MFG3914

Templates, Configurations, and Containers for agile prototype machining in Fusion

Philip Mestenhauser
Meta

Learning Objectives:

- Create a flexible Template design that can easily scale to other machines
- Learn how containers power automated CAM workflows
- Understand how CAD configurations can cut through the complexity of modular work-holding systems into a few source CAD files.
- Replace, Derive, or Reconfigure? Understand the nuance behind these tools. Leverage Toolpath Templates with CAD configurations

Description

Have you ever attempted to transfer your machining process to another machine, only to find yourself dealing with lost CAD and CAM associations; wanting to start-over from scratch?

In this session we will take a deep-dive into Fusion Configurations to build robust assemblies of your work-holding systems that will elegantly repair CAD and CAM associations when you need to change course.

Through this course attendees will develop robust CAD frameworks capable of supporting various CNC workflows for both milling and turning that empower the machine simulation environment.

Building on these frameworks we will explore the potential of Geometry Features to automate the machining process for common part families using selectionless toolpaths, ultimately streamlining routine cutting operations.

[Download the Framework for yourself]

Philip Mestenhauser

Philip Mestenhauser is a CNC machinist, and optomechanical prototyping engineer focused on product development prototyping at Meta.

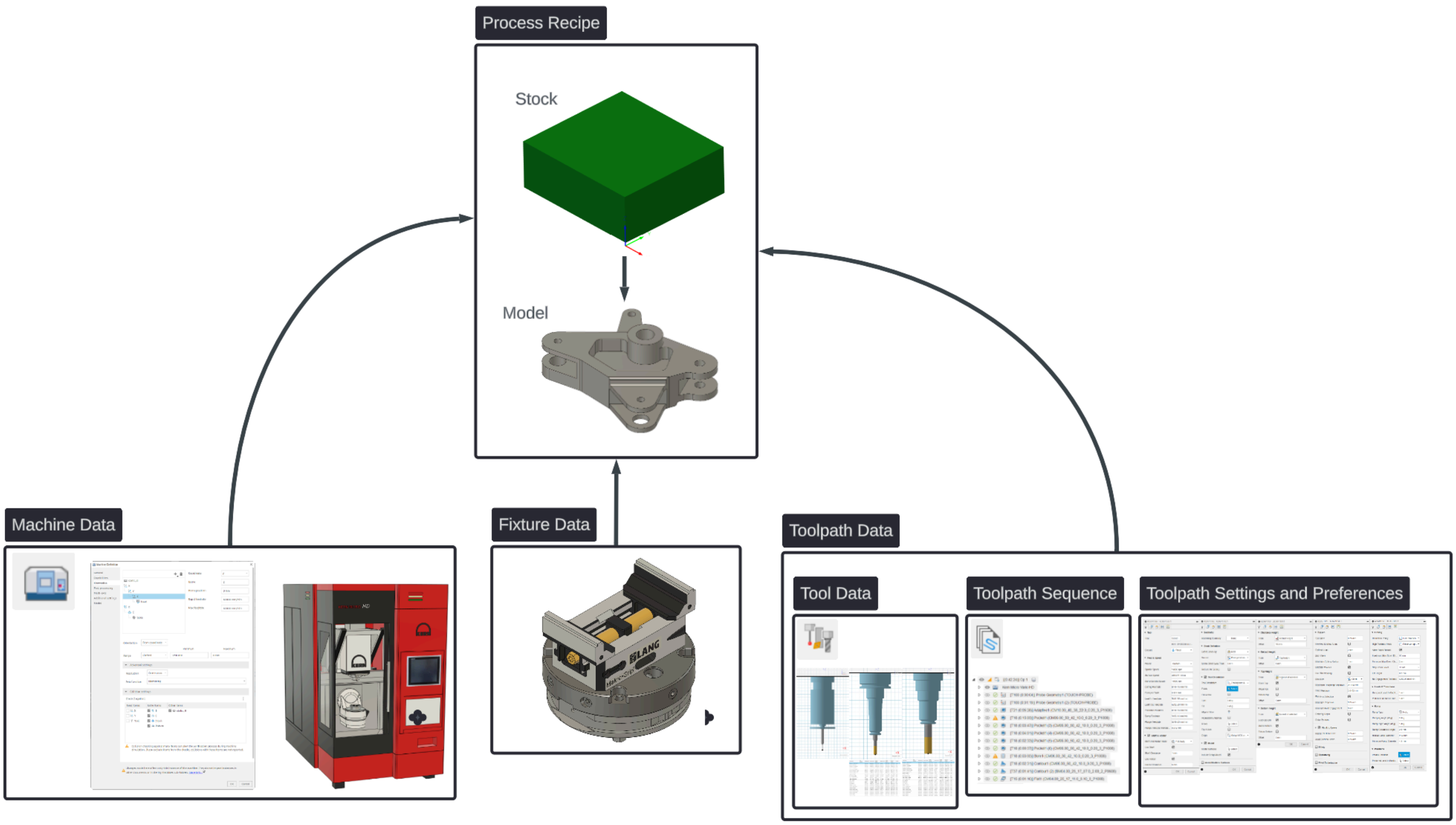
Fusion is missing something..

Fusion has Libraries for Tools

Fusion has libraries for Toolpaths

Fusion has libraries for Machines, and posts..

Fusion needs libraries for Fixtures



Unfortunately Autodesk doesn't provide a cloud based solution in Fusion for a Fixturing library.

Fixturing is a critical component of the machining recipe, and we need a solution similar to a cloud tool library that we can share with our team.

This handout will provide a detailed breakdown of key concepts to build your own **Fixture Assemblies** by using component containers, joint origins, and configurations.

You can insert these **Fixture Assemblies** into a **CAM Template** for an agile CNC programming environment that will allow you to replace single vise elements, or whole assemblies to quickly move existing programs machinery as you respond to the needs of your shop and workflows.

Table of Contents

Background info

[Fusion - Assemblies Master Class | Kevin Schneider]

[Tips and Tricks: Get the Most Out of Configurations in Fusion 360]

[Streamlining CAM Workflows with Templates | Rob Lockwood]

CAM Templates for predefined workflows

Container templates are very simple designs for starting a specific machining workflow.
They are well documented by Rob Lockwood; who provides an excellent presentation about how these template documents can reduce repetition in our programming.
Watch it before continuing through this handout or watching my presentation.
A container template is the most important concept a CNC machinist using Fusion must know.

Components are Containers for model CAD

A **Body** is simply geometry and must exist inside a **Component**

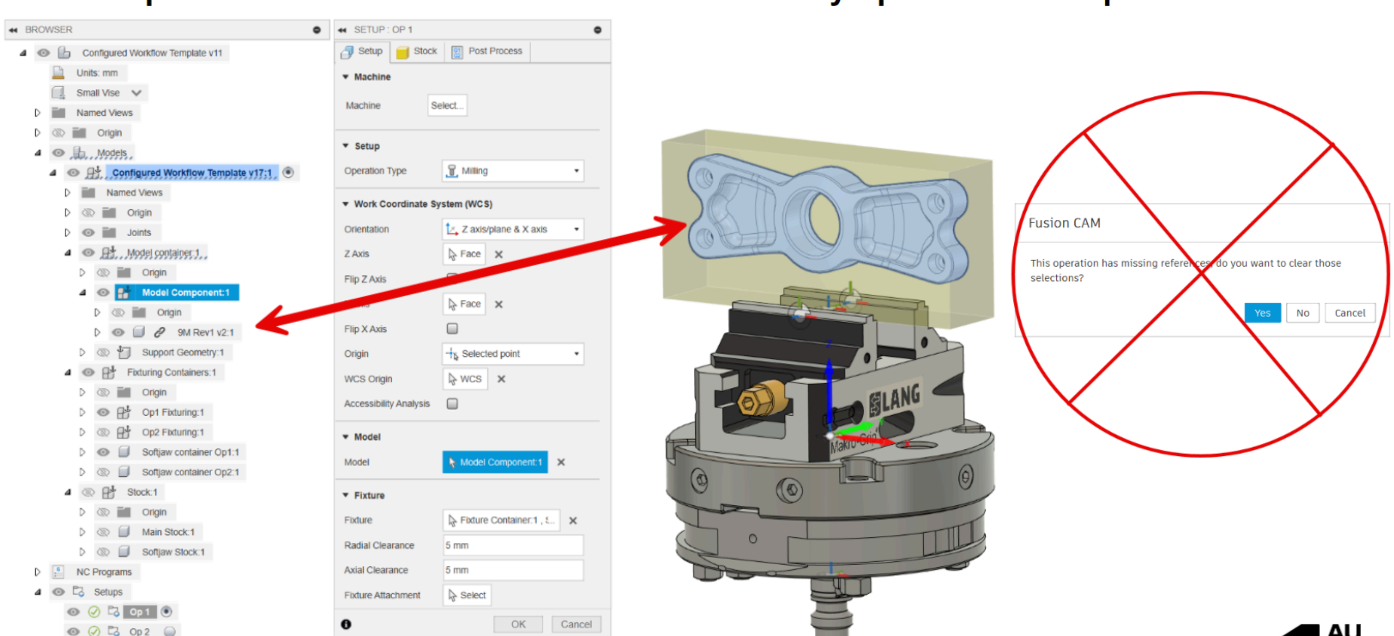
Any **Component** in Fusion is a container for other geometry or CAD objects. It can contain other components, or simply be 'empty' with only its Origin point (XYZ Axes, and XY, XZ, YZ planes).

Using a **component container** for your **Model** is a powerful technique for maintaining CAM model selections. Instead of selecting the model itself in your CAM setup you select the component container that holds the model.

If we do not use a container for a CAM setup and directly select the model when we try to replace, or edit the model - you may see this warning that those selections have gone missing.

Every program is a template if you use a Model Container

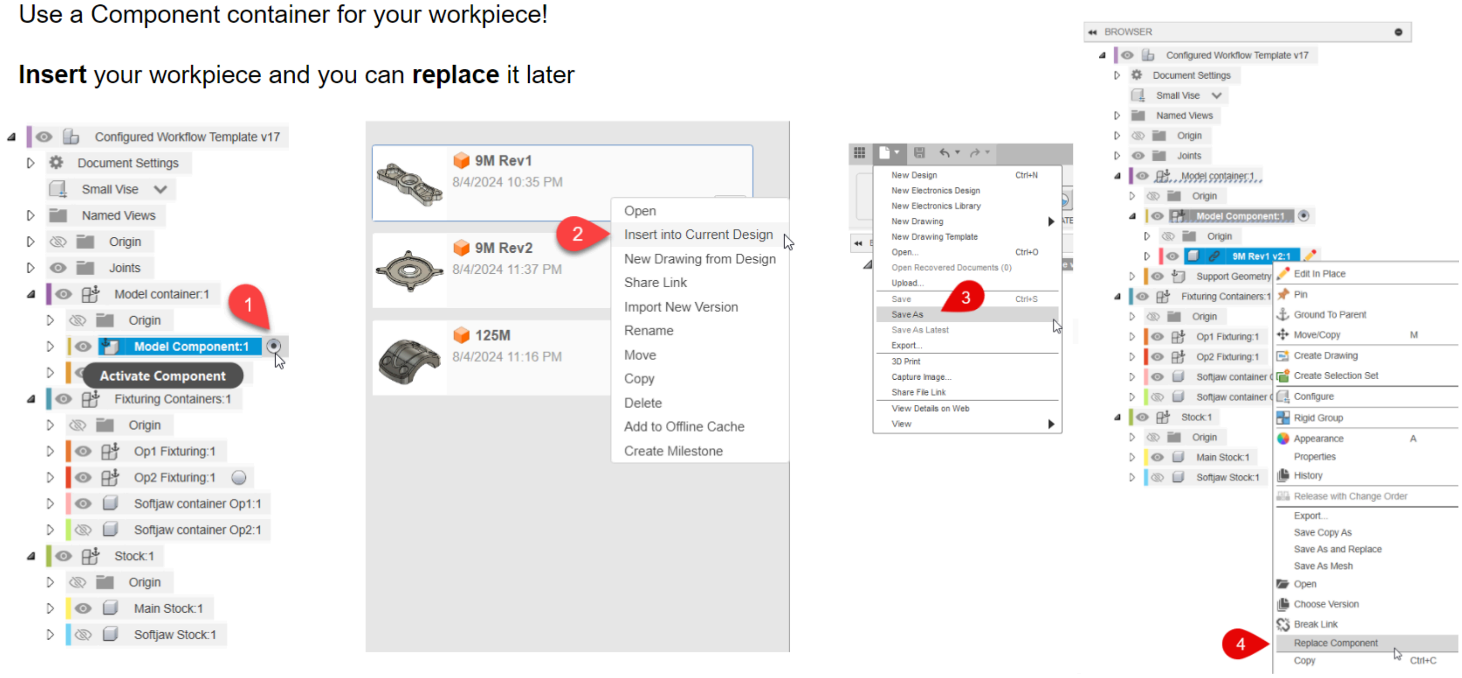
The Component Container selection will automatically update the Setup Selections



Every program is a template if you use a Model Container

Use a Component container for your workpiece!

Insert your workpiece and you can **replace** it later



Insert your model into this Component Container from the data panel it will become an **External Reference** with a chain-link or Flag next to the component icon in the browser.

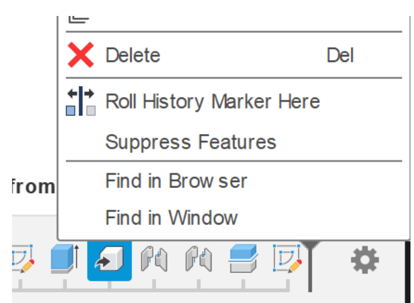
Any External Reference can be **replaced** with any other External Reference in the Right-click context menu.

When you **replace** the model within the container (to machine a different part using the same toolpaths), the CAM setup retains its selection of the container's contents
.
The setup doesn't care what's inside the container, only that the container itself is still selected.

If you begin programming with a simple Model-Container, **every program will become a Template** for similar geometry, or future revisions of that part. Simply open the old design, **Save-as**, Replace the Model and begin programming using the same setups and toolpaths to capture and re-use the process knowledge stored in the Template

With a couple intentional iterations, you will refine a starting point that has most of your repetitive programming already completed. Ready for you to focus on the finer details.

You may be tempted if your design is local to fusion to **Derive** in the Model for machining into this container.
You must make the component Container active when you issue the Derive command as you cannot move a derived component in the browser.
Unfortunately a Derived Component cannot be replaced, so It will need to be manually deleted from the timeline before inserting a new revision of your model to re-use CAM programming work.



Fusion Joints 101:

We use the **Joint** tool in Fusion to assemble our different workholding to match the assemblies we build in the physical world. Most Joints for workholding will be **Rigid, Slider,** or **Planar** types

- You can only create a joint between two components. You may pick the contents of a component however fundamentally **you need a component** to create a joint.

- The input to a joints can be a Vertice, Edge, Face, Component Origin, or **Joint Origin**

- Joints follow a **From-To selection order** where the first selection will move to the second selection. This is not critical to follow the selection order once everything is fully constrained, but it's good to know.

- **Rename your joints!**
  I recommend following the from-to naming convention: *ie*; 'Vise to Clamping unit' or 'Stock to Vise'

  Naming your joints will help you debug, and others understand your design much easier.

- The joint feature will be listed in the timeline, and in the browser.

  The Joint will place itself inside the browser at the highest common parent component between the input selections. This means once a joint is created it cannot be edited to select a different component than was initially selected.

  This limitation is why we must build a framework of 'Typed files' if we want to replace one of the joint targets with a new component.

- Fusion will try to automatically repair a joint when the model changes, by searching for that same Vertice, Edge, Face, Component Origin or Joint Origin by their *EntityID* a value only available to users through API tools like [Fusion Utilities]

  If the Entity ID cannot be found because your model timeline changed, or because you added a filet to the target edge, or replaced the entire component the joint will not be able to compute and will be broken in 'Error' or 'warning' state. This leaves the last known solution cached in the timeline. Creating further issues downstream.

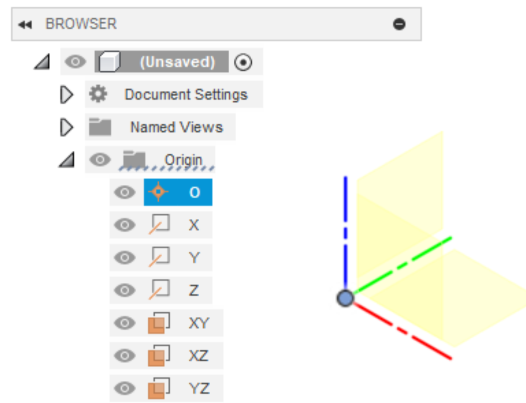
- Rigid, Slider, and Planar joints are all the same, they simply define how many Degrees of Freedom the joint should have constrained.
  - Rigid - all 6 Degrees of freedom are constrained
  - Slider - 5 Degrees of freedom
  - Planar 3 Degrees of freedom

Mastering Joint behavior is critical if you use **Replace** on components that - I find [Fusion - Assemblies Master Class by Kevin Schneider] very helpful to understand Assembly best practices in Fusion for predictable Joint behavior

Save-as lineage:
I used [This video] during my presentation to demonstrate how save-as can be used to make multiple versions of a design that can be replaced for one another without joint failure.

Replacing two different components will cause any dependent joints to fail unless the incoming and outgoing component have a shared EntityID, or common reference found in both designs.

A simple way to ensure your joint never breaks from replacing a component is to use the **Root component origin** as the snap selection for the joint.

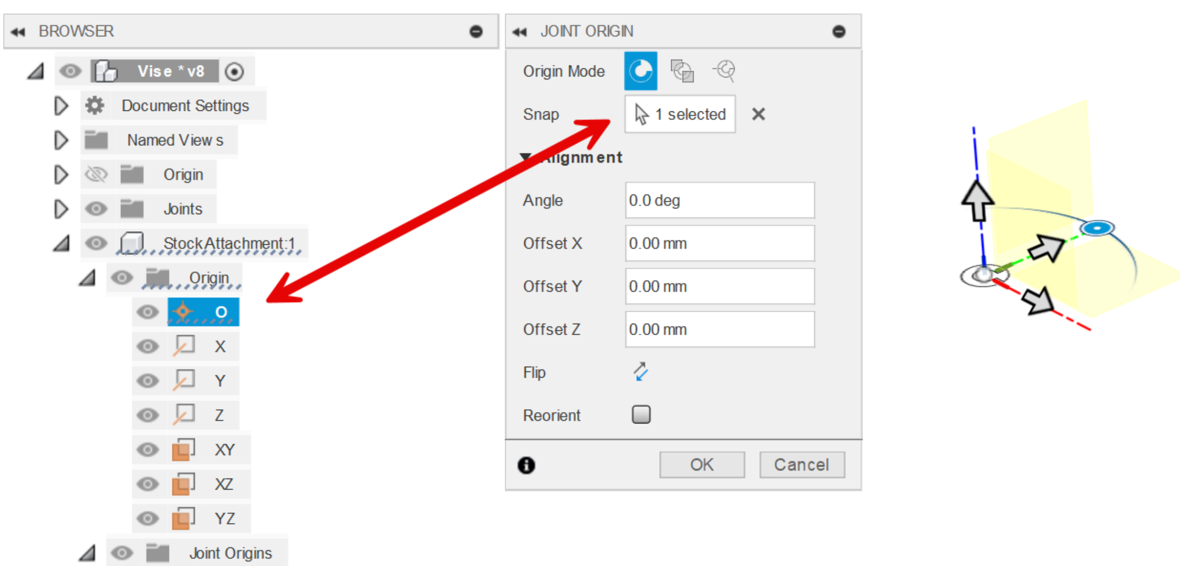


Since all components have this Root origin the joint can find a matching EntityID in the incoming design. And the joint will maintain its integrity.

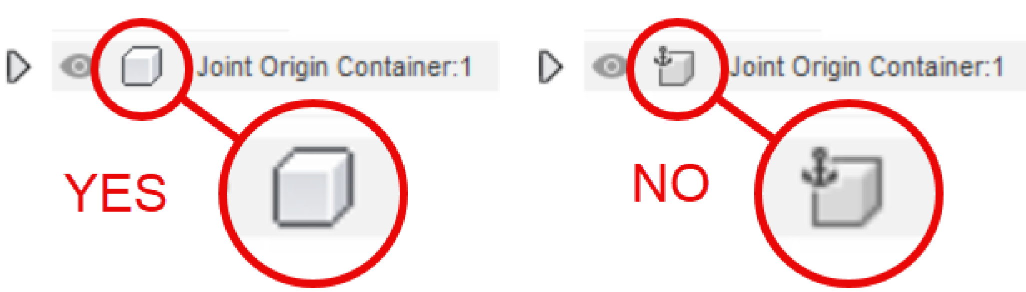
Joint Origin Container (JOC) component
Since we're making an attachment framework we need more flexibility than just the Root component origin offers.

We will **create a new component** at the top level of the model hierarchy. This will be an empty component aside from a **Joint Origin** inside it.

Create the joint origin with the snap on the coordinate **origin** of JOC. This puts the joint origin inside this container.
This will be a robust reference for any higher order assemblies.



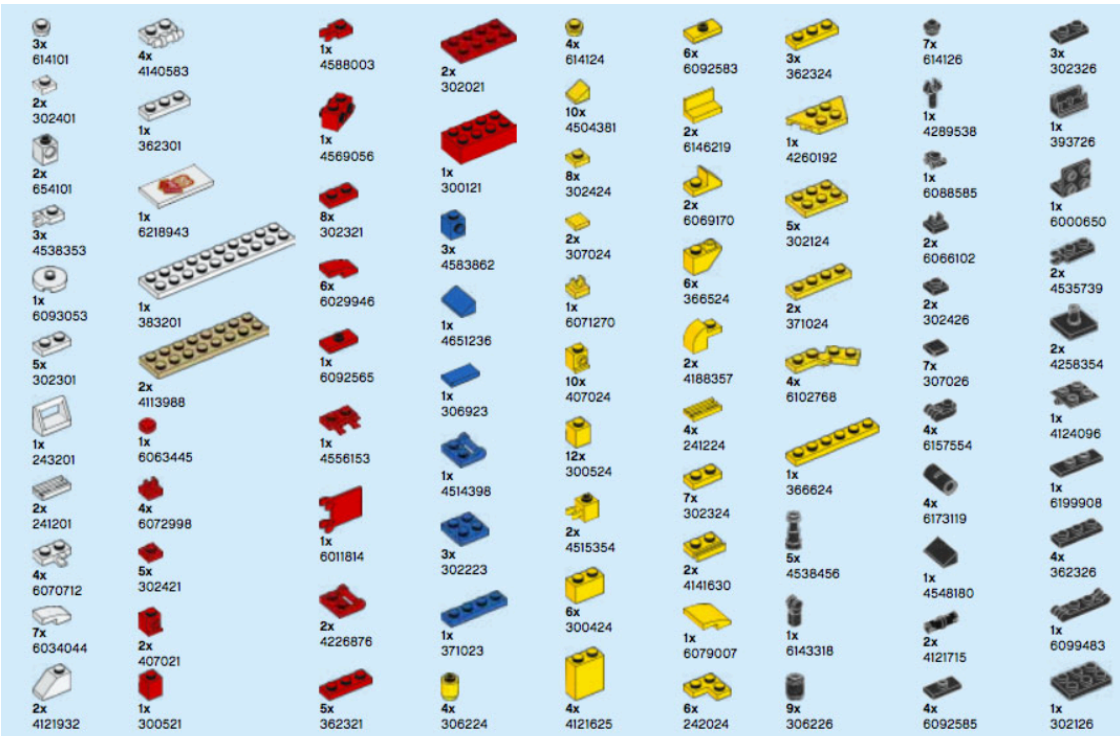
Make sure The Joint Origin Container component is not grounded to parent



Attaching the JOC to the your geometry

Now we need to mount the Joint Origin onto our model.
Since the JOC component is unconstrained, make a new joint. Pick the new joint origin as the first snap reference, and pick any other point on your geometry as the other reference.
This constrains the new joint origin to be fixed to the object.

Think of it like a LEGO stud.



A piece may have 1 stud or many, creating a standardized way to attach two items with positional context.

Regardless of the shape or size of the geometry in-between the studs we know things should only attach together on specific points with discrete orientations.

We are doing the same thing with Joint Origins and their JOC.

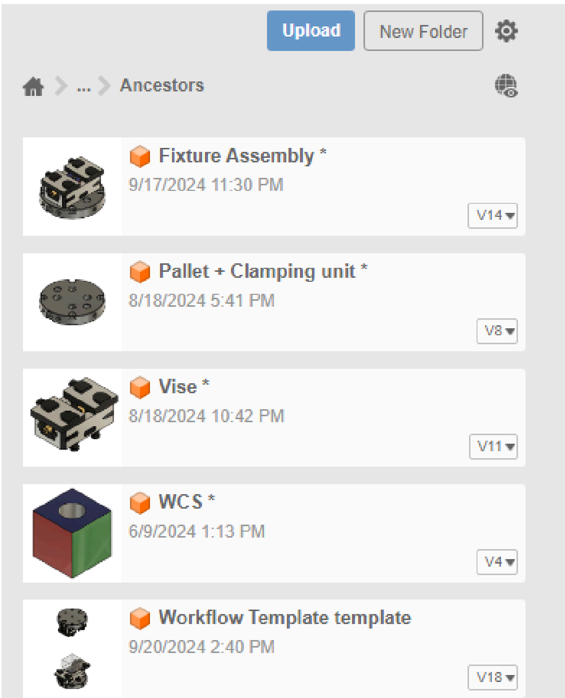
The Joint origin is the stud. The JOC component is the container. Since the JOC is an unconstrained component it can be joined *onto* any geometry you need to put a standardized attachment point on.

With save-as lineage this design can be duplicated, where we are free to edit, or entirely delete and import new geometry, joining on simply leaving the JOC in the design if unused.

Now, we can **Insert** this design into an assembly and any associations to that Joint origin or JOC will be repaired if the design is replaced with a related version that has been saved-as from a common ancestor. Effectively making replaceable, typed parts.

Section 1: Replaceable Fixturing Assembly - WCS, Vise, and Clamping unit.

Build your own Fixture assembly with your team's CAD using my framework available on Github



Download the *Workflow Template template.f3z* and upload it into your Fusion Team hub

Consider uploading the framework in a folder that you protect with read-only permissions.

These are the '**Ancestor designs**' that we will be creating a save-as lineage from.

We can replace similar designs with one another just as we do on our machines.

This works if we always keep the JOC and Joint origins in these designs, **if any JOC are deleted accidentally they cannot be recovered.**

Respect your Ancestors.

The **WCS, Vise,** and **Clamping unit** designs are the building blocks for us to build a flexible **fixturing assembly** which we will insert into Templates for CAM programming.

Consider them to be **Typed files.**

Each of these documents has specific Joint origins in component containers to create robust attachments between them, and any higher level references a CAM programmer needs to make.

You simply need to download these, make copies of each type with **save-as,** strip out the placeholder geometry, import your own geometry, and attach the Joint origins onto your geometry.

The **Workflow Template Template** contains two instances of the **Fixture Assembly**. All hooked up to CAM setups that are pointing at the **WCS**, **Vise**, and **Clamping unit** designs respectively as a starting point for you to build your own Templates that reflect your CNC programming workflows with your fixturing.

