CP473669 - Debugging your Fusion Design: Let’s Get Rid of Those Red and Yellow Features

Jeff Strater
Sr Software Architect, Autodesk

Phil Eichmiller
Principal Quality Engineer, Autodesk
Phil Eichmiller

Phil Eichmiller is a Senior Software Quality Assurance Engineer for Autodesk, on the Fusion 360 team. He enjoys helping the Fusion online community and at night he shares his knowledge by teaching Fusion 360 for the CAD program at Portland Community College. Roller Derby is his favorite pastime, especially watching his daughters, who are both derby stars in Portland, Oregon.

Jeff Strater

Jeff Strater is a Senior Software Architect in the Fusion 360 team. I've been with Fusion since the very beginning. My focus is on general modeling/sketching. Before that, I was a developer and architect on Inventor, also before R1. So, I'm a long-time CAD guy. When not working with Fusion or its customers, I like to run, cycle, hike, and read science fiction.

About the speakers
Class Outline

• Class goals, and starting principles
• Intro to Errors and Warnings in Parametric Design
• Understanding the Causes of Errors/Warnings
• Debugging/Fixing Strategies
• Preventing Errors and Warnings
Class Goals and Some Starting Principles
Class Goals

• Teach you how to fix and prevent this:

• Design Workspace only, Parametric designs only

• Understand some of how Fusion detects and reports these errors
  o To help in your ability to deal with and prevent failures
Some Starting Principles

• Fix it when you see it
• Warnings are serious, too
• Compute All is your friend
• Editing when rolled back can hide errors
Some Starting Principles
Errors and Warnings in Fusion’s Parametric Design Environment
Errors vs. Warnings

• Error: A failure that prevents the feature from computing at all
Errors vs. Warnings

- **Error**: A failure that prevents the feature from computing at all
- **Warning**: A failure that still allows the feature to produce *some* result
- **Failure propagation**
  - Error – dependent features will also be in an Error state
  - Warning - downstream features can be OK
Parametric Feature Design 101

- Parametric design is a sequence of features
  - Entries in the Timeline
  - Computed in order
  - Order determined by feature dependencies
Feature Dependencies

- If a feature uses geometry from another feature, that is a *dependency*
- Always a result of a geometry selection in a command
Dependency Example
Feature Dependencies From this Example
The Causes of Errors/Warnings
What Causes a Design to Fail?

- Errors are always caused by edits to the design
- Two main categories of design errors:
  - Geometry Errors
  - Dependency Errors
Geometry Errors

• Happens when the modeling kernel fails

The fillet/chamfer could not be created at the requested size. This might be occurring at the ends of the selected edges. Try adjusting the size or using multiple separate operations. Check that the selected edge chain ends at a sensible position, and if not try selecting more edges.

The selected faces could not be offset. Try adjusting the offset/thickness value or changing the selection of faces.
Dependency Errors

- More common error after a design edit
- Caused when a Geometry Dependency fails
Types of Dependency Failures

- **No match found**

- **Too many matches found**
  - Expected to find N edges, found >N
  - Usually not a Warning/Error

- **Mixed results**
  - Some, but not all matches found
    - only applies to features than can accept multiple inputs
Two Kinds of Dependency

• **Geometry Dependency**
  - The geometry of the selection is required
    - Offset Workplane (needs the plane from the selection)
    - Sketch Edge Projection (needs the line or curve from the edge)

• **Topology Dependency**
  - The actual entity is required
    - Fillet (cannot fillet a line, you need the edge)
    - Shell (cannot shell a plane, you need the face)
Why is This Important to Know?

- **Geometry Dependency** failures result in **Warnings**
- **Topology Dependency** failures result in **Errors**
- **Why?**
  - Geometry can be cached. Topology cannot
Debugging and Error Fixing Strategies
General Strategies

- Read the error messages
- Fix Errors/Warnings when they happen
- Fix the Errors in Timeline order
Fixing Sketch Errors
**Sketch Errors**

- **Sketch Errors come in 3 flavors:**
  - Sketch Plane failures
  - Sketch solve failures
  -Projected geometry failures
Fixing Sketch Plane Failures

• Sketch Plane is a Geometry Dependency
  o Failure is a Warning
  o Only available fix is Redefine Sketch Plane
    ▪ Be careful here – Redefine can cause unexpected results
Fixing Sketch Plane Failures

- **Sketch Plane is a Geometry Dependency**
  - Failure is a Warning
  - Only available fix is Redefine Sketch Plane
    - Be careful here – Redefine can cause unexpected results
Fixing Sketch Plane Failures

- Sketch Plane is a Geometry Dependency
  - Failure is a Warning
  - Only available fix is Redefine Sketch Plane
    - Be careful here – Redefine can cause unexpected results
Fixing Sketch Plane Failures

- **Sketch Plane is a Geometry Dependency**
  - Failure is a Warning
  - Only available fix is Redefine Sketch Plane
    - Be careful here – Redefine can cause unexpected results
Fixing Sketch Solve Failures

- Usually, this is because of a conflict between dimensions and/or constraints
  - You need to find the conflict and remove it
Projected Geometry Failures

• Most common sketch failure
Fixing Projected Geometry Failures

- In the April, 2019 update, Fusion added “Manage Lost Projections”
Fixing Projected Geometry Failures
Fixing Modeling Errors
Geometry Modeling Failures

