers—around each hole.

The **Enable Washers** option is only available for sheet bodies when **Use Adaptive Sizing**, **Capture Curvature**, and **Capture Proximity** are all set to **No**. If the model contains both solid and sheet bodies, **Enable Washers** is available, but washers will only be generated for the sheet bodies.

Figure 34: Washers Generated Around Two Holes



If you enable washers, you can specify the height of the washers and whether the washer element nodes should be moved for holes that are close to a boundary. You should be aware of the scenarios in which a washer might not be generated for a hole.

Height of Washer

Height of Washer controls the height of the washer elements to be generated around each hole. By default, the height is set to the **Defeature Size** (p. 106).

You should ensure that the height of the washer is set to a value between the target minimum element size in the model and the **Element Size** (p. 98) (the nominal uniform element size expected in the overall model).

Note:

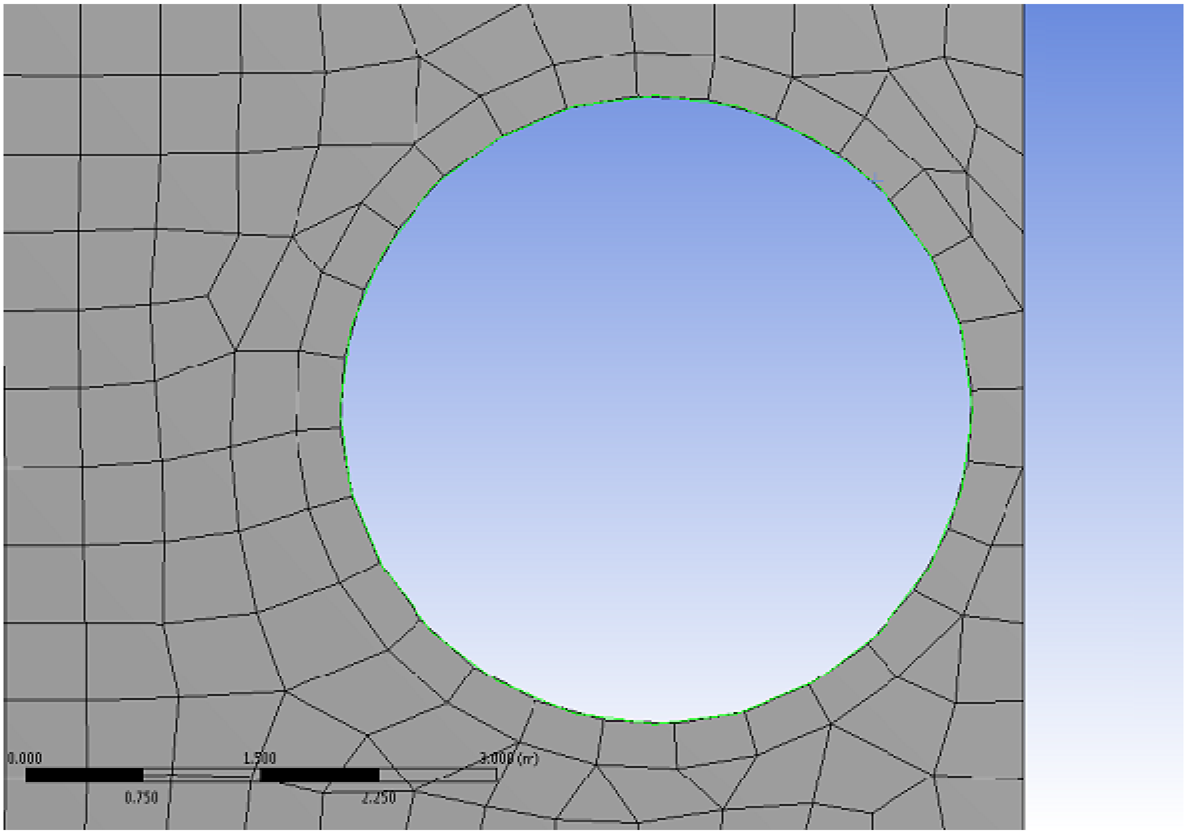
The mesher attempts to respect the minimum size and **Element Size** when the mesh is generated. However, you may notice variations in the final mesh due to local feature sizes, transition areas, or other geometric anomalies.

Allow Nodes to be Moved off Boundary

Allow Nodes to be Moved off Boundary controls the placement of the nodes for holes that are too close to a boundary to place the washer elements. If you select **Yes**, the nodes will be moved such that the washer elements can be placed without compromising any boundaries. The default value for **Tolerance for Moving Nodes** is 10% of the value specified for **Height of Washer**. If you select **No**, the mesher will not generate any washer elements in any locations in which the elements do not fit.

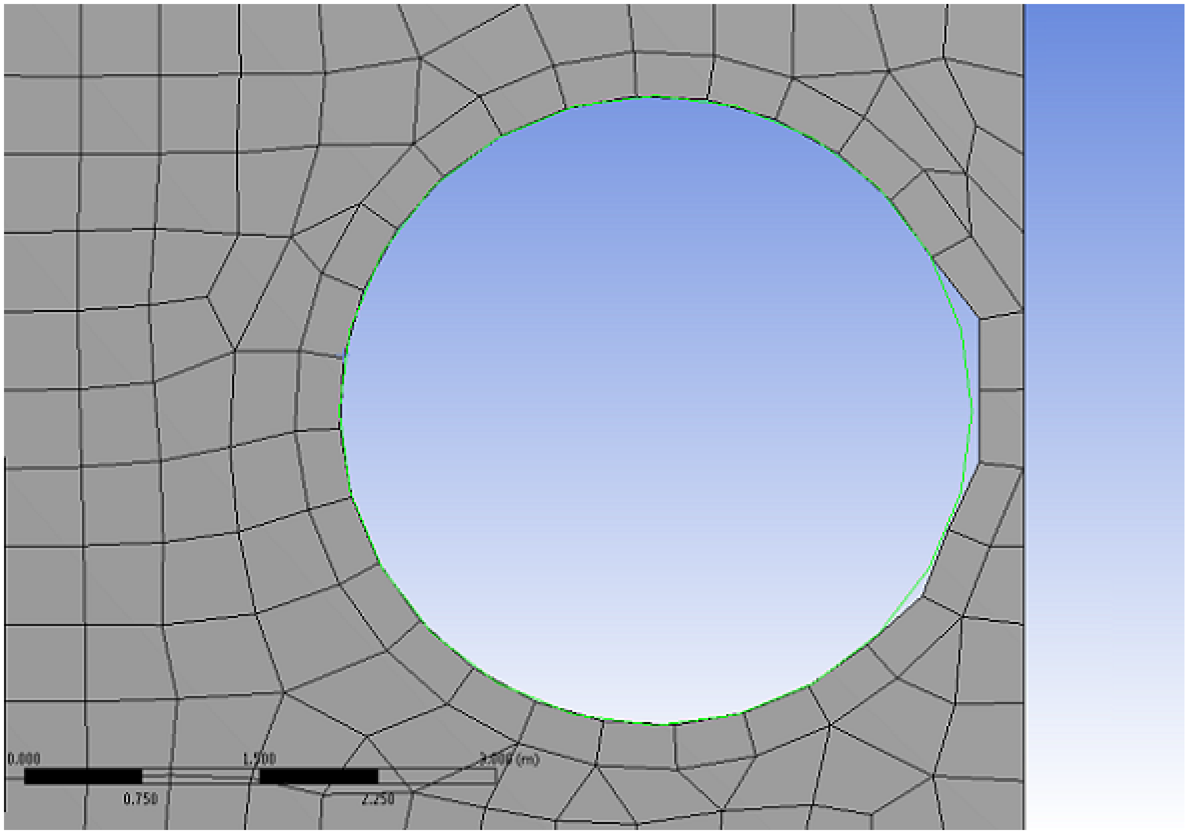
The following figure shows a hole that is too close to an edge for the washer elements to be generated between the hole and the edge.

Figure 35: Washer Element Nodes Not Moved



After setting **Allow Nodes to be Moved off Boundary** to **Yes**, the washer elements are now generated by slightly moving some of the nodes away from the edge of the hole.

Figure 36: Washer Element Nodes Moved



Limitations for Washers

The following table describes the scenarios in which a washer might not be generated for a hole:

Table 1: Washer Limitations

Scenario	Description
A hole is shared by two faces	The hole is not detected, and a washer is not generated around the hole. To ensure that a washer is generated, you should merge the faces (p. 502).
A local mesh control (p. 195) is set	Washers are not generated on any face to which a local mesh control has been applied.
The **Defeature Size** (p. 106) is greater than the **Minimum Edge Length**	Washers might not be generated.
A hole has a diameter less than the **Minimum Edge Length**	A washer will not be generated for the hole.
The **Min Size** and **Element Size** (p. 98) are set to the same value	Washers might not be generated.

Bounding Box Diagonal

Bounding Box Diagonal provides a read-only indication of the length of the assembly diagonal.

Average Surface Area

Average Surface Area provides a read-only indication of the average surface area of the model.

Minimum Edge Length

Minimum Edge Length provides a read-only indication of the smallest edge length in the model.

Quality Group

The Mesh Quality Workflow (p. 117) describes the steps for configuring mesh quality. The **Quality** group includes:

The Mesh Quality Workflow

The general workflow for configuring mesh quality is as follows:

1. Set Physics Preference (p. 93).
2. (Optional) If **Physics Preference** is **Mechanical**, set Error Limits (p. 118).
3. Set Check Mesh Quality (p. 118) depending on how you want the mesher to respond when elements reach error or warning limits.
4. Set appropriate quality targets based on your physics preference:
   - Target Quality (p. 121)
   - Target Skewness (p. 122)
   - Target Jacobian Ratio (Corner Nodes) (p. 122)
5. Generate the mesh. (p. 486)
6. Review warning or error messages. Use Show Elements (p. 495) if needed.
7. Review mesh metrics (p. 123).
8. Perform troubleshooting (p. 535) if needed.

Check Mesh Quality

Check Mesh Quality determines how the software behaves with respect to error and warning limits (p. 118):

- **Yes, Errors** (default) - If the meshing algorithm cannot generate a mesh that passes all error limits, an error message is printed and meshing fails.
- **Yes, Errors and Warnings** - If the meshing algorithm cannot generate a mesh that passes all error limits, an error message is printed and meshing fails. In addition, if the meshing algorithm cannot generate a mesh that passes all warning (target) limits, a warning message (p. 495) is printed.
- **No** - Mesh quality checks are done at various stages of the meshing process (for example, after surface meshing prior to volume meshing). The **No** setting turns off most quality checks, but some minimal checking is still done. In addition, even with the **No** setting, the target quality metrics are still used to improve the mesh. The **No** setting is intended for troubleshooting (p. 535) and should be used with caution as it could lead to solver failures or incorrect solution results.

Note:

Changing the **Check Mesh Quality** setting after you have meshed affects the mesh status as follows:

- If, after meshing, you change the **Check Mesh Quality** setting from **No** to another setting, or you change the setting from **Yes, Errors** to **Yes, Errors and Warnings**, the mesh becomes out-of-date and requires action.
- If, after meshing, you change the **Check Mesh Quality** setting from **Yes, Errors and Warnings** to another setting, or you change the setting from **Yes, Errors** to **No**, the mesh will not go out-of-date. However, some old messages may not directly apply anymore.

Error and Warning Limits

During mesh generation, an element's quality (which is sometimes called the element shape) is calculated. The meshing algorithm uses error limits to obtain a valid mesh. It performs extra mesh cleanup to ensure the error limits are met, such that a valid mesh is one that satisfies the necessary (minimum) conditions and can be consumed by the solver. The meshing algorithm then attempts to improve quality based on warning (target) limits. Error and warning (target) limits can be defined further as:

- An error limit is the value at which an element's quality is not suitable for the solver being used, and by default the mesh would fail (see Check Mesh Quality (p. 118)). Priority is given to ensure there are no elements below the error limits. Error limits are determined by the physics preference as shown in the table below. You cannot change error limits. However, if you are using the **Mechanical** physics preference, you can choose from two sets of error limits: **Standard Mechanical** or **Aggressive Mechanical**. You can think of the error limits as the minimum quality criteria of the meshing.
- A warning (target) limit has two purposes:

1. It is used as a warning limit. If the mesh has elements that are questionable for the solver being used, these elements can be flagged for you via the warning limit. To configure this warning behavior, set Check Mesh Quality (p. 118) to **Yes, Errors and Warnings**.

2. It is used as a target limit. The mesh methods will first try to improve the mesh to ensure there are no elements that do not pass the error limits. If successful, the mesh methods do further improvements to try to meet the target limits. You can think of the target limits as the quality goals for the meshing. If the mesher cannot meet the goals, a warning can be issued. To configure this warning behavior, set Check Mesh Quality (p. 118) to **Yes, Errors and Warnings**.

Note:

All mesh methods use the warning (target) limits to flag warning elements if Check Mesh Quality (p. 118) is set to **Yes, Errors and Warnings**. However, not all mesh methods use the target limits to improve the mesh. Currently, only the Patch Conforming Tetra (p. 200) mesh method uses the target limits to improve the mesh.

You can control how the mesher responds when it reaches error and warning limits by setting the **Check Mesh Quality** (p. 118) option.

The error limits for each physics preference are:

- **Mechanical** - Uses either of these sets of error limits, depending on the setting of the **Error Limits** option:

  – **Standard Mechanical** - These error limits have proven to be effective for linear, modal, stress, and thermal problems.

  – **Aggressive Mechanical** - These error limits are more restrictive than the error limits for **Standard Mechanical**. **Aggressive Mechanical** may produce more elements, fail more often, and take longer to mesh. You can use these error limits by setting **Error Limits** to **Aggressive Mechanical**. As an alternative, you can set **Physics Preference** to **Nonlinear Mechanical**. However, doing so changes other defaults and may significantly change the mesh size and/or which features the mesh is capturing, and therefore may have a big impact on mesh quality. This is the default when **Physics Preference** is set to **Mechanical**, but you can use the **Error Limits** option to change it.

- **Nonlinear Mechanical** - Uses error limits as shown in the table below to produce a high quality mesh that meets the shape checking requirements of tetrahedral elements for nonlinear analysis. If the element quality cannot meet the error limits, the mesh is not desirable for nonlinear analysis. These error limits are used whenever **Physics Preference** is set to **Nonlinear Mechanical**; you cannot change them.

Note:

Using the **Nonlinear Mechanical** option typically produces more elements and longer meshing times. If the element size is too coarse, meshing robustness may be problematic because it is sometimes difficult to get a good quality mesh that not only meets the coarse element size but also captures the features of the model. In such cases, you should reduce the element size, simplify the model, or set Check Mesh Quality (p. 118) to **No** to turn off the error checks.

- **Electromagnetics** - Uses error limits based on element volume, face warping, and face angle. These error limits are used whenever **Physics Preference** is set to **Electromagnetics**; you cannot change them.
- **CFD** - For non-assembly meshing algorithms, uses error limits based on element volume. For assembly (p. 367) meshing algorithms, uses error limits based on orthogonal quality (p. 142). These error limits are used whenever **Physics Preference** is set to **CFD**; you cannot change them.
- **Explicit** - Uses error limits based on Jacobian ratio and element volume. These error limits are used whenever **Physics Preference** is set to **Explicit**; you cannot change them.

The following table presents the error and warning (target) limits for different values of Physics Preference (p. 93). The **Hydrodynamics** physics preference does not have error and warning limits, so it is not included in the table.

Physics Preference	**Mechanical**			**Nonlinear Mechanical**		**Electromagnetics**		**CFD**		**Explicit**	
Criterion	**Standard Mechanical Error Limit**	**Aggressive Mechanical Error Limit**	**Warning (Target) Limit**	**Error Limit**	**Warning (Target) Limit**	**Error Limit**	**Warning (Target) Limit**	**Error Limit**	**Warning (Target) Limit**	**Error Limit**	**Warning (Target) Limit**
Element Quality (p. 130)	< 5 x 10^{-6} for 3D < 0.01 for 2D < 0.75 for 1D	< 5 x 10^{-4} for 3D < 0.02 for 2D < 0.85 for 1D	< 0.05 (default)	< 5 x 10^{-4} for 3D < 0.02 for 2D < 0.85 for 1D	N/A	N/A	< 0.05 (default)	N/A	N/A	N/A	< 0.05 (default)
Jacobian Ratio (p. 132) (Gauss Points)	< 0.025	N/A	N/A	< 0.025	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Jacobian Ratio (p. 132) (Corner Nodes)	N/A	< 0.025	N/A	< 0.001	< 0.04 (default)	N/A	N/A	N/A	N/A	< 0.001	N/A

Physics Preference	Mechanical			Nonlinear Mechanical		Electromagnetics		CFD		Explicit	
Skewness (p. 140)	N/A	N/A	N/A	N/A	> 0.9 (default)	N/A	N/A	N/A	> 0.9 (default)	N/A	N/A
Orthogonal Quality (p. 142)	N/A	N/A	N/A	N/A	N/A	N/A	N/A	<= 0 for assembly meshing, not used for other methods	<= 0.05 for assembly meshing, not used for other methods	N/A	N/A
Element Volume	< 0	< 0	N/A	< 0	N/A	< 10^{-30}	N/A	< 10^{-32}, not used for assembly meshing	N/A	< 0	N/A
Aspect Ratio (for triangles (p. 130) and quadrilaterals (p. 131))	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Face Angle	N/A	N/A	N/A	N/A	N/A	> 150	N/A	N/A	N/A	N/A	N/A
Face Warping (p. 136)	N/A	N/A	N/A	N/A	N/A	> 0.4	N/A	N/A	N/A	N/A	N/A

Target Quality

The **Target Quality** global option allows you to set a target element quality (p. 130) that you would like the mesh to satisfy.

The target quality value drives improvements to tetrahedral elements. If you set the target quality and the mesh contains tetrahedral elements, the mesher will attempt to improve the tetrahedral elements to meet the target quality that you specified. If the target quality cannot be met, a valid mesh may still be generated. In addition, if Check Mesh Quality (p. 118) is set to **Yes, Errors and Warnings**, a warning message is displayed to help you address the issues preventing the mesh from satisfying the target quality. You can right-click the Message field and select the Show Elements (p. 495) option from the context menu to create Named Selections for the elements that don't meet the target.

You should set the target quality if you intend to run a simulation that is sensitive to mesh quality. However, because setting the target quality increases memory usage and the time required to generate the mesh, you should not set the quality any higher than necessary.

To set the **Target Quality**, enter a value between **0** (lower quality) and **1** (higher quality). The default is 0.05.

Note:

- **Target Quality** is supported for the Patch Conforming Tetra (p. 200) mesh method only.
- The Adaptive sizing can result in coarse mesh sizes with stretched elements that cannot be improved with a higher target quality value. Therefore, if you are using the Adaptive sizing, you should set the **Target Quality** to a value < 0.1. Alternatively, you could use a different sizing option (such as Curvature (p. 102)).

Target Skewness

The **Target Skewness** global option allows you to set a target skewness (p. 140) that you would like the mesh to satisfy.

The target skewness value drives improvements to tetrahedral elements. If you set the target skewness and the mesh contains tetrahedral elements, the mesher will attempt to improve the tetrahedral elements to meet the target skewness that you specified. If the target skewness cannot be met, a valid mesh may still be generated. In addition, if Check Mesh Quality (p. 118) is set to **Yes, Errors and Warnings**, a warning message is displayed to help you address the issues preventing the mesh from satisfying the target skewness. You can right-click the Message field and select the Show Elements (p. 495) option from the context menu to create Named Selections for the elements that don't meet the target.

You should set the target skewness if you intend to run a simulation that is sensitive to mesh quality. However, because setting the target skewness increases memory usage and the time required to generate the mesh, you should not set the skewness any lower than necessary.

To set the **Target Skewness**, enter a value between **0** (higher quality) and **1** (lower quality). The default is 0.9. For a tetrahedral mesh, you should not set **Target Skewness** to a value < 0.8.

Note:

Target Skewness is supported for the Patch Conforming Tetra (p. 200) mesh method only.

Target Jacobian Ratio (Corner Nodes)

The **Target Jacobian Ratio (Corner Nodes)** global option allows you to set a target Jacobian ratio (p. 132) that you would like the mesh to satisfy.

The target Jacobian ratio value drives improvements to tetrahedral elements. If you set the target Jacobian ratio and the mesh contains tetrahedral elements, the mesher will attempt to improve the tetrahedral elements to meet the target Jacobian ratio that you specified. If the target Jacobian ratio cannot be met, a valid mesh may still be generated. In addition, if Check Mesh Quality (p. 118) is set to **Yes, Errors and Warnings**, a warning message is displayed to help you address the issues preventing the mesh from satisfying the target Jacobian ratio. You can right-click the Message field and select the Show Elements (p. 495) option from the context menu to create Named Selections for the elements that don't meet the target.

You should set the target Jacobian ratio if you intend to run a simulation that is sensitive to mesh quality. However, because setting the target Jacobian ratio increases memory usage and the time required to generate the mesh, you should not set the Jacobian ratio any higher than necessary.

To set the **Target Jacobian Ratio (Corner Nodes)**, enter a value between **0** (lower quality) and **1** (higher quality). The default is 0.04.

Note:

Target Jacobian Ratio (Corner Nodes) is supported for the Patch Conforming Tetra (p. 200) mesh method only.

Smoothing

Smoothing attempts to improve element quality by moving locations of nodes with respect to surrounding nodes and elements. The **Low**, **Medium**, or **High** option controls the number of smoothing iterations along with the threshold metric where the mesher will start smoothing.

Note:

- When **Smoothing** is set to **High**, additional smoothing of inflation layers occurs. This may slow down the prism generation process.
- For details about **Smoothing** settings and their effects on assembly meshing (p. 367), refer to Setting Sizing Options (p. 390).

Mesh Metric

The **Mesh Metric** option allows you to view mesh metric information and thereby evaluate the mesh quality. Once you have generated a mesh, you can choose to view information about any of the following mesh metrics: Element Quality (p. 130), **Aspect Ratio** for triangles (p. 130) or quadrilaterals (p. 131), Jacobian Ratio (p. 132) (MAPDL, corner nodes, or Gauss points), Warping Factor (p. 136), Parallel Deviation (p. 138), Maximum Corner Angle (p. 139), Skewness (p. 140), Orthogonal Quality (p. 142), and Characteristic Length (p. 144). Selecting **None** turns off mesh metric viewing.

When you select a mesh metric, its **Min**, **Max**, **Average**, and **Standard Deviation** values are reported in the Details View, and a bar graph is displayed under the **Geometry** window. The graph is labeled with color-coded bars for each element shape represented in the model's mesh, and can be manipulated to view specific mesh statistics of interest (p. 124).

Note:

If the model contains multiple parts or bodies, you can view the mesh metric information for an individual part or body. To do so, return to the Tree Outline. Under the **Geometry** object, click the specific part or body of interest. The Details view displays the **Nodes**,

Elements, **Min**, **Max**, **Average**, and **Standard Deviation** values for the selected metric and part/body under **Statistics**. (The graph is not available at the part/body level.)

Accessing Mesh Metric Information

To access mesh metric information:

1. Generate the mesh. You can view mesh metric information for any mesh that was successfully generated using the Generate Mesh (p. 486), Preview Surface Mesh (p. 489), or Preview Inflation (p. 492) feature.
2. Click the **Mesh** object in the Tree Outline.
3. In the Details View, expand the **Quality** folder.
4. For the **Mesh Metric** control, select the metric of interest from the drop-down menu.

   By default, the **Min**, **Max**, **Average**, and **Standard Deviation** values for the selected metric are reported in the Details View. In addition, a bar graph (p. 124) is displayed under the **Geometry** window.

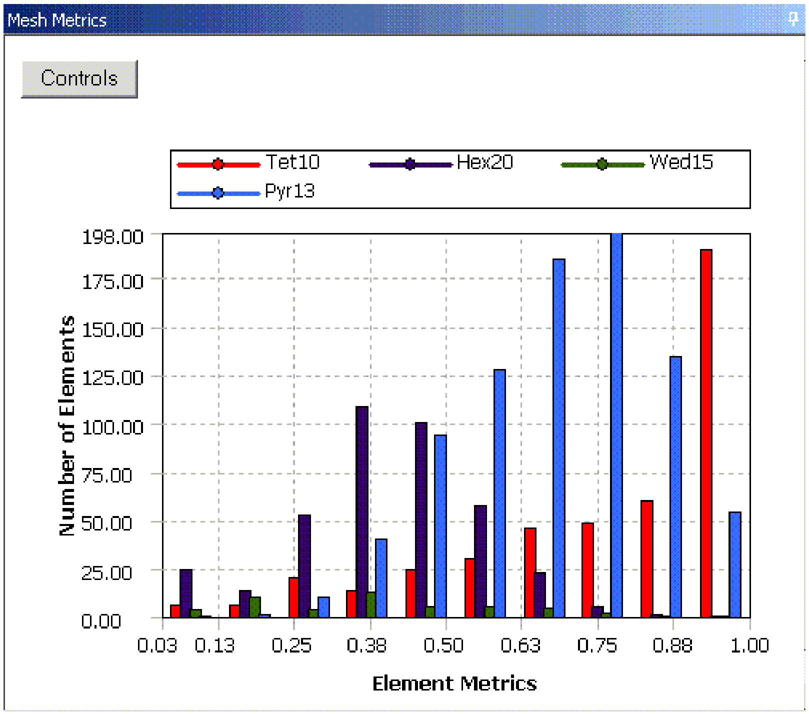
Note:

To view the numbers of **Nodes** and **Elements** in the meshed model, expand the Statistics (p. 193) folder in the Details View.

Viewing Advanced Mesh Statistics

When you select a mesh metric, a bar graph is displayed as shown in Figure 37: Mesh Metrics Bar Graph (p. 125). For this illustration, the **Element Quality** mesh metric was selected in the Details View, so the bar graph displays the minimum to maximum **Element Quality** values over the entire mesh.

Figure 37: Mesh Metrics Bar Graph



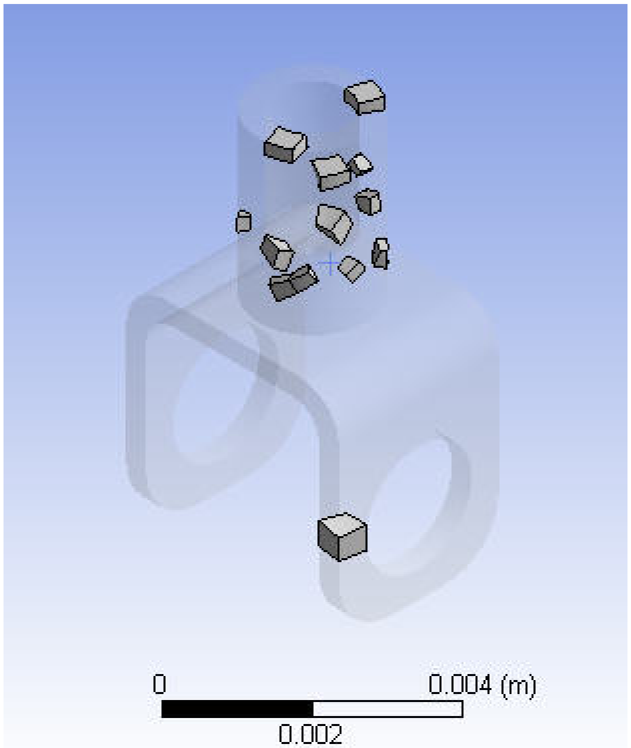
In Figure 37: Mesh Metrics Bar Graph (p. 125), the X-axis represents the value of the selected mesh metric. Using the **Y-Axis Option** setting described in Using the Bar Graph Controls (p. 128), you control whether the Y-axis represents the number of elements within a particular quality factor range (the default), or the percentage of the total volume represented by the elements within a particular quality factor range. In Figure 37: Mesh Metrics Bar Graph (p. 125), the Y-axis represents the number of elements. The alternative would be for the Y-axis to represent the percentage of the total volume. Remember that a model could have a large number of poorly shaped elements that are confined to a small local area. The total volume of these elements might not be significant compared to the volume of the entire model. As a result, the bar corresponding to this low quality factor may not be significant. The **Mesh Metric** option displays the selected mesh metric without qualifying the elements for acceptability.

Additional characteristics of the bar graph include:

- The graph is displayed only when a mesh metric is selected. If you set **Mesh Metric** to **None**, the graph is not displayed. Alternatively, you can click the **Metric Graph** button on the toolbar to hide/show the graph.
- Resuming a model retains the last-saved state of the graph.
- Clicking the **Controls** button accesses the graph controls described in Using the Bar Graph Controls (p. 128).
- The location of an individual bar along the X-axis is the mid-point of the range of metric values covered by that bar.
- Clicking an individual bar on the graph (or in the column of white space above the bar) changes the view in the **Geometry** window. The geometry becomes transparent and only those elements meeting the criteria values corresponding to the selected bar are displayed, as shown in Fig-

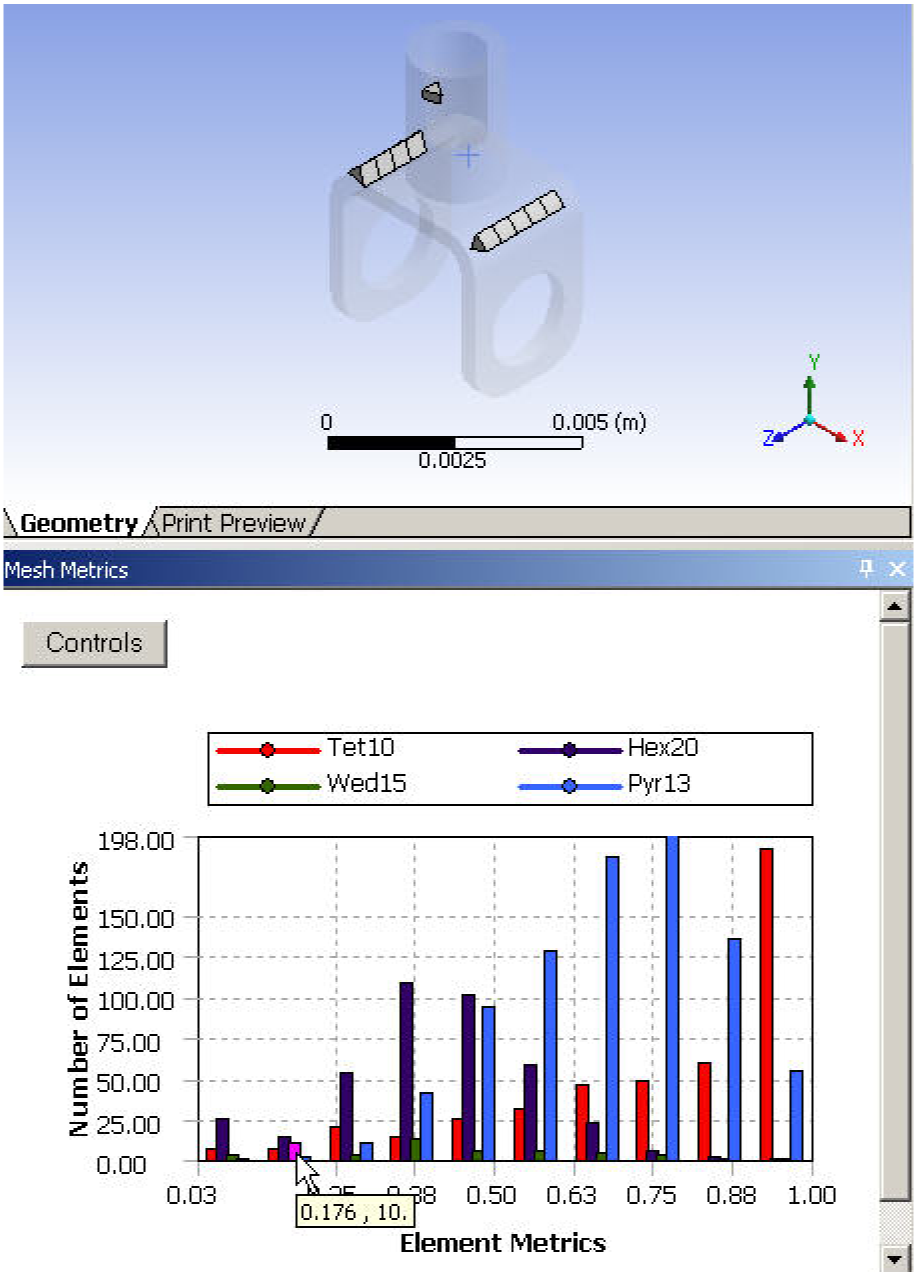
ure 38: Geometry View After Selecting an Individual Bar (p. 126). (The option to click in the column above the bar is helpful if the graph contains very short bars that are difficult to click.)

Figure 38: Geometry View After Selecting an Individual Bar



- If you click and hold the cursor on an individual bar or column, you see a tooltip showing the metric value associated with the bar, along with either a number of elements or the percent of total volume represented by the elements (depending on the **Y-Axis Option** setting). For example, in Figure 39: Clicking and Holding on an Individual Bar (p. 127), 0.176 is the mid-point of the range of metric values covered by the selected bar, and there are 10 elements with values that fall within that range. The 10 elements are displayed in the **Geometry** window.

Figure 39: Clicking and Holding on an Individual Bar



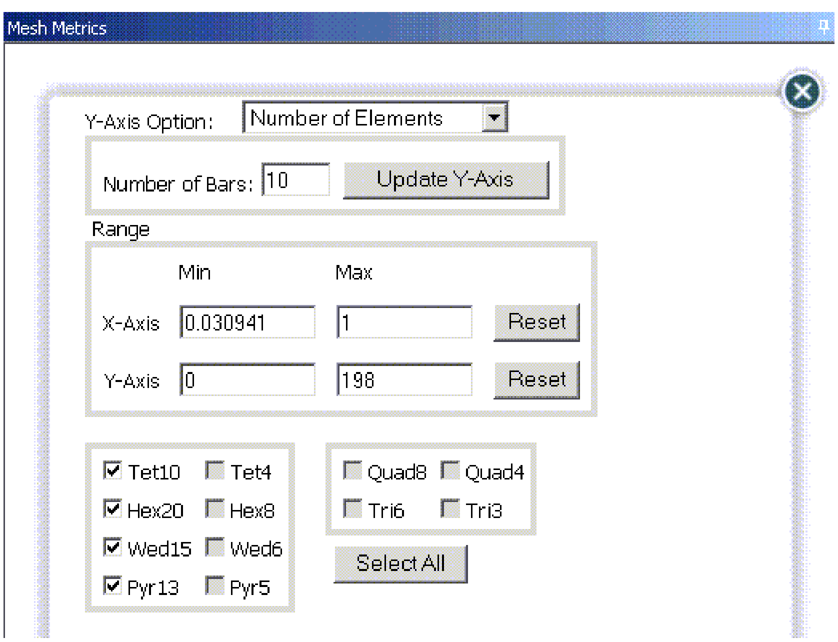
- To select multiple bars, hold the **Ctrl** key and click all desired bars. All elements corresponding to all selected bars are displayed in the **Geometry** window.
- To return the **Geometry** window to the full mesh view (no transparency; all elements are displayed), click on *empty white space* on the graph. Empty white space does not include the column of white space above a bar, as clicking in this area selects the bar and displays only those elements associated with it.
- If you click in a column for which there are 0 elements, all that is displayed in the **Geometry** window is the transparent geometry.

- The graph can be filtered based on element types. See Using the Bar Graph Controls (p. 128) for more information.
- The graph respects section planes and hiding of bodies in the **Geometry** window. For example, if you hide a body and then click an individual bar to view the elements corresponding to the selected bar, elements in the hidden body are not displayed in the **Geometry** window, even if they meet the criteria that the bar represents.
- To zoom the graph, hold the **ALT** key and use your mouse to define a selection box on the graph (click the graph and drag the mouse downward and to the right to define the area to zoom; then release the mouse button). To reset the graph to its initial view, hold the **ALT** key, click the graph and drag the mouse downward and to the left; then release the mouse button.
- The values of the X-axis and Y-axis labels on the graph correspond to the *visible* ranges, rather than to global values. For example, the value 198 in Figure 39: Clicking and Holding on an Individual Bar (p. 127) is the maximum end of the range for the Y-axis, based on the current content of the graph. If you zoom the graph or define a new range of values to display as described in Using the Bar Graph Controls (p. 128), the values of the X-axis and Y-axis labels change accordingly along with the content of the graph.

Using the Bar Graph Controls

When you click the **Controls** button on the graph, the graph is replaced by the controls page as shown in Figure 40: Bar Graph Controls Page (p. 128). Clicking the **X** button applies any changes on the controls page and returns you to the graph.

Figure 40: Bar Graph Controls Page



From the controls page shown in Figure 40: Bar Graph Controls Page (p. 128), you can set the following values:

- **Y-Axis Option** - Determines what the heights of the bars represent. Options include **Number of Elements** and **Percent of Volume/Area**. The default is **Number of Elements**.

- **Number of Bars** - Determines the number of bars to include in the graph. You can enter any whole number greater than or equal to 0. The default is 10. When you click **Update Y-Axis**, the **Min** and/or **Max** values for the **Y-Axis** are recomputed so that the graph and the **Y-Axis** values on the controls page reflect the new number of bars.

- **Range** - Defines a range for the selected metric to display only those elements that fall within the specified range.

  - **X-Axis** - Specify a **Min** and/or **Max** value. To locate and estimate the number of worst elements in the mesh, adjust the **Min** and **Max** values to the lower or upper end of the quality criterion (depending on metric) and click **Update Y-Axis**. (Determining the distribution and location of all the bad elements at one time is helpful in cases where you may need to re-import your model into the DesignModeler application to remove the corresponding problematic geometry.) Click **Reset** to return to the **X-Axis** defaults. (Note: Negative values are acceptable.)

  - **Y-Axis** - Specify a **Min** and/or **Max** value. By lowering the **Max** value, you can clip the Y-axis for easier visualization of small bars, especially as they relate to different element types. Click **Reset** to return to the **Y-Axis** defaults.

- List of element types - Determines which element types to include in the graph. Element types that do not appear in the mesh are read-only on the controls page. Select the element types that you want to include in the graph, or click **Select All** to include all available element types in the graph. By default, all available element types are selected.

Note:

Because the bars approximate the metrics across the range of the **X-Axis**, choosing a very small number of bars over a large **X-Axis** range may move the bars away from the actual average metric of the elements represented by each bar.

Calculation Details

For information about the calculations that are performed for each metric, refer to:

Element Quality

Aspect Ratio Calculation for Triangles

Aspect Ratio Calculation for Quadrilaterals

Jacobian Ratio

Warping Factor

Parallel Deviation

Maximum Corner Angle

Skewness

Orthogonal Quality

Characteristic Length

Element Quality

The **Element Quality** option provides a composite quality metric that ranges between 0 and 1. This metric is based on the ratio of the volume to the sum of the square of the edge lengths for 2D quad/tri elements, or the square root of the cube of the sum of the square of the edge lengths for 3D elements. A value of 1 indicates a perfect cube or square while a value of 0 indicates that the element has a zero or negative volume.

This can also be expressed as follows:

- For two-dimensional quad/tri elements:

$$Quality = C\left(area \Big/ \sum (EdgeLength)^2 \right)$$

> **Note:**
>
> For the **Mechanical** (p. 93) physics preference, if **Error Limit** is set to **Aggressive Mechanical** (p. 118) and the Jacobian Ratio (Corner Nodes) (p. 132) is less than zero, an error occurs.

- For three-dimensional brick elements:

$$Quality = C\left[volume \Big/ \sqrt{\left[\Sigma\left(Edge\,length\right)^2\right]^3} \right]$$

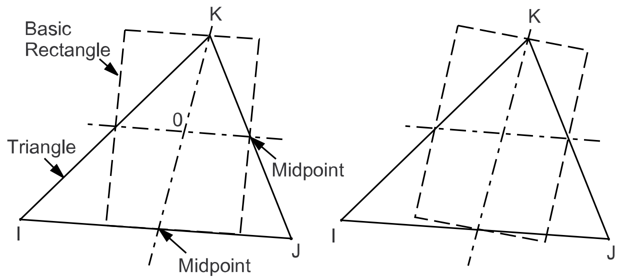
The following table lists the value of C for each type of element:

Element	Value of C
Triangle	6.92820323
Quadrangle	4.0
Tetrahedron	124.70765802
Hexagon	41.56921938
Wedge	62.35382905
Pyramid	96

Aspect Ratio Calculation for Triangles

The aspect ratio for a triangle is computed in the following manner, using only the corner nodes of the element (Figure 41: Triangle Aspect Ratio Calculation (p. 131)):

Figure 41: Triangle Aspect Ratio Calculation



1. A line is constructed from one node of the element to the midpoint of the opposite edge, and another through the midpoints of the other 2 edges. In general, these lines are not perpendicular to each other or to any of the element edges.
2. Rectangles are constructed centered about each of these 2 lines, with edges passing through the element edge midpoints and the triangle apex.
3. These constructions are repeated using each of the other 2 corners as the apex.
4. The aspect ratio of the triangle is the ratio of the longer side to the shorter side of whichever of the 6 rectangles is most stretched, divided by the square root of 3.

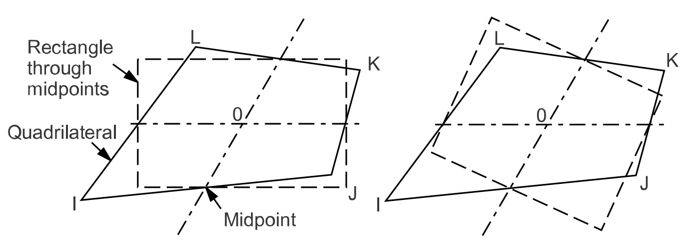
The best possible triangle aspect ratio, for an equilateral triangle, is 1. A triangle having an aspect ratio of 20 is shown in Figure 42: Aspect Ratios for Triangles (p. 131).

Figure 42: Aspect Ratios for Triangles

Aspect Ratio Calculation for Quadrilaterals

The aspect ratio for a quadrilateral is computed by the following steps, using only the corner nodes of the element (Figure 43: Quadrilateral Aspect Ratio Calculation (p. 131)):

Figure 43: Quadrilateral Aspect Ratio Calculation



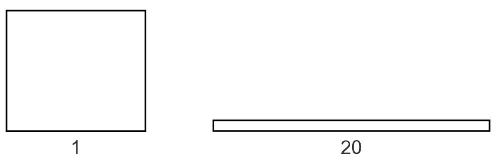
1. If the element is not flat, the nodes are projected onto a plane passing through the average of the corner locations and perpendicular to the average of the corner normals. The remaining steps are performed on these projected locations.

2. Two lines are constructed that bisect the opposing pairs of element edges and which meet at the element center. In general, these lines are not perpendicular to each other or to any of the element edges.

3. Rectangles are constructed centered about each of the 2 lines, with edges passing through the element edge midpoints. The aspect ratio of the quadrilateral is the ratio of a longer side to a shorter side of whichever rectangle is most stretched.

4. The best possible quadrilateral aspect ratio, for a square, is one. A quadrilateral having an aspect ratio of 20 is shown in Figure 44: Aspect Ratios for Quadrilaterals (p. 132).

Figure 44: Aspect Ratios for Quadrilaterals



Jacobian Ratio

The Jacobian ratio is a measurement of the shape of a given element compared to that of an ideal element. If an element has a bad quality Jacobian ratio, the element may not map well from element space to real space, thereby making computations based on the element shape less reliable. The ideal shape of an element depends on element type.

Jacobian ratio is not calculated for triangles and tetrahedra that are linear (no midside nodes) or have straight-sided midside nodes. This is because the shape function for these types of elements is linear, and the partial derivatives of linear functions are constant. Therefore, the determinant of the Jacobian ratio for these types of elements is constant over the entire element, which means $R_J(min) = R_J(max)$. For such elements, the Jacobian ratio is always 1.

Jacobian Ratio Calculation

There are two ways to calculate the Jacobian ratio: either based on corner nodes (nodal points) or based on Gauss points (integration points).

When the Jacobian ratio calculation is based on...	Be aware that...
A sampling of element corner nodes	• The calculation is more restrictive. • To view mesh metric information based on this calculation, you must set Mesh Metric (p. 123) to one of the following: – **Jacobian Ratio (Corner Nodes)**, which is bounded by -1 (worst) and 1 (best) on the mesh metrics bar graph (p. 125). An element with a Jacobian ratio <= 0 should be avoided. – **Jacobian Ratio (MAPDL)**, which is used by the MAPDL solver, is the inverse of **Jacobian Ratio (Corner Nodes)**. **Jacobian Ratio (MAPDL)** is bounded by negative infinity and positive infinity,

	but all negative-value elements are collected and arbitrarily assigned the value -100 for purposes of the mesh metrics bar graph (p. 125). An element with a Jacobian ratio <= 0 should be avoided. A Jacobian ratio whose value is close to 1 is best. **Note:** When **Physics Preference** is set to **Mechanical**, the Error Limits (p. 118) option for shape checking is set to **Standard Mechanical** by default. The Jacobian ratio calculation used for **Standard Mechanical** checks Gauss points but does not check corner nodes. This may lead to a situation where you have an element that has a **Jacobian Ratio (Gauss Points)** that is > 0 and a **Jacobian Ratio (Corner Nodes)** that is <= 0.
A sampling of element Gauss points	• The calculation is less restrictive. • To view mesh metric information based on this calculation, you must set Mesh Metric (p. 123) to **Jacobian Ratio (Gauss Points)**, which is bounded by -1 (worst) and 1 (best) on the mesh metrics bar graph (p. 125). An element with a Jacobian ratio <= 0 should be avoided.

An element's Jacobian ratio is computed by the following steps, using the full set of nodes for the element:

1. Sampling locations are based on the selected **Mesh Metric** option [**Jacobian Ratio (MAPDL)**, **Jacobian Ratio (Corner Nodes)**, or **Jacobian Ratio (Gauss Points)**]. At each sampling location listed in the table below, the determinant of the Jacobian matrix is computed and called R_J. R_J at a given point represents the magnitude of the mapping function between element natural coordinates and real space. In an ideally-shaped element, R_J is relatively constant over the element, and does not change sign.

Element Shape	**R_J Sampling Locations for Jacobian Ratio (MAPDL) and Jacobian Ratio (Corner Nodes)**	**R_J Sampling Locations for Jacobian Ratio (Gauss Points)**
10-node tetrahedra	Corner nodes	Four Gauss quadrature points
5-node or 13-node pyramids	Base corner nodes and near apex node (apex R_J factored so that a pyramid having all edges the same length will produce a Jacobian ratio of 1)	5-node pyramids use 1 Gauss quadrature point 13-node pyramids use 8 Gauss quadrature points
8-node quadrilaterals	Corner nodes and centroid	Four Gauss quadrature points
20-node bricks	All nodes and centroid	Eight Gauss quadrature points

all other elements	Corner nodes	Choose the optimal number of Gauss quadrature points for integration

2. For **Jacobian Ratio (MAPDL)**, the Jacobian ratio of the element is the ratio of the *maximum* to the *minimum* sampled value of R_J, while for **Jacobian Ratio (Corner Nodes)** and **Jacobian Ratio (Gauss Points)**, it is the ratio of the *minimum* to the *maximum*. For **Jacobian Ratio (MAPDL)**, if the maximum and minimum have opposite signs, the Jacobian ratio is arbitrarily assigned to be -100 (and the element is clearly unacceptable).

3. If the element is a midside-node tetrahedron, an additional R_J is computed for a fictitious straight-sided tetrahedron connected to the 4 corner nodes. For **Jacobian Ratio (MAPDL)**, if that R_J differs in sign from any nodal R_J (an extremely rare occurrence), the Jacobian ratio is arbitrarily assigned to be -100. For **Jacobian Ratio (Corner Nodes)** and **Jacobian Ratio (Gauss Points)**, the Jacobian ratio is assigned to be -1.

4. If the element is a line element having a midside node, the Jacobian matrix is not square (because the mapping is from one natural coordinate to 2-D or 3-D space) and has no determinant. For this case, a vector calculation is used to compute a number which behaves like a Jacobian ratio. This calculation has the effect of limiting the arc spanned by a single element to about 106°.

The Jacobian ratio of various element shapes may worsen as follows:

- A triangle or tetrahedron has a Jacobian ratio of 1 if each midside node, if any, is positioned at the average of the corresponding corner node locations. This is true no matter how otherwise distorted the element may be. Hence, this calculation is skipped entirely for such elements. Moving a midside node away from the edge midpoint position will worsen the Jacobian ratio. If the node is moved significantly, the Jacobian ratio will become negative and the element is invalid.

- Any rectangle or rectangular parallelepiped having no midside nodes, or having midside nodes at the midpoints of its edges, has a Jacobian ratio of 1. Moving midside nodes toward or away from each other can worsen the Jacobian ratio. If the node is moved significantly, the Jacobian ratio will become negative and the element is invalid.

- A quadrilateral or brick has a Jacobian ratio of 1 if (a) its opposing faces are all parallel to each other, and (b) each midside node, if any, is positioned at the average of the corresponding corner node locations. As a corner node moves near the center, the Jacobian ratio worsens. If the node is moved significantly, the Jacobian ratio will become negative and the element is invalid.

The figures below illustrate mesh quality by color for **Jacobian Ratio (MAPDL)** and **Jacobian Ratio (Corner Nodes)**.

Note:

To view mesh quality by color, in the Details view, set Display Style to the desired metric. The use of **Display Style** to color mesh according to quality and the use of mesh metrics (p. 123) to view quality statistics are mutually exclusive. For example, when you are viewing mesh metrics, you cannot also view the mesh quality by color.

Figure 45: Jacobian Ratio (MAPDL)

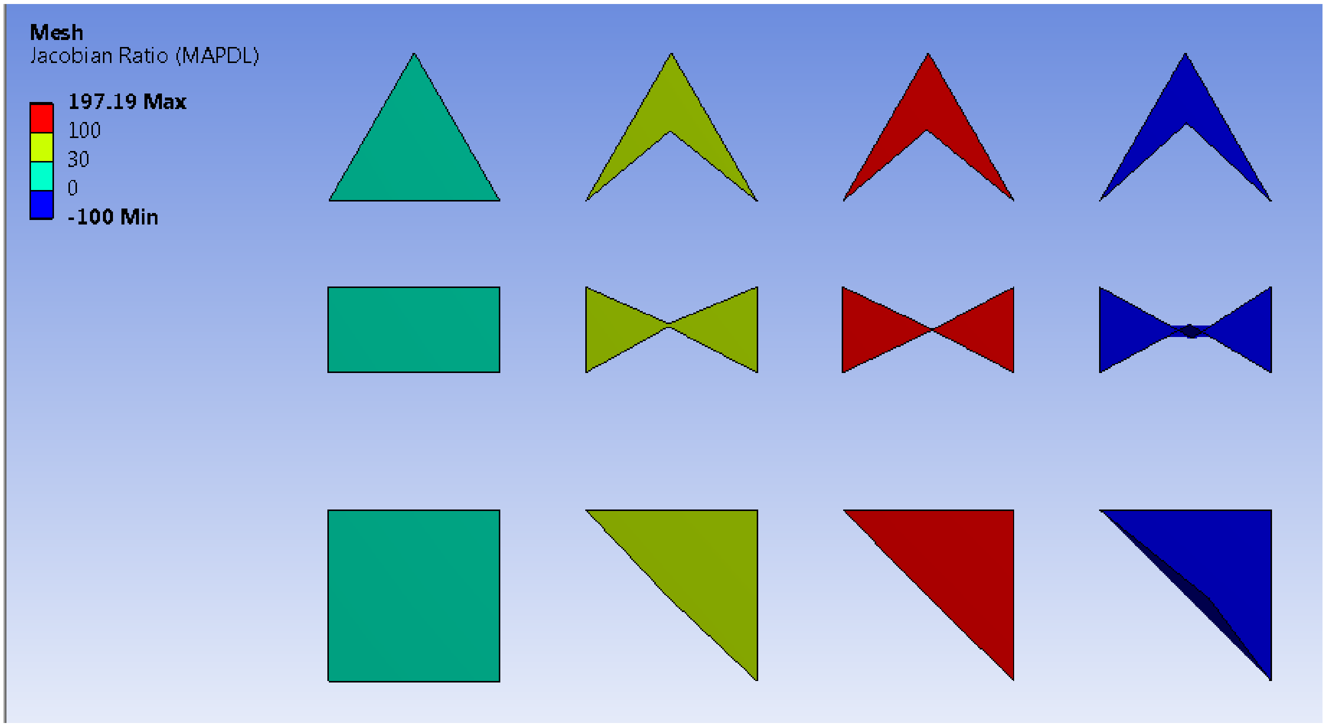
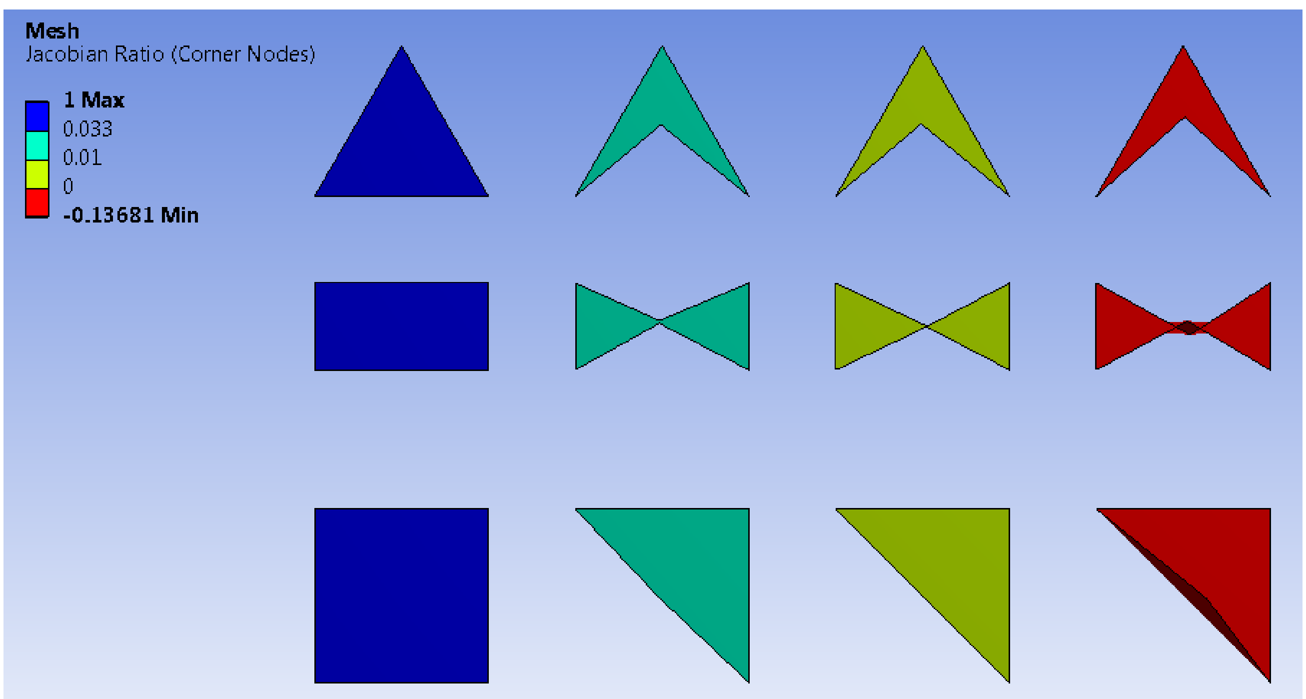
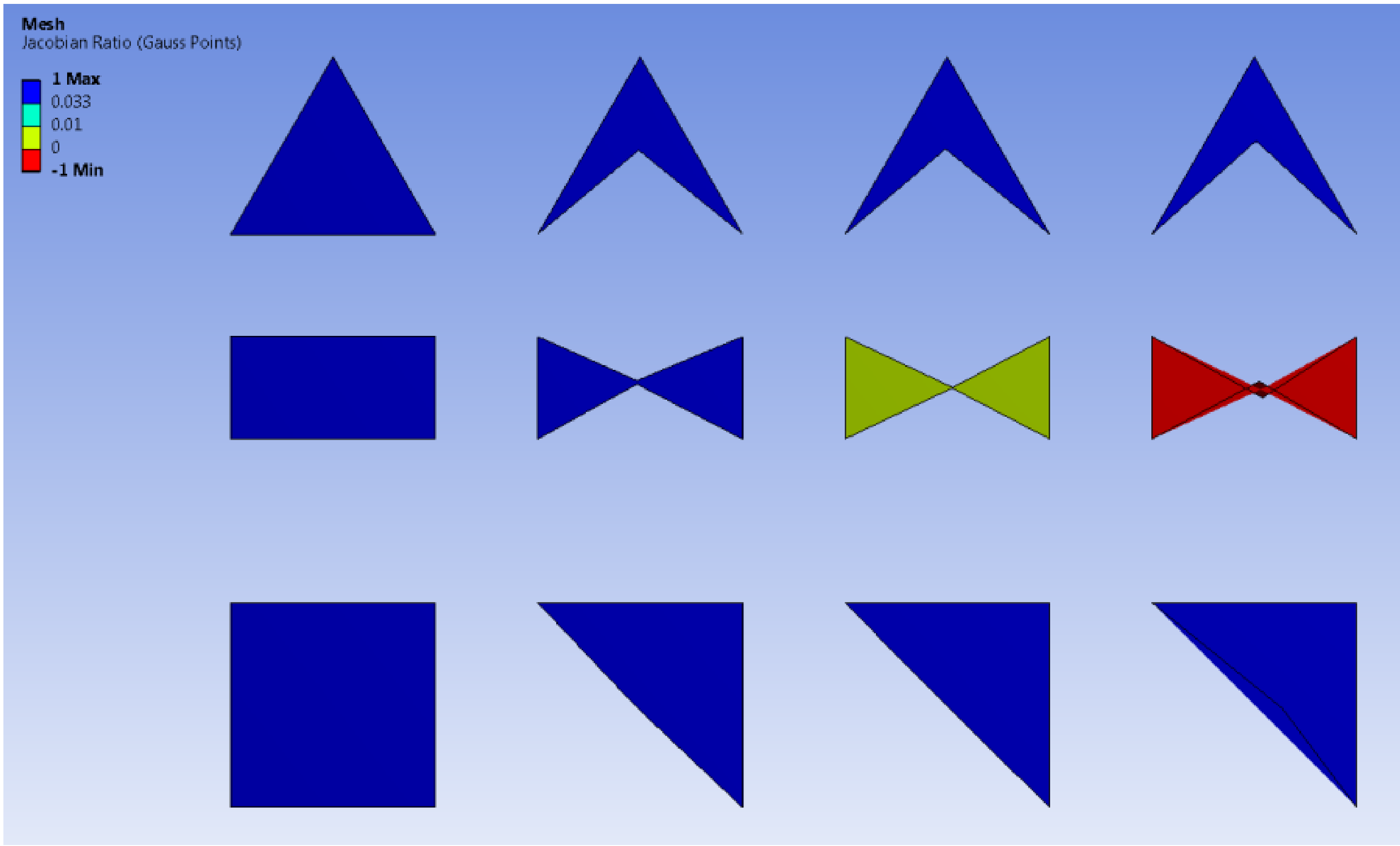


Figure 46: Jacobian Ratio (Corner Nodes)



Jacobian Ratio (Gauss Points) is a good indicator of quality for quadratic tetrahedrons because its formulation is similar to that used by the solver. However, it is not very helpful for shell meshes. For shell meshes, Element Quality (p. 130) is a better indicator of mesh quality. For example, the 2D elements that register as bad in the preceding figures will not necessarily register as bad elements for the **Jacobian Ratio (Gauss Points)** mesh metric, as shown below:

Figure 47: Jacobian Ratio (Gauss Points)



Warping Factor

Warping factor is computed and tested for some quadrilateral shell elements, and the quadrilateral faces of bricks, wedges, and pyramids. A high factor may indicate a condition the underlying element formulation cannot handle well, or may simply hint at a mesh generation flaw.

Warping Factor Calculation for Quadrilateral Shell Elements

A quadrilateral element's warping factor is computed from its corner node positions and other available data by the following steps:

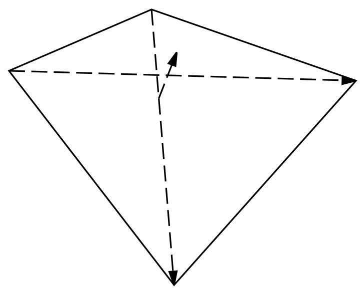
Note:

When computing the warping factor for a quadrilateral shell element, the Meshing application assumes 0 thickness for the shell.

---

1. An average element normal is computed as the vector (cross) product of the 2 diagonals (Figure 48: Shell Average Normal Calculation (p. 137)).

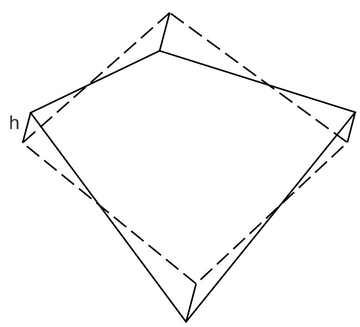
Figure 48: Shell Average Normal Calculation



2. The projected area of the element is computed on a plane through the average normal (the dotted outline on Figure 49: Shell Element Projected onto a Plane (p. 137)).

3. The difference in height of the ends of an element edge is computed, parallel to the average normal. In Figure 49: Shell Element Projected onto a Plane (p. 137), this distance is 2h. Because of the way the average normal is constructed, h is the same at all four corners. For a flat quadrilateral, the distance is zero.

Figure 49: Shell Element Projected onto a Plane



4. The "area warping factor" (F_a^w) for the element is computed as the edge height difference divided by the square root of the projected area.

5. For all shells except those in the "membrane stiffness only" group, if the thickness is available, the "thickness warping factor" is computed as the edge height difference divided by the average element thickness. This could be substantially higher than the area warping factor computed in 4 (above).

6. The warping factor tested against warning and error limits (and reported in warning and error messages) is the larger of the area factor and, if available, the thickness factor.

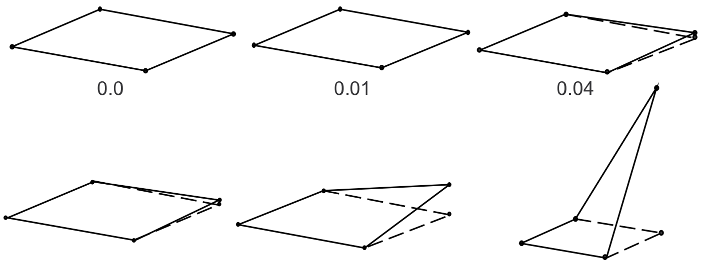
7. The best possible quadrilateral warping factor, for a flat quadrilateral, is zero.

Figure 50: Quadrilateral Shell Having Warping Factor (p. 138) shows a "warped" element plotted on top of a flat one. Only the right-hand node of the upper element is moved. The element is a unit square, with a real constant thickness of 0.1.

When the upper element is warped by a factor of 0.01, it cannot be visibly distinguished from the underlying flat one.

When the upper element is warped by a factor of 0.04, it just begins to visibly separate from the flat one.

Figure 50: Quadrilateral Shell Having Warping Factor



Warping of 0.1 is visible given the flat reference, but seems trivial; however, it is well beyond the error limit for a membrane shell. Warping of 1.0 is visually unappealing. This is the error limit for most shells.

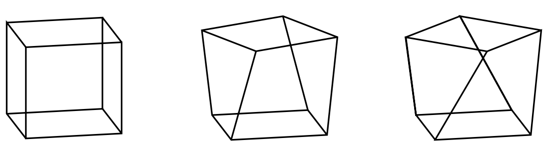
Warping beyond 1.0 would appear to be obviously unacceptable; however, SHELL181 permits even this much distortion. Furthermore, the warping factor calculation seems to peak at about 7.0. Moving the node further off the original plane, even by much larger distances than shown here, does not further increase the warping factor for this geometry. Users are cautioned that manually increasing the error limit beyond its default of 5.0 for these elements could mean no real limit on element distortion.

Warping Factor Calculation for 3-D Solid Elements

The warping factor for a 3-D solid element face is computed as though the 4 nodes make up a quadrilateral shell element with no real constant thickness available, using the square root of the projected area of the face as described in 4 (above).

The warping factor for the element is the largest of the warping factors computed for the 6 quadrilateral faces of a brick, 3 quadrilateral faces of a wedge, or 1 quadrilateral face of a pyramid. Any brick element having all flat faces has a warping factor of zero (Figure 51: Warping Factor for Bricks (p. 138)).

Figure 51: Warping Factor for Bricks



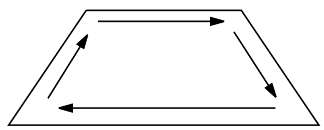
Twisting the top face of a unit cube by 22.5° and 45° relative to the base produces warping factors of about 0.2 and 0.4, respectively.

Parallel Deviation

Parallel deviation is computed using the following steps:

1. Ignoring midside nodes, unit vectors are constructed in 3-D space along each element edge, adjusted for consistent direction, as demonstrated in Figure 52: Parallel Deviation Unit Vectors (p. 139).

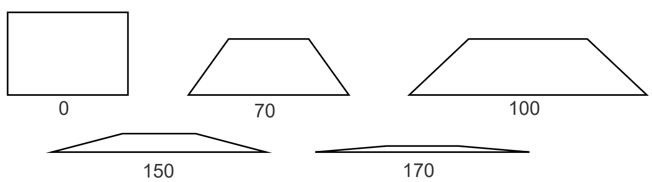
   Figure 52: Parallel Deviation Unit Vectors

2. For each pair of opposite edges, the dot product of the unit vectors is computed, then the angle (in degrees) whose cosine is that dot product. The parallel deviation is the larger of these 2 angles. (In the illustration above, the dot product of the 2 horizontal unit vectors is 1, and acos (1) = 0°. The dot product of the 2 vertical vectors is 0.342, and acos (0.342) = 70°. Therefore, this element's parallel deviation is 70°.)

3. The best possible deviation, for a flat rectangle, is 0°. Figure 53: Parallel Deviations for Quadrilaterals (p. 139) shows quadrilaterals having deviations of 0°, 70°, 100°, 150°, and 170°.

   Figure 53: Parallel Deviations for Quadrilaterals

Maximum Corner Angle

Maximum corner angle is computed and tested for all except Emag elements. Some in the finite element community have reported that large angles (approaching 180°) degrade element performance, while small angles don't.

Maximum Corner Angle Calculation

The maximum angle between adjacent edges is computed using corner node positions in 3-D space. (Midside nodes, if any, are ignored.) The best possible triangle maximum angle, for an equilateral triangle, is 60°. Figure 54: Maximum Corner Angles for Triangles (p. 139) shows a triangle having a maximum corner angle of 165°. The best possible quadrilateral maximum angle, for a flat rectangle, is 90°. Figure 55: Maximum Corner Angles for Quadrilaterals (p. 140) shows quadrilaterals having maximum corner angles of 90°, 140° and 180°.

Figure 54: Maximum Corner Angles for Triangles

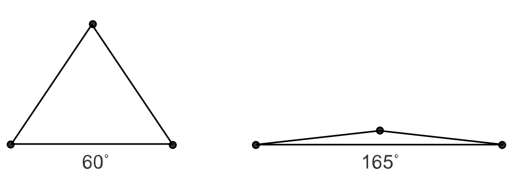
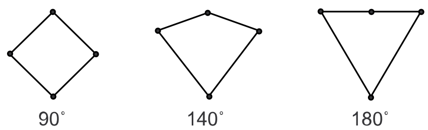


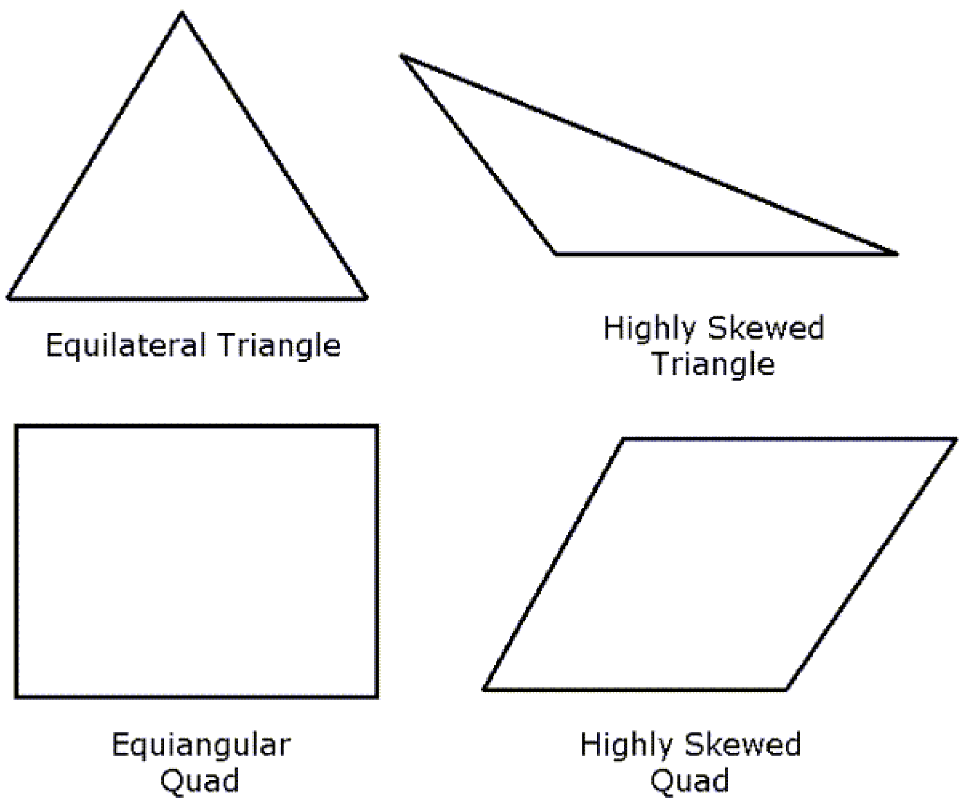
Figure 55: Maximum Corner Angles for Quadrilaterals



Skewness

Skewness is one of the primary quality measures for a mesh. Skewness determines how close to ideal (equilateral or equiangular) a face or cell is (see Figure 56: Ideal and Skewed Triangles and Quadrilaterals (p. 140)).

Figure 56: Ideal and Skewed Triangles and Quadrilaterals



The following table lists the range of skewness values and the corresponding cell quality.

Value of Skewness	**Cell Quality**
1	degenerate
0.9 — <1	bad (sliver)
0.75 — 0.9	poor
0.5 — 0.75	fair
0.25 — 0.5	good
>0 — 0.25	excellent
0	equilateral

According to the definition of skewness, a value of 0 indicates an equilateral cell (best) and a value of 1 indicates a completely degenerate cell (worst). Degenerate cells (slivers) are characterized by nodes that are nearly coplanar (colinear in 2D).

Highly skewed faces and cells are unacceptable because the equations being solved assume that the cells are relatively equilateral/equiangular.

Two methods for measuring skewness are:

- Based on the equilateral volume (applies only to triangles and tetrahedra).
- Based on the deviation from a normalized equilateral angle. This method applies to all cell and face shapes, including pyramids and prisms.

Equilateral-Volume-Based Skewness

In the equilateral volume deviation method, skewness is defined as

$$\text{Skewness} = \frac{\text{Optimal Cell Size - Cell Size}}{\text{Optimal Cell Size}}$$

where, the optimal cell size is the size of an equilateral cell with the same circumradius.

Quality meshes have a skewness value of approximately 0.1 for 2D and 0.4 for 3D. The table above provides a general guide to the relationship between cell skewness and quality.

In 2D, all cells should be good or better. The presence of cells that are fair or worse indicates poor boundary node placement. You should try to improve your boundary mesh as much as possible, because the quality of the overall mesh can be no better than that of the boundary mesh.

In 3D, most cells should be good or better, but a small percentage will generally be in the fair range and there are usually even a few poor cells.

Note:

The Equilateral-Volume-Based Skewness quality metric applies to any mesh element that includes a triangular face. For triangular and tetrahedral elements, all faces of which are strictly triangular, the Equilateral-Volume-Based Skewness metric applies directly. For wedge or pyramidal elements, which include combinations of triangular and quadrilateral faces, the Meshing application computes both Equilateral-Volume-Based Skewness metrics (for the triangular faces) *and* Normalized Equiangular Skewness metrics (for the quadrilateral faces and 3-D element, itself) and reports the maximum computed metric as the element skewness. As a result, Equilateral-Volume-Based Skewness metrics reported for meshes that contain wedge and/or pyramidal elements may include skewness values attributable to Normalized Equiangular Skewness computations.

Normalized Equiangular Skewness

In the normalized angle deviation method, skewness is defined (in general) as

$$\max\left[\frac{\theta\max-\theta_e}{180-\theta_e},\frac{\theta_e-\theta_{\min}}{\theta_e}\right]$$

where

θ_{max} = largest angle in the face or cell

θ_{min} = smallest angle in the face or cell

θ_e = angle for an equiangular face/cell (60 for a triangle, 90 for a square)

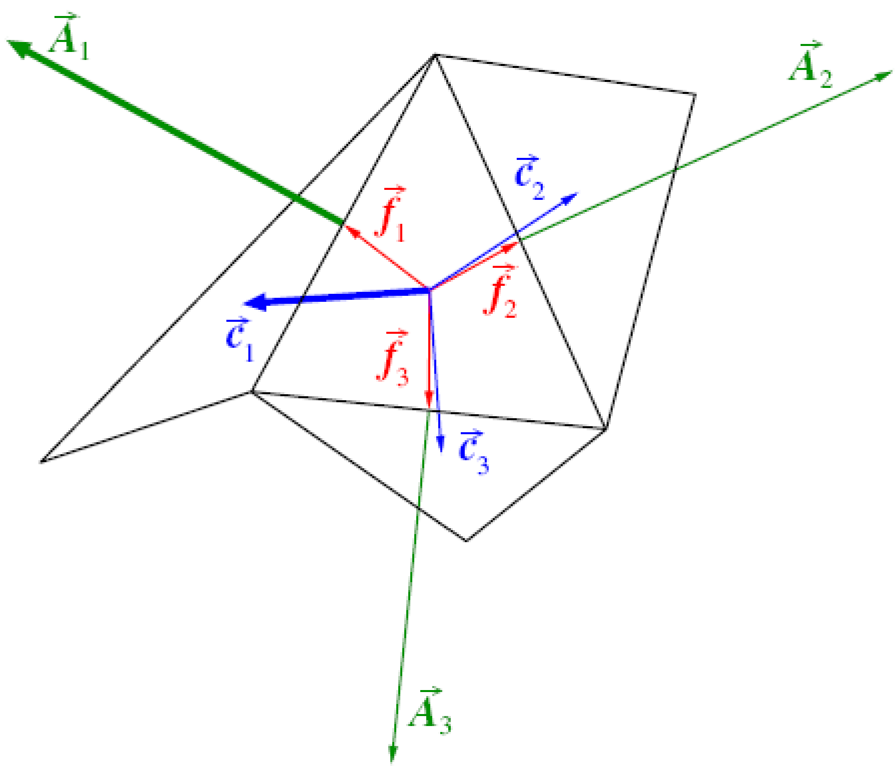
For a pyramid, the cell skewness will be the maximum skewness computed for any face. An ideal pyramid (skewness = 0) is one in which the 4 triangular faces are equilateral (and equiangular) and the quadrilateral base face is a square. The guidelines in the table above apply to the normalized equiangular skewness as well.

Orthogonal Quality

The range for orthogonal quality is 0-1, where a value of 0 is worst and a value of 1 is best.

The orthogonal quality for cells is computed using the face normal vector, $\vec{A}_i$ for each face; the vector from the cell centroid to the centroid of each of the adjacent cells, $\vec{c}_i$; and the vector from the cell centroid to each of the faces, $\vec{f}_i$. Figure 57: Vectors Used to Compute Orthogonal Quality for a Cell (p. 142) illustrates the vectors used to determine the orthogonal quality for a cell.

Figure 57: Vectors Used to Compute Orthogonal Quality for a Cell



For each face, the cosines of the angle between $\vec{A}_i$ and $\vec{c}_i$, and between $\vec{A}_i$ and $\vec{f}_i$, are calculated. The smallest calculated cosine value is the orthogonality of the cell. Finally, **Orthogonal Quality** depends on cell type:

- For tetrahedral, prism, and pyramid cells, the Orthogonal Quality is the minimum of the orthogonality and (1 - cell skewness).
- For hexahedral and polyhedral cells, the Orthogonal Quality is the same as the orthogonality.

Note:

- When the cell is located on the boundary, the vector $\vec{c}_i$ across the boundary face is ignored during the quality computation.
- When the cell is separated from the adjacent cell by an internal wall (a baffle), the vector $\vec{c}_i$ across the internal boundary face is ignored during the quality computation.
- When the adjacent cells share a parent-child relation, the vector $\vec{f}_i$ is the vector from the cell centroid to the centroid of the child face while the vector $\vec{c}_i$ is the vector from the cell centroid to the centroid of the adjacent child cell sharing the child face.

Orthogonal quality in the Meshing application is equivalent to Inverse Orthogonal Quality in Ansys Fluent Meshing, except that the scale is reversed:

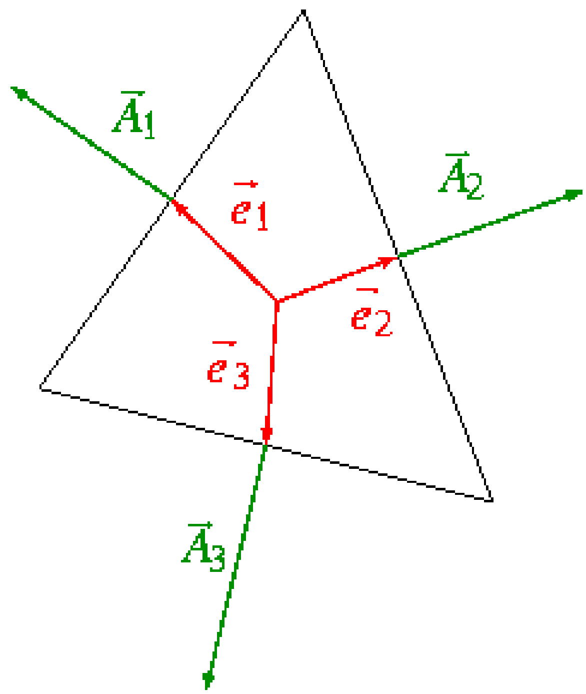
Inverse Orthogonal Quality = 1 – Orthogonal Quality

The orthogonal quality values may not correspond exactly with the inverse orthogonal quality values in Ansys Fluent because the computation depends on boundary conditions on internal surfaces (*WALL* vs. *INTERIOR/INTERNAL/FAN/RADIATOR/POROUS-JUMP*). Ansys Fluent may return different results which reflect the modified mesh topology on which CFD simulations are performed. Also, for **CutCell** (p. 367) meshes, the elements in the Meshing application are "traditional" (hex/tet/wedge/pyramid) elements. When a **CutCell** mesh is exported from the Meshing application to Ansys Fluent, elements that are connected to parent faces are exported in polyhedral format, while all others retain their type. Note that this behavior is only true for the **CutCell** assembly algorithm; the **Tetrahedrons** assembly algorithm uses only traditional element types.

For more information about Inverse Orthogonal Quality, see Quality Measure.

In a similar way, orthogonal quality for faces is computed as the smallest cosine of the angle between the edge normal vector, $\vec{A}_i$ for each edge and the vector from the face centroid to the centroid of each edge, $\vec{e}_i$. Figure 58: Vectors Used to Compute Orthogonal Quality for a Face (p. 144) illustrates the vectors used to determine the orthogonal quality for a face.

Figure 58: Vectors Used to Compute Orthogonal Quality for a Face



Characteristic Length

Characteristic length (also sometimes called characteristic dimension) is used to compute the time step that satisfies the Courant-Friedrichs-Lewy (CFL) condition for a given analysis setup.

The CFL condition is of interest mostly in explicit dynamics and computational fluid dynamics analyses. It governs the maximum time step for which a solution will be stable, and it must be met for the solution to converge. The CFL condition can be expressed as follows:

$$\Delta t \leq f * \left[\frac{h}{c} \right]_{min}$$

where:

f = time step safety factor (commonly/default 0.9)

h = characteristic length

c = material sound speed

such that if you know the characteristic length and material sound speed, you can determine the time step safety factor.

As h decreases, so does the time step. The definition of h varies based on element type:

Element Type	Definition of Characteristic Length (*h*)
Hexahedral or wedge	The volume of the element divided by the square of the longest diagonal and scaled by $\sqrt{2/3}$
Tetrahedral	The minimum distance of any element node to its opposing element face

Characteristic Length calculated by the Mesh Metrics and the User Defined Function are the same.

Inflation Group

Inflation is useful for CFD boundary layer resolution, electromagnetic air gap resolution or resolving high stress concentrations for structures. Inflation is supported for the following mesh methods:

Volume Meshing:

- Patch Conforming (p. 200)
- Patch Independent (p. 200)
- Sweep (p. 223)

> **Note:**
>
> Inflation is supported for the sweep mesh method only when the **Src/Trg Selection** option is set to either **Manual Source** or **Manual Source and Target**.

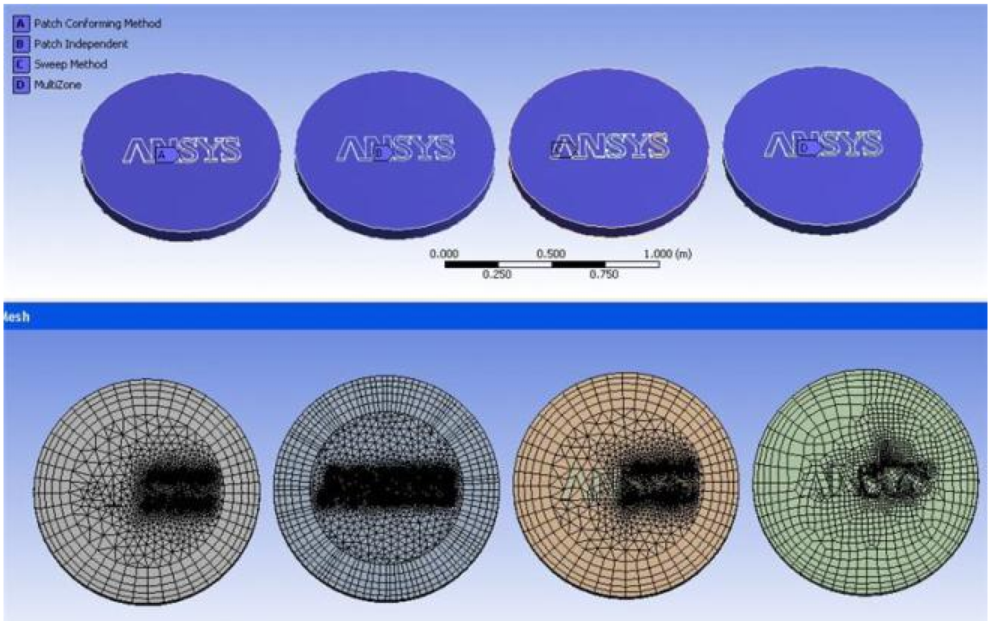
- MultiZone (p. 228)
- Cartesian (p. 236)
- Assembly meshing algorithms (CutCell and Tetrahedrons) (p. 367)

Surface Meshing:

- Quad Dominant (p. 245)
- All Triangles (p. 246)
- MultiZone Quad/Tri (p. 246)

Figure 59: Inflation into Volume Mesh Methods (p. 146) illustrates inflation into the patch conforming, patch independent, sweep, and MultiZone mesh methods respectively.

Figure 59: Inflation into Volume Mesh Methods



Defining Global Inflation Controls

The **Inflation** group of global mesh controls appears in the Details View when the **Mesh** object is selected in the Tree Outline. The options in the **Inflation** group provide global control over all inflation boundaries.

In most cases, the controls in the **Inflation** group apply to both 3D and 2D inflation. Additional information that is specific to 2D inflation is noted where applicable.

Basic options include:

- Use Automatic Inflation
- Inflation Option
- Transition Ratio
- Maximum Layers
- Growth Rate
- Number of Layers
- Maximum Thickness
- First Layer Height
- First Aspect Ratio
- Aspect Ratio (Base/Height)
- Inflation Algorithm
- View Advanced Options

Defining Local Inflation Controls

In addition to setting global inflation controls, you can use local (scoped) inflation controls to apply inflation to specific boundaries. In most cases, the values that you set globally will be populated to the local inflation controls. If you subsequently make changes to the local inflation settings, the local settings will override the global settings. For details, refer to Inflation Control (p. 291).

Inflation and Mesh Method Controls

For steps to follow to assign inflation depending on the selected mesh method, refer to Inflation Controls (p. 414). For general information on applying inflation controls in combination with the various mesh method controls, refer to Meshing: Mesh Control Interaction Tables (p. 435).

Use Automatic Inflation

You can set the **Use Automatic Inflation** control so that inflation boundaries are selected automatically depending on whether or not they are members of Named Selections groups. The following options are available:

None

Program Controlled

All Faces in Chosen Named Selection

Remember the following information when using inflation:

- Automatic inflation is supported only for 3D inflation on volume models. It is not supported for 2D inflation on shell models. You cannot select Program Controlled (p. 148) or All Faces in Chosen Named Selection (p. 149) for the **Use Automatic Inflation** option to mesh a 2D model. To apply 2D inflation on a shell model, use local inflation mesh controls (p. 291) instead.
- In the following scenarios, using inflation results in automatic suppression of the refinement (p. 264) control:
  - When automatic inflation (either Program Controlled (p. 148) or All Faces in Chosen Named Selection (p. 149)) is used with refinement in the same model
  - When local inflation (p. 291) is used with refinement in the same body or in the same part
- In general, using a mixture of local inflation and automatic inflation within the same part is *not recommended*. When you mix local and automatic inflation, the local inflation settings will be used for the bodies to which they are applied. The automatic inflation settings will create inflation *only* on those bodies that *do not* have local inflation applied to them. The automatic inflation settings will ignore all bodies and all attached faces to which local inflation settings are applied.

  If an assembly meshing algorithm is being used, a mixture of local and automatic inflation is *not supported*. Refer to The Assembly Meshing Workflow (p. 372) for details.

None

If you select **None**, inflation boundaries are not selected globally. Instead, the inflation boundaries that you identify using the local inflation mesh controls (p. 291) are used. This is the default.

Program Controlled

Program Controlled inflation operates differently depending on whether meshing occurs at the part/body level or assembly level.

Note:

Program Controlled inflation is only supported for 3D models.

Program Controlled Inflation for Part/Body Level Meshing

If you are using a mesh method that operates at the part/body level and you select **Program Controlled** inflation, all faces in the model are selected to be inflation boundaries, except for the following:

- Faces in Named Selection(s)

  Note:

  By default, faces in Named Selections are not selected to be inflation boundaries when **Use Automatic Inflation** is set to **Program Controlled**. However, you can select specific Named Selections to be included in **Program Controlled** inflation. For details, see the discussion of Named Selections and **Program Controlled** inflation in the Mechanical help.

- Faces in Contact region(s)
- Faces in Symmetry definition
- Faces that belong to a part/body that has a mesh method defined on it that does not support 3D inflation definitions (mesh method is set to Sweep (p. 223) or Hex Dominant (p. 222))
- Faces in sheet bodies
- Faces on bodies that have manual inflation controls

When **Program Controlled** automatic inflation is used for part/body level meshing, the mesher inflates in the following manner:

- For single body parts, the face will always inflate into the body.
- For multibody parts with a mix of fluid and solid parts:
  - Faces on fluid region will inflate into fluid region; solid region will not be inflated.
  - Faces on parts of the same material will not be inflated.
- For parts of the same material, shared faces will not be inflated.

The manner in which inflation is applied depends on values that you enter for the following options:

- Inflation Option (p. 150)
- Inflation Algorithm (p. 154)
- View Advanced Options (p. 158)

Note:

When **Program Controlled** automatic inflation is being used for part/body level meshing, you can view the surfaces that have been selected for inflation by using the **Show Program Controlled Inflation Surfaces** (p. 493) feature.

Program Controlled Inflation for Assembly Level Meshing

If you are using an assembly meshing algorithm (p. 367) and you select **Program Controlled** inflation, all faces in the model are selected to be inflation boundaries, except for the following:

- Faces in Named Selection(s)

Note:

By default, faces in Named Selections are not selected to be inflation boundaries when **Use Automatic Inflation** is set to **Program Controlled**. However, you can select specific Named Selections to be included in **Program Controlled** inflation. For details, see the discussion of Named Selections and **Program Controlled** inflation in the Mechanical help.

- Faces in Symmetry definition

When **Program Controlled** automatic inflation is used for assembly level meshing, the mesher inflates in the following manner:

- All fluid bodies, either real or virtual, will be inflated based on the rules above.
- Solid bodies will not be inflated (where a "solid" body is a volume body with **Fluid/Solid** (p. 379) set to **Solid**).

Note:

The **Show Program Controlled Inflation Surfaces** (p. 493) feature is not supported for assembly meshing algorithms.

All Faces in Chosen Named Selection

If you select **All Faces in Chosen Named Selection**, a **Named Selection** field is displayed to let you scope inflation to the Named Selection. The manner in which inflation is applied to the Named Selections group depends on values that you enter for the following options:

- Inflation Option (p. 150)

- Inflation Algorithm (p. 154)
- View Advanced Options (p. 158)

Note:

The **All Faces in Chosen Named Selection** option is not supported for assembly (p. 367) meshing algorithms. If this option is specified and you select an assembly meshing algorithm, the option will be changed automatically to Program Controlled (p. 148) and a warning will be issued.

Inflation Option

The **Inflation Option** settings determine the heights of the inflation layers. The following options are available:

- **Smooth Transition** - This is the default. The **Smooth Transition** option uses the local tetrahedral element size to compute each local initial height and total height so that the rate of volume change is smooth. Each triangle that is being inflated will have an initial height that is computed with respect to its area, averaged at the nodes. This means that for a uniform mesh, the initial heights will be roughly the same, while for a varying mesh, the initial heights will vary.

  The computations used for prism layer growth are as follows:

  – The following value is computed at each node on the prism base:

    Height of last prism (H) = Transition_Ratio (p. 152) * average_edge_length

  – The height of the first layer (h) is computed using the following formula, where g = Growth Rate (p. 153) and n = Number of Layers (p. 153):

    H = h * (g ^ (n-1))

  Increasing the value of the **Growth Rate** control reduces the total height of the inflation layer. The total height approaches an asymptotic value with respect to the number of inflation layers.

  For details about the additional controls that appear when **Smooth Transition** is selected, refer to the descriptions of the Transition Ratio (p. 152), Maximum Layers (p. 153), and Growth Rate (p. 153) controls.

  Note:

  The **Smooth Transition** option works differently for the **MultiZone** mesh method. See MultiZone Support for Inflation (p. 364) for details.

- **Total Thickness** - The **Total Thickness** option creates constant inflation layers using the values of the **Number of Layers** and **Growth Rate** controls to obtain a total thickness as defined by the value of the **Maximum Thickness** control. Unlike inflation with the **Smooth Transition** option, with the **Total Thickness** option the thickness of the first inflation layer and each following layer is constant.

For details about the additional controls that appear when **Total Thickness** is selected, refer to the descriptions of the Number of Layers (p. 153), Growth Rate (p. 153), and Maximum Thickness (p. 154) controls.

- **First Layer Thickness** - The **First Layer Thickness** option creates constant inflation layers using the values of the **First Layer Height**, **Maximum Layers**, and **Growth Rate** controls to generate the inflation mesh. Unlike inflation with the **Smooth Transition** option, with the **First Layer Thickness** option the thickness of the first inflation layer and each following layer is constant.

  For details about the additional controls that appear when **First Layer Thickness** is selected, refer to the descriptions of the First Layer Height (p. 154), Maximum Layers (p. 153), and Growth Rate (p. 153) controls.

- **First Aspect Ratio** - The **First Aspect Ratio** option creates inflation layers using the values of the **First Aspect Ratio**, **Maximum Layers**, and **Growth Rate** controls to generate the inflation mesh.

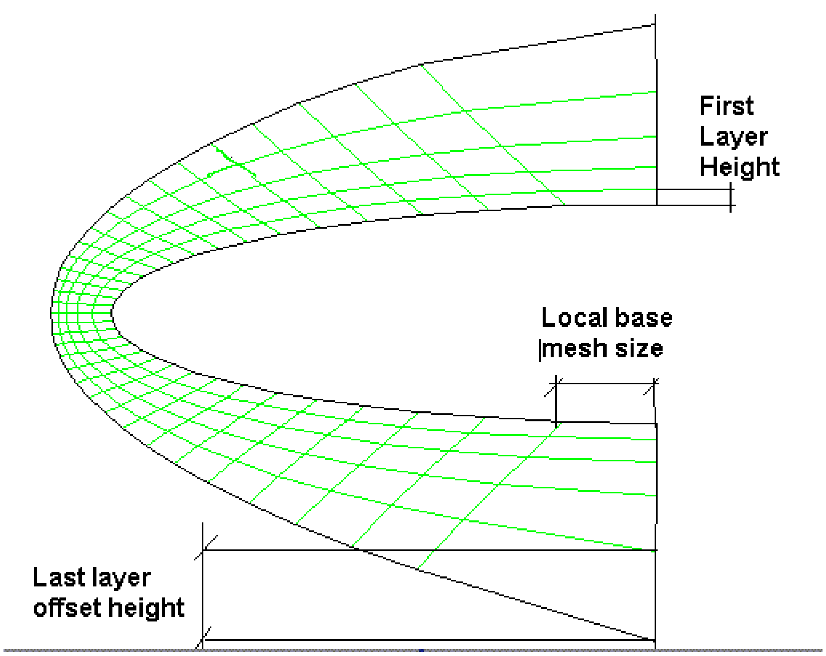
  Post inflation (p. 158) is not supported when **Inflation Option** is **First Aspect Ratio**.

  For details about the additional controls that appear when **First Aspect Ratio** is selected, refer to the descriptions of the First Aspect Ratio (p. 154), Maximum Layers (p. 153), and Growth Rate (p. 153) controls.

- **Last Aspect Ratio** - The **Last Aspect Ratio** option creates inflation layers using the values of the **First Layer Height**, **Maximum Layers**, and **Aspect Ratio (Base/Height)** controls to generate the inflation mesh.

  Figure 60: Last Aspect Ratio Option (p. 151) illustrates this option. With the **Last Aspect Ratio** method, the **First Layer Height** is specified. The offset height for the last layer is calculated from the local base mesh size and specified **Aspect Ratio (base/height)**. For example, if you specify a value of 3 for **Aspect Ratio (base/height)**, the offset height of the last layer will be the local base mesh size divided by 3. The local growth rate is calculated using **Maximum layers** to create exponential growth through the intermediate layers.

Figure 60: Last Aspect Ratio Option



Post inflation (p. 158) is not supported when **Inflation Option** is **Last Aspect Ratio**.

For details about the additional controls that appear when **Last Aspect Ratio** is selected, refer to the descriptions of the First Layer Height (p. 154), Maximum Layers (p. 153), and Aspect Ratio (Base/Height) (p. 154) controls.

Note:

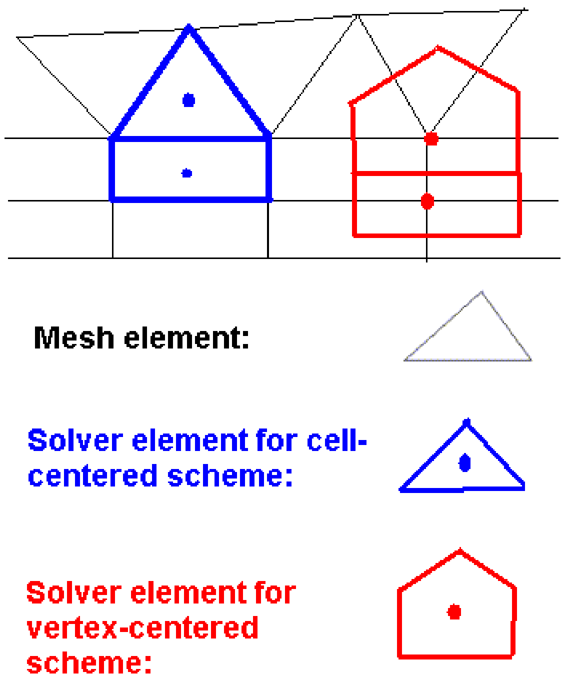
For bodies meshed with the **Body Fitted Cartesian** method, the Inflation option cannot be changed. If **Physics Preference** is set to **CFD**, then three inflation layers are created with total thickness proportional to **Element Size**. For any other physics preference, a single inflation layer is created with thickness proportional to element size.

Transition Ratio

In basic terms, the **Transition Ratio** control determines the rate at which adjacent elements grow. It is the volume-based size change between the last layer of elements in the inflation layer and the first elements in the tetrahedron region (or hexahedron region for **CutCell** meshing (p. 367)). The value of **Transition Ratio** is an ideal value and should produce accurate size change for inflation from a planar boundary. However, be aware that areas of strong curvature will introduce an inaccuracy into the size change. If proximity (p. 100) detection is activated, elements in proximity and elements with prism aspect ratios meeting the value defined by the Maximum Height over Base (p. 161) control will ignore this transition ratio.

The **Transition Ratio** control is applicable only when Inflation Option (p. 150) is **Smooth Transition**. Valid values for **Transition Ratio** are from **0** to **1**. When Physics Preference (p. 93) is set to **CFD** and Solver Preference (p. 95) is set to **CFX**, the default for **Transition Ratio** is **0.77**. For all other physics preferences, including **CFD** when **Solver Preference** is set to either **Fluent** or **Polyflow**, the default is **0.272**.

The reason for this difference is because the **Fluent** and **Polyflow** solvers use a cell-centered scheme for transition ratio. In a cell-centered scheme, the fluid flow variables are allocated at the center of the computational cell, and the mesh-element is the same as the solver-element. In contrast, the **CFX** solver uses a vertex-centered scheme. In a vertex-centered scheme, the fluid flow variables are stored at the cell vertex, and the solver-element is a "dual" of the mesh-element. This means that the vertex of the mesh-element *is* the center of the solver-element. The difference between the two schemes is illustrated in the figure below.



Maximum Layers

The **Maximum Layers** control determines the maximum number of inflation layers to be created in the mesh. Valid values are from **1** to **1000**. The default is **5** for solid bodies and **2** for surface bodies.

If you set a different number of maximum layers on adjacent regions, stair stepping will be used between the regions.

The **Maximum Layers** control is applicable only when Inflation Option (p. 150) is **Smooth Transition**, **First Layer Thickness**, **First Aspect Ratio**, or **Last Aspect Ratio**.

Growth Rate

The **Growth Rate** control determines the relative thickness of adjacent inflation layers. As you move away from the face to which the inflation control is applied, each successive layer is approximately one growth rate factor thicker than the previous one. Valid values are from **0.1** to **5.0**. The default is **1.2**.

The **Growth Rate** control is applicable only when Inflation Option (p. 150) is **Smooth Transition**, **First Layer Thickness**, **Total Thickness**, or **First Aspect Ratio**.

Number of Layers

The **Number of Layers** control determines the actual number of inflation layers in the mesh, except in places where layers are removed locally for reasons of improving mesh quality (for example, in areas where inflation layers would otherwise collide with each other). Valid values are from **1** to **1000**. The default is **5** for solid bodies and **2** for surface bodies.

If you set a different number of layers on adjacent regions, stair stepping will be used between the regions.

The **Number of Layers** control is applicable only when Inflation Option (p. 150) is **Total Thickness**.

Maximum Thickness

The **Maximum Thickness** control determines the desired thickness of the inflation layer. You must enter a value for this control, and it must be greater than 0.

The **Maximum Thickness** control is applicable only when Inflation Option (p. 150) is **Total Thickness**.

First Layer Height

The **First Layer Height** control determines the height of the first inflation layer. This first inflation layer consists of a single layer of prism elements that is formed against the faces of the inflation boundary. You must enter a value for this control, and it must be greater than 0.

The **First Layer Height** control is applicable only when Inflation Option (p. 150) is **First Layer Thickness** or **Last Aspect Ratio**.

First Aspect Ratio

By choosing the **First Aspect Ratio** option for the **Inflation Option** control, you can control the heights of the inflation layers by defining the aspect ratio of the inflations that are extruded from the inflation base. The *aspect ratio* is defined as the ratio of the local inflation base size to the inflation layer height. Use the **First Aspect Ratio** control to specify the first aspect ratio to be used. Enter a value greater than **0**. The default is **5**.

The **First Aspect Ratio** control is applicable only when Inflation Option (p. 150) is **First Aspect Ratio**.

Aspect Ratio (Base/Height)

By choosing the **Last Aspect Ratio** option for the **Inflation Option** control, you can control the heights of the inflation layers by defining the aspect ratio of the inflations that are extruded from the inflation base. The *aspect ratio* is defined as the ratio of the local inflation base size to the inflation layer height. Use the **Aspect Ratio (Base/Height)** control to specify the aspect ratio to be used. Enter a value between **0.5** and **20**. The default is **1.5** when Solver Preference (p. 95) is **CFX**, and **3** when Solver Preference (p. 95) is **Fluent** or **Polyflow**.

The **Aspect Ratio (Base/Height)** control is applicable only when Inflation Option (p. 150) is **Last Aspect Ratio**.

Inflation Algorithm

The **Inflation Algorithm** control determines which inflation algorithm will be used. Options for **Inflation Algorithm** are **Pre** and **Post** and are dependent upon the selected mesh method.

Note:

Post Inflation is being deprecated and will be removed in future releases.

The following table shows which inflation algorithms are applicable to each mesh method. For information on how the inflation algorithm is handled when a combination of mesh methods is being used, see Interactions Between Mesh Methods (p. 435).

Note:

The **Inflation Algorithm** control is hidden when an assembly (p. 367) meshing algorithm is selected. Refer to The Assembly Meshing Workflow (p. 372) for details.

Mesh Method	Inflation Algorithm	
	Pre	Post
Patch Conforming Tetrahedrons (p. 200)	Yes, 3D	Yes, 3D
Patch Independent Tetrahedrons (p. 200)	N/A	Yes, 3D
Hex Dominant (p. 222)	N/A	N/A
Sweep (p. 223)	Yes, 2D. Occurs in the following manner: 1) source face is meshed with triangles, 2) inflation occurs on tri surface mesh, and 3) source is swept. Intervals on source and target are fixed.	N/A
MultiZone (p. 228)	The **Inflation Algorithm** displays as **Pre** but an O-grid-based algorithm specific to **MultiZone** is used. As with the **Pre** inflation algorithm, the mesh is inflated during the meshing process.	N/A
MultiZone Quad/Tri (p. 246)	The **Inflation Algorithm** displays as **Pre** but an O-grid-based algorithm specific to **MultiZone Quad/Tri** is used. As with the **Pre** inflation algorithm, the mesh is inflated during the meshing process.	Yes, 2D
Quad Dominant (p. 245)	Yes, 2D	N/A
All Triangles (p. 246)	Yes, 2D	N/A
Cartesian (p. 236)	The **Inflation Algorithm** displays as **Pre** but algorithm specific to **Body Fitted Cartesian** is used. As with the **Pre** inflation	N/A

Mesh Method	Inflation Algorithm	
	Pre	**Post**
	algorithm, the mesh is inflated during the meshing process. **Note:** • 3D only • If **Physics Preference** is **CFD**, then 3 layers are created. For other physics preferences, only one layer is created.	

Pre

When **Pre** is selected, the surface mesh will be inflated first, and then the rest of the volume mesh will be generated. This is the default for all physics types.

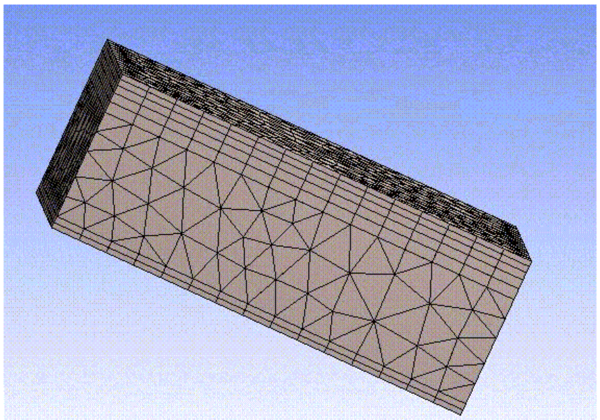
Remember the following information:

- **Inflation Algorithm** must be set to **Pre** for the Previewing Inflation (p. 492) feature to work.
- When Inflation Option (p. 150) is either **First Aspect Ratio** or **Last Aspect Ratio**, **Inflation Algorithm** is set to **Pre** and is read-only.
- Hard points are not supported and may be ignored for 3D Pre inflation.
- Hard edges are not supported and may be ignored for Pre inflation.
- In some cases an additional smoothing attempt will be performed automatically. Refer to Smoothing (p. 123) for details.
- Match controls (p. 280) on faces are supported with Pre inflation, regardless of whether inflation is set to Program Controlled (p. 148) or has been set through any global or local inflation definition. In contrast, match controls on edges are not supported with Pre inflation. Match controls (both faces and edges) are not supported with Post inflation (p. 158). For all these non-supported cases, Ansys Workbench automatically suppresses/disables the Match Control feature.
- For swept meshes with inflation (p. 414) and match control (p. 280), inflation is performed ahead of the match mesh and sweeping. This can affect the sizings on the match controls, which can in turn lead to meshing failure. Therefore, when using both match controls and inflation with sweeping, it might improve meshing robustness if you assign hard edge sizings (p. 252) to the high and low edges of the source face for the sweep.

Pre Inflation and Different Numbers of Layers on Adjacent Faces

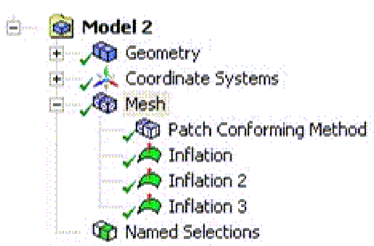
The **Pre Inflation Algorithm** does not support the definition of different numbers of inflation layers on adjacent faces. For example, Figure 61: Different Numbers of Layers Are Respected (p. 157) shows a case in which different numbers of inflation layers have been specified on two faces. Since the faces are not connected, the different numbers of layers are respected.

Figure 61: Different Numbers of Layers Are Respected



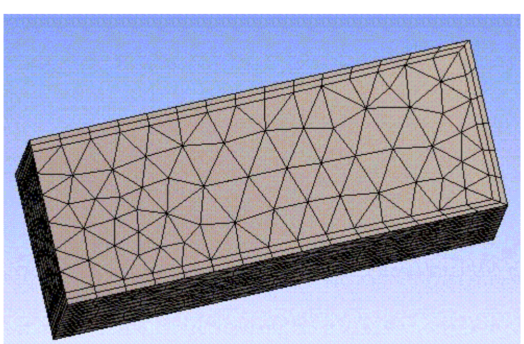
Now assume that the **Inflation** controls shown in Figure 62: Portion of Project Tree (p. 157) define four inflation layers for the **Inflation** control, five inflation layers for the **Inflation 2** control, and two inflation layers for the **Inflation 3** control, and that the controls are defined on adjacent faces.

Figure 62: Portion of Project Tree



In this case, although different numbers of inflation layers were defined, a two-layer (fewest number of layers defined) inflation mesh is generated as shown in Figure 63: Different Numbers of Layers Are Not Respected (p. 157).

Figure 63: Different Numbers of Layers Are Not Respected



Post

When **Post** is selected, a postprocessing technique that works after the tetrahedral mesh is generated is used. A benefit of this option is that the tetrahedral mesh does not have to be generated each time the inflation options are changed.

Remember the following information:

- Match controls (p. 280) on faces are supported with Pre inflation (p. 156), regardless of whether inflation is set to Program Controlled (p. 148) or has been set through any global or local inflation definition. In contrast, match controls on edges are not supported with Pre inflation. Match controls (both faces and edges) are not supported with Post inflation. For all these non-supported cases, Ansys Workbench automatically suppresses/disables the Match Control feature.
- Post inflation is not supported when there is a mixture of tetrahedron and non-tetrahedron mesh methods applied to the bodies in a multibody part. If you want to apply Post inflation to a multibody part, all bodies in the part must have a tetrahedron mesh method applied to them.
- Post inflation is not supported when Inflation Option (p. 150) is either **First Aspect Ratio** or **Last Aspect Ratio**.
- Refer to Using the Mesh Worksheet to Create a Selective Meshing History (p. 409) for information about how Post inflation operations are processed by the **Mesh** worksheet.

View Advanced Options

The **View Advanced Options** control determines whether advanced inflation options appear in the Details View. Choices are **No** (default) and **Yes**. When this control is set to **Yes**, the following options are available:

Collision Avoidance

Maximum Height over Base

Growth Rate Type

Maximum Angle

Fillet Ratio

Use Post Smoothing

Smoothing Iterations

Note:

In addition to viewing and/or modifying the advanced inflation options in the Details View, you can view and/or modify them by using the **Options** dialog box (p. 317).

Collision Avoidance

The **Collision Avoidance** control determines the approach that is to be taken in areas of proximity to avoid collisions that may occur from marching inflated surface meshes from opposite sides into each other.

For 2D inflation, the **Collision Avoidance** control detects geometry limitations in faces that would otherwise cause inflation mesh to overlap or cross face boundaries, or result in a space that is so small that it leads to bad quality mesh for the remaining region of inflation mesh. When **Collision Avoidance** is set to **Layer Compression** or **Stair Stepping**, the value of the Gap Factor (p. 161) control, along with the local mesh size, will determine how much space will be adequate for the remaining region of inflation mesh.

Note:

- The option that you choose for **Collision Avoidance** is used *only in areas of proximity*. In areas of proximity, if the option is set to **Layer Compression**, layer compression is performed; if it is set to **Stair Stepping**, stair stepping is performed; if it is set to **None**, no collision checking is performed. However, in all other problematic scenarios (for example, invalid normals, quality failure, bad surface mesh, and so on), local stair stepping is performed regardless of which option you choose.

- When **Collision Avoidance** is set to **Layer Compression** and local stair stepping occurs after compression, poor quality pyramids may be introduced into the mesh. Because of this possibility, a warning message will appear whenever stair stepping occurs after compression. The message will not identify the location of the stair stepping. However, the location of the stair stepping with added pyramids often coincides with the location of the worst quality element (specifically, when considering the **Skewness** metric). For this reason, using the Meshing application's **Mesh Metric** (p. 123) feature to locate the worst quality element (based on **Skewness**) is also likely to locate the pyramids.

- The **Collision Avoidance** option is not used for **MultiZone** (p. 364) as the inflation layers are created within the blocking approach.

- For assembly meshing algorithms, **Collision Avoidance** is set to **Layer Compression** and is read-only. Refer to Assembly Meshing (p. 367) for details.

The following options are available:

- **None** - The **None** option does not check for layer collisions. Selecting this option speeds up inflation layer computation time. However, it can result in an invalid mesh and mesh failures. For these reasons, this option is not recommended.

  For 2D inflation, if a collision/proximity limitation is detected during layer creation, creation of inflation layers stops with the previous layer. (Inflation stops *completely*; contrast with **Stair Stepping** below.)

- **Layer Compression** - The **Layer Compression** option compresses inflation layers in areas of collision. In these areas, the defined heights and ratios are reduced to ensure the same number of layers throughout the entire inflation region. Generally, this option is best for avoiding the creation of pyramids in the mesh. **Layer Compression** is the default only when the **Physics Preference** (p. 93) is set to **CFD** and the **Solver Preference** (p. 95) is set to **Fluent**; otherwise, the default is **Stair Stepping**.

  For 2D inflation, if a collision/proximity limitation is detected during layer creation, inflation heights will shrink locally. If Fix First Layer (p. 161) is set to **Yes**, the First Layer Height (p. 154) will not be scaled.

For details about the additional controls that appear when **Layer Compression** is selected, refer to the descriptions of the Fix First Layer (p. 161) and Gap Factor (p. 161) controls.

- **Stair Stepping** - Rather than compressing the prism layers, with **Stair Stepping** the prism layers are "stair stepped" in the proximity region to avoid collision and to maintain the gap defined by Gap Factor (p. 161). The **Stair Stepping** approach to inflation growth locally reduces inflation layers to avoid collisions, as well as bad quality elements in sharp or tight corners. The term "stair stepping" refers to the steps created between one layer and the next. Using this approach, special logic is used to fill the steps with pyramid and tetrahedron elements for prism steps, or prism, pyramid, and tetrahedron elements for hex steps. This special logic helps the mesher obtain a high-quality transition to the tetrahedral mesh. **Stair Stepping** is the default, unless the **Physics Preference** (p. 93) is set to **CFD** and the **Solver Preference** (p. 95) is set to **Fluent**, in which case the default is **Layer Compression**.

  For 2D inflation, if a collision/proximity limitation is detected during layer creation, creation of inflation layers stops *locally*. (Contrast with **None** above.)

  An additional control, for **Gap Factor** (p. 161), appears when **Stair Stepping** is selected.

The figures below illustrate how the **Layer Compression** and **Stair Stepping** options differ.

Figure 64: Layer Compression vs. Stair Stepping Option (Full Mesh View)

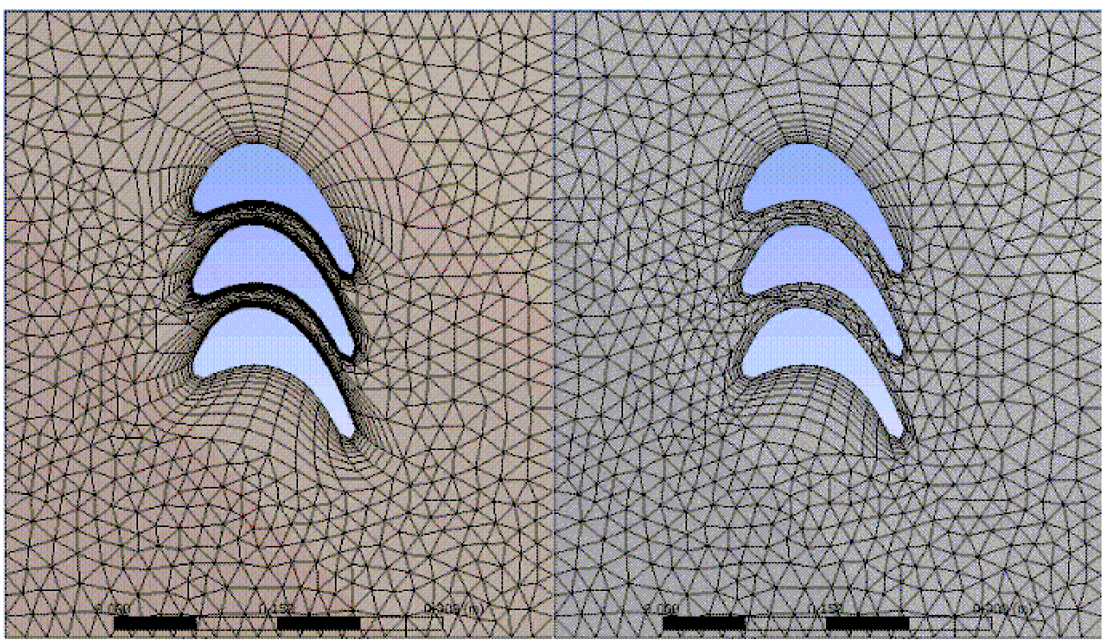
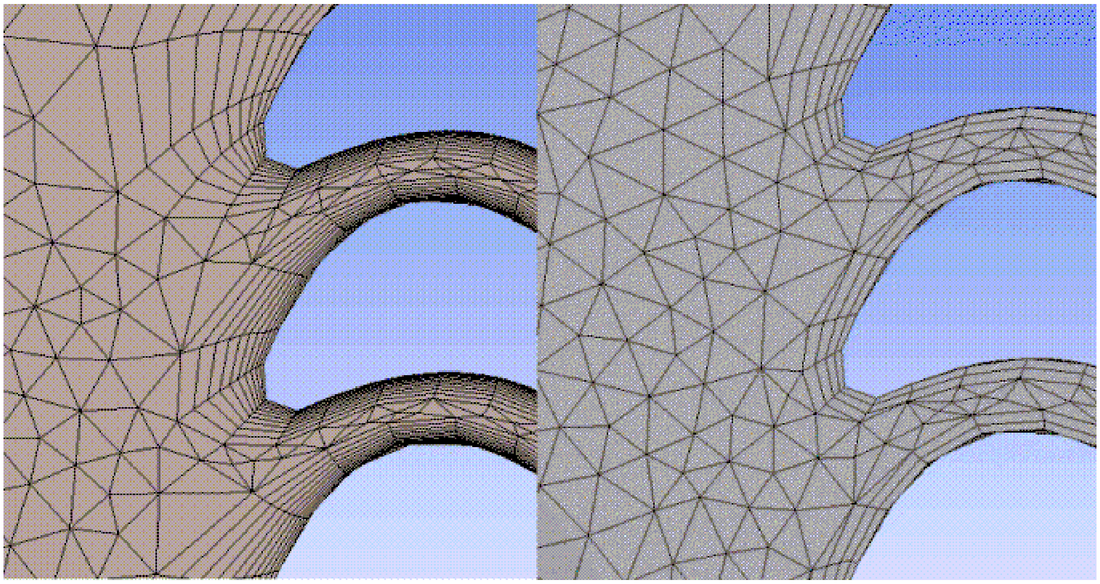


Figure 65: Layer Compression vs. Stair Stepping Option (Detail View)



Fix First Layer

The **Fix First Layer** control determines whether the heights or ratios of the first inflation layer will be modified to avoid collision. Valid values are **Yes** and **No**. The default is **No**. This option will not allow the value that is set for the First Layer Height (p. 154) control to be changed.

The **Fix First Layer** control is applicable only when Collision Avoidance (p. 158) is **Layer Compression**.

Gap Factor

The **Gap Factor** control allows maintenance of the gap between intersecting prisms. Valid values are from **0** to **2**. The default is **0.5**. A value of **1** means a gap equal to the ideal tet cell height based on base face size in proximity to each other is maintained.

The **Gap Factor** control is applicable only when Collision Avoidance (p. 158) is **Layer Compression** or **Stair Stepping**.

Note:

Refer to the discussion of inflation controls in Selecting an Assembly Mesh Method (p. 375) for information about specifying **Gap Factor** for assembly meshing algorithms.

Maximum Height over Base

The **Maximum Height over Base** control sets the maximum allowable prism aspect ratio (that is, the ratio of height over base of the base triangle). When the prism aspect ratio reaches this value, the height of the prisms stops growing. That is, new prisms continue to form, but the heights of the prisms will not increase. Valid values are from **0.1** to **5**. The default is **1.0**.

For 2D inflation, the **Maximum Height over Base** control helps to maintain a good size ratio. Once the inflation height is greater than the local mesh size multiplied by the value of **Maximum Height over Base**, the inflation height stops growing. New layers continue to form, but the heights of the inflation layers will not increase. New layers that form will be equal to the inflation base size multiplied by the value of **Maximum Height over Base**.

Growth Rate Type

The **Growth Rate Type** control determines the height of the inflation layers given the initial height and height ratio. The following options are available:

- **Geometric** - This is the default. With this option, the prism height of a particular layer is defined by $h*r^{(n-1)}$, where h = initial height, r = height ratio, and n = layer number. The total height at layer n is: $h(1-r^n)/(1-r)$.
- **Exponential** - With this option, the prism height of a particular layer is defined by $h*e^{(n-1)p}$, where h = initial height, p = exponent, and n = layer number.
- **Linear** - With this option, the prism height of a particular layer is defined by h(1+(n-1)(r-1)), where h = initial height, r = height ratio, and n = layer number. The total height at layer n is: nh((n-1)(r-1)+2)/2.

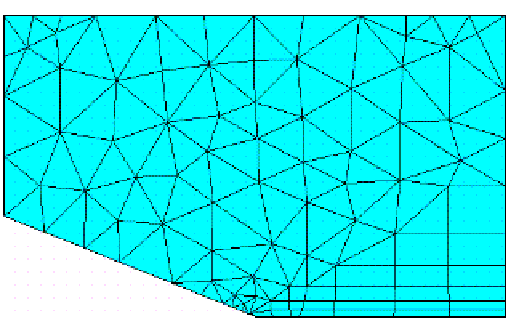
Maximum Angle

The **Maximum Angle** control determines prism layer growth around angles and when prisms will adhere (project) to adjacent surfaces/walls. If the inflated mesh involves extruding from one surface and not its neighbor, and the angle between the two surfaces is less than the specified value, the prisms (sides) will adhere (project) to the adjacent wall. Valid values are from **90** to **180** (degrees). Typically, a value between **120** and **180** is desirable. The default is **140**. Refer to the figures below for examples of maximum angle.

For 2D inflation, the **Maximum Angle** control determines whether an edge that is adjacent to an inflation edge can be imprinted with inflation mesh. If the angle between the two edges is smaller than the value of **Maximum Angle**, the inflation mesh will be imprinted on the adjacent edge. On the other hand, if the angle between the two edges is larger than the value of **Maximum Angle**, the inflation mesh will not be imprinted on the adjacent edge.

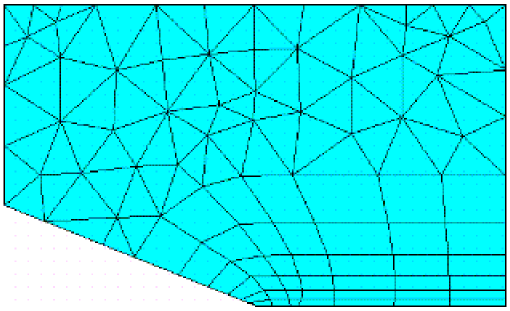
In Figure 66: Maximum Angle = 140 (p. 162), the angle between the planes is 158.2 (21.8) degrees. Since the maximum angle is less than the angle between the walls, the prism layers are capped with pyramids.

Figure 66: Maximum Angle = 140



In Figure 67: Maximum Angle = 180 (p. 163), the maximum angle exceeds the separation angle between the surfaces, so the prism remains attached to the adjacent surface.

Figure 67: Maximum Angle = 180



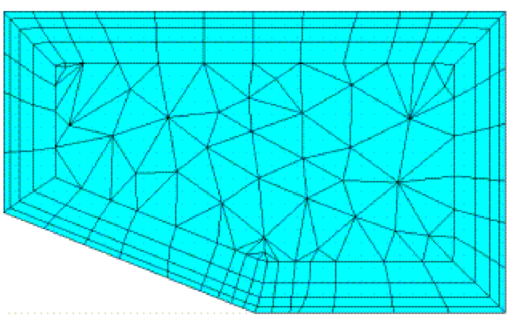
Fillet Ratio

The **Fillet Ratio** control determines whether a fillet proportional to the total height of a prism element will be created when a prism element is generated in the corner zone of a tetrahedral mesh. Creating a fillet proportional to the total height of the prism makes it possible to control the smoothness of the prism layer. Valid values are from **0** to **1** (decimal values are allowed). A value of **0** means no fillets. The default is **1**. Refer to the figures below for examples of fillet ratio.

Note:

For meshing corners with angles less than 60 degrees, there may not be space for a fillet.

Figure 68: Fillet Ratio = 0.0



In the figure below, the radius of the inner prism fillet is 0.5 times the height of the total prism thickness.

Figure 69: Fillet Ratio = 0.5

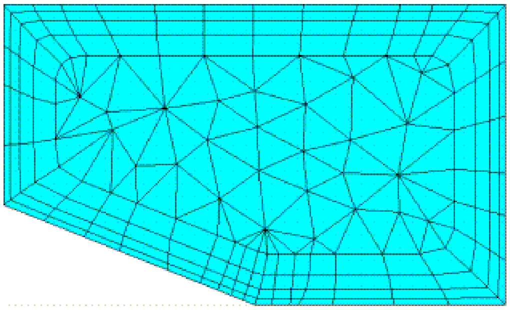
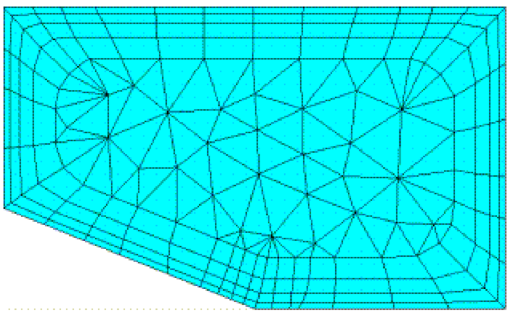


Figure 70: Fillet Ratio = 1.0



Use Post Smoothing

The **Use Post Smoothing** control determines whether post-inflation smoothing will be performed. Smoothing attempts to improve element quality by moving the locations of nodes with respect to surrounding nodes and elements. Valid values are **Yes** and **No**. The default is **Yes**. When this control is set to **Yes**, the Smoothing Iterations (p. 164) control appears in the Details View.

Smoothing Iterations

The **Smoothing Iterations** control determines the number of post-inflation smoothing iterations that will be performed to improve the mesh. Valid values are from **1** to **20**. The default is 5.

The **Smoothing Iterations** control is applicable only when Use Post Smoothing (p. 164) is **Yes**.

Assembly Meshing Group of Controls

Note:

Meshing does not support Assembly Meshing. To enable **Assembly Meshing**, Open Ansys Workbench, click **Tools > Options > Appearance**, select **Unsupported Features** and click **OK** .

"Assembly meshing (p. 367)" refers to meshing an entire model as a single mesh process, as compared to part- or body-based meshing, in which meshing occurs at the part or body level respectively. If the assembly meshing **Method** control is set to **None**, Ansys Workbench meshing operates at the part level, but if it is set to **CutCell** or **Tetrahedrons**, the entire assembly will be meshed at one time using the selected assembly meshing algorithm.

Assemblies can also be meshed using part-based meshing methods, but in such cases the mesher operates one part at a time, and therefore cannot mesh virtual bodies or evaluate parts that occupy the same space.

The **Assembly Meshing** group of global mesh controls is available in both the Meshing application and the Mechanical application, but it is exposed only when **Physics Preference** is set to **CFD** and **Solver Preference** is set to either **Fluent** or **Polyflow**.

Note:

Meshes generated using assembly meshing are not supported for Mechanical solvers. If you try to use a Mechanical solver to solve an analysis of an assembly mesh, the solution is blocked and an error message is issued. Refer to Method (p. 165) for details.

The **Assembly Meshing** group allows you to control these options:

Method

Feature Capture

Tessellation Refinement

Intersection Feature Creation

Morphing Frequency

Keep Solid Mesh

Method

The **Method** control determines whether an assembly meshing algorithm will be used and filters user interface components appropriately. The following options are available:

- **None** - This is the default. Assembly meshing will not be used and controls are not exposed.
- **CutCell** - Selects the **CutCell** strategy for assembly meshing. Exposes assembly meshing controls and hides controls that are not applicable to assembly meshing. The **CutCell** option is supported only in the Meshing application. See also Assembly Meshing (p. 367).

  Note:

  If **CutCell** is active in a Mesh system and you replace the Mesh system with a Mechanical Model system, you will not be able to use **CutCell** in the Mechanical Model system.

- **Tetrahedrons** - Selects the **Tetrahedrons** strategy for assembly meshing. Exposes assembly meshing controls and hides controls that are not applicable to assembly meshing. The **Tetrahedrons** option is available and supported in the Meshing application. **Tetrahedrons** is also available in the Mechanical application; however, meshes generated using assembly meshing are not supported for Mechanical solvers. If you try to use a Mechanical solver to solve an analysis of an assembly mesh, the solution is blocked and an error message is issued. To proceed using a Mechanical solver, you must first deactivate assembly meshing (set **Method** to **None**) and then regenerate the mesh. See also Assembly Meshing (p. 367).

Note:

Changing the **Method** control from **None** to **Tetrahedrons** while using assembly meshing to mesh an entire model may result in differing named selection titles in the system file. If so, the new mesh will be incompatible with the **Polyflow** data file.

Feature Capture

The **Feature Capture** control determines which CAD features are captured for assembly meshing (p. 367). The following options are available:

- **Program Controlled** - This is the default. A feature angle of 40 degrees is used to determine which features are captured. If the shared faces on an edge form an angle smaller than (180 - 40) degrees, the edge is selected for assembly meshing.
- **Feature Angle** - Exposes an additional **Feature Angle** field, where you can set a value from 0 to 90 degrees instead of using the default of 40. The smaller the angle, the higher the number of features that are captured. If you specify a value greater than 90, a feature angle of 90 is used. Setting a negative value resets the feature angle to its default, while setting the value to 0 captures all features.

Tessellation Refinement

The **Tessellation Refinement** control specifies the value to be used for tessellation (faceting) refinement for assembly meshing (p. 367). The following options are available:

- **Program Controlled** - This is the default. Sets tessellation refinement to 10% of the value of Curvature Min Size (p. 108)/Proximity Min Size (p. 110) (whichever is smaller). This is the recommended value for most assembly meshing operations.
- **Absolute Tolerance** - Exposes an additional **Absolute Tolerance** field, where you can set a numerical value for refinement. The recommended range is between 5 and 10% of the value of Curvature Min Size (p. 108)/Proximity Min Size (p. 110) (whichever is smaller). A value on the lower end of the range may work better if you have problems with gaps.
- **None** - Sets tessellation refinement to the CAD program or DesignModeler application default setting.

Note:

- To assist you in working with assembly meshing (p. 367), you may want to use the **Show Missing Tessellations** (p. 498) feature prior to mesh generation.
- Also see Avoiding Bad Feature Capturing in Assembly Meshing (p. 548).

Intersection Feature Creation

In cases where two parts/bodies overlap in space, the **Intersection Feature Creation** control determines whether the intersection between faces is computed. When activated, **Intersection Feature Creation** computes additional feature edges to be respected during the snapping that occurs within Assembly Meshing. Activating this feature is very useful for avoiding zigzag boundaries at an intersection, because it ensures that the "real" intersection lines are respected. However, this operation can be computationally expensive, so you should de-activate it if you have many non-intersecting bodies in the model. The following options are available:

- **Program Controlled** — This is the default, which activates this feature whenever you have assemblies (intersecting or not intersecting).
- **No** — Deactivates the feature.
- **Yes** — Activates the feature.

Morphing Frequency

In the **CutCell** inflation algorithm, inflation layers are grown into the **CutCell** mesh. The volume mesh is morphed so the boundary of the **CutCell** mesh matches the cap of the inflation. The value set for the **Morphing Frequency** determines how often the morphing is repeated. For example, if the **Morphing Frequency** is set to **5** (the default) and more than five layers are grown, then morphing is repeated at every five layers. Similarly, if the **Morphing Frequency** is set to **3** and more than three layers are grown, then morphing is repeated at every three layers.

Note:

Morphing Frequency is applicable only to **CutCell** meshing with inflation.

Keep Solid Mesh

The **Keep Solid Mesh** control determines whether the mesh for any body marked as a solid is discarded or kept. A body may be marked as a solid based on the definition of a solid that the Meshing application uses for exporting the assembly mesh (p. 367) to the solver. The following options are available:

- **No** - This is the default. All solid mesh is discarded. Bodies whose mesh is discarded remain in a meshed state.
- **Yes** - All solid mesh is kept.

Note:

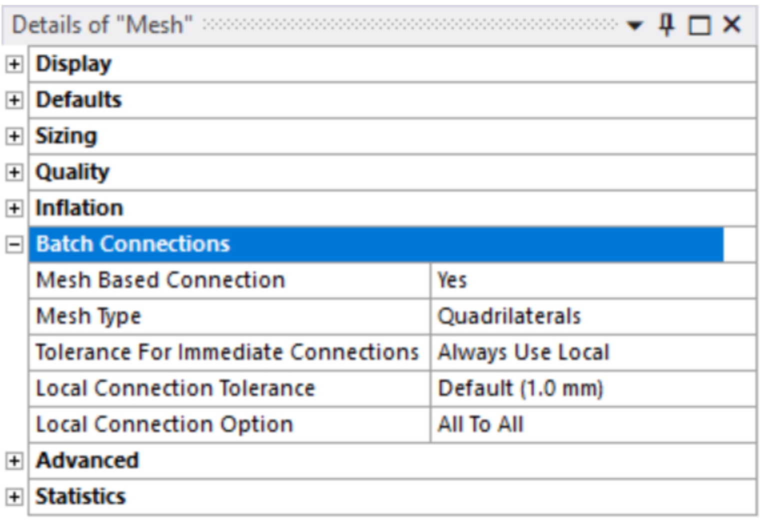
The **Keep Solid Mesh** control does not appear until a virtual body (p. 379) has been created.

Batch Connections

Batch Connections are used for achieving connections in structures made of sheets and beams. **Batch Connections** uses **Connect** control to connect the beams and shells. **Batch Connections** uses **Repair Topology** to defeature the model at mesh level.

Note:

All legacy databases saved with **Mesh Based Connection** set to **Yes** will resume with **Connect** control scoped to the entire assembly.



- **Mesh Based Connection**: Activates mesh based connection. The default value is **No**. When set to **Yes**, the following options are available:

- **Mesh Type**: Allows you to select the type of mesh to be used for meshing. The mesh types available are Triangles and Quadrilaterals.

- **Tolerance for Immediate Connections**: The field is only applicable when you use the **Connect and Mesh Selected Entities** option. There are three options. They are:

  – **Always Use Local**: When you select **Always Use Local**, the following options are available:

    → **Local Connection Tolerance**: Allows you to provide the Connection Tolerance value.

    → **Local Connection Option**: Allows you to select the **Connection Option**. The connection options are **All to All**, **Free to All**, **Free to Free**.

  The default option is **Always Use Local**.

  – **Prompt Each Time**: Allows you to enter the desired **Connection Tolerance** and **Connection Option** while performing selective meshing.

When you select **Quality** in **Batch Connections**, you have the following options:

- **Check Mesh Quality** (p. 118): The default value for **Check Mesh Quality** is **Yes**.

- **Error Limits** (p. 118): Allows you to select two options. They are:
  - **Standard Mechanical** (p. ?)
  - **Aggressive Mechanical** (p. ?)
- **Mesh Metrics** (p. 123)

When you select **Advanced**, the following option is available:

- **Topology Checking** (p. 179)
- **Loop Removal Tolerance** (p. 193)

Local Controls in Batch Connections

Batch Connections support local controls. The local controls supported are :

- **Sizing**

  Local Sizing controls are supported in **Batch Connections**. Sizing controls can be applied to edges and faces.

  - **Face Sizing** allows you to set the **Type** as only **Element Size**. It supports only uniform size functions.
  - **Edge Sizing** allows both **Element size** and **Number of Divisions** to be set.

  For both Face Sizing and Edge Sizing, the default value for **Element size** is same as the **Global Element Size**.

  Note:

  Local sizing controls on vertex and body are not supported in Batch Connections.

  When you select **Sizing** in **Batch Connections**, the following options are available:

  - **Growth Rate**: The default value for **Growth Rate** is 1.2.
  - **Capture Curvature** (p. 102): The default value is **No**. When the **Capture Curvature** is set to **Yes**, the following options are available:
    - → **Curvature Min Size** (p. 108)
    - → **Curvature Normal Angle** (p. 109)
  - **Capture Proximity** (p. 102): The default value is **No**. When the **Capture Proximity** is set to **Yes**, the following options are available:
    - → **Proximity Min Size** (p. 110)
    - → **Num Cells Across Gap** (p. 110)

→ **Proximity Size Function Sources** (p. 110)

- **Repair Topology** (p. 297)
- **Connect** (p. 299)
- **Weld** (p. 302)
- **Mapped Meshing** (p. 265)

  When **Batch Connections** is set to **Yes**, right-click the **Mesh** folder, click **Insert >Mapped Meshing**. You can select the faces and click **Mapped Meshing** (p. 265) to create meshes of the given **Element Size**.

- **Selective Meshing** (p. 404)

  Batch Connections allows you to perform selective local connection by selecting the entities to be connected, then right-click and select **Connect and Mesh Selected Entities** to connect and mesh the selected entities.

