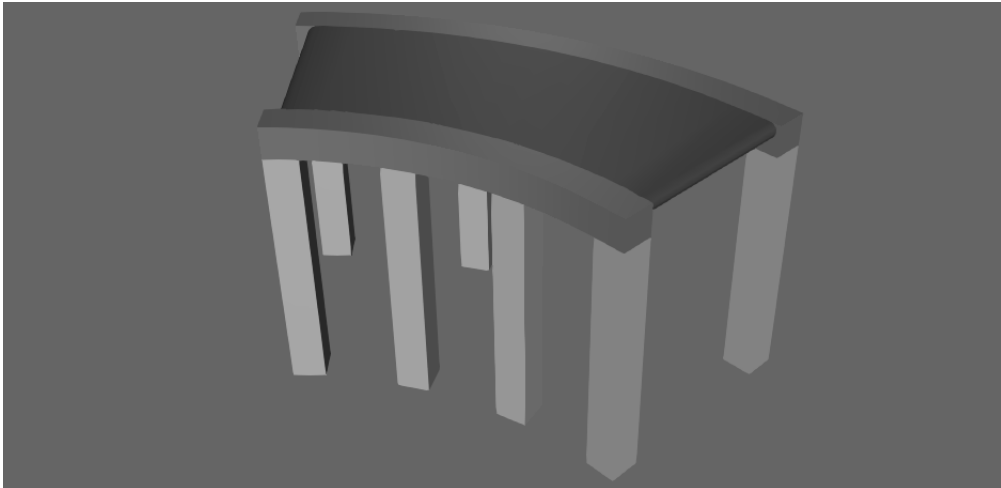


# Model a Parametric Conveyor

Visual Components 4.0 | Version: March 24, 2017



Some conveyors allow you to move components from one location to another along a curved path. In some cases, you need to make a conveyor parametric to support different dimensions and geometry, such as roll bars and belts.

In this tutorial you learn how to:

- Rebuild geometry and write expressions using component properties.
- Transform geometry and Frame features.
- Clone geometry along an axis and angle.
- Revolve geometry using a sweep profile.
- Assign materials to objects.
- Create custom features using Python API.
- Use a wizard to build the path and interfaces of a conveyor.

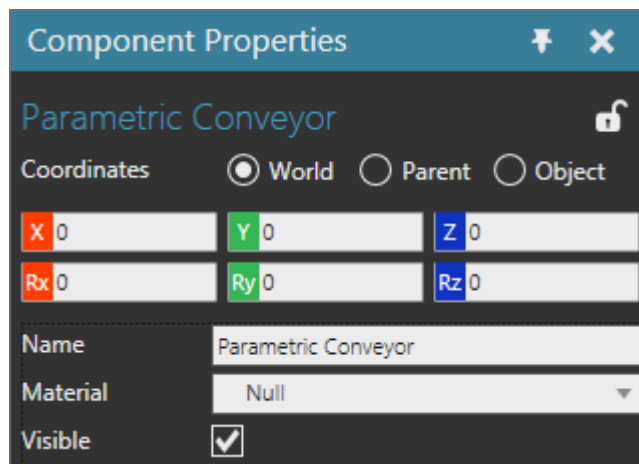
**Support**  
support@visualcomponents.com

**Community**  
community.visualcomponents.net

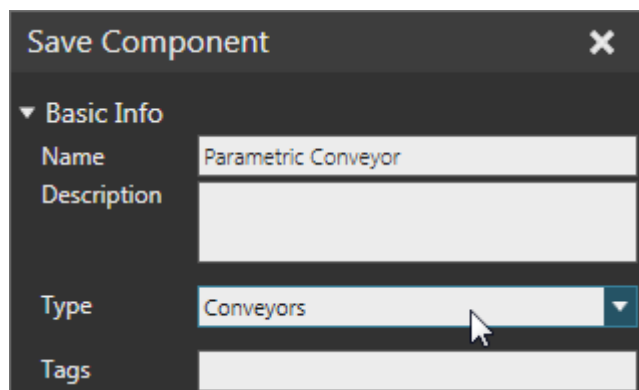
# Getting Started

This tutorial does not require you to import any geometry. Primitives and other types of features will be used to model the conveyor geometry.

1. Press CTRL+N to clear the 3D world.
2. Click the **Modeling** tab, and then in the Component group, click **New** to create a new component.
3. In the Component Properties panel, set Name to **Parametric Conveyor**.



4. On the Modeling tab, in the Component group, click **Save**.
5. In the Save Component task pane, under Basic Info, set Type to **Conveyors**.



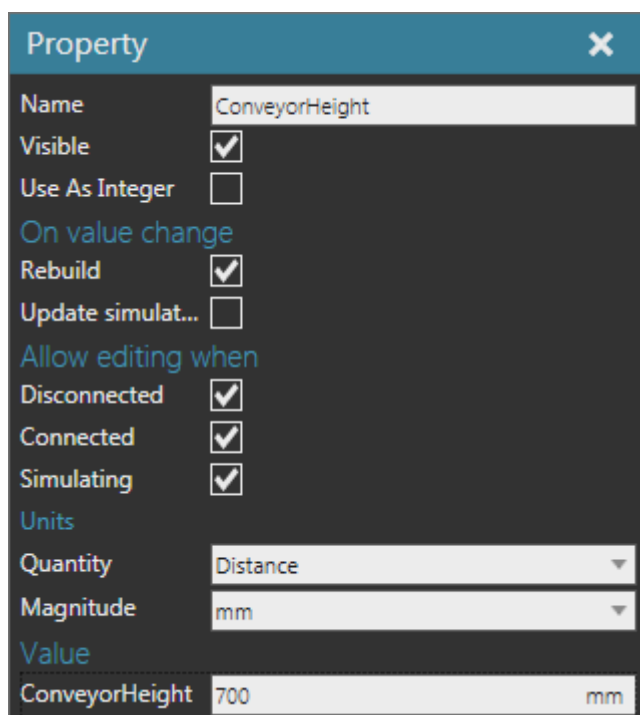
6. Now click **Save**, and then save the new file in your My Models folder.

## Create Leg

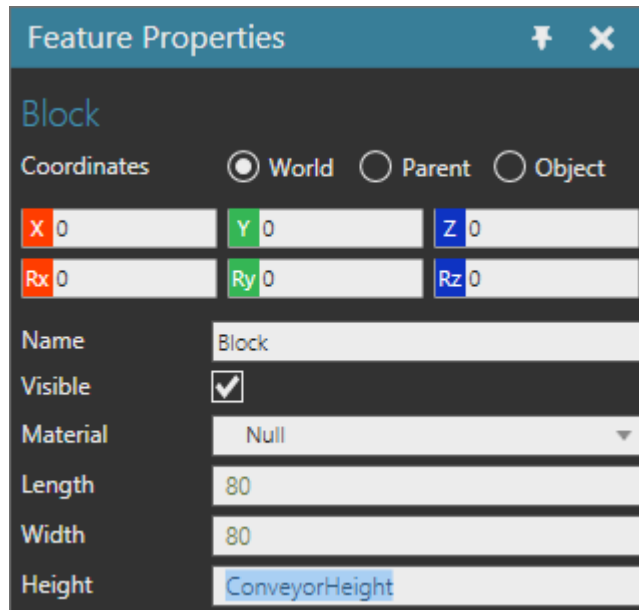


A primitive Block feature can be used to represent a conveyor leg. A component property will be used to control the height of the block, thereby the height of conveyor legs.

1. On the Modeling tab, in the Properties group, click the **Properties** arrow, and then under Basic, click **Real**.
2. In the Property task pane, set Name to **ConveyorHeight**, Quantity to **Distance** and Value to **700**.



3. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Primitive Geometry, click **Box**.
4. In the Feature Properties panel, set Length and Width to **80** and Height to **ConveyorHeight**.



The screenshot shows the 'Feature Properties' panel for a 'Block' feature. The panel has a title bar with a pin icon and a close icon. Below the title, the word 'Block' is displayed. Under 'Coordinates', there are three radio buttons: 'World' (selected), 'Parent', and 'Object'. Below these are six input fields for coordinates: X (0), Y (0), Z (0), Rx (0), Ry (0), and Rz (0). Below the coordinates are five more properties: 'Name' (Block), 'Visible' (checked), 'Material' (Null), 'Length' (80), 'Width' (80), and 'Height' (ConveyorHeight).

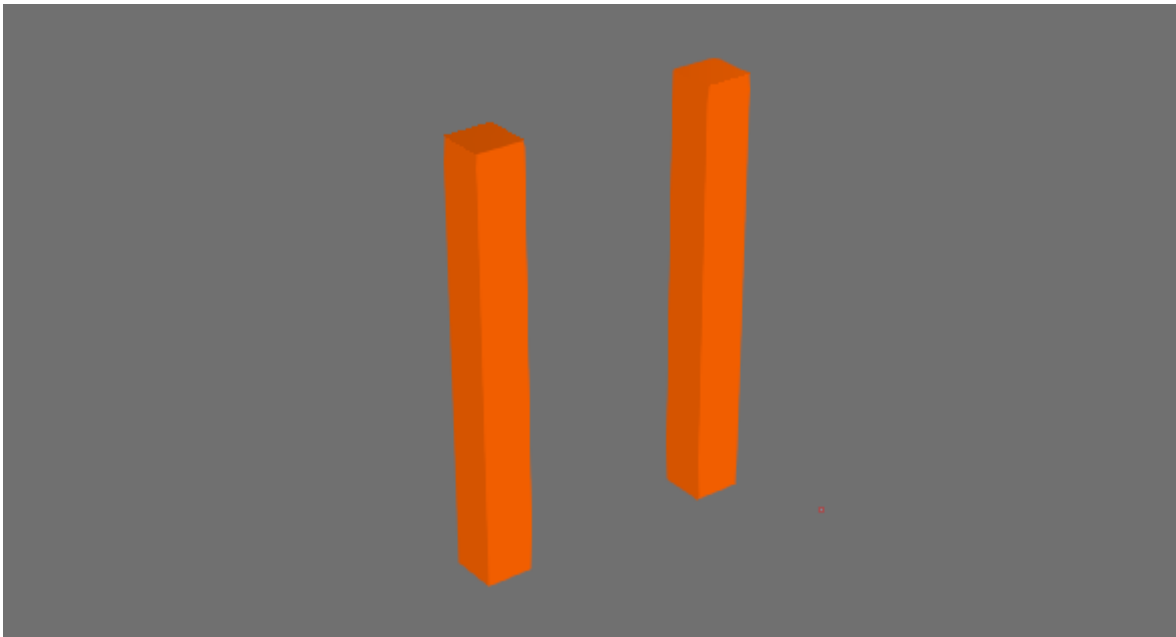
Feature Properties		
Block		
Coordinates <input checked="" type="radio"/> World <input type="radio"/> Parent <input type="radio"/> Object		
X	0	Y 0
Rx	0	Ry 0
Z	0	Rz 0
Name	Block	
Visible	<input checked="" type="checkbox"/>	
Material	Null	
Length	80	
Width	80	
Height	ConveyorHeight	

One or more properties of a feature may allow you to assign an expression. When the expression is evaluated it returns a calculated value.

If you are using component properties in an expression, the "On value change" attributes of a component property serve an important role in updating feature properties and/or rebuilding geometry.

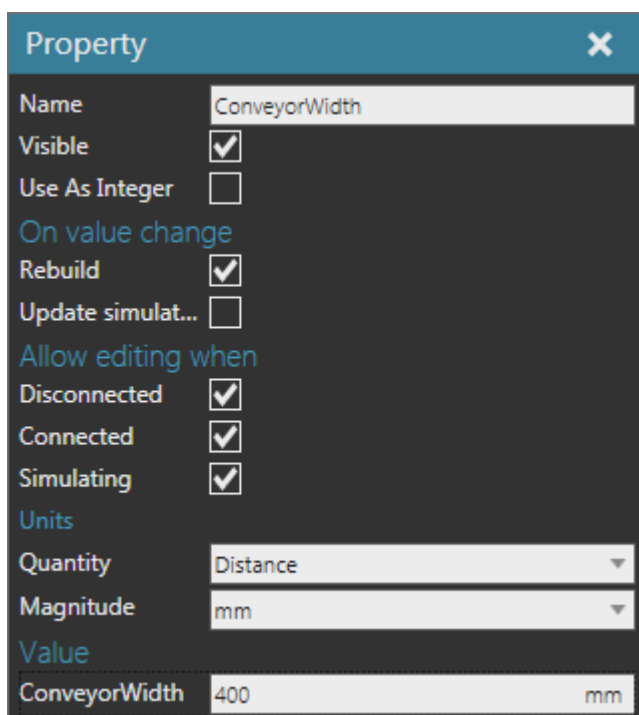
**NOTE!** Rebuilding geometry when there is a change to a property can be an intensive task during a simulation, especially in components with a lot of geometry.