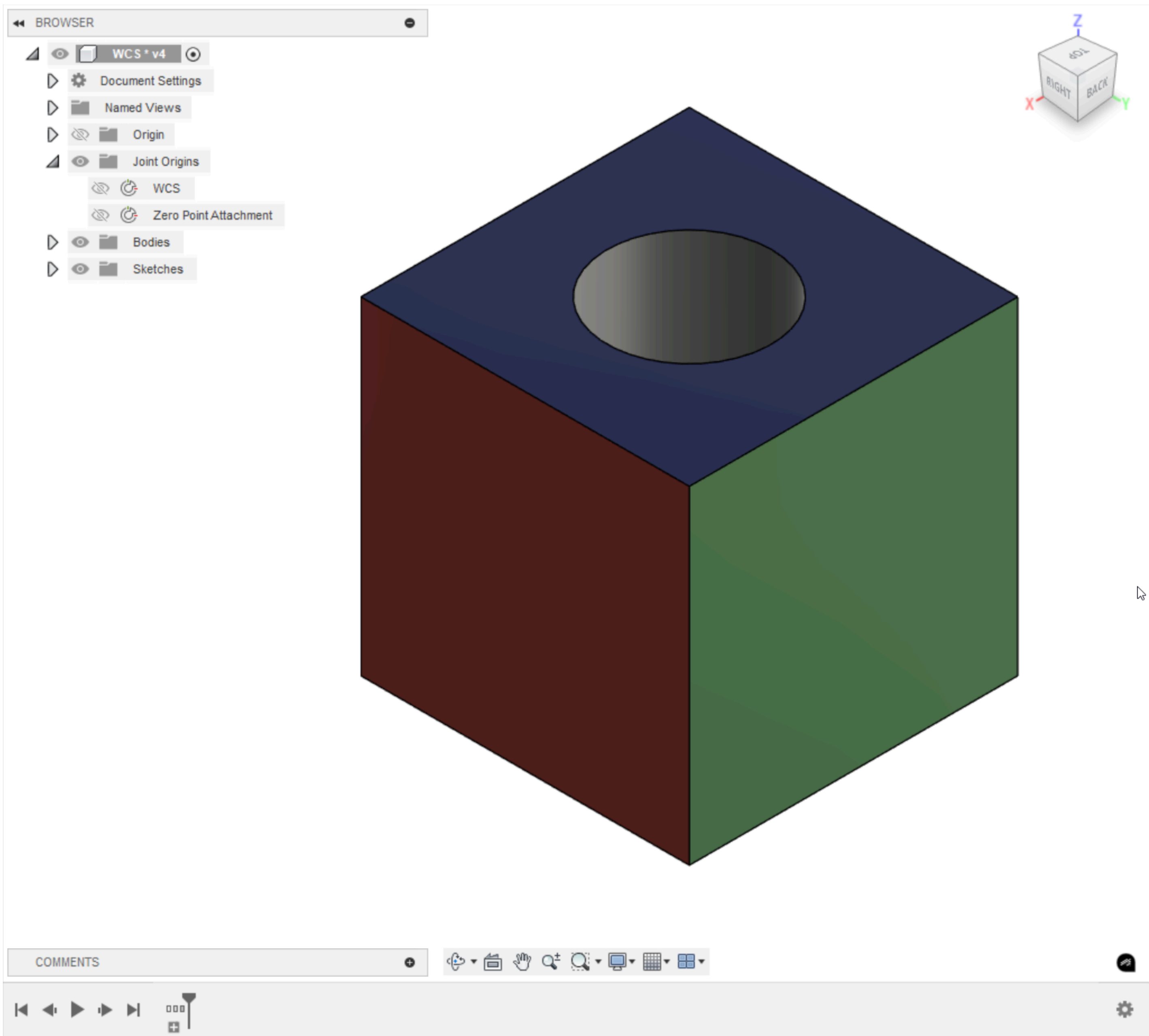
As each of these designs you make will be saved from the common Ancestor, you can replace any vise for another simply. Without concern of a broken timeline.

WCS - Work Coordinate System

This geometry is used for the Setup coordinate system. To fully define a WCS we need:

- Z reference - *Blue Face*
- X reference - *Red Face*
- WCS point - *WCS Joint Origin*

We Join this geometry onto the Clamping Unit with the *Zero Point Attachment* Joint Origin

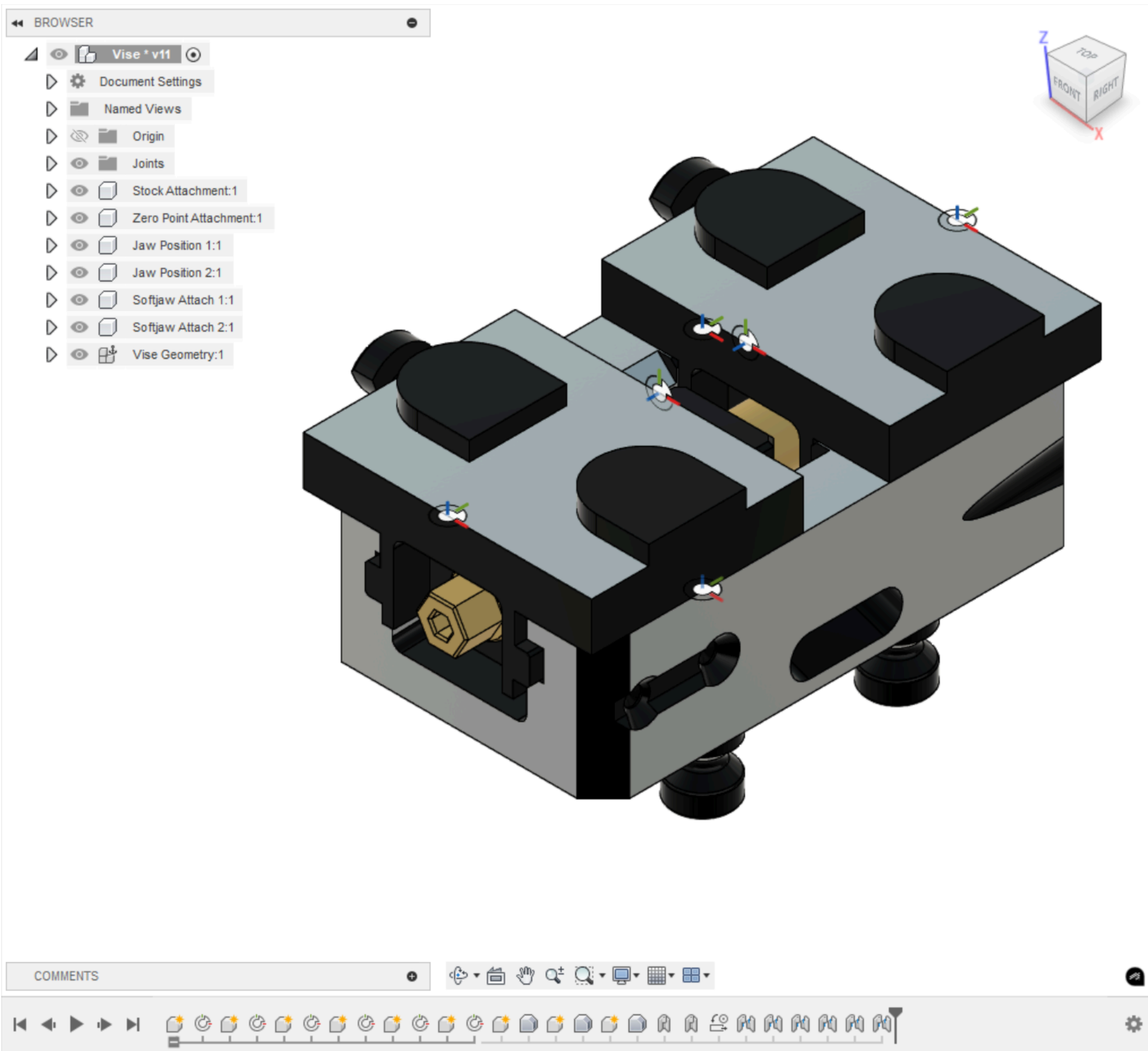


Vise

Stock and Soft-jaws need to be local to the CAM programming in the Template document so they can be parametric; editable geometry based on our workpiece size and shape.

The stock is joined to the vise on the ***Stock Attachment*** and the jaws close onto the stock with their respective ***Jaw Attachment****.* Soft Jaws can be joined onto the ***Soft Jaw Attachment***

The Vise is Joined to the Clamping Unit with the *Zero Point Attachment* Joint Origin

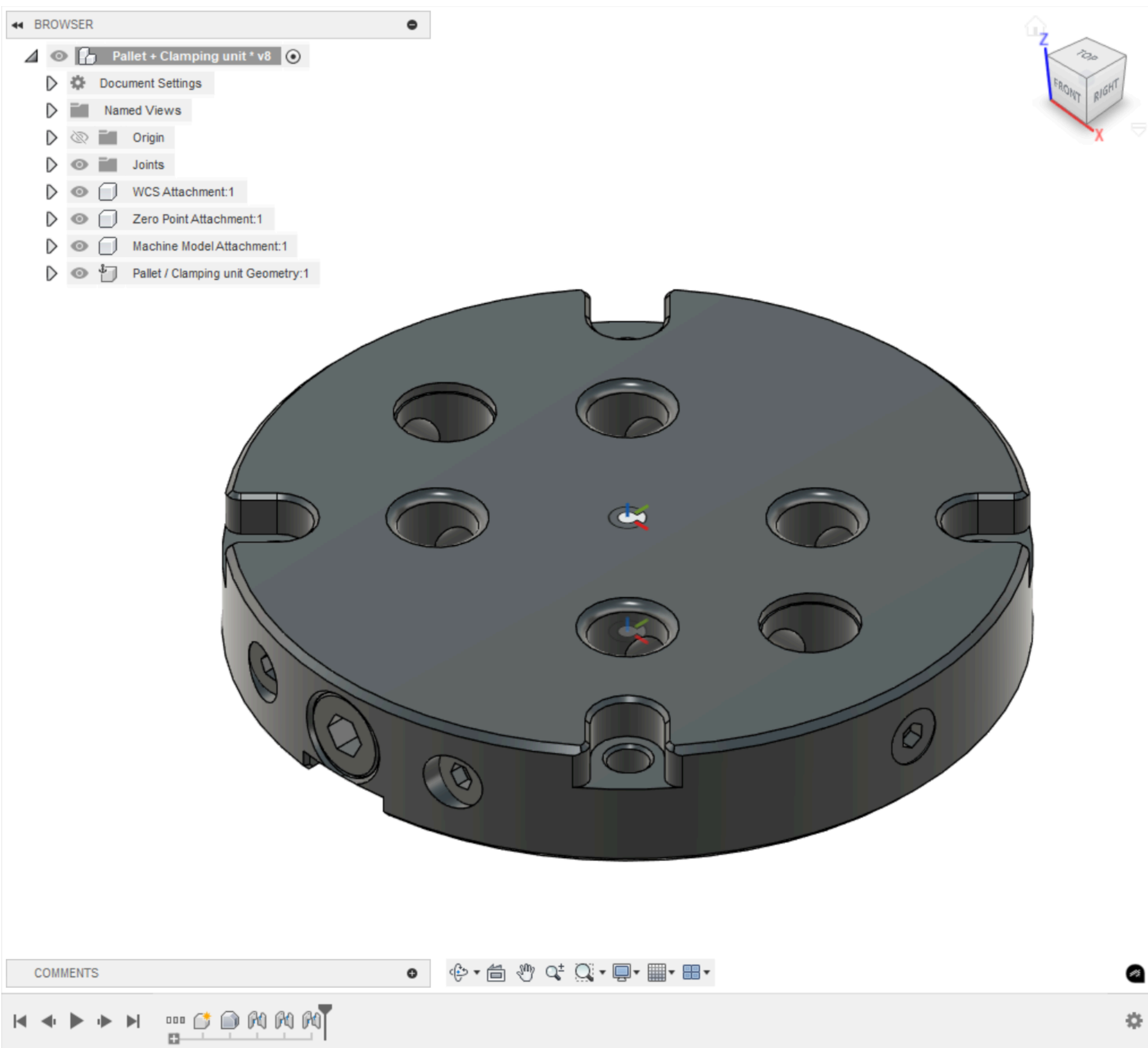


Pallet/Clamping unit

The Clamping unit connects the Vise and WCS as a '**zero point**' system.
It may be an assembly of CAD designs to represent all of the fixturing you need to consider in simulation that isn't a vise.

If you simply use a Vise on a machine without any pallet or clamping unit, you can leave the JOC attachments and remove the geometry.
Then **Rigid Group** the root component, and including all children and the JOC components to fully constrain the design.

- WCS Attachment - Joint target for the WCS cube
- Zero Point Attachment - Joint target for the Vise - **there may be many Zero Point JOCs**
- Machine Model attachment - Location of Machine Simulation model selected at Setup.

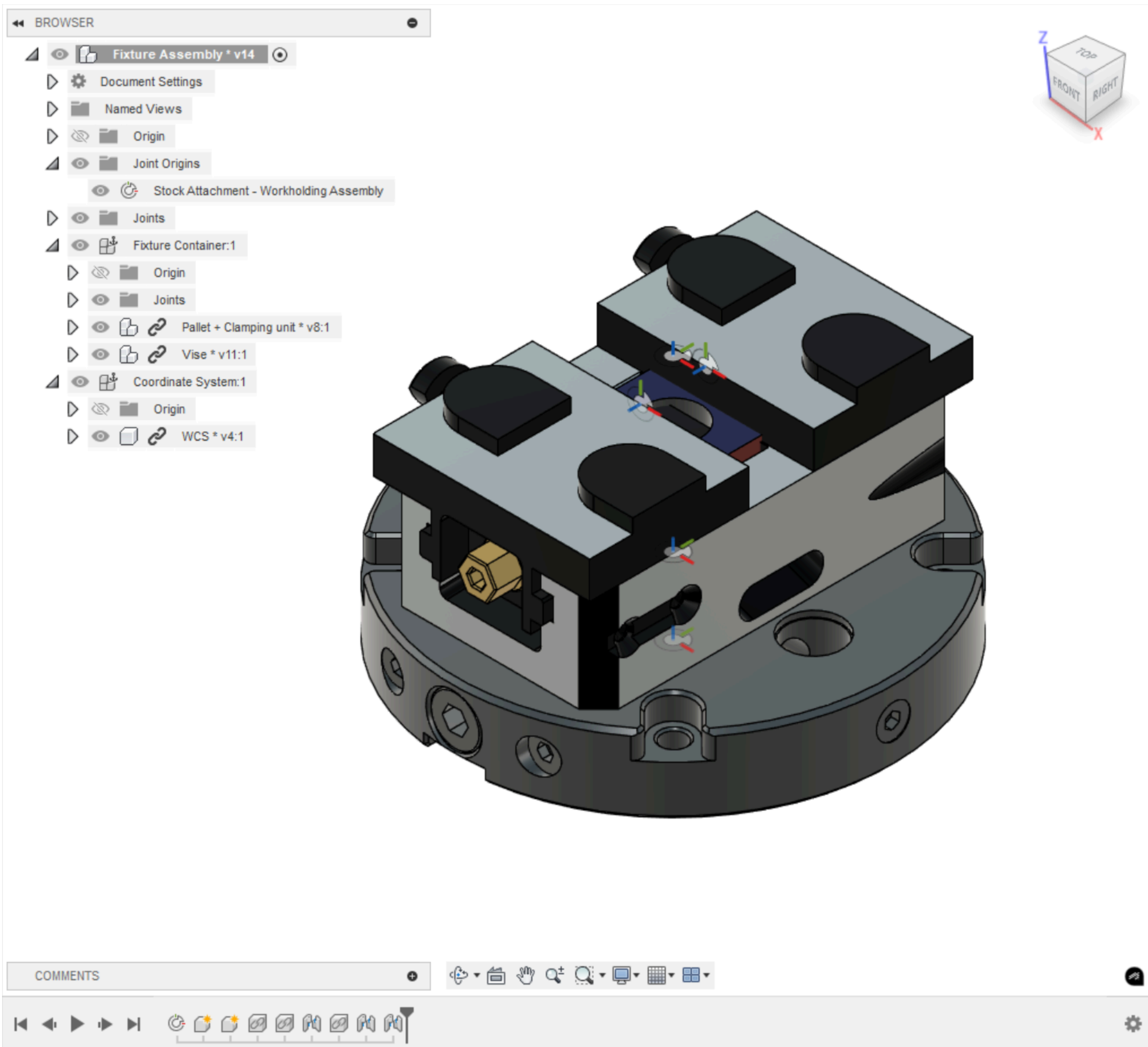


Fixturing Assembly

The Fixturing assembly brings the Vise, WCS, and Clamping unit together into two ‘containers’.

Fixture Container component - Is selected by the Setup as ‘Fixture’, and all of the contents will be considered in simulation for collision detection and avoidance.

Coordinate System - Is a component container to ensure the WCS is always found by the setup if the Fixturing Assembly is replaced.



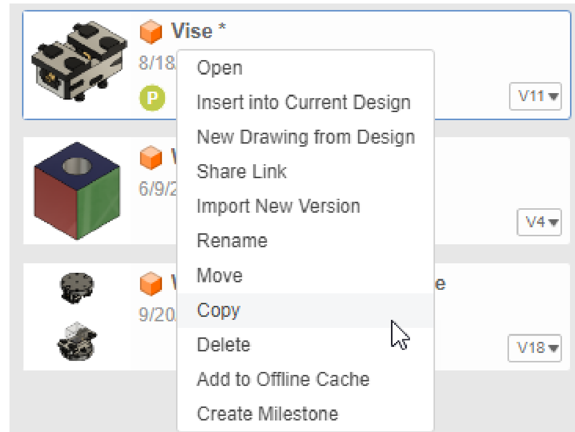
Importing your vise into the framework:

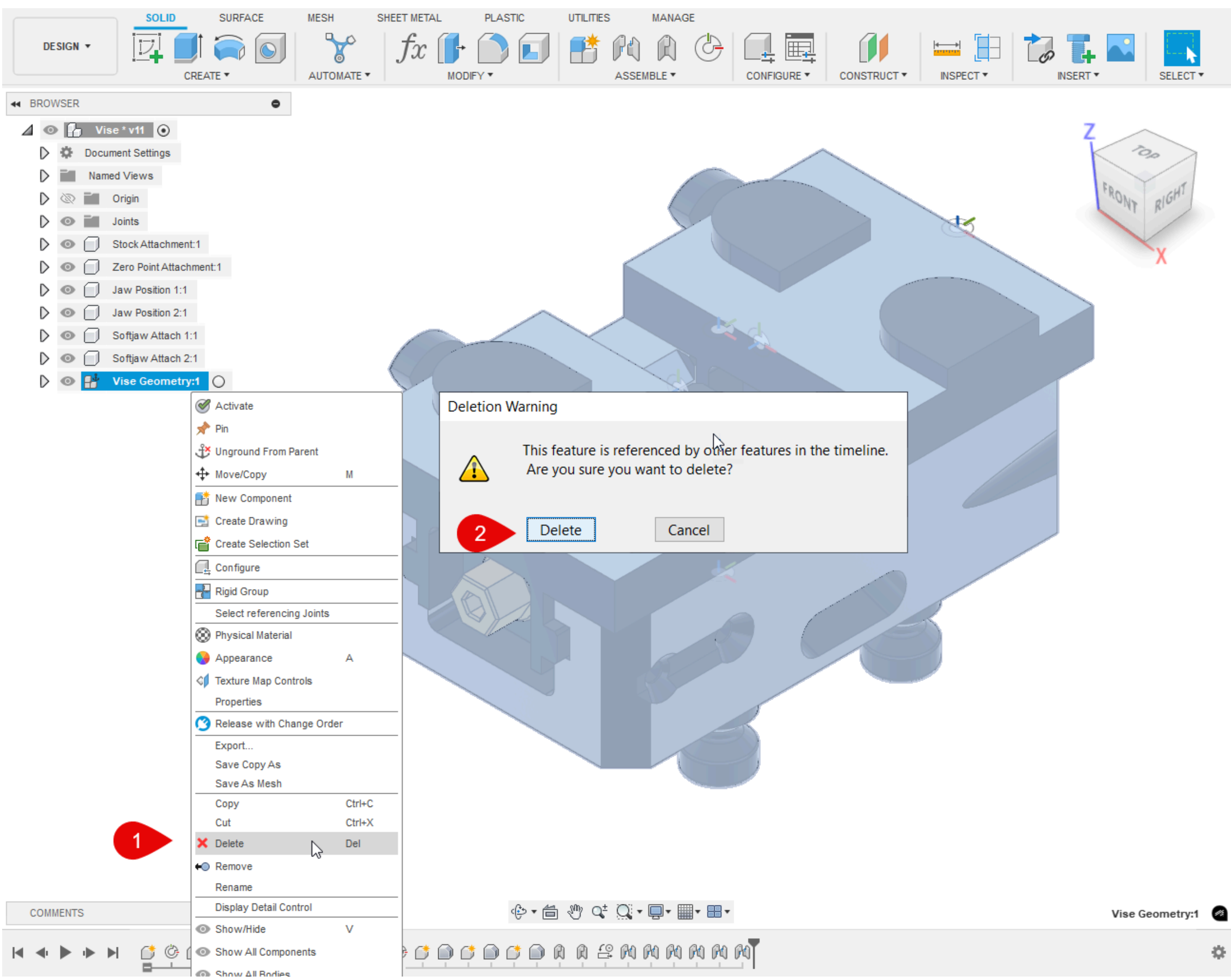
1. Begin by making a **Copy** of the Vise design.

2. Delete the placeholder geometry, you will be warned that downstream features will be affected.

   So long as the JOC Attachment components and Joint Origins remain in the design you can remove everything else.

.



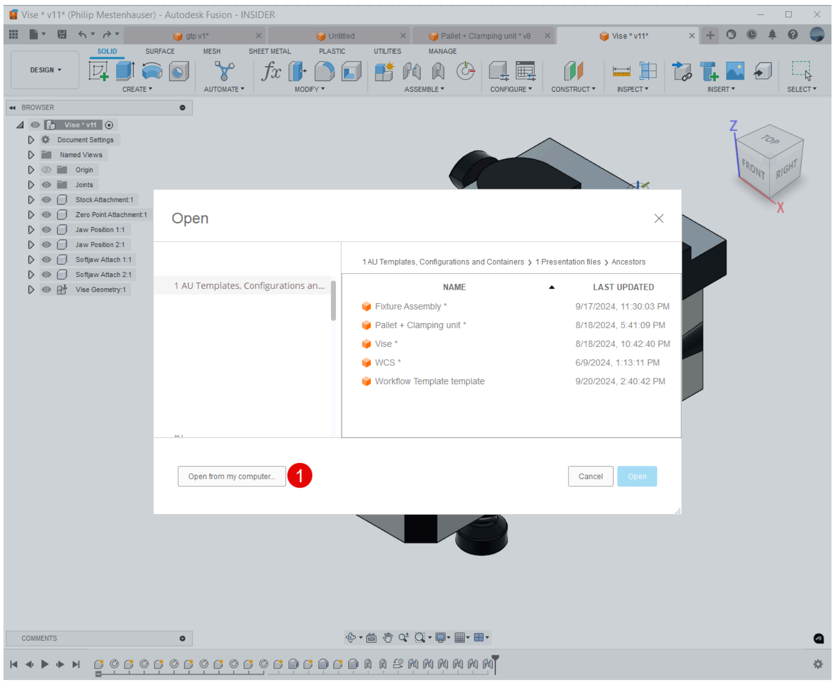


Open, simplify, and prepare data to import into the framework

3. Insert your existing Vise geometry into the Vise design.

You can insert your existing design as an external reference and **break the link**. Be sure to **delete any existing Joint origins** that aren't part of the JOC framework.

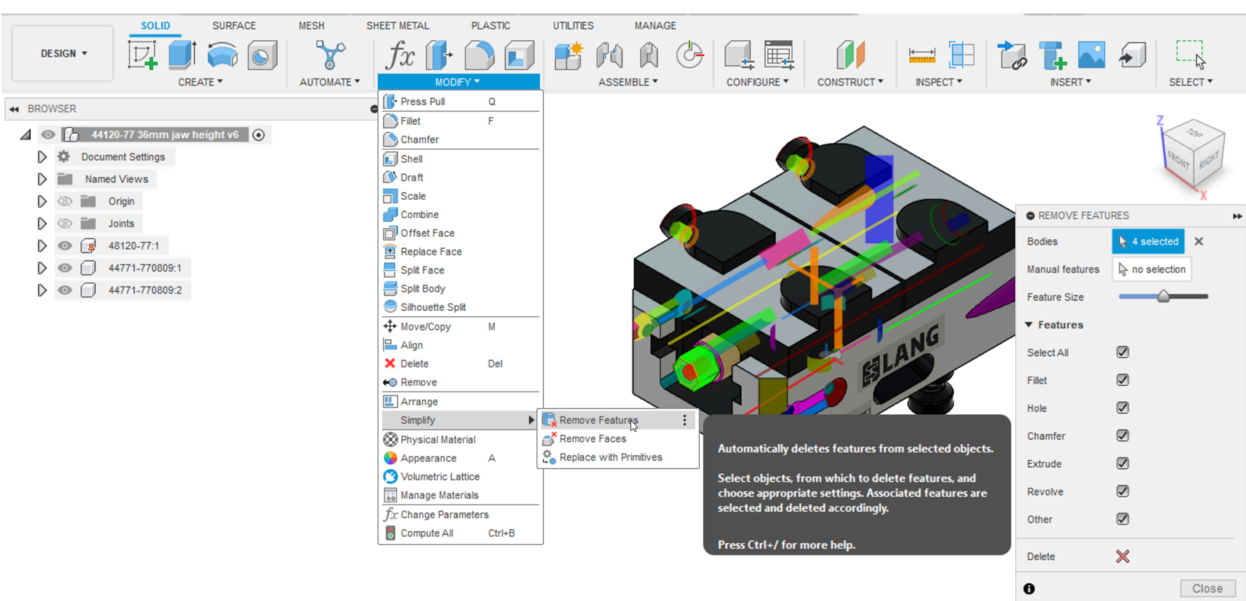
Alternatively, you can use the File>'Open from my Computer' to open new vise geometry from the manufacturer's website.



4. Simplify the design **within this temporary design.** We will copy it into the

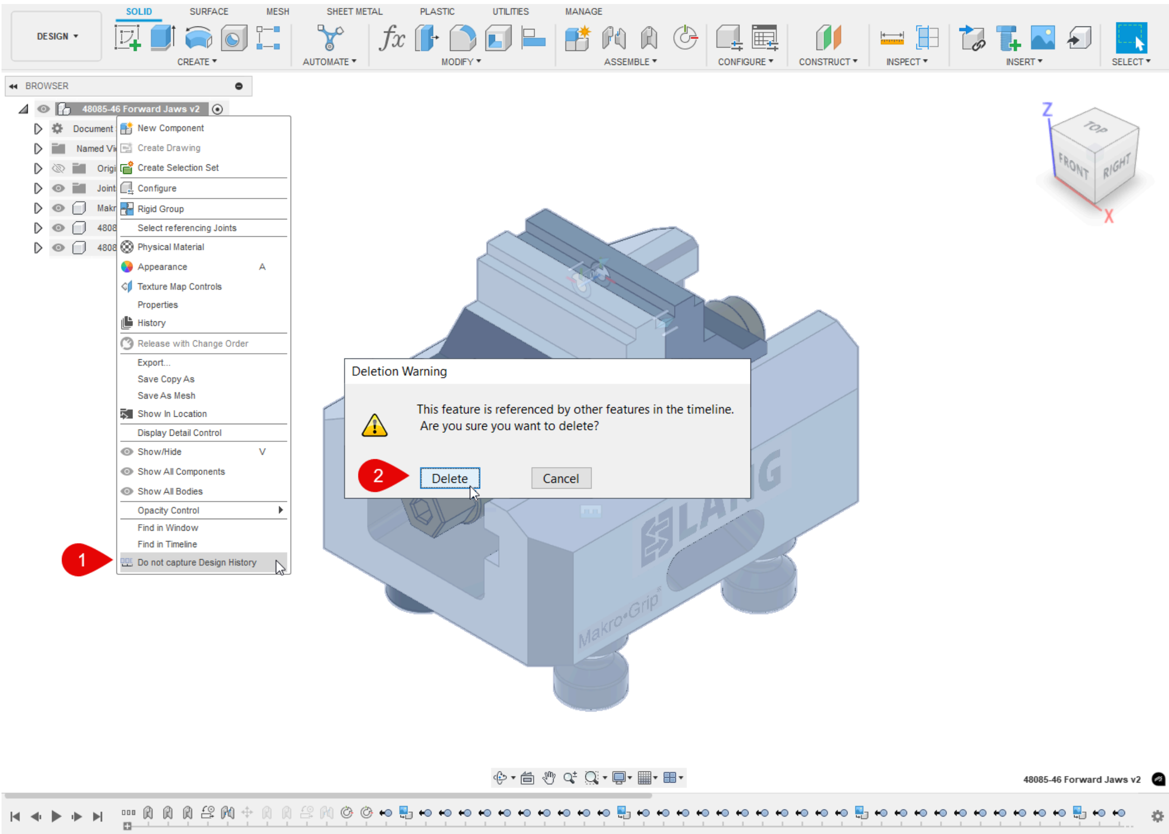
There is some benefit to keeping fixture CAD simple and reducing detailed CAD down to primitive shapes to avoid excessive toolpath computation and simulation time.

Using the **Simplify** - **Remove Features** command to reduce the number of faces is very efficient and effective. Note that this command is only available when the design history is active.