Best Practices

The recommended inputs for batch connections are as follows:

- Set the facet quality to 7 before importing model into Mechanical. You can set the facet quality in SpaceClaim by navigating through **SpaceClaim > SpaceClaim Options > Rendering quality**. The facet quality in DesignModeler can be set by navigating through **DesignModeler > Tools > Options >Graphics > Facet Quality**.
- In SpaceClaim, DesignModeler or in other upstream CAD package, **Extend** operation should be performed. The largest connection gap to be resolved should be smaller than the specified **Element size**.
- Remove overhangs in upstream CAD. Overhangs are small penetrations between two surfaces. When unresolved overhangs can lead to mesh failure or mesh quality issues.
- Imprinting edges or performing share topology at CAD level is not recommended. If CAD model has edges already imprinted, then use **Virtual Topology** (p. 501) to remove them.
- If the model has faces with missing facets, then you should fix them in upstream CAD.

The suggested general connection strategy is for clean models that do not have much difference between the smallest and the largest gap to be connected across. The best practices to follow while connecting two entities are as follows:

- Find the smallest and largest gap between the entities to be connected.
  - Define a list of connection tolerance such that the smallest gap is less than the smallest connection tolerance and the largest connection tolerance is greater than the largest gap.
  - **Worksheet** is recommended to resolve large gaps rather than using large connection tolerance globally. Define named selection on the entities which requires large tolerance to be connected. Then use this named selection in the worksheet as a step.

- Check for any warning message on unconnected edges reported during meshing.
- Right-click the warning message and select **Show Problematic Geometry**. Use model **Walk** to analyze the unconnected regions.
- Create named selections on unconnected entities.

• Activate mesh **Worksheet**.

- Add a step in the **Worksheet** and scope the created named selection.
- Add a step for **All Bodies** with connection tolerance(s) used previously as global tolerance(s).

Remember the following while performing **Batch Connections**:

• The gap size between the entities to be connected must be smaller than the **Element Size**. For example, if the biggest gap size to be resolved is 1 mm then the element size must be greater than 1 mm.

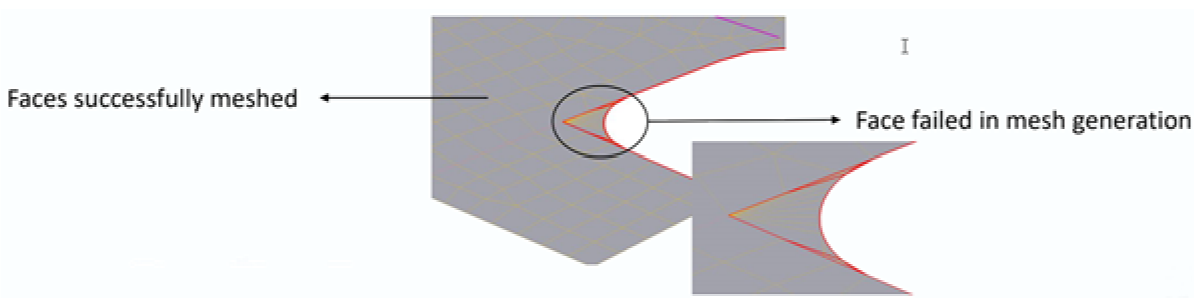
• Faces (width) and edges (length) of the connected entities below the connection tolerance(s) are defeatured.

• Mesh is not associated with defeatured entities (edge, face, body/part).

• Mesh is not associated with vertices.

• In case of quadrilateral mesh failure, triangle mesh is generated with a warning message displayed.

- Right-click the warning message and select **Show Problematic Topology** to display the bodies on which triangle mesh was generated.
- If triangle mesh also fails, then the mesh failure is reported and CAD facets will be displayed.



Limitations

The limitations of batch connections are as follows:

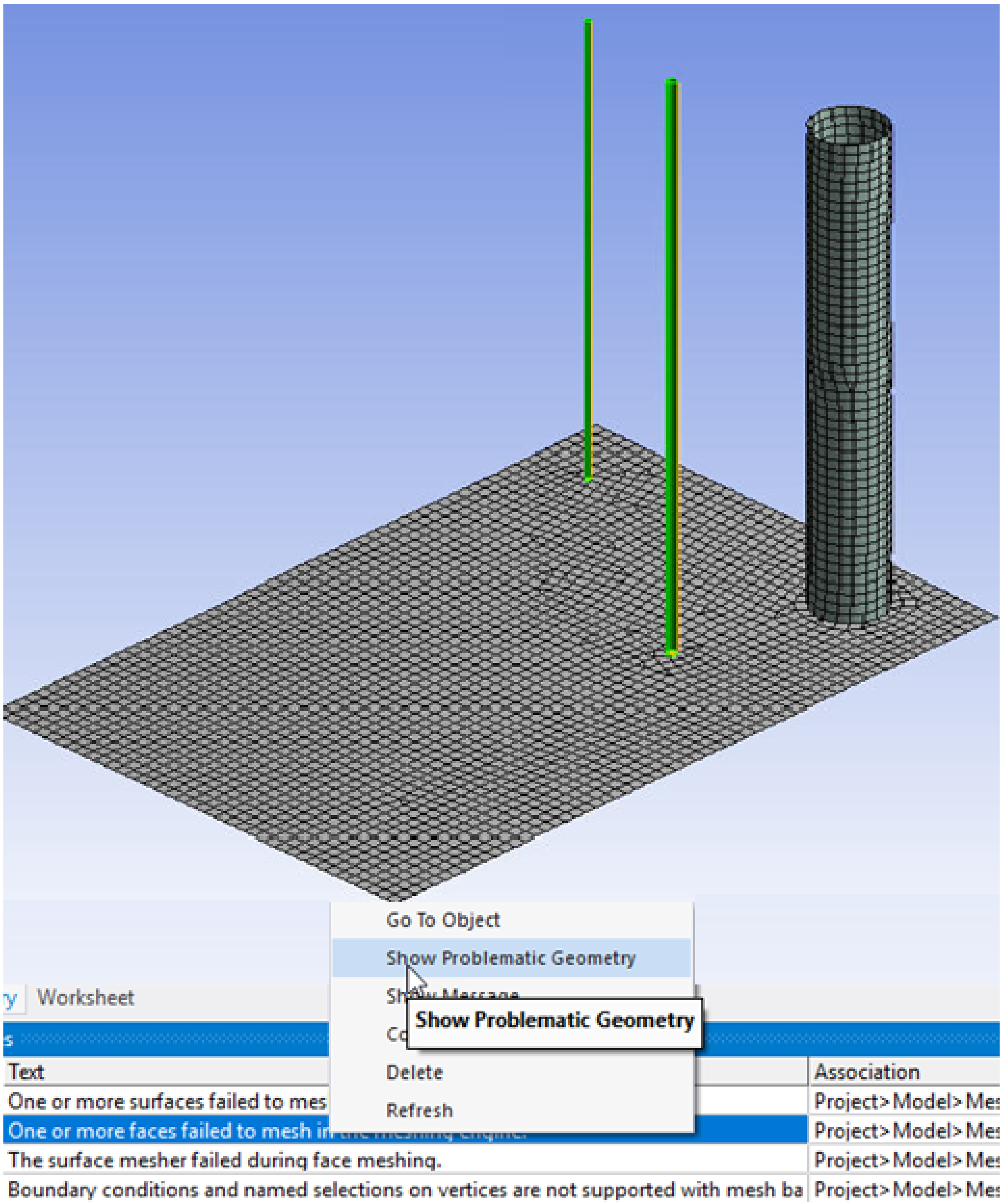
• Overlapping beams are not supported. Mesh in the overlapping portion of the beams will be assigned to either one of the involved beams.

• Selective meshing using **Connect and Mesh Selected Entities** does not allow to save and resume the project to re-mesh the remaining parts of the model. That is, when you mesh a part of the model, then save and close the project for future use in Mechanical, you will not be allowed to mesh the remaining part of the model using **Connect and Mesh Selected Entities** on reopening the project. When you reopen the saved project to continue meshing, all the mesh will be cleared. You may need to restart meshing from the beginning.

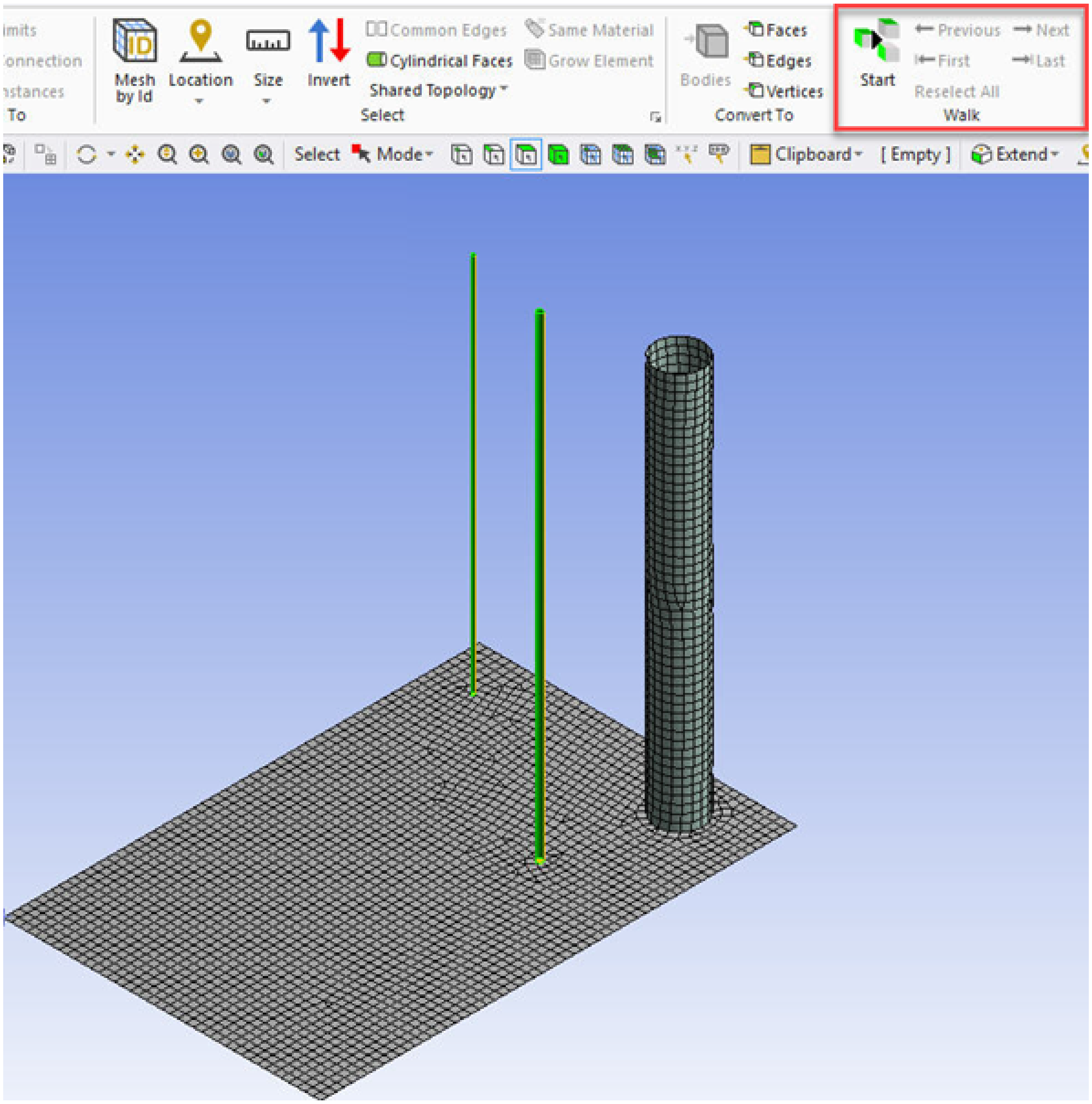
- **Mapped Meshing** does not support mapping of annular region.
- **Weld** and Mapped control override all other sizing and local control settings.
- **Weld** control does not support **Connect and Mesh Selected Entities** option.
- **Batch Connections** does not support multi-body parts which includes solid body parts.
- **Batch Connections** does not create conformal meshes between solids.

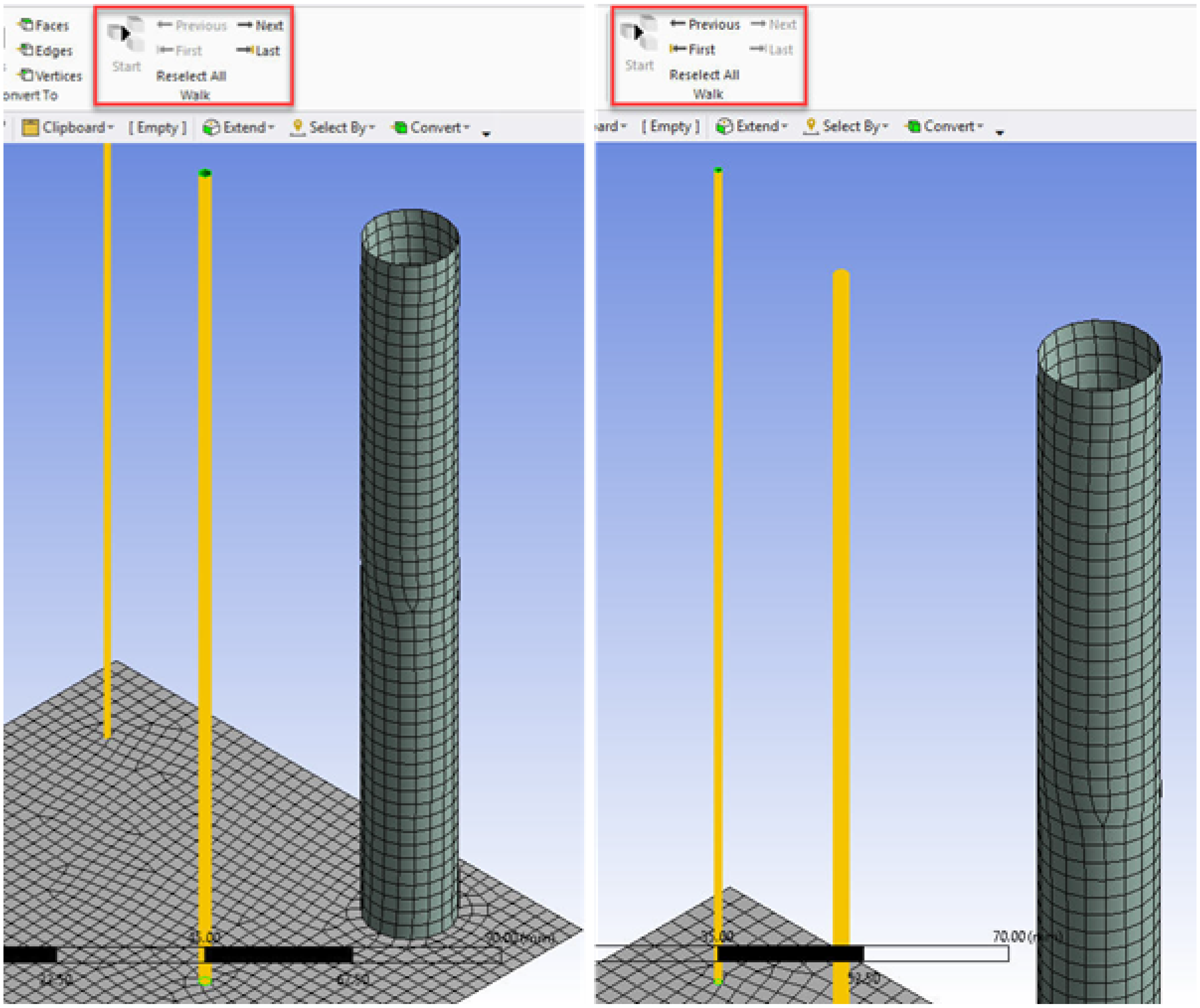
Ease of Use Features

When an error or warning such as unconnected edges or mesh failure occurs, you can right-click the corresponding message and select **Show Problematic Geometry** to highlight the location at which error occurred in the model.



You can click **Selections** tab in the ribbon and click **Start** in **Walk** group to walk through the error locations. You can use **Next** to walk to the next location. You can click **Previous** to go back to the previous error location.


Mesh by Id
Location
Size
Invert
Common Edges
Same Material
Cylindrical Faces
Grow Element
Shared Topology
Select
Bodies
Faces
Edges
Vertices
Convert To
Start
Previous
Next
First
Last
Reselect All
Walk
Select
Mode
Clipboard
[Empty]
Extend



You can view the unconnected faces after meshing. To view the unconnected faces, right-click on the Geometry window. Select **Diagnostics > Find Unconnected Faces**. The unconnected faces get highlighted in the model.

When entities scoped to **Named Selection** with **Protected** set to **Yes** are modified while performing batch connections, you will receive a warning message.

Advanced Group

The **Advanced** group allows you to control these options:

- Number of CPUs for Parallel Part Meshing
- Straight Sided Elements
- Rigid Body Behavior
- Triangle Surface Mesher
- Topology Checking
- Pinch
- Loop Removal

The **Advanced** group of controls is inaccessible when an assembly (p. 367) meshing algorithm is selected.

Number of CPUs for Parallel Part Meshing

Number of CPUs for Parallel Part Meshing: Sets the number of processors to be used for parallel part meshing. Using the default for specifying multiple processors will enhance meshing performance on geometries with multiple parts. For parallel part meshing, the default is set to **Program Controlled** or **0**. This instructs the mesher to use all available CPU cores. The Default setting inherently limits 2 GB memory per CPU core. An explicit value can be specified between **0** and **256**, where **0** is the default. Refer to Parallel Part Meshing (p. 433) for more details.

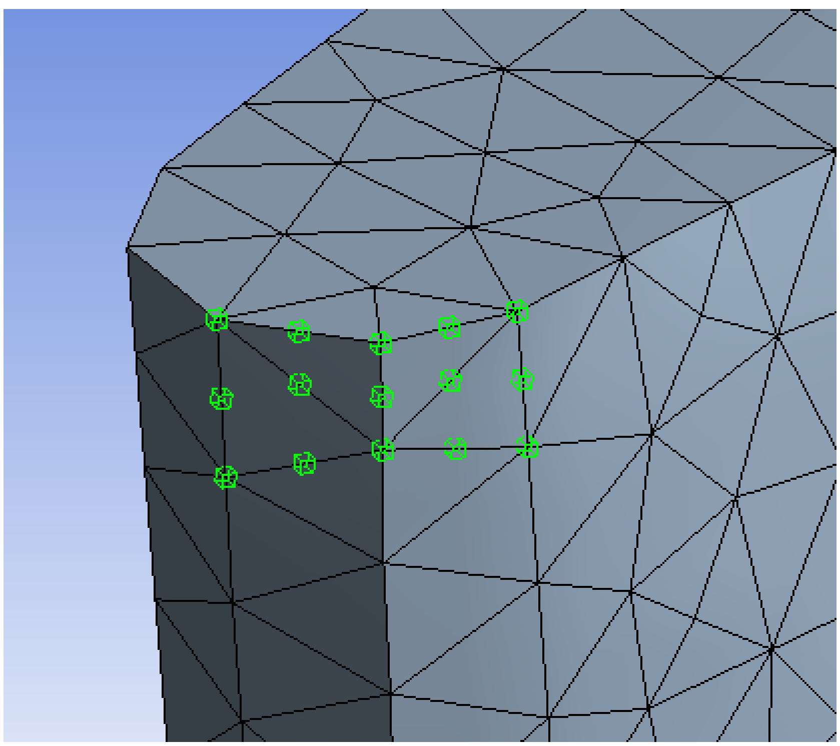
Straight Sided Elements

The **Straight Sided Elements** option (which is displayed when the model includes an enclosure from the DesignModeler application), specifies meshing to straight edge elements when set to **Yes**. You must set this option to **Yes** for Electromagnetic simulations.

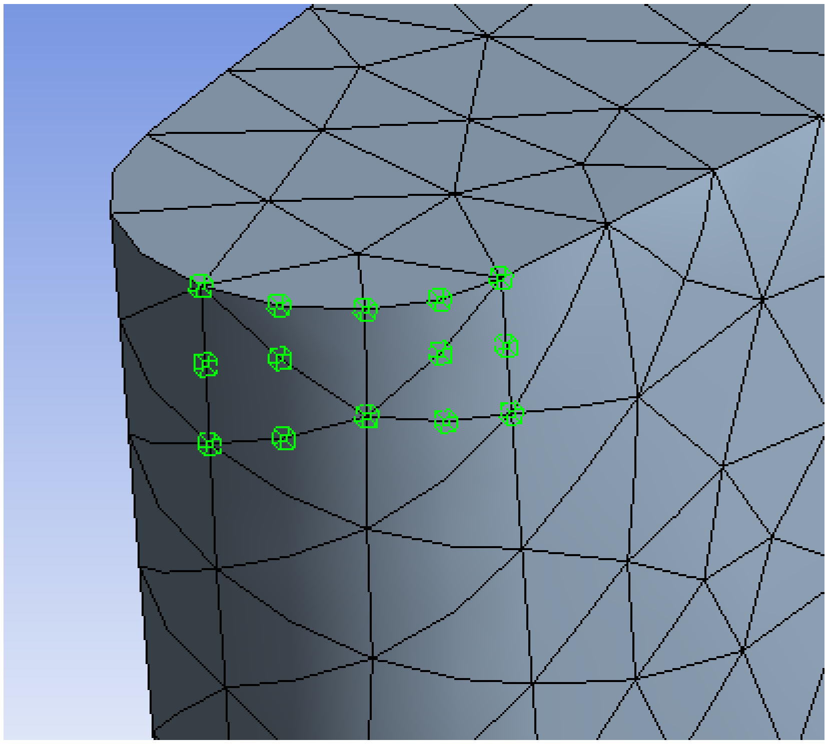
This option may affect the placement of midside nodes if the **Element Order** option is set to **Quadratic**.

Example 3: Midside Node Placement with Straight Sided Elements

In this example, the mesh is generated with straight sided elements with midside nodes.



Generating the mesh without straight sided elements results in midside nodes that capture the curvature of the model. If an element edge corresponds to geometry that is curved, the element's edge will be curved. Likewise, if an element edge corresponds to geometry that is straight, the element's edge will be straight.



Straight Sided Elements is not available if the **Element Order** option is set to **Linear**.

Rigid Body Behavior

The **Rigid Body Behavior** option determines whether a full mesh is generated for a rigid body, rather than a surface contact mesh. **Rigid Body Behavior** is applicable to all body types. Valid values for **Rigid Body Behavior** are **Dimensionally Reduced** (generate surface contact mesh only) and **Full Mesh** (generate full mesh). The default is **Dimensionally Reduced** unless the **Physics Preference** (p. 93) is set to **Explicit**. For more information, refer to Rigid Body Meshing (p. 424).

For Explicit, the default behavior is **Full Mesh**, but **Dimensionally Reduced** is also available. The differences between the two options when the **Physics Preference** is set to **Explicit** are highlighted in the table below.

Explicit behavior	Full Mesh (default)	Dimensionally Reduced
Rigid properties	From Mesh, computed in Explicit Solver	From Geometry, computed in Mechanical
Mesh	Tets/hexas for solids, Quad/tria for shells	Quad/tria for both solids and shells

Triangle Surface Mesher

The **Triangle Surface Mesher** control determines which triangle surface meshing strategy will be used by patch conforming meshers. In general, the advancing front algorithm provides a smoother size variation and better results for skewness (p. 140) and orthogonal quality (p. 142). This control is inaccessible when an assembly (p. 367) meshing algorithm is selected. The following options are available:

- **Program Controlled** - This is the default. The mesher determines whether to use the Delaunay or advancing front algorithm based on a variety of factors such as surface type, face topology, and defeatured boundaries.
- **Advancing Front** - The mesher uses advancing front as its primary algorithm, but falls back to Delaunay if problems occur.

Note:

If the mesher falls back to Delaunay, the edge mesh from the advancing front algorithm could still be used and in some rare cases could lead to meshing failures. Switching to the **Program Controlled** option could fix the issue as the starting edge mesh could be better.

The figures below illustrate the difference between the **Program Controlled** and **Advancing Front** options.

Figure 71: Triangle Surface Mesher = Program Controlled

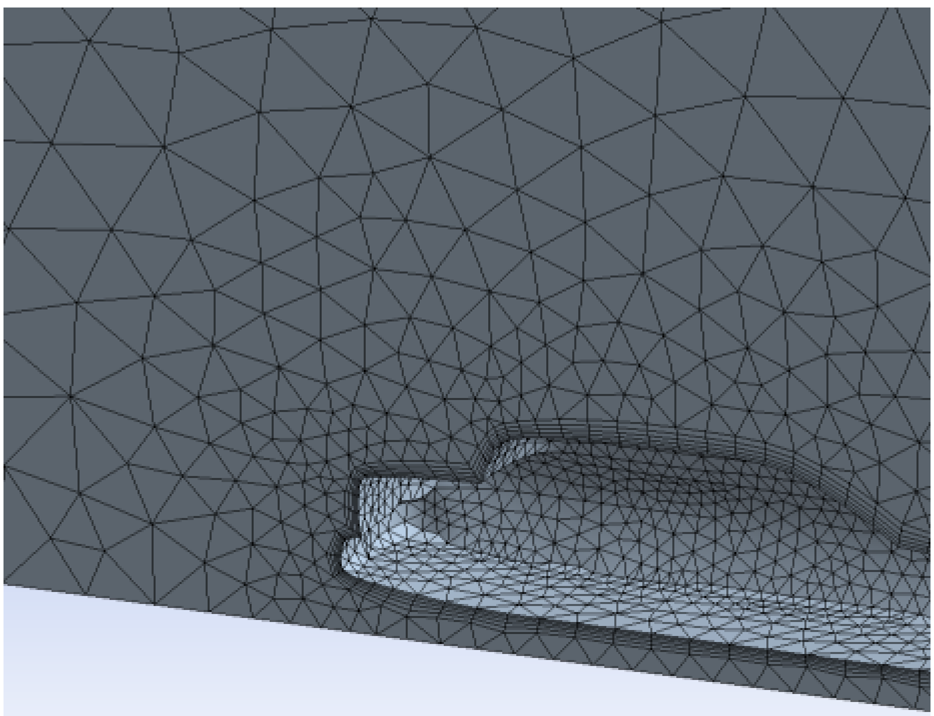
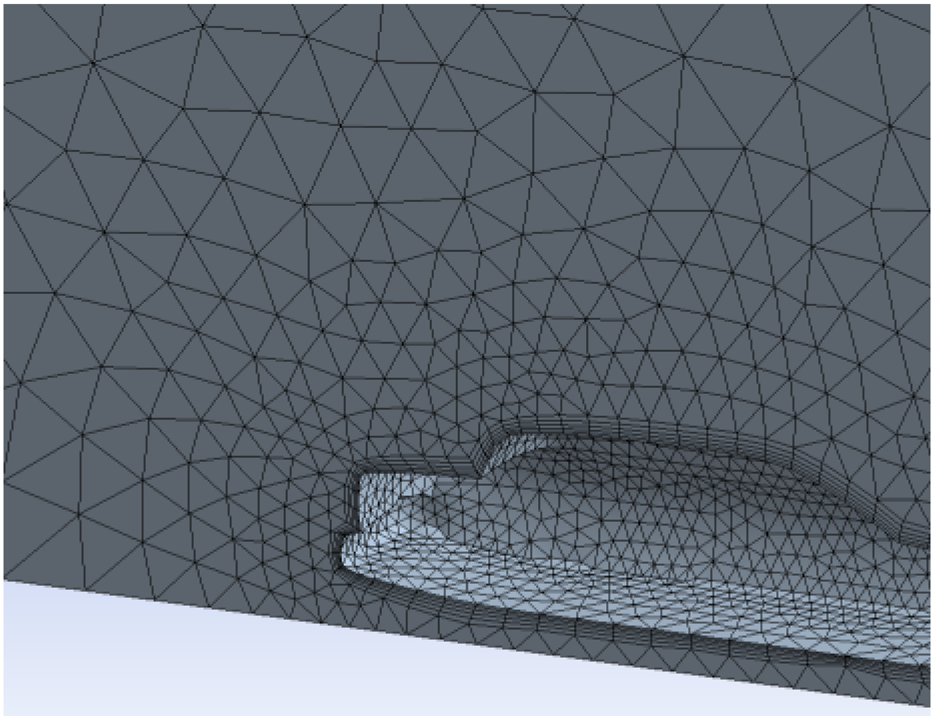


Figure 72: Triangle Surface Mesher = Advancing Front



Topology Checking

The **Topology Checking** option controls what happens when a user scopes an object (such as loads, boundary conditions, Named selections and so on) to geometry (bodies, faces, edges, and vertices) after the mesh has been generated. If **Topology Checking** is set to **Yes** (default), the software will check to see if the scoped geometry has mesh properly associated to it. If the associations are incorrect, the scoping of the object will force the mesh to be out of date. The mesh would need to be re-generated to get proper associations. If the associations are correct, the scoping is performed without any change to the mesh and the mesh stays up to date. Set **Topology Checking** to **No** to avoid the checks and always keep the mesh up to date.

Note:

- **Topology Checking** is defaulted to **Yes** as it is important to have proper associations between geometry and mesh in order to properly transfer the loads to the solver.
- The best practice is to define all loads and boundary conditions prior to meshing so that the topology is properly captured during meshing.

You can override the default of the **Topology Checking** control by setting the **Topology Checking** option on the **Options** panel (p. 317).

The following sections describe how the mesher handles Topology Protection:

Protecting Topology Defined Prior to Meshing

Protecting Topology Post Meshing

Protecting Topology Defined Prior to Meshing

Protected topology will result in having a better association between the mesh topology (nodes, faces, elements) and the geometry features (topology).That is,

- A geometry vertex should have a node associated to it.
- A geometry edge should have mesh nodes/edges associated to it.
- A geometry faces should have mesh nodes/faces/elements associated to it.
- A geometry body should have mesh nodes/elements associated to it.

Since boundary conditions are ultimately applied to the mesh, it is important to have proper associations to get proper loading. Thus, if protected topologies are ignored, an error or warning message may be issued by the mesher.

There are two levels of topology protection, namely **hard** and **soft**.

- Hard protected topology can be defined by scoping Named Selections and/or Contact region objects to geometry (bodies, faces, edges, and vertices) and setting the **Protected** option to **Yes**. This instructs the mesher to give higher priority to those geometry features. Thus, during mesh generation, the outer boundaries of a collection of hard protected topologies will be maintained. The mesher will return an error state and message if the outer boundaries of the hard protected topologies cannot be protected.
- Soft protected topology can be defined by scoping Named Selections to geometry (bodies, faces, edges, and vertices) and setting the **Protected** option to **Program Controlled**. For soft protected topologies, mesh-based defeaturing takes priority. Thus, the outer boundaries of soft protected topologies may be altered by the mesher. In such situations, mesher will return a warning message. You can right-click on the warning message and use the **Show Problematic Geometry** option to visualize problematic geometries. You should verify that the mesh is acceptable in case of defeaturing of outer boundaries of the soft protected topologies.

Setting up Hard Protected Topology

- For contacts, set **Protected** to **Yes**.
- For Named Selections, set **Protected** to **Yes**.

Setting up Soft Protected Topology

For Named Selections, set **Protected** to **Program Controlled**. When **Program Controlled** is selected, the scoped object(s) will be considered as soft protected. The scoped objects will not receive additional protection by the mesher, even if the Named Selection is used for boundary conditions, symmetry, other types of loads for the solver, as well as match controls and hard sizing controls for meshing. Mesh-based defeaturing will have priority and the outer boundaries of these topologies may be altered by the mesher.

Resuming Legacy Databases (prior to Release 19.0)

If you resume a legacy database (prior to Release 19.0) with protected topologies:

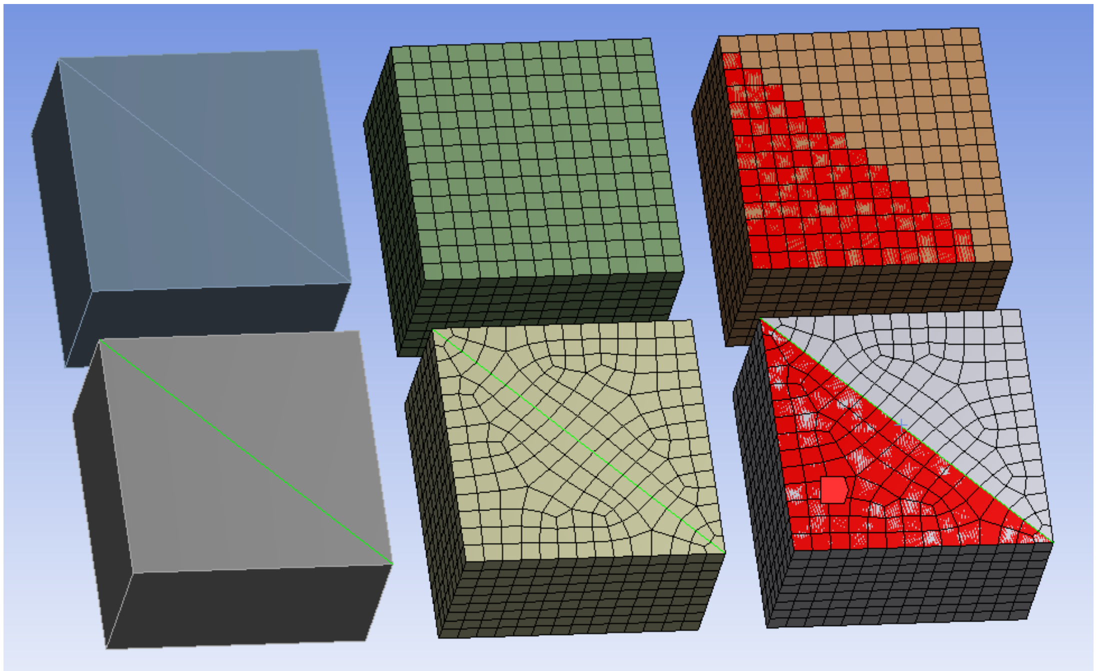
- Contacts are resumed with **Protected** set to **No**.
- Named selections are resumed with **Protected** set to **Program Controlled**.

Note:

- If surface bodies have differing thicknesses, the edges between the faces will be protected unless using **MultiZone Quad/Tri** mesh method. If using the **MultiZone Quad/Tri** mesh method, use **Preserve Boundaries** = **All**, or put the faces into separate named selections.
- If you use the **MultiZone** mesh method and set **Preserve Boundaries** (p. 228) to **All**, the **MultiZone** method will protect all boundaries.
- Virtual Topology can still be used with patch independent meshing; however, the boundaries of the virtual cells do not have to be protected unless the virtual topology is scoped to something. In other words, the virtual cells replace the underlying geometry but follow the same protection rules.
- You should apply loads/boundary conditions prior to meshing as it is the most robust process to get the proper mesh to respect the boundary conditions.
- The mesh is associated to the geometry even for bodies, faces, edges, and vertices that are not protected, but in such cases there may not be the one-to-one relationship that exists between the mesh and geometry for protected topology. For example, in Figure 73: Protecting Topology (p. 182) the highlighted edge is protected for the boxes in the bottom row, but not for the boxes in the top row.

  In the two boxes on the right, the faces with red mesh indicate face mesh that is associated with the triangular face on the left of the geometry. Due to the sharp features of the box, the side faces all have a one-to-one relationship. You can check the association by using the **Plot Elements Attached to Named Selections** option.

Figure 73: Protecting Topology



Protecting Topology Post Meshing

After meshing, if a new object is scoped to the geometry, the associations to the mesh may not match (see Figure 73: Protecting Topology (p. 182)). In this case, the new object is added to the list of protected topologies, but you may or may not want to re-mesh because the associations may be fine. For example, if you scope a control to one of the side faces of the box, the associated mesh will be fine. However, if you select one of the triangular faces, the associated mesh may be problematic.

You can set the Topology Checking (p. 179) control in the **Advanced** group to control the re-meshing behavior. By default, **Topology Checking** is set to **Yes**, which forces a re-mesh.

- If **Topology Checking** is set to **Yes** (default), the mesh goes out-of-date, because the state manager must revalidate that all scoped topology is associated properly as protected topology. If you then attempt to re-mesh, the software runs the topology checks and ensures all protected topology is respected. If the topology checks are successful, the mesh is validated but not re-meshed. If the topology checks are unsuccessful, the software re-meshes the geometry, treating the newly scoped objects as protected topology.

- If **Topology Checking** is set to **No**, the software does not check to ensure that the mesh is associated to the topology properly, so you must validate the associations manually if you have concerns.

Pinch

The **Pinch** feature lets you remove small features (such as short edges and narrow regions) at the mesh level in order to generate better quality elements around those features. The **Pinch** feature provides an alternative to Virtual Topology (p. 501), which works at the geometry level. The two features work in conjunction with one another to simplify meshing constraints due to small features in a model that would otherwise make it difficult to obtain a satisfactory mesh.

When **Pinch** controls are defined, the small features in the model that meet the criteria established by the controls will be "pinched out," thereby removing the features from the mesh. You can instruct the Meshing application to automatically (p. 189) create pinch controls based on settings that you specify, or you can manually (p. 286) designate the entities to be pinched. Pinch controls can be applied to solid and shell models, with certain restrictions as shown in the table below.

The **Pinch** feature is supported for the following mesh methods:

Volume Meshing:

- Patch Conforming (p. 200)
- Thin Solid Sweeping (p. 224)
- Hex Dominant Meshing (p. 222)

Surface Meshing:

- Quad Dominant (p. 245)
- All Triangles (p. 246)
- MultiZone Quad/Tri (p. 246)

The table below shows the types of model (solid or shell), mesh methods, pinch creation methods (auto or manual), and pinch behaviors that are supported for each type of pinch control.

Note:

With Ansys Workbench Release 16.0, post pinch behaviors are migrated into **Mesh Connections**. When you regenerate a mesh that was created using **Pinch Behavior** settings, the new mesh might report different results than the previous mesh.

Entities in Pinch Control: Primary > Secondary	**Model Type**		**Mesh Method**						**Pinch Creation Method**	
	Solid	**Shell**	**Patch Conforming**	**Thin Solid Sweeping**	**Hex Dominant**	**Quad Dominant**	**All Tri**	**MultiZone Quad / Tri**	**Auto Pinch**	**Manual Pinch**
Edge > Edge	X	X	X	X	X	X	X	X	X	X
Edge > Vertex	X	X	X	X	X	X	X	X		X

Entities in Pinch Control: Primary > Secondary	Model Type		Mesh Method						Pinch Creation Method	
	Solid	Shell	Patch Conforming	Thin Solid Sweeping	Hex Dominant	Quad Dominant	All Tri	MultiZone Quad / Tri	Auto Pinch	Manual Pinch
Vertex > Vertex	X	X	X	X	X	X	X	X	X	X
Face > Edge		X				X	X	X		X
Face > Vertex		X				X	X	X		X

Examples of a Mesh With and Without Pinch

The figures below illustrate the effect of pinch controls on the mesh. The first figure shows where the pinch controls have been defined in the model. The second and third figures show the meshes that are generated without the pinch controls and with the pinch controls, respectively.

Figure 74: Locations of Pinch Controls

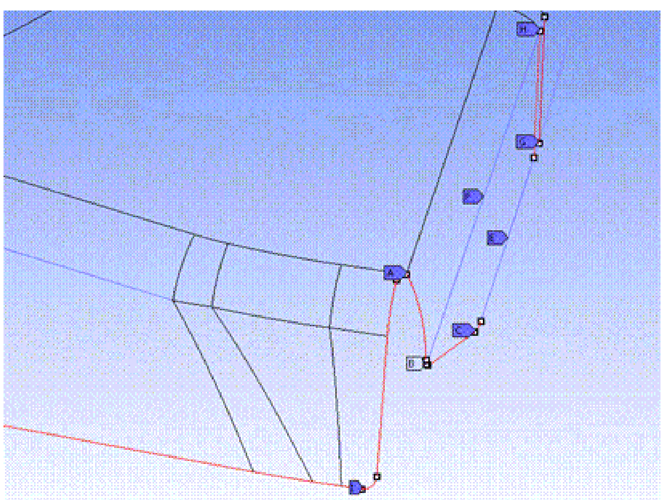


Figure 75: Mesh Generated Without Pinch Controls

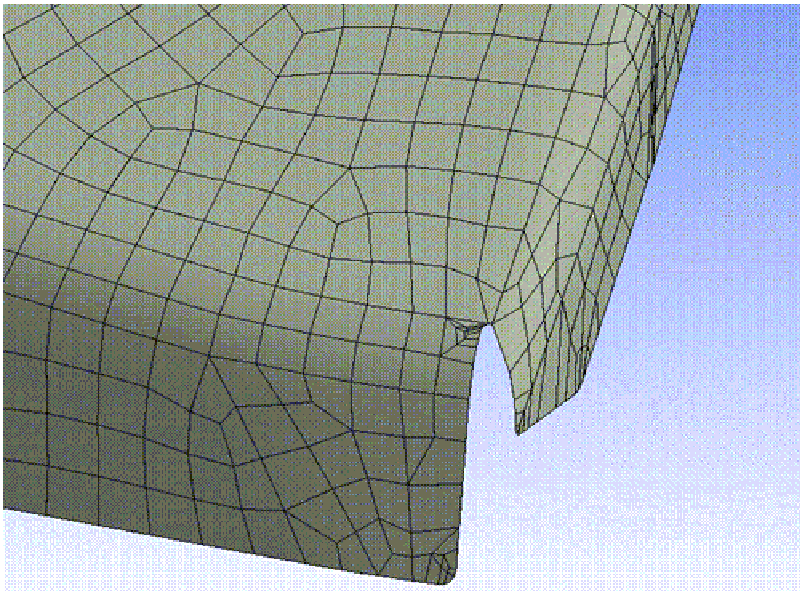
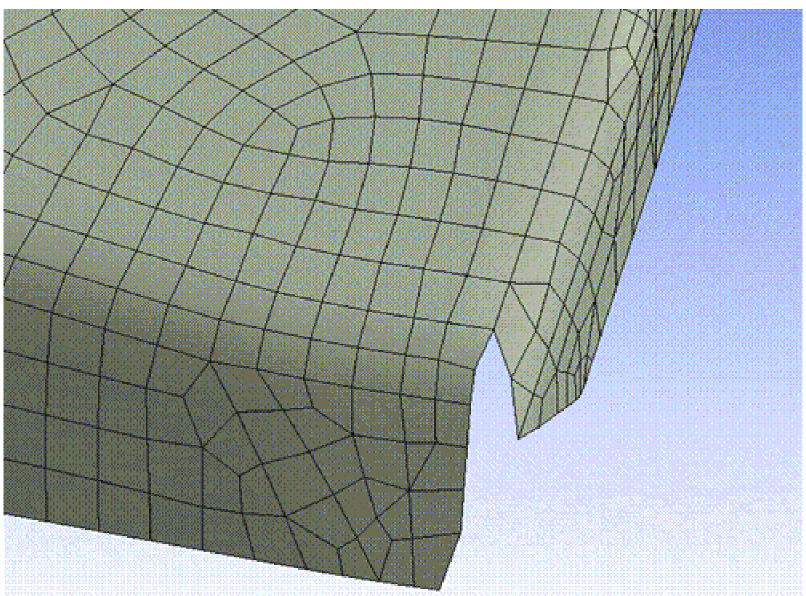


Figure 76: Mesh Generated With Pinch Controls



For More Information

For general information on applying pinch controls in combination with the various mesh method controls, refer to Meshing: Mesh Control Interaction Tables (p. 435).

Additional pinch topics include:

- Pinch Control Automation Overview
- How to Define Pinch Control Automation
- How to Define or Change Pinch Controls Manually
- Usage Information for Pinch Controls

Pinch Control Automation Overview

The **Advanced** group of global mesh controls appears in the Details View when the **Mesh** object is selected in the Tree Outline. By setting pinch options in the **Advanced** group, you can instruct the Meshing application to automatically generate pinch controls according to your settings. The Meshing application can generate the pinch controls based off shell thickness (for surface models only), or by identifying small features in the model that fall within a tolerance you specify.

Understanding the Automatic Pinch Control Algorithm

The Meshing application uses four major criteria for generating automatic pinch controls. These criteria (in order of importance) include:

1. Capture geometry of sheet bodies.

2. Edges adjacent to flat surfaces are primaries.

3. Cluster primaries together so that they are adjacent to one another.

4. All things being equal, longer edges are primaries.

Note:

- The primary geometry is the entity that retains the profile of the original geometry. The secondary geometry is the entity that changes in order to move towards the primary geometry. Depending on the tolerance, the pinch control will pinch out the entire secondary entity or only a portion of the secondary entity into the primary.

- The automatic pinch control algorithm supports only one primary for each pinch control.

- Once the automatic pinch control algorithm has paired two edges to use as a primary and a secondary in an automatic pinch control, the algorithm cannot use either of those same two edges as primary or secondary in any other automatic pinch control. For example, in the geometry shown in Figure 77: Automatic Pinch Control for Edges on Left; Manual Pinch Control Required for Edges on Right (p. 187), an automatic pinch control has been created for the annotated secondary (red) edges and the primary (blue) edge on the left side of the model. However, a pinch control will not be created automatically for the very similar configuration of edges on the right side of the model. For the Meshing application to be able to generate the mesh shown in Figure 78: Mesh Generated with Automatic Pinch Control and Manual Pinch Control on Similar Geometry (p. 187), a manual pinch control had to be created for the edges on the right.

Figure 77: Automatic Pinch Control for Edges on Left; Manual Pinch Control Required for Edges on Right

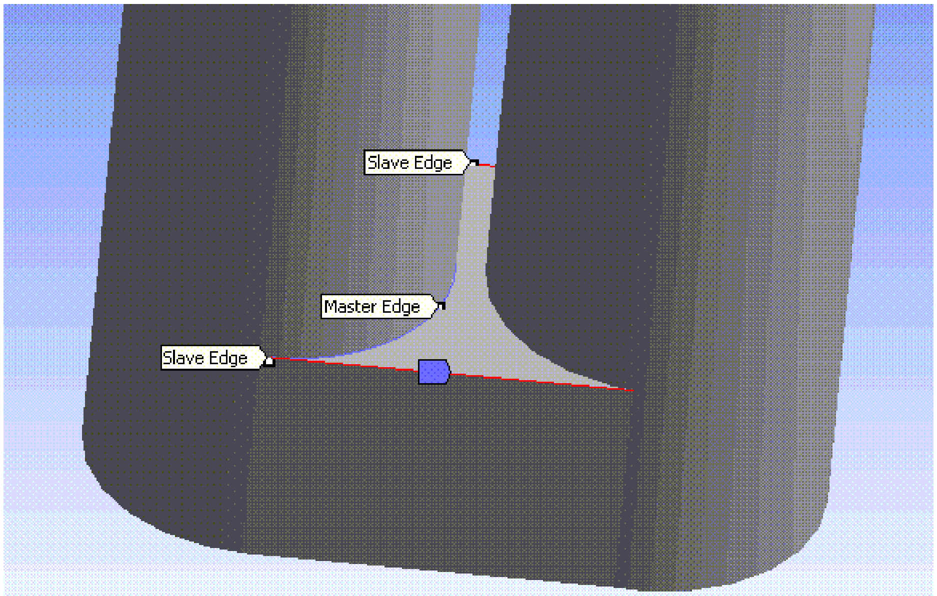
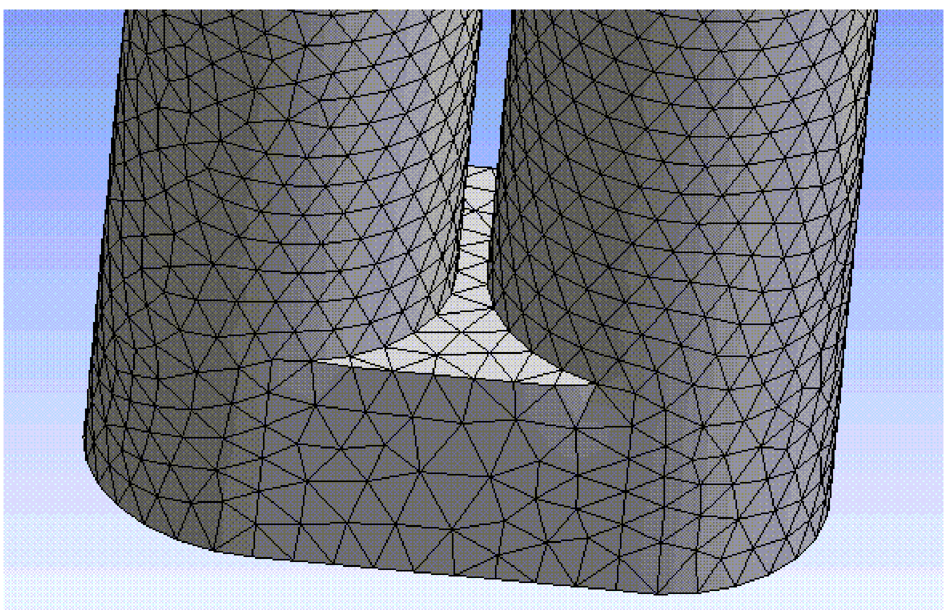


Figure 78: Mesh Generated with Automatic Pinch Control and Manual Pinch Control on Similar Geometry



The basic options for defining automatic pinch controls are described below.

Use Sheet Thickness for Pinch

The **Use Sheet Thickness for Pinch** control determines whether automatic pinch controls will be based on the shell thickness of the model, rather than on a specified pinch tolerance (p. 188). When the sheet thickness method of automatic pinch control is used, the pinch algorithm uses a pinch tolerance equal to 1/2 of the shell thickness.

Valid values are **Yes** and **No**. The default is **No**.

Note:

- The **Use Sheet Thickness for Pinch** control is available only for sheet (surface) models. If your model contains a combination of surface and solid bodies, the **Use Sheet Thickness for Pinch** control is available only if all solid bodies are suppressed (p. 486).
- You cannot use the sheet thickness method of automatic pinch control for a surface model that has no thickness defined or has a 0 (zero) thickness.

Pinch Tolerance

The **Pinch Tolerance** control allows you to specify a tolerance for the Meshing application to use when it generates automatic pinch controls. Vertex-vertex pinch controls will be created on any edge with a length less than the specified tolerance, and edge-edge pinch controls will be created on any face for which two edges are within proximity according to the specified tolerance. For the Meshing application to automate pinch control, you must specify a value for **Pinch Tolerance** unless **Use Sheet Thickness for Pinch** (p. 187) is set to **Yes**. Specify a value greater than **0.0**.

Note:

- The value that you specify for **Pinch Tolerance** should be smaller than the mesh size around the region in which the pinch control is being applied. For example, if an Edge Sizing control has been placed on an edge, a **Pinch Tolerance** value that is greater than that edge sizing may cause the mesher to fail.
- In general, the value that you specify for **Pinch Tolerance** should be greater than the value that appears in the **Sizing** > **Minimum Edge Length** (p. 117) field. Exceptions to this guideline include models without seam edges, such as elliptical prism, cylinder, and sphere models.
- Do not specify an overly high value for the **Pinch Tolerance** control. The **Pinch** feature allows the mesher to mesh over geometry features as if they were not there, and a tolerance that is set too high can cause inverted elements.
- When **Capture Curvature** and/or **Capture Proximity** is set to **Yes**, the default pinch tolerance is 90% of the value of **Curvature Min Size** (p. 108)/ **Proximity Min Size** (p. 110) (whichever is smaller). This differs from the tolerance used by the default mesh based defeaturing, refer to Mesh Defeaturing (p. 106) for details.

Generate Pinch on Refresh

The **Generate Pinch on Refresh** control determines whether pinch controls will be regenerated following a change made to the geometry (such as a change made via a DesignModeler application operation such as a merge, connect, etc.). If **Generate Pinch on Refresh** is set to **Yes** and you change the geometry, all pinch controls that were created automatically will be deleted and recreated based on the new geometry. If **Generate Pinch on Refresh** is set to **No** and you update the geometry, all pinch controls related to the changed part will appear in the Tree Outline but will be flagged as undefined.

Valid values are **Yes** and **No**. The default is **No**.

Note:

Only **Automatic** pinch controls are regenerated. That is, if a pinch control has a **Scope Method** of **Manual** (either because it was created manually or because you made a change to an **Automatic** pinch control), the pinch control will never be regenerated on refresh. See Changing Pinch Controls Locally (p. 290) for information about making changes to pinch controls.

How to Define Pinch Control Automation (p. 189) provides the steps for defining automatic pinch controls.

How to Define Pinch Control Automation

The following sections provide the steps for defining pinch control automation. Pinch can be automated based on either shell thickness or a user-defined tolerance.

Note:

Use of pinch control automation will delete all existing pinch controls that have a **Scope Method** of **Automatic** before creating the new pinch controls.

Defining Pinch Control Automation Based on Shell Thickness

This section describes the steps for defining pinch control automation based on shell thickness. This procedure applies to sheet (surface) models only.

To define pinch control automation based on shell thickness:

1. In the Details View of the **Mesh** folder, expand the **Advanced** group of controls.

2. Set **Use Sheet Thickness for Pinch** (p. 187) to Yes.

   Notice that the value of the **Pinch Tolerance** (p. 188) control changes to **Based on Sheet Thickness** and is grayed out.

3. Change the value of the **Generate Pinch on Refresh** (p. 188) control if desired.

4. Right-click the **Mesh** folder and select **Create Pinch Controls** from the context menu.

   A pinch control object is automatically inserted into the Tree for each region containing features that meet the criteria established by the pinch control settings. To display details about an individual pinch control, highlight it in the Tree and information about it appears in the Details View. For information about making changes to this information, refer to Changing Pinch Controls Locally (p. 290).

Defining Pinch Control Automation Based on a Specified Pinch Tolerance

This section describes the steps for defining pinch control automation based on a tolerance that you specify.

To define pinch control automation based on pinch tolerance:

1. In the Details View of the **Mesh** folder, expand the **Advanced** group of controls.

2. Specify a value for **Pinch Tolerance** (p. 188).

3. Change the value of the **Generate Pinch on Refresh** (p. 188) control if desired.

4. Right-click the **Mesh** folder and select **Create Pinch Controls** from the context menu.

   A pinch control object is automatically inserted into the Tree for each region containing features that meet the criteria established by the pinch control settings. To display details about an individual pinch control, highlight it in the Tree and information about it appears in the Details View. For information about making changes to this information, refer to Changing Pinch Controls Locally (p. 290).

How to Define or Change Pinch Controls Manually

As an alternative to defining pinch control automation (p. 189), you can define local controls to pinch scoped entities. You can also make changes to pinch controls, regardless of whether they were created automatically or manually. For details, refer to Pinch Control (p. 286) in the local mesh controls section of the Meshing help.

Usage Information for Pinch Controls

Remember the following information when using the **Pinch** feature:

- The **Pinch** feature works on faces, edges, and vertices only, bodies cannot be pinched. Refer to the table in Pinch (p. 182) for restrictions related to entity types.

- The automatic pinch control algorithm supports only one primary for each pinch control. In manual pinch controls, you can specify multiple faces or multiple edges to act as primaries, but only one vertex can act as primary.

- When defining manual pinch controls, using the same primary in more than one pinch control is supported. This is true for all types of manual pinch controls: edge-edge, edge-vertex, vertex-vertex, face-edge, and face-vertex. When multiple pinch controls use the same primary, the aggregate of the pinch controls is used to determine the pinch. Note that this behavior differs from that of other mesh controls when multiples are specified. With other mesh controls, the control that appears lowest in the Tree is honored.

- If there are hard size constraints on a primary, the pinch control will be skipped completely. If there are hard size constraints on secondaries, only the secondaries with the constraints will be skipped. In either case, a warning message will be issued.

- If your model contains multibody parts and you want pinch controls to operate on selected parts/bodies only, you must first suppress (p. 486) the parts/bodies that you do not want the pinch controls to apply to. Then follow the steps outlined in How to Define Pinch Control Automation (p. 189).

Alternatively you can use mesh connections, which supports automatic and manual mesh connections. With mesh connections, you can change the settings in the Details View of the **Connection Group** folder to automatically generate mesh connections on scoped geometry only. For more information, see Common Connections Operations for Auto Generated Connections in the Mechanical help.

- If the geometry fails to mesh correctly due to the pinched features, an error message is generated. To highlight the geometry that is responsible for the message, select the message, right-click, and select **Show Problematic Geometry** from the context menu.

- When a face contains an internal loop with a pinch control and the edges of the loop become a "single internal edge" due to pinching, the surface mesher may completely ignore the "single" edge (that is, the surface mesher may mesh over the edge). The reason that this may occur is that a pinch control never changes the topology of a model. When a surface mesher collects all boundary edge meshes before performing surface meshing, it considers the newly created "single" edge to be a regular edge rather than a hard edge, which most users would expect. As a result, all edge meshes along the "single" edge may be ignored.

- After pinch controls are generated:

  – If you highlight a pinch control in the Tree, the pinch region is flagged in the **Geometry** window. For more information, see the descriptions of **Primary Geometry** and **Secondary Geometry** pinch controls in Pinch Control Automation Overview (p. 186).

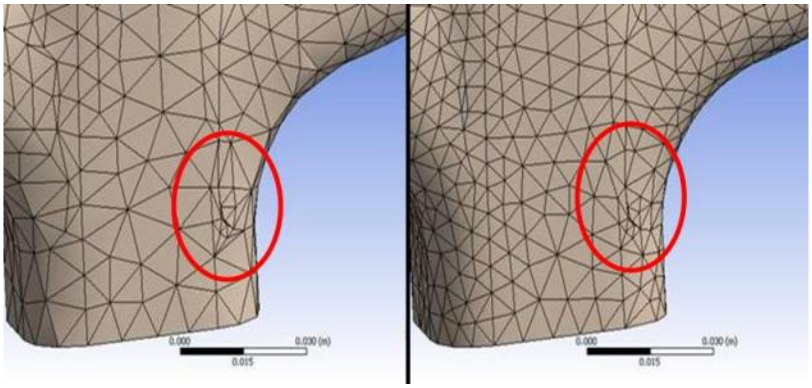
  – You can make changes to pinch controls whether they were generated automatically or created manually. To do so, in the **Mesh** folder, highlight the **Pinch** object that you want to change. As a result, the Details of the pinch appear in the Details View, where you can change its **Scope** and **Definition**. Making changes to a pinch that was generated automatically causes the value of the **Scope Method** control to change from **Automatic** to **Manual**. For details about defining or changing pinch controls manually, see Pinch Control (p. 286) in the local mesh controls section of the Meshing help.

- If a pinch control has a pinch tolerance defined for it that falls below one or more Hard (p. 262) scoped size controls, a warning will be issued. The warning will suggest that you either modify the pinch tolerance or remove any pinch control(s) in close proximity to the Hard size control(s) in question; otherwise, surface meshing may fail.

- There is no guarantee that features will be preserved when using pinch controls. For this reason, it is best practice to check the mesh where pinch controls have been defined close to features. If a problem exists in the mesh, flipping the primary and secondary entities will be sufficient to solve the problem in many cases.

- Pinch controls can be used for models involving multiple complications in one location (such as slivers, sharp angles, and short edges within the same pinch tolerance) as well as for models containing isolated problem spots. However, when used in combination with the Sizing Options (p. 89), pinch controls are best used for isolated problems. For example, refer to the meshes in the figure below, which show the results of applying pinch controls in combination with other sizing options. For the mesh on the left, a **Pinch Tolerance** (p. 188) of 3e-3 and a **Curvature Min Size** (p. 108) of 6.e-3 were specified. For the mesh on the right, a **Pinch Tolerance** of 3e-3 and a **Min Size** of 4.e-3 were specified. Neither is acceptable due to the presence of high aspect ratio triangles in the mesh. In such cases, the use of Virtual Topology or defeaturing within the DesignModeler application is recommended as an alternative to pinch.

Figure 79: Pinch Not Recommended for Models with Multiple Complications



- In a face-edge pinch control, the mesh on the edges within the specified tolerance is "snapped" to the primary face. You must choose the primary and secondaries in such a way that the elements on the face whose edges are defined as secondaries will be stretched onto the primary face. If the edges would be "squashed," no pinch will be created.
- When a face pinch control and a Face Meshing (p. 265) control are applied to the same face, the mesher attempts to generate a mapped mesh for the face. If the mesher cannot retain the mapped mesh pattern, it will generate a free mesh instead and issue a warning.
- When using a Face as the primary geometry, then the pinch control is applied post-processing and does not support mixed dimension meshing. When using an Edge as the primary geometry, then the pinch control is applied pre-processing, and is recommended in mixed dimension situations.
- If you apply a match control (p. 280) and a face-edge pinch control to the same topology, a warning is issued.
- Since **Pinch** objects cannot be duplicated, they cannot be used as template objects for the Object Generator.

Loop Removal

The **Advanced** group of global mesh controls appears in the Details View when the **Mesh** object is selected in the Tree Outline. The Meshing application automatically removes loops according to the criteria you specify for the loop removal options in this group. Prior to meshing, you can use the **Show Removable Loops** (p. 495) feature to preview the loops that will be removed according to the current settings.

The loop removal feature is supported for the following mesh methods:

Surface Meshing:

- Quad Dominant (p. 245)
- All Triangles (p. 246)

- MultiZone Quad/Tri (p. 246)

Note:

- The loop removal controls are passed to the MultiZone Quad/Tri method as described in MultiZone Quad/Tri Method Control (p. 246).
- If you are meshing with loop removal on (using the Quad Dominant or MultiZone Quad/Tri method), making changes to a loop after meshing (such as adding a load on a loop) invalidates the mesh and you will need to re-mesh. You should apply loads to the model before meshing when using these controls. Refer to Protecting Topology Defined Prior to Meshing (p. 180) for related information.

The options for defining loop removal are described below.

Sheet Loop Removal

The **Sheet Loop Removal** control determines whether loops will be removed (meshed over) by the mesher. When **Sheet Loop Removal** is set to **Yes**, the mesher removes any loop with a radius less than or equal to the value of **Loop Removal Tolerance**.

Valid values are **Yes** and **No**. The default is **No**.

Loop Removal Tolerance

The **Loop Removal Tolerance** control sets the tolerance for loop removal. Any loop with a radius less than or equal to the value of **Loop Removal Tolerance** will be meshed over by the mesher.

Specify a value greater than **0.0**.

Statistics Group

The **Statistics** group lets you view and request information about these options:

Nodes

Elements

Nodes

The **Nodes** option provides a read-only indication of the number of nodes in the meshed model. If the model contains multiple parts or bodies, you can view the number of nodes in an individual part or body by highlighting it under the **Geometry** object in the Tree Outline.

Elements

The **Elements** option provides a read-only indication of the number of elements in the meshed model. If the model contains multiple parts or bodies, you can view the number of elements in an individual part or body by highlighting it under the **Geometry** object in the Tree Outline.

Meshing: Local Mesh Controls

Local mesh controls are available when you highlight a **Mesh** object in the tree and choose a tool from either the **Mesh Control** drop-down menu, or from first choosing **Insert** in the context menu (displayed when you right-click a **Mesh** object). You can specify the scoping of the tool in the tool's Details View under **Method** to either a **Geometry Selection** or to a **Named Selection**.

Note:

Be aware of the following items regarding mesh control tools:

- The Object Generator enables you to make one or more copies of a template object, scoping each to a different piece of geometry. When defining mesh controls, you can use the Object Generator to make copies of a template mesh control, which may reduce the necessity to manually define multiple related mesh controls. For details, refer to Generating Multiple Objects from a Template Object in the Mechanical help.

- For most mesh controls, the latest control that you add on a particular geometry overrides any prior controls that you already have added on that geometry. For example, if you apply a Sizing control setting of 0.5 to faces A,B,C then apply a setting of 1.0 to face B, faces A and C will retain the 0.5 setting, but the setting for face B will be 1.0. This is also useful when you want to force sweep many bodies of a multibody part and only tet mesh one or specify special sweeping options on one. For example, you can select all 1000 parts and then override one or 10 part(s) instead of picking 999 (990) and then selecting one (10).

  Exceptions include the MultiZone Quad/Tri, MultiZone, and All Tetrahedrons - Patch Independent controls. For information about how these controls interact with other controls, refer to Meshing: Mesh Control Interaction Tables (p. 435), Interactions Between Mesh Methods (p. 435), and Interactions Between Mesh Methods and Mesh Controls (p. 438).

- If you suppress a mesh control tool, the Suppress symbol appears ("x" adjacent to the name of the tool) and **Suppressed** is set to **Yes** in the Details View of the tool. Situations can occur when you do not suppress a mesh control tool, and the Suppress symbol appears adjacent to the tool but **Suppressed** is set to **No** in the Details View of the tool. In these situations, refer to the mesh control's **Active** read-only field for the reason why the tool is suppressed. Examples are a control applied to a uniform surface body mesh (not supported), a control scoped to suppressed geometry, or a **Contact Sizing** control scoped to a suppressed **Contact Region**.

The following local mesh controls are available:

Method Control

Mesh Grouping Control

Sizing Control

Method Control

The **Method** control is valid only for a body. The default value selects meshing methods that provide a successful automated mesh. By default, the application attempts to use auto Fing for solid models and quadrilateral element generation for surface body models. If a model is not sweepable, the **Patch Conforming** mesher under **Tetrahedrons** is used.

To set the values for the **Method** control, click **Mesh** on the Tree Outline, right-click to view the menu, and select **Insert > Method**. You can also click **Mesh** on the Tree Outline, select the **Mesh Control** button on the Context Toolbar, and select **Method**.

Note:

The **Method** control is not supported for assembly (p. 367) meshing algorithms.

In the Details View for the scoped local method, you can set **Method** based on whether you want to apply the method to a solid body or a surface body. For more information, refer to:

Method Controls and Element Order Settings

When setting the Method control (p. 196) to a scoped body, you can control whether meshes are to be created on the scoped body with midside nodes or without midside nodes by using the **Element Order** setting under **Definition** in the Details View. When setting the **Element Order** option for a scoped body, choices include **Use Global Setting**, **Linear**, and **Quadratic**.

If you select **Use Global Setting**, **Element Order** will be handled as dictated by the global **Element Order** option (p. 96). The remaining choices—**Linear** and **Quadratic**—have the same descriptions as their counterparts under the global **Element Order** option (p. 96). Setting **Element Order** to **Linear** or **Quadratic** for a scoped body will override the setting of the global **Element Order** option.

If the **Element Order** is set to **Quadratic**, and if **Straight Sided Elements** is set to **No**, the midside nodes will be placed on the geometry so that the mesh elements properly capture the shape of the geometry. However, if the location of a midside node might affect the mesh quality, the midside node may be relaxed to improve the element shape. Therefore, some midside nodes might not follow the shape of the geometry precisely.

For more information about how the **Straight Sided Elements** control affects midside nodes, see Straight Sided Elements (p. 176).

Mixed Order Meshing

Mixed order meshing is supported across bodies for the following mesh methods:

- For solid meshing:
  - Patch Conforming Tetrahedron (p. 200)
  - Patch Independent Tetrahedron (p. 200)
  - MultiZone (p. 228)
  - General Sweep (p. 323)
  - Thin Sweep (p. 330)
  - Hex Dominant (p. 222)
  - Cartesian (p. 236)
- For surface meshing:
  - Quad Dominant (p. 245)
  - All Triangles (p. 246)
  - MultiZone Quad/Tri (p. 246)

This means that when scoping one of these mesh methods to bodies in a multibody part, you can set the **Element Order** option to **Quadratic** (resulting in higher order elements) for some bodies and to **Linear** (resulting in lower order elements) for others.

Mixed order meshing is supported whether you are performing Selective Meshing (p. 404) or meshing all of the bodies in the part at the same time. The behavior, and your resulting mesh, is dependent on the meshing order:

- For *simultaneous* meshing, the Quadratic bodies generally have precedence at the interface. In such cases, all of the elements in a lower order body that are adjacent to a higher order body will be higher order elements, thereby creating one layer of quadratic elements at the interface face. These elements will be higher order at the interface face but with dropped midside nodes where adjacent to the linear elements in the mesh.

An exception occurs when a solid body and a sheet body share an interface. In this case, the solid body takes precedence and will be meshed first using its defined element order. Then the sheet body is meshed with midside nodes handled as described for **Selective Meshing**, below.

- For Selective Meshing (p. 404), the order in which you mesh the bodies determines the precedence. If you first mesh a linear body followed by meshing an adjacent quadratic body, then the linear body has precedence. Elements in the quadratic body that are adjacent to the linear body will be lower order elements. Midside nodes will be dropped from the quadratic elements at the interface face.

  If you first mesh a quadratic body followed by meshing an adjacent linear body, then the quadratic body has precedence. Elements in the linear body that are adjacent to the quadratic body will be higher order elements. Midside nodes will be added to the linear elements at the interface face.

The figures below illustrate an example of mixed order meshing. To obtain the mesh shown in Figure 80: Mixed Order Meshing of a Multibody Part (p. 198), the global **Element Order** option was set to **Quadratic**, resulting in a mesh of quadratic tet elements for the topmost body. The sweep method was applied to the remaining bodies, with the **Element Order** option set to **Linear** on the **Sweep Method** (p. 223) control. This resulted in a mesh of primarily linear hex/wedge elements for the swept bodies, with the hex/wedge elements that are attached to the common interface being mixed order (see Figure 81: Mixed Order Elements (p. 199)).

Figure 80: Mixed Order Meshing of a Multibody Part

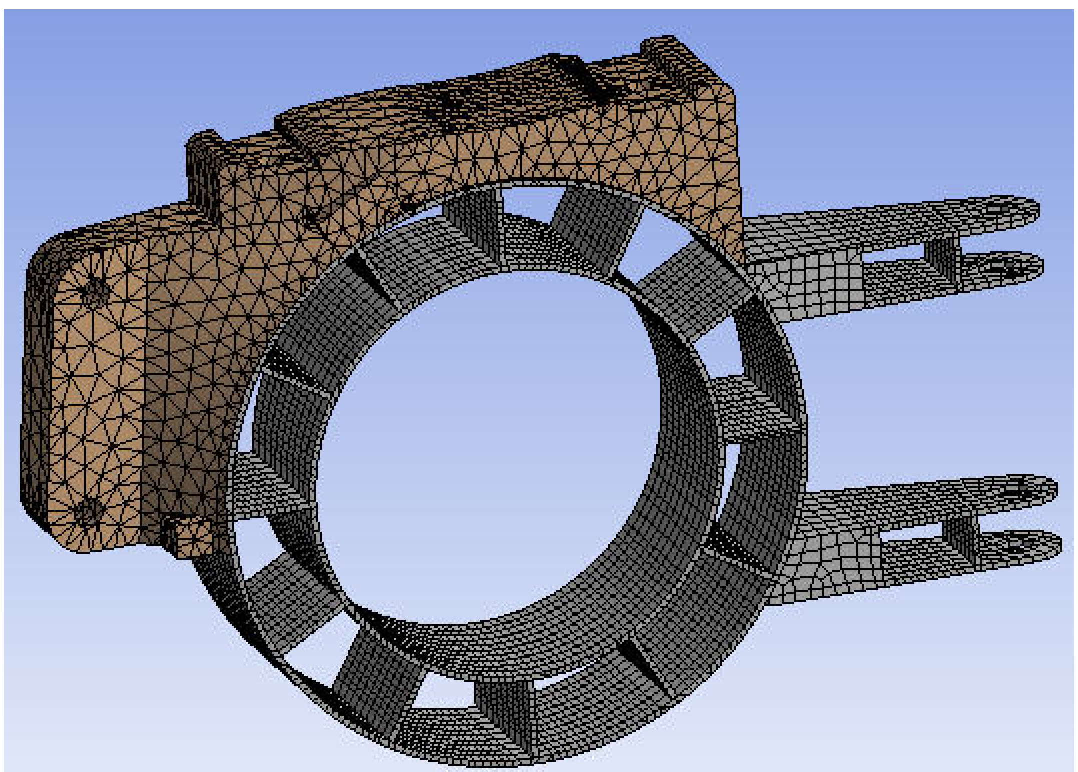
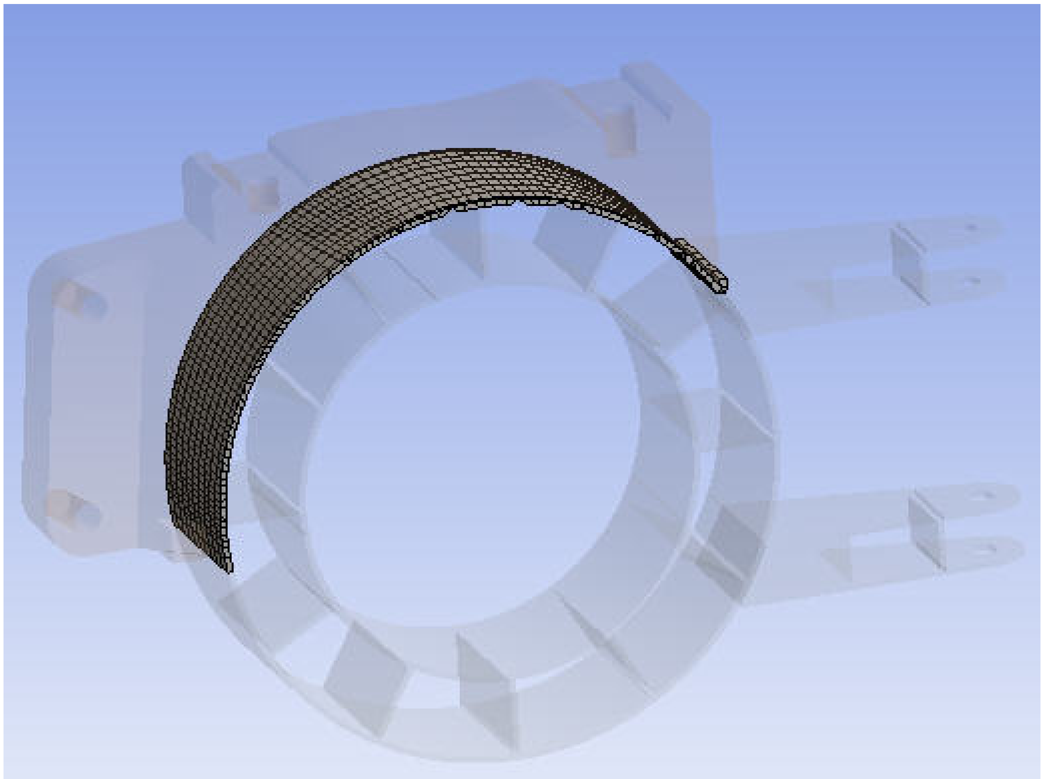


Figure 81: Mixed Order Elements (p. 199) shows the mixed order hex/wedge elements that are attached to quadratic pyramid elements at the interface. On the **Mesh Metrics** (p. 123) bar graph, mixed order elements are displayed as quadratic element types.

Figure 81: Mixed Order Elements



Setting the Method Control for Solid Bodies

The options described below are available for solid bodies.

- Automatic Method Control
- Tetrahedrons Method Control
- Hex Dominant Method Control
- Sweep Method Control
- MultiZone Method Control
- Cartesian Method Control
- Layered Tetrahedrons Method Control
- Particle Method

Automatic Method Control

By default, the application uses the **Automatic Method** control, which attempts to use sweeping (p. 223) for solid models and quadrilateral element generation for surface body models. If solid bodies can't be swept, the body is meshed with the **Patch Conforming Tetrahedron** (p. 200) mesher.

You can preview the bodies that can be swept meshed by right-clicking **Mesh** on the Tree Outline and choosing **Show>Sweepable Bodies** from the context menu.

Scoping a mesh method control (**Sweep** or **MultiZone**) is a way to force a body to be meshed with **Sweep** or **MultiZone**. To use the MultiZone mesh method in place of **Sweep**, turn on **Tools>Options>Meshing>Meshing: Use MultiZone for Sweepable Bodies**. When using **MultiZone** in place of **Sweep**, Sweepable bodies (p. 323) are meshed with **MultiZone**.

Tetrahedrons Method Control

If you select the **Tetrahedrons** method, an all tetrahedral mesh is created. An **Algorithm** setting is displayed allowing you to choose how the tetrahedral mesh is created based on your choice of one of the following options:

Patch Conforming Algorithm for Tetrahedrons Method Control

The **Patch Conforming** Tetra mesh method is a Delaunay tetra mesher with an advancing-front point insertion technique used for mesh refinement. The Patch Conforming Tetra mesh method provides:

- Support for 3D inflation (p. 145)
- Built-in pyramid layer for conformal quad-tet transition (p. 423)
- Built-in growth and smoothness control. The mesher will try to create a smooth size variation based on the specified growth factor.
- Removal of features under a certain size (Mesh Defeaturing (p. 106)).

Remember the following information when using the Patch Conforming Tetra mesh method:

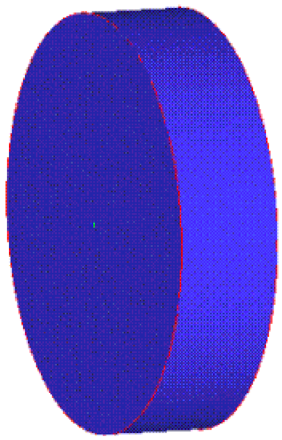
- For problematic geometry given by the Patch Conforming Tetra mesher or regions that self intersect, you can possibly remedy the problem by adding an extremely large face sizing control in the region of the self intersection. However, best practice would be to remove the problematic geometry in the DesignModeler application or your CAD system.
- When Patch Conforming Tetra meshing (with **Physics Preference** (p. 93) of **CFD**) fails due to lack of available memory, an error message will be issued. However, this error message will not identify insufficient memory as the cause. Because meshing will stop before the memory limit is reached, you may not notice any unusual behavior.
- For information about the **Element Order** option, refer to Method Controls and Element Order Settings (p. 196).

Patch Independent Algorithm for Tetrahedrons Method Control

The Patch Independent mesh method for tetrahedrons is based on the following spatial subdivision algorithm: This algorithm ensures refinement of the mesh where necessary, but maintains larger elements where possible, allowing for faster computation. Once the "root" tetrahedron, which encloses the entire geometry, has been initialized, the Patch Independent mesher subdivides the root tetrahedron until all element size requirements (that is, the prescribed local mesh sizes) are met.

At each subdivision step, the edge length of the tetrahedron (=size) is divided by 2. This means that the prescribed sizes should all differ by factors that are an integer power of 2. The size of the root tetra is set to the smallest given size multiplied by 2^n. All other prescribed sizes are approximated by subdividing the root tetra. Refer to the series of figures below, which illustrate the process that is followed by the Patch Independent tetra mesher.

Figure 82: Geometry Input to Patch Independent Tetra Mesher



At this point, the Patch Independent tetra mesher balances the mesh so that elements sharing an edge or face do not differ in size by more than a factor of 2.

Figure 83: Full Tetrahedron Enclosing the Geometry

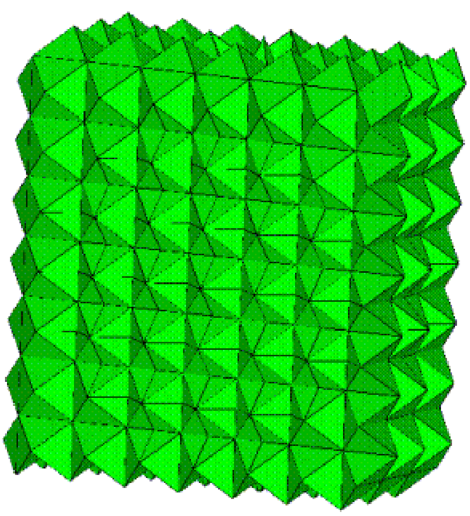


Figure 84: Full Tetrahedron Enclosing the Geometry in Wire Frame Mode

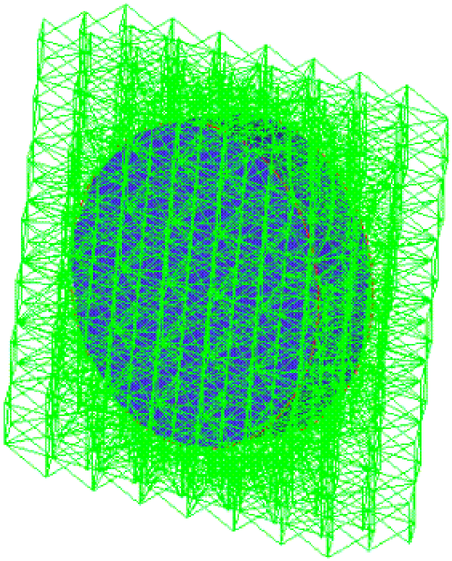
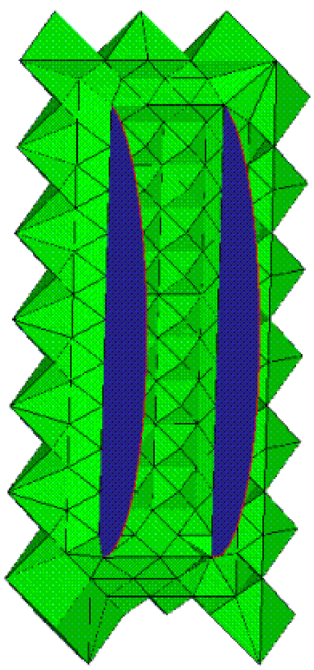


Figure 85: Cross-Section of the Tetrahedron



After this is done, the Patch Independent tetra mesher makes the mesh conformal; that is, it guarantees that each pair of adjacent elements will share an entire face. The mesh does not yet match the given geometry, so the mesher next rounds the nodes of the mesh to the prescribed points, prescribed curves, or model surfaces. The Patch Independent tetra mesher then "cuts away" all of the mesh, which cannot be reached by a user-defined material point without intersection of a surface.

Figure 86: Mesh After Capture of Surfaces and Separation of Useful Volume

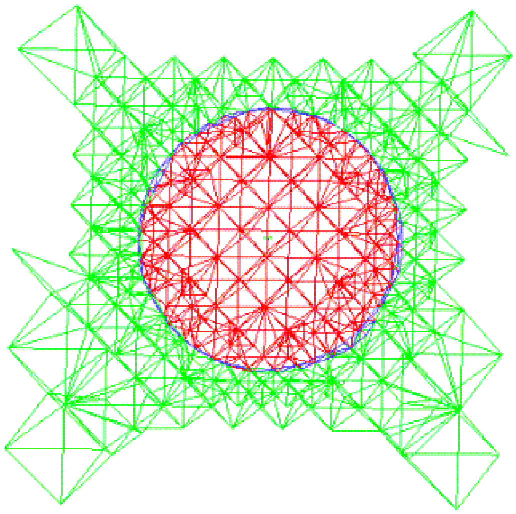
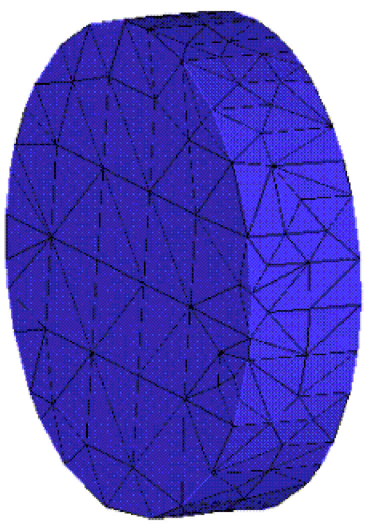
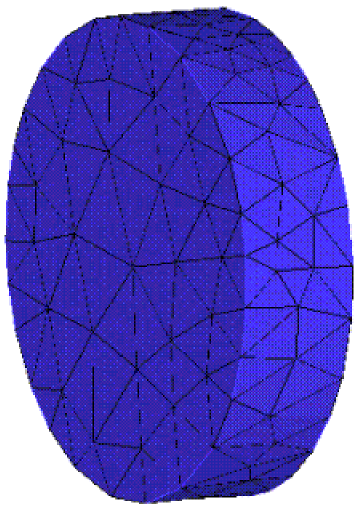


Figure 87: Final Mesh Before Smoothing



Finally, the mesh is smoothed by moving nodes, merging nodes, swapping edges and in some cases, deleting bad elements.

Figure 88: Final Mesh After Smoothing



Note:

The **Patch Independent Tetrahedrons** method is being deprecated and will be removed in future releases.

The **Patch Independent** mesh method includes the following settings:

- **Element Order** - Refer to Method Controls and Element Order Settings (p. 196).
- **Defined By** - Choices are **Max Element Size** and **Approx Number of Elements**.
- **Max Element Size** - The size of the initial element subdivision. The default value depends on the sizing options selected:
  - If **Use Adaptive Sizing** is set to **No**, the default value of **Max Element Size** is inherited from the global **Max Size** (p. 105) value.
  - If **Use Adaptive Sizing** is set to **Yes**, the default value of **Max Element Size** is inherited from the global **Element Size** (p. 98) value.

  In either case, you can change the value if you want to apply a specific value locally. In such cases, the maximum size comes from the larger value of the global controls (that is, **Max Size** or **Element Size**, as described above) *OR*, the largest scoped body size or face size that Patch Independent is also scoped to. A scoped edge size is not respected if it is larger than either the global size or the size on an attached face.

  With the Patch Independent mesh method, scoped body sizing is supported as follows:

  - If a local body size is defined and it is smaller than the global maximum size, the scoped body size will be assigned inside the volume.
  - If the global maximum size is smaller than any scoped body, face or edge sizing, the global maximum size (**Element Size** when **Use Adaptive Sizing** is set to **Yes**; **Max Size** when **Use Adaptive Sizing** is set to **No**) will be changed to be the same as the largest

sizing within the mesher. For example, if Patch Independent is defined on two bodies, and the setup is as follows:

→ Global Max Size = 4

→ Local body size scoped to Body1 = 8

→ No local body size is scoped to Body2
The Patch Independent maximum size will be 8, and the global Max Size of 4 will be used for the sizing of Body2.

Note:

The maximum element size inside the volume of Body2 could grow to 8. Because setting local sizings affects the largest element size in the model, you should avoid setting local sizes that are larger than the global maximum size.

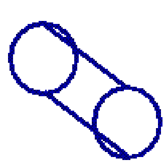
- **Approx Number of Elements** - Prescribes an approximate number of elements for the mesh. The default is 5.0E+05. Specifying a prescribed number of elements for the Patch Independent method is applicable only if the method is being applied to a single part.

- **Feature Angle** - Specifies the minimum angle at which geometry features will be captured. If the angle between two faces is less than the specified **Feature Angle**, the edge between the faces will be ignored, and the nodes will be placed without respect to that edge. If the angle between two faces is greater than the **Feature Angle**, the edge should be retained and mesh aligned and associated with it (note the edge could be ignored due to defeaturing, and so on). You can specify a value from 0 (capture most edges) to 90 (ignore most edges) degrees or accept the default of 30 degrees.

- **Mesh Based Defeaturing** - Ignores edges based on size. **Off** by default. If set to **On**, a **Defeature Size** field appears where you may enter a numerical value greater than 0.0. By default, the value of this local **Defeature Size** field is the same as the global Defeature Size (p. 106). If you specify a different value here, it will override the global value. Specifying a value of 0.0 here resets the tolerance to its default. If multiple Patch Independent tetra mesh method controls are defined with different tolerances, the smallest tolerance is respected.

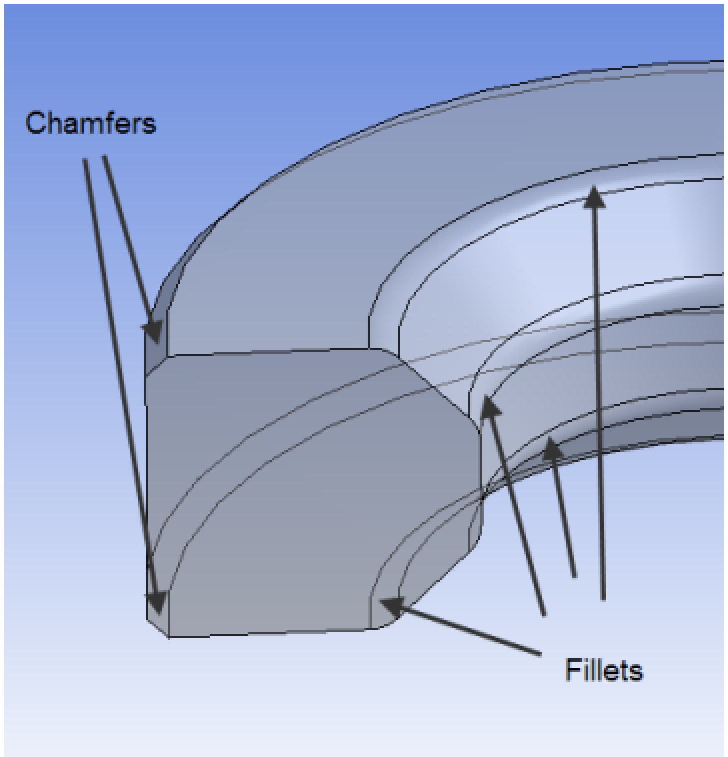
  There are several basic cases, including the following:

  – A small hole with a diameter smaller than the tolerance as shown below.

  No edges are dropped. You should defeature manually in this case.

  – Two approximately parallel spaced edges (fillet or chamfer), as shown below.



To determine whether a face is a fillet/chamfer, the Patch Independent mesher evaluates the face's geometric features. To be considered a fillet/chamfer:

- A face must be at least twice as long as it is wide.
- A fillet/chamfer has either three or four sides (that is, two long sides and one or two short sides), all with angles <= 135 degrees.

In the case of a fillet, which is a curved or rounded face, the angle between the fillet and a face attached to one of its long sides is 0 degrees (not 180 degrees). In contrast, a chamfer is a planar face and the angle between the chamfer and a face attached to one of its long sides is larger than 0 degrees.

For defeaturing of fillets/chamfers, the mesher considers the fillet/chamfer face as well as the faces adjacent to it (the faces attached to its long sides). The dihedral angles between these faces are evaluated to determine whether the attached edges of adjacent faces will be respected (that is, whether nodes will be placed with respect to the edges at the long sides of the fillet/chamfer).

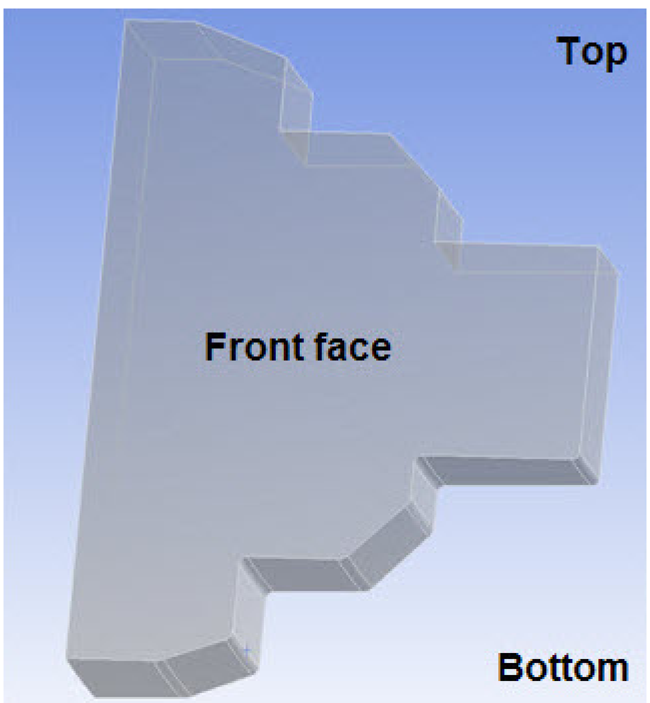
There are three dihedral angles occurring at a fillet/chamfer:

- One dihedral angle occurs between the two faces "touching," or adjacent to, the fillet/chamfer face. When this angle is compared with the **Feature Angle**, the angle is measured between the face normals at the imaginary edge where the two faces (virtually) meet.
- Two dihedral angles occur between the fillet/chamfer face and the respective faces "touching," or adjacent to, the two long sides of the fillet/chamfer. The angles are evaluated as the angles between the face normals at the common edge of the fillet/chamfer and the attached face.

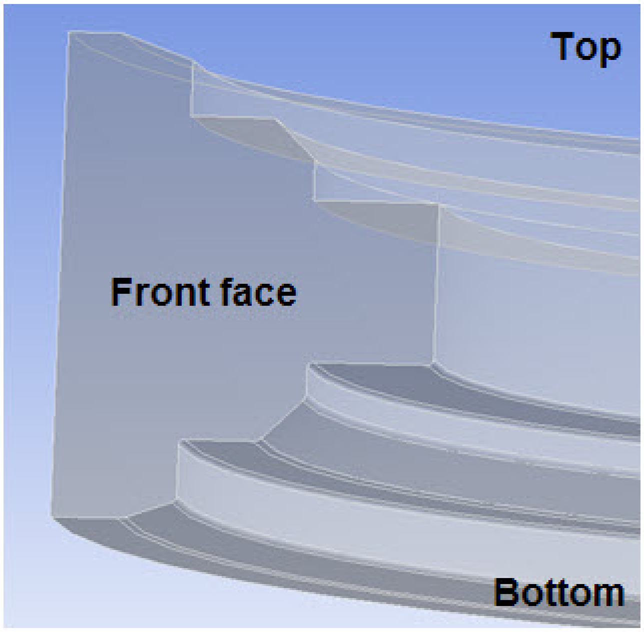
Defeaturing occurs as follows:

- If the angle between the two faces adjacent to the fillet/chamfer face is greater than the **Feature Angle**, and the angles between the fillet/chamfer face and the faces attached to its long sides are less than the **Feature Angle**, and the minimum fillet/chamfer width is *greater than* the **Defeature Size**, both long sides/edges are respected.
- If the angle between the two faces adjacent to the fillet/chamfer face is greater than the **Feature Angle**, and the angles between the fillet/chamfer face and the faces attached to its long sides are less than the **Feature Angle**, and the minimum fillet/chamfer width is *less than* the **Defeature Size**, only one long side/edge is respected.
- If only one angle between the fillet/chamfer face and the faces attached to its long sides is greater than the **Feature Angle**, only one long side/edge is respected.
- If none of the angles are greater than the **Feature Angle**, none of the long sides/edges are respected.

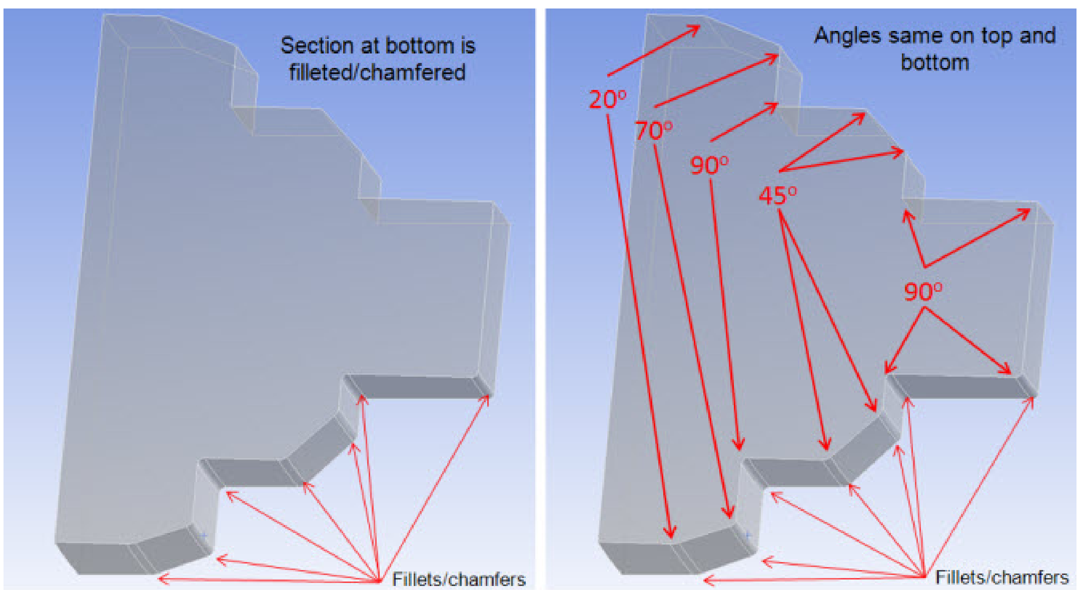
The following series of figures illustrates fillet/chamfer detection. In this example, a cross-section is revolved. The top and bottom of the section are identical, except the bottom has fillets/chamfers at each corner and the top does not. Because the definition of a fillet/chamfer is somewhat general, two cases are presented, each with a different angle of revolution. The angle of revolution is 5 degrees in the first case, as shown below. In this case, only the small faces fit the criteria of fillets/chamfers.



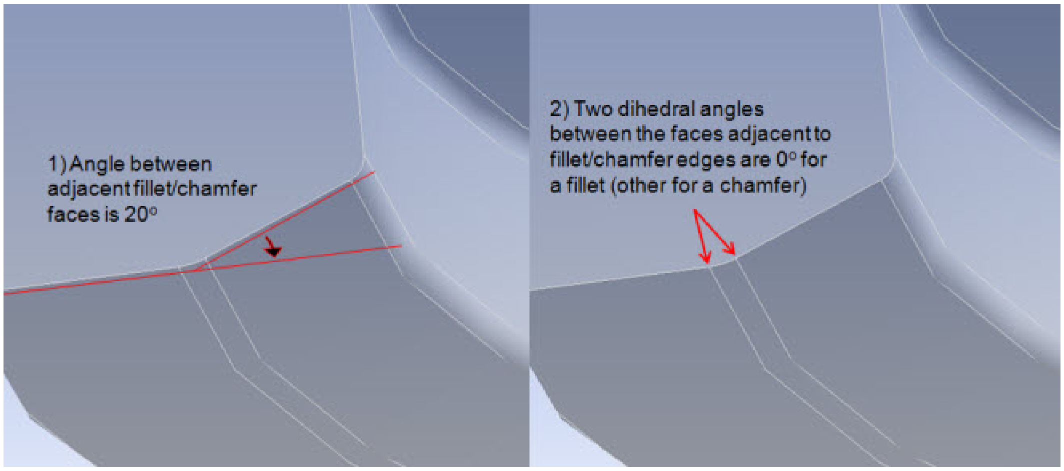
In the second case, the angle of revolution is 180 degrees, as shown below. In this case, all faces fit the criteria of fillets/chamfers, except for the front/back faces of the extrusion.



The figure below shows the angles that are considered for fillet/chamfer detection, and the small faces that are found to be fillets/chamfers.



As described earlier, the angles that are considered for a given fillet/chamfer are 1) the angle between adjacent fillet/chamfer faces and 2) the two angles attached to the fillet/chamfer. Notice the angles in the figure below.

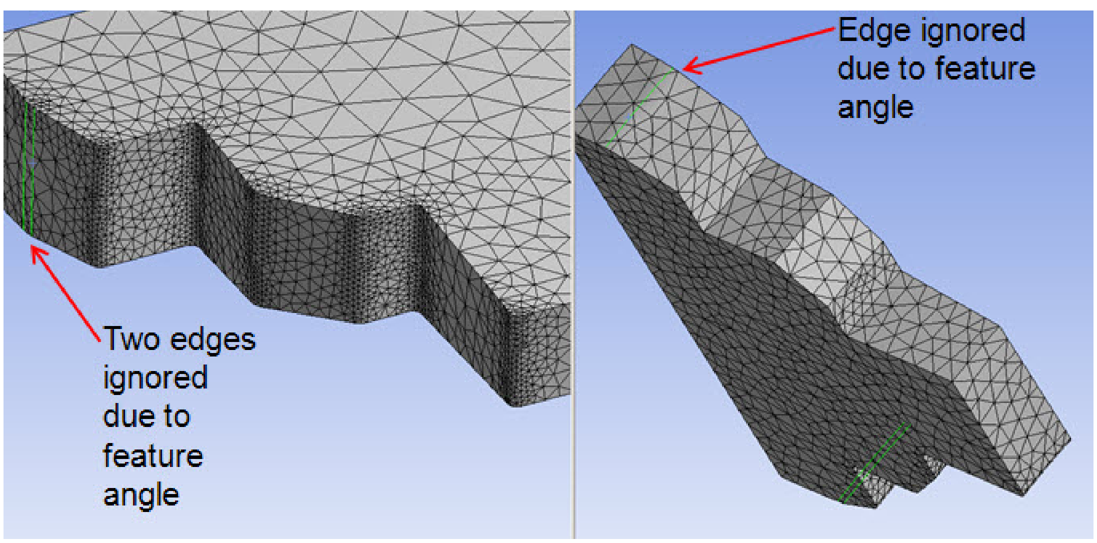


Notice the settings shown below, with **Feature Angle** set to 30 and **Mesh Based Defeaturing** turned off.

Advanced

Defined By	Max Element Size
Max Element Size	8. mm
Feature Angle	30.0 °
Mesh Based Defeaturing	Off
Refinement	Proximity and Curvature
Min Size Limit	1. mm

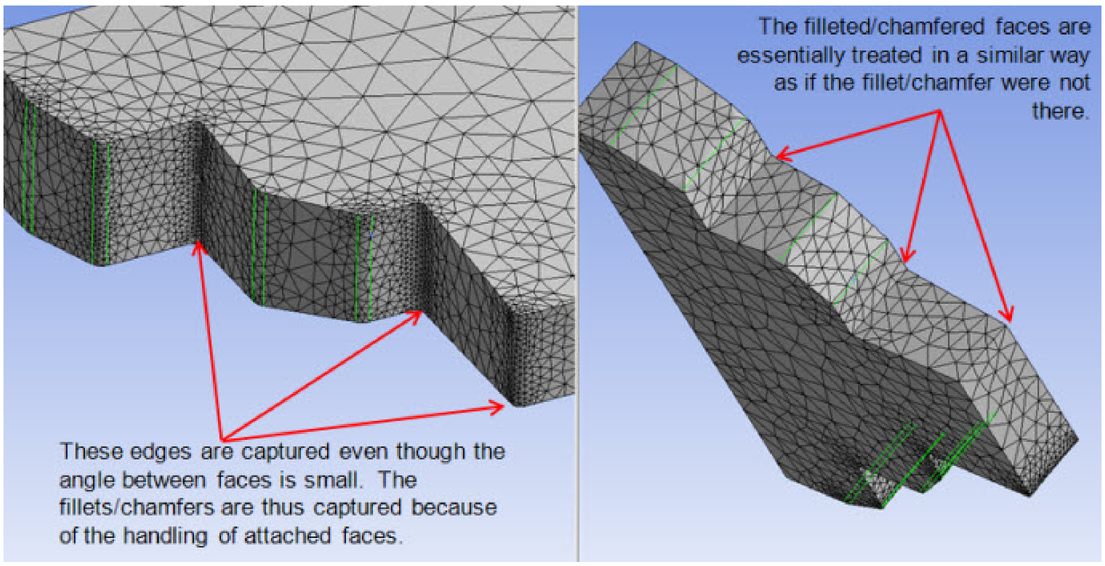
In the figure below, the highlighted edges are the edges that are ignored with the settings shown above. All edges are captured except for locations where the angle between faces or adjacent fillet/chamfer faces (two bottom edges) is 20 degrees. Changing the **Feature Angle** to a value below 20 will result in the mesher capturing those edges, while increasing the angle will result in more edges being ignored.



In the settings shown below, **Feature Angle** is changed to 80 but the other settings used before are retained.

Advanced	
Defined By	Max Element Size
Max Element Size	8. mm
Feature Angle	80.0 °
Mesh Based Defeaturing	Off
Refinement	Proximity and Curvature
Min Size Limit	1. mm

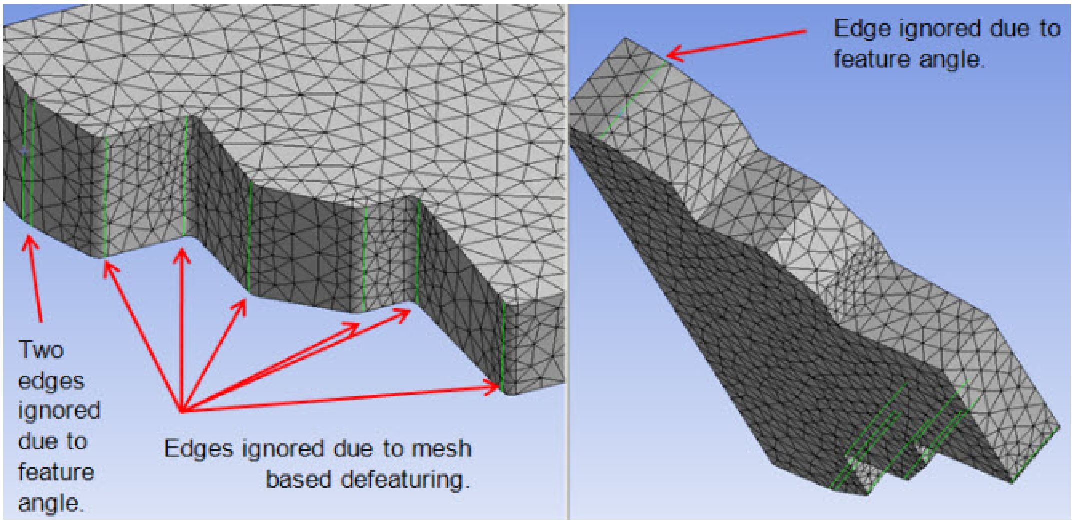
In the figure below, the highlighted edges are the edges that are ignored with the settings shown above. All edges are ignored except for those at angles of 90 degrees, both with or without fillets/chamfers.



Now consider the settings shown below. Here the **Feature Angle** is set back to 30, but **Mesh Based Defeaturing** is turned on. Both the **Defeature Size** and the **Min Size Limit** are set to 2.5 mm, which is larger than the bottom fillets/chamfers.

Advanced	
Defined By	Max Element Size
Max Element Size	8. mm
Feature Angle	30.0 °
Mesh Based Defeaturing	On
Defeaturing Tolerance	2.5 mm
Refinement	Proximity and Curvature
Min Size Limit	2.5 mm

In the figure below, the highlighted edges are the edges that are ignored with the settings shown above. The same edges as before are ignored due to the feature angle, but in addition every other edge along the bottom fillets/chamfers is ignored.

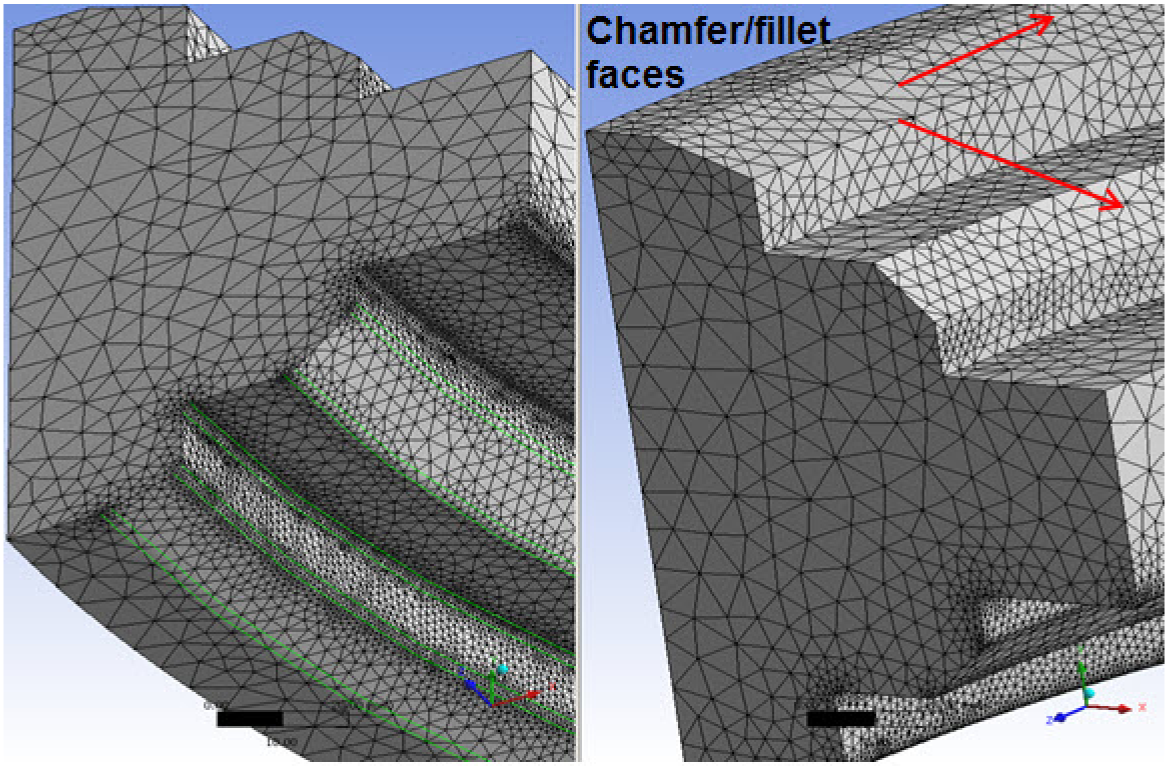


The last part of this example involves the case in which the angle of revolution is 180 degrees. Once again the **Feature Angle** is set to 80 but **Mesh Based Defeaturing** is turned off.

Advanced

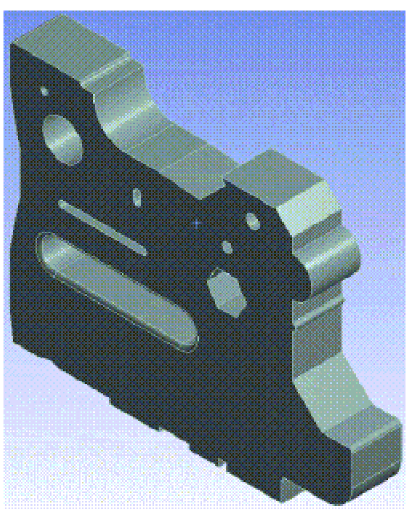
Defined By	Max Element Size
Max Element Size	8. mm
Feature Angle	80.0 °
Mesh Based Defeaturing	Off
Refinement	Proximity and Curvature
Min Size Limit	1. mm

In the figure below, the highlighted edges are the edges that are ignored. With the settings shown above and the longer extrusion, more faces are found to be fillets/chamfers when compared to the case of the shorter extrusion. In comparison, the bottom section is identical as all faces are found to be fillets/chamfers (so the meshing behavior does not change). However, with the inclusion of all faces on top being considered chamfers, the meshing behavior does change.



The following series of figures shows examples of the Patch Independent Tetrahedron mesher with various settings. Figure (a) shows the base geometry.

Figure 89: Example (a) Showing Base Geometry



Figures (b) through (f) below show examples of the Patch Independent Tetrahedron mesher under the conditions noted.

Figure 90: Example (b) Min Size Limit (Described Below) Set to 1

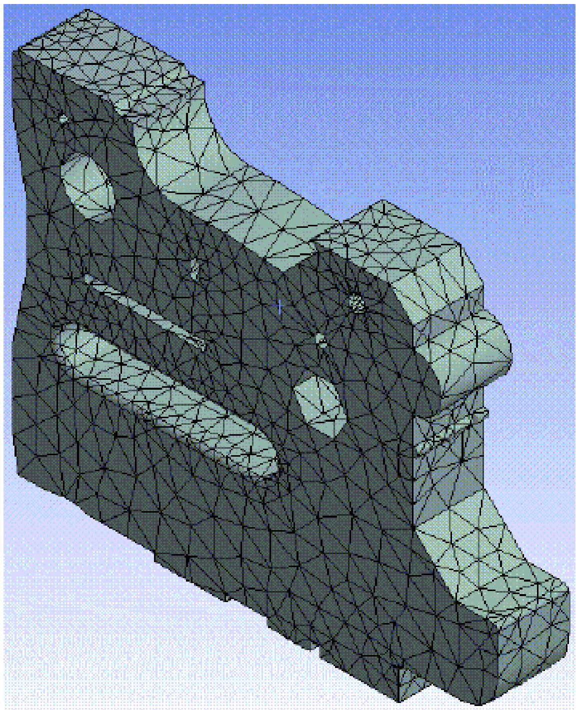


Figure 91: Example (c) Min Size Limit (Described Below) Set to 0.5

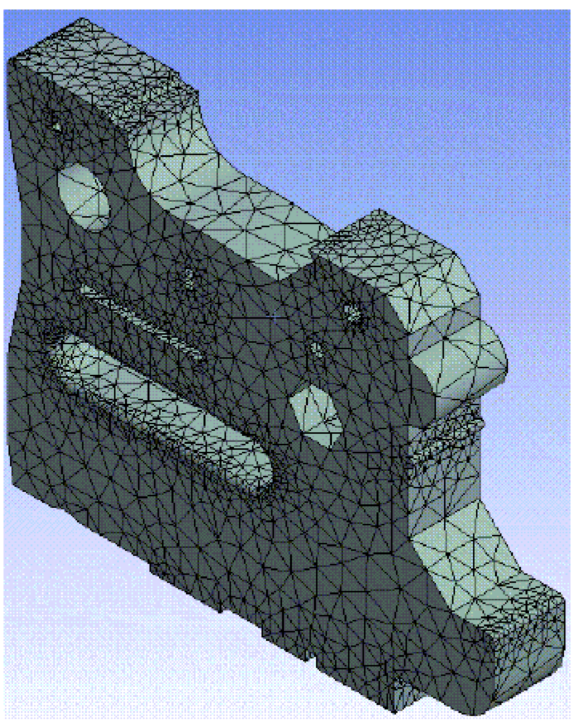


Figure 92: Example (d) Defeature Size Set to 1

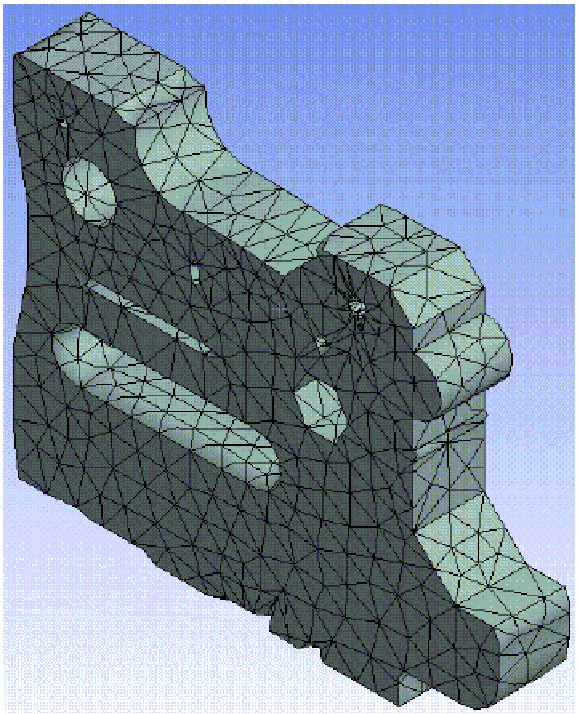


Figure 93: Example (e) Defeature Size Set to 1 and Element Order Set to Linear

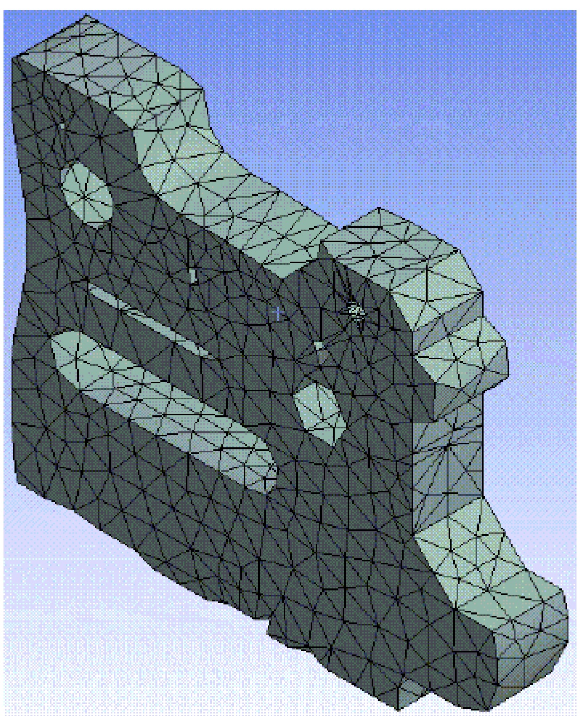
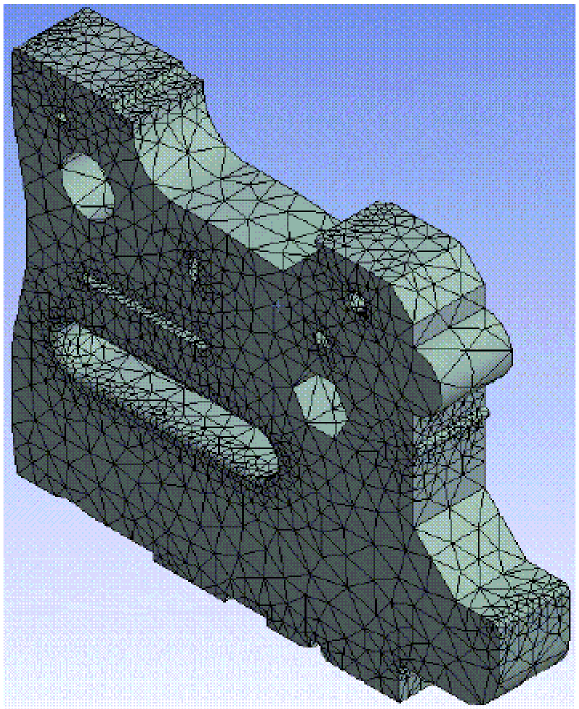


Figure 94: Example (f) Defeature Size Set to 1 and Min Size Limit Set to 0.5



- **Refinement** and **Min Size Limit** - When **Refinement** is set to **Proximity and Curvature**, **Curvature**, or **Proximity**, the mesh is automatically refined based on geometry curvature and/or proximity. This will result in larger elements on flat planar faces and smaller elements in areas of high curvature or within small gaps. In addition, a **Min Size Limit** field is displayed, in which you enter a numerical value. (The default of **Refinement** is **Proximity and Curvature**, unless **Physics Preference** (p. 93) is set to **Explicit**, in which case the default is **No**.)

  Curvature or proximity based refinement will subdivide the elements until this **Min Size Limit** is reached. However, projection to geometry and smoothing may push the size even smaller for some elements. The **Min Size Limit** prevents curvature or proximity based refinement from generating elements that are too small. The default value of **Min Size Limit** depends on whether **Use Adaptive Sizing** is set to **Yes** or **No**:

  – If **Use Adaptive Sizing** is set to **No**, the default value of **Min Size Limit** is inherited from the global **Curvature Min Size** (p. 108)/**Proximity Min Size** (p. 110) control based on the **Refinement** type set. The mesher will also use the Min Size values defined locally.

    → When **Refinement** is set to **Proximity and Curvature** (default), the maximum of the global **Proximity Min Size** (p. 110) and **Curvature Min Size** (p. 108) is used as the default **Min Size Limit**.

    → When **Refinement** is set to **Proximity**, the global **Proximity Min Size** (p. 110) is used as the default **Min Size Limit**.

→ When **Refinement** is set to **Curvature**, the global **Curvature Min Size** (p. 108) is used as the default **Min Size Limit**.

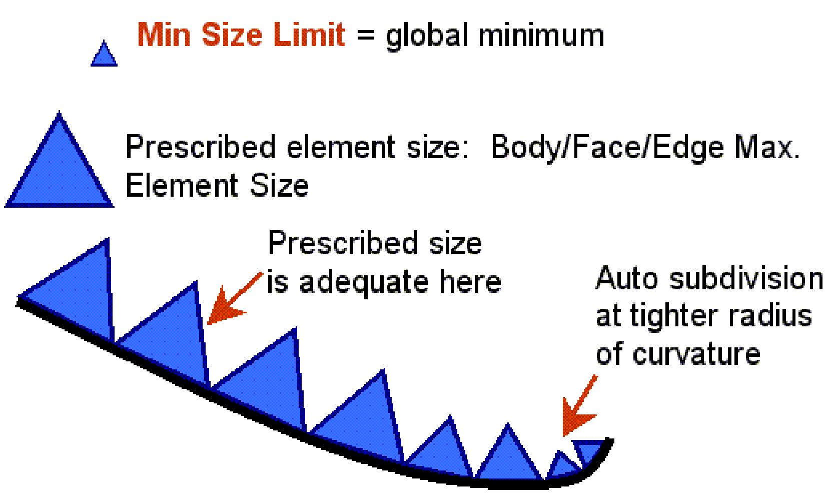
Note:

The global **Curvature Min Size** (p. 108)/**Proximity Min Size** (p. 110) values will be used to determine the default value for **Min Size Limit** based on the **Refinement** type set for the **Patch Independent** method control, irrespective of the global refinement type set (for example, even if **Capture Curvature** and **Capture Proximity** are both set to **No**).

– If **Use Adaptive Sizing** is set to **Yes**, you must specify a value for **Min Size Limit**.

Note:

Also see Notes on Element Size Settings for the Patch Independent Tetra Mesher (p. 220).

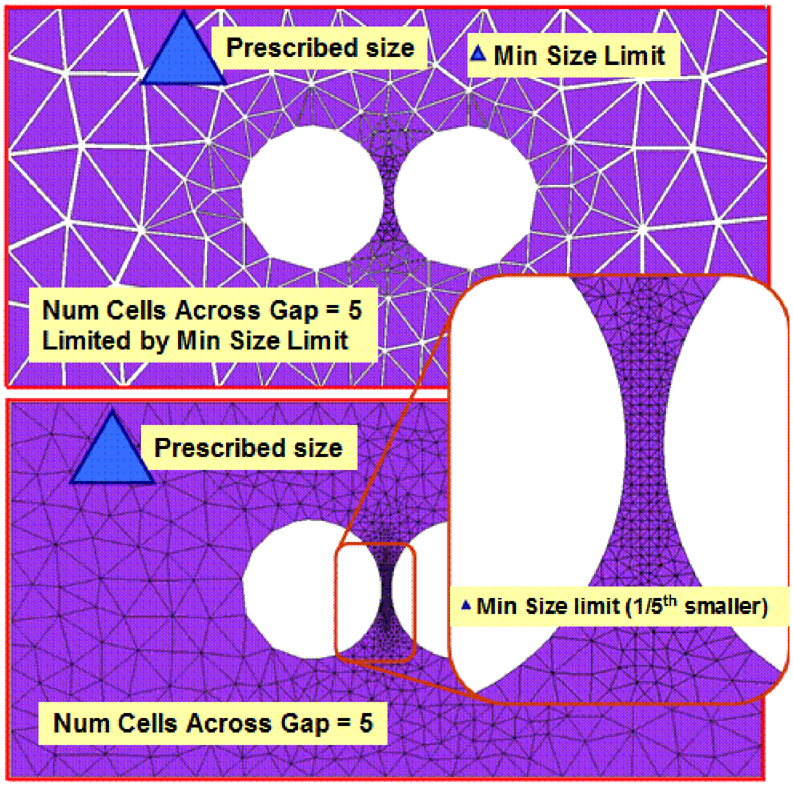


- **Num Cells Across Gap** - (Displayed only when **Refinement** is set to **Proximity and Curvature** or **Proximity**.) The number of cells desired in narrow gaps. This sets the goal for the proximity based refinement. The mesh will subdivide in tight regions toward this goal, but the refinement is limited by the **Min Size Limit**. It will not override this limit. The default value depends on the Sizing Options (p. 100):

  – If **Use Adaptive Sizing** is set to **No**, the default value of **Num Cells Across Gap** is inherited from the global **Num Cells Across Gap** (p. 110) value.

  – If **Use Adaptive Sizing** is set to **Yes**, the default value of **Num Cells Across Gap** is 3.

  In either case, you can change the value if you want to apply a specific value locally.



- **Curvature Normal Angle** - (Displayed only when **Refinement** is set to **Proximity and Curvature** or **Curvature**.) Sets the goal for the curvature based refinement. The mesh will subdivide in curved regions until the individual elements span this angle. This refinement is also limited by the **Min Size Limit**. You can specify a value from 0 to 180. The default value depends on the Sizing Options (p. 100):
  - If **Use Adaptive Sizing** is **No**, the default value of **Curvature Normal Angle** is inherited from the global **Curvature Normal Angle** (p. 109) value.
  - If **Use Adaptive Sizing** is **Yes**, the default value of **Curvature Normal Angle** will be computed based on the value of the Span Angle Center (p. 107) global option.

  In either case, you can change the value to define specific values locally, but be aware that with PI Tet, only one value for curvature normal angle (num cells in gap) is used by the mesher. The smallest curvature normal angle (largest num cells in gap) will be applied globally. A warning message will indicate this for you.

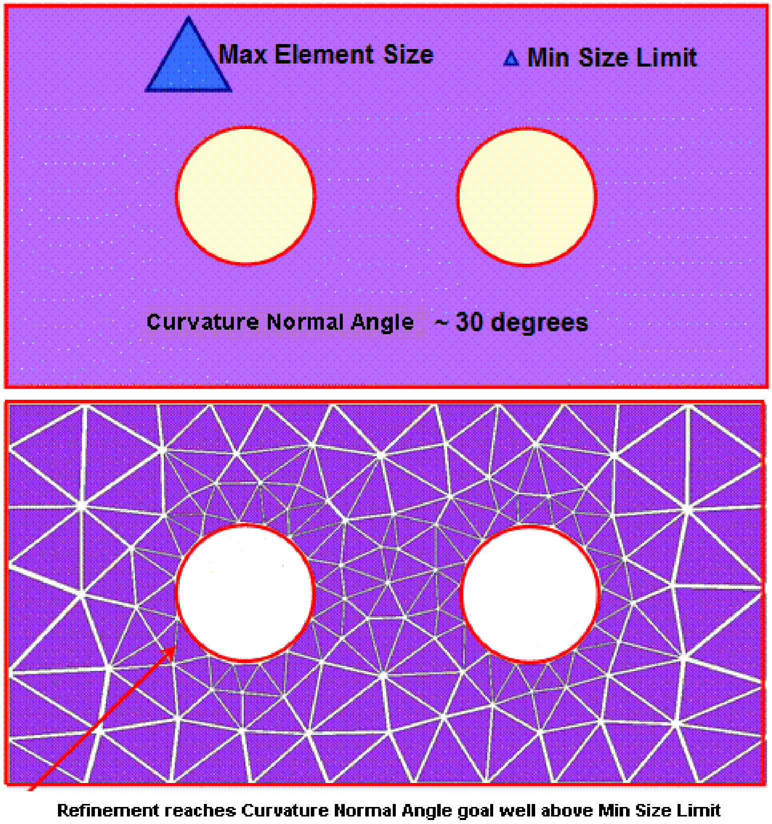


Figure 95: Example (a) Showing Base Geometry

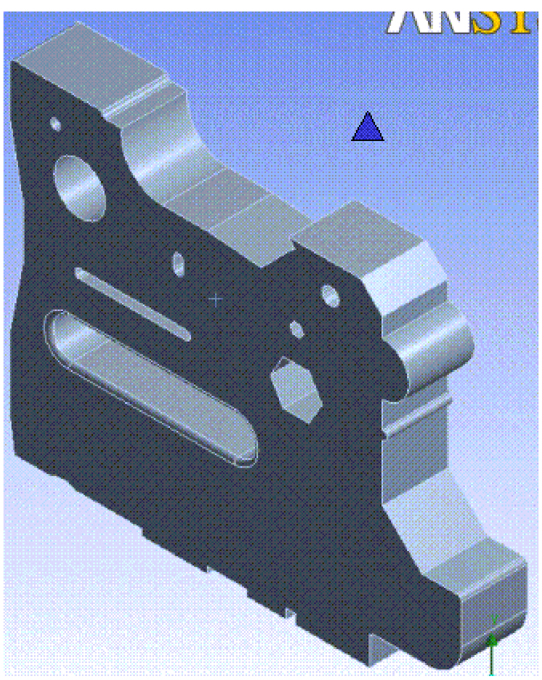


Figure 96: Example (b) Default Patch Independent Tetrahedron Mesher

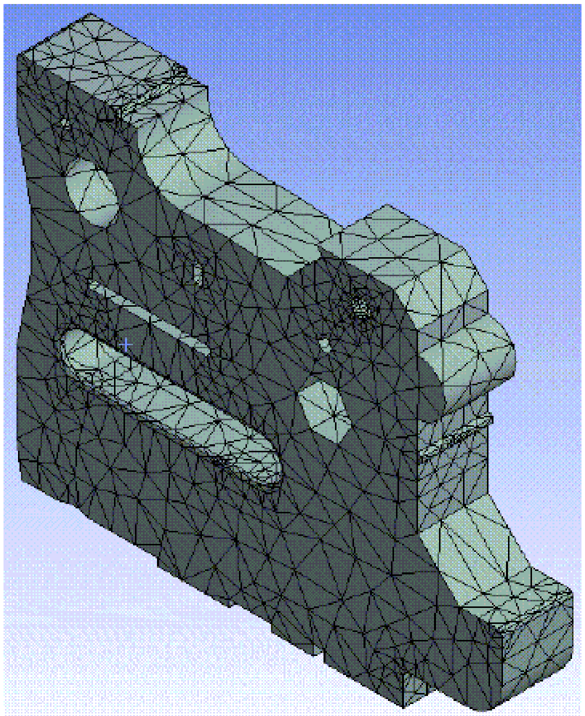
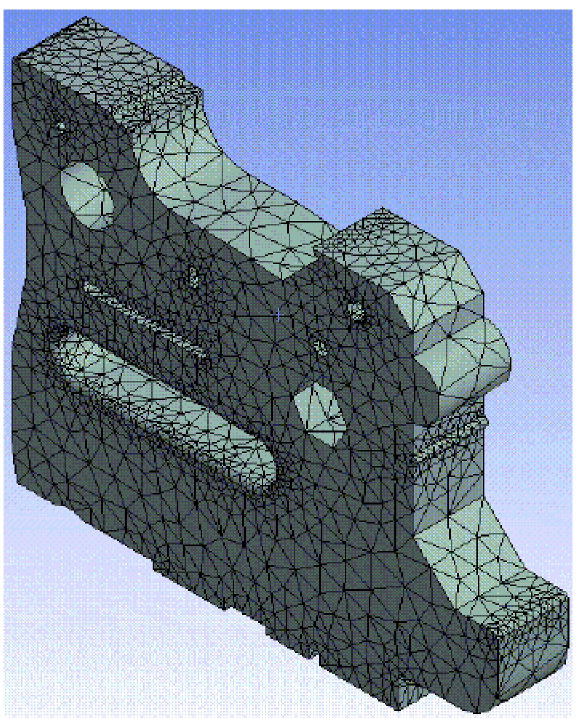


Figure 97: Example (c) Patch Independent Tetrahedron Mesher with Min Size Limit Set to Capture Curvature

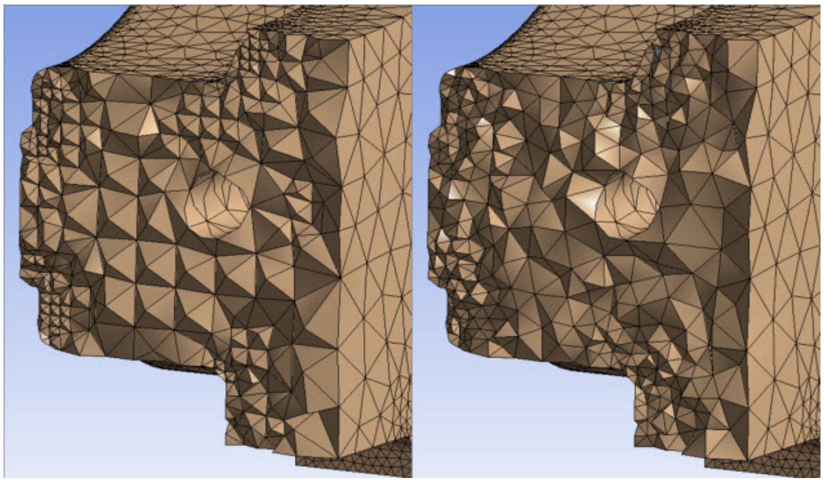


- **Smooth Transition** - Determines whether the Octree volume mesh generated from the Patch Independent mesh method should be kept or whether it should be replaced with a Delaunay volume mesh starting from the Patch Independent surface mesh. Options are **On** or **Off** (default).

If set to **On**, the volume mesh will be a Delaunay mesh. If set to **Off**, the volume mesh will be an Octree mesh.

Figure 98: Effect of Smooth Transition Setting (p. 220) illustrates the effect of setting **Smooth Transition** to **Off** (Octree volume mesh on the left) or **On** (Delaunay volume mesh on the right).

Figure 98: Effect of Smooth Transition Setting



- **Growth Rate** - Represents the increase in element edge length with each succeeding layer of elements. For example, a growth rate of 1.2 results in a 20% increase in element edge length with each succeeding layer of elements. Specify a value from 1.0 to 5.0 or accept the **Default**. When set to **Default**, the value is the same as the global growth rate (p. 105). If **Use Adaptive Sizing** is set to **Yes**, the **Default** is set differently based on whether **Smooth Transition** is **Off** (default is 2.0) or **On** (default is 1.2).

> **Note:**
>
> If **Smooth Transition** is set to **Off**, the growth rate is very approximate since the volume is filled with an Octree meshing approach which requires 2-to-1 transitions. Thus in such cases, the growth rate relates only to when the transitions occur through the mesh.

- **Minimum Edge Length** - Read-only indication of the smallest edge length in the part.

- **Match Mesh Where Possible** - The **Match Mesh Where Possible** control is applicable to contact definitions between faces. Options are **Yes** and **No**. The default is **Yes**. If contact is defined by a single face that topologically belongs to two different bodies, setting this option to **Yes** has no effect. However, if there are independent faces on the two bodies, setting this option to **Yes** causes the Patch Independent mesh method to create nodes on both sides of the contact. The nodes are not connected but have identical coordinates.

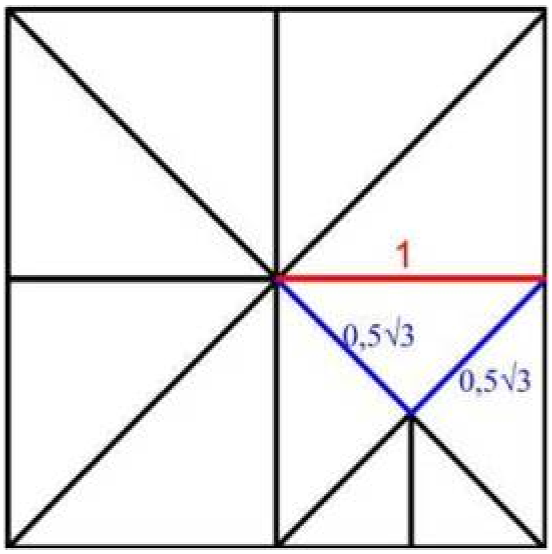
- **Write ICEM CFD Files** - Sets options for writing Ansys ICEM CFD files. Refer to Writing Ansys ICEM CFD Files (p. 84) for details.

Notes on Element Size Settings for the Patch Independent Tetra Mesher

Remember these notes when using the Patch Independent tetra mesher:

- If you are specifying element sizes with the Patch Independent mesher, you may notice that some element edge lengths are less than the size that you have entered. For example, if your element size is 1, the resulting elements in a uniform tetrahedron mesh will have tetrahedron with edges of length $3^{1/2}/2$ and edges of length 1. A single tetrahedron in this mesh will have two edges of length 1 and four edges of length $3^{1/2}/2$. Two of the three dimensions of the bounding box of this tetrahedron will have length of 1 while the other dimension will have the length of 0.5. This correlates to an element size of 1.

  Figure 99: Element Edge Lengths Smaller Than Specified Element Size

- If you are using Curvature and Proximity Refinement, you may notice that your elements are always less than the maximum size specified. Element growth rates with this mesher are always based on powers of 2. For instance if your minimum size limit is set to 1 and your maximum element size limit is set to 5, and you have curvature in your model that warrants curvature based mesh refinement down to the minimum element size, you will see that the largest elements are not size 5 but size 4. This happens because in order to maintain elements at the minimum size limit, the initial tetrahedron must be some power of 2 larger than the minimum element size, which in this example case is 1.

