



# Lecture-08

## Assemblies

---

In this section, you use the vanity cabinet parts described and built in **Chapter 2 "Parts"** to build subassemblies, such as the spigot and the faucet handles. Then you bring the subassemblies together to create an assembly, the vanity.



This chapter includes the following topics:

- **Assembly Definition**
- **Assembly Design Methods**
- **Prepare an Assembly**
- **Mates**
- **In-Context Design**
- **Load an Assembly**
- **Examine the Assembly**

## Assembly Definition

An assembly is a collection of related parts saved in one SOLIDWORKS document file with a `.sldasm` extension.

Assemblies:

- Contain anywhere from two to over one thousand components, which can be parts or other assemblies called subassemblies
- Display movement between related parts within their degrees of freedom

The components in an assembly are defined in relation to each other using assembly mates. You attach the assembly components using various types of mates such as coincident, concentric, and distance mates. For example, the faucet handle components are mated to the faucet base component using concentric and coincident mates. The mated components create the spigot subassembly. Later, you include this subassembly in the main vanity assembly, mating it to the other components in the vanity assembly.

## Assembly Design Methods

You create assemblies using two basic methods: bottom-up design and top-down design.

You can also use a combination of the two methods. With either method, your objective is to mate the components to create the assembly or subassembly (see [Mates](#) on page 57).

### Bottom-up Design

In bottom-up design, you create parts, insert them into an assembly, and mate them as required by your design. Bottom-up design is the preferred technique when you use previously constructed, off-the-shelf parts.

An advantage of bottom-up design is that because components are designed independently, their relationships and regeneration behavior are simpler than in top-down design. Working bottom-up allows you to focus on the individual parts. It is a good method to use if you do not need to create references that control the size or shape of parts with respect to each other.

Most of the vanity cabinet uses bottom-up design. You create the components such as the sink and the spigot in their own part windows. Then you open an assembly document, bring the components into the assembly, and add various mates.

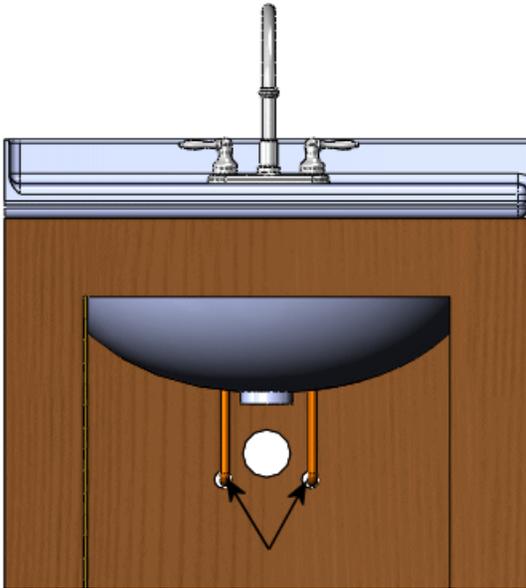
### Top-down Design

In top-down design, you start your work in the assembly. You can use the geometry of one part to help define other parts, to create features that affect multiple parts, or to create machined features that are added only after the parts are assembled. For instance, you can start with a layout sketch or define fixed part locations, then design the parts referencing these definitions.

Top-down design is also known as in-context design.

For example, you can insert a part in an assembly, then build a fixture based on this part. Working top-down, creating the fixture in context, allows you to reference model geometry, so you can control the dimensions of the fixture by creating geometric relations to the original part. That way, if you change a dimension of the part, the fixture is updated automatically.

The vanity cabinet also uses top-down design. You create the two supply pipes within the context of the assembly. Then you reference the size and location of the faucet subassembly and the vanity cabinet to define the supply pipes.