## Linear Clone Leg

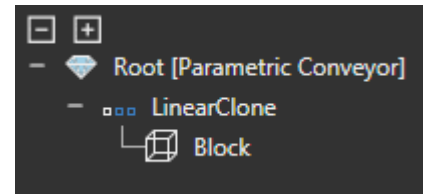
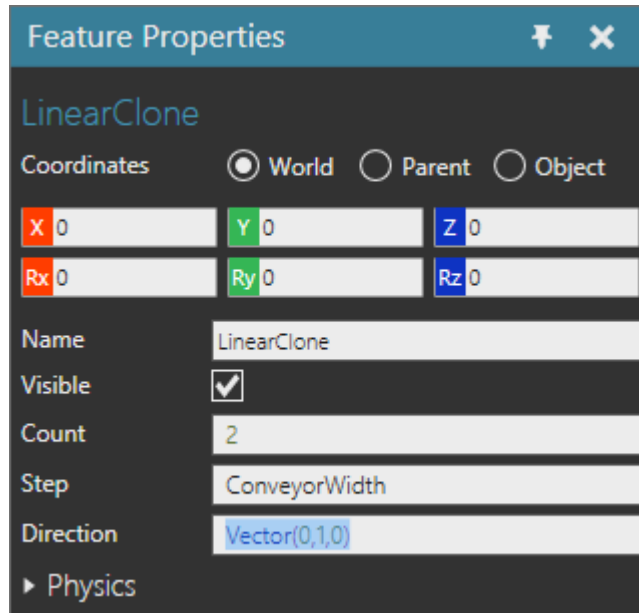


The conveyor leg can now be cloned using an operation feature. This will form the front legs of the conveyor. A component property will be used to control the step or distance between the legs, thereby the width of the conveyor.

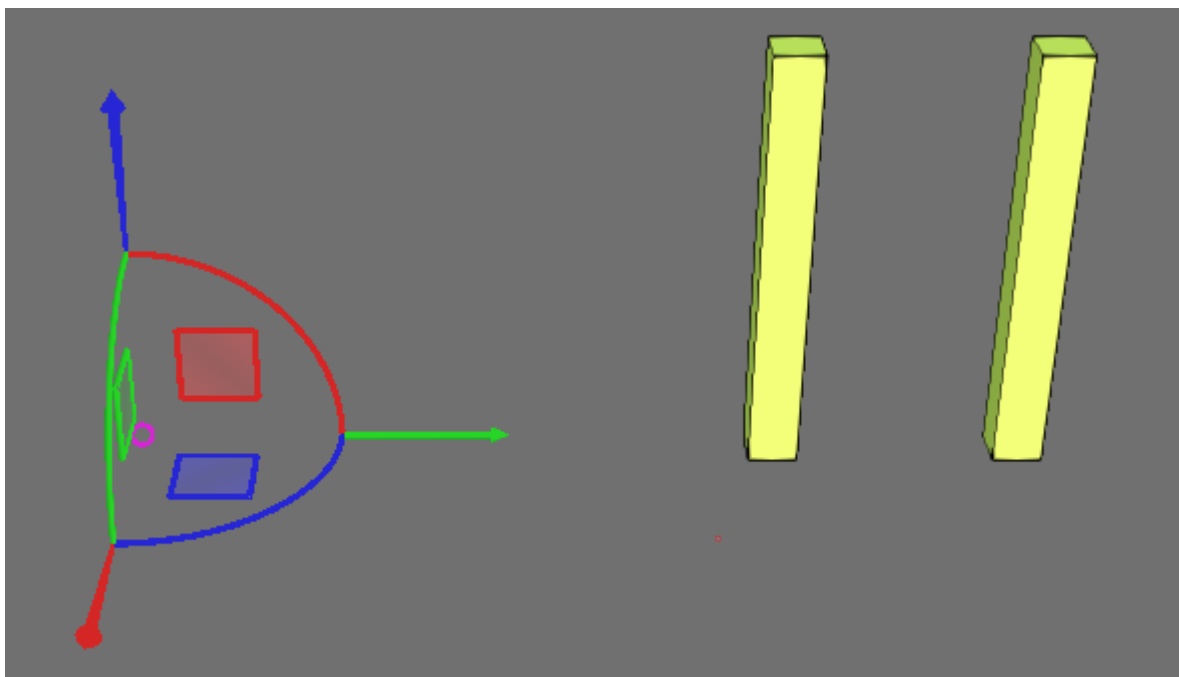
1. On the Modeling tab, in the Properties group, click the **Properties** arrow, and then under Basic, click **Real**.
2. In the Property task pane, set Name to **ConveyorWidth**, Quantity to **Distance** and Value to **400**.



3. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Clone, click **Linear Clone**.
4. In the Feature Properties panel, set Count to **2**, Step to **ConveyorWidth** and Direction to **Vector(0,1,0)** to clone in the Y-axis direction.
5. In the Component Graph panel, drag **Block** onto **LinearClone** to apply the operation and clone the conveyor leg.

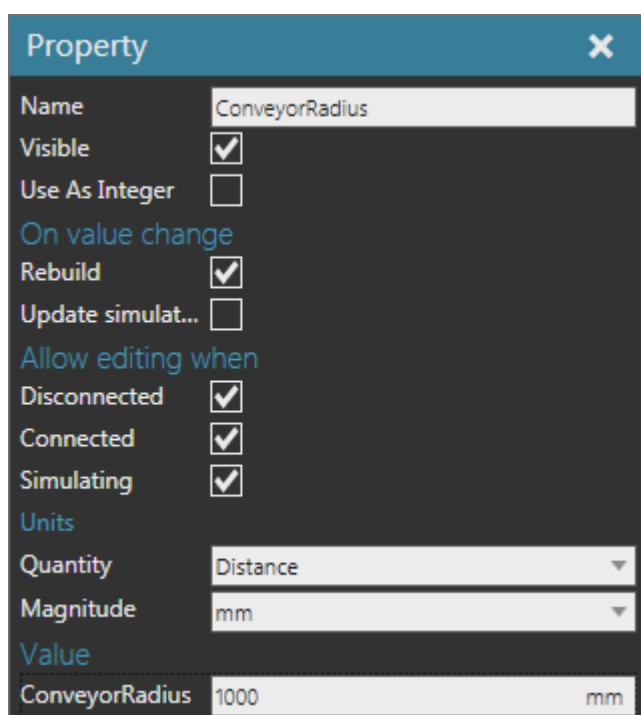


# Transform Legs

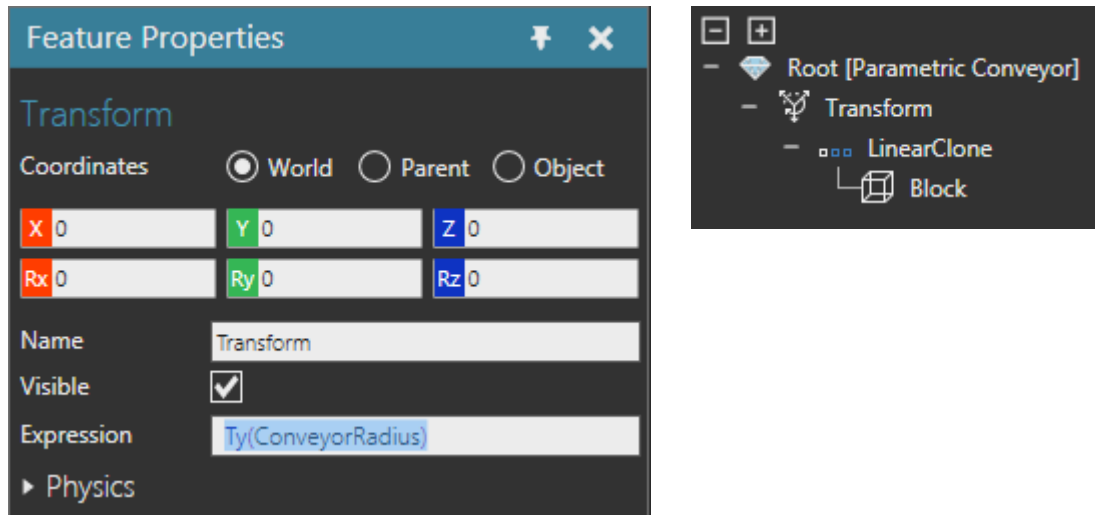


The front legs of the conveyor will now be offset from the component origin, thereby creating a radius that can be used for cloning geometry at an angle around a center point. A component property will be used to control the value of the radius, and a Transform feature will be used to offset the conveyor legs in the Y-axis.

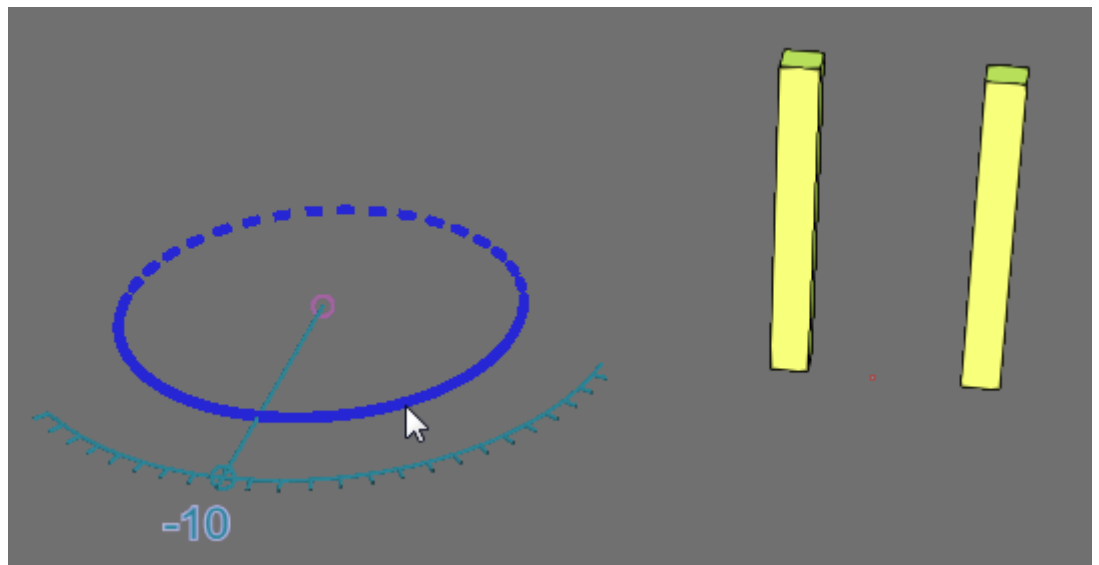
1. On the Modeling tab, in the Properties group, click the **Properties** arrow, and then under Basic, click **Real**.
2. In the Property task pane, set Name to **ConveyorRadius**, Quantity to **Distance** and Value to **1000**.



3. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Transform, click **Transform**.
4. In the Feature Properties panel, set Expression to "Ty(ConveyorRadius)".
5. In the Component Graph panel, drag **LinearClone** and its tree onto **Transform** to apply the operation and translate the conveyor legs along the Y-axis.



6. Select the **Transform** feature, and then use the **Move** tool to rotate the feature around the Z-axis.

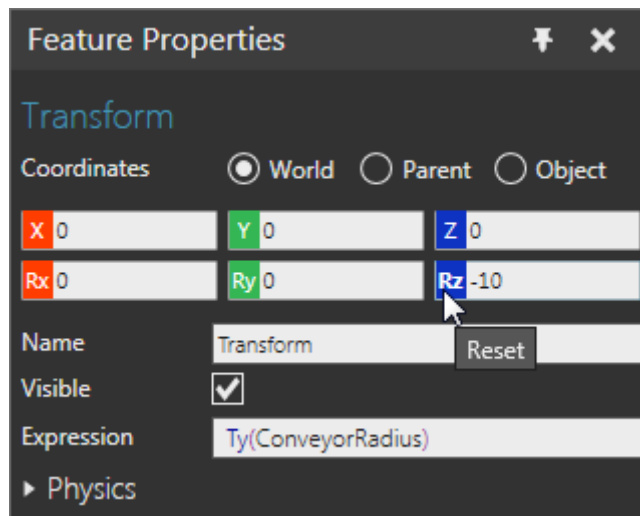


The purpose of this step is to help you understand that the Transform feature is parent to its subfeatures. By default, the Move Mode setting should be Hierarchy, so the conveyor legs move when you rotate the Transform feature. Notice that the legs are at the same distance (radius) from the origin of the Transform feature. That is, the position and orientation of the legs have not changed in their parent coordinate system.