- Small features of Named Selections will be checked in comparison to element size settings prior to meshing. If the minimum element size seems to be too big to capture the essential features of the geometry, a warning will be issued if small entities could cause the mesher to fail.

Notes on Scoping for the Patch Independent Mesher

You can use the Patch Independent tetra mesh method in combination with other solid mesh methods in a multibody part, and the bodies will be meshed with conformal mesh. Refer to Conformal and Non-Conformal Meshing (p. 21) for information about conformal meshing.

Notes on Virtual Topologies and the Patch Independent Mesher

Virtual topologies may affect the success of meshing with the Patch Independent tetra mesh method. Because virtual topologies are often a coarse approximation of the original faces or edges, the resulting small inaccuracies (gaps and overlaps) may cause the Patch Independent tetra mesher to miss some parts of the boundary of the virtual topology. As a result, the mesher may not accurately model the respected topology and may fail.

Since in general, the Patch Independent tetra mesh method does not require the use of virtual topologies to clean up the geometry, you can remove some of the problematic virtual topology and use Named Selections for boundary conditions instead, as appropriate.

Miscellaneous Notes for the Patch Independent Mesher

The Patch Independent tetra mesh method does not support mesh connections (p. 444), contact matches (p. 455), pinch controls (p. 182), match controls (p. 280), or face meshing controls (p. 265).

Hex Dominant Method Control

If you select the **Hex Dominant** method, a free hex dominant mesh is created. If you are interested in a hex mesh, this option is recommended for bodies that cannot be swept. To preview any bodies that can be swept, click **Mesh** on the Tree Outline and right-click the mouse. Select **Show> Sweepable Bodies** from the context menu to display bodies that fulfill the requirements of a sweepable body (other than axis sweeping).

The **Hex Dominant** mesh method includes the following settings:

- **Element Order** - Refer to Method Controls and Element Order Settings (p. 196).
- **Free Face Mesh Type** - Determines the shape of the elements used to fill the body. Allows you to choose **Quad/Tri** or **All Quad**. The default is **Quad/Tri**.

Hex dominant meshing adds the most value under the following conditions:

- Meshing bodies with large amounts of interior volume.
- Meshing bodies that transition from sweepable bodies in a body that has been decomposed for sweeping. However, it is better to use **Body**/**Face Sizing** to obtain more uniform face meshing, which leads to more hexes by volume.

Hex dominant meshing adds little value under the following conditions:

- Meshing thin complicated bodies (like a cellular phone case). The number of elements may actually increase compared to a tetrahedron mesh since the element size must be much smaller for this class of body when using hex dominant meshing to create well shaped hexes.
- A body is sweepable or can easily be decomposed to multiple sweepable bodies. The quality of a swept mesh is usually superior to that of a hex dominant mesh.

- Models where fast transition of the mesh can result in poor solution accuracy (such as CFD models). The Hex dominant approach can have very fast transitions at the core of the volume.

Note:

- Mesh Matching for cyclic symmetry is not supported for hex dominant meshing.
- Workbench assists you in determining if hex dominant meshing is applicable to your situation. When you apply the **Hex Dominant** option on a body or group of bodies, Workbench calculates the normalized volume to surface area ratio (p. 223). If it detects a ratio less than 2, **Control Message** appears in a highlighted row under **Definition** in the Details View. If you click **Yes, Click To Display**, a warning message states that a low percentage of hex elements or poorly shaped hex elements may result. Suggestions are included for alternative meshing schemes.

  The normalized volume to surface area ratio is defined by the following expression:

  (Volume of body/(Surface area of body)$^{3/2}$)/factor

  where factor, the ratio for a unit sphere = (4/3 π)/(4 π)$^{3/2}$
- Adaptive refinement starting from a hex dominant mesh will result in remeshing of the structure with tetrahedrons.
- If you apply a local **Sizing** control (p. 248) to a solid body with a **Method** control set to **Hex Dominant** (p. 222) or **Sweep** (p. 223), or to a sheet body with a **Method** control set to **Quadrilateral Dominant** (p. 245), a near uniform quadrilateral mesh will result on all affected faces on a body meshed with **Hex Dominant**, on the source face meshed with **Sweep**, and on all affected faces meshed with **Quadrilateral Dominant**. To obtain even more of a uniform quadrilateral mesh, set the **Behavior** (p. 262) of the **Sizing** control to **Hard**.

Sweep Method Control

If you select the **Sweep** method, a swept mesh is forced on "sweepable" bodies (p. 323) (including axis-sweepable bodies, which are not displayed when you use the **Show Sweepable Bodies** feature). The mesher will fail if a swept mesh cannot be generated on a body with a **Sweep Method** control.

Use the **Sweep** option for any of the following situations:

- A swept mesh is required.
- You want a swept mesh on a model that revolves around an axis where the source and target faces share topology.

When you choose the **Sweep** option, the Details View expands to include additional settings, many of which are unique to this option. For basic usage that involves obtaining a swept mesh, the procedure is to apply a **Method Control** to one or more bodies, set **Method** to **Sweep**, and accept the default values of the various settings.

For advanced or specialized usage such as meshing thin models or axis sweeping, adjust the settings as needed. The following is a description of each of these settings.

- **Algorithm** - Allows you to choose sweep algorithm. The default is set to **Program Controlled** and you can also select **Axisymmetric**.
  - **Program Controlled** - This algorithm can be applied only on the traditional sweepable models.
  - **Axisymmetric** - This algorithm can be applied to all axisymmetric models. Axisymmetric Sweeper does not support shared topology. Axisymmetric Sweeper supports **Contact Sizing** (p. 263) and **Topology Protection** (p. 180)

    Note:

    Mesh Based Defeaturing (p. ?) is available only when **Axisymmetric** algorithm is selected.

- **Mesh In Center** - Defines the type of mesh in the centre of the Axisymmetric model. The default value is **Hexahedra**. When you select **Hexahedra**,O-Grid mesh is generated at the centre of the axisymmetric model. You can specify the number of division for the O-Grid using the **O-Grid Divisions**.
- **Element Order** - Refer to Method Controls and Element Order Settings (p. 196).
- **O-Grid Divisions** - Number of divisions per quadrant from the center of the O-Grid to the boundary (or outside) of the O-Grid. The default value for**O-Grid Divisions** is 4
- **Project Corners to Top** - Projects the nodes on the steps along the body of interest to the top surface of the Axisymmetric model. The default value is **Yes**.
- **Src/Trg Selection** - Defines the source and target selection type according to the following choices:
  - **Automatic** - The program determines the best source and target for the body.
  - **Manual Source** - You select the source and the program determines the target. This choice is useful when there are multiple source target pairs and you want to specify the source in order to get the correct bias through the sweep direction. Another application is when your cross section is changing and the mesh quality would be better when sweeping from one side vs. another.
  - **Manual Source and Target** - The sweeper will revolve the mesh around common edges/vertices. This choice is useful when you want to sweep a body where the source and target faces share vertices and/or edges.
  - **Automatic Thin** (p. 330) - This choice is for thin models and thin sheet metal parts, or any application where you want one hex or wedge through the thickness, in preparation for using the Mechanical APDL application's SOLSH190 element or the LS-DYNA thick shell element. (See the description for the Mechanical APDL element in the *Element Reference* within the Mechanical APDL help.) For this choice, the face with the largest face area is selected as the primary source and the algorithm determines the rest of the source faces. For multibody parts, only

one division through the thickness is possible. For single body parts, you can define multiple elements through the thickness using the **Sweep Num Divs** control. Biasing is not available. **User Defined Criteria** allows you to select the type of **Automatic Thin** sweep. This option is available only when the **Src/Trg Selection** is set to **Automatic Thin**. The available options are **Program Controlled** and **Protect Internal Edges**. When you select **Program Controlled**, the interior loops in the model are not preserved while performing thin sweep. When you select **Protect Internal Edges**, the interior loops in the model are preserved while performing thin sweep. An **Element Option** setting is included that instructs the solver to use the **Solid Shell** element where possible, or to always use a **Solid** element.

- **Manual Thin** (p. 330) - The same restrictions apply as described above for **Automatic Thin**. However, with this choice, you can do *any* of the following:

  → Pick one source face and allow the program to determine the rest.

  → Pick all of the source faces and allow the program to do nothing but mesh the source faces and sweep them to the target.

  → Pick multiple source faces and mesh one target face.

Note:

→ The **Sweep** mesh method does not support the **Manual Source**, **Manual Source and Target**, or **Manual Thin** settings for **Src/Trg Selection** if **Sweep** is applied to more than one part, even if you suppress all of the other parts.

→ In some cases, the thin model sweeper may want to swap source and target faces based on meshing conditions in neighboring bodies. In such cases, a warning message will be issued to alert you.

Refer to Considerations for Selecting Source Faces for the Thin Model Sweeper (p. 331) for details.

→ To make source/target face selection easier, select **Annotation Preferences** from the Toolbar and then deselect **Body Scoping Annotations** in the **Annotation Preferences** option box to toggle the visibility of annotations in the **Geometry** window. For example, after scoping **Sweep** to a body, the body will be displayed using a blue solid annotation. Turn off the body scoping annotations, then select the source/target faces. For picking internal faces, the **Hide Faces** right-click option may help you to see inside a body. For example, you can select external faces in the **Geometry** window and then use the **Hide Faces** option to hide the selected faces (making it easier to select the internal faces).

- **Free Face Mesh Type** - Determines the shape of the elements used to fill the swept body (pure hex, pure wedge, or a combination of hex/wedge). Allows you to choose **All Tri**, **Quad/Tri**, or **All Quad** meshing when **Src/Trg Selection** is **Automatic**, **Manual Source**, or **Manual Source**

and Target. Allows you to choose **Quad/Tri** or **All Quad** meshing when **Src/Trg Selection** is **Automatic Thin** or **Manual Thin**. The default in all cases is **Quad/Tri**.

Note:

– If the source face is also a side face of another body, the face will always be quad mapped.

– When **Free Face Mesh Type** is set to either **Quad/Tri** or **All Quad** and the source face can be mapped meshed, the face will sometimes be mapped meshed even if it means applied sizing controls (such as **Contact Sizing** (p. 263), **Sphere of Influence** (p. 256), etc.) will be ignored.

– In some cases when **Src/Trg Selection** is set to **Automatic**, the source face that is selected by the software must be quad mapped in order for the sweep method to be successful. In these cases, the value that is specified for **Free Face Mesh Type** may be ignored.

- **Type** - Allows you to specify a **Number of Divisions** or **Element Size** through the sweep direction. When sweeping generalized bodies that share faces, the **Element Size** is a soft constraint on interval assignment and the **Number of Divisions** is a hard constraint. If you have conflicting **Number of Divisions** constraints, the sweeper will fail and yield a message. To obtain a regular mesh in the sweep direction, the guiding edges must have consistent lengths. You can define virtual split edges to achieve consistent lengths for these edges (see Creating and Managing Virtual Split Edges (p. 517)). Also see Sizing Control (p. 248) for more information.

Note:

The Type **ElementSize** is not available for **Axisymmetric** algorithm.

- **Sweep Bias Type** - Specify bias in the same manner as edge biasing for the **Bias Type** setting in a **Sizing** (p. 248) mesh control. There is no graphical feedback for biasing on a **Method** control. Biasing direction is based from the source to the target.

Note:

Sweep Bias Type is not available for **Axisymmetric** algorithm.

- The **Constrain Boundary** setting is available for multibody parts only (both for general sweeping and thin sweeping). Specify whether you want to allow the mesher to split the elements at the boundary of a swept mesh region to aid in meshing. You can choose **Yes** (constrain boundary, no splitting is allowed) or **No** (do not constrain boundary, splitting is allowed). Choosing **Yes** prevents tets from entering the swept body. The default is **No**.

Note:

Constrain Boundary setting is not available for **Axisymmetric** algorithm.

Note:

- For gasket simulations, set the **Stiffness Behavior** of the body to **Gasket** and proceed with adjusting mesh settings as described in the Gasket Meshing section located under Gasket Bodies in the Mechanical application help.
- In models with swept regions, the sizing controls will affect the mesh gradation in the swept region. You can override this effect by specifying any **Sweep Bias** value (including a value of 1), **Sweep Element Size** value, or **Sweep Num Divs** value in the Details View when defining the sweep method.
- There is a system limitation when using the sweep method with the Size Function (p. 89). The Size Function may have nodes slightly off because the spacing is queried. The sweeper then tries to match that spacing, which may lead to unexpected mesh results.
- If you apply a local **Sizing** control (p. 248) to a solid body with a **Method** control set to **Hex Dominant** (p. 222) or **Sweep** (p. 223), or to a sheet body with a **Method** control set to **Quadrilateral Dominant** (p. 245), a near uniform quadrilateral mesh will result on all affected faces on a body meshed with **Hex Dominant**, on the source face meshed with **Sweep**, and on all affected faces meshed with **Quadrilateral Dominant**. To obtain even more of a uniform quadrilateral mesh, set the **Behavior** (p. 262) of the **Sizing** control to **Hard**.

Limitations

The limitations for **Axisymmetric** sweep are as follows:

- **Topology Protection** (p. 180) is not supported in **Axisymmetric** algorithm.
- A warning message appears, when the nodes are off the mesh. You can click the warning message to locate that area on the model and can redefine mesh using **Mesh Based Defeaturing** (p. ?) for the particular area.
- Axisymmetric body with less than 360 degree revolution is not supported by this algorithm.
- **Axisymmetric** algorithm does not support models with share-topology.
- **Axisymmetric Sweep** does not support multibody.

MultiZone Method Control

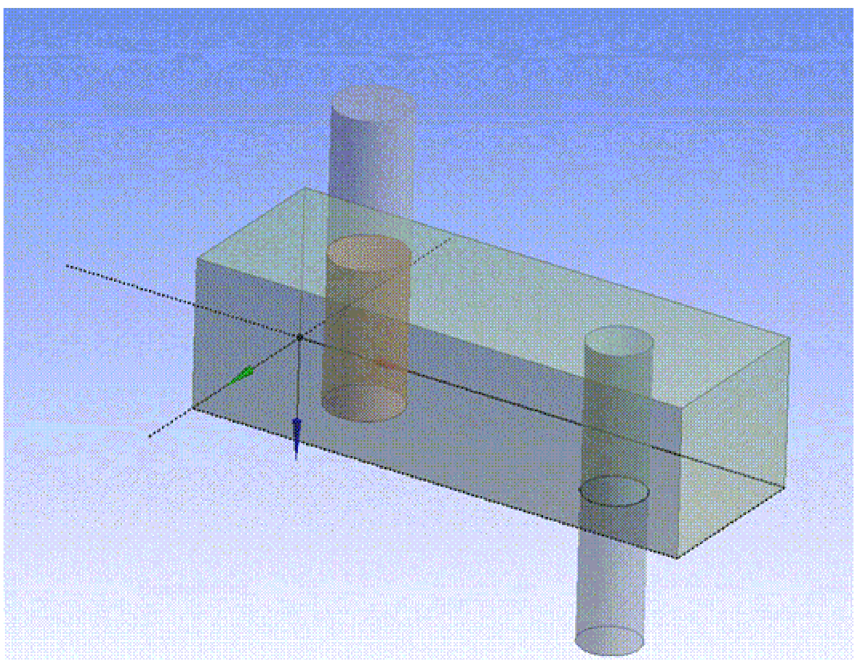
The **MultiZone** mesh method provides automatic decomposition of geometry into mapped (sweepable) regions and free regions. When the **MultiZone** mesh method is selected, all regions are meshed with a pure hexahedral mesh if possible. To handle cases in which a pure hex mesh will not be possible, you can adjust your settings so that a swept mesh will be generated in structured regions and a free mesh will be generated in unstructured regions.

For example, using the **Sweep** mesh method, you would need to slice the part below into five bodies as shown to obtain a pure hex mesh:

Note:

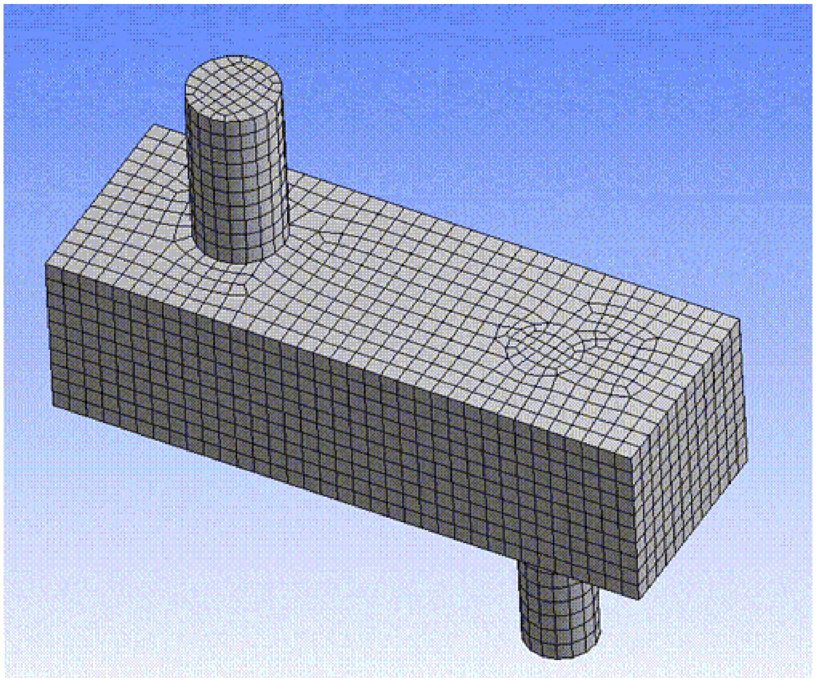
This section describes method control settings. See MultiZone Meshing (p. 343) for detailed algorithm and usage information.

Figure 100: Sweep Method Would Require Slicing to Obtain Pure Hex Mesh



In contrast, using the **MultiZone** mesh method requires no slicing. **MultiZone** automates the geometry decomposition and generates the pure hex mesh shown in Figure 101: MultiZone Generates Pure Hex Mesh without Slicing (p. 229).

Figure 101: MultiZone Generates Pure Hex Mesh without Slicing



When you choose the **MultiZone** mesh method, the Details View expands to expose various settings, including several that are unique to **MultiZone**. For basic usage that involves obtaining a **MultiZone** mesh, the procedure is to apply a **Method Control** to one or more bodies, set **Method** to **MultiZone**, and accept the default values of the various settings.

For advanced or specialized usage, adjust the settings as needed. The following is a description of each of these settings.

- **Mapped Mesh Type** - Determines the shape of the elements used to fill structured regions according to the following choices (the default is **Hexa**):
  - **Hexa** - A mesh of all hexahedral elements is generated for the part the method is scoped to.
  - **Hexa/Prism** - A mesh of hexahedral and prism/wedge elements is generated for the part the method is scoped to. The main difference between the **Hexa/Prism** option and the other options is that for swept regions, the surface mesh can allow triangles for quality and transitioning. The triangles are later extruded to prisms/wedges.
  - **Prism** - A mesh of all prism elements is generated for the part the method is scoped to. This option is sometimes useful if the source face mesh is being shared with a tet mesh, as pyramids are not required to transition to the tet mesh.
- **Surface Mesh Method** - Specifying a value for **Surface Mesh Method** instructs **MultiZone** to use the **Program Controlled**, **Uniform**, or **Pave** method to create the surface mesh.
  - **Program Controlled** - Automatically uses a combination of **Uniform** and **Pave** mesh methods depending on the mesh sizes set and face properties. This is the default method.
  - **Uniform** - Uses a recursive loop-splitting method which creates a highly uniform mesh. This option is generally good when all edges have the same sizing and the faces being meshed do

not have a high degree of curvature. The orthogonality of the mesh from this method is generally very good.

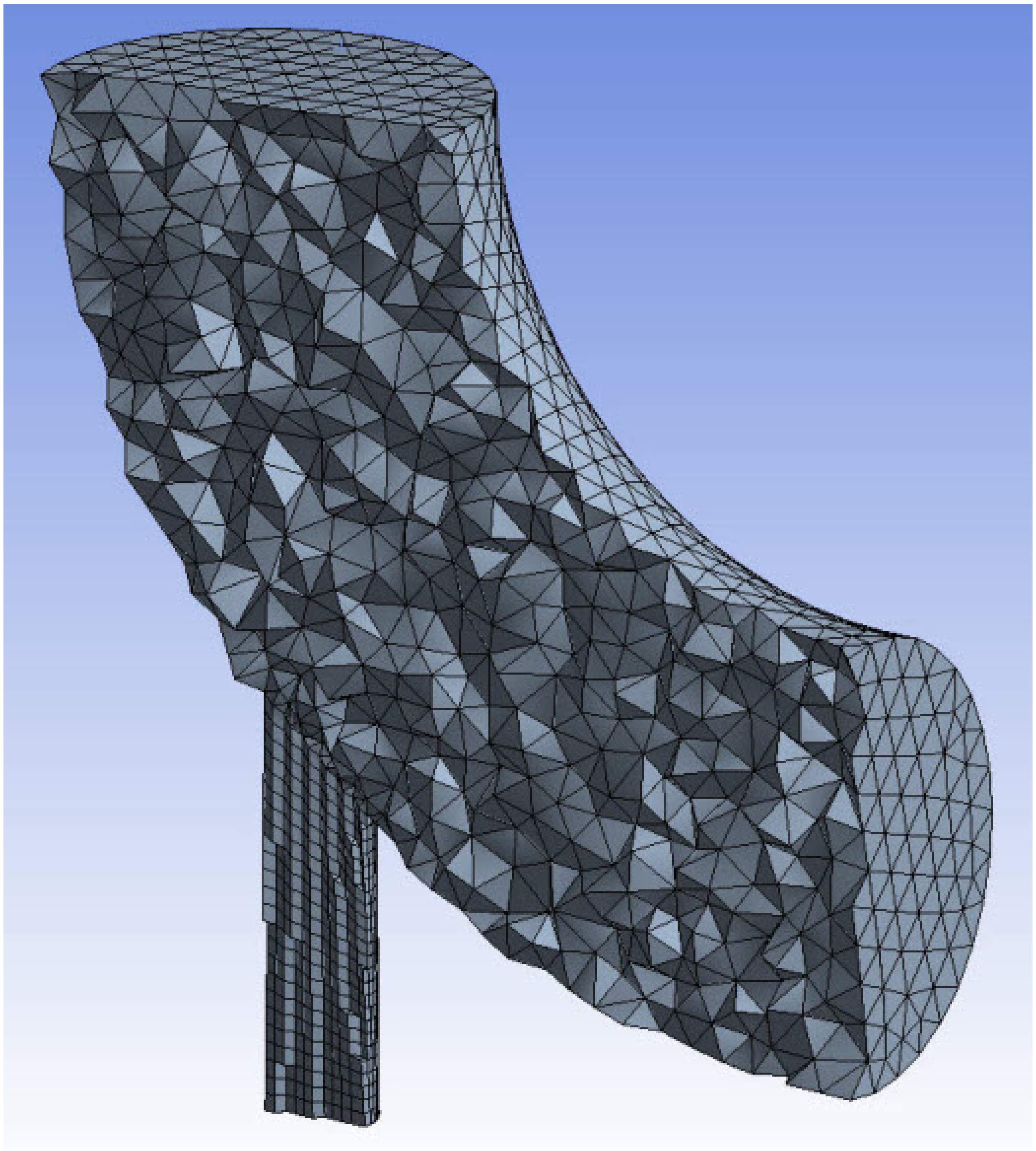
- **Pave** - Uses a paving mesh method which creates a good quality mesh on faces with high curvature, and also when neighboring edges have a high aspect ratio. This approach is also more reliable to give an all-quad mesh.

Note:

The **Surface Mesh Method** is applicable only to faces that are free meshed. If a face can be mapped meshed, it will be. See **Face Meshing Control** (p. 265) for more information.

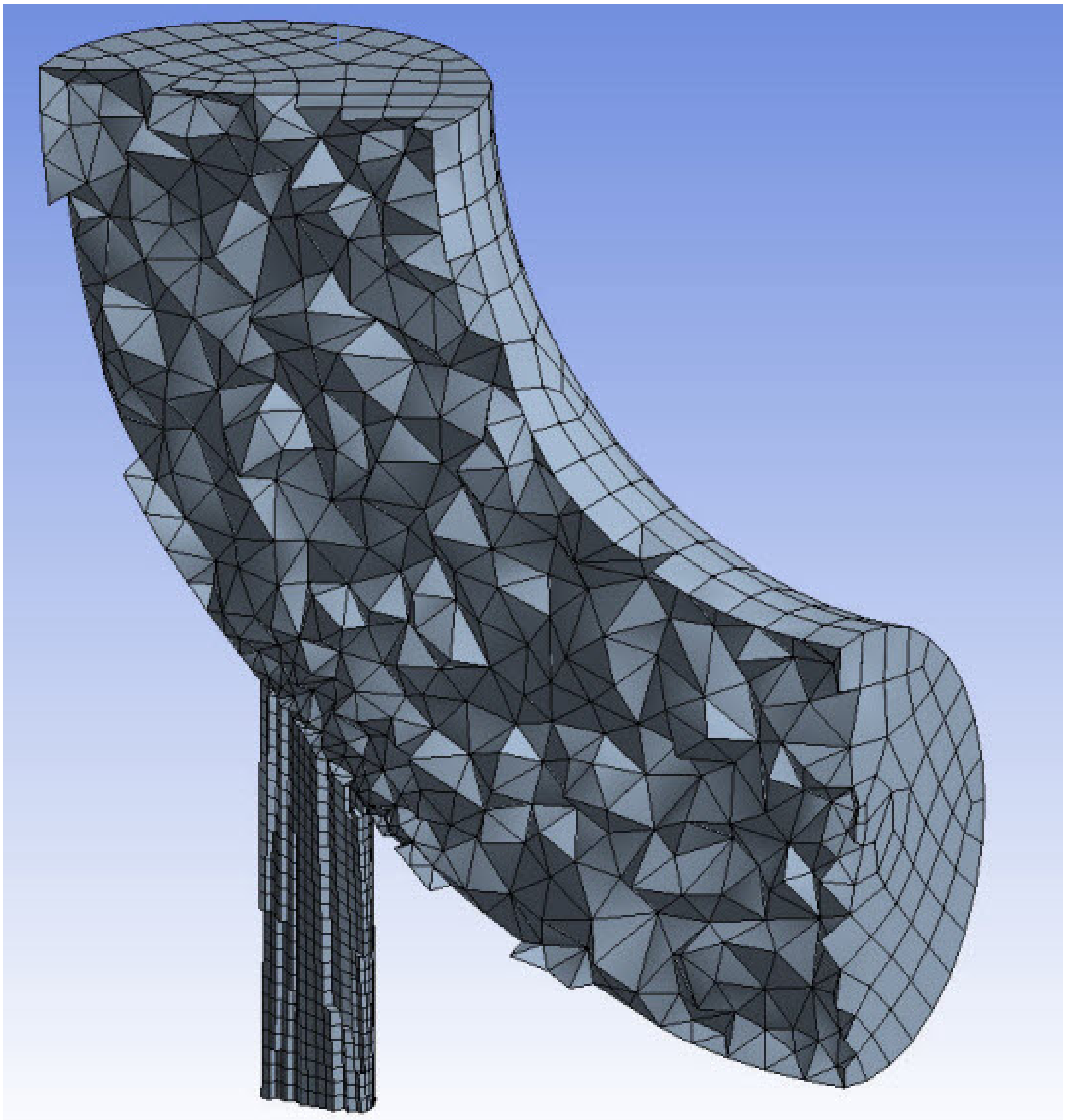
- **Free Mesh Type** - Specifying a value for **Free Mesh Type** will instruct **MultiZone** to allow a free mesh if it is not possible (without slicing) to generate a pure hex or hex/prism mesh. The value of **Free Mesh Type** determines the shape of the elements used to fill unstructured regions according to the following choices (the default is **Not Allowed**):
  - **Not Allowed** - Choose this option if you require a mapped mesh.
  - **Tetra** – Regions of the model that cannot be meshed with a mapped mesh will be filled with a tetrahedral mesh. Figure 102: Free Mesh Type = Tetra (p. 231) shows a **MultiZone** mesh that was generated when **Free Mesh Type** was set to **Tetra**. Notice the lower section that was able to be mapped meshed, and the upper section that was free meshed because it could not be map meshed. Refer to Patch Conforming Algorithm for Tetrahedrons Method Control (p. 200) for more information.

Figure 102: Free Mesh Type = Tetra



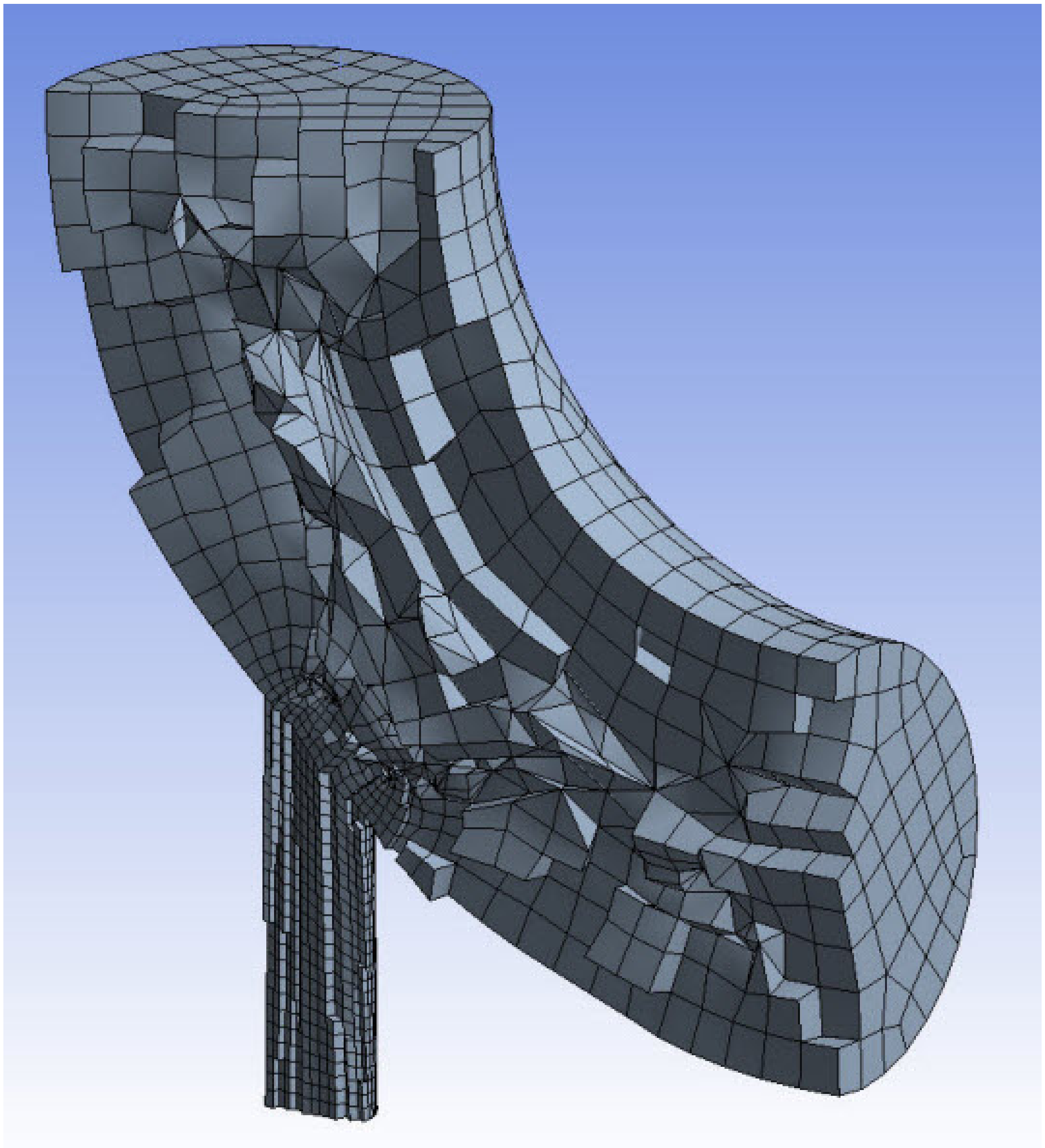
- **Tetra/Pyramid** – Regions of the model that cannot be meshed with a mapped mesh will be filled with a tetrahedral mesh with pyramids at the faces. Figure 103: Free Mesh Type = Tetra/Pyramid (p. 232) shows a **MultiZone** mesh that was generated when **Free Mesh Type** was set to **Tetra/Pyramid**. Notice the lower section that was able to be mapped meshed, and the upper section that was free meshed because it could not be map meshed. Refer to Patch Conforming Algorithm for Tetrahedrons Method Control (p. 200) for more information.

Figure 103: Free Mesh Type = Tetra/Pyramid



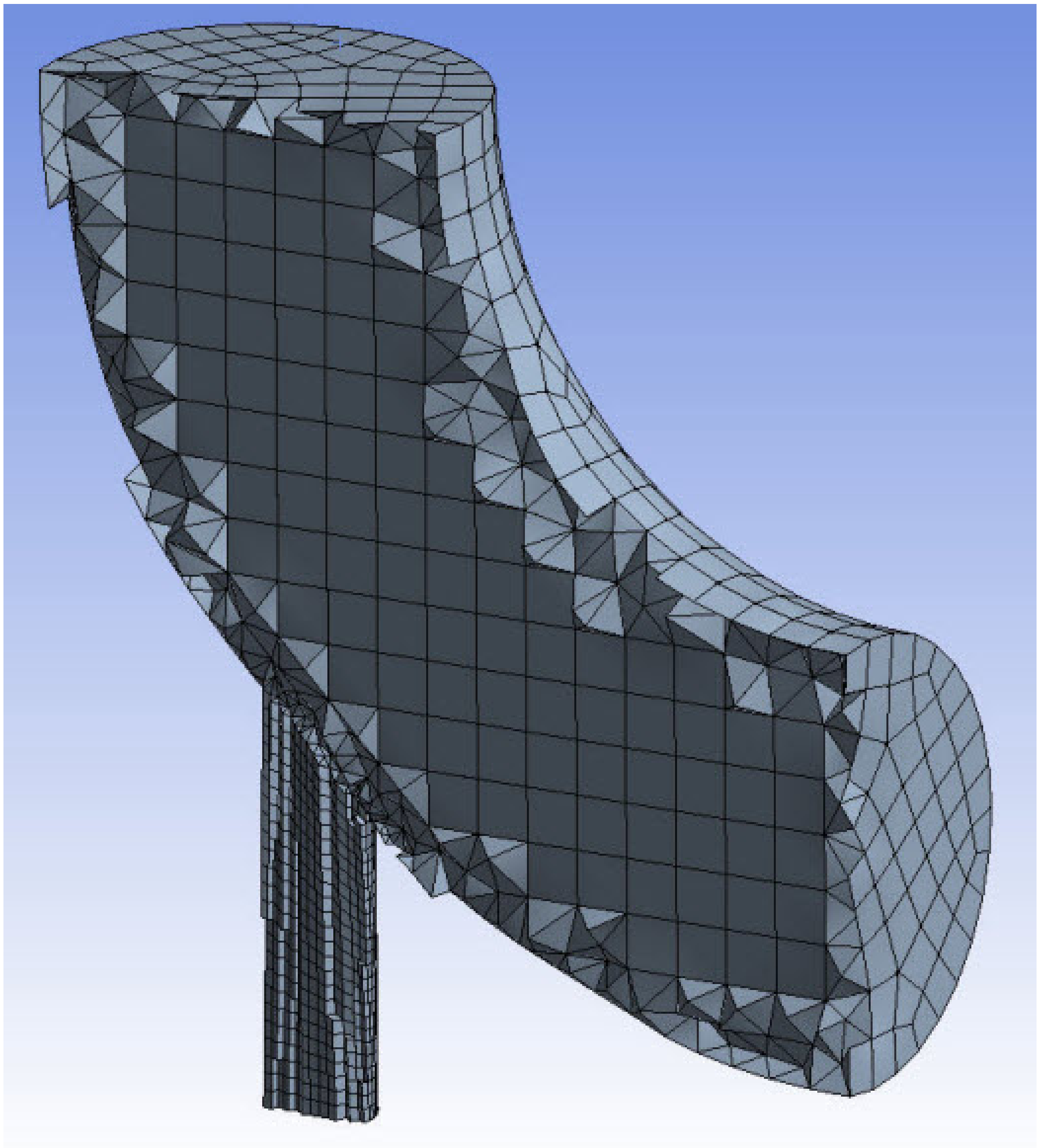
– **Hexa Dominant** — Regions of the model that cannot be meshed with a mapped mesh will be filled with a hex dominant mesh. Figure 104: Free Mesh Type = Hexa Dominant (p. 233) shows a **MultiZone** mesh that was generated when **Free Mesh Type** was set to **Hexa Dominant**. Notice the upper section that was able to be meshed mapped meshed, and the lower section that was free meshed because it could not be mapped meshed. Refer to Hex Dominant Method Control (p. 222) for more information.

Figure 104: Free Mesh Type = Hexa Dominant



– **Hexa Core** - Regions of the model that cannot be meshed with a mapped mesh will be filled with a hexa core mesh. Hexa Core meshes can be generated where the majority of the volume is filled with a Cartesian array of hexahedral elements essentially replacing the tetras. This is connected to the remainder of a prism/tetra hybrid by automatic creation of pyramids. Hexa Core allows for reduction in number of elements for quicker solver run time and better convergence. Figure 105: Free Mesh Type = Hexa Core (p. 234) shows a **MultiZone** mesh that was generated when **Free Mesh Type** was set to **Hexa Core**. Notice the upper section that was able to be mapped meshed, and the lower section that was free meshed because it could not be mapped meshed.

Figure 105: Free Mesh Type = Hexa Core



- **Element Order** - Refer to Method Controls and Element Order Settings (p. 196). The default is **Use Global Setting**.
- **Src/Trg Selection** - Defines the source and target selection type according to the following choices (the default is **Automatic**):
  - **Automatic** - The **Automatic** option generally works fine for simple sweep configurations, but if there are multiple levels of sweeps it is often best to manually define the source faces.
  - **Manual Source** - You select the faces that will be used as sources (and targets) using the **Source Scoping Method** you specify. **MultiZone** treats all sources/targets as sources, as imprinting can occur from either side. For additional details, refer to the description of **Source Scoping Method**, **Source**, and **Source Named Selection** below.

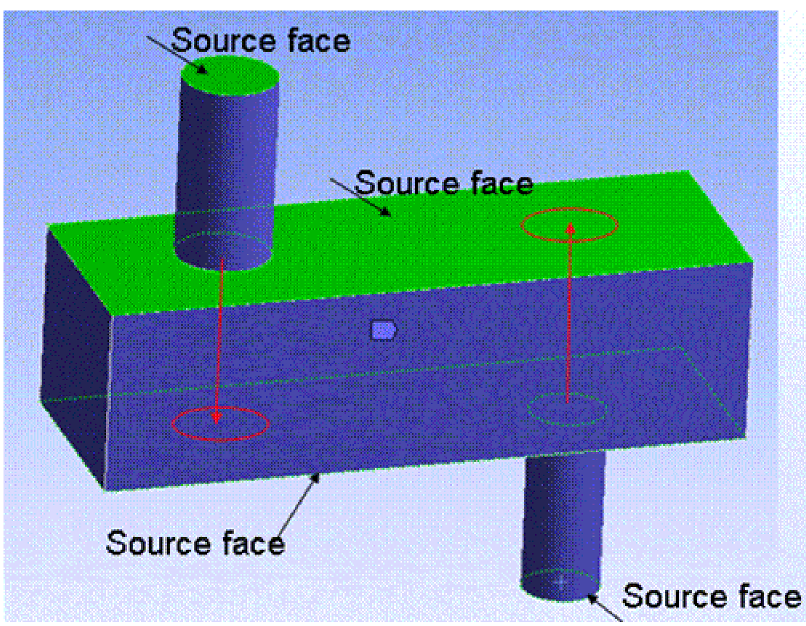
- **Source Scoping Method** -Defines the method for choosing a source face. **Geometry Selection** enables you to select sources/targets manually using the **Source** option. **Named Selection** enables you to choose one Named Selection as a source/target using the **Source Named Selection** option.

- **Source** - Select the faces that need to be imprinted for proper geometry decomposition. This option is available if you select **Geometry Selection** as your **Source Scoping Method**. The faces you select can be either "sources" or "targets," but all of them will be treated as sources by **MultiZone**, as shown in Figure 106: Source Face Selection for MultiZone (p. 235).

Note:

To make source face selection easier, select **Annotation Preferences** from the Toolbar and then deselect **Body Scoping Annotations** in the **Annotation Preferences** option box to toggle the visibility of annotations in the **Geometry** window. For example, after scoping **MultiZone** to a body, the body will be displayed using a blue solid annotation. Turn off the body scoping annotations, then select the source faces. For picking internal faces, the **Hide Faces** right-click option may help you to see inside a body. For example, you can select external faces in the **Geometry** window and then use the **Hide Faces** option to hide the selected faces (making it easier to select the internal faces).

Figure 106: Source Face Selection for MultiZone



- **Source Named Selection** - Choose an existing Named Selection to select the faces that need to be imprinted for proper geometry decomposition. This option is available if you select **Named Selection** as your **Source Scoping Method**.

- **Sweep Size Behavior** – Enables you to set a **Sweep Element Size** to define the mesh spacing (default), or to select **Sweep Edges** to remove edges and prevent them from constraining the source faces.

- **Sweep Element Size** - Enables you to set an element size to define the mesh spacing along the sweep path from source to target faces. If this control is set to a non-zero value, sizing controls applied to the selected bodies as curvature and proximity refinement and/or local sizing are ignored.

  The **Sweep Element Size** setting is ignored if hard size controls are applied to side edges/faces. If multiple bodies with the same sweep direction have different sizes set for **Sweep Element Size**, the smallest size is used and the others are ignored.

  Clicking the check box adds this setting to the Workbench parameters, enabling you to use element size settings as a variable design point when creating multiple solutions.

- **Sweep Edges**- This option should be used with an edge sizing control. The edge sizing control defines the distribution along the sweep path, and can also affect the source face. Use this option to remove the influence of the edge sizing from the source face mesh. That is, the edges selected will only influence the sweep path and not the source faces.

- **Preserve Boundaries**- Preserves only the protected topologies (See Protecting Topology Defined Prior to Meshing (p. 180)) or all features in the model. **Protected** is the default.

- **Mesh Based Defeaturing** - "Filters" edges in/out based on size. **Off** by default. If set to **On**, a **Defeature Size** field appears where you may enter a numerical value greater than 0.0. By default, the value of this local **Defeature Size** field is the same as the global Defeature Size (p. 106). If you specify a different value here, it will override the global value. Specifying a value of 0.0 here resets the tolerance to its default.

- **Minimum Edge Length** - Read-only indication of the smallest edge length in the model.

- **Write ICEM CFD Files** - Sets options for writing Ansys ICEM CFD files. Refer to Writing Ansys ICEM CFD Files (p. 84) for details.

Note:

For detailed information about **MultiZone**, refer to MultiZone Meshing (p. 343). For general information on applying **MultiZone** in combination with other mesh method controls, refer to Meshing: Mesh Control Interaction Tables (p. 435).

Notes on Scoping for the MultiZone Mesh Method

You can use the **MultiZone** mesh method in combination with other solid mesh methods in a multibody part, and the bodies will be meshed with conformal mesh.

If a multibody part contains some bodies that are scoped to be meshed with **MultiZone** and other bodies that are not scoped with any mesh method, these other bodies will be meshed with the default mesh method.

Refer to Conformal and Non-Conformal Meshing (p. 21) for information about conformal meshing.

Cartesian Method Control

The **Cartesian** method creates unstructured hexa mesh of mostly uniform size, aligned to the specified coordinate system, and fits it to the geometry. The element size should be smaller than the thickness of the model to prevent the mesher from defeaturing (not capturing) that portion of the model. Alternatively, the defeaturing could be helpful to eliminate "dirty" geometry smaller than the element size.

This method is useful when the geometry features align well with a coordinate system and a regular mesh is desired. Models for explicit dynamics, organic models (models without many feature edges), process industry and electronic components are good examples that could benefit from this mesh method. This method is also recommended for simulating the printing process in Additive Manufacturing.

Note:

The **Cartesian** method must be applied to an entire part. If a body in a multibody part is selected, all bodies in the part are added to the Geometry selection.

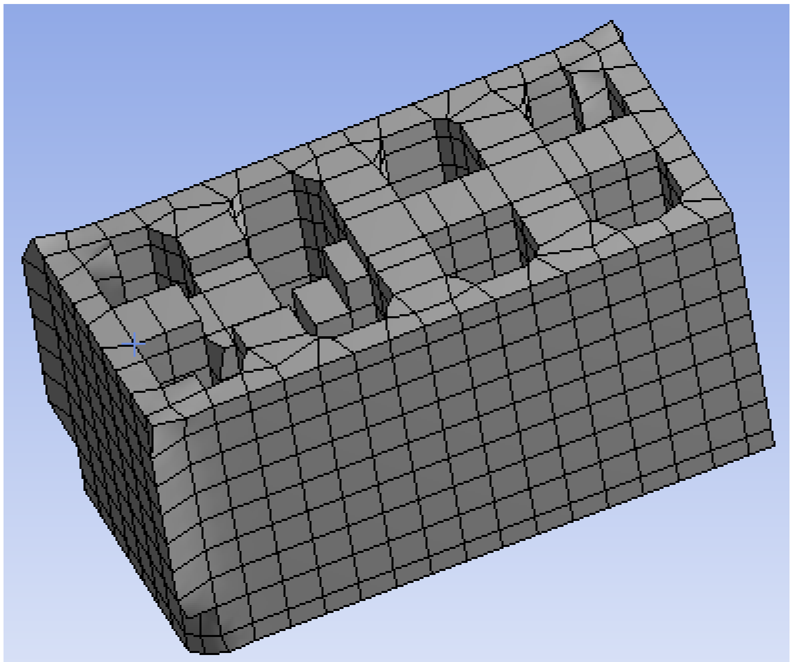
When you choose the **Cartesian** Method option, the Details View expands to include additional settings, many of which are unique to this option. For basic usage that involves obtaining a Cartesian mesh, the procedure is to apply a **Method Control** to one or more bodies, set **Method** to **Cartesian**, and accept the default values of the various settings.

For advanced or specialized usage, adjust the settings as needed. The following is a description of each of these settings.

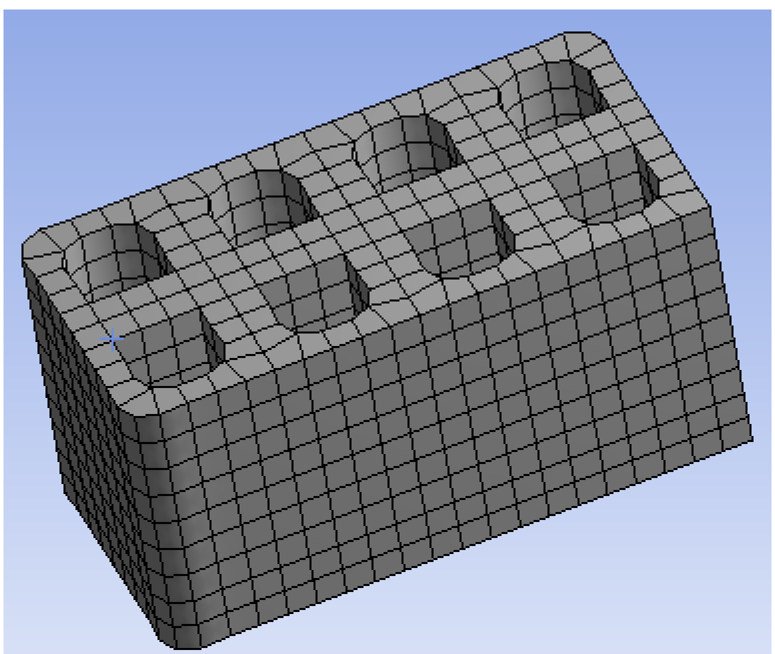
- **Element Order** - Refer to Method Controls and Element Order Settings (p. 196).
- **Type** allows you to specify how edge size is determined. Edge size is considered a soft size, allowing for small variations depending on geometry and other settings.
  - **Element Size** - You can use the **Default** setting, which is based on global element size, or manually set the element size.

    You can parameterize the **Element Size** by clicking the check box to the left of the label.
  - **Number of Divisions** - You can use the **Default** setting, which is obtained from the model's bounding box size in the z-direction divided by the global element size, or manually set a **Number of Divisions in Z-Dir**. The nominal size found from the computation of the number of divisions is also used for the spacing in the x and y dimensions.

    You can parameterize the **Number of Divisions in Z-Dir** by clicking the check box.
- **Algorithm** is set to **Body Fitted**.
- **Spacing Option** allows you to force split lines to better capture model features. Select one of the following options.
  - **Default** uses the specified element size for X, Y and Z spacing. This may result in distorted or missing elements.



– **User Controlled**, with **Key Points Selection** set to **Automatic**, looks at all vertices in the model and adds split lines to follow the features, within the specified **Tolerance**. It will then adjust the mesh spacing between neighboring split lines so that the hex mesh follows the features.

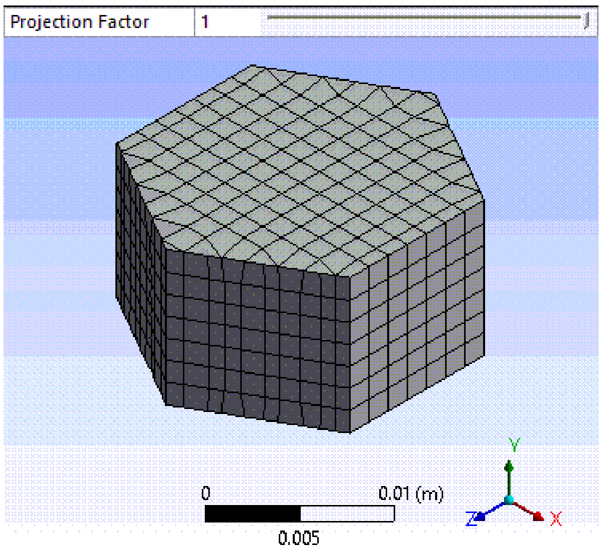


– **User Controlled**, with **Key Points Selection** set to **Manual**, allows you to select vertices manually to create split lines in each of the X, Y and Z directions. Mesh spacing is adjusted between neighboring splits. Additional split lines may be added by the mesher as required to follow the features.

- **Projection Factor** allows you to set the balance between mesh quality and capturing the geometry.

Set a value between 0 and 1. If the **Projection Factor** is set to 0, the Cartesian mesh will have high quality hexa elements that approximate the geometry surface via stairstepping. Increasing the value of **Projection Factor** will force the mesh to more closely follow the geometry surface, but with reduced element quality. If the **Projection Factor** is set to 1, most of the boundary nodes will project to the geometry and perturb only those causing bad element quality. However, some hexa elements may have pairs of coplanar faces to fit a planar geometry surface.

*The following demo is presented as an animated GIF. View online if you are reading the PDF version of the help. Interface names and other components shown in the demo may differ from those in the released product.*The image shows the effect of decreasing the **Projection Factor** for a simple geometry with two sides aligned with the global coordinate system.



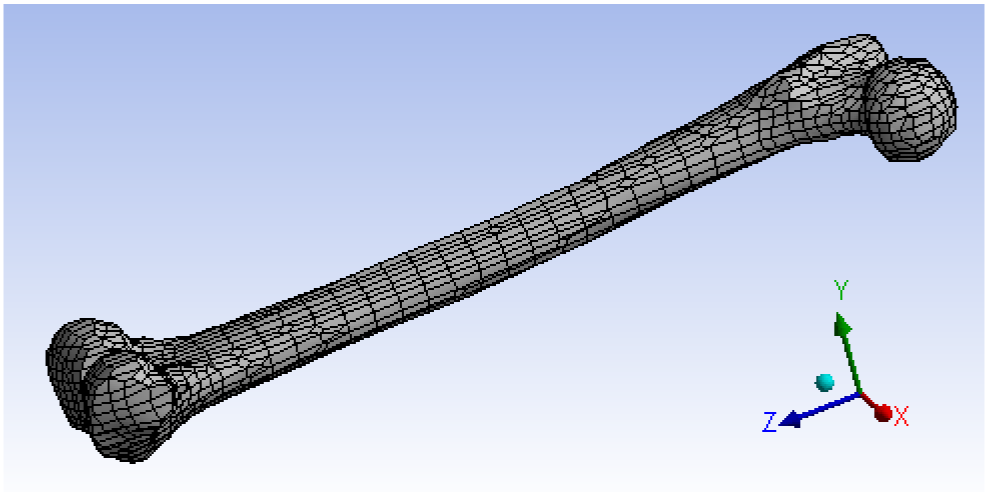
The default value of **Projection Factor** is 0 for Additive Manufacturing Process simulations. Specifically, if the AM Process object is present in the Project tree when a Cartesian method is added, the **Projection Factor** is automatically set to 0. When you change the value to something other than 0 for AM simulations, you are generating supports from the meshed body, the **Projection Factor** *must* be set to 0.

- **Project in constant Z-Plane** - This option is useful for print simulation in Additive Manufacturing.

  If enabled, the x and y coordinates of the Cartesian mesh are modified while maintaining a constant height in the Z-direction. The final mesh is much tighter to the geometry. This option is enabled by default if the AM Process object is present in the Project tree.

- **Stretch Factor in X** - (Also available for **Y** and **Z** direction). You can use these controls to modify the Aspect Ratio of the hexa mesh in the selected dimension(s).

  This is useful to reduce the element count for a geometry that may be elongated or shortened in one dimension. In this image, **Stretch Factor in Z** has been set to 3.0 while X and Y remain at 1.0. Mesh elements sizes can be up to about 3x larger in the z-dimension than x-dimension and y-dimension, within the constraint of the **Projection Factor**.



Note:

The process for applying the **Stretch Factor** depends on how edge sizing is detemined, however the end result on aspect ratio is the same.

- If **Type** = **Element Size**, then the stretch factor will apply a multiplier to the element size, aligned to a selected dimension. For example, if **Element Size** = 0.1 and stretch factors of 0.5, 1, and 2 are applied, then the dimensional mesh element size will be approximately 0.05, 0.1 and 0.2.
- If **Type** = **Number of Divisions**, then the stretch factor will scale the other dimensions while preserving the specified number of divisions in the z-dimension. For example, if stretch factors are 0.5, 1, and 2, then the nominal element size will be scaled by approximately 1/4, 1/2 and 1 to preserve the number of divisions in the z-dimension.

- **Coordinate System** allows you to choose the coordinate system to which the mesh is aligned. The global coordinate system will be used by default.

Layered Tetrahedrons Method Control

The **Layered Tetrahedrons** mesh method creates unstructured tetrahedral mesh in layers based on a specified layer height and fits it to the geometry.

This method can be used for simulating the printing process in Additive Manufacturing as the build parts must conform to a mesh with fixed step sizes in the global Z direction.

The meshing process involves the following approach:

- An initial surface mesh is generated based on the settings defined.
- The mesher generates mesh layers based on the layer height specified. The starting layer is the z-location of the plane from which the mesh layers will be generated.

- Nodes on the geometry within the tolerance specified (**Relative Tolerance**) to the layer plane are projected to the plane during the initial layering operation.

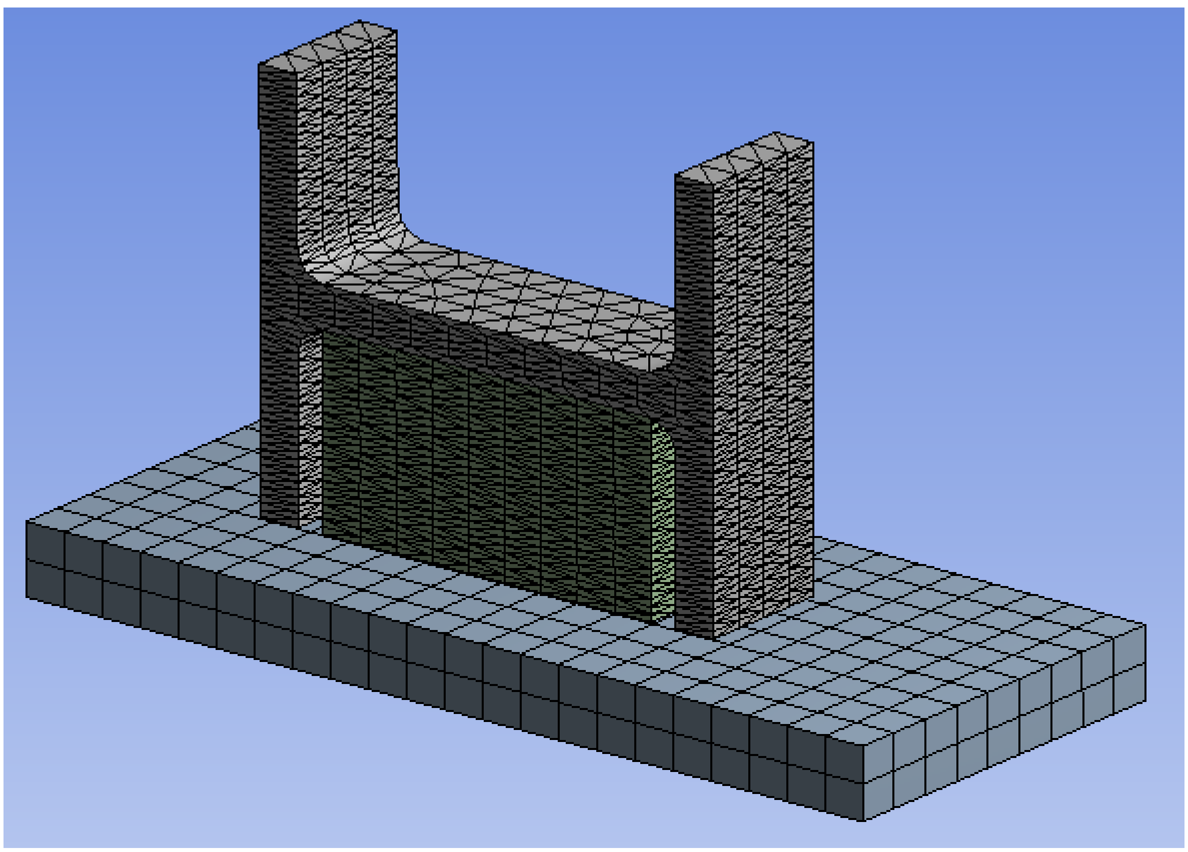
  Feature nodes and corner nodes determined based on the **Feature Angle** and the **Corner Angle**, respectively, will be preserved.

- Problematic sliver faces are identified based on **Sliver Triangle Height** and are collapsed or fixed to improve quality.

- Geometry faces overlapping the layer plane are identified based on the **Overlapping Angle** and are inflated. Additionally, geometry faces close to the layer planes which will lead to bad quality tets are identified based on the **Layer Height** and proximity to the plane. These faces will be inflated away from the layer planes to create space for better quality tets based on the **Inflate Relative Tolerance** value specified.

- The improved surface mesh is then filled with tetrahedral mesh conforming to the mesh layers and the tetrahedral mesh quality is improved.

Figure 107: Layered Tetrahedrons Mesh



Note:

- Bodies with shared topology are not supported with this mesh method. Conformal mesh will not be created even when the bodies have shared interface(s).

- This method cannot be used in conjunction with mesh controls such as **Inflation**, **Refinement**, **Match Control**, **Pinch**, **Face Meshing** and **Edge Sizing** controls.

- The mesh is not associated back to geometry. See Association Using Named Selections (p. 244) for more details.

When you choose the **Layered Tetrahedrons** Method option, the Details View expands to include additional settings, many of which are unique to this option. For basic usage, the procedure is to apply a Method Control to the body and set **Method** to **Layered Tetrahedrons**. Set the **Layer Height** and accept the default values of the various settings.

Meshing Recommendations

- The **Layered Tetrahedrons** method is available only when **Use Adaptive Sizing** is set to **No**.
- Minimum and maximum sizes should be decided based on the model.
  - The minimum size should be decided based on the features which are to be resolved and the **Layer Height**. You should set a value smaller than the **Layer Height**.
  - The **Element Size** size should not be greater than 6 times the minimum size specified. The mesh quality reduces as the **Element Size** / **Min Size** ratio increases.
  - The **Max Size** does not influence the layered tet mesher.
- Set the **Growth Rate**, the recommended range of values is 1.2 to 2.
- The recommended range for the **Curvature Normal Angle** is up to 36 degrees.
- The **Sliver Triangle Height** is based on the minimum size (p. 108) specified. The default value is 10% of the minimum size

  You should use a sliver height not more than 50% of min size.
- Generating thin pockets of mesh can be avoided by increasing **Relative Tolerance** and / or **Inflate Relative Tolerance**. The recommended range for **Relative Tolerance** is between 0.01 to 0.02. The recommended range for **Inflate Relative Tolerance** is between 0.1 to 0.3.

The **Layered Tetrahedrons** mesh method includes the following advanced settings:

- **Element Order**: The default is **Use Global Setting**. See Method Controls and Element Order Settings (p. 196) for details.

  Note:

  Only straight sided mid-nodes are available for quadratic element type.

- **Generate Layers Using Facets**: Generates layered tetrahedron mesh for the given model using facets. This allows you to skip the surface mesh generation before slicing. Thus, slicing operation is performed directly on CAD. The default value is **No**.
- **Repair Facets**: Enables you to repair CAD facets. When set to **Yes**, **Repair Facets** allows you to fix Sliver facets and facet intersections aggressively. The default value is **No**.

- **Layer Start**: This is the z-location of the plane from which the mesh layers will be generated. The default value is the min Z-coordinate of the bounding box enclosing the scoped bodies. For Additive Manufacturing simulations — more specifically, when the AM Process object is available in the Project tree — the value is automatically set to the Z-coordinate of the top of the baseplate by default.

- **Relative Tolerance**: Nodes within the given tolerance value to the layer plane will be projected to the plane during the initial layering operation. The default value is 0.01 (1%) which can be used for most cases. The recommended range of values is 0.01 to 0.02 (1-2%). For extreme cases, you may use a value of up to 0.05.

- **Inflate Relative Tolerance**: This relative tolerance is used to improve the surface mesh in the thin layer regions after the layering operation is done. This tolerance is used to move (inflate) nodes away to the value specified to improve the quality. The default is 0.1 i.e., 10% (of the layer height). One usually gets good results when this tolerance is used in range of 0.1 to 0.3. Largest acceptable value for this setting is 0.5, which should be used very carefully.

- **Overlapping Angle**: Geometry faces overlapping the layer plane are identified based on the overlapping angle and are inflated. The default is 155 degrees. The acceptable range is an angle greater than the **Feature Angle** and less than 180 degrees.

- **Defeature Layer Volume**: Specifies the threshold volume for defeaturing a thin end layer of cells. This setting is used only if a very thin end layer is expected since such layers may cause mesh quality issues.

- **Aggressive Inflate Option**: Enables inflation of faces in the proximity of the layer planes to improve mesh quality in addition to inflation to resolve overlapping faces. This option is set to **Yes**, by default. In some cases, the aggressive inflation may introduce some intersections in the surface mesh, in which case you can set the **Aggressive Inflate Option** to **No**.

- **Aggressive Tetrahedrons Improvement**: Activates the tetrahedron mesh improvement routines. It allows you to remove the thin tetrahedrons formed during meshing. The default value is **No**. When set to **Yes**, allows the mesh to do aggressive tetrahedron improvements.

- **Sliver Triangle Height**: Problematic sliver faces are identified based on **Sliver Triangle Height** and are collapsed or fixed to improve quality. The **Sliver Triangle Height** is based on the minimum size (p. 108) specified. The default value is 10% of the minimum size. Based on the model, you may need to increase the value, however, it is recommended that the value should not be greater than 50% of the min size.

- **Feature Angle**: Feature nodes determined based on the **Feature Angle** will be preserved during meshing. The default feature angle is 40 degrees. The acceptable range is from 0 degrees to an angle less than the specified **Overlapping Angle**.

- **Corner Angle**: Corner nodes determined based on the **Corner Angle** will be preserved during meshing. The default is 90 degrees. The acceptable range is from 0 to 180 degrees.

Association Using Named Selections

The mesh generated using the Layered Tetrahedrons method is not fully associated to the geometry. To have mesh associated to the geometry, define Named Selections on the faces on which association is required prior to meshing.

Note:

Edge and vertex named selections are not considered for mesh association.

Limitations

The **Layered Tetrahedrons** method has the following limitations:

- Multi-body parts with and without shared topology are not supported. You should separate them into individual parts in CAD.
- The model should not have other suppressed bodies.
- The mesh is not associated back to geometry. See Association Using Named Selections (p. 244) for more details.
- Only straight sided mid-nodes are available for quadratic element type.
- Contact must be used between two bodies even if they have a shared interface.

Note:

Conformal mesh will not be generated even if there is a shared interface between bodies.

Particle Method

Particle Method is available only when the **Physics Preference** is set to **Explicit** in the **Mesh Details** view and for bodies with **Reference Frame** set to **Particle**. **Particle Method** generates cloud of particles of specified diameter within the body.

Note:

- **Particle Method** does not support multi body parts.
- **Particle Method** supports only solid bodies.
- Solid bodies must have water-tight facets to be meshed with **Particle Method**.

Right-click the **Mesh** object and click **Insert > Method > Particle** to access the **Particle Method**. In the **Details** view, the following options are available:

Scope

Scoping Method: Allows you to select **Geometry** or **Named Selection**. The default value is **Geometry**.

Definition

Suppressed: Allows you to suppress the method control. The default value is **No**. When is set to **Yes**, the **Active** field is available. **Active** is a read only field and provides the number of suppressed parts.

Method: Denotes the selected method.

Particle Diameter: Allows you to define the diameter of particles to be generated.

Control Message: Provides message when the **Particle Method** is scoped to an unsupported body. You can click **Control Message** to view the error message.

Note:

- If **Thick Shells and Beams** option is turned off, the particles are displayed as crosses based on their centers. To visualize the particles as spheres **Thick Shells and Beams** option must be turned on.
- Error message for mesh failure referring to non-conformal facets means that the input body does not have water-tight facets. The input must be corrected in the geometry. Refaceting the geometry can help to resolve this issue in some cases.

Setting the Method Control for Surface Bodies

The options described below are available for surface bodies.

Quadrilateral Dominant Method Control

If you select the **Quadrilateral Dominant** method (default), the body is free quad meshed. The **Quadrilateral Dominant** mesh method includes the following settings:

- **Element Order** - Refer to Method Controls and Element Order Settings (p. 196).
- **Free Face Mesh Type** - Available for most analyses and can be set to either **Quad/Tri** (default) or **All Quad**.

Note:

- If you are using the Quadrilateral Dominant mesh method with inflation (p. 291) and the Size Function (p. 100) is on, the mesh size of the last inflation layer will be used for the corresponding Quadrilateral Dominant boundary mesh size.

- If you apply a local **Sizing** control (p. 248) to a solid body with a **Method** control set to **Hex Dominant** (p. 222) or **Sweep** (p. 223), or to a sheet body with a **Method** control set to **Quadrilateral Dominant** (p. 245), a near uniform quadrilateral mesh will result on all affected faces on a body meshed with **Hex Dominant**, on the source face meshed with **Sweep**, and on all affected faces meshed with **Quadrilateral Dominant**. To obtain even more of a uniform quadrilateral mesh, set the **Behavior** (p. 262) of the **Sizing** control to **Hard**.

- If you are meshing a multibody part that contains a mix of line bodies and surface bodies, all surface bodies and all line bodies that share edges with surface bodies will be meshed with the selected surface mesh method. Any remaining line bodies (where only vertices are shared with surface bodies) will always be meshed with the **Quadrilateral Dominant** mesh method.

Triangles Method Control

If you select the **Triangles** method, an all triangle mesh is created. For information on the **Element Order** option, refer to Method Controls and Element Order Settings (p. 196).

MultiZone Quad/Tri Method Control

If you select the **MultiZone Quad/Tri** method, a mesh of quads and/or triangles is created over the entire part of the selected body, depending on values that you enter for the options described below.

MultiZone Quad/Tri is a patch independent method.

Surface Mesh Method - Instructs **MultiZone Quad/Tri** to use the **Program Controlled**, **Uniform**, or **Pave** method to create the mesh.

- **Program Controlled** - Automatically uses a combination of **Uniform** and **Pave** mesh methods depending on the mesh sizes set and face properties. This is the default method.

- **Uniform** - Uses a recursive loop-splitting method which creates a highly uniform mesh. This option is generally good when all edges have the same sizing and the faces being meshed do not have a high degree of curvature. The orthogonality of the mesh from this method is generally very good.

- **Pave** - Uses a paving mesh method which creates a good quality mesh on faces with high curvature, and also when neighboring edges have a high aspect ratio. This approach is also more reliable to give an all-quad mesh.

Note:

The **MultiZone Quad/Tri** method ignores the Sizing Options (p. 100) when **Surface Mesh Method** is set to **Uniform**. In such cases, **Element Size** acts as a hard size.

Element Order - Refer to Method Controls and Element Order Settings (p. 196).

Free Face Mesh Type - Determines the shape of the mesh elements. Can be **All Quad**, **All Tri**, or **Quad/Tri** (default) for a mesh of pure quad, pure tri, or a combination of quad/tri elements respectively.

Element Size - Allows you to specify the element size used for the selected geometry. Applicable only when **Surface Mesh Method** is set to **Uniform**. Otherwise, uses the global **Element Size** (p. 98).

Mesh Based Defeaturing - "Filters" edges in/out based on size and angle. Can be **On** or **Off**. By default, this local **Mesh Based Defeaturing** setting is the same as the setting of the global Mesh Defeaturing (p. 106) control. When **Mesh Based Defeaturing** is **On**, a **Defeature Size** field appears where you may enter a numerical value greater than 0.0. By default, the value of this local **Defeature Size** field is the same as the global Defeature Size (p. 106). If you specify a different value here, it will override the global value. A recommended setting is at least one-half the value set for **Element Size** to assure a successful mesh. Specifying a value of 0.0 here resets the tolerance to its default.

Note:

When the global Mesh Defeaturing (p. 106) control is on but **Use Adaptive Sizing** is set to **Yes**, the default defeaturing performed for **MultiZone Quad/Tri** includes defeaturing based on the dihedral angle between the faces as well as edge length defeaturing based on the smallest element size set by the user.

Sheet Loop Removal - Removes holes on surface bodies based on size. If set to **Yes**, a **Loop Removal Tolerance** field appears where you may enter a numerical value greater than 0.0. By default, the value of this local **Loop Removal Tolerance** field is the same as the global Loop Removal Tolerance (p. 193). If you specify a different value here, it will override the global value. The mesh simply paves over holes smaller than the **Loop Removal Tolerance** setting. Holes with boundary conditions applied to them will not be removed from the mesh. Any boundary conditions applied to holes that were removed from the mesh will not be respected by the solver.

Minimum Edge Length - Read-only indication of the smallest edge length in the part.

Write ICEM CFD Files - Sets options for writing Ansys ICEM CFD files. Refer to Writing Ansys ICEM CFD Files (p. 84) for details.

Usage Information for the MultiZone Quad/Tri Mesh Method Control

The following usage information is applicable to the **MultiZone Quad/Tri** mesh method:

- Base mesh caching is not supported for **MultiZone Quad/Tri**, so a change to inflation controls requires remeshing.
- Using **MultiZone Quad/Tri** may allow meshing over very small bodies in a multibody part. This may lead to a solver error if a body load is associated with that body. If this is the case, you must suppress the body (p. 486) before solving your model.
- Surface bodies with specified variable thickness are not protected topology. To prevent faces and their boundaries from being meshed over, create an individual Named Selection for each thickness.
- The **MultiZone Quad/Tri** mesh method supports mesh connections (p. 444).

- You can use the **MultiZone Quad/Tri** mesh method in combination with other surface mesh methods in a multibody part, and the bodies will be meshed with conformal mesh. If you select the **MultiZone Quad/Tri** mesh method to mesh a multibody part that contains a mix of line bodies and surface bodies, all surface bodies and all line bodies that share edges with surface bodies will be meshed with the selected method. Any remaining line bodies (where only vertices are shared with surface bodies) will be meshed with the **Quadrilateral Dominant** (p. 245) mesh method. Refer to Conformal and Non-Conformal Meshing (p. 21) for more information about conformal meshing.

- When meshing multibody parts having solid bodies and sheet bodies with some faces shared, it is best to mesh the shared faces with **MultiZone Quad/Tri**. If the faces are meshed using the solid body mesh method control, the **MultiZone Quad/Tri** body will often fail.

Caution:

Multiple environments with different loadings may over-constrain the **MultiZone Quad/Tri** mesher such that the mesher may not be able to return a mesh for the given inputs. If discretization error is not an issue, the mesher will be less constrained if you duplicate the model and change the environment instead of adding multiple environments under the same model.

Mesh Grouping Control

Mesh Grouping identifies bodies that should be grouped together for assembly meshing (p. 367) algorithms and is available only when assembly meshing algorithms are being used. See Defining Mesh Groups (p. 389) for details.

Sizing Control

The Sizing control sets:

- The element size for a selected body, face, or edge.
- The number of divisions along an edge.
- The element size within a user-defined "sphere of influence" that can include a selected body, face, edge, or vertex. This control is recommended for local mesh sizing. The control must also be attached to a coordinate system if it is to be scoped to anything other than a vertex.
- The element size within a user-defined "body of influence." The body of influence will influence the mesh of the body to which it is scoped, but the body of influence itself will not be meshed.
- The scale factor for a selected body, face, or edge. The scale factor enables you to define the local element, minimum, and defeature sizes as factors of the global element size.
- The minimum mesh sizing used for a selected body, face, or edge. This setting overrides the default global sizing.

The Sizing control is described in the following sections:

Notes on Element Sizing

Remember the following notes when using the Sizing control:

- Visual aids are available to assist you. When you pick an edge, the edge length is displayed. A circle is displayed adjacent to the cursor whose diameter indicates the current setting in the **Element Size** field. The scale ruler is displayed below the graphic and provides a good estimate of the scale of the model. Also for edge sizing, if you specify a bias, and if you set **Element Size** to a value other than **Default**, the size control will be displayed graphically with the initial mesh density (including any specified bias) in the **Geometry** window.

- **When Applying Sizes to Faces:** Faces adjacent to a face that has a scoped size control applied to it respect the source as part of the sizing control. Meshes on the adjacent faces will transition smoothly to the size on the scoped face. When size controls that have differing sizes are on adjacent entities, the adjacent topology receives the smallest size.

- **When Applying Sizes to Edges:** If possible, the meshing algorithm places the requested number of divisions on the specified edge. Otherwise, the algorithm adjusts the number to allow a successful mesh generation. When **Behavior** is set to Hard, the number of divisions is given a higher priority and the mesher may fail rather than adjust the number of divisions. In case of failure, inspect the messages for more information as to what constraints may be causing it.

  Note:

  - In **MultiZone**, a closed edge is represented by more than one blocking topological edges. Therefore, user-specified edge node distribution size may not be respected. However, the number of intervals will still be respected.

  - For split curves in rotated edge association, only interval count, not distribution, will be respected.

- **When Sweeping (p. 323):** Consider the following when applying size controls to source and target geometry:

  - If your sizing controls are scoped to either the source or target face, the mesher will transfer the size control to the opposite face. If you have a size control on both faces, the size on one of the faces will be used. That face is automatically determined by the software. However the size on the edges of the target face will not be affected if no sizes are explicitly defined on these edges.

  - Edge sizing applied to a target face is respected only if **Behavior** is **Hard**.

  - If you have a sphere of influence on a possible source or target face, the face with the most spheres will be chosen as the source face. The edge mesh of the source face affected by the sphere of influence will not affect the target face. This may prevent the model from sweeping with acceptable element quality. To avoid this, place the sphere of influence on the edges of both the source and target face.

– Applying sizes, regardless of type (that is, size, number of divisions, sphere of influence), to the edges of possible source and target faces will only affect the faces that use these edges.

If you want to control a side area, the problem must be properly constrained such that the interval assignment does not override your size control. The divisions on the edge may decrease in order to make the body sweepable. When using a meshing process other than swept meshing, the divisions can only increase. When applying a size to a part that is sweepable, the resulting mesh may have fewer divisions on the edge than specified due to the interval assignment logic of the sweepers.

When sweeping a model, if you use the **Sphere of Influence** sizing control and the sphere is not touching the edges of a side area or is totally enclosed in the body, the sphere will have no effect.

When sweeping a closed torus (shown below) with an applied size on the face of the torus, the number of divisions that will result on the torus is governed by the arc length between the caps of the surface on the inside of the torus.

Figure 108: Sweeping a Closed Torus

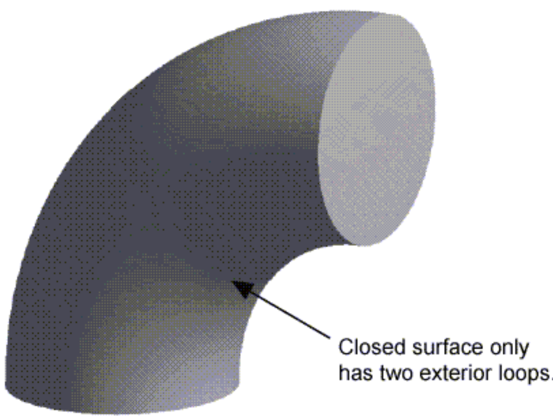
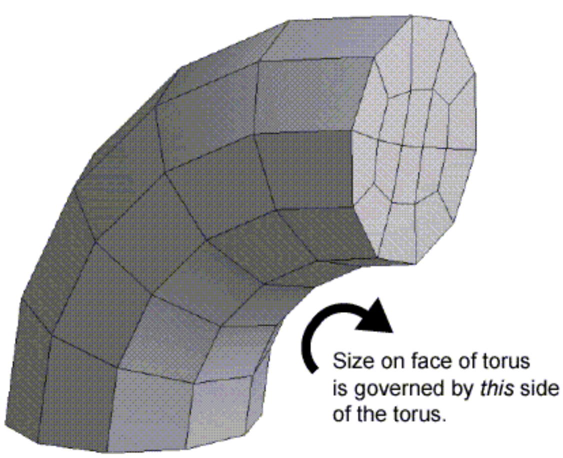


Figure 109: Resulting Mesh for Closed Torus



- Using the **Sphere of Influence** sizing control may not have any effect on the generated mesh if the control is scoped to the Body of a Line Body.

- In general, users are discouraged from defining a body of influence and a sphere of influence such that the regions of influence overlap. In cases where elements fall within overlapping bodies/spheres of influence, elements will be created using the **Sphere of Influence** sizing that appears lowest in the Tree.

- Regardless of the value for the sizing control you set, other factors such as edge and face curvature and the proximity of small features may override the effect of the sizing control.
- When using mapped face controls, or **Sweep** or **MultiZone** parallel edge assignments are handled automatically for mapped faces. That is, for a mapped face, there are two sets of parallel edges. If you increase or decrease the sizing on one edge, the same increase or decrease occurs automatically on the other edge to ensure a mapped mesh is possible. If a model contains a row of mapped faces (such as the sides of a box), you can set a number of elements on one edge and the same number of elements will be forced on all side/parallel edges. Setting **Behavior** to **Hard** will give that edge priority. In case of conflict between two edges that should be parallel but have different hard sizing controls assigned to them, the software will either fail and print a message about the conflict, or a free mesh will be given for that face.
- When using **MultiZone**:
  - If you scope bias settings to parallel edges, and the sizings create conflicts, they are applied according to the following priority:
    1. Double bias edge (highest priority)
    2. Single bias edge
    3. No bias edge
  - Setting **Behavior** to **Hard** on an edge gives that edge a higher priority. That is, a hard edge sizing takes priority over a soft edge sizing, but among sizing controls that are hard or soft, the order of priority is double, single, then no biasing.
- If you apply a local **Sizing** control (p. 248) to a solid body with a **Method** control set to **Hex Dominant** (p. 222) or **Sweep** (p. 223), or to a sheet body with a **Method** control set to **Quadrilateral Dominant** (p. 245), a near uniform quadrilateral mesh will result on all affected faces on a body meshed with **Hex Dominant**, on the source face meshed with **Sweep**, and on all affected faces meshed with **Quadrilateral Dominant**. To obtain even more of a uniform quadrilateral mesh, set the **Behavior** of the **Sizing** control to **Hard**.
- If several sizing controls are attached to the same edge, face, or body, the last control is applied. If a sizing control is placed on an edge and then another is placed on a face or body that contains that edge, the edge sizing takes precedence over the face or part sizing.
- If you have adjusted the element size, then changed length units in a CATIA or ACIS model, when you choose **Update** or **Clear Generated Data** at a Model or Project node in the Tree Outline, you may need to re-adjust the element size. The sizing control does not automatically re-adjust to match this situation.
- When using **MultiZone Quad/Tri** (p. 246) and **Surface Mesh Method** is set to **Uniform**, the **Element Size** will take priority over local face or body sizings unless the specified face or body size is smaller than the **Element Size**. This means you can use face or body sizing to obtain a finer mesh, but not to obtain a coarser mesh.
- When using Assembly Meshing (p. 367):

  - The **Element Size** (p. 255) option for **Type** is supported for local body, face, and edge sizing. Any bias options (p. 262) applied with edge sizing are ignored.
  - The **Body of Influence** (p. 257) option for **Type** is supported for local body sizing, but the body of influence cannot be scoped to a line body.
  - If you want to use a body of influence with a virtual body (p. 379), you can scope the body of influence to any body in the geometry. The body of influence does not have to be inside or even in contact with the scoped body.
  - The **Sphere of Influence** (p. 256) and **Number of Divisions** (p. 258) options for **Type** are not supported.
  - No local vertex sizing (p. 252) is supported.
  - **Contact Sizing** (p. 263) is supported. However, if contact sizing is applied to entities on a body that is scoped to a body of influence, the contact sizing is ignored.

  If any unsupported local size controls are defined prior to selection of an assembly meshing algorithm, they are suppressed when an assembly meshing algorithm is selected.

- For a table summarizing the behaviors of local sizing controls when used with various mesh methods, refer to Interactions Between Mesh Methods and Mesh Controls (p. 438).

Applying a Local Sizing Control

You can set the local sizing on a body, face, edge, or vertex.

1. Insert a Sizing control by doing one of the following:
   - Right-click a body, face, edge, or vertex, and then select **Insert** > **Sizing**.
   - In the Tree Outline, right-click the Mesh object and select **Insert** > **Sizing**.
2. If necessary, in the Details view, define the scope of the selection:

To apply local sizing to...	Do this...
A geometry selection	1. Click **Scoping Method** and select **Geometry Selection**. 2. Select a body, face, edge, or vertex. 3. In the **Geometry** field, click **Apply**.
A named selection	1. Click **Scoping Method** and select **Named Selection**. 2. Select a **Named Selection**.

3. Do one of the following, depending on what entities you selected and how you want to control the mesh sizing:

If you selected...	And you want to...	Do this...
Bodies, faces, or edges	Specify an element size to control mesh sizing	1. In the **Type** field, select **Element Size**. 2. Define the **Element Size** (p. 255).
Bodies, faces, or edges	Specify a scale factor to define the local minimum and defeature sizes as factors of the global element size	1. In the **Type** field, select **Factor of Global Size**. 2. Define the **Factor of Global Size** (p. 258).
Bodies, faces, edges, or vertices	Apply mesh sizing within the confines of a sphere	1. In the **Type** field, select **Sphere of Influence**. 2. Define the **Sphere of Influence** (p. 256).
Bodies	Apply mesh sizing by specifying an element size and controlling the mesh density based on neighboring bodies of influence	1. In the **Type** field, select **Body of Influence**. 2. Define the **Bodies of Influence** (p. 257). 3. Define the **Element Size** (p. 255).
Edges	Control mesh sizing according to a discrete number of divisions along the edge	1. In the **Type** field, select **Number of Divisions**. 2. Define the **Number of Divisions** (p. 258).

4. If necessary, define any advanced options.

   The values of the advanced options are set by default. You can change these values to apply greater control over the local sizing on an entity.

For more information on the advanced options, see Descriptions of Local Sizing Control Options (p. 254).

Note:

If **Use Adaptive Sizing** is set to **No**, any local size applied to an entity is also applied to all lower topology entities.

Descriptions of Local Sizing Control Options

Use the options described below to further define local mesh sizing. The choices that are available depend on the selected topology, physics preference, and as noted in the individual option descriptions. You can set the following local sizing options:

Definition Options:

- Element Size (p. 255)
- Sphere of Influence (p. 256)
- Body of Influence (p. 257)
- Factor of Global Size (p. 258)
- Number of Divisions (p. 258)

Advanced Options:

- Defeature Size (p. 259)
- Defeature Size Scale (p. 259)
- Influence Volume (p. 259)
- Affected Distance (p. 260)
- Capture Curvature (p. 260)
- Capture Proximity (p. 260)
- Growth Rate (p. 261)
- Local Min Size (p. 261)
- Curvature Min Size Scale (p. 261)
- Curvature Normal Angle (p. 261)
- Proximity Min Size (p. 262)
- Proximity Min Size Scale (p. 262)
- Num Cells Across Gap (p. 262)

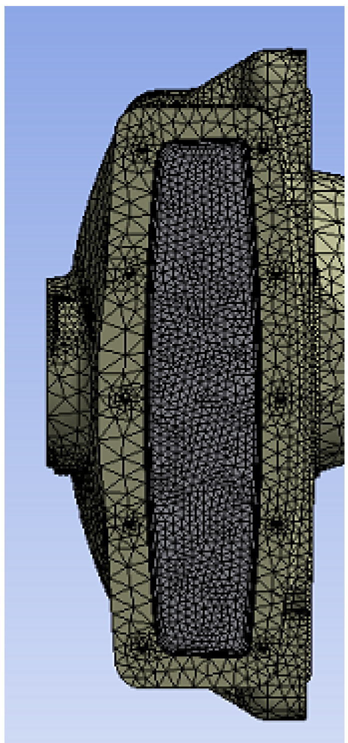
- Proximity Size Function Sources (p. 262)
- Behavior (p. 262)
- Bias Type and Bias Option (p. 262)

Element Size

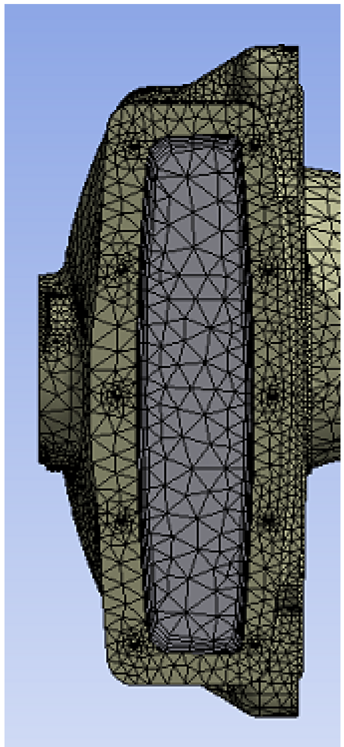
If you selected a body, face, or edge, **Element Size** (default) is one of the available options in the **Type** field. Enter a positive value (decimals are allowed) in this field. Smaller values generate more divisions. A value of "0" instructs the sizing control to use its defaults.

The following series of figures shows the effect of the **Element Size** option applied to the central face at 5mm.

Element Size set to 5mm.



Element Size default.



Additional details about the **Element Size** option differs depending on the sizing options set. The description of **Element Size** is as follows:

- For an *edge*, **Element Size** is the maximum size on the edge. It takes priority over the global Max Size (p. 105), and any face or body sizing control that includes the edge(s). If two edge sizing controls are attached to the same edge, the latter size control takes priority.
- For a *face*, **Element Size** is the maximum size on the face, which is then propagated down to the edges (unless a more local edge control is assigned). The value of **Element Size** takes priority over Max Size (p. 105) and any body size control that includes the face(s). If two face sizing controls are attached to the same face, the latter size control takes priority.
- For a *body*, **Element Size** is the maximum size on the body, *and* the maximum sizes on faces and edges of the body (unless a more local face or edge size control is assigned). The value of **Element Size** takes priority over the global Max Size (p. 105). If the body size behavior is **Hard** (p. 262), the value of **Element Size** also takes priority over the global Max Size (p. 105).

Sphere of Influence

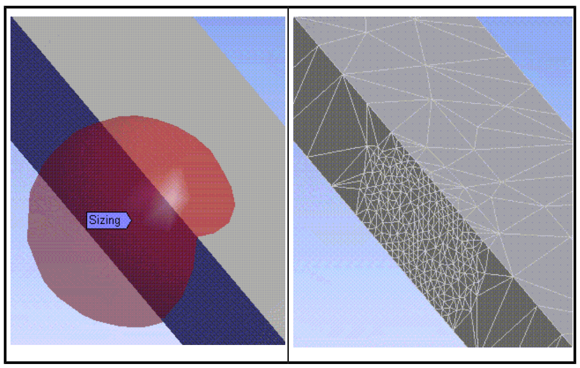
The **Sphere of Influence** option is available in the **Type** field after you select an entity such as a body, face, edge, or vertex.

If the **Sphere of Influence** is scoped to a body or vertex, the **Sphere of Influence** affects the entire body regardless of sizing options being used. If the **Sphere of Influence** is scoped to a face or edge and **Use Adaptive Sizing** is set to **Yes**, the **Sphere of Influence** only affects the face(s) or edge(s) that are scoped to the control and the transition away from those entities. If the **Sphere of Influence** is scoped to a face or edge and any other sizing option is used, the **Sphere of Influence** will affect the whole body (not just scoped face(s) or edge(s)).

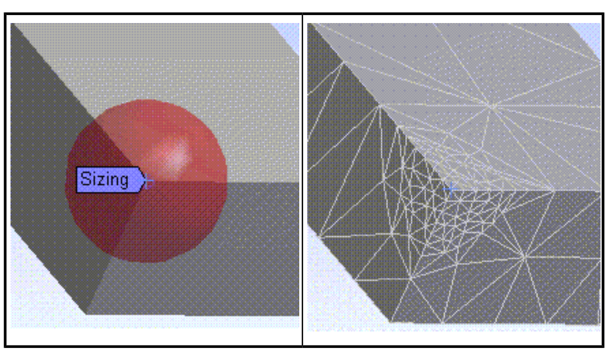
Although the **Behavior** option is not available for **Sphere of Influence**, **Sphere of Influence** behaves as a **Hard** setting. That is, in the vicinity of a **Sphere of Influence**, the **Sphere of Influence** sizing overrides pre-existing sizing information regardless of whether the pre-existing sizes are larger or smaller than the **Sphere of Influence** sizing. This is in contrast to the **Body of Influence** (p. 257) option, which behaves as a **Soft** setting.

For bodies, faces, and edges, **Sphere of Influence** allows you to apply mesh sizing within the confines of a sphere in space that you define as follows:

1. Create a local coordinate system whose origin you intend to be the center of the sphere.
2. Select this coordinate system in the **Sphere Center** field.
3. Enter the radius of the sphere in the **Sphere Radius** field.
4. Enter a value in the **Element Size** field. The element size will be applied to all topologies within the confines of the sphere. For example, if you are applying the element size to a face, the size will also be applied to the edges of that face, and to the vertices of those edges, but only within the confines of the sphere. An example is shown below.



If you selected a vertex, the *only* option available in the **Type** field is **Sphere of Influence**. The description is the same as presented above except that the center of the sphere *is* the vertex. There is no need to create or use a local coordinate system to define the center of the sphere. After applying element size to a vertex using **Sphere of Influence**, the element size is applied to all topologies connected to that vertex, such as all edges and faces containing that vertex, if they fall within the sphere. An example is shown below.



Body of Influence

The **Body of Influence** option is available in the **Type** field if you selected a body and **Use Adaptive Sizing** is set to **No**. Using this option, you can set one body as a source of another body (that is, a

Body of Influence). The **Body of Influence** will influence the mesh density of the body that it is scoped to, but it will not be a part of the model geometry nor will it be meshed. **Body of Influence** bodies are noted in the Details View of each prototype.

Although the **Behavior** option is not available for **Body of Influence**, **Body of Influence** behaves as a **Soft** setting. That is, in the vicinity of a **Body of Influence**, the specified **Body of Influence** sizing must be larger than the smallest Curvature Min Size (p. 108)/Proximity Min Size (p. 110) *and* smaller than the global Max Size (p. 105) to have an effect on the mesh size distribution. The specified **Body of Influence** sizing imposes a local maximum size on all elements that are inside the boundary of the body. This is in contrast to the **Sphere of Influence** (p. 256) option, which behaves as a **Hard** setting.

Remember the following notes when using **Body of Influence**:

- If your source body is a sphere, it is best for you to use the **Sphere of Influence** option instead of **Body of Influence**. **Body of Influence** is intended for non-spherical bodies.
- In general, users are discouraged from defining a **Body of Influence** and a **Sphere of Influence** such that the regions of influence overlap. In cases where elements fall within overlapping bodies/spheres of influence, elements will be created using the **Sphere of Influence** sizing that appears lowest in the Tree.
- You cannot apply loads, mesh controls, etc. on bodies of influence. Bodies of influence are used only to influence the sizing controls and therefore only sizing attributes can be applied to them.
- You can suppress and unsuppress bodies of influence.
- If you are using **Body of Influence** with **Match Control**, be aware that the body of influence will not be copied from one matched entity to the other. As a workaround, you can copy the body in the DesignModeler application and use both bodies as your source.
- When using Assembly Meshing (p. 367), the body of influence cannot be scoped to a line body. If you want to use a body of influence with a virtual body (p. 379), you can scope the body of influence to any body in the geometry. The body of influence does not have to be inside or even in contact with the scoped body. Although **Contact Sizing** (p. 263) is supported for assembly meshing algorithms, if it is applied to entities on a body that is scoped to a body of influence, the contact sizing is ignored.

Factor of Global Size

If you select a body, face, or edge, **Factor of Global Size** is one of the available options in the **Type** field (unless **Use Adaptive Sizing** is set to **Yes**). Choose **Factor of Global Size** and enter a value for **Element Size Factor** to define the local minimum and defeature sizes as factors of the global element size. If local sizings are defined as a factor of global size and you modify the **Element Size** (p. 98), the local sizings are recalculated.

Number of Divisions

Number of Divisions can be used with all meshers except for Assembly Meshing (p. 367) algorithms.

If you select an edge, the options available in the **Type** field are **Element Size** and **Sphere of Influence**, along with the **Number of Divisions** option. Choosing **Number of Divisions** and entering a value in the **Number of Divisions** field is an alternative to choosing **Element Size** if you are interested in having the mesh be sized according to a discrete number of divisions along an edge. If you set **Number of Divisions** to a value greater than 1000, the number of divisions will not be drawn on the edge in the **Geometry** window.

Defeature Size

For body sizing and face sizing controls, sets the local tolerance for defeaturing. Features smaller than or equal to this value will be removed when the mesh is generated. You can specify any value greater than 0.0.

If the local sizing is **Uniform**, the default local **Defeature Size** is set to the smaller of the following two values:

- Global **Defeature Size**
- 50% of the value of the local **Element Size**

For all other local sizing controls, the default local **Defeature Size** is set to the smaller of the following two values:

- Global **Defeature Size**
- 50% of the value of the local **Curvature Min Size** or **Proximity Min Size** (whichever is smaller)

For more information about setting the defeature size, see Defeature Size (p. 106).

Defeature Size Scale

Specify the scale factor for the defeature size. The default values comes from the scale factors for **Mechanical Defeature Size Factor** and **CFD Defeature Size Factor** depending on the physics preference (**CFD** physics preference uses the **CFD Defeature Size Factor**, other physics preferences use the **Mechanical Defeature Size Factor**). The default value or the value that you specify is multiplied by the global element size to determine the local defeature size.

Influence Volume

When **Element Size** is selected, the **Influence Volume** appears with **Yes** or **No** option. The default option is **No**. If you select **Yes**, the **Affected Distance** field appears. When influencing the volume, the face size control generates spheres of influence internally by automatic determination of radius and size based on the **Affected Distance**.

Note:

- Cartesian method does not support **Influence Volume** meshing.
- **Layered Tetrahedrons Method** does not support **Influence Volume** meshing.
- When **Adaptive Sizing** is set to **Yes**, **Influence Volume** option does not have effect.

Affected Distance

When **Influence Volume** is set as **Yes**, the **Affected Distance** option appears. You can enter the distance up to which the defined **Element Size** will affect the volume mesh.

Capture Curvature and Capture Proximity

If a global sizing option is defined for the model, then you can set a local sizing on a body, face, or edge to further refine the local entity's sizing. When you set a local sizing on an entity, the local sizing options default to the same values that are defined for the global size function. You can change any of these values, and the local values will take precedence over the global values when the mesh is generated.

For example, if **Capture Curvature** and/or **Capture Proximity** are selected, and the local sizing is **Uniform** (or vice versa), the local sizing option takes precedence.

Example 4: Setting a Local Sizing Option

If the global sizing option is set to **Curvature** with a 20° **Curvature Normal Angle**, and you set a local sizing of **Curvature** on a face with a 60° **Curvature Normal Angle**, the local **Curvature Normal Angle** will take precedence when the mesh is generated. In this example, the face with the local sizing control will use a **Curvature Normal Angle** of 60° even though the global value is 20°.

If using **Nonlinear Mechanical** or **CFD** physics preference, or when **Use Adaptive Sizing** is set to **No**; the following options are available and work as follows:

- **Capture Curvature** = **Yes** turns on the Curvature-based Sizing (p. 102).
- **Capture Proximity** = **Yes** turns on the Proximity-based Sizing (p. 102).
- If **Capture Curvature** and **Capture Proximity** are both set to **Yes**, then both Curvature-based Sizing (p. 102) and Proximity-based Sizing (p. 102) will be used.
- If **Capture Curvature** and **Capture Proximity** are both set to **No**, then Uniform (p. 103) sizing will be used.

Note:

Local sizing limitations include:

- If **Use Adaptive Sizing** is set to **Yes**, then you cannot set a local size function.
- If the global sizing option is set to **Capture Proximity** or both **Capture Proximity** and **Capture Curvature**, and the local sizing option is set to **Capture Curvature** or both **Capture Proximity** and **Capture Curvature** (or vice versa), the mesh on the entity is generated as though both **Capture Proximity** and **Capture Curvature** are set locally.
- If the global sizing option is set to **Capture Proximity**, and the local sizing option is set to **Capture Curvature** (or vice versa), the option that has the smallest size specified for it (either **Num Cells Across Gap** for the **Proximity** sizing, or **Curvature Normal Angle** for the **Curvature** sizing) takes precedence.

- The local **Proximity** sizing is not supported for sheet models. If both the local **Proximity** and **Curvature** sizing is defined on a sheet model, only the **Curvature** sizing takes effect.
- Local proximity sizing is not supported for edge sizing controls. To set a local edge sizing control, you must choose either **Capture Curvature** or **Uniform** (both **Capture Curvature** and **Capture Proximity** set to **No**).

Growth Rate

You may specify a growth rate for the scoped soft size of an entity (*body, face, or edge only*). The description of the scoped **Growth Rate** is the same as that of the global growth rate (p. 105) that you can set in the Details View when the **Mesh** object is selected in the Tree Outline. However, the growth rate you specify for a scoped entity *must always be smaller than or equal to* the specified global growth rate (p. 105). **Growth Rate** is not available for **Sphere of Influence**. Specifying a growth rate for a face or body affects the growth on the face or on the boundary of the body, and its effect continues outside of the scoped entity as well.

Local Min Size

You can set **Local Min Size** to specify a value that takes priority over the global **Min Size** when you select a *body*, *face*, or *edge*. If **Local Min Size** is set to **Default**, the mesher uses either the global **Min Size** or the local **Element Size** (if defined), whichever is smaller. This setting is available when **Capture Curvature** is set to **Yes**, and is useful when you want to refine the mesh based on curvature. When you set the **Local Min Size**, the mesher refines from the **Element Size** to the **Local Min Size** in curved areas, but retains the **Element Size** for flat areas.

Note:

Local Min Size may not be respected by the Sweeping or MultiZone Methods due to the interval assignment used to generate structured meshes. The min size is used on source faces, but not necessarily on side faces.

Curvature Min Size Scale

Specify the scale factor for the curvature minimum size. The value that you specify is multiplied by the global element size to determine the local curvature minimum size. The default is determined by the **Mechanical Min Size Factor** or **CFD Min Size Factor** option (p. 319) in the Options dialog box, depending on the physics preference.

Curvature Normal Angle

You may specify **Curvature Normal Angle** for a scoped entity (*body, face, or edge only*). **Curvature Normal Angle** is the maximum allowable angle that one element edge is allowed to span given a particular geometry curvature. Available only when **Capture Curvature** is set to **Yes**. You can specify a value from 0 to 180 degrees to override the global setting (where a value of 0 resets the option to its default). The default for the global setting is calculated based on the Physics Preference (p. 93).

For more information, see Curvature Normal Angle (p. 109).

Proximity Min Size

This option allows you to specify a minimum size to be used in a local proximity sizing calculation, in addition to the **Local Min Size** (p. 261). By default, **Proximity Min Size** is set equal to the default of **Local Min Size**. You can accept the default or specify a value greater than 0. For more information, see Proximity Min Size (p. 110).

Proximity Min Size Scale

Specify the scale factor for the proximity minimum size. The value that you specify is multiplied by the global element size to determine the local proximity minimum size. The default is determined by the **Mechanical Min Size Factor** or **CFD Min Size Factor** option (p. 319) in the Options dialog box, depending on the physics preference.

Num Cells Across Gap

Specify the minimum number of layers of elements to be generated in gaps when **Capture Proximity** is set to **Yes**.

Proximity Size Function Sources

This option is available if **Capture Proximity** is set to **Yes**. It determines whether regions of proximity between faces and/or edges are considered when proximity sizing calculations are performed.

For more information, see Proximity Size Function Sources (p. 110).

Behavior

If the global option **Use Adaptive Sizing** is **Yes**, or the local sizing is Uniform (both **Capture Curvature** and **Capture Proximity** set to **No**), the **Behavior** option is available for bodies, faces, and edges.

You can specify either **Soft** (default) or **Hard**, but the effect depends on the type of mesh being generated. Typically, **Hard** results in a stricter size setting than **Soft**. For example, with a tri or tet mesh, the **Hard** setting would not allow as much transition from one element size to another element size due to influences of sizing on neighboring objects. With a **Hard** setting, the element size defined would be maintained for the object to which the setting is scoped. With a **Soft** setting, the element size defined may be modified in order to respect other size settings in the vicinity of the object to which the setting is scoped.

You should take care when applying **Hard** sizes. With a mapped quad/tri or swept mesh, a **Hard** setting forces the interval edge assignment on the object to be met, and if the mesher cannot achieve the setting, the mesh could fail.

Bias Type and Bias Option

For edges only, use **Bias Type** to adjust the spacing ratio of nodes on an edge. This feature is useful for any engineering problem where nodes need to be clustered on an edge or group of edges, or if there is a need to bias the mapped mesh of a face towards a specific direction with respect to the edges of the face. **Bias Type** can be used with all meshers except the **Patch Independent Tetrahedron** method and the assembly meshing algorithms (p. 367). To use **Bias Type**, choose one of the four pre-determined patterned options depicted pictorially from the **Bias Type** drop-down menu:

> **Note:**
>
> The **MultiZone** meshing method respects all **Bias Type** options. The nodes generally follow the distribution. However, some optimization can be done to improve transition with other mesh, or to improve the quality. Setting the **Behavior** to **Hard** will give higher priority to the bias settings defined than the optimization.

Then specify a **Bias Option**. The drop-down menu enables you to choose **No Bias**, **Bias Factor**, or **Smooth Transition**:

- **Bias Factor** is defined as the ratio of the largest edge to the smallest edge. To set the Bias Factor, choose **Bias Factor** from **Bias Option** and enter a value into the field to define the ratio.

- **Smooth Transition** is defined as Growth Rate = Bias Factor^(1/(n-1)), where *n* is the number of divisions. To define Smooth Transition, choose **Smooth Transition** from **Bias Option**, then enter a value for **Growth Rate**.

> **Note:**
>
> If **Behavior** is set to **Hard**, then the number of divisions and the bias cannot be changed by the mesher. If **Behavior** is set to **Soft**, then the edge divisions can be changed but the edge will be initially meshed with the specified **Bias Factor**.

If you want to select multiple edges to apply sizing, but some of the edges do not have the same orientation, you can use the **Reverse Bias** option to manually select all of the edges. **Reverse Bias** is available when the control is applied to the edges, defined as an **Element Size** or **Number of Divisions**, and has a bias towards one of the vertices ("- -- --- ----" or "---- --- -- -").

To apply **Reverse Bias**, select a group of edges, then choose **Reverse Bias** and click **Apply**. Only edges that are part of the main scoping of the control are applied. All others are ignored.

To undo **Reverse Bias**, select an edge that is not part of the main scoping. Then choose **Reverse Bias** and click **Apply**.

Contact Sizing Control

Contact Sizing creates elements of relatively the same size on bodies from the faces of a face to face or face to edge contact region. This control generates spheres of influence (p. 256) internally with automatic determination of radius and size (if **Resolution** is selected for **Type**). You may want to apply a method control (p. 196) on sweepable bodies to force the elements to be tetrahedron in the case where the sweeper is not providing enough local sizing near your contact region. Your swept mesh may be quite dense if the contact size is small on the source and target faces of the body. You may also see very little effect on swept bodies in the case where a contact size is applied to a very small region of a large source face. You can apply contact sizing using any of the following procedures:

- Choose **Contact Sizing** from the **Mesh Control** drop-down menu, or from the context menu when you right-click a **Mesh** object (**Insert> Contact Sizing**). Select a specific contact region under **Scope** in the Details View, then under **Type**, choose **Resolution** for a relative size (and enter a value or use the slider), or **Element Size** (and enter a value) for an absolute size.

- Drag a **Contact Region** object onto the **Mesh** object, then in the Details View, under **Type**, choose **Resolution** for a relative size (and enter a value or use the slider), or **Element Size** (and enter a value) for an absolute size.

- Drag the **Contacts** folder onto the **Mesh** object, which creates a **Contact Sizing** control for each of the contact regions in the folder. Then in the Details View for each contact region, under **Type**, choose **Resolution** for a relative size (and enter a value or use the slider), or **Element Size** (and enter a value) for an absolute size.

- Select the **Contacts** folder or an individual **Contact Region** in the Tree and use the RMB option **Create** > **Contact Sizing** to create **Contact Sizing** controls for the selected contact regions. Then in the Details View for each contact region, under **Type**, choose **Resolution** for a relative size (and enter a value or use the slider), or **Element Size** (and enter a value) for an absolute size.

Note:

- You can select two bodies in the Geometry window and use the **Go To** > Contact Sizing Common to Selected Bodies option to identify any contact sizing controls that exist between the two bodies. This feature provides an easy way for you to delete the common controls.

- Because **Contact Sizing** objects cannot be duplicated, they cannot be used as template objects for the Object Generator.

- Contact sizing works differently for assembly meshing algorithms. See Applying Contact Sizing (p. 398) for details.

Refinement Control

Refinement controls specify the maximum number of times you want an initial mesh to be refined. You can specify refinement controls for faces, edges, and vertices.

To insert a refinement control:

1. Click **Mesh** on the Tree Outline.

2. Do one of the following:

   - Right-click and select **Insert> Refinement** from the context menu.

   - Click **Mesh Control** on the Context Toolbar and select **Refinement** from the drop-down list.

3. In the Details View, scope the geometry whose mesh you want to be refined.

4. Specify a **Refinement** value between 1 (minimum refinement) and 3 (maximum refinement). If you attach several controls to the same entity, the last control applied takes precedence.

Some refinement controls can override or affect other refinement controls that have been applied to connected topology. A face refinement control overrides a refinement control on any of the face's edges or vertices. An edge refinement control overrides a refinement control on either of the edge's vertices. Basically, a refinement control will lower the value of an overridden control by its own value.

For example, consider a face refinement control with a refinement value of 1, where one of the face's edges has a refinement control with a value of 2, and one of the edge's vertices has a refinement control with a value of 2. In this example, the face refinement control reduces the value of the edge refinement control by 1, and it also reduces the value of the vertex refinement control by 1. The edge refinement control now has a value of 1, so it reduces the vertex's refinement control by 1. Now the vertex refinement control has a value of zero, which essentially means the refinement control has no effect.

Note:

- Refinement controls are not available for the **MultiZone**, **Patch Independent Tetra**, or **MultiZone Quad/Tri** mesh methods, or for assembly (p. 367) meshing algorithms. If you are using the **Automatic Method** (p. 199) and you have enabled the Use MultiZone for Sweepable Bodies (p. 317) option, refinement controls on sweepable bodies behave similarly to how they behave when the **Sweep** mesh method is used.
- In the following scenarios, refinement controls are automatically suppressed:
  - When automatic inflation (p. 147) (either Program Controlled (p. 148) or All Faces in Chosen Named Selection (p. 149)) is used with refinement in the same model.
  - When local inflation (p. 291) is used with refinement in the same body or in the same part.
- If you apply a refinement control to a part that was either swept meshed (p. 223) or hex dominant meshed (p. 222), and then you delete the refinement control, the intermediate tetrahedral mesh will be retained unless you invalidate the state of the part (for example, by clearing the database). An intermediate tetrahedral mesh is created when you try to refine non-tetrahedral solid elements.
- Refinement controls are not supported on shared faces between solid bodies and sheet bodies in a multibody part.
- Refinement controls are not supported for Mixed Order Meshing (p. 422).
- Special processing of refinement operations occurs when you use the **Mesh** worksheet to create a selective mesh history. Refer to Using the Mesh Worksheet to Create a Selective Meshing History (p. 409) for details.

Face Meshing Control

Face meshing controls enable you to generate a free or mapped mesh on selected faces. The Meshing application determines a suitable number of divisions for the edges on the boundary face automatically. If you specify the number of divisions on the edge with a **Sizing** control, the Meshing application attempts to enforce those divisions.

To set the **Face Meshing** controls, highlight **Mesh** in the Tree Outline, and right-click to view the menu. Select **Insert> Face Meshing**. You can also click **Mesh** in the Tree Outline, and select the **Mesh Control** Context Toolbar, then select **Face Meshing** from the drop-down menu.

Definition>Mapping is set to **Yes** by default, also exposing **Constrain Boundary** (p. 266) and **Advanced** (p. 267) settings. If Mapping is set to **No**, the Mesher will perform a free mesh and the **Constrain Boundary** and **Advanced** settings are not available.

Note:

To assist you in defining face meshing controls, you can use the **Show Mappable Faces** (p. 498) feature to select all mappable faces automatically and highlight them in the **Geometry** window.

Mapped Face Meshing is supported for the following mesh methods:

Volume Meshing:

- Sweep (p. 223)
- Patch Conforming Tetrahedron (p. 200)
- Hex Dominant (p. 222)
- MultiZone (p. 228)

Surface Meshing:

- Quadrilateral Dominant (p. 245)
- Triangles (p. 246)
- MultiZone Quad/Tri (p. 246)

Face Meshing control topics include:

Note:

For general information about applying mapped Face Meshing controls in combination with the various mesh method controls, refer to Meshing: Mesh Control Interaction Tables (p. 435).

Setting Basic Face Meshing Controls for Mapped Meshing

This section describes the steps for setting basic **Face Meshing** controls for mapped face meshing.

To set basic Face Meshing controls for mapped meshing:

1. Insert a mapped face meshing control by highlighting **Mesh** in the Tree and right-clicking to view the menu. Select **Insert> Face Meshing**. **Definition>Mapped Mesh** is set to **Yes** by default.

2. For the **Definition> Method** control, choose **Quadrilaterals** or **Triangles: Best Split**. (The **Triangles: Best Split** option is available only for sheet models.)

3. For the **Definition> Internal Number of Divisions** control, specify the number of divisions across annular regions or seamless cylinders. (The **Internal Number of Divisions** option is activated when the **Face Meshing** control **Definition** is set to **Mapped Mesh** and the control is scoped to faces made up of two loops.) The default value is 0.

4. For the **Definition> Constrain Boundary** control, specify whether you want to allow the mesher to split the boundary of a mapped mesh region to aid in meshing of adjacent faces. You can choose **Yes** (constrain boundary; no splitting is allowed) or **No** (do not constrain boundary; splitting is allowed). The default is **No**. See Notes on Face Meshing Controls for Mapped Meshing (p. 274) for related information.

5. Generate the mesh by right-clicking the **Mesh** object in the Tree and selecting **Generate Mesh**.

Understanding Advanced Mapped Face Meshing Controls

When you apply advanced mapped Face Meshing controls for mapped meshing to a face, the Meshing application divides the face into one or more mappable regions and creates a mapped mesh in each region. Advanced mapped face meshing controls are subject to restrictions related to vertex types (p. 268) and restrictions related to edge mesh intervals (p. 269).

The *advanced* Face Meshing controls for mapped meshing are supported for the following mesh methods only:

Volume Meshing:

- Sweep (p. 223)
- Patch Conforming Tetrahedron (p. 200)
- Hex Dominant (p. 222)
- MultiZone (p. 228)

Surface Meshing:

- Quadrilateral Dominant (p. 245)
- Triangles (p. 246)
- MultiZone Quad/Tri (p. 246)

Advanced mapped Face Meshing topics include:

Effect of Vertex Type on Face Meshes

Setting Advanced Face Meshing Controls for Mapped Meshing

Note:

For general information on applying Face Meshing controls for mapped meshing in combination with the various mesh method controls, refer to Meshing: Mesh Control Interaction Tables (p. 435).

Restrictions Related to Vertex Types

To constitute a submappable face, a face must possess only End, Side, Corner, and Reversal vertices. In addition, the total number of End vertices, N_E, must satisfy the following equation:

$$N_E = 4 + N_C + 2N_R$$

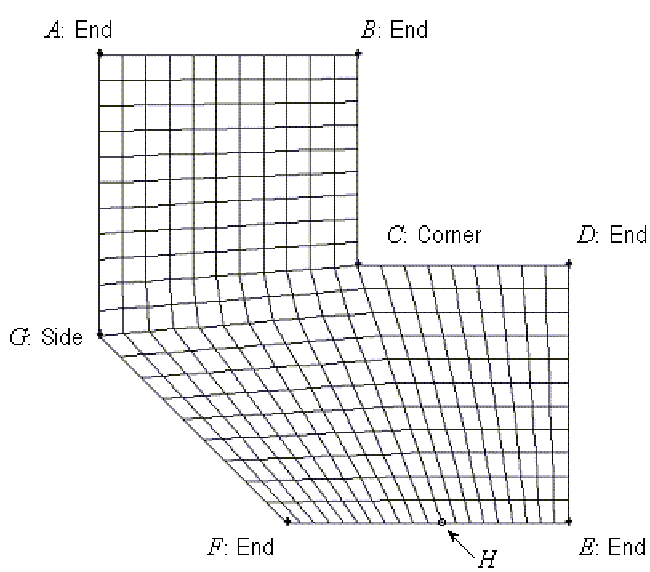
where N_C and N_R are the total numbers of Corner and Reversal type vertices, respectively, on the face. That is, for every Corner type vertex, the face must possess an additional End vertex, and for every Reversal vertex, the face must possess two additional End vertices.

Note:

You cannot specify Reversal vertices. Reversal vertices are used internally by the Meshing application to determine whether the face is mappable.

The shape of the mesh generated by means of the advanced face meshing controls depends on the type and arrangement of vertex types on the face. As an example of the effect of vertex types, consider the face shown in Figure 110: Inside Corner Vertex (p. 268), which consists of a planar L-shaped face, one corner of which is truncated at an angle.

Figure 110: Inside Corner Vertex



In Figure 110: Inside Corner Vertex (p. 268), the inside corner vertex (*C*) is designated as a Corner vertex, therefore, in order to be submappable, the face must possess five End type vertices (*A*, *B*, *D*, *E*, and *F*). The advanced mapped Face Mesh control divides the face into the following two mapped regions:

- *A*, *B*, *C*, *H*, *F*, *G*
- *C*, *D*, *E*, *H*

Note:

If you enforce an advanced mapped face mesh control on a face, the Meshing application evaluates the face with respect to its vertex type designations. If the vertex types do not meet the criteria outlined above, the Meshing application attempts to change the vertex types so that the face is submappable.

For most submappable faces, there are multiple configurations of vertex types that satisfy the vertex type criteria. Each vertex type configuration results in a unique node pattern for the submapped mesh. When the Meshing application automatically changes vertex types, it attempts to employ the configuration that minimizes distortion in the mesh. To enforce a specific node pattern on a submapped mesh, manually select the vertices such that they meet the advanced mapped mesh control vertex type criteria outlined above. (See Selecting the Vertex Type and Picking Vertices (p. 270).)

Restrictions Related to Edge Mesh Intervals

If you specify a bias on the edge of a face before applying an advanced mapped Face Mesh control to the face, you must specify the bias on all parallel edges of the face.

Selecting Faces and Vertices

To use advanced mapped Face Mesh controls on a face, you must do the following:

- Select the face upon which the vertex types are to be defined
- Select the vertex type (using the **Specified Sides**, **Specified Corners**, and **Specified Ends** controls (p. 272))
- Pick the vertices to which the vertex type specification is to be applied

Selecting the Face

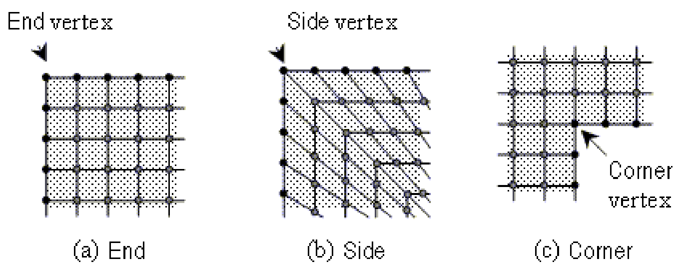
The Meshing application vertex types are specific to the faces upon which they are set. Therefore, to specify the type designation of an individual vertex, you must first select a face to be associated with that vertex. An individual vertex may possess as many vertex type designations as the number of faces to which it is attached. For example, it is possible for a vertex to possess a Side type designation with respect to one face and an End type designation with respect to another, as long as two separate mapped face meshing controls are defined for the two faces. For more information, refer to Setting Advanced Face Meshing Controls for Mapped Meshing (p. 272).

Selecting the Vertex Type and Picking Vertices

The structure of any face mesh in the vicinity of an individual vertex on its boundary is a function of the vertex type. There are three vertex types that you can specify.

- End
- Side
- Corner

Figure 111: Face Vertex Types



An individual vertex may possess only one vertex type designation. For example, you cannot designate a vertex as type "side" and also designate that same vertex as type "end." For more information, refer to Setting Advanced Face Meshing Controls for Mapped Meshing (p. 272).

Each vertex type differs from the others in the following ways:

- The number of face mesh lines that intersect the vertex
- The angle between the edges immediately adjacent to the vertex

The following table summarizes the characteristics of the vertex types shown in Figure 111: Face Vertex Types (p. 270).

Note:

If a face has only 4 vertices and 4 edges, the maximum for the range of the angle of a Side vertex type is 179°, and the acceptable range shifts accordingly.

Vertex Type	Intersecting Grid Lines	Range of Angle Between Edges
End	0	0° — 135°
Side	1	136° — 224°
Corner	2	225° — 314°
Reversal	3	315° — 360° (You cannot specify Reversal vertices. The range for Reversal vertices is used internally by the Meshing application to determine whether the face is mappable.)

The following sections describe the general effect of the End, Side, and Corner vertex types on the shape of the face mesh in the vicinity of a specified vertex.

End Vertex Type

When you specify a vertex as the End vertex type (**Specified Ends** control (p. 272)), the Meshing application creates the face mesh such that only two mesh element edges intersect at the vertex (see (a) in Figure 111: Face Vertex Types (p. 270)). As a result, the mapped and submapped face mesh patterns on both sides of the End vertex terminate at the edges adjacent to the vertex. Assigning the End vertex type to a vertex whose adjacent edges form an angle greater than 180° will likely result in mesh failure.

Side Vertex Type

When you specify a vertex as the Side vertex type (**Specified Sides** control (p. 272)), the Meshing application creates the face mesh such that three mesh element edges intersect at the vertex (see (b) in Figure 111: Face Vertex Types (p. 270)). The Meshing application treats the two topological edges that are adjacent to the vertex as a single edge for the purposes of meshing.

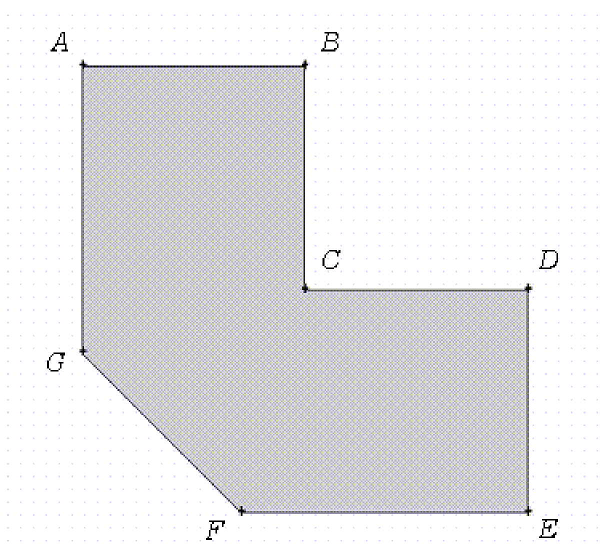
Corner Vertex Type

When you specify a vertex as the Corner vertex type (**Specified Corners** control (p. 272)), the Meshing application creates the face mesh such that four mesh element edges intersect at the vertex (see (c) in Figure 111: Face Vertex Types (p. 270)). Assigning the Corner vertex type to a vertex whose adjacent edges form an angle less than 180° will create an unnecessarily bad quality mesh (although the mesh will be valid).

Effect of Vertex Type on Face Meshes

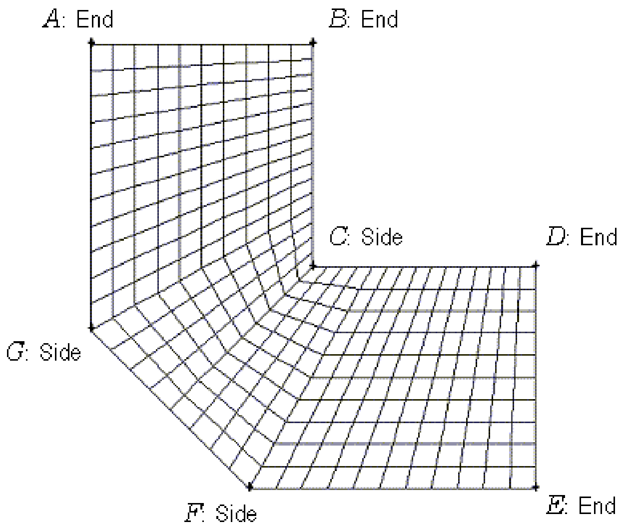
As an example of the general effects of vertex types on face meshes, consider the planar face shown in Figure 112: Seven-sided Planar Face (p. 271). The following two examples illustrate the effects of different vertex type specifications applied to vertices *C*, *F*, and *G* on the shape of the resulting mesh.

Figure 112: Seven-sided Planar Face



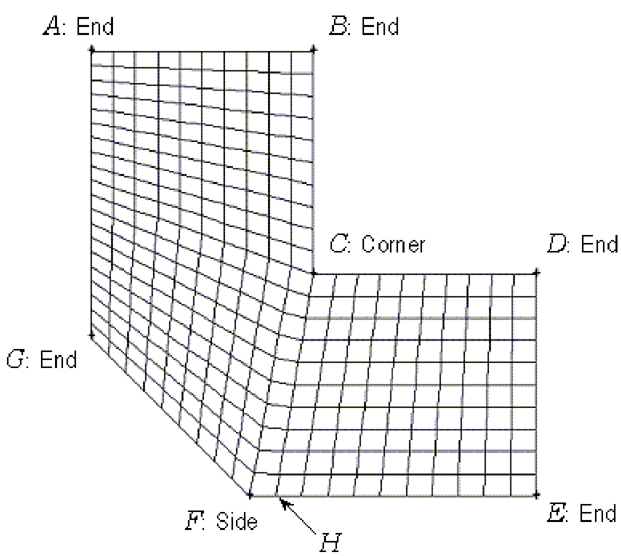
In Figure 113: Example Face Mesh—Side Inside Corner Vertex (p. 272), vertices *C*, *F*, and *G* are specified as Side vertices; therefore, the Meshing application treats sides *BCD* and *EFGA* as if each were a single edge. As a result, the entire face represents a mappable region, and the Meshing application creates a single checkerboard pattern for the mesh.

Figure 113: Example Face Mesh—Side Inside Corner Vertex



In Figure 114: Example Face Mesh—Corner Inside Corner Vertex (p. 272), vertices *C*, *F*, and *G* are specified as Corner, Side, and End type vertices, respectively. As a result, the face is submappable, and the Meshing application creates two separate checkerboard patterns for the mesh. The upper-left submapped region is defined by the polygon *ABCHFG*. The lower-right submapped region is defined by *CDEH*. For both regions, the node at point *H* serves as an End type vertex for the purposes of mesh creation.

Figure 114: Example Face Mesh—Corner Inside Corner Vertex



Setting Advanced Face Meshing Controls for Mapped Meshing

This section describes the basic steps for setting Face Meshing controls for mapped meshing.

To set advanced Face Meshing controls for mapped meshing:

1. Insert a **Face Meshing** control by highlighting **Mesh** in the Tree and right-clicking to view the menu. Select **Insert> Face Meshing**. **Definition>Mapped Mesh** is set to **Yes** by default.

2. Select the face upon which the vertex types are to be defined by scoping the face in the **Mapped Face Meshing** Details View. (Refer to Selecting the Face (p. 269) for more information.)

3. For the **Definition> Method** control, choose **Quadrilaterals** or **Triangles: Best Split.**

4. Enter additional **Definition** settings, as desired, in the Details View.

5. Use the **Specified Sides**, **Specified Corners**, and **Specified Ends** controls in the **Advanced** section of the Details View to select the desired vertices in the **Geometry** window and apply your selections. To do so, pick the desired vertex/vertices in the **Geometry** window and then click the **Specified Sides**, **Specified Corners**, or **Specified Ends** control to assign the vertex/vertices to the desired vertex type. (Refer to Selecting the Vertex Type and Picking Vertices (p. 270) for more information.)

Note:

If you select a vertex by mistake and want to de-select it, click the control in question in the **Advanced** section of the Details View, clear the selection by clicking in an "empty" portion of the **Geometry** window, and then click **Apply**. For example, assume that you mistakenly selected 1 vertex for the **Specified Corners** control. To clear the selection:

- In the **Specified Corners** control in the **Advanced** section of the Details View, click your selection (that is, the text "1 Vertex").

  The **Apply**/**Cancel** buttons will appear in the **Specified Corners** control and the vertex will be highlighted in green in the **Geometry** window.

- Click in an empty portion of the **Geometry** window.

- Click **Apply** in the **Specified Corners** control.

Note:

An individual vertex may possess as many vertex type designations as the number of faces to which it is attached. For example, it is possible for a vertex to possess a Side type designation with respect to one face and an End type designation with respect to another, as long as two separate mapped face meshing controls are defined for the two faces. Conversely, a single mapped Face Meshing control cannot specify the same vertex as more than one vertex type. That is, you cannot designate a vertex as type Side and also designate that same vertex as type End in a single mapped face meshing control. If you attempt to do so, the second and any subsequent assignments for that vertex will result in the control being highlighted in yellow in the **Advanced** section of the Details View, and you will not be able to generate a mesh. If this occurs, use the procedure noted above to de-select the unwanted vertex assignment(s).

6. Generate the mesh by right-clicking the **Mesh** object in the Tree and selecting **Generate Mesh**.

Notes on Face Meshing Controls for Mapped Meshing

Remember the following notes when using the Face Meshing controls for mapped meshing:

- The blue status icon that may appear in the Tree Outline indicates that a mapped mesh cannot be provided on the scoped topology. One of three scenarios triggers the icon:

  1. The face cannot be map meshed.

  2. The quality of the mapped mesh was not acceptable and a free mesh was generated.

  3. If **Constrain Boundary** (p. 266) is set to **Yes**, the mesher will fail if the boundary of a mapped mesh must be modified.

- For mixed/solid shell parts, a Face Meshing control cannot be scoped to a sheet face if the face is adjacent to a solid body. In such cases, the meshing of the sheet face will fail.

- To assist you in working with mapped face meshing, you may want to use the **Show Mappable Faces** (p. 498), **Show Sweepable Bodies** (p. 494), and/or **Preview Surface Mesh** (p. 489) features. The **Show Mappable Faces** feature selects all mappable faces automatically and highlights them in the **Geometry** window. By using the **Show Sweepable Bodies** feature, you can find out whether bodies are sweepable (before and after modifying vertex types). By using the **Preview Surface Mesh** feature, you can verify that your mesh settings are correct.

- When sweeping (p. 323):

  - If the sweep method is applied to a body and mapped face meshing is defined for either the body's source or target face, the sweep mesher will fail if a mapped mesh cannot be obtained for the face.

  - When mapped face meshing is defined for a side face, the mapped mesher will loosen its tolerances on determining whether a face is mappable.

  - When sweeping and using *advanced* mapped meshing controls (p. 269), you must set vertex types for both the source and target faces.

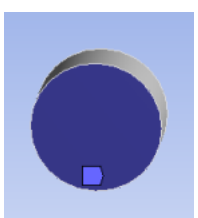
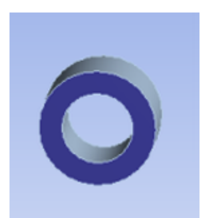
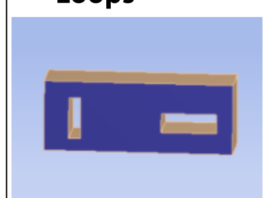
- When a face has only 4 vertices and 4 edges, and mapped Face Meshing controls are applied, the only factor that will determine a successful mapped mesh is element quality.

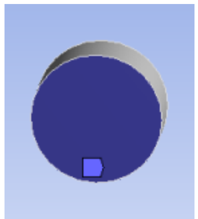
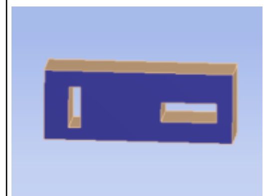
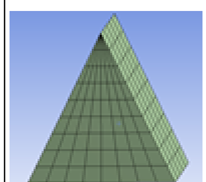
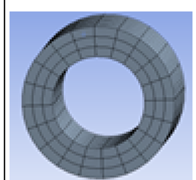
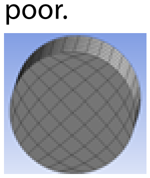
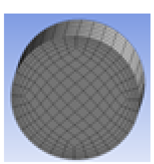
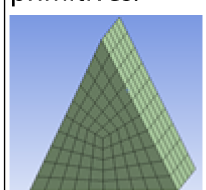
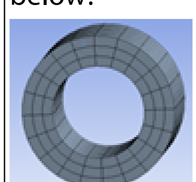
Note:

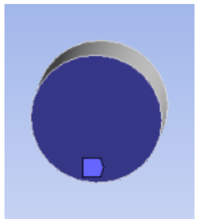
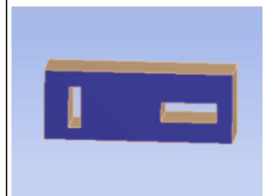
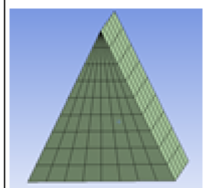
It is often helpful to use the **Show Vertices** option to ensure edges are complete and do not have unintended segmentation. If an edge is segmented, it could mean that a face you think should be successfully mapped actually has 5 vertices and 5 edges. To help resolve such issues, you can define a virtual edge (p. 502) or use advanced mapped Face Meshing controls (p. 267).

- An effective technique for mapped meshing on surface bodies is to select all faces on the body and let the mesher determine which faces should be map meshed and which faces should be free meshed.
- Mapped Face Meshing controls are not supported for assembly (p. 367) meshing algorithms.
- If the mapped Face Meshing controls are attached to faces with exactly two boundary edges or two sets of boundary edges, an additional option, **Internal Number of Divisions**, is available. This option allows you to specify the number of layers of elements that will be generated between the two boundary edges.
  - If there is a conflict between **Internal Number of Divisions** and a **Face** sizing control, the mapped face control's Internal Number of Divisions value will take priority.
  - If there is a conflict between **Internal Number of Divisions** and a parallel **Edge** sizing control, the Sweep mesh method will respect the Internal Number of Divisions unless the size is hard, in which case it will return an error notifying you of the conflict. The **MultiZone** mesh method will respect the **Edge** sizing control.
  - If there is a conflict between the **Internal Number of Divisions** and the number of divisions on a Sweep mesh method, the Sweep mesh method will return an error notifying you of the conflict.
  - See Face Characteristics for Annular faces in the table below for more information.

The table below provides an overview of types of faces and how various mesh methods handle when mapped Face Meshing controls are applied to them.

Mesh Method	**Face Characteristic**			
	Circular 	**Triangle** 	**Annular** 	**Internal Loops** 
• Sweep (p. 223) • Thin Sweep (p. 330) • Hex Dominant (p. 222) • Patch Conforming Tetrahedron (p. 200)	Not supported. You must free mesh these faces.	Not supported. You must free mesh these faces. As side faces, triangle faces can be	Supported. In the Details View, the **Internal Number of Divisions** (p. 266) option is activated so	Not supported. You must free mesh these faces, but note the following: If there is just one internal loop, it is

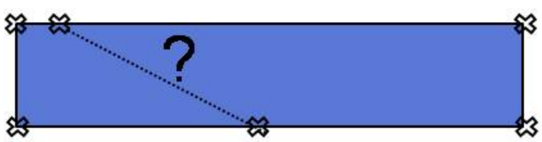
Mesh Method	Face Characteristic			
	Circular 	Triangle 	Annular 	Internal Loops 
• Quadrilateral Dominant (p. 245) • Triangles (p. 246)		mapped to obtain a wedge mesh at one corner, depending on source face selection. 	you can specify the number of divisions across the annular region. The option is set to **3** in the example below: 	treated as an annular case. For example, if the model above were split in half, you would have a square annulus which would mesh similar to the circular annulus model.
• MultiZone (p. 228) • MultiZone Quad/Tri (p. 246)	Supported, but mesh quality is poor.  You can use inflation to obtain an O-Grid for better quality in corners. 	Supported. Triangle faces are submapped to tri primitives.  As side faces, triangle faces can be mapped to obtain a wedge mesh at one corner, depending on source face selection.	Supported. In the Details View, the **Internal Number of Divisions** (p. 266) option is activated so you can specify the number of divisions across the annular region. The option is set to **3** in the example below: 	Ignored for source faces. However, internal loops are supported for side faces. If this example is meshed top to bottom, and has mapped faces defined on the sides, it meshes as follows:  For more information on using mapped Face Meshing with

Mesh Method	Face Characteristic			
	Circular 	**Triangle** 	**Annular** 	**Internal Loops** 
				side faces, see Side Face Handling of Imprinted Regions (p. 356).

With fillets, chamfers, or large or small angles

- It is not always clear which vertex(es) should be used as the corner(s). Using virtual topology (p. 501) to merge edges or split faces, or using mapped face meshing vertex controls may help.

- When there are a large number of segments along a side. This situation may create difficulties in assigning incremental edge assignments that lead to good quality mesh. In these cases, adding a face split is a good way to ensure the interval edge assignment is done correctly.



Concerns over the mappability of faces may be different depending on whether the faces are source/target faces or side faces. For sweeping to be successful (regardless of whether Sweep (p. 223), Thin Sweep (p. 330), or MultiZone (p. 228) is used), all side faces (that is, all non-source/target faces) must be mappable. Ensuring side face mappability is most critical to ensure a successful swept mesh. Ensuring a source face can be mapped should be a lower priority, and your strategy for sweeping should account for this difference. You can use the **Show>Mappable Faces** (p. 498) option to help determine mappability of faces.

