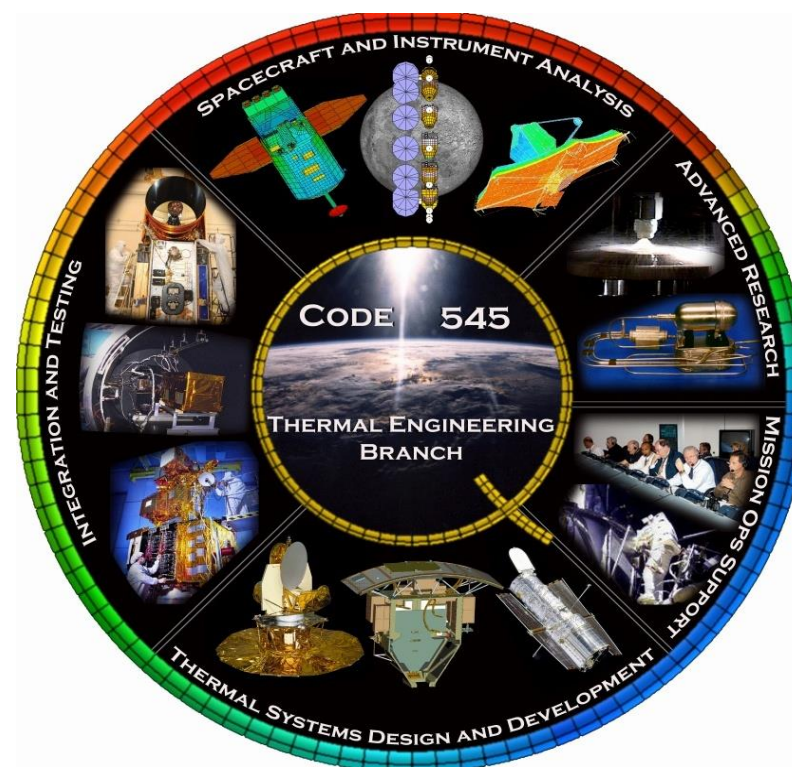# *Building Thermal Models*

Hume Peabody
Staff Engineer
Code 545
Hume.L.Peabody@nasa.gov
(301) 286-9141

Acronyms

1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional
α	Solar Absorptance
Alt	Altitude
ASSY	Assembly
Bij	Interchange Factor
BOL	Beginning of Life
CAD	Computer Aided Design
Cp	Specific Heat
DT	Temperature Difference
ε	Infrared Emissivity
ε^*	Effective Emissivity
EOL	End of Life
FD	Finite Difference
FE	Finite Elements
GOLD	Goddard Open Learning Design
GSFC	Goddard Space Flight Center
h_{cond}/h_{evap}	Condensor / Evaporator Heat Transfer Coefficient
HP	Heat Pipe
HR	Heat Rate
ID	Identifier

IF	Interface
Incl	Inclination
IR	Infra Red
k	Thermal Conductivity
MCRT	Monte Carlo Ray Trace
MLI	Multi-Layer Insulation
NCG	Non-Condensable Gas
PCM	Phase Change Material
PID	Proportional-Integral-Derivative
ρ	Density
RAAN	Right Ascension of Ascending Node
Radk	Radaition coupling
RES	Reservoir
SC	Spacecraft
SINDA	Systems Improved Numerical Differences Analyzer
STEP	Standard for the Exchange of Product Model Data
STOP	Structural-Thermal-Optical-Performance
TD	Thermal Desktop
TEC	Thermo Electric Cooler
UV	Ultraviolet-Visible
VCHP	Variable Conductance Heat Pipe
VP	View Port

PURPOSE

This presentation is meant to be an overview of the model building process

It is based on typical techniques (Monte Carlo Ray Tracing for radiation exchange, Lumped Parameter, Finite Difference for thermal solution) used by the aerospace industry

This is not intended to be a "How to Use Thermal Desktop" course. It is intended to be a "How to Build Thermal Models" course and the techniques will be demonstrated using the capabilities of ThermalDesktop. Other codes may or may not have similar capabilities…

The General Model Building Process can be broken into four top level steps:

1. ***Build Model***

2. ***Check Model***

3. ***Execute Model***

4. ***Verify Results***

BUILD MODEL: General Model Building Process

1. **Define Thermo-Optical and Thermophysical Properties**
2. **Get CAD from Mechanical Designer – STEP format preferred, make sure it is solids and not only surfaces**
   - *Process CAD into layers for controlling visibility*
3. **Build and define each CAD component**
   - *Geometric Surfaces, Nodes, Elements (Plate, Solid) – Add Comments to help*
   - *For Solid Element geometry, apply surface coat of zero thickness plates*
   - *Determine Node Subdivisions and Edge vs. Centroid Nodalization*
   - *Merge Nodes*
   - *Assign Node Numbers and Submodels*
   - *Assign Material and Thickness (Orienters for Solid Elements)*
   - *Assign Active Sides, Optical Properties, and Radiation Analysis Groups*
   - *Assign MLI*
4. **Determine Variable Parameters**
   - *Add Symbols (Thickness, Power, Multipliers, IF Values, etc) and assign to geometry if needed*
5. **Accommodate Moving Geometry**
   - *Define Assemblies and/or Trackers and Assign Geometry (Include nodes to move FE)*
6. **Establish Model Connectivity**
   - *Add Contactors between surfaces/edges at interfaces*
   - *Add Conductors between nodes/surfaces*
7. **Establish Boundary Conditions**
   - *Add and Define Heat Loads*
   - *Add and Define Heaters*

BUILD MODEL: Define ThermoPhysical and ThermoOptical Properties

- **Add ThermoOptical Properties:**
  - ✓ Open/Create Property Database
  - ✓ Add Property and Specify Name
  - ✓ Define α and ε
  - ✓ Include comment/source
  - ✓ Repeat until all properties added…
  - ✓ Repeat for EOL or other cases
  - ✓ Can import from other models
  - ✓ Define any Property Aliases that may be used (e.g. Radiator_Coating). Use ALIAS_ prefix
- **Add ThermoPhysical Properties:**