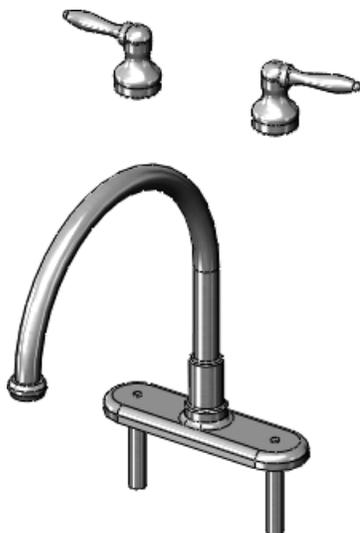
Supply pipes

## Prepare an Assembly

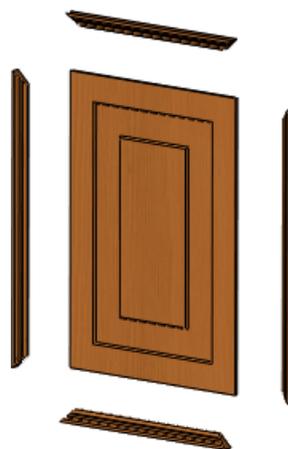
Before you build an assembly, you need to prepare the assembly components.

Throughout this section, you use the parts for the vanity cabinet created in [Parts](#) on page 37. The vanity includes the following subassemblies:

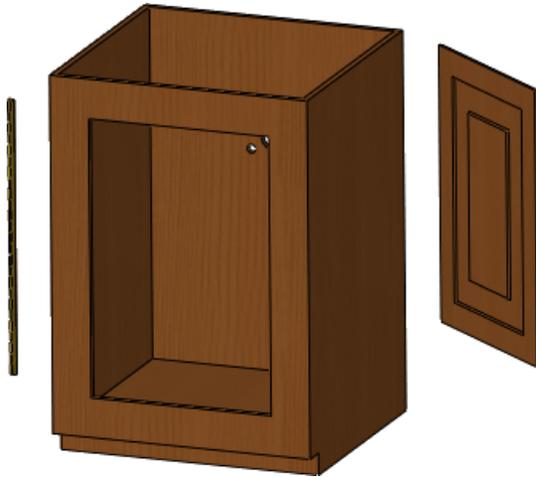
- Faucet and faucet handles
- Door and moldings
- Door subassembly, cabinet, and hinge



Faucet and faucet handles



Door and moldings



Door subassembly, cabinet, and hinge

For each subassembly document, you do the following prior to mating the components:

- Load and anchor the first component to the assembly origin
- Load the additional components
- Move and position the components

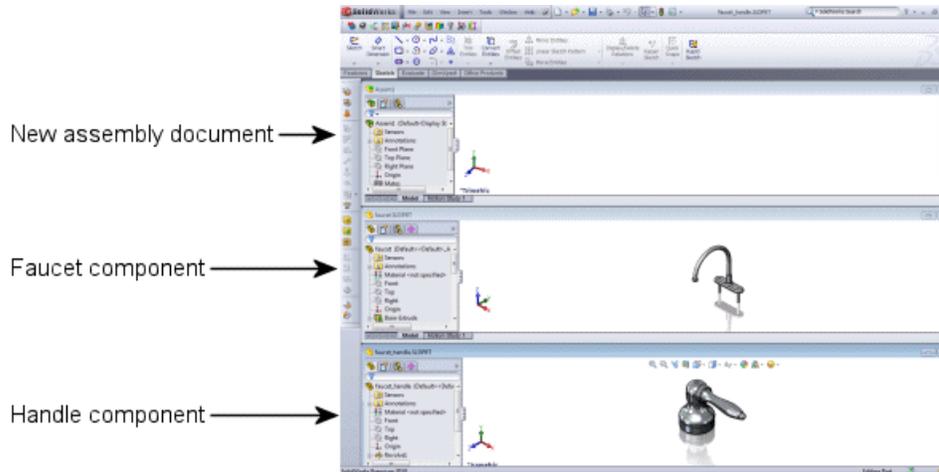
## Mates

Mates position the components in an assembly precisely with respect to each other.

Positioning the components defines how they move and rotate with respect to each other. Mates create geometric relations, such as coincident, perpendicular, and tangent. Each mate is valid for specific combinations of geometry such as cones, cylinders, planes, and extrusions. For example, if you mate a cone to another cone, the valid types of mates you can use include coincident, concentric, and distance (see [Coincident Mate](#) on page 60).