In addition to the mappability of each individual side face, the collective set of faces for a given side may present a problem. For example, for a collective set of faces, all parallel edges need to have the same number of divisions from the top to the bottom of the sweep. Ensuring all side faces of a swept body are mappable does not always ensure the body is sweepable. For example, if parallel edges of a mapped face change in direction, a source edge could become a side edge and make the body impossible to sweep. Also, edge splits on one face need to propagate through the collective set of faces while maintaining a reasonable quality mesh. Reducing the number of edge splits may simplify the sweeping and lead to a better quality mesh. This effect is the result of the simplification in the "interval edge assignment," or the requirement for the mesher to have the same number of divisions along the sweep path.

Extending splits through the set of side faces may also help the mesher with the incremental edge assignments, as well as with constraining the grid lines along the sweep path to control the mesh quality. The imprinting that occurs with **MultiZone** creates further complications. See Side Face Handling of Imprinted Regions (p. 356) for more information.

Mesh Copy Control

The **Mesh Copy** control enables you copy mesh from one body to another. This option can be used to reduce the mesh setup time for repetitive bodies/parts. Association to CAD is maintained after performing mesh copy.

Mesh controls are scoped only to the source anchor body. When the mesh is generated, the source anchor body is meshed and the mesh is then copied to targets.

You can scope the Mesh Copy control to either a geometry selection or a named selection as follows:

1. Insert a Mesh Copy control by right-clicking the Mesh object in the Tree Outline and selecting **Insert** > **Mesh Copy**.

2. In the Details view, define the scope of the selection:

   - To apply the mesh copy to a geometry selection, set the **Scoping Method** to **Geometry Selection**. Select a face for the **Source Anchor**.

   - To apply the mesh copy to a named selection, set the **Scoping Method** to **Named Selection**. Select the appropriate Named Selection for the **Source Anchor**.

3. Define the scope for the target:

   - To apply the mesh copy to a geometry selection, set the **Target Scoping** to **Geometry Selection**. Select the appropriate faces for the **Target Anchors**.

   - To apply the mesh copy to a named selection, set the **Target Scoping** to **Named Selection**. Select the appropriate Named Selection for the **Target Anchors**.

Note:

While scoping the source and target anchors, note the following:

- The source and target face area should be the same.

- The associated source and target bodies should have the same volume.

- The source and target configuration should be identical (for example, a circle to rectangle mapping is incorrect, even if the face area is the same).

- If the above is not true, the mesh may be copied, but the nodes may not be associated properly to the target bodies, or the copied mesh transformation may be incorrect.

Figure 115: Mesh Copy Scope (p. 279) shows the set up of the Source face (blue) and Target Anchor faces (red) for the Mesh Copy control. Figure 116: Generated Mesh (p. 280) shows the mesh that was generated.

Figure 115: Mesh Copy Scope

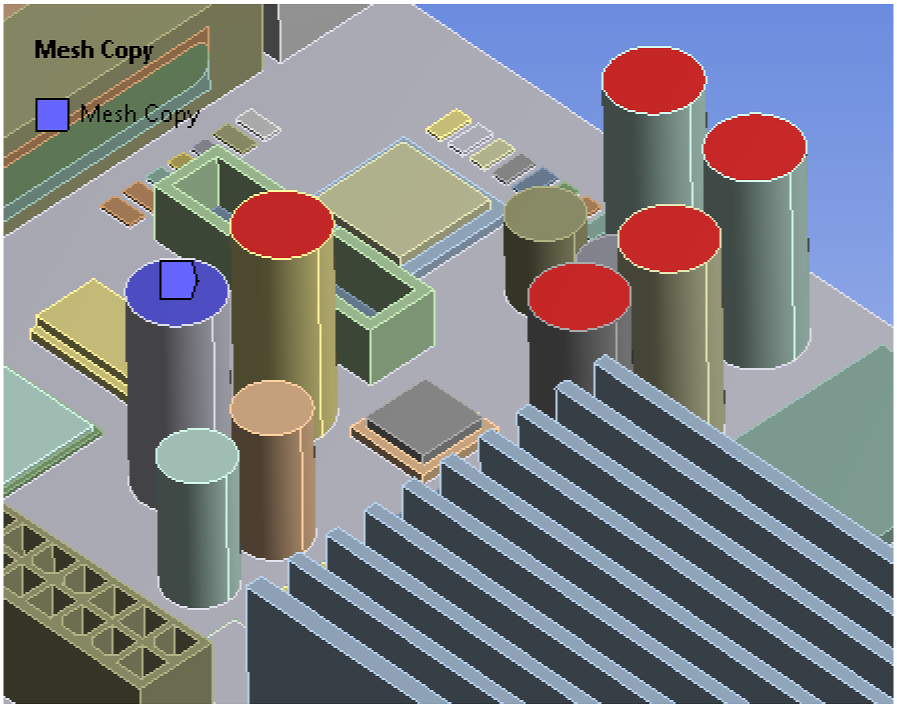
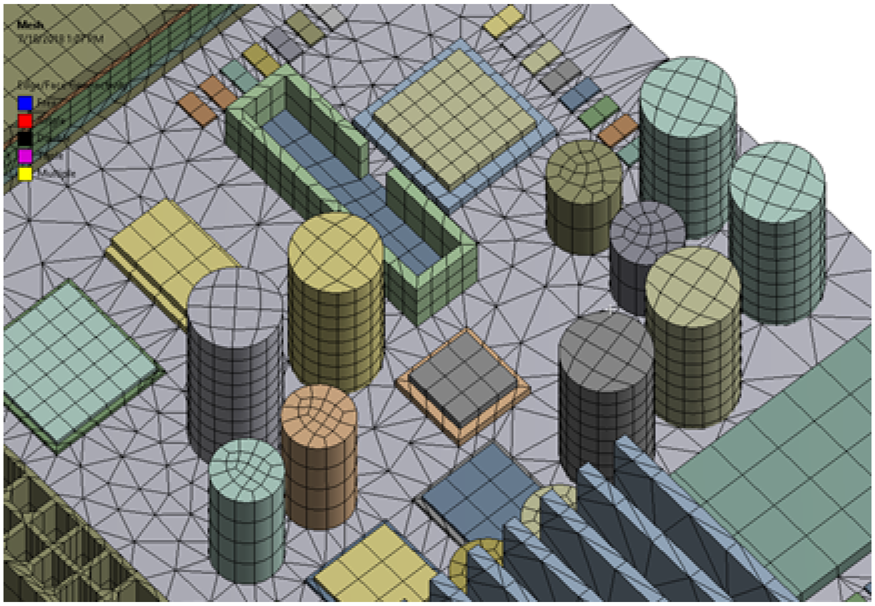


Figure 116: Generated Mesh



Note:

- The mesh copy control is supported only for solid bodies.
- The mesh copy control is not supported when previewing the surface mesh, or previewing inflation.
- Mesh copy is supported for multibody parts with share topology. Bodies which are scoped to a **Mesh Copy** control are meshed first, and then the remaining bodies are meshed.
- Selective meshing (p. 404) is supported with mesh copy only if the source bodies are meshed before the target bodies. If the target bodies are meshed first, the mesh will not be copied.
- The Mesh Copy target body is a reflection of the source. Element connectivities will be different. This may cause the solver to fail. Choosing a different target anchor face might help in some cases. True reflection is not supported.
- Mesh copy is not supported for Assembly meshing (p. 367).
- Mesh controls set up on the target bodies will be ignored when the mesh is generated.
- Sizing controls set up on the target bodies will not be copied to the source body.

Match Control

The **Match Control** matches the mesh on two or more faces or edges in a model. The Meshing application provides two types of match controls—cyclic (p. 283) and arbitrary (p. 284).

The **Match Control** is supported for the following mesh methods:

Volume Meshing:

- Sweep (p. 223)
- Patch Conforming (p. 200)
- MultiZone (p. 228)

Surface Meshing:

- Quad Dominant (p. 245)
- All Triangles (p. 246)

Remember the following information when using the match control feature:

- Edge meshes are matched for sheet, 2D, and 3D bodies. Face meshes are matched across bodies.
- A single match control with one high face and one low face cannot be applied across multiple parts. If there are multiple faces on the high side and multiple faces on the low side, the software does its best to match the high and low sides on a part-by-part basis. For example, the match control will support situations in which there are two parts, each having one face on the high side and one face on the low side (for a total of two high faces and two low faces). However, for more complex situations, you must be careful to ensure the proper matching is done.
- Matching will fail if the high and low faces are on two separate bodies that have other bodies (being meshed with a method other than Sweep) or a space between them.
- The faces or edges that you select must be topologically and geometrically the same.

  This means:

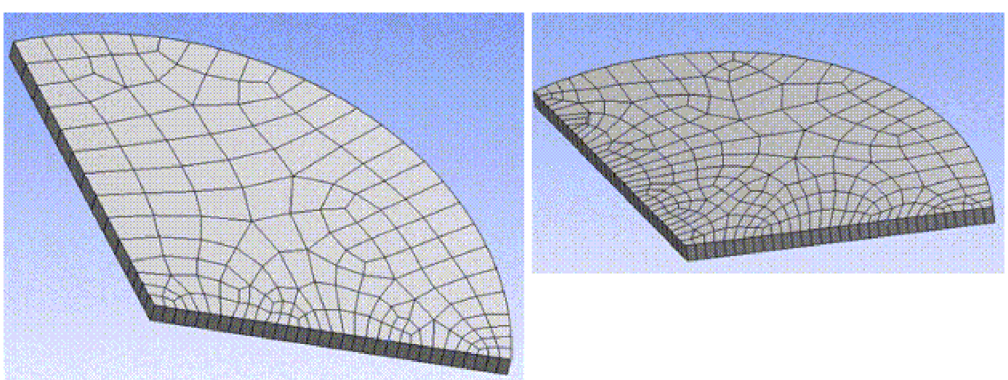
  - They have the same number of vertices on the high and low sides.
  - They have similar surface area or length.
  - The high and low sides have similar transformation.

    The mesher will move nodes away from the geometry to meet the transformation.
- If the low and high sides of a geometry do not match, the high side geometry is meshed and the low side will be meshed using the transformation from the match control. In this case, the low side geometry and mesh might be slightly different, so a warning is displayed prompting you to check this deviation and decide whether it is acceptable or that you should modify the geometry.
- Multiple match controls can be associated with a single entity, but multiple associations can result in conflicts among match controls. If a conflict occurs, the Meshing application issues an error message, and matching fails. For example, a match control conflict may occur if the two faces adjacent to an edge have two different match controls applied to them. If the two match controls use two different sets of coordinate systems the mesher may produce an error even if the transformation between the two match controls is the same. However, for this case, if the two match controls refer to the same set of coordinate systems there won't be any conflict.
- A match control can only be assigned to one unique face pair. Assigning the same face as **High/Low Geometry** in more than one match control is not supported. If multiple match controls assign the

same face as a **High/Low Geometry** entity, the match control that appears lowest in the Tree is honored.

- Match controls are not respected with refinement or adaptivity.
- When match is used with the Sizing Options (p. 100), the effect of a sizing on the high or low side will be transferred bidirectionally from the high side to the low side and vice versa. This means that if the low side has a sizing control and the high side does not, the Sizing control will use the low sizing control on the high side.
- Match controls on faces are supported with Pre inflation (p. 156), regardless of whether inflation is set to Program Controlled (p. 148) or has been set through any global or local inflation definition. In contrast, match controls on edges are not supported with Pre inflation. Match controls (both faces and edges) are not supported with Post inflation (p. 158). For non-supported cases, Ansys Workbench automatically suppresses/disables the Match Control feature.
- Match controls are not enforced when previewing inflation (p. 492).
- You cannot apply a match control to topology on which a face-edge pinch (p. 182), mesh connection, or symmetry control has been applied. In any of these cases, an error will be issued when you generate the mesh.
- Match controls are not supported for assembly (p. 367) meshing algorithms.
- Match controls on edges are not supported for the MultiZone (p. 228) mesh method. When match controls on faces are used with MultiZone, only one periodic or cyclic transformation is supported (MultiZone can support multiple match controls, as long as they use the same coordinate system and have the same angle/translation). In addition, MultiZone does not support matching of free meshed regions (p. 229).
- Match controls can be used with thin sweeping (p. 330), as shown in the figures below. In the figure on the left, a match control was applied to the top and bottom faces. In the figure on the right, a match control was applied to the side faces.

Figure 117: Match Controls Used with Thin Sweeping



Match control topics include:

Arbitrary Match Control

Note:

For general information on applying match controls in combination with the various mesh method controls, refer to Meshing: Mesh Control Interaction Tables (p. 435).

Cyclic Match Control

The cyclic matching process involves copying the mesh of the first selected faces or edges in the **Match Control** (the **High Geometry Selection** scoped in the Details View of the **Match Control**) to the second selected faces or edges in the control (the scoped **Low Geometry Selection**).

You can scope a Cyclic Match Control to either a geometry selection or a named selection.

If you want to automatically generate solver constraints for periodic mesh in addition to matching the mesh, you should use the **Symmetry** feature in the Mechanical application.

1. Insert a Match Control by right-clicking the Mesh object in the Tree Outline and selecting **Insert** > **Match Control**.

2. In the Details view, define the scope of the selection:

To apply a match control to...	Do this...
A geometry selection	1. Click **Scoping Method** and select **Geometry Selection**. 2. Select one or more faces or edges, and then in the **High Geometry Selection** field, click **Apply**. 3. Select one or more faces or edges, and then in the **Low Geometry Selection** field, click **Apply**.
A named selection	1. Click **Scoping Method** and select **Named Selection**. 2. In the **High Boundary** and **Low Boundary** fields, select the appropriate named selections.

3. In the Details View, select **Cyclic** as the **Transformation** type and a coordinate system with its z-axis aligned to the axis of rotation for the geometry.

4. Generate the mesh by right-clicking the **Mesh** object and selecting **Generate Mesh** from the context menu.

Note:

When a match is successful, the number of elements in the matched faces/edges will be the same, and there will be a direct one-to-one mapping between their nodes. A small blue status icon () appears to the left of the **Match Control** object icon in the Tree Outline if the **Match Control** fails on the face or edge pair.

When a cyclic **Match Control** is used together with **Sizing** (p. 248) controls, the controls on the high side have the higher precedence. Whatever controls are on the high faces or edges will be honored on the low faces or edges in the **Match Control**. **Sizing** controls applied to the low faces or edges will be honored only if the high side does not have the same controls, and only if the sizing is applied *directly* on the low topology (that is, applying the sizing on connected topologies will have no effect).

Note:

The Meshing application inserts match controls for periodic regions automatically. See the Match Meshing and Symmetry (p. 423) section for more information.

Arbitrary Match Control

This feature lets you select multiple pairs of faces or edges in a model to create a match control that will consequently generate exactly the same mesh on the high geometry as it does on the low geometry. You can select a set of high faces belonging to different parts, and low faces belonging to different parts, as long as the high and low pairs are on the same body. So, for example, you can have a single match control consisting of multiple pairs of faces or edges across bodies.

However, unlike cyclic match controls (p. 283), which require you to select a coordinate system with its z-axis of rotation aligned to the geometry's axis of rotation, for arbitrary match controls the faces or edges to be matched can be arbitrarily located, and the match control is based on two coordinate systems that you select.

You can scope an Arbitrary Match Control to either a geometry selection or a named selection.

If you want to automatically generate solver constraints for periodic mesh in addition to matching the mesh, you should use the **Symmetry** feature in the Mechanical application.

1. Insert a Match Control by right-clicking the Mesh object in the Tree Outline and selecting **Insert** > **Match Control**.

2. In the Details view, define the scope of the selection:

To apply a match control to...	Do this...
A geometry selection	1. Click **Scoping Method** and select **Geometry Selection**.

To apply a match control to...	Do this...
	2. Select one or more faces or edges, and then in the **High Geometry Selection** field, click **Apply**. 3. Select one or more faces or edges, and then in the **Low Geometry Selection** field, click **Apply**.
A named selection	1. Click **Scoping Method** and select **Named Selection**. 2. In the **High Boundary** and **Low Boundary** fields, select the appropriate named selections.

3. Change the value of the **Transformation** control to **Arbitrary**.

4. Choose the coordinate systems for the selected high and low geometry entities. The applicable settings in the Details View are:

   - **High Coordinate System**: Choose the coordinate system for the faces/edges assigned by the **High Geometry Selection** control.
   - **Low Coordinate System**: Choose the coordinate system for the faces/edges assigned by the **Low Geometry Selection** control.

   Note:

   All the coordinate systems currently defined for the model appear in the **High Coordinate System** and **Low Coordinate System** drop-down menus. You may choose coordinate systems from the list, or you may need to define additional coordinate systems. In order for the match control to be honored, the coordinate systems that you choose must be defined such that a valid transformation matrix can be calculated. In other words, the two coordinate systems must be created such that when the coordinates of every point of the **Low Geometry Selection** in the **Low Coordinate System** are placed into the **High Coordinate System**, the high and low faces/edges match exactly. Refer to the Coordinate Systems Overview in the Mechanical help for information on coordinate systems and how to create them.

5. Generate the mesh by right-clicking on the **Mesh** object and selecting **Generate Mesh** from the context menu.

   When a match is successful, the number of elements in the matched faces/edges will be the same, and there will be a direct one-to-one mapping between their nodes. If the mesh is generated but the match was unsuccessful, a small blue status icon ( ) displays to the left of the **Match Control** object icon in the Tree Outline.

Example 5: Arbitrary Mesh Matching

The figures below show an example of arbitrary mesh matching. Figure 118: Coordinate Systems for Arbitrary Mesh Matching (p. 286) shows the selected coordinate systems, and Figure 119: Matched Mesh (p. 286) shows the resulting matched mesh. Edge and face sizings were also applied.

Figure 118: Coordinate Systems for Arbitrary Mesh Matching

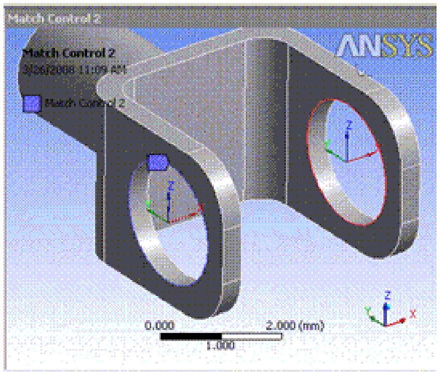
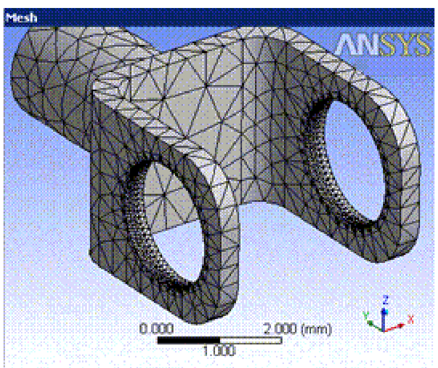


Figure 119: Matched Mesh



Pinch Control

The **Pinch** control lets you remove small features (such as short edges and narrow regions) at the mesh level in order to generate better quality elements around those features.

When **Pinch** controls are defined, the small features in the model that meet the criteria established by the controls will be "pinched out," thereby removing the features from the mesh. You can instruct the Meshing application to automatically create pinch controls based on settings that you specify (as described in Pinch Control Automation Overview (p. 186)), or you can manually designate the entities to be pinched, as described below in Defining Pinch Controls Locally (p. 287).

Local pinch control topics include:

Defining Pinch Controls Locally

Changing Pinch Controls Locally

For an overview of pinch controls and details on pinch control automation, refer to Pinch (p. 182). For general information on applying pinch controls in combination with the various mesh method controls, refer to Meshing: Mesh Control Interaction Tables (p. 435). For general information about pinch control usage, see Usage Information for Pinch Controls (p. 190).

Defining Pinch Controls Locally

This section describes the steps for defining pinch controls locally.

To define pinch controls locally:

1. Click the **Mesh** folder in the Tree Outline. Right-click and select **Insert** > **Pinch** from the context menu.

   Or

   Click **Mesh Control** on the toolbar and choose **Pinch** from the menu.

   A pinch control object is inserted into the Tree.

2. In the **Geometry** window, pick one or more faces, one or more edges, or one vertex that you want to define as the Primary. (A Primary geometry is an entity that retains the profile of the original geometry.) Refer to Pinch (p. 182) for a table that summarizes the valid entities you can pick for each type of pinch control.

3. Use either of these methods to apply your selection:

   - Right-click in the **Geometry** window to display the context menu and select **Set As Pinch Primary**.
   - Click in the **Primary Geometry** field in the Details View.

   The pinch region is flagged in the **Geometry** window. The color of each selected entity changes to blue to identify it as primary geometry. If you want to adjust your selections, you can re-pick geometry and then apply the new selections to overwrite the existing primary geometry. If using the context menu method to apply selections, you can also pick additional geometry and select **Add To Pinch Primary** to add the geometry to the existing primary geometry. Also see Changing Pinch Controls Locally (p. 290).

4. In the **Geometry** window, pick one or more edges or vertices that you want to define as the secondary. (A secondary geometry is an entity that changes in order to move towards the primary geometry. Depending on the tolerance, the pinch control will pinch out the entire secondary entity or only a portion of the secondary entity into the primary. Faces cannot be defined as secondary.)

5. Use either of these methods to apply your selection:

   - Right-click in the **Geometry** window to display the context menu and select **Set As Pinch Secondary**.
   - Click in the **Secondary Geometry** field in the Details View.

   The color of each selected edge/vertex changes to red to identify it as secondary geometry. If you want to adjust your selections, you can re-pick geometry and then apply the new selections to overwrite the existing secondary geometry. If using the context menu method to apply selections, you can also pick additional geometry and select **Add To Pinch Secondary** to add the geometry to the existing secondary geometry. Also see Changing Pinch Controls Locally (p. 290).

6. Change the value of the **Suppressed** control if desired.

   By default, the value of **Suppressed** is **No**. If you change the value to **Yes**, this pinch control has no effect on the mesh (that is, the small features you picked for this pinch control are not removed and will affect the mesh). In addition, an **Active** control with a read-only setting of **No, Suppressed** appears under the **Suppressed** control when **Suppressed** is set to **Yes**.

7. Change the value of the **Tolerance** control if desired.

   By default, the value of **Tolerance** is based on the global pinch control tolerance (p. 188). If you specify a different value here, it overrides the global value.

8. Change the value of the **Snap to Boundary** control if desired.

   By default, the value of **Snap to Boundary** is **Yes**. **Snap to Boundary** is applicable only for pinch controls in shell models for which a face has been defined as the primary geometry and one or more edges are defined as the secondary. In such cases, when the value of **Snap to Boundary** is **Yes** and the distance from a secondary edge to the closest mesh boundary of the primary face is within the specified snap to boundary tolerance, nodes from the secondary edge are projected onto the boundary of the primary face. The joined edge will be on the primary face along with other edges on the primary face that fall within the defined pinch control tolerance. Refer to the figures below to see the effect of the **Snap to Boundary** setting.

   Note:

   For edge-to-edge pinch controls in shell models, the snap tolerance is set equal to the pinch tolerance internally and cannot be modified.

   ---

9. Change the value of the **Snap Type** control if desired. **Snap Type** appears only when the value of **Snap to Boundary** is **Yes**.

   - If **Snap Type** is set to **Manual Tolerance** (the default), a **Snap Tolerance** field appears where you may enter a numerical value greater than 0. By default, the **Snap Tolerance** is set equal to the pinch tolerance but it can be overridden here.

   - If **Snap Type** is set to **Element Size Factor**, a **Primary Element Size Factor** field appears where you may enter a numerical value greater than 0. The value entered should be a factor of the local element size of the primary topology.

Note:

When you define a pinch control locally, the value of **Scope Method** is **Manual** (read-only). If you make changes to a pinch control that was created through pinch control automation (p. 189), the value of the **Scope Method** field for that pinch control changes from **Automatic** to **Manual**.

Figure 120: Snap to Boundary Set to Yes (p. 289) shows the mesh when **Snap to Boundary** is set to **Yes** (default).

Figure 120: Snap to Boundary Set to Yes

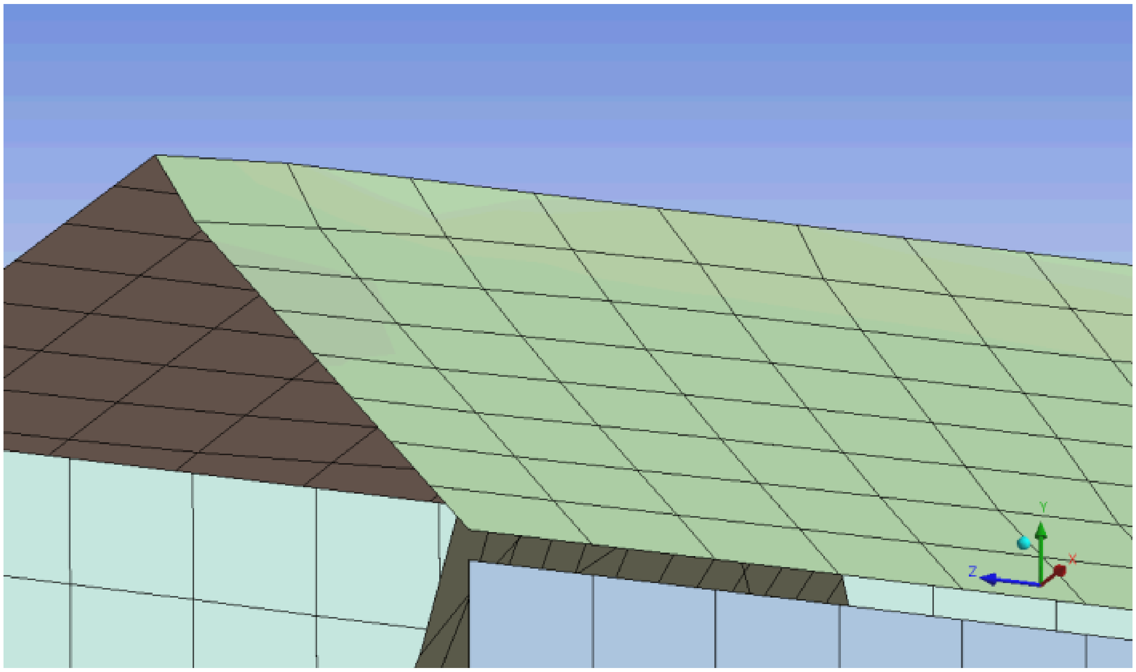
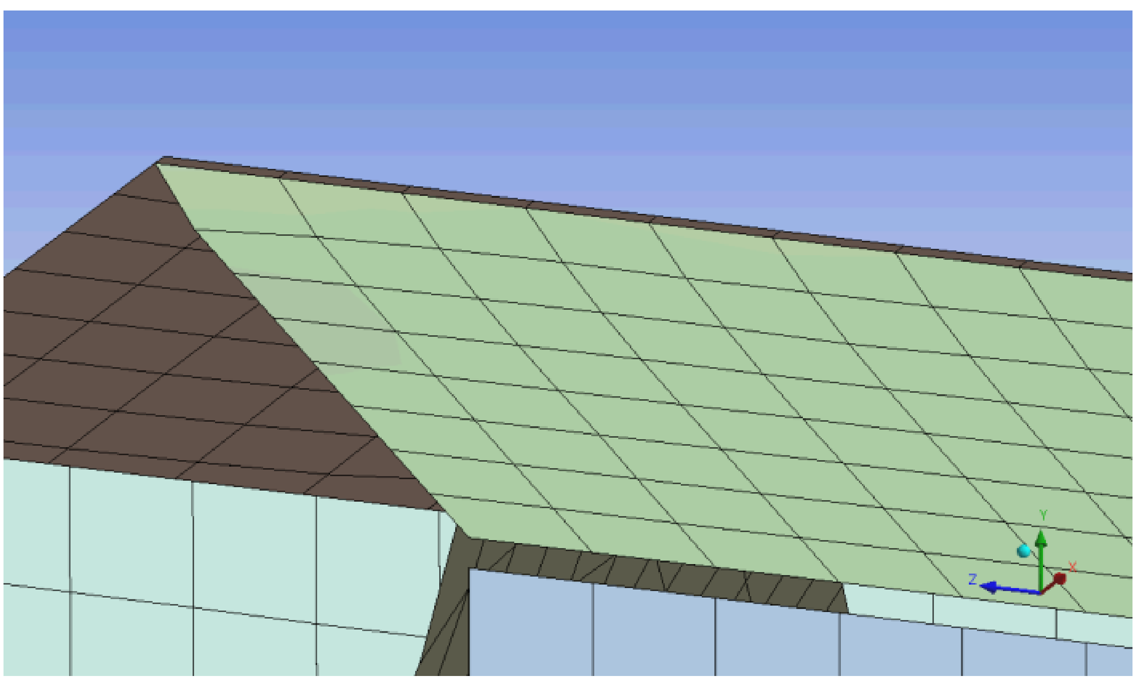


Figure 121: Snap to Boundary Set to No (p. 289) shows the mesh for the same model when **Snap to Boundary** is set to **No**.

Figure 121: Snap to Boundary Set to No



Changing Pinch Controls Locally

This section describes the steps for changing pinch controls locally. You can make changes to pinch controls regardless of whether they were created automatically or manually. You can select multiple pinch controls to make the same changes to all selected pinch controls at one time.

To change pinch controls locally:

1. In the Tree Outline, select the pinch control(s) that you want to change.

2. In the **Geometry** window, pick one or more faces, one or more edges, or one vertex that you want to define as primary. Refer to Pinch (p. 182) for a table that summarizes the valid entities you can pick for each type of pinch control.

3. Use either of these methods to apply your selection:

   - Right-click in the **Geometry** window to display the context menu and select **Set As Pinch Primary** or **Add To Pinch Primary**.

   - Click in the **Primary Geometry** field in the Details View.

   The pinch region is flagged in the **Geometry** window. The color of each selected entity changes to blue to identify it as primary geometry. If you want to further adjust your selections, you can re-pick geometry and then apply the new selections to overwrite or add to the existing primary geometry. To add to the geometry, you must use **Add To Pinch Primary** in the context menu.

4. In the **Geometry** window, pick one or more edges or vertices that you want to define as secondary. (Faces cannot be defined as secondary.)

5. Use either of these methods to apply your selection:

   - Right-click in the **Geometry** window to display the context menu and select **Set As Pinch Secondary** or **Add To Pinch Secondary**.

   - Click in the **Secondary Geometry** field in the Details View.

   The color of each selected edge/vertex changes to red to identify it as secondary geometry. If you want to further adjust your selections, you can re-pick geometry and then apply the new selections to overwrite or add to the existing secondary geometry. To add to the geometry, you must use **Add To Pinch Secondary** in the context menu.

6. Change the value of the **Suppressed** control if desired.

   By default, the value of **Suppressed** is **No**. If you change the value to **Yes**, this pinch control has no effect on the mesh (that is, the small features you picked for this pinch control are not removed and will affect the mesh). In addition, an **Active** control with a read-only setting of **No, Suppressed** appears under the **Suppressed** control when **Suppressed** is set to **Yes**.

7. Change the value of the **Tolerance** control if desired.

   By default, the value of **Tolerance** is based on the global pinch control tolerance (p. 188). If you specify a different value here, it overrides the global value.

8. Change the value of the **Snap to Boundary** control if desired.

By default, the value of **Snap to Boundary** is **Yes**. **Snap to Boundary** is applicable only for pinch controls in shell models for which a face has been defined as the primary geometry and one or more edges are defined as the secondary. In such cases, when the value of **Snap to Boundary** is **Yes** and the distance from a secondary edge to the closest mesh boundary of the primary face is within the specified snap to boundary tolerance, nodes from the secondary edge are projected onto the boundary of the primary face. The joined edge will be on the primary face along with other edges on the primary face that fall within the defined pinch control tolerance. Refer to Figure 120: Snap to Boundary Set to Yes (p. 289) and Figure 121: Snap to Boundary Set to No (p. 289) to see the effect of the setting.

Note:

For edge-to-edge pinch controls in shell models, the snap tolerance is set equal to the pinch tolerance internally and cannot be modified.

9. Change the value of the **Snap Type** control if desired. **Snap Type** appears only when the value of **Snap to Boundary** is **Yes**.

    - If **Snap Type** is set to **Manual Tolerance** (the default), a **Snap Tolerance** field appears where you may enter a numerical value greater than 0. By default, the **Snap Tolerance** is set equal to the pinch tolerance but it can be overridden here.
    - If **Snap Type** is set to **Element Size Factor**, a **Primary Element Size Factor** field appears where you may enter a numerical value greater than 0. The value entered should be a factor of the local element size of the primary topology.

Note:

If you make changes to a pinch control that was created through pinch control automation (p. 189), the value of the **Scope Method** field for that pinch control changes from **Automatic** to **Manual**.

Inflation Control

Inflation is useful for CFD boundary layer resolution, electromagnetic air gap resolution or resolving high stress concentrations for structures. It is supported for the mesh methods listed in the section Inflation Group (p. 145). You can use local inflation mesh controls to apply inflation to specific boundaries. When an inflation control is scoped to a solid model, every scoped geometry must have a boundary defined for it. The settings of the local inflation controls will override global inflation control (p. 145) settings.

You can define local inflation controls either by inflating a method or by inserting individual inflation controls.

Note:

If you are using an assembly meshing algorithm (p. 367), refer to The Assembly Meshing Workflow (p. 372) for inflation control procedures specific to those algorithms.

Inflating a Method

1. Insert a mesh method.

2. Associate (scope) the desired bodies with the method.

3. Right-click the method and choose **Inflate This Method** from the menu.

   Note:

   For example, if you right-click a **Tetrahedrons** method control or a **Sweep** method control and specify the source, then an **Inflate This Method** menu option is available. Choosing this option inserts an inflation control on every body to which the **Tetrahedrons** meshing method with the selected **Algorithm** is applied, or on every face to which the **Sweep** method is applied. Similarly, an inflation control will be inserted into the Tree Outline for each body/face. (Sweeping with inflation is the same as inflation with tetrahedrons except that with sweeping you pick faces and edges instead of bodies and faces.)

4. Highlight one of the inflation controls that was inserted into the Tree Outline.

5. Change the value of the **Suppressed** control if desired.

   By default, the value of **Suppressed** is **No**. If you change the value to **Yes**, this inflation control has no effect on the mesh. In addition, an **Active** control with a read-only setting of **No, Suppressed** appears under the **Suppressed** control when **Suppressed** is set to **Yes**.

6. Use either of these methods to specify the inflation boundaries:

   - In the Details View, set **Boundary Scoping Method** to **Geometry Selection**, pick the entities in the **Geometry** window, and click the **Boundary** field in the Details View to **Apply**.

   - In the Details View, set **Boundary Scoping Method** to **Named Selections**, select a Named Selection from the **Boundary** drop-down, and press **Enter**.

     Note:

     – To select multiple Named Selections to be used as inflation boundaries, press and hold the **Ctrl** key while selecting the Named Selections from the **Boundary** drop-down, and then press **Enter**.

– If none of the predefined Named Selections include the correct topology to be used as an inflation boundary, no Named Selections will be available in the drop-down. For example, if you scoped a surface body with the method in step 2, a Named Selection containing an edge must exist. Otherwise, you cannot select anything from the drop-down.

7. Specify additional inflation options as desired. For details about options, refer to the notes below and to Inflation Group (p. 145).
8. Repeat steps 4 through 7 for each inflation control in the Tree Outline.

Inserting Individual Inflation Controls

1. Optionally, select the desired bodies or faces in the **Geometry** window.
2. Use either of these methods to insert the inflation control:
   - Click **Mesh Control** on the toolbar and choose **Inflation** from the menu.
   - Right-click in the **Geometry** window and choose **Insert** > **Inflation** from the menu.
3. If you selected the bodies or faces in step 1, go directly to step 4. If not, use either of these methods to scope inflation to the desired bodies or faces:
   - In the Details View, set **Scoping Method** to **Geometry Selection**, pick the entities in the **Geometry** window, and click the **Geometry** field in the Details View to **Apply**.
   - In the Details View, set **Scoping Method** to **Named Selection**, and select a Named Selection from the **Named Selection** drop-down.
4. Change the value of the **Suppressed** control if desired.

   By default, the value of **Suppressed** is **No**. If you change the value to **Yes**, this inflation control has no effect on the mesh. In addition, an **Active** control with a read-only setting of **No, Suppressed** appears under the **Suppressed** control when **Suppressed** is set to **Yes**.
5. Use either of these methods to specify the inflation boundaries:
   - In the Details View, set **Boundary Scoping Method** to **Geometry Selection**, pick the entities in the **Geometry** window, and click the **Boundary** field in the Details View to **Apply**.
   - In the Details View, set **Boundary Scoping Method** to **Named Selections**, select a Named Selection from the **Boundary** drop-down, and press **Enter**.

Note:

– To select multiple Named Selections to be used as inflation boundaries, press and hold the **Ctrl** key while selecting the Named Selections from the **Boundary** drop-down, and then press **Enter**.

– If none of the predefined Named Selections include the correct topology to be used as an inflation boundary, no Named Selections will be available in the drop-down. For example, if you selected a face in step 1, a Named Selection containing an edge must exist. Otherwise, you cannot select anything from the drop-down.

6. Specify additional inflation options as desired. For details about options, refer to the notes below and to Inflation Group (p. 145).

Notes on Defining Local Inflation Controls (2D Only)

In most cases, the controls in the global Inflation (p. 145) group apply to both 3D and 2D inflation, and the values set globally will be populated to the local inflation controls. Changes that you make to the local inflation settings will override the global settings. Exceptions and special considerations for defining 2D local inflation are described here.

- The ability to define 2D (face) inflation is not supported for assembly (p. 367) meshing algorithms.
- When defining 2D local inflation, the available options for the **Inflation Option** (p. 150) control are **Smooth Transition** (default), **First Layer Thickness**, and **Total Thickness**. If the **Inflation Option** control is set to **First Aspect Ratio** or **Last Aspect Ratio** globally, it will be set to **Smooth Transition** locally.
  - When **Inflation Option** is **Smooth Transition**, you can set values for Transition Ratio (p. 152), Maximum Layers (p. 153), and Growth Rate (p. 153).
  - If you select **First Layer Thickness**, you can set values for First Layer Height (p. 154), Maximum Layers (p. 153), and Growth Rate (p. 153). The value of **Growth Rate** is used to calculate the heights of the successive layers, as follows:

    Growth Rate = h_{n+1} / h_n

    where h_n = height of layer n
  - If you select **Total Thickness**, you can set values for Number of Layers (p. 153), Growth Rate (p. 153), and Maximum Thickness (p. 154). The first layer height is computed based on these three values. The heights of the successive layers are computed using the same formula shown above under **First Layer Thickness**.

  When 2D inflation is applied to a 3D model (that is, to the face of a 3D body), the local value of **Maximum Layers** will be set equal to the global value by default. However, when it is applied to a 2D model (i.e, to the face of a surface body), the local value of **Maximum Layers** will be set to 2.
- If you are using the Quadrilateral Dominant (p. 245) mesh method with inflation and the Size Function (p. 100) is on, the mesh size of the last inflation layer will be used for the corresponding Quadrilateral Dominant boundary mesh size.

Notes on Defining Local Inflation Controls (3D and 2D)

- To make inflation boundary selection easier, select **Annotation Preferences** from the Toolbar and then deselect **Body Scoping Annotations** in the **Annotation Preferences** option box to toggle the visibility of annotations in the **Geometry** window. For example, after scoping inflation to a body, the body will be displayed using a blue solid annotation. Turn off the body scoping annotations; then select the desired faces as boundaries. For picking internal faces, the **Hide Faces** right-click option may help you to see inside a body. For example, you can select external faces in the **Geometry** window and then use the **Hide Faces** option to hide the selected faces (making it easier to select the internal faces).

- Multiple **Inflation** controls can be scoped to the same body or face with different inflation options on the faces/edges.

- If the mesh method is **Automatic**, the **Patch Conforming** tetrahedron method will be used for inflation on a body and the **Sweep** method will be used for inflation on a face.

- If the mesh method is **Cartesian** (3D only) and **Physics Preference** is set to **CFD**, then three boundary layers are created with total thickness proportional to element size. If **Cartesian** and physics is not **CFD**, then a single boundary layer is added with thickness proportional to element size.

- In the following scenarios, using inflation results in automatic suppression of the refinement (p. 264) control:

  – When automatic inflation (p. 147) (either Program Controlled (p. 148) or All Faces in Chosen Named Selection (p. 149)) is used with refinement in the same model

  – When local inflation (p. 291) is used with refinement in the same body or in the same part

- For information on setting global inflation controls and descriptions of all of the individual inflation controls, refer to Inflation Group (p. 145). For steps to follow to assign inflation depending on the selected mesh method, refer to Inflation Controls (p. 414). For general information on applying inflation controls in combination with the various mesh method controls, refer to Meshing: Mesh Control Interaction Tables (p. 435).

Gasket Control

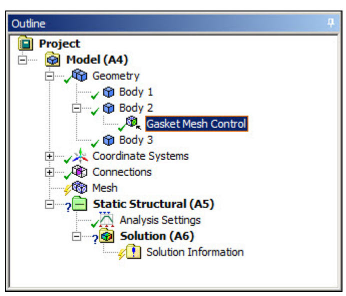
Available when **Body** object's **Stiffness Behavior** is set to **Gasket**. The control applies a sweep mesh in a chosen direction and drops midside nodes on gasket elements that are parallel to the sweep direction. You can directly access **Gasket** from the **Mesh** tab.

Tree Dependencies:

- **Valid Parent Tree Object: Body**

- **Valid Child Tree Objects:** None.

Insertion Options: Appears automatically when a **Body** object's **Stiffness Behavior** is set to **Gasket**.



Additional Related Information:

- Gasket Bodies

- Gasket Meshing

Object Properties

The ***Details view*** properties for this object include the following.

Category	Properties
Definition	**Free Face Mesh Type** **Mesh Method** **Element Order**
Scope	**Src/Trg Selection** **Source** **Target**

Sharp Angle Tool

Using the Sharp Angle Tool, you control the capture of features with sharp angles, such as the edge of a knife or the region where a tire meets the road. It can also be used for improved feature capturing in general, even if the faces that you pick to define a control do not form a sharp angle. The Sharp Angle Tool is available only when assembly meshing (p. 367) algorithms are being used and ensures that the desired features are captured in the assembly mesh.

Using the Sharp Angle Tool

The steps for using the Sharp Angle Tool are presented here.

1. Highlight the **Mesh** object in the Tree Outline and then do one of the following:

   - Right-click the object and select **Insert > Sharp Angle** from the context menu.
   - Select **Mesh Control> Sharp Angle** on the **Mesh** context toolbar.
   - Right-click the **Geometry** window and select **Insert> Sharp Angle** from the context menu.

   These methods insert a sharp angle control beneath the **Mesh** object and activate the **Face** selection filter.

2. Use either of these methods to scope the control to the desired faces:

   - In the Details View, set **Scoping Method** to **Geometry Selection**, pick the faces in the **Geometry** window, and click the **Geometry** field in the Details View to **Apply**.
   - In the Details View, set **Scoping Method** to **Named Selection**, and select a Named Selection from the **Named Selection** drop-down.

   Note:

   If none of the predefined Named Selections include faces, no Named Selections will be available in the drop-down.

3. Change the value of the **Suppressed** control if desired.

   By default, the value of **Suppressed** is **No**. If you change the value to **Yes**, this control has no effect on the mesh (that is, the sharp angle features will not be captured). In addition, an **Active** control with a read-only setting of **No, Suppressed** appears under the **Suppressed** control when **Suppressed** is set to **Yes**.

Notes on the Sharp Angle Tool

- To suppress, unsuppress, or delete a sharp angle control, highlight it in the Tree Outline, right-click, and select the appropriate option from the context menu. You cannot copy a sharp angle control.
- If any sharp angle controls are defined and you change from an assembly meshing algorithm to another mesh method, the controls are suppressed automatically. In such cases, an **Active** control with a read-only setting of **No, Invalid Method** will appear under the **Suppressed** control in the Details View, but the value of the **Suppressed** control will still be set to **No**.

Repair Topology

Repair Topology is available only when **Mesh Based Connections** is set to **Yes** under **Batch Connections** in the **Mesh** Details view.

To Access Repair Topology:

1. Right-click the **Mesh** folder in the Tree Outline.
2. Select **Insert > Repair Topology**.

The following options are available in the **Details** view for **Repair Topology**:

Merge Face Options

- **Merge Faces**: Merges the selected faces. The default value is **No**. When the value is set to **Yes**, the following option appears:
  - **Scoping**: Allows you to select either **Geometry Selection** or **Named Selection**. You can select multiple named selections from the list of available face named selections. After selecting, you must press **Enter** to apply the selection. The default value is **Geometry Selection**. Based on the **Scoping** selected, the number of faces used for the respective selection is displayed.

Supress Edge Options

- **Supress Edges**: Suppress the selected edges. The default value is **No**. When the default value is set to **Yes**, the **Scoping** option appears which works same as in the **Merge Faces** option.

Repair Thin Faces Options

- **Remove Thin Faces**: Removes the thin faces by merging them with the neighboring faces. The default value is **No**. When the value is set to **Yes**, the **Thin Face Width** and **Use Local Geometry Scoping** options appear. The default value for **Thin Face Width** is same as the **Connection**

Tolerance. When **Use Local Geometry Scoping** is set to **No**, performs **Remove Thin Faces** operation on the entire model.

Repair Short EdgesOptions

- **Collapse Short Edges**: Collapses the edges below the specified tolerance. Feature edges and Protected edges are not collapsed. The default value is **No**. When the **Collapse Short Edges** is set to **Yes**, the **Short Edge Length** and **Use Local Geometry Scoping** option is displayed.
  - **Short Edge Length**: Allows you to specify the shortest edge length to performs collapse operation.
  - **Use Local Geometry Scoping**: Allows you to select the scoping method. The default value is **No**. When set to **Yes**, the **Scoping** appears with **Geometry Selection** as default value. When **Scoping** is set to **Geometry Selection**, the **Thin Face Geometry** is displayed that allows you to apply the scoping to the selected faces. When **Scoping** is set to **Named Selection**, allows you to select the available named selections.

Repair Sharp Angle Options

- **Remove Sharp Angle Faces**: Removes the sharp angle faces below the specified angle tolerance by merging them to the neighboring faces. The default value is **No**. When the value is set to **Yes**, the **Sharp Angle** and **Use Local Geometry Scoping** options appear. The default value for **Sharp Angle** is 10 degrees. When **Use Local Geometry Scoping** is set to **No**, performs **Remove Sharp Angle Faces** operation on the entire model.

Repair Pinch Faces Options

- **Pinch Faces**: The default value is **No**. When set to **Yes**, the **Pinch Tolerance** and **Use Local Geometry Scoping** options appear. The default value of **Pinch Tolerance** is same as the **Connection Tolerance**. The default value for **Use Local Geometry Scoping** is **No**. When the **Use Local Geometry Scoping** is set to **Yes**, **Scoping** appears and allows you to select both **Geometry Selection** and **Named Selection**. The default value is **Geometry Selection**. When **Use Local Geometry Scoping** is set to **No**, performs **Repair Pinch Faces** operation on the entire model.

Fill Hole Options

- **Fill Hole**: Fills the selected hole. The default value is **No**. When set to **Yes**, **Scoping** appears and allows you to select either **Geometry Selection** or **Named Selection**.
  - **Geometry Selection**: Allows you to scope only edges for fill hole option. The default value is Geometry Selection.
  - **Named Selection**: Allows you to scope edge-based name selection for fill hole option.

Remove Thin Faces, **Remove Sharp Angle Faces** and **Pinch Faces** allow you to define local scoping using the **Use Local Geometry Scoping** option. The default value for **Use Local Geometry Scoping** is **No**. When the **Use Local Geometry Scoping** is set to **Yes**, **Scoping** appears and allows you to select either **Geometry Selection** or **Named Selection**. The default value is **Geometry Selection**.

Repair Topology provides options to find the unconnected edges and overlapping faces. You can right-click on the **Geometry** window, select **Diagnostics >Find Unconnected Edges** to display the unconnected edges. You can also access the overlapping faces by selecting **Diagnostics >Find Overlapping Faces**.

Note:

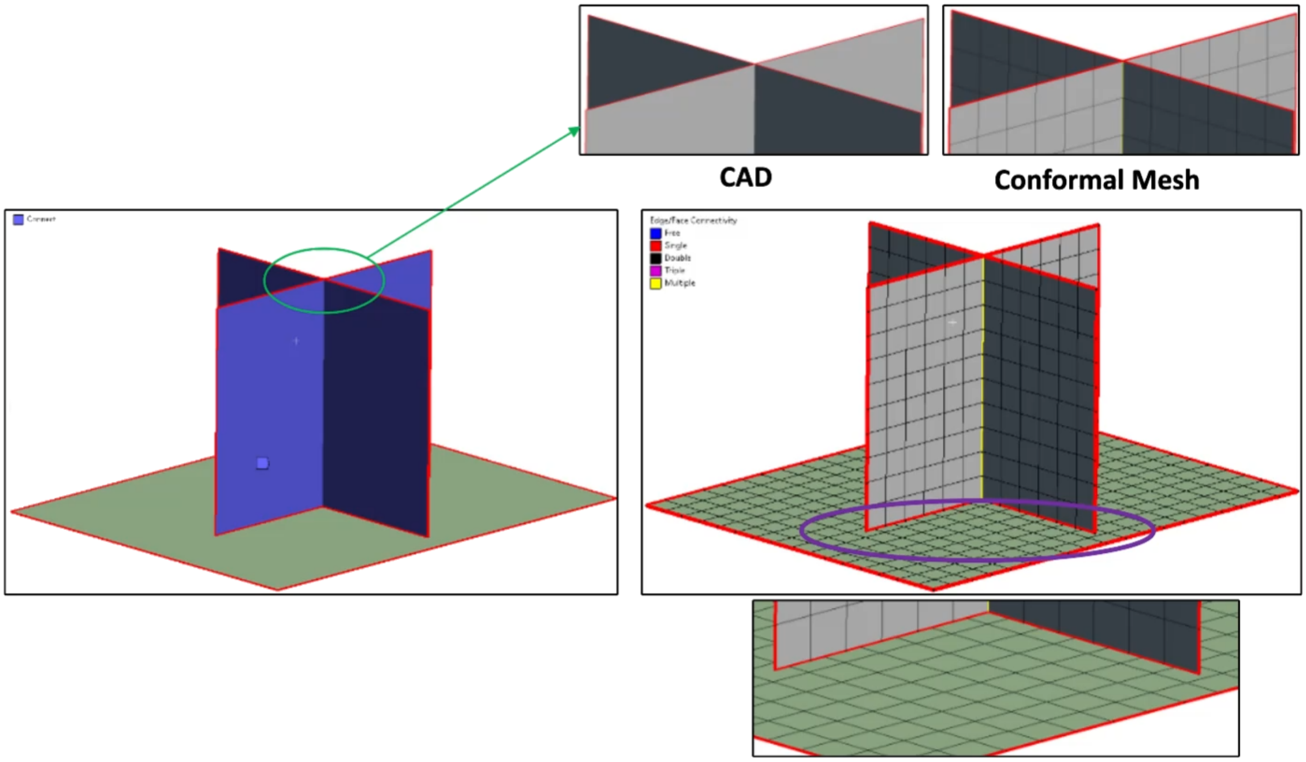
- **Repair Topology** operations are performed at the mesh level. Hence, the changes are not reflected on the input geometry. You should carefully examine mesh association to geometry when using repair topology.
- **Repair Topology** operations respects **Protected Entities**. Hence, **Repair Topology** options do not alter **Protected Entities**.

Connect

Connect control allows you to create conformal mesh between the selected entities. **Connect** control is enabled only when **Mesh Based Connection** is set to **Yes** in **Batch Connections**. To access **Connect** control, right-click **Mesh** object and click **Insert > Connect**. When you click **Connect**, the **Details** view displays the **Connect** options.

Details of "Connect" - Connect

Scope	
Scoping Method	Geometry Selection
Geometry	7 Faces
Use Worksheet	No
Definition	
Multiple Connection Steps	No
Connection Tolerance	Default (1.0 mm)
Connection Size	Default (10.0 mm)
Connection Option	All To All
Coplanar Angle Tolerance	Default (25.0°)
Perform Intersections	Yes



Note:

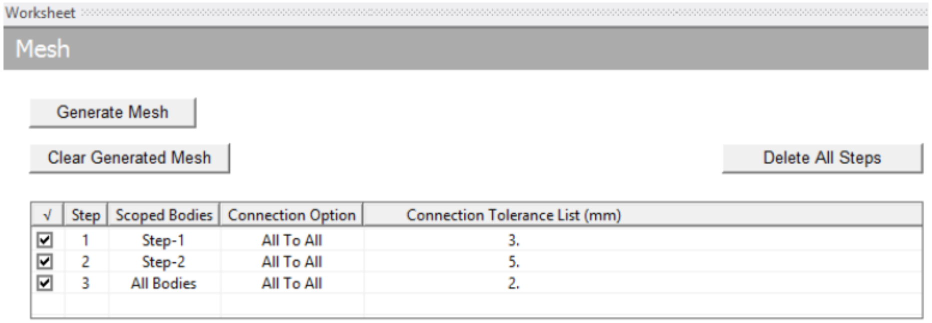
- **Connect** control does not support scoping to solids entities.
- All legacy databases saved with **Batch Connections** will resume with **Connect** control.

Scope

- **Scoping Method**: Allows you to select **Geometry** or **Named Selection** for scoping.
- **Use Worksheet**: Allows you to scope using worksheet when set to **Yes**. The default value is **No**. When **Use Worksheet** is set to **Yes**, the connections are made using worksheet.

  Connections Using Worksheet

  Connections can also be performed using **Worksheet**. **Worksheet** allows you to define the order in which connections are performed and allows you to define varying connection parameters. **Worksheet** is context sensitive. To perform connections using **Worksheet**, you must be in the **Mesh** folder. Click **Worksheet** in **Tools** under the **Home** tab to open the **Worksheet**.



Note:

You can activate, setup and visualize mesh worksheet only from the Mesh folder.

Worksheet allows you to scope named selections. Named selections can either be body based or face based. For each step you can change connection option and specify connection tolerance(s). The available connection options are:

- **All to All**: Connects all possible entities in the scope during the connection process. When **All to All** is selected, sliver faces and short edges below the connection tolerance value are removed as a part of the connection process.
- **Free to All**: Connects only unconnected edges to rest of the entities in the scope.
- **Free to Free**: Connects only unconnected edges in the scope.

The steps involved in **Worksheet** based connection are as follows:

1. Right- click the **Worksheet**.
2. Click **Add** to add a step.
3. Specify scope defined by named selection from the drop-down list in the **Scoped Bodies** field. **AllBodies** is a named selection available by default.
4. Select the **Connection Option**.
5. Specify connection tolerance or list of tolerances for the step. The connection tolerance(s) behaves same as global connection tolerance and list of tolerance.
6. Add as many steps as required for the connection.

When **Generate Mesh** is clicked connections are established in the sequence defined in the worksheet. The mesh is generated after all the connections are made.

Note:

All Bodies is available by default under scoped bodies in the **Worksheet**. This named selection is scoped to all bodies in the assembly.

Definition

- **Multiple Connection Steps**: Allows you to provide multiple values for connection tolerance. The default value is **No**. When you set **Multiple Connection Steps** to **No**, **Connection Tolerance** field is available to provide the tolerance value for connection. When you set the **Multiple Connection Steps** to **Yes**, the **Connection Tolerance List** is available to provide multiple values for connection tolerance. You can specify any number of connection tolerance values separated by a space. The first value in **Connection Tolerance List** performs face to face intersections, short edge removal, thin face removal and so on. From the second tolerance value onwards only the unconnected (free) edges are considered for performing connections.

- **Connection Size**: Defines the size with which you discretize the edges before connecting them. By default, the connection size is same as **Element Size**.

- **Coplanar Angle Tolerance**: Checks whether the two faces to be connected are in the same plane or not. When the faces are in the same plane, intersection is not performed. The value of **Coplanar Angle Tolerance** is in degrees.

- **Perform Intersections**: Allows you to skip intersection while connecting the selected entities. The default value is **Yes**.

Note:

When you use **Connect and Mesh Selected Entities** on legacy files which have partially saved mesh, the previously saved mesh gets cleared. You may have to regenerate mesh for the entire model.

Weld

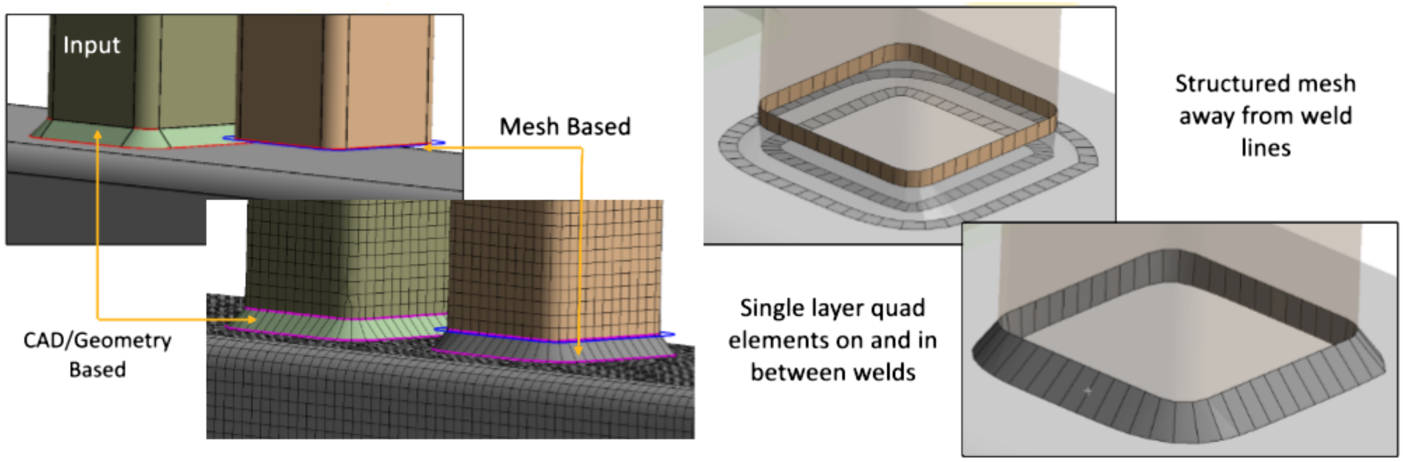
Weld control allows you to create weld bodies and, or generate layers of quad elements on the weld faces (tent and extension faces), in between the tent and extension faces and on the faces that are welded together along the edges shared with the tent and extension faces. When you select **Source** as **Mesh** and **Create using** options like **Curves**, **Curves and Bodies** or **Curves and Faces**, tent and extension faces are created during meshing using defined **Weld Curve**. The faces are associated to a weld body that is created during meshing. The weld body is available under the **Geometry** tree object. The weld body name is same as that of the **Weld** control. One weld body gets created per **Weld Curve**.

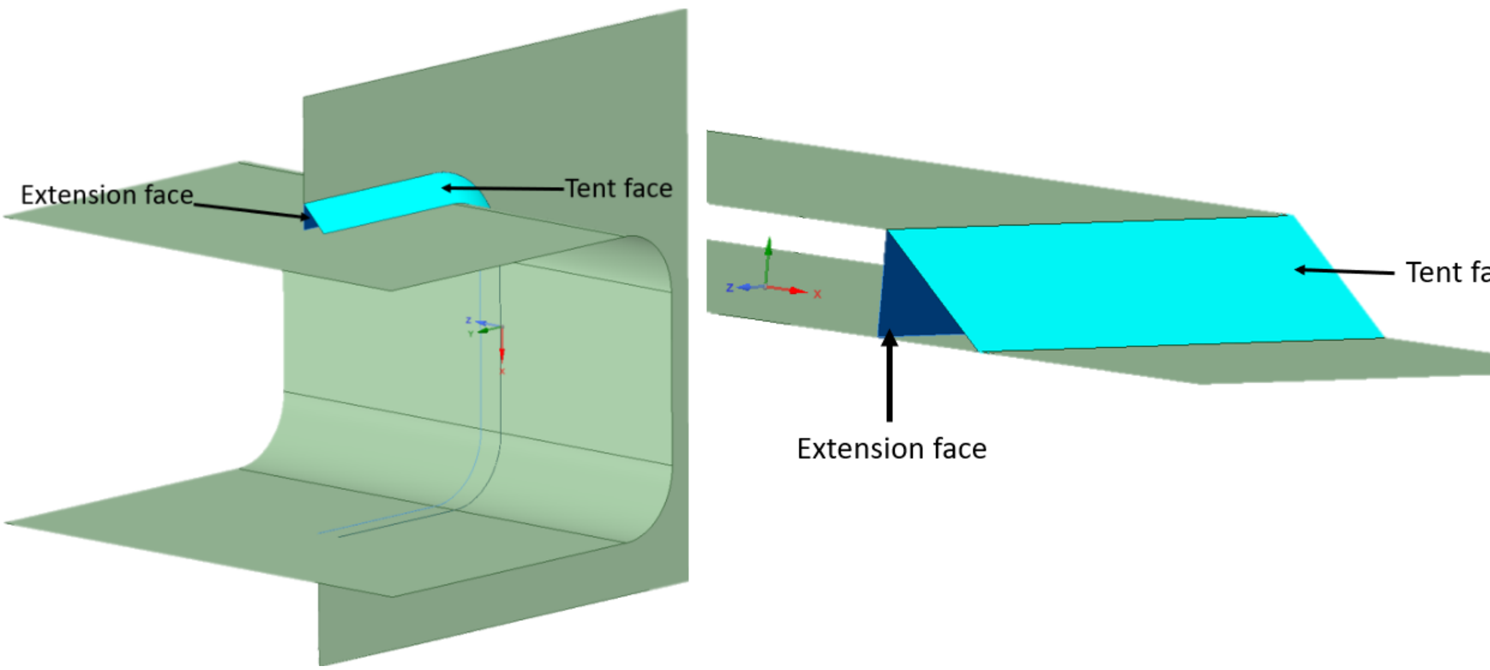
Note:

Weld bodies created at the mesh level are deleted, when the following occurs:

- Mesh is cleared.

- Geometry is updated.





You can access the weld control as follows:

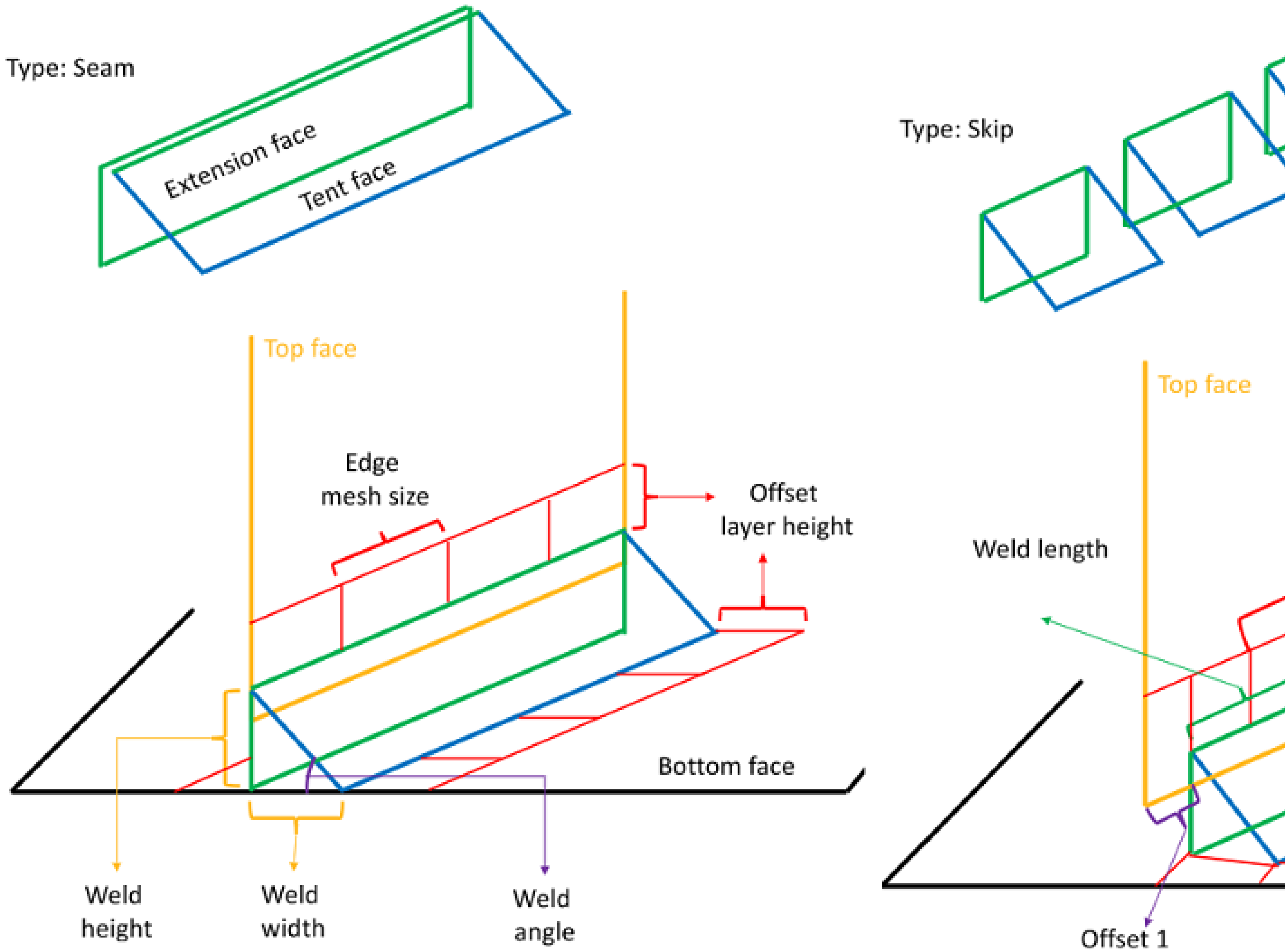
1. In the Tree Outline, click **Mesh** folder, set **Mesh Based Connections** to **Yes** in the **Details View**.
2. Right-click the **Mesh** folder in the Tree Outline.
3. Select **Insert > Weld**.

The following options are available in the **Details** view for **Weld**:

Scope

- **Scoping Method**: You can use **Geometry Selection** or **Named Selections** for scoping. The default value is **Geometry Selection**. **Scoping Method** is not available only when the **Create Using** is set to **Curves**.
- **Type**: Allows you to select the type of weld control. The available types are:
  - **Seam**: Allows you to perform continuous welding.

– **Skip**: Allows you to perform discontinuous welding. When you select **Type** as **Skip**, the **Source** is **Mesh** by default and is a read only field.



- **Source**: Allows you to select **Geometry** or **Mesh** for welding. **Source** represents where the weld faces are modeled. You can select **Geometry**, for weld faces created in CAD. You can select **Mesh**, when the weld entities are to be created during mesh generation.

- **Modeled As**: Allows you to choose or define how weld entities are modeled. When the **Source** is **Geometry** and **Type** is **Seam**, you need to define what the weld faces represent. The available options are:

  – **Tent**: Creates mesh only on the tent face for the weld.

  – **Extension**: Creates mesh only on the extension face for the weld.

  – **Tent and Extension**: Creates mesh on the tent or extension faces with one layer of quad mesh on both tent and extension with the user defined offset layers on top and bottom faces. When you select **Tent and Extension**, the **Extension Surface** and **Tent Surfaces** fields are available.

– **Parent Bodies Connection**: Connects the top and bottom faces without creating single quad layer on the tent or extension face. When you select **Parent Bodies Connection**, the **Bottom Faces** and **Top Faces** fields are available.

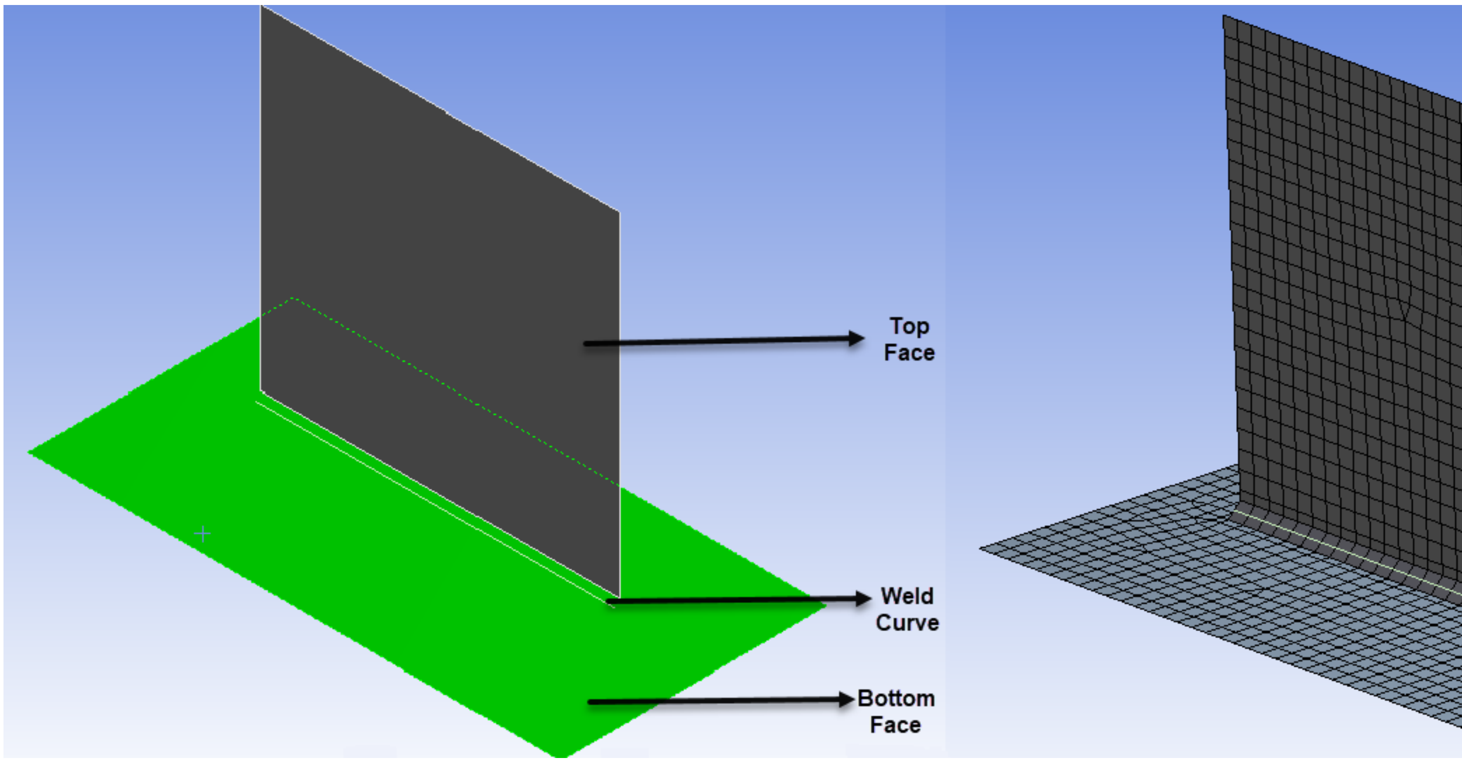
When you select the **Source** as **Mesh**, the available options are:

– **Tent and Extension**: Creates both tent and extension face for the weld.

– **Tent**: Creates only the tent face for the weld.

– **Extension**: Creates only the extension face for the weld.

– **1D**: Creates extension only weld using beams. Here beams are conformally connected to the top and bottom faces. For input edge or curves, the beams created are represented as a single body. **1D** is available only when the **Source** is **Mesh**.



- **Create Using**: Allows you to create weld using **Curves**, **Curves and Bodies**, **Curves and Faces**, **Edges**, **Edges and Bodies** or **Edges and Faces**. Available only when the **Source** is **Mesh**.

  – **Curves**: Allows you to pick a beam body and use it as weld curve. Software automatically determines the faces in proximity of the defined weld curve to be welded. When you select **Curves**, the **Weld Curve** option is available.

  – **Curves and Bodies**: Allows you to define bodies to be welded together in addition to the weld curve. When you select **Curves and Bodies**, the **Bottom Bodies**, **Top Bodies**, **Weld Curve** options are available.

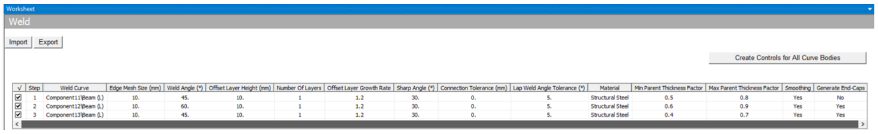
  – **Curves and Faces**: Allows you to define faces to be welded together in addition to the weld curve. This option gives you better control on defining the scope for weld creation during mesh generation. When you select **Curves and Faces**, the **Bottom Faces**, **Top Faces**, **Weld Curve** options are available.

- **Edges**: Allows you to scope geometry edges to create welds. The edges scoped to a weld control or in a row of worksheet must be continuous. When you select **Edges**, the **Edges** option is available.
- **Edges and Bodies**: Allows you to define bodies to be welded together in addition to the geometry edges. When you select **Edges and Bodies**, the **Bottom Bodies**, **Top Bodies**, **Edges** options are available.
- **Edges and Faces**: Allows you to define faces to be welded together in addition to the geometry edges. This option gives you better control on defining the scope for weld creation during mesh generation. When you select **Edges and Faces**, the **Bottom Faces**, **Top Faces**, **Edges** options are available.

Note:

If the **Curves** or **Edges** option is providing incorrect result for weld creation, then using **Curves and Faces** or **Edges and Faces** is recommended.

- **Tent Direction**: Allows you to specify the direction of tent creation. The available options are **Normal**, **Reversed** and **Both**.
  - **Normal**: Allows you to create tent in the same direction of the face normal direct for edges-based inputs and along the weld curve for curve-based input.
  - **Reversed**: Allows you to create tent in the opposite direction of the face normal direct for edge-based input and opposite to the weld curve for curve-based input.
  - **Both**: Allows you to create tent along both the same direction and opposite direction of the face normal direct for edge-based inputs and both the same and opposite sides of the weld curve for curve-based input.
- **Use Worksheet**: Allows you to use worksheet for weld controls. **Worksheet** enables you to combine multiple weld controls in a single weld control. The default value is **No**. When **Use Worksheet** is set to **Yes, Scope** with **Worksheet** appears in the **Details** view which is a read-only field. The **Worksheet** opens and allows you to create weld controls for the **Named Selections**. **Worksheet** provides different options based on the selected **Source** and **Type**. The options available in **Worksheet** are read only in the **Details** view. You can define these options in the **Worksheet**.



Note:

Worksheet allows you to create weld controls for curved bodies using **Body Selection**. Worksheet allows you to import and export the worksheet parameters as .CSV file format. You can also edit the CSV file if required.

The .CSV file import format should be as given below:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	#	0	1	0	3	0	0	0							
2	!	0	1	5	6	8	14	15	25	17	18	19	20	24	16
3	Active	Step Number	Edge	Edge Mesh Size	Weld Angle	OffsetLayer Height	Number of Layers	OffsetLayer Growth Rate	Sharp Angle	Connection Tolerance	Lap Angle Tolerance	Material Id	Smoothing	Generate EndCaps	
4	Yes	1	panel-1:Pipe-right:weld 1	3	45	2	1	1.2	90	0	5	10	Yes	Yes	

Rows 1 and 2 are used to populate the weld control Details view. The values in Row 1 and 2 should not be modified. The values in Row 1 represents the following:

	A	B	C	D	E	F	G	H
1	**Number**	**Type** 0 - Seam 1 - Skip	**Source** 0 - Geometry 1 - Mesh	**Modeled As** 0 - Tent and Extension 1 - Parent Bodies Connection 2 - Tent 3 - Extension 4 - 1D	**Create Using** 0 - Curves 1 - Curves and Bodies 2 - Curves and Face 3 - Edges 4 - Edges and Bodies 5 - Edges and Faces	**Adjust weld Height** 0 - No 1 - Yes	**Creation Criteria** 0 - Angle Based 1 - Width Based	**Thickness Assignment** 0 - Program Controlled 1 - Min Parent Thickness Factor 2 - Max Parent Thickness Factor 3 - Both 4 - User Defined

In Row 2, the numbers indicate the columns in worksheet. The details are as follows:

- 0 - Active
- 1 - StepNumber
- 2 - Bottom Face/Body/ Extension surface
- 3 - Top Face/Body/Tent surface
- 4 - Weld Curve
- 5 - Edge
- 6 - Edge Mesh Size
- 7 - Adjust Height
- 8 - Weld Angle,
- 9 - Weld Length

- 10 - Weld Pitch
- 11 - Number of Welds
- 12 - Offset1
- 13 - Offset2
- 14 - Offset Layer Height
- 15 - Number of Layers
- 16 - Generate Endcaps
- 17 - Sharp Angle
- 18 - Connection Tolerance
- 19 - Lap Angle Tolerance
- 20 - Material Id
- 21 - Thickness Value
- 22 - Min Thickness Factor
- 23 - Max Thickness Factor
- 24 - Smoothing
- 25 - Offset Layer Growth Rate

Note:

- Named selections for edges should be unique.
- For each row, the edges scoped to the named selection must be continuous.
- If welds are defined using **Curves**, then the names of weld curves must be unique.

Worksheet enables you to handle errors and failures efficiently. The **Worksheet** rows highlighted in yellow denotes warning messages and in red denotes error messages. You can right-click **Weld>Deactivate Problematic Worksheet Entries** to deactivate the row which is causing error while meshing.

- **Curve Scoping**: Available only when the **Source** is **Mesh** and **Creating Using** is set to either **Curves**, **Curves and Bodies** or **Curves and Faces**. The available options are **Geometry Selection** and **Body Selection**. The default value is **Geometry Selection**. **Body Selection** allows you to select the weld curve from the list of beam bodies.

> **Note:**
>
> When you use beam bodies as weld curve, **Treatment** is automatically set to **Construction Body**.
>
> **Treatment** does not reset automatically, when the beam body is removed from the **Weld Curve** scope.

Definition

- **Suppressed**: The default value is **No**. When the value is set to **Yes**, the **Active** options appears with a read-only setting **No,Suppressed**.
- **Adjust Weld Height**: Allows you to adjust the weld height. The default value is **No**. When the **Adjust Weld Height** is set to **Yes**, the **Weld Height** field appears.
  - **Weld Height**: Allows you to define the height of the weld created during mesh generation. The default value is calculated as an average of the sheet thicknesses of the bodies connected by the weld. If the sheet thicknesses of the bodies are not available, no height adjustment is performed during mesh generation.
- **Weld Length**: Allows you to define the length of each weld in the skip. The **Weld Length** is available only when the selected **Type** is **Skip**. There is no default value for weld length. Thus, you must define the weld length manually.
- **Weld Pitch**: Allows you to define the pitch of the weld. **Weld Pitch** is the distance from mid-point of one weld to the mid-point of adjacent weld in a skip weld. **Weld Pitch** is therefore the sum of the weld length and the gap between each of the welds. There is no default value for **Weld Pitch**. Therefore, you must define the **Weld Pitch** manually. **Weld Pitch** must be greater than **Weld Length**.
- **Number of Welds**: Allows you to define the number of skip welds on the **Weld Curve** or **Edges**. You can also specify the number of skip welds. The skip welds are created based on the specified **Weld length** and **Weld Pitch**.Once the defined **Number of Welds** is achieved, no further welds are created on the scoped **Weld Curve** or **Edges**.
- **Offset 1**: Allows you to specify the distance between the start vertex and the first skip weld. The default value is 0.
- **Offset 2**: Allows you to specify the distance between the last skip weld and and the end vertex in the scoped **Weld Curve** or **Edges**. The default value is 0.
- **Creation Criteria**: Allows you to define the criteria for the weld creation. The field is only available when the **Source** is **Mesh**. The available options are:

- **Angle Based**: Allows you to control the angle between the tent face and bottom face. When you select **Angle Based** for creating weld, the **Weld Angle** appears. The default value is 45 degrees. You can select any value from 30 to 60 degrees for Weld Angle.

  Note:

  Extension face is always created perpendicular to the bottom face.

- **Width Based**: Allows you to create welds of constant width along the weld line. When you select **Width Based** for creating weld, the **Weld Width** appears. The default value is to use **Weld Height**. If **Adjust Weld Height** is set to **No**, the default width is calculated like default **Weld Height** calculation.

• **Edge Mesh Size**: Allows you to define the mesh size used to discretize the edges defined as weld lines. The default value for **Edge Mesh Size** is same as the global **Element Size**.

Note:

The **Edge Mesh Size** should be reduced for welds involving high curvature.

• **Offset Layer Height** : Allows you to define the height for each quad layer. The default value is the **Global Element Size**.

• **Number Of Layers**: Allows you to define the number of quad layers generated from the weld lines. The default value for **Number Of Layers** is **1** and the maximum number of layers that can be generated is **3**.

• **Generate End-Caps**: Allows you to generate triangular end-caps at the free ends of the welds. The default value is **Yes**. When the value is set to **No**, the following options appear:

- **Write Definition File**: Allows you to write the FE-Safe weld definition files at the defined location. The default value is **Yes**.

- **File Location**: Writes the Weld definition files in the project directory or at location specified by you. Weld definition files allows you to import **Weld** information into FE-Safe.

Note:

- End-caps are not created when **Modeled As** is set to **Extensions**.

- End-caps are created when **Modeled As** is set to **Tent**. End-caps are usually created in plane of the tent face. However, if there is no enough space to create in-plane end-caps, then they can be created out of tent face plane.

• **Generate Named Selection**: Available only when the **Source** is **Mesh**.The default value is **No**.The available options are:

- **Weld**: Creates FE named selection on the weld body for the respective mesh control. This named selection is added under Named Selection tree-node with the name same as that of the weld control. The default value is **No**.
- **HAZ Layer 1**: Creates named selection on mesh elements for the first offset layer.
- **HAZ Layer 2**: Creates named selection on mesh elements for the second offset layer.
- **HAZ Layer 3**: Creates named selection on mesh elements for the third offset layer.

Note:

The **HAZ Layer** option created depends on the **Number Of Layers**. That is, if the **Number of Layer** is set to 1, only **HAZ Layer 1** option is created. The **HAZ Layer 2** and **HAZ Layer 3** options are not created when the **Number of Layers** is 1.

Mechanical Properties

- **Material**: Allows you to define the material for the weld bodies created during meshing. You can only select the defined materials for weld bodies from the drop-down menu. The default option is **None**.
- **Cross Section**: Allows you to assign cross section to weld bodies **Modeled As 1D**. This option is available only when **Modeled As** is set to **1D**.
- **Thickness Assignment**: Allows you to select different options to define the thickness of weld bodies created during meshing. The available options are:
  - **Program Controlled**: Allows the application to control the thickness of the weld bodies during meshing. The default value is Program Controlled. Here thickness is calculated as follows:

    Weld Thickness = (Average thickness of top and bottom weld bodies) / $\sqrt{2}$
  - **Min Parent Thickness Factor**: Allows you to specify the minimum thickness factor for weld bodies during meshing. Here thickness is calculated as follows:

    Weld Thickness= Min Parent Thickness Factor * min (Average thickness of all top and bottom faces for the created weld)
  - **Max Parent Thickness Factor**:Allows you to define the maximum thickness factor for weld bodies during meshing. Here thickness is calculated as follows:

    Weld Thickness= Max Parent Thickness Factor * max (Average thickness of all top and bottom faces for the created weld).
  - **Both**: Allows you to specify the minimum and maximum thickness factor for weld bodies during meshing. Here thickness is calculated as follows:

    Weld Thickness= Min Parent Thickness Factor * min(Average thickness of all top and bottom faces for the created weld) + Max Parent Thickness Factor * max(Average thickness of all top and bottom faces for the created weld).

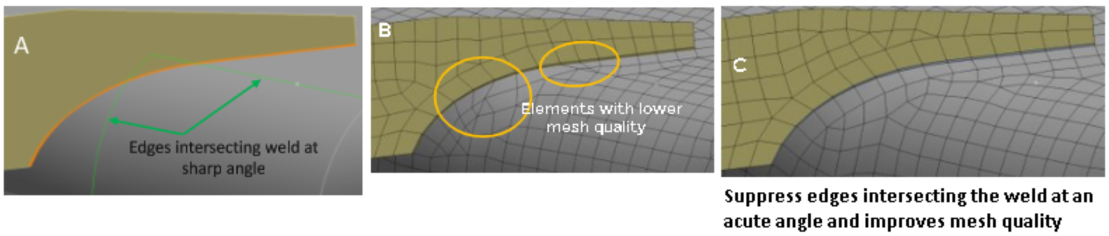
– **User Defined**: Allows you to define the **Thickness** value for weld bodies during meshing.

Note:

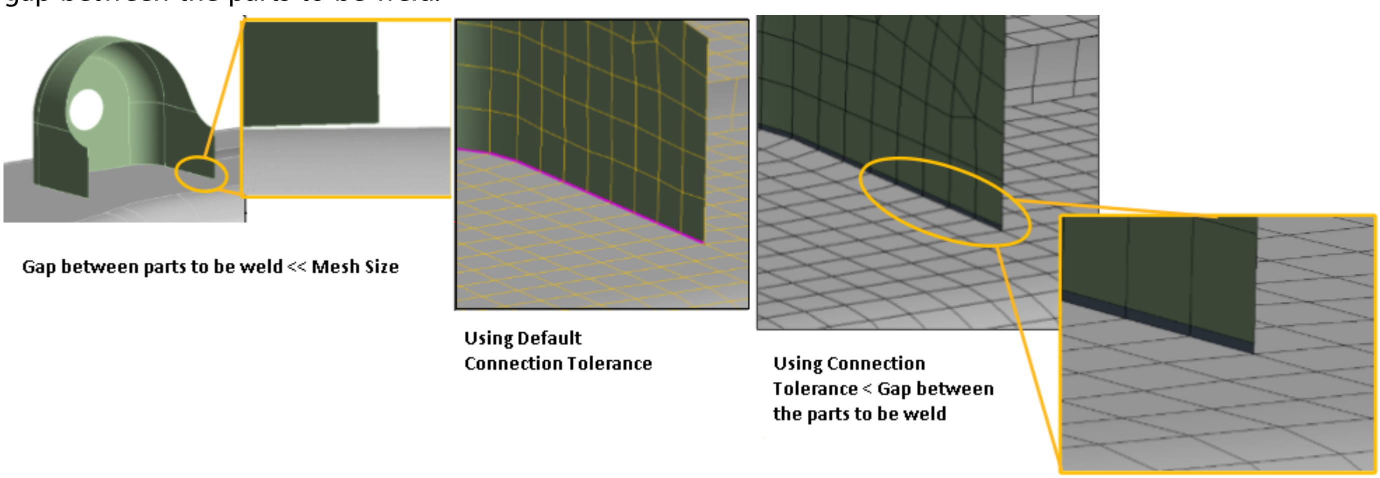
The **Mechanical Properties** options are not available when the **Source** is **Geometry**.

Advanced

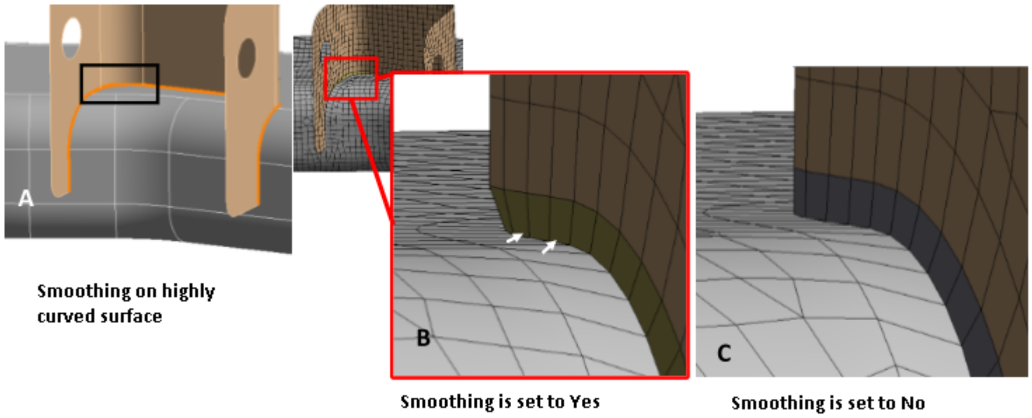
- **Sharp Angle**: Allows you to automatically merge the bottom faces if the weld faces intersect them at an angle less than the specified value. The default value is **30** degrees. The **Sharp Angle** allows you to provide any value from **0** to **90** degrees.



- **Connection Tolerance**: Allows you to connect the weld bodies created during mesh generation to the parent metal bodies. Available only when the **Source** is **Mesh**. When two parts to be welded together are close to each other, the **Connection Tolerance** should be less than the gap between the parts to be weld.



- **Smoothing**: Available only when the **Source** is **Mesh**. The default value is **Yes**. **Smoothing** should be set to **No** for highly curved surfaces to achieve better results.



- **Lap Weld Angle Tolerance**: Allows you to adjust tent direction by specifying appropriate angle. **Lap Weld Angle Tolerance** is available only when the **Source** is **Mesh** and **Modeled As** is either **Tent** or **Tent and Extension**. The default value is **5** degree. The acceptable range of **Lap Weld Angle Tolerance** is **0** to **90** degrees.

Note:

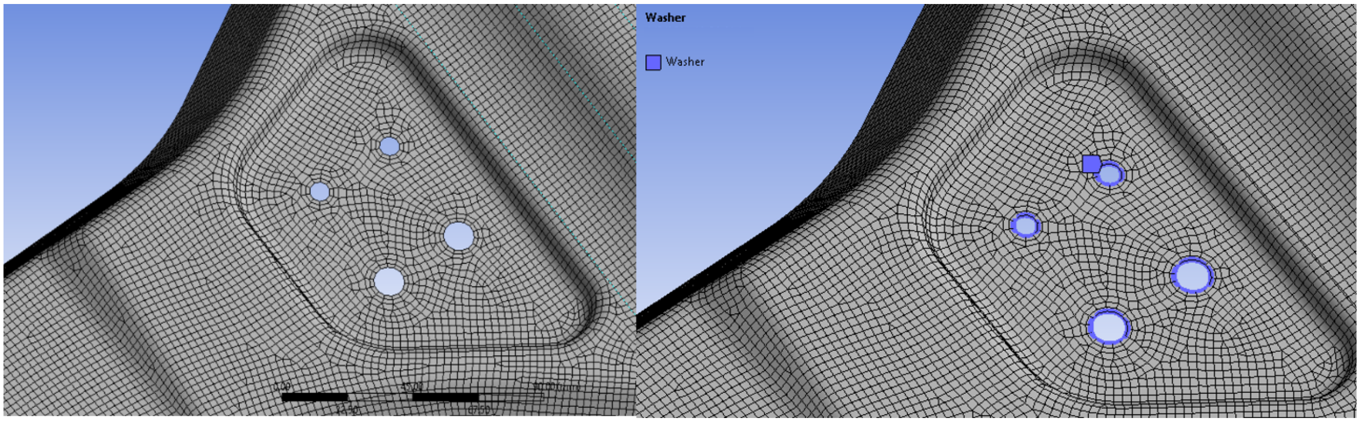
Cross hair is implemented for Edge Mesh Size, Offset Layer Height and Connection Tolerance, the diameter of the cross hair is equal to the value of the corresponding parameter.

The **Weld** control has the following limitations:

- **Weld** control does not support **Connect and Mesh Selected Entities** option.
- Combined length of all tent faces and combined length of all extension faces should be same.
- **Weld** control overrides all other sizing and local control settings.
- The selection option for **Weld Curve** does not support multiple bodies.
- CAD edges scoped in a weld control or in one row of weld control worksheet must be continuous.
- **Weld** control does not support intersecting welds for all sources.
- **Weld** Control does not support triangular mesh, when you select **Mesh Type** as **Triangles**.
- **Weld Worksheet** allows you to import only files with plain CSV format.

Washer

Washer control allows you to create layer of quadrilateral elements around holes. **Washer** control enables you to scope the edge loops or edge based named selections. Washer control is available only when the **Mesh Based Connections** is set to **Yes** under **Batch Connections** in **Mesh** Details view. To access the **Washer** control, right-click **Mesh** object and click **Insert** > **Washer**.



When you click **Washer**, the Details view displays the **Washer** options:

Scope

- **Scoping Method**: Allows you to select the scoping method for washer control. The default method is **Geometry**.
  - **Geometry**: Allows you to scope the edge loops.
  - **Named selection**: Allows you to scope the edge based named selection.

Definition

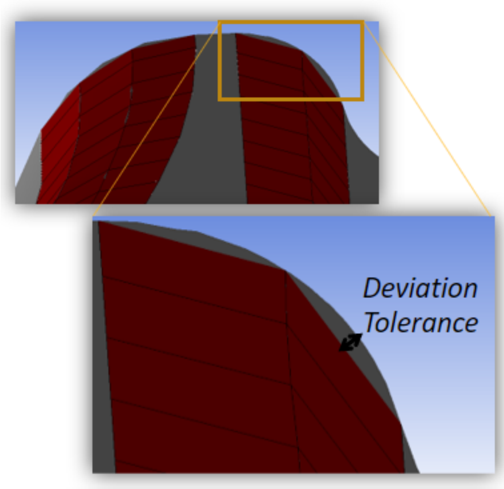
- **Suppressed**: Allows you to suppress the washer control. The default value is **No**. When **Suppressed** is set to **Yes**, the **Active** field displays the status of the washer control. The **Active** field is read-only.
- **Number of Layers**: Allows you to specify the required number of quad layers to be created. The default value is 1 and the maximum allowed number of layers is 3.
- **Layer Height**: Allows you to specify the height of the washer layers. The default value is same as the **Element Size**.
- **Growth Rate**: Allows you to specify the width ratio between the adjacent washer layers. The default value is 1.2
- **Allow node movement**: Allows you to move or shrink the hole with the specified tolerance. The default value is **No**. You can select Move or Shrink based on the operation you want to perform. When you select **Yes**, the washer mesh edges deviate slightly at the location to provide better mesh quality. This option provides flexibility to improve the quality of the mesh.
  - **Node move tolerance**: Allows you to specify the tolerance for Move or Shrink on a hole.

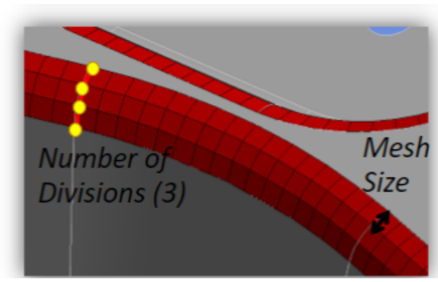
Limitations

- **Washer** Control does not support **Sizing Option**.
- **Washer** Control supports only quadrilateral mesh.

Deviation

Deviation control enables you to scope faces or face based named selections. **Deviation** control is available only when the **Mesh Based Connections** is set to **Yes** under **Batch Connections** in **Mesh** Details view. To access the **Deviation** control, right-click **Mesh** object and click **Insert > Deviation**.





When you click **Deviation**, the Details view displays the deviation options:

Scope

- **Scoping Method**: Allows you to select the scoping method for **Deviation** control. The default method is **Geometry**.
  - **Geometry**: Allows you to scope the faces.
  - **Named Selection**: Allows you to scope the face based named selection.

Definition

- **Suppressed**: Allows you to suppress the **Deviation** control. The default value is **No**. When **Suppressed** is set to **Yes**, the **Active** field displays the status of the **Deviation** control. The **Active** field is read-only.
- **Type**: Allows you to define the type of deviation control specifications. The available options are **Deviation Tolerance**, **Mesh Size**, **Number of Divisions**.
  - **Deviation Tolerance**: Allows you to specify the maximum distance between the mesh and the curvature of the fillet. You can provide smaller **Deviation Tolerance** to get a better curvature resolution.
  - **Mesh Size**: Allows you to specify the mesh size on the shorter sides of the fillets. The default value for **Mesh Size** is 75% of the chosen **Element Size**.
  - **Number of Divisions**: Allows you to specify the number of divisions on the curved sides of the faces.

Limitations

- In the **Mesh** Details view, when the **Capture Curvature** is set to **Yes**, the **Deviation** control is not considered while meshing.
- **Deviation** control supports only quadrilateral mesh.
- **Deviation** control does not work on fillets having sharp angle edges.
- **Deviation** control only meshes continuous body. It does not mesh any exclusion or inclusion in the model.
- When **Deviation** control cannot be applied, the default quadrilateral dominant mesh is generated.
- When there is an extra topological node on one side of the shorter edge of the fillet, the node gets projected to the other shorter edge of the fillet to get all quadrilateral mesh for **Deviation** control.
- The **Deviation** control generates quadrilateral mesh on triangular and polygonal shaped fillets.

Meshing: Options

You can control the behavior of functions in the Meshing application through the **Options** dialog box. For more information, refer to:

Accessing the Options Dialog Box

To access the Meshing application options (p. 317):

1. From the main menu, choose **Tools**> **Options**. An **Options** dialog box appears and the options are displayed on the left.

2. Click on a specific option on the left (either **Meshing** or **Export**).

3. Change any of the option settings by clicking directly in the option field on the right. You will first see a visual indication for the kind of interaction required in the field (examples are drop-down menus, secondary dialog boxes, direct text entries).

4. Click **OK**.

Note:

- To cancel your changes and restore the default settings, click **Reset** on the **Options** dialog box.

- Option settings within a particular language are independent of option settings in another language. If you change any options from their default settings, then start a new Workbench session in a different language, the changes you made in the original language session are not reflected in the new session. You are advised to make the same option changes in the new language session.

Meshing Options on the Options Dialog Box

The options that appear in the right pane of the **Options** dialog box depend on which category is selected in the left pane:

- When **Meshing** is selected in the left pane, these subcategories appear:
  - Meshing

  - Virtual Topology
  - Sizing
  - Quality
  - Inflation
- When **Export** is selected in the left pane, these subcategories appear:
  - CGNS
  - Ansys Fluent

Meshing

- **Highlight Topology Being Meshed if Possible**: Controls the default for highlighting of topologies during mesh processing. When **Yes** (default), the topology that is currently being processed by the mesher is highlighted in the **Geometry** window, which may help with troubleshooting (p. 536). This highlighting is not supported for the **Patch Independent Tetra** or **MultiZone** mesh methods, or when assembly meshing is being used. Refer to Generating Mesh (p. 486) for details.
- **Allow Selective Meshing**: Allows/disallows selective meshing. Choices are **Yes** and **No**. The default value is **Yes**. Refer to Selective Meshing (p. 404) for details.
- **Number of CPUs for Meshing Methods**: Specifies the number of processors to be used by the meshing operation. There is no counterpart setting in the Details View. Specifying multiple processors will enhance the performance of the **MultiZone Quad/Tri**, **Patch Independent Tetra**, and **MultiZone** mesh methods. This option has no effect when other mesh methods are being used. You can specify a value from **0** to **256** or accept the default. The default is **0**, which means the number of processors will be set automatically to the maximum number of available CPUs. The **Number of CPUs** option is applicable to shared memory parallel (SMP) meshing (multiple cores; not supported for clusters). Refer to Parallel Part Meshing (p. 433).
- **Default Physics Preference**: Sets the default option for the **Physics Preference** (p. 93) in the Details View of a **Mesh** object. Choices are **Mechanical**, **Nonlinear Mechanical**, **Electromagnetics**, **CFD**, **Explicit**, and **Hydrodynamics**.
- **Default Method**: Sets the default **Method** setting in the Details View of a **Method** (p. 196) control object. This option only affects **Method** controls that are added manually. Choices are **Automatic (Patch Conforming/Sweeping)**, **Patch Independent**, and **Patch Conforming**. When the geometry is attached, the default method of all options except **Automatic** will be scoped to all parts in the assembly. The **Automatic** option has no effect.

  Note:

  Changing the **Default Method** changes the default mesh method for all future analyses, regardless of analysis type.

- **Use MultiZone for Sweepable Bodies**: If set to **On**, the mesher uses the MultiZone (p. 343) method instead of General Sweeping (p. 323) for sweepable bodies. The default setting is **Off**. See MultiZone for Sweepable Bodies (p. 346) for more information.

- **Topology Checking**: Sets the default value for the Topology Checking (p. 179) control. The default value is **Yes**.

- **Verbose Messages from Meshing**: Controls the verbosity of messages returned to you. If set to **On** and you are meshing a subset of bodies, the message "These bodies are going to be meshed" appears, and you can click the right mouse button on the message to see the bodies. The default is **Off**. Regardless of the setting, when meshing completes and any bodies failed to mesh, the message "These bodies failed to be meshed" appears, and you can click the right mouse button to see them.

- **Number of CPUs for Parallel Part Meshing**: Sets the default number of processors to be used for parallel part meshing. You can change this value in the **Advanced** (p. 93) group under the Details View. Using the default for specifying multiple processors will enhance meshing performance on geometries with multiple parts. For parallel part meshing, the default is set to **Program Controlled** or **0**. This instructs the mesher to use all available CPU cores. The Default setting inherently limits 2 GB memory per CPU core. An explicit value can be specified between **0** and **256**, where **0** is the default. Refer to Parallel Part Meshing (p. 433) for more details.

Virtual Topology

Merge Edges Bounding Manually Created Faces: Sets the default value for the **Merge Face Edges** setting in the Details View of a **Virtual Topology** object. Choices are **Yes** and **No**. The default value is **Yes**.

Sizing

- **Adaptive Resolution**: Sets the resolution for mesh sizing when **Use Adaptive Sizing** is set to **Yes**. The default setting is **Program Controlled**. The range of values that can be set is 0 to 7, with the mesh resolution changing from coarse (**0**) to fine (**7**). See Resolution (p. 104) for more information.

- **Mechanical Min Size Factor (Default: 0.01)**: Sets your preference for the scale factor that will be used to calculate the default minimum size when the physics preference is **Mechanical**, **Electromagnetics**, or **Explicit**. The value that is specified here is multiplied by the global element size to determine the default minimum size.

- **CFD Min Size Factor (Default: 0.01)**: Sets your preference for the scale factor that will be used to calculate the default minimum size when the physics preference is **CFD**. The value that is specified here is multiplied by the global element size to determine the default minimum size.

- **Mechanical Defeature Size Factor (Default: 0.005)**: Sets your preference for the scale factor that will be used to calculate the default defeature size when the physics preference is **Mechanical**, **Electromagnetics**, or **Explicit**. The value that is specified here is multiplied by the global element size to determine the default defeature size.

- **CFD Defeature Size Factor (Default: 0.005)**: Sets your preference for the scale factor that will be used to calculate the default defeature size when the physics preference is **CFD**. The value that is specified here is multiplied by the global element size to determine the default defeature size.

- **Bounding Box Factor (Default: 0.05)**: Helps set the default Element size as follows: (Bounding Box Diagonal * Bounding Box Factor = Default Element size). This is only used when **Use Adaptive Sizing** is set to **No** and only solid parts are present in the model. **Adaptive** sizing uses its own default.

- **Surface Area Factor (Default: 0.125)**: Helps set the default Element size as follows: (Average Surface Area * Surface Area Factor = Default Element Size). This is only used when **Use Adaptive Sizing** is set to **No** and sheet bodies are present in the model. **Adaptive** sizing uses its own default.

- **MultiZone Sweep Sizing Behavior**: If set to **Use Size Function**, then any applied sizing controls (curvature and proximity refinement, and/or local sizing) are evaluated in all directions of the selected bodies. If set to **Ignore Size Function**, curvature and proximity refinement and/or local sizing along the sweep path are ignored and the spacing along the sweep path is determined either by the global sizes or local sizes that are explicitly set on edges along the sweep direction.

  You can also override the sizing by locally specifying hard edge sizes or using the **Sweep Size Behavior** control in the **MultiZone** method control to locally adjust the sizing for the bodies the control is scoped to. See MultiZone Method Control (p. 228) for more information.

Quality

- **Check Mesh Quality**: Sets the default quality behavior (p. 118) with respect to how the mesher responds to error and warning limits. Choices are:
  - **Default** - With this setting, the behavior changes as appropriate when you change the setting of **Physics Preference**.
  - **Yes, Errors** - If the meshing algorithm cannot generate a mesh that passes all error limits, an error message is printed and meshing fails.
  - **Yes, Errors and Warnings** - If the meshing algorithm cannot generate a mesh that passes all error limits, an error message is printed and meshing fails. In addition, if the meshing algorithm cannot generate a mesh that passes all warning (target) limits, a warning message is printed.
  - **No** - Mesh quality checks are done at various stages of the meshing process (for example, after surface meshing prior to volume meshing). The **No** setting turns off most quality checks, but some minimal checking is still done. In addition, even with the **No** setting, the target quality metrics are still used to improve the mesh. The **No** setting is intended for troubleshooting (p. 535) and should be used with caution as it could lead to solver failures or incorrect solution results.

- **Mechanical Error Limit**: Sets the default error limit (p. 118) when **Physics Preference** is set to **Mechanical**. Choices are **Standard Mechanical** and **Aggressive Mechanical**.

- **Target Quality (0 = Program Default)**: Sets the default target element quality (p. 130). When you modify this value, the value you enter becomes the new default for the Target Quality (p. 121) in the Details View. You can enter 0 on the **Options** panel to revert to the program default.

- **Target Skewness (0 = Program Default)**: Sets the default target skewness (p. 140). When you modify this value, the value you enter becomes the new default for the Target Skewness (p. 122) in the Details View. You can enter 0 on the **Options** panel to revert to the program default. For a tetrahedral mesh, you should not set **Target Skewness** to a value < 0.8.

- **Target Jacobian Ratio (Corner Nodes) (0 = Program Default)**: Sets the default target Jacobian ratio (p. 132). When you modify this value, the value you enter becomes the new default Target Jacobian

Ratio (Corner Nodes) (p. 122) in the Details View. You can enter 0 on the **Options** panel to revert to the program default.

Inflation

Inflation-related options that can be set on the **Options** dialog box include:

- **Maximum Height over Base** (p. 161)
- **Gap Factor** (p. 161)
- **Growth Rate Type** (p. 162)
- **Maximum Angle (Degrees)** (p. 162)
- **Fillet Ratio** (p. 163)
- **Use Post Smoothing** (p. 164)
- **Smoothing Iterations** (p. 164)

CGNS

- **File Format**: Sets the file format to be used for CGNS Export (p. 64) operations. There is no counterpart setting in the Details View. Choices are:
  - **ADF** (default) - Exports the mesh in ADF (Advanced Data Format).
  - **HDF5** - Exports the mesh in HDF5 (Hierarchical Data Format version 5).
- **CGNS Version**: Sets the CGNS library version to be used for CGNS Export (p. 64) operations. There is no counterpart setting in the Details View. Choices are **3.1** (the default), **3.0**, **2.5**, **2.4**, **2.3**, **2.2**, and **2.1**.
- **Export Unit**: Defines the unit of measurement for the mesh when exported to CGNS. The default is **Use Project Unit**, which means the mesh is not scaled. If you change this to another value (centimeter, millimeter, micrometer, inch, or foot), the mesh is scaled according to the export unit you select.

Ansys Fluent

- **Format of Input File (*.msh)**: Sets the file format to be used for Fluent Mesh Export (p. 43) operations. There is no counterpart setting in the Details View. Choices are **Binary** (the default) and **ASCII**.
- **Auto Zone Type Assignment**: When set to **On** (the default), zone types are automatically assigned, as described in Fluent Mesh Export (p. 43). When set to **Off**, assigns all boundary zones as the default WALL, enabling you to assign your own zone type assignments for Fluent Mesh Export (p. 43) operations.

Licensing Option

In the Mesh System, you can access **Meshing License** options from **File> Licensing**. The **License Options** allows you to set your license preferences. The available License Options are:

- **CFD PrepPost**
- **Ansys Mechanical Enterprise**
- **Ansys Mechanical Premium**
- **Ansys Mechanicl Pro**
- **Ansys Mechanical Enterprise PrepPost**
- **Ansys LS-DYNA**

In **2021 R2** release, Ansys Meshing does not support **Advanced Meshing**(advanced_meshing license feature). However, you can access the **Advanced Meshing** in the earlier versions of Ansys Meshing providing backward compatibility.

Meshing App sets the **CFD PrepPost** by default. You can also disable the license by selecting the license to be disabled and click the **Disable** button. You can access the legacy features by selecting the **Show Legacy Licenses** and the following license options are available:

- **CFD PrepPostPro**
- **Ansys Professional NLT**
- **Ansys Structural**
- **Ansys Professional NLS**
- **Ansys Multiphysics**
- **Ansys ExplicitSTR**
- **Ansys Emag**
- **Ansys AUTODYN PrepPost**
- **Ansys Mechanical/Emag**

All meshing features and options are available for the supported license features, but Fracture Meshing is available only for Ansys Enterprise License.

Meshing: Specialized Meshing

You can use the meshing features in Ansys Workbench to perform various types of specialized meshing. For more information, refer to:

Mesh Sweeping

This method of meshing complements the free mesher. If a body's topology is recognized as sweepable, the body can be meshed very efficiently with hexahedral and wedge elements using this technique. The number of nodes and elements for a swept body is usually much smaller than ones meshed with the free mesher. In addition, the time to create these elements is much smaller.

Workbench will automatically check to see if the body fulfills the topological requirements for sweeping. It will then choose two faces that are topologically on the opposite sides of the body. These faces are called the source and target faces. Workbench will mesh the source face with quadrilateral and triangular elements and then copy that mesh onto the target face. It then generates either hexahedral or wedge elements connecting the two faces and following the exterior topology of the body.

Note:

- This information applies to general sweeping. For requirements and usage information specific to thin model sweeping, see Thin Model Sweeping (p. 330).
- For descriptions of the sweep option settings, see Sweep Method Control (p. 223).

Requirements for General Sweeping

A body cannot be swept if any of these conditions exist:

- There is a completely contained internal void in the body.
- A source and target pair cannot be found. That is, the sweeper cannot find at least one path from a source surface to a target surface connected by edges or closed surfaces.
- If a **Sizing** control (p. 248) is used on a body with hard edge sizing and the source and target faces contain hard divisions which are not the same for each respective edge.

When sweeping it is only necessary to apply hard divisions to one leg of the sweep path. If the path has multiple edges, you should apply your controls to that path.

If the sweep path is shared by another body and that path lies on the body's source or target face then more hard divisions may be needed to constrain the sweeper.

When using Virtual Topology with sweeping, avoid creating virtual cells that result in a fully closed surface. Fully closed surfaces cause difficulties for the swept mesher and may result in poor meshes. When selecting adjacent faces for inclusion in a virtual cell, it is best to use Virtual Topology to merge some (but not all) of the faces. A good approach is to use Virtual Topology for the smaller faces, but omit any larger faces from the virtual cell.

To preview any bodies that can be swept meshed, click Mesh on the Tree Outline and right-click the mouse. Select **Show> Sweepable Bodies** from the context menu to display bodies that fulfill the requirements of a sweepable body. However, even if these requirements are met, the shape of the body may at times still result in poorly shaped elements. In these cases, the tetrahedron mesher is used to mesh the body.

The **Show Sweepable Bodies** feature only displays bodies that can be swept in terms of topology where the source and target are not adjacent on an axis. It cannot automatically determine axis-sweepable bodies. However, these bodies can be meshed if a Sweep mesh method is applied and source and target faces are defined. A sweepable body may not be Sweep meshed if the body geometry is not suitable.

Show Mappable Faces is a good tool to diagnose side faces. All side faces should be mappable, but if they are not found to be mappable, it indicates there may be a problem with the topology. For help in diagnosing problems when using the **Sweep** method, refer to the description of the **Edge** group in the Mechanical help. This toolbar provides access to features that are intended to improve your ability to distinguish edge and mesh connectivity.

Other Characteristics of General Sweeping

Other characteristics of sweeping include the following:

- The general sweeper ignores the **Num Cells Across Gap** (p. 110) setting, which is used to help define the proximity-based sizing (p. 102).

- Hard entities (p. 429) are not supported for the general sweeper.

- If the sweep method is applied to a body and a mapped Face Meshing (p. 265) control is defined for either the body's source or target face, the sweep mesher will fail if a mapped mesh cannot be obtained for the face. See Notes on Face Meshing Controls for Mapped Meshing (p. 274) for related information.

- The source and target faces do not have to be flat or parallel.

- If the topology of the source and target face is the same, the sweeping operation will often succeed even if the shape of the source face is different from the shape of the target face. However, drastically different shapes can cause element shape failures.

- Sweeping does not require your model to have a constant cross section. However, the best results are obtained for constant or linearly varying cross sections.

- For swept meshes with inflation (p. 414) and match control (p. 280), inflation is performed ahead of the match mesh and sweeping. This can affect the sizings on the match controls, which can in turn lead to meshing failure. Therefore, when using both match controls and inflation with sweeping, it might improve meshing robustness if you assign hard edge sizings (p. 252) to the high and low edges of the source face for the sweep.

Figure 122: Axis Sweep Representation

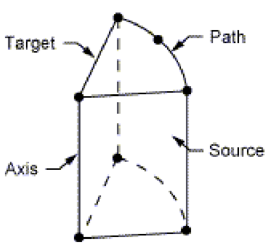


Figure 123: Edge Only Sweep Path

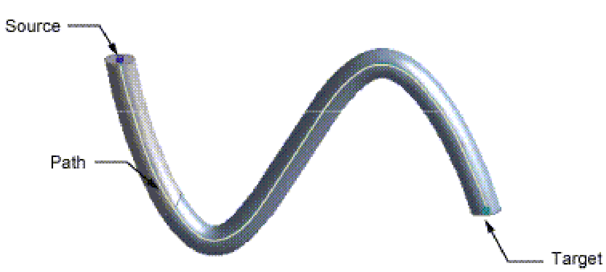


Figure 124: Edge Plus Closed Surface Sweep Path

Rules Followed By the General Sweeper

In deciding which area should be designated as the source area for general sweeping, the program uses the following rules, in the order as listed below. The sweeper will check all the rules until it finds a rule to use. Once a higher order rule is used, all the lower rules will not be considered. For example, if none of the first five rules apply, it will check against rule 6. In this case, if a face is a plane (flat) and the other face is not a plane (not flat), the flat face will be picked as the source and the test will be terminated.

1. **Manually set control - specify both source and target:** The source and target for sweeping will be exactly as you specify. This is the fastest way of meshing. It will eliminate searching for a possible source and target. For axis-sweeping, this method *must* be used. If a face is a source of another body and is not picked as a source of the current body, the aforementioned face will be used as a source.

2. **Manually set control - specify source:** Once the user specifies a source area, the program will try to find the target suitable to the source. The source will be exactly as specified.

3. **Face Mesh control:** The program finds the face with a mapped **Face Mesh** control applied to it.

4. **Number of loops:** The face with the largest number of loops will be picked as source face.

5. **Number of lines:** The face with the largest number of lines will be picked as source face.

6. **Flat face:** A flat face has higher priority for being a source face.

7. **Less sharing:** In most cases, a face might be used by one or a maximum of two bodies. If every one is flat (plane), the one used by the least number of bodies (that is, used by just one body) will be picked as source face.

8. **Larger area:** The largest area will be picked as the source.

Topological Requirements of the General Sweeper

The general sweeper must have at least one path between the source face and target face. The side faces of the sweep do not need to be singular but they must all be submappable and have single loops. The source face cannot be a closed analytic such as a full cylinder, torus or sphere. However, partial analytics are acceptable as source and target faces.

Note:

Creo Parametric creates unique topological models that no other CAD system creates. In all other CAD systems, non-periodic faces can have only one exterior topological loop. On the other hand, models in Creo Parametric can have non-periodic faces with multiple exterior loops. This type of topology does not pose a problem for the free meshers in the Meshing application. However, it does pose a problem for the general sweeper. As noted above, side faces of the sweep must have single loops. They cannot have multiple exterior loops because if they do, a single path from the source to the target cannot be determined.

Importing the model into the DesignModeler application breaks the face with multiple exterior loops into multiple faces with single loops because the DesignModeler kernel does not support the Creo Parametric topology. Exporting the model from Creo Parametric to IGES or STEP format will also resolve this issue.

Figure 125: Example (a) Showing Invalid Closed Cylindrical Face as Source Face

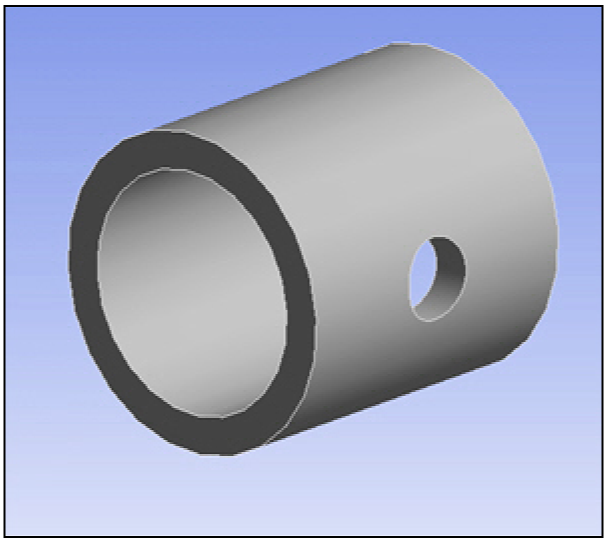
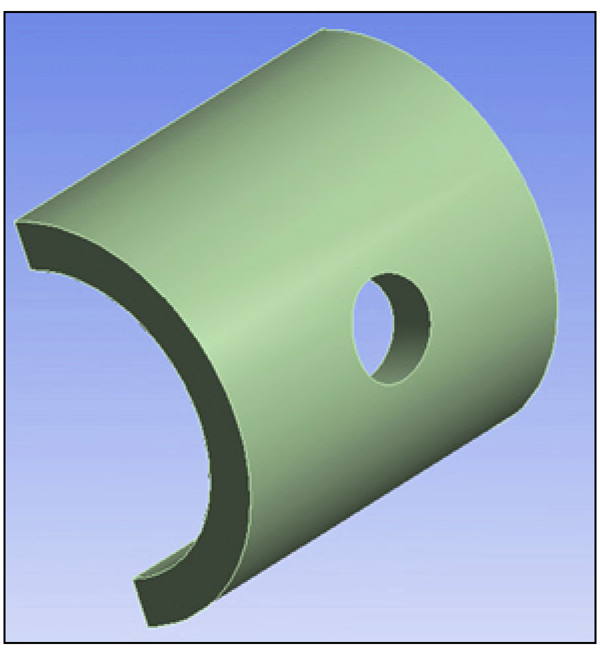


Figure 126: Example (b) Valid Open Cylindrical Face as Source Face



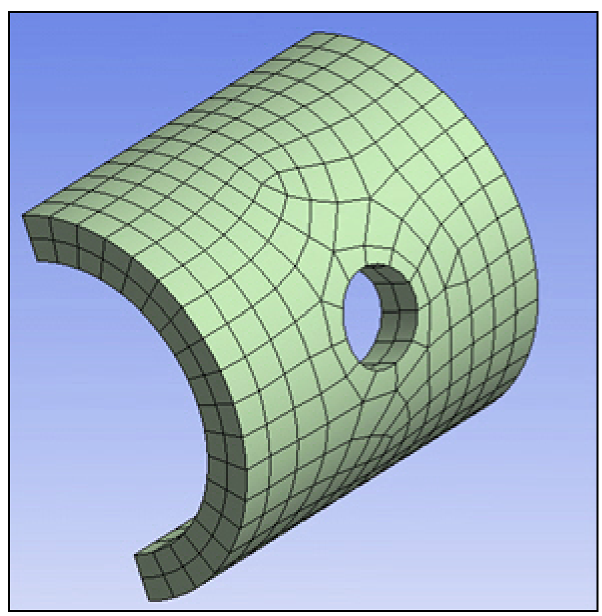
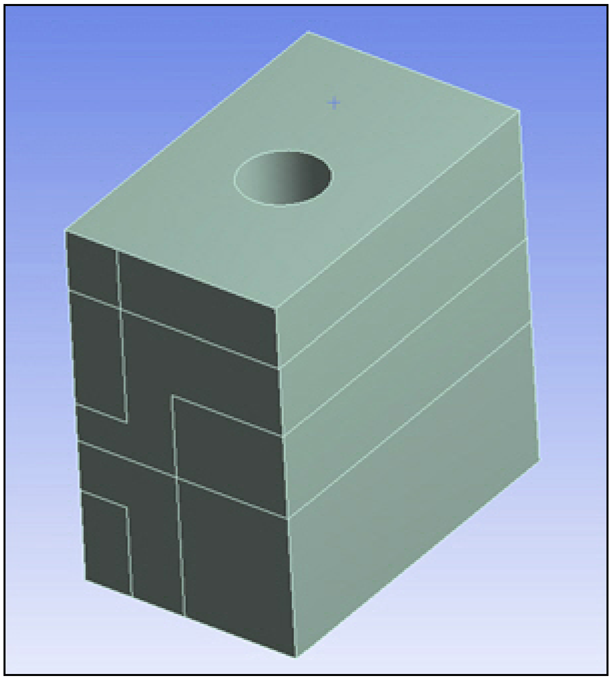
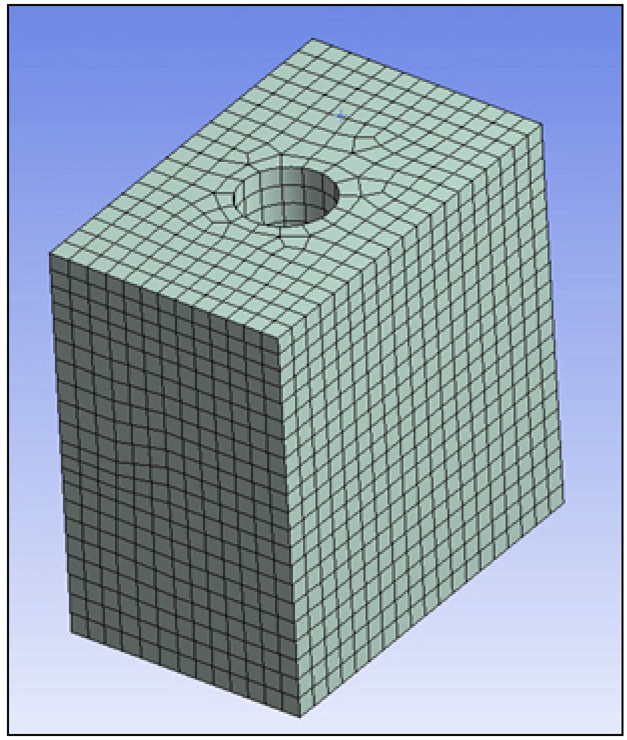


Figure 127: Example (c) Multiple Connected Side Faces

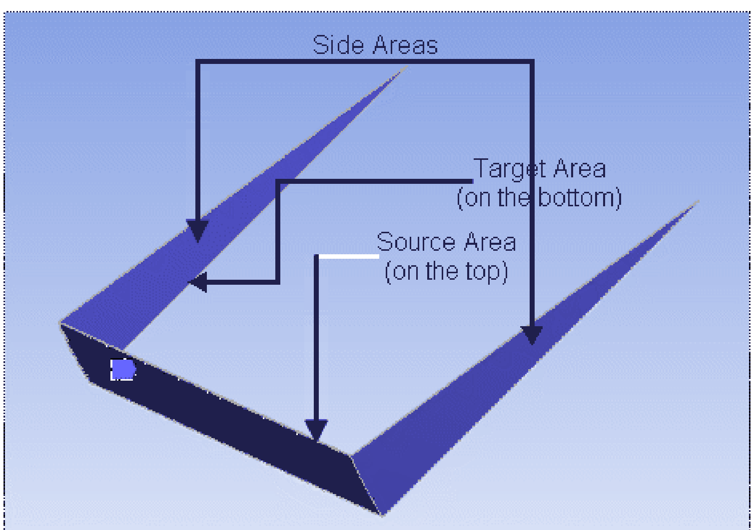




Using General Sweep to Mesh a Narrow Channel Body

The series of images below illustrates the use of general sweep, along with mapped face meshing (p. 265) and hard edge sizing (p. 252) controls, to mesh a narrow channel body. Figure 128: Axial Sweep Model (p. 328) shows the source, target, and side areas of the axial sweep model used in this example.

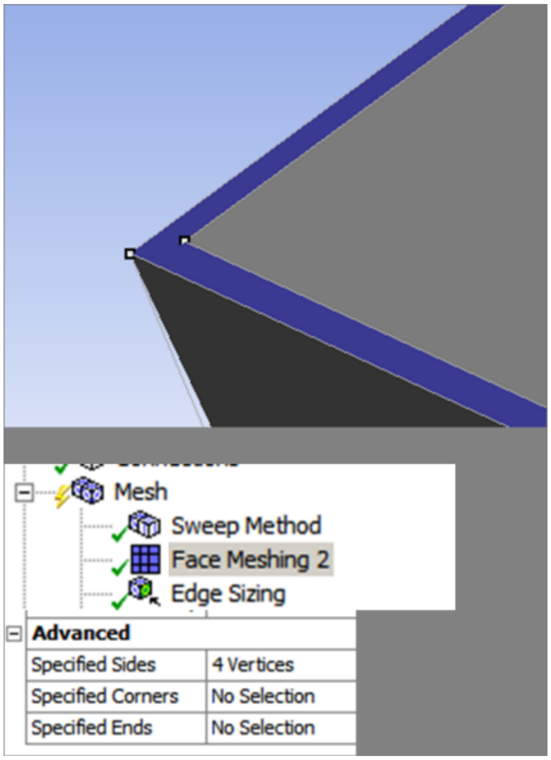
Figure 128: Axial Sweep Model



Because the source and target areas to be meshed are a narrow channel and you want them to be meshed with map mesh, it may present difficulties to the mesher. In Figure 129: Axial Sweep Model:

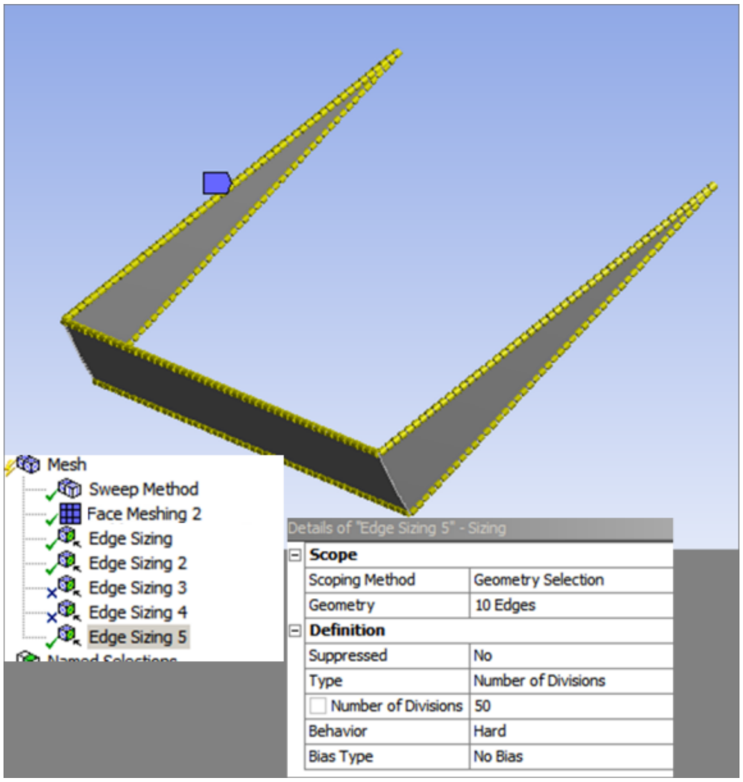
Face Meshing Control (p. 329), a mapped **Face Meshing** control is defined on the source face. Four vertices (two on each side area) have been selected for the **Specified Sides** control.

Figure 129: Axial Sweep Model: Face Meshing Control



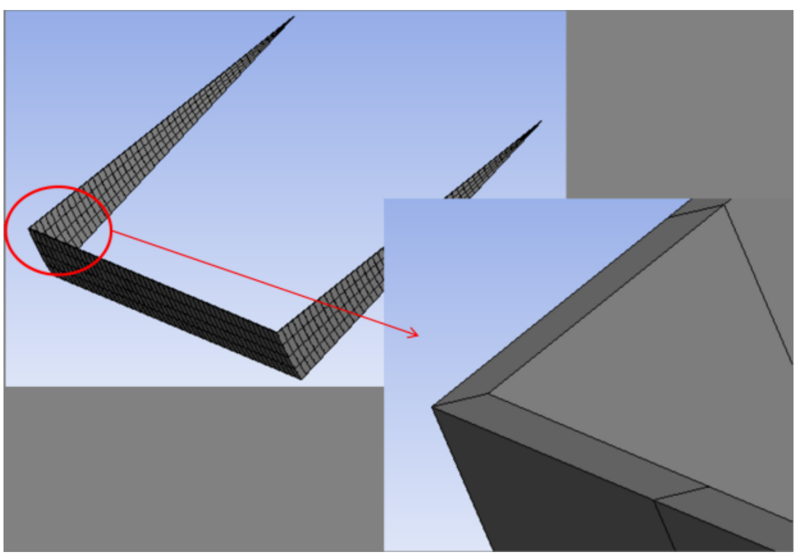
In cases similar to this example, the key to obtaining a successful mesh is the definition of a hard edge sizing control to make the two paired parallel edges. As shown in Figure 130: Axial Sweep Model: Hard Edge Sizing Control (p. 330), set the **Type** to **Number of Divisions** and enter a value in the **Number of Divisions** field (in this case, 50). The **Hard** option ensures the number of divisions are the same on the pair of edges.

Figure 130: Axial Sweep Model: Hard Edge Sizing Control



Finally, Figure 131: Axial Sweep Model: Meshed (p. 330) shows the mesh obtained using the settings described above.

Figure 131: Axial Sweep Model: Meshed



Thin Model Sweeping

Similar to the behavior of the general sweeper (p. 323), the thin model sweeper creates a structured hexahedral/wedge mesh, but for a thin model. It meshes one side of the thin solid (the source), and then sweeps the mesh to the other side (the target). Unlike the general sweeper, the thin model sweeper does not require a topological one-to-one match of source to target; the model may have multiple source and/or target surfaces. (Refer to Topological Requirements of the Thin Model Sweeper (p. 332) for examples.) In addition, the thin model sweeper can perform some edge defeaturing and allowing it to mesh models that have reasonably small features.

Requirements and usage information specific to the thin model sweeper include the following:

- The model must be thin—if the model is too thick, the thin model sweeper algorithm may fail.
- The source(s) and target(s) cannot touch each other.
- The model must have an obvious "side" that is perpendicular to the source and target; all of the side areas must connect directly from source to target.
- Mesh controls defined on the target may not be respected.
- Multibody parts are supported.
- For multibody parts, only one division through the thickness is possible. For single body parts, you can define multiple elements through the thickness using the **Sweep Num Divs** control in the Details View of the **Sweep Method**. (See steps below.)
- The thin model sweeper ignores the **Num Cells Across Gap** (p. 110) setting, which is used to help define the proximity-based sizing (p. 102). Using the proximity-based sizing in combination with the thin model sweeper may lead to an unnecessarily long computation time.
- If two bodies intersect to make a "T" connection, the thin model sweeper does not require that a mapped mesh control be defined at the junction of the two bodies.
- The **Preview Source and Target Mesh** and **Preview Surface Mesh** features do not support the thin model sweeper.
- See Notes on Face Meshing Controls for Mapped Meshing (p. 274) for information about using mapped Face Meshing controls with the thin model sweeper.

Considerations for Selecting Source Faces for the Thin Model Sweeper

The thin model sweeper meshes one side of a thin solid (the source), and then sweeps the mesh to the other side (the target). You can control which side the mesher uses as the source by selecting source faces manually. (To do so, set the **Src/Trg Selection** control to **Manual Thin** as described below.)

For most geometries, you can select just 1 of the faces in the complete set of faces that you want to be used as the source set, and the mesher will properly identify the other faces that are a part of that source set. However, for more complicated models (such as those containing multibody parts), you need to select all source faces in the source set in order for the mesher to be successful in finding the complete set of source faces.

A general rule of thumb is if you can select a single face and then extend the selection to its limits, the mesher can also identify the proper complete set of source faces. (For details about extending selections, refer to the description of the Extend Selection command in the Mechanical help.) If the geometry contains sharp angles that make the limit extension selection difficult, it will also be difficult for the mesher to use a single face for the source face definition, and you should select the complete set of source faces.

Topological Requirements of the Thin Model Sweeper

If a thin model mesh fails, turn on the **Edge Coloring > By Connection** option to see if the edge connectivity is unusual. In some cases, the geometry connectivity may not be as expected, and this may create problems during meshing. These problems can be fixed in the DesignModeler application, the CAD package, or possibly through the use of virtual topologies (p. 501)

The thin model sweeper supports M source faces to N target faces, where M and N can be any positive whole numbers. Between source faces and target faces, there must be "side faces." The angles between side faces and either source faces or target faces must be sharp enough that the faces are NOT considered to be smoothly connected. Therefore, a knife with a thin blade would not be appropriate for thin model sweeping because the cutting edge (blade) does not form a "side face." During the thin model sweeping meshing process, the features (vertices, edges, and faces) on the target may not be preserved and therefore, you should avoid applying boundary conditions to the target. The side faces must connect to both source and target. No edges or vertices are allowed on side faces. In this sense, no hard edges (p. 429) on side faces are allowed. Side edges must connect directly from source to target. You can use virtual topology to eliminate some features.