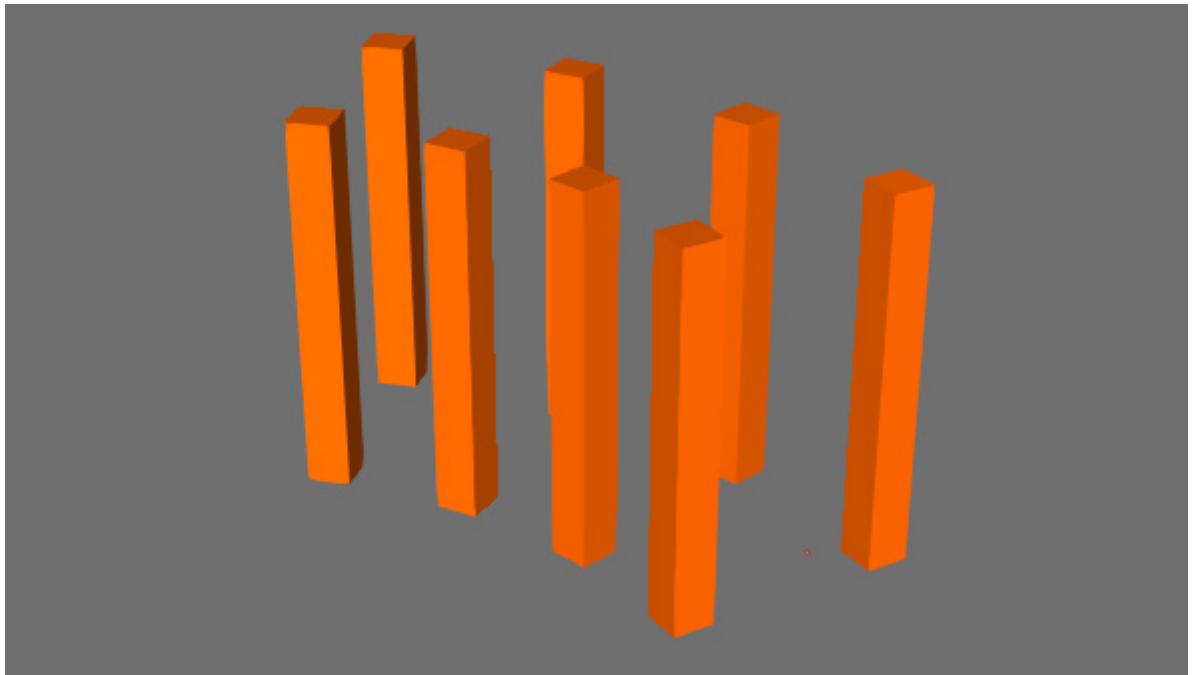
**NOTE!** In this case, the parent of the conveyor leg is the Linear Clone feature and its parent is the Transform feature.



7. In the Feature Properties panel, click **Rz** to reset the rotation around the Z-axis to zero.

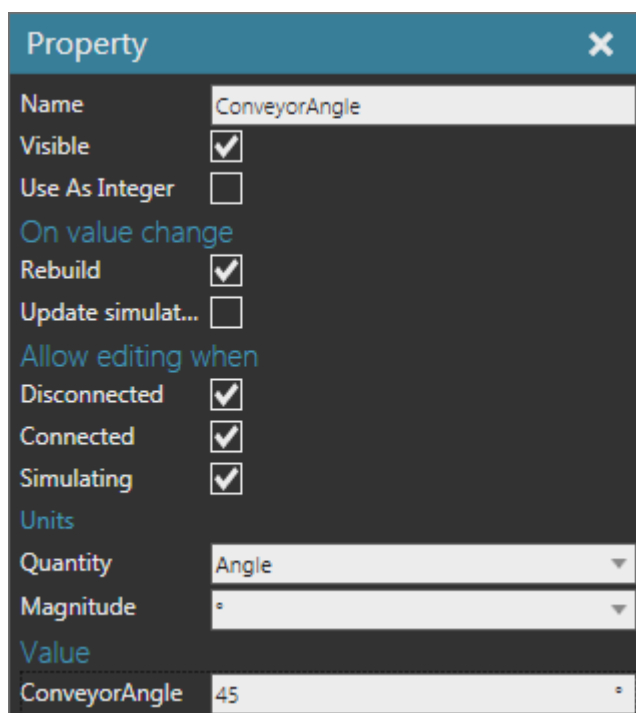


## Angular Clone Legs

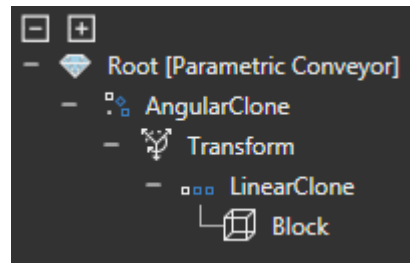
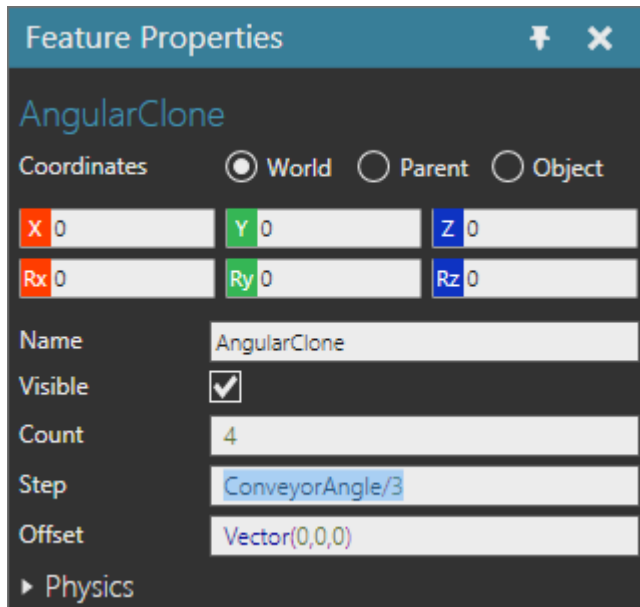


The conveyor legs can now be cloned along an angle, thereby determining the length of the conveyor. A component property will be used to control the angle or curve of the conveyor.

1. On the Modeling tab, in the Properties group, click the **Properties** arrow, and then under Basic, click **Real**.
2. In the Property task pane, set Name to **ConveyorAngle**, Quantity to **Angle** and Value to **45**.

A screenshot of the 'Property' task pane in a software application. The pane has a title bar with a close button. It contains several sections: 'Name' with the value 'ConveyorAngle'; 'Visible' with a checked checkbox; 'Use As Integer' with an unchecked checkbox; 'On value change' with 'Rebuild' checked and 'Update simulat...' unchecked; 'Allow editing when' with 'Disconnected', 'Connected', and 'Simulating' all checked; 'Units' with 'Quantity' set to 'Angle' and 'Magnitude' set to '°'; and 'Value' with 'ConveyorAngle' set to '45'.

3. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Clone, click **Angular Clone**.
4. In the Feature Properties panel, set Count to **4** and Step to "ConveyorAngle/3".
5. In the Component Graph panel, drag **Transform** and its tree onto **AngularClone** to apply the operation and clone the conveyor legs along a 45 degree angle.



## Create Roll Bar

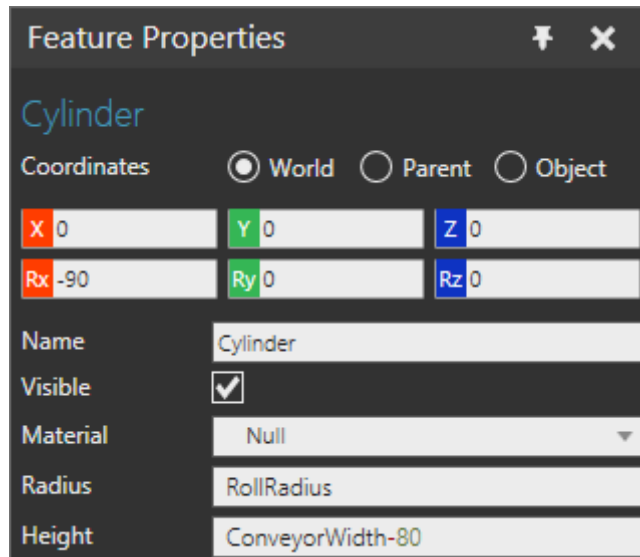


A primitive Cylinder feature can be used to represent a conveyor roll bar. A component property will be used to control the radius of the cylinder, thereby the width of conveyor roll bars.

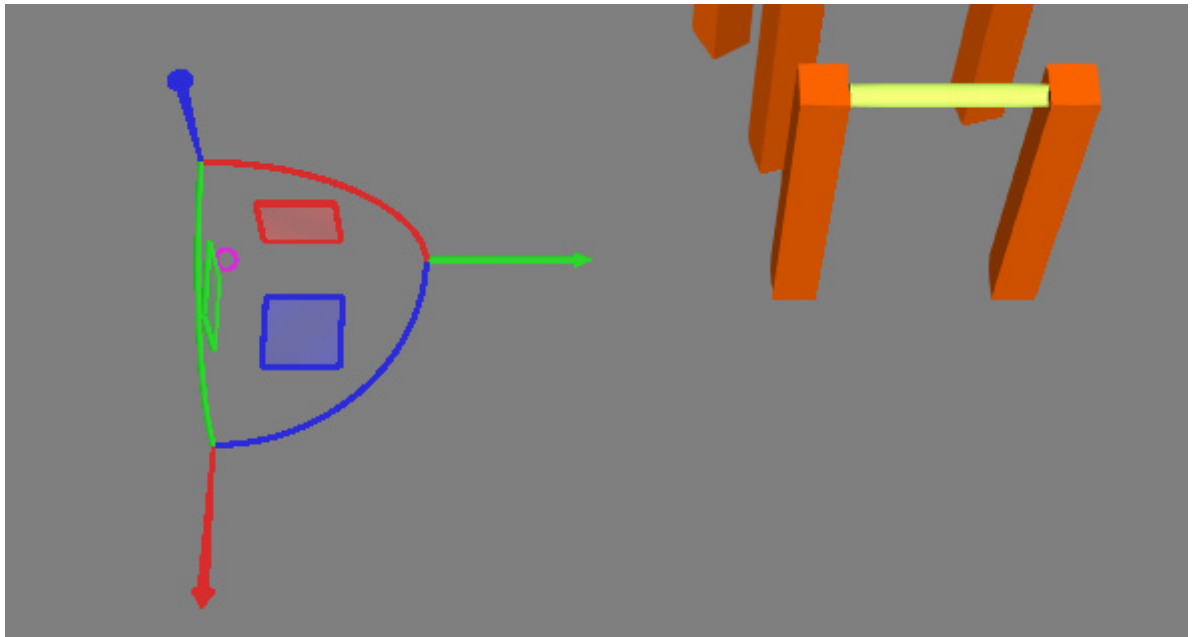
1. On the Modeling tab, in the Properties group, click the **Properties** arrow, and then under Basic, click **Real**.
2. In the Property task pane, set Name to **RollRadius**, Quantity to **Distance** and Value to **20**.

Property	
Name	RollRadius
Visible	<input checked="" type="checkbox"/>
Use As Integer	<input type="checkbox"/>
On value change	
Rebuild	<input checked="" type="checkbox"/>
Update simulat...	<input type="checkbox"/>
Allow editing when	
Disconnected	<input checked="" type="checkbox"/>
Connected	<input checked="" type="checkbox"/>
Simulating	<input checked="" type="checkbox"/>
Units	
Quantity	Distance
Magnitude	mm
Value	
RollRadius	20 mm

3. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Primitive Geometry, click **Cylinder**.
4. In the Feature Properties panel, set Radius to **RollRadius** and Height to **ConveyorWidth-80** and then set Rx to **-90** in World coordinates to rotate the conveyor roll bar so that its top and bottom face front to back.