## Faucet Subassembly

Depending on the complexity of the assembly (the number of separate components), you can open one or all of your components. In the faucet example, there are only two components (the faucet and the handle), so you can tile the two documents. After you open the components, you need to open a new assembly document into which you bring the components.



You can add more than one instance of the same part to an assembly. You do not have to create a unique part for each component in the assembly.

You want to place the bottom of the handle component on the flat base of the faucet component, so the handle sits on the faucet. You also need to center the handle components over the faucet stems to position them correctly. To position the components, you apply a coincident mate and a concentric mate.

## Load the First Assembly Component

When creating an assembly, start with the component that does not move with respect to the other components. This is the component you anchor or fix to the assembly origin. In the example of the faucet subassembly, you anchor the faucet component.



Anchoring the first component ensures that the planes in both documents are aligned.

Bring the first component into the `.sldasm` document as follows:

- Select the component name in the FeatureManager design tree of the `.sldprt` document and drag it into the `.sldasm` document.
- To position the first component on the origin of the `.sldasm` document, drop it on the origin in the graphics area or anywhere in the FeatureManager design tree. Dropping it in the FeatureManager design tree requires less fine mouse movement and will automatically align the part's origin and the assembly origin.



As you bring each component into the `.sldasm` document, the component appears in the FeatureManager design tree.

## Load the Additional Components

You load the other components of the assembly by selecting the component in the FeatureManager design tree of the `.sldprt` document, and dragging the component into the graphics area of the `.sldasm` document. In the example of the faucet subassembly, you drag in two instances of the handle.

The first component you add to an assembly is fixed in space by default, which is useful for mating of components. It is common to choose a component that you wish to be fixed; however, you can change which component is fixed later.



Faucet component with origin (assembly origin and component origin)



First handle component added



Second handle component added

## Position the Additional Components

When you bring the additional components into the assembly, you can position them anywhere in the graphics area. Then you can use the left mouse button to drag a component closer to the first, anchored component. You can use the right mouse button to rotate a component into the proper orientation.

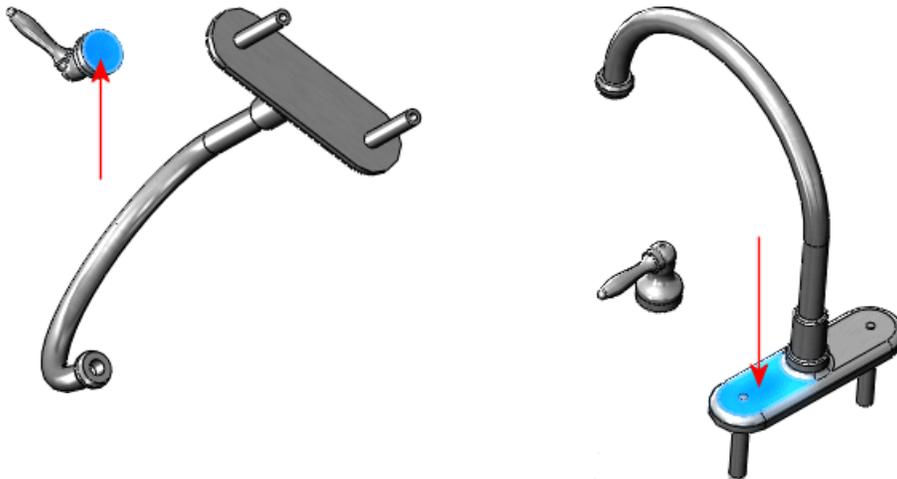
Leave some space between components to view the relevant component areas. You can use the following methods to change the orientation of the components:

- Middle mouse button: Rotates all components.
- Middle mouse button with **Ctrl**: Pans all components.
- Middle mouse wheel: Zooms all components in or out.

These mouse functions help select edges, faces, or other entities needed to apply mates.

## Coincident Mate

To create a coincident mate between the handle component and the faucet component, attach the flat bottom face of the handles to the flat top face of the faucet.