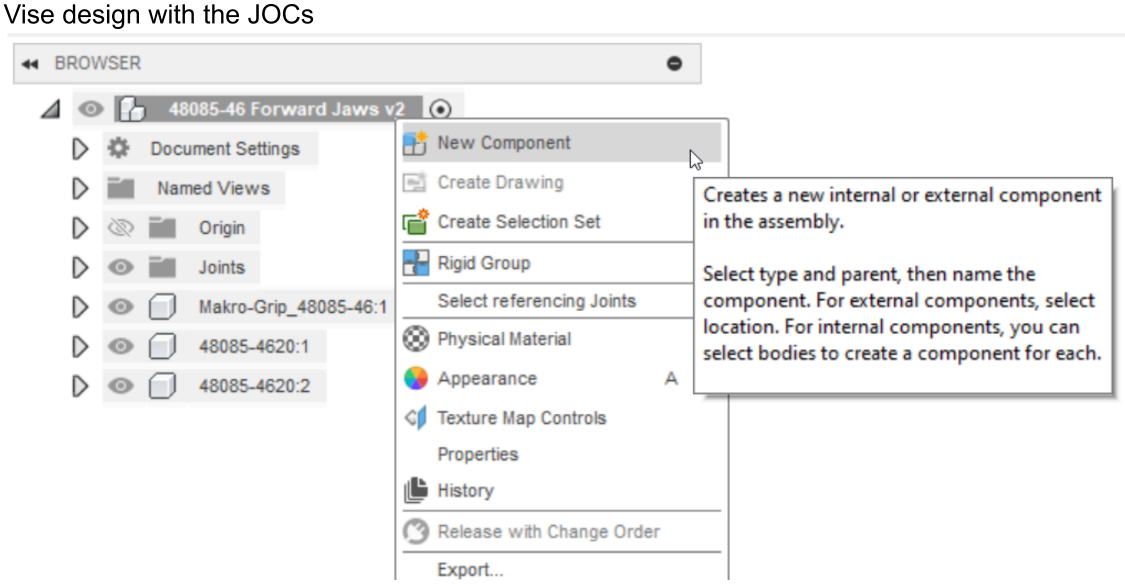


Declutter the timeline

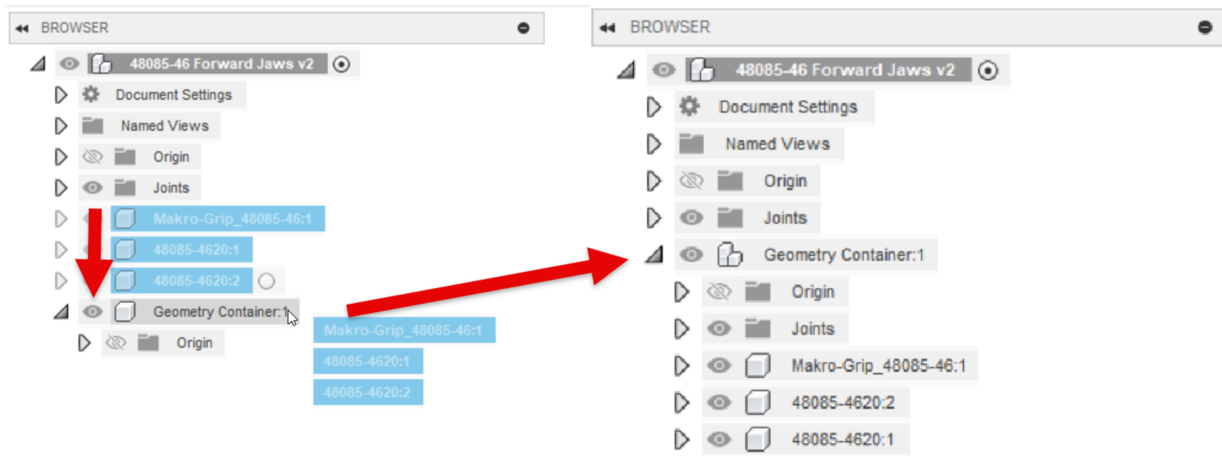
5. Once you are done simplifying your model, **turn off design history** to clean up the timeline
   Be sure to do this in a separate design from the Vise with the JOC attachments.



6. Create a **New Component** container for the Vise geometry. This will make it easy to copy into the Vise design with the JOCs

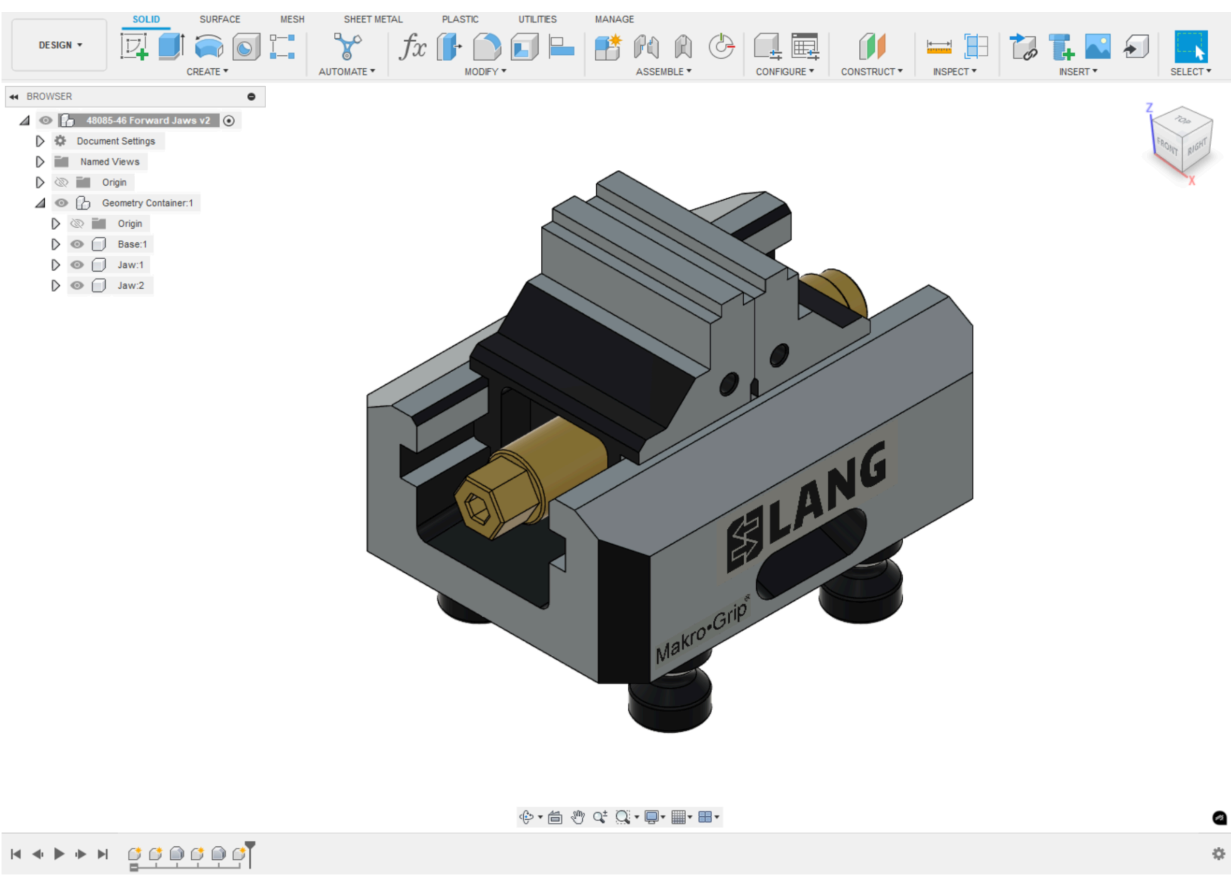


7. click and drag bodies into the fewest required components to create the motion of the vise/chuck. Move all of these components into the container component.

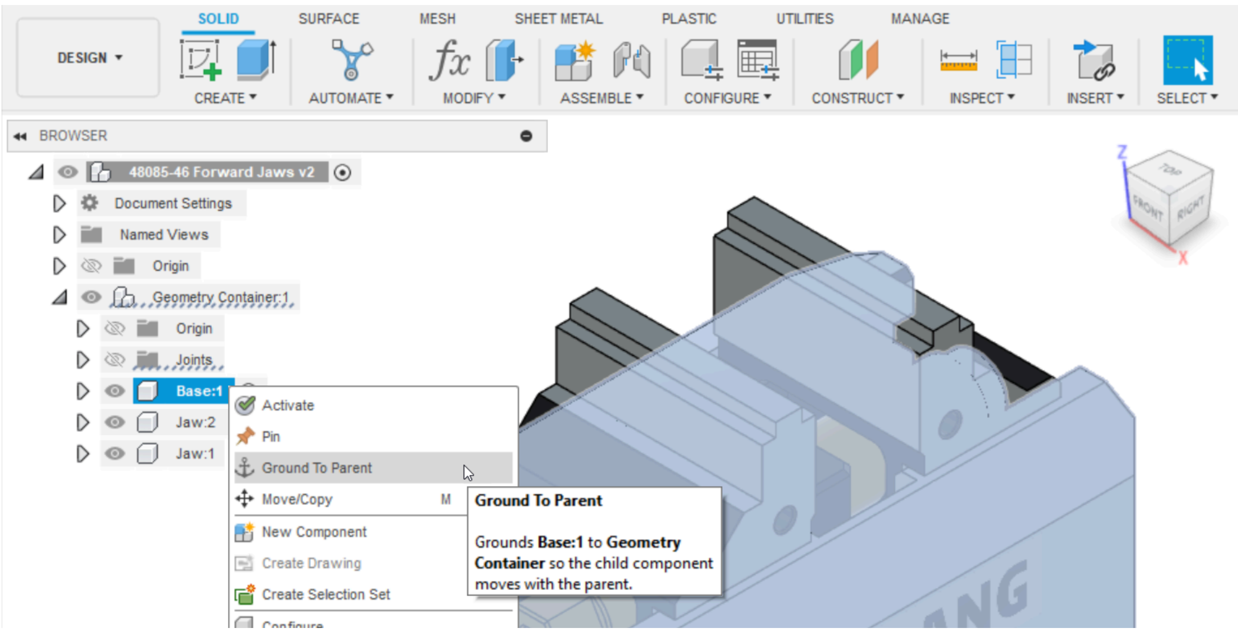


8. Use **Move**, and **Align** to position the **Base** and **Jaw** components relative to the **Origin** of this container if necessary.

9. Turn the design history back on, the timeline will be simplified to basic components, and Base Features for the geometry.

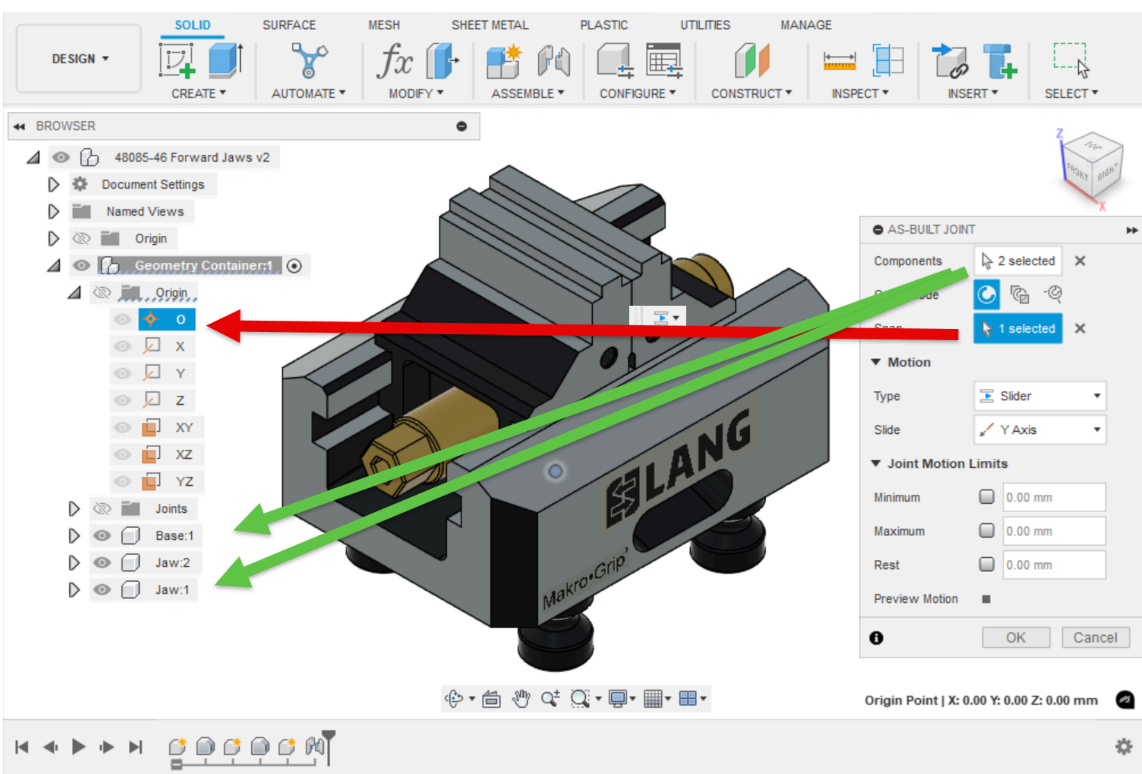


10. **Ground to Parent** the Base.
    This will constrain the base to the component container, and should be the anchor to fully constrain a self centering vise.



11. Apply the relevant motion joints between each component of the Vise.
    An **As-Built Joint** is a great way to apply slider joints between the jaw and base.
    The Joint is placed at the highest common reference in the browser- so be sure to select the Geometry container Origin as the 'Snap

12. Apply a **Motion link** Between the two slider joints.



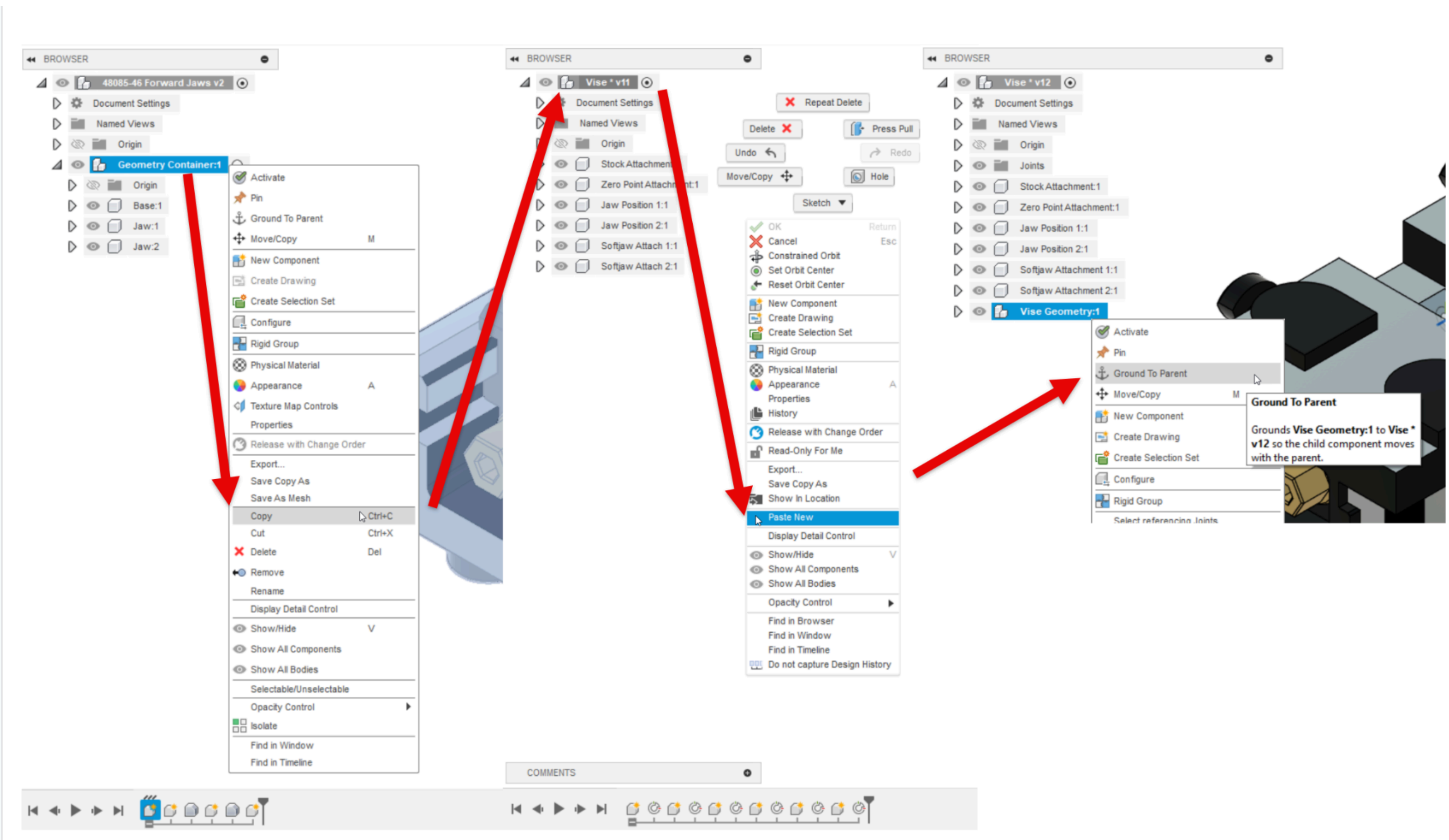
Ideally, all of this simplification and preparation is still within the source CAD prior to inserting it into the Vise type design with the JOC components
We still need to get the geometry into the Vise design we duplicated in Step 1.

13. **Copy** the geometry container.

14. Navigate to the Vise design we copied in step 1.

15. Right click on the root component, **Paste New** to grab the component container with all of the vise geometry, and place it in the **Vise** typed design.

16. **Ground to parent** this container of geometry

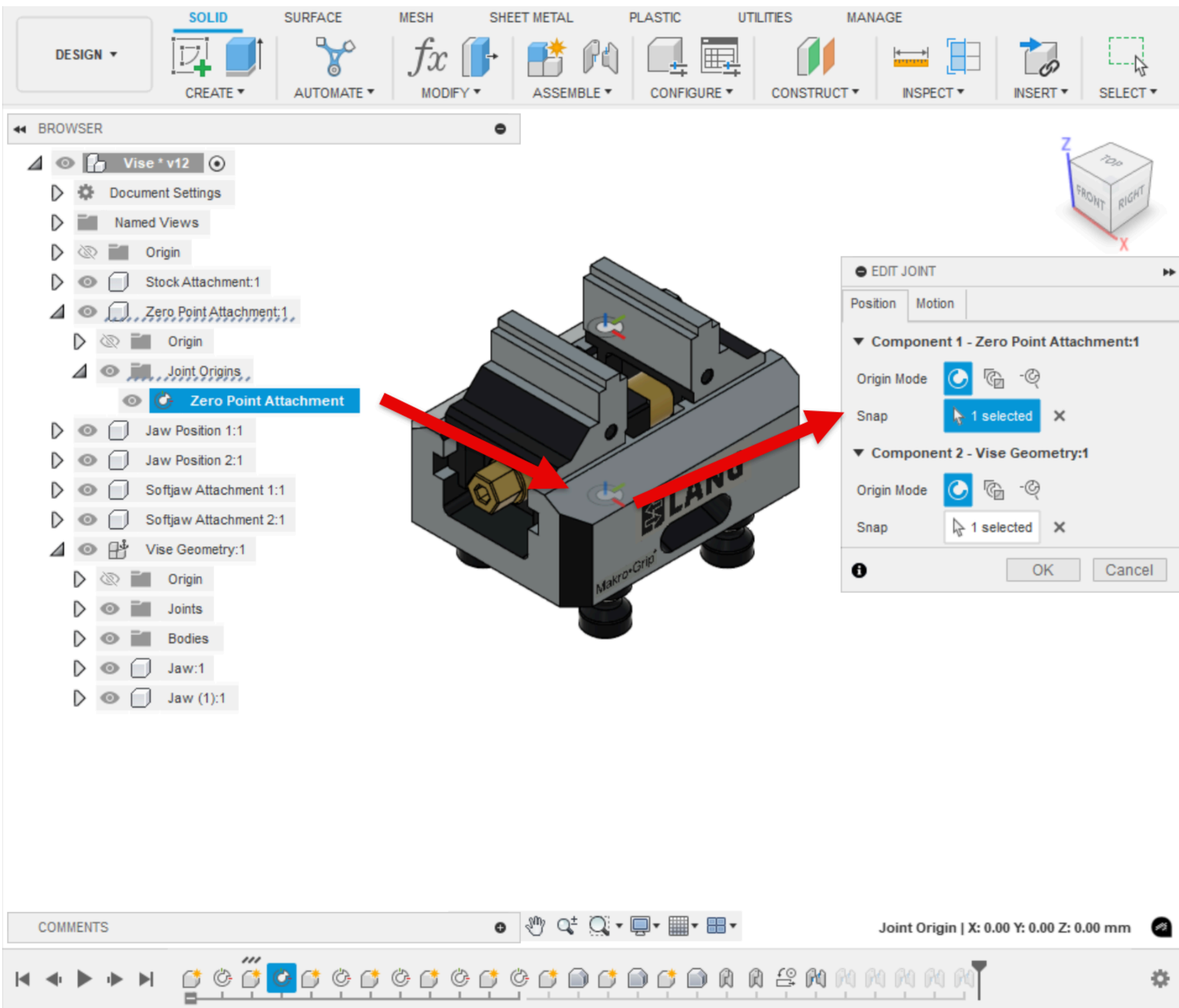


Rigging the Attachments onto the vise.

Now that the geometry is pasted into the design with the JOC attachments, we need to Join the Joint Origin attachments onto the functional locations of the geometry.

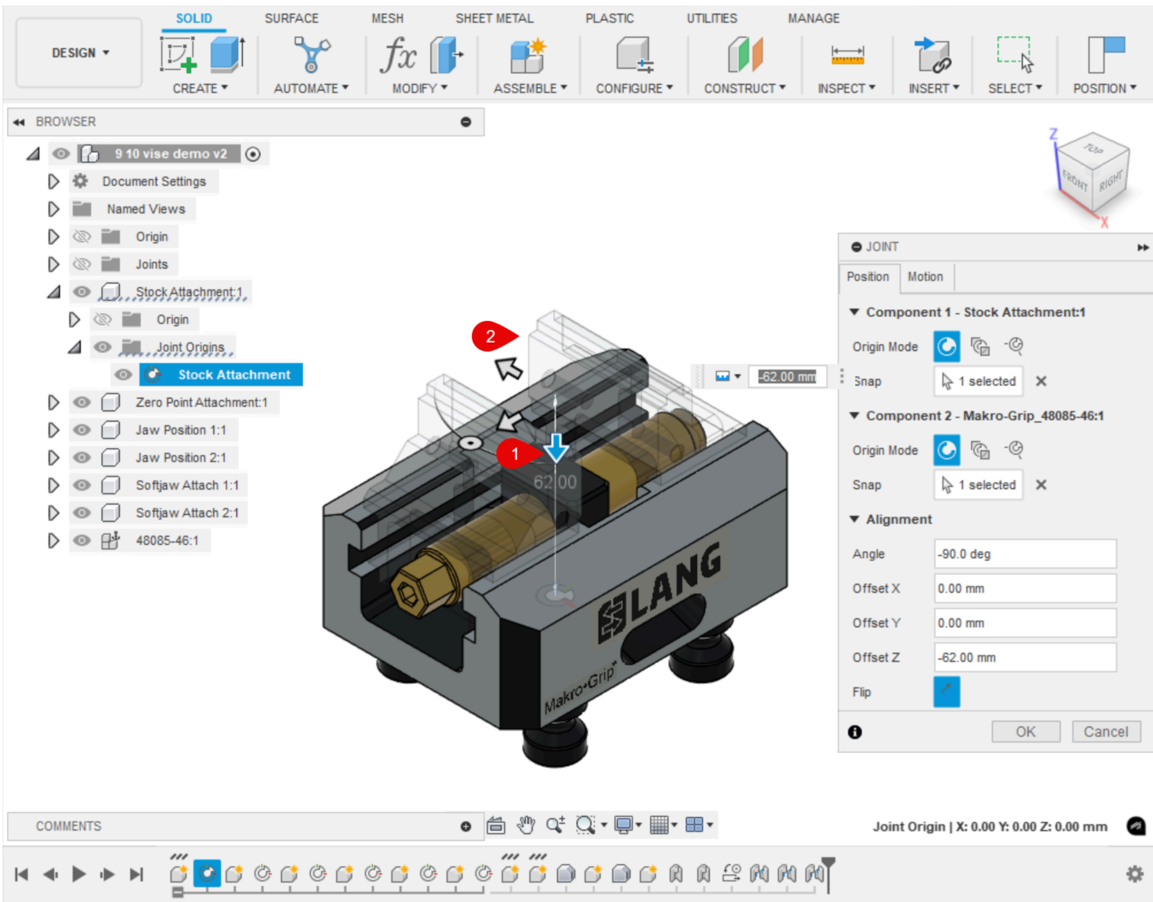
17. Use the Joint command to join the *Zero Point Attachment* **Joint origin** onto the ***geometry*** you imported.
    Be sure to select geometry for the second selection, and not the document Origin.

    The orientation of the Joint origin should match how you want the Vise to be mounted.

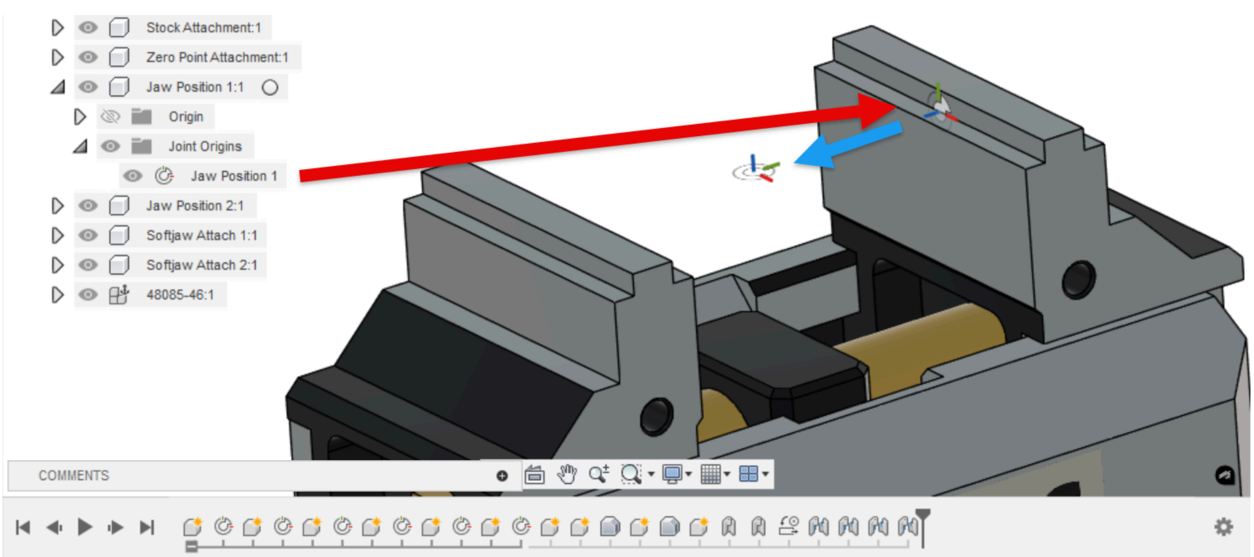


18. Use the Joint command to join the *Stock Attachment* **Joint origin** onto the *geometry* you imported, and apply a value to the *Offset Z* alignment.

A fast way to do this, is to click the drag handle to begin the 'input' of the corresponding offset value. Make your next click on the surface of your jaw-step where the stock is gripped, Fusion will 'measure' this distance, and apply the correct offset.



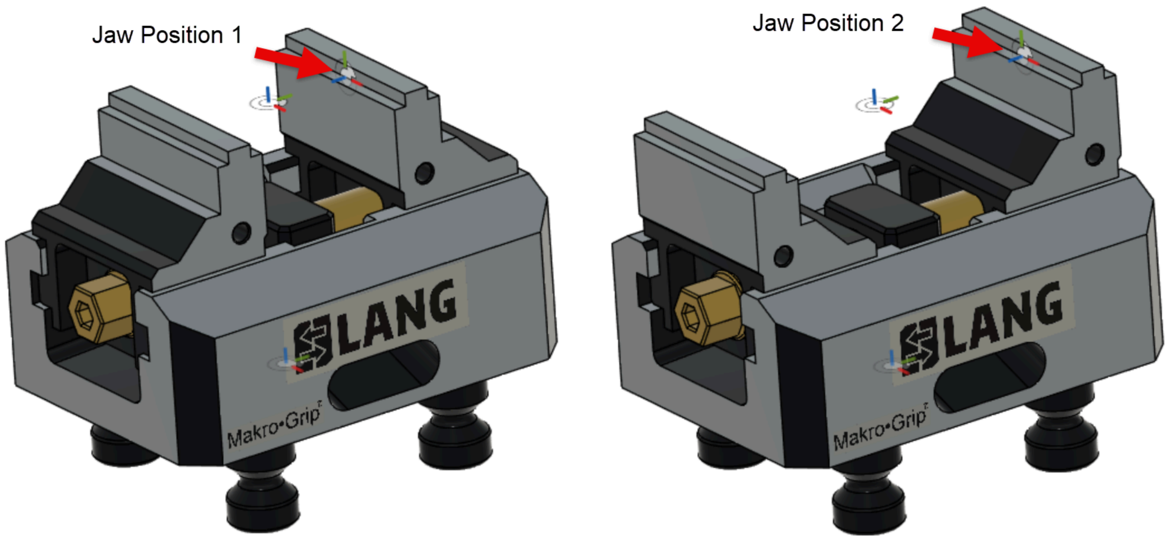
19. The Jaw Attachments need to join onto the Y+ Surface of the stock. The Joint origin Icon must point back to the Stock Attachment Joint Origin as pictured.