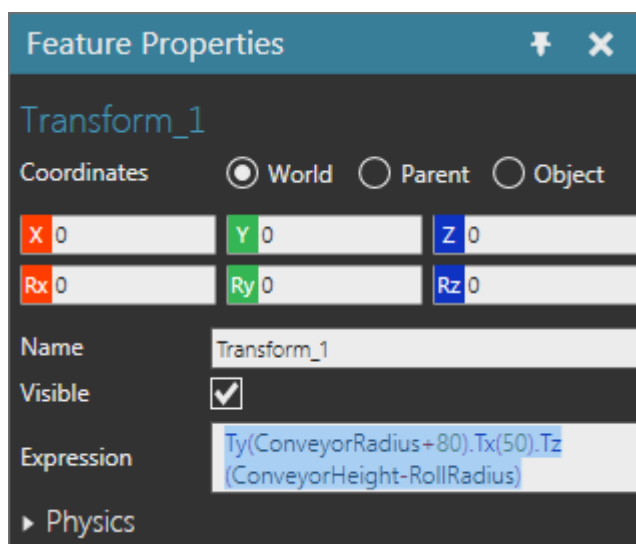


## Transform Roll Bar



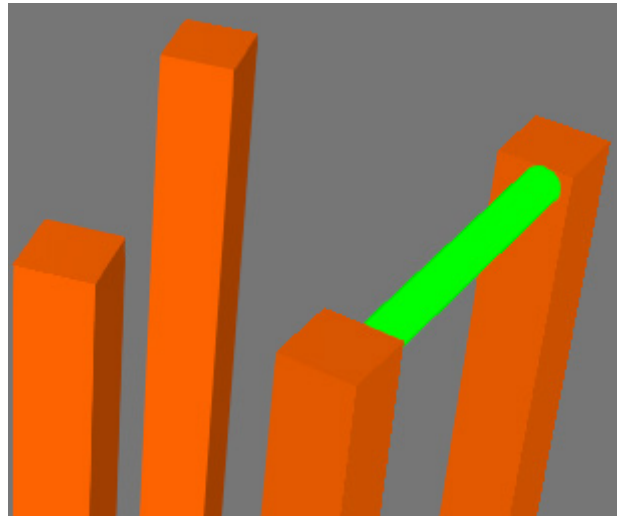
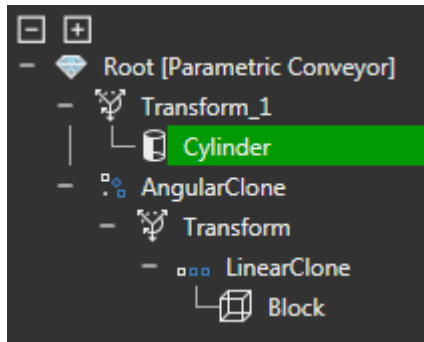
The conveyor roll bar can now be offset to be at the start of the conveyor and in between its first two legs.

1. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Transform, click **Transform**.
2. In the Feature Properties panel, set Expression to "Ty(ConveyorRadius+80).Tx(50).Tz(ConveyorHeight-RollRadius)".

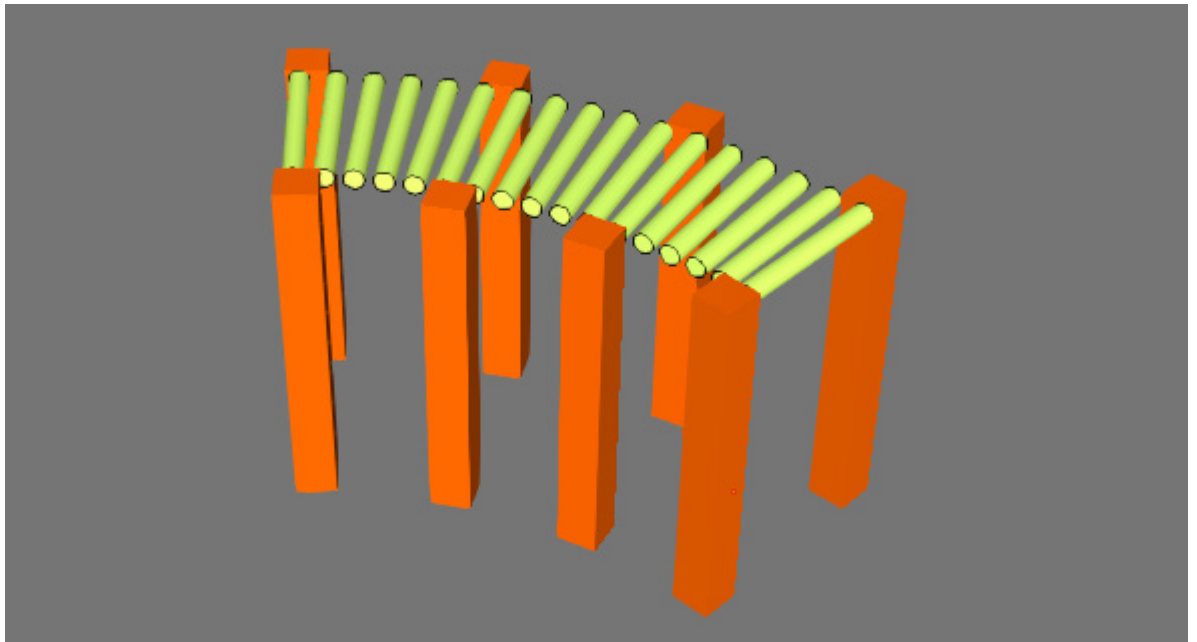


**NOTE!** The translation along the X-axis can be set to 40 or 55. The expression is evaluated from right to left. For example, the roll bar is first translated in the Z-axis, then it is moved in the X-axis before being rotated around the Y-axis.

3. In the Component Graph panel, drag **Cylinder** onto **Transform\_1** to apply the operation and translate the conveyor roll bar to be in between the first two conveyor legs.

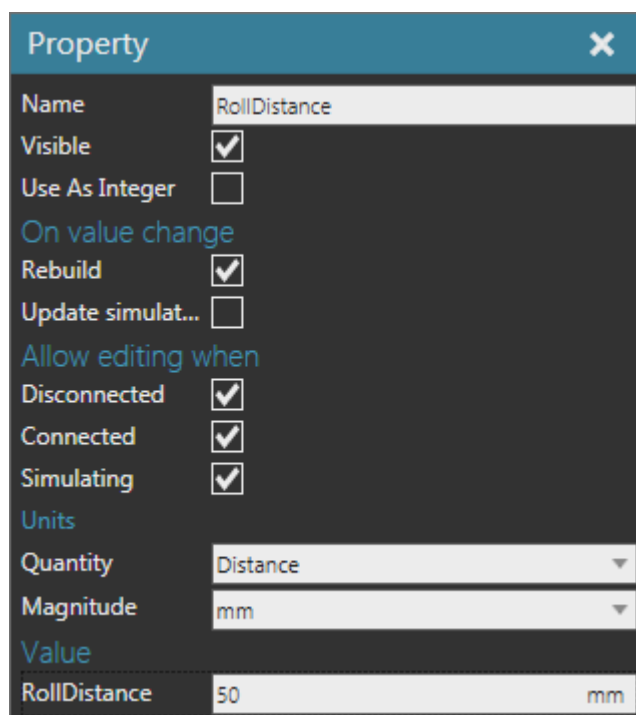


## Angular Clone Roll Bar



The conveyor roll bars can now be cloned along the curve or arc of the conveyor. In this case, you will use the conveyor angle and radius to determine the arc length. A component property will be used to calculate and control the distance between each roll bar.

1. On the Modeling tab, in the Properties group, click the **Properties** arrow, and then under Basic, click **Real**.
2. In the Property task pane, set Name to **RollDistance**, Quantity to **Distance** and Value to **50**.

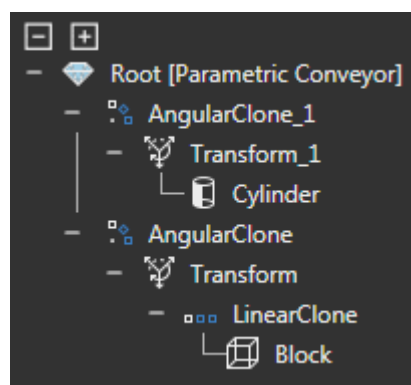
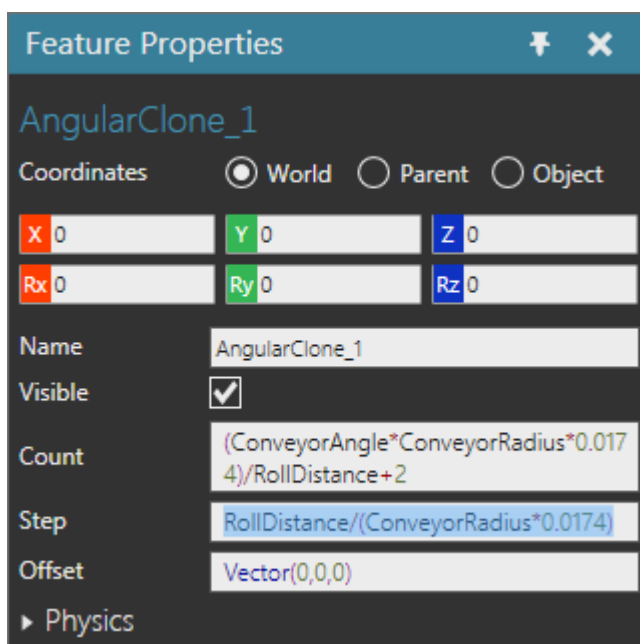




3. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Clone, click **Angular Clone**.

**NOTE!** Radians will be used, so an easy conversion from degrees to radians is to multiply the degree value by 0.0174, which is  $\pi/180.0$  degrees.

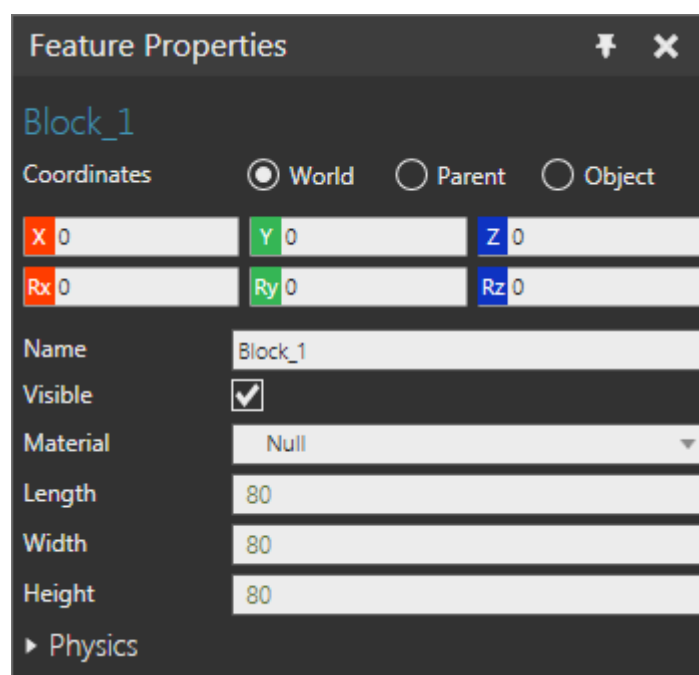
4. In the Feature Properties panel, do all of the following:
  - Set Count to  $(\text{ConveyorAngle} * \text{ConveyorRadius} * 0.0174) / \text{RollDistance} + 2$ .
  - Set Step to  $\text{RollDistance} / (\text{ConveyorRadius} * 0.0174)$ .
5. In the Component Graph panel, drag **Transform\_1** and its tree onto **AngularClone\_1** to apply the operation and clone the conveyor roll bars along an arc.



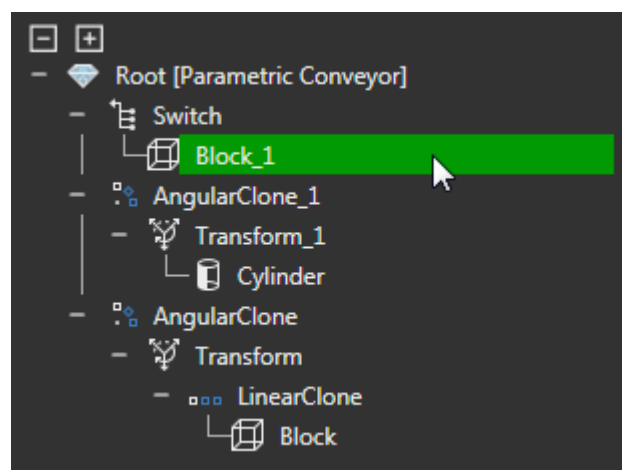
# Create Side Rail

A primitive Block feature or Python feature can be used to represent a conveyor side rail. In this case, you will learn how to use both features as well as a Switch feature for enabling or disabling a feature tree.

1. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Other, click **Switch**.
2. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Primitive Geometry, click **Box**.
3. In the Feature Properties panel, set Length, Width and Height to **80**.