Flat bottom face of the faucet handles

Flat top face of the faucet base component

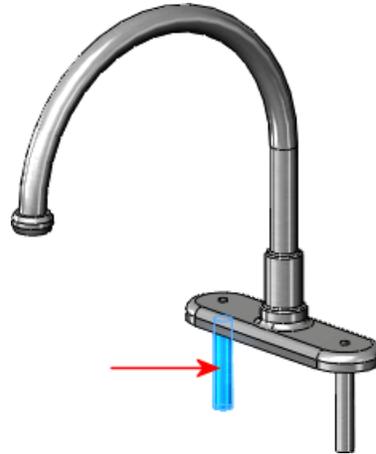
When you apply the coincident mate, the faucet handle component moves closer to the faucet component. Note that you can still slide the handle anywhere along the top face of the faucet by dragging it with the left mouse button, indicating that a second mate is required to further define the position of the two components.

## Concentric Mate

Select any round face on the faucet handle. Then select the round face of the faucet stem (the portion of the component that slides into the counter top, and connects to the supply pipe).



Round face on the faucet handle



Round face on the stem

Once you apply the concentric mate between the faucet handle component and the faucet component, you can no longer move the faucet handle along the top face of the faucet to shift its position. You can, however, use the left mouse button to drag the faucet handle on its axis.



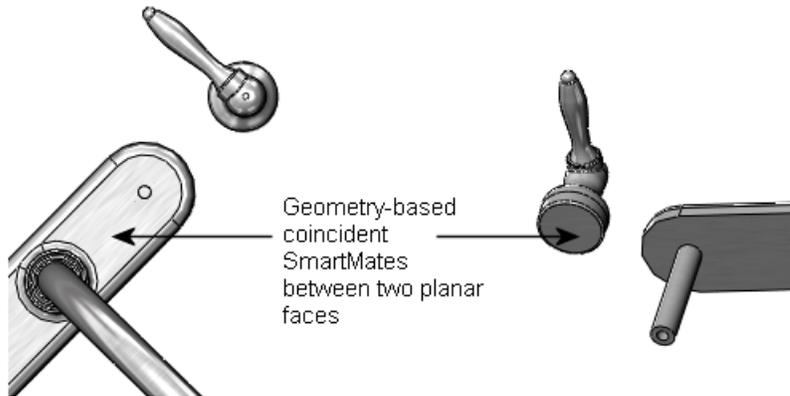
For a lesson on assembly mates, see the *Assembly Mates* tutorial.

## Faucet Subassembly - Alternate Design Approach

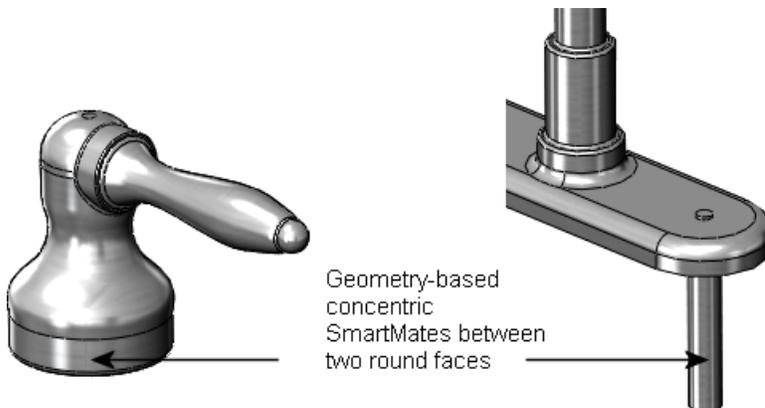
Another approach to mating the faucet and handle components is to use SmartMates. With SmartMates, the system automatically creates some mates. SmartMates are based on the entity you use to drag the component.

When you drag components into assemblies, you infer the geometry of existing components to create mates. SmartMates automatically infer mate partners and eliminate the need to use the **Mate** PropertyManager.