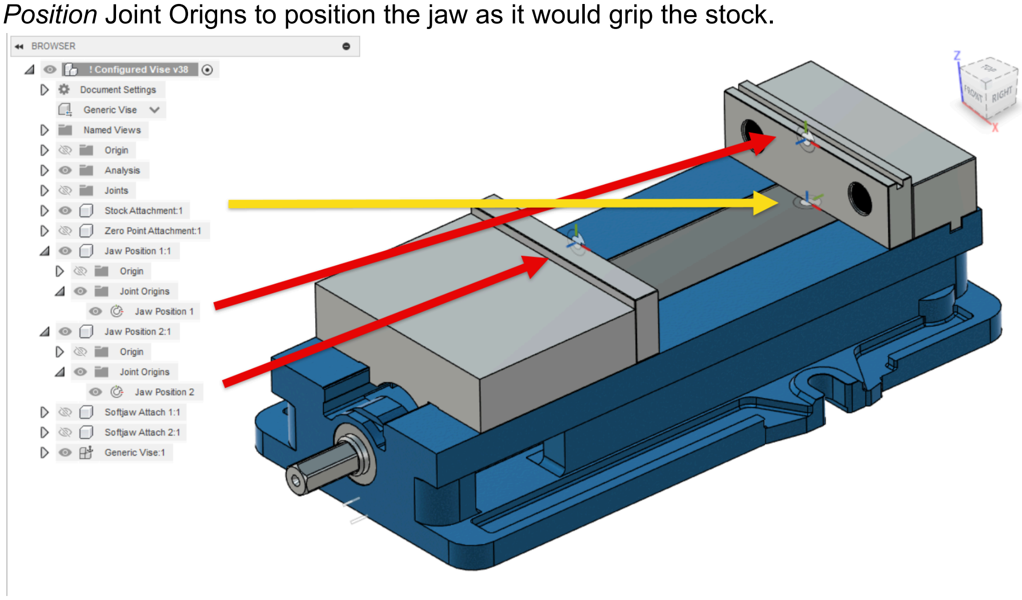
Jaw Attachments for fixed Jaw, and self centering vises

We have 2 Jaw Position attachments to support self-centering vises that can reverse the jaw orientation, and fixed-jaw vises.

For self **centering vises**;
At the Template level of the framework where we model our stock, we mount the stock to the vise with a **rigid joint** to the Vise *Stock Attachment,* and a **Planar joint** between the stock and the **one of** the *Jaw Position* Joint origins to position the jaw as it would grip the stock. The Planar joint is tolerant to any micro-misalignment in the CAD between the Jaws/Base.



For **fixed jaw vises**;
At the Template level of the framework where we model our stock, we mount the stock to the vise with a **Slider joint** to the **Vise** *Stock Attachment,* and a **Planar joint** between the stock and **both of** the *Jaw Position* Joint Origns to position the jaw as it would grip the stock.



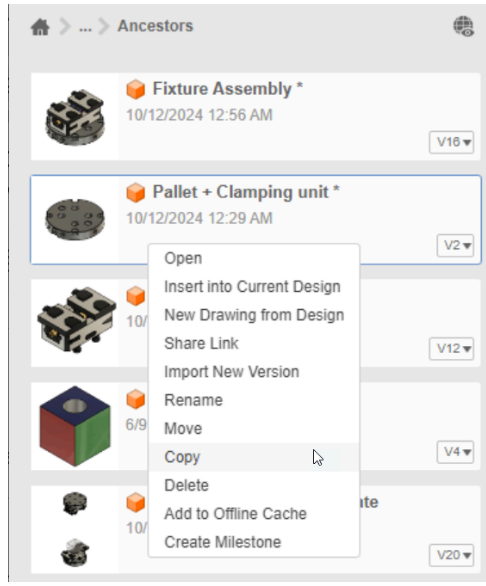
If a Joint Origin/ JOC is not relevant, it may be simply left unconstrained. *do not delete it.*

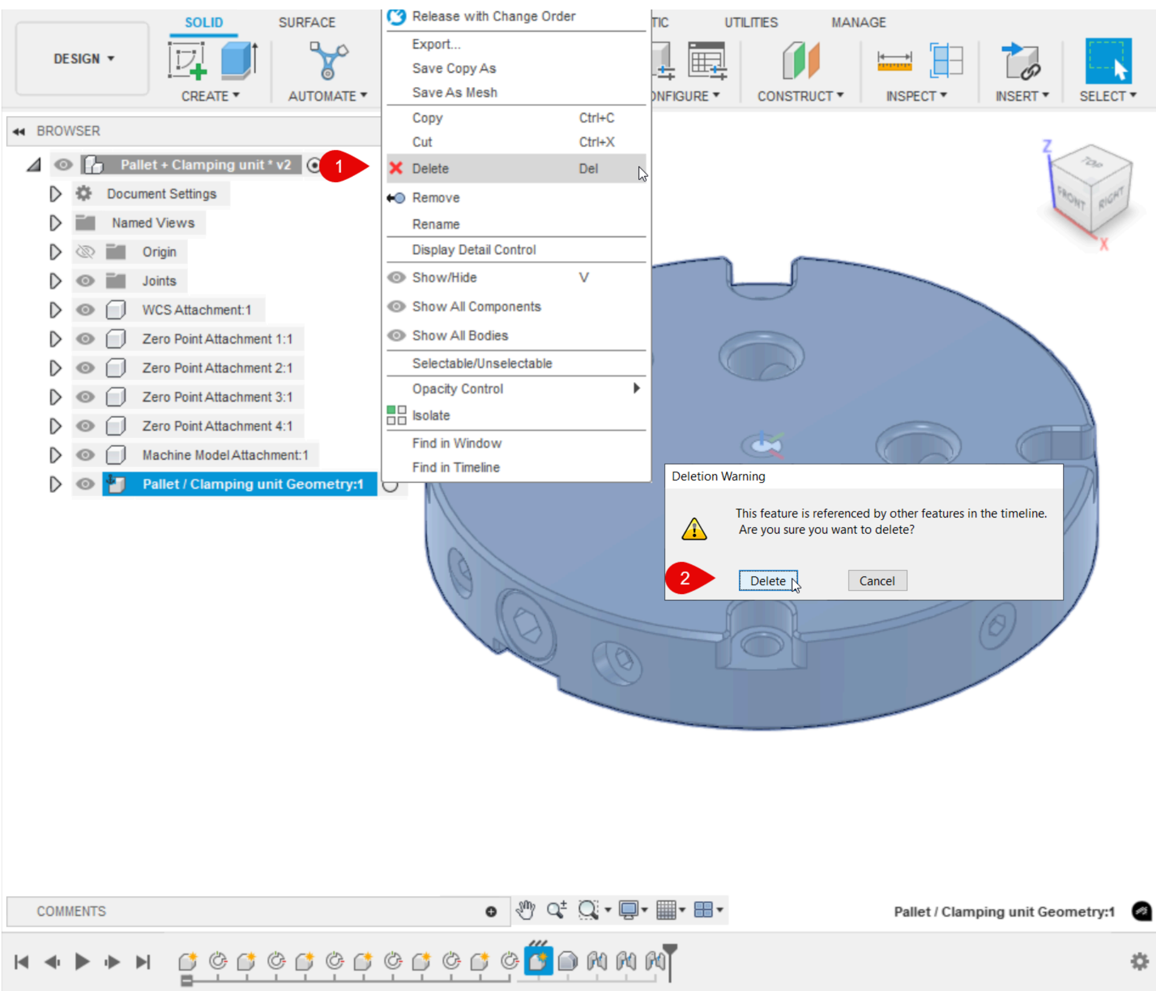
Importing your Clamping unit into the framework:

1. Begin by making a **Copy** of the Vise design.

2. Delete the placeholder geometry, you will be warned that downstream features will be affected.

   So long as the JOC Attachment components and Joint Origins remain in the design you can remove everything else.



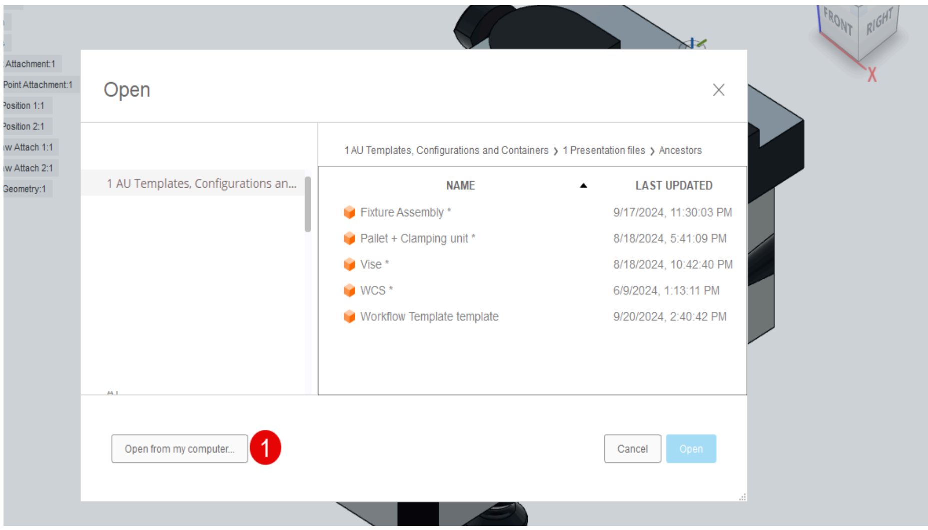


Open, simplify, and prepare data to import into the framework

3. Insert your existing Clamping unit geometry into the Clamping unit design.

You can insert your existing design as an external reference and **break the link**. Be sure to **delete any existing Joint origins** that aren't part of the JOC framework.

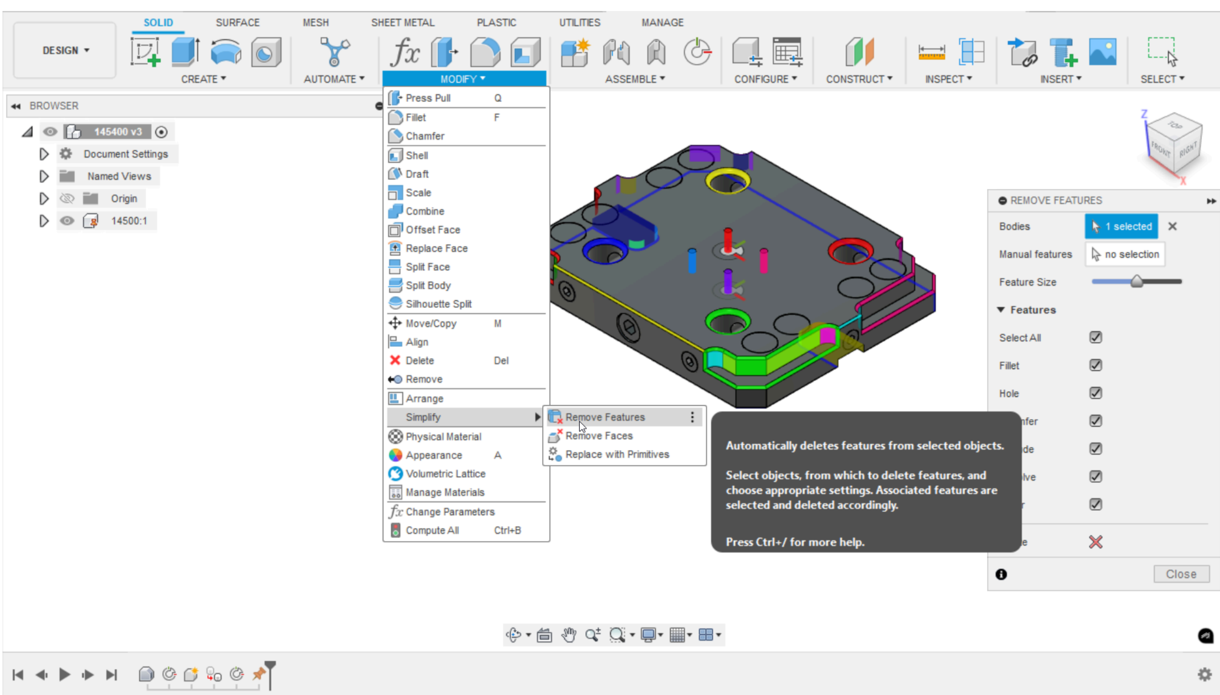
Alternatively, you can *File* > *'Open from my Computer'* to open new Clamping unit geometry from the manufacturer's website.



4. Simplify the design

It is wise to edit the fixture CAD down to simple primitive shapes to avoid excessive toolpath computation and simulation time.

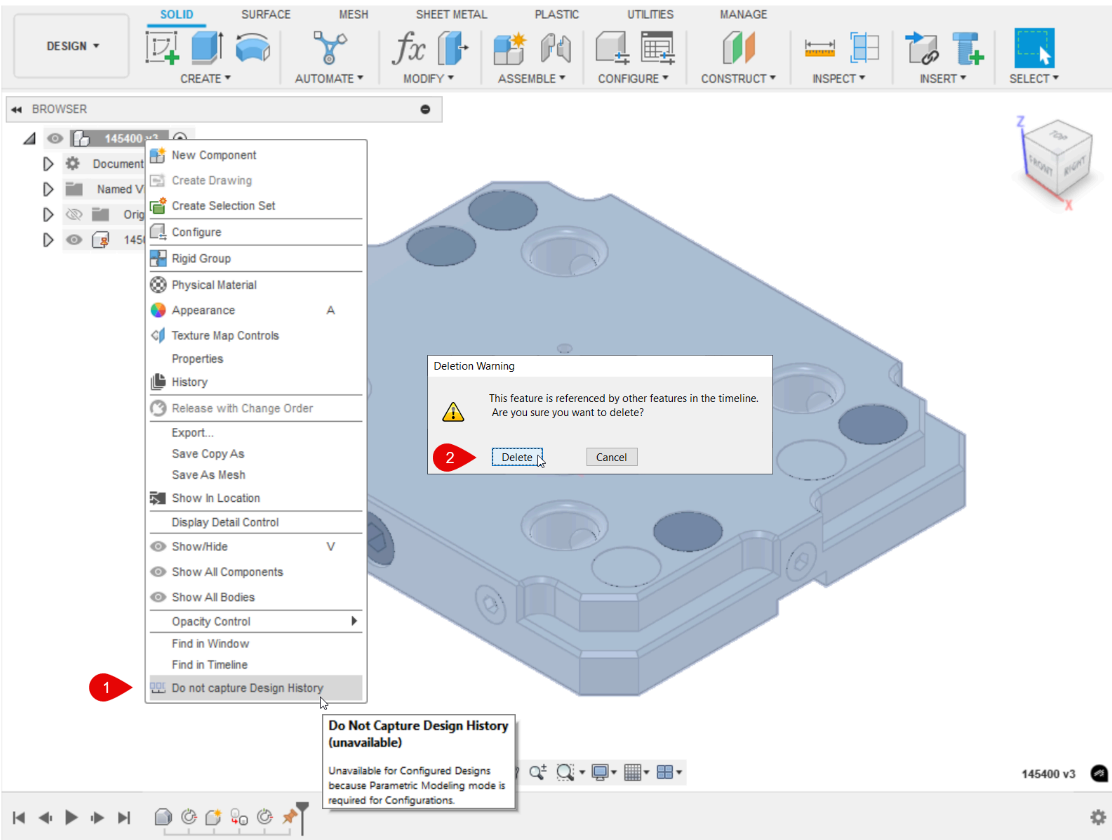
Using the **Simplify**, **Remove Features** command to reduce the number of faces is very efficient and effective. Note that this command is only available when the design history is active.



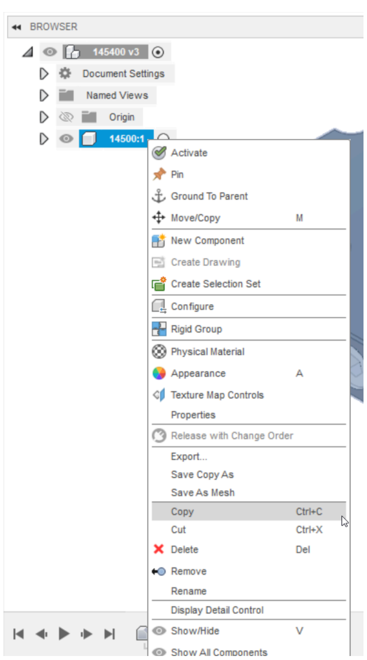
Declutter the timeline

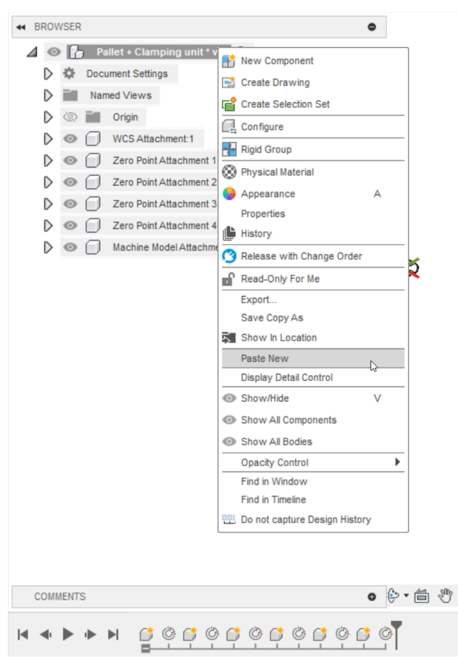
5. Once you are done simplifying your model, **turn off design history** to clean up the timeline, remove any Joints, or Joint Origins, and Pin/Unpin timeline features.

   Be sure to do this in a separate design from the Vise with the JOC attachments.



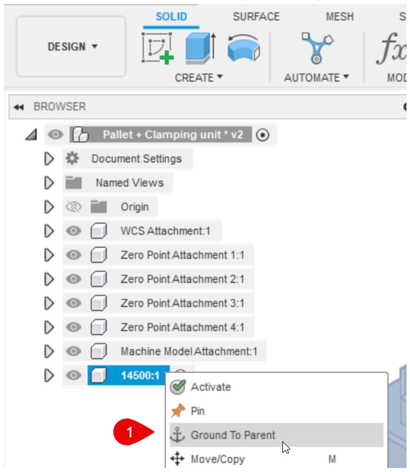
6. Organize the relevant geometry into a Component container, copy it, and **Paste New** into the Clamping Unit design with the JOCs





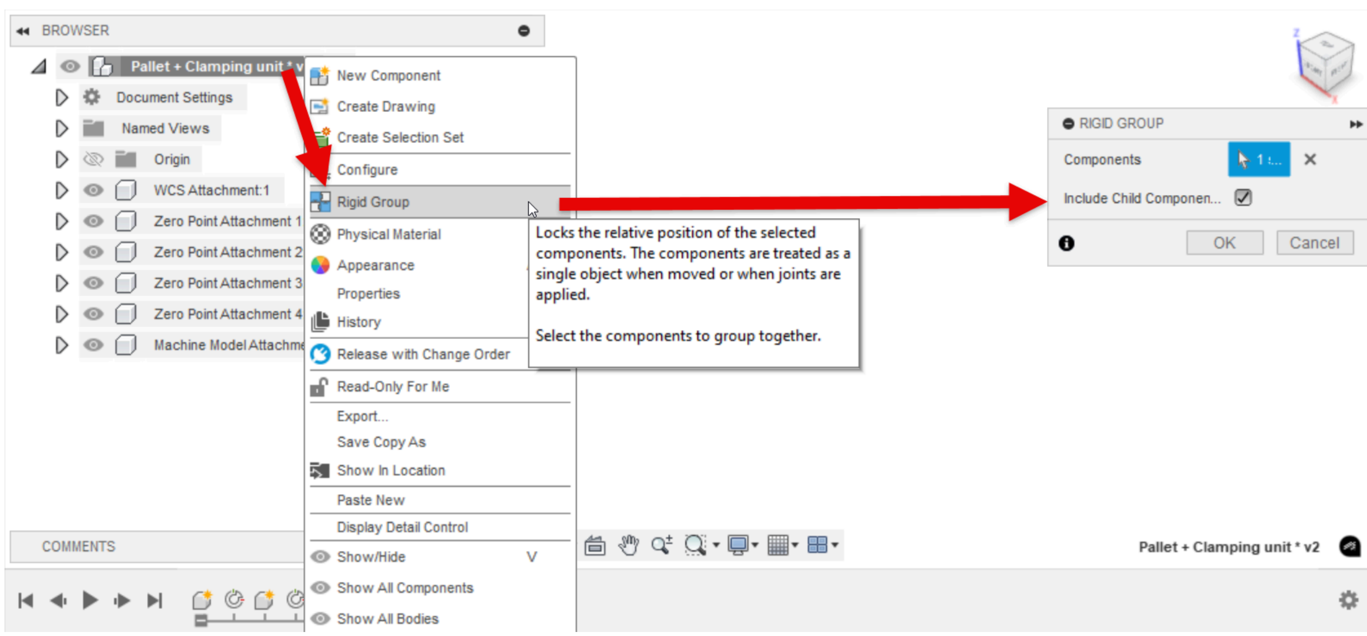
Rigging the Attachments onto the Clamping unit.

7. Ground the Clamping unit geometry to parent.



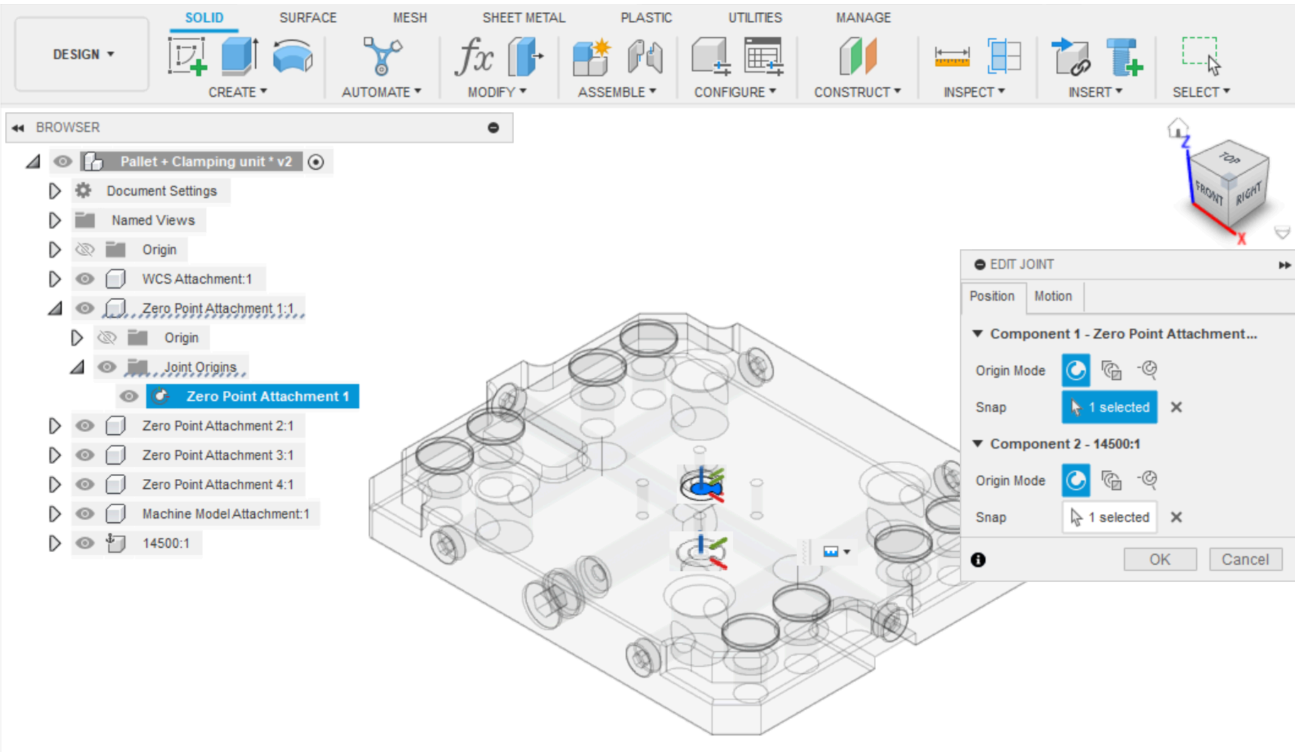
You can insert multiple Components into this Clamping unit design. It should be an assembly of any non-vise geometry you would like to include in your cutting simulations for collision checking.

If you simply use a Vise on a machine without any pallet or clamping unit, leave the JOC attachments and remove the geometry, then **Rigid group** the root component, including all of the JOC components to fully constrain the design.

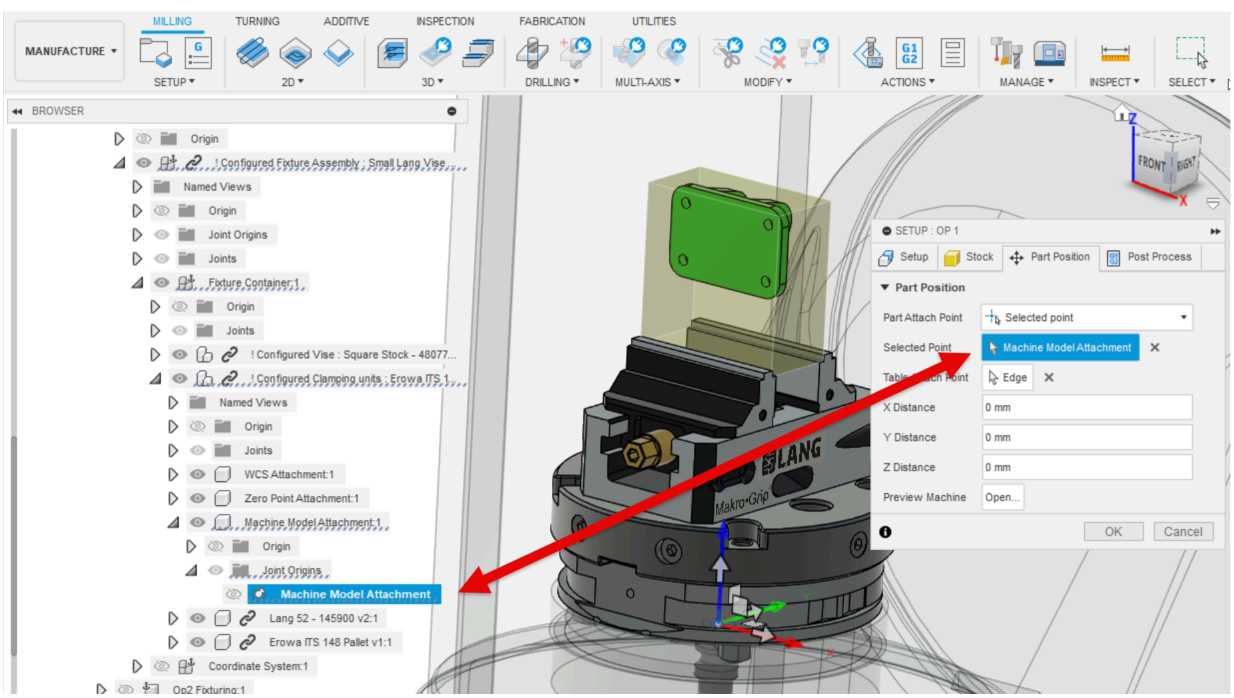


Now that the geometry is assembled and constrained in the **Clamping Unit** design with the JOC attachments, we need to join the attachments onto the functional locations of the geometry.