Figure 132: Example (a) N Source to 1 Target or 1 Target to N Source Topology

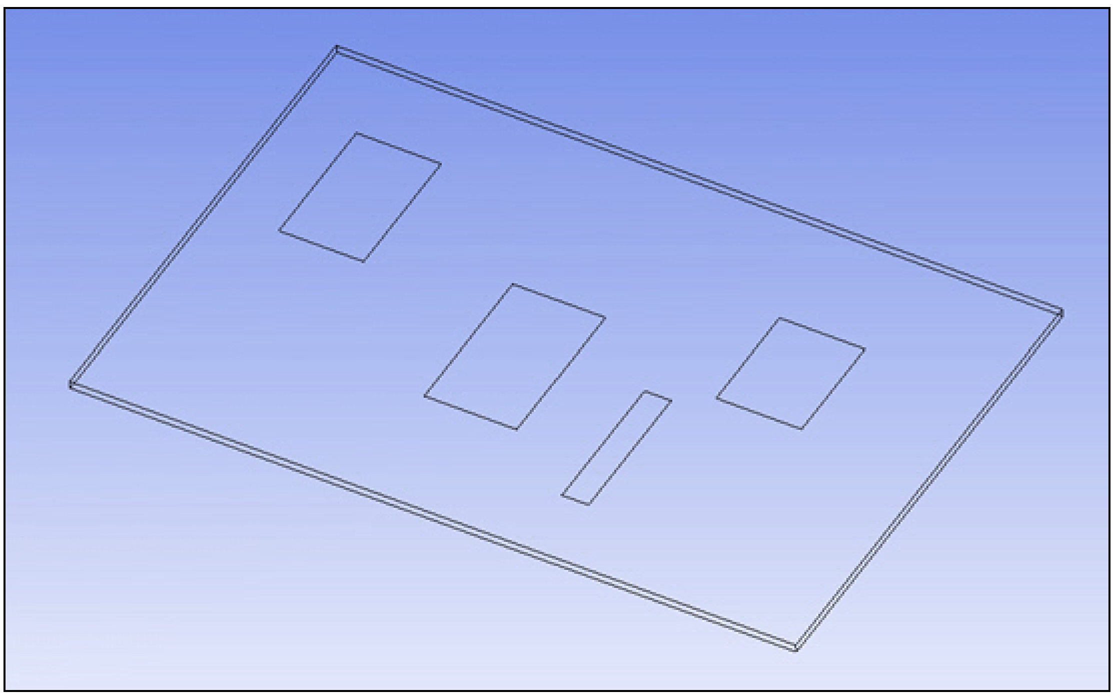


Figure 133: Example (b) N Source to N Target Topology

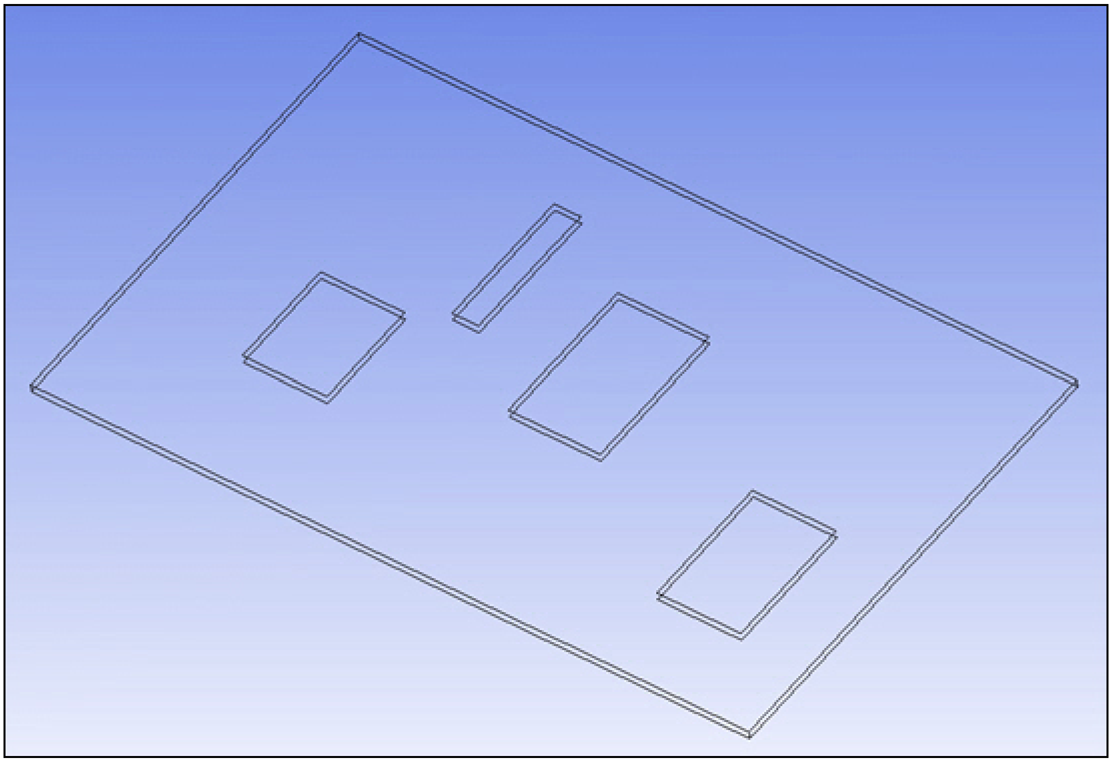


Figure 134: Example (c) 1 Source to N Target Mesh

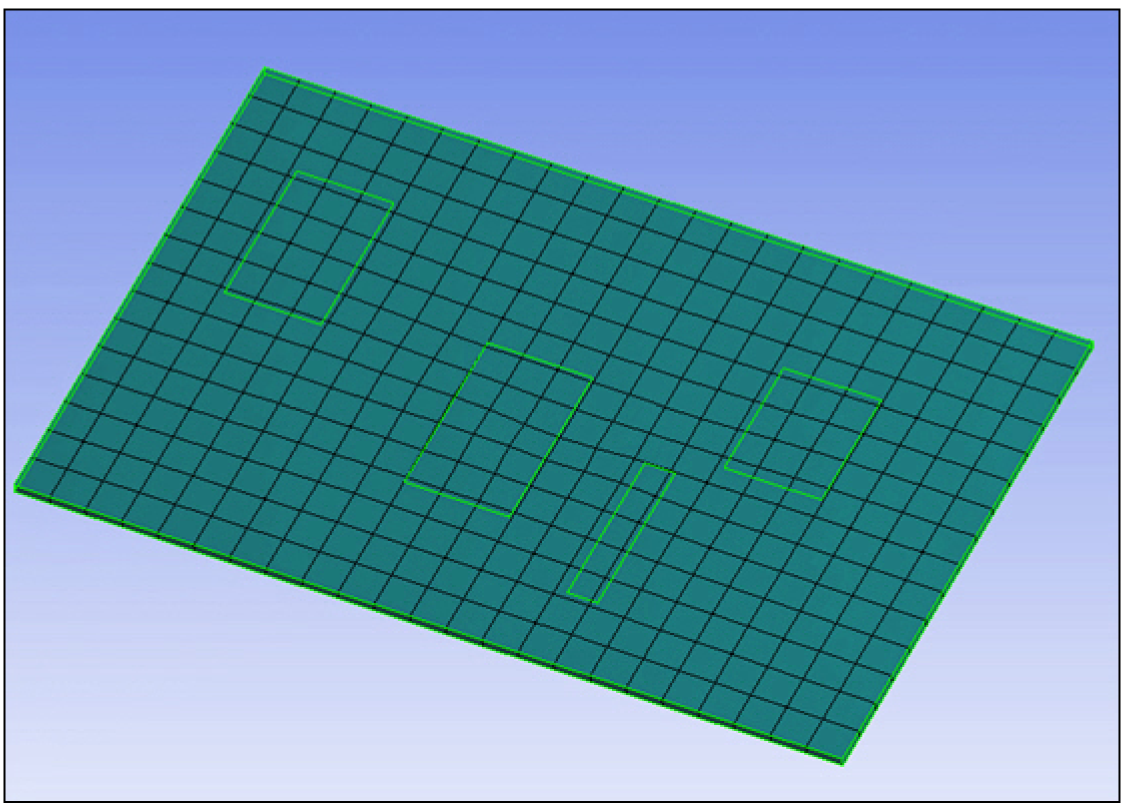


Figure 135: Example (d) N Source to 1 Target Mesh

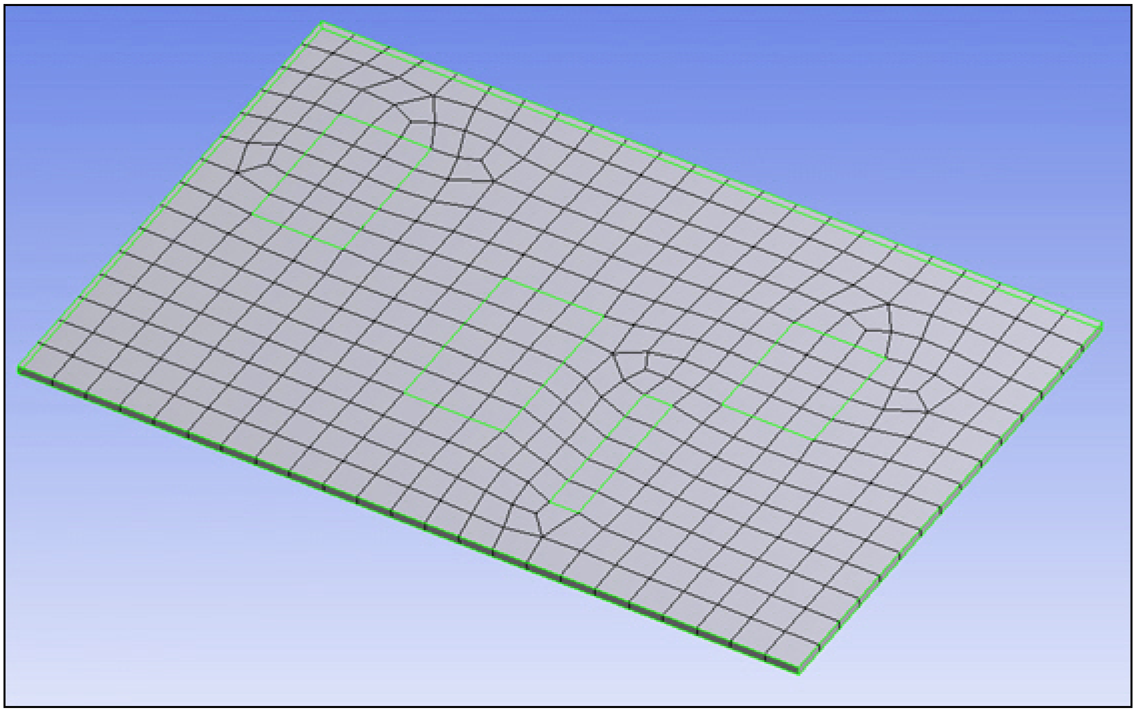
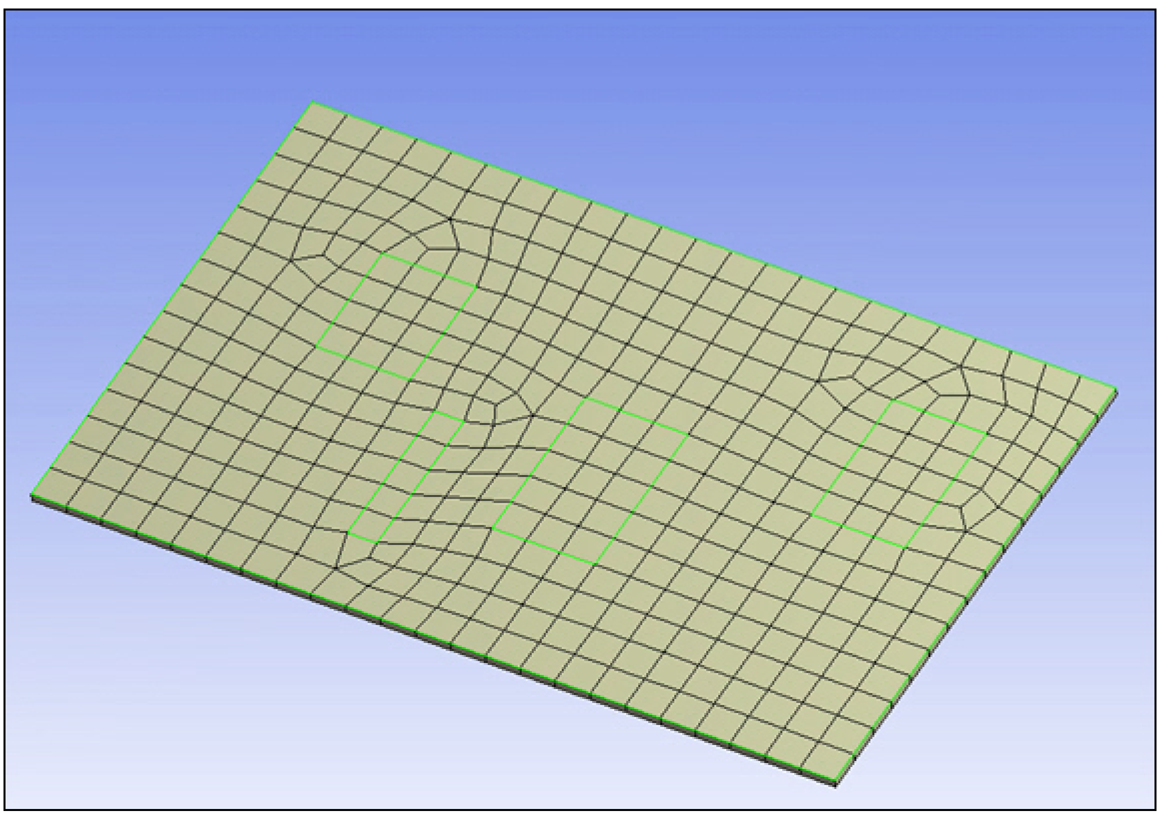
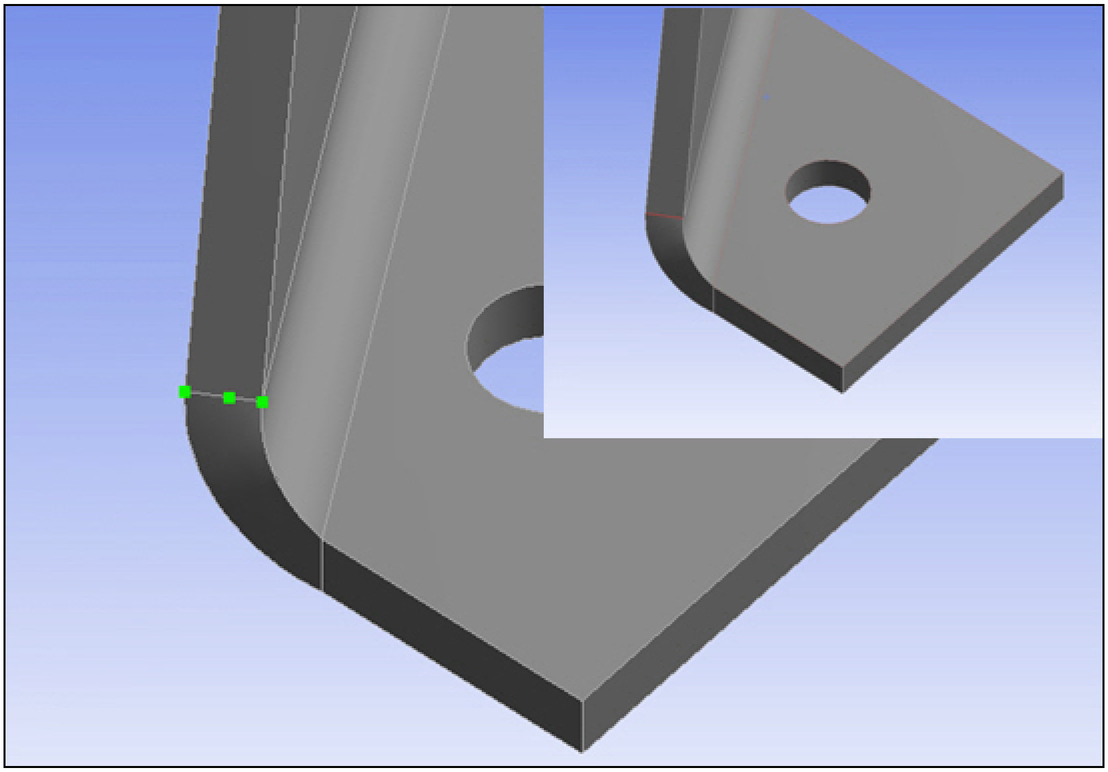


Figure 136: Example (e) N Source to N Target Mesh



Use Virtual Topology to create a single edge between source and target faces.

Figure 137: Using Virtual Topology to Create Single Edge Between Source/Target Faces

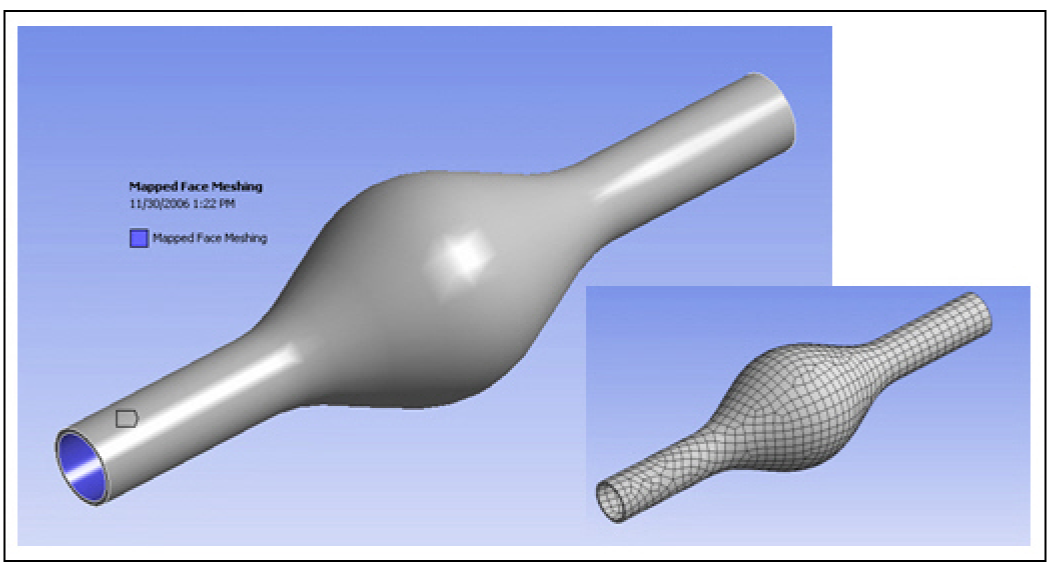


Mesh Controls and the Thin Model Sweeper

Mesh Controls applied on the target faces/edges are ignored. Only mesh controls applied to the source faces/edges are respected.

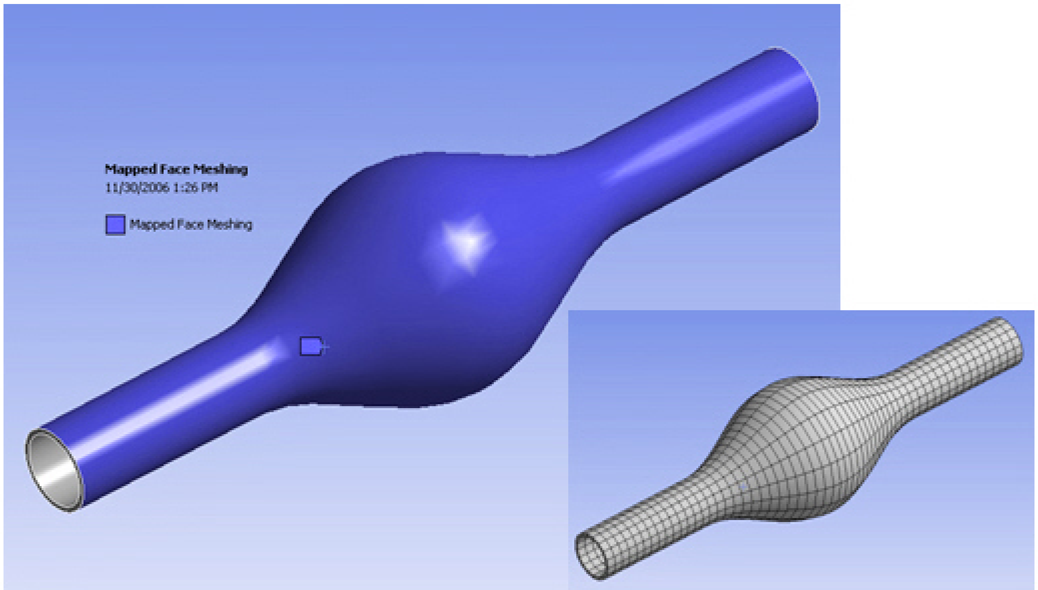
In example (a) below, the Mapped Face Control is ignored because it is applied to the target face.

Figure 138: Example (a) Mapped Face Control Applied to Target Is Ignored



In example (b) below, the Mapped Face Control is respected because it is applied to the source face.

Figure 139: Example (b) Mapped Face Control Applied to Source Is Respected



Thin Model Sweeping for Single Body Parts

This section provides the basic steps for using thin model sweeping to mesh a single body part.

To use the Thin Model Sweeper to mesh a single body part:

1. Click the **Mesh** object in the Tree and select **Insert> Method** from the context menu.
2. Scope the **Method** control to the thin body.
3. In **Details> Definition**, set **Method** to **Sweep**.
4. Set **Src/Trg Selection** to **Manual Thin** or **Automatic Thin**.

   Although **Automatic Thin** may work for simple cases, you may need to select **Manual Thin** depending on the complexity of the model.
5. If you selected **Manual Thin**, scope the source face(s), remembering the recommendations provided in Considerations for Selecting Source Faces for the Thin Model Sweeper (p. 331).
6. Enter additional sweep option settings, as desired, in the Details View. These may include **Free Face Mesh Type**, **Sweep Num Divs**, and **Element Option**. For descriptions of these options, see Sweep Method Control (p. 223).
7. Define other mesh controls, as desired.
8. Generate the mesh.

Figure 140: Thin Solid Sweeper Used to Mesh a Single Body Part (p. 337) shows a model of a timing cover that consists of a single body. The thin solid sweeper was used to mesh the body. To obtain this mesh, **Free Face Mesh Type** was set to **Quad/Tri**, **Sweep Num Divs** was set to **2**, and **Element Option** was set to **Solid Shell**.

Figure 140: Thin Solid Sweeper Used to Mesh a Single Body Part

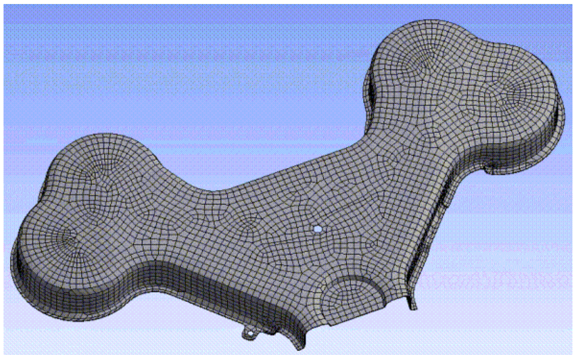
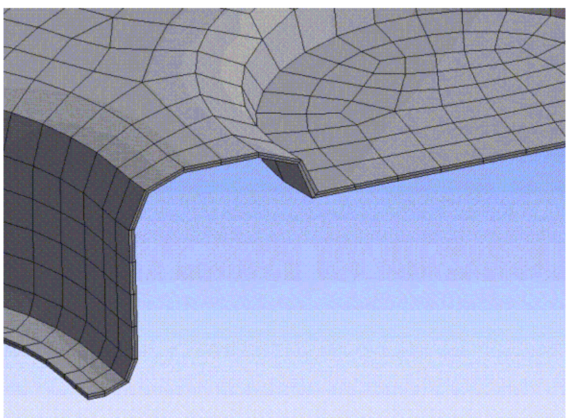


Figure 141: Thin Solid Sweeper Used to Mesh a Single Body Part: Detail (p. 337) shows detail of the timing cover. The **Sweep Num Divs** setting of **2** is apparent in this view.

Figure 141: Thin Solid Sweeper Used to Mesh a Single Body Part: Detail



Thin Model Sweeping for Multibody Parts

This section provides the basic steps for using thin model sweeping to mesh multibody parts. You can define thin sweep for each thin body in the multibody part.

To use the Thin Model Sweeper to mesh a multibody part:

1. Select a thin body in the **Geometry** window, right-click, and select **Insert> Method**.
2. Set **Method** to **Sweep**.
3. Set **Src/Trg Selection** to **Manual Thin** or **Automatic Thin**.

   Although **Automatic Thin** may work for simple cases, you may need to select **Manual Thin** depending on the complexity of the model.
4. If you selected **Manual Thin**, scope the source face(s) of the thin body, remembering the recommendations provided in Considerations for Selecting Source Faces for the Thin Model Sweeper (p. 331).

5. Enter additional sweep option settings for the thin body, as desired, in the Details View. These may include **Free Face Mesh Type** and **Element Option**. For descriptions of these options, see Sweep Method Control (p. 223).

6. If the part contains multiple thin bodies, repeat step 1 through step 5 for each.

7. If the part contains any thick sweepable bodies, repeat step 1 through step 5 for each, but set **Src/Trg Selection** to **Automatic**, **Manual Source**, or **Manual Source and Target** (depending on complexity of the model).

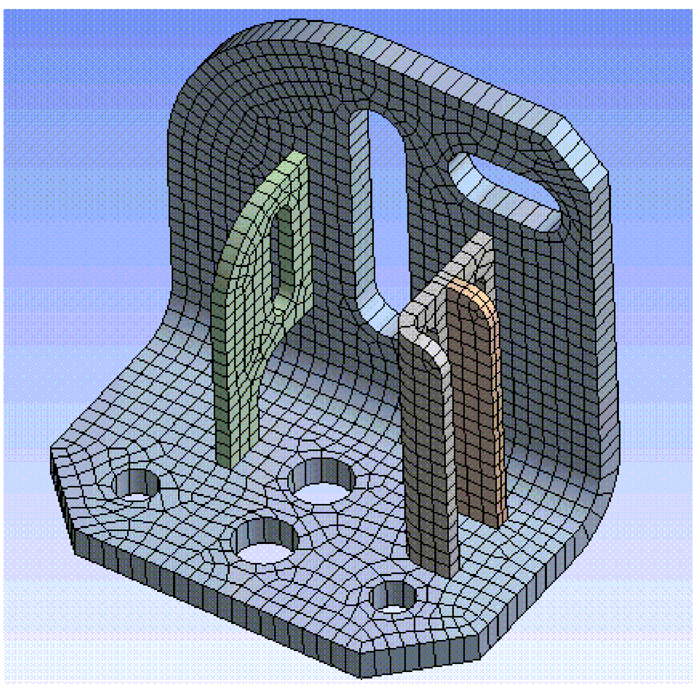
8. If the part contains any non-sweepable bodies, define mesh methods for each, if desired. If the mesh methods are left undefined, the Meshing application will determine the most appropriate methods to use for the non-sweepable bodies.

9. Define other mesh controls, as desired.

10. Generate the mesh.

Figure 142: Thin Solid Sweeper Used to Mesh a Multibody Part (p. 338) shows a model of a bracket that consists of four bodies. The thin solid sweeper was used to mesh the bodies. To obtain this mesh, **Free Face Mesh Type** was set to **Quad/Tri** and **Element Option** was set to **Solid**.

Figure 142: Thin Solid Sweeper Used to Mesh a Multibody Part

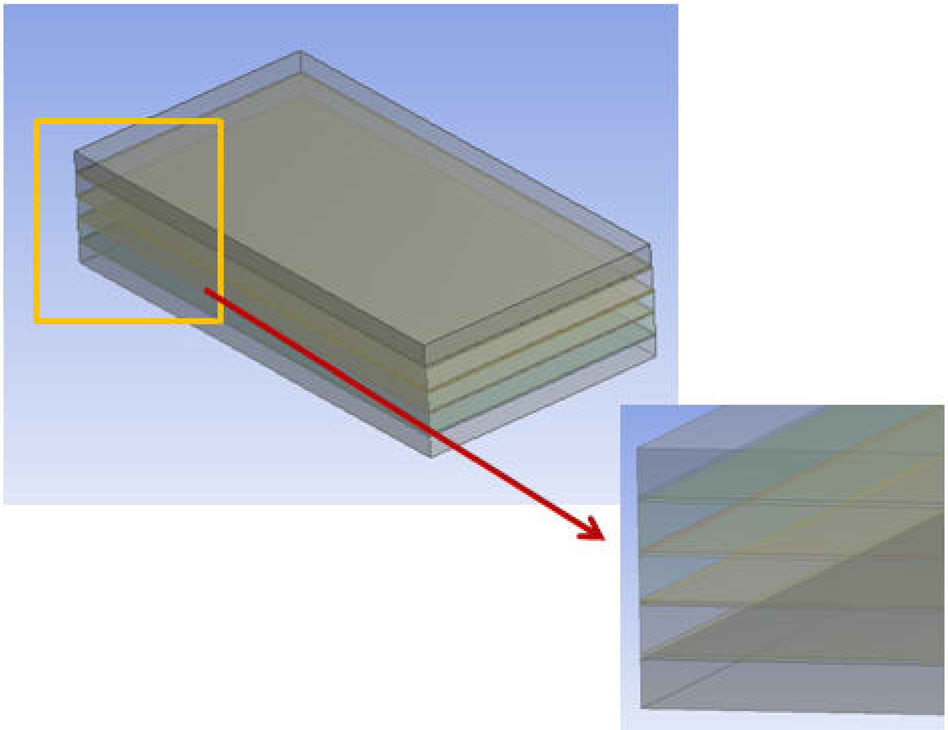


Additional Considerations for Using the Thin Model Sweeper

This section describes several models and scenarios to consider before using the thin model sweeper.

The first example involves a multibody part that models a laminated composite material, as shown below. Defining source faces for such models may be confusing.

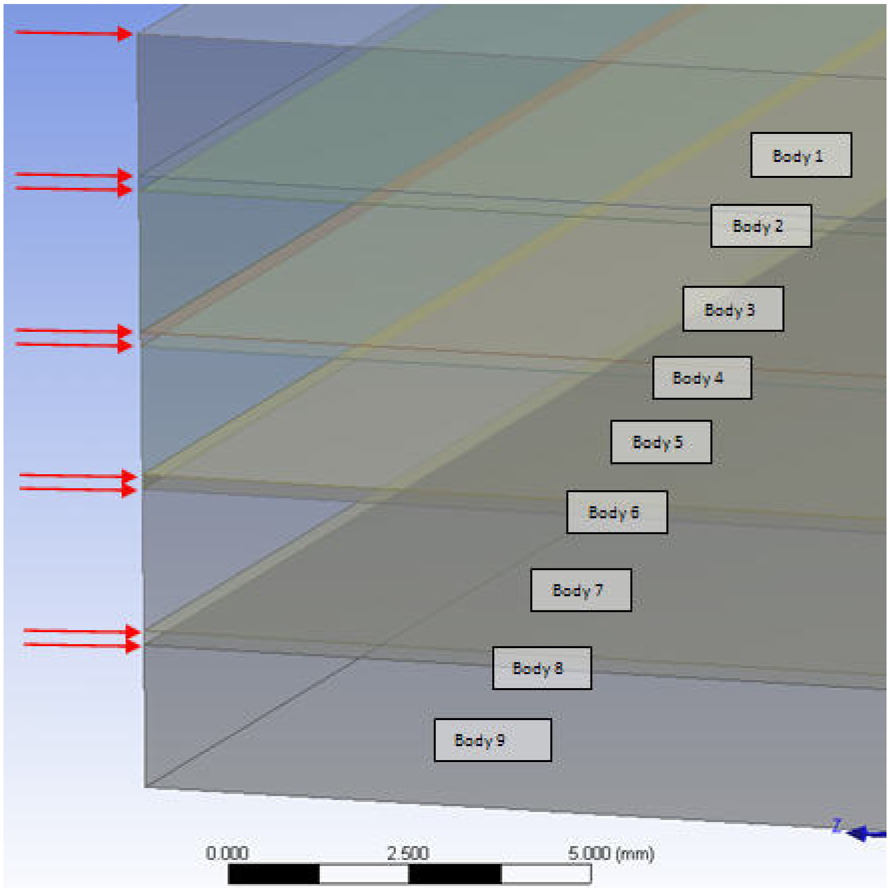
Figure 143: Thin Solid Sweeper and Laminated Composite Models



The part contains nine bodies. Assume that the **Manual Thin** option for **Src/Trg Selection** will be applied to each of them. With the **Manual Thin** option, source faces must be defined for each selected body (each body must have at least one face selected as its source face). There are different ways that you can select faces to meet this requirement, and it is logical to assume that defining nine source faces (one for each body) is one way that will work. However, in cases of laminated composites, we recommend that you specify *every other face* as a source face.

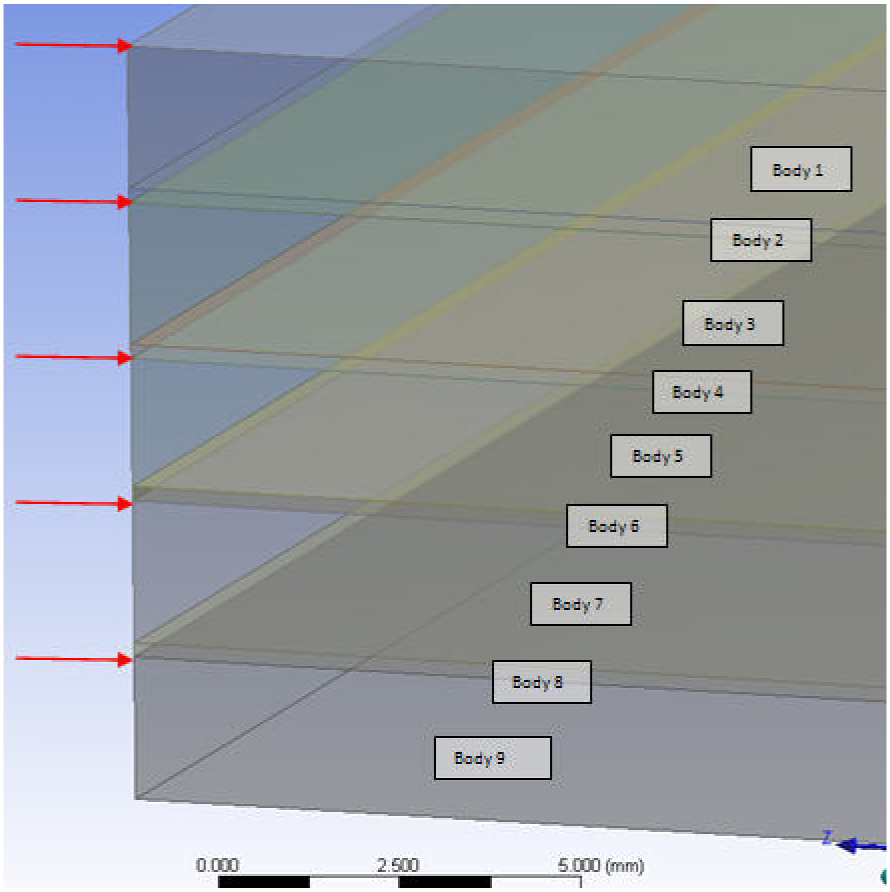
Consider the figure below, in which nine faces (indicated by arrows) are defined as source faces for the nine bodies. As illustrated by the figure, Body 1 has two faces defined as source faces, and the same is true for bodies 2 through 8. This source face definition causes ambiguity for the thin sweep mesher, which will have trouble determining a target face in bodies 1 through 8 and may fail.

Figure 144: Ambiguous Source Face Definition for Laminated Composite Model



Now look at the figure below. Here every other face has been selected as a source face, for a total of five. With this source face definition, each body still has one face selected to be its source face, so the requirement for **Manual Thin** source face selection has been met. With this source face definition, the thin sweep mesher will have no problem determining target faces for each of the bodies.

Figure 145: Recommended Source Face Definition for Laminated Composite Model



Note:

For cases in which the **Automatic Thin** option can be used, an alternative method to consider is to apply the **Manual Thin** option to only one body, define its source face, and apply the **Automatic Thin** option to the remaining bodies.

Before using thin solid sweeping, remember that the mesher meshes one side of faces and then sweeps the mesh to the second side of faces. Consider Figure 146: Thin Solid Sweeper Limitation (p. 342), which shows a model containing three plates. In the thin sweep operation, the edges that are common to two source faces are present on the source side. If the edges are different on the opposite side, the mesher will use the nodes from the source side in the mesh on the opposite side anyway. Thus, if there are features on the non-source side that are unique and need to be captured, you should not use the thin solid sweeping approach, as the mesher will ignore these features.

In Figure 146: Thin Solid Sweeper Limitation (p. 342), there is no valid way to mesh the middle plate with the thin solid sweep method, as there is an imprint coming from both the plate above and the plate below the middle plate, unless:

1. The plate is decomposed (sliced) to ensure all target face(s) have a corresponding source face.

2. The multibody part is broken into single parts (non-conformal mesh at common interfaces).

3. Some other mesh method is used. (In Figure 146: Thin Solid Sweeper Limitation (p. 342), a tet mesh method is used.)

4. The source and target faces have similar pairs, and the source faces are selected properly (described below).

Figure 146: Thin Solid Sweeper Limitation

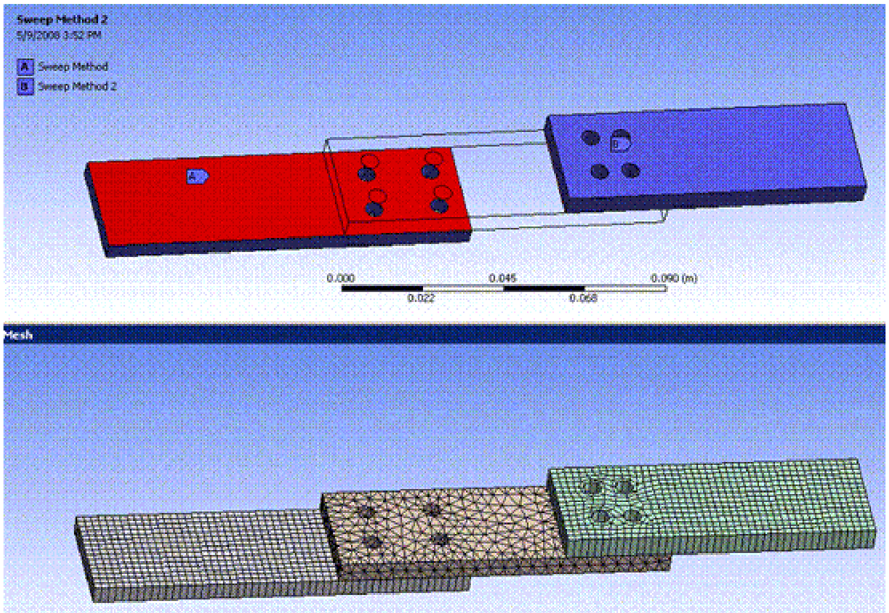
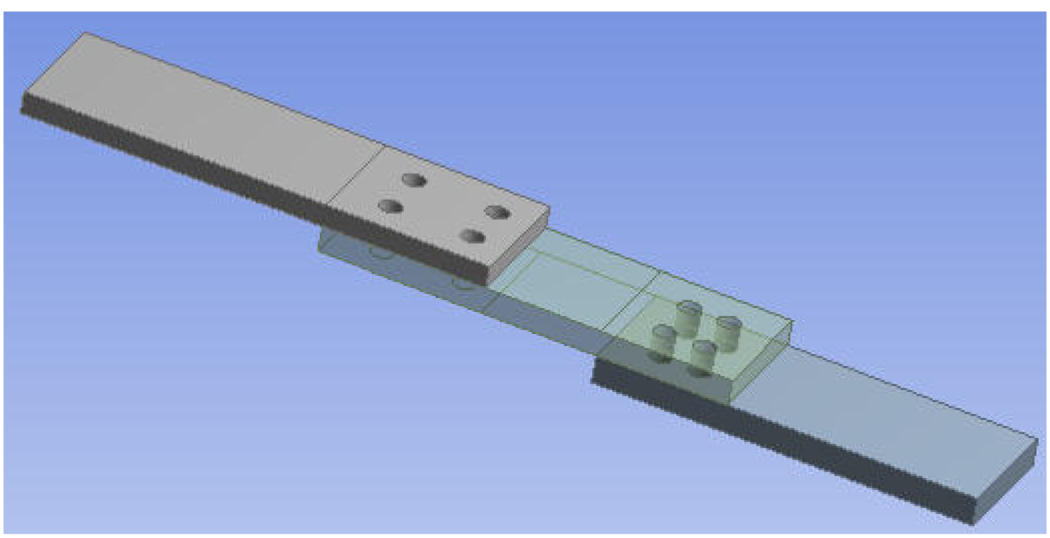


Figure 147: Adding Face Projections (Splits) in the DesignModeler Application (p. 342) illustrates an alternative approach to meshing the model above. In the DesignModeler application, the Projection feature allows face(s) to be split so that the source and target pairs will align. (For this model, the Edges on Face type of projection was used.)

Figure 147: Adding Face Projections (Splits) in the DesignModeler Application



With the addition of the face splits, the model can be meshed successfully with the thin solid sweep method by applying the **Manual Thin** option for **Src/Trg Selection** to all three bodies, and defining the top surface of each body as its source faces, as shown below. In this example, two faces are selected as source faces for the body on the left, three for the middle body, and two for the body on the right. Defining the source faces in this way ensures that everything is meshed from one side of the multibody part to the other.

Figure 148: Defining Source Faces when Face Splits Are Present

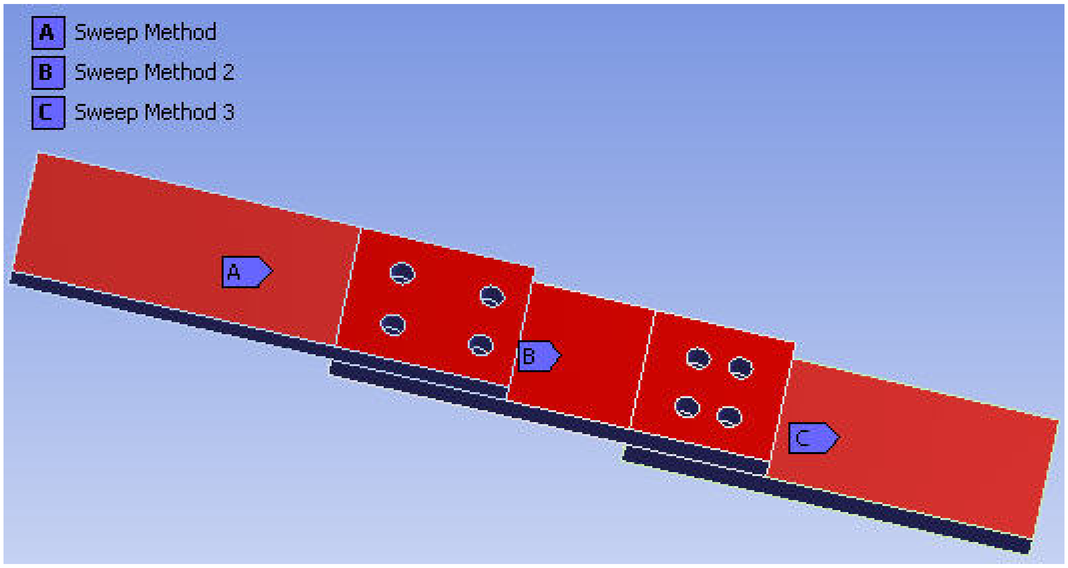
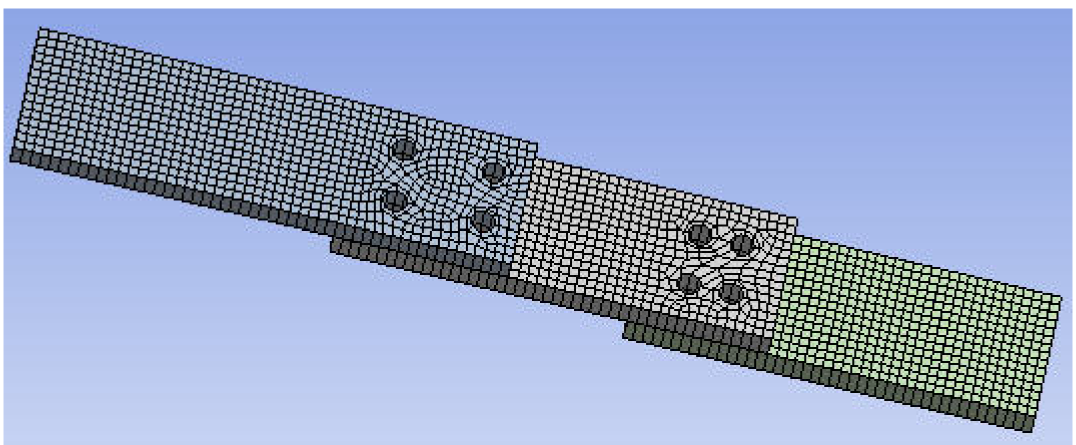


Figure 149: Three Plates Model Meshed with Thin Solid Sweeper (p. 343) shows the meshed model.

Figure 149: Three Plates Model Meshed with Thin Solid Sweeper



MultiZone Meshing

The **MultiZone** mesh method provides automatic decomposition of geometry into mapped (structured/sweepable) regions and free (unstructured) regions. It automatically generates a pure hexahedral mesh where possible and then fills the more difficult to capture regions with unstructured mesh. The **MultiZone** mesh method and the **Sweep** mesh method operate similarly; however, **MultiZone** has capabilities that make it more suitable for a class of problems for which the **Sweep** method would not work without extensive geometry decomposition.

MultiZone meshing topics include:

MultiZone Algorithms

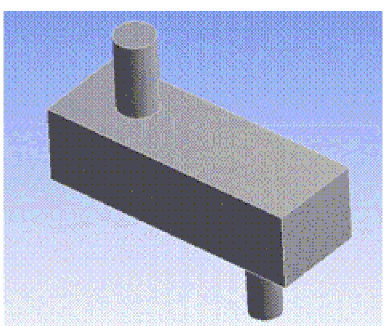
The **MultiZone** mesh method, which is based on the blocking approach used in Ansys ICEM CFD Hexa, starts by automatically blocking out surfaces. If the surface blocking can form a closed volume, then the volume may be filled automatically with a number of structured, swept, or unstructured blocks. The structured blocks can be filled with **Hexa**, **Hexa/Prism**, or **Prism** elements and the unstructured blocks can be filled with **Tetra**, **Hexa Dominant**, or **Hexa Core** elements depending on your settings, as described in MultiZone Method Control (p. 228).

The blocking algorithm and the meshing algorithm used to generate a **MultiZone** mesh are detailed below.

MultiZone Blocking Algorithm

The blocking algorithm used to generate a **MultiZone** mesh can be described as follows. The series of figures illustrates the process, assuming the geometry shown in Figure 150: Blocking Algorithm—Sample Geometry (p. 344).

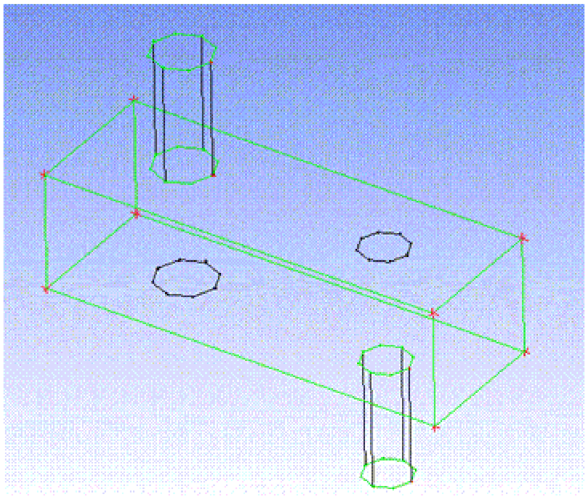
Figure 150: Blocking Algorithm—Sample Geometry



1. Creates automatic surface blocking (2D) by performing a surface analysis. In this step the algorithm:

   - Uses the **MultiZone Quad/Tri** mesh method to generate 2D blocking.
   - Uses mapped controls to try to force mapping. In general, surfaces with four sides are mapped (structured) and other surface patches are free (unstructured).
   - Uses source face selection to try to imprint faces into the source/target faces. (**MultiZone** treats all sources/targets as sources, as imprinting can occur from either side. For additional details, refer to MultiZone Source Face Selection Tips (p. 350).)

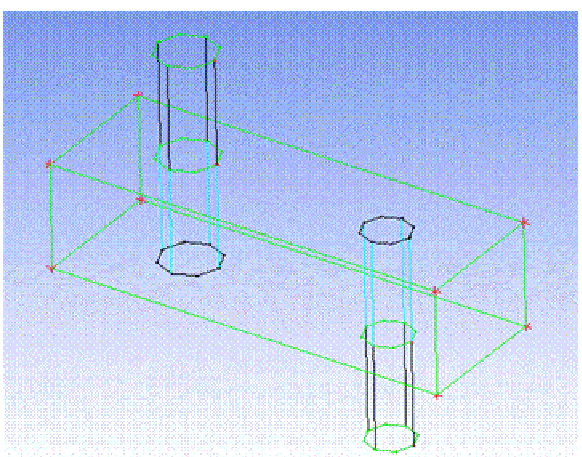
Figure 151: Blocking Algorithm—Step 1: 2D Blocking



2. Creates a **MultiZone** (3D) structure. In this step the algorithm:

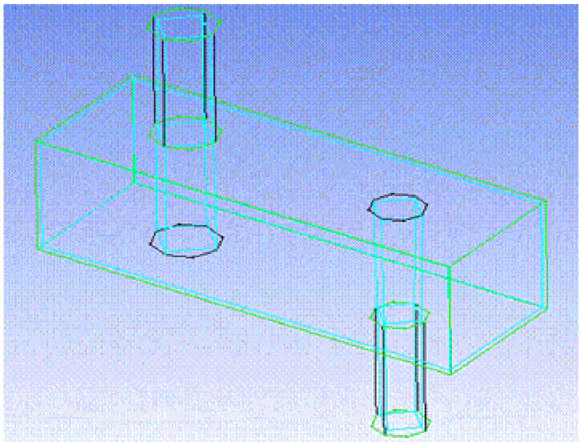
   - Processes surface blocks to match swept sections.
   - Connects surface blocks to create volume blocking (3D) using unique heuristic approaches to avoid some of the traditional limitations of sweeping algorithms.
   - Resolves the volume region with structured, swept, and unstructured blocks. Mapped faces become structured blocks and free faces become unstructured blocks.

Figure 152: Blocking Algorithm—Step 2: 3D Blocking



3. O-Grid creates boundary blocks automatically, and the algorithm extrudes O-Grid to create inflation.

Figure 153: Blocking Algorithm—Step 3: Inflation



Note:

Inflation can be performed on a combination of side and/or source faces.

MultiZone Meshing Algorithm

The meshing algorithm used to generate a **MultiZone** mesh can be described as follows:

- Assigns intervals by transferring sizing from the topology to blocking.
- Meshes structured faces (those that can be mapped meshed) using transfinite interpolation and unstructured (free) faces with all quadrilateral or a combination of quadrilateral/triangle elements.
- Interpolates, sweeps, and fills the volume mesh for the structured, swept, and unstructured blocks respectively. The structured blocks can be filled with **Hexa**, **Hexa/Prism**, or **Prism** elements and the unstructured blocks can be filled with **Tetra**, **Hexa Dominant**, or **Hexa Core** elements depending on your settings, as described in MultiZone Method Control (p. 228).

MultiZone for Sweepable Bodies

If using Use MultiZone for Sweepable bodies (p. 317), you can use the Show Sweepable Bodies (p. 494) setting to identify bodies that will be meshed using the **MultiZone** method. Although **MultiZone** can mesh other, more complex bodies, it does not attempt this unless you manually insert **MultiZone** mesh method controls.

Other notes about the behavior of **MultiZone** when meshed in this way:

1. When meshing using **Automatic** approach, **MultiZone** uses **Preserve Boundaries** = **All** (Note this is different than the default when scoping a body to a MultiZone mesh method) and no defeaturing is done.
2. If **MultiZone** fails to generate a valid mesh due to errors in the mesh quality or topology, the body will be meshed with **Free Mesh Type** = **Tetra**.

3. Bodies scoped to an **Automatic** Mesh Method will also be meshed in the same way.

Using MultiZone

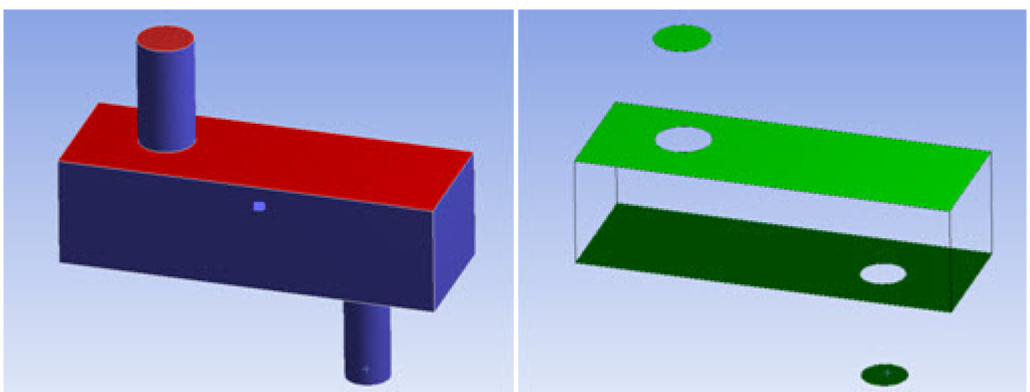
To help you to determine whether the **MultiZone** mesh method is the most appropriate method for your problem and to ensure blocking is constructed properly, you must consider the following characteristics of the problem:

- Sources
- Sweep paths (side faces)
- Intersections (levels)

To classify a **MultiZone** problem, consider which faces will be sources and in turn, which sources will cause imprinting. (Note that imprinting, which is described in MultiZone Source Face Imprinting Guidelines (p. 351), has its own classifications that further define the problem.)

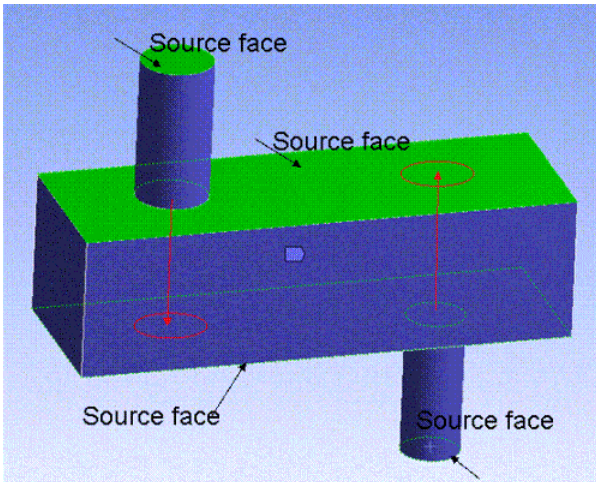
In Figure 154: Classifying the Problem: Sources (p. 347), four faces have been selected as sources.

Figure 154: Classifying the Problem: Sources



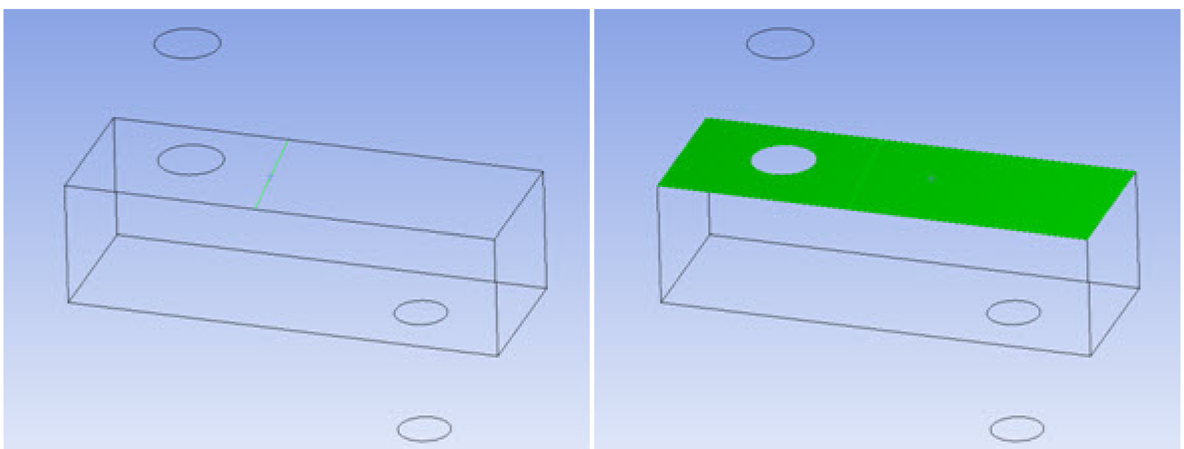
You must also consider the sweep path, or sides, of the mesh. This includes how many potential sweep directions exist in the problem and if there is more than one, how imprints should be handled. See Figure 155: Classifying the Problem: Handling of Paths and Imprints (p. 348).

Figure 155: Classifying the Problem: Handling of Paths and Imprints



A collective set of attached faces in a body is used for imprinting. For example, if one of the faces in this model has a split in it as shown in Figure 156: Collective Source Faces (p. 348), the number of source faces increases by one. However, the set of source faces remains the same.

Figure 156: Collective Source Faces

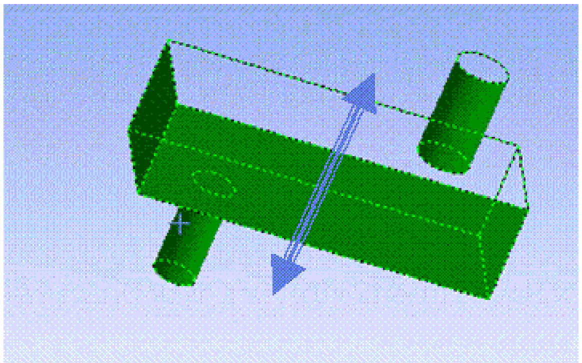


Thus, there are still four levels of source faces. However, now there are additional considerations necessary for imprinting, and handling of side faces.

All faces that are not sources are side faces, and they make up the path. In a simple box example, the box could be swept in any one of three directions, but it is the source selection that determines the path. If no source is selected, **MultiZone** determines the path arbitrarily.

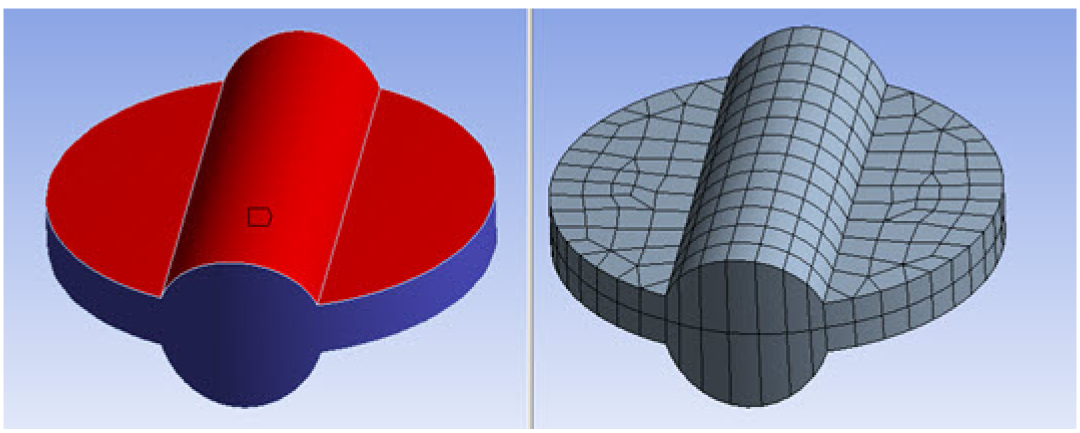
The model shown in Figure 157: Classifying the Problem: Sweep Path (p. 349) shows one clear sweep direction. The green faces are the side faces.

Figure 157: Classifying the Problem: Sweep Path



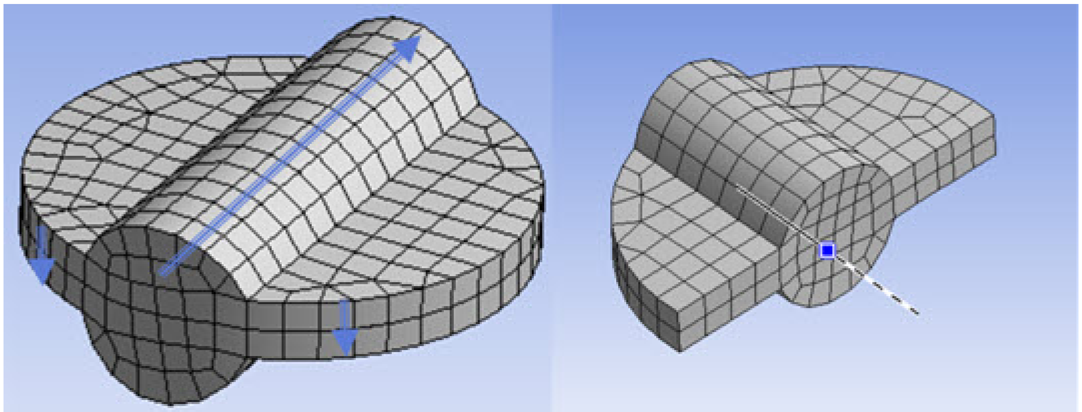
By default in **MultiZone**, the mesher is set to automatically determine the sweep path. Since 2D regions and 3D regions are decomposed automatically, this allows the mesher the freedom to choose paths that may not be possible without first subdividing the model. For example, in cases like the valve body shown below, with a traditional sweep meshing approach you would either need to split this body into three regions (center region with path along the axis, and two half cylinders with path top to bottom), or mesh the whole body from top to bottom.

Figure 158: Valve Body: Traditional Approach



With **MultiZone**, the mesher determines that certain portions of the cylindrical region face need to be mapped as a side face, and other portions of it that need to be considered as source faces, and therefore it can decompose the sweep paths automatically.

Figure 159: Valve Body: Automatic Source Faces with MultiZone



In general, you should allow **MultiZone** to define its own sweep path. However, if the path is obvious and there are clear source and side faces, or if **MultiZone** is not able to determine the sweep path on its own, you can guide the mesher by setting manual source faces and ensuring side faces are mappable.

For more information, refer to:

MultiZone Source Face Selection Tips

MultiZone Source Face Imprinting Guidelines

MultiZone Face Mappability Guidelines

Using Virtual Topology to Handle Fillets in MultiZone Problems

MultiZone Source Face Selection Tips

When you choose the **MultiZone** mesh method, the Details View expands to expose various settings (p. 228), one of which is **Src/Trg Selection**. **Src/Trg Selection**, which defines the source and target selection type for the **MultiZone** mesh method, can be set to either **Automatic** (the default) or **Manual Source**. Remember these points when selecting faces:

- Select both source and target faces. **MultiZone** internally decides which faces to take as sources and which to take as targets.
- To do imprinting of face loops, all swept section faces (that is, those that are perpendicular to the sweep direction) should be selected.
- Source faces will generally be meshed with a free mesh. Depending on the geometry, you can add mapped face meshing controls (p. 265) to these faces to get mapped meshing.
- At an interface in a multibody part (in which two bodies share the same face), there is not always a distinction of which body the face belongs to. When using **MultiZone** in such cases, setting the face to be inflated may affect both bodies because both bodies are meshed with **MultiZone** at the same time.
- To make source face selection easier, select **Annotation Preferences** from the Toolbar and then deselect **Body Scoping Annotations** in the **Annotation Preferences** option box to toggle the visibility of annotations in the **Geometry** window. For example, after scoping **MultiZone** to a body, the body will be displayed using a blue solid annotation. Turn off the body scoping annota-

tions; then select the source faces. For picking internal faces, the **Hide Faces** right-click option may help you to see inside a body. For example, you can select external faces in the **Geometry** window and then use the **Hide Faces** option to hide the selected faces (making it easier to select the internal faces).

Refer to Figure 106: Source Face Selection for MultiZone (p. 235) for more information.

MultiZone Source Face Imprinting Guidelines

Imprinting has its own classifications that further define the problem. Models often exhibit combinations of these different situations when trying to sort out the sweep paths between pairs of source face(s).

When selecting source faces, it is important to recognize that the pairs of source faces need to have proper matches (that is, each level must be resolved with other levels). If source faces cannot be paired, imprinting must be done by the software to create proper pairs. The software supports only certain types of imprinting, so it is important to understand these cases.

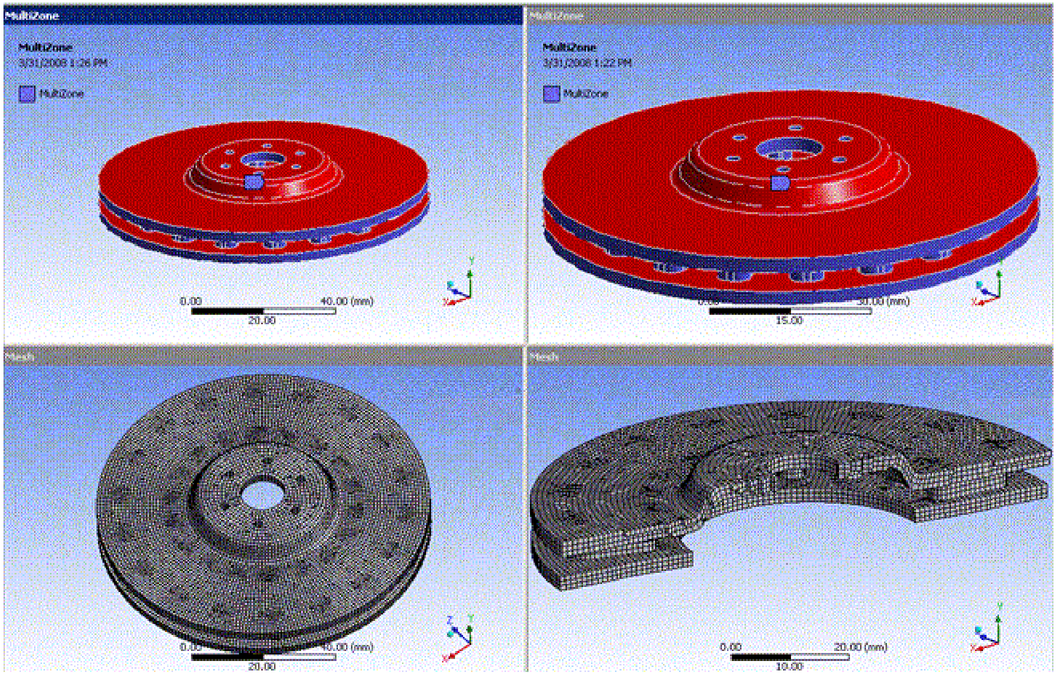
Imprinting classifications include:

It is also important to note that if you set the **Free Mesh Type** (p. 228) to anything other than **Not Allowed**, some of the imprinting will not be done, as this can lead to some problems in the creation of the free blocks. For example, on some models you could get a pure hex mesh if you set **Free Mesh Type** to **Not Allowed**, but not get a pure hex mesh if you set **Free Mesh Type** to some other value.

Internal Loops

Internal loops are internal regions that imprint to other faces, such as the fins in the rotor model shown below.

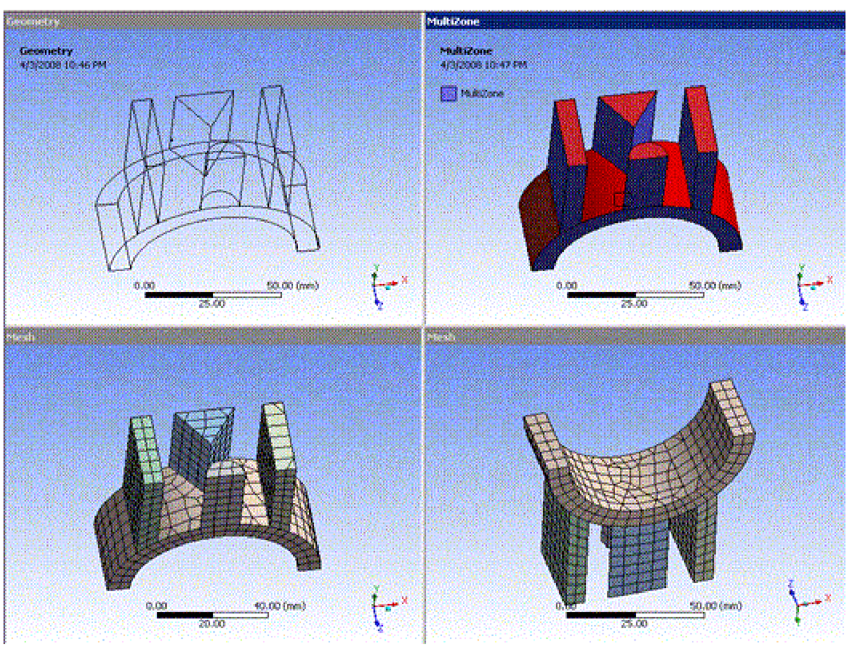
Figure 160: Source Imprinting Classifications: Internal Loops



Boundary Loops

Boundary loops are splits in a face that extend to the boundary, such as those shown in the model below.

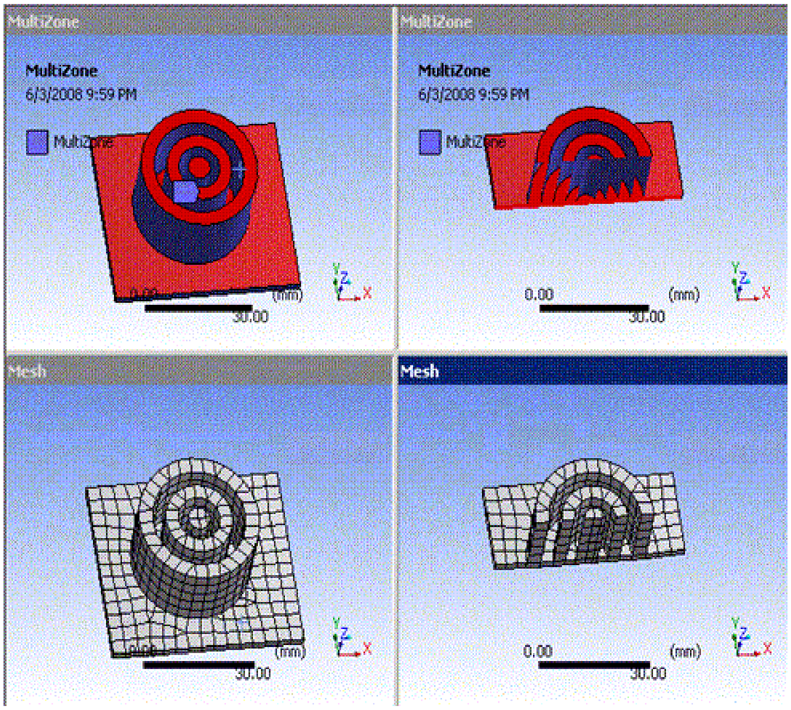
Figure 161: Source Imprinting Classifications: Boundary Loops



Multiple Internal Loops

Some models exhibit multiple internal loops, such as those in the model below.

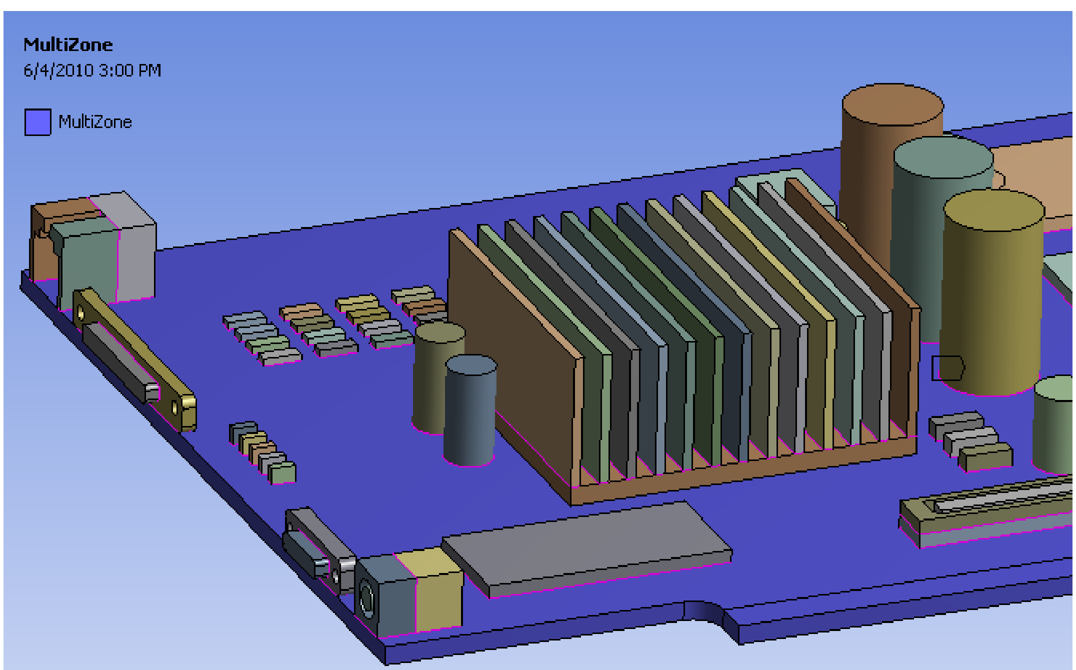
Figure 162: Source Imprinting Classifications: Multiple Internal Loops



Multiple Connected Internal Loops

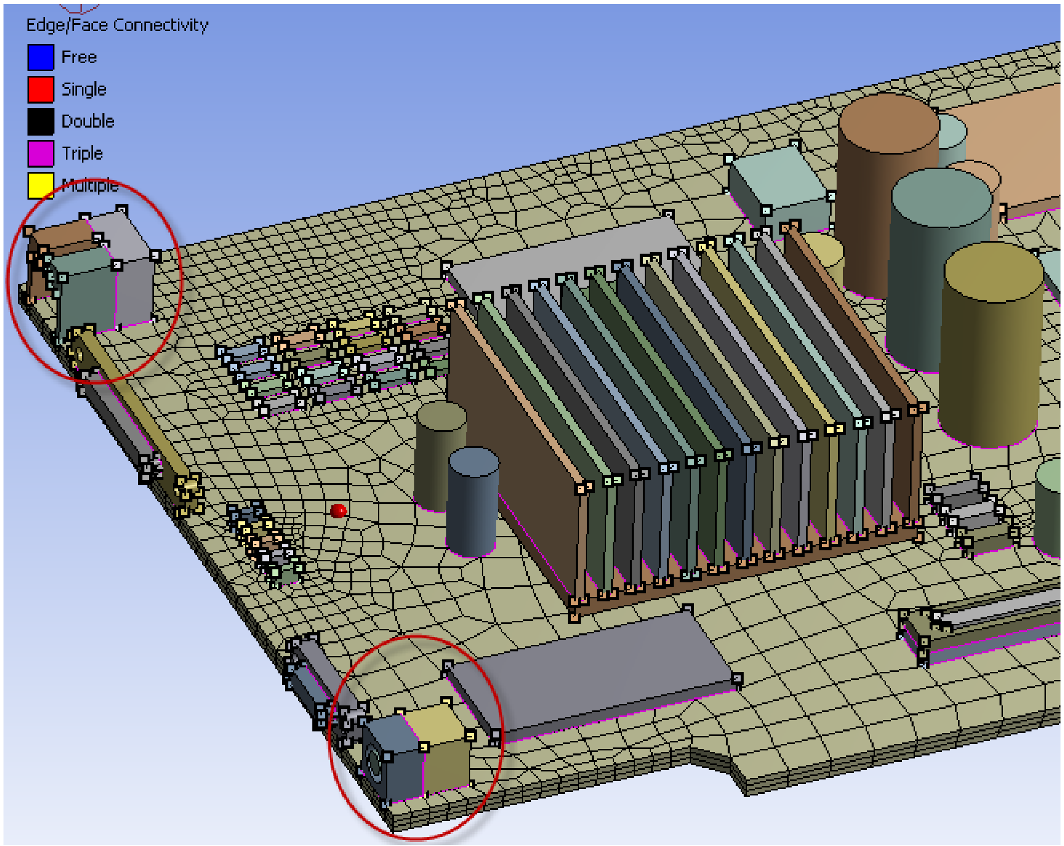
Some models exhibit multiple connected internal loops, such as those in the model of the circuit board below. The **MultiZone** method has been applied to the underlying board, which is highlighted in blue. The **Edge Coloring>By Connection** feature is on in this view.

Figure 163: Source Imprinting Classifications: Multiple Connected Internal Loops View 1



In the figure below, the body corresponding to the underlying board was selected and meshed. In this view, both the **Edge Coloring>By Connection** and **Show Vertices** features are on.

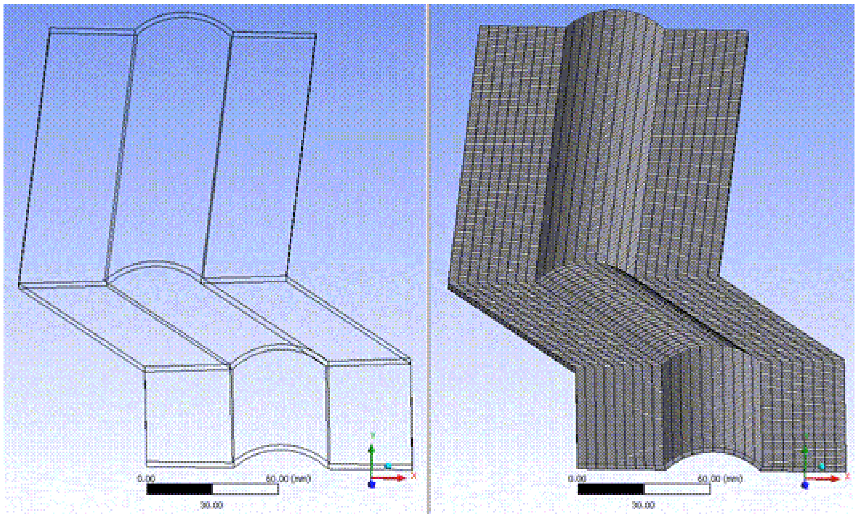
Figure 164: Source Imprinting Classifications: Multiple Connected Internal Loops View 2



Parallel Loops

Some models exhibit parallel loops, such as those in the model below. If the software can match all pairs these models should work; however, the angle between pairs can cause problems (due to difficulties finding the pairs), and if there are non-matched pairs they will cause problems. For non-matched pairs, virtual topologies may solve the problem. For these kinds of examples, using automatic source face selection is often more robust than manual source face selection.

Figure 165: Source Imprinting Classifications: Parallel Loops



Intersecting Loops

Two views of a model with intersecting loops are shown below. Notice that the sources on the top and bottom would intersect when being imprinted. These types of cases are not supported yet. You can use geometry decomposition as a workaround.

Figure 166: Source Imprinting Classifications: Intersecting Loops View 1

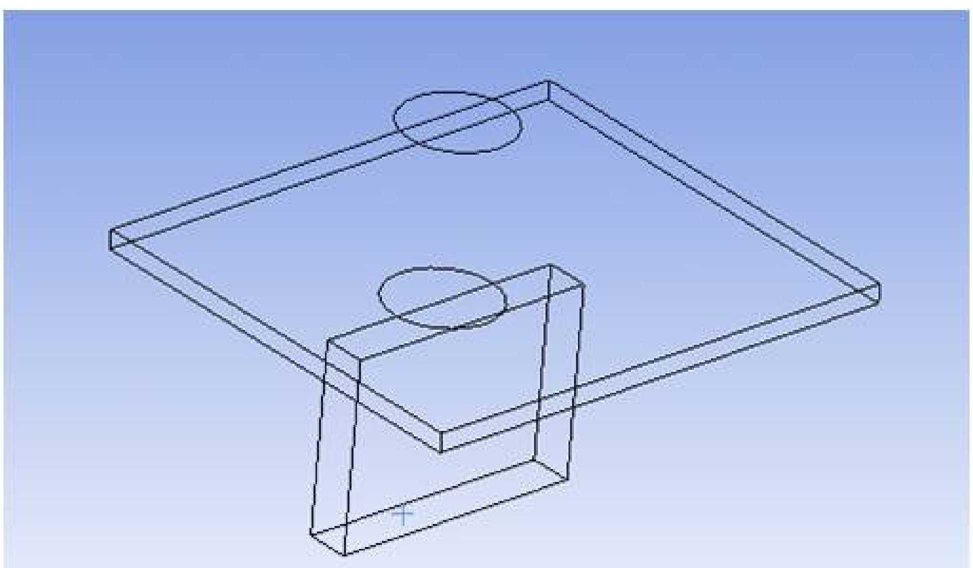
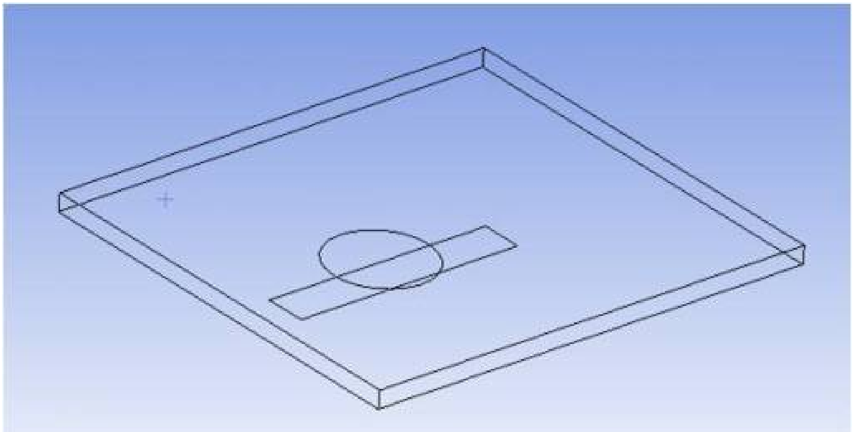


Figure 167: Source Imprinting Classifications: Intersecting Loops View 2



MultiZone Face Mappability Guidelines

To construct a swept or all hex mesh on typical geometries, side faces generally have to be able to be mapped or submapped. In most cases, if manual source faces are selected with **MultiZone**, all other faces are treated as side faces and are mapped/submapped if possible. However, certain model characteristics can lead to problems in mapping/submapping side faces, and ultimately lead to meshing failure. You can use virtual topologies to correct most issues and obtain successful side face handling and mesh generation. In some cases, you may also want source faces to be mapped/submapped, but that should be a lower priority.

In addition, imprinted regions must have a clear path of connecting side faces and handling intersections throughout the path.

Note:

If all faces of a body have a face meshing control set to **Mapped**, MultiZone will perform additional steps to ensure an all-mapped hex mesh is created. This can be important for cases where source faces of one body are side faces of an adjacent body so the entire body must be meshed with mapped faces in order to mesh the full model.

For guidelines that explain some common situations and steps you can take to resolve faces into mappable regions, see:

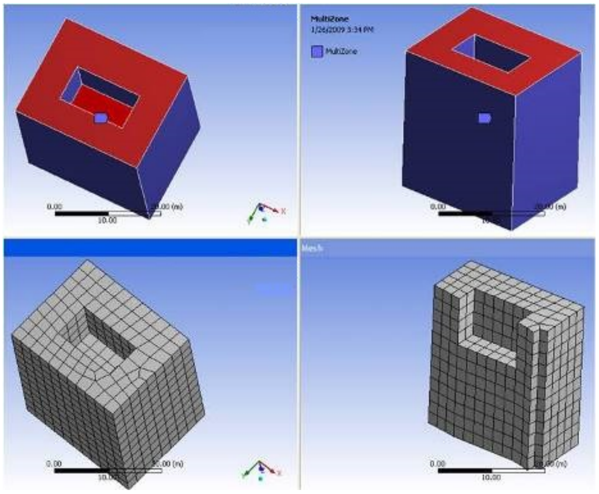
- Face Meshing Control (p. 265)
- Side Face Handling of Imprinted Regions (p. 356)

Side Face Handling of Imprinted Regions

When an imprint is made to connect two sets of source faces, side faces are constructed in the process. (See Figure 155: Classifying the Problem: Handling of Paths and Imprints (p. 348).) Generally, the pairs of imprints create a natural set of side faces that are mappable; however, if there are several sets of side faces along the sweep path, the interval edge assignment of the internal mapped faces can become tricky. Because there are not physical faces or edges to help define the interval edge assignment, you are sometimes better off adding more decomposition to help control this (either by splitting some of the external faces, or by slicing the geometry).

The simple cutout case shown below illustrates this.

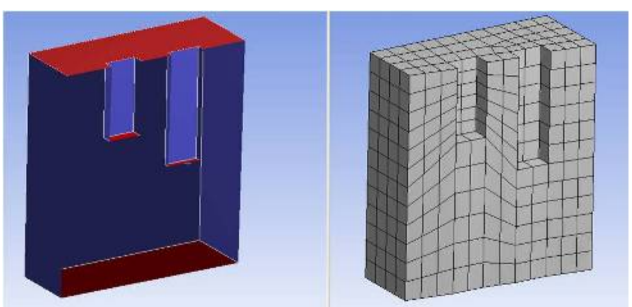
Figure 168: Simple Cutout Case



In the case above, there is a set of source faces at top, center, and bottom. Internal side faces are constructed from the center to the bottom and the interval edge assignment for these internal edges is found by subtracting the number of divisions for the top to center region of sweep path from the entire sweep path.

However, in cases in which the internal cutouts are at multiple levels or the sides do not provide a clear connected path, **MultiZone** could have difficulties constructing side faces with appropriate interval edge assignment unless you set the **Free Mesh Type** (p. 228) to something other than **Not Allowed**, or you perform manual geometry decomposition. Refer to the figure below for an example showing cutouts at multiple levels.

Figure 169: Cutouts at Multiple Levels



In the figure above, notice that cutouts at different levels are meshed at the same level with regard to the grid lines.

When there are intersections between the imprints and the side faces along the sweep path, some complications may arise. For example, the case shown in Figure 170: Intersections Between Levels and Sides (p. 358) has one clear sweep path (top to bottom), six levels (sets of sources), and a need for imprints.

The legend below explains the symbols and display options used in Figure 170: Intersections Between Levels and Sides (p. 358).

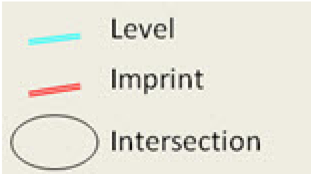
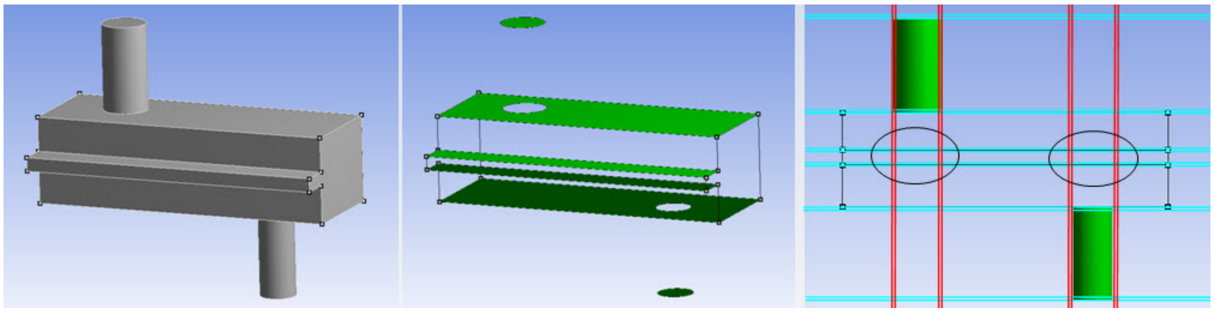
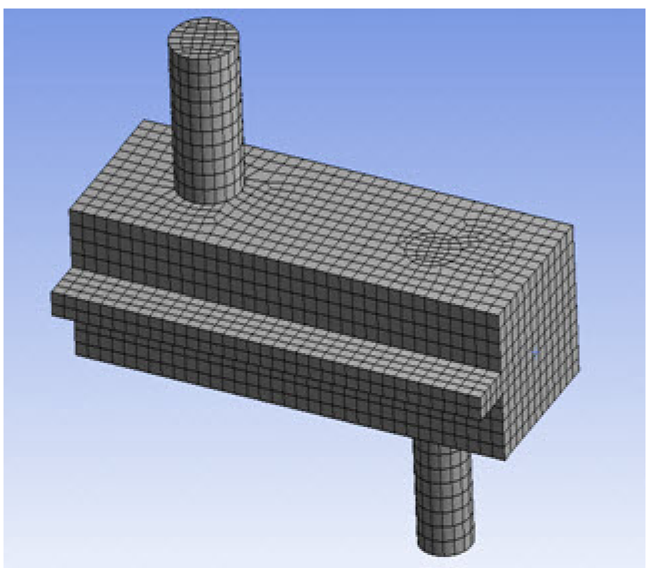


Figure 170: Intersections Between Levels and Sides



To construct a swept mesh on this model, **MultiZone** needs to resolve the intersections between the cylinders, and the side rib. In this case, meshing is successful because the sources are well-defined, and the side face handling is clear (side faces can be submapped appropriately).

Figure 171: Meshed Model

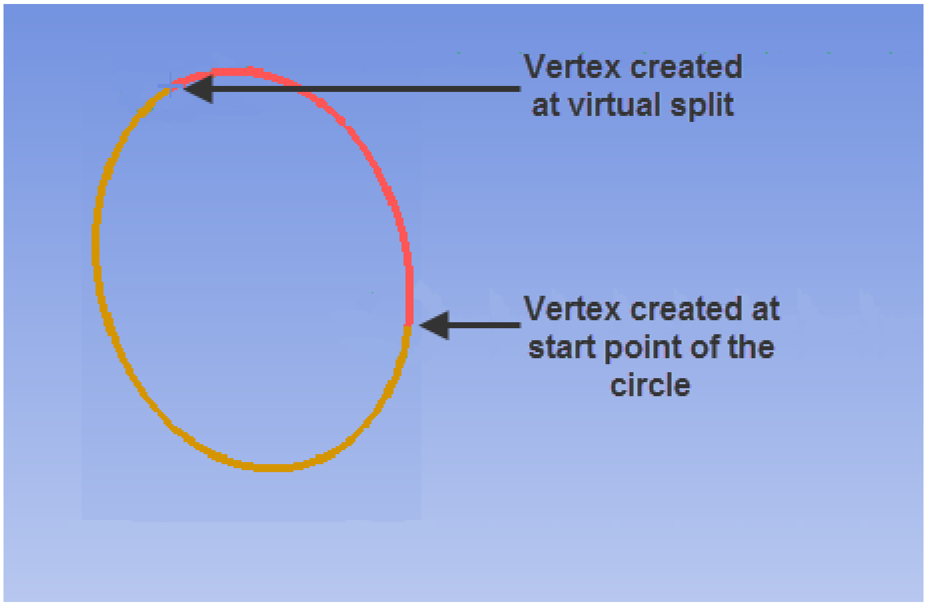


However, if the sources are not well-defined (or incomplete), or the side faces cannot be mapped, **MultiZone** will have difficulties. To fix such problems, add extra decomposition or mesh with the **Free Mesh Type** option.

Cylindrical Side Faces

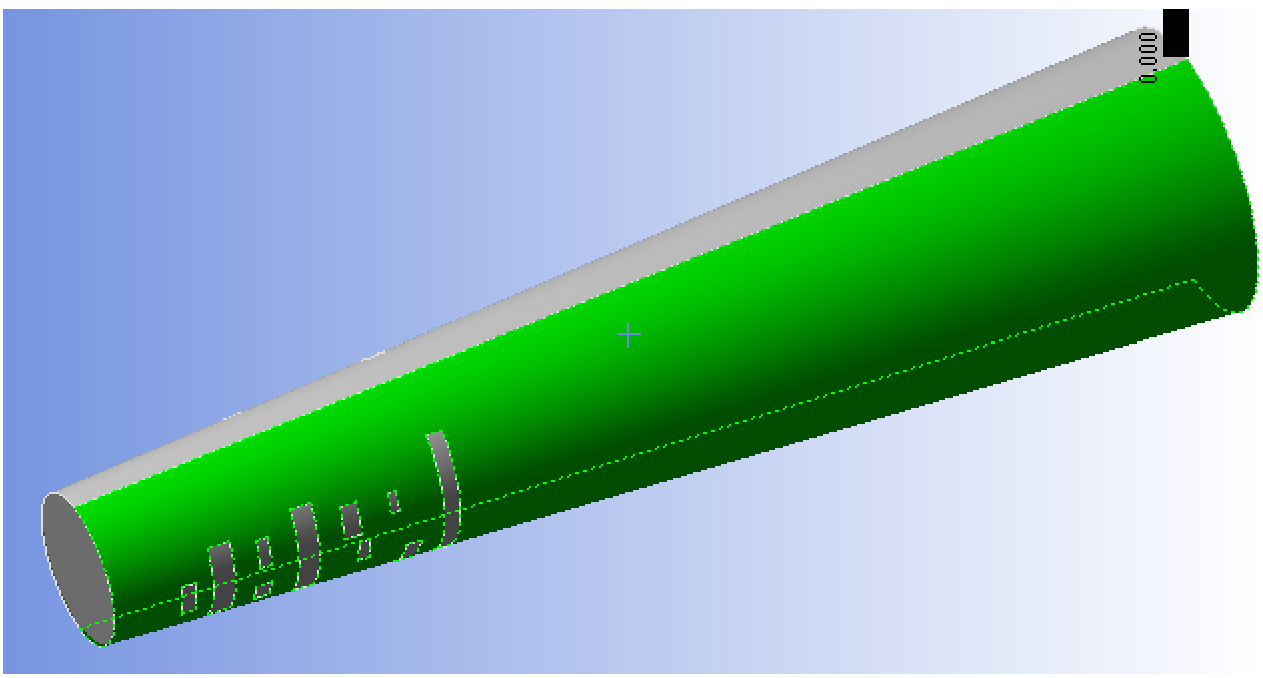
Cylinders have a start and end point which, at times, can affect the quality of the meshing. When you add a first virtual edge to split a circle (the side face of a cylinder), you create two vertices: one where you split the circle and one at the end point, as shown below.

Figure 172: Vertices in a Split Circle



If **MultiZone** fails to mesh a cylindrical surface with cut-outs folded at greater than 180 degrees, splitting the cylinder along the seam to get a sub-mapped face mesh may help. If you use this method, check to ensure that the split does not hamper mesh quality.

Figure 173: 360 ° Cutout

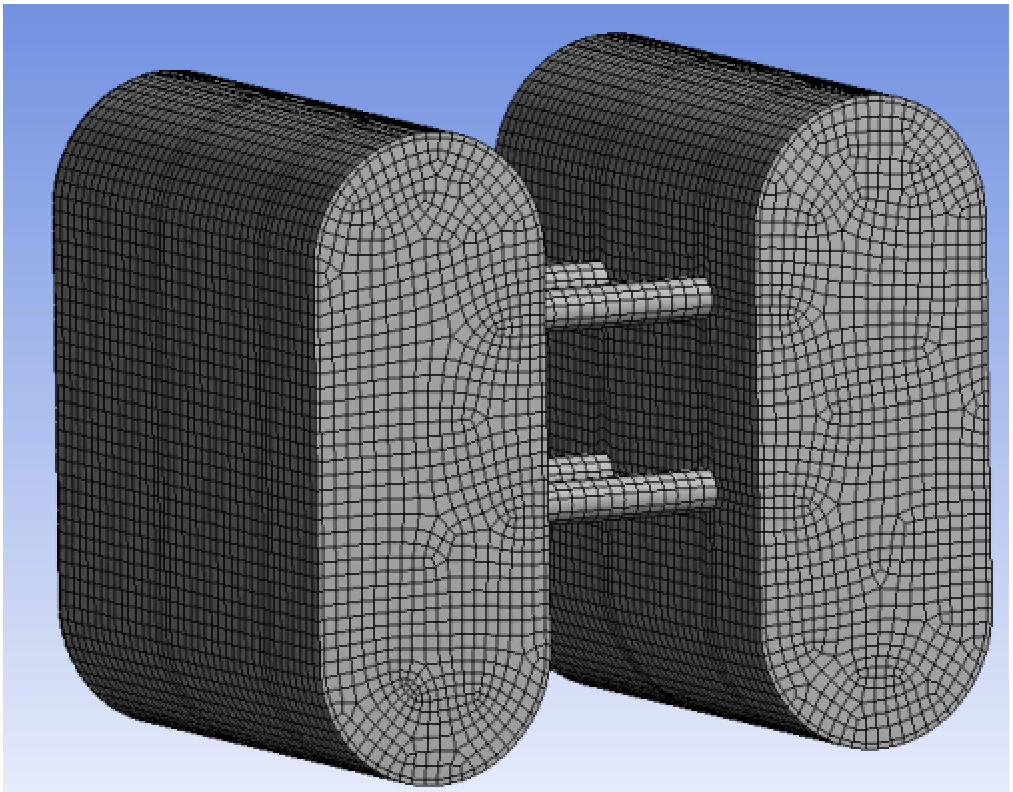


For more information about virtual split edges, see Creating and Managing Virtual Split Edges (p. 517).

Internal Loop Side Faces

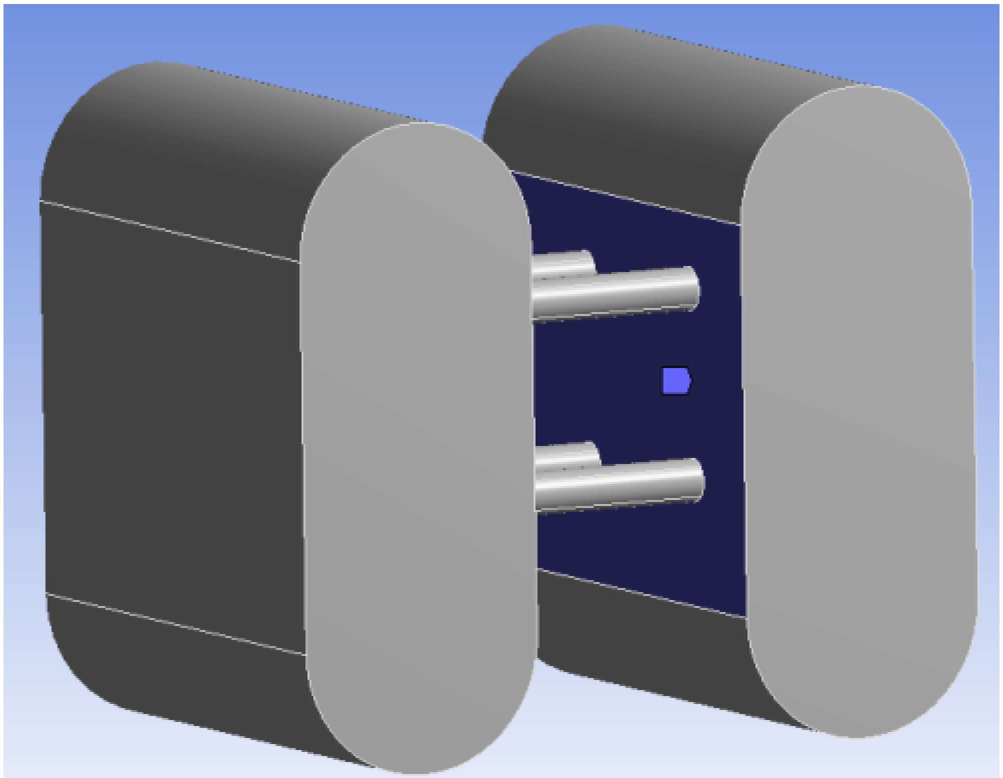
When internal loops exist along the side faces of the sweep path, as shown in Figure 174: Internal Loops along Side Faces of the Sweep Path (p. 360) below, the following tips might help:

Figure 174: Internal Loops along Side Faces of the Sweep Path



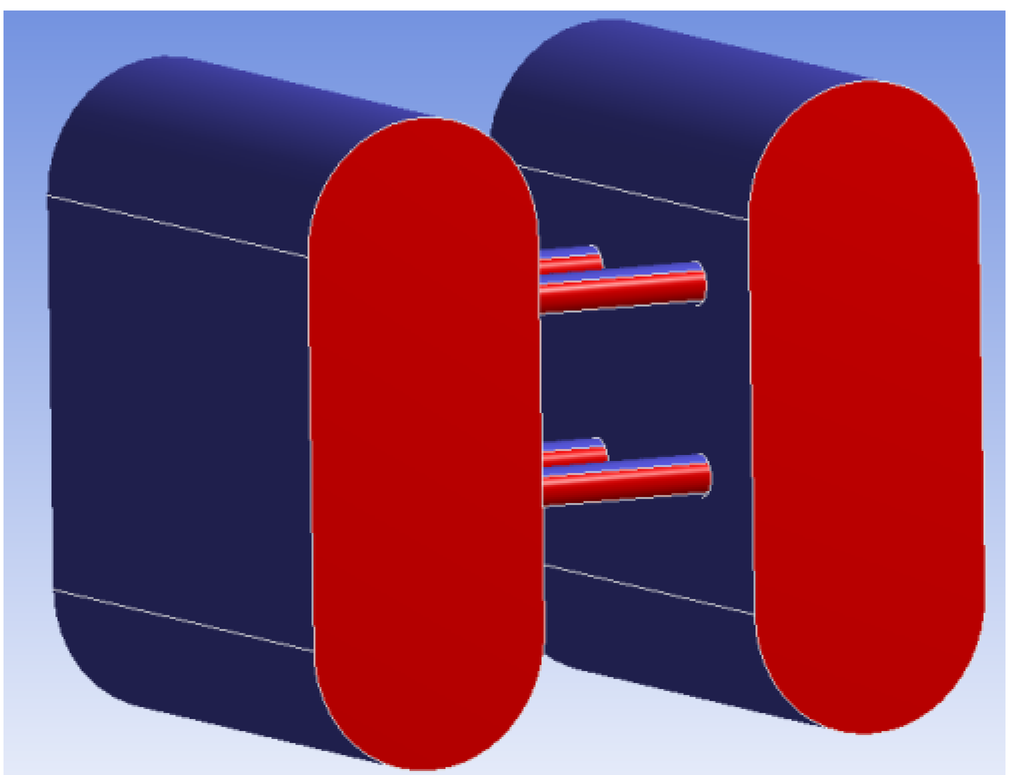
1. Assign a mapped face control to the side face(s) with the internal cutouts to help ensure those faces are mapped.

Figure 175: Map Face Control Assigned to Side Faces



2. Assign front/back of connecting faces as source faces for MultiZone.

Figure 176: Connecting Faces Assigned as Source Faces



3. If side faces are cylindrical, use inflation to get reasonable quality mesh.

Figure 177: Using Inflation on Cylindrical Side Faces

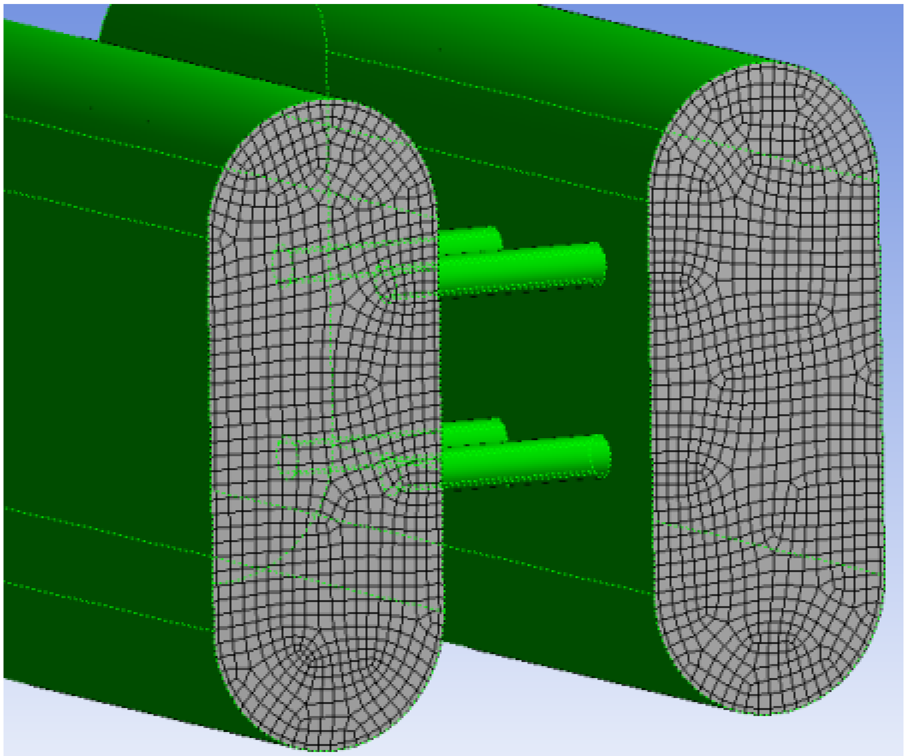
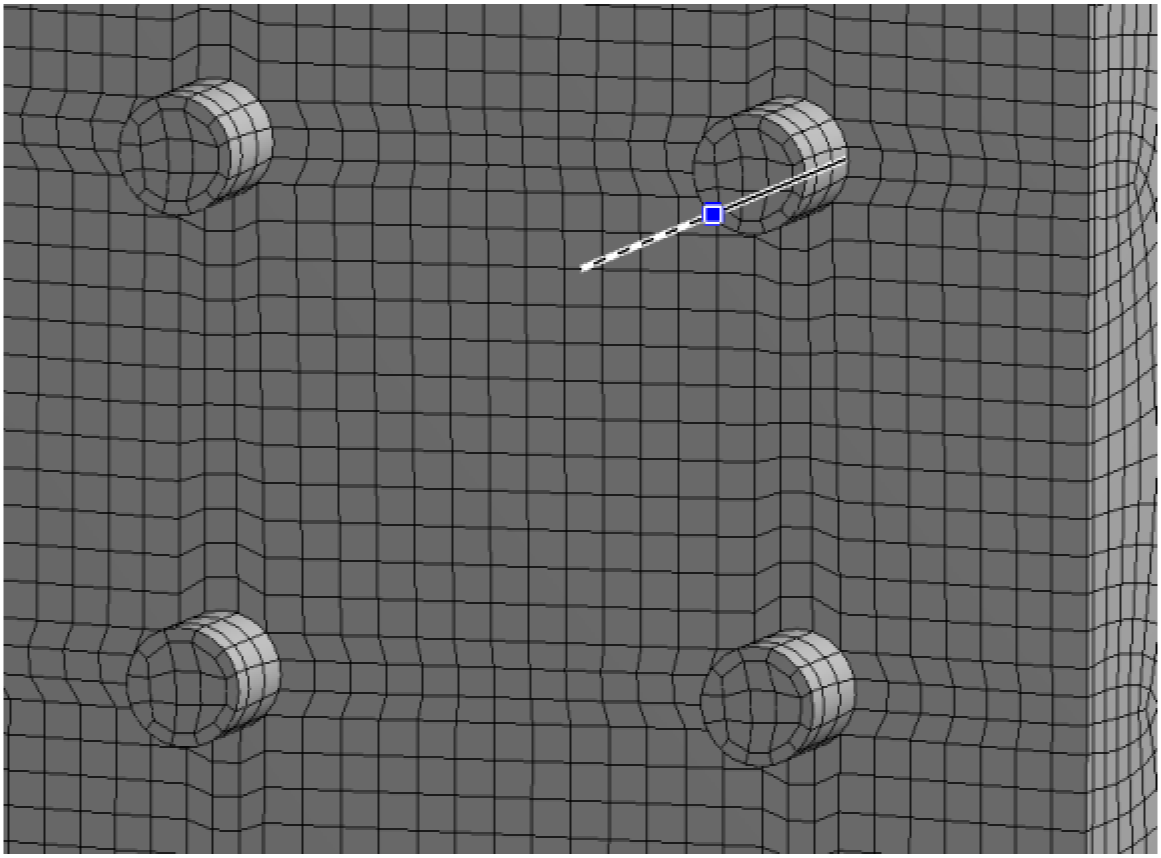


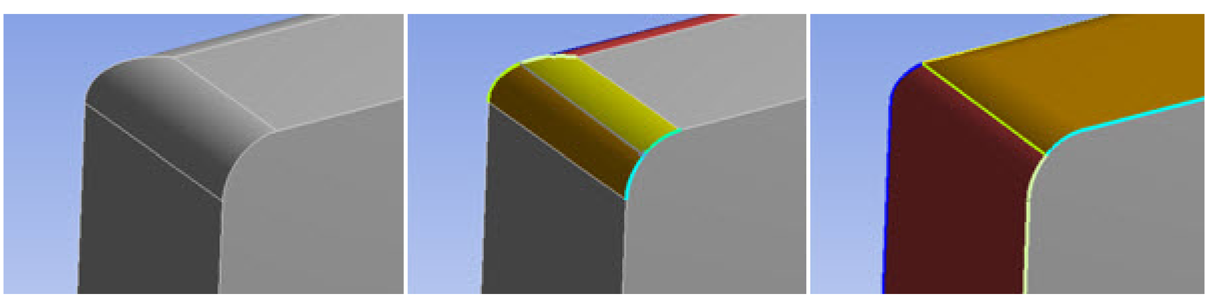
Figure 178: Using Inflation on Cylindrical Side Faces



Using Virtual Topology to Handle Fillets in MultiZone Problems

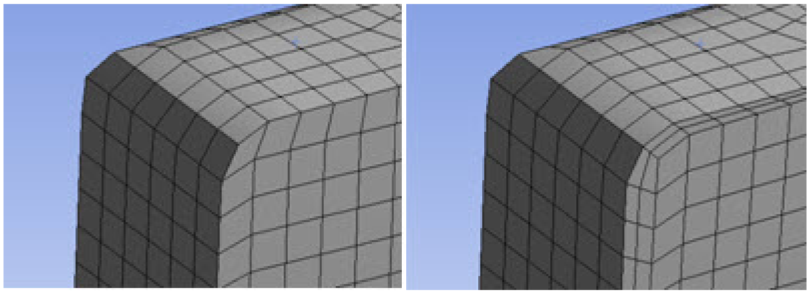
The presence of fillets in a model can cause problems for **MultiZone**, since arguably a fillet could be either a side face or a source face and the mesh quality in the corner could be problematic. Ideally, a fillet should be split in half, with one half going to the source face and the other half going to the side face.

Figure 179: Fillets and MultiZone



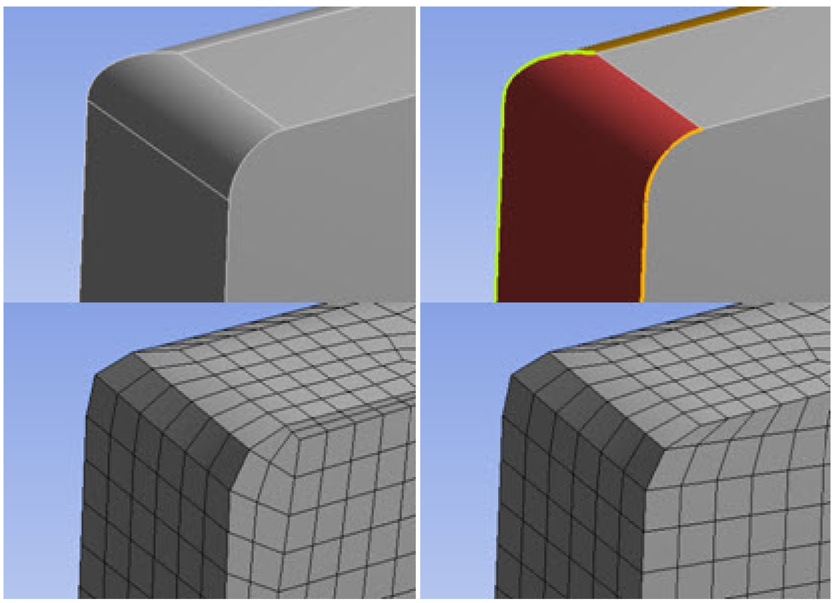
In addition, inflation is generally recommended to improve the high angled elements that would be formed along the edge that is shared between the source and side faces. Inflation allows the mesh to transition away from the boundary and reduce the angle. See Figure 180: Fillets and Inflation (p. 364).

Figure 180: Fillets and Inflation



Alternatively, the fillet could be removed by assigning the fillet to the side face through a virtual topology operation.

Figure 181: Fillets as Side Faces



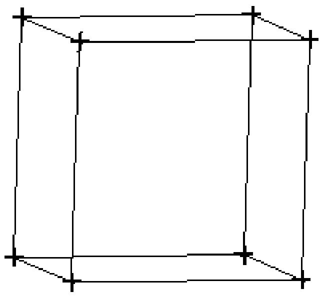
Refer to Meshing: Virtual Topology (p. 501) for more information.

MultiZone Support for Inflation

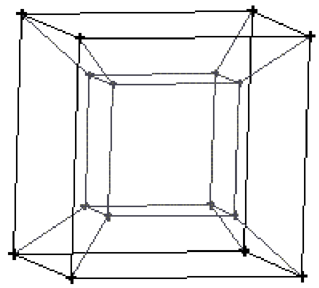
The Inflation Option (p. 150) setting determines the heights of the inflation layers. For the **MultiZone** mesh method, you can set **Inflation Option** to **Smooth Transition**, **Total Thickness**, or **First Layer Thickness**. **First Aspect Ratio** and **Last Aspect Ratio** are not supported.

The **Inflation Option** for **MultiZone** is set to **Smooth Transition** by default. The approach **Smooth Transition** uses for computing each local initial height for **MultiZone** differs from the approach used for tet mesh methods. This is because an inflated tet mesh contains different types of volume elements (tets and prisms where the ratio takes into consideration the difference in volume based on element shape), while in an inflated **MultiZone** mesh the elements generally will be the same type.

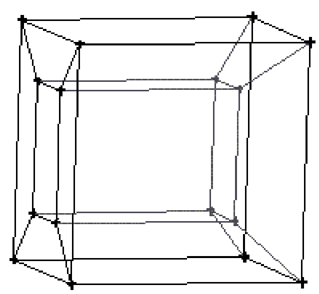
Inflation in **MultiZone** is done by offsetting the topology to construct inflation regions called an O-Grid. In the O-Grid creation, only the faces that are being inflated will be offset. For example, in a simple box (or sphere) like the following:



If all faces are inflated, you would get a complete O-Grid.



If you only select some faces of the box, the faces that are not inflated do not get an O-Grid region, and the inflation layers attach to that face. In the following image, all faces are inflated except the left face.



Inflation controls define:

- How thick the O-Grid region is.
- The sizing along the O-Grid.

When the **Smooth Transition** option is used with **MultiZone**, the O-Grid edge length varies based on the number of elements, and the local last inflation height is computed as Transition Ratio (p. 152) * local_mesh_size. As with other mesh methods when **Smooth Transition** is used, the inflation layers are created using the values of the Transition Ratio (p. 152), Maximum Layers (p. 153), and Growth Rate (p. 153) controls.

Note:

- Because the **MultiZone** mesh method uses an O-Grid technique for inflation, it cannot stairstep or peel back layers if there is insufficient room for all layers. Because of this, the default values for **Smooth Transition** inflation could be aggressive, depending on the model. Reducing the number of layers or switching to a different type of inflation definition may be more suitable for some models.

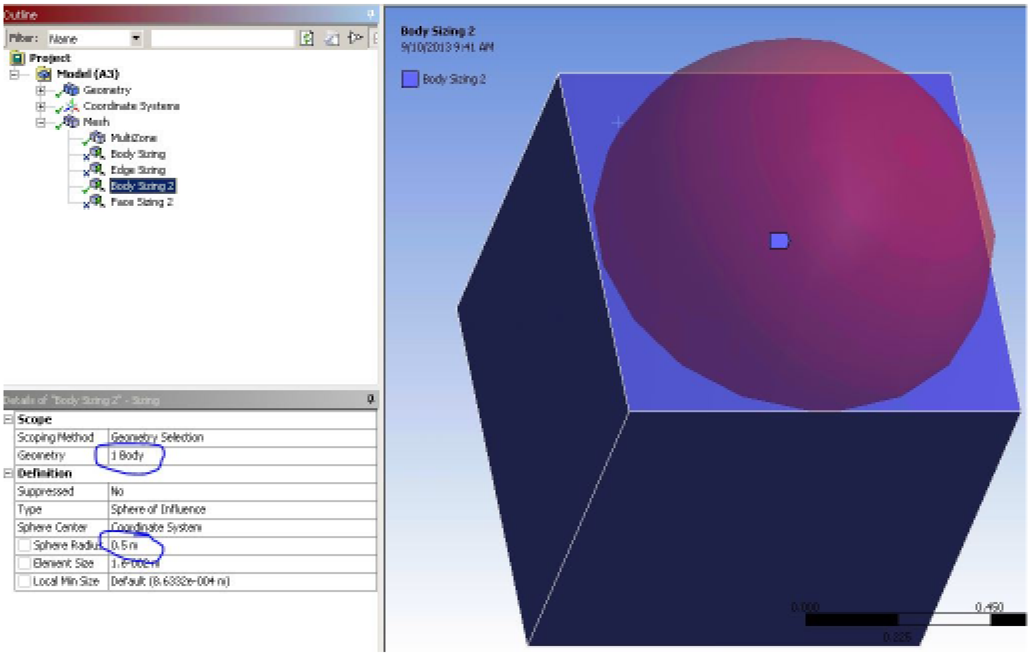
- **MultiZone** supports Program Controlled (p. 148) inflation.
- The Collision Avoidance (p. 158) option is not used for **MultiZone** as the inflation layers are created within the blocking approach.

MultiZone Limitations and Hints

Be aware of the following limitations and hints when using **MultiZone**:

- Pinch controls (p. 182) are not supported.
- Only three-dimensional bodies can be used as Bodies of Influence (BOIs) (p. 257).
- The hard-sizing behavior of Sphere of influence (SOI) (p. 256) controls is not supported; you can define a refinement with an SOI control but not a coarsening. The SOI element size is only applicable in the mesh if it is smaller than the local mesh size without SOI.
- BOI controls are interpreted as an agglomerate of SOI controls that fill the body of influence. Therefore, on the boundaries of that body, the size might be a little larger because the location is not in one of the spheres.
- The scope is not used for the SOI/BOI definition. The SOI/BOI does not act only on the scoped topology, but on all topologies within the region of the SOI/BOI.
- In **MultiZone**, size controls (p. 256) are first applied on the edges. For structured blocks, the mapped mesh of structured blocks is completely determined by the mesh on the attached edges; you cannot apply refinement or coarsening to the faces as you can for unstructured blocks. Therefore, to see the impact of the SOI on a mapped face, you must increase the radius of the SOI or lower the Growth Rate (p. 105) to influence the mesh size on the edges.

Figure 182: Sphere of Influence on Face that Doesn't Intersect Edges



- In **MultiZone**, a closed edge is represented by more than one blocking topological edges. Therefore, user-specified edge node distribution size may not be respected. However, the number of intervals will still be respected,

- For split curves in rotated edge association, only interval count, not distribution, will be respected.

- Some inflation cases are not supported; for example, when the inflation does not form a closed loop (which would lead to non-conformal mesh that is not allowed). In these cases the inflation will be skipped. Also see MultiZone Support for Inflation (p. 364).

- The Show Sweepable Bodies (p. 494) feature is a good tool to detect bodies that should mesh with MultiZone.

- There is no access to underlying blocks except by writing out the Ansys ICEM CFD files (p. 84).

- For help in diagnosing problems when using the **MultiZone** method, refer to the description of the **Edge** group in the Mechanical help. This toolbar provides access to features that are intended to improve your ability to distinguish edge and mesh connectivity.

- See Handling General MultiZone Meshing Failures (p. 546) for more information.

For additional information, refer to Interactions Between Mesh Methods and Mesh Controls (p. 438) (Patch Independent Mesh Methods table).

Assembly Meshing

Two algorithms are available for assembly meshing (p. 164):

- **CutCell**

- **Tetrahedrons**

CutCell Cartesian meshing is a general purpose meshing method designed for Ansys Fluent. The **Tetrahedrons** assembly meshing algorithm is a derivative of the **CutCell** algorithm, with strengths and weaknesses similar to those of **CutCell**. The **Tetrahedrons** method starts from the **CutCell** mesh and through various mesh manipulations creates a high quality unstructured tet mesh.

The **CutCell** method uses a top down volume meshing approach (surface mesh automatically created from boundary of volume mesh) without the need for manual geometry cleanup or decomposition, thereby reducing the turnaround time required for meshing. The **CutCell** method is useful for meshing fluid bodies in single body parts and multibody parts and assemblies of unconnected solids; it cannot be used to mesh a collection of loosely closed surface patches.

The **CutCell** algorithm is suitable for a large range of applications, and due to the large fraction of hex cells in the mesh, often produces better results than tetrahedral methods.

CutCell is supported in the Meshing application only. The **Tetrahedrons** algorithm is available in both the Meshing application and the Mechanical application; however, meshes generated using assembly meshing are not supported for Mechanical solvers. Refer to Method (p. 165) for details.

Note:

Assembly meshing or Fluent mesh export (p. 43) may fail if you are using shared licensing, no licenses are available, and Ansys Fluent is running already. In such cases, the error is due to shared licensing restrictions, but the error message that is issued does not identify licensing as the cause.

Note:

Assembly Meshing is being deprecated and will be removed in future releases.

Assembly meshing topics include:

The Assembly Meshing Process

The assembly meshing process involves the following approach:

1. Sizing controls and virtual bodies (optional) are defined.

   Note:

   You can define virtual bodies to represent flow volumes if your model does not contain a "solid" that represents the fluid region. In this way, virtual bodies allow you to mesh fluid regions without having to use the DesignModeler application or another solid modeler to model them. The **Fluid/Solid** material property (p. 379) for virtual bodies is always set to **Fluid** (read-only).

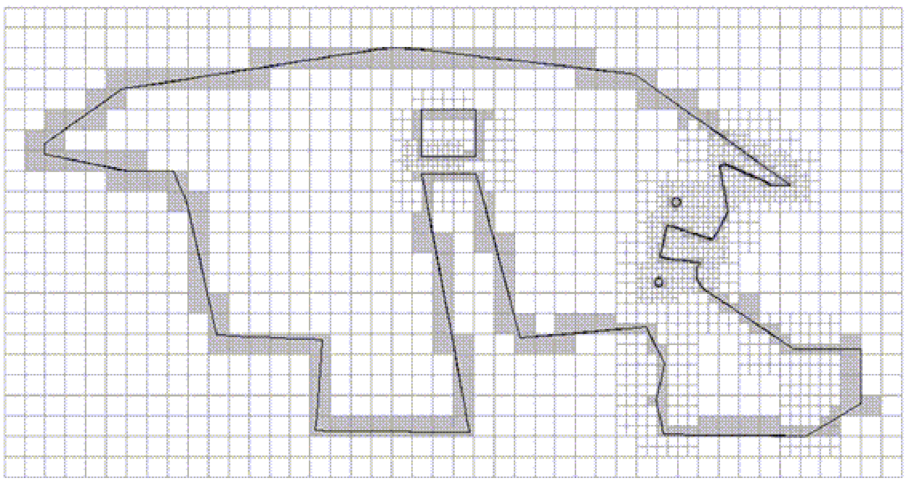
2. The initial size of the Cartesian grid is computed based on the minimum and maximum size set for the sizing controls.

3. A uniform Cartesian grid is created within the bounding box for the geometry.

4. The sizing values are computed and the grid is then adaptively refined based on the local sizing values.

   Figure 183: Mesh After Refinement (p. 369) shows the mesh after refining the initial grid based on sizing controls.

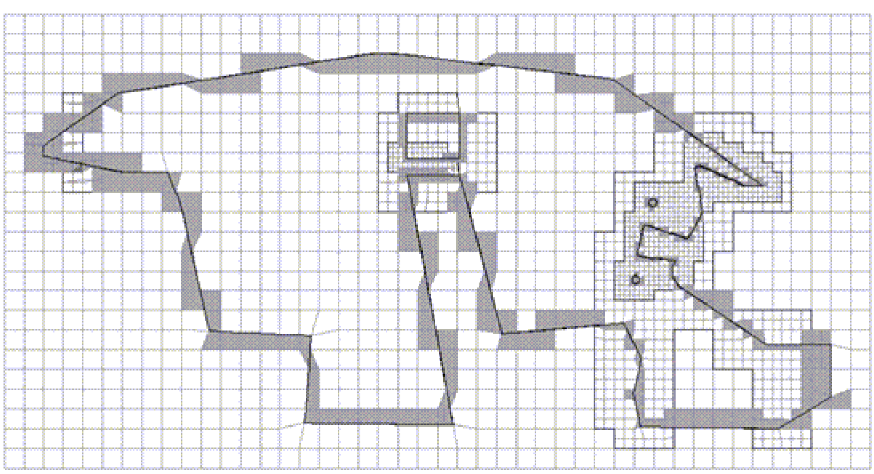
   Figure 183: Mesh After Refinement



5. The cells intersected by the geometry are marked. Only nodes on marked cells are considered for projection. The nodes are projected to the geometry (corner, edge, and face in order of reducing priority).

   Figure 184: Mesh After Projection (p. 369) shows the mesh after node projection.

   Figure 184: Mesh After Projection



6. The edges intersected by the geometry are identified. Mesh edges to be preserved/recovered are determined, and are used to construct mesh faces. Once the mesh faces are identified, cells are decomposed to recover these faces. The cells are decomposed based on a number of templates.

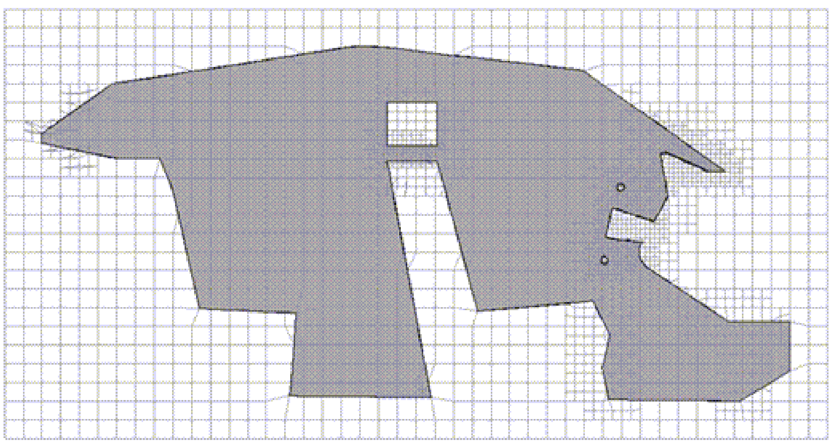
7. The mesh outside the fluid bodies is removed. The mesh inside the solid bodies may also be removed for assembly meshing, depending on the setting of the Keep Solid Mesh (p. 167) control.

8. Therefore, the quality of the cells generated is improved.

9. Cells are separated into cell zones based on the respective virtual bodies. A cell included in multiple virtual bodies will be included with the body having the highest priority. The smallest body, based on the size of the bounding box enclosing it, has the highest priority to guarantee that any voids are recovered properly.

   Figure 185: Cells Separated After Decomposition (p. 370) shows the cells separated into respective cell zones after decomposition.

   Figure 185: Cells Separated After Decomposition

10. The boundary mesh is recovered and separated based on the underlying geometry.

    - Faces whose adjacent neighboring cells are in different cell zones automatically constitute the boundary mesh.
    - The neighboring cells of a face on an internal baffle are in the same cell zone. In such cases, faces close to and nearly parallel to the baffle surface are recovered to represent the baffle surface.
    - As each cell zone is a closed region, the mesh boundary is conformal.

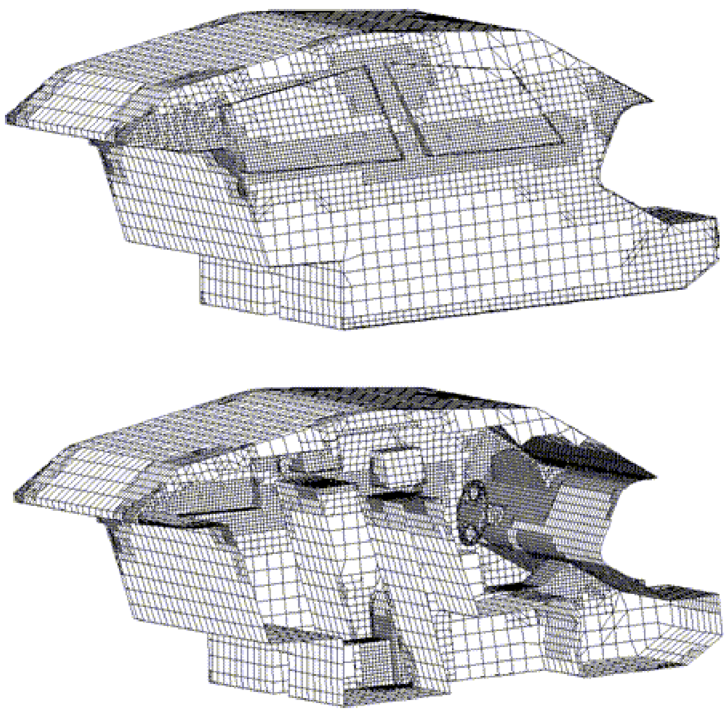
The boundary zone types are assigned based on the Named Selections defined on the (underlying) geometry faces.

Note:

Named Selection names for internal face zones are not interpreted. In cases where two enclosed voids share a face, the face zone is assigned type *WALL* automatically regardless of whether a Named Selection has been defined for the face. In these cases, the mesh generation cannot cross any boundary so you must define a virtual body with material point for each flow volume void in order for the volumes to be meshed.

Figure 186: CutCell Mesh After Boundary Recovery (p. 371) shows the **CutCell** mesh after the boundary mesh is recovered.

Figure 186: CutCell Mesh After Boundary Recovery



When the mesh is exported to Ansys Fluent, a cell zone type of either *FLUID* or *SOLID* is assigned to each body based on its material properties. Refer to Fluent Mesh Export (p. 43) for details.

For information about material properties and which faces are selected to be inflation boundaries when **Program Controlled** inflation is defined for assembly meshing, refer to Program Controlled (p. 148).

11. If the **Tetrahedrons** assembly meshing algorithm was selected, tet meshing extensions are applied. The **Tetrahedrons** algorithm starts with the **CutCell** mesh and transforms it into an unstructured tet mesh.

The Assembly Meshing Workflow

This section describes the workflow for using assembly meshing, which is different from the workflow for the other Meshing application mesh methods. Rather than being applied locally through insertion of a mesh method control, assembly meshing is a global operation as implied by its name and cannot be used in combination with other meshing methods.

The general workflow for assembly meshing is as follows:

1. Set prerequisites (p. 374).
2. Select an assembly mesh method (**CutCell** or **Tetrahedrons**) (p. 375).
3. Change **Fluid/Solid** material property settings (optional) (p. 379).
4. Define virtual bodies (optional) (p. 379).
5. Define mesh groups (optional) (p. 389).
6. Set global assembly meshing options (p. 390) (**Feature Capture**, **Tessellation Refinement**, and **Keep Solid Mesh**).
7. Define sharp angle controls (optional) (p. 390).
8. Set sizing options (p. 390).
9. Find thin sections (optional) (p. 393).
10. Find contacts (optional) (p. 395).
11. Generate the mesh (p. 396).
12. Apply contact sizing (p. 398).
13. Set global inflation controls (p. 400).
14. Generate the inflation mesh (p. 401).
15. Apply local (scoped) inflation controls and regenerate the inflation mesh (optional) (p. 401).
16. Export the mesh (p. 404).

You may modify the workflow to handle your specific assembly meshing problem. Some examples are presented below.

Sample Assembly Meshing Workflow for Oil Drill Bit or Valve

Assume the case involves a model of an oil drill bit or valve with these characteristics:

- The model consists of a single or multibody part.
- Shared topology is working, and there are only single interior faces in the model.
- You already know that there are no unresolved gaps or thin sections in the model.

- You know that the model has sections with acute angles.

The following workflow is appropriate for this case:

1. Set prerequisites (p. 374).
2. Select an assembly mesh method (**CutCell** or **Tetrahedrons**) (p. 375).
3. Change **Fluid/Solid** material property settings (optional) (p. 379).
4. Set global assembly meshing options (p. 390) (**Feature Capture** and **Tessellation Refinement**).
5. Define sharp angle controls (p. 390).
6. Set sizing options (p. 390).
7. Generate the mesh (p. 396).
8. Set global inflation controls (p. 400).
9. Generate the inflation mesh (p. 401).
10. Apply local (scoped) inflation controls and regenerate the inflation mesh (optional) (p. 401).
11. Export the mesh (p. 404).

Sample Assembly Meshing Workflow for an Automotive Assembly

Assume the case involves a model of an automotive assembly with these characteristics:

- You do not know all of the details of the model.
- You need to mesh the flow volume, and you may also need to mesh the assembly.

The following workflow is appropriate for this case:

1. Set prerequisites (p. 374).
2. Select an assembly mesh method (**CutCell** or **Tetrahedrons**) (p. 375).
3. Change **Fluid/Solid** material property settings (optional) (p. 379).
4. Define virtual bodies (p. 379).
5. Define mesh groups (optional) (p. 389).
6. Set global assembly meshing options (p. 390) (**Feature Capture**, **Tessellation Refinement**, and **Keep Solid Mesh**).
7. Define sharp angle controls (optional) (p. 390).
8. Set sizing options (p. 390).
9. Find thin sections (optional) (p. 393).

10. Find contacts (p. 395).
11. Generate the mesh (p. 396).
12. Apply contact sizing (p. 398).
13. Set global inflation controls (p. 400).
14. Generate the inflation mesh (p. 401).
15. Apply local (scoped) inflation controls and regenerate the inflation mesh (optional) (p. 401).
16. Export the mesh (p. 404).

Setting Prerequisites

To expose the **Assembly Meshing** group of controls, you must set the following prerequisites:

1. Set Physics Preference (p. 93) to **CFD**.
2. Set Solver Preference (p. 95) to either **Fluent** or **Polyflow**.

As a result, the **Assembly Meshing** group of global controls appears in the Details View with **None**, **CutCell**, and **Tetrahedrons** as options for **Method**. Refer to Method (p. 165) for details.

Note:

- When **Physics Preference** is set to **CFD** and assembly meshing is active, a shape checking algorithm based on orthogonal quality (p. 142) is used. Orthogonal quality, which is the recommended quality criterion for CFD simulations, can be used for all types of meshes including **CutCell** and polyhedral. Note that the skewness quality criterion is not recommended for **CutCell** meshes.
- Orthogonal quality (p. 142) in the Meshing application is equivalent to Inverse Orthogonal Quality in Ansys Fluent Meshing, except that the scale is reversed:

  Inverse Orthogonal Quality = 1 – Orthogonal Quality

  The orthogonal quality values may not correspond exactly with the inverse orthogonal quality values in Ansys Fluent because the computation depends on boundary conditions on internal surfaces (*WALL* vs. *INTERIOR*/*INTERNAL*/*FAN*/*RADIATOR*/*POROUS-JUMP*). Ansys Fluent may return different results which reflect the modified mesh topology on which CFD simulations are performed. Also, for **CutCell** meshes, the elements in the Meshing application are "traditional" (hex/tet/wedge/pyramid) elements. When a **CutCell** mesh is exported from the Meshing application to Ansys Fluent, elements that are connected to parent faces are exported in polyhedral format, while all others retain their type. Note that this behavior is only true for **CutCell**; the **Tetrahedrons** algorithm uses only traditional element types.

Selecting an Assembly Mesh Method

Selecting an assembly meshing algorithm will expose the assembly meshing controls and hide controls that are not applicable to assembly meshing:

1. In the **Assembly Meshing** group of global controls in the Details View, set Method (p. 165) to **CutCell** or **Tetrahedrons**.

After selecting a method, you will continue to have access to the following mesh controls with the noted exceptions and additions that are unique to assembly meshing algorithms. Controls are discussed in more detail in the appropriate workflow steps.

Sizing Controls

Most Sizing controls (p. 100) are supported.

- Adaptive Sizing (p. 90) is not supported.
- Uniform Sizing (p. 103) is not supported.

Local (Scoped) Size Controls

Some local (scoped) size (p. 248) controls are supported.

- The **Element Size** (p. 255) option for **Type** is supported for local body, face, and edge sizing. For edge sizing, the **Number of Divisions** option for **Type** is not supported. Any bias options (p. 262) applied with edge sizing are ignored.
- The **Body of Influence** (p. 257) option for **Type** is supported for local body (p. 248) sizing, but the body of influence cannot be scoped to a line body.
- If you want to use a body of influence with a virtual body (p. 379), you can scope the body of influence to any body in the geometry. The body of influence does not have to be inside or even in contact with the scoped body.
- The **Sphere of Influence** (p. 256) and **Number of Divisions** (p. 258) options for **Type** are not supported.
- No local vertex sizing (p. 252) is supported.
- **Contact Sizing** (p. 263) is supported. However, if contact sizing is applied to entities on a body that is scoped to a body of influence, the contact sizing is ignored.

If any unsupported local size controls are defined prior to selection of an assembly meshing algorithm, they are suppressed when an assembly meshing algorithm is selected.

Inflation Controls

3D inflation controls are supported. This includes global (p. 145) (automatic **Program Controlled**) and local (p. 291) (scoped) controls.

- A mixture of global and local inflation is *not supported* for assembly meshing algorithms. Consider the following when determining whether to use global or local inflation:

  - For inflation on virtual bodies, you must use automatic **Program Controlled** inflation; you cannot use local controls to inflate virtual bodies. Thus in general, if you are using virtual bodies to represent flow volumes in your model, plan to use automatic inflation. Automatic inflation is specified globally by setting Use Automatic Inflation (p. 147) to Program Controlled (p. 148). With **Program Controlled** inflation, faces on real solid bodies will inflate into the virtual bodies. The **Fluid/Solid** designation on real bodies will be respected (that is, faces on real fluid bodies will inflate into the fluid region, but the solid region will not be inflated).

  - Alternatively, you can set **Use Automatic Inflation** to None (p. 147) and define local inflation controls. This approach is appropriate if your model contains real bodies that represent the fluid regions.