There are different types of SmartMates. You can use geometry-based SmartMates to create coincident mates between planar faces. For example, use SmartMates to create a coincident mate between the faucet component and each of the faucet handles in the faucet subassembly. Use **Alt** and drag the bottom face of the handle to create a coincident mate between the handle and faucet.



You can use another type of geometry-based SmartMate to create the concentric mate between the two round faces to completely define the faucet subassembly.



There are other types of SmartMates, including feature-based SmartMates and pattern-based SmartMates. For more information, see *SmartMates Overview* in the Help.

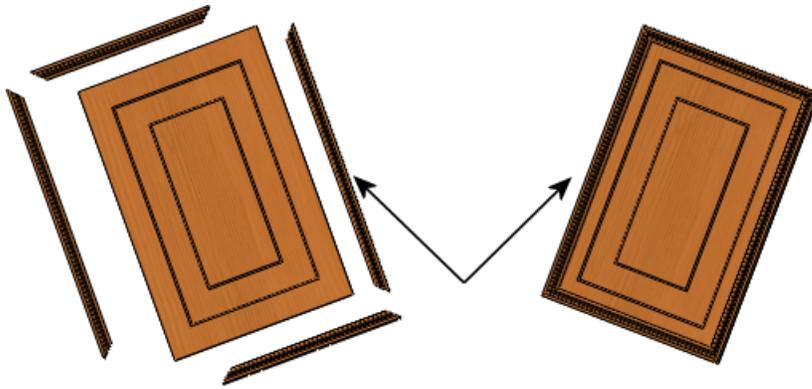
## Door Subassembly

The cabinet door uses coincident mates between the door component and the four molding components. It also uses configurations of the molding as a time-saving design step.

Configurations let you create multiple variations of a part or an assembly within a single document. Configurations provide a convenient way to develop and manage families of models with different dimensions, components, or other parameters (see [Use Configurations of a Part](#) on page 49).

As stated earlier, you can use the same part more than once in an assembly. Each instance of the part can also use a different configuration.

The door subassembly uses configurations. There are four instances of the molding component. Two of the instances use the **short** configuration and fit across the short sides of the door. The other two instances use the **long** configuration.



Door subassembly with molding components

## Cabinet Subassembly

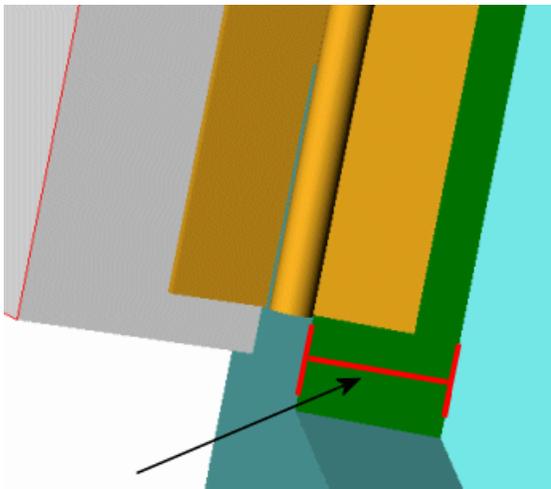
The cabinet subassembly uses concentric and coincident mates. It also uses a distance mate between the cabinet and one of the hinge components.

### Distance Mate

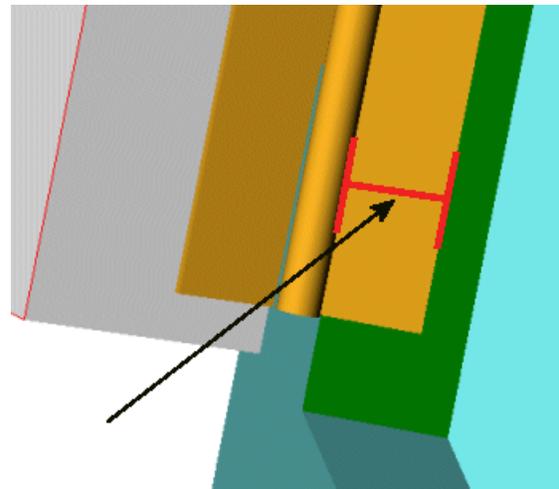
A distance mate uses a value you assign to separate the two entities.