- Usually happen deep in the modeling kernel
- No general solutions here
- Each feature has different common failures
Fillet Geometry Failures

- Areas of high curvature
- Near tangent edges
Fillet Geometry Failures

- Areas of high curvature
- Near tangent edges
Fillet Areas of High Curvature Fixes

- Identify areas of high curvature using Curvature Comb Analysis
- Only fix is to reduce the curvature
- Use Curvature Combs in sketch, then adjust the spline for less curvature
Fixing Fillet Near Tangencies

- Identify the edges involved
- Add constraints to force tangency
- Text is hard to fix
Shell/Offset Geometry Failures

- Main error is self-intersecting surfaces
Fixing Shell/Offset Failures

• Main approach here is to reduce surface curvature
Shell Example Demo
Loft Geometry Errors

- Self intersections (detecting a pattern here?)
- Rail issues
Fixing Loft Self Intersections

- Modify the profiles/rails to correct the self-intersection
Fixing Loft Rail Failures

- For tangency failures, add tangent constraint
- For intersection errors, find the point of non-intersection, add constraints
Combine/Join/Cut/Intersect Geometry Failures

• Near coincident geometry
  o Split Body example:

  ![Split Body example image](image)

  Using face coincident to other faces fails

• Other examples of errors
  o Combine fails:
Combine/Join/Cut/Intersect Geometry Failures

Error

Failed to perform boolean operation.

Failed to Boolean bodies together
Fixing Combine Failures

- Avoid or fix any “near coincidences” in the geometry
- Make sure there are clear overlaps between target and tool
  - Especially with spline geometry – planes, cylinders, etc. are usually OK

Using a plane does not fail
Sweep Geometry Failures

• Self-intersecting geometry (again…)
Fixing Sweep Geometry Failures

- Reduce curvature, or profile size
- Cheat and use surface sweep
1 Reference Failures
The model is using cached geometry to solve. Please reselect reference geometry for failed features in the timeline.

Edge 1 missing
Dependency Modeling Failures

Fillet1

1 Reference Failures

- The edge reference is lost, try editing this feature to reselect the lost edge.

- Edge 1 missing

- Compute Failed
Dependency Modeling Failures

Sketch2

1 Reference Failures
The sketch plane is lost, Cache is used.

⚠️ Please redefine the sketch to select other plane!

Face 1 missing

Project1

1 Reference Failures
⚠️ The project source is lost, Cache is used!
Fixing Modeling Dependency Failures

• **Undo the edit that caused it**
  - Find another approach to achieve the same thing
  - Edit at the end of the Timeline

• **Use Edit Feature to re-select the failed reference**

• **Special case: Sketch plane dependency failure**
  - Use Redefine Sketch Plane
Fixing Assembly Errors
Joint Failures

- Most common joint errors are caused by modeling inaccuracies
Fixing Joint Failures

- Find areas of inaccurate modeling and fix them
- Add degrees of freedom to joints
Strategies to Prevent Errors From Happening
Dependency Management

- Minimize unnecessary dependencies
- Choose the most stable dependency possible
- Be careful when editing sketches
- Use parameters instead of dependencies
Minimize Unnecessary Dependencies

• Optimize preferences
• Be careful with sketch dimensions and constraints
• Use “To Object” sparingly
Choose the Most Stable Dependency

- In decreasing order of stability under edit:
  - Origin Work Geometry
  - Work Geometry that references Origin Work Geometry
  - Sketch Geometry
  - Solid/Surface Dependencies:
    - Body
    - Face
    - Edge
    - Vertex
Be Careful When You Edit Sketches

- The entire entity tracking system in Fusion is based on sketch object IDs.
- Deleting and re-drawing a sketch curve will get a new ID.
- **Edit** consumed sketches when possible, don’t delete.
- Be aware of some modification commands’ effects:
  - Fillet, Trim
Be Careful When You Edit Sketches

• The entire entity tracking system in Fusion is based on sketch object IDs
• Deleting and re-drawing a sketch curve will get a new ID
• *Edit* consumed sketches when possible, don’t delete
• Be aware of some modification commands’ effects
  o Fillet, Trim
Be Careful When You Edit Sketches

- The entire entity tracking system in Fusion is based on sketch object IDs
- Deleting and re-drawing a sketch curve will get a new ID
- **Edit** consumed sketches when possible, don’t delete
- Be aware of some modification commands’ effects
  - Fillet, Trim
Be Careful When You Edit Sketches

- The entire entity tracking system in Fusion is based on sketch object IDs
- Deleting and re-drawing a sketch curve will get a new ID
- **Edit** consumed sketches when possible, don’t delete
- Be aware of some modification commands’ effects
  - Fillet, Trim
Be Careful When You Edit Sketches

• The entire entity tracking system in Fusion is based on sketch object IDs
• Deleting and re-drawing a sketch curve will get a new ID
• Edit consumed sketches when possible, don’t delete
• Be aware of some modification commands’ effects
  o Fillet, Trim
Using Parameters Instead of Dependencies

• You can avoid geometric dependencies with parameters
• E.g. instead of “To Object” Extrude, use parameters to enforce geometry matches up
Bookshelf With Dependencies

- This version uses lots of geometric dependencies
- Each of those is a possible dependency failure under edit
Resources

- Forums
Resources

- Forums
- Learning and Documentation
Resources

- Forums
- Learning and Documentation
- Chat
Resources

- Forums
- Learning and Documentation
- Chat
- Support (Commercial only)