  The restriction from mixing global and local inflation results in the following behaviors:

  - If **Program Controlled** inflation and local inflation are both defined and you select an assembly meshing algorithm, **Program Controlled** inflation overrides the local inflation controls. The local inflation controls become inactive.

  - If an assembly meshing algorithm is selected and local inflation is defined, the local inflation controls become inactive if you select **Program Controlled** inflation.

  - If an assembly meshing algorithm is selected and **Program Controlled** inflation is defined, the option to insert local inflation will be grayed out (unavailable). You must set **Use Automatic Inflation** to **None** to be able to insert local inflation.

- By default, Inflation Option (p. 150) is set to **Smooth Transition** and Transition Ratio (p. 152) is set to **0.272**. If you set **Transition Ratio** prior to selecting assembly meshing, your setting will be ignored for assembly meshing but will be restored if you subsequently deselect assembly meshing and return to another mesh method.

- The Inflation Algorithm (p. 154) control, which is used to select either the **Pre** or **Post** inflation algorithm for other mesh methods, is hidden when an assembly meshing algorithm is selected.

  For the **CutCell** algorithm, inflation is neither **Pre** nor **Post**. Rather, it may be considered a hybrid of the two, in that the technology used is like that of the **Pre** algorithm, but inflation occurs **Post** mesh generation. For the **Tetrahedrons** algorithm, **Pre** inflation is used, with inflation behaviors and limitations very similar to those of the Patch Conforming Tetrahedron (p. 200) mesh method.

  If you set **Inflation Algorithm** prior to selecting an assembly meshing algorithm, your setting will be ignored for assembly meshing but will be restored if you subsequently deselect assembly meshing and return to another mesh method.

- Collision Avoidance (p. 158) is set to **Layer Compression** and is read-only. Note, however, that layer compression is used in areas of proximity and bad normals. In other problematic scenarios (for example, non-manifold nodes, bad surface mesh, and so on), local stair stepping is performed. As a result of local stair stepping, poor quality cells may be introduced into the mesh. Because of this possibility, a warning message will appear whenever stair stepping occurs. The message will not identify the location of the stair stepping; however, it often coincides with the location of the worst quality cells. For this reason, using the Mesh Metric (p. 123) feature to locate the worst quality cells is also likely to locate the areas where stair stepping occurred. To avoid stair stepping, make sure that the correct faces have been picked for inflation and that small features are properly resolved, as stair stepping also may be related to bad resolution of acute angles.

- Inflation on baffles is supported for the **Tetrahedrons** algorithm only. Inflation can be applied to baffles using global (automatic Program Controlled (p. 148) inflation) or local controls. When **Program Controlled** inflation is used, baffles are automatically selected to be inflation boundaries unless they are part of a Named Selection (with the **Program Controlled Inflation** option set to **Exclude**). Only two-sided growth cases for inflation are supported. Inflation layers will stair step at free boundary edges of the baffles.

- By default, Gap Factor (p. 161) is set to **1.5** for the **CutCell** algorithm. For the **Tetrahedrons** algorithm, **Gap Factor** is set equal to the value that is specified for non-assembly mesh methods (**0.5** by default). The **Gap Factor** settings for the **Tetrahedrons** algorithm and the non-assembly mesh methods are synchronized such that if one is changed, the other is updated accordingly. However, changing the value specified for the **CutCell** algorithm has no effect on the value being used by the **Tetrahedrons** algorithm and non-assembly mesh methods, and vice versa.

  Gap Factor for the **Tetrahedrons** algorithm follows the same logic that is used by the Patch Conforming Tetrahedrons (p. 200) mesh method.

  When you are using an assembly meshing algorithm, if the aspect ratio of cells in the inflation layer reaches 50, a warning message is issued. The warning message will suggest that you reduce the gap factor, as doing so may improve the aspect ratio of these cells. Although reducing the gap factor (to 0.5, for example) may improve the quality, it may also have a negative impact on robustness. A higher value (1.5, for example) is generally more robust, but may not result in the best mesh quality.

- If you make changes to inflation settings after generating the mesh using an assembly meshing algorithm, each subsequent re-mesh begins with the initial (cached) mesh and inflates from there.

Statistics Controls

Statistics (p. 193) controls are supported.

Virtual Topology

The use of virtual topology (p. 501) with assembly meshing is generally not recommended as it can lead to poor faceting, resulting in poor feature capturing and/or mesh quality in assembly meshes. For these reasons, a warning message is issued when virtual topologies are invoked with assembly meshing.

Unsupported Controls

Controls and features that are inaccessible when an assembly meshing algorithm is selected include those listed below.

- The global **Triangle Surface Mesher** (p. 177) control

- The global **Advanced** (p. 175) group of controls

- The following global Defeaturing controls:

  - **Automatic Mesh Based Defeaturing** (p. 106)

  - **Pinch** (p. 182)

  - **Loop Removal** (p. 192)
- The following local (scoped) controls:
  - **Method** (p. 196)
  - **Refinement** (p. 264)
  - **Mapped Face Meshing** (p. 265)
  - **Match** (p. 280)
  - **Pinch** (p. 286)
  - 2D (face) **Inflation** (p. 291)
- Rigid bodies (p. 424)
- Symmetry—In the case of the Symmetry feature, it is accessible when an assembly meshing algorithm is selected but it will not be respected. If you attempt to use Symmetry with an assembly meshing algorithm, a warning message is issued.
- The following RMB menu options:
  - **Preview> Inflation** (p. 492)
  - **Preview> Source and Target Mesh** (p. 491)
  - **Preview> Surface Mesh** (p. 489)
  - **Show> Mappable Faces** (p. 498)
  - **Show> Program Controlled Inflation Surfaces** (p. 493)
  - **Show> Removable Loops** (p. 495)
  - **Show> Sweepable Bodies** (p. 494)

Note:

For supported mesh methods, the **Preview Surface Mesh** feature helps you to verify that your mesh settings are correct by allowing you to visualize and examine the surface mesh prior to generating the full mesh. The inaccessibility of **Preview Surface Mesh** for assembly meshing does not present an obstacle for internal flow problems, as you can easily see the mesh. However, since external flow problems involve a void, the following alternatives are recommended:

- Use a Section Plane to look at the surface from the inside. This may be sufficient for simple models.
- For more complex models, define a Named Selection that includes all of the internal boundaries of the model, except those for which a Named Selection already exists. You can then view the surface mesh by viewing the Named Selection.

Changing Fluid/Solid Material Property Settings

You can use the **Fluid/Solid** material property setting to control the physics that will occur on a model:

1. Select a prototype (**Body** object) in the Tree Outline.
2. In the Details View, select the appropriate option for **Fluid/Solid**. Valid options are **Fluid**, **Solid**, and **Defined By Geometry**.

   When set to **Defined By Geometry**, the value is based on the **Fluid/Solid** material property that was assigned to the body in the DesignModeler application.

Note:

- **Fluid/Solid** also appears in the Details View if you select a **Virtual Body** object in the Tree Outline, but in such cases it is always set to **Fluid** (read-only).
- Refer to Fluent Mesh Export (p. 43) for information about this setting and how material properties are translated for use in Ansys Fluent.
- **Fluid/Solid** is available in the Meshing application only; it is not available if you are using the meshing capabilities from within the Mechanical application.

Defining Virtual Bodies

Virtual bodies are supported for assembly meshing algorithms (p. 367) only. If your model does not contain a "solid" to represent a flow region, a virtual body can be defined so that you can mesh the flow region without having to use the DesignModeler application or another solid modeler to model it. Virtual bodies are extremely beneficial in cases where it is not practical to extract flow volumes in a modeler. They can be defined by material point only, or by material point and a group of fluid surfaces.

As defined by a material point, the virtual body is an object that is defined by a coordinate system representing the origin of a material, and which extends from the origin until it reaches the face of another solid (such as when a ray is traced from the origin to the closest solid).

The two main reasons for using a group of fluid surfaces to help define a virtual body are performance and mesh quality.

For example, consider a case involving large assembly for which you want to analyze a small flow volume but have no interest in the solid mesh. If you define the virtual body by material point only, assembly meshing refines and resolves every solid in the assembly, and finally, using mesh separation and the material point, identifies the flow volume and discards the solid mesh. However, if you select a group of fluid surfaces to help define the virtual body, assembly meshing resolves only the boundaries of the fluid surfaces in the group; essentially, it ignores all other boundaries. This approach typically speeds up mesh generation by a factor of two or more.

With respect to mesh quality, using a group of fluid surfaces to help define a virtual body is a good idea if you have a fluids-only problem (**Keep Solid Mesh** (p. 167) is set to **No**) where within the solids there are 3D details and/or bodies that are smaller than the minimum size and hence would be problematic to resolve.

If fluid surface creation using the **Extend to Connection** option "leaks" to the outside of the domain, the leak must be closed. This approach provides an alternative way of detecting leaks. **Extend to Connection** is discussed in more detail below.

Note:

- Virtual bodies and fluid surface objects are not supported when Meshing application capabilities are accessed from within the Mechanical application.
- A virtual body and the optional group of fluid surfaces that is used to help define it are represented in the Tree Outline by **Virtual Body** objects and **Fluid Surface** objects respectively. A material point must be specified when a **Fluid Surface** object is being used to help define a virtual body. Both the **Fluid Surface** object and the **Virtual Body** object will remain underdefined until the material point is specified.
- The **Fluid/Solid** material property for a virtual body is always set to **Fluid** (read-only).

Defining a Virtual Body

The steps for defining a virtual body are presented here.

1. Select the **Geometry** object in the Tree Outline.
2. Insert a **Virtual Body** object into the Tree using either of these methods:
   - Right-click the **Geometry** object and select **Insert** > **Virtual Body** from the context menu.
   - Choose **Virtual Body** on the **Geometry** context toolbar.

   Note:

   If this is the first **Virtual Body** object to be inserted, a **Virtual Body Group** object will be inserted into the Tree along with the **Virtual Body** object as its child object.

   ---

3. In the Details View, specify a value for **Visible**. The default is **Yes**. If set to **No**, the resultant mesh of the virtual body will not appear in the **Geometry** window.
4. Specify a value for **Suppressed**. The default is **No**. If set to **Yes**, the virtual body will be suppressed and will not be meshed, nor will it be sent to the solver if it is already meshed.
5. In the Details View, **Used By Fluid Surface** specifies whether the virtual body is being used by a group of fluid surfaces. The default is **No**. If the virtual body will be used by a group of fluid surfaces, insert a **Fluid Surface** object into the Tree by right-clicking on the **Virtual Body** object and selecting **Insert** > **Fluid Surface** from the context menu.
   - Alternatively, you can change the setting to **Yes** in the Details View, and a new **Fluid Surface** object will be inserted under the **Virtual Body** object.

- Only one **Fluid Surface** object can be associated to any one **Virtual Body** object.
- If you switch the value of **Used By Fluid Surface** from **Yes** to **No**, the **Fluid Surface** object will be hidden.

6. Click the **Virtual Body** object in the Tree to continue defining it. For **Material Point**, specify the coordinate system to be used for the virtual body, and the faces will be oriented accordingly. The default is **Please Define**.

   - The **Fluid Surface** object and **Virtual Body** object will remain underdefined until a material point is specified.
   - You can either select the default coordinate system or define a local coordinate system. In either case, the setting will be retained, even if the **Used By Fluid Surface** setting is changed later.

   The remaining fields for the virtual body are read-only:

   - **Fluid/Solid** - Always set to **Fluid** for virtual bodies.
   - **Nodes**, **Elements**, **Mesh Metric** - Data associated with the virtual body when meshed.

7. If **Used By Fluid Surface** is **No**, the definition of the virtual body is now complete. If the virtual body will be used by a group of fluid surfaces (that is, **Used By Fluid Surface** is **Yes**), expand the **Virtual Body** object in the Tree Outline to expose the **Fluid Surface** object.

8. To define and make adjustments to the group of fluid surfaces, click the **Fluid Surface** object and use the following controls. These controls appear in the Details View of the **Fluid Surface** object.

   - **Faces To Group**: In the **Geometry** window, select the faces that should be members of the group. Refer to the example of the aero valve assembly below for information about automating face selection.
   - **Primary Virtual Body**: Read-only name of the primary virtual body.
   - **Priority**: Determines which group will claim cells in cases where groups overlap. The priority is initially based on the rule: the smaller the volume, the higher the priority.
   - **Suppressed**: Read-only setting that is inherited from the virtual body.

Notes on Virtual Bodies

Remember the following information when defining virtual bodies:

- Connections are not supported.
- Statistics are supported.
- When coordinate systems are created they are represented in the Tree as objects named **Coordinate System**, **Coordinate System 2**, **Coordinate System 3**, and so on.
- Toggling the value of **Used By Fluid Surface** from **Yes** to **No** will result in the following behaviors:

  - If you switch from **Yes** to **No**, the **Fluid Surface** object will be hidden.
  - If you switch from **No** to **Yes**, the **Fluid Surface** object will be inserted into the Tree Outline, and the virtual body will become the primary body.

- If a group of fluid surfaces is being used to help define a virtual body, its suppression status is inherited from the virtual body and is read-only.
- You should not use a group of fluid surfaces to help define a virtual body if one or more internal baffles (**Surface Body** with free edges) is located inside the flow volume void where you specified the material point. These baffles do not resolve properly in Ansys Fluent.
- For flow volume extraction, make sure that capping faces have been created in your CAD package. Capping faces must be selected manually when defining virtual bodies. See Using the Extend to Connection Option with Fluid Surface Grouping (p. 386) for an example.
- Using one surface body to cap multiple inlets/outlets is not supported. Make sure that each opening is capped with at least one face.
- The mesh around the edges of a capping surface must be properly resolved both by size and feature to ensure that no leakage occurs.
- Contact sizing might be used to close gaps smaller than 1/4 of the minimum size. Note that contact sizing will not properly recover features around the gap, therefore you should exclude inflation in this area.
- It is not recommended to use contact sizing to close a gap if faces in the connection are already included in the sharp angle control.
- Contact sizing tools are not supported on baffle surfaces or capping surfaces.
- If you create a fluid body using Boolean operations from solid volumes, but shared topology fails, the recommended action is to not use this geometry configuration in assembly meshing. Instead remove the fluid volumes and add a material point to extract the fluid volume inside meshing. Also see Handling of Interfaces Between Bodies in Assembly Meshing (p. 383).
- The **Mesh** cell state will go out-of-date and will require updating if:
  - You suppress/unsuppress a virtual body.
  - You delete a virtual body.
  - You change the location of the material point that you selected for a virtual body.
  - You select a different material point for a virtual body.
- Named Selection names for internal face zones are not interpreted. In cases where two enclosed voids share a face, the face zone is assigned type *WALL* automatically regardless of whether a Named Selection has been defined for the face. In these cases, the mesh generation cannot cross any boundary so you must define a virtual body with material point for each flow volume void in order for the volumes to be meshed.
- When an assembly mesh is exported to Ansys Fluent, **Virtual Body Group** names and **Virtual Body** names are handled the same as part names and body names respectively. The zone naming

rules that are applied to *real* part and body names are also applied to **Virtual Body Group** names and **Virtual Body** names. For example, part "part" and body "solid body" will result in a zone name of "part-solid_body." Similarly, **Virtual Body Group** "virtual_body_group" and **Virtual Body** "virtual_body" will result in a zone name of "virtual_body_group-virtual_body." Refer to Fluent Mesh Export (p. 43) for details.

For fluids of different properties (for example, air, water, etc.), it is recommended that you create virtual bodies within different virtual body groups to make sure the fluid zones are separated inside Ansys Fluent.

- As described in Define Connections the Mechanical help, connections can be contact regions, joints, mesh connections, and so on. To aid in picking faces related to flow volumes, you can use the **Extend to Connection** option on the **Extend Selection** drop-down menu. **Extend to Connection** searches for faces that are adjacent to the current selection as well as all faces that are adjacent to each of the additional selections within the part, up to and including all connections on the selected part. For example, if you are using a group of fluid surfaces to help define a virtual body, you can generate connections, pick one face on each body of the flow volume, and then select **Extend to Connection**. As a result, the faces related to the flow volume are picked. See Using the Extend to Connection Option with Fluid Surface Grouping (p. 386) for an example.

  The extent of the faces that will be included when **Extend to Connection** is used depends greatly on the current set of connections, as defined by the specified connections criteria (for example, **Connection Type**, **Tolerance Value**, and so on). By modifying the criteria and regenerating the connections, a different set of faces may be included. Refer to Common Connections Operations for Auto Generated Connections in the Mechanical help for more information.

- **Extend to Connection** is also helpful in cases where you want to scope a size locally to the faces of a virtual body. For example, consider a case in which you want to assign a smaller curvature angle to a fluid region and you want to keep the solid mesh (**Keep Solid Mesh** (p. 167) is set to **Yes**). When **Physics Preference** is **CFD**, the default curvature angle of 18 degrees has been determined to be suitable for use in flow simulations. For solids, the angle can probably be as large as 30-40 degrees. Hence, by setting the global **Curvature Normal Angle** (p. 109) to 30 degrees and scoping a curvature normal angle (p. 261) of 18 degrees locally to the faces of the virtual body (after roughly picking the faces and using **Extend to Connection** to find adjacent faces), you can avoid unnecessary mesh refinement in the solid parts.

Handling of Interfaces Between Bodies in Assembly Meshing

In cases where fluid extraction is successful in the DesignModeler application (or any other CAD package) but Shared Topology fails, and you plan to solve a conjugate heat transfer problem and hence keep the solid mesh (p. 167), two problems may occur because you now have two boundaries all along the surface of the fluid domain:

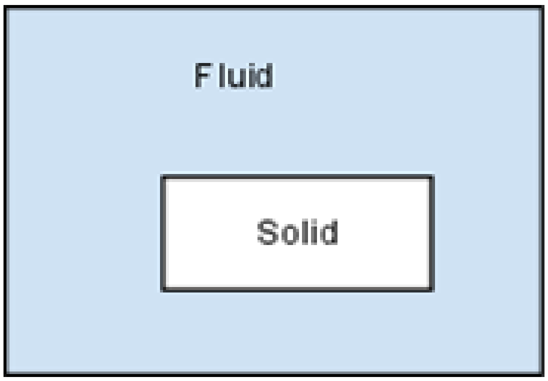
- The Shared Topology failure indicates that the fluid domain does not match the original boundaries very well, which may lead to bad quality cells.
- Any Named Selection created on the solid bodies may be lost in cases where the corresponding fluid wall is present. Due to the conformal mesh that will be generated by assembly meshing, only one of the geometry faces in contact will get surface mesh elements associated to it. Alternatively, the Named Selection may be added to the fluid wall as well.

To avoid these problems, you can suppress the fluid volume and use material point-based extraction inside assembly meshing.

For the example shown below, in the case of a Shared Topology failure, you could:

1. Keep the Boolean subtraction operation but suppress the solid body and use a virtual body instead.

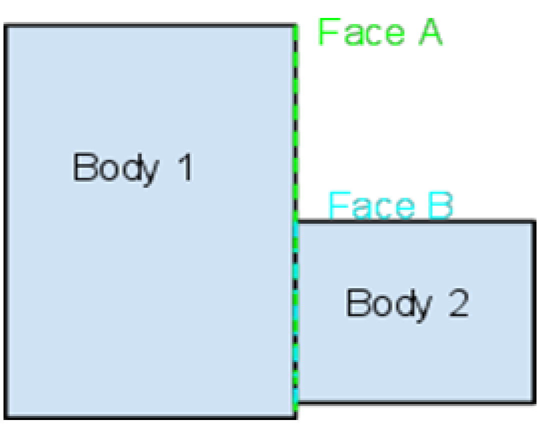
2. Remove the Boolean subtraction operation in the DesignModeler application or CAD package so that the solid overlaps the same volume as the fluid (the meshing will do the volume extraction).



Alternatively, if the solids/fluids only partially touch as in the case shown below, mesh at the interface between **Body 1** and **Body 2** may get associated to either **Face A** or **Face B**. If you need to define a Named Selection for **Face B**, it should include both **Face A** and **Face B** to ensure proper association.

Note:

You must use a similar approach when applying inflation to shared faces between a solid body and a fluid body in cases involving multiple parts. That is, select both sets of overlapping faces when defining the local inflation control. See Applying Local (Scoped) Inflation Controls and Regenerating the Inflation Mesh (p. 401) for related information.



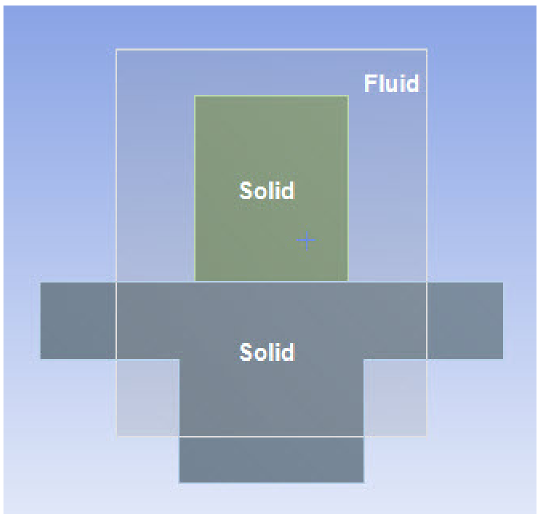
Because this approach may result in the Named Selection including the larger area (all of **Face A**), you may need to split out **Face A** with imprints to get the proper topology.

Using a Virtual Body to Separate a Fluid Region for Meshing

In cases where a solid spatially divides a fluid into multiple regions and you want to recover the mesh in only one region, you can use a virtual body and material point to define the region to be meshed.

For example, in Figure 187: Solid Bodies Dividing a Fluid Body (p. 385), the image on the left shows a mesh group (p. 389) that consists of two solid bodies, and which divides a fluid body into multiple regions. The image on the right shows the corresponding mesh that is obtained using the **CutCell** algorithm.

Figure 187: Solid Bodies Dividing a Fluid Body



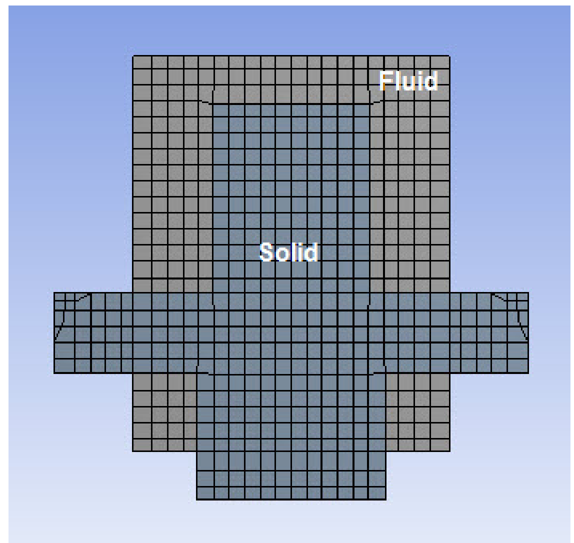
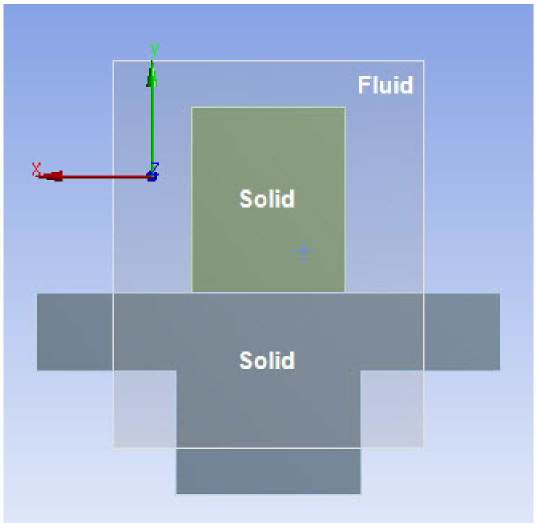
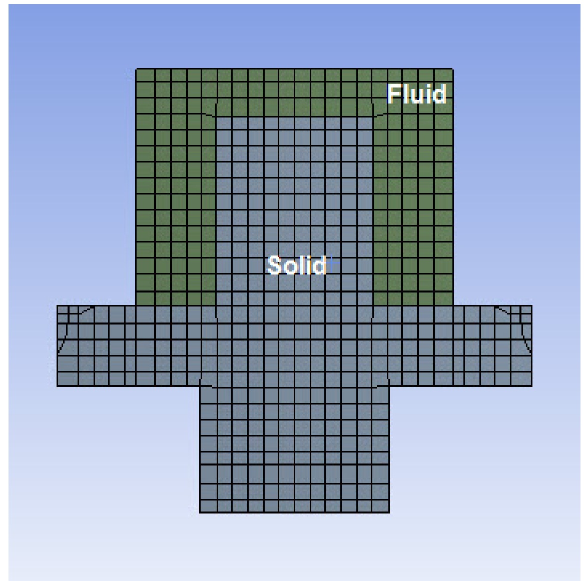


Figure 188: Virtual Body Defined to Separate Fluid Region (p. 386) shows the same model, but in this case a virtual body and material point have been defined to further separate the fluid region. The image on the right shows the corresponding mesh that is obtained using the **CutCell** algorithm.

Figure 188: Virtual Body Defined to Separate Fluid Region

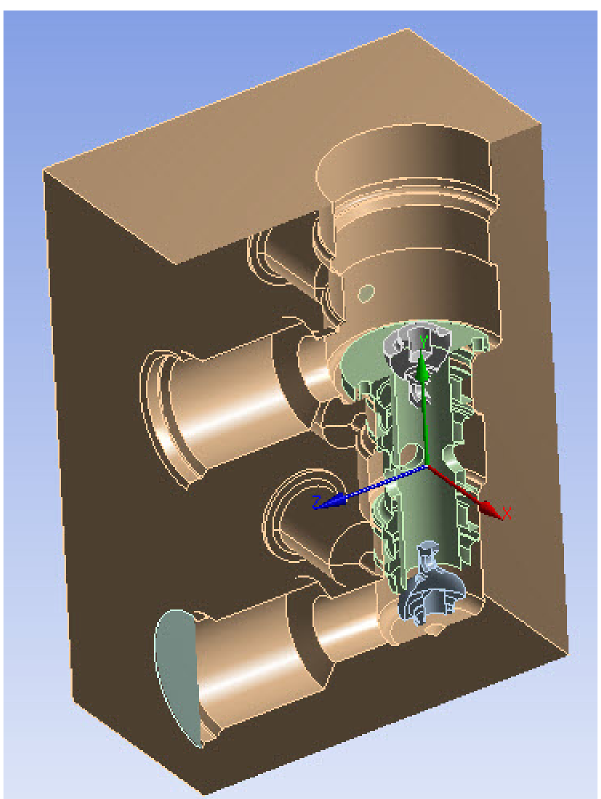




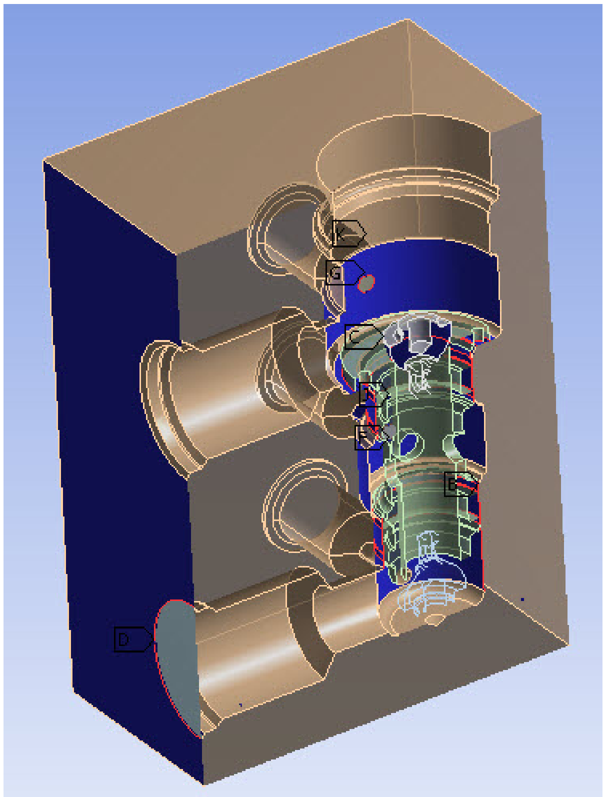
Using the Extend to Connection Option with Fluid Surface Grouping

This example illustrates how you can use the **Extend to Connection** option and fluid surface grouping together to solve assembly meshing problems.

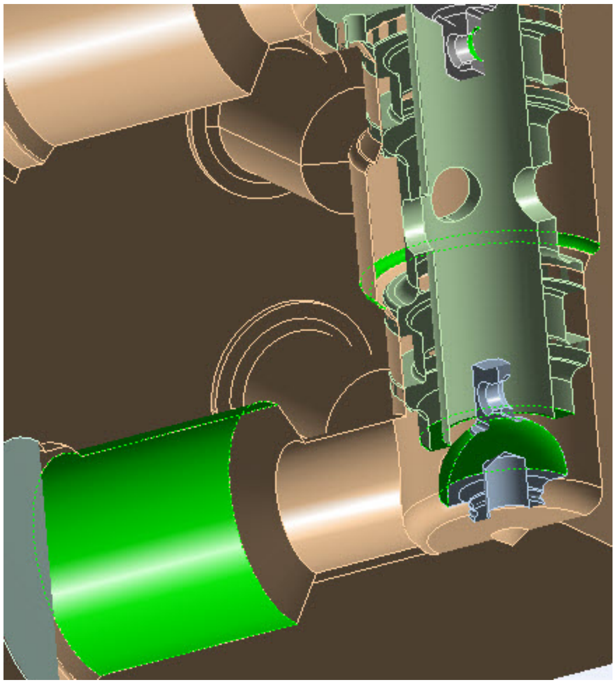
The first figure shows a model of an aero valve, for which a virtual body is being used to represent the flow volume. Notice the **Material Point** that is inside the virtual body.



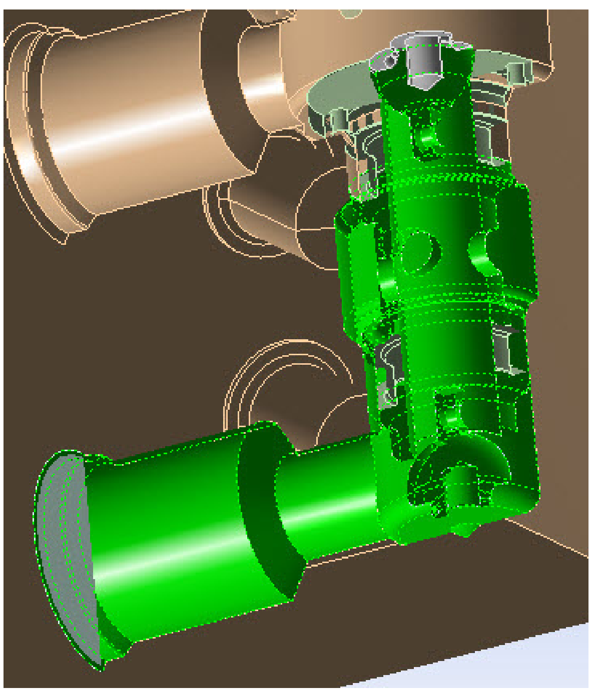
Automatic connections were generated for the model and are highlighted in the figure below.



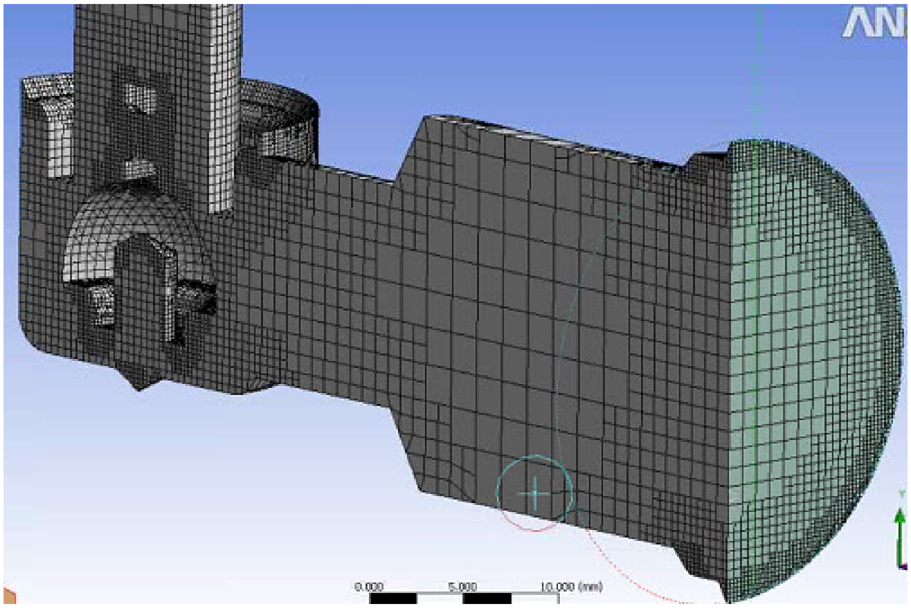
As described in the procedure above, to use a group of fluid surfaces to help define a virtual body, all of the faces that should be members of the group must be selected and assigned as the **Faces To Group** in the Details View. Using the face picker to select all of the faces manually in the **Geometry** window may be difficult. However, by using the **Extend to Connection** option, you need to select only one face on each body of the flow volume (that is, one face on each of the domains that will be "closed" by the connections). In the figure below, five faces have been picked.



After you select one face on each body of the flow volume, select the **Extend to Connection** option. In this example, the figure below shows the faces that are selected automatically after the **Extend to Connection** option is chosen. The option captures all faces that will "touch" the flow volume, except for any capping faces. The capping faces, such as the inlet face, must be picked manually. After all capping faces are picked, you can click the **Faces To Group** field in the Details View to apply your selection and then proceed with generating the mesh.



The figure below shows a view of the resulting mesh, with the inlet on the right.



Defining Mesh Groups

You can use mesh grouping to identify bodies that should be grouped together for assembly meshing algorithms.

Mesh Group objects appear in the Tree Outline under the **Mesh** object. The purpose of a **Mesh Group** object is to unite bodies that have the same properties and do not have gaps between them. For example, this approach may be useful if you decomposed a fluid volume into multiple volumes for hybrid meshing but now want to use the **CutCell** approach to mesh the same model. With **CutCell**, decomposed volumes are no longer needed. Grouping tells the mesher to treat certain solid parts as one part. You can define a mesh group to merge volumes, and the mesh generated on the combined parts (the group) will be associated with the mesh of the selected primary body.

Note:

Only solid (3D) bodies can be selected for grouping. Surface bodies cannot be selected.

Defining a Mesh Group

The steps for defining a mesh group are presented here.

1. To insert a **Mesh Group** object into the Tree, highlight the **Mesh** object (or its **Mesh Grouping** or **Mesh Group** child object if any exist) and then do one of the following:

   - Select **Mesh Control> Mesh Group** on the **Mesh** context toolbar.
   - Click the right mouse button on the object you highlighted and select **Insert> Mesh Group** from the context menu.

   These methods insert a **Mesh Group** object beneath the **Mesh Grouping** object. The **Mesh Grouping** object is inserted automatically when the first **Mesh Group** object is inserted.

2. To define and make adjustments to an individual group, click a **Mesh Group** object and use the following controls. These controls appear in the Details View of the **Mesh Group** object.

   - **Bodies To Group**: In the **Geometry** window, select the bodies that should be members of the group.
     - All bodies within a group, including the **Primary Body**, should be of the same type (**Fluid** or **Solid**, as defined by the **Fluid/Solid** material property (p. 379)). Otherwise, unexpected results may occur.
     - Surface bodies cannot be selected for grouping.
   - **Primary Body**: In the **Geometry** window, select the body that should act as the primary of the group. The primary body is the body to which all mesh of the group members will be associated. By default, the first body that is selected for **Bodies To Group** is the **Primary Body**.
   - **Priority**: Determines which group will claim cells in cases where groups overlap. The priority is initially based on the rule: the smaller the volume, the higher the priority.
   - **Suppressed**: Toggles suppression of the selected group. The default is **No**. If set to **Yes**, the group will be suppressed.

Setting Global Assembly Meshing Options

The next step in the assembly meshing workflow is to define global assembly meshing options:

1. Select the appropriate option for Feature Capture (p. 166).
2. Select the appropriate option for Tessellation Refinement (p. 166).
3. Select the appropriate option for Keep Solid Mesh (p. 167).

Defining Sharp Angle Controls

To ensure that the desired features are captured in the assembly mesh, you can use the Sharp Angle Tool to control the capture of features with sharp angles, such as the edge of a knife or the region where a tire meets the road. It can also be used for improved feature capturing in general. Refer to Sharp Angle Tool (p. 296) for details.

Setting Sizing Options

The next step in the assembly meshing workflow is to set sizing options.

Of great importance to assembly meshing are the values of the Curvature Min Size (p. 108) and Proximity Min Size (p. 110) options. Make sure that the values of these options truly represent the smallest size that you want the curvature and proximity size functions to capture and that they are set appropriately before invoking the **Find Thin Sections** (p. 393) or **Find Contacts** (p. 395) features. By default, these features operate based on the smaller of these two minimum size values.

When either the **CutCell** method is selected, or the **Tetrahedrons** method is selected and **Physics Preference** is set to **CFD**, **Capture Curvature** (p. 102) is set to **Yes** by default. If you set **Capture**

Proximity (p. 102) to **Yes**, the Proximity Size Function Sources (p. 110) control appears. Its value determines whether regions of proximity between faces, edges, or both are considered when proximity-based sizing calculations are performed.

The remainder of this section describes points to remember when setting sizing options specifically for assembly meshing algorithms. Refer to Sizing Options (p. 100) for details about setting additional sizing options. Also refer to Handling Assembly Meshing Failures Due to Min Size (p. 548).

Effect of the Smoothing Option:

The setting of the Smoothing (p. 123) option controls the quality threshold at which the assembly meshing algorithm will start smoothing. The table below presents the **Smoothing** options that are available in the Meshing application (**Low**, **Medium**, and **High**) and their corresponding quality limits. All cells below the specified quality limit will be considered for improvement.

Note:

Orthogonal quality in the Meshing application is equivalent to Inverse Orthogonal Quality in Ansys Fluent Meshing, except that the scale is reversed:

Inverse Orthogonal Quality = 1 – Orthogonal Quality

Smoothing Option	Orthogonal Quality Limit (without Inflation)	Orthogonal Quality Limit (with Inflation)
Low	0.1	0.01
Medium	0.15	0.05
High	0.2	0.1

Refer to Orthogonal Quality (p. 142) for more information.

Note:

When the **Tetrahedrons** algorithm is being used and **Smoothing** is set to **Low**, only minimal surface mesh improvement occurs, which provides a faster turnaround for large cases having many faces. Thus, you can choose between higher quality mesh vs. faster turnaround.

Rules for Computing Min Size and Max Size:

Assembly meshing algorithms use the following rules for computing **Min Size** and **Max Size** values, where the value of **Min Size** is the smaller of the two minimum size values (**Curvature Min Size** (p. 108) or **Proximity Min Size** (p. 110)). In general, the **Max Size** is set based on the **Element Size** defined prior to turning on Assembly meshing. If sheets exist in the model, the default **Max Size** will be equal to the default **Element size**. If sheets do not exist in the model, the default **Max Size** will be equal to 2 times the default **Element Size**. The **Max Size** and **Min Size** values will be set such that a 2^n ratio is maintained because assembly meshing techniques use Octree subdivision. Therefore:

1. Default **Min Size** = Default **Max Size** / 128.

   Note:

   Assembly meshing does not use the **CFD Min Size Factor**.

2. The ratio between **Min Size** and **Max Size** can be any one of the powers of two from 0 to 13 (shown in the table below). Thus, 14 levels of difference between **Min Size** and **Max Size** are allowed:

2^0 = 1	2^7 = 128
2^1 = 2	2^8 = 256
2^2 = 4	2^9 = 512
2^3 = 8	2^10 = 1024
2^4 = 16	2^11 = 2048
2^5 = 32	2^12 = 4096
2^6 = 64	2^13 = 8192

   Note:

   The value of **Max Size** cannot be greater than (2^13 * **Min Size**). If the value you set for **Max Size** is too high, **Max Size** is set to its maximum limit (2^13 * **Min Size**) automatically.

3. **Max Size** may be converted to the power of two that is nearest to the *intended* value of **Max Size**, where the intended value of **Max Size** is either the default value or the user input value of **Max Size**.

Consider this example, which shows the **Min Size** and **Max Size** values at each step in the given sequence:

1. Select an assembly meshing algorithm. Default **Min Size** = 5, so default **Max Size** = 5 * 128 = 640.

   Min Size, **Max Size** = Default(5.0), Default(640.0)

2. If you set **Min Size** to any one of {0.0625, 0.125, 0.25, 0.5, 1, 2, 4, ... 512, ... 2^n}, then **Max Size** = 512 (not default). For example, if you set **Min Size** = 1, (**Min Size**, **Max Size**) = (1, 512).

3. Set **Max Size** = 40,000, **Max Size** = 8192 based on the maximum limit rule described above.

   Min Size, **Max Size** = 1, 8192

4. Set **Max Size** = 48.5, **Max Size** is converted to the nearest power of two, so **Max Size** = 64.

   Min Size, **Max Size** = 1, 64

5. Set **Max Size** = 47.5, **Max Size** is converted to the nearest power of two, so **Max Size** = 32.

Min Size, **Max Size** = 1, 32

6. Set **Min Size** = 0.25, **Max Size** = 32.

   Min Size, **Max Size** = 0.25, 32

7. Set **Min Size** = 0, **Min Size** = Default(5.0) and **Max Size** = 40.

   Min Size, **Max Size** = Default(5.0), 40

8. Deselect assembly meshing, user input values for **Min Size** and **Max Size** are 0, 47.5 respectively.

   Min Size, **Max Size** = Default(0.0123), 47.5

Finding Thin Sections

You can use the **Find Thin Sections** option as a diagnostics tool to locate thin sections in an assembly. By default, **Find Thin Sections** looks for all sections with a size equal to or less than the current **Curvature Min Size** (p. 108)/**Proximity Min Size** (p. 110) (whichever is smaller).

Find Thin Sections uses the Mechanical application's technology for automatic connection detection to provide these diagnostics for assembly meshing. The technology generates diagnostics data in the form of contact regions for the thin sections automatically. Each contact region is named appropriately based on the names of the items in the **Geometry** branch of the tree that make up that contact region (for example, **`Bonded - Solid_3 To Solid_3`**).

Be aware that from the perspective of the Mechanical application, the connection detection technology is dependent on physics properties. When the technology is used for assembly meshing diagnostics, physics properties are not considered.

Connection detection options that are of particular importance to finding thin sections for assembly meshing are mentioned below. They are set optimally by default to provide assembly meshing diagnostics and should not be changed.

If you are interested in learning more about connection detection settings and how they are used by the Mechanical application, refer to Common Connections Operations for Auto Generated Connections in the Mechanical help.

The **Find Thin Sections** option is available only when assembly meshing (p. 367) algorithms are being used. Use of the **Find Thin Sections** option is an optional step in the assembly meshing workflow.

The steps for using the **Find Thin Sections** option are presented here.

1. Make sure that **Curvature Min Size** (p. 108)/**Proximity Min Size** (p. 110) is set appropriately.

2. Highlight the **Mesh** object or **Connections** object in the Tree Outline.

3. Do one of the following:

   - Click the right mouse button on the object and select **Find Thin Sections** from the context menu.

- Click the right mouse button on the **Geometry** window and select **Find Thin Sections** from the context menu.

Either of these methods inserts an **Assembly Meshing Thin Sections** folder beneath the **Connections** object and populates the folder with appropriately named contact regions.

Note:

If **Find Thin Sections** does not find any thin sections that meet the tolerance criteria, the folder is inserted but it will be empty.

The figure below shows the default settings that appear in the Details View when the folder is selected. *The defaults should not be changed.*

- The **Tolerance Value** is based on the value of **Curvature Min Size** (p. 108)/**Proximity Min Size** (p. 110), which was set to 0.250 mm for this example.
- **Face/Face** is set to **Yes** to detect thin sections between faces.
- **Group By** is set to **Parts**, meaning connection faces that lie on the same parts will be included in a single connection object.
- **Search Across** is set to **Anywhere**, enabling automatic connection detection regardless of where the geometry lies. Contacts that occur across bodies are deleted automatically (see note below).

Details of "Assembly Meshing Thin Sections"	
Definition	
Connection Type	Contact
Scope	
Scoping Method	Geometry Selection
Geometry	All Bodies
Auto Detection	
Tolerance Type	Value
Tolerance Value	0.25 mm
Use Range	No
Face/Face	Yes
Face/Edge	No
Edge/Edge	No
Priority	Include All
Group By	Parts
Search Across	Anywhere

Note:

- If you change **Min Size**/**Proximity Min Size**, you must rerun the option to regenerate the data. Each time you rerun the option an additional folder is added to the Tree Outline (**Assembly Meshing Thin Sections 2**, **Assembly Meshing Thin Sections 3**, and so on). The lowest folder in the tree contains the most recent data.

- The **Find Thin Sections** option automatically deletes contacts that occur across bodies (for example, from Body_A to Body_B).

Finding Contacts

You can use the **Find Contacts** option as a diagnostics tool to locate contacts in an assembly. By default, **Find Contacts** detects face-edge contacts using a tolerance based on the current **Curvature Min Size** (p. 108)/**Proximity Min Size** (p. 110) (whichever is smaller) as follows:

- Contacts between 10% of **Curvature Min Size**/**Proximity Min Size** and 0

The search for contacts passes feature information down to the meshing algorithm to ensure that desired features are preserved. In addition to the edges that are determined to be feature edges based on the assembly meshing angle criterion, edges referenced in contact regions are also considered as feature edges in assembly meshing. **Find Contacts** is particularly useful for assemblies in which fillets of bodies are adjacent to other bodies, forming a sharp angle, as it will preserve the edges of these fillets independent of the feature angle settings.

Find Contacts uses the Mechanical application's technology for automatic connection detection to provide these diagnostics for assembly meshing. The technology generates diagnostics data in the form of contact regions for the contacts automatically. Each contact region is named appropriately based on the names of the items in the **Geometry** branch of the tree that make up that contact region (for example, **`Bonded - Solid_2 To Solid_4`**).

Be aware that from the perspective of the Mechanical application, the connection detection technology is dependent on physics properties. When the technology is used for assembly meshing diagnostics, physics properties are not considered.

Connection detection options that are of particular importance to finding contacts for assembly meshing are mentioned below. They are set optimally by default to provide assembly meshing diagnostics and should not be changed.

If you are interested in learning more about connection detection settings and how they are used by the Mechanical application, refer to Common Connections Operations for Auto Generated Connections in the Mechanical help.

The **Find Contacts** option is available only when assembly meshing (p. 367) algorithms are being used.

The steps for using the **Find Contacts** option are presented here.

1. Make sure that **Curvature Min Size** (p. 108)/**Proximity Min Size** (p. 110) is set appropriately.
2. Highlight the **Mesh** object or **Connections** object in the Tree Outline.
3. Do one of the following:
   - Click the right mouse button on the object and select **Find Contacts** from the context menu.
   - Click the right mouse button on the **Geometry** window and select **Find Contacts** from the context menu.

Either of these methods inserts an **Assembly Meshing Contacts** folder beneath the **Connections** object and populates it with appropriately named contact regions.

Note:

If **Find Contacts** does not find any contacts that meet the tolerance criteria, the folder is inserted but it will be empty.

The figure below shows the default settings that appear in the Details View when the **Assembly Meshing Contacts** folder is selected. *The defaults should not be changed.*

- The **Tolerance Value** is set to 10% of the value of **Curvature Min Size** (p. 108)/**Proximity Min Size** (p. 110) which for this example was set to 0.250 mm.
- **Face/Edge** is set to **Yes** to detect contacts between faces and edges.
- **Group By** is set to **Bodies**, meaning connection faces and edges that lie on the same bodies will be included in a single connection object.
- **Search Across** is set to **Bodies**, enabling automatic connection detection between bodies.

Details of "Assembly Meshing Contacts"	
Definition	
Connection Type	Contact
Scope	
Scoping Method	Geometry Selection
Geometry	All Bodies
Auto Detection	
Tolerance Type	Value
Tolerance Value	2.5e-002 mm
Use Range	No
Face/Face	No
Face/Edge	Yes
Edge/Edge	No
Priority	Include All
Group By	Bodies
Search Across	Bodies

Note:

If you change **Min Size**/**Proximity Min Size**, you must rerun the option to regenerate the data. Each time you rerun the option an additional folder is added to the Tree Outline (**Assembly Meshing Contacts 2**, **Assembly Meshing Contacts 3**, and so on). The lowest folder in the tree contains the most recent data.

Generating the Mesh

The next step in the assembly meshing workflow is to generate the base mesh:

1. Select the **Mesh** object or any mesh control object.
2. Right-click to display the context menu, or choose the **Mesh** drop-down menu from the toolbar.
3. Select **Generate Mesh** (p. 486) in the menu. The part is meshed. The mesh is displayed when you select the **Mesh** object.
4. Examine the mesh to determine whether it is acceptable. If it is acceptable, you can proceed to the next section. If it is unacceptable, try adjusting the settings you specified above and then regenerate the mesh.

Note:

- Although **Generate Mesh** (p. 486) is valid, its options to generate the mesh on selected bodies or parts are invalid and therefore inaccessible.
- If you make changes to inflation settings after generating the mesh using an assembly meshing algorithm, each subsequent re-mesh begins with the initial (cached) mesh and inflates from there.

As assembly meshing algorithms are patch independent, the generated mesh goes through the following validation checks to confirm the mesh is valid and is not missing mesh at any location. During validation the assembly meshing algorithms consider factors such as number of elements/faces/nodes associated to the mesh (including initial mesh vs. inflated mesh), number of elements/faces/nodes associated to entities contained in Named Selections, orthogonal quality measures, and number of inflation layers:

- The following conditions must be met, or the returned mesh is considered to be invalid (p. 538) and an error is issued:
  - Orthogonal quality must be > 0 for all volume elements.
  - No leaks (p. 549) were detected during flow volume extraction.
  - No failures related to user-defined objects (such as **Virtual Body**, **Fluid Surface**, or **Mesh Group** objects) occurred.
- If any of the following conditions exist, the returned mesh is considered to be valid but a warning is issued:
  - The initial mesh is returned but inflation failed.
  - Orthogonal quality for one or more volume elements is <=0.05.
  - A Named Selection does not contain any elements or exterior faces.

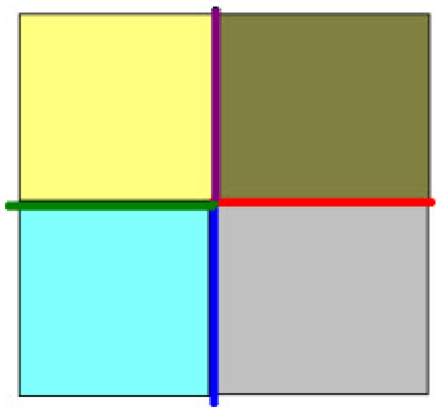
Note:

- A body that does not contain any elements may result in a warning or an error, depending on the circumstances:

  – When the initial mesh is returned but a body has no mesh associated to it, a warning is issued for the failed body. In such cases, the state of the body in the Tree Outline may indicate that the body is meshed.

  – When generation of the inflation mesh results in deletion of mesh from the initial (cached) mesh, an error is issued.

- The assembly meshing algorithms may not be able to support cases where surfaces form a cross. For example, the four bodies in the image below form a cross as depicted by the thick lines. In such cases, the feature recovery/surface recovery along the center line (shown as the center point in the 2D image below) may be poor.



- Using a different facet quality in the DesignModeler application could have an impact on the performance and quality of assembly meshing. Increasing the facet quality could improve feature capturing, reduce leakage, and improve quality (refer to Avoiding Bad Feature Capturing in Assembly Meshing (p. 548)), but at the same time it could increase the assembly meshing time, particularly for tetrahedron.

- Refer to Meshing: Troubleshooting (p. 535) for additional tips and strategies for handling problems that may occur during assembly meshing.

Applying Contact Sizing

When used with assembly meshing, contact sizing provides a method for closing leaks discovered during meshing. Typically, leakage occurs if any contact is larger than 1/10 of the local minimum size. If a leak is discovered during meshing and the gap is up to 1/3 of the local mesh size, you can close the gap at the location of contact by using a two-step process. First, scope a new contact to the existing **Find Contacts** (p. 395) list by picking all of the faces on one side and all of the faces (or edges) on the other side of the leak. Second, scope a contact sizing control to this new contact.

For assembly meshing, the default **Element Size** used by contact sizing controls is *minimum_size/5*, where *minimum_size* is the value of **Curvature Min Size** (p. 108) or **Proximity Min Size** (p. 110), whichever is smaller. This **Element Size** is appropriate for the vast majority of cases. However, you may need to adjust the value of **Element Size** in these situations:

- If hard sizing that is smaller than **Curvature Min Size**/**Proximity Min Size** is applied in the vicinity of the contact region, you may need to reduce the **Element Size** used by contact.

- If local sizing that is significantly larger than **Curvature Min Size**/**Proximity Min Size** is applied in the vicinity of the contact region, you may need to increase the **Element Size** used by contact.

Note:

- Each face-edge contact region in a contact sizing control consists of one or more faces and one or more edges. That is, in some cases a contact region may consist of one face to one edge in contact, while in others it may consist of many faces to many edges.
- Use care in specifying **Element Size** for contact sizing. A value that is too large may lead to inaccurate geometry, while a value that is too small may fail to close the leak. A good guideline is to specify an **Element Size** that is 1/5 of the local element size.
- In some cases the gap at the location of contact is closed but the mesh in that location will not be well resolved. This occurs because there are no features to be captured at the location of the gap, which in turn causes the edges to become jagged.

You can apply contact sizing using either of the following methods:

Method 1

1. In the Tree Outline, select the **Mesh** object or any mesh control object.
2. Do one of the following:
   a. Choose **Contact Sizing** from the **Mesh Control** drop-down menu.
   b. Right-click the selected object and select **Insert> Contact Sizing** from the context menu.

   A new **Contact Sizing** control is added to the Tree Outline.
3. In the Details View, select a specific **Contact Region** to scope it to the sizing control.
4. Enter a value for **Element Size**. By default, **Element Size** is set equal to the value of *minimum_size/5*.

Method 2

1. In the Tree Outline, drag a contact region from the **Assembly Meshing Contacts** (p. 395) folder onto the **Mesh** object.

   A new **Contact Sizing** control is added to the Tree Outline.
2. In the Tree Outline, highlight the new **Contact Sizing** control.

3. Enter a value for **Element Size**. By default, **Element Size** is set equal to the value of *minimum_size/5*.

Note:

- The **Element Size** option for **Type** specifies an absolute size for the contact sizing. The **Resolution** option for **Type**, which is sometimes used to specify a relative size in contact sizing controls (p. 263), is not supported for assembly meshing algorithms.
- If contact sizing is applied to entities on a body that is scoped to a body of influence, the contact sizing is ignored.
- If you apply contact sizing and a sharp angle control (p. 296) to the same face, the sharp angle features may not be captured accurately.

Setting Global Inflation Controls

After you determine the base mesh is acceptable, the next step in the assembly meshing workflow is to set global inflation controls:

Note:

When an assembly meshing algorithm is being used, a mixture of global (automatic **Program Controlled**) and local (scoped) inflation is *not supported*. See Selecting an Assembly Mesh Method (p. 375) for details about behaviors resulting from this restriction and guidelines for using global vs. local inflation.

---

1. Set Use Automatic Inflation (p. 147) to Program Controlled (p. 148).

   The **All Faces in Chosen Named Selection** option is not supported for assembly meshing algorithms. If this option is specified and you select an assembly meshing algorithm, the option will be changed automatically to **Program Controlled** and a warning will be issued.

2. Set additional global inflation controls as desired. Refer to Inflation Group (p. 145) for descriptions.

Note:

- The **Fluid/Solid** material property for virtual bodies is always set to **Fluid** (read-only). With **Program Controlled** inflation, faces on real solid bodies will inflate into the virtual bodies. The **Fluid/Solid** designation on real bodies (p. 379) will be respected (that is, faces on real fluid bodies will inflate into the fluid region, but the solid region will not be inflated).
- Attempting to grow thicker prism layers in areas where the aspect ratio of the base to the prism cap is very large may result in an invalid (p. 535) mesh. In such cases (for example, external flow problems), you should use aspect ratio based growth to avoid problems with invalid meshes.

Generating the Inflation Mesh

The next step in the assembly meshing workflow is to generate the inflation mesh:

1. Select the **Mesh** object or any mesh control object.
2. Right-click to display the context menu, or choose the **Mesh** drop-down menu from the toolbar.
3. Select **Generate Mesh** (p. 486) in the menu. The inflation mesh is generated and is displayed when you select the **Mesh** object.
4. Examine the inflation mesh, which was obtained using automatic **Program Controlled** inflation. If it is acceptable, you can proceed directly to Exporting the Mesh (p. 404). If it is unacceptable, you can either adjust the global inflation controls described above or proceed to the next section to set local inflation controls, and then regenerate the inflation mesh.

Note:

- For assembly meshing algorithms, inflation is a post process for the mesher after it has created the hexahedron or tetrahedron elements. A benefit of this approach is that the hexahedral/tetrahedral mesh does not have to be generated each time inflation options are changed. You can add/delete/modify/suppress your inflation settings, and the meshing process will begin with the initial (cached) mesh and inflate from there. This is important in case you need to make adjustments to obtain the desired results.
- In cases in which Smoothing (p. 123) is set to **High** and **CutCell** meshing is being used, additional smoothing of inflation layers occurs. This may slow down the prism generation process.

Applying Local (Scoped) Inflation Controls and Regenerating the Inflation Mesh

If the inflation mesh that was generated using global (automatic **Program Controlled**) inflation is acceptable, you can proceed directly to Exporting the Mesh (p. 404). If it is unacceptable, you can apply local inflation controls to try to improve the mesh.

Note:

- When an assembly meshing algorithm is being used, a mixture of global and local inflation is *not supported*. See Selecting an Assembly Mesh Method (p. 375) for details about behaviors resulting from this restriction and guidelines for using global vs. local inflation.
- If mesh groups (p. 389) are present, you must scope local inflation controls to either the primary body or to all bodies in the mesh group. Otherwise, mesh generation fails. Alternatively, you can set the **Fluid/Solid** material property (p. 379) settings for the bodies to **Fluid** and use global (**Program Controlled**) inflation instead. Refer to Program Controlled (p. 148) for more information.

To add boundary layers to a face for assembly meshing:

1. In the global inflation controls, set Use Automatic Inflation (p. 147) to None (p. 147). Otherwise, you will not be able to insert local inflation.

2. Optionally, select the desired bodies in the **Geometry** window. These are the bodies that you want to scope inflation to.

3. Use either of these methods to insert the inflation control:

   - Click **Mesh Control** on the toolbar and choose **Inflation** from the menu.

   - Right-click in the **Geometry** window and choose **Insert** > **Inflation** from the menu.

4. If you selected the bodies in step 2, go directly to step 5. If not, use either of these methods to scope inflation to the desired bodies:

   - In the Details View, set **Scoping Method** to **Geometry Selection**, pick the bodies in the **Geometry** window, and click the **Geometry** field in the Details View to **Apply**.

   - In the Details View, set **Scoping Method** to **Named Selection**, and select a Named Selection from the **Named Selection** drop-down.

5. Change the value of the **Suppressed** control if desired.

   By default, the value of **Suppressed** is **No**. If you change the value to **Yes**, this inflation control has no effect on the mesh. In addition, an **Active** control with a read-only setting of **No, Suppressed** appears under the **Suppressed** control when **Suppressed** is set to **Yes**.

6. Use either of these methods to specify the inflation boundaries (that is, select the faces that you want the inflation layers to grow away from):

   - In the Details View, set **Boundary Scoping Method** to **Geometry Selection**, pick the faces in the **Geometry** window, and click the **Boundary** field in the Details View to **Apply**.

   - In the Details View, set **Boundary Scoping Method** to **Named Selections**, select a Named Selection from the **Boundary** drop-down, and press **Enter**.

Note:

– To select multiple Named Selections to be used as inflation boundaries, press and hold the **Ctrl** key while selecting the Named Selections from the **Boundary** drop-down, and then press **Enter**.

– If none of the predefined Named Selections include the correct topology to be used as an inflation boundary, no Named Selections will be available in the drop-down. For assembly meshing algorithms, the correct topology is always a face because 2D inflation is not supported (see notes below). If any 2D inflation controls are defined prior to selection of an assembly meshing algorithm, they are suppressed when an assembly meshing algorithm is selected.

– For assembly meshing algorithms only, the scoped body and the face that you select to be the inflation boundary do not have to be on the same part. In other words, the face does not have to be attached to the body. Such controls are invalid for part/body level meshing, and will be flagged as invalid if any have been defined and you subsequently deselect assembly meshing. The scoped body will be retained but you will have to select a new inflation boundary.

– To apply inflation to shared faces between a solid body and a fluid body in cases involving multiple parts, you must select both sets of overlapping faces when defining the local inflation control.

7. Specify additional inflation options, as desired, in the Details View. For descriptions of additional options, refer to Inflation Group (p. 145).

8. Select the **Mesh** object or any mesh control object.

9. Right-click to display the context menu, or choose the **Mesh** drop-down menu from the toolbar.

10. Select **Generate Mesh** (p. 486) in the menu. The inflation layers are regenerated, and the mesh is displayed when you select the **Mesh** object.

11. Continue making adjustments and experimenting with automatic **Program Controlled** inflation vs. local inflation until your results are satisfactory.

Note:

- For inflation on virtual bodies, you must specify automatic **Program Controlled** inflation. You cannot specify local controls to inflate virtual bodies.

- Assembly meshing algorithms do not support 2D inflation (inflation scoped to faces with edges selected as boundaries).

- Assembly meshing algorithms do not support inflation on both sides of a face zone. If you apply inflation to a face zone that is shared by two cell zones of type fluid, the desired inflation will not occur.

- Inflation on both sides of a baffle is supported for the **Tetrahedrons** algorithm only. Refer to the discussion of inflation controls in Selecting an Assembly Mesh Method (p. 375) for more information.

- If multiple inflation controls are defined and different numbers of layers are defined in them, the smallest defined number of layers will be respected for assembly meshing algorithms.

- If multiple inflation controls are defined, each with different values for Aspect Ratio (Base/Height) (p. 154), the last value is the one that will be taken (the value specified for the inflation control that appears lowest in the Tree Outline is honored) and will be used for all inflation controls for which Inflation Option (p. 150) is set to **Last Aspect Ratio**.

Exporting the Mesh

After you determine the inflation mesh is acceptable, the mesh is ready for export.

Points to remember when exporting an assembly mesh:

- Names of parts, bodies, and Named Selections should be limited to 64 characters.
- If you encounter a problem when using assembly meshing algorithms in the Meshing application, you can export the faceted geometry (p. 74) to Fluent Meshing where you can display, interrogate, and repair the faceted data.
- When a **CutCell** mesh is exported from the Meshing application to Ansys Fluent (p. 43), elements that are connected to parent faces are exported in polyhedral format, while all others retain their type. Note that this behavior is only true for **CutCell**; the **Tetrahedrons** algorithm uses only traditional element types.
- When the mesh is exported to Ansys Fluent, a cell zone type of either *FLUID* or *SOLID* is assigned to each body based on its material properties. Refer to Fluent Mesh Export (p. 43) for details.
- Due to the 3D nature of the underlying meshing approach, mesh is not re-associated to any Named Selection applied on vertices or edges. Hence, these Named Selections are not transferred to Ansys Fluent.
- When assembly meshing algorithms are used, mesh is associated to surface bodies in the Meshing application. The association is required in the Meshing application to ensure proper handling of the mesh topology during internal processes. This requirement means that if your model contains surface bodies, the element count reported in the Meshing application will differ from the cell count reported in Ansys Fluent for the same case.
- You cannot use assembly meshing in the Meshing application if you need to maintain the original boundaries of overlapping surfaces and plan to use a non-conformal mesh interface in Ansys Fluent. If non-conformal mesh is needed, for example for sliding mesh problems, another mesh method must be used.
- Refer to Defining Virtual Bodies (p. 379) for information about export of virtual bodies to Ansys Fluent.

Selective Meshing

Using selective meshing, you can selectively pick bodies and mesh them incrementally. After meshing a body, you can mesh the entire part or assembly or continue meshing individual bodies. To generate the rest of the mesh in the model, use the **Generate Mesh** (p. 486) feature.

The following mesh methods are supported:

- For solid meshing:
  - Patch Conforming Tetrahedron (p. 200)
  - Patch Independent Tetrahedron (p. 200)

  - MultiZone (p. 228)
  - Sweep (p. 223)
  - Hex Dominant (p. 222)
- For surface meshing:
  - Quad Dominant (p. 245)
  - All Triangles (p. 246)
  - MultiZone Quad/Tri (p. 246)

Remember the following information when using selective meshing:

- Selective meshing is enabled by default. You can use the **Allow Selective Meshing** (p. 317) option to disable it. See Disabling Selective Meshing (p. 408) for information about Meshing application behaviors when selective meshing is disabled.
- Selective meshing is not persistent for a geometry update or re-mesh operation. However, you can use the **Mesh** worksheet to create a selective meshing history so that your meshing steps can be repeated in the desired sequence. Otherwise, you must manually re-mesh bodies in the desired sequence. Refer to Using the Mesh Worksheet to Create a Selective Meshing History (p. 409) for details.
- When using the **Preview Surface Mesh** (p. 489), **Preview Source and Target Mesh** (p. 491), or **Preview Inflation** (p. 492) feature during selective meshing, the previewed mesh will be discarded when you perform a subsequent preview or full mesh operation. The previewed mesh will not be used to seed the subsequent mesh operation.
- After meshing, the meshed status icon (p. 535) appears in the Tree Outline for a meshed body within the **Geometry** folder, or for a multibody part whose child bodies are all meshed. If you make changes after meshing that invalidate (p. 535) the mesh for an individual body (such as adding sizing to the body), you will need to re-mesh that body only.
- In a multibody part, if any child bodies have been meshed and refined, another child body is unmeshed, and you subsequently mesh the unmeshed body, the mesh state of all refined bodies in the part will be invalidated and re-meshed during mesh generation. Similarly, if one body is unmeshed and refinement is needed on another, generating the mesh will result in meshing and refinement of the entire part. In addition to cases involving refinement, this behavior applies in cases where post inflation is used.
- When meshing a body that is part of a symmetry object, match control (p. 280), or pinch control (p. 182), all bodies to which the control is applied need to be meshed at the same time. Also, if a body that is part of a symmetry object, mesh connection object, contact match object, match control, or pinch control fails to mesh, the body will have an invalid mesh state (p. 535) that will propagate to all other bodies that are part of the respective object/control.
- Mesh connections (p. 444), contact matches (p. 455), post inflation (p. 154), and refinement (p. 264) are not performed until all body meshes have been generated. These operations will either be performed when the last body is meshed through selective meshing, or in the last step of the **Generate Mesh** (p. 486) operation.

- Selective meshing is boundary constrained. That is, if you add a size control to a face that is adjacent to an up-to-date body, the edges of that face will be recovered from the existing mesh. Due to the boundary constraints, the mesher cannot split the edges to aid in meshing and will fail if it attempts to do so.

- When you mix mesh methods in multibody parts, the manner in which topology shared by multiple bodies is protected depends on whether adjacent bodies are being meshed with Patch Independent methods (Patch Independent Tetrahedron (p. 200), MultiZone (p. 228), or MultiZone Quad/Tri (p. 246)) and/or Patch Conforming methods (Patch Conforming Tetrahedron (p. 200), Sweep (p. 223) [general or thin], or Hex Dominant (p. 222)):

  - The interface between a Patch Conforming method and a Patch Conforming method is not protected.

  - The interface between a Patch Conforming method and a Patch Independent method is completely protected.

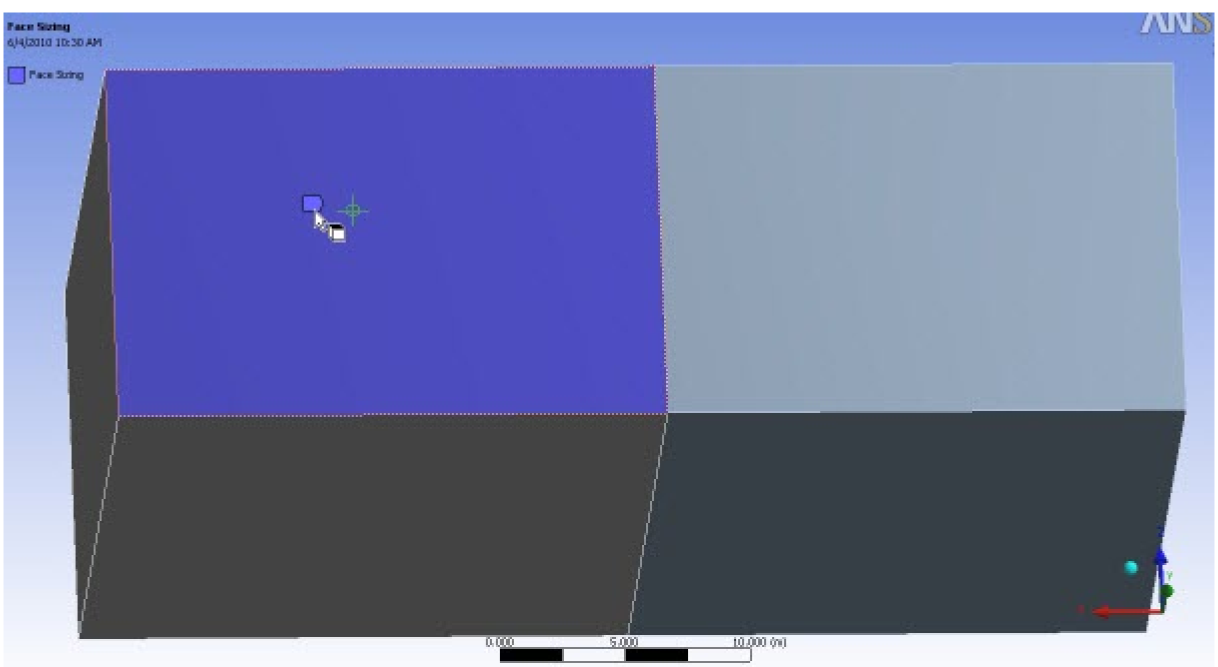
  - Only the boundary is protected at the interface between a Patch Independent method and a Patch Independent method.

- You can use the **Verbose Messages from Meshing** (p. 317) option to control the verbosity of messages returned to you. Depending on the setting, before meshing a message reports the subset of bodies that is going to be meshed and/or after meshing a message reports the subset of bodies that failed to mesh.

- Size controls on neighboring bodies are not considered if you are performing selective meshing. This limitation is applicable to all mesh methods that support selective meshing; however, its impact may differ depending on the methods being used.

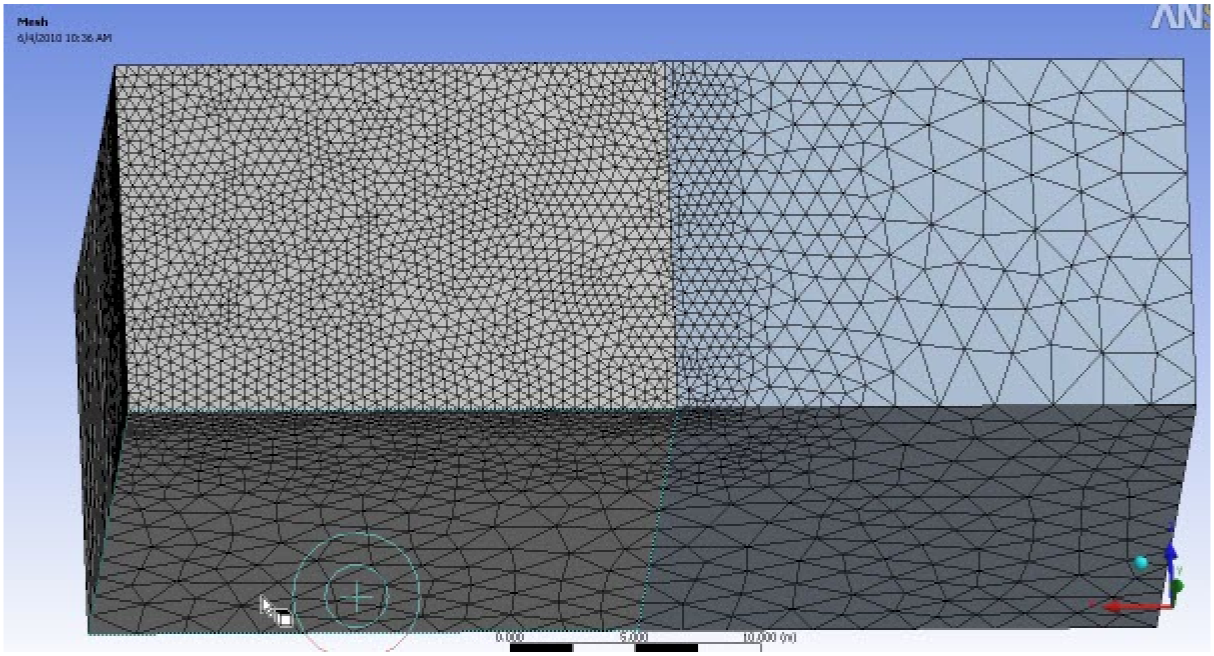
  For example, consider the simple model below, which consists of two boxes to which the Patch Independent Tetra mesh method has been applied. A local size control that defines a much smaller **Element Size** than the global size has been scoped to the top face of the box on the left.

  Figure 189: Two Boxes with Sizing on One Face



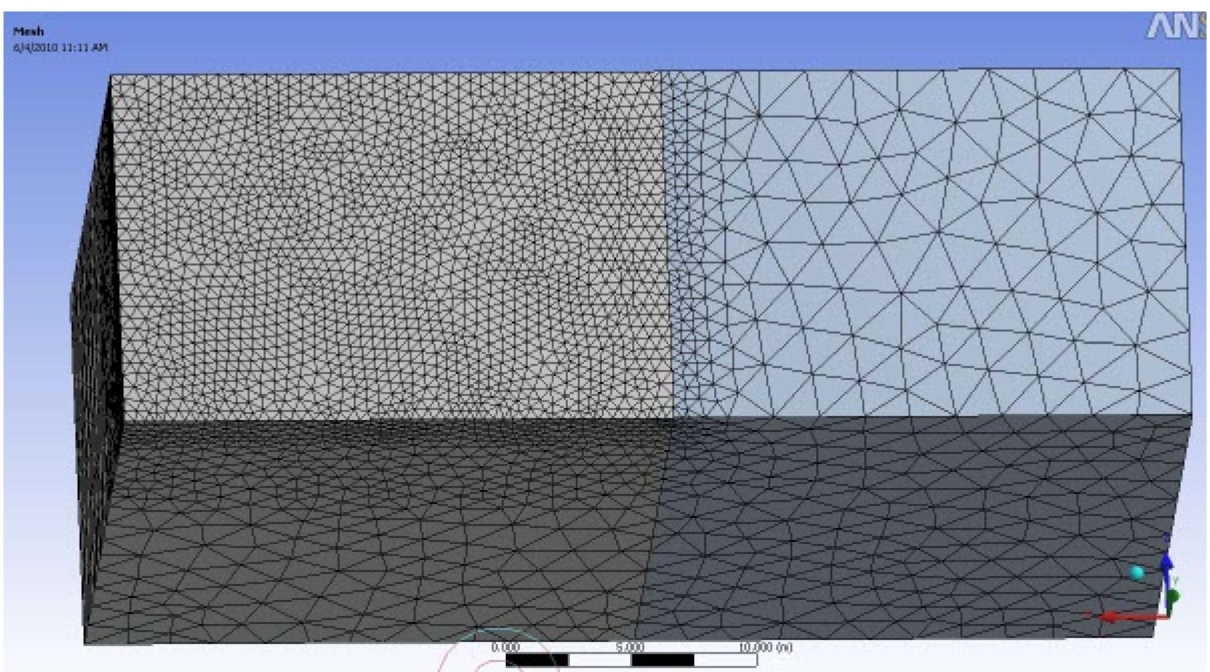
When the mesh is generated in one step (for the entire part rather than body by body), there is a smooth transition from the fine element size to the coarse element size, as shown in Figure 190: Mesh Generated for Entire Part (p. 407).

Figure 190: Mesh Generated for Entire Part



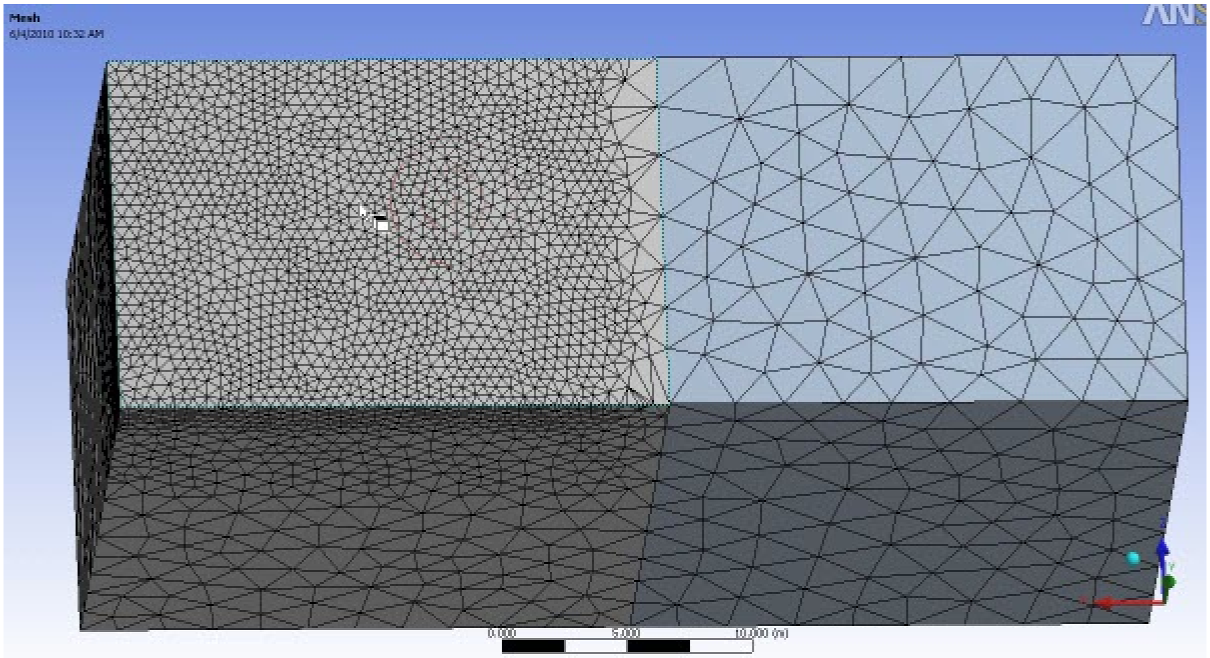
However, the mesh will differ if selective (body by body) meshing is performed. Figure 191: Selective Meshing: Left Body First (p. 407) shows the mesh when the body on the left is meshed first, and the body on the right is meshed second. In this case although the results are different than those in the figure above, the mesh may still be acceptable because the impact of the local size control on the left body has influenced the boundary mesh of the right body.

Figure 191: Selective Meshing: Left Body First



In Figure 192: Selective Meshing: Right Body First (p. 408), the body on the right was meshed first, and the body on the left was meshed second. When this meshing sequence is used, the mesh on the right body does not recognize the size control that is scoped to the body on the left. This results in a coarse mesh on the right body with the transition region occurring on the left body.

Figure 192: Selective Meshing: Right Body First



- When **Mesh Based Connection** is set to **Yes**,**Selective Meshing** is not supported.

Disabling Selective Meshing

Set the **Allow Selective Meshing** (p. 317) option to **No** to disable selective meshing. The Meshing application behaves as follows when selective meshing is disabled:

- If you make changes after meshing that invalidate (p. 535) the mesh for an individual body in a multibody part (such as adding sizing to the body), the mesh for all bodies in the part is invalidated and you will need to re-mesh all bodies.
- If one body in a multibody part is suppressed and you mesh all the other bodies, unsuppressing the suppressed body invalidates (p. 535) the mesh for all the bodies within that part. When you regenerate the mesh, all the bodies within that part will be re-meshed. If the model contains multiple parts, bodies in the other parts will not be affected and will not be re-meshed.
- The **Generate Mesh** (p. 486), **Preview Surface Mesh** (p. 489), and **Clear Generated Data** (p. 496) RMB menu options are unavailable for individual bodies in multibody parts in the Tree Outline. To use these features for a multibody part, you must right-click at the part level in the Tree Outline.

- There is no **Parts>** flyout menu for the **Generate Mesh** (p. 486), **Preview Surface Mesh** (p. 489), and **Clear Generated Data** (p. 496) RMB menu options in the **Geometry** window. If you select one of these options, the action is performed on the entire part.

Note:

In some cases, the mesh may be generated in a selective fashion (body by body) even if selective meshing is currently disabled. For example, if you use selective meshing to mesh some of the bodies in a part, then disable selective meshing, and then generate the mesh *and the mesh process does not invalidate any bodies*, the mesh is generated using selective meshing processes. To avoid this behavior, you can use the **Clear Generated Data** (p. 496) feature or force a change of the mesh state on the part. Non-selective meshing will be used for all subsequent meshing.

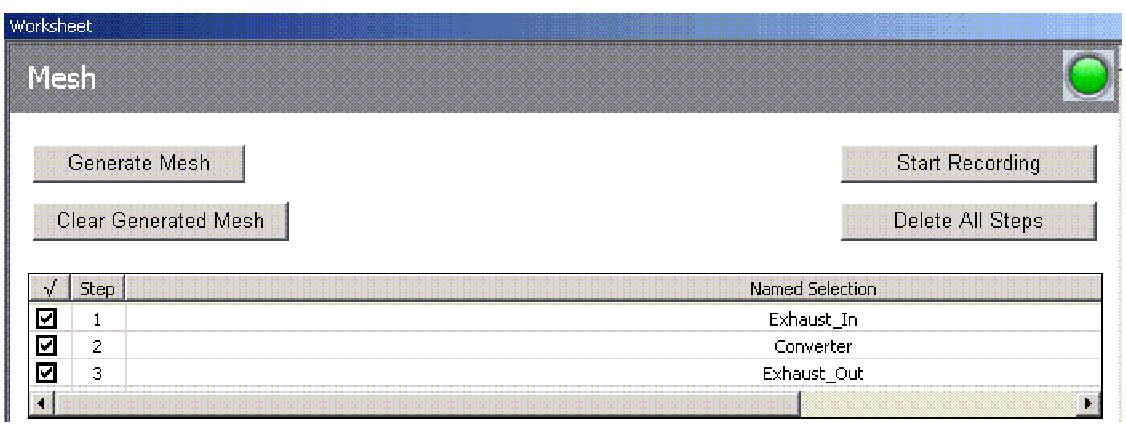
Using the Mesh Worksheet to Create a Selective Meshing History

When you perform selective meshing, you control the sequence in which bodies are meshed. You can use the **Mesh** worksheet to create a selective meshing history, so that your meshing steps can be repeated in the desired sequence for any geometry update or re-mesh operation. Figure 193: Mesh Worksheet (p. 409) shows a sample **Mesh** worksheet. Each row in the worksheet corresponds to a step in the meshing sequence. When you generate the mesh, the software processes the steps one by one. For each step, the software selects the bodies identified by the specified Named Selection and meshes those bodies using the meshing controls that are applied to them.

Note:

The **Mesh** worksheet is a helpful meshing tool, but it does not manage mesh state. State management is handled at the prototype (**Body** object) and mesh levels. Mesh state is visible in the Tree Outline in the **Geometry** folder and in the **Mesh** folder. See Understanding States (p. 538) for more information.

Figure 193: Mesh Worksheet



There are two ways to create a selective meshing history:

- By recording meshing steps as you perform them
- By adding meshing steps to the worksheet manually

To create a selective meshing history by recording your steps:

1. Click the **Mesh** object in the Tree Outline.
2. Do one of the following:

   Right-click the **Mesh** object and select **Start Recording** from the context menu. As a result, the worksheet opens in recording mode automatically.

   -or-

   Click the **Worksheet** button on the toolbar and click the **Start Recording** button on the worksheet.

3. Move the worksheet to the desired location to dock it. The location will persist whenever the **Mesh** object or one of its child objects is highlighted in the Tree Outline. For example, you may want to dock the worksheet alongside the **Geometry** window, allowing you to view both at once.

   Note:

   The worksheet is not dockable on Linux platforms. On Linux, you can drag the worksheet off the Meshing application interface, and it will then appear in its own separate window.

4. Do one of the following:

   Select one or more bodies or parts in the **Geometry** window, right-click, and select **Generate Mesh on Selected Bodies**. If you did not dock the worksheet, you may need to click the **Worksheet** button on the toolbar or the **Graphics** tab at the bottom of the worksheet to return to the **Geometry** window first.

   -or-

   Select one or more bodies or parts in the Tree Outline, right-click, and select **Generate Mesh**.

   As a result, a Named Selection for the selected set is generated (named **Meshing_1**, **Meshing_2**, **Meshing_3**, and so on) and is added to the worksheet automatically.

   During recording, other steps in the worksheet are ignored while the mesh for the selected set is being generated. See below for additional notes on recording and playback behaviors.

5. Repeat step 4 for each meshing step in the desired sequence.
6. When you are done meshing bodies/parts, do one of the following:

   Right-click the **Mesh** object and select **Stop Recording** from the context menu.

   -or-

Click the **Stop Recording** button on the worksheet.

Note:

You do not have to record the meshing for all bodies, only those for which you want to record the meshing order. See Mesh Worksheet Recording and Playback Behaviors (p. 411) for details.

To create a selective meshing history manually:

1. Click the **Mesh** object in the Tree Outline.
2. Click the **Worksheet** button on the toolbar.
3. Move the worksheet to the desired location to dock it. The location will persist whenever the **Mesh** object or one of its child objects is highlighted in the Tree Outline. For example, you may want to dock the worksheet alongside the **Geometry** window, allowing you to view both at once.

Note:

The worksheet is not dockable on Linux platforms. On Linux, you can drag the worksheet off the Meshing application interface, and it will then appear in its own separate window.

---

4. Add a row to the worksheet by right-clicking on the table and selecting **Add** from the context menu.
5. In the new row, click the **Named Selection** column and select a Named Selection from the **Named Selection** drop-down.
6. Repeat steps 4 and 5 for each meshing step in the desired sequence.

Mesh Worksheet Recording and Playback Behaviors

Remember the following information regarding recording and playback behaviors:

- During recording, the button in the upper-right corner of the worksheet is red. When recording is stopped, the button is green.
- When you start recording, the software checks for meshed bodies and verifies that each meshed body is accounted for in the worksheet. If any extraneous meshed bodies are found, recording does not occur and an error message is issued. You must clear the mesh entirely (or at least clear the mesh from the bodies that are not being used by the worksheet) before you can begin recording. To do so, use the **Clear Generated Data** (p. 496) option or click the **Clear Generated Mesh** button on the worksheet.
- When you start recording, the states of the bodies being used by the worksheet are checked. If any body is not in a meshed state, recording does not occur and an error message is issued. You must bring the mesh up-to-date before you can begin recording. To do so, right-click the last step in the worksheet and select **Generate Mesh Through This Step** from the context menu. See below for more information about using this option.

- As steps are being recorded, they are appended to the existing steps in the worksheet.
- Other steps in the worksheet are ignored while the mesh for the selected set is being generated.
- After you have entered recording mode, recording stops when the mesh is up-to-date. An exception to this behavior occurs if you record one or more steps but then suppress all remaining unmeshed bodies. In such cases, the mesh will be up-to-date but recording will not stop automatically. You must click **Stop Recording**.
- During recording/playback, post mesh operations (including mesh connections (p. 444), contact matches (p. 455), post inflation (p. 154), and refinement (p. 264)) do not occur until the last step in the worksheet, after all meshing is complete.
- If you select the Preview Surface Mesh (p. 489), Preview Source and Target Mesh (p. 491), or Preview Inflation (p. 492) feature, recording stops and a warning message is issued.
- The Generate Mesh (p. 486) option is context sensitive. It behaves differently depending on where you invoke it. If you invoke it from the **Mesh** object in the Tree Outline (RMB on **Mesh** folder > **Generate Mesh**), it behaves similar to the **Preview** options in that recording stops. If you invoke it from the **Geometry** object (RMB on **Geometry** folder > **Generate Mesh**) or from the **Geometry** window (RMB > **Generate Mesh On Selected Bodies**), the operation is treated as a selective meshing step and is recorded.
- If you insert a step manually, recording stops and a warning message is issued.
- If you delete a step manually, recording stops and a warning message is issued.
- If the mesh fails, recording stops and a warning message is issued.
- To replay steps incrementally, right-click in the row of interest and select **Generate Mesh Through This Step** from the context menu. As a result, recording stops, any existing mesh is cleared, and all meshing steps prior to and including the selected step are replayed in the **Geometry** window.

Mesh Worksheet Named Selection Behaviors

Remember the following information regarding Named Selection behaviors:

- Only Named Selections that consist of bodies are selectable from the worksheet's **Named Selection** drop-down.
- If the model contains bodies that are not included in any Named Selection, these bodies are meshed last.
- If a Named Selection is used by the worksheet, the **Used by Mesh Worksheet** field in the Details View for that Named Selection is set to **Yes**, even if the corresponding worksheet step is inactive.
- For Named Selections that are generated automatically by the worksheet, the **Send to Solver** (p. 78) option in the Details View is set to **No** by default. The default is **Yes** for Named Selections that you create. The worksheet respects the user-defined settings. That is, if you create a Named Selection, retain the **Send to Solver** setting of **Yes**, and subsequently use that Named Selection in the worksheet, the Named Selection will be passed to the solver as expected.

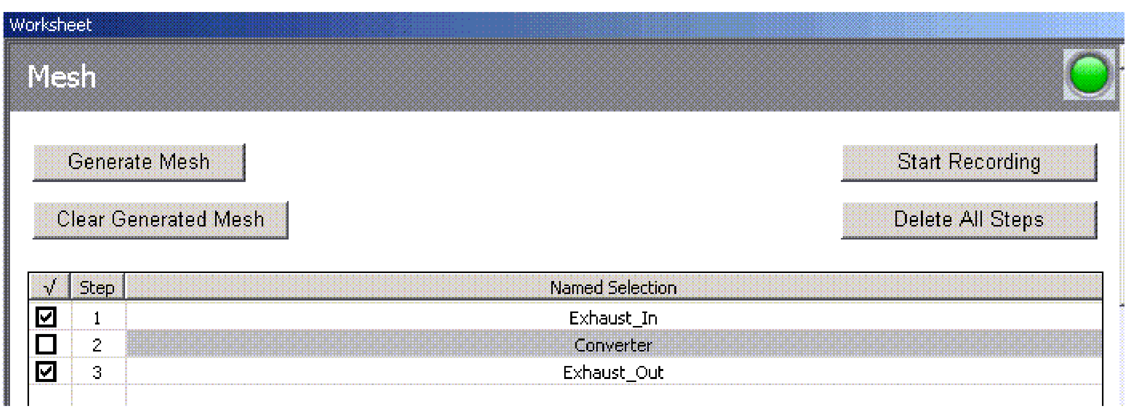
- If a change to a Named Selection being used by the worksheet causes the Named Selection to become underdefined or invalid, the corresponding worksheet step will be deactivated without invalidating the worksheet. Reasons a Named Selection may become underdefined or invalid include:
  - You change the scope of a Named Selection from body to face/edge
  - A geometry update changes the scope of a Named Selection, or causes the software to try to delete the Named Selection
- If you delete a named selection manually, recording stops and a warning message is issued.
- From the Tree Outline, you can suppress/unsuppress Named Selections or bodies included in Named Selections that are being used by the worksheet, and the corresponding worksheet step will be deactivated/activated accordingly without invalidating the worksheet.

  You also can activate/deactivate steps directly on the worksheet. By default, a check mark (√) appears on the worksheet, meaning steps that correspond to all unsuppressed bodies/Named Selections are active. To deactivate a single step, clear the corresponding check box. To deactivate all active steps, click √ and it is replaced by ▫. Toggling between √ and ▫ activates/deactivates the steps corresponding to all unsuppressed bodies/Named Selections. Toggling step activation on the worksheet has no effect on the suppressed status of the corresponding bodies/Named Selections in the Tree Outline.

  When a step is inactive, its row in the worksheet is grayed out. Inactive steps are skipped during mesh generation and other worksheet operations.

  To obtain the example shown in Figure 194: Mesh Worksheet Step Deactivation (p. 413), the Converter Named Selection was suppressed in the Tree Outline.

Figure 194: Mesh Worksheet Step Deactivation



Miscellaneous Points to Remember

- To delete a row from the worksheet, right-click in the row of interest and select **Delete** from the context menu. The worksheet, including the meshing sequence, will be updated automatically.
- Click the **Delete All Steps** button to clear all data from the worksheet.

- If the mesh fails at any point in the process, the process terminates but returns as much of the mesh as possible.
- If you change the worksheet after you mesh, those changes will not be reflected in your meshing state as the worksheet does not affect meshing state. The changes will take effect the next time you generate a mesh.
- If you select bodies to mesh individually (by using the Generate Mesh On Selected Bodies (p. 486) option in the **Geometry** window; the Generate Mesh (p. 486) option in the **Geometry** folder; or by using the Preview Surface Mesh (p. 489), Preview Source and Target Mesh (p. 491), or Preview Inflation (p. 492) feature), the Meshing application ignores the worksheet and generates the mesh for the selected bodies.
- If you select **Part** or **Body** objects in the **Geometry** folder in the Tree Outline, right-click, and then select Generate Mesh (p. 486) from the context menu, the Meshing application ignores the worksheet and generates the mesh for the selected parts/bodies.
- This feature is not available when assembly meshing algorithms (p. 367) are being used.

Inflation Controls

Inflation is useful for CFD boundary layer resolution, electromagnetic air gap resolution or resolving high stress concentrations for structures.

The following sections provide the high-level steps to follow to assign inflation depending on the selected mesh method.

For information on setting global inflation controls and descriptions of all of the individual inflation controls, refer to Inflation Group (p. 145). Alternatively, you can use local inflation mesh controls to apply inflation to specific boundaries. For details, refer to Inflation Control (p. 291). For general information on applying inflation controls in combination with the various mesh method controls, refer to Meshing: Mesh Control Interaction Tables (p. 435).

Inflation Controls With Sweeper

Inflation is a pre process for the sweeper (p. 223). The source face is meshed and then inflated before sweeping with the sweeper.

Note:

- Inflation is supported only on the source face(s) of the sweep (that is, inflation on the side faces). Inflation away from the source face(s) is not supported.
- You do not have to select a source face for sweeping with inflation. You can simply pick faces for inflation and the Meshing application will internally place a Sweep method on the adjacent bodies using the inflated faces as the sources (unless another method already exists).

To add boundary layers to a source face for sweeping:

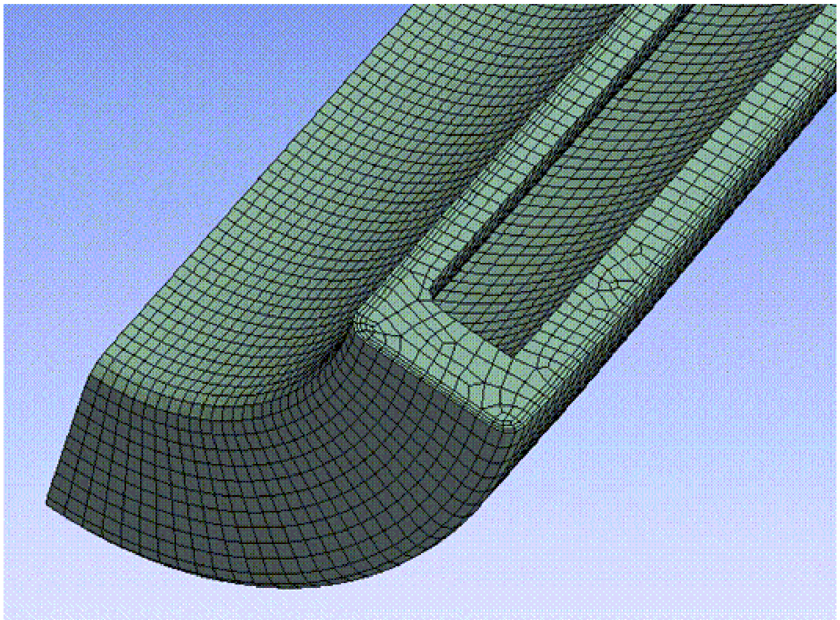
1. Apply a **Method** control to a body.
2. Set **Method** to **Sweep**.
3. Set **Src/Trg Selection** to **Manual Source** or to **Manual Source and Target**.
4. Scope the source (and target if **Manual Source and Target** was selected).
5. Set **Free Face Mesh Type** to **All Quad**, **All Tri**, or **Quad/Tri**. Your selection determines the shape of the elements used to fill the swept body (pure hex, pure wedge, or a combination of hex/wedge respectively). The boundary region of the source/target faces will always be meshed with quad layers. Refer to Figure 195: Sweep Method With Inflation: Hex Fill (p. 416) and Figure 196: Sweep Method With Inflation: Wedge Fill (p. 416).
6. Enter additional sweep settings, as desired, in the Details View.
7. Select the source face and insert an **Inflation** control.
8. Select the outer boundary edges of the source face for inflation (p. 293) (the edges that you want inflation to grow away from).
9. Enter additional inflation settings, as desired, in the Details View.
10. Mesh the body.

Note:

If the target face has a different number of edges than the source face, the bias of the boundary layer may not be transferred correctly.

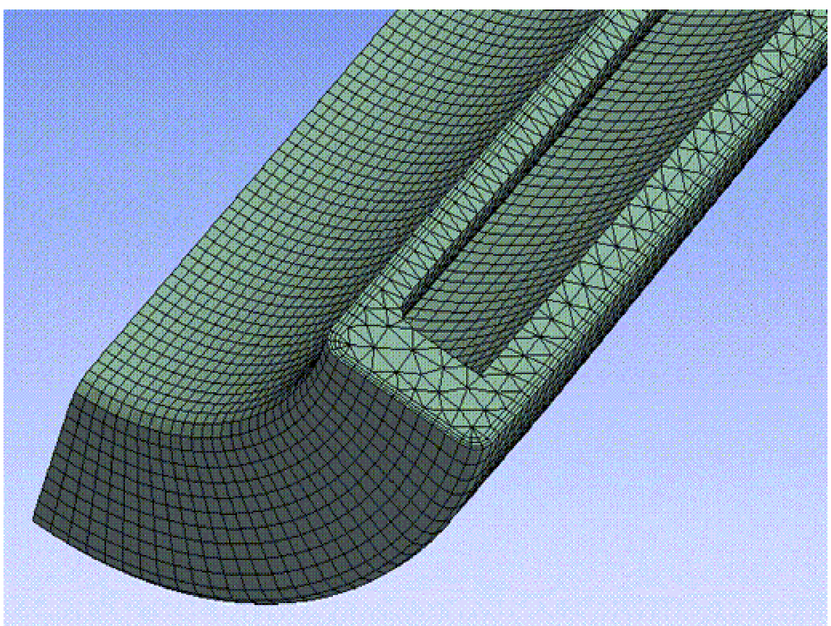
To obtain the mesh shown in Figure 195: Sweep Method With Inflation: Hex Fill (p. 416), **Free Face Mesh Type** was set to **All Quad**. Notice the boundary region is meshed with quad layers.

Figure 195: Sweep Method With Inflation: Hex Fill



To obtain the mesh shown in Figure 196: Sweep Method With Inflation: Wedge Fill (p. 416), **Free Face Mesh Type** was set to **All Tri**. Notice the boundary region is meshed with quad layers.

Figure 196: Sweep Method With Inflation: Wedge Fill



Inflation Controls With Patch Conforming Mesher

Inflation can be either a pre process or a post process for the patch conforming (p. 200) mesher.

To add boundary layers to a face using the Patch Conforming Mesher:

1. Apply a **Method** control to a body.

2. Set **Method** to **Tetrahedrons**.
3. Set the tetrahedrons **Algorithm** to **Patch Conforming**.
4. Select the body and insert an **Inflation** control.
5. Select the faces to be inflated (p. 293) (the faces that you want the inflation layers to grow away from).
6. Enter additional settings, as desired, in the Details View.
7. Mesh the body.

Inflation Controls With Patch Independent Mesher

Inflation is a post process for the patch independent (p. 200) mesher after it has created the tetrahedron elements.

To add boundary layers to a face using the Patch Independent Mesher:

1. Apply a **Method** control to a body.
2. Set **Method** to **Tetrahedrons**.
3. Set the tetrahedrons **Algorithm** to **Patch Independent**.
4. Set the **Min Size Limit**.
5. Select the body and insert an **Inflation** control.
6. Select the faces to be inflated (p. 293) (the faces that you want the inflation layers to grow away from).
7. Enter additional settings, as desired, in the Details View.
8. Mesh the body.

Inflation Controls With MultiZone

To add boundary layers to a face using the MultiZone (p. 228) Mesher:

1. Apply a **Method** control to a body.
2. Set **Method** to **MultiZone**.
3. Select the body and insert an **Inflation** control.
4. Select the faces to be inflated (p. 293) (the faces that you want the inflation layers to grow away from).
5. Enter additional settings, as desired, in the Details View.
6. Mesh the body.

For more information, see MultiZone Support for Inflation (p. 364).

Inflation Controls With MultiZone Quad/Tri Mesher

To add boundary layers to a face using the MultiZone Quad/Tri (p. 246) Mesher:

1. Apply a **Method** control to a body.
2. Set **Method** to **MultiZone Quad/Tri**.
3. Select a body or face and insert an **Inflation** control.
4. Select the edges to be inflated (p. 293) (the edges that you want inflation to grow away from).
5. Enter additional settings, as desired, in the Details View.
6. Mesh the body.

Note:

Base mesh caching is not supported for **MultiZone Quad/Tri**, so a change to inflation controls requires remeshing.

Inflation Controls With Quadrilateral Dominant or All Triangles Mesher

Inflation is a pre process for the quadrilateral dominant (p. 245) mesher or all triangles (p. 246) mesher.

To add boundary layers to a face using the Quadrilateral Dominant or All Triangles Mesher:

1. Apply a **Method** control to a body.
2. Set **Method** to **Quadrilateral Dominant** or **Triangles**.
3. Select a body or face and insert an **Inflation** control.
4. Select the edges to be inflated (p. 293) (the edges that you want inflation to grow away from).
5. Enter additional settings, as desired, in the Details View.
6. Mesh the body.

Inflation Controls With Cartesian Mesher

Inflation is a **Pre** process only for the Body Fitted Cartesian (p. 236) mesher. For **CFD** physics only, three layers are created with total thickness proportional to **Element Size**; for other physics preferences, only only layer is created.

To add boundary layers to a body using the Cartesian Mesher:

1. Apply a **Method** control to a body.
2. Set **Method** to **Cartesian**.

3. Select the **Method** and add an **Inflation** (p. 292) control.

4. Enter additional settings, as desired, in the Details View.

5. Mesh the body.

Inflation Handling Between Bodies With Different Methods

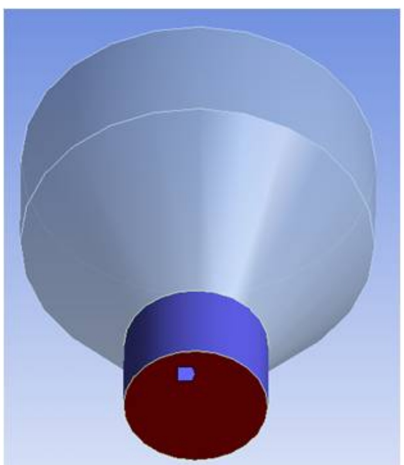
The inflation handling between bodies where one body is meshed with the sweep method and one body is meshed with the patch conforming tetrahedral method requires some special consideration to ensure inflation layers propagate through the common interface. There are two such cases to consider:

- The case in which the common interface of two bodies is also a source/target face of the swept body

- The case in which the common interface of two bodies is also a side face of the swept body

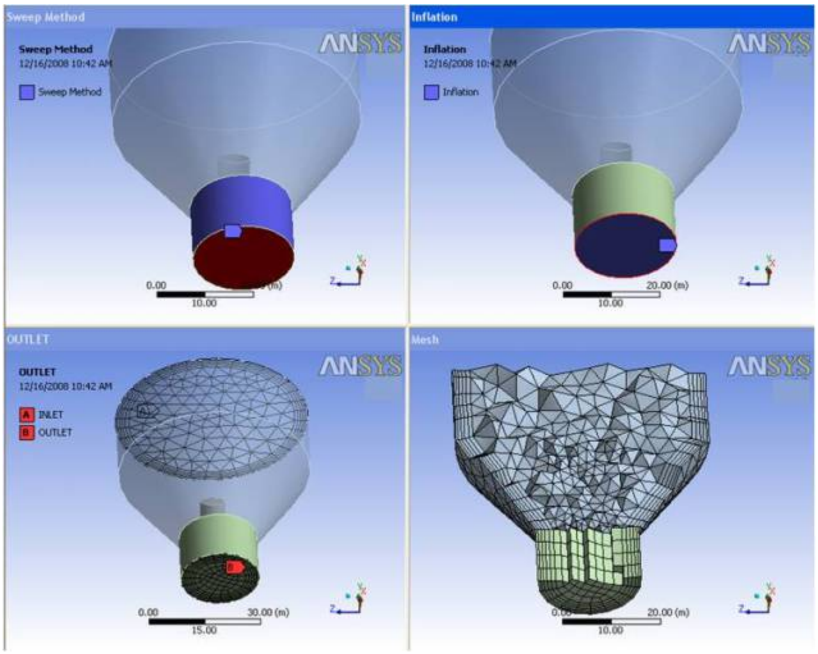
The model below will be used to explain the first case, in which the common interface of two bodies is also a source/target face of the swept body.

Figure 197: Swept Body Shares Source/Target Face With Tet Body



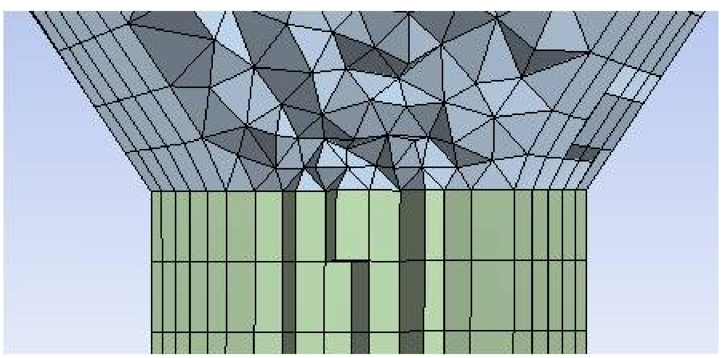
In this case, inflation on the patch conforming tetrahedral method must be defined off the faces of the wall (not common interface), or by using Program Controlled (p. 148) inflation (which ignores faces in Named Selections and common interfaces between bodies). The swept body needs the source face to be selected, and 2D inflation must be defined on the source face. Since 2D inflation does not support the Smooth Transition (p. 150) option, it is best to use another option so that the inflation between bodies will properly align.

Figure 198: Defining Inflation for a Swept Body Sharing Source/Target Face With Tet Body



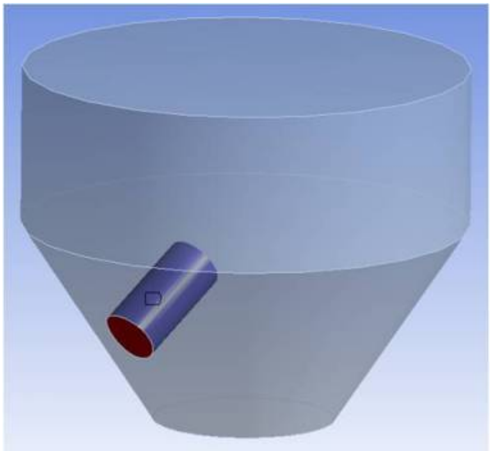
After properly setting up the model and ensuring the inflation of the tet body and the swept body have similar near-wall spacings, a mesh can be generated where the inflation layers will pass from one body to the next with proper connections on the common interface, as shown below.

Figure 199: Detail of Proper Connections on the Common Interface



The model below will be used to explain the second case, in which the common interface of two bodies is also a side face of the swept body.

Figure 200: Tet Body Surrounds Swept Body



In this case, inflation on the patch conforming tetrahedral method must be defined off the faces of the wall (not common interface), or by using Program Controlled (p. 148) inflation (which ignores faces in Named Selections and common interfaces between bodies). To properly align the inflated tet mesh to the side faces of the swept body, a biasing must be used along the sweep direction. Since the biasing along the sweep direction uses a different formulation than the inflation biasing, the following notes may be helpful.

The sweep bias ratio is the ratio of largest edge to smallest edge along sweep path, the growth ratio for inflation is a factor of the growth from the first element to the second element, etc. These relate as described below. The equation to get the inflation growth rate to align to the swept body is:

$$I_r = S_b^{(1/N-1)}$$

where

S_b = Sweep Bias

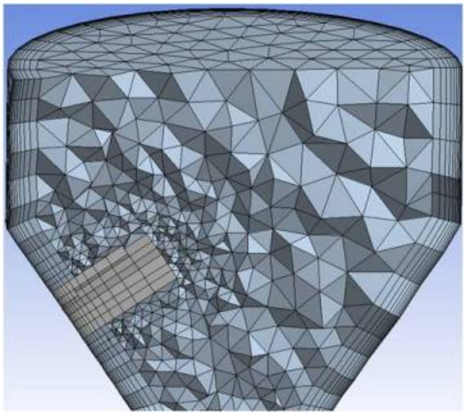
N = Number of Divisions along sweep

I_r = Inflation Growth Rate

Also, to get proper alignment between the first layer of inflation and sweep, you need to use the first layer height of the swept mesh as first layer height for sweep.

Because the first layer height is computed by the software from the length of the sweep path, the sweep bias, and the number of divisions along the sweep—and there is no easy way to get the length of the sweep path—you should mesh the swept body first, measure the first layer height, and use this value as input for the First Layer Height (p. 154) option when defining inflation controls. The inflation growth rate can then be calculated using the formula above. This leads to well-aligned layers between the sweep and tet regions, as shown below.

Figure 201: Detail of Well-aligned Layers Between the Swept and Tet Regions



Inflation and Baffles

The Meshing application provides support for meshing 0–thickness walls, or baffles, as non-manifold faces of a solid body. Inflation is supported. See Baffle Meshing (p. 431) for more information.

Mesh Refinement

Mesh refinement is a postprocess in the mesh generation process in which the elements on the selected topology are split. This is useful for local mesh sizing control. See the Refinement Control (p. 264) section for more information.

Mixed Order Meshing

The **Method Controls and Element Order Settings** allow you to specify whether the mesh to be generated for a given body is **Quadratic** (high order) or **Linear** (low order). **Mixed Order Meshing** refers to meshing a multibody part having shared topology with some bodies as **Quadratic** and some bodies as **Linear**. In such situations, "transitional elements" are required to connect the mesh at any linear-to-quadratic interface. These "transitional elements" are treated as quadratic elements with dropped midside nodes, and the side of the interface on which they appear is determined by your meshing process. See Method Controls and Element Order Settings (p. 196) for more information.

Contact Meshing

Enhancing convergence or quality of results for structural contact analysis may require the use of contact sizing to create similarly sized meshes where faces and bodies are coming in contact. See the Contact Sizing Control (p. 263) section for more information.

For information about using contact meshing for rigid bodies, refer to Rigid Body Meshing (p. 424).

Winding Body Meshing

Winding body meshing creates special element types depending on the attributes given to bodies in the DesignModeler application. No mesh controls are supported for winding bodies because of the nature of the required mesh.

Wire Body Meshing

Wire body meshing meshes the wire bodies in an assembly, respecting any mesh controls applied to the edges of the wire body.

Note:

Wires and beams are both considered to be line bodies and are handled in the same way by the mesher.

Pyramid Transitions

Pyramid transitions occur when a swept or hex dominant body abuts, that is, shares faces with, a body meshed with tetrahedrons. The mesher will try to insert the pyramids into the body meshed with tetrahedrons. If that is not possible, the hexahedron at the boundary will be split into pyramids and tetrahedrons to create a conformal mesh between the two bodies. Pyramids will also be formed at the interface of an end cap of inflation on quad surface mesh and a tet body.

Match Meshing and Symmetry

For parts that are symmetric about a cylindrical axis, you can match the mesh by using either cyclic mesh matching or the **Symmetry** feature in the Mechanical application. The following table describes when to use each method:

Table 2: Mesh Matching for Symmetrical Parts

If you want to...	Do this...
Match the mesh, but you do not want to automatically generate solver constraints for periodic mesh	Apply a **Cyclic Mesh Match** control (p. 283).
Match the mesh, and automatically generate solver constraints for periodic mesh	Define the symmetry in the model by applying the necessary **Symmetry regions**, **Periodic regions**, or **Cyclic regions**.

When **Periodic Region** or **Cyclic Region** objects exist in the **Symmetry** folder, match face mesh controls will be created internal to the mesher. If the mesher cannot match the mesh on the objects in the **Symmetry** folder, it will return a failure or informational message.

Related topics include:

- For a description of the **Symmetry** folder and its support of **Periodic Region** and **Cyclic Region** objects, see Defining Symmetry in the Mechanical help.

- For an overview of the match control and its limitations, see Match Control (p. 280).
- For general information on applying match controls in combination with the various mesh method controls, see Meshing: Mesh Control Interaction Tables (p. 435).

Rigid Body Meshing

Rigid body meshing simplifies the representation of a model by reducing it to the contact regions and the centroid of the model. In other words, when a part is defined as a rigid body, the mesher knows to mesh only the contact regions, and to create a single mass element at the centroid of the model. (The centroid is represented by an **Inertial Coordinate System**. Refer to the discussion of Rigid Bodies in the Mechanical help for more information.)

Rigid body meshing can be used for both 2D and 3D contact. For 2D models, only the edges of the rigid surface in contact are meshed because the surface mesh is not needed for the analysis. Similarly, for 3D models, only the faces of the rigid body in contact are meshed because the volume mesh is not needed. The elements used to mesh the contact surfaces can be quadrilateral or triangular in shape, with or without midside nodes.

The following surface mesh methods can be applied to rigid bodies:

- Quadrilateral Dominant (p. 245)
- Triangles (p. 246)

Rigid body contact meshing respects mapped face controls (p. 265) and sizing (p. 248) controls. If 2D inflation (p. 145) is applied, inflation layers are generated for the surfaces in contact.

If a method control is scoped to a rigid body, the **Method** control is set to **Quadrilateral Dominant** by default, but you can change the value to **Triangles**. When **Method** is set to **Quadrilateral Dominant**, an additional option called the **Free Face Mesh Type** control is available for most analyses and can be set to either **Quad/Tri** (default) or **All Quad**.

For Transient Structural, Rigid Dynamics, and Explicit Dynamics analyses, certain default behaviors related to rigid body meshing differ depending on analysis type and solver. Additionally, the **Free Face Mesh Type** option is replaced by the **Rigid Face Mesh Type** option to determine the default element shape for rigid body face meshing. The table below provides information you should be aware of when selecting one of these analysis types from the Toolbox and adding it to a Workbench project.

Table 3: Rigid Body Meshing: Default Behaviors for Rigid Dynamics, Transient Structural, and Explicit Dynamics Analyses

Standard Analysis Type	Solver	Element Order (p. 96)	Straight Sided Elements (p. 176)	Rigid Body Behavior (p. 177)	Rigid Face Mesh Type
Rigid Dynamics	Rigid Body Dynamics	-	Not applicable	Full Mesh	Quad/Tri
Transient Structural	Mechanical APDL	Program Controlled	No	Dimensionally Reduced	Quad/Tri
Explicit Dynamics	Autodyn	Linear	Not applicable	Full Mesh	Quad/Tri

Standard Analysis Type	Solver	Element Order (p. 96)	Straight Sided Elements (p. 176)	Rigid Body Behavior (p. 177)	Rigid Face Mesh Type
Explicit Dynamics (LS-DYNA Export)	LS-DYNA	Linear	Not applicable	Full Mesh	Quad/Tri

For information about generating a full mesh on rigid bodies instead of a surface mesh, refer to the description of the Rigid Body Behavior (p. 177) control.

Using 2D Rigid Body Contact Meshing

This section describes the basic steps for using 2D rigid body contact meshing.

To define a 2D rigid body for contact meshing:

1. Open the model in the Mechanical application.
2. In the Tree, expand the **Geometry** object so that the body objects are visible.
3. Click the body that you want to define as the rigid body.
4. In the **Details> Definition** view for the body, change the value of the Stiffness Behavior control to **Rigid**.

   Note:

   When you change the **Stiffness Behavior** to **Rigid**, an **Inertial Coordinate System** object is added to the Tree automatically. This **Inertial Coordinate System** represents the centroid of the body.

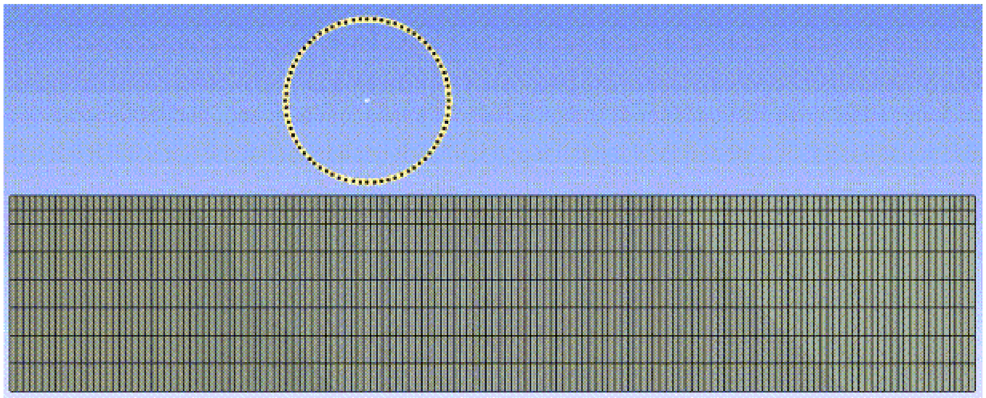
5. If desired, change the value of the **Element Order** (p. 96) control.
6. Generate the mesh by right-clicking on the **Mesh** object in the Tree and selecting **Generate Mesh**.

   Note:

   The mesh for the 2D rigid body is created only in the contact region (edges in contact). See the figure below for an example.

In the figure below, which shows a model of a slab and a cylinder, the cylinder has been defined as a rigid body. When the mesh is generated, the cylinder is meshed with line elements as shown.

Figure 202: 2D Rigid Body Contact Meshing



Using 3D Rigid Body Contact Meshing

This section describes the basic steps for using 3D rigid body contact meshing.

To define a 3D rigid body for contact meshing:

1. Open the model in the Mechanical application.

2. In the Tree, expand the **Geometry** object so that the body objects are visible.

3. Click the body that you want to define as a rigid body.

4. In the **Details> Definition** view for the body, change the value of the Stiffness Behavior control to **Rigid**.

   Note:

   When you change the **Stiffness Behavior** to **Rigid**, an **Inertial Coordinate System** object is added to the Tree automatically. This **Inertial Coordinate System** represents the centroid of the body.

5. If you wish to control the mesh method, insert a mesh method by right-clicking on the **Mesh** object in the Tree and selecting **Insert> Method**.

   Note:

   The Automatic method appears in the Details View.

6. In the Details View, scope the mesh method to the rigid body.

   Note:

   By default, the **Method** control is set to **Quadrilateral Dominant** for rigid bodies, but you can change the value to **Triangles**. When **Method** is set to **Quadrilateral Dominant**, an additional option called the **Free Face Mesh Type** control is available and can be set to either **Quad/Tri** (default) or **All Quad**.

7. If desired, change the value of the **Element Order** (p. 96) control.

8. Generate the mesh by right-clicking on the **Mesh** object in the Tree and selecting **Generate Mesh**.

Note:

The mesh for the 3D rigid body is created only in the contact region (faces in contact).

Thin Solid Meshing

Thin solid meshing is useful for thin solid bodies where one element through the thickness is desired. This meshing also takes advantage of the Mechanical APDL application's SOLSH190 element or the LS-DYNA thick shell element .

- It may be advantageous to use a **Sizing** control on the faces/body along with mapped Face Meshing controls to give a uniform mesh.

- Virtual Topology (p. 501) may be necessary to satisfy the topological criterion for thin solid meshing.

CAD Instance Meshing

The Meshing application supports pattern instances that have been defined for part features or assembly components in a CAD system, such as Creo Parametric, Parasolid, or Solid Edge. These instance definitions remain in the CAD system. When a model with instances is read in to Workbench, the geometry is read once and then copied for each instance. Similarly, when meshing, the Meshing application generates the mesh once and then copies it for each instance. Support for pattern instances provides these benefits:

- Improved geometry import speed because only one instance of a part is read in

- Improved meshing speed because only one instance of a part is meshed; copies of the first instance's mesh are used for the remaining instances

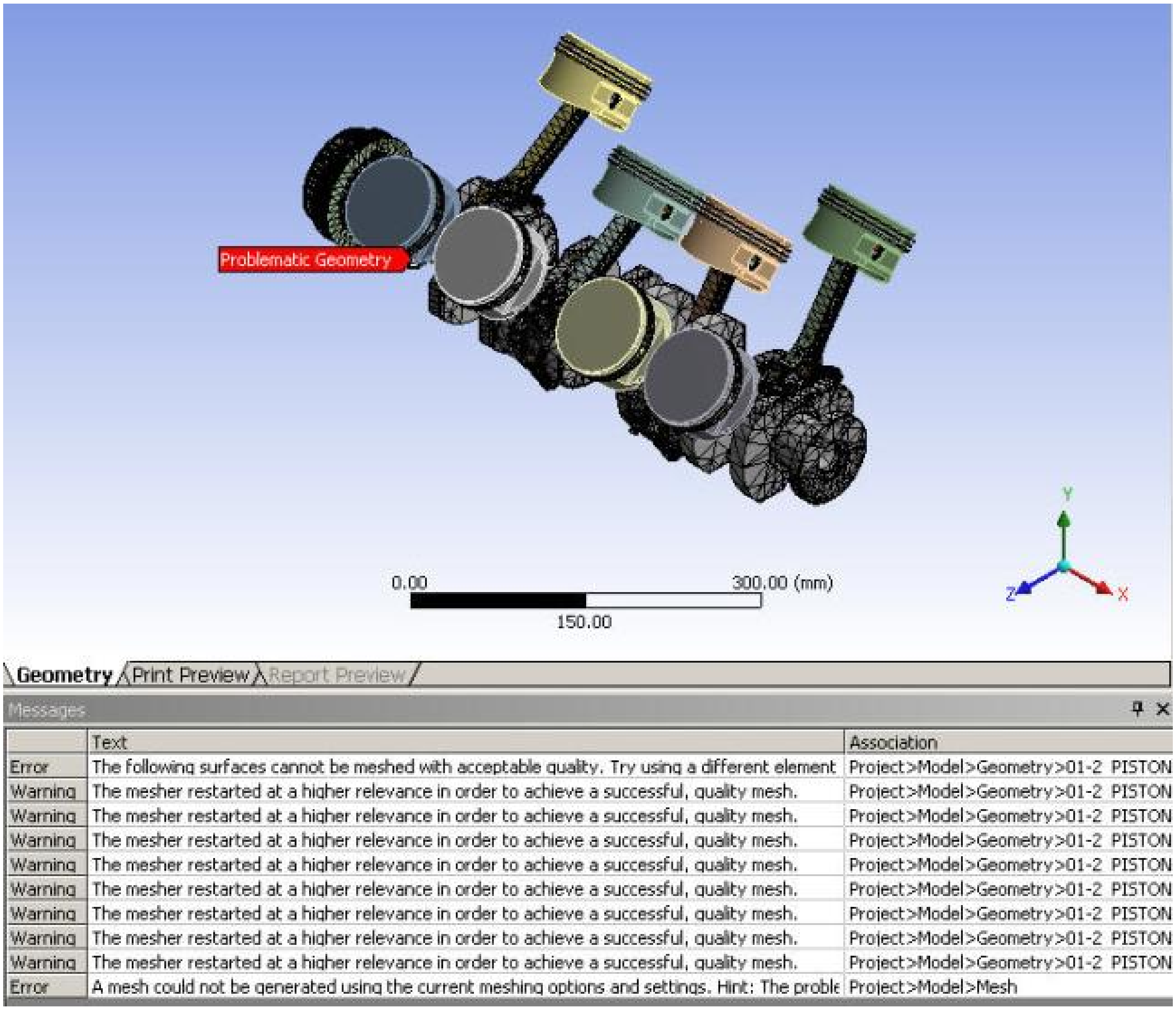
Remember the following information when working with instances:

- Instances of *bodies* are not supported. Single body part instancing is supported, as well as certain variations of instancing of multibody parts (for example, multibody parts can be instanced, but a body cannot be instanced within a single part). For more information, refer to the discussion of feature modeling's effect on instance data in the DesignModeler help.

- If you apply a mesh control to a part that is instanced, each instance must have the same control applied to it or the part will be meshed individually. For example, if your model consists of four parts that are instanced but only one part has an edge sizing control applied to it, that part will be meshed individually and the other three parts will be meshed once and instanced.

- Instancing is not supported for the following controls. Each part that has any of these controls applied to it will be meshed individually, regardless of whether the control in question is applied to all instances:
  - Sphere of influence (p. 256) sizing
  - Body of influence (p. 257) sizing
  - Contact sizing (p. 263)
  - Rigid body meshing (p. 424) (presence of any rigid bodies in the part)
- You can use the **Extend Selection** command to select a set of instances, which can be useful for performing tasks such as applying loads or creating a Named Selection. To select a set of instances, first select one instance, then click the **Extend Selection** drop-down menu and choose **Extend to Instances**. **Extend to Instances** searches for all remaining instances that are defined for the currently selected instance. The **Extend to Instances** menu option is enabled only when pattern instances are defined in the current model.
- When using mesh methods or controls that require both body scoping and face scoping (such as Sweep (p. 223) with source face selection; or inflation (p. 291)), you can use the Object Generator to set up the model for instancing. For details, refer to Generating Multiple Objects from a Template Object in the Mechanical help.
- Because the mesh is copied from the first instance to the remaining instance, it follows that each instance will be meshed with the same number of nodes and elements. To confirm this, after CAD instances are successfully meshed, click the **Geometry** object in the Tree Outline. Then click the **Worksheet** toolbar button. When the **Worksheet** appears, click the **Nodes** heading, which sorts the data on that column and allows you to view the matching numbers of nodes and elements for each instance.
- Error handling for instances is also copied. For example, if the mesher fails on one instance, all instances will fail; however, you will receive a warning message for each instance. Refer to Figure 203: Error Handling for Instances (p. 429), which shows a model containing eight identical pistons.

Figure 203: Error Handling for Instances



Meshing and Hard Entities

The two types of hard entities are hard edges and hard points. Hard entities are usually defined in the DesignModeler application or a CAD system.

A hard point is an embedded point in a face or edge. Hard points are captured by nodes during the meshing process. Hard points are not supported for the following mesh methods or controls:

- General sweeping (p. 323)
- 3D inflation (p. 145)

A hard edge is an embedded edge in a face. There are two main types of embedded edges:

- An edge in which one vertex of the edge touches the face boundary, but the other vertex does not touch a face boundary

- An edge in which neither of the vertices touches the face boundary

Hard edges are not supported for the following mesh methods or controls:

- General sweeping (p. 323)
- 2D or 3D inflation (p. 145)

Note:

- If the methods and controls listed above are required in your mesh, insert a Virtual Topology and use the **Simplify Faces** (p. 508) option to remove the hard entities.
- Other mesh methods have certain limitations in how hard points and hard edges are handled. For more information, refer to Limitations of Using Hard Entities with Other Mesh Controls (p. 430).

Spot Welds

Spot welds are used to connect individual surface body parts together to form surface body model assemblies, just as contact is used for solid body part assemblies. Spot welds are usually defined as hard points in the DesignModeler application or a CAD system:

- When a model is imported into the Meshing application, the mesher simply treats the hard points as embedded points.
- Upon import to the Mechanical application, spot welds are automatically generated where hard points are defined in the model.

For related information, refer to Point in the DesignModeler help and Spot Welds in the Mechanical help.

Limitations of Using Hard Entities with Other Mesh Controls

Limitations of hard entities include the following:

- 3D inflation (p. 145) does not support hard entities of either type. If inflation is applied, a warning message is issued to indicate the hard entities were ignored.
- 2D inflation (p. 145) supports hard points only.
- General Sweeping (p. 323) does not support hard points.
- The MultiZone Quad/Tri (p. 246) and MultiZone (p. 228) mesh methods do not respect hard entities unless their topology is protected (p. 180).
- For the MultiZone (p. 228) mesh method only, the faces that contain the hard entities must be selected as source faces for the hard entities to be respected.
- Hard edges may exist accidentally in a CAD model due to Boolean operations with tight tolerances or other such operations. These accidental hard edges may be undesired, in which case you

should remove them by using the **Virtual Topology: Face Simplify** feature or defeaturing them within the DesignModeler application or a CAD system.

Baffle Meshing

The Meshing application provides support for meshing 0-thickness walls, or baffles, as non-manifold faces of a solid body. For such models, you do not have to adjust the mesh size to capture the thin regions.

Characteristics and limitations of baffle meshing include:

- For part/body level meshing, baffle meshing is supported by the Patch Conforming Tetra (p. 200), Patch Independent Tetra (p. 200), and MultiZone (p. 228) mesh methods only. If you apply any other mesh method to a body containing baffles, the mesh method will be suppressed, and the reason (not supported) will be reported in the **Active** read-only field in the Details View. In such cases, the body will be meshed with the Patch Conforming Tetra mesh method.

  Baffle meshing is also supported for assembly meshing algorithms. See Assembly Meshing (p. 367).

- When the MultiZone mesh method is used, the body with a baffle must be meshed with a free mesh of tetrahedral elements. For this reason, you must set the **Free Mesh Type** to **Tetra** for bodies with baffles.

- The Patch Conforming Tetra and Patch Independent Tetra mesh methods support inflation of baffles. The MultiZone mesh method does not support inflation of baffles.

- When the Patch Conforming Tetra mesh method is used, inflation layers will stair step (p. 158) at free boundary edges of the baffles.

- When the Patch Independent Tetra mesh method is used and Collision Avoidance (p. 158) is set to **Stair Stepping**, inflation layers will stair step at free boundary edges of the baffles. However, if **Collision Avoidance** is set to **Layer Compression**, full prism columns appear at the free boundary edges.

- **Program Controlled** (p. 148) inflation is supported (that is, if you select **Program Controlled** inflation, baffles are automatically selected to be inflation boundaries unless they are in a Named Selection).

- Only two-sided growth cases for inflation are supported.

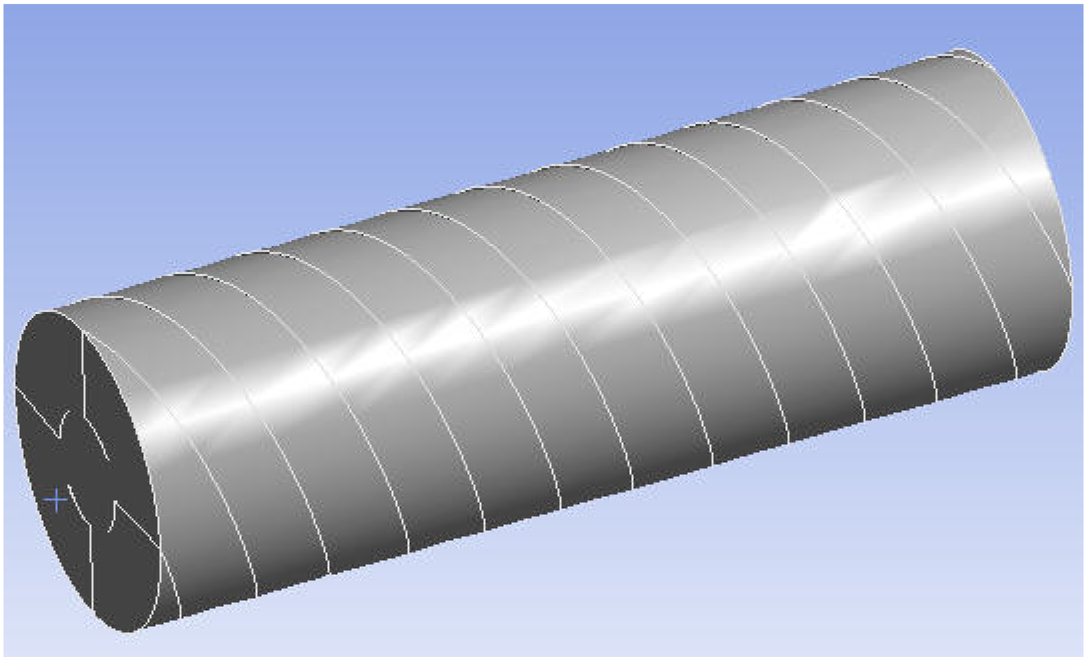
- Pyramid transitions are supported.

- Prism/pyramid elements are not supported for meshing crossed/intersecting baffles.

- There is a single set of nodes on the internal face.

Figure 204: Cylinder Containing Baffles (p. 432) shows a model of a cylinder that contains a series of baffles.

Figure 204: Cylinder Containing Baffles



In Figure 205: Section Cut Showing Baffle Meshing (p. 432), the model was meshed using **Program Controlled** inflation. Since the baffles were not in a Named Selection, they were automatically selected to be inflation boundaries. A section plane was activated to view a section cut through the model, and the baffle faces were selected in the **Geometry** window.