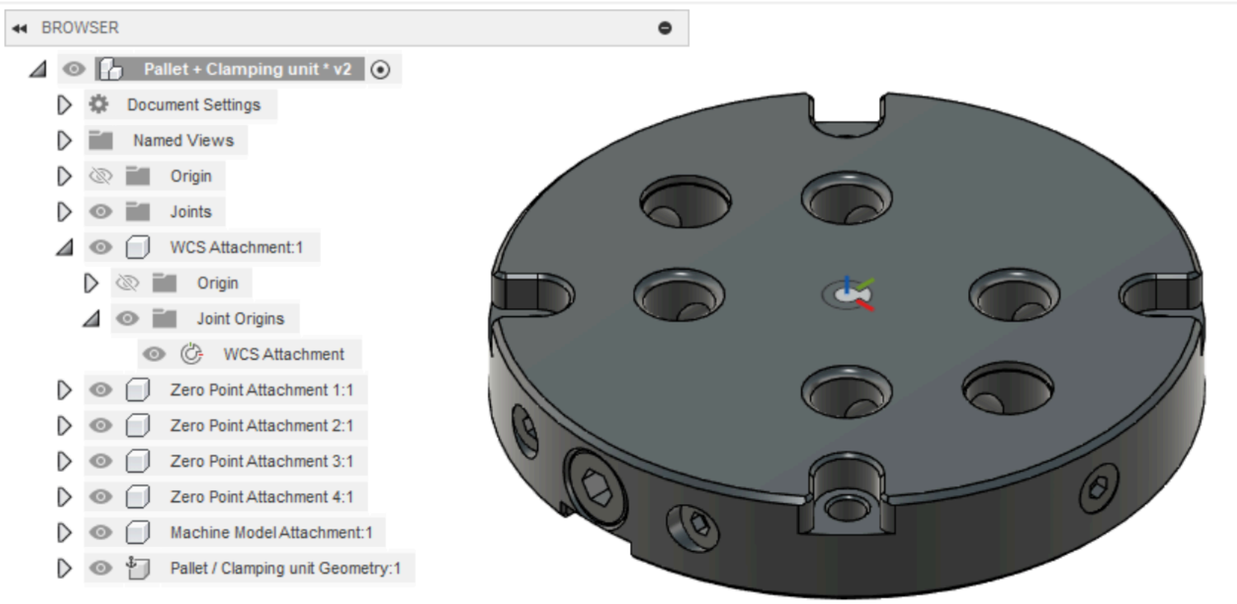
8. Join the *Zero Point Attachment* to the Zero point on the clamping unit geometry.



9. Join the *Machine Model Attachment* to the location your Machine Simulation model will connect to the 'Part Position' in the Setup.



10. Join the *WCS Attachment* to the location where you calibrate your Work-offset, or mount your WCS artifact used to calibrate the machine. We will Join the **WCS** type design onto this location in the **Fixturing Assembly** design.



Once the Zero Point attachment, and Stock Attachment are constrained to geometry you can save the design.

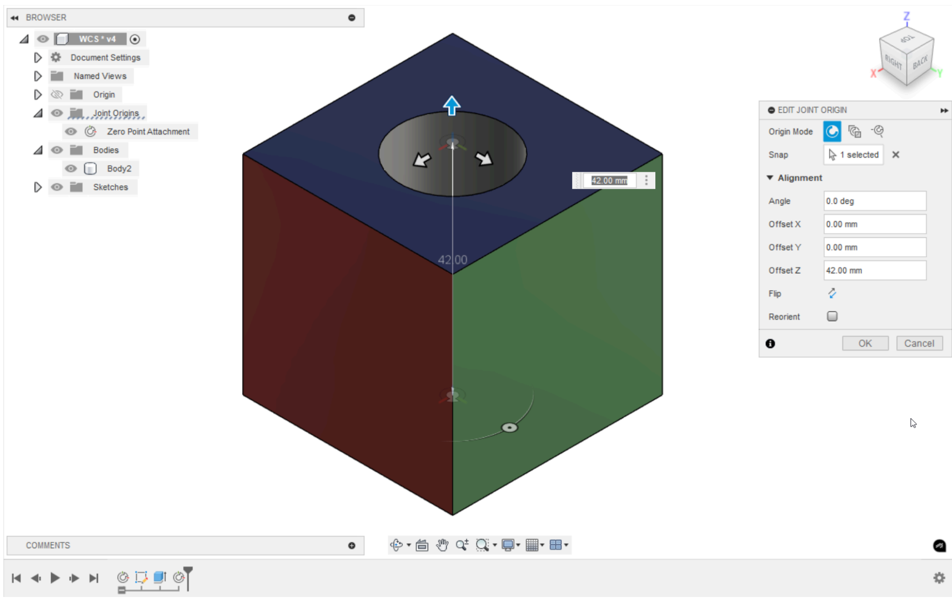
If a Joint Origin/ JOC is not relevant, it may be simply left unconstrained. *do not delete it.*

The initial example designs provided include 4 Zero Point Attachment JOC’s to support multiple vises, but you are free to make as many as you like for your application.

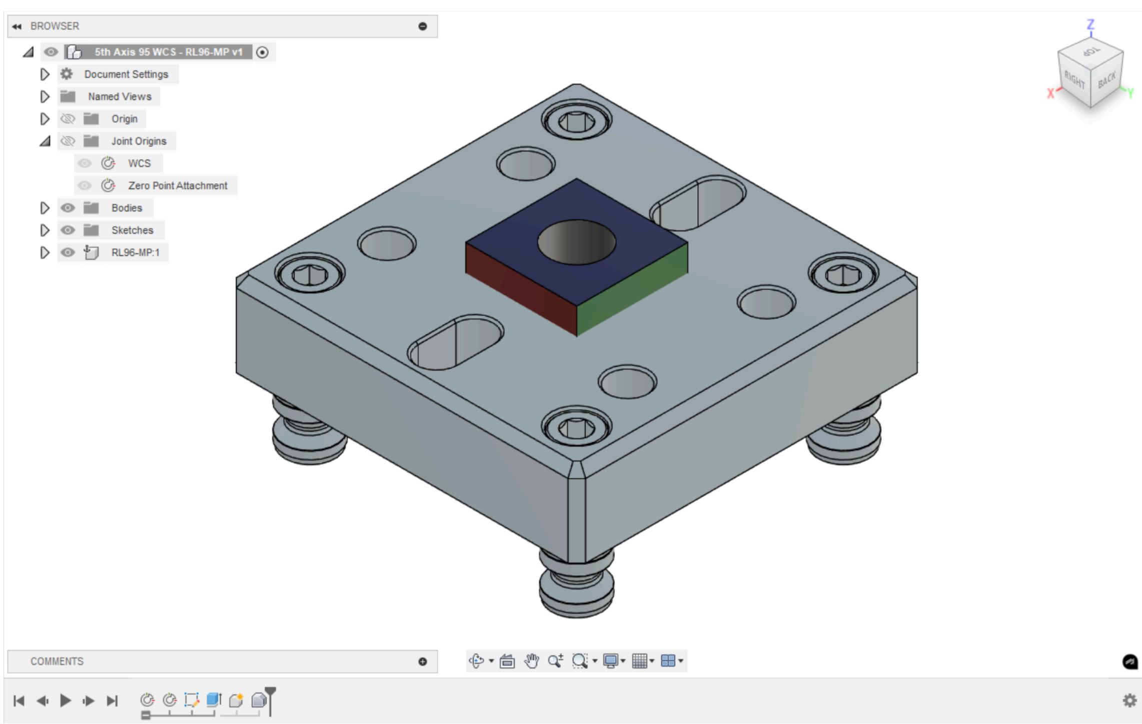
The WCS

The WCS notably does not use Joint Origin Container components for the Joint origins, as there is no need for motion, or realignment between the WCS point and the Zero point where this artifact is mounted.

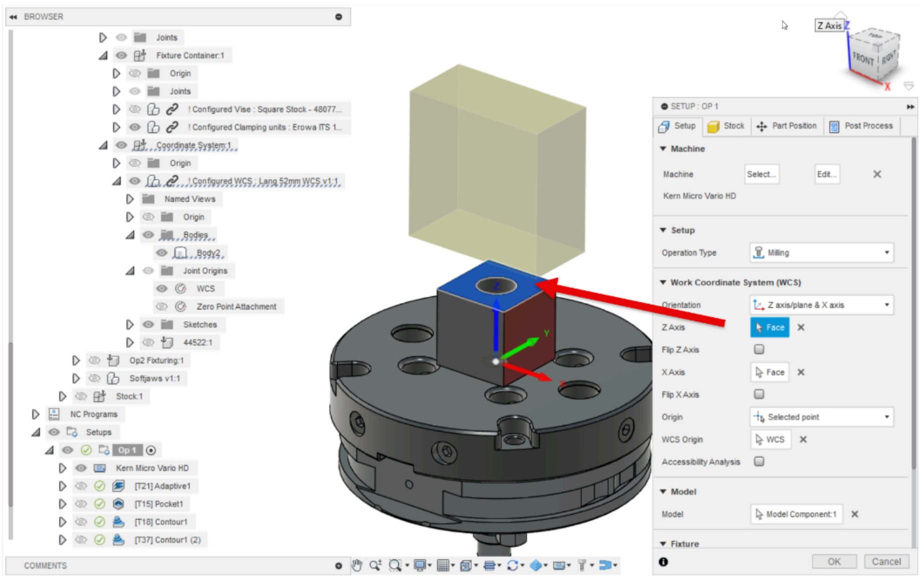
If these two intended coordinates are not coincident, a simple offset may be made to the Joint origin feature to position it as needed.

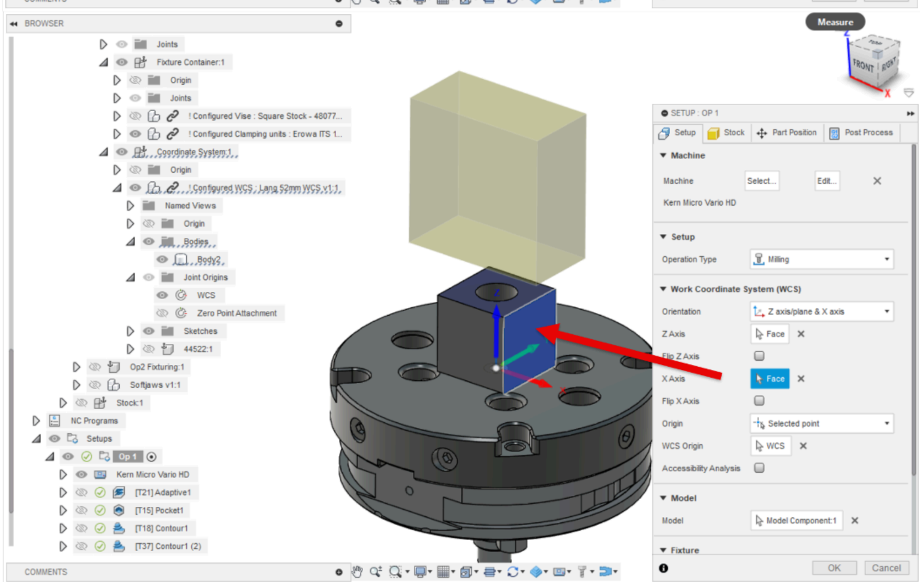


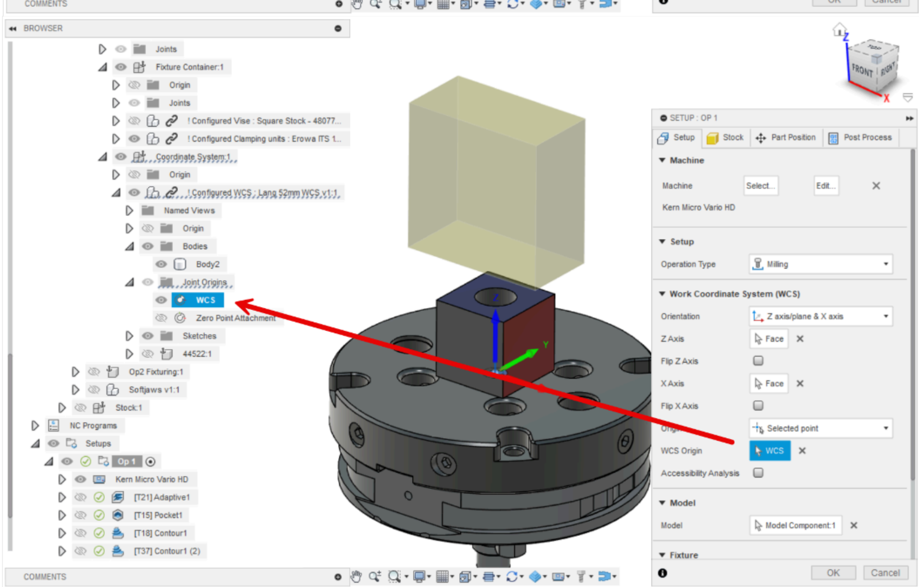
You are free to include other geometry to depict the orientation and assembly context of your WCS artifact, this geometry will not be considered for collision avoidance in the setup.



The Blue, and Red surfaces of the cube are used to define the Z and X directions in the CAM Setup, to explicitly define the WCS precisely as you intend on the WCS Joint Origin.







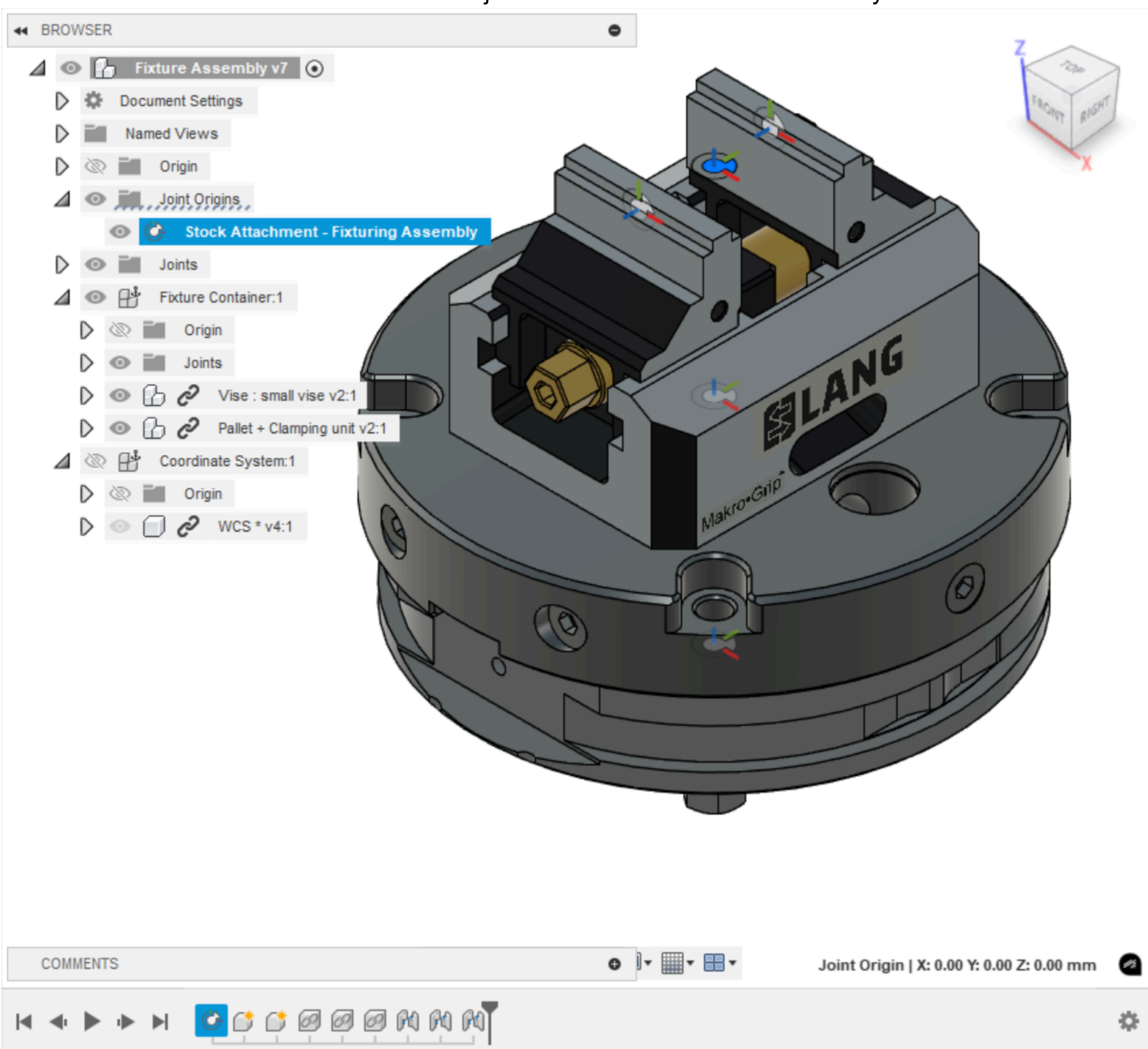
The Fixturing Assembly

A **Vise**, **WCS**, and **Clamping unit** together make a Replaceable **Fixturing Assembly** design type, that we will insert into Templates for CAM programming.

There's a lot of components in the browser, with every attachment being in a JOC component, the user experience clicking through the browser in the CAM environment is getting a bit rough.

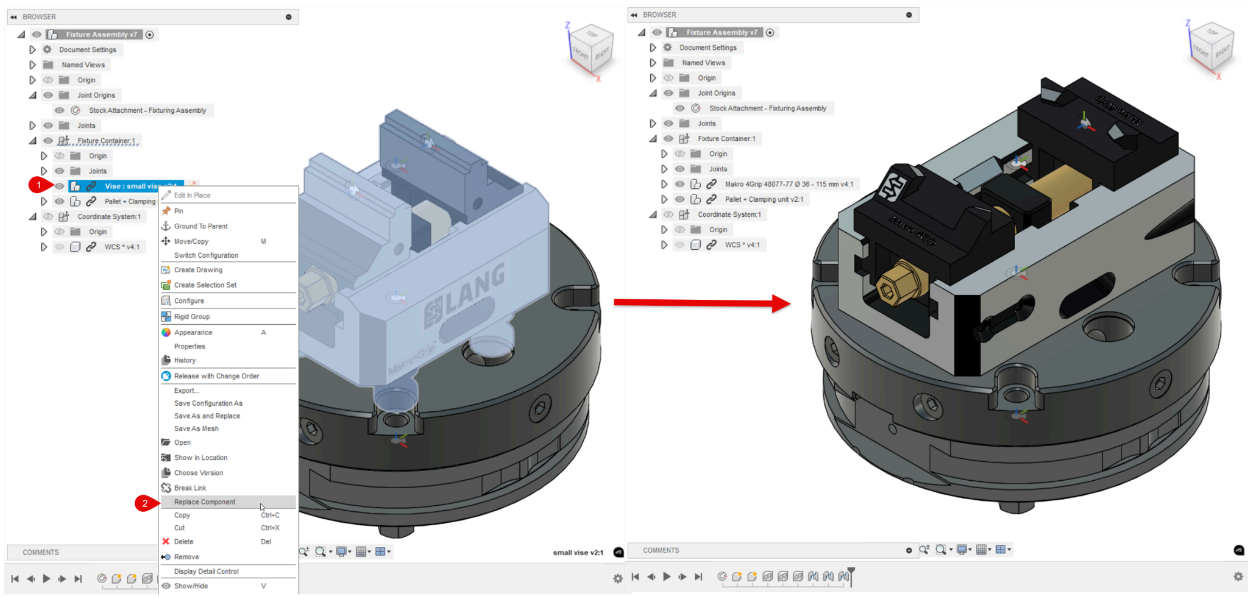
To cut down on this, the RFA has a **redundant** *Stock Attachment - Fixturing Assembly* Joint origin placed at the Root Origin.

The *Stock Attachment* of the Vise should be joined to it to constrain the assembly.



We will use the Joint Origins to join each design together, so any vise can be simply replaced for another. The 3 joints in this design will resolve without error because they are pointing at Joint Origins that have a lineage back to a common ancestor found in the incoming vises.

The Vise is joined from the *Stock Attachment* to the *Stock Attachment - Fixturing Assembly*
The Vise *Zero Point Attachment* is joined to the Clamping unit *Zero Point Attachment*
The WCS *Zero Point Attachment* is Joined to the Clamping unit *Zero Point Attachment*



We can Keep Saving new versions of **Vises**, **Clamping units**, and **WCS** designs, which can all seamlessly be replaced with one-another in our **Fixture Assemblies**, which can be inserted into our CAM programs, allowing us to move CAM operations “from one machine, to another” by replacing one **Fixture Assembly** another.

Or..

We could use *Configurations..*

Section 2: Configurations

This framework of Lineage, and JOC containers allows us to build very sophisticated Replaceable Fixture Assemblies, but it can quickly become a lot of work to manage all of the permutations of designs for a small team.

Since we are including the intent of the machine pallet system, multi-location zero point systems, and modular vises - every time we add a new Machine, or Vise to the mix, we have that many more permutations of designs that need to be copied, replaced, and saved-as before they can be used.

Configurations can leverage the same JOC's to compress all of our fixturing CAD into 4 configured designs that can be simply re-configured while maintaining CAM selections, and all the joints between each fixture.

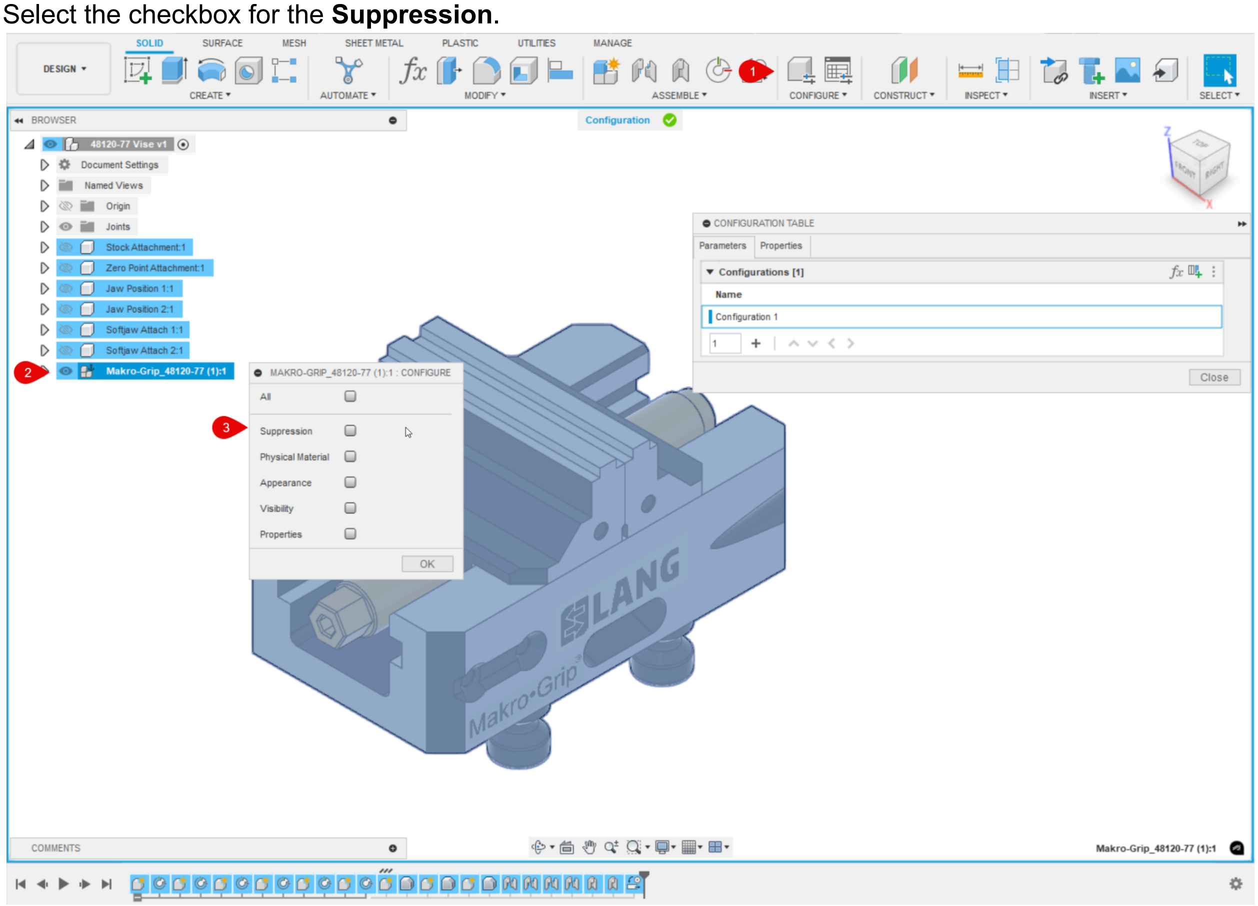
Configured Vise

Begin with one of your Vise designs, and select the Configure tool in the Ribbon.

This will open the Configuration Table, and a new selection mode within Fusion. Anything in the Browser, or the Timeline that is highlighted blue, is configurable.

Select the geometry component container and a dialog will open with some checkboxes for the configurable aspects of each component.

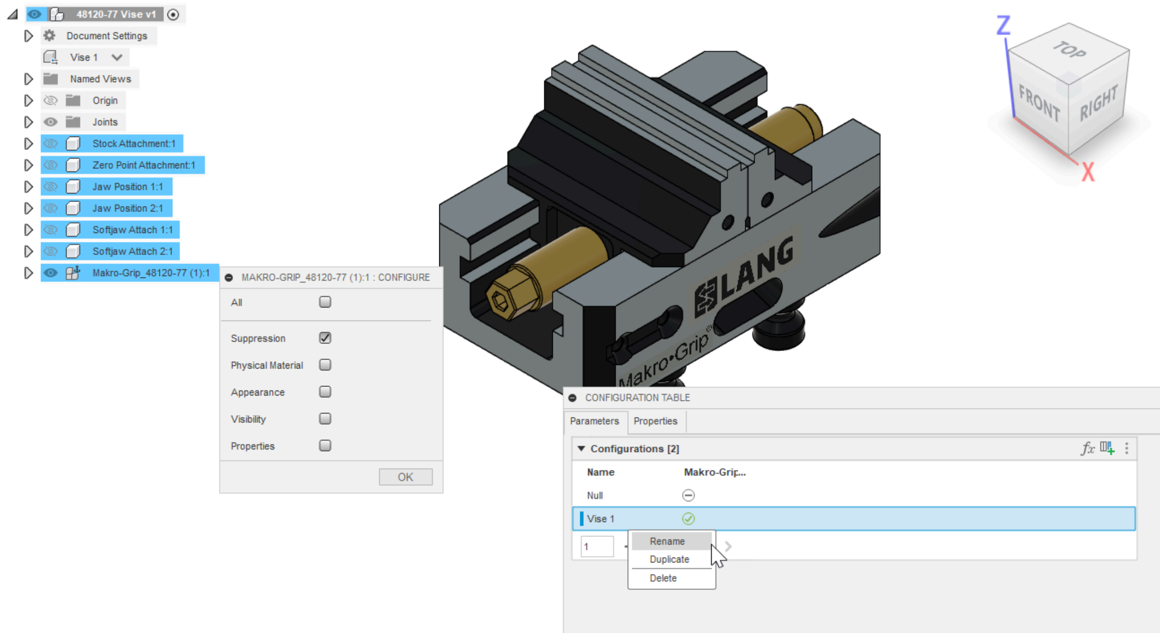
Select the checkbox for the **Suppression**.



Now we will see a new column in our Configuration Table with a checkbox we can toggle, to **suppress**, or unsuppress the Vise.

Right Click on the first Configuration Name, and Rename it to “**Null**” with the Geometry suppressed

Press the + Icon to add another Configuration for the Vise and have it **unsuppressed**

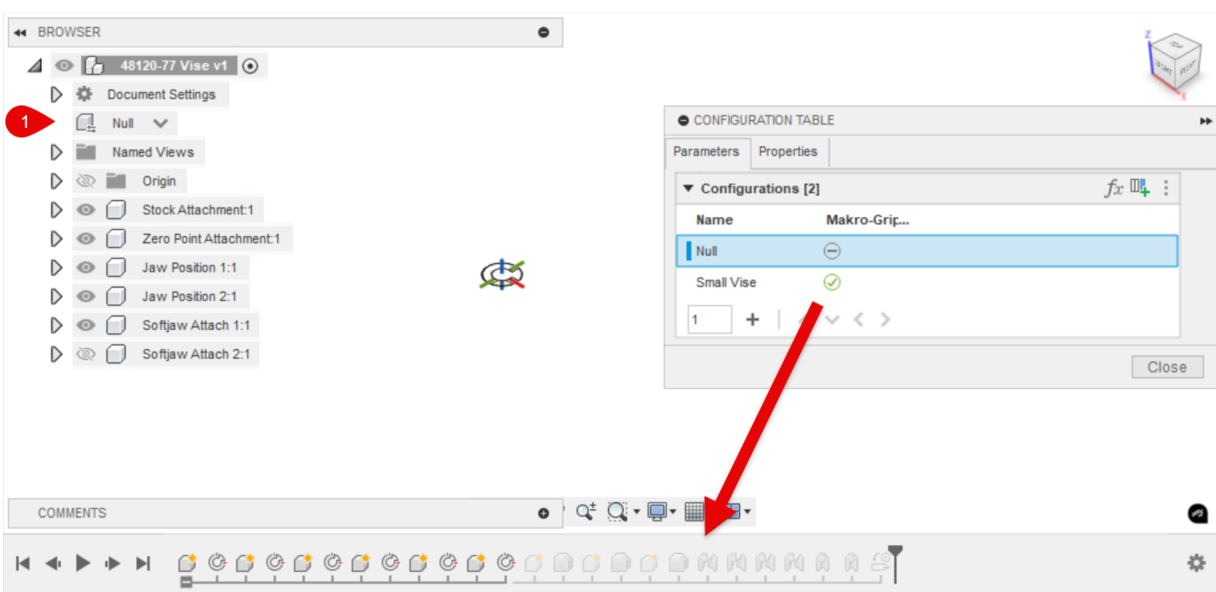


You can switch configurations from the new dropdown in the browser, or by double-clicking on a row in the configuration table.