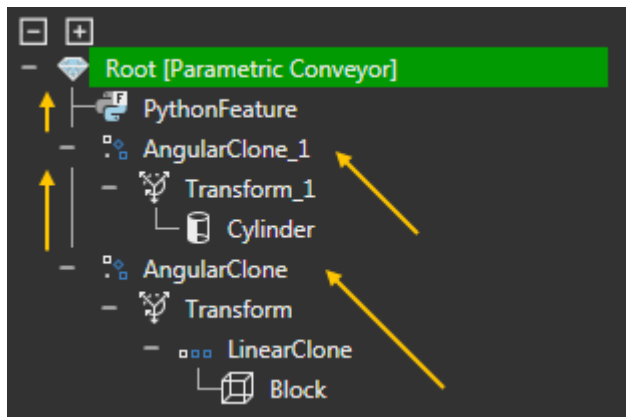


4. In the Component Graph panel, drag **Block\_1** onto **Switch** to apply the operation. The block will disappear in the 3D world because the Choice property of the Switch is zero. That is, the Switch has not selected one of its subfeatures.



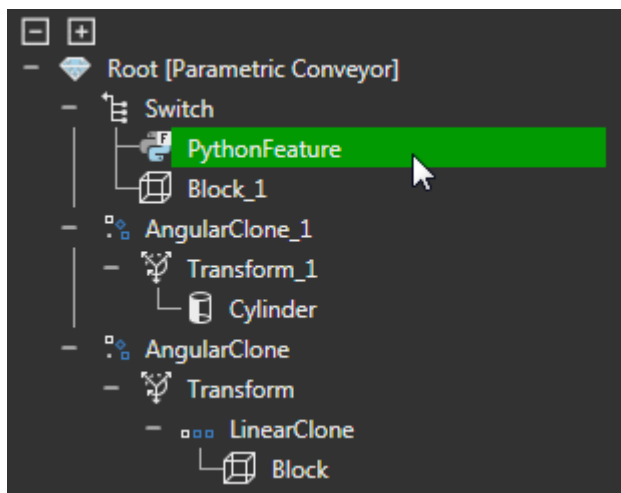
# Python Feature

A Python feature allows you to create a custom feature by writing a script with Python API. The script is evaluated like any other feature in a node feature tree, from lowest to highest level until reaching the root level.



In some cases, a Python feature can be used like a Transform feature to transform its subfeatures. In other cases, a Python feature can be used to create simple and complex geometry as well as manipulate mesh topology and geometry. For more information, refer to the "Python Feature" topic in Help documentation.

1. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Other, click **Python**.
2. In the Component Graph panel, drag **PythonFeature** onto **Switch** to apply the operation.



3. In the Feature Properties panel, click **Open In Editor** to display its script editor.

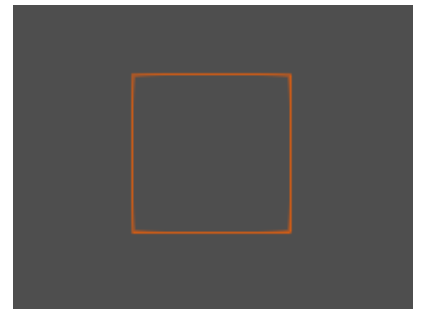
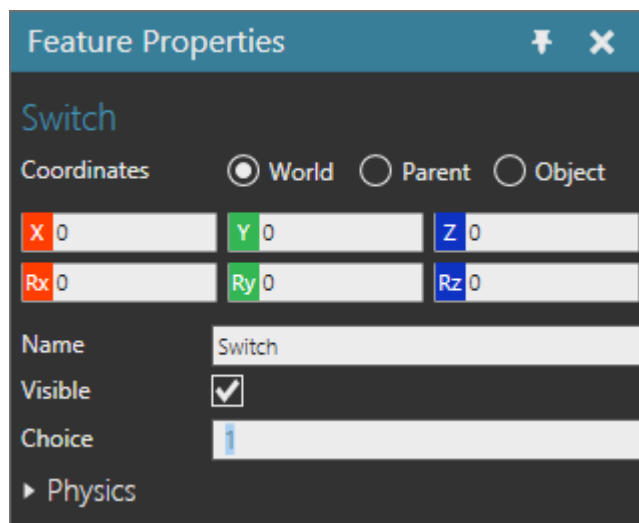
4. In the script editor, add the following code to create a rectangle, and then compile the code.

```
from vcFeature import *
import vcVector

def OnEvaluate(feature, target, matrix, containers):
    target.clear()
    c_set = target.createGeometrySet(VC_COMPACTLINESET)
    line = [vcVector.new(0,0,0),vcVector.new(0,0,80),vcVector.new(0,80,80),\
vcVector.new(0,80,0),vcVector.new(0,0,0)]
    c_set.addLine(line)

    for c in containers:
        c.moveGeometrySets(target,matrix)
```

5. In the Component Graph panel, select the **Switch** feature, and then in the Feature Properties panel, set Choice to **1** to select the first subfeature of the Switch.



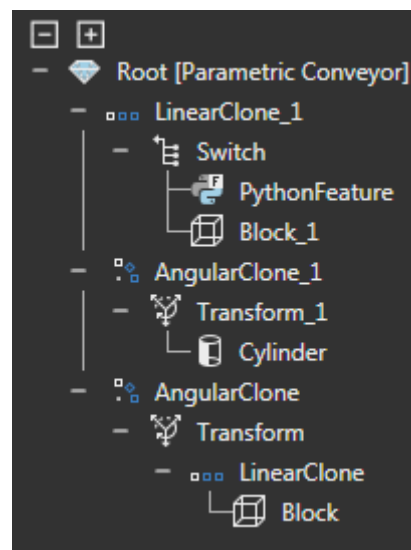
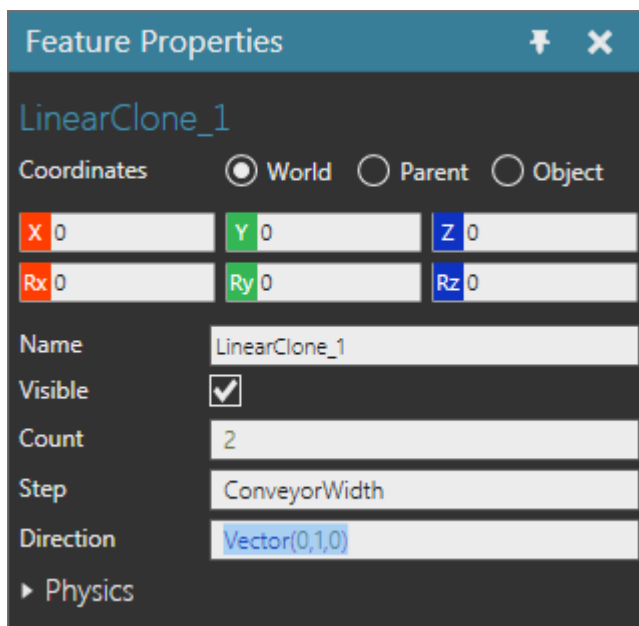
In this case, the first subfeature of the Switch is the Python feature you used to create a 2D rectangle/square. If you prefer to use a Block feature as the side rail then set Choice to 2.

# Linear Clone Side Rail

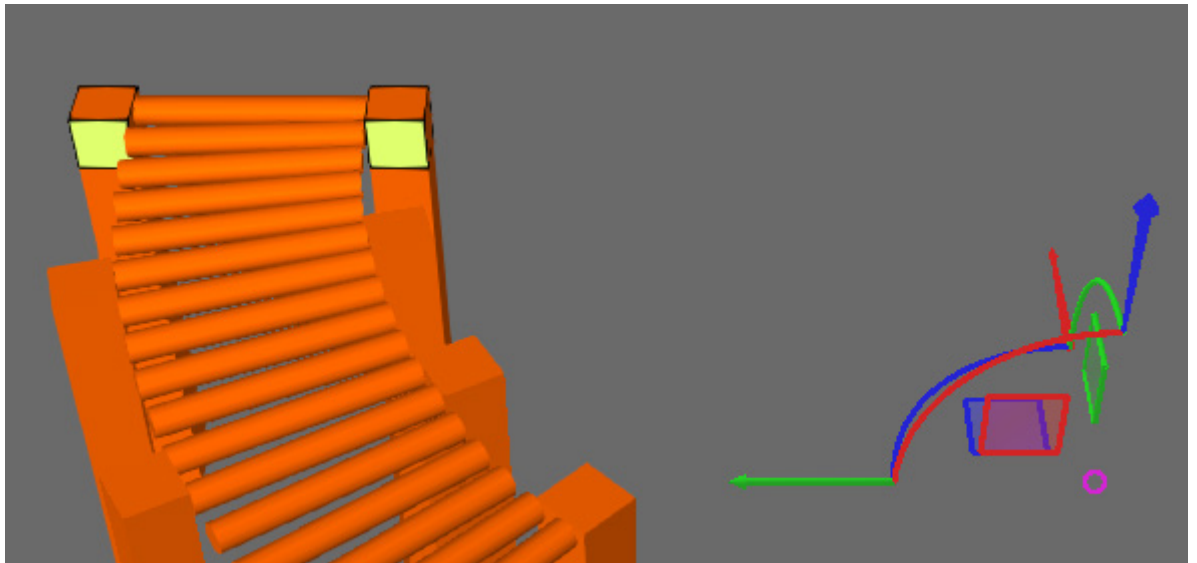


The side rail of the conveyor can now be cloned to create a set of two side rails.

1. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Clone, click **Linear Clone**.
2. In the Feature Properties panel, set Count to **2**, Step to **ConveyorWidth** and Direction to **Vector(0,1,0)** to clone in the Y-axis direction.
3. In the Component Graph panel, drag **Switch** onto **LinearClone\_1** to apply the operation and clone the conveyor side rail.

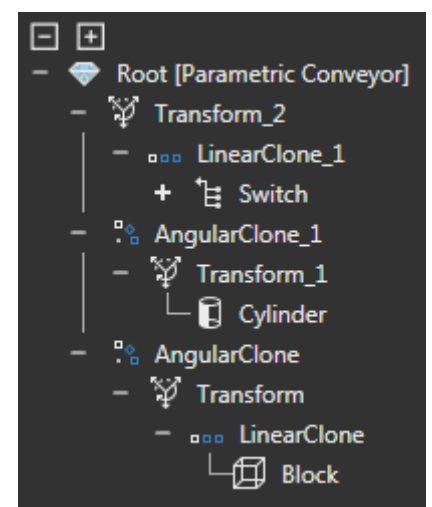
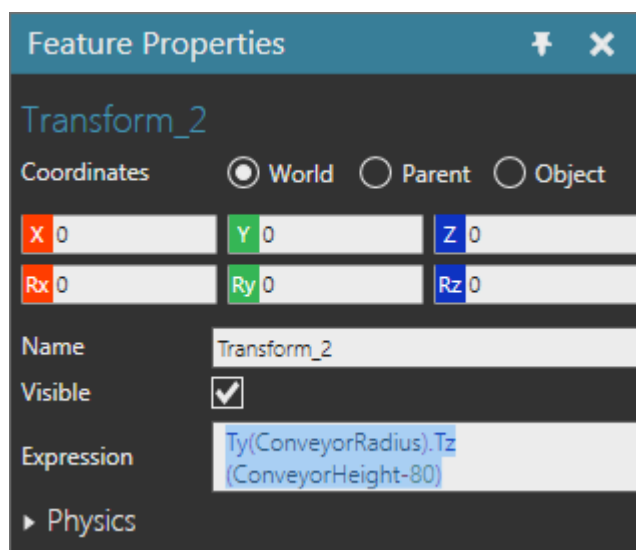


## Transform Side Rails

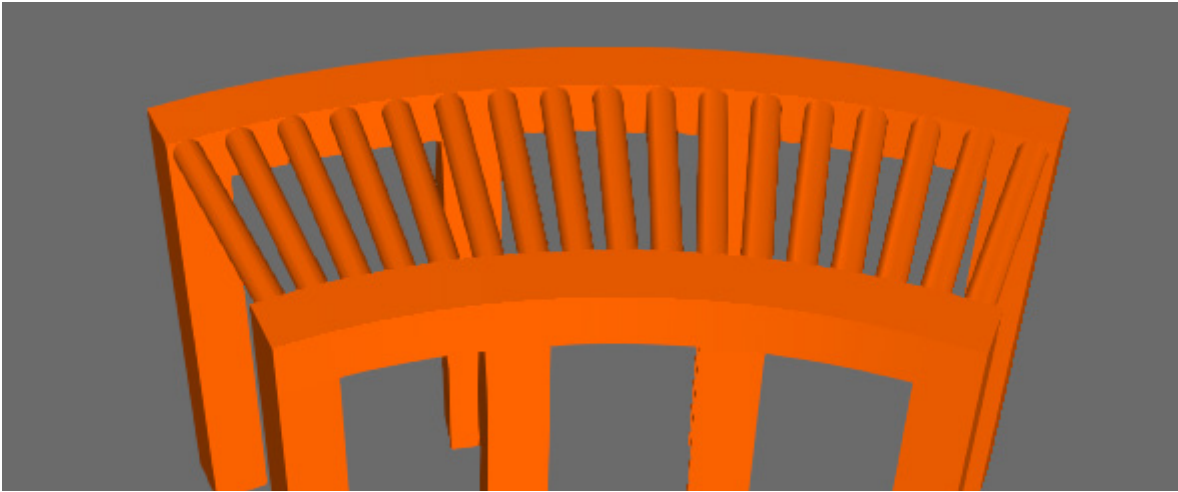


The conveyor side rails now need to be offset from the component origin and placed at the start of the conveyor. This could be as simple as using the Move tool, but in the next section you will need to revolve (sweep) the side rail geometry to make a curved surface. Therefore, a Transform feature will be used to offset the side rails.

1. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Transform, click **Transform**.
2. In the Feature Properties panel, set Expression to "Ty(ConveyorRadius).Tz(ConveyorHeight-80)".
3. In the Component Graph panel, drag **LinearClone\_1** and its tree onto **Transform\_2** to apply the operation and translate the conveyor side rails to be flush with the right side of two conveyor legs.

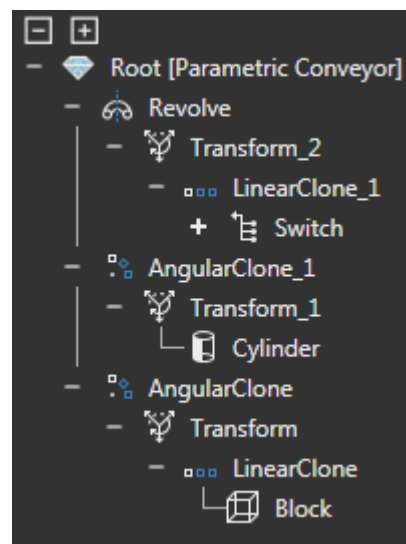
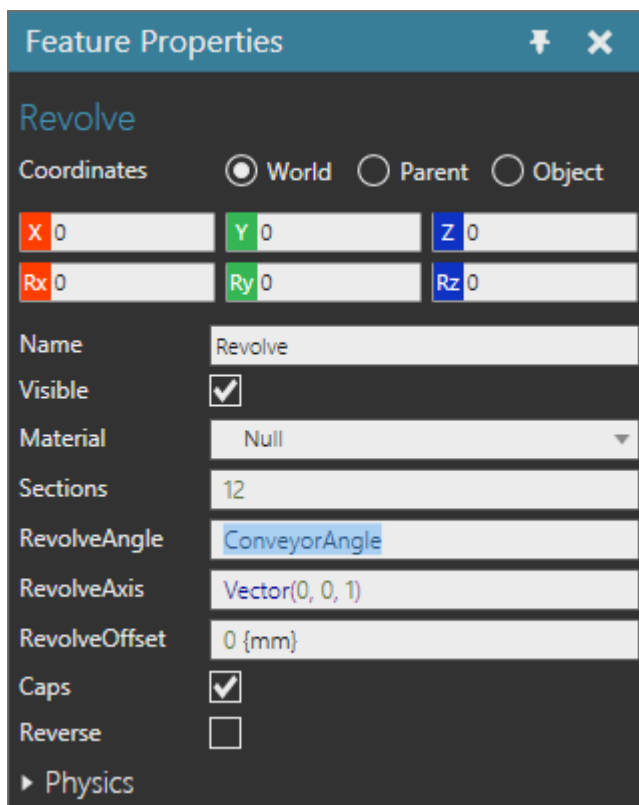


## Revolve Side Rails

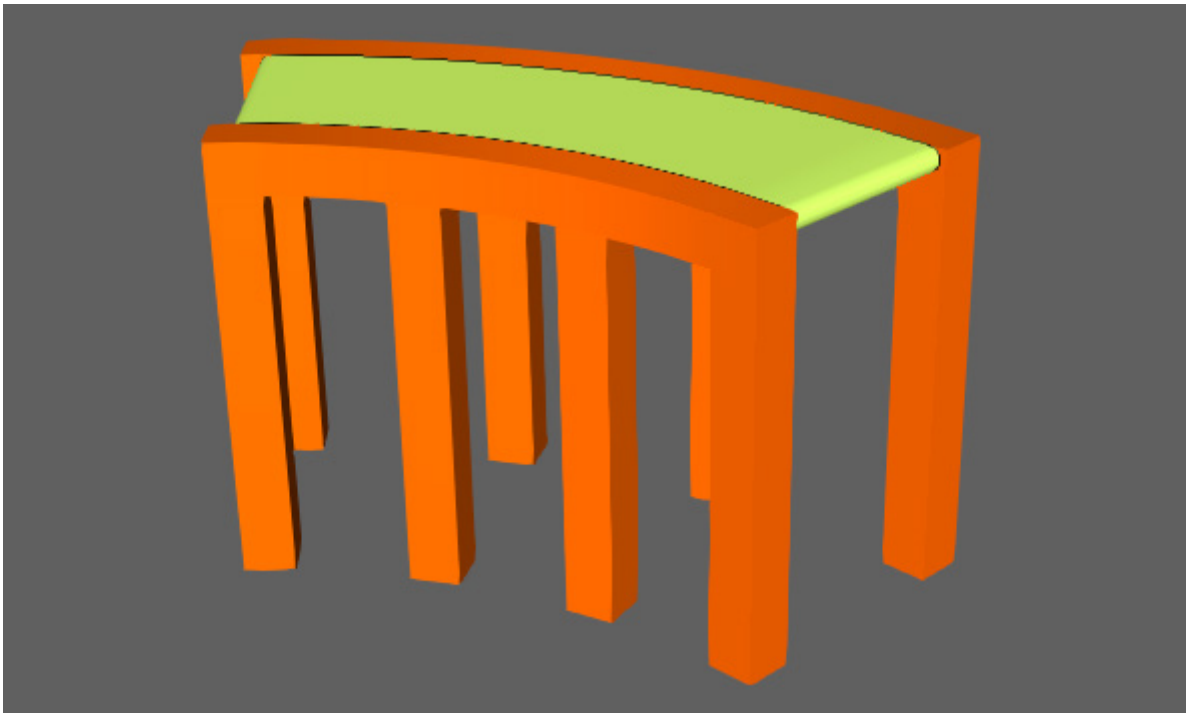


The conveyor side rails can now be revolved or swept along an axis to create a curved surface. In this case, the geometry of the side rails is the sweep profile.

1. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Generate, click **Revolve**.
2. In the Feature Properties panel, set RevolveAngle to **ConveyorAngle**.
3. In the Component Graph panel, drag **Transform\_2** and its tree onto **Revolve** to apply the operation and sweep the geometry of the side rails along an arc.

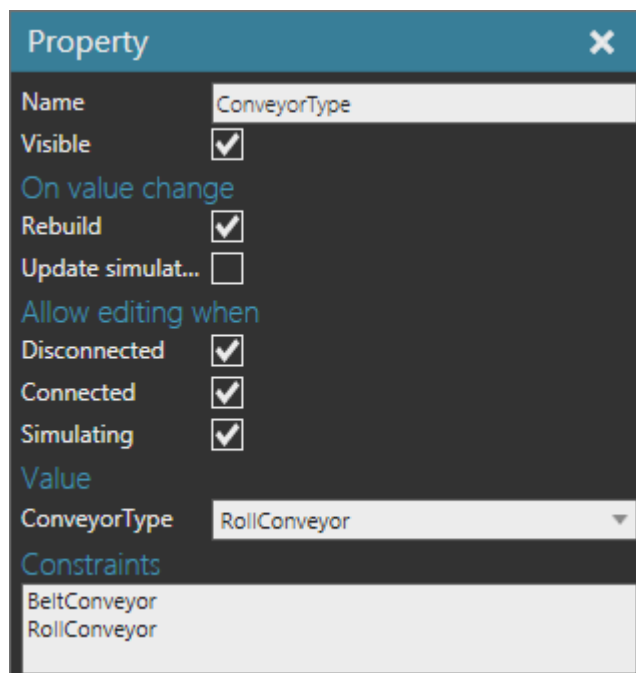


## Exercise - Create Conveyor Belt



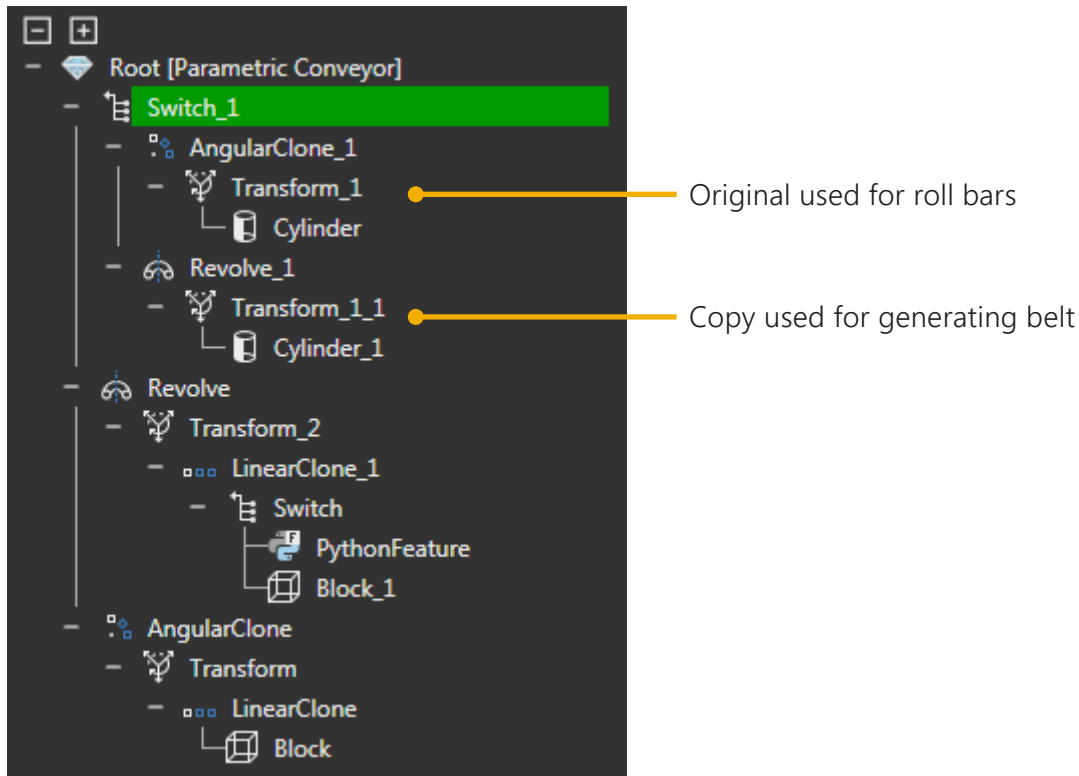
You know how to transform, clone and revolve features as well as choose which subfeatures are affected by operations. You also know how to use component properties to rebuild geometry. The curve conveyor can use roll bars or a belt to move parts. Try to create a belt for the conveyor on your own or follow the given steps.

1. Add a String property to the component and name it ConveyorType. The property should have a step constraint with two values: BeltConveyor and RollConveyor with a default value of RollConveyor.





2. Add a Switch feature, and then copy Transform\_1 feature and its tree, and then paste the copy inside the Switch\_1 feature. You only need to copy the Transform\_1 feature because its tree is copied with it.
3. Add a Revolve feature and set its RevolveAngle to ConveyorAngle. In the node feature tree, move Revolve\_1 inside Switch\_1, and then make Revolve\_1 the parent of the copied roll bar tree.
4. Reorder the node feature tree so that the tree for the original roll bars is nested in Switch\_1.



5. Edit the Choice property of Switch\_1 to use the following expression:
  - `ConveyorType=="RollConveyor"?1:(ConveyorType=="BeltConveyor"?2:0)`

This type of expression might be new to you. It uses a condition operator to evaluate the value of the ConveyorType property. If the ConveyorType property is set to RollConveyor the switch will select its first subfeature, which needs to be the feature tree used for generating roll bars. If the first condition is False, the next condition is evaluated and the switch will either select its second subfeature or none. In this case, the second subfeature is the feature tree used for generating the belt.

**TIP!** Some modelers prefer to use a Python Script behavior to handle property change events and update features. This approach is outside the scope of this tutorial, but lessons about this topic can be found on the Visual Components Academy website.

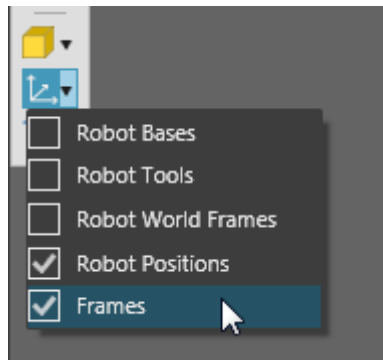
6. Test your work by changing the value of ConveyorType to BeltConveyor and then back to RollConveyor.