In the vanity cabinet, the distance mate positions the hinge optimally, so that it functions freely. You determine the correct mate distance using the **Measure** tool.

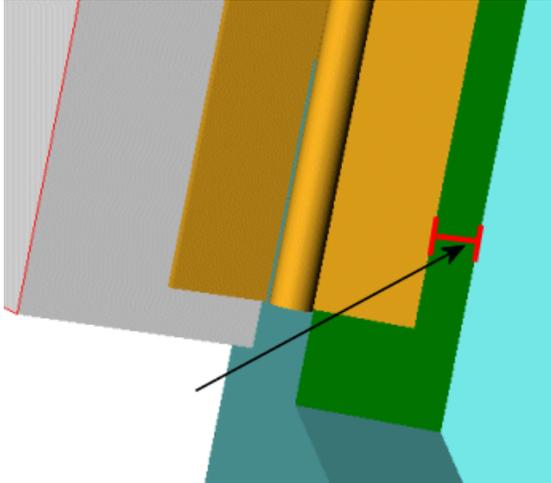
By measuring the entities of different components, you can determine at what position to place the hinge so that it does not bind when you open the cabinet door. Once you know the thickness of the door opening and the width of the hinge, you can position the hinge using a distance mate.



Measure the width of the inside for the cabinet door opening.



Measure the width of the hinge that you attach to the inside of the cabinet door opening.



Apply a distance mate, based on measurements of the cabinet and of the hinge.

## In-Context Design

You can create a new part within an assembly document (in the context of an assembly).

In addition to creating or editing components in their own part windows, the SOLIDWORKS software lets you create or edit components in the assembly window. The advantage is that you can reference the geometry of one component to create or modify another component. By referencing the geometry of another component, you ensure that the components fit together correctly. This method of design is called top-down or in-context design because you are working in the context of the assembly.

In the vanity assembly, there are two examples of in-context design. One example is the diameter of the supply pipe component and the waste pipe component. The pipe components are both new parts that you create in the context of the assembly. The other example is the cut feature for the holes in the back of the vanity cabinet. The vanity cabinet is an existing part that you edit in the context of the assembly. These examples are discussed in the next two sections.

As you create an in-context part, the software includes notations and options with information about the relationships in the features.

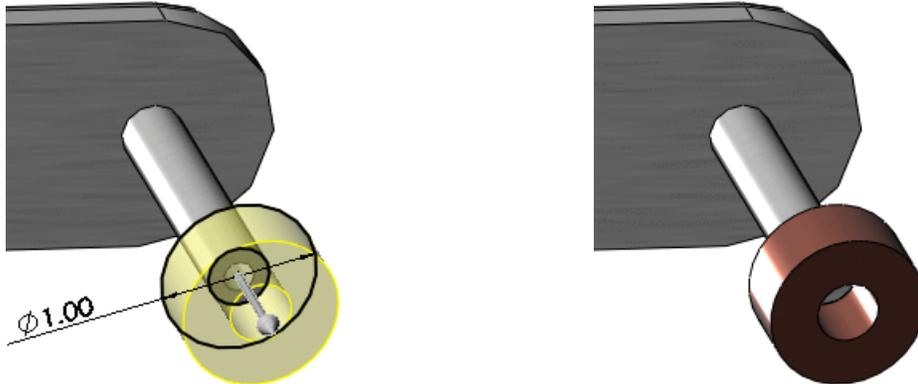


For more information on creating in-context components, see *Creating a Part in an Assembly* in the Help.

## Create an Assembly Component In-Context

The diameter of the supply pipe component depends on the diameter of the faucet stem. It is a good idea to create the supply pipe component in the assembly so you can reference the geometry of the faucet stem. You use the **Convert Entities** and **Offset Entities** sketch tools to reference the geometry of the faucet stem for a sketch in the supply pipe component. This reference ensures that the size of the supply pipe changes if you change

the size of the faucet stem. You can use the same method to create the waste pipe component, which depends on the diameter of the exit stem at the bottom of the basin.