Observe the timeline to see how the geometry suppression will also suppress the joints that rig the JOC onto the geometry, letting them return to the Origin of the document.

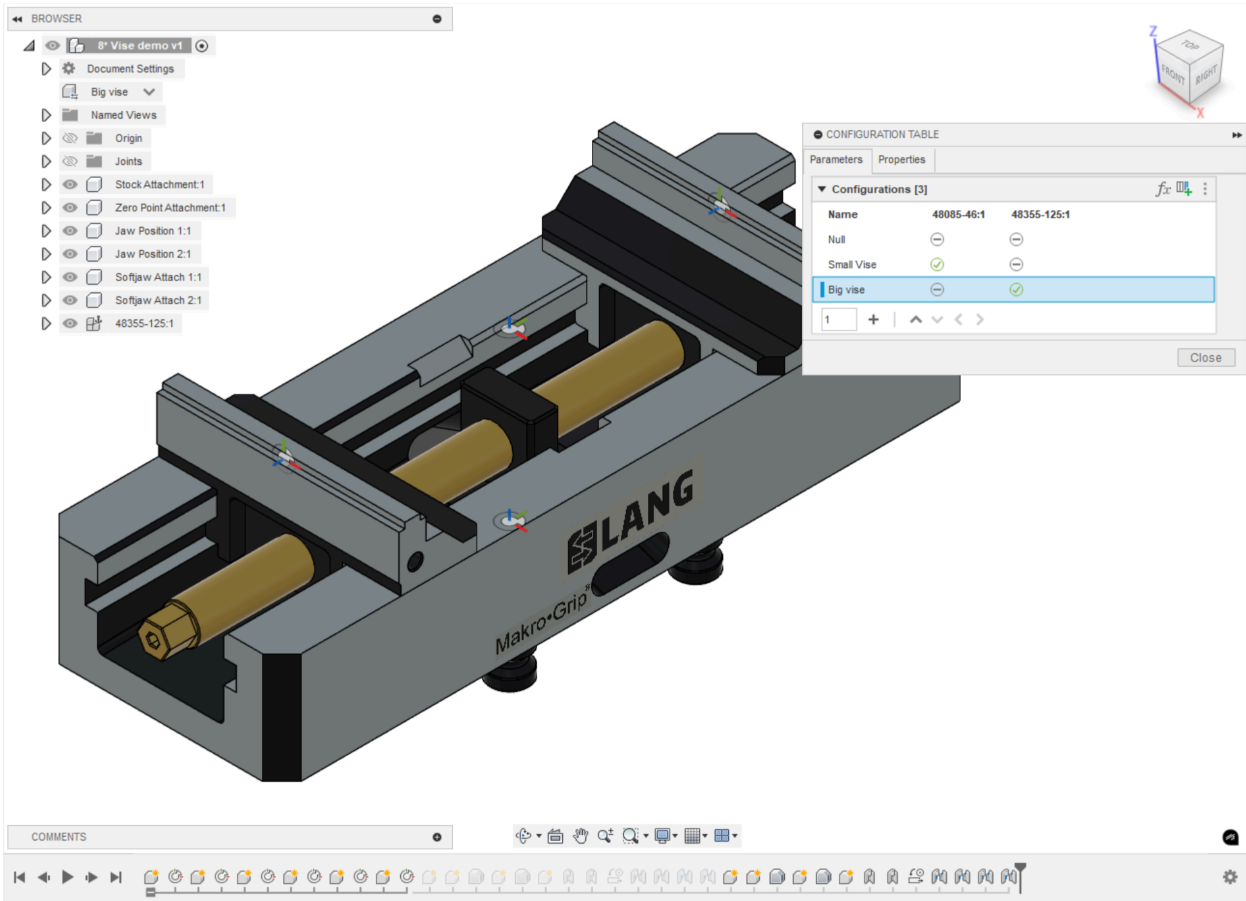
This is because we selected the suppressed geometry for one of the joint input selections, and configurations will suppress downstream dependencies.

We can leverage this behavior by inserting another Vise, and rigging the same JOC attachments onto it while the first vise is suppressed.

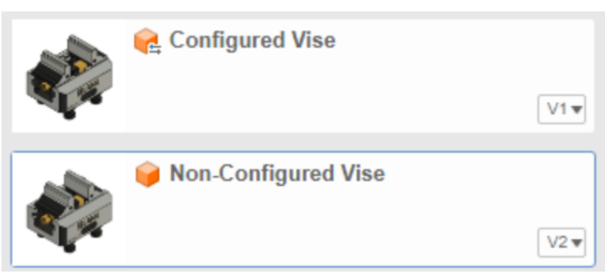


Follow the steps from [Importing your Vise into the framework]() to insert another Container of Vise geometry into this Configured Vise design and rig on the attachments

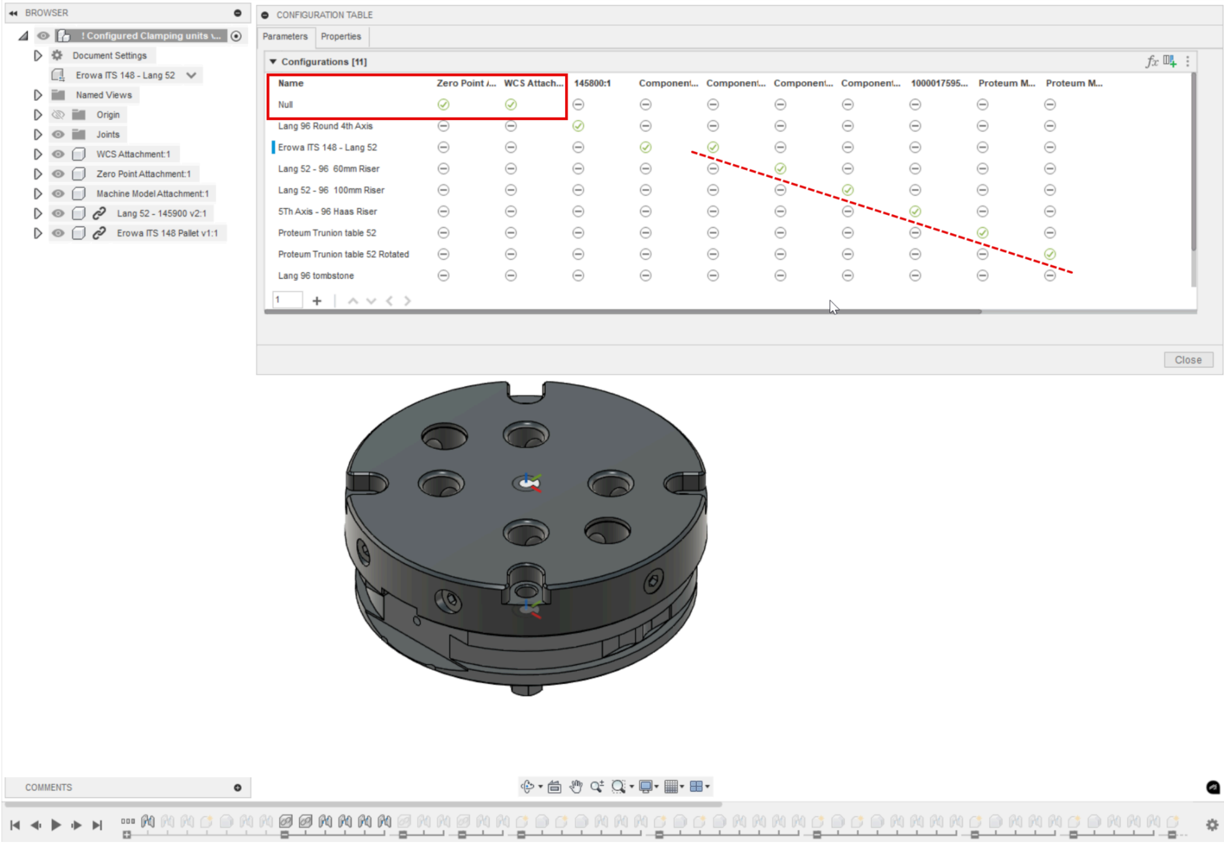
Add a configuration for the second vise, and continue building the **suppression matrix** with only one active vise geometry, and set of rigging joints active at a time.



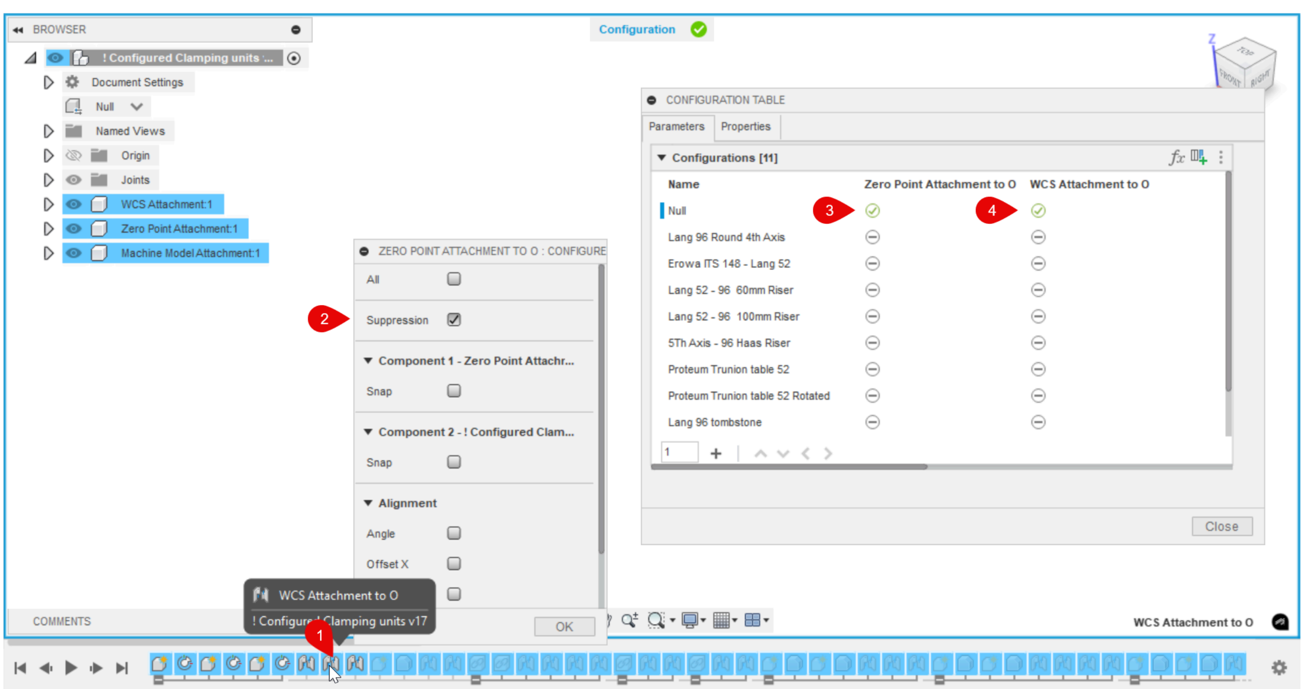
Save this design, and it will become a Configured design, Denoted by the subtle change in icon.



Configured Clamping unit
Clamping units are configured in the same way as **Vises**, with a simple suppression matrix.



Begin with a **Null** configuration that has the *Machine Model Attachment, WCS Attachment, and Zero Point Attachment*, Joined to the **Root Origin** of the document.
These joints will fully constrain the null configuration for the case where you want no Clamping unit geometry present.
Configure the suppression of these Joints that constrain the *Zero Point Attachment* and *WCS Attachment*



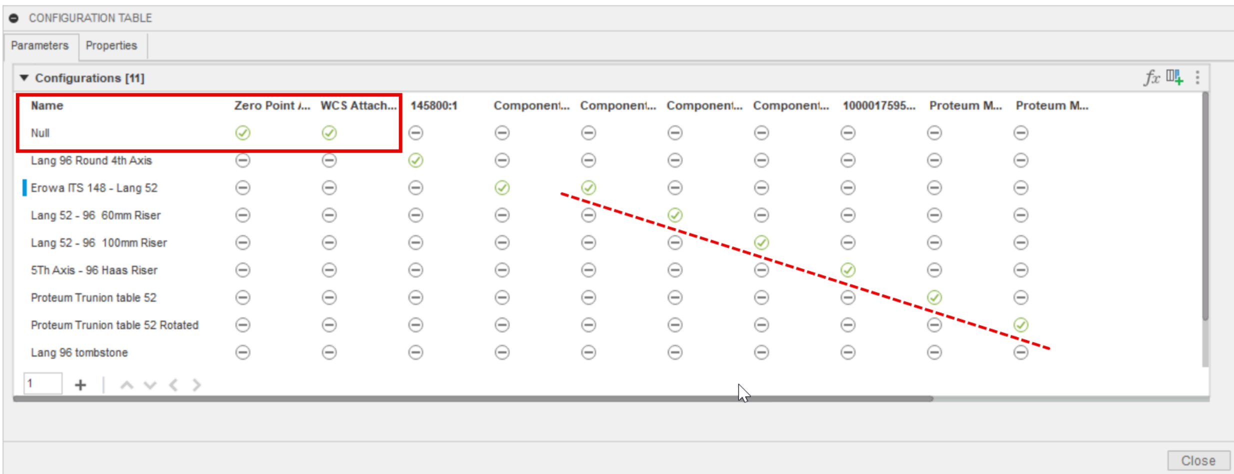
The *Machine Model Attachment* to Root Origin Joint should stay unsuppressed, and unconfigured.

This way, as we join each Joint Origin Attachment onto the geometry, it will become fully constrained.

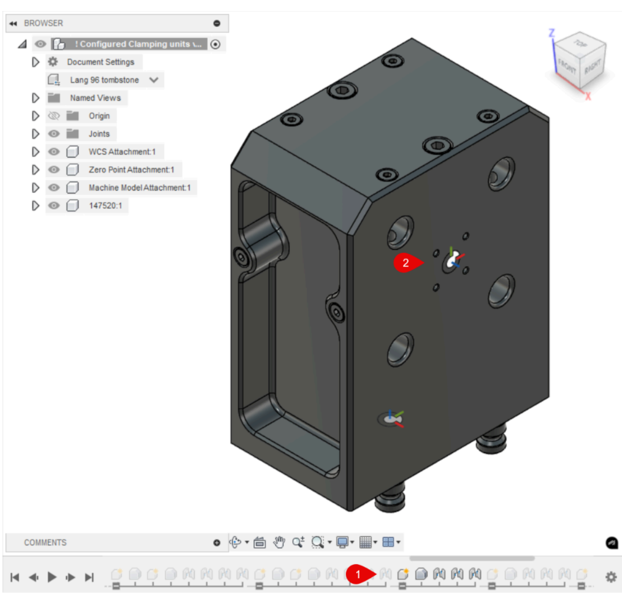
Now we can Paste/Insert new instances of **Clamping unit** geometry into the document, Using the rigging joint to the *Machine Model Attachment* to constrain it.

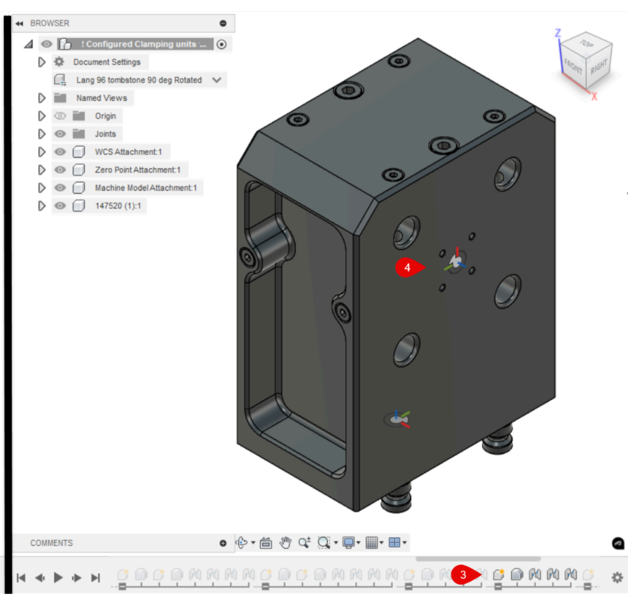
Join the *WCS Attachment* and *Zero Point Attachment* Joint Origins on to the geometry.

Add each piece of Geometry to the **Configuration Table** as a **Suppression,** building out a matrix of suppressions that enable each set of Rigging joints for the active Geometry you need to represent your fixturing.



If you need multiple orientations of a **Vise** given **Clamping unit**, you can simply just insert a second instance of that geometry, and re-orient the *Zero Point Attachment* as necessary.

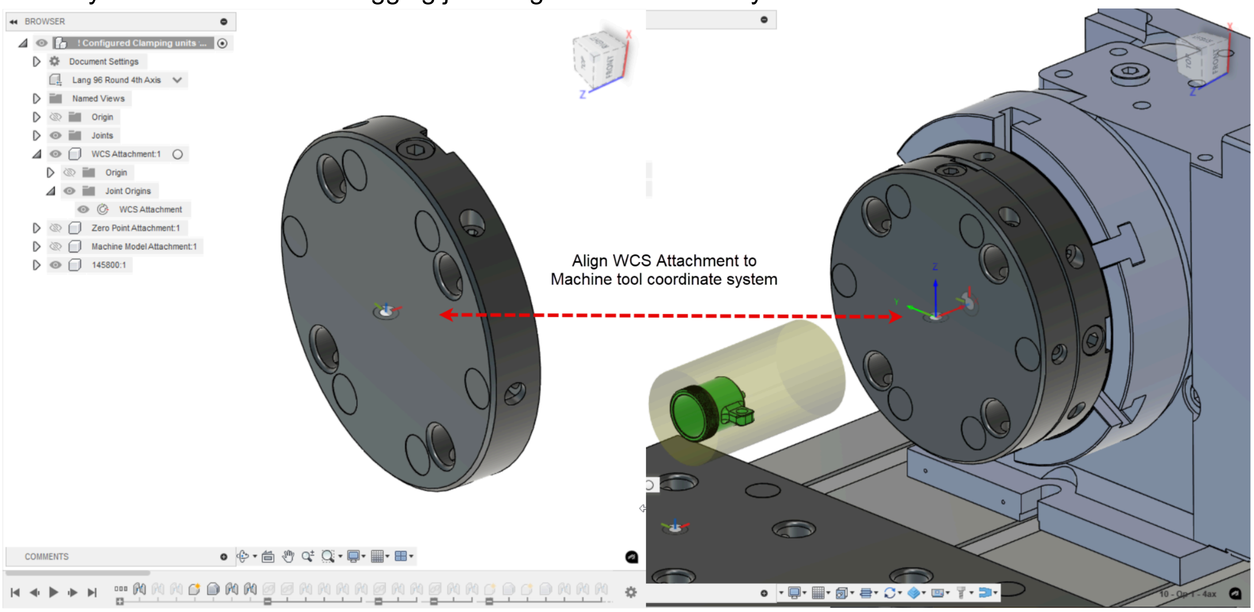




Configured WCS

The WCS Attachment orientation will control the orientation of your fixture group on the Machine Model.

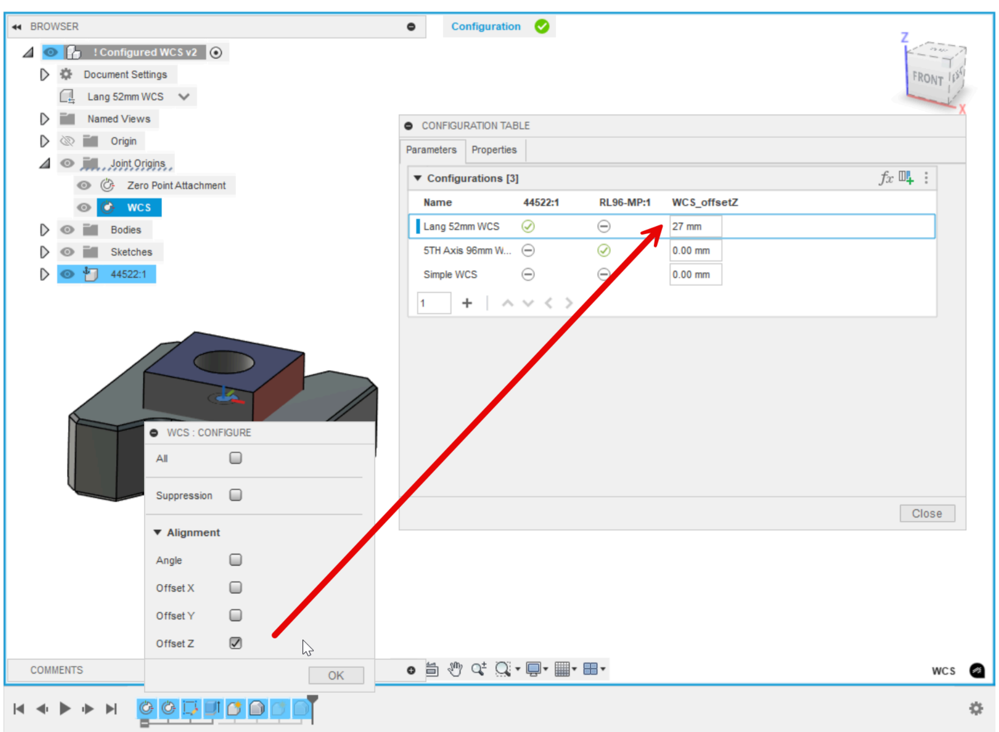
You may need to re-orient the rigging joint to get the orientation you desire.



You may have numerous Zero Point systems in your shop with different distances between the artifact mounting point, and the location the machine calibrates.

You can configure the Offset Angle/X/Y/Z of the Joint origin to position it as needed.

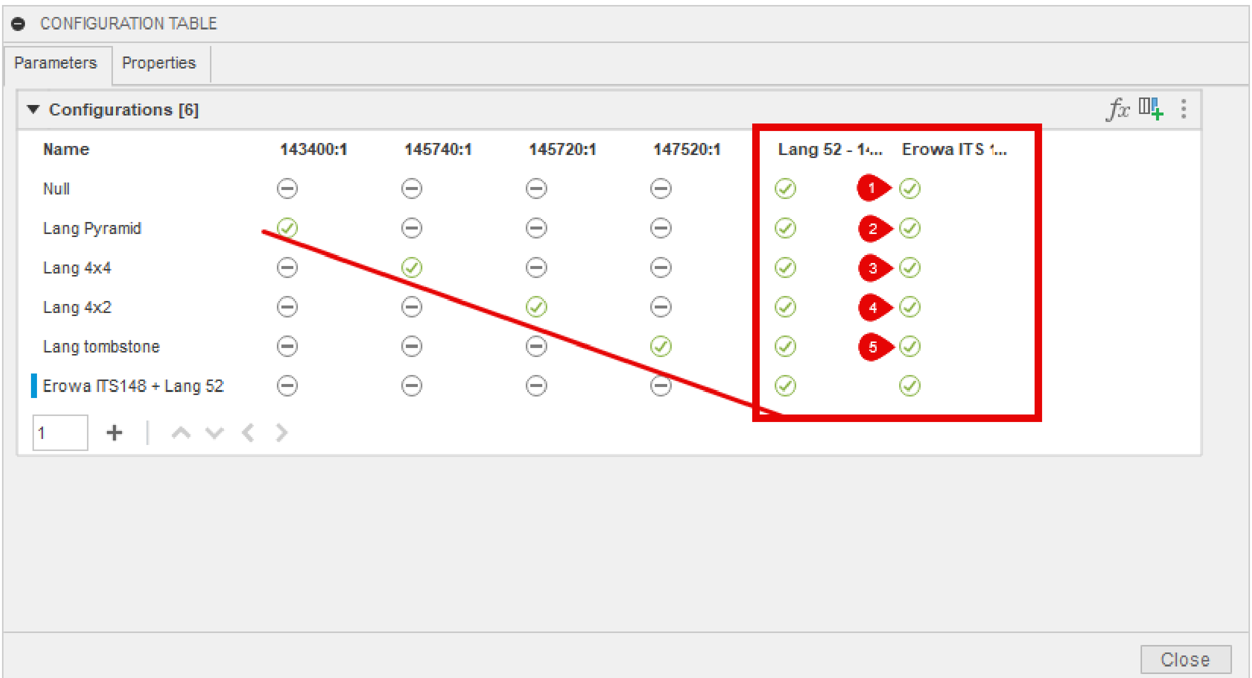
Similarly, you can suppress/unsuppress any relevant geometry you want so long as you **leave the WCS cube.**



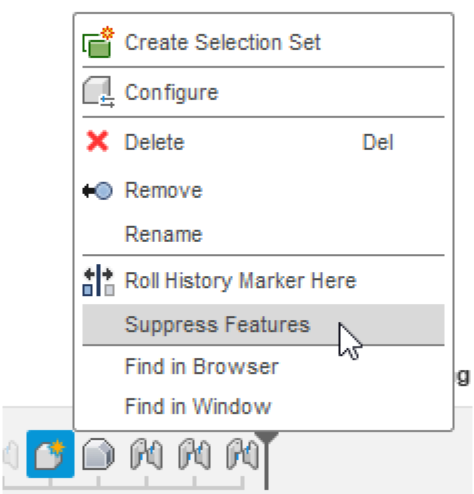
Working with large Configurations.

When you configure a suppression it will be added to the configuration table in it's active state.

It can be tedious to click each suppression off in the matrix of suppressions we are building with most components unsuppressed in any given configuration.



To avoid this, suppress the feature before you configure it and the default state will be suppressed



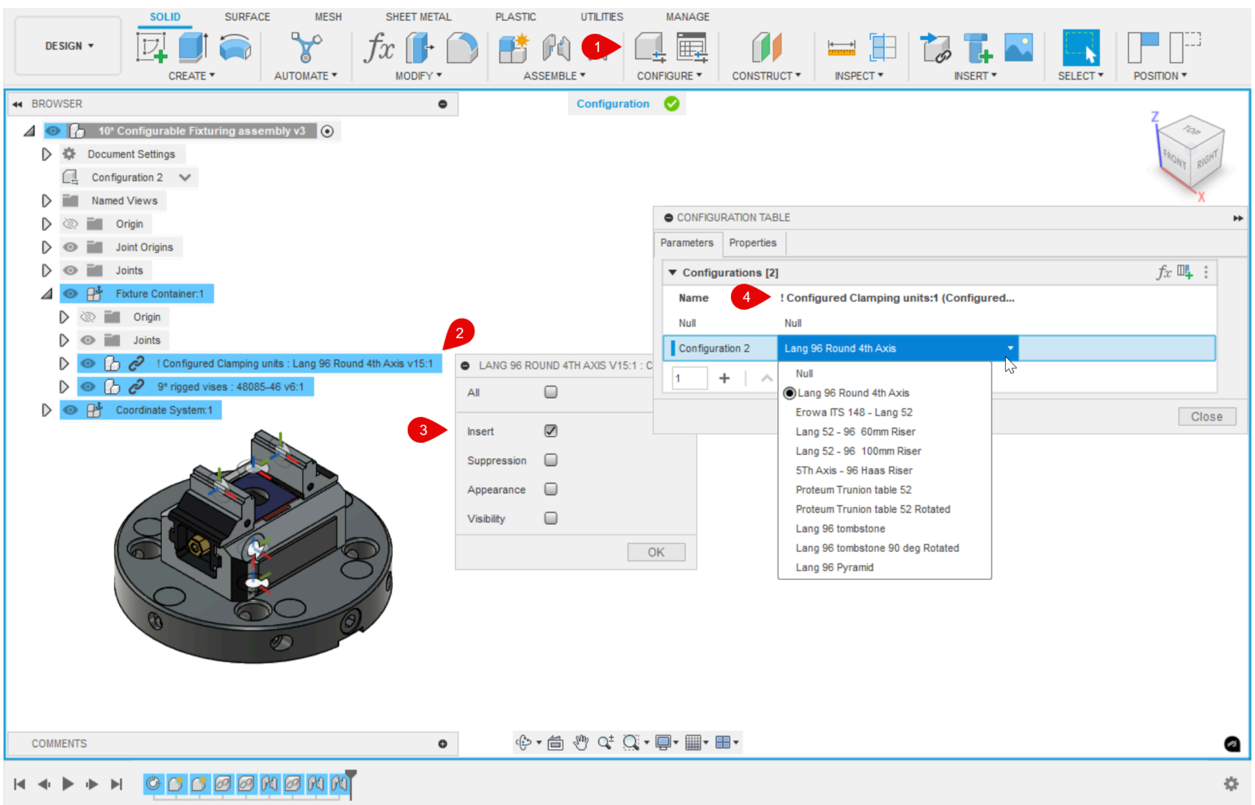
Section 3: Nested Configurations - Configurable Fixture Assembly

Once a **Vise**, **WCS**, or **Clamping unit** design type is configured, you can use **replace** on a similar type design to bring it into a Fixture Assembly.