Figure 205: Section Cut Showing Baffle Meshing

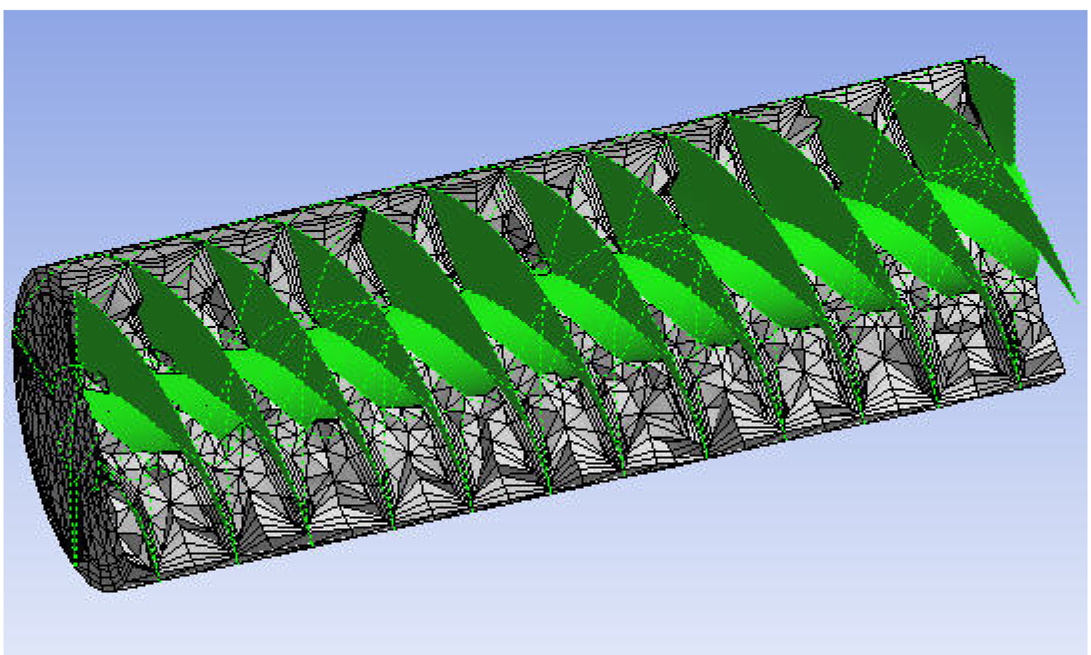
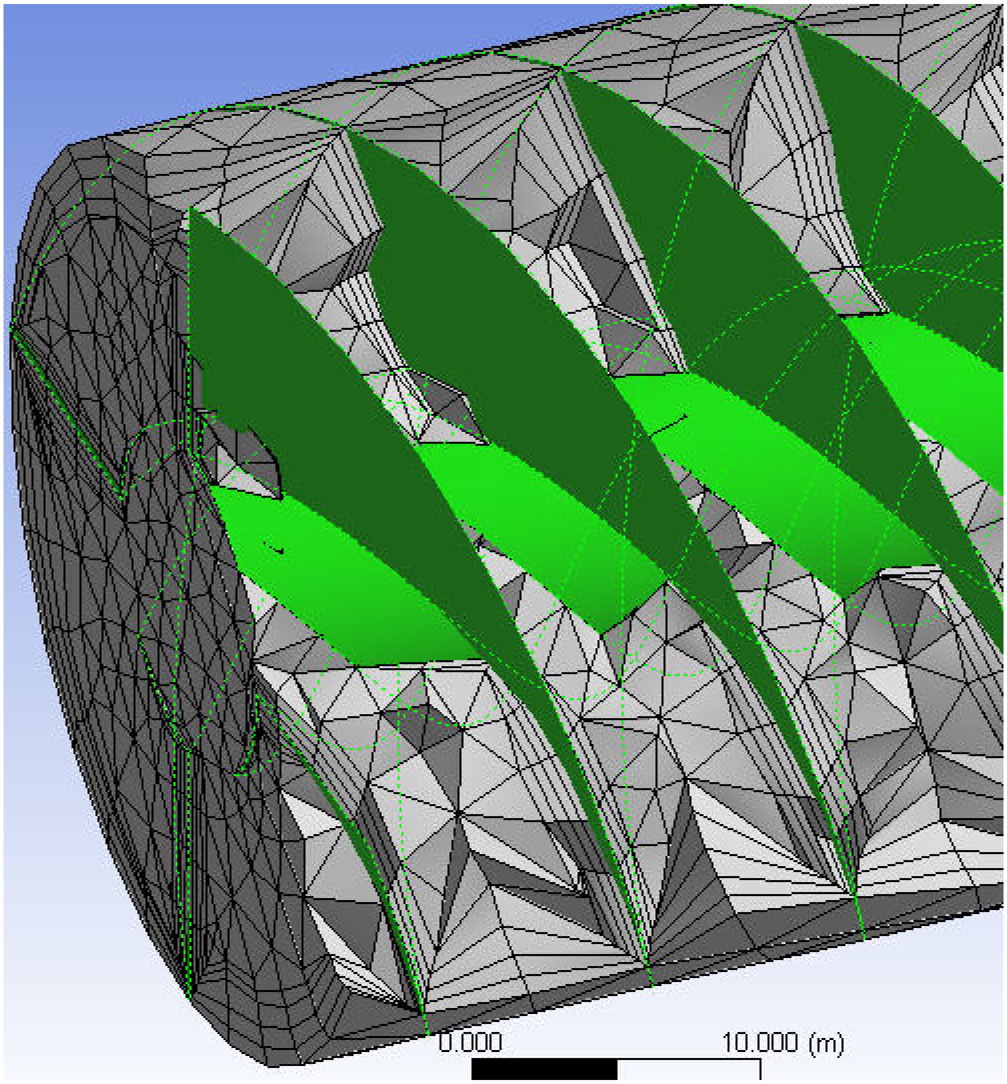


Figure 206: Detail of Inflation on Baffles (p. 433) shows a detailed view of the inflation layers on the baffles.

Figure 206: Detail of Inflation on Baffles



Parallel Part Meshing

You can control three mechanisms in Ansys Workbench that operate in a parallelized manner:

- Remote Solve Manager Design Point updates. Refer to RSM Configuration.
- Parallel Part meshing: **Tools**> **Options**> **Number of CPUs for Parallel Part Meshing**
- Individual mesh methods (**MultiZone Quad/Tri**, **Patch Independent Tetra**, and **MultiZone** only): **Tools**> **Options**> **Number of CPUs for Meshing Methods**

For the most efficient use of machine resources, it is important that the running processes do not over-saturate the processing cores or the available memory. You must allocate processing cores to each of these mechanisms in a way that provides the most benefit for your workflow. When parallel part meshing is invoked with the default number of CPUs, it automatically uses the cores of all available CPUs with the inherent limitation of 2 gigabytes per CPU core.

Parallel Part Meshing Best Practices

Best practices include:

- Know how many physical processing cores are available.
- If you are using Remote Solve Manager (RSM), meshing is done serially. This option cannot be overridden.
- For non-RSM Design Point updates, meshing is done serially by default. You can override this option by setting the option **Number of CPUs for Parallel Part Meshing** explicitly under **Meshing>Advanced Options**.
- If you are using more than one processor for individual meshing methods, be sure to set a balance between the two meshing options if you are using methods that support the **Number of CPUs for Meshing Methods** option (**MultiZone Quad/Tri**, **Patch Independent Tetra**, and **MultiZone**). They should be set to an explicit value greater than 0.

  For example, if you are using an 8 core system, setting **Number of CPUs for Parallel Part Meshing** to **3** and **Number of CPUs for Mesh Methods** to **3** will provide a good balance. If the mesh methods that you typically use support the **Number of CPUs for Meshing Methods** option, setting the **Number of CPUs for Parallel Part Meshing** to **2** and the **Number of CPUs for Meshing Methods** to **4** or **5** may potentially provide more benefit.
- If you are using parallel part meshing only, you can set the **Number of CPUs for Parallel Part Meshing** to **0**. In such cases, the software uses as many cores as possible.
- For Parallel Part meshing we recommend turning off hyper-threading as this may lead to degradation of parallel performance.
- Note that Parallel Part meshing does not support the following mesh controls:
  - Assembly meshing
  - Contact Sizing
  - Fracture
  - Mesh Match via Symmetry
  - Morph Service/Morphing
  - Pinch
  - Post Connection
  - Post-Inflation
  - Preview Surface Mesh/Preview Inflation
  - Retry
  - Refinement

Meshing: Mesh Control Interaction Tables

This section presents the effects of applying combinations of mesh controls on the same part or body. Topics include the meshing implication when one mesh method is applied in combination with another mesh method, and the effects of applying various mesh controls in combination with the various mesh methods.

Interactions Between Mesh Methods

The tables below present the effects of meshing two or more bodies in a multibody part using a combination of different mesh methods:

- Using combinations of surface mesh methods
- Using combinations of solid mesh methods
- Applying a single 3D inflation control on more than one solid body when a combination of mesh methods has been scoped to the bodies
- Applying a 3D inflation control on a solid body when more than one mesh method has been scoped to the body

Note:

- Assembly meshing algorithms (p. 367) cannot be used in combination with any other mesh method.
- The Cartesian (p. 236) mesh method operates at the part level, and does not support interactions with other mesh methods. If one body in a multibody part is scoped to be meshed with the Cartesian mesh method, all bodies will be added to the scoping.
- Refer to Conformal and Non-Conformal Meshing (p. 21) for information about conformal meshing.

The table below describes the automatic sequencing of surface mesh methods when two mesh methods are being used. If all three methods are being used, the automatic sequence is:

1. All Triangles (p. 246)
2. Quad Dominant (p. 245)

3. MultiZone Quad/Tri (p. 246)

Note:

If you are performing selective meshing, you control the sequence. Refer to Selective Meshing (p. 404) for usage notes.

Surface Mesh Method	Surface Mesh Method		
	All Triangles (p. 246)	**Quad Dominant (p. 245)**	**MultiZone Quad/Tri (p. 246)**
All Triangles (p. 246)	N/A	**All Triangles** first	**All Triangles** first
Quad Dominant (p. 245)	**All Triangles** first	N/A	**Quad Dominant** first
MultiZone Quad/Tri (p. 246)	**All Triangles** first	**Quad Dominant** first	N/A

The table below describes the automatic sequencing of solid mesh methods when two methods are being used. If more than two methods are being used, the automatic sequence is:

1. MultiZone (p. 228)
2. General Sweep (p. 323)
3. Thin Sweep (p. 330)
4. Hex Dominant (p. 222)
5. Patch Conforming Tetra (p. 200)
6. Patch Independent Tetra (p. 200)

Note:

- If you are performing selective meshing, you control the sequence. Refer to Selective Meshing (p. 404) for additional usage notes.
- During automatic sequencing of solid mesh methods when inflation has been applied, **Post** inflation is always applied last and uses as its input mesh the complete currently existing part mesh.

Solid Mesh Method	Solid Mesh Method					
	Hex Dominant (p. 222)	**General Sweep (p. 323)**	**Thin Sweep (p. 330)**	**Patch Conforming Tetra (p. 200)**	**Patch Independent Tetra (p. 200)**	**MultiZone (p. 228)**
Hex Dominant (p. 222)	N/A	**General Sweep** first	**Thin Sweep** first	**Hex Dominant** first	**Hex Dominant** first	**MultiZone** first
General Sweep (p. 323)	**General Sweep** first	N/A	**General Sweep** first	**General Sweep** first	**General Sweep** first	**MultiZone** first[1]

Solid Mesh Method	Solid Mesh Method					
	Hex Dominant (p. 222)	**General Sweep (p. 323)**	**Thin Sweep (p. 330)**	**Patch Conforming Tetra (p. 200)**	**Patch Independent Tetra (p. 200)**	**MultiZone (p. 228)**
Thin Sweep (p. 330)	**Thin Sweep** first	**General Sweep** first	N/A	**Thin Sweep** first	**Thin Sweep** first	**MultiZone** first
Patch Conforming Tetra (p. 200)	**Hex Dominant** first	**General Sweep** first	**Thin Sweep** first	N/A	**Patch Conforming Tetra** first	**MultiZone** first
Patch Independent Tetra (p. 200)	**Hex Dominant** first	**General Sweep** first	**Thin Sweep** first	**Patch Conforming Tetra** first	N/A	**MultiZone** first
MultiZone (p. 228)	**MultiZone** first	**MultiZone** first[1]	**MultiZone** first	**MultiZone** first	**MultiZone** first	N/A

1–While mixing **Sweep** and **MultiZone** mesh methods, pre-meshed faces may be used in these ways:

- Mapped faces can be supported as side faces when **MultiZone** or **Sweep** is used to mesh subsequent bodies. The pre-meshed faces may have been generated using either **General Sweep** or **MultiZone**. There are limitations on how the face is mapped. Simple mapped faces (that is, 4-sided) are supported; however, more complicated submapped cases may cause problems.
- Mapped faces can be supported as source faces.
- Free faces (where mesh does not have a quad mapped pattern) can be supported as source faces only.

The table below describes how inflation is handled if you apply a single 3D inflation control on more than one solid body when a combination of mesh methods has been scoped to the bodies.

Method 1	Method 2	Supported Inflation Algorithm
Patch Conforming Tetra (p. 200)	Patch Independent Tetra (p. 200)	Post (p. 158) inflation only
Patch Conforming Tetra (p. 200)	MultiZone (p. 228)	Pre (p. 156) inflation only[1]
Patch Independent Tetra (p. 200)	MultiZone (p. 228)	No inflation allowed; inflation is suppressed
Patch Conforming Tetra (p. 200)	Patch Independent Tetra (p. 200) (+ Method 3 of MultiZone (p. 228))	No inflation allowed; inflation is suppressed
General Sweep (p. 323)	Any other method	No inflation allowed; inflation is suppressed
Thin Sweep (p. 330)	Any other method	No inflation allowed; inflation is suppressed
Hex Dominant (p. 222)	Any other method	No inflation allowed; inflation is suppressed

1– In such cases involving **MultiZone**, the value of the **Inflation Algorithm** control displays as **Pre** but an O-grid-based algorithm specific to **MultiZone** is used. As with the **Pre** inflation algorithm, the mesh is inflated during the meshing process.

The table below describes how inflation is handled if you apply a 3D inflation control on a solid body when more than one mesh method has been scoped to the body. In such cases, the method control that appears lowest in the Tree is respected and therefore inflation is handled as it would normally be handled for that method.

Lowest Method in Tree	Supported Inflation Algorithm(s)
Patch Conforming Tetra (p. 200)	Post (p. 158) or Pre (p. 156) inflation
Patch Independent Tetra (p. 200)	Post (p. 158) inflation only
MultiZone (p. 228)	Pre (p. 156) inflation only[1]
General Sweep (p. 323)	No inflation allowed; inflation is suppressed
Thin Sweep (p. 330)	No inflation allowed; inflation is suppressed
Hex Dominant (p. 222)	No inflation allowed; inflation is suppressed

1– In such cases involving **MultiZone**, the value of the **Inflation Algorithm** control displays as **Pre** but an O-grid-based algorithm specific to **MultiZone** is used. As with the **Pre** inflation algorithm, the surface mesh is inflated first and then the rest of the volume mesh is generated.

Interactions Between Mesh Methods and Mesh Controls

The tables in this section present the effects of applying various mesh controls in combination with the various mesh methods, and include:

- Using mesh controls with the solid meshing methods
- Using mesh controls with the surface meshing methods
- Using mesh controls with the assembly meshing methods

The table below describes the effects of applying the mesh control on the left with each of the solid meshing methods.

Mesh Control	Solid Meshing Methods						
	Patch Conforming Tetra (p. 200)	Patch Independent Tetra (p. 200)	MultiZone (p. 228)	General Sweep (p. 323)	Thin Sweep (p. 330)	Hex Dominant (p. 222)	Cartesian (p. 236)
Body Sizing Control (p. 248)	Supported	Supported	Supported	Supported	Supported	Supported	Supported

Mesh Control	Solid Meshing Methods						
	Patch Conforming Tetra (p. 200)	Patch Independent Tetra (p. 200)	MultiZone (p. 228)	General Sweep (p. 223)	Thin Sweep (p. 230)	Hex Dominant (p. 222)	Cartesian (p. 236)
Face Sizing Control (p. 248)	Supported	Supported	Supported	Supported	Supported	Supported	Supported
Edge Sizing Control (p. 248)	Supported	Supported	Supported	Supported	Supported	Supported	Supported
Sphere of Influence Control (p. 256)	Supported	Supported	Supported, but only influences source face(s).	Supported, but only influences source face.	Supported, but only influences source face(s).	Supported	Not supported
Body of Influence Control (p. 257)	Supported	Supported	Supported, but only influences source face(s).	Supported, but only influences source face.	Supported, but only influences source face(s).	Supported	Not supported
Contact Sizing Control (p. 263)	Supported	Supported	Supported	Supported	Supported	Supported	Not supported
Refinement Control (p. 264)	Supported	Not supported	Not supported	Supported	Supported	Supported	Not supported
Mapped Face Control (p. 265)	Supported	Not supported	Supported	Supported	Supported	Supported	N/A. All faces are essentially mapped.
Match Control (p. 280)	Supported	Not supported	Supported, with limitations (p. 280).	Supported	Supported	Not supported	Not supported
Pinch Control (p. 286)	Supported	Not supported	Not supported	Not supported	Supported on sources and targets.	Supported	Not supported
Inflation Control (p. 291)	Supported	Supported	Supported	Supported on source via 2D inflation.	Not supported	Not supported	Supported

The table below describes the effects of applying the mesh control on the left with each of the surface meshing methods.

Mesh Control	Surface Mesh Methods		
	All Triangles (p. 246)	Quad Dominant (p. 245)	MultiZone Quad/Tri (p. 246)
Body Sizing Control (p. 248)	Size control affects elements on body and lower topological entities.		Supported
Face Sizing Control (p. 248)	Size control affects elements on face and lower topological entities.		Supported
Edge Sizing Control (p. 248)	Size control affects element edge lengths on edge.		Supported
Sphere of Influence Control (p. 256)	Inserts elements of specified size within sphere.		Not supported
Body of Influence Control (p. 257)	Inserts elements of specified size within body. Only available when Size Function is on.		Not supported
Contact Sizing Control (p. 263)	Inserts spheres of influence on contact faces in regions within contact tolerance.		Not supported
Refinement Control (p. 264)	Refines elements as post-process.		Not supported
Mapped Face Control (p. 265)	Mapped faces are meshed before any other faces. Interval assignment may affect edge divisions. No sphere of influence support. No mesh based defeaturing or pinch support.		
Match Control (p. 280)	Edge meshes are matched for sheet, 2D, and 3D bodies. Face meshes are matched across bodies. Match controls cannot be applied across multiple parts. Additional restrictions (p. 280) apply.		Not supported
Pinch Control (p. 286)	Supported		Not supported
Inflation Control (p. 291)	Supported		

The table below describes the effects of applying the mesh control on the left with the assembly meshing methods.

Mesh Control	Assembly Meshing Algorithms (p. 367)
Body Sizing Control (p. 248)	Supported
Face Sizing Control (p. 248)	Supported
Edge Sizing Control (p. 248)	Supported
Sphere of Influence Control (p. 256)	Not supported
Body of Influence Control (p. 257)	Supported
Contact Sizing Control (p. 263)	Supported
Refinement Control (p. 264)	Not supported
Mapped Face Control (p. 265)	Not supported
Match Control (p. 280)	Not supported
Pinch Control (p. 286)	Not supported
Inflation Control (p. 291)	Supported

Meshing: Miscellaneous Tools

The miscellaneous meshing tools described in the following sections include:

Generation of Contact Elements

When you load a model into the Meshing or Mechanical application, by default, connections are found between parts that have faces in proximity of each other. Depending on the application, you may want the boundaries common to two parts to be similar, so that contact definitions or non-conformal interface definitions may be more accurate. To get common boundaries between parts in an assembly, you should first imprint all the parts with each other in SpaceClaim or DesignModeler (p. 21). Then, when you edit the model in the Meshing or Mechanical application, you should define specific contact conditions.

One of those conditions is tolerance, which controls the extent of contact between parts in an assembly. Tolerance is set as a percentage of the bounding box of the assembly. The bounding box is the smallest volume that the assembly will fit in. You can change the tolerance (between -100 and 100) in the **Options** dialog box under the Mechanical application's **Connections** category.

The higher the number, the tighter the tolerance. A loose tolerance generally increases the number of contact faces and areas of contact between parts, while a tight tolerance will decrease the number of contact faces.

Each face of a part is checked against the faces of other parts in the assembly. A connection is generated between any faces within the tolerance. You can use overlap tolerances to further limit which faces are in contact if you want only the faces that fully overlap to be found in contact.

When solving in the Mechanical solver, the elements for the two sets of faces that make up a contact pair are compared. Contact elements are generated for element pairs that are within the tolerance, but element pairs outside the tolerance are ignored.

Note:

This discussion is applicable to part-based meshing. The concept of an assembly of parts should not be confused with assembly meshing (p. 20). For assembly meshing, contact provides feature capturing (p. 395) and contact sizing (p. 398) to close gaps; no contact elements or special interface handling are involved.

Recommendations for Defining Contact for CFD Analyses

CFD users should be aware of the following recommendations:

- The **Auto Detect Contact On Attach** option controls whether contact detection is computed upon geometry import. If you do not want contact detection to be computed, make sure that it is disabled by selecting **Tools> Options** from the Ansys Workbench main menu, and then selecting either the Mechanical or Meshing category as appropriate. The option is enabled by default in both applications.

- If you are an Ansys Fluent user, you generally want to imprint all the parts with each other in SpaceClaim or DesignModeler (p. 21) as mentioned above. Failing to imprint parts may lead to connections that have cyclic redundancy and may fail to output to the solver.

- For Ansys Fluent users, a boundary zone type of *INTERFACE* is assigned automatically to the contact source and contact target entities that compose contact regions at the time of mesh export. See Special Cases (p. 54) for details.

Renaming Mesh Control Tools

You can rename any of the mesh control tool objects to include the name assigned to the part or body. To do this, use a right mouse button click on the object and choose **Rename Based on Definition** from the context menu. For example, if you scope a **Refinement** tool to a body named **Tube** and choose **Rename Based on Definition**, the mesh control tool name changes from **Refinement** to **Refinement on Tube**. The name change is reflected both in the tree and as a label on the body.

Mesh Numbering

The Mesh Numbering feature allows you to renumber the node and/or element numbers of a generated meshed model consisting of flexible parts. The feature is useful when exchanging or assembling models and could isolate the impact of using special elements such as superelements. For details, refer to Mesh Numbering in the Mechanical help.

Mesh Editing

Mesh Editing enables you to improve or refine the quality of a mesh and more efficiently create continuous, conformal meshes for large models with multiple parts. You can move individual nodes, merge nodes together, match nodes, or use mesh connections to join the meshes of topologically disconnected surface bodies and solids.

If the nodes on two different parts are coincident (there are duplicate nodes at the same location), it is faster to merge the nodes to join them. If the nodes are at different locations, you should use mesh connections or contact matches.

Mesh Connections work only for sheet bodies, and **Contact Matches** work only for solid bodies. You can use **Node Move** and **Node Merge** for solid, sheet, and line bodies.

Note:

Ansys DesignSpace licenses do not support the Mesh Editing feature.

The Mesh Editing feature is described in the following sections:

Inserting a Mesh Edit Object

Mesh Editing tools, such as Mesh Connections (p. 444), Contact Matches (p. 455), Node Merge (p. 467), and Node Move (p. 471) are listed in the tree hierarchy as children of the **Mesh Edit** object.

To insert a **Mesh Edit** object:

- Right-click the **Model** object and choose **Insert** > **Mesh Edit**.

  If you have already created a **Mesh Edit** object, the option will not be available from the **Insert** menu.

- Right-click the **Mesh** object and choose **Insert**, then choose any of the following:

  – **Mesh Connection Group**

  – **Manual Mesh Connection** (for manual mesh connections)

  – **Contact Match Group**

  – **Contact Match** (for manual mesh contact match)

  – **Node Merge Group**

  – **Node Merge**

  – **Node Move**

  – **Pull**

  A parent **Mesh Edit** object is created automatically.

- On the **Mesh Edit** toolbar, select **Mesh Edit**, and then select an option.

  A parent **Mesh Edit** object is created automatically.

When you add a **Manual Mesh Connection**, **Contact Match**, or **Node Merge** object, the corresponding group object is created as well.

Mesh Connections

Mesh connections enable you to join the meshes of topologically disconnected surface bodies that may reside in different parts. They are an alternate option to connecting the geometry (for example, by using the DesignModeler application to repair small gaps). However, geometry tolerances are tighter than the tolerances used by mesh connections and often lead to problems in obtaining conformal mesh.

Scoping

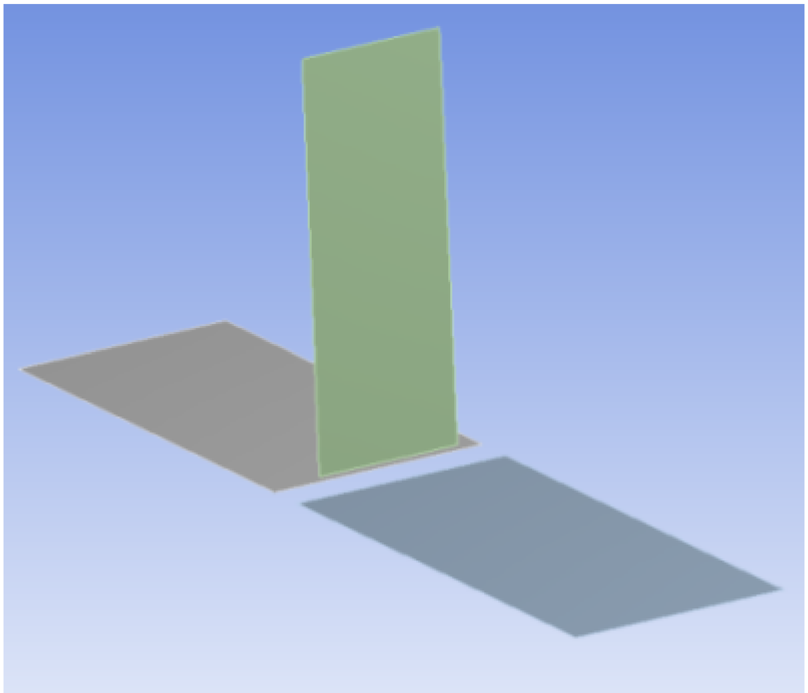
With mesh connections, the connections are made at the mesh level and tolerance is based locally on mesh size. Connections are made edge-to-face or vertex-to-face; they connect edge(s) or vertices on face(s) to another face to pinch out the gap and create conformal mesh between the edge(s) and face(s).

Since mesh connections are a post mesh process–the mesh is pinched in a separate step after meshing is complete–the base mesh is stored to allow for quicker updates. That is, if you change a mesh connection or meshing control, only local re-meshing is required to clean up the neighboring mesh.

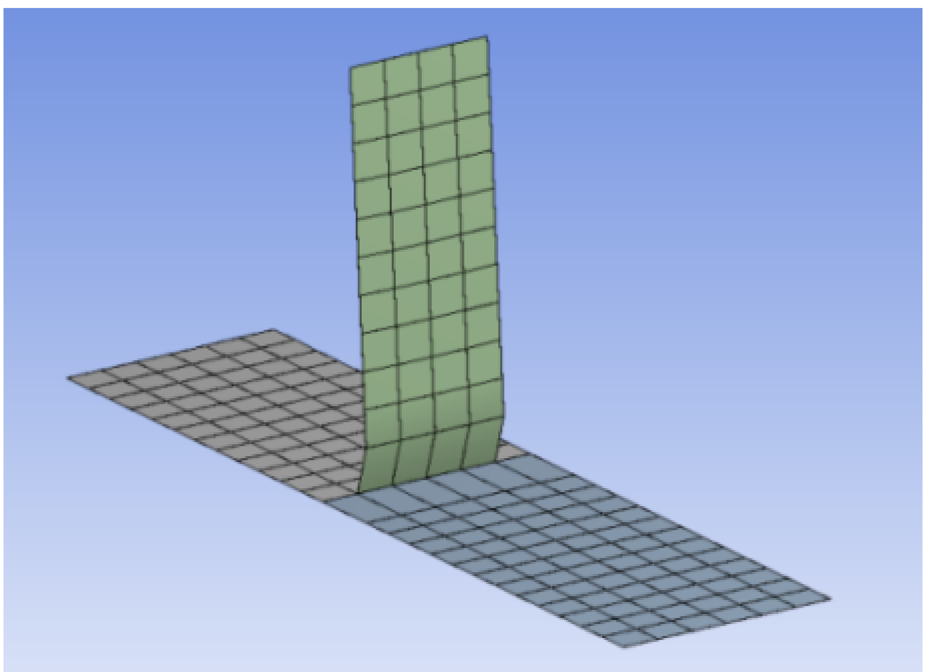
Note:

With Ansys Workbench Release 16.0, post pinch behaviors are migrated into **Mesh Connections**. When you regenerate a mesh that was created using **Pinch Behavior** settings, the new mesh might report different results than the previous mesh.

Surface Bodies With No Shared Topology:



Same Surface Bodies With Edge-To-Edge Mesh Connection Established:



Treatment of Legacy Databases

Resuming a legacy database (prior to v16), the application moves **Mesh Connection** objects and the associated **Connection Group** folders that contain them to the new **Model**-level **Mesh Edit** parent folder.

Application

To apply mesh connections:

1. Insert **Mesh Connection** objects automatically or manually.

   - Mesh connections can be automatically generated using the **Detect Mesh Connections** option available from the right-click context menu of the **Mesh Connections** or **Mesh Connection Group** folder. The **Tolerance Value**, pairing type, and other properties used for auto detection can be set in the Details view of the **Mesh Connection Group** folder under the **Auto Detection** category. Sheet thickness can also be used as a Tolerance Value.

     The automatic mesh connections feature is very helpful, but it can only detect edge-to-face connections. If you need to define edge-to-edge connections, you will need to define them manually. The feature can also find and create connections that you may not want. Always review the connections, or at least be aware that if problems arise, they may be due to automatically generated mesh connections.

   - For more control, or to control the engineering design, you may want to insert **Mesh Connection** objects manually.

     Highlight the **Mesh Connection Group** folder and select the **Manual Mesh Connection** option on the **Mesh Edit** context tab, or right-click the object and select **Insert** > **Manual Mesh Connection**.

     The **Manual Mesh Connection** option is also available when the top-level **Mesh/Mesh Edit** folder is selected. Selecting the option at this level adds a **Mesh Connection Group** object in addition to a **Mesh Connection** object.

     You can also select one or more Contact Regions or the **Contacts** folder, right-click, and select **Create**>**Mesh Contact(s) or Connection(s)**. This option enables you to create Mesh Connection objects from Contact Regions. The application scopes the new Mesh Connection objects to the geometries of the Contact Region(s) and sets the tolerance to be equal to the trim tolerance of the contact region. The Mesh connections are added into a new Mesh Connection Group folder.

2. In the Details view specify **Primary Geometry** and **Secondary Geometry**.

   - "Primary" indicates the topology that will be captured after the operation is complete. In other words, it is the topology to which the secondary topologies in the connection are projected.

   - "Secondary" indicates the topology that will be pinched out during the operation. In other words, it is the topology that is projected and merged with the primary.

The primary geometry can be one or more faces or edges. The secondary geometry can only be one or more edges or vertices. When specifying faces, the annotation is displayed on both sides of the faces.

> **Note:**
>
> Mesh connections support common imprints, which involve multiple secondaries connected at the same location to a common primary. See Common Imprints and Mesh Connections (p. 450).

3. In the Details view specify **Tolerance**. The **Tolerance** here has a similar meaning to the **Tolerance Value** global connection setting, and is represented as a transparent sphere. See Tolerances Used in Mesh Connections (p. 448) for details about **Tolerance** and how it relates to the **Snap Tolerance** described below.

4. For edge-to-face mesh connections only, in the Details view specify **Snap to Boundary** and **Snap Type**. When **Snap to Boundary** is **Yes** (the default) and the distance from a secondary edge to the closest mesh boundary of the primary face is within the specified snap to boundary tolerance, nodes from the secondary edge are projected onto the boundary of the primary face. The joined edge will be on the primary face along with other edges on the primary face that fall within the defined pinch control tolerance. See Pinch Control (p. 286) for details.

   Snap Type appears only when the value of **Snap to Boundary** is **Yes**.

   - If **Snap Type** is set to **Manual Tolerance** (the default), a **Snap Tolerance** field appears where you may enter a numerical value greater than 0. By default, the **Snap Tolerance** is set equal to the pinch tolerance but it can be overridden here. See Tolerances Used in Mesh Connections (p. 448) for details about **Snap Tolerance** and how it relates to the **Tolerance** described above.

   - If **Snap Type** is set to **Element Size Factor**, a **Primary Element Size Factor** field appears where you may enter a numerical value greater than 0. The value entered should be a factor of the local element size of the primary topology.

5. Highlight the **Mesh Edit** folder and choose **Generate** (right-click and choose from context menu). The surface bodies are displayed and show the mesh connections.

6. If necessary, review the mesh connections:

   a. Select one or more **Mesh Connection** or **Mesh Connection Group** objects, right-click, and select **Create Named Selections**.

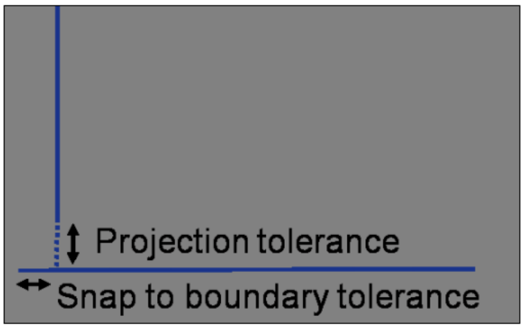
      A named selection is created for each mesh connection you selected. If you selected a mesh connection group, a named selection is created for each mesh connection within the group. Each named selection is automatically given the same name as the mesh connection from which you created it.

   b. Click a named selection to view the mesh for the mesh connection.

Tolerances Used in Mesh Connections

You can set two separate tolerances to define mesh connections. Setting appropriate tolerances is often critical to obtaining high quality mesh that adequately represents the geometry you want to capture.

- **Tolerance** – Projection tolerance to close gaps between bodies.
- **Snap Tolerance** – Snap to boundary tolerance to sew up mesh at the connection (applicable to edge-to-face mesh connections only).



The **Tolerance** value is used to find which bodies should be connected to which other bodies. Setting a larger **Tolerance** connects more bodies together, while setting it smaller may cause some connections to be missed. For this reason, you might want to set this to a larger value than needed. Setting a smaller value can avoid problems in automatic mesh connection creation, but can also result in other problems because the tolerance used in meshing is inherited from automatic mesh connection detection settings.

Using a Large Tolerance Value

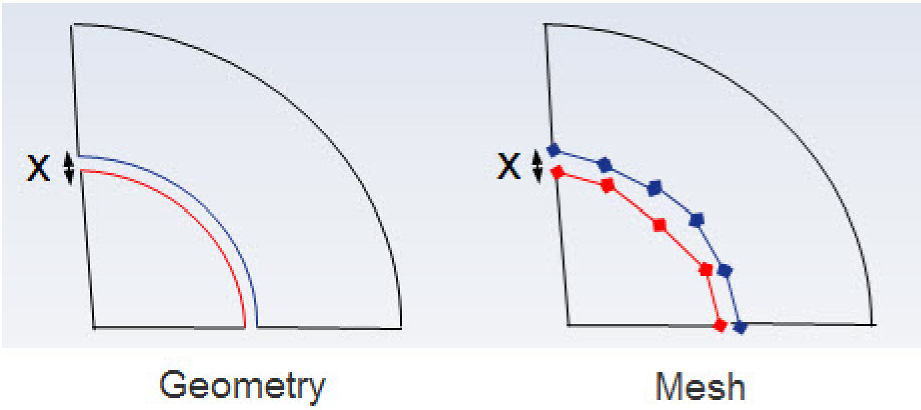
For a large assembly for which you do not want to define mesh connections manually, automatic mesh connection detection provides many benefits. Setting a large **Tolerance** value to find connections yields more connections, which provides a higher level of comfort that the model is fully constrained. However, larger values can be problematic for the following reasons:

- When more automatic mesh connections are created, more duplicates can be created and the mesher decides ultimately which connections to create. In general, making these decisions yourself is a better approach.
- The **Snap Tolerance** defaults to the same value as the **Tolerance**. If the value of **Tolerance** is too large for **Snap Tolerance**, the mesher may be too aggressive in pinching out mesh at the connection, and hence the mesh quality and feature capturing may suffer.

Using a Small Tolerance Value

When mesh connections are generated automatically, the **Tolerance** is used on the *geometry* edges and faces to determine which entities should be connected. However, the connections themselves are not generated until meshing occurs. Because the connections are performed on nodes and elements of the mesh rather than on the geometry, the tolerances do not translate exactly.

For example, in the case below, you would want to set a **Tolerance** that is slightly larger than the gap in the geometry. If the gap is defined as *x* and the tolerance is set to *x*, automatic mesh connection detection could find the connection, but the meshing process may result in mesh that is only partially connected.



Tips for Setting Tolerances

As detailed above, setting the correct tolerance can be very important, and in some cases may require some speculation and/or experimentation. The following tips may help:

- You can adjust the **Tolerance** used to generate automatic mesh connections after the connections are found. Sometimes it is a good idea to use one **Tolerance** value to find the mesh connections, select all the mesh connections, and then reduce or increase the **Tolerance** later.
- Having **Snap to Boundary** turned on and using a **Snap Tolerance** are not always advisable. It depends on the model and the features you want to capture.

Mesh Sizing and Mesh Connections

Mesh size has an effect on the quality and feature capture of a mesh connection as follows:

- Mesh size always affects the base mesh, as features are only captured relative to mesh size.
- During mesh connection processing, the base mesh is adjusted according to the common imprint/location. In cases where there is a large projection or a large difference in mesh sizes between the primary entity and the secondary entity, the common edge between bodies can become jagged. Also, as local smoothing takes place, there can be some problems in transition of element sizes. You can often use one of the following strategies to fix the problem:
  - Use more similar sizes between source and target.
  - Improve the tolerance used by mesh connections (either for projection, or for snapping to boundary).
  - Adjust the geometry's topology so that the base mesh is more accommodating to the mesh connection.

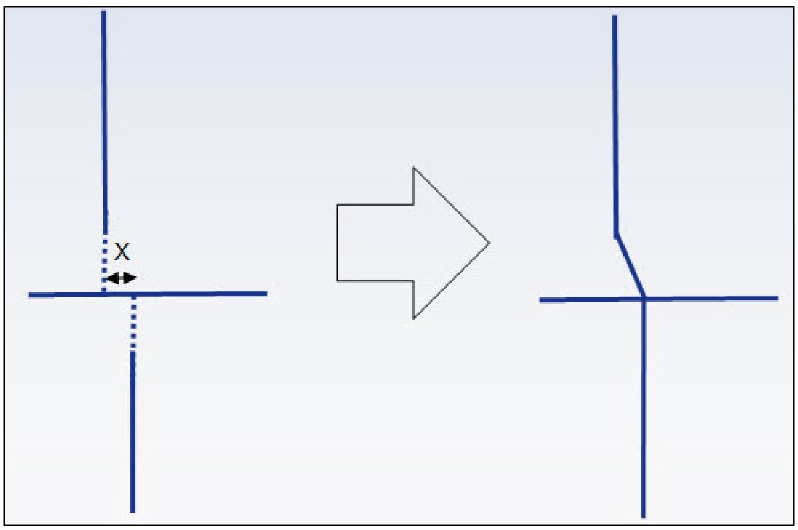
See Common Imprints and Mesh Connections (p. 450).

Common Imprints and Mesh Connections

The tolerance for common imprints comes from the minimum element size in the footprint mesh, which is the horizontal plate in the example below. Common imprints are made if the gap between imprints is smaller than or equal to 5 percent of the element size in the connection region. For this reason, setting the mesh size appropriately is important to control whether the imprints will be common or not.

For example, in the case shown below, if you want a common imprint, the minimum element size is set to **Yes**) should be >*x*.

In this case, you could scope local face mesh sizing on the horizontal plate to control the sizing.



Mesh Connections for Selected Bodies

You can select a geometric entity and lookup the mesh connection object in the tree outline. To find the relevant mesh connection object:

- Right-click a geometric entity, and then click **Go To** > **Mesh Connections for Selected Bodies**.

Mesh Connections Common to Selected Bodies

You can select a pair of geometric entities and lookup the shared **Mesh Connection** object in the tree outline. To find a relevant mesh connection object:

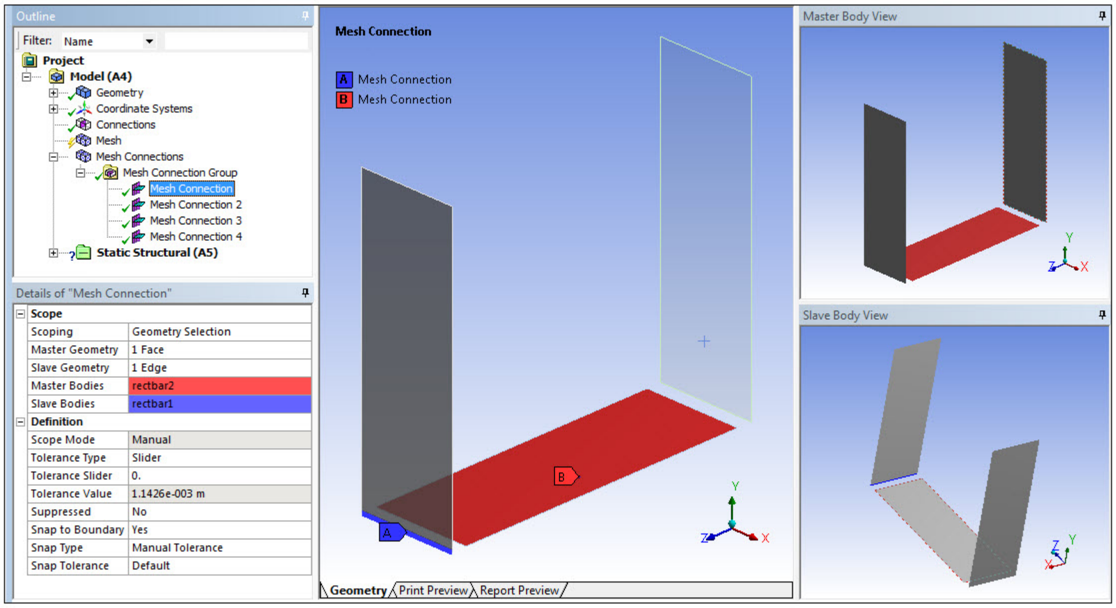
- Select the appropriate pair, and then click **Go To** > **Mesh Connections Common to Selected Bodies**.

This option can be helpful for finding spurious mesh connections, in which case duplicates can be removed.

Displaying Multiple Views of Mesh Connections

Use the **Body Views** button on the **Mesh Edit** Context tab to display parts in separate auxiliary windows.

For closer inspection of mesh connections, you can use the **Show Mesh** option on the Display Context tab along with **Body Views** and the **Sync Views** toggle button. When the Body Views button is engaged, any manipulation of the model in the Geometry window will also be reflected in both auxiliary windows. The **Body Views** toggle button enables you to display parts in separate windows and the **Sync Views** toggle button, when activated, causes any change in the Geometry window to also be reflected in the auxiliary windows.



Merging Mesh Connections

Mesh connections can be merged by selecting the desired objects, right-clicking, and selecting the Merge Selected Mesh Connections option. During the process, the application deletes the original objects and creates a new Mesh Connection object with a combined scoping.

Mesh Connections can only be merged under the following criteria:

- The mesh connections are not already connected.
- The primary and secondary geometry must have similar topology for the connections being merged. For example, if you are merging two connections and the first has a face for the primary geometry and the second has an edge for the primary geometry, the connections cannot be merged. If both primaries are faces and both secondaries are edges, the connections can be merged.

When mesh connections are merged, the new mesh connections contain the merged set of entities as primaries and secondaries.

Note:

Be aware that the merge operation process can create undesired connections. This can create a connection that is not appropriate for mesh generation.

Diagnosing Failed Mesh Connections

The state of each mesh connection is displayed in the Tree Outline. For a description of the various states, see Understanding Mesh Connection and Contact Match States (p. 539).

General Failures

In the event of a general mesh connection failure, the following approach is recommended:

1. Select an ignored or failed mesh connection shown in the tree and look at **Control Messages** in the Details View.

   Note:

   You can use the **Filter** to identify **Mesh Connection** objects that are **State>Ignored**. However, if a mesh connection is in an “error” state, it cannot be filtered in the tree.

2. Click **Yes, Click to Display** to display related error messages.

3. Right-click on the error messages:

   a. If a message provides “Problematic Geometry” information:

      i. Select the message, right-click, and select **Show Problematic Geometry** from the context menu.

         This action highlights the geometry in the Geometry window that is responsible for the message.

         Note:

         Any error message that is related to a specific mesh connection will be associated with the secondary geometry in the connection.

      ii. Select the problematic bodies, right-click, and select **Go To > Mesh Connections for Selected Bodies**.

          This action highlights all mesh connections attached to the problematic geometry.

      iii. Review the tolerances and mesh sizes associated with the highlighted connections.

   b. If a message provides “Go to Body” information:

      i. Select the message, right-click, and select **Go to Body** from the context menu.

         This action highlights the object in the Details view that is responsible for the message.

      ii. Review the tolerances and mesh sizes associated with the highlighted body or bodies.

This action highlights all mesh connections attached to the problematic geometry.

Mesh Connection Failure

If you receive an error or warning message about one or more mesh connections:

1. Highlight and review the message.

2. Right-click the message and select the option **Go To Object**. The corresponding **Mesh Connection** object that is at issue becomes active in the tree.

3. Verify that all of the associated properties are properly defined.

4. Right-click the message and select the option **Show Problematic Geometry**. The corresponding **Primary**/**Secondary** geometry that is at issue becomes highlighted in the **Graphics** window.

5. Verify that the all of the associated geometries are properly defined.

Failures Due to Defeaturing from MultiZone Quad/Tri Meshing and/or Pinch Controls

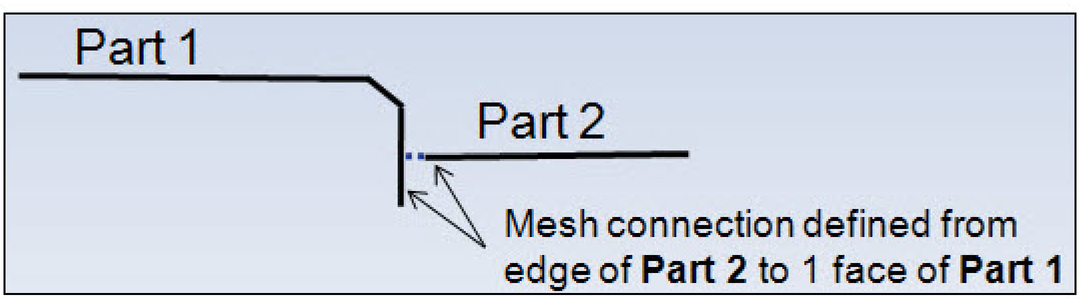
Due to the patch independent nature of the MultiZone Quad/Tri (p. 246) mesh method, a connection may fail because the mesh is associated with some face of the body but not with the face that is involved in the connection. This type of mesh connection failure, which may also occur when pinch controls are defined, is the result of the part mesh being significantly defeatured prior to mesh connection generation. To avoid mesh connection failures when using **MultiZone Quad/Tri** and/or pinch controls, use one of the following approaches:

- Use virtual topology (p. 501) to merge the faces of interest with the adjacent faces to create large patches, and then apply mesh connections to the patches.

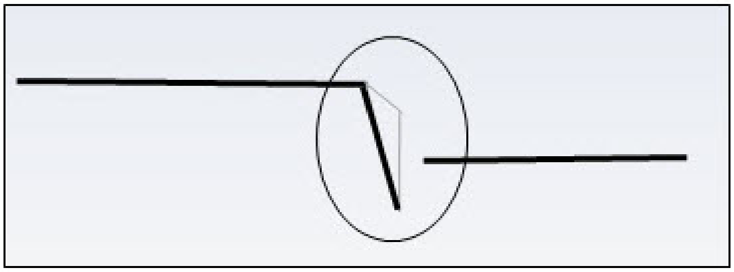
- Protect small faces in mesh connections by defining Named Selections.

The software does not automatically extend the connection region because doing so may lose the engineering intent of the model.

For example, consider the two parts shown below.



If you are using the **MultiZone Quad/Tri** mesh method or pinch controls, the part mesh may look like the one shown below. Notice that one face has been defeatured out.



In this case:

- If the defeatured face is the one defined in the mesh connection, the connection will fail.
- If the other face is the one defined in the mesh connection, the connection will succeed.
- If you include both faces in the mesh connection, the connection will succeed.

Since you cannot always control which face is defeatured, the most robust and recommended approach is to include both faces in the mesh connection.

Points to Remember

- After model assembly, you cannot generate new mesh connections in a mesh that already has mesh connections. Mesh connections only work with a model assembly if the mesh is unconnected in the upstream systems. Because the base meshes from the upstream systems are not available to the assembled model, you must regenerate the mesh to use mesh connections.
- The mesh must be up-to-date before you can generate mesh connections.

  If the mesh is not up-to-date, then the base mesh will be regenerated when you generate the mesh connections.
- Although the tolerance used for finding mesh connections and for generating mesh connections may be the same value, the tolerance itself has slightly different meanings in the two operations. When finding mesh connections, the tolerance is used to identify pairs of geometry edges or face(s)/edge(s). When generating mesh connections, the tolerance is used in pinching together the edge mesh or edge/face mesh. Since the geometry consists of NURBS, and the mesh consists of linear edges, the same tolerance may mean something slightly different in the two operations.

  For example, consider a geometry that consists of two cylindrical sheet parts that share an interface constructed from the same circle. Also consider that you are finding mesh connections with a tolerance of 0.0. In this case, the mesh connection is easily found because the two edges are exactly the same. However, when the mesh connection is being formed, some segments of the edge may fail to be pinched together if the mesh spacing of the two parts is different and thus the tolerance of the edge mesh is different. Also see Tolerances Used in Mesh Connections (p. 448).
- For a higher order element, a midside node along the connection between a secondary and a primary is located at the midpoint between its end nodes, instead of being projected onto the geometry.
- Although mesh connections do not alter the geometry, their effects can be previewed and toggled using the Display Context tab.

- For Error Limits (p. 118), mesh connections support the **Standard Mechanical** option only.
- If you define a mesh connection on topology to which a match control (p. 280), Face Meshing control (p. 265), or inflation control (global (p. 145) or local (p. 291)) is already applied, the mesh connection may alter the mesh, which in turn may eliminate or disable the match, mapped face meshing, or inflation control.
- Mesh connections cannot be mixed with refinement (p. 264) or post inflation (p. 154) controls.
- A mesh connection scoped to geometries (for the primary and the secondary) that lie on the same face are ignored by the mesher, and, as a result, no mesh connection is generated.
- Refer to Clearing Generated Data (p. 496) for information about using the **Clear Generated Data** option on parts and bodies that have been joined by mesh connections.
- Refer to Using the Mesh Worksheet to Create a Selective Meshing History (p. 409) for information about how mesh connection operations are processed by the **Mesh** worksheet.
- Mesh connections are not supported for external mesh models.
- Mesh connections are not supported between solid bodies and sheet bodies in a multibody part, or between sheet bodies and line bodies in a multibody part.

Contact Matches

Contact matches enable you to match mesh nodes between topologically disconnected solids within a specified tolerance. They are an alternate option to imprinting faces (for example, by using the DesignModeler application). Geometry tolerances are typically tighter than the tolerances used by contact matches, which can lead to problems in obtaining conformal mesh. In these scenarios, contact matches provide a more robust option.

Similar to mesh connections, contact matches are performed on mesh nodes. Contact matches can only be face-to-face between solid bodies.

Contact matches are a post-mesh operation, performed after the base mesh has been generated. The base mesh is then stored so that if you change a contact match, only local re-meshing is required to clean up the neighboring mesh. Likewise, if you make any changes to the base mesh, the contact matches must be re-generated.

Considerations for Contact Matches

You should be aware of the following points regarding contact matches:

- After model assembly, you cannot generate new contact matches in a mesh that already has contact matches.

  Contact matches only work with a model assembly if the mesh is unconnected in the upstream systems. Because the base meshes from the upstream systems are not available to the assembled model, you must regenerate the mesh to use contact matches.

- The mesh must be up-to-date before you can generate contact matches.
- Contact matches are only supported for the patch conforming mesh method.

  Contact matches are not supported for the following mesh methods:

  - Assembly meshing (p. 367)
  - Mixed order meshing (p. 196)

- For a higher order element, a midside node along the connection between a secondary and a primary is located at the midpoint between its end nodes, instead of being projected onto the geometry.
- For Error Limits (p. 118), contact matches support the **Standard Mechanical** option only.
- If you define a contact match on a topology to which a match control (p. 280), Face Meshing control (p. 265), or inflation control (global (p. 145) or local (p. 291)) is already applied, the contact match may alter the mesh, which in turn may eliminate or disable the match, mapped face meshing, or inflation control.
- Contact matches cannot be mixed with refinement (p. 264) or post inflation (p. 154) controls.
- A contact match scoped to geometries (for the primary and the secondary) that lie on the same face are ignored by the mesher, and, as a result, no contact match is generated.
- Contact matches are not supported for external mesh models.

How Mesh Size Affects Contact Matches

Mesh size affects the quality and feature capture of a contact match as follows:

- Mesh size always affects the base mesh, as features are only captured relative to mesh size.
- During mesh contact match processing, the base mesh is adjusted according to the common imprint and location.

  In cases where there is a large projection or a large difference in mesh sizes between the primary entity and the secondary entity, there could be problems in the transition of the mesh away from the contact match area. If the mesh size difference is too great, the contact match will not be generated. You can often use one of the following strategies to fix the problem:

  - Use more similar sizes between source and target.
  - Improve the tolerance used by contact matches.

- Adjust the geometry's topology so that the base mesh is more accommodating to the contact match.

- The **Tolerance** also affects how the mesh is matched for these types of bodies. For more information, see How the Tolerance Affects Gaps and Boundaries (p. 458).

How Tolerances Affect Contact Matches

Setting appropriate tolerances is critical to obtaining high quality mesh that adequately represents the geometry you want to capture.

The **Tolerance** value is used to find which mesh nodes on a body should be matched to mesh nodes on another body. Setting a larger **Tolerance** matches more nodes, while setting it smaller may cause some nodes not to be matched. For this reason, you might want to set this to a larger value than needed. Setting a smaller value can avoid problems in automatic contact matching, but can also result in other problems because the tolerance used in meshing is inherited from automatic contact match detection settings.

Considerations for Using a Large Tolerance Value

For a large assembly for which you do not want to define contact matches manually, automatic mesh contact match detection provides many benefits. Setting a large **Tolerance** value to find contact matches yields more matches.

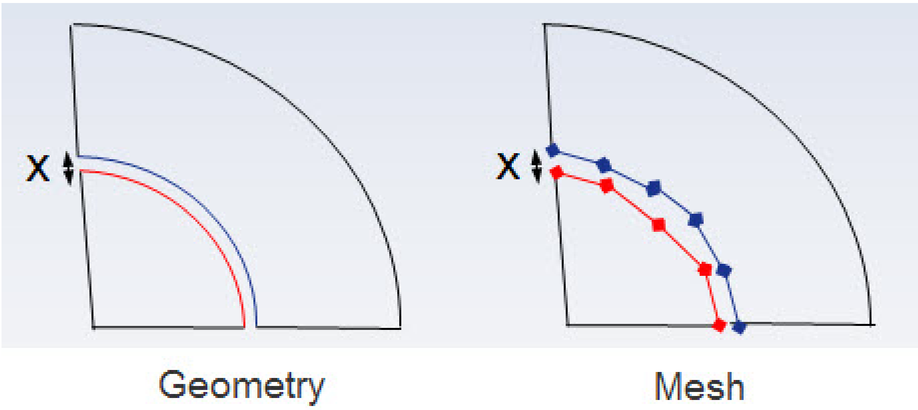
However, larger values can be problematic. When more automatic contact matches are created, more duplicates can be created, which can cause problems when attempting to match the mesh. In general, making these decisions yourself is a better approach.

Considerations for Using a Small Tolerance Value

When contact matches are generated automatically, the **Tolerance** is used on the geometry edges and faces to determine which entities should be matched. However, the contact matches themselves are not generated until after the mesh has been generated. Because the contact matches are performed on nodes and elements of the mesh rather than on the geometry, the tolerances do not translate exactly.

For example, in the case below, you would want to set a **Tolerance** that is slightly larger than the gap in the geometry. If the gap is defined as *x*, and the tolerance is set to *x*, automatic mesh contact match detection could find the connection, but the meshing process may result in mesh that is only partially matched.

Figure 207: Setting the Contact Match Tolerance

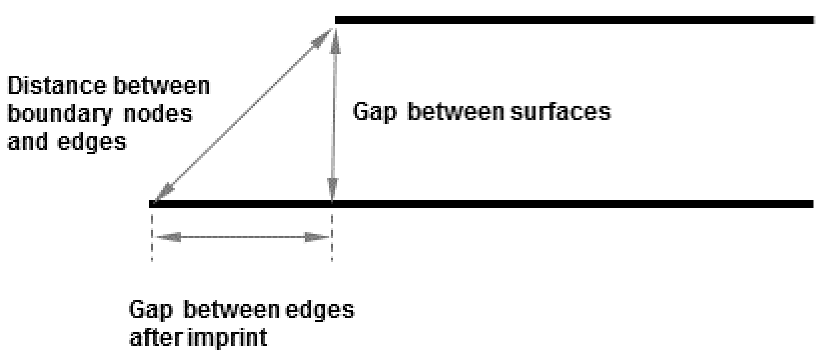


How the Tolerance Affects Gaps and Boundaries

The tolerance also controls how the mesh nodes are matched when there is a gap between the boundary edges of the "Primary" and "Secondary" bodies, or a gap between the bodies themselves, or both.

The following figure shows how the tolerance can be used to match mesh nodes between two solid bodies. In this example, there is a gap between the bodies, as well as a gap between the boundary edges.

Figure 208: Contact Match with Gaps Between "Primary" and "Secondary" Bodies



Assuming that the mesh size is not too large, the following table describes how the mesh will be matched when there are gaps between bodies and boundaries:

Table 4: Mesh Matching for Gaps

If...	Then...
There is a gap between the bodies only	The mesh is matched between the bodies as long as the gap between surfaces is within the specified tolerance. The gap will remain, but the mesh nodes will be matched.

If...	Then...
There is a gap between the boundary edges, but not between the bodies	This mesh is generated first, and the parts are meshed separately. The gap may be meshed depending on the mesh sizes being used and whether there is an imprint in the geometry. **Note:** If there is an imprint in the geometry and you want to remove the gap, you can remove it by inserting a pinch control prior to meshing. For more information, see Pinch Control (p. 286).
There is a gap between both the bodies and the boundary edges	The mesh is matched along the boundaries as long as the gap is within the specified tolerance, and the mesh size is less than or equal to the size of the gap. If the gap is not within the tolerance, or if the mesh size is too large, then the mesh is matched from the edge to the interior.

The mesh size also affects how the mesh is matched for these types of bodies. For more information, see How Mesh Size Affects Contact Matches (p. 456).

Applying Contact Matches

You can apply contact matches by using any of the following methods:

- Apply contact match controls from contact regions (p. 459)
- Automatically apply contact matches (p. 461)
- Manually apply contact matches (p. 462)

Applying Contact Match Controls from Contact Regions

1. In the Tree view, click the **Contacts** object and verify that, in the **Auto Detection** category, the **Tolerance Value** is set appropriately.

   This value controls the relative distance that the application will use to search for contact between parts. However, if the tolerance is set too high, then extraneous contact matches might be generated.

   For more information about auto detection tolerance, see Connection Group.

2. Select the contact regions for which you want to create contact matches by doing one of the following:

- To create contact matches for one or more contact regions, select the **Contact Regions**, right-click, and select **Create** > **Mesh Contact(s) or Connection(s)**
- Or, to create contact matches for all of the contact regions, right-click the **Contacts** folder and select **Create** > **Mesh Contact(s) or Connection(s)**

The application scopes the new **Contact Match** objects to the contact region(s), and sets the tolerance to be equal to the trim tolerance of the contact region. The **Contact Match** objects are added into a new **Contact Match Group** folder.

3. In the Details view, specify the tolerance by setting the **Tolerance Type** and **Tolerance Value**.

   The tolerance here has a similar meaning to the **Tolerance Value** global connection setting, and is represented as a transparent sphere.

   Setting the correct tolerance can be very important, and in some cases may require some speculation and experimentation. You can adjust the tolerance after the contact matches are generated by selecting the contact matches and then changing the **Tolerance Value**.

   For details on setting the tolerance, see How Tolerances Affect Contact Matches (p. 457)

4. Generate the contact matches by doing one of the following:

To...	Do this...
Generate all contact matches	Right-click the **Mesh Edit** folder and select **Generate**.
Generate contact matches for a contact match group	Right-click the **Contact Match Group** object and select **Generate**.

   If the base mesh is out-of-date, it is regenerated. The nodes are matched between the primary and secondary geometries, and a message appears displaying the number of node pairs that were matched.

   If the normals between the primary and secondary faces are misaligned, some contact matches may not be generated.

5. If necessary, review the contact matches:

   a. Select one or more **Contact Match** or **Contact Match Group** objects, right-click, and select **Create Named Selections**.

      A named selection is created for each contact match you selected. If you selected a contact match group, a named selection is created for each contact match within the group. Each named selection is automatically given the same name as the contact match from which you created it.

   b. Click a named selection to view the mesh for the contact match.

      To better view the mesh on the contact region, click the **Wireframe** button on the **Graphics** toolbar.

6. If desired, merge the nodes by dragging the contact matches into a **Node Merge Group**.

The mesh nodes are matched during the contact match operation, but they are not merged. If you want conformal mesh, you should merge the nodes.

For more information about merging nodes, see Node Merge (p. 467).

7. To convert the contact matches to geometry selections, select the contact regions, right-click, and select **Convert To** > **Geometry Selection**.

   Converting the contact matches is helpful if you want to delete the contact regions, but wish to retain the contact matches.

Automatically Applying Contact Matches

1. If necessary, insert a Contact Match Group folder (p. 443).

2. Select the **Contact Match Group** folder and, in the Details view, set the **Auto Detection** properties as needed.

3. Right-click the **Contact Match Group** folder, and then select **Detect Connections**.

4. In the Details view, verify the properties.

   Primary Geometry, Secondary Geometry

   "Primary" indicates the topology that will be captured after the operation is complete. In other words, it is the topology to which the nodes in the secondary topologies are matched. The primary geometry can be one or more faces.

   "Secondary" indicates the topology that will be matched to the primary during the operation. The secondary geometry can be one or more faces.

   Tolerance

   The **Tolerance** here has a similar meaning to the **Tolerance Value** global connection setting, and is represented as a transparent sphere.

   Setting the correct tolerance can be very important, and in some cases may require some speculation and experimentation. You can adjust the tolerance after the contact matches are generated by selecting the contact matches and then changing the **Tolerance Value**.

   For details on setting the tolerance, see How Tolerances Affect Contact Matches (p. 457)

5. Generate the contact matches by doing one of the following:

To...	Do this...
Generate all contact matches	Right-click the **Mesh Edit** folder and select **Generate**.
Generate contact matches for a contact match group	Right-click the **Contact Match Group** object and select **Generate**.

   If the base mesh is out-of-date, it is regenerated. The nodes are matched between the primary and secondary geometries, and a message appears displaying the number of node pairs that were matched.

The automatic contact match detection might detect contact matches that are undesirable. Therefore, if any contact matches fail, you should verify that the contact match is necessary before attempting to correct the error (p. 464).

If the normals between the primary and secondary faces are misaligned, some contact matches may not be generated.

6. If necessary, review the contact matches:

   a. Select one or more **Contact Match** or **Contact Match Group** objects, right-click, and select **Create Named Selections**.

      A named selection is created for each contact match you selected. If you selected a contact match group, a named selection is created for each contact match within the group. Each named selection is automatically given the same name as the contact match from which you created it.

   b. Click a named selection to view the mesh for the contact match.

      To better view the mesh on the contact region, click the **Wireframe** button on the **Graphics** toolbar.

7. If desired, merge the nodes by dragging the contact matches into a **Node Merge Group**.

   The mesh nodes are matched during the contact match operation, but they are not merged. If you want conformal mesh, you should merge the nodes.

   For more information about merging nodes, see Node Merge (p. 467).

Manually Applying Contact Matches

1. If necessary, insert a Contact Match Group folder (p. 443).

2. Do one of the following:

   - Right-click the **Contact Match Group** folder and select **Insert** > **Contact Match**.
   - Highlight the **Contact Match Group** folder, and then click the **Contact Match** option on the **Mesh Edit** toolbar.

3. In the Details view, verify the properties.

   Primary Geometry, Secondary Geometry

   "Primary" indicates the topology that will be captured after the operation is complete. In other words, it is the topology to which the nodes in the secondary topologies are matched. The primary geometry can be one or more faces.

   "Secondary" indicates the topology that will be matched to the primary during the operation. The secondary geometry can be one or more faces.

Tolerance

The **Tolerance** here has a similar meaning to the **Tolerance Value** global connection setting, and is represented as a transparent sphere.

Setting the correct tolerance can be very important, and in some cases may require some speculation and experimentation. You can adjust the tolerance after the contact matches are generated by selecting the contact matches and then changing the **Tolerance Value**.

For details on setting the tolerance, see How Tolerances Affect Contact Matches (p. 457)

4. Generate the contact matches by doing one of the following:

To...	Do this...
Generate all contact matches	Right-click the **Mesh Edit** folder and select **Generate**.
Generate contact matches for a contact match group	Right-click the **Contact Match Group** object and select **Generate**.

If the base mesh is out-of-date, it is regenerated. The nodes are matched between the primary and secondary geometries, and a message appears displaying the number of node pairs that were matched.

If the normals between the primary and secondary faces are misaligned, some contact matches may not be generated.

5. If necessary, review the contact matches:

   a. Select one or more **Contact Match** or **Contact Match Group** objects, right-click, and select **Create Named Selections**.

      A named selection is created for each contact match you selected. If you selected a contact match group, a named selection is created for each contact match within the group. Each named selection is automatically given the same name as the contact match from which you created it.

   b. Click a named selection to view the mesh for the contact match.

      To better view the mesh on the contact region, click the **Wireframe** button on the **Graphics** toolbar.

6. If desired, merge the nodes by dragging the contact matches into a **Node Merge Group**.

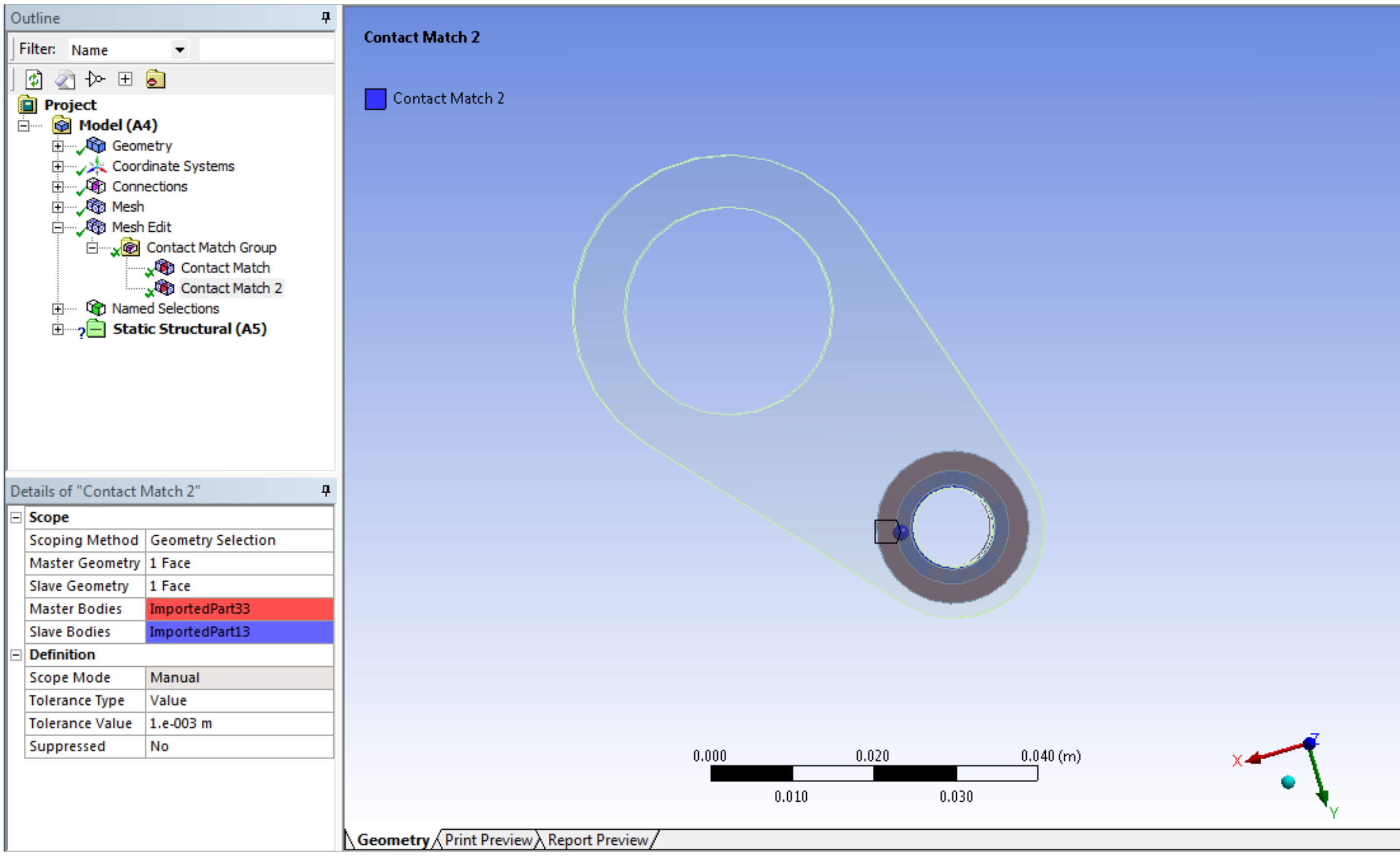
The mesh nodes are matched during the contact match operation, but they are not merged. If you want conformal mesh, you should merge the nodes.

For more information about merging nodes, see Node Merge (p. 467).

Displaying Multiple Views of Contact Matches

To more closely inspect a contact match, you can display the "Primary" and "Secondary" bodies in auxiliary windows next to the Geometry window.

Figure 209: Viewing the "Primary" and "Secondary" Bodies in Auxiliary Windows



1. Select a contact match.
2. On the **Mesh Edit** toolbar, click the **Body Views** button.

   The "Primary" and "Secondary" bodies are displayed in auxiliary windows.
3. For closer inspection of contact matches, click the **Show Mesh** button on the Graphics Options toolbar.
4. If desired, synchronize the views between the Geometry window and the auxiliary windows by clicking the **Sync Views** button on the **Mesh Edit** toolbar.

   By synchronizing the views, any change in the Geometry window will be reflected in the auxiliary windows.

Troubleshooting Failed Contact Matches

Diagnosing Failed Contact Matches

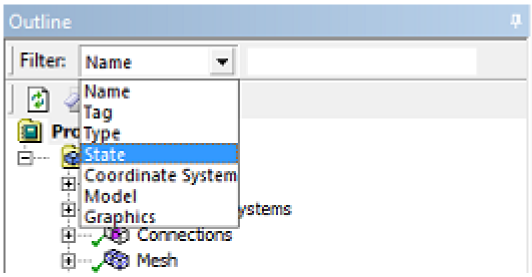
If a contact match fails, the Contact Match object may be in the "Ignored" state in the Tree Outline (), or a warning or error message might be displayed.

For a description of the various contact match states, see Understanding Mesh Connection and Contact Match States (p. 539).

To diagnose a failed contact match:

1. If necessary, use the **Filter** to identify any contact matches that are in the "Ignored" state.

   a. In the Outline, click the **Filter** drop-down menu and select **State**.

   b. In the adjacent drop-down menu, select **Ignored**.

2. If the contact match is in the "Ignored" state, then display the associated error message:

   a. In the Tree Outline, select the **Contact Match** object.

   b. In the Details view, click **Click to Display** to display the associated error message.

   You can use the **Filter** to identify contact matches that are in the "Ignored" state. However, if a contact match is in an error state, it cannot be filtered in the tree.

3. For each error message, take corrective action:

If the error message...	Do this...
Provides "Problematic Geometry" information	1. Right-click the error message and select **Show Problematic Geometry**. The geometry that is responsible for the message is highlighted in the Geometry window. Any error message that is related to a specific contact match will be associated with the secondary geometry. 2. Right-click the problematic bodies and select **Go To** > **Contact Matches for Selected Bodies**. All contact matches that are attached to the problematic geometry are highlighted. 3. Review the tolerances and mesh sizes associated with the highlighted contact matches.

If the error message...	Do this...
	For more information, see Correcting Contact Match Errors (p. 466).
Provides "Go to Body "information	1. Right-click the error message and select **Go to Body**. The object that is responsible for the message is highlighted in the Details view. 2. Review the tolerances and mesh sizes associated with the highlighted body or bodies. For more information, see Correcting Contact Match Errors (p. 466). This action highlights all contact matches attached to the problematic geometry.
Provides "Go to Object" information	1. Right-click the message and select **Go To Object**. The corresponding **Contact Match** object that is at issue becomes active in the tree. 2. Verify that all of the associated properties are properly defined.

Correcting Contact Match Errors

Contact match errors may occur due to tolerance issues or mesh size differences. Automatic contact match detection (p. 461) can result in undesirable contact matches. Therefore, if you used this method to apply the contact matches, you should verify that the failed contact match is necessary before correcting the error.

1. If there is a tolerance error, verify that the **Tolerance Value** is correct.

   If the distance between the "Primary" and "Secondary" bodies is greater than the **Tolerance Value**, then increase the tolerance.

2. Create a named selection for the failed contact matches that should have the same mesh size:

   a. Select all of the failed contact matches, right-click, and select **Create Named Selection**.

      A named selection is created for each failed contact match.

b. Select the named selections that should have similar mesh size, right-click, and select **Merge Selected Named Selections**.

   When choosing named selections, you should consider the location of the contact matches in the assembly, and the mesh size in the regions surrounding those contact matches.

3. Define the mesh size for the faces in the named selection:

   a. Right-click the **Mesh** object and select **Insert** > **Sizing**.

   b. In the Details view, set the **Scoping Method** to **Named Selection**, and then select the named selection that you created.

   c. Define the mesh sizing by modifying the settings in the **Definition** group as needed.

      For more information about the face sizing settings, see Descriptions of Local Sizing Control Options (p. 254).

4. Right-click the **Mesh** object and select **Generate**.

   The base mesh is regenerated with the face sizing control.

5. Right-click the **Mesh Edit** object and select **Generate**.

   The contact matches are regenerated.

Node Merge

Node Merge is a mesh editing tool that enables you to merge mesh nodes within a specified tolerance, making the mesh conformal across bodies, parts, and assemblies. Node merge can be used in conjunction with Node Move (p. 471) to remove large gaps in meshes without degrading mesh quality significantly. Node Merge can be performed on solid, sheet, and line bodies.

Scoping

Similar to Mesh Connections, Node merges are performed on mesh nodes. You must, however, explicitly specify the tolerance for node merges. Node merges can be face-to-face, face-to-edge, or edge-to-edge. Node Merge is a post-mesh operation, performed after the base mesh is generated. The base mesh is then stored so that if you change a node merge, only local re-meshing is required to clean up the neighboring mesh.

Requirements

For a Node merge to be successful, the mesh needs to be of similar size, and the number of mesh nodes to be merged on both topology entities need to be equal. If nodes on the primary and secondary do not have a 1–1 correspondence, then the resulting mesh after node merge can have holes.

Note:

Mesh Connections take priority over Node Merge operations. If nodes are connected by mesh connections, then the Node Merge operation will ignore the connection.

Application

Node merge operation supports two methods for connecting the mesh: *Automatic* and *Manual*. They both work similarly, with the exception that in manual node merge the node merge objects are explicitly populated in the tree, but are hidden in automatic mode. Automatic node merge is very useful when working on very large assemblies, as large number of node merge objects can slow down performance. Manual node merge, however, allows you to review individual node merge objects in detail.

A node merge operation is executed as a single operation on the base mesh of the whole model. If you have multiple node merge groups, executing **Generate** on a single group will merge only the nodes in that group, and will return the mesh to the state prior to the node merge operations on the other groups. To connect all of the node merge groups, execute **Generate** at the **Mesh Edit** folder level.

If you have applied any mesh connections or contact matches, you should generate them before applying node merges.

To automatically apply node merges:

1. Right-click on a **Model** object in the Tree and choose **Mesh Edit**, or select the **Model** object and choose **Mesh Edit** from the **Model** toolbar.

2. Right-click on a **Mesh Edit** object and choose **Insert**>**Node Merge Group**, or choose **Node Merge Group** from the **Mesh Edit** toolbar.

3. Select the **Node Merge Group** and set the **Method** to **Automatic Node Merge** in the Details view (this is the default setting).

4. Set the **Scoping Method** to **Geometry Selection** (the default) or to **Named Selection** and choose the body or bodies to search for node merges.

5. Modify the **Tolerance** settings in the Details view. The **Tolerance** value is used to find which bodies should be connected to which other bodies. For a discussion of Tolerance settings, see Tolerances Used in Mesh Connections (p. 448), although **Snap Boundaries** are not available for Node Merges.

    - **Tolerance Type**: Options include **Slider**, **Value**, and **Use Sheet Thickness**. Bodies in an assembly that were created in a CAD system may not have been placed precisely, resulting in small overlaps or gaps along the connections between bodies. You can account for any imprecision by specifying connection detection tolerance. This tolerance can be specified by value when the type is set to **Slider** and **Value**, or sheet thickness of surface bodies when the type is set to **Use Sheet Thickness**.

- **Tolerance Slider**: Appears if **Tolerance Type** is set to **Slider**. To tighten the connection detection, move the slider bar closer to +100 and to loosen the connection detection, move the slider bar closer to -100. A tighter tolerance means that the bodies have to be within a smaller region (of either gap or overlap) to be considered in connection; a looser tolerance will have the opposite effect. Be aware that as you adjust the tolerance, the number of connection pairs could increase or decrease.
- **Tolerance Value**: Appears if **Tolerance Type** is set to **Slider** or **Value**. This field will be read-only if the **Tolerance Type** is set to **Slider** showing the actual tolerance value based on the slider setting. When the **Tolerance Type** is set to **Value**, you will be able to provide an exact distance for the detection tolerance.

  After you provide a greater than zero value for the **Tolerance Value**, a circle appears around the current cursor location. The radius of the circle is a graphical indication of the current **Tolerance Value**. The circle moves with the cursor, and its radius will change when you change the **Tolerance Value** or the **Tolerance Slider**. The circle appropriately adjusts when the model is zoomed in or out.
- **Use Range**: Appears when the **Tolerance Type** property is set to **Slider** or **Value**. Options include **Yes** and **No** (default). If set to **Yes**, you will have the connection detection searches within a range from **Tolerance Value** to **Min Distance Value** inclusive.
- **Min Distance Percentage**: Appears if **Use Range** is set to **Yes**. This is the percentage of the Tolerance Value to determine the Min Distance Value. The default is 10 percent. You can move the slider to adjust the percentage between 1 and 100.
- **Min Distance Value**: Appears if **Use Range** is set to **Yes**. This is a read-only field that displays the value derived from: **Min Distance Value** = **Min Distance Percentage** * **Tolerance Value**/100.
- **Face/Face**: Options include **Yes** and **No** (default). Detects connection between the faces of different bodies. The maximum allowable difference in the normals for which contact is detected is 15 degrees.
- **Face/Edge**: Options include **Yes** and **No** (default). Detects connection between faces and edges of different bodies. Faces are designated as targets and edges are designated as contacts. Saying Yes exposes **Face Angle Tolerance** and **Edge Overlap Tolerance**.
- **Edge/Edge**: Options include **Yes** (default) and **No**. Detects connection between edges of different bodies.
- **Search Across**: This property enables automatic connection detection through the following options:
  - **Bodies** (default)
  - **Parts**: Between bodies of different parts, that is, not between bodies within the same multibody part.

  - **Anywhere**: Detects any connections regardless of where the geometry lies, including different parts. However, if the connections are within the *same body*, this option finds only Face/Face connections, even if the **Face/Edge** setting is turned **On**.

- **Face Angle Tolerance**: Available only if **Face/Edge** is set to **Yes**. For faces that will be excluded from the proximity detection pair, this property defines the minimum angle between the primary face and secondary edge entity, above which the two face pairs will be ignored from proximity detection. The default value is **`70°`**.

- **Edge Overlap Tolerance**: Available only when **Face/Edge** is set to **Yes**. This tolerance value is the minimum percentage that an edge may overlap the face and is included as a valid proximity detection pair. The default value is **25%**.

6. Generate the node merge by doing one of the following:

To...	Do this...
Generate all node merges	Right-click **Mesh Edit** and choose **Generate**.
Generate node merges for a Node Merge Group	a. For each **Node Merge Group** that should not be generated, right-click the **Node Merge Group** and choose **Suppress**. b. Right-click the **Node Merge Group** and choose **Generate**.

To manually apply Node Merges using the **Node Merge Group**:

1. Right-click on a **Model** object in the Tree and choose **Mesh Edit**, or select the **Model** object and choose **Mesh Edit** from the **Model** toolbar.

2. Right-click on a **Mesh Edit** object and choose **Insert**>**Node Merge Group**, or choose **Node Merge Group** from the **Mesh Edit** toolbar.

3. Select the **Node Merge Group** and set the **Method** to **Manual Node Merge** in the Details view.

   An additional control, **Group By** is exposed in the Details view. Options for **Group By** include **None**, **Bodies**, and **Parts**, and **Faces**. This property allows you to group the automatically generated connections objects. Setting **Group By** to **Bodies** (default) or to **Parts** means that connection faces and edges that lie on the same bodies or same parts will be included into a single connection object. The **Faces** option is only available if the **Face/Face** or **Face/Edge** controls are set to **Yes**.

   Setting **Group By** to **None** means that the grouping of geometries that lie on the same bodies or same parts will *not* occur. Any connection objects generated will have only one entity scoped to each side (that is, one face or one edge). If there are a large number of source/target faces in a single region. Choosing **None** avoids excessive contact search times in the solver.

4. Right-click the **Node Merge Group** and select **Detect Connections**. The states of the **Node Merge** connection are denoted using the same symbols as are used for **Mesh Connections** (p. 539).

5. Generate the node merge by doing one of the following:

To...	Do this...
Generate all node merges	Right-click **Mesh Edit** and choose **Generate**.

To...	Do this...
Generate node merges for a Node Merge Group	a. For each **Node Merge Group** that should not be generated, right-click the **Node Merge Group** and choose **Suppress**. b. Right-click the **Node Merge Group** and choose **Generate**.
Generate individual node merges	Select each node merge that you want to generate, right-click, and choose **Generate**.

To manually apply Node Merges individually:

1. Right-click on a **Model** object in the Tree and choose **Mesh Edit**, or select the **Model** object and choose **Mesh Edit** from the **Model** toolbar.
2. Right-click on a **Mesh Edit** object and choose **Insert**>**Node Merge**, or choose **Node Merge** from the **Mesh Edit** toolbar.

   The object is added as a child of a new **Node Merge Group**.

   Alternatively, you can select a **Node Merge Group** and set the **Method** to **Manual Node Merge** in the Details view. Then right-click on the **Node Merge Group** and choose **Insert**>**Node Merge**.
3. Set the **Scoping Method** to **Geometry Selection** (the default) or to **Named Selection** to determine how you will select the **Primary Geometry** and **Secondary Geometry**.
4. Select the **Node Merge** object and choose the **Primary** and **Secondary** Geometries:
   - "Primary" indicates the topology that will be captured after the operation is complete.
   - "Secondary" indicates the topology that will be pinched out during the operation.
5. Set the controls under **Definition** as desired.
6. Generate the node merge by doing one of the following:

To...	Do this...
Generate all node merges	Right-click **Mesh Edit** and choose **Generate**.
Generate the individual node merge	Right-click the node merge and choose **Generate**.

Note:

The mesher does not check the quality of the mesh after a Node Merge is performed. Therefore, you should perform a manual quality check any time you insert a Node Merge.

Node Move

The **Node Move** feature enables you to select and then manually move a specific node on the mesh to improve the local mesh quality.

Requirements

This feature has the following topological requirements:

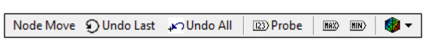
- Node movement is restricted to the target part. That is, nodes cannot be moved outside of the part.
- Face/Edge nodes can be moved along the corresponding face/edge only.
- The application locks corner nodes to the underlying vertex.
- Moving a node on a solved analysis causes the solution data to become obsolete. However, the state does not become obsolete for other objects that depend on the mesh (for example, imported loading conditions, Element Orientations). If your node movement changes affect other objects, you will need to manually clear and refresh these objects to apply the new changes.
- The **Node Move** feature is disabled if you are working with Section Planes.

Application

The **Node Move** object is a child object of **Mesh Edit**. It is inserted into the tree by selecting the **Node Move** button on the **Mesh Edit** toolbar or by selecting the **Mesh Edit** object, right-clicking, and selecting **Insert**>**Node Move**.

To use this feature, you need to generate the mesh on your model. This can be done before or after you have inserted the object into the tree. Once generated, the node selection options are assigned automatically. The Select Type is set to Select Mesh and Vertex is the required picking tool. Moving the cursor across the mesh of your model displays the available mesh nodes. You may then select and move nodes.

In addition, once the object is placed in the tree, the **Node Move** toolbar displays.



Node Move toolbar options include the following:

Undo Last

Cancels the last node movement performed on the mesh. Operations that change the original mesh may make this option unavailable.

Undo All

Cancels all of the node movements that you have made to the mesh. Operations that change the original mesh may make this option unavailable.

Probe, Max, and Min

These are annotation options. Selecting the **Max** and/or **Min** buttons displays the maximum and minimum values for mesh criteria (Element Quality, Jacobian Ratio, etc.) that you have selected. The **Probe** feature is also criteria-based. You place a **Probe** on a point on the model to display an annotation on that point. **Probe** annotations show the mesh criterion-based value at the location of the cursor. When created, probe annotations do not trigger the database to be marked for the file needing to be saved (that is, you will not be prompted to save). Be sure to issue a save if you wish to retain these newly created probe annotations in the database. These options are not visible if the **Mesh** object **Display Style** property is set to the default setting, **Body Color**.

Edges Options

This drop-down menu provides options to change the display of your model, including:

- **No Wireframe**: displays a basic picture of the body.
- **Show Elements**: displays element outlines.

These options are the same options that are available on the Meshing Context Toolbar.

Free Mode

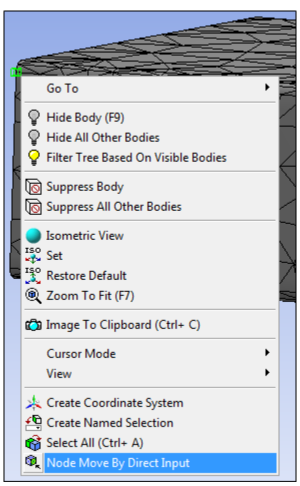
You can depress the **F4** key while you are moving a node to remove certain movement restrictions. This mode enables you to move an edge, vertex, or face node anywhere on a given face.

Note:

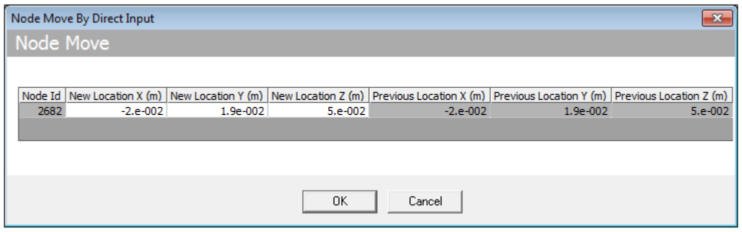
If moving nodes on an assembly mesh, only face nodes are available.

Direct Node Movement

You can also move a node by manually selecting the desired node, right-clicking, and then selecting the option **Node Move By Direct Input**.

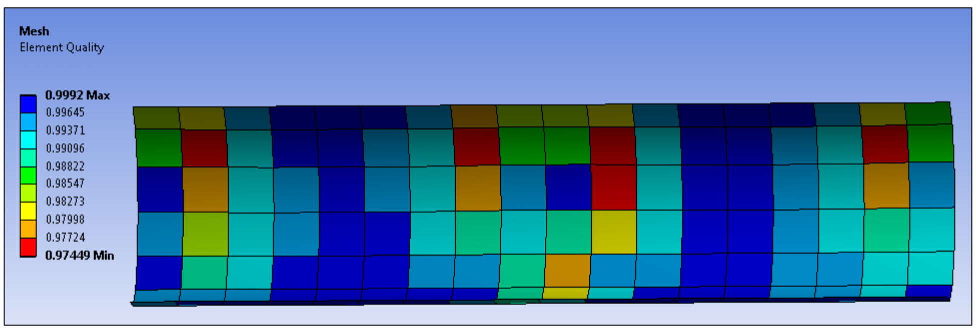


This option displays an entry window where you can manually change the X, Y, Z location of the node.



Element Quality Display

To enhance the presentation of your selections and movements, make sure that you set the **Display Style** property of the **Mesh** to **Element Quality**. An example of this setting is provided below. The exact display is provided on the **Node Move** object (and the legend title is also **Mesh**).

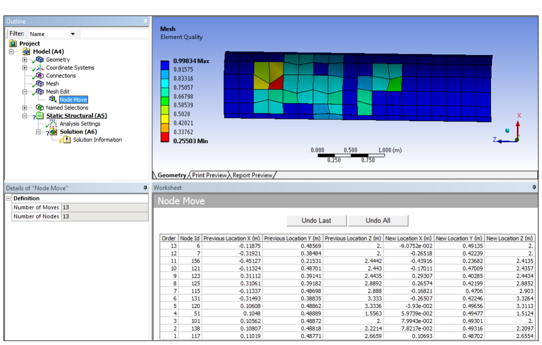


Worksheet

The **Worksheet** works in combination with the **Node Move** feature. The **Worksheet** records all of the node movements performed. In addition, and as illustrated below, the **Worksheet** provides information about the selection order of node movements, node numbers, coordinate-based location information (previous and new), as well as the options to undo the last movement or all of the movements that were made.

Note:

If you update your mesh (**Mesh** object>**Update**), the application maintains your movements in the **Worksheet** until the mesh is cleared (zero nodes) using the RMB option **Clear Generated Data** or you refresh data from the CAD source.



The illustration above also shows the **Worksheet** docked in the lower portion of the screen so that you can easily see all of your node movement information while also being able to see the model. Docking is possible with the docking tool shown here. This tool displays when you drag a window's title bar. Hovering the window over one of the blue arrows highlights the arrow. Releasing the mouse button docks the window in that screen location. It may also be useful to display the node numbers on your model using the display option **Node Number** available through Annotation Preferences.

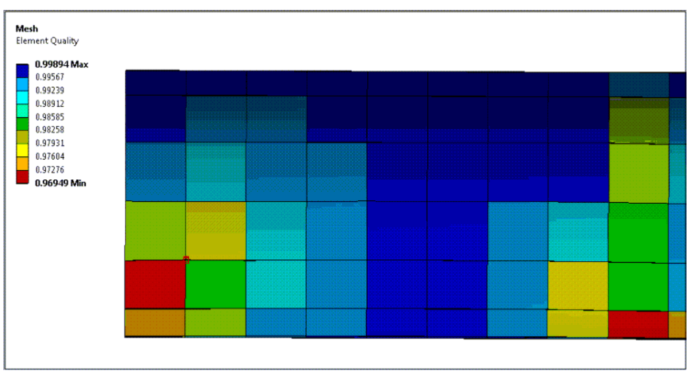
Note:

Node move operations are not persistent. The **Worksheet** view gives a history of what has been done based on the current Node IDs, but new Node IDs are created when the model is re-meshed. Therefore, the history recorded in the Worksheet is rendered out-of-date after re-meshing. For this reason, Node Move operations should be used sparingly, and only to fix small issues.

See the Windows Management Help section for additional information.

Example

The following is an animated example of the use of the feature in tandem with the **Element Quality** display. View online if you are reading the PDF version of the help. Interface names and other components shown in the demos may differ from those in the released product.



Pull

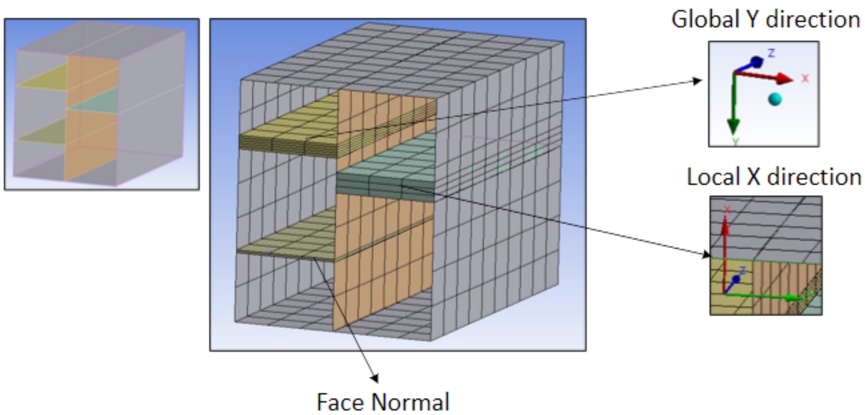
Pull method enables you to extrude or revolve element faces, geometric faces or scoped bodies (only shell body) from a surface or solid mesh. **Pull** also allows you to create surface coating on solid bodies, element faces and geometric faces belonging to the solid bodies. **Pull** generates volume of solid elements for the specified height or revolved volume from the source surface mesh along the specified coordinated system or the face normal vector.

Extrude

Extrude scopes shell bodies, elements belonging to shell bodies, geometric faces of shell or solid bodies, element faces of solid bodies.

To access **Extrude** ,

1. Right-click **Mesh object >Insert>Mesh Edit**.
2. Right-click **Mesh Edit >Insert>Pull**.
3. Click **Extrude** to perform extrude mesh.



To duplicate the **Pull** object, right-click the **Pull** object and click **Duplicate**. A new **Pull** object is created with the same parameters as the already created one. When you select multiple **Pull** objects, the **Duplicate** option is not available. **Pull** creates linear elements if the base mesh is having linear elements and quadratic elements if the base mesh is quadratic.

In the **Tree** view under **Mesh** object, right-click **Pull > Generate Selected Pull** to create **Pull Part** under the **Geometry** tree. The **Pull** part created is independent of the contents of the **Pull** mesh. When you **Extrude** or **Revolve** object from mesh data, the base mesh consists of solid element faces or surface elements. Each connected set of elements forms a body. Each body has a top, bottom and side face. Two edges are created bounding the top and bottom face. These edges are called ring edges and they do not have vertices.

When you **Extrude** or **Revolve** object from CAD topology, bodies of the **Pull** part match the bodies of the base mesh. Bottom faces of the pull bodies match those of base mesh and top face match the corresponding face of the base mesh. A side face is created for each edge of the CAD topology. Edges and vertices of **Pull** bodies are created to form watertight solids.

The exceptions to the above rule are the following:

- When the mesh body having multiple parts are meshed using **Batch Connections**, a **Pull** part is created for a **Pull** feature.

- When the **Pull** feature is Extrude up to target and the target is such that some of the base mesh must be removed to perform extrusion, you must apply the rules of extrusion of the mesh objects. Hence, the pull sides are not defined by the original CAD topology edges.

Note:

- For **Pull** defined by CAD topology, if the **Pull** mesh is cleared or CAD model is refreshed, all loads and boundary conditions behavior follow the one defined by the CAD topology entities.
- For **Pull** defined by mesh elements or element faces, if **Pull** mesh is cleared or CAD model is refreshed, all loads and boundary conditions need to be redefined.

The Details view displays the **Pull(Extrude)** options :

Scope

Scoping Method: Allows you to scope the model based on your selection. There are two options. They are **Geometry Selection** and **Named Selection**.

Geometry Selection: Allows you to scope elements, element faces, geometric faces and sheet bodies for **Extrude**.

Named Selection: Allows you to select the element facets and elements belonging to sheet bodies for **Extrude**.

Definition

Method: Displays the selected method of **Pull**.

Extrude: Allows you to generate solid elements for specified number of layers and height

Height: Allows you to specify the height for the volume of solid elements. **Height** allows only positive values.

Number of Layers:Allows you to specify the number of layers to be used for **Extrude**.

Extruded By: Allows you to specify the method of extrusion. The available options are **Use Coordinate System**, **Face Normal** and **Face Normal(Reversed).**

- **Use Coordinate System**: Allows you to generate elements in the specified coordinate system. When you select **Use Coordinate System** option in **Extruded By**, the **Coordinate System** and **Use Coordinate System** fields appear. The **Coordinate System** allows you to select the **Global Coordinate System** by default. The **Use Coordinate System** allows you to select any of the XYZ coordinate axis.
- **Face Normal**: Allows you to generate elements along the face normal vector.
- **Face Normal Reversed**: Allows you to generate elements along the opposite direction of the face normal vector.

Extrude Upto: Allows you to generate elements up to the specified target. The target can be faces from solids or sheet bodies and multiple faces with sharing geometric edge.

Suppressed: Allows you to suppress the selected entities. You can select **Yes** to suppress the selected entities and **No** to unsuppress the selected entities. The default value is **No**.

Part Properties

Material: Allows you to select the material of your choice. You can also select a different material from the parent body material. The available options are **None** and **Structural Steel**.

Revolve

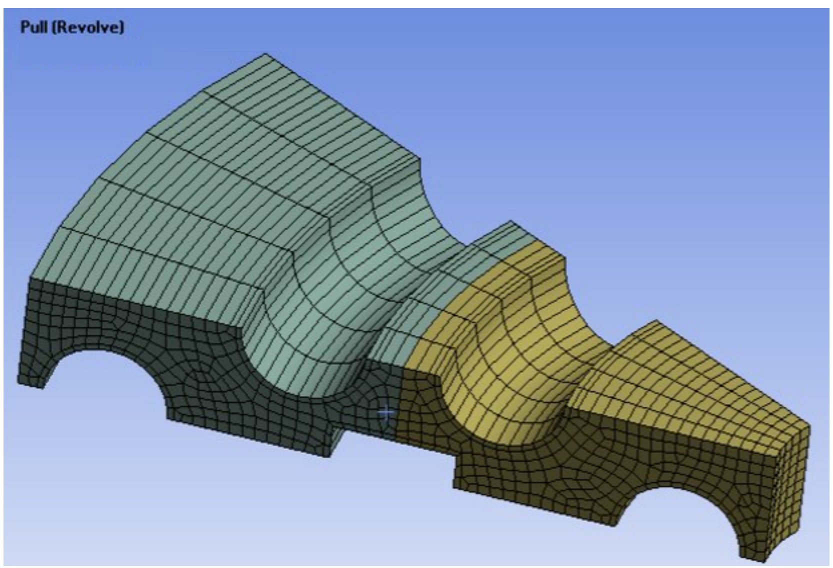
Revolve scopes shell bodies, elements belonging to shell bodies, geometric faces of shell or solid bodies, element faces of solid bodies.

To access **Revolve**,

1. Right-click **Mesh object >Insert>Mesh Edit**.

2. Right-click **Mesh Edit >Insert>Pull**.

3. Click **Revolve** to generate pull elements along the revolution angle.



The Details view displays the **Revolve** options:

Scope

Scoping Method: Allows you to scope the model based on your selection. There are two options. They are **Geometry Selection** and **Named Selection**.

Geometry Selection: Allows you to scope elements, element faces, geometric faces and sheet bodies for **Revolve**.

Named Selection: Allows you to select the element facets and elements belonging to sheet bodies for **Revolve**.

Definition

Method: Displays the selected method of **Pull**. **Revolve** allows you to generate solid elements for the specified number of layers and revolution angle along the edge coordinate system.

Revolution Angle: Allows you to specify angle of revolution for the solid elements. When the **Revolution Angle** is set to 360 degree, the **Merge Nodes** option is available. **Merge Nodes** allows you to merge the beginning and end nodes of the revolving body when set to **Yes**. The default value is **No**. **Revolution Angle** allows only positive values.

Number of Layers: Allows you to specify the number of layers to be used for **Revolve**.

Coordinate System: Allows you to select the **Global Coordinate System** by default. This option supports only **Cartesian System**.

Axis of Revolution: Allows you to select the coordinate axis along which the selected elements can be revolved. The available options are **X Axis**, **Y Axis**, **Z Axis**.

Suppressed: Allows you to suppress the selected entities. You can select **Yes** to suppress the selected entities and **No** to unsuppress the selected entities. The default value is **No**.

Part Properties

Material: Allows you to select the material of your choice. You can also select a different material from the parent body material. The available options are **None** and **Structural Steel**.

Surface Coating

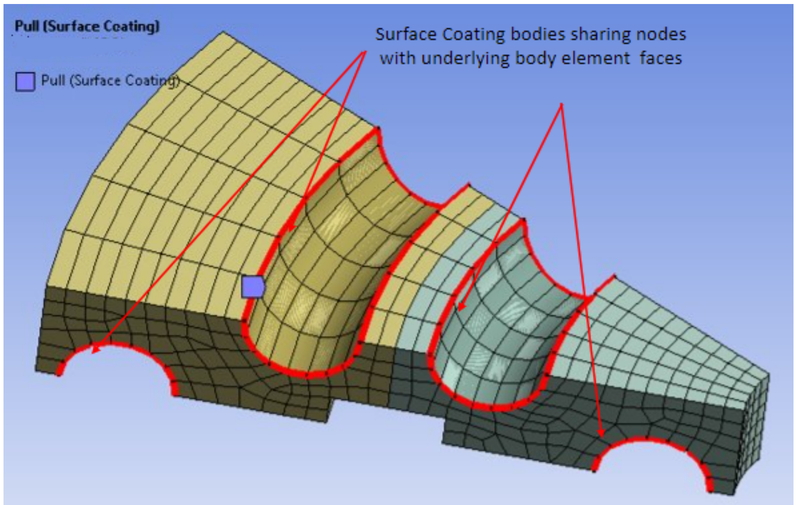
Surface Coating allows you to scope solid element faces. For mesh body, the bodies are created according to the bodies that own solid element faces. Each element face creates a topological face. Also, creates edges and vertices to form the topology. When you perform **Pull** on solid bodies with boundaries, bodies are created for each body that owns faces in the surface coating body and creates faces, edges, and vertices to match the topology of the base mesh.

To access **Surface Coating**,

1. Right-click **Mesh object >Insert>Mesh Edit**.

2. Right-click **Mesh Edit >Insert>Pull**.

3. Click **Surface Coating** to generate solid elements on the surface.



The Details view displays the **Surface Coating** options:

Scope

Scoping Method: Allows you to scope the model based on your selection. There are two options. They are **Geometry Selection** and **Named Selection**.

Geometry Selection: Allows you to scope elements, element faces of solid bodies for**Surface Coating**.

Named Selection: Allows you to scope element faces, geometric faces and solid bodies for **Surface Coating**.

Definition

Method: Displays the selected method of Pull. **Surface Coating** allows you to create shell elements coating for the outer layer of the 3D objects. You can scope solid element faces, CAD faces and solid bodies through **Geometry Selection** and **Named Selection**.

Suppressed: Allows you to suppress the selected entities. You can select **Yes** to suppress the selected entities and **No** to remove selection from the selected entities. The default value is **No**.

Part Properties

Material: Allows you to select the material of your choice. You can also select a different material from the parent body material. The available options are **None** and **Structural Steel**.

Stiffness Option: Allows you to provide the stiffness behavior of the shell body created by surface coating. The available options are **Stress Evaluation Only**, **Membrane Only**, **Membrane and Bending**.

Thickness: Allows you to specify the thickness for surface coating. The **Thickness** option is available only when the **Stiffness Behavior** is set to **Membrane Only** or **Membrane and Bending**.

Limitations

- When you perform **Pull** on intersecting bodies having mesh, **Pull** cannot detect the intersection between the existing mesh and the created **Pull** or intersection between the two **Pull** objects. **Pull** can only detect self-intersection within a single **Pull**. Then, it provides error message without creating **Pull** mesh.

- When you select two element faces, geometric faces or disconnected bodies from disjoint bodies in the same model for performing **Pull** operation, two **Pull** objects are created under the generated **Pull** object.
- When **Pull** is performed on a highly curved surface, the generated pulled elements may converge or intersect. Such elements are excluded from extrusion when **Extrude Upto** is set to **Yes**. When **Extrude Upto** is set to **No**, the **Pull** operation is aborted providing an error message.
- When a solid object is added through Construction Geometry after generating the **Pull** object, you cannot generate mesh on the solid body. You must insert and mesh all solid objects before generating the first **Pull** object.
- When you try to **Pull(Extrude)** using solid element faces, only single side face (perpendicular to extrude direction) is created instead of creating multiple faces as with the extruding topological entity.
- When you select multiple **Pull** objects for performing **Pull** operations, you do not have the option to generate pull. However, to generate multiple **Pull**, you can select the **Pull** objects right-click **Mesh Edit > Generate**.
- When you suppress a **Pull** part under the **Geometry** tree, the **Pull** object under the **Mesh Edit** folder is not suppressed. It remains unchanged.
- When you have a **Pull** object generated, the **Node Merge** option is not available when you try to insert it by right-clicking **Mesh Edit> Insert>Node Merge**.
- **Pull** does not support cylindrical coordinate system.
- **Pull (Surface Coating)** does not support rigid bodies.
- **Refresh** is not supported when you scope mesh elements or element faces for **Pull** generation.
- For geometries created using the **Mesh Pull** feature and exported as a **Part Manager Database** file, the Ansys Mechanical does not currently support re-meshing the part using normal mesh methods when you import the .pmdb file into a new **Mechanical** session.
- When you perform **Pull** on a model having both solid and shell bodies, **Pull** scopes element faces of both solid and shell bodies without providing an error message.
- When same elements are scoped through two different Name Selection, multiple **Pull** can be used to generate the for **Pull** the same elements without providing any warning message.
- When you apply **Pull** on a model with multiple body parts, it creates conformal mesh and shares topology between the bodies. Hence, if you do not want to share topology, you must transfer the model to Ansys Mechanical as multi-part assemblies.
- For **Extrude** and **Revolve**, **Pull** automatically suppresses the base body and hides it in the graphics when the:
  - Sheet body is scoped
  - All the geometric faces of a sheet or solid body is scoped
  - All elements of a sheet body are scoped.

- When you apply **Revolve** on a model having multibody parts and if the axis of revolution coincides with the multiple edges of the model, then **Pull** mesh fails.
- When a single **Pull** is applied on the multipart assembly, the **Pull** creates non conformal mesh and may have self-intersection causing **Pull** failure. You may define separate **Pull** for each part of the multipart assembly for successful pull generation.

Common Display Features

This section describes some common display features:

Hiding or Suppressing Bodies

For a quick way to hide bodies (that is, turn body viewing off) or suppress bodies (that is, turn body viewing off and remove the bodies from further treatment in the analysis), select the bodies in the tree or in the **Geometry** window (choose the **Body** select mode, either from the toolbar or by a right-click in the **Geometry** window). Then right-click and choose **Hide Body** or **Suppress Body** from the context menu. Choose **Show Body**, **Show All Bodies**, **Unsuppress Body**, or **Unsuppress All Bodies** to reverse the states.

The following options are also available:

- **Hide All Other Bodies**: show only selected bodies.
- **Hide or Show**: contains menu options to hide specific body types. Based on the body types of your model, options include: **Solid Bodies**, **Surface Bodies**, and **Line Bodies**.
- **Invert Visibility**: inverts the visibility of hidden bodies versus those that are visible. When selected, all hidden bodies become visible and all visible bodies become hidden.
- **Suppress All Other Bodies**: enables you to unsuppress only selected bodies.
- **Invert Suppressed Body Set**: enables you to reverse the suppression state of all bodies (unsuppressed bodies become suppressed and suppressed bodies become unsuppressed).

Note:

- If another model level object, such as a Remote Point, Joint, or Contact Region, is scoped to a Body that becomes Suppressed, that object also becomes suppressed until it is rescoped or the body is Unsuppressed.
- Results from hidden bodies are used in the formulation of the maximum and minimum values in the contour legend and in the Details View.

  Results from suppressed bodies are suppressed and are **not** used in the formulation of maximum and minimum values.

Hiding or Showing Faces

You can hide selected faces on a model such that you are able to see inside the model. This feature is especially useful for bodies with interior cavities, such as engine blocks. To use the feature, first select faces on the model that you want to hide, then right-click anywhere in the **Geometry** window and choose **Hide Face(s)** in the context menu. This menu choice is only available if you have already selected faces.

Choose **Show Hidden Face(s)** from the context menu to restore the visibility of faces previously hidden using **Hide Face(s)**. The **Show Hidden Face(s)** menu choice is only available if there are hidden faces from choosing **Hide Face(s)**. It *cannot* be used to restore the visibility of faces previously hidden by setting **Visible** to **No** in the Details view of a Named Selection object.

Note:

The selected faces will appear hidden only when you view the geometry. The feature is not applicable to mesh displays or result displays.

Creating Section Planes

For viewing purposes, you can use the **Create Section Plane** option to slice the graphical image of your model based on a predefined coordinate system. For details, refer to Creating Coordinate-Based Section Planes in the Mechanical help.

Meshing: Ease of Use Features

The features described in this section are intended to assist you in meshing.

Updating the Mesh Cell State

In contrast to the **Generate Mesh** (p. 486) feature (which only produces the mesh), the **Update** feature determines whether the geometry needs to be updated, refreshes the geometry if necessary, generates the mesh if necessary, and also writes the output data for any connected cells:

- The **Generate Mesh** feature is useful when you are investigating the impact of different settings on the mesh but you are not ready to export the mesh files.
- The **Update** feature is useful if you make a connection in the Ansys Workbench Project Schematic from a Mesh cell to a system that requires a new type of output data (for example, if you make a connection from the Mesh cell to a Fluid Flow (CFX) or Fluid Flow (Fluent) analysis system). In such cases, the Mesh cell will go out-of-date. To bring the Mesh cell up-to-date, you can perform an **Update** on the Mesh cell from within the Project Schematic, or follow the procedure below from within the Meshing application.
- If you connect a Mesh component system to an analysis system and the mesh file contains an assembly mesh, the mesh file must be an Ansys Fluent mesh file (*.msh) for it to be consumed

by the downstream system's solver. If the mesh file is of any other type, the **Update** fails and an error message is issued.

- CFX does not support Assembly Mesh.

To update the Mesh cell:

1. Select the **Mesh** object or any mesh control object.
2. Right-click to display the context menu.
3. Select **Update** in the menu.

Note:

As an alternative to steps 2 and 3, you can click the **Update** button on the **Mesh** toolbar.

Generating Mesh

The **Generate Mesh** operation uses all defined meshing controls as input to generate a mesh. **Generate Mesh** operates only on active objects, meaning that if bodies or controls are suppressed, they are ignored by the meshing operation. You can generate mesh on the entire (active) model, or selectively on (active) parts and/or bodies. This includes single body parts, multibody parts, individual bodies, or multiple selected bodies across different parts or within the same part.

Note:

- Selecting **Generate Mesh** generates a mesh based on the current mesh settings. It does not write the output data for any connected cells (downstream systems). **Generate Mesh** is useful when you are investigating the impact of different settings on the mesh but you are not ready to export the mesh files. Refer to Updating the Mesh Cell State (p. 485) for related information.

- Using selective meshing, you can selectively pick bodies and mesh them incrementally. After meshing a body, you can mesh the whole part or assembly or continue meshing individual bodies. Refer to Selective Meshing (p. 404) for additional information.

Monitoring the Meshing Process

The **Ansys Workbench Mesh Status** dialog box contains a **Highlight** check box that you can use to control whether the topology that is currently being processed by the mesher is highlighted in the **Geometry** window, which may help with troubleshooting (p. 536).

You can enable and disable the **Highlight** check box during the meshing process. Meshing performance should be similar regardless of whether topology highlighting is enabled, but it may be less distracting to disable it. If topology highlighting is enabled and you stop the meshing process, the highlighted topology is selected for you automatically.

This topology highlighting is not supported for the **Patch Independent Tetra** or **MultiZone** mesh methods, or when assembly meshing is being used. For information about how to set the default for topology highlighting, refer to Meshing Options on the Options Dialog Box (p. 317).

Suppressing and Unsuppressing Bodies in a Model

When there is a combination of suppressed and unsuppressed (active) bodies in a model, the Meshing application meshes only the active bodies. This is true regardless of mesh method. In addition, all influence of the suppressed bodies on neighboring bodies and their meshes is suppressed. For example, if a size control is applied to a suppressed body, the size control will not affect that body, nor will it influence neighboring bodies (in general, if a size control is assigned to a suppressed body, that control is also suppressed unless it is also attached to other active bodies). Refer to Selective Meshing (p. 404) for additional information.

To generate the mesh for all active bodies:

1. Select the **Mesh** object or any mesh control object.
2. Right-click to display the context menu, or choose the **Mesh** drop-down menu from the toolbar.
3. Select **Generate Mesh** in the menu.

   All active bodies are meshed. If the model includes multiple parts, they are meshed in parallel. The **Ansys Workbench Mesh Status** dialog box appears, displaying the meshing progress and highlighting each entity as it is meshed.

   After the mesh has been generated, it is displayed when you select the **Mesh** object or the **Show Mesh** display option.

4. If necessary, stop the meshing process:

   a. In the **Ansys Workbench Mesh Status** dialog box, click **Stop**.

      To see which parts have been meshed, expand the **Geometry** object in the Tree Outline. A green status icon (✓) indicates that the part has been meshed.

   b. To restart the meshing process, right-click the **Mesh** object or any mesh control object and select **Update**.

      The meshing process resumes and meshes only the parts that have not yet been meshed.

To generate the mesh for individual active bodies:

1. Select the bodies by doing one of the following:
   - In the Tree Outline, select one or more **Body** objects.
   - Select one or more bodies in the Geometry window.

2. Right-click to display the context menu.
3. Select **Generate Mesh** in the menu.

The bodies that you selected are meshed. If you selected multiple parts, they are meshed in parallel. The **Ansys Workbench Mesh Status** dialog box appears, displaying the meshing progress and highlighting each entity as it is meshed.

After the mesh has been generated, it is displayed when you select the **Mesh** object or the **Show Mesh** display option.

4. If necessary, stop the meshing process:

   a. In the **Ansys Workbench Mesh Status** dialog box, click **Stop**.

      To see which parts have been meshed, expand the **Geometry** object in the Tree Outline. A green status icon (✘) indicates that the part has been meshed.

   b. To restart the meshing process, right-click the **Mesh** object or any mesh control object and select **Update**.

      The meshing process resumes and meshes only the parts that have not yet been meshed.

To generate the mesh for individual active parts:

1. Select the parts by doing one of the following:

   - In the Tree Outline, select one or more **Part** objects.
   - In the Geometry window, select one or more parts.

2. Right-click to display the context menu.

3. Select **Generate Mesh** in the menu.

   The parts that you selected are meshed. If you selected multiple parts, they are meshed in parallel. The **Ansys Workbench Mesh Status** dialog box appears, displaying the meshing progress and highlighting each entity as it is meshed.

   After the mesh has been generated, it is displayed when you select the **Mesh** object or the **Show Mesh** display option.

4. If necessary, stop the meshing process:

   a. In the **Ansys Workbench Mesh Status** dialog box, click **Stop**.

      To see which parts have been meshed, expand the **Geometry** object in the Tree Outline. A green status icon (✘) indicates that the part has been meshed.

   b. To restart the meshing process, right-click the **Mesh** object or any mesh control object and select **Update**.

      The meshing process resumes and meshes only the parts that have not yet been meshed.

After successfully generating a mesh, you can view mesh statistics and mesh metric information that you can use to evaluate the mesh quality. For more information, see Statistics Group (p. 193) and Quality Group (p. 117).

To re-mesh:

1. Select the **Mesh** object.
2. Right-click to display the context menu and select **Clear Generated Data** in the menu.
3. Confirm that you want to clear the mesh by clicking the **Yes** button.
4. Right-click the **Mesh** object to display the context menu again and select **Generate Mesh** in the menu.

Note:

- The order of topological entities is not guaranteed during a CAD source refresh. In cases in which you mesh, refresh, and re-mesh, the mesher may not produce exactly the same mesh if the refresh caused the topological entities to be reordered. As a result of this reordering, the mesher meshes the entities in a different order as well, producing a slightly different result.
- When selected from the **Geometry** object in the Tree Outline, the **Generate Mesh** RMB menu option behaves slightly differently than when it is selected from the **Mesh** object in the Tree Outline. Refer to Selective Meshing (p. 404) for details.
- Refer to Meshing: Troubleshooting (p. 535) for tips and strategies for handling problems that may occur during meshing.

Previewing Surface Mesh

You can preview the surface mesh for all unsuppressed parts, individual unsuppressed parts, or individual unsuppressed bodies. This includes single body parts, multibody parts, individual bodies, or multiple selected bodies across different parts or within the same part. You can also export the previewed surface mesh file in Fluent format, as described in Exporting a Previewed Surface Mesh in Fluent Format (p. 491).

Note:

- This feature is not supported for the **Patch Independent** and **MultiZone** mesh method controls. It is also not supported for thin model sweeping (that is, use of the **Sweep** mesh method control with **Src/Trg Selection** set to **Manual Thin** or **Automatic Thin**) or for assembly meshing algorithms. (Refer to The Assembly Meshing Workflow (p. 372) for alternative approaches when using assembly meshing algorithms.)
- When previewing surface mesh on bodies that are being meshed with **Sweep**, not all sizing information is used in the calculation. Previewing surface mesh on such bodies is a crude check to get a general idea whether the surface mesh is appropriate, but the full mesh

may look different due to differences in sizing calculations when taking all meshing constraints into account.

- Refer to Selective Meshing (p. 404) for general information about selective meshing and limitations related to using the **Preview Surface Mesh** feature with selective meshing.

To preview the surface mesh for all unsuppressed parts:

1. Select the **Mesh** object or any mesh control object.
2. Right-click to display the context menu, or choose the **Mesh** drop-down menu from the toolbar.
3. Select **Preview> Surface Mesh** in the context menu or **Preview Surface Mesh** in the drop-down menu. The surface mesh is displayed for the model when you select the **Mesh** object.

To preview the surface mesh for individual unsuppressed parts - from the object tree:

1. Select the **Part** objects.
2. Right-click to display the context menu.
3. Select **Preview> Surface Mesh** in the menu. The surface mesh is displayed for the parts when you select the **Mesh** object.

To preview the surface mesh for individual unsuppressed parts - from the Geometry window:

1. Select the **Mesh** object.
2. Select the parts in the **Geometry** window.
3. Right-click to display the context menu.
4. Select **Parts> Preview Surface Mesh** in the menu. The surface mesh is displayed for the parts when you select the **Mesh** object in the tree.

To preview the surface mesh for individual unsuppressed bodies - from the object tree:

1. Select the **Body** objects.
2. Right-click to display the context menu.
3. Select **Preview> Surface Mesh** in the menu. The surface mesh is displayed for the bodies when you select the **Mesh** object.

To preview the surface mesh for individual unsuppressed bodies - from the Geometry window:

1. Select the **Mesh** object.
2. Select the bodies in the **Geometry** window.
3. Right-click to display the context menu.

4. Select **Preview Surface Mesh On Selected Bodies** in the menu. The surface mesh is displayed for the bodies when you select the **Mesh** object in the tree.

After successfully previewing the surface mesh, you can view statistics about it. These statistics include mesh metric information that you can use to evaluate the mesh quality. For more information, see Statistics Group (p. 193).

Exporting a Previewed Surface Mesh in Fluent Format

Follow the steps below to export a previewed surface mesh in Fluent format:

To export a previewed surface mesh in Fluent format:

1. Follow the procedure to preview a surface mesh (p. 489).
2. Select **File> Export** from the main menu to export the surface mesh.
3. In the **Save As** dialog box, choose a directory and specify a file name for the file. Then choose **Fluent Input Files** from the **Save as type** drop-down menu and click **Save**.

   A `.msh` file suitable for import into Fluent will be created in the requested directory.

Previewing Source and Target Mesh

This feature allows you to preview the source and target meshes for scoped bodies. You can preview the source and target mesh on individual bodies or multiple selected bodies across different parts or within the same part. This feature applies only to a **Method** (p. 196) control set to the **Sweep** option.

Note:

- This feature is not supported for thin model sweeping (that is, use of the **Sweep** mesh method control with **Src/Trg Selection** set to **Manual Thin** or **Automatic Thin**).
- Refer to Selective Meshing (p. 404) for general information about selective meshing and limitations related to using the **Preview Source and Target Mesh** feature with selective meshing.

To preview the source and target mesh:

1. Scope the body or bodies of interest.
2. Right-click on the **Mesh** object and insert a **Method** (p. 196) control.
3. In the Details View, set **Method** to the **Sweep** option.
4. Right-click on the **Sweep Method** option in the tree to display the context menu, or choose the **Mesh** drop-down menu from the toolbar.
5. Select **Preview> Source and Target Mesh** in the context menu or **Preview Source and Target Mesh** in the drop-down menu. The source and target meshes are displayed when you select the **Mesh** object.

Previewing Inflation

The **Preview Inflation** feature helps you identify possible problems with inflation before you generate the mesh. You can preview inflation on single body parts, multibody parts, individual bodies, or multiple selected bodies across different parts or within the same part. You can also export the previewed inflation mesh file in Fluent format, as described in Exporting a Previewed Inflation Mesh in Fluent Format (p. 493).

Remember the following information when using the **Preview Inflation** feature:

- This feature is not supported for the **Patch Independent Tetra** and **MultiZone** mesh method controls. It is also not supported for assembly (p. 367) meshing algorithms.
- Match controls (p. 280) are not enforced when previewing inflation.
- In certain cases, **Preview Inflation** may return an invalid (p. 535) mesh. **Generate Mesh** (p. 486) may return a valid mesh with inflation, a valid mesh without inflation (because inflation failed), or result in a mesh failure (because tet meshing failed). For these reasons if **Generate Mesh** fails, using either the **Preview Surface Mesh** (p. 489) or **Preview Inflation** feature to locate the worst quality element is also likely to locate the cause of the mesh failure.
- When previewing inflation on bodies that are being meshed with **Sweep**, not all sizing information is used in the calculation. Previewing inflation on such bodies is a crude check to get a general idea whether inflation will work and/or if there will be problems with inflation prior to generating the full mesh, but the full mesh may look different due to differences in sizing calculations when taking all meshing constraints into account.
- This feature applies only when the **Inflation Algorithm** control is set to Pre (p. 156).
- After successfully previewing inflation, you can view statistics about it. These statistics include mesh metric information that you can use to evaluate the mesh quality. For more information, see Statistics Group (p. 193).
- Refer to Selective Meshing (p. 404) for general information about selective meshing and limitations related to using the **Preview Inflation** feature with selective meshing.

To preview inflation:

1. Apply inflation to the desired boundaries.
2. Select an object in the Tree Outline (the **Mesh** object, a mesh **Method** object, or an **Inflation** object) and right-click to display the context menu.
3. Select **Preview> Inflation** in the menu. In response, Workbench generates the inflation layers only and displays them in the **Geometry** window. You may need to click the **Mesh** object in the Tree Outline before you can view the inflation layers.

Figure 210: Previewed Inflation Mesh (p. 493) shows a model of an auto manifold to which inflation was applied. The **Preview Inflation** feature was selected, and the inflation layers were generated and displayed in the **Geometry** window.

Figure 210: Previewed Inflation Mesh

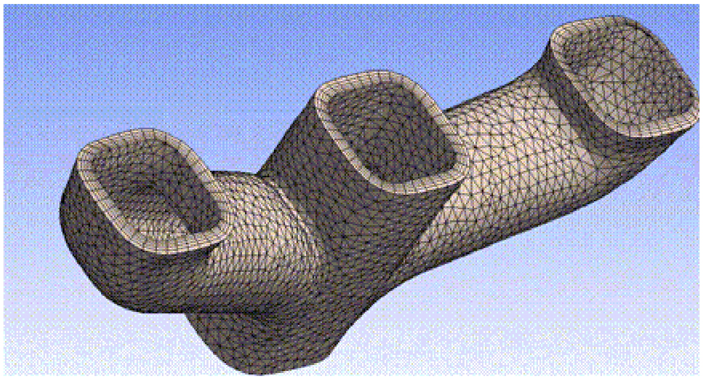
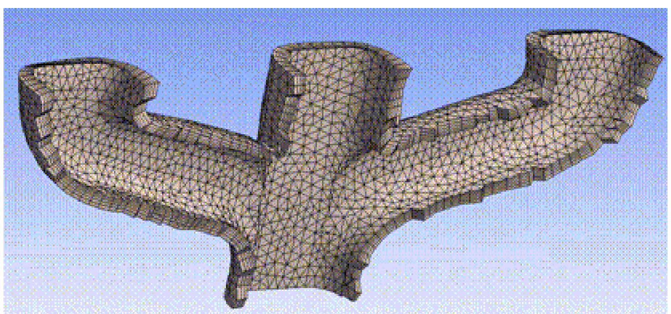


Figure 211: Section Plane View of Previewed Inflation Mesh (p. 493) shows a different view of the auto manifold model. For this view, a Section Plane was defined so that the inflation layers could be viewed internally.

Figure 211: Section Plane View of Previewed Inflation Mesh



Exporting a Previewed Inflation Mesh in Fluent Format

Follow the steps below to export a previewed inflation mesh in Fluent format:

To export a previewed inflation mesh in Fluent format:

1. Follow the procedure to preview inflation (p. 492).
2. Select **File> Export** from the main menu to export the inflation mesh.
3. In the **Save As** dialog box, choose a directory and specify a file name for the file. Then choose **Fluent Input Files** from the **Save as type** drop-down menu and click **Save**.

   A `.msh` file suitable for import into Fluent will be created in the requested directory.

Showing Program Controlled Inflation Surfaces

If you select **Program Controlled** as the value for the Use Automatic Inflation (p. 147) control, all surfaces in the model are selected to be inflation boundaries, except:

- Surfaces that are members of Named Selections groups
- Faces on non-enclosure bodies if an enclosure body is present

- Faces on bodies that have manual inflations on them
- Interfaces between bodies in a multibody part
- Faces used in contact
- Faces used in symmetry
- Faces on bodies being meshed with Hex Dominant or manual Sweep controls

To view the faces that have been selected for inflation:

1. Select the **Mesh** object in the Tree Outline.
2. Right-click to display the context menu.
3. Select **Show> Program Controlled Inflation Surfaces** in the context menu.

Note:

This feature is not supported for assembly (p. 367) meshing algorithms.

Showing Sweepable Bodies

You can display bodies that are "sweepable" according to the criteria detailed under Mesh Sweeping (p. 323).

To display sweepable bodies:

1. Select the **Mesh** object or any mesh control object.
2. Right-click to display the context menu.
3. Select **Show> Sweepable Bodies** in the menu. All sweepable bodies are displayed.

Note:

This feature is not supported for assembly (p. 367) meshing algorithms.

Showing Problematic Geometry

If problematic geometry causes meshing to fail, Workbench alerts you by:

- Displaying a **Problematic Geometry** message annotation in the **Geometry** window
- Displaying messages in the Messages window to describe the problem

For related information on viewing problematic geometry, refer to:

- Show Problematic Geometry in the DesignModeler help

- Meshing: Troubleshooting (p. 535)

Show Problematic Location

If problematic location causes meshing to fail, Workbench alerts you by:

- Displaying a **Show Problematic Location** message annotation in the **Geometry** window
- Displaying messages in the Messages window to describe the problem

Note:

When the problematic location is inside the body, you can use wire frame mode to visualize that particular location.

For related information on viewing problematic location, refer to:

- Show Problematic Geometry in the DesignModeler help
- Meshing: Troubleshooting (p. 535)

Showing Elements that Do Not Meet the Target Metric

If Check Mesh Quality (p. 118) is set to **Yes, Errors and Warnings**, and the generated mesh contains elements that do not meet the specified target metric, a warning message is displayed in the Messages window to indicate the problem.

You can create a Named Selection for these elements by right-clicking the Messages field and selecting **Show Elements** in the context menu. A Named Selection named **Warning Elements** will be created. You can optionally choose to add the attached elements as well.

Refer to Quality Group (p. 117) for related information.

Showing Removable Loops

You can use the **Show Removable Loops** feature prior to meshing to view loops that will be removed according to the criteria defined by the global **Sheet Loop Removal** (p. 193) and **Loop Removal Tolerance** (p. 193) controls.

This feature applies only to sheet models, and it responds only to the settings of the global loop removal controls. For example, if you set loop removal controls locally and your model contains loops that will be removed according to your local criteria, the **Show Removable Loops** feature will return nothing if the global **Sheet Loop Removal** control is off (set to **No**).

To view removable loops:

1. Select the **Mesh** object in the Tree Outline.
2. Right-click to display the context menu.

3. Select **Show> Removable Loops** in the context menu.

Note:

This feature is not supported for the **MultiZone Quad/Tri** (p. 246) mesh method or assembly (p. 367) meshing algorithms.

Inspecting Large Meshes Using Named Selections

You can use Named Selections to inspect only a portion of the total mesh. Although this feature is available regardless of mesh size, it is most beneficial when working with a large mesh (greater than 5 - 10 million nodes). For details, refer to Displaying Named Selections in the Mechanical help.

Generating Multiple Mesh Controls from a Template

The Object Generator enables you to make one or more copies of a template object, scoping each to a different piece of geometry. When defining mesh controls, you can use the Object Generator to make copies of a template mesh control, which may reduce the necessity to manually define multiple related mesh controls. For details, refer to Generating Multiple Objects from a Template Object in the Mechanical help.

Clearing Generated Data

You can clear generated data from the database using a right-mouse click menu item. You can either clear all mesh and results data (if applicable) from a model, or clear the mesh data on a selected part or body.

Note:

- When you clear the mesh, the status of the part or body will indicate that it is not meshed.
- When used on parts and bodies that have been joined by mesh connections (p. 444), the **Clear Generated Data** option works as follows, where the "base" mesh, which is stored in a temporary file, is the mesh in its unsewn (pre-joined) state:
  - If a base mesh is available, the mesh is reverted to the base mesh and the requested parts/bodies are cleared.
  - If no base mesh is available, the entire mesh is cleared and a warning message is issued. Reasons the base mesh may not be available include situations in which you have deleted your temporary files, exported a .mechdat file for someone else to use, or moved your project database to a different computer.
  - All mesh connections in the model, including those not associated with the selected body or part, are cleared.

– Because a Node Move cannot be undone, **Clear Generated Data** is not available from the **Mesh Edit** context menu when the **Mesh Edit** object has only **Node Move** objects as children.

To clear all mesh and results data from a model - from the object tree:

1. Select the **Mesh** object in the Tree Outline.
2. Right-click to display the context menu.
3. Select **Clear Generated Data** in the context menu.
4. When asked whether you want to clear the data, click **Yes**.

To clear the mesh data from the selected part or body - from the object tree:

1. Select the part or body in the Tree Outline.
2. Right-click to display the context menu.
3. Select **Clear Generated Data** in the context menu.
4. When asked whether you want to clear the mesh, click **Yes**.

To clear the mesh data from the selected body - from the Geometry window:

1. Select the **Mesh** object.
2. Select the bodies in the **Geometry** window.
3. Right-click to display the context menu.
4. Select **Clear Generated Data On Selected Bodies** in the context menu.
5. When asked whether you want to clear the mesh, click **Yes**.

To clear the mesh data from the selected part - from the Geometry window:

1. Select the **Mesh** object.
2. Select the part in the **Geometry** window.
3. Right-click to display the context menu.
4. Select **Parts> Clear Generated Data** in the context menu.
5. When asked whether you want to clear the mesh, click **Yes**.

Showing Missing Tessellations

Geometry with missing facets can lead to incorrect geometry representation by the mesher. Using the **Show Missing Tessellations** feature, you can highlight geometry with missing facets, which will allow you to detect and resolve problems prior to mesh generation.

Note:

This feature is available only for the **Patch Independent Tetra** (p. 200) mesh method and assembly (p. 367) meshing algorithms (both the **CutCell** and **Tetrahedrons** algorithms).

To highlight missing tessellations:

1. Select the **Geometry** object in the Tree Outline.

2. Right-click on the **Geometry** object or in the **Geometry** window to display the context menu.

3. Select **Show Missing Tessellations** in the context menu.

If a face without tessellations is found:

- The boundary of the face will be highlighted in the **Geometry** window.

- A warning message will be displayed in the **Messages** window advising you to use the **Show Problematic Geometry** (p. 494) context menu option to locate the problem areas.

Showing Mappable Faces

You can display faces that are "mappable" according to the criteria detailed under Face Meshing Control (p. 265).

To display mappable faces:

1. Select the **Mesh** object or any mesh control object.

2. Right-click to display the context menu.

3. Select **Show> Mappable Faces** in the menu. All mappable faces are highlighted in the **Geometry** window.

4. Right-click on the **Mesh** object or mesh control object and select **Insert> Face Meshing**. In the Details View, the **Geometry** field shows the number of faces that are mappable and therefore were selected.

By default, **Definition>Mapped Mesh** is set to **Yes**, which means the faces in the **Face Meshing** control are mapped by default.

To edit the selected set of faces, click the **Geometry** field in the Details View to activate it. Then in the **Geometry** window, select the mappable faces that you want to use in the **Face Meshing** control. Click

Apply in the Details View to complete your selection. Then proceed with the procedure described in Face Meshing Control (p. 265).

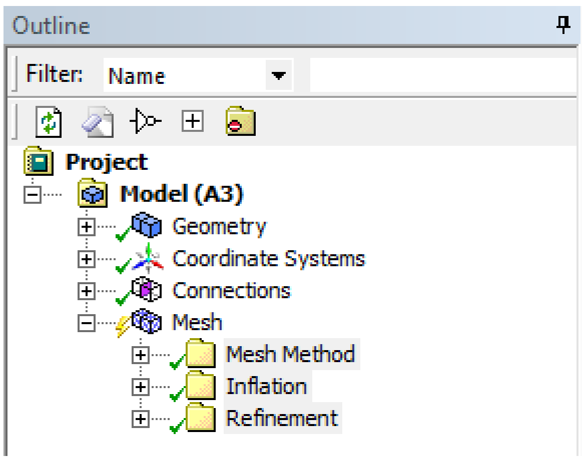
Note:

This feature is not supported for assembly (p. 367) meshing algorithms.

Grouping Mesh Objects By Type

If you have many different types of mesh objects, you can use the **Group All Similar Children** option to organize the objects into folders based on their type. In the following figure, the mesh methods, inflation control, and refinement have been grouped into separate folders.

Figure 212: Mesh Objects Grouped By Type



For more information, see Group Tree Objects.

Meshing: Virtual Topology

The following sections cover these topics:

Introduction

Uses of virtual topology include:

- Before performing analysis of a CAD model, you may want to group faces/edges together to form virtual cells. In such cases, virtual topology can aid you in reducing the number of elements in the model, simplifying small features out of the model, and simplifying load abstraction.
- You can split a face to create two virtual faces, or split an edge to create two virtual edges. For example, in the case of a rectangular face in which a single edge on one side of the face corresponds to two edges on the opposite side of the face, you can split the single edge so that node alignment across the face can have similar spacing.
- When needed, you can create virtual hard vertices to facilitate split face operations.
- Virtual topology can be helpful for handling fillets for **MultiZone** and sweep meshing. See Using Virtual Topology to Handle Fillets in MultiZone Problems (p. 363).

A CAD Model has two parts:

1. **Topology:** The connectivity of a CAD model, meaning: vertices are connected to edges, which are connected to faces, which are connected to volumes. Each one of these entities is referred to as a cell.
2. **Geometry:** The geometry of the CAD model is the underlying mathematical definition of the aforementioned cells.

A virtual cell in the Mechanical application or the Meshing application modifies the topology of only the local copy in the Mechanical application or Meshing application. Your original CAD model remains unchanged. New faceted geometry is also created with virtual topology. However, the mesher may project the nodes back to the original geometry where applicable.

You can use Virtual Topology to simplify the geometry to aid in meshing. There are several ways to simplify the topology using either automatic, manual, or a combination of automatic and manual approaches. The best approach to use is generally based on the meshing objectives (number of elements desired) and the cleanliness of the CAD model.

Virtual cells are often created to:

- Reduce the element count (increase the mesh size).
- Fix the topology of bodies that are not sweepable, to have mappable faces so that the bodies are now sweepable.
- Fix meshing problems.

To reduce the element count, it is often a good idea to first use **Automatic** or **Repair** operations to reduce the number of faces/edges, and then use **Manual** virtual topology operations to ensure important topology is respected and/or features that could create mesh quality problems are removed.

To fix the topology of bodies make them sweepable. **Automatic** and/or **Repair** operations might be helpful, but it is often very dependent on the geometry. **Manual** virtual topology operations give you more control and are often the better approach.

Meshing problems sometimes occur because of the topology. When a meshing failure occurs it generally points to the offending topology. Using **Manual** virtual topologies is a good way to fix such problems.

For more information about the Automatic and Manual approaches to creating virtual cells, see:

- Creating Virtual Cells Manually (p. 505)
- Creating Virtual Cells Automatically Using Automatic Mode (p. 508)
- Creating Virtual Cells Automatically Using Repair Mode(p. 512)

Note:

- There are geometric limitations to creating virtual cells, including those related to cells that would have too much curvature, or other limitations in trying to represent a group of faces by a single face.
- The tesselation of models from CATIA4 may not be appropriate for virtual topology, which could prevent the creation of virtual cells for these models.