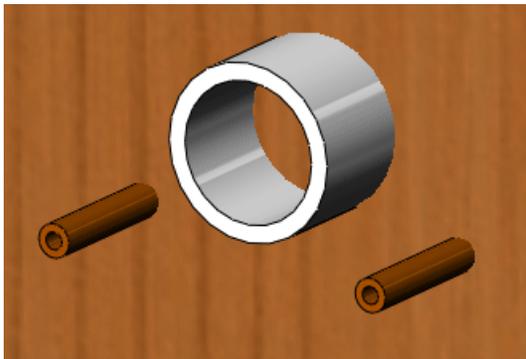


Use **Convert Entities** and **Offset Entities** to create the sleeve between the faucet stem and the supply pipe.

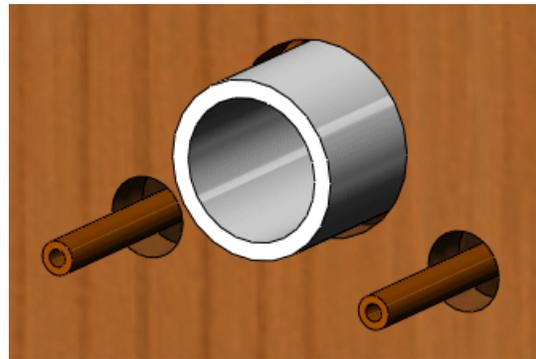
Extrude the sketch to create the sleeve between the faucet stem and the supply pipe.

## Modify a Part In-Context of an Assembly

The positions of the holes in the back of the vanity cabinet depend on the length of the supply pipe and the waste pipe components. It is a good idea to edit the vanity cabinet component in the assembly so you can reference the geometry of the supply pipes and waste pipe. You use the **Offset Entities** sketch tool to reference the geometry of the pipes for a sketch of the cut in the vanity cabinet component. This reference ensures that the position and size of the holes changes if you change the position and size of the supply pipes or waste pipe.



Supply and waste piping before in-context cut



Supply and waste piping after in-context cut

## Load an Assembly

You can improve performance of large assemblies significantly by using lightweight components.

After you create an assembly, you can load it with its active components fully resolved or lightweight.

- When a component is fully resolved, all of its model data is loaded in memory.
- When a component is lightweight, only a subset of its model data is loaded in memory. The remaining model data is loaded on an as-needed basis.

Loading an assembly with lightweight components is faster than loading the same assembly with fully resolved components.

Lightweight components are efficient because the full model data for the components is loaded only as it is needed.

Assemblies with lightweight components rebuild faster because fewer details are evaluated. However, mates on a lightweight component are solved, and you can edit existing mates.



The vanity cabinet is a relatively simple assembly, so any performance gains using lightweight components are minimal.

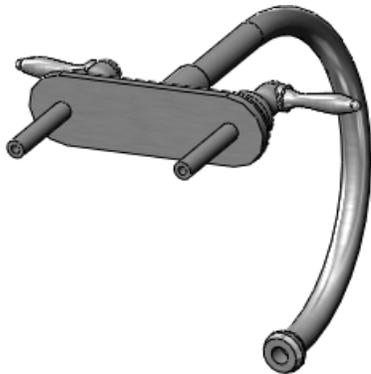
## Examine the Assembly

The SOLIDWORKS software includes various assembly tools that can display, test, and measure your assembly components once you apply the mates.

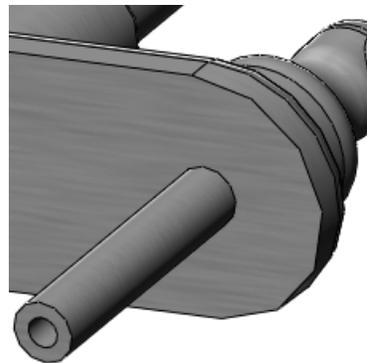
Some of the assembly tools include:

### Hide and Show Components

You can hide or show components in the graphics area. Hiding components often facilitates component selection when you add mates or when you create in-context parts. For example, to select the inner and outer diameters of the faucet stems, you can hide all components except the faucet subassembly, and then zoom, rotate, or change the view as needed.