Use the Configure mode, to select the configurable aspects of the Configured design.

Select the "**Insert**" and it will add a column in the configuration table with a drop-down for all of the configurations found in that inserted design.

This is a **Nested Configuration**



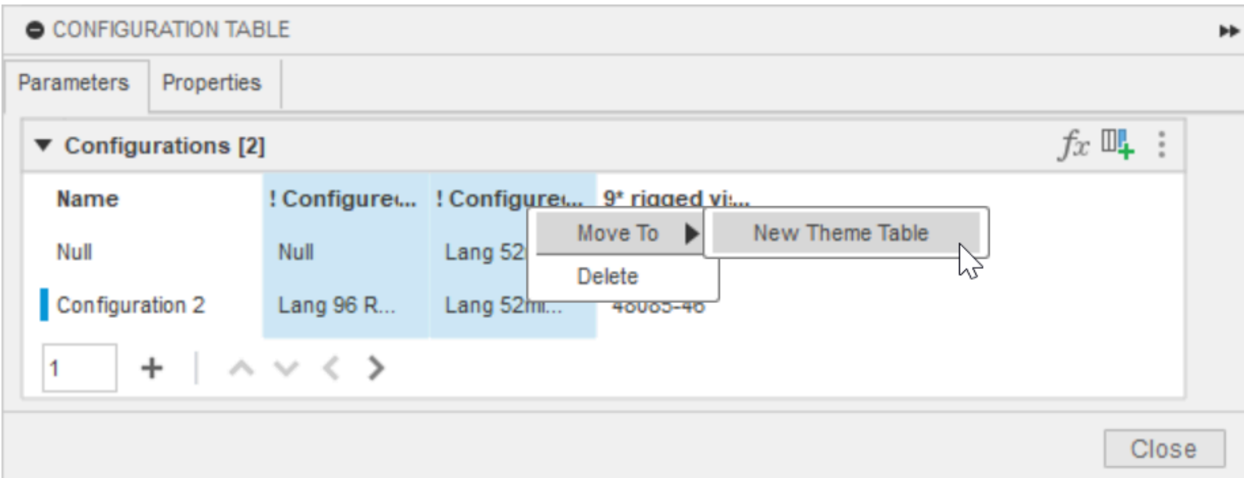
This will create a **Configured Fixturing Assembly.**

Name the first configuration in the table to be **Null** and set that Configured Vise to **Null** for the first configuration.

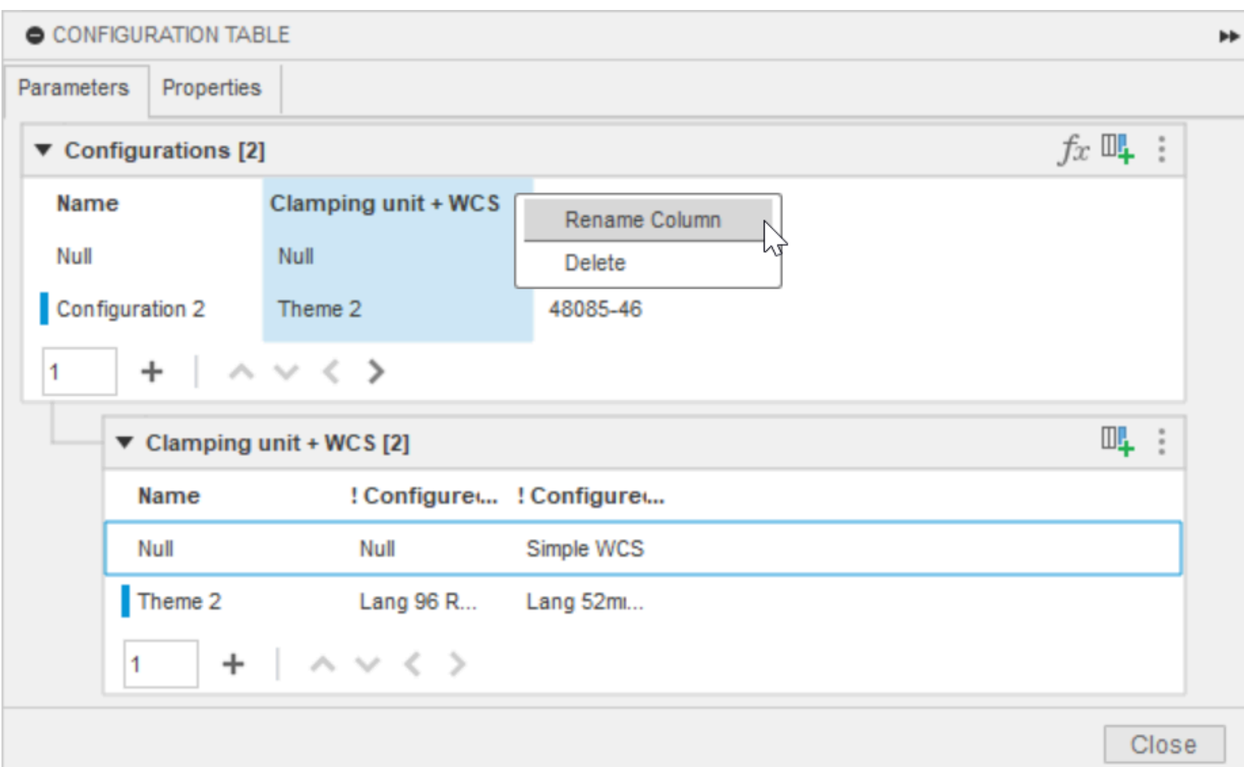
This enables us to insert many configured vises to join onto each *Zero Point Attachment* of the configured **Clamping unit**, and set them to Null when the Zero Point is not being used.

Theme tables create relationships between configured aspects
Objects like the **WCS** and the **Clamping unit** often have physical compatibility limitations, and we can define this relationship within the configuration table by right clicking the column header, and selecting 'Move to new Theme table'

Move the configured Insert of the WCS and the Clamping unit into a new theme table.



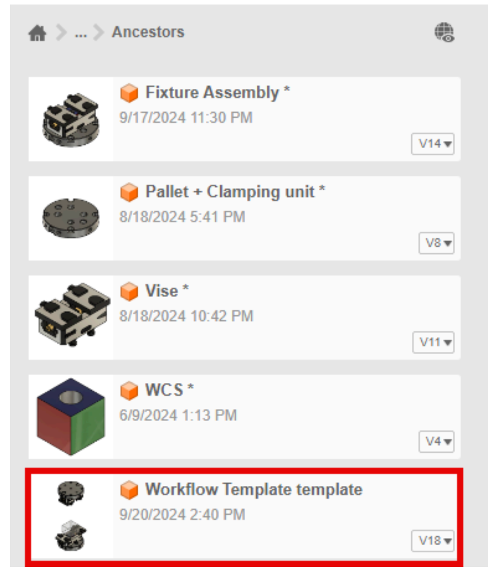
You can rename this Theme Table by right clicking on the New column that is created
We want the first entry in this Theme table to be **NULL** with the Simple WCS for situations where we don't need a clamping unit present.



Section 4: Using Configured Fixture Assemblies in Workflow Templates.

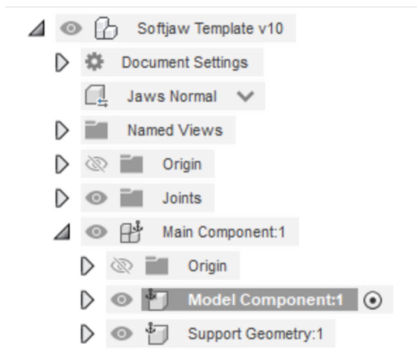
Workflow template template

Return to the Workflow Template Template design that we initially downloaded from the [Github]

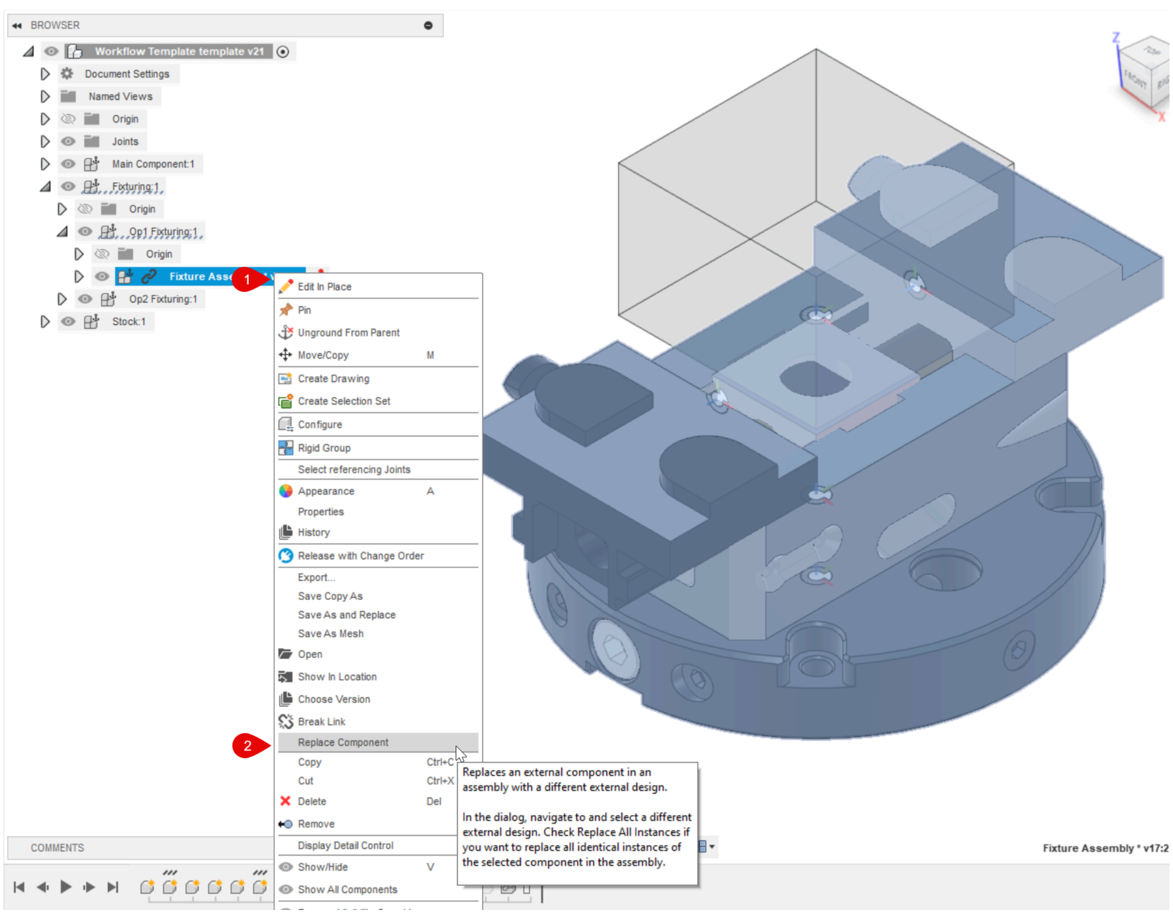


The Workflow template is a great starting point for you to create a 'workflow', or sequence of setups that have the machine, fixturing, stock, and model pre-selected for a given manufacturing strategy.

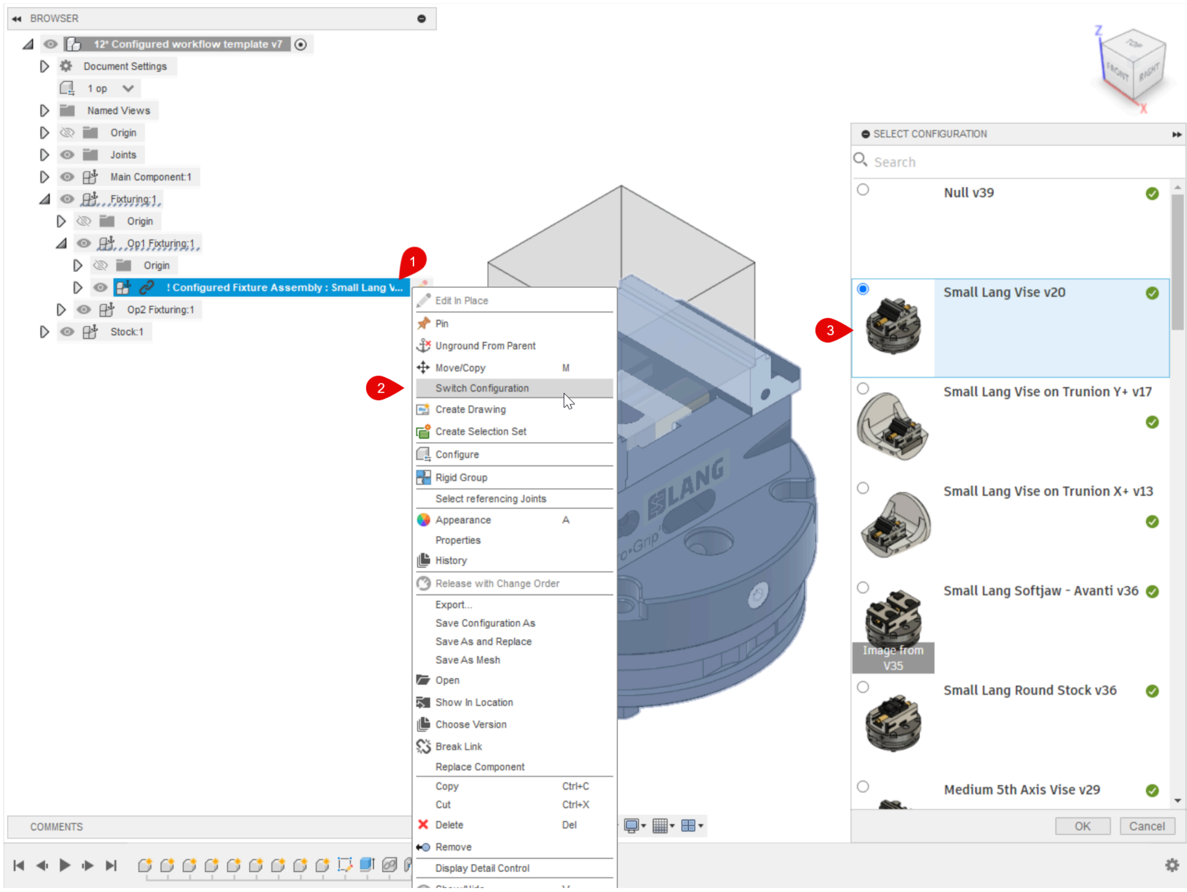
The best way to use a template is to **copy** the last similar program, and replace or insert your model into the Model Component container.



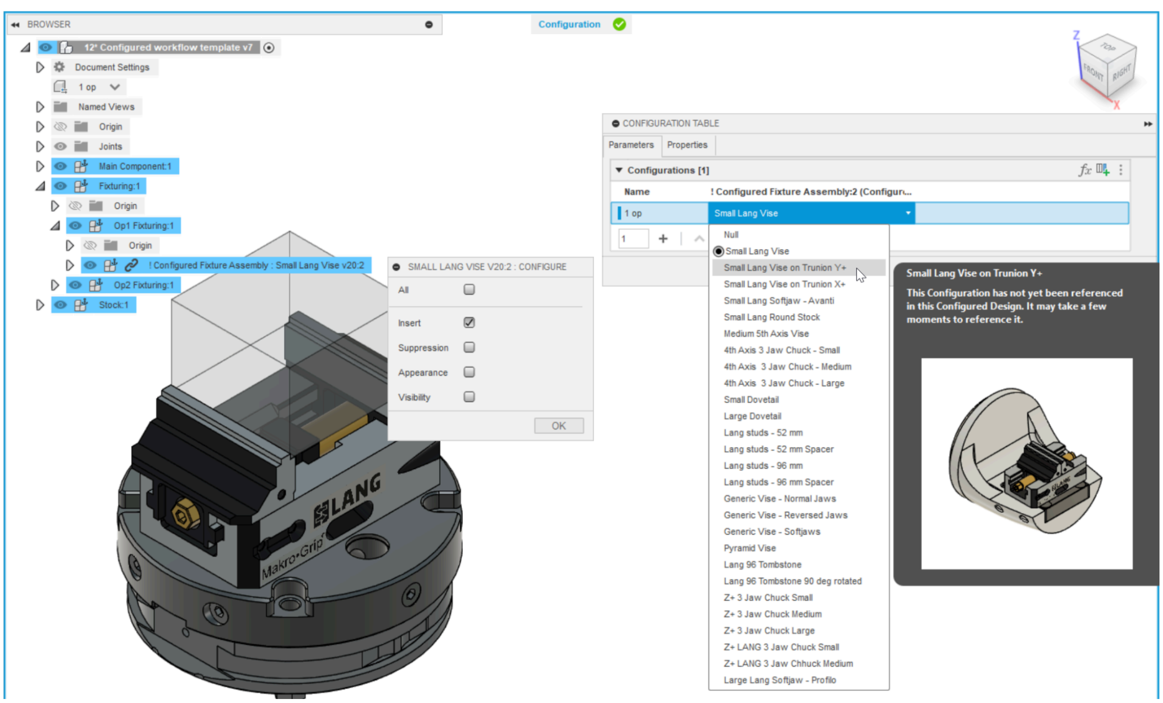
Use **Replace** on the fixturing Assembly, and bring in your configured Fixturing Assembly into the framework and build yourself a CAM programming environment that can automate your programming of your common part families.



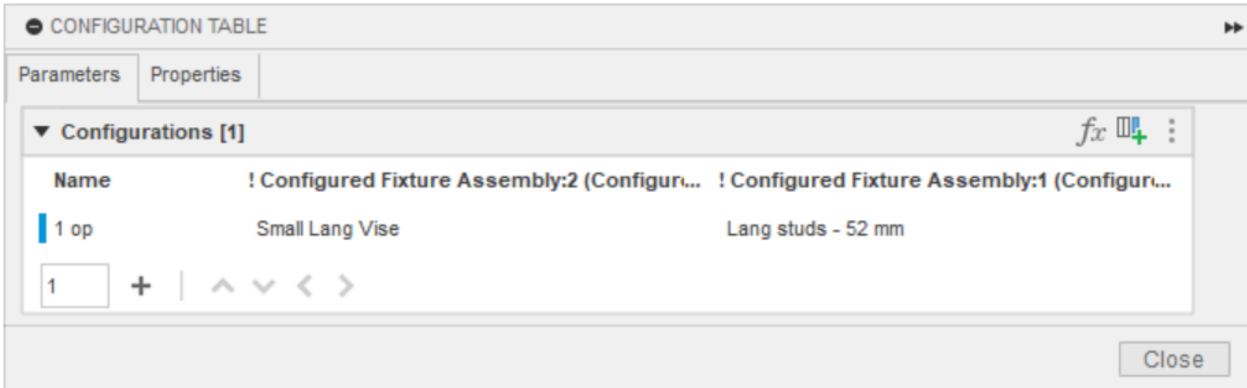
Now we can simply Right click on the Configured Fixture Assembly, and select **Switch Configuration** to select any of the configurations of the Fixturing Assembly.



We can configure the Fixturing assembly with the Configured Insert to select via the configuration table. Simply select which Configured Fixturing Assembly is needed from the Dropdown.



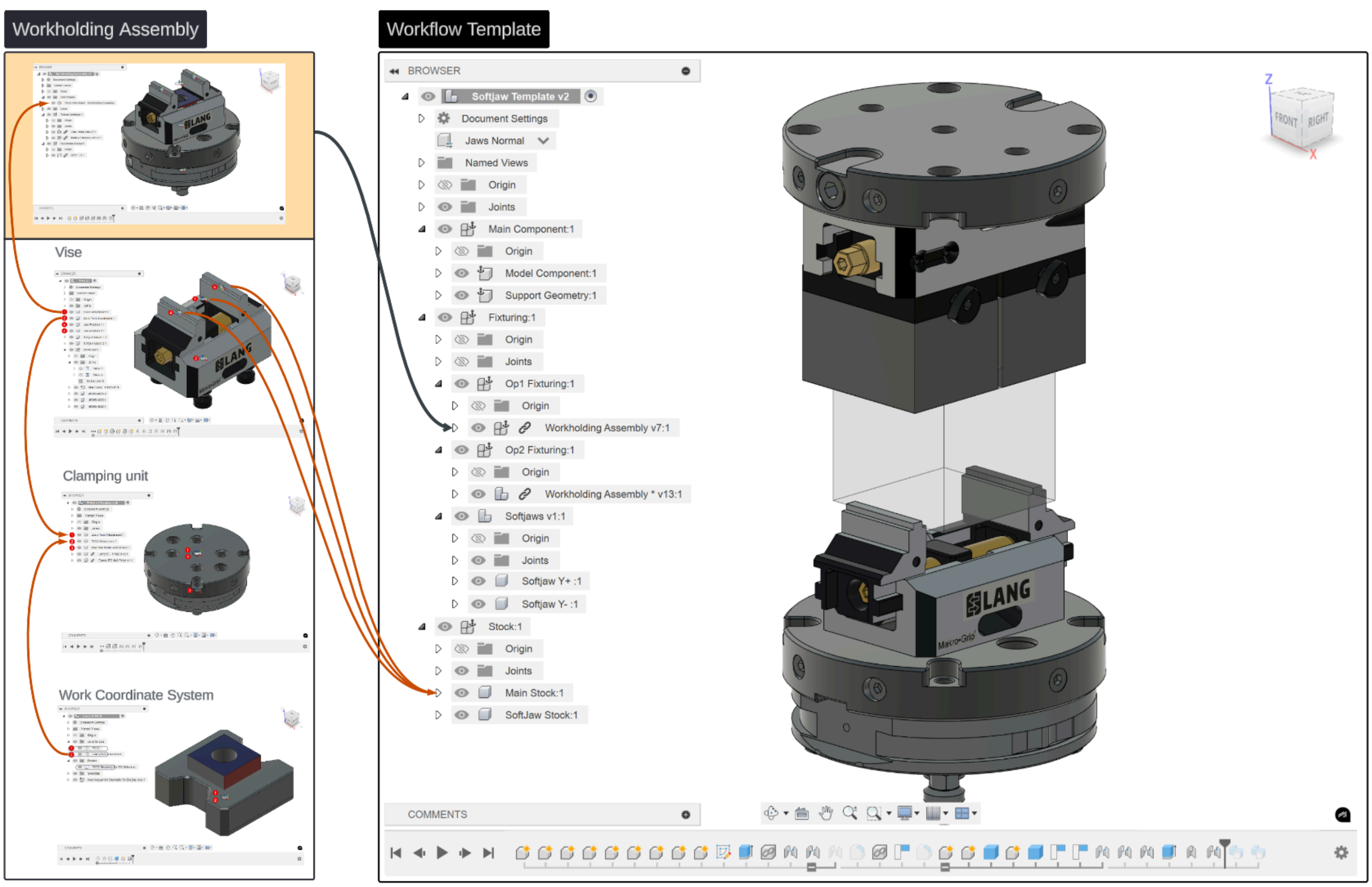
Configure the Insert for both of the Fixturing Assemblies to the configuration table, and you can now select what fixture is used for Setup 1, and Setup 2 in the Manufacture workspace.



Joints in the Template Framework

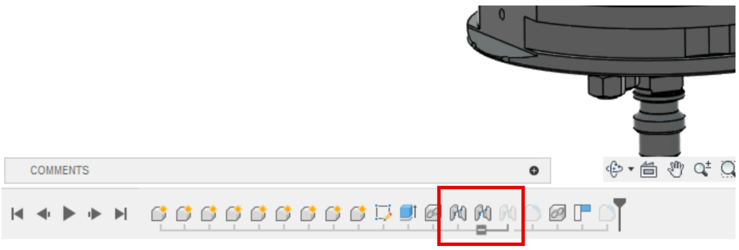
All of the Joints between the fixture and stock should update to the new geometry without any errors when we reconfigure each fixturing assembly.

This is because all of the joints in this framework are selecting Joint Origins that share a lineage with each other.

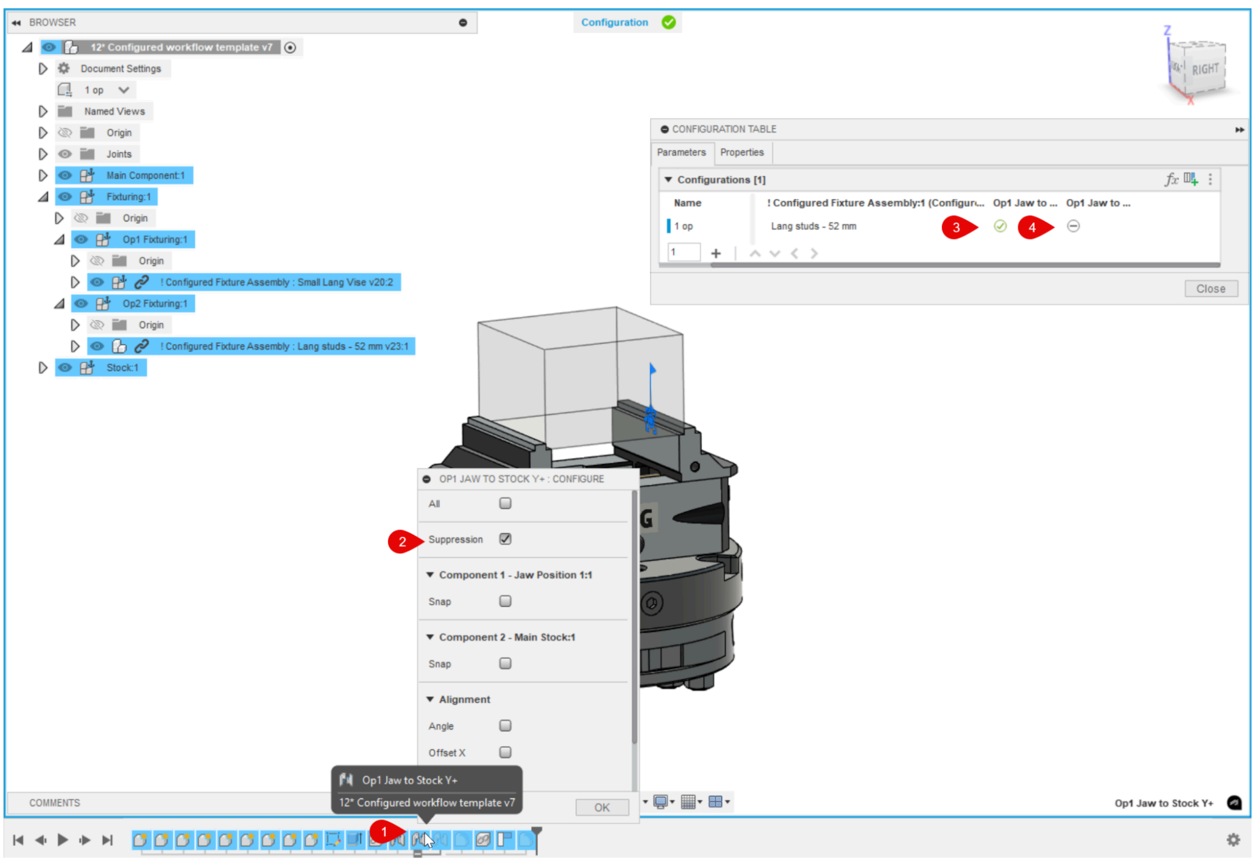


There are 3 Joints between a vise and the stock in a Template.
A self centering vise uses a Rigid Joint and one Jaw Attachment, and a fixed jaw vise uses a Slider Joint with both Attachments.