Creating and Managing Virtual Cells

Notes on Virtual Cell Creation

- Scoped objects except other virtual cells may need to be relinked to the new virtual cells when that virtual cell is composed of entities in the scoped object. When the virtual cell is deleted the object may need to be rescoped to the original entities.

- All scoped objects except for mesh controls and other virtual cells will be protected during automatic virtual cell generation. This will allow users to load their models and run auto virtual topology without deleting loads. All faces within a protected object may be merged with faces in the same protected object and not in any other protected object.
- If **Generate on Update** is set to **Yes** and you update the geometry, all **Virtual Cell** objects that were created automatically will be deleted and recreated based on the new geometry. Any loads that were attached to geometry within the deleted **Virtual Cell** objects will need to be reattached to the new geometry.
- If **Generate on Update** is set to **No** and you update the geometry, all **Virtual Cell** objects that were created automatically should remain persistent barring major topology changes of the model being updated. Reapplication of loads may not be necessary.
- For any virtual cells that were generated, you can choose whether the nodes will be projected back to the underlying geometry (in contrast to the virtual cell's faceted geometry). To do so, select the desired virtual cell and use the **Virtual Topology Properties** dialog (p. 529) to set the **Project to Underlying Geometry** option to **Yes** or **No**. The default is **Yes** for analytical geometry; otherwise, the default is **No**. Computational expense increases when the option is set to **Yes**. Use these guidelines:
  - **Yes** is recommended if:
    - → You are using virtual topology to fine tune the topology of your mesh and need precise control from the Size Function (p. 100).
    - → Your geometry is valid and you want the mesh to capture it accurately. In this case there may be a slight impact on performance.
  - **No** is recommended if:
    - → You are using virtual topology to simplify bad/corrupt geometry or topology.
    - → You are trying to grossly defeature a model (for example, remove bosses, serial numbers, etc.).

Virtual Cell Logic and Usage

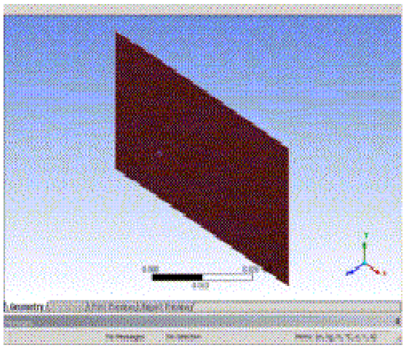
The Virtual Topology feature is a cell dependency operation where existing virtual cells can be used to create new virtual cells both manually and automatically. Within the cell hierarchy, a virtual cell depends on another virtual cell if the latter is used to create the former.

Virtual Face Dependency

A virtual face depends on a virtual edge if during the face creation the virtual edge is used to create virtual face loops. A virtual face depends on another virtual face if the latter was used to create the former.

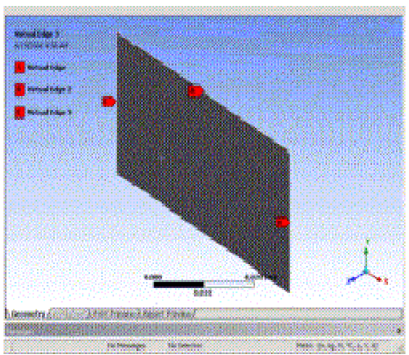
The following image represents the Merge Face Edges in the off position.

Figure 213: Merge Face Edges Off



The following image represents the Merge Face Edges in the on position.

Figure 214: Merge Face Edges On



Virtual Edge Dependency

A virtual edge depends on a virtual face if during the edge creation an owner loop belongs to the said face. A virtual edge depends on another virtual edge if the latter was used to create the former.

Note:

- If a virtual edge was created from a virtual split edge, you cannot delete the virtual split edge without first deleting the virtual edge. Conversely, if a virtual split edge was created from a virtual edge, you cannot delete the virtual edge without first deleting the virtual split edge.

  A warning message appears in the **Messages** window for each failed deletion. To highlight the geometry that is responsible for a message, select the message, right-click, and select **Show Problematic Geometry** from the context menu.

- In addition, if the virtual edge belongs to a virtual face, the virtual face will not be deleted either.

  A warning message appears in the **Messages** window for the virtual face, and you can use the **Show Problematic Geometry** option to highlight the face.

You can manually designate faces and edges for inclusion into a virtual cell, or you can have the Mechanical application or the Meshing application automatically create virtual cells based on settings that you specify. You can use the **Automatic** mode to globally reduce the number of faces and edges where possible, or use **Repair** to focus more closely on problematic faces and edges. The geometry under a virtual cell is represented by the underlying cell's graphic resolution.

Note:

- There are geometric limitations to creating virtual cells, including those related to cells that would have too much curvature, or other limitations in trying to represent a group of cells by a single cell.
- The tessellation of models from CATIA4 may not be appropriate for virtual topology, which could prevent the creation of virtual cells for these models.

Creating Virtual Cells Manually

1. Insert a **Virtual Topology** object in the tree.
2. Choose the face (**Ctrl**+**F**) or edge (**Ctrl**+**E**) selection filter, and then pick one or more faces or one or more edges that you want to include in the virtual cell(s).

   You can use the **Close Vertices** button to identify tightly clustered vertices that might need to be merged.
3. Create the **Virtual Cell** object(s) by doing one of the following:
   - Choose **Merge Cells** on the **Virtual Topology** context toolbar. You can also use Ctrl+M to merge either the faces after selecting them, or the common edge between faces.
   - Click the right mouse button on the Virtual Topology object and select **Insert> Virtual Cell** from the context menu.
   - Click the right mouse button in the **Geometry** window and select **Insert> Virtual Cell** from the context menu.

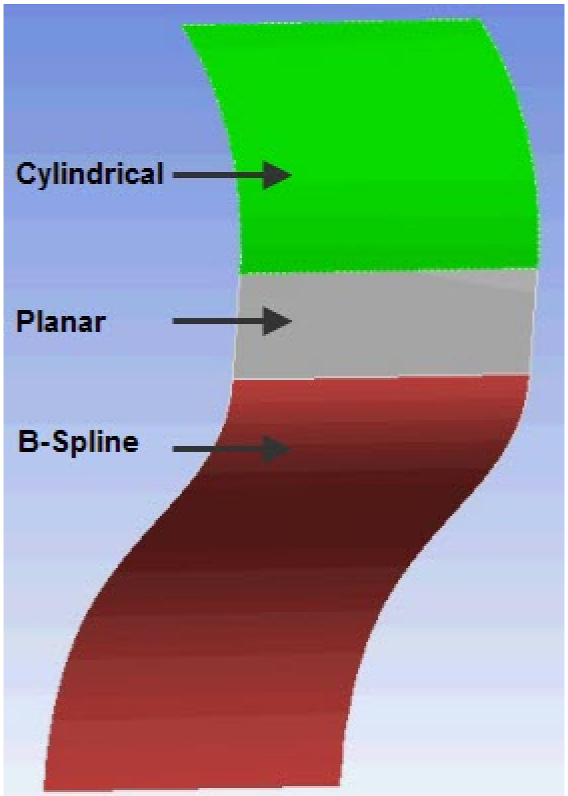
   From the selected set of faces or edges, the software creates the virtual cell(s). During this process, adjacent selected entities are grouped appropriately to form virtual cell(s), while any single selected entity (that is, one that is selected but is not adjacent to any other selected entity) forms its own virtual cell. An error message appears in the **Messages** window for each subset of failed topologies. To highlight the geometry that is responsible for a message, select the message, right-click, and select **Show Problematic Geometry** from the context menu. Refer to the examples below.

Note:

- A virtual cell cannot be created on a single edge that is straight or enclosed with no vertices.
- A virtual cell cannot be created on a single face that is cylindrical or planar. For example, if you select either the top face or middle face in the figure below and try to create a vir-

tual cell, no virtual cell will be created. However, selecting the bottom face will result in creation of a virtual cell.

Figure 215: Single Face Virtual Cell Limitations



Examples of Virtual Cell Formation

In the example shown in Figure 216: Formation of Virtual Faces (p. 507), suppose that you select faces A, B, E, and F and then select **Insert> Virtual Cell**. As a result, virtual faces **AB** and **EF** will be formed, as shown in Figure 217: Virtual Faces After Operation (p. 507).

The edges would be handled according to the **Merge Face Edges** setting. For example, if **Merge Face Edges** is set to **Yes**, virtual edges **ac** and **km** also will be formed.

As another example, using the same starting point, if you select faces A, E, and F and then select **Insert> Virtual Cell**, virtual faces **A** and **EF** will be formed. In this case, if **Merge Face Edges** is set to **Yes**, virtual edge **km** also will be formed.

Figure 216: Formation of Virtual Faces

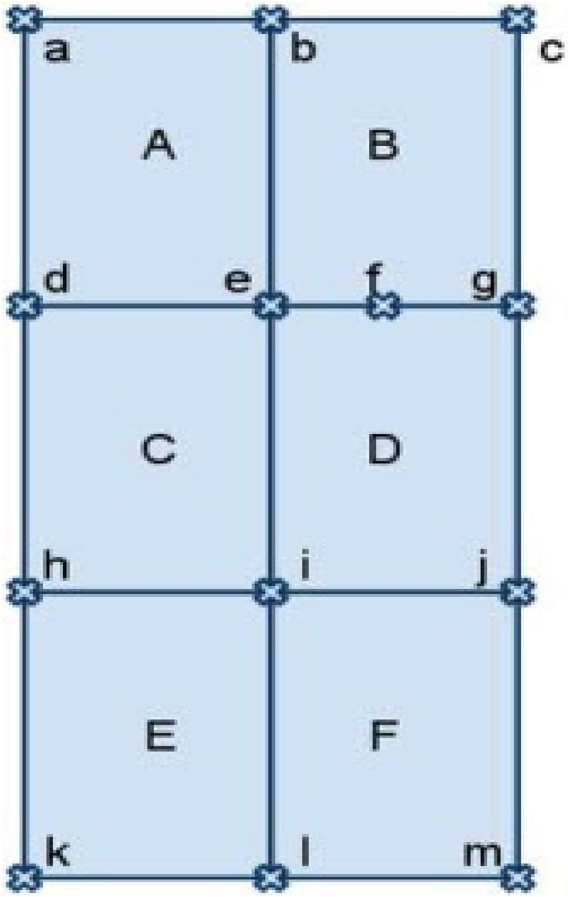
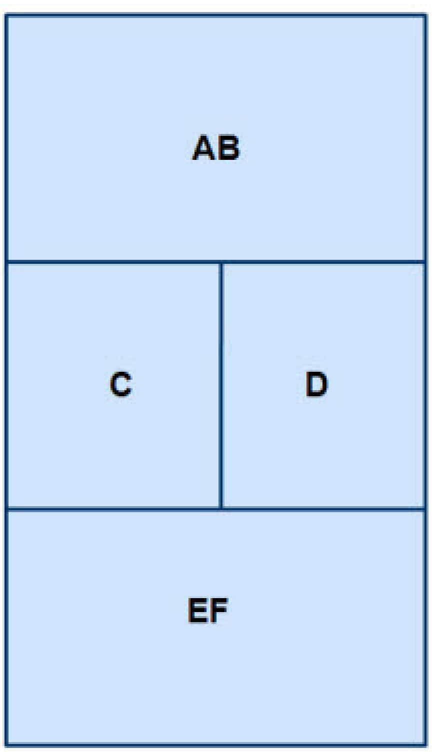
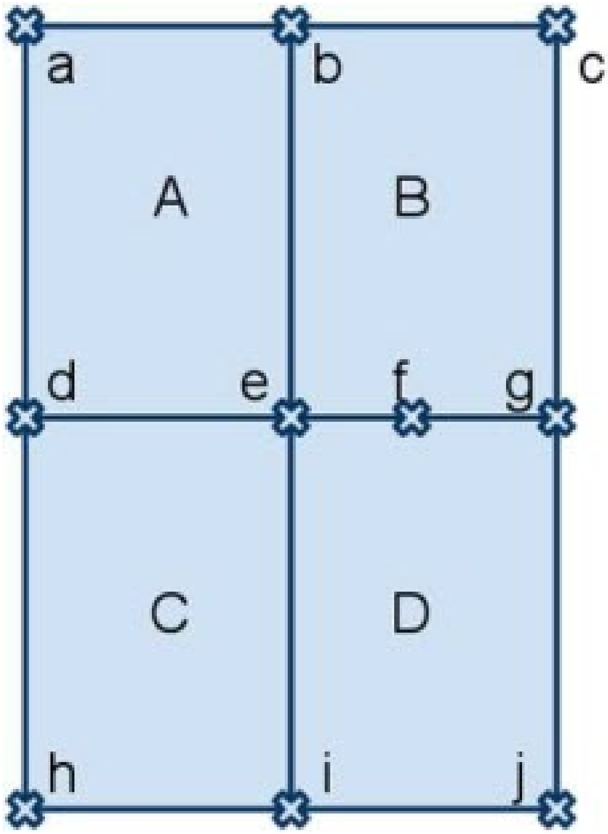


Figure 217: Virtual Faces After Operation



In the example shown in Figure 218: Formation of Virtual Edges (p. 508), suppose that you select all edges and then select **Insert> Virtual Cell**. In this case, only virtual edge **eg** will be formed because only those virtual edges that can be formed without forcing face merges will be created.

Figure 218: Formation of Virtual Edges



Creating Virtual Cells Automatically Using Automatic Mode

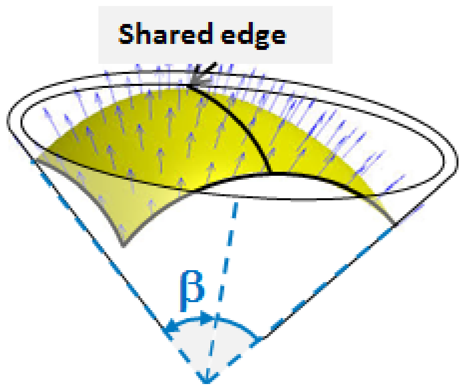
1. Insert a **Virtual Topology** object in the tree.
2. In the Details View, ensure that **Method** is set to **Automatic**.
3. Make adjustments as needed to any of the following settings in the Details View:
   - **Behavior** – Determines how aggressively the face(s) and edge(s) are merged. The choices are **Low**, **Medium**, **High**, **Edges Only**, and **Custom**. The **Edges Only** setting will merge only edges. The **Custom** setting exposes **Custom** properties (**Curvature** and **Feature Angles**) and **Advanced Custom** properties (**Aspect Ratio**, **Contact Angle**, **Edge Angle**, and **Shared Boundary Ratio**). These properties enable you to set parameters that control the creation of automatic Virtual Topologies.

     Note:

     Setting any of the Custom or Advanced Custom Properties to **-1** resets the value of that property back to its default.

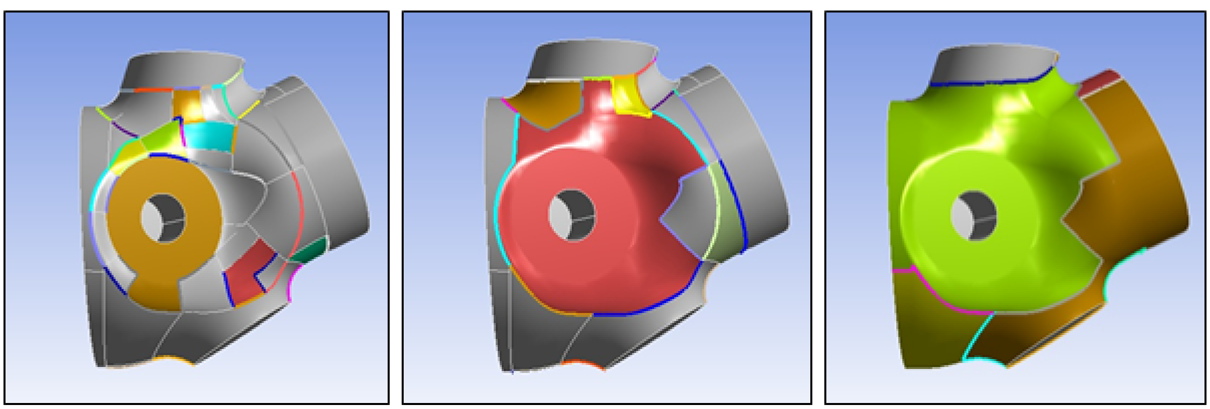
- **Gauss Curvature Angle** – Represents the flatness of the resultant face. If Angle β is greater than the Gauss Curvature Angle, the faces will stay separate. Increasing the Curvature Angle causes a greater number of faces to be grouped into fewer, larger faces, which might mean that there are fewer resulting Virtual Topologies.

  Figure 219: Gauss Curvature Angle

  The **Gauss Curvature Angle** can range from 0-180 degrees. The default setting is 60 degrees. The following figure shows the results of setting the **Gauss Curvature Angle** to 25, 60, and 120 degrees respectively.

  Figure 220: Curvature Angle at 25, 60, and 120 degrees

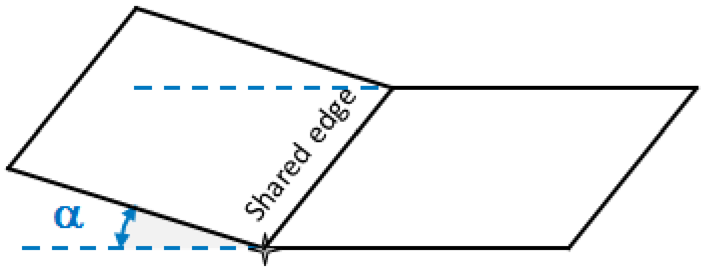
  

  Note:

  For best results, set the angle between 20 and 120 degrees. If the angle is set below this range, few faces will be merged. Setting the value above this range could result in problems with the mesh. Setting the value above this range could also result in large clusters of faces that ultimately fail to merge into a virtual face.

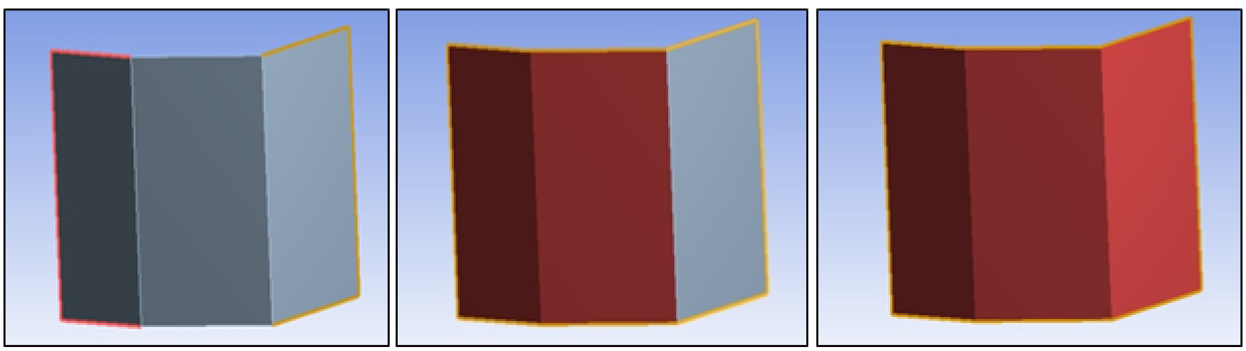
- **Feature Angle** – Sets the minimum angle between faces at a common edge. If Angle α, as shown in the following figure, is greater than the **Feature Angle**, the faces will stay separate. Faces are merged as the **Feature Angle** increases.

Figure 221: Feature Angle



The **Feature Angle** can range from 0 to 180 degrees. The default setting is 40 degrees. The following figure shows the results of setting the **Feature Angle** to 20, 40, and 80 degrees respectively.

Figure 222: Feature Angle at 20, 40, and 80 Degrees

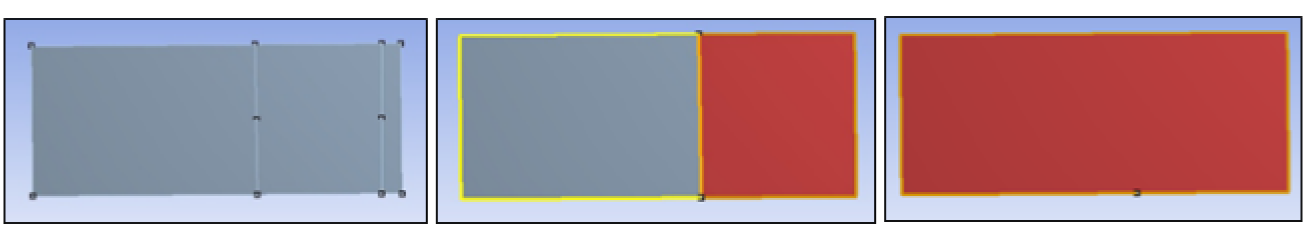


Note:

For best results, set the angle between 30 and 90 degrees. If the angle is set below this range, few faces will be merged. If the value is set above this range, it could result in large clusters of faces that ultimately fail to merge into a virtual face.

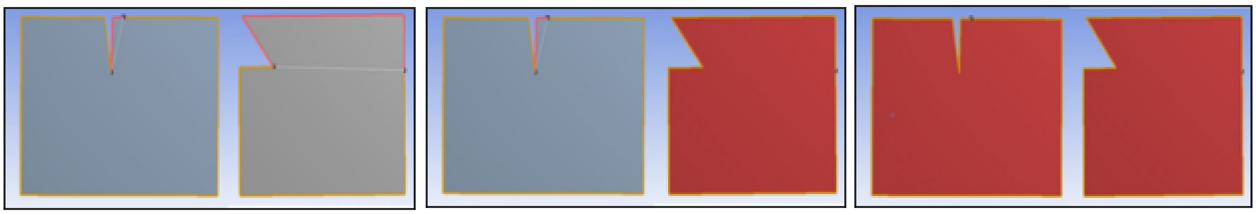
- **Aspect Ratio** – The ratio between the area of the face group to the square of the shared boundary length between the face and the Face Group. This setting controls how faces are grouped; increasing the Aspect Ratio causes a greater number of faces to be grouped into fewer, larger faces. The Aspect Ratio can range from 0 to 1. The default setting is 1. The following figure shows the results of setting the Aspect Ratio to 0.2, 0.5, and 0.9, respectively.

Figure 223: Aspect Ratio at 0.2, 0.5, and 0.9



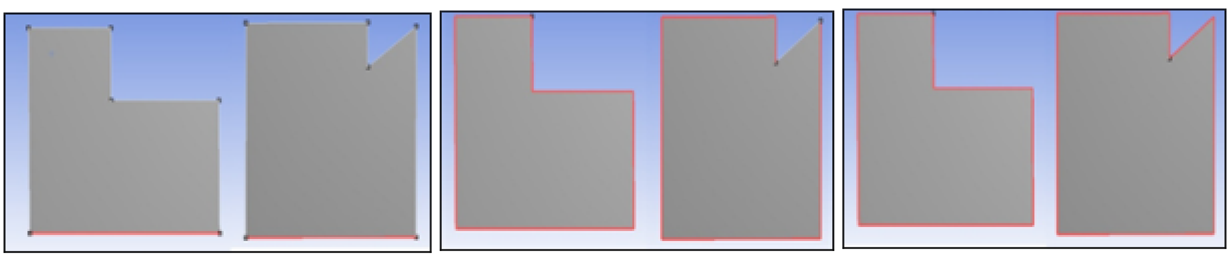
- **Contact Angle** – The angle introduced at common vertices between faces. This setting prevents complicated boundaries when grouping faces (for example, angles introduced at contact points between faces). Increasing the Contact Angle causes a greater number of faces to be grouped into fewer, larger faces. The angle can range from 0 to 360 degrees. The default setting is 360 degrees. The following figure shows the results of setting the **Contact Angle** to 270 degrees, 330 degrees, and 355 degrees, respectively.

Figure 224: Contact Angle at 270, 330, and 355 degrees



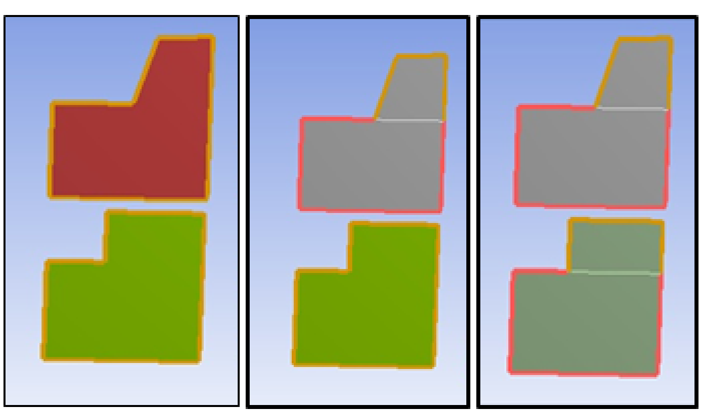
- **Edge Angle** – The feature angle between edges at their common vertex. Increasing the Edge Angle causes a greater number of edges to be grouped together. The angle can range from 0 to 180 degrees. The default setting for the Edge Angle is 80 degrees. The following figure shows the results of setting the **Edge Angle** to 80 degrees, 100 degrees, and 150 degrees, respectively.

Figure 225: Edge Angle at 80, 100, and 150 degrees



- **Shared Boundary Ratio** – The ratio of the length of the common boundary to the length of the smallest perimeter. Increasing the Shared Boundary Ratio causes the number of grouped faces decreases. The ratio can range from 0 to 0.5. The default setting for the ratio is 0. The following figure shows the results of setting the **Shared Boundary Ratio** to 0.2, 0.3, and 0.4, respectively.

Figure 226: Shared Boundary Ratio at 0.2, 0.3, and 0.4



4. Make adjustments to the **Advanced** (p. 528) settings in the Details view.

5. Do one of the following:

   - Click right mouse button on the **Virtual Topology** object and choose **Generate Virtual Cells** from the context menu. Virtual cells are automatically created for each region that meets the criteria established by the settings in step 2.

     These virtual cells remain valid after a geometry update.

   - Choose the face (**Ctrl**+**F**) or body selection filter and pick two or more faces or two or more bodies. Click right mouse button on the **Virtual Topology** object and choose **Generate Virtual Cells on Selected Entities** from the context menu. From the selected set of faces or bodies, the software groups adjacent entities appropriately and automatically creates virtual cells for each region that meets the criteria established by the settings in step 3.

     These virtual cells become invalid after a geometry update.

Creating Virtual Cells Automatically Using Repair Mode

1. Insert a **Virtual Topology** object in the tree.

2. In the Details View change the **Method** to **Repair**.

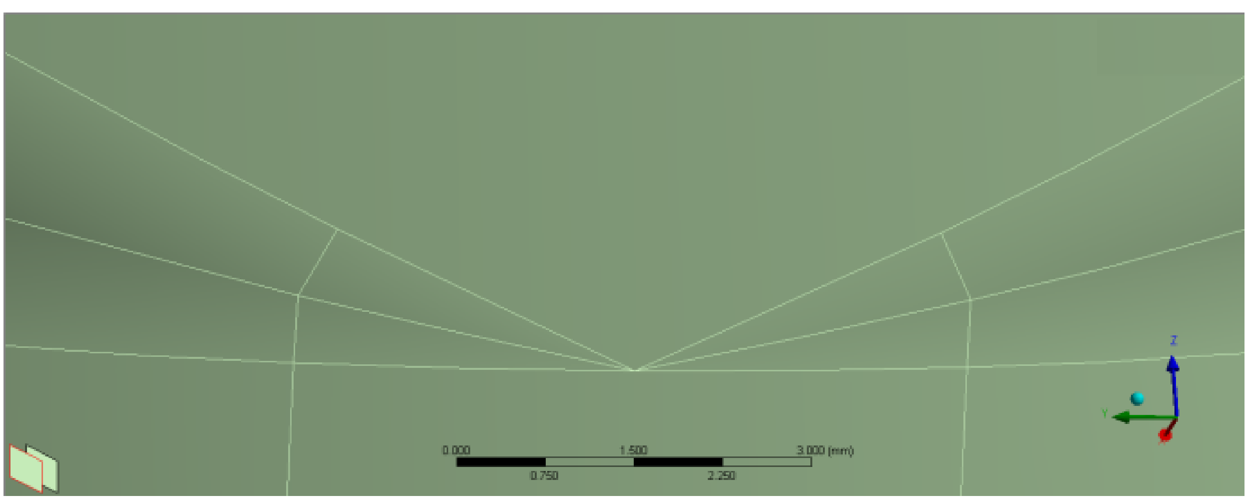
3. Make adjustments as needed to any of the following settings in the Details View:

   - **Behavior** – Determines the type of repair to be performed. The choices are **Repair All**, **Repair Small Edges**, **Repair Slivers**, and **Repair Small Faces**.

     – The **Repair Small Edges** setting tries to remove all the small edges, satisfying the **Repair Settings** criteria displayed in the Details View. The small edges are removed either by merging their attached faces or by merging the small edge with an adjacent edge. The **Repair Small Edges** setting exposes the **Max Edge Length** and **Min Edge Length** settings. The edges with lengths between **Max Edge Length** and **Min Edge Length** are repaired.

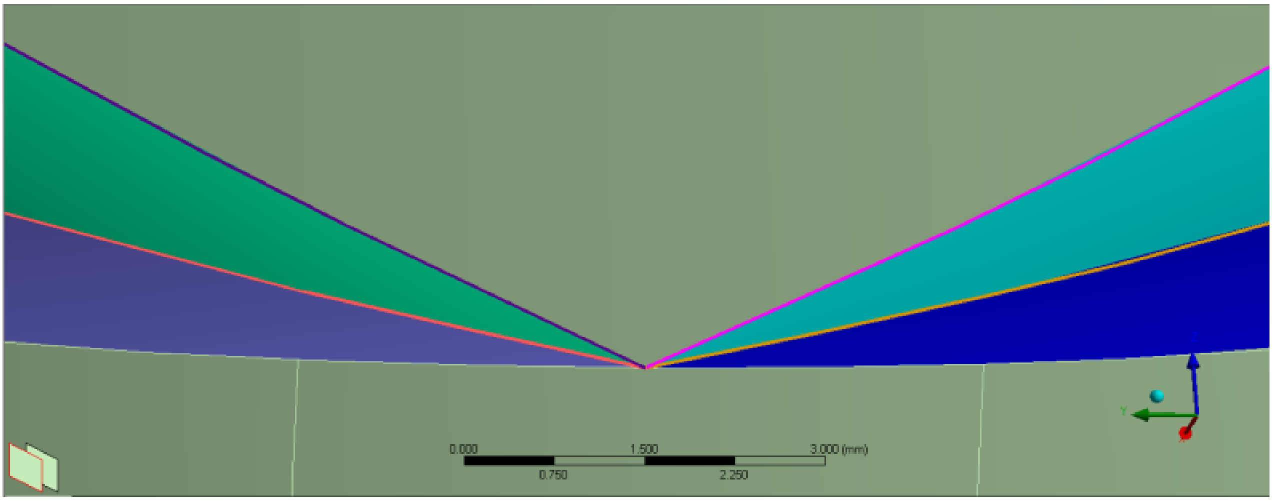
       The following example, the first figure shows small edges between faces.

       Figure 227: Small Edges Between Faces



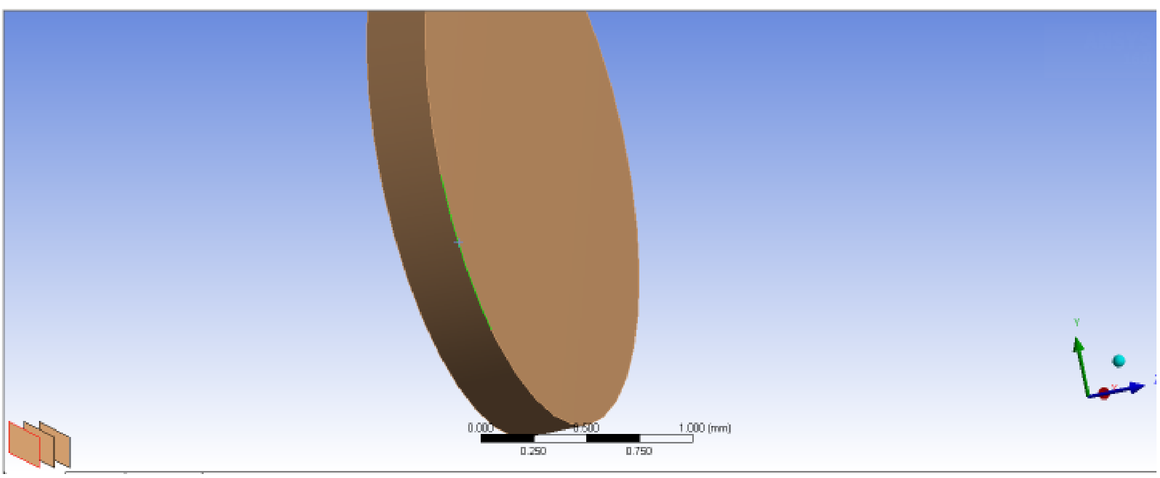
After **Small Edge Repair** (as shown in the following figure) with **Max Edge Length** set to 0.5 mm and **Min Edge Length** set to 0 mm, the small edges are removed by the faces being merged.

Figure 228: Small Edges Removed



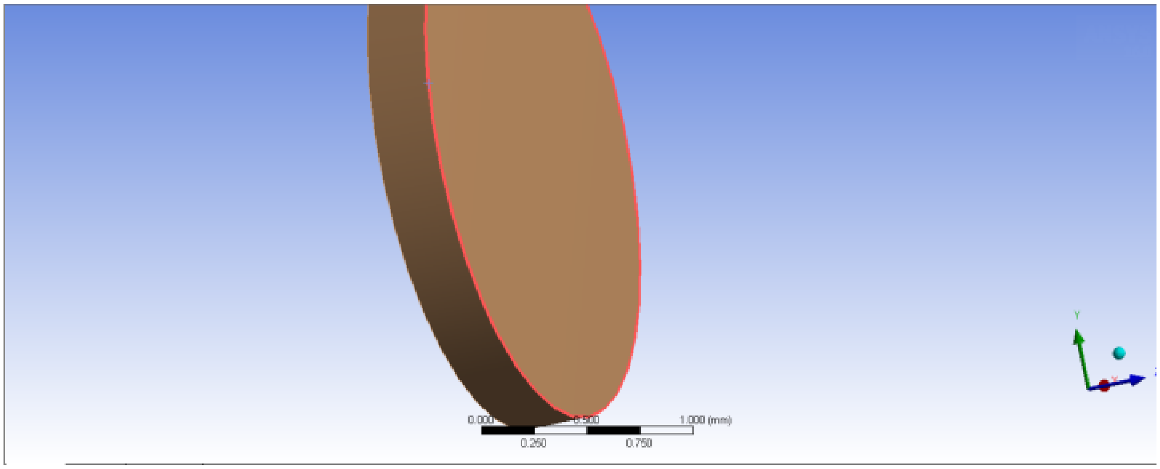
In the following figure, small edges are attached to the same faces.

Figure 229: Small Edges Attached to the Same Faces



After **Small Edge Repair** with **Max Edge Length** set to 0.9 mm and **Min Edge Length** set to 0 mm, the small edges are merged.

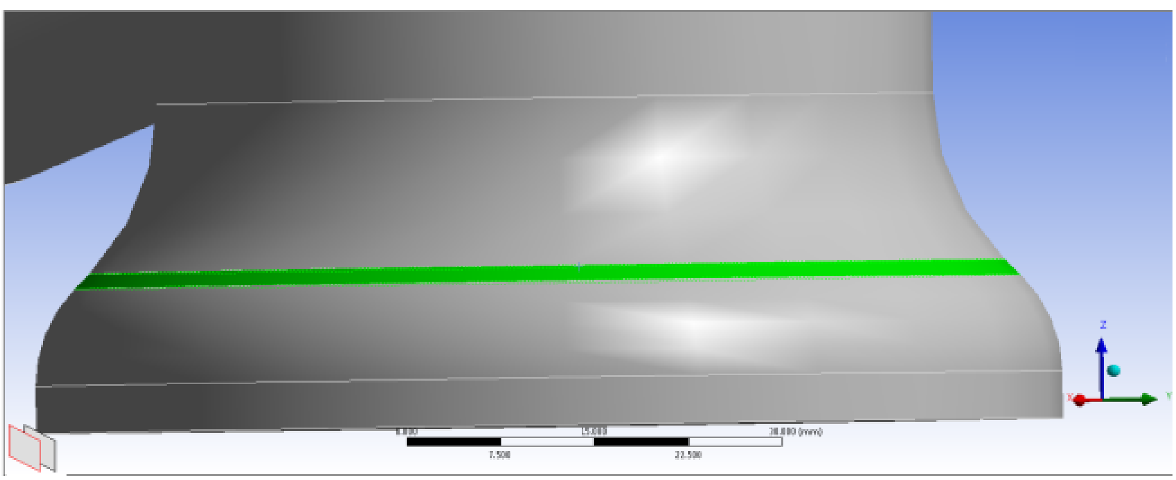
Figure 230: Small Edge Repair with Edge Merge



- The **Repair Slivers** option tries to remove all the slivers, satisfying the **Repair Settings** criteria shown in the Details View. The slivers are removed by merging the sliver face with another adjacent face. The **Repair Slivers** option exposes the **Max Sliver Width** and **Min Sliver Width** options. The sliver faces with width between **Max Sliver Width** and **Min Sliver Width** are repaired.

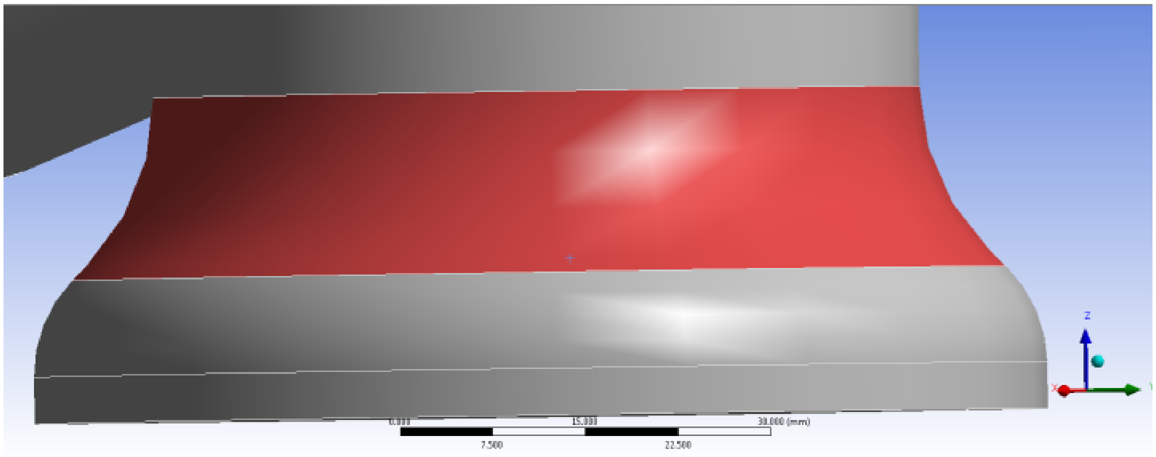
  The following figure shows a geometry with a sliver face.

  Figure 231: Sliver Face

  After **Sliver Repair** with **Max Sliver Length** set to 2 mm and **Min Sliver Length** to 0 mm, the sliver face is merged with an adjacent face.

Figure 232: Sliver Repair

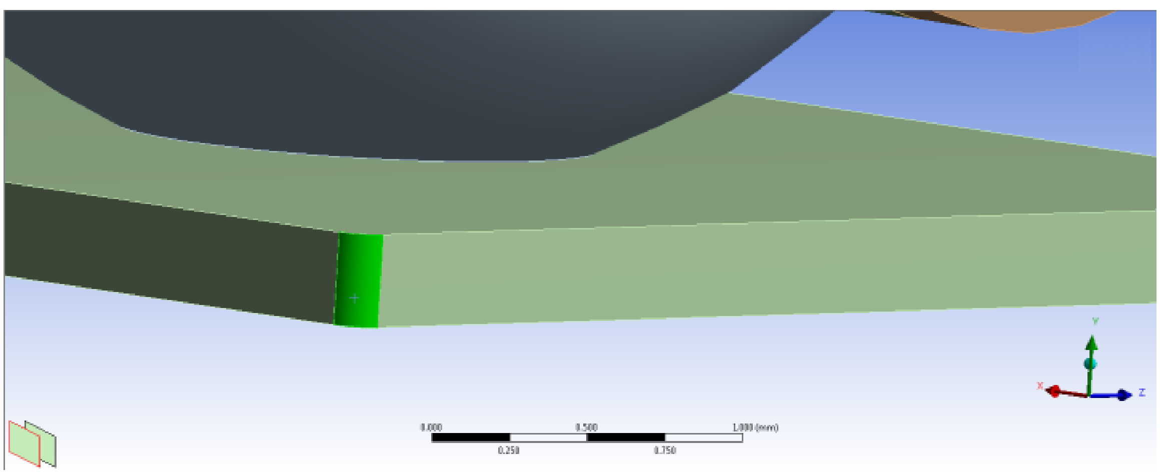


– The **Repair Small Faces** option tries to remove all the small faces satisfying the **Repair Settings** criteria shown in the Details View. The small faces are removed by merging the small face with another adjacent face.

The **Repair Small Faces** option exposes the **Max Face Area** and **Min Face Area** options. The faces with area between **Max Face Area** and **Min Face Area** are repaired.

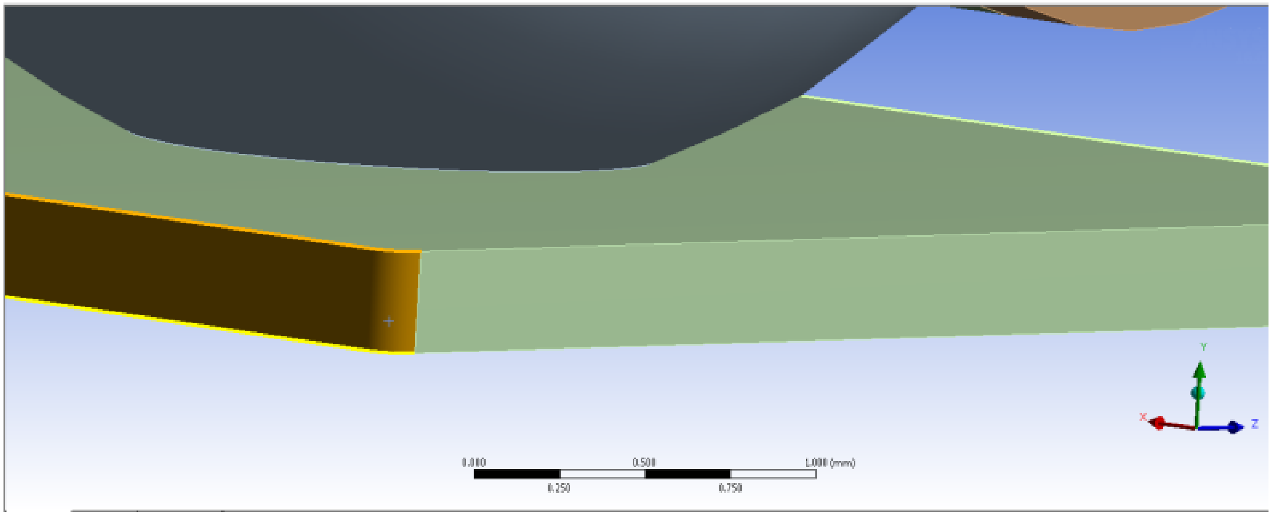
The following figure shows a small face.

Figure 233: Small Face



After small face repair with **Max Face Area** set to 0.04 mm^2 and **Min Face Area** set to 0 mm^2, the small face is merged with an adjacent face.

Figure 234: Small Face Repair



– The **Repair All** setting tries to remove all of the small edges, sliver faces, and small faces that satisfy the **Repair Settings** criteria shown in the Details View.

Repair Settings– Exposes settings like **Max Edge Length** and **Min Edge Length**, depending on the **Behavior** settings. The default values for different **Repair Settings** are:

→ **Max Edge Length**: Default is set to **½ Min Size** (available under **Sizing** in mesh when **Size Function** is turned ON)

→ **Min Edge Length**: Default is **0**, or no lower limit

→ **Max Face Area**: Default is set to (½ Min Size)^2.

→ **Min Face Area**: Default is 0, or no lower limit

→ **Max Sliver Width**: Default is set to ½ Min Size

→ **Min Sliver Width**: Default is 0, or no lower limit

Note:

Setting any of the Repair Settings to **-1** resets the value of that setting back to its default.

---

4. Make adjustments to the Advanced (p. 528) settings in the Details view.

5. Do one of the following:

   • Click right mouse button on the **Virtual Topology** object and choose **Generate Virtual Cells** from the context menu. Virtual cells are automatically created for each region that meets the criteria established by the settings in step 2.

     These virtual cells remain valid after a geometry update.

- Choose the face (**Ctrl**+**F**) or body selection filter and pick two or more faces or two or more bodies. Click right mouse button on the **Virtual Topology** object and choose **Generate Virtual Cells on Selected Entities** from the context menu. From the selected set of faces or bodies, the software groups adjacent entities appropriately and automatically creates virtual cells for each region that meets the criteria established by the settings in step 3.

  These virtual cells become invalid after a geometry update.

Creating and Managing Virtual Split Edges

When preparing geometry for meshing, it may be advantageous to split an edge into two virtual edges separated by a vertex. For example, in the case of a rectangular face in which a single edge on one side of the face corresponds to two edges on the opposite side, you can split the single edge so that node alignment across the face can have similar spacing. This can be achieved by creating **Virtual Split Edge** objects. You can also modify the split interactively by dragging the newly created vertex along the length of the original edge thus altering the split location.

Splitting an Edge

1. Insert a **Virtual Topology** object in the tree.
2. Choose the edge selection filter (**Ctrl**+**E**) and then in the **Geometry** window, pick the edge that you want to split. The selected edge can be either a "real" edge or a previously-defined virtual edge.

   Note:

   To simplify specification of the split location, when picking the edge to split you should position your cursor at the point on the edge where you want the split to occur. Then select **Virtual Split Edge at +** as described in step 3.

3. Create the **Virtual Split Edge** using either of these methods:

   - To define the split location according to your cursor location on the edge, right-click in the **Geometry** window and select **Insert> Virtual Split Edge at +** from the context menu, or choose **Split Edge at +** on the **Virtual Topology** context toolbar.
   - To define the split without specifying the location, select the edge you want to split, right-click in the **Geometry** window, and select **Insert> Virtual Split Edge** from the context menu, or choose **Split Edge** on the **Virtual Topology** context toolbar. By default the split ratio will be set to 0.5, but you can change it later by using the **Virtual Topology Properties** dialog (p. 529).

   Note:

   If the software cannot create the split, the error message “Unable to split the edge at selected location or with given split ratio” will appear and the split will not be created.

4. In step 3, if you created the **Virtual Split Edge** object by selecting **Virtual Split Edge at +**, the split ratio is determined automatically by the software. If you created it by selecting **Virtual Split Edge**, you can either accept the default of 0.5 or specify a different split ratio by right-clicking and selecting **Edit Selected Virtual Entity Properties...**. Then edit the **Split Ratio** field on the **Virtual Topology Properties** dialog.

   The split ratio defines the location of the split by specifying the ratio between the distance from the start point of the edge to the split location and the overall length of the edge. Specify a value from 0 to 1. For example, a value of 0.5 will split the edge into two edges of equal length. A value of 0.75 will split the edge into two edges where the first edge is three quarters of the length of the original edge, and the second edge is only one quarter of the length of the original edge. A value of 0 or 1 is valid only if the selected edge is a closed edge. In the case of a closed edge, such as a circle, the edge will be split into two edges, and the new vertex will be placed along one of the new edges.

   You can change the location of the split at any time by accessing the **Virtual Topology Properties** (p. 529) dialog and editing the **Split Ratio** field, or by modifying the edge split interactively.

   To modify an edge split interactively, select any portion of the original edge or its vertex split location in the **Geometry** view. Then, while pressing **F4** on the keyboard, drag the mouse along the length of the edge to redistribute the split ratio. The display indicates the initial 3-D location of the split, together with a preview of its new location and split ratio.

   When you change the split ratio of an edge split that is attached to a face split, both the edge split and the face split are adjusted accordingly.

Virtual Split Edge Dependency

Existing virtual edges and/or existing virtual split edges can be used to create new virtual split edges. Within the split edge hierarchy, a virtual split edge depends on a virtual edge if the latter is used to create the former. Similarly, a virtual split edge depends on another virtual split edge if the latter is used to create the former.

Note:

If a virtual edge was created from a virtual split edge, you cannot delete the virtual split edge without first deleting the virtual edge. Conversely, if a virtual split edge was created from a virtual edge, you cannot delete the virtual edge without first deleting the virtual split edge.

A warning message appears in the **Messages** window for each failed deletion. To highlight the geometry that is responsible for a message, select the message, right-click, and select **Show Problematic Geometry** from the context menu.

Locking Locations of Dependent Virtual Split Edges

In cases involving virtual split edge dependency, you can choose whether the locations of dependent splits are locked or unlocked when the location of the parent split is modified. If unlocked, the location of the dependent split moves to maintain its defined split ratio when the parent is modified. If locked, the location of the dependent split persists when the parent is modified. Regardless of the setting, the location of the dependent split will move if preserving its location would invalidate the split (see Figure 238: Overridden Locked Dependent Splits (p. 520) for an example).

Note:

For parametric updates, all virtual split edges are treated as unlocked.

1. Highlight the **Virtual Topology** object in the tree.

   Note:

   Locking must be set globally on the **Virtual Topology** object. It cannot be set locally for individual **Virtual Split Edge** objects.

2. In the Details View, set the **Lock position of dependent edge splits** option to **Yes** (default) or **No**.

The following figures illustrate locking. Figure 235: Original Virtual Split Edge with Dependent Virtual Split Edge (p. 519) shows an example in which a split at .5 was defined on original edge **ab**, creating two new edges—**ac** and **cb**. A second split at .5 was then defined on edge **cb**, creating two new edges—**cd** and **db**.

Figure 235: Original Virtual Split Edge with Dependent Virtual Split Edge

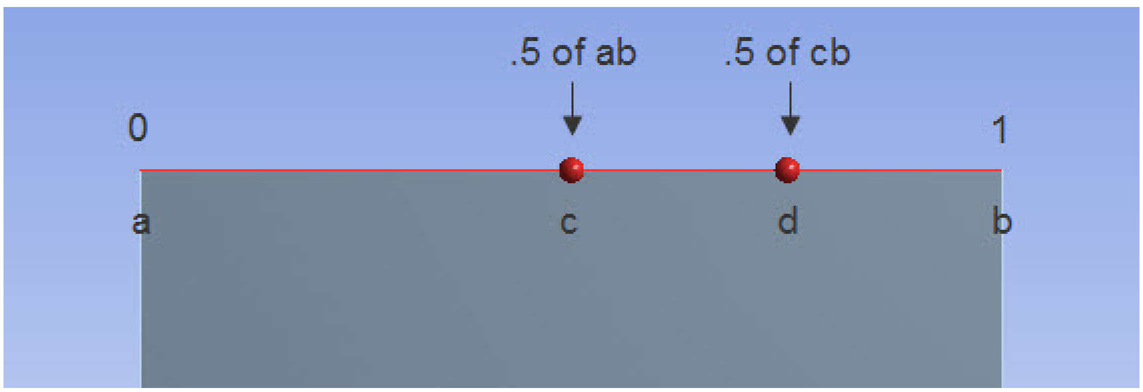


Figure 236: Unlocked Dependent Splits (p. 520) shows the expected behavior when **Lock position of dependent edge splits** is set to **No**, and the split located at point **c** is changed from .5 to .1. Notice that point **d** has moved to maintain its defined split ratio at .5 of **cb**.

Figure 236: Unlocked Dependent Splits

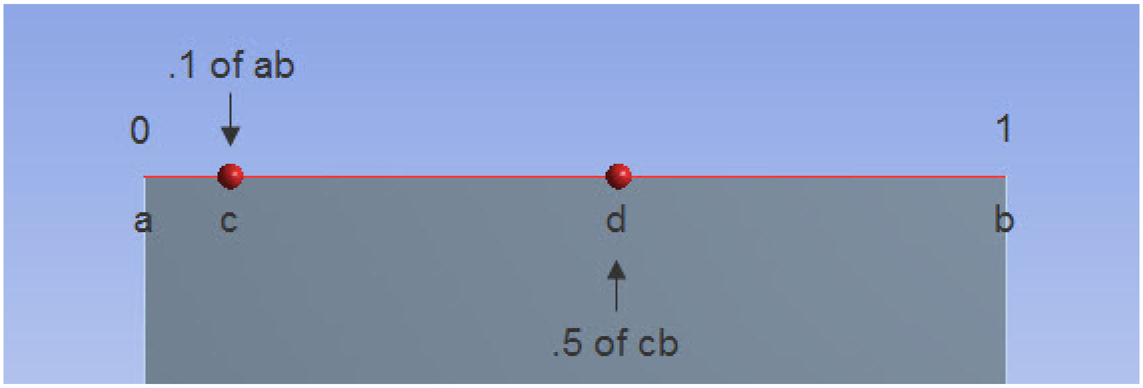
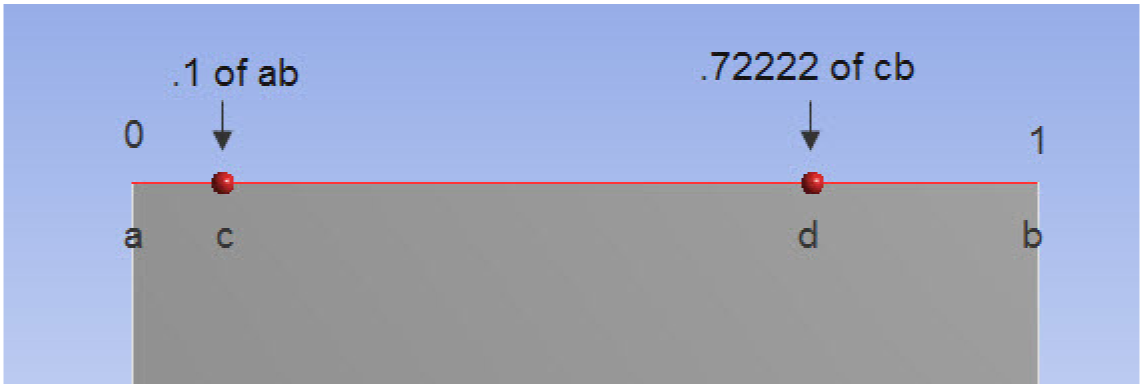


Figure 237: Locked Dependent Splits (p. 520) shows the expected behavior when **Lock position of dependent edge splits** is set to **Yes**, and the split located at point **c** is changed from .5 to .1. Notice that in this case point **d** has not moved, and the split ratio is now .72222 of **cb**.

Figure 237: Locked Dependent Splits



In Figure 238: Overridden Locked Dependent Splits (p. 520), the split located at point **c** was changed from .5 to .9. Even with locking set to **Yes**, in this case point **d** was moved because preserving its location would have invalidated the split (that is, if its original location had been preserved, point **d** would no longer be located on edge **cb**).

Figure 238: Overridden Locked Dependent Splits

Notes on Virtual Split Edges

- When the **Sweep** mesh method (p. 223) is used, the guiding edges must have consistent lengths to obtain a regular mesh in the sweep direction. You can define virtual split edges to achieve consistent lengths for these edges.
- If your **Geometry** view is configured to display **Shaded Exterior and Edges**, you can conveniently review virtual split edges by clicking the **Virtual Topology** group in the Tree Outline. In this mode, split edges will be highlighted in two different colors (automatically assigned) to draw attention to the splits.
- You cannot use the virtual split edge feature to split edges belonging to line bodies.

Creating and Managing Virtual Split Faces

When preparing geometry for meshing, it may be advantageous to split a face. This can be achieved by creating **Virtual Split Face** objects, which allow you to split a face along two vertices to create 1 to N virtual faces. The selected vertices must be located on the face that you want to split.

Splitting a Face

1. Insert a **Virtual Topology** object in the tree.
2. Choose the vertex selection filter and then in the **Geometry** window, pick two vertices on the face that you want to split.

   Note:

   - The vertices must be attached to the same face and in the same part.
   - Optionally, you can create one or more **Virtual Hard Vertex** objects to facilitate the split face operation. **Virtual Hard Vertex** objects allow you to define a hard point according to your cursor location on a face, and then use that hard point in the split face operation. Refer to Creating and Managing Virtual Hard Vertices (p. 524).
   - To see all vertices in the **Geometry** window, including any virtual hard vertices, make sure that the **Show Vertices** option is enabled.

3. Create the **Virtual Split Face** by doing one of the following:
   - Choose **Split Face at Vertices** on the **Virtual Topology** context toolbar.
   - Click right mouse button on the **Virtual Topology** object and select **Insert> Virtual Split Face at Vertices** from the context menu.

- Click right mouse button in the **Geometry** window and select **Insert> Virtual Split Face at Vertices** from the context menu.

Note:

- If the software cannot create the split, the error message "Unable to split face at selected vertices, please ensure vertices are attached to the same face and in the same part" will appear and the split will not be created.
- To change the location of a virtual hard vertex interactively, select it in the **Geometry** view. Then, while pressing **F4** on the keyboard, drag the mouse to move the vertex to the desired location. You cannot move the vertex beyond the face on which it was created. The display indicates the initial 3-D location of the vertex, together with a preview of its new location.

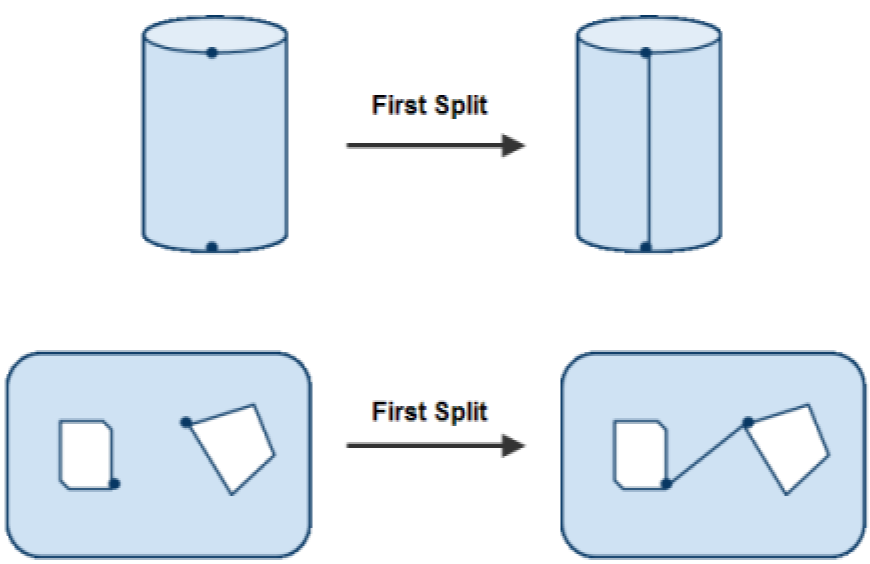
  When you change the location of a virtual hard vertex, the corresponding face split is adjusted accordingly.

  You cannot use the **Virtual Topology Properties** dialog to change the location of a virtual hard vertex.

Special Considerations for Virtual Split Faces

Certain types of faces cannot be split by a single split operation; they require two splits. For example, refer to the cylindrical face in the figure below. As a result of the first split, a seam edge is created and the face is no longer a periodic face. To actually split the cylindrical face, you then must select two more vertices and split the face a second time.

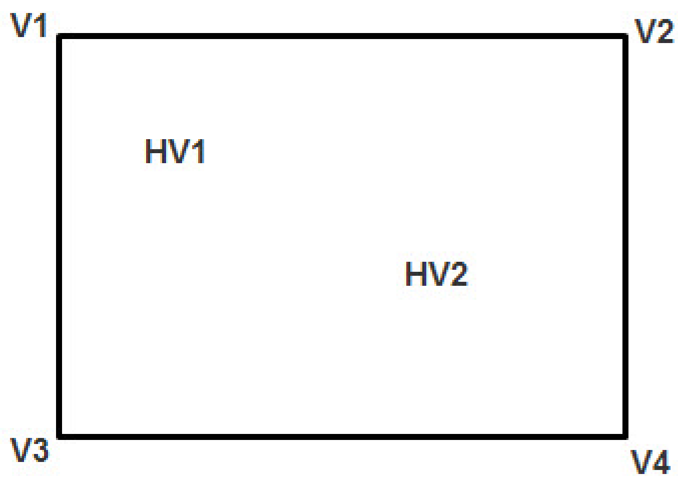
Figure 239: Types of Faces Requiring Two Virtual Split Face Operations



To split a face along more than two vertices you must perform a series of split face operations using two vertices at a time. For example, refer to the figure below, where V1, V2, V3, and V4 are vertices, and HV1 and HV2 are virtual hard vertices. Splitting the face through V1, HV1, HV2, and V4 would require three split operations:

1. Select V1 and HV1 and split the face.
2. Select HV1 and HV2 and split the face.
3. Select HV2 and V4 and split the face.

Figure 240: Splits Requiring a Series of Virtual Split Face Operations



Virtual Split Face Dependency

Existing virtual faces and/or existing virtual split faces can be used to create new virtual split faces. Within the split face hierarchy, a virtual split face depends on a virtual face if the latter is used to create the former. Similarly, a virtual split face depends on another virtual split face if the latter is used to create the former.

Faces can be split by selecting vertices of virtual split edges, providing the vertices meet the requirements described in the procedure above.

Note:

If a virtual face was created from a virtual split face, you cannot delete the virtual split face without first deleting the virtual face. Conversely, if a virtual split face was created from a virtual face, you cannot delete the virtual face without first deleting the virtual split face.

If a virtual split face was created from vertices of virtual split edges, you cannot delete the virtual split edges without first deleting the virtual split face.

If a virtual split face was created from virtual hard vertices, you cannot delete the virtual hard vertices without first deleting the virtual split face.

A warning message appears in the **Messages** window for each failed deletion. To highlight the geometry that is responsible for a message, select the message, right-click, and select **Show Problematic Geometry** from the context menu.

Creating and Managing Virtual Hard Vertices

You can create one or more **Virtual Hard Vertex** objects to facilitate a split face operation (p. 521). **Virtual Hard Vertex** objects allow you to define a hard point according to your cursor location on a face, and then use that hard point in the split face operation.

Creating a Virtual Hard Vertex

1. To create a virtual hard vertex (hard point), choose the face selection filter (**Ctrl**+**F**) and then in the **Geometry** window, pick the face that you want to split. Position your cursor on the face where you want the hard point to be located, left-click, and do one of the following:
   - Right-click in the **Geometry** window and select **Insert**> **Virtual Hard Vertex at +** from the context menu.
   - Choose **Hard Vertex at +** on the **Virtual Topology** context toolbar.

Hanging Edges Resulting From Split Faces

It is important to understand the effect hard vertices and hanging edges can have on meshing. If a face is not completely split, the partial face split is treated as a hanging edge and can create constraints on the mesher. If a virtual hard vertex is created but is not used for a split (or it comes in from the geometry), it can also create constraints on the mesher.

Common Virtual Topology Operations

This section describes common virtual topology operations.

Importing Legacy Models

Upon import of a legacy model into release 2021 R2, suppressed virtual topology entities will be deleted. This includes any virtual topology entities that were suppressed manually (for example, by right-clicking on the virtual topology entity in the Tree Outline and selecting **Suppress** from the context menu), but it does not include virtual topology entities that are suppressed because the body containing them is suppressed. If entities are deleted, a warning message will be issued advising you to import the model into an earlier release, unsuppress the affected entities, and save the model for use in release 2021 R2.

Handling of Geometry Changes That Result in Incomplete Virtual Entities

The Virtual Topology object that appears in the Tree Outline represents all definitions of virtual face or virtual edge groups, and all definitions of virtual split edges, virtual split faces, and virtual hard vertices within a model. Individual objects for these virtual entities do not appear in the Tree. If a geometry operation invalidates a virtual entity, refreshing the geometry does not cause the **Virtual Topology** object in the Tree Outline to become underdefined. For example, if you include a fillet and one neighboring face in the creation of a virtual cell, but later remove the fillet from the CAD model and refresh the geometry, that individual virtual cell will become underdefined (as it only includes the one neighboring face), but it will not be deleted, and there will be no change in the Tree Outline. If in a later operation, the fillet is re-added to the CAD model and refreshed, the virtual cell will be restored. When a virtual entity becomes underdefined due to a geometry operation, a message is issued. You might be able to view partial Virtual Entities by right-clicking on the **Show Problematic Geometry** message.

Underdefined Virtual Topologies are not displayed in Virtual Topology graphics, nor are they included in Virtual Topology statistics.

Note:

If you suppress a part or all bodies in a multibody part in the DesignModeler application, and refresh the geometry in the Meshing application, any virtual topology that had been defined on those bodies will be removed. The virtual topology will not be removed if only some bodies within a multibody part are suppressed in DesignModeler.

Using Named Selections with Virtual Topology

If you are using Named Selections with virtual topology and you subsequently modify the virtual topology, you must manually update the Named Selections. For example, if you create a Named Selection to define local inflation and then define virtual topology on that Named Selection, you must update the Named Selection before generating the mesh or the inflation will not be defined correctly. This limitation does not always occur if you perform a similar operation using the DesignModeler application. For example, after you perform a merge operation in DesignModeler, the software may be able to relink the Named Selection to the topology automatically when the geometry is refreshed.

Cycling Through Virtual Entities in the Geometry Window

You can use the ← and → buttons on the **Virtual Topology** context toolbar to cycle through virtual topology entities in the sequence in which they were created and display them in the **Geometry** window.

1. In the **Geometry** window, select a virtual entity.
2. On the **Virtual Topology** context toolbar, click ←or →.

Remember the following information when using ← and →:

- ← and → are grayed out until at least one virtual entity has been defined.
- If no virtual entities are selected, clicking →displays the first virtual entity in the sequence and clicking ← displays the last virtual entity in the sequence.

- If the currently selected virtual entity is the last in the sequence, clicking → displays the first virtual entity in the sequence. If the currently selected virtual entity is the first in the sequence, clicking ← displays the last virtual entity in the sequence.
- If using → when multiple virtual entities are selected, the entity that has the highest ID (based on the order in which the entities were created) is considered to be the current selection, and clicking → displays the entity that follows it.
- If using ← when multiple virtual entities are selected, the entity that has the lowest ID (based on the order in which the entities were created) is considered to be the current selection, and clicking ← displays the entity that precedes it.
- If the current multiple selection contains no virtual entities, ← and → work as though there are no selections (That is, clicking → displays the first virtual entity in the sequence and clicking ← displays the last virtual entity in the sequence). If the current multiple selection contains a mixture of virtual entities and non-virtual entities, the non-virtual entities are ignored.
- If a split is selected, both edges or all faces of the split are displayed.
- If any virtual entities are deleted or merged, the sequence is adjusted automatically. For example, the following behaviors occur if you create 14 virtual entities and then perform these actions in order:

  1. If you select the fifth virtual entity and click →, the sixth virtual entity is displayed.
  2. Click ←, and the fifth virtual entity is displayed.
  3. Delete the sixth virtual entity, select the fifth virtual entity, and click →. The seventh virtual entity is displayed. This occurs because the sixth virtual entity was deleted.
  4. Merge the seventh and eighth virtual entities, which creates a fifteenth virtual entity.
  5. Select the fifth virtual entity and click →. The ninth virtual entity is displayed. This occurs because the sixth virtual entity was deleted and the seventh and eighth were merged to form a new entity.

Deleting All Virtual Entities

You can use the **Delete All Virtual Entities** option to delete all virtual topology entities from a model at one time, regardless of type.

1. Highlight the **Virtual Topology** object in the Tree Outline or select any virtual topology entity in the **Geometry** window.
2. Right-click and select **Delete All Virtual Entities**.
3. Answer **Yes** at the prompt.

Deleting All Virtual Cells

You can use the **Delete All Virtual Cells** option to delete all virtual topology cells (the virtual topology group itself, including any virtual split edges, virtual split faces, or virtual hard vertices, will not be deleted). This option is available only when the **Virtual Topology** object in the Tree Outline is highlighted and virtual cells, as well as virtual split edges, virtual split faces, and/or virtual hard vertices exist. Otherwise, use the **Delete All Virtual Entities** option.

1. Highlight the **Virtual Topology** object in the Tree Outline.
2. Right-click on either the **Virtual Topology** object in the Tree Outline, or in the **Geometry** window. Select **Delete All Virtual Cells**.
3. Answer **Yes** at the prompt.

Deleting All Virtual Split Edges

You can use the **Delete All Virtual Split Edges** option to delete all virtual split edges (the virtual topology group itself, including any virtual cells, virtual split faces, or virtual hard vertices, will not be deleted). This option is available only when the **Virtual Topology** object in the Tree Outline is highlighted and virtual split edges, as well as virtual split faces and/or virtual hard vertices exist. Otherwise, use the **Delete All Virtual Entities** option.

1. Highlight the **Virtual Topology** object in the Tree Outline.
2. Right-click on either the **Virtual Topology** object in the Tree Outline, or in the **Geometry** window. Select **Delete All Virtual Split Edges**.
3. Answer **Yes** at the prompt.

Deleting All Virtual Split Faces

You can use the **Delete All Virtual Split Faces** option to delete all virtual split faces (the virtual topology group itself, including any virtual cells, virtual split edges, or virtual hard vertices, will not be deleted). This option is available only when the **Virtual Topology** object in the Tree Outline is highlighted and virtual split faces, as well as virtual split edges and/or virtual hard vertices exist. Otherwise, use the **Delete All Virtual Entities** option.

1. Highlight the **Virtual Topology** object in the Tree Outline.
2. Right-click on either the **Virtual Topology** object in the Tree Outline, or in the **Geometry** window. Select **Delete All Virtual Split Faces**.
3. Answer **Yes** at the prompt.

Deleting All Virtual Hard Vertices

You can use the **Delete All Virtual Hard Vertices** option to delete all virtual hard vertices (the virtual topology group itself, including any virtual cells, virtual split faces, or virtual split edges, will not be deleted). This option is available only when the **Virtual Topology** object in the Tree Outline is highlighted and virtual hard vertices, as well as virtual split faces and/or virtual split edges exist. Otherwise, use the **Delete All Virtual Entities** option.

1. Highlight the **Virtual Topology** object in the Tree Outline.
2. Right-click on either the **Virtual Topology** object in the Tree Outline, or in the **Geometry** window. Select **Delete All Virtual Hard Vertices**.

3. Answer **Yes** at the prompt.

Deleting Selected Virtual Entities

You can use the **Delete Selected Virtual Entities (And Dependents)** option to delete selected virtual topology entities, along with any dependents if applicable.

To use a right mouse button click:

1. Highlight any object in the Tree Outline (for example, the **Geometry** or **Mesh** object).
2. In the **Geometry** window, select the virtual entities that you want to delete.
3. Right-click and select **Delete Selected Virtual Entities (And Dependents)**.
4. Answer **Yes** at the prompt.

To use the **Virtual Topology** context toolbar:

1. Highlight the **Virtual Topology** object in the Tree Outline.
2. In the **Geometry** window, select the virtual entities that you want to delete.
3. Choose **Delete** on the **Virtual Topology** context toolbar.
4. Answer **Yes** at the prompt.

Also see:

- Creating and Managing Virtual Cells (p. 502)
- Creating and Managing Virtual Split Edges (p. 517)
- Creating and Managing Virtual Split Faces (p. 521)
- Creating and Managing Virtual Hard Vertices (p. 524)

Common Virtual Topology Features

This section describes features that are common to all types of virtual topology.

Setting Advanced Properties

The following Advanced settings are available in the Details View for **Automatic** and **Repair** modes.

- **Generate on Update** – Sets whether you want to include the settings in this Details View when you update the geometry.
- **Simplify Faces** – Removes hard edges and hard vertices from the selection.
  - **Yes**: If the **Simplify Faces** property is turned on, the program removes hard edges and hard vertices.

  - **No**: If the **Simplify Faces** property is turned off, faces are not simplified.

- **Merge Face Edges** – The property is relevant only during the virtual face creation process. It applies only to manually-created virtual faces and can be modified at any time, but the modification will have no effect on previously-created virtual faces.

  - **Yes**: If the **Merge Face Edges** property is turned on, the program will attempt to merge bounding edges of a newly manually-created virtual face and create virtual edges. The criterion used to merge edges is based on the **Behavior** setting.

  - **No**: If the **Merge Face Edges** property is turned off, only a virtual face will be created out of selected faces.

- **Lock position of dependent edge splits** – See Locking Locations of Dependent Virtual Split Edges (p. 518).

- **Virtual Faces**, **Virtual Edges**, **Virtual Split Edges**, **Virtual Split Faces**, **Virtual Hard Vertices**, **Total Virtual Entities** – Read-only indications of corresponding counts in the model. See Viewing Virtual Topology Statistics (p. 532).

Using the Virtual Topology Properties Dialog to Edit Properties

You can use the **Virtual Topology Properties** dialog to edit the properties of selected virtual topology entities.

To use a right mouse button click:

1. Highlight any object in the Tree Outline (for example, the **Geometry** or **Mesh** object).

2. In the **Geometry** window, select the virtual entities whose properties you want to edit.

3. Right-click and select **Edit Selected Virtual Entity Properties...**.

4. Make the desired changes in the **Virtual Topology Properties** dialog.

5. To apply any changes and/or exit, press **Enter** on your keyboard or click **X** on the dialog.

To use the **Virtual Topology** context toolbar:

1. Highlight the **Virtual Topology** object in the Tree Outline.

2. In the **Geometry** window, select the virtual entities whose properties you want to edit.

3. Choose **Edit** on the **Virtual Topology** context toolbar.

4. Make the desired changes in the **Virtual Topology Properties** dialog.

5. To apply any changes and/or exit, press **Enter** on your keyboard or click **X** on the dialog.

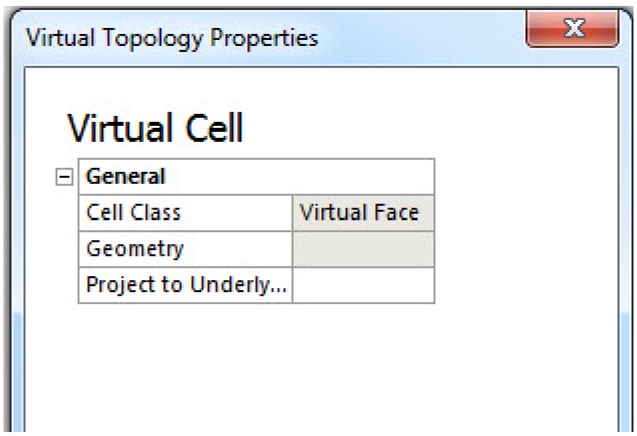
Remember the following information when using the **Virtual Topology Properties** dialog:

- If all selected virtual entities have the same value for a particular property, that value appears in the **Virtual Topology Properties** dialog. Otherwise, the value for that property is blank.

- Fields that are grayed out are read-only.
- The changes you make in the **Virtual Topology Properties** dialog will be applied to all selected virtual entities.
- If you change a split location, the graphic in the **Geometry** window will be redrawn.

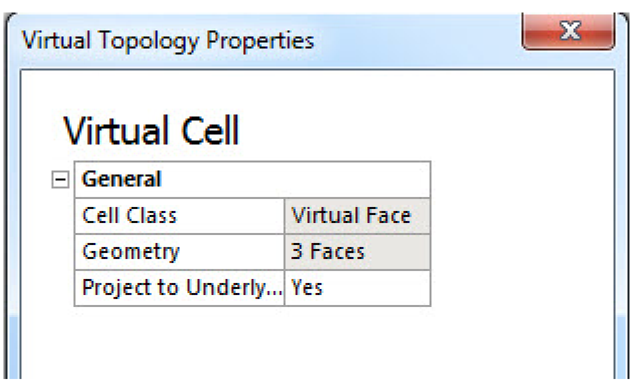
Consider the examples below. For the first example, two virtual faces were selected. One virtual face was composed of five faces, and its **Project to Underlying Geometry** option was set to **No**. The other virtual face was composed of three faces, and its **Project to Underlying Geometry** option was set to **Yes**.

Figure 241: Virtual Topology Properties Dialog: Example 1



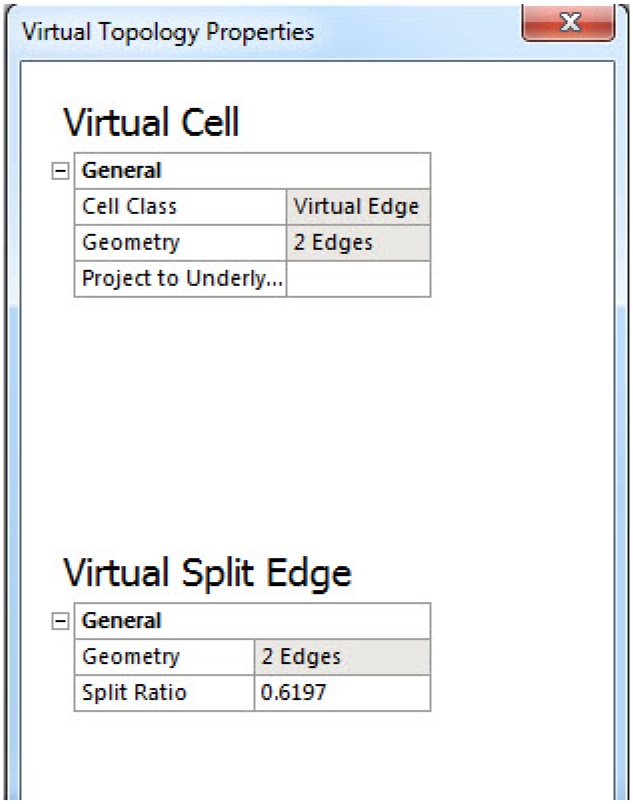
In the example below, two virtual faces were selected. In this case each virtual face was composed of three faces, and **Project to Underlying Geometry** was set to **Yes** for both virtual faces.

Figure 242: Virtual Topology Properties Dialog: Example 2



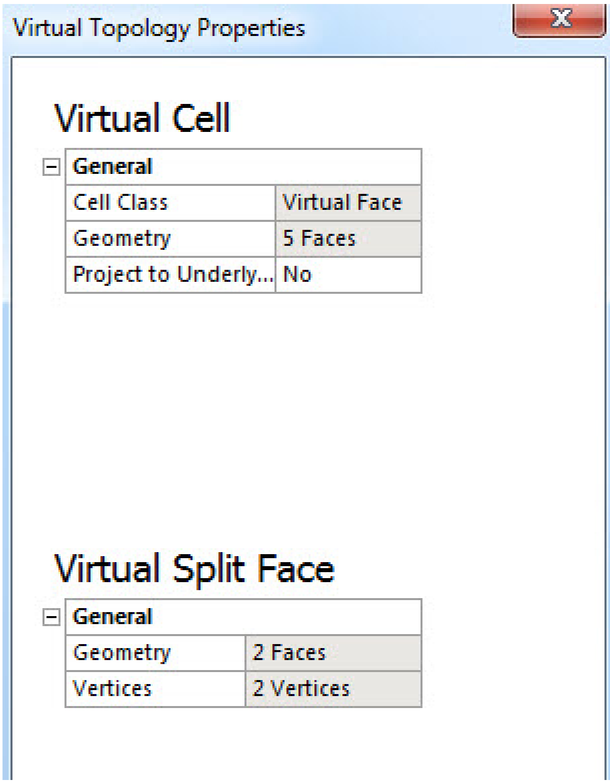
In the example below, two virtual edges and one virtual split edge were selected. Both virtual edges were composed of two edges, but **Project to Underlying Geometry** was set to **Yes** for one virtual edge and to **No** for the other.

Figure 243: Virtual Topology Properties Dialog: Example 3



In the example below, one virtual face and one virtual split face were selected. The virtual face was composed of five faces, and its **Project to Underlying Geometry** option was set to **No**.

Figure 244: Virtual Topology Properties Dialog: Example 4

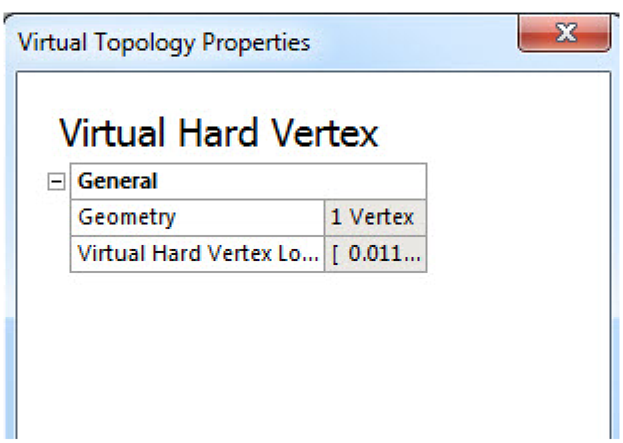


In the example below, one virtual hard vertex was selected.

Note:

Although you cannot use the dialog to modify the **Virtual Hard Vertex Location**, you can do so interactively using the **F4** key. See Creating and Managing Virtual Split Faces (p. 521) for details.

Figure 245: Virtual Topology Properties Dialog: Example 5



Viewing Virtual Topology Statistics

The Details View of the **Virtual Topology** object provides counts of the following:

- Virtual faces
- Virtual edges
- Virtual split edges
- Virtual split faces
- Virtual hard vertices
- Total virtual entities

Remember the following information when viewing virtual topology statistics:

- Statistics are updated whenever a geometry or virtual topology change occurs.
- If any body is suppressed, the virtual topology entities on that body are not counted.
- If you create a virtual topology entity and then use it to create another virtual topology entity, the former exists in the background in a suppressed state and is not counted.

Meshing: Troubleshooting

This section is intended to provide you with tips and strategies for avoiding and handling problems that may occur when using the Meshing application. Topics include:

- Using Lustre Parallel File Systems on Linux (p. 553)

Meshing Process

When you generate mesh, the mesh is generated as a separate process per part (unless using assembly meshing where all parts are meshed in one process). If there are multiple parts, each part can be meshed in parallel. While meshing, the status window provides details on what is happening in the meshing process. Other information, such as the amount of memory used can be found by using a task manager.

While the mesher is working at certain points it will highlight topology (edges, faces, bodies). If the mesher gets stuck for a long time on a particular topology, you should inspect the highlighted topology and possibly merge it with another topology using virtual cells, or adjust the mesh sizes in that area.

For details about topology highlighting during the meshing process, refer to Generating Mesh (p. 486). For information about how to set the default for topology highlighting, refer to Meshing Options on the Options Dialog Box (p. 317).

Identifying Poor Quality Mesh

The following approaches are recommended to improve the mesh quality and obtain a valid mesh:

- Set the **Display Style** to identify quality by color.
- Use the **Mesh Metric** (p. 123) option to view information about a number of quality statistics and set **Min/Max** values to find poor quality elements.
- Use the Worksheet as your **Scoping Method** and use worksheet data to define criterion-based measurements.

Recommended First Course of Action for Meshing Failures

If your mesh generation fails, it may be a partial or complete meshing failure. Your first course of action should be to examine any messages that the mesher returns to the **Messages** window. The messages include hints that explain why the meshing completely or partially failed. In some cases, you can right-click the message and select **Show Problematic Geometry** to highlight any entities associated with the message in the **Geometry** window and see what the failed mesh looks like.

The mesher also provides visual cues to identify obsolete and/or failed meshes. As shown in the figures below, failed meshes are shaded in maroon and obsolete meshes are colored yellow. The approximate location of the cause of the meshing failure is identified by a convergence of white lines.

Figure 246: Obsolete Mesh

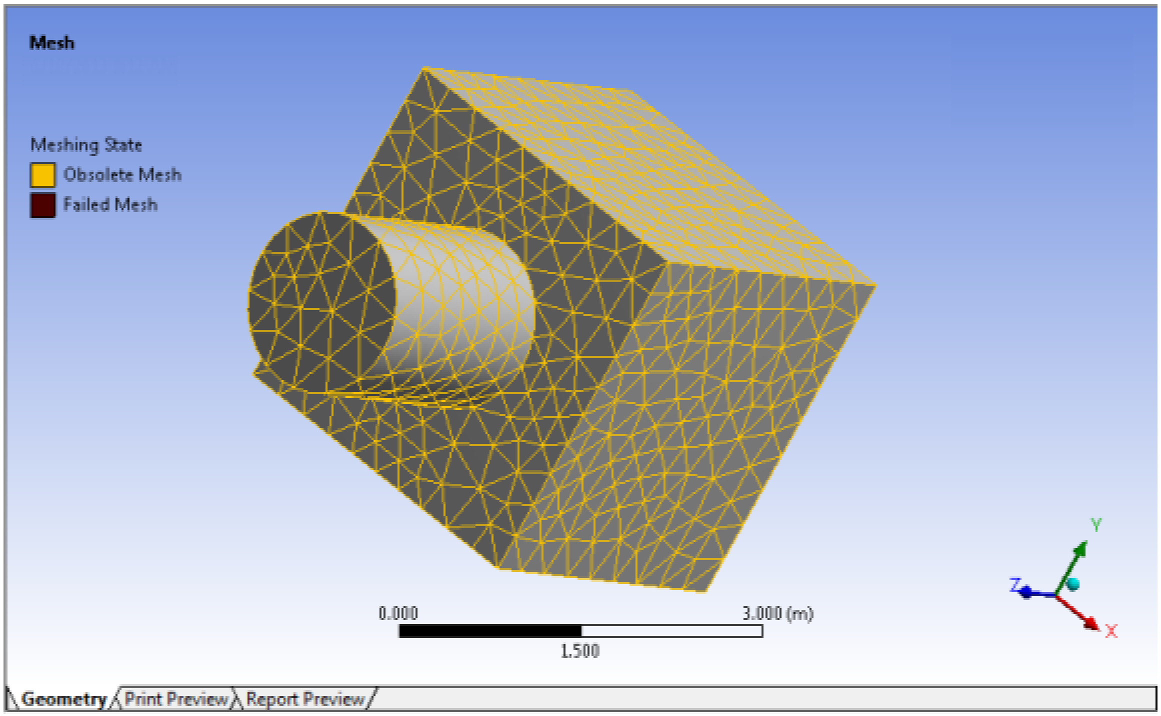
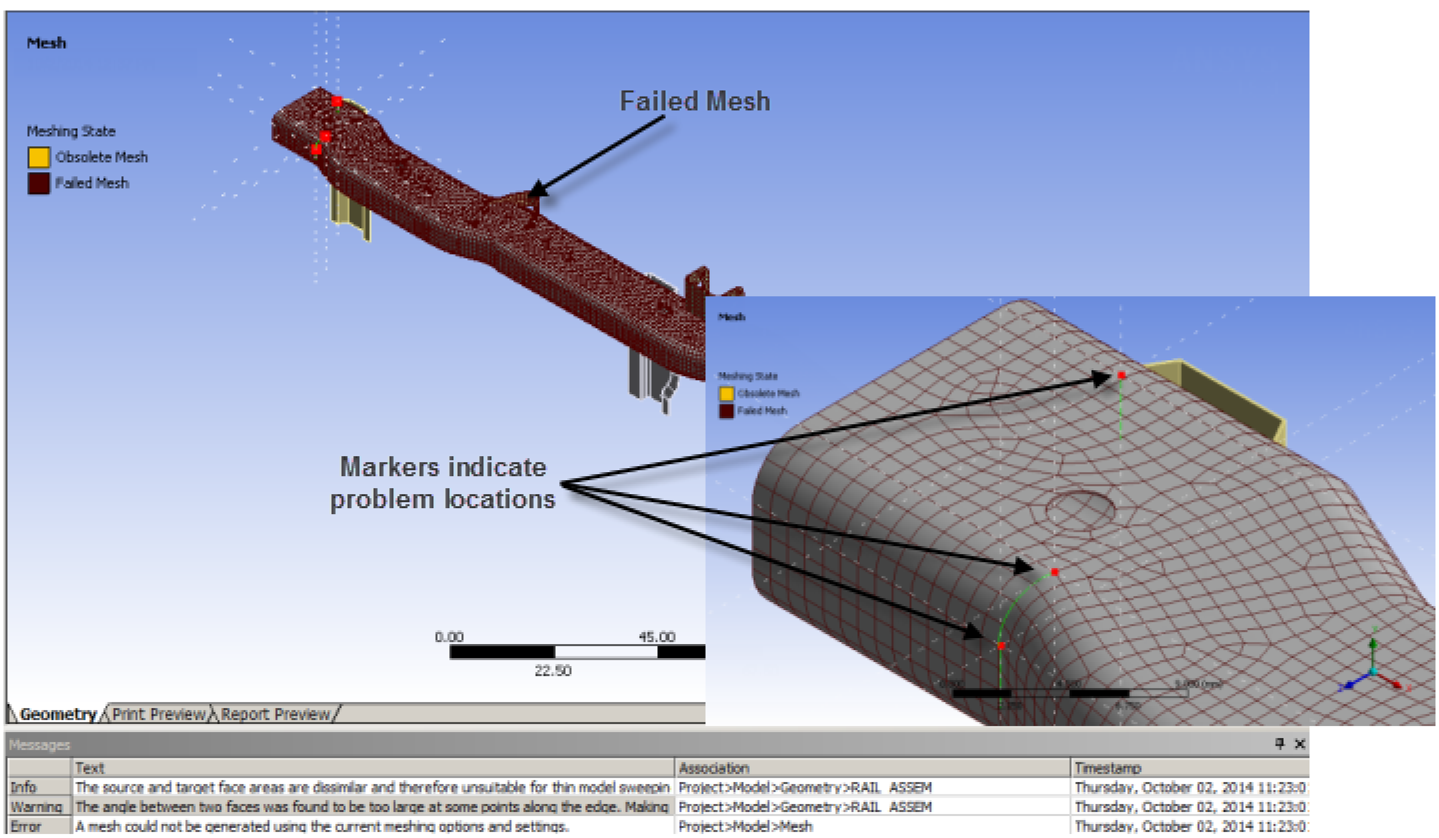


Figure 247: Failed Mesh



If the entities are very small, you can refer to the status bar at the bottom of the window to view statistics related to the entities. Then create a Named Selection to retain information about the problematic entities. Continue reading below for more information about messaging.

Understanding Messaging

The **Messages** window prompts you with feedback concerning meshing operations. Every message returned from the mesher is not necessarily an error. Messages come in three forms:

- **Error**: Requires action from you to obtain a mesh.
- **Warning**: Guides you in best practices or alternative courses of action. A warning does not require you to fix any problems, but may give an indication of a subsequent related error.
- **Information**: Helps you make better decisions related to meshing your model or provides you with information about the approach the mesher used to mesh your model.

Once messages are displayed, you can:

- Double-click a message to display its contents in a pop-up dialog box.
- Highlight a message and then press the key combination **Ctrl+C** to copy its contents to the clipboard.
- Press the **Delete** key to remove a selected message from the window.
- Select one or more messages and then use the right mouse button click to display the following context menu options:
  - **Go To Object** - Selects the object in the Tree Outline that is responsible for the message.
  - **Show Problematic Geometry** - Highlights the geometry in the **Geometry** window that is responsible for the message. This option is not always available.
  - **Show Message** - Displays the selected message in a pop-up dialog box.
  - **Copy** - Copies the selected messages to the clipboard.
  - **Delete** - Removes the selected messages.
  - **Refresh** - Refreshes the contents of the **Messages** window as you edit objects in the Tree Outline.

Understanding States

Sometimes the mesher returns an invalid mesh. Refer to the state of the body in the Tree Outline to determine whether a body was meshed:

- A check mark denotes a body in a fully defined but unmeshed state.
- A check mark with a hash denotes a meshed body.
- A question mark denotes a body that needs more information before it can be sent to the solver.

When your model contains an active unmeshed body, the **Mesh** object in the Tree Outline is preceded by a lightning bolt to indicate a body is out-of-date and requires action:

When your model is fully meshed (all bodies are in a meshed state), the **Mesh** object is preceded by a check mark to indicate that the meshing data is fully defined and ready for the next stage of your analysis (that is, an update in the Meshing application or a solve in the Mechanical application):

Understanding Mesh Connection and Contact Match States

Mesh connections and contact matches use the following symbols to denote mesh state:

- A check mark denotes a connection or match that is fully defined and ready to be connected or matched.
- A check mark with a hash denotes a successful connection or match.
- A question mark denotes a connection or match that is invalid that needs more information before it can be connected or matched.
- A circle with a bar in it denotes a connection or match that is ignored. Check the Details view of the mesh connection or contact match for the reason.
- A lightning bolt indicates that the connection or match is out-of-date and requires action.
- An 'X' denotes a suppressed connection or match. Unsuppressing the connection or match requires it to be reconnected or rematched even if it was previously connected or matched.

Shape Checks and Meshing Failures

Meshing may fail if the mesh quality does not meet the criterion of the defined shape checks (p. 118). The following approaches are recommended to improve the mesh quality and obtain a valid mesh:

1. To identify faces that do not meet the shape checking criteria, right-click the warning message and select **Show Problematic Geometry**.
2. Use a different shape check setting.

   Some shape checks (p. 118) have a stricter set of criterion than others. By using a different shape check setting a mesh might be generated, and the mesh metrics bar graph (p. 123) can be used to find the mesh violating the stricter shape checks. In this way, locating the problem is the first step to fixing it.

Note:

You can turn off most shape checks altogether by setting Check Mesh Quality (p. 118) to **No**.

3. Use the methods described in Identifying Poor Quality Mesh (p. 536) to determine the quality of the mesh.

4. Use the **Preview Surface Mesh** (p. 489) and/or **Preview Inflation** (p. 492) features.

   With this approach the boundary mesh is generated even if the mesh would violate the defined shape checks. Once the previewed mesh is generated, use the mesh metrics bar graph (p. 123) to determine the location of bad quality elements. Generally, fixing the bad quality surface mesh is the best way to fix the volume mesh because bad quality mesh is usually a result of the geometry over-constraining the mesh topology. Using defeaturing controls (such as Loop Removal (p. 192) and Mesh Defeaturing (p. 106)), pinch (p. 182) controls, and virtual topologies (p. 501) are all good strategies to remove geometry features that may cause problems for the meshing algorithms.

   Note:

   Not all mesh methods support the use of **Preview Surface Mesh** (p. 489) and **Preview Inflation** (p. 492).

For additional information about the shape checking acceptance criterion used by Ansys Workbench, refer to Ansys Workbench and Mechanical APDL Application Meshing Differences (p. 87).

Handling Selective Meshing Failures

Selective meshing (p. 404) may lead to unexpected results in cases where a mesh control change affects only one body. This may in turn lead to sweep mesh failure because the source and target meshes no longer align or the resultant change makes a body unsweepable. If desired, you can set the **Allow Selective Meshing** (p. 317) option to **No** to disable selective meshing and allow the mesh control changes to ripple through the entire part.

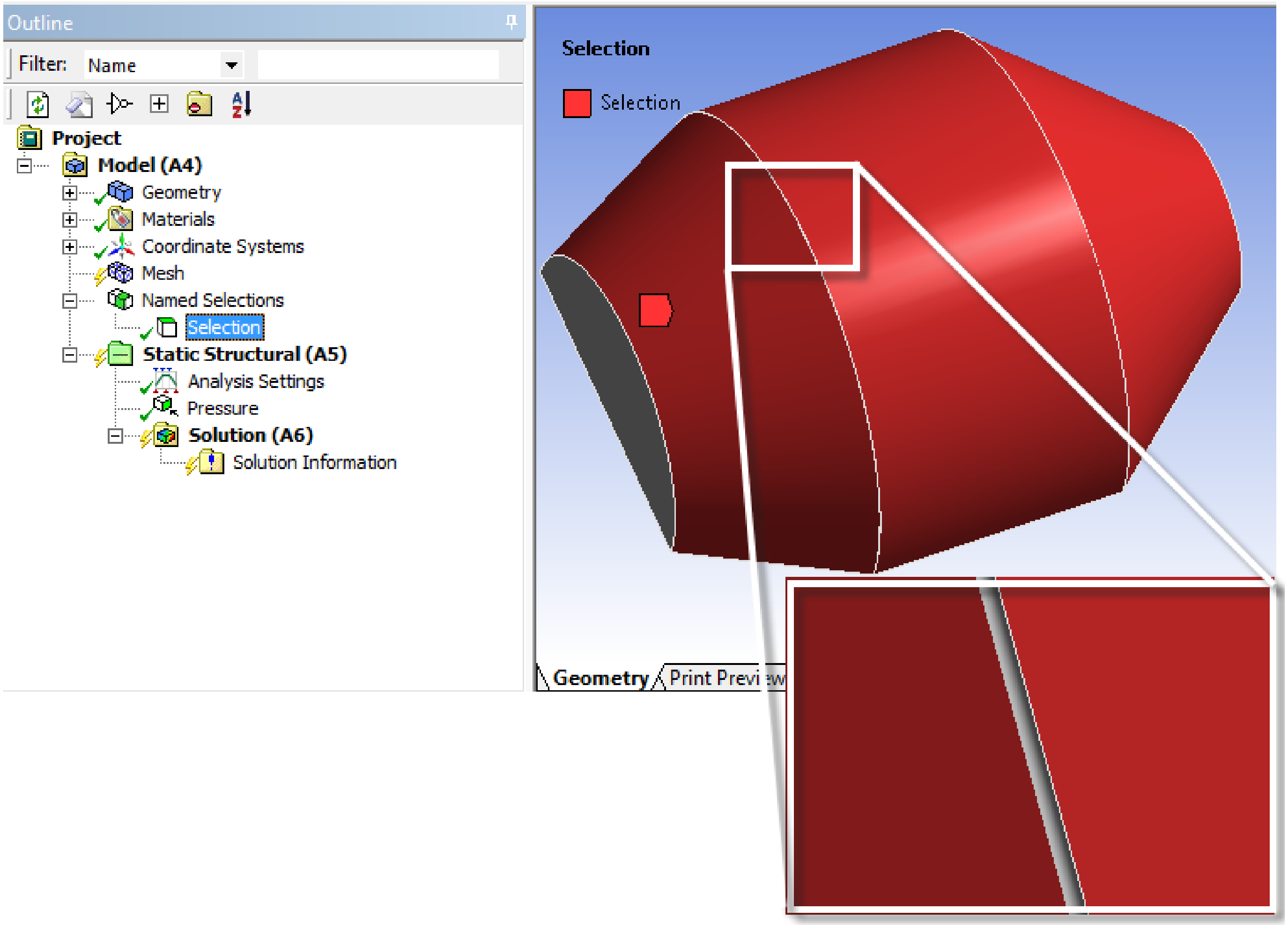
Selective meshing is not persistent for a geometry update or re-mesh operation. However, you can use the **Mesh** worksheet to create a selective meshing history so that your meshing steps can be repeated in the desired sequence. Otherwise, you may need to go through your body meshing steps manually if the single mesh update does not satisfy your meshing requirements. Refer to Using the Mesh Worksheet to Create a Selective Meshing History (p. 409) for details.

Handling Failures due to Protected Topology

Improper set up of hard protected topology (p. 180) may result in meshing failures if the mesher cannot return a reasonable mesh while also respecting the topology.

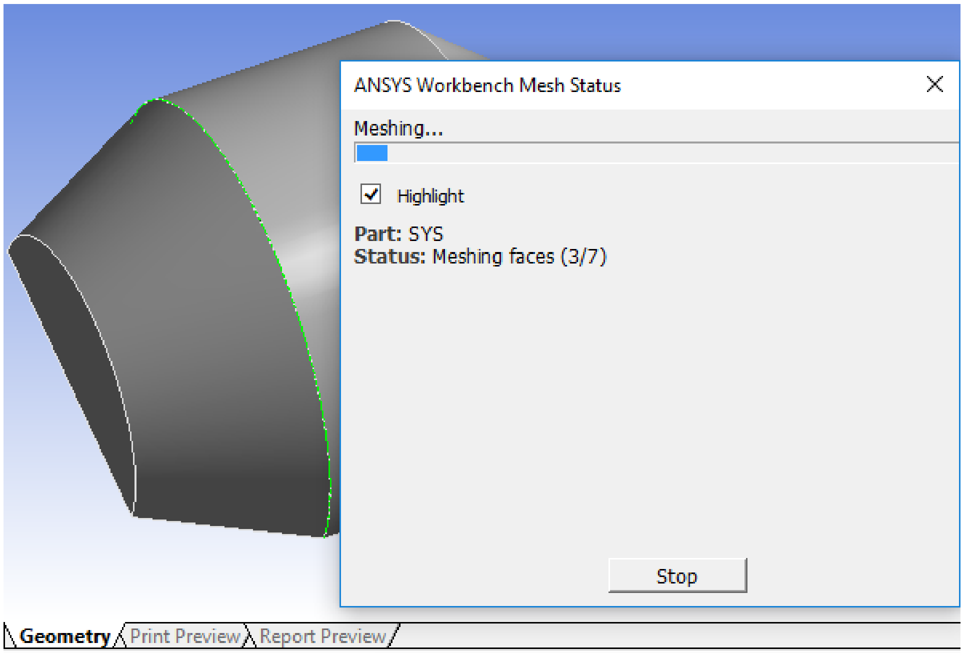
In the example, a sliver face is missing from the Named Selection defined. The **Protected** option is set to **Yes**.

Figure 248: Example with Missing Face



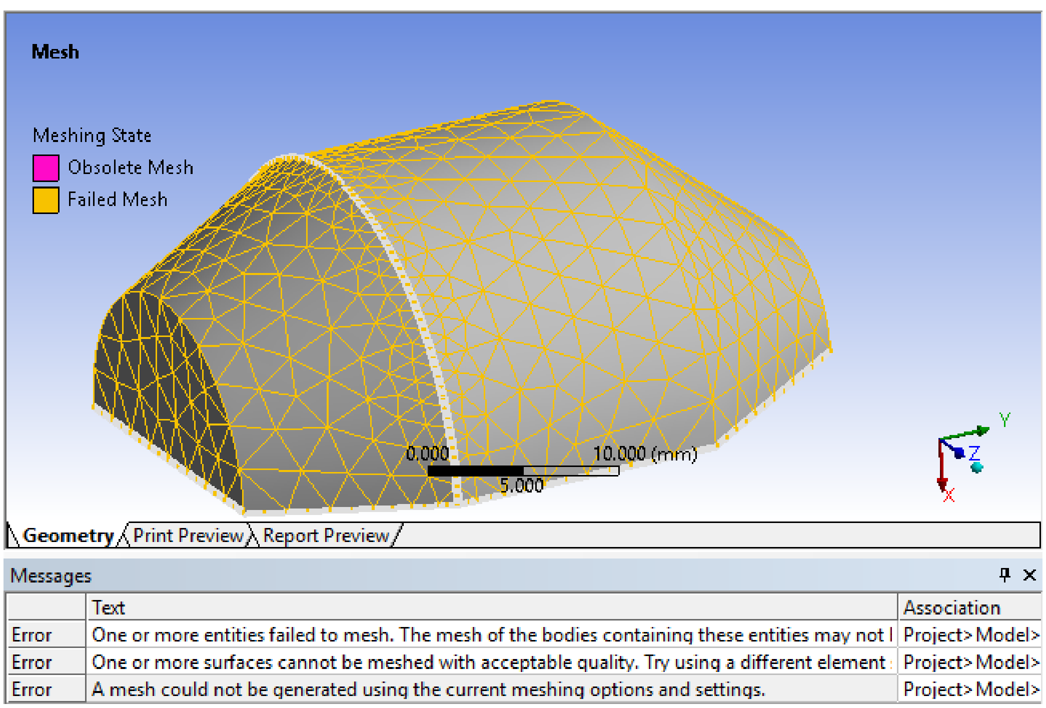
The missing face could lead to problems during meshing since boundaries of hard protected topologies receive special treatment to ensure the mesh is properly associated. The special protection ensures the outer boundary is captured accurately, however, it could have a negative impact on meshing since the mesher is forced to capture the sliver surface. With the **Highlight** option enabled, the problematic face is highlighted during the meshing process (Figure 249: Problematic Topology Highlighted During Meshing (p. 542)). If a face is highlighted for a long period of time it can indicate a problem meshing the face. In this case, the problem comes from the protected topology.

Figure 249: Problematic Topology Highlighted During Meshing



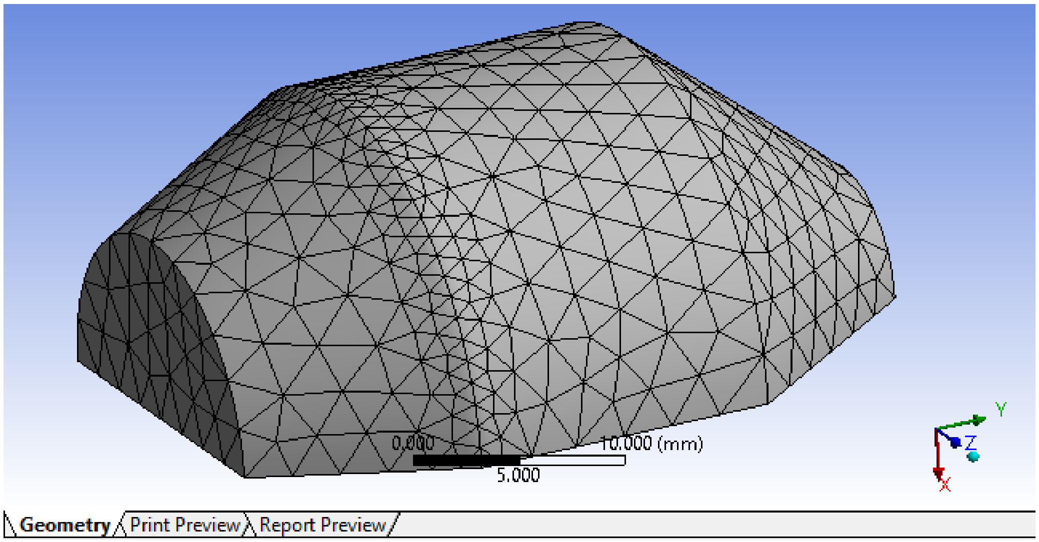
The missing face could also lead to other meshing failures. For example, the sizing applied may be such that the mesher cannot return a reasonable mesh while also respecting the topology. In this case, the mesher may fail in face meshing while trying to protect the boundaries of the Named Selection. The faces that fail to mesh are indicated in the message window (use the right mouse button to select the error message to show the problematic geometry).

Figure 250: Failed Surface Mesh Due to Protected Topology



Examine the model to locate objects scoped to either the sliver face or the neighboring faces of the sliver face (Named Selection). By including the sliver face in the Named Selection definition, the mesh can be generated while respecting the sizing and the topology.

Figure 251: Mesh Respecting Protected Topology



Handling Patch Independent Tet Meshing Failures

If there are gaps in your geometry and Patch Independent tet meshing fails, the mesh size may be set smaller than the size of the gaps in the geometry. In such cases, try adjusting the size of the mesh in those regions so they are larger than the geometry gap size.

Figure 252: Patch Independent Tet Mesh Failure Due to Geometry Gap

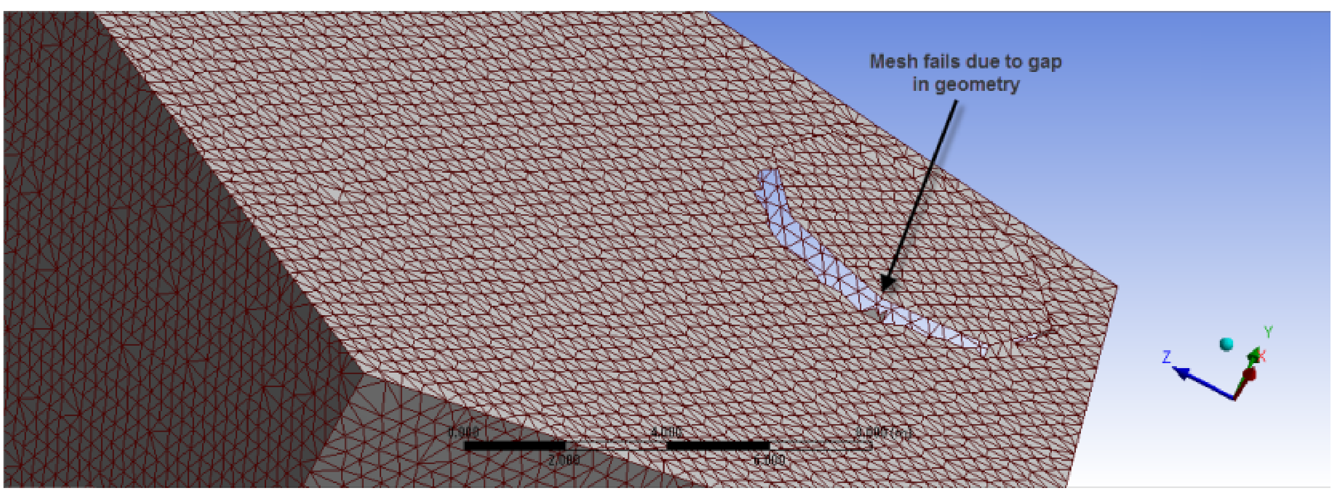
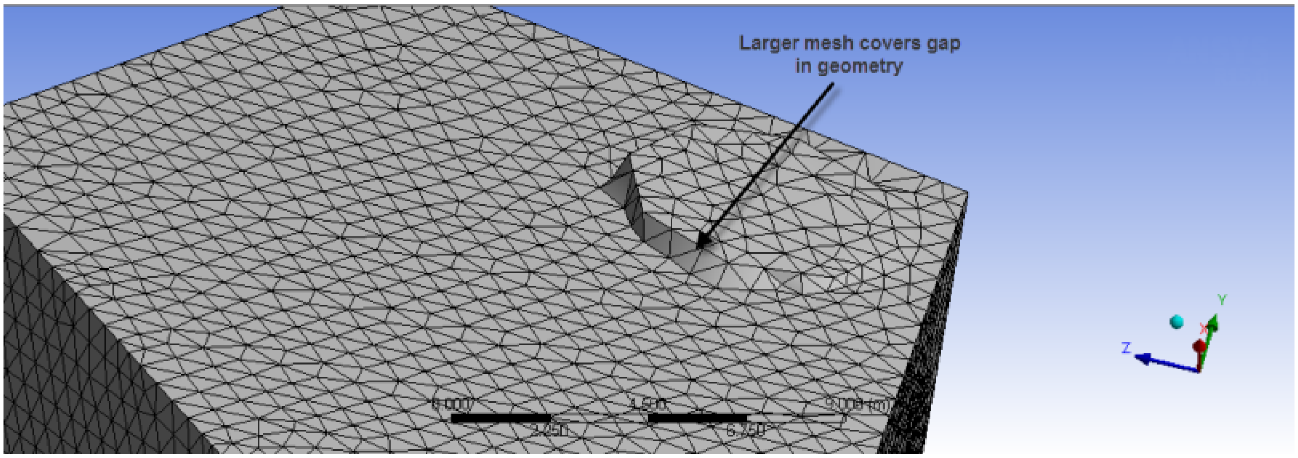


Figure 253: Patch Independent Tet Mesh Failure Corrected with Larger Mesh



Handling Patch Conforming Tetrahedral, Hex Dominant, Quad Dominant, and All Triangle Meshing Failures

Some mesh failures are due to an inappropriate defeature size (either default or user input) or dirty geometry. Use the following guidelines to determine which issue is the cause of the failure:

With Sizing Options (p. 100) turned off:

1. Determine whether the model is a multiple part assembly, a multibody part, or a single body part.

2. If a message provides "Problematic Geometry" information, use it to determine which portions of the model fail. Quite often, one or more faces fail to mesh.

3. If a face fails to mesh, check whether the face is "regular"; that is, make sure that it is not too skinny or has skinny sections with misaligned edge spacings which would make it difficult to get a good mesh. Use of virtual topologies (p. 501), pinch (p. 182) controls, etc. may help in these situations.

4. If a face fails to mesh, check to see if the edges attached to that face may be problematic. For example:

   - Turn on the **Show Vertices** and **Close Vertices** options to see if any edge is significantly faceted (that is, it has many edge splits in comparison to mesh size), or if there are any unexpected clusters of vertices. The mesher will try to place a node on each vertex, so unnecessary vertices can lead to complications in meshing that may be avoidable. Use of virtual topologies (p. 501), pinch (p. 182) controls, and so on may help in these situations.

   - Turn on the **Edge Coloring > By Connection** option to see if the edge connectivity is unusual. In some cases, the geometry connectivity may not be as expected, and this may create problems during meshing. These problems could be fixed in the DesignModeler application, the CAD package, or possibly through the use of virtual topologies (p. 501) or pinch (p. 182) controls.

   - For a multibody part, turn on the **Edge Coloring > By Body Connection** option to see if the edge connectivity is unusual between bodies.

Handling General Sweep Meshing Failures

In the event of a sweep mesh failure, the following approaches are recommended:

1. Check for contradicting edge sizings.

2. For **Src/Trg Selection**, use a manual setting instead of automatic.

3. Check side faces to see if they are mappable. Use of virtual topologies (p. 501) can help make bodies sweepable:

   - Turn on the **Show Vertices** and **Close Vertices** options to see if any edges have unnecessary splits. Extra edge splits can make faces that appear to be mappable more difficult to map. Virtual edges (p. 502) can help in these cases.

   - Use of virtual face splits (p. 521) can help make faces more mappable, as can use of the mapped face mesh control (p. 265) and its advanced options.

4. Turn on the **Edge Coloring > By Connection** option to see if the edge connectivity is unusual. In some cases, the geometry connectivity may not be as expected, and this may create problems during meshing. These problems could be fixed in the DesignModeler application, the CAD package, or possibly through the use of virtual topologies (p. 501).

5. For a multibody part, turn on the **Edge Coloring > By Body Connection** option to see if the edge connectivity is unusual between bodies.

For detailed information about the requirements and characteristics of sweep meshing, refer to Mesh Sweeping (p. 323).

For additional information, refer to Figure : Strategies for Avoiding Stretched Elements in the Mechanical APDL help.

Handling Thin Sweep Meshing Failures

In the event of a thin sweep mesh failure, first refer to Thin Model Sweeping (p. 330) for detailed information about the requirements and characteristics of thin sweep meshing.

The **Preview Source and Target Mesh** and **Preview Surface Mesh** features do not support the thin model sweeper. Thus, if a failure occurs, you must use the feedback in the **Messages** window to determine the problem:

- If **Src/Trg Selection** is set to **Automatic Thin**, determine whether the correct source/target faces are being used. You can review the source/target faces by right-clicking the appropriate message. If incorrect source/target faces are being used, select the **Manual Thin** option on the sweep method and pick the correct faces manually.

- In many cases, the messages tell you to use virtual topology (p. 501) to merge an edge/face. Thin sweeping requires one division through the thickness. Side edges must connect directly from source to target to comply with this rule.

- If a message tells you the target faces are not meshed, try swapping the source/target faces.

- In cases where a thin sweep body and a general sweep body are neighbors, the general sweep body has higher priority and is meshed first. The general sweep operation may place nodes on the side area/edge of the thin sweep body. If you receive a message describing this situation, you must apply additional mesh controls to prevent it.

- If both source and target areas are meshed, thin sweep will fail. In this case, you must find some way to eliminate the situation.

- Thin sweep may issue warning messages telling you that the source you picked was swapped with the target, and that some controls on the target faces were ignored as a result. These types of warning messages are for your reference only.

Handling General MultiZone Meshing Failures

In the event of a **MultiZone** mesh failure, the following approaches are recommended:

1. If using automatic source face selection, try using manual source face selection (or vice versa). For manual source face selection, ensure that all sources and targets are selected. Refer to Using MultiZone (p. 347) for more information.

2. Ensure that all side faces are mappable. Refer to MultiZone Face Mappability Guidelines (p. 356) for more information. Use of virtual topologies can help make bodies sweepable:

   - Turn on the **Show Vertices** and **Close Vertices** options to see if any edges have unnecessary splits. Extra edge splits can make faces that appear to be mappable more difficult to map. Virtual edges (p. 502) can help in these cases.

   - Use of virtual face splits (p. 521) can help make faces more mappable, as can use of the mapped face mesh control (p. 265).

3. If MultiZone doesn't respect edge biasing, as shown in Figure 254: Edge Biasing Not Respected by MultiZone (p. 547) below, it may be because the opposite edge is split. To work around this, perform the edge biasing on the opposite edges to get a better edge distribution, as shown in Figure 255: Edge Biasing Respected by MultiZone (p. 547).

Figure 254: Edge Biasing Not Respected by MultiZone

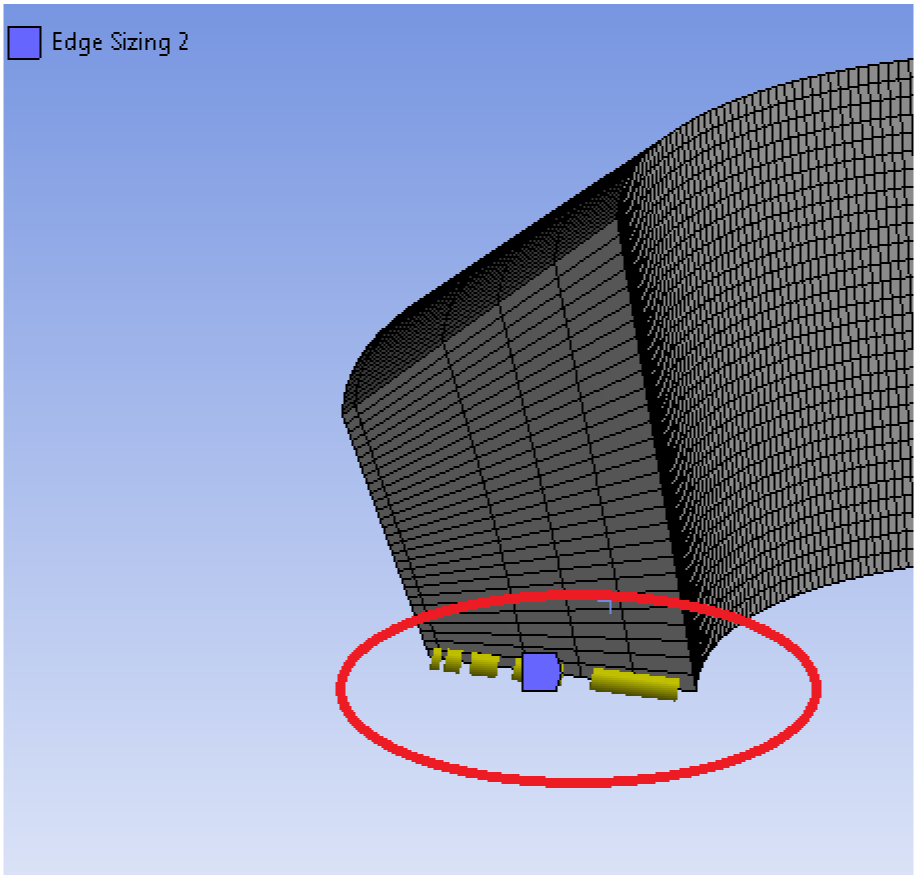
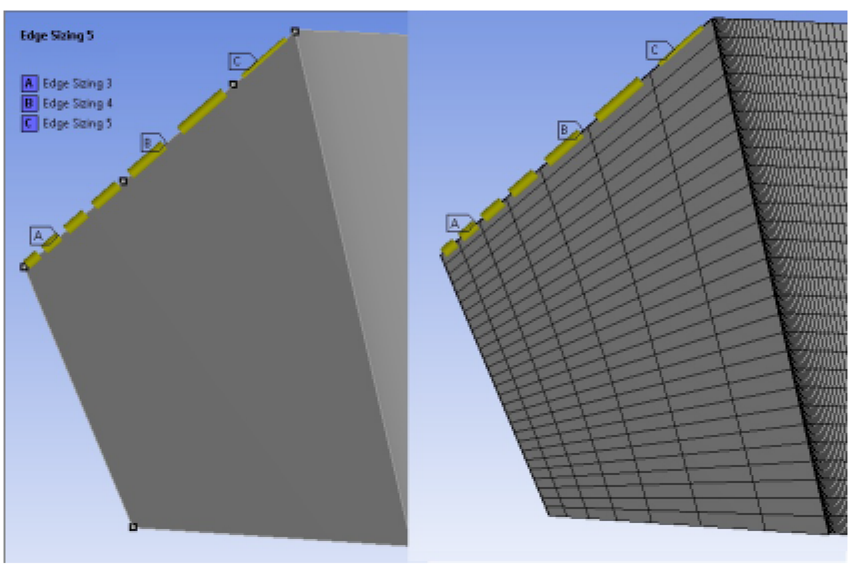


Figure 255: Edge Biasing Respected by MultiZone



Note:

When a curve with bigeometric distribution is split, the curve is split into GEO1 and GEO2 starting at the split point.

---

4. Turn on the **Edge Coloring > By Connection** option to see if the edge connectivity is unusual. In some cases, the geometry connectivity may not be as expected, and this may create problems during meshing. These problems could be fixed in the DesignModeler application, the CAD package, or possibly through the use of virtual topologies (p. 501).

5. For a multibody part, turn on the **Edge Coloring > By Body Connection** option to see if the edge connectivity is unusual between bodies.

For detailed information about the requirements and characteristics of **MultiZone**, refer to MultiZone Meshing (p. 343).

Handling Failed Mesh Connections

In the event of a mesh connection failure, refer to Diagnosing Failed Mesh Connections (p. 452).

Handling Failed Contact Matches

In the event of a contact match failure, refer to Troubleshooting Failed Contact Matches (p. 464).

Avoiding Bad Feature Capturing in Assembly Meshing

In some cases, you may encounter bad feature capturing when using assembly (p. 367) meshing in the Meshing application, even though the faceting of the same model looks fine in Ansys Fluent. The bad faceting may be apparent in Ansys Fluent only if you turn off the viewing of edges and view surfaces only. The following approaches are recommended to avoid bad feature capture:

- If you are experiencing bad feature capture, increasing facet quality can help. If you are using the DesignModeler application, increasing the value of the **Facet Quality** option (**Tools** > **Options** > **DesignModeler** > **Graphics** > **Facet Quality**) from 5 (the default) to 10 can lead to significant improvements. This **Facet Quality** option affects models entering the Meshing application via DesignModeler only; CAD packages have their own separate faceting controls. In particular, you should increase facet quality in DesignModeler or your CAD package if your model contains high order NURBS surfaces and/or your applications require the highest fidelity to your input CAD (such as aerospace applications, external aero applications, etc.). However, be aware that higher settings create large numbers of facets, which may slow down processing or possibly lead to failures in facet generation due to insufficient memory. Refer to **Facet Quality** for details about this setting in DesignModeler.

- In cases of tolerant models, the Tessellation Refinement (p. 166) control tries to avoid re-projections to the underlying curves and surfaces at locations where the CAD tolerance exceeds the desired refinement tolerance. In cases of significant tolerances, you may want to relax the **Tessellation Refinement** tolerance or even set it to **None** to avoid re-projections that may contribute to problems with faceting. In cases of accurate models, the default settings typically work well.

- The Sharp Angle Tool (p. 296) can be used to control the capture of features with sharp angles, as well as to improve feature capture in general.

Handling Assembly Meshing Failures Due to Min Size

Failure in the assembly (p. 367) meshing algorithms is almost always related to faceting issues in relation to minimum size. Make sure that the values of the Curvature Min Size (p. 108) and Proximity Min Size (p. 110) options truly represent the smallest sizes that you want the curvature and proximity size functions to capture. By default, many meshing features operate based on the smaller of these two minimum size values. Consider the following:

1. It is strongly recommended that you ALWAYS adjust the value of **Curvature Min Size**/**Proximity Min Size** as appropriate for your problem. Make sure that the minimum size is 1/2 of any small feature or gap that you need to capture. Similarly, the minimum size should be about 1/10 of the diameter of the smallest pipe. For very simple cases, make sure to increase the minimum size appropriately. Failure to do so may result in an over-refined mesh with a huge number of facets.

2. Use local (scoped) size controls (p. 252) to ensure two layers of elements in any gap/thickness. If the scoped sizing is smaller than the minimum size, you must adjust the **Tessellation Refinement** (p. 166) accordingly. If you add a hard size that is smaller than the minimum size, make sure that the tolerance specified by the **Tessellation Refinement** control is now 10 times smaller than the specified hard size.

3. If you receive a warning about missing tessellations, it may help to lower the tessellation tolerance by 50%.

4. In some cases, small defects in the faceting may lead to bad quality meshes. In many of these cases, a minor modification of the minimum size or tessellation tolerance can rectify the problem.

Handling Assembly Meshing Failures Due to Flow Volume Leaks

Virtual bodies are used with assembly meshing to represent flow volumes in a model so that you can mesh flow regions without having to model geometry to represent them. These flow volumes are extracted during meshing; however, an extraction failure may occur if there are gaps between bodies and/or faces such that the extracted flow volume would not be watertight and therefore would leak. If a leak is present, the flow volume mesh will contain only elements from the leak path (that is, surface and line elements will be returned but volume elements will not). As a result, assembly mesh generation will be unsuccessful and an error message will be issued. In such cases, the assembly meshing algorithms detect and trace the leaks and display their leak paths graphically as follows:

- Any time there is an exterior leak (that is, the mesh leaks to the outside of the model), it will be detected by the Meshing application. However, the leak path will be displayed only if you have defined a virtual body (p. 379) and specified a material point in the flow volume void. For exterior leaks, tracing occurs from the bounding box to all defined material points.

  When a material point is inside a virtual body, you must also set **Used By Fluid Surface** to **Yes** when defining the virtual body, and then define the appropriate set of fluid surfaces by selecting the faces of the virtual body and assigning them to the group.

- Interior leaks involve leak tracing between material points. Thus, for an interior leak to be identified, you must define two or more virtual bodies and specify a material point in each flow volume void.

If a leak is detected, the leak path should clearly indicate its location, in which case you should return the model to the DesignModeler application or your CAD system to close the gap that is causing the leak (for example, by adding a face or moving a body).

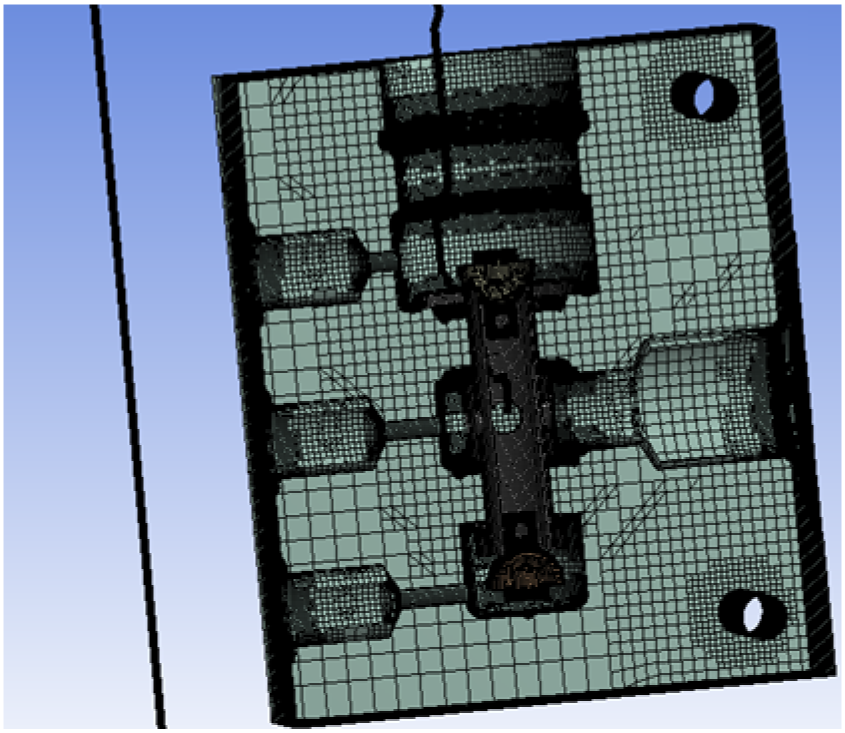
Note:

- If you suppress a virtual body, any leak path associated with it will be hidden from view in the **Geometry** window.

- In some assembly meshing cases, contact sizing can also be used for closing leaks discovered during meshing. Refer to Applying Contact Sizing (p. 398) for details.

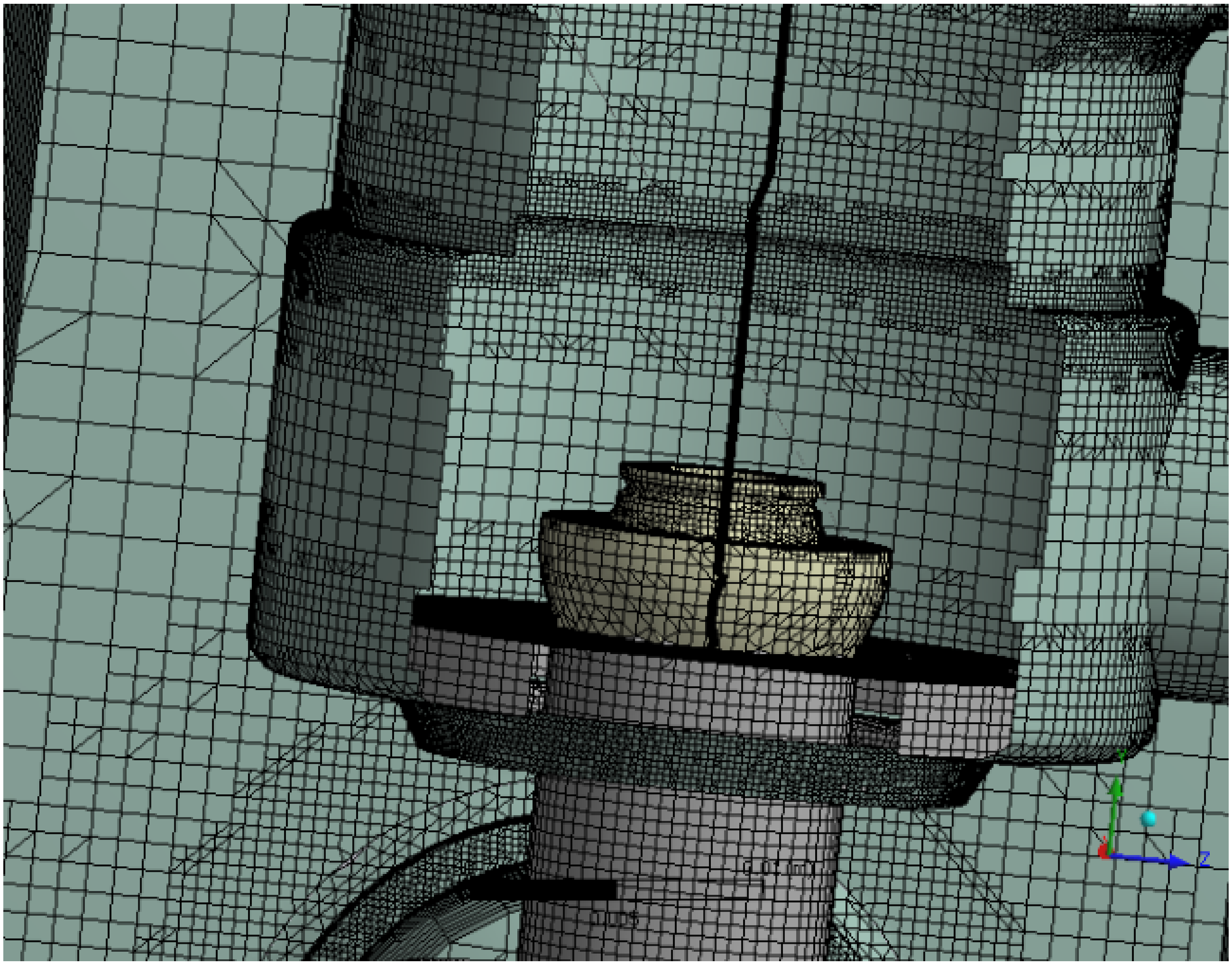
The figure below shows the leak path for a failed assembly mesh.

Figure 256: Leak Path for a Failed Assembly Mesh



The figure below shows a closer view of the leak.

Figure 257: Closer View of Leak Path



Handling Assembly Meshing Inflation Problems

If a high aspect ratio is obtained for cells in the inflation layer when an assembly meshing algorithm (p. 367) is being used, reducing the value of Gap Factor (p. 161) may help. Refer to the discussion of inflation controls in Selecting an Assembly Mesh Method (p. 375) for more information about specifying **Gap Factor** for assembly meshing algorithms.

The CutCell algorithm does not support very thick inflation layers, so instead of using an Inflation Option (p. 150) of **First Layer Thickness** or **Total Thickness**, use aspect ratio-based inflation.

Tips for Using Virtual Topology

Virtual topology (p. 501) surfaces made up of two loops are not automatically mappable. For a faceted surface made up of two loops to be map meshed, a mapped **Face Meshing** control must be scoped to it or it must be a side area of a general sweep body.

Meshing Complicated Models

Meshing a complicated model may require special attention and experimentation. In such cases, the following strategies and guidelines are recommended for obtaining a successful mesh:

1. Analyze the model to determine its complexity:
    - Identify the small features that you do (and do not) want to retain.
    - Consider the model's size and its relationship to the element size transitions that are appropriate for the mesh. A smoother transition from the fine element size to the coarse element size will result in a larger number of elements, which should be considered (especially if the model is quite large). Coarser transitions will result in a smaller number of elements. You must determine what is acceptable.
    - Refer to the value in the **Minimum Edge Length** (p. 117) field. This field provides a read-only indication of the minimum edge length in the model.
    - Think about the element size that you expect to obtain, especially the desired *minimum* element size. To help you determine the desired size, in the **Geometry** window, select the edges of small features that you want to retain and refer to the status bar for feedback about the selections.
2. Perform a low-effort mesh evaluation by using appropriate sizes as determined from #1, but without controls such as inflation, match mesh controls, etc. that add constraints to the mesher. Also, try to start with a coarser mesh size and refinement in later steps.
    - If the mesh is successful, examine it to see whether the mesh size and transition rates are acceptable. In most cases, you will need to make some adjustments to obtain the desired results.
    - If the mesh fails, examine any messages that the mesher returned to the **Messages** window, as described elsewhere in this Troubleshooting (p. 535) section.
3. Adjust settings to retain desired small features:
    - In many cases, small features are either small holes or channels in the model and are associated with high curvature. For this reason, using the **Curvature-based sizing** (p. 102) is a good strategy for retaining these features.
    - Be careful when using **Proximity-based sizing** (p. 102). If the value of **Minimum Edge Length** (p. 117) is too small, using Proximity-based sizing may lead to meshing problems.
4. Adjust settings to defeature (remove) unimportant small features:
    - The Meshing application automatically defeatures small features according to the specified Defeature Size (p. 106). Refer to the **Minimum Edge Length** (p. 117) value to help determine which small features will be defeatured automatically.
    - For solid models, **Defeature Size** is set to 50% of the value of **Curvature Min Size** (p. 108) by default. If you set a larger **Defeature Size**, you must also set a larger **Curvature Min Size** because the defeature size cannot be as large as the minimum element size.
5. Adjust the mesh settings to achieve the desired quality.

Continue making adjustments until your results are satisfactory. Try adjusting controls such as face sizing (p. 252), edge sizing (p. 252), transition rate (p. 107), and smoothing (p. 123). You may also want to experiment with virtual topology (p. 501).

Using a Localized Operating System on Linux

If you are using a localized operating system (such as French or German) on Linux, you must perform additional steps in order for the Meshing application to recognize the correct numerical format. Refer to the platform details section of the Ansys, Inc. Linux Installation Guide for details.

Using Lustre Parallel File Systems on Linux

Meshing application projects created prior to Release 16.0 need to be updated before they can be used on Lustre parallel file systems on Linux. To do this:

1. Load the project into Release 16 software on a system that does not use a Lustre parallel file system.
2. Perform an operation that changes each model in the Meshing application (for example, hide and then show a part). If systems share the same model, the change needs to be done for only one of the systems.
3. Save the project.

Index

Symbols

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P