Hide all components except the one you need



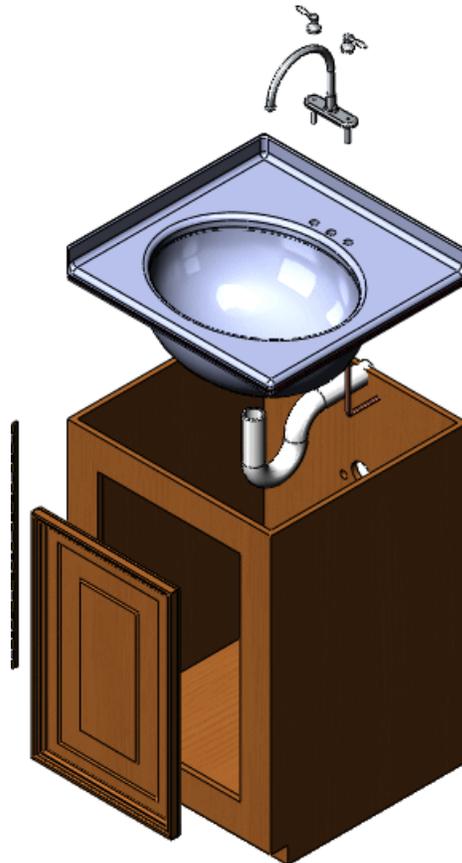
Zoom, rotate, and change the view if necessary to select the feature



**Hide Components** and **Show Components** do not affect the mates between the components. They affect only the display.

## Explode the Assembly

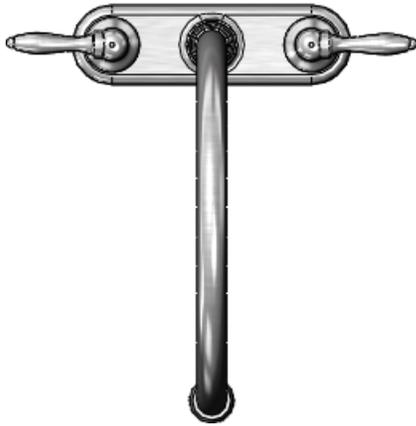
An exploded view separates the components in an assembly to facilitate viewing. Exploded views include many options such as which components to include, what distances to use, and in which direction to display the exploded components. The exploded view is saved with a configuration of the assembly or subassembly.



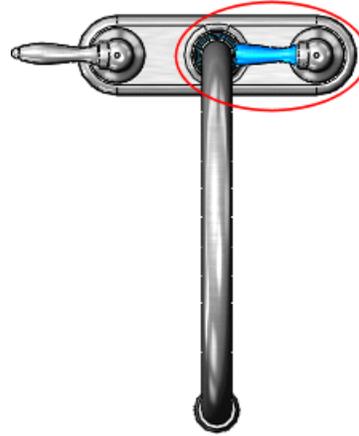
## Detect Collisions Between Components

You can detect collisions with other components when you move or rotate a component. The SOLIDWORKS software can detect collisions with the entire assembly or a selected group of components that move as a result of mates.

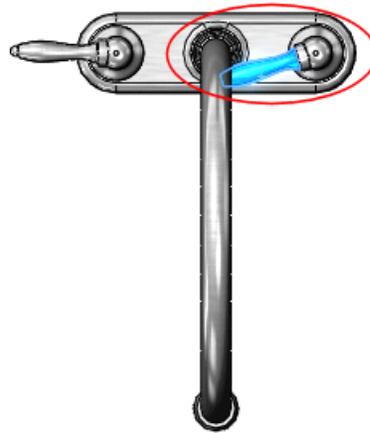
In the faucet subassembly, note how the faucet handles collide with the faucet. You can set the **Stop at collision** option to determine where the components collide.



Normal position of handles



**Collision Detection** without **Stop at collision** active. Notice the handle moves inside the faucet.



**Collision Detection** with **Stop at collision** active. Notice the handle cannot move inside the faucet.