# Define Path Frames

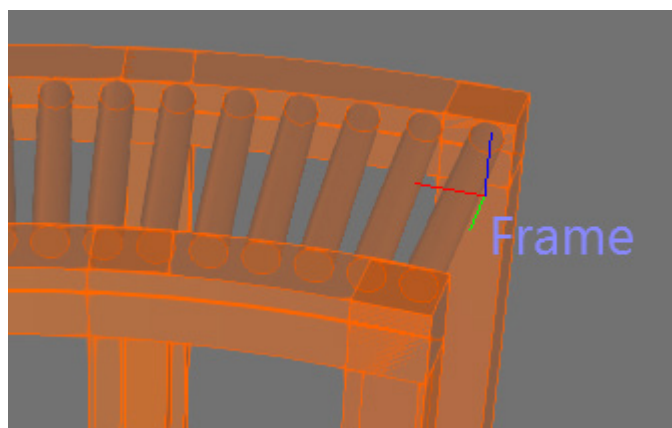
Frame features can be added to the conveyor and used as waypoints for moving parts. In this case, frames will be rotated and transformed to form a curved path from beginning to end of the conveyor.

## Start Frame

1. On the 3D world toolbar, click the **Frame Types** arrow, and then select the **Frames** check box to turn on the global visibility of Frame features.

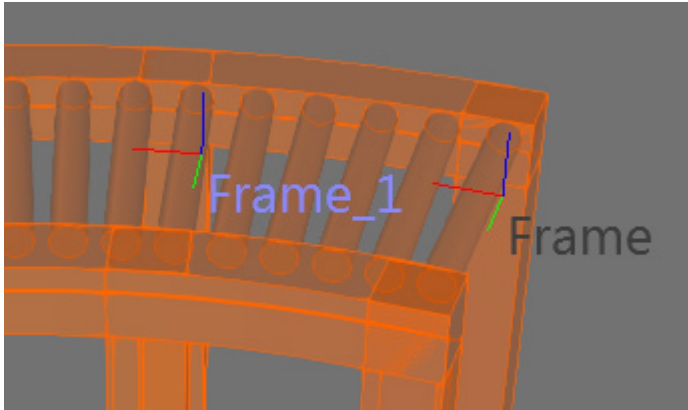


2. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Transform, click **Transform**.
3. In the Feature Properties panel, set Expression to  $\text{"Tz(ConveyorHeight).Ty(ConveyorRadius+(ConveyorWidth+80)/2.0).Tx(80)"}$ .
4. On the Modeling tab, in the Geometry group, click the **Features** arrow, and then under Other, click **Frame**.
5. In the Feature Properties panel, set Rz to **180.0** in World coordinates so that the positive X-axis of the frame faces the direction parts will move along the conveyor.
6. In the Component Graph panel, drag **Frame** onto **Transform\_3** to apply the operation and reposition the frame at the start of the conveyor.



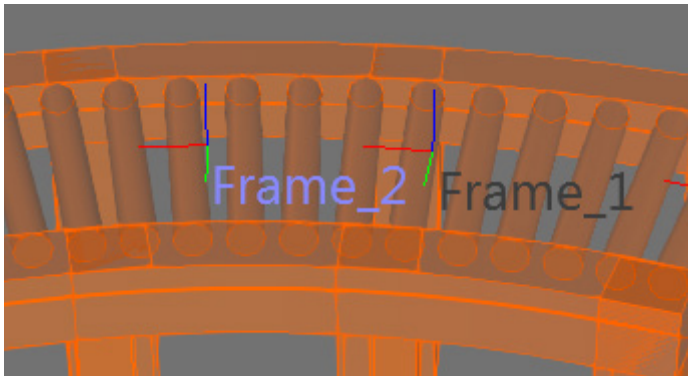
## Mid Frame 1

1. Create a **Transform** feature, and then set its Expression to  $\text{Tz}(\text{ConveyorHeight}).\text{Rz}(\text{ConveyorAngle}/4.0).\text{Ty}(\text{ConveyorRadius}+(\text{ConveyorWidth}+80)/2.0)$ .
2. Create a **Frame** feature, rotate it **180.0** around its Z-axis, and then drag the frame onto the Transform feature created in the previous step.



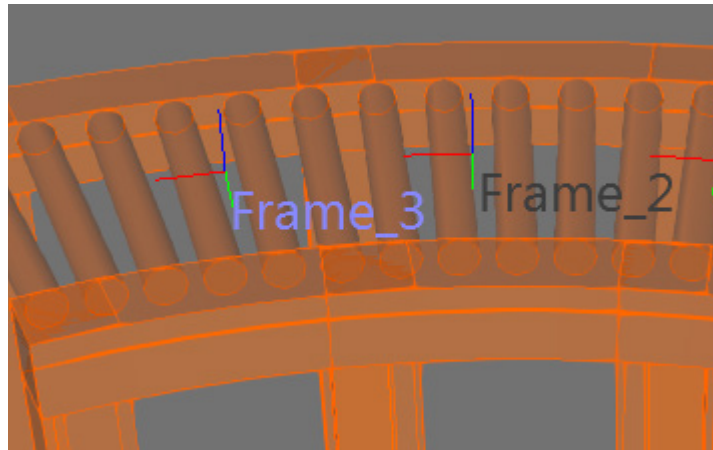
## Mid Frame 2

1. Create a **Transform** feature, and then set its Expression to  $\text{Tz}(\text{ConveyorHeight}).\text{Rz}(\text{ConveyorAngle}*2.0/4.0).\text{Ty}(\text{ConveyorRadius}+(\text{ConveyorWidth}+80)/2.0)$ .
2. Create a **Frame** feature, rotate it **180.0** around its Z-axis, and then drag the frame onto the Transform feature created in the previous step.



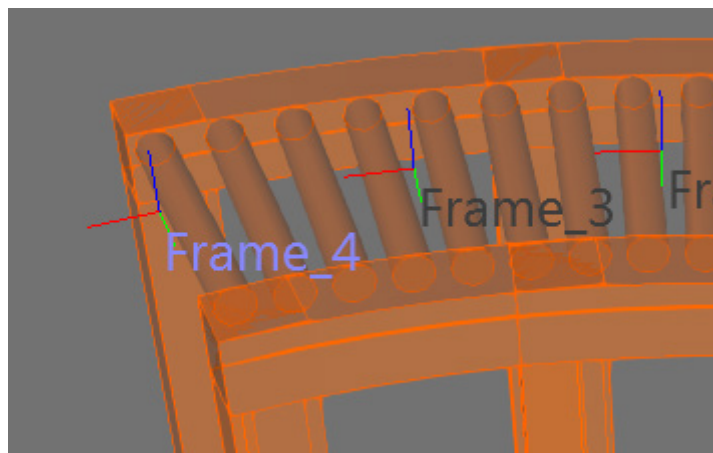
## Mid Frame 3

1. Create a **Transform** feature, and then set its Expression to `"Tz(ConveyorHeight).Rz(ConveyorAngle*3.0/4.0).Ty(ConveyorRadius+(ConveyorWidth+80)/2.0)"`.
2. Create a **Frame** feature, rotate it **180.0** around its Z-axis, and then drag the frame onto the Transform feature created in the previous step.



## End Frame

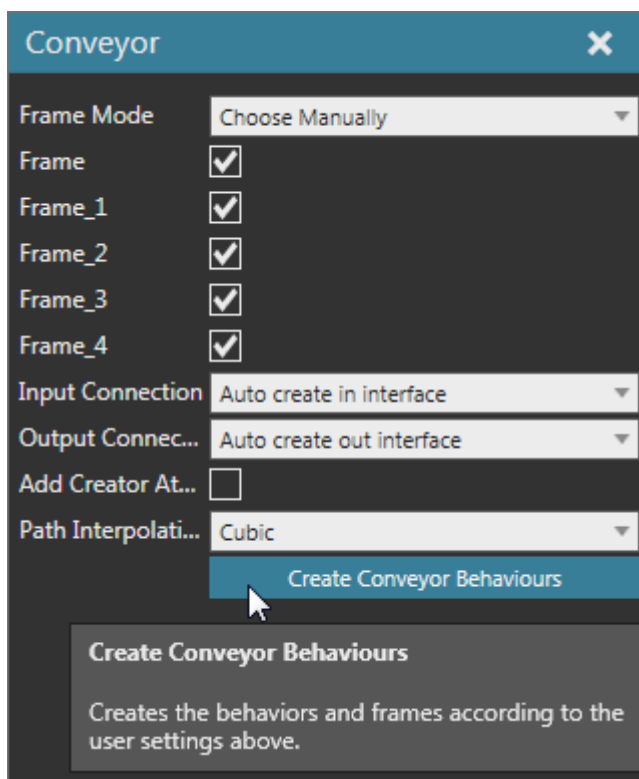
1. Create a Transform feature, and then set its Expression to `"Tz(ConveyorHeight).Rz(ConveyorAngle).Ty(ConveyorRadius+(ConveyorWidth+80)/2.0)"`.
2. Create a **Frame** feature, rotate it **180.0** around its Z-axis, and then drag the frame onto the Transform feature created in the previous step.



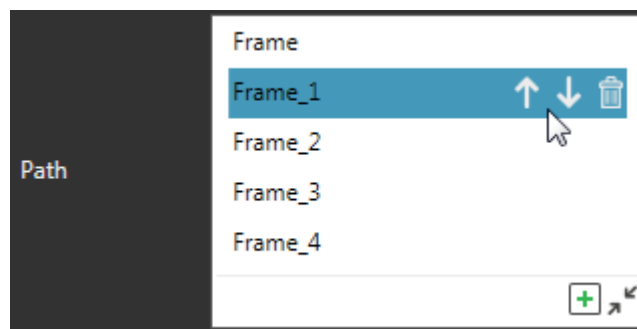
# Conveyor Wizard

A Conveyor wizard can be used to quickly set up the path and interfaces of the conveyor. These behaviors will allow the conveyor to move parts to and from other components during a simulation.

1. On the Modeling tab, in the Extra group, click the **Wizards** arrow, and then click **Conveyor**.
2. In the Conveyor task pane, set Frame Mode to **Choose Manually**, and then select the check box of every listed **Frame** feature.
3. Now set Path Interpolation to **Cubic**, and then click **Create Conveyor Behaviours**, and then click **Close**.

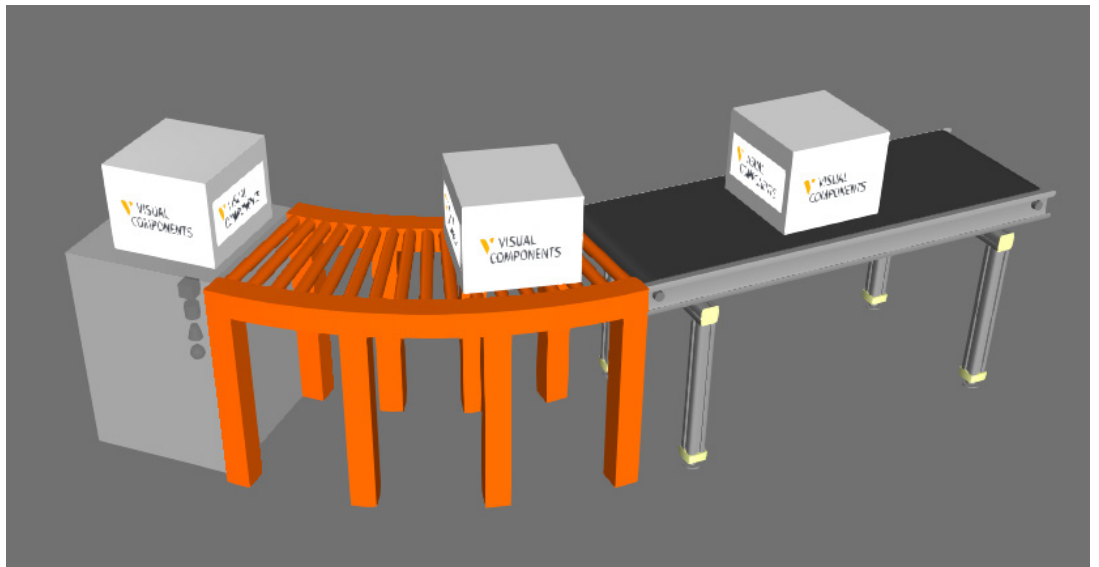


**NOTE!** In some cases, you may need to reorder the path frames by editing the Path property of the path behavior.



## Test Path and Interfaces

1. Save the component.
2. Click the **Home** tab, and then in the Component Properties panel, set ConveyorWidth to **500**.
3. Click the **Home** tab, and then in the eCatalog panel, Collections view, under Models by Type, click **Feeders**, and then add and connect a **Shape Feeder** to the beginning end of the **conveyor** in the 3D world.
4. In the eCatalog panel, Collections view, under Models by Type, expand **Conveyors**, click **Visual Components**, and then add and connect a **Conveyor** to the other end of the **conveyor** in the 3D world.
5. (Optional) Add and connect a **Conveyor Sensor** to the path of the **Parametric Conveyor**. The sensor would be connected to the conveyor and attached to its path behavior as an external sensor.
6. Run the simulation, verify boxes move from one conveyor to another, and then reset the simulation.