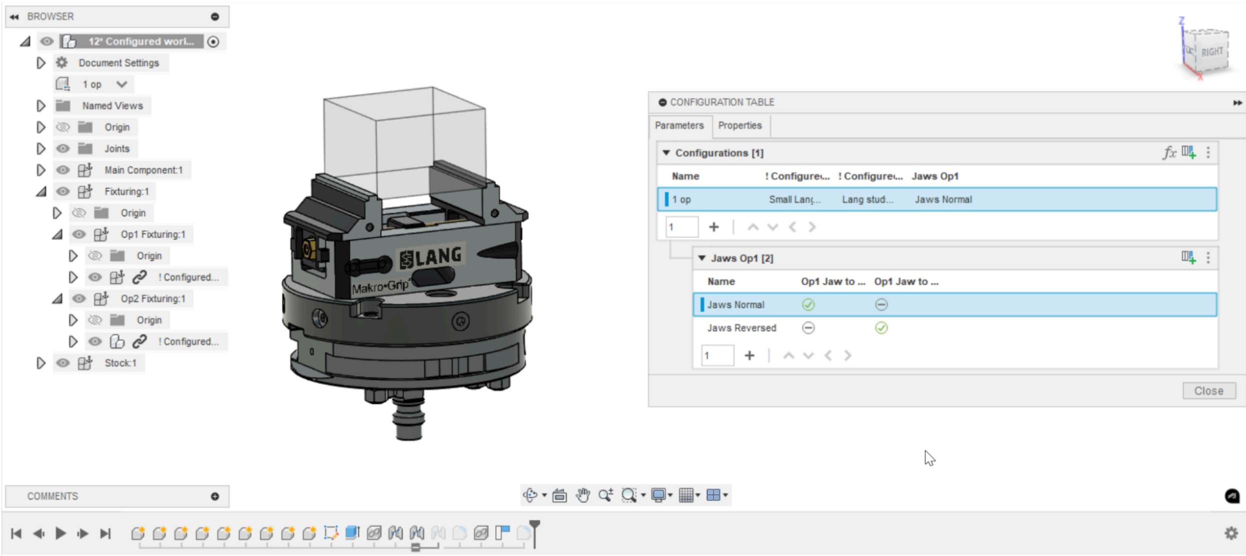
- **Rigid/Slider -** Stock Bottom to *Stock Attachment - Fixturing Assembly* Joint Origin
- **Planar -** Stock Y+ to *Jaw Attachment 1*
- **Planar -** Stock Y+ to *Jaw Attachment 2*

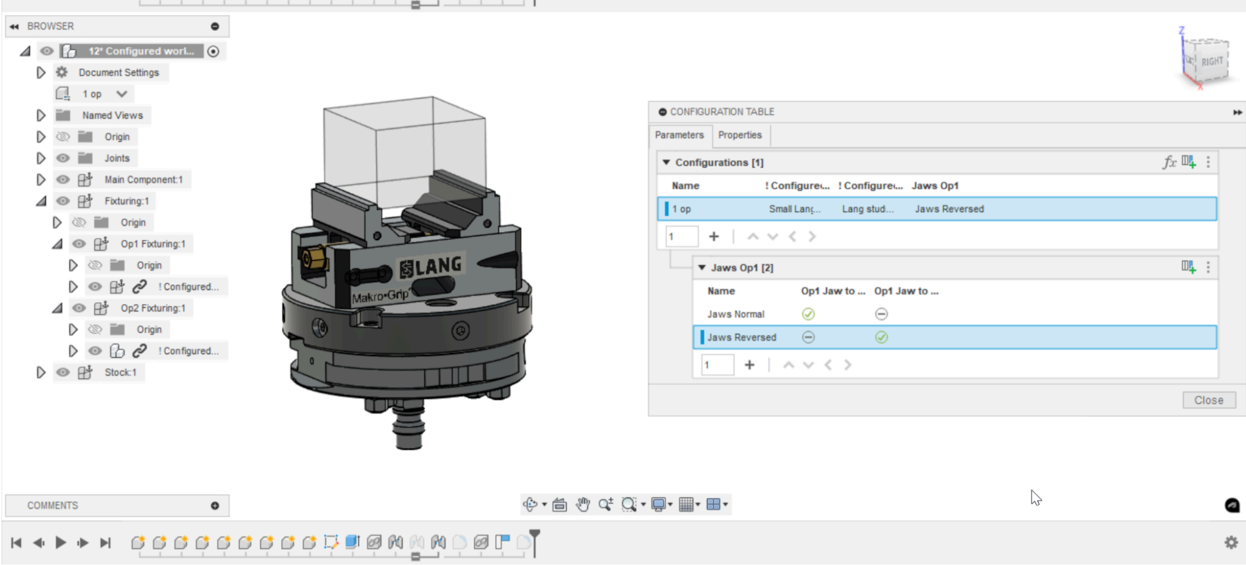


If all 3 Joints are unsuppressed, the two Jaw to Y+ Joints will be in conflict.
These two joints to the *Jaw Attachments* are grouped together in the timeline.
Use configurations to toggle the suppression of these joints, creating a reversed jaw, or fixed jaw functionality.

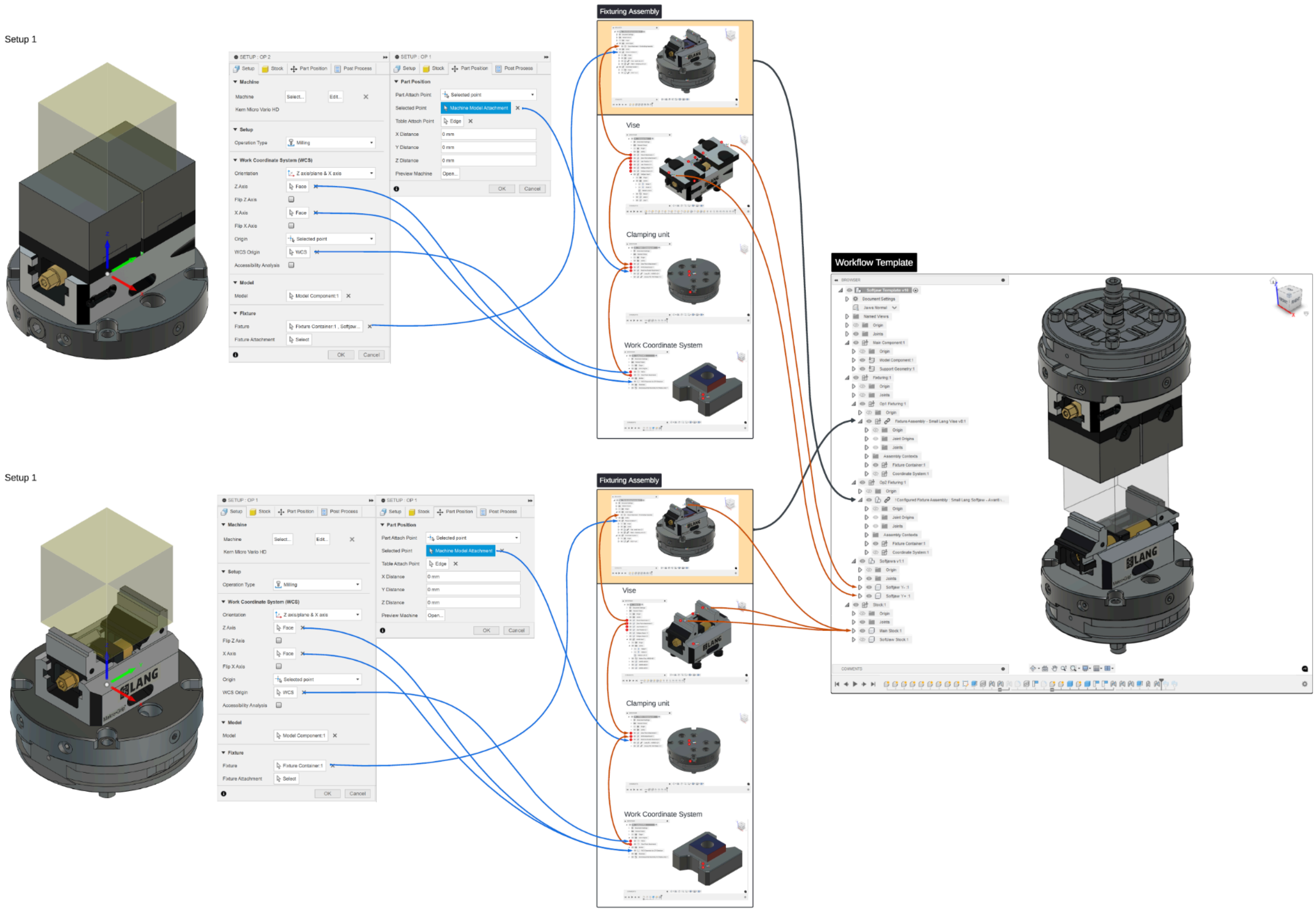


We can move these Joints into a new **Theme Table** For **Jaws Op1** and simply select which jaw configuration we would like to use.
Self centering vises should configure **one** of the Joints to the Y+ Stock face and Jaw Attachment to be active at any time.





Fixed jaw Vises will need to suppress the Rigid Joint to the *Stock Attachment*, and unsuppress a Slider Joint to the *Stock Attachment,* as well as a joint between *Jaw Attachment 2* and the Y- face of the stock. .



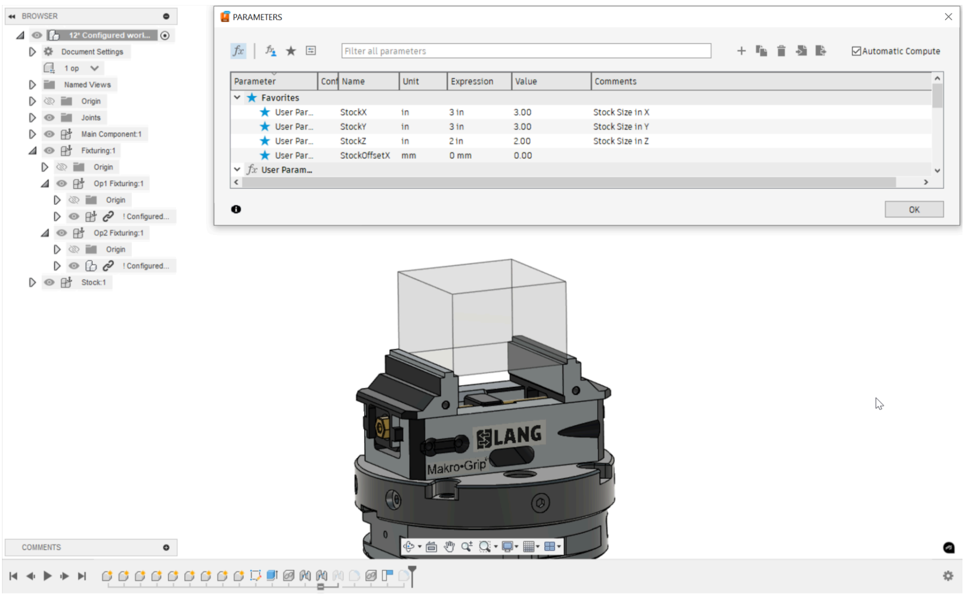
This Image is available in higher resolution on my github

Parametric Stock

Programmers need the stock to be editable in context of the Model, so it must be created in the Workflow Template design.

The Template Template has a handful of **User Parameters** that are driving the stock body size, and the offset of the Joint to the Stock to shift the stock within the jaws in the X axis.

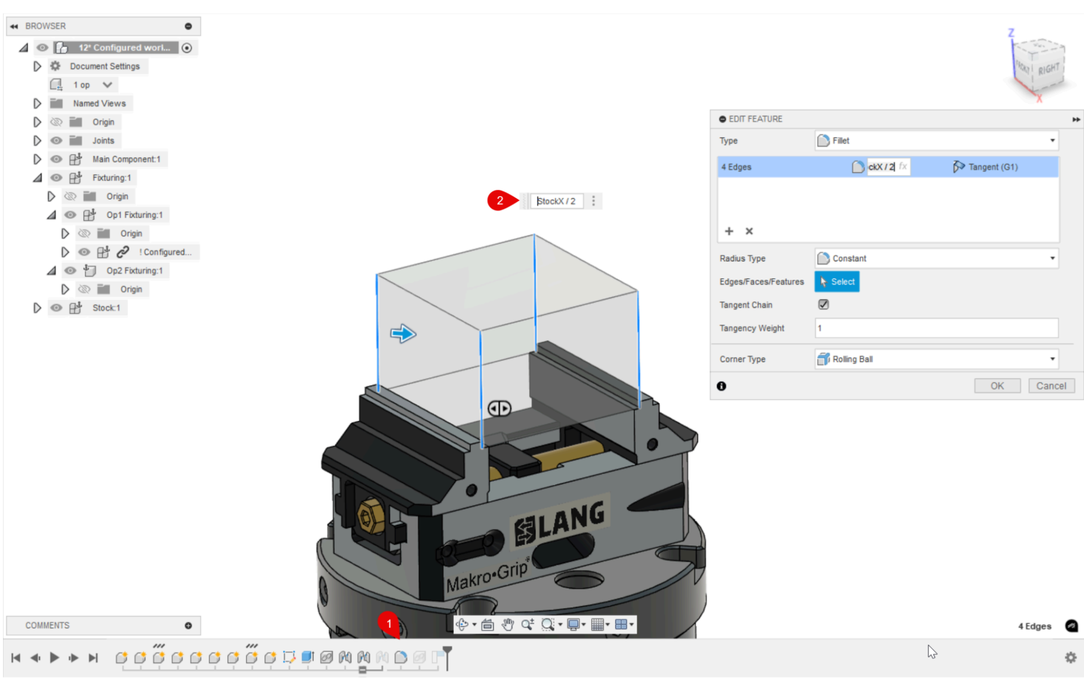
Similar parameters can be used to offset the stock to create a condition similar to parallels for fixed jaw vises



Round vs Square stock

There is a suppressed **Fillet feature** in the timeline that has each of the vertical edges of the stock selected, with a size of **StockX/2**

When StockX=StockY the stock geometry will become a cylind



https://conferences.autodesk.com/flow/autodesk/au2024/sessioncatalog/page/inperson/session/1714501559813001D0Kr

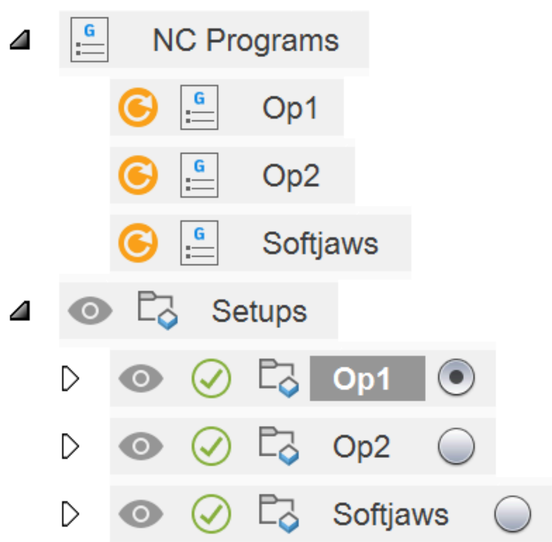
Section 5 - CAM Programming with Workflow templates.

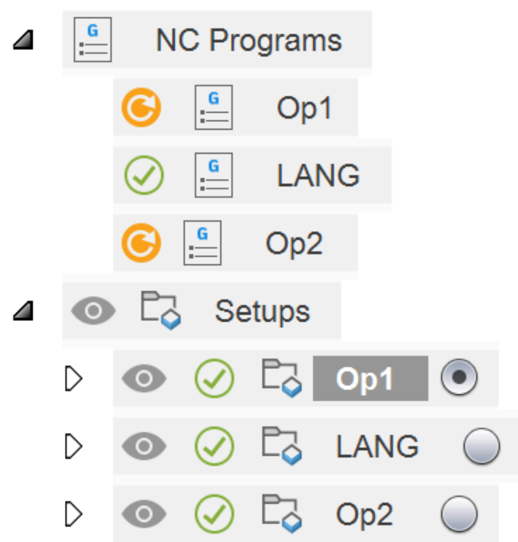
With a configured, or replicable fixturing assembly in the template framework, you can easily copy the workflow template template design, and begin defining your common workflows.

I consider any sequence of CAM setups to be a workflow, as the document will contain the context for the machine, fixturing, stock, and possibly toolpaths with machine specific tools.

Templates work best if they are fully pre-defined so your programmer may simply open, drop in a model and begin programming within the context of that specific machine, and setup sequence.

A simple template may be a single setup, but they really shine when you create a sequence of setups that share in-process stock with rest machining from the previous setups, or predefine multiple adjacent setups like softjaw/fixture machining.

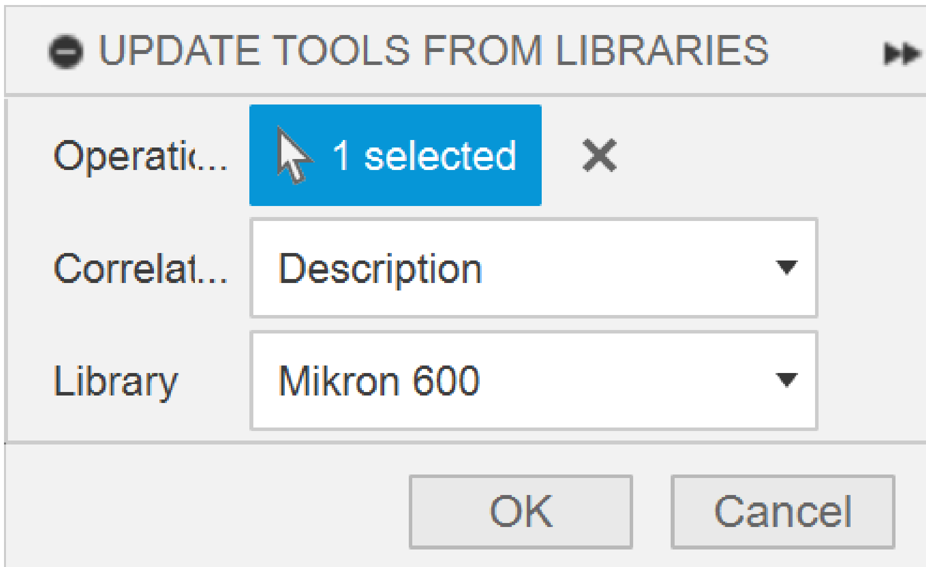




Since we cannot reconfigure the machine in the setup or NCProgram yet, many different versions of each workflow will need to be produced for each machine you want to support.

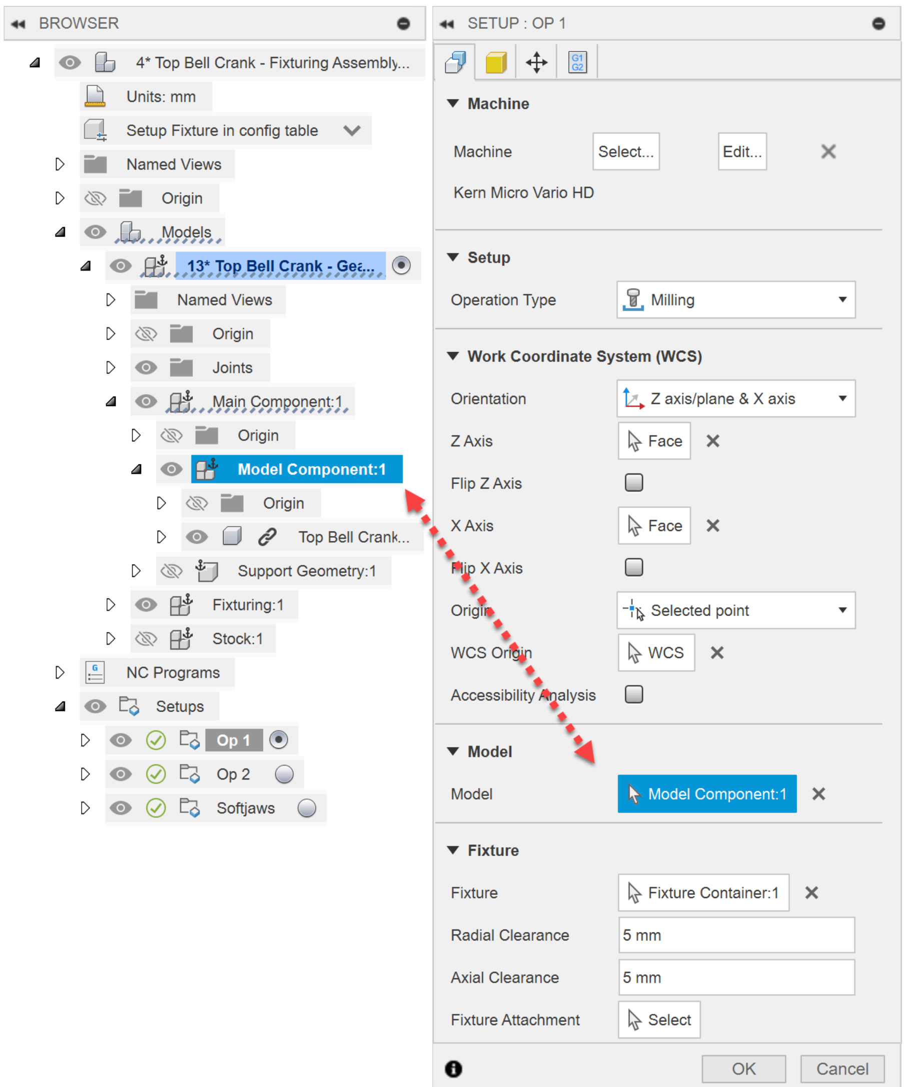
This is where the configured fixture assembly really shines. You can simply reconfigure the fixturing assembly to a different machine's fixture arrangement, update the machine and tools in the manufacturing workspace to migrate a program to a new machine.

I highly recommend the [Fusion Essentials Addin] to assist in moving programs from one machine to another, as the update tools function can replace tools in a document with those from a different cloud library if they are similar.



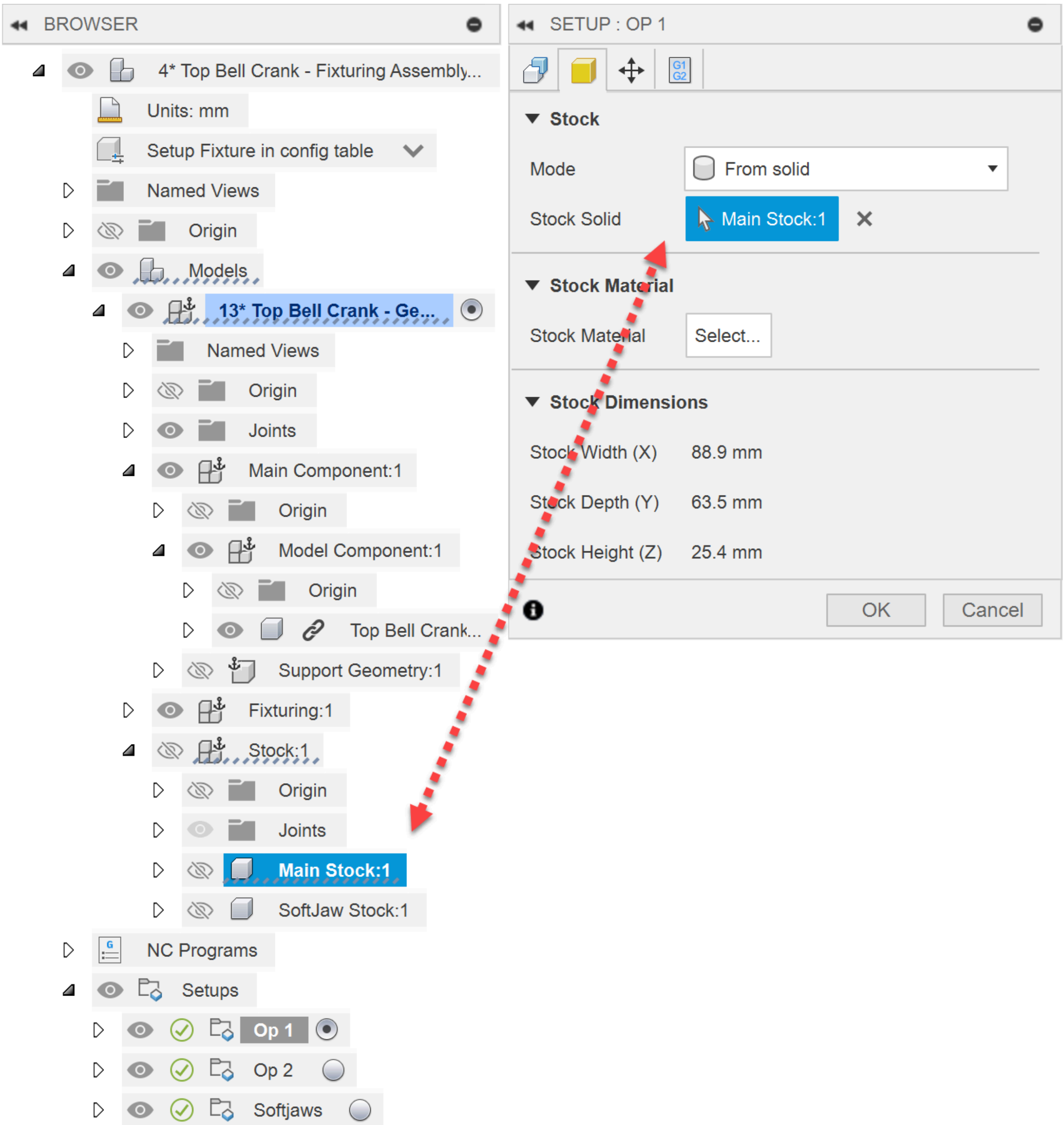
Containers as inputs to the setup.
The model container is selected by each setup by default, this means any model you insert into the design will be selected for machining automatically.

If you insert your model as an external reference and maintain the link (chain link or flag icon) you may replace the model with future versions, allowing you to very quickly reuse your old programming.



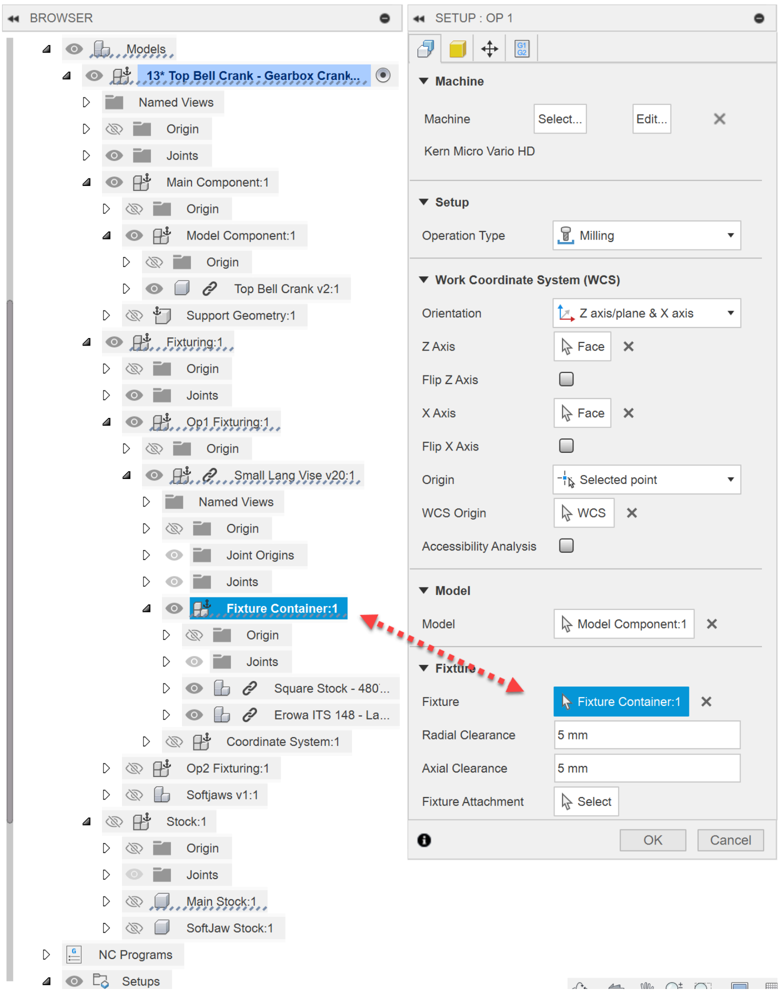
Stock container
The stock is modeled local to the template because it needs to be parametric and editable in context of the model.

Any bodies you add to this container will become part of your stock.



The fixture container

This container inside the fixture assembly will automatically select the vise and the clamping unit, even if they are reconfigured/replaced.



These items will be passed into the setup automatically, where we can assign a Radial and Axial clearance for all toolpaths inside the setup to keep away from the fixture, both axial and radially.

These values will be added to the values selected by the fixture offset values in the touch/avoid table under the geometry tab.