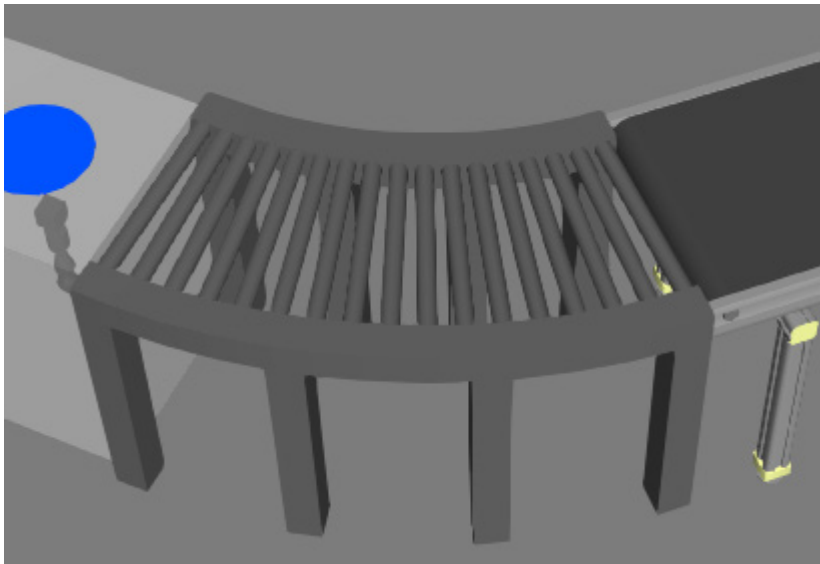


7. Save the layout.

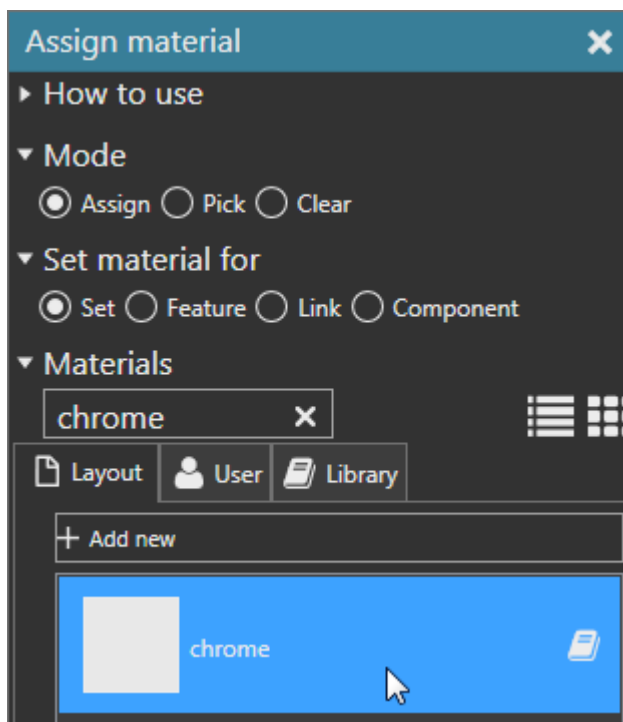
# Assign Materials

You can assign materials to component geometry at different levels. For example, you can assign materials to geometry sets, features, links or the entire component.

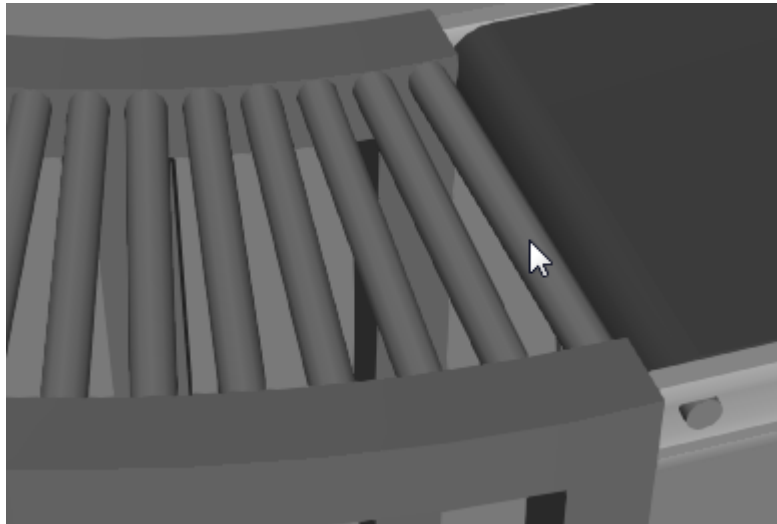
1. In the 3D world, select **Parametric Conveyor**, and then in the Component Properties panel, set Material to **steel\_crosspattern**. This affects all features in the component because you have not assign them their own material.



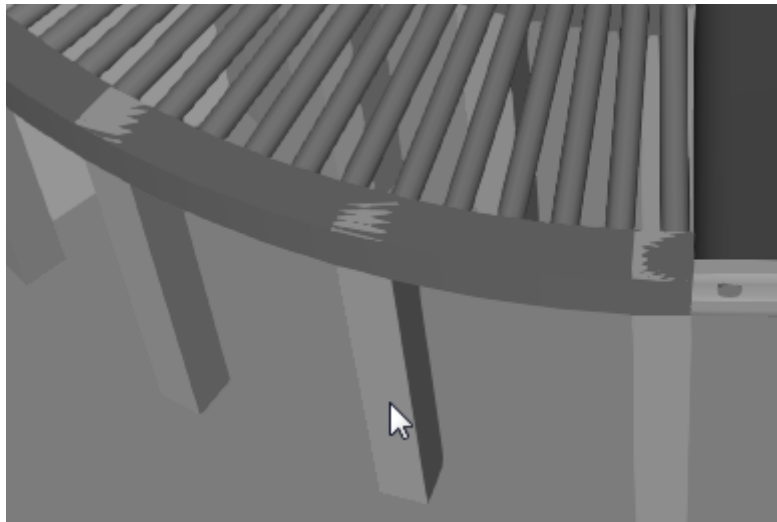
2. Click the **Modeling** tab, and then in the Geometry group, click the **Tools** arrow, and then under Material, click **Assign**.
3. In the Assign material task pane, under Materials, search for and select **chrome**.



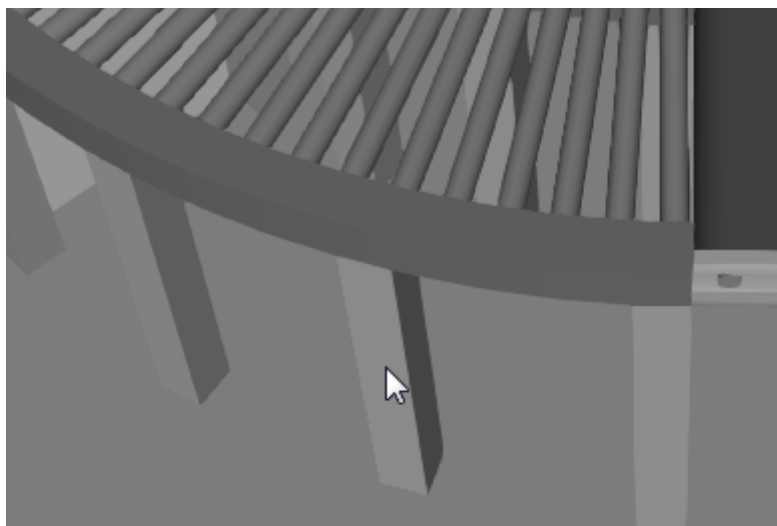
4. In the 3D world, click a roll bar to assign chrome to every roll bar.



5. Assign **aluminium** to the legs of the conveyor.

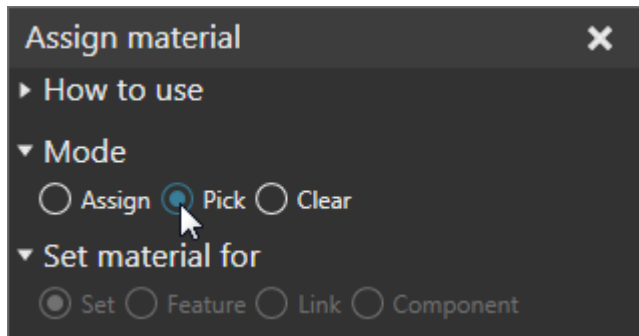


6. There is a noticeable plane intersection between the conveyor legs and side rails, so select the **Block** feature of the conveyor leg and set its Height to **ConveyorHeight-80** to fix this issue.

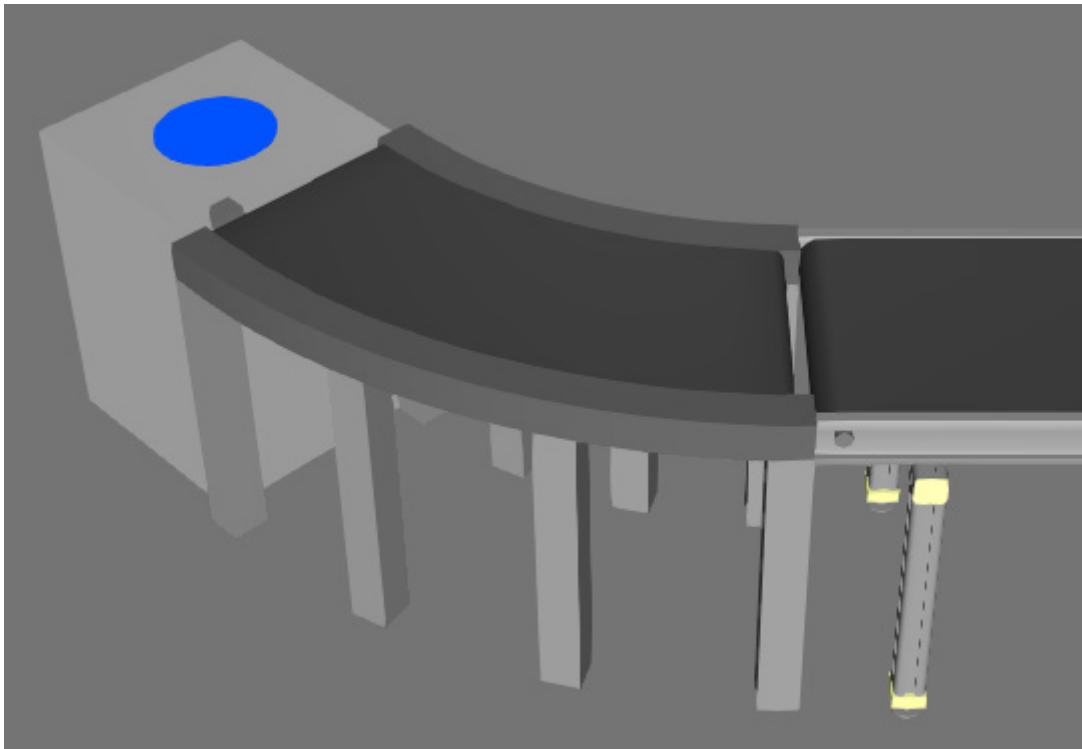




7. In the Assign material task pane, under Mode, select **Pick** to inspect the material of an object.



8. In the 3D world, click the belt of the other conveyor. The inspected material should be rubber. In this case, you are selecting a material not a feature in the 3D world.
9. Select the **Revolve\_1** feature that is used to generate the curve conveyor belt, and then set its Material to **rubber**.
10. Set ConveyorType to **BeltConveyor** to verify the belt was assigned the rubber material, and then save the component.



## Review

In this tutorial you learned how to model a parametric conveyor by using features, properties, expressions and wizards. You know how to clone geometry along an axis and angle. You also understand that in some cases geometry must be transformed before it can be cloned or revolved. You know how to assign materials to a component as well as its features and geometry sets. You also know how to switch the geometry of a component based on property values. Finally, you know how to create a custom feature using Python API and use a wizard to create the behaviors of a conveyor.