  - ✓ Open/Create Property Database
  - ✓ Add Property and Specify Name
  - ✓ Define ρ, C_p, and k (or k_x, k_y, k_z)
    - ✓ For MLI, define $\varepsilon*$
    - ✓ For PCM, define FUSION
  - ✓ Can import from other models
  - ✓ For MLI, recommend using ρ of 0.6 kg/m^3 and k and C_p of 0. For MLI assignments, define thickness for MLI as 1.0 m
    - ✓ This will allow mass of MLI to be estimated based on 0.6 kg/m^2
  - ✓ Define any Property Aliases that may be used (e.g. Panel_Material). Use ALIAS_ prefix
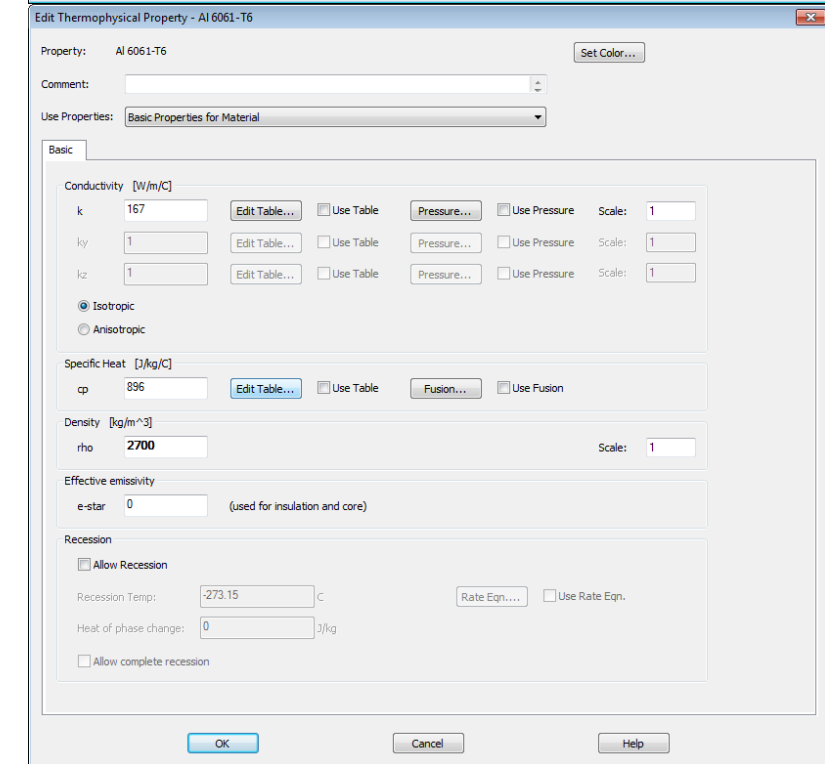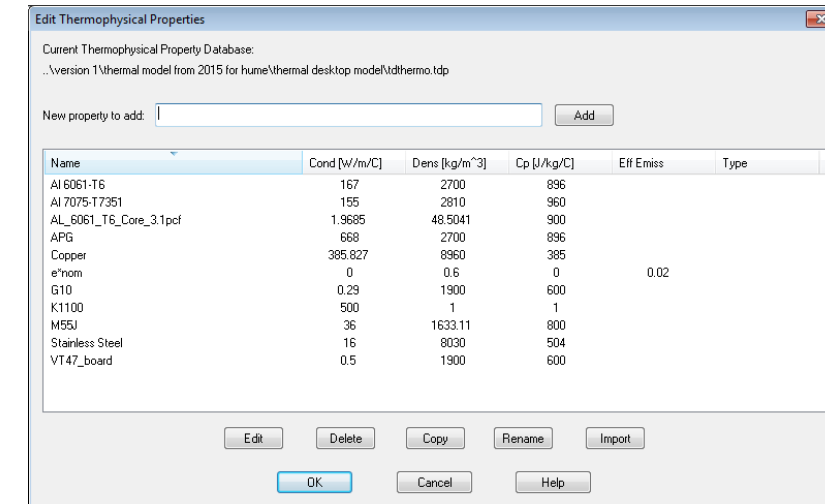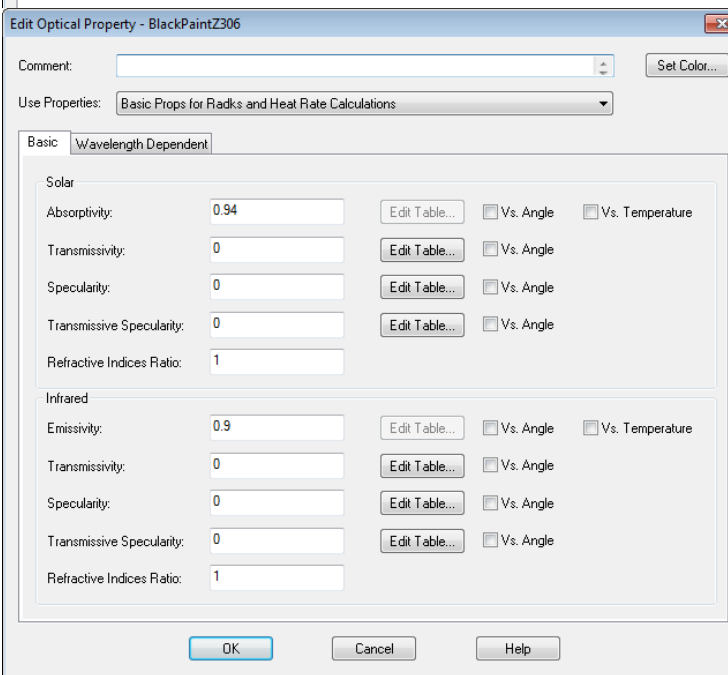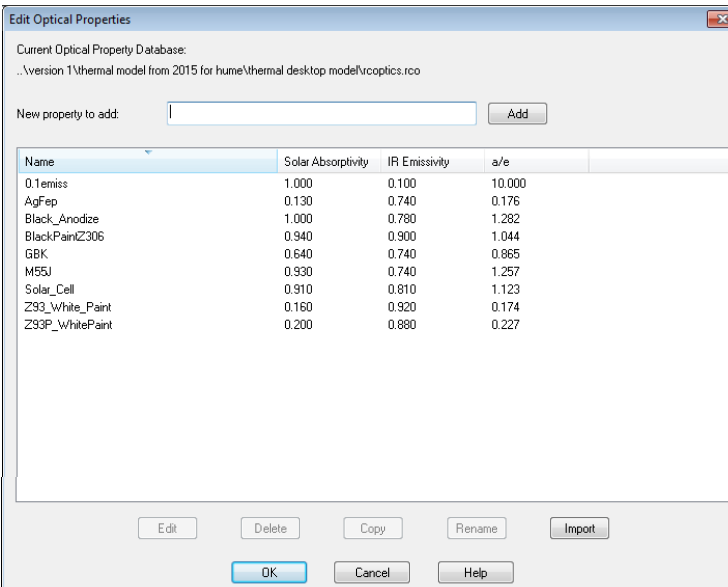
BUILD MODEL: Importing CAD

- To import a STEP file:
  - ✓ Type *import*, select file, and wait for notification in bottom right
    (Known issue with this command with AutoCAD 2014 in Windows 10)
  - ✓ Type *explode* and select object (this will explode the block into constituent parts)
  - ✓ Type *explode* followed by *'filter ('* makes this command transparent within explode command)
  - ✓ Select *Block* in the *Select Filter* DropDown and Click *Add to List* if not already there
  - ✓ Click *Apply* and type *all* at command prompt
  - ✓ If number of objects found is not zero, hit return twice (to end filter and to end explode)
  - ✓ Repeat the Explode/Filter combination until zero objects are found
  - ✓ At this point, everything should be a 3DSOLID or non-block entity type
  - ✓ Create layers for all the components for the design (*layer*)
  - ✓ All Solids can now be placed on layers and have colors changed to help with visibility control
  - ✓ Solids can also be copied and sliced (*slice*) to take measurements and "see" what is inside
  - ✓ Solids can have a material applied to them make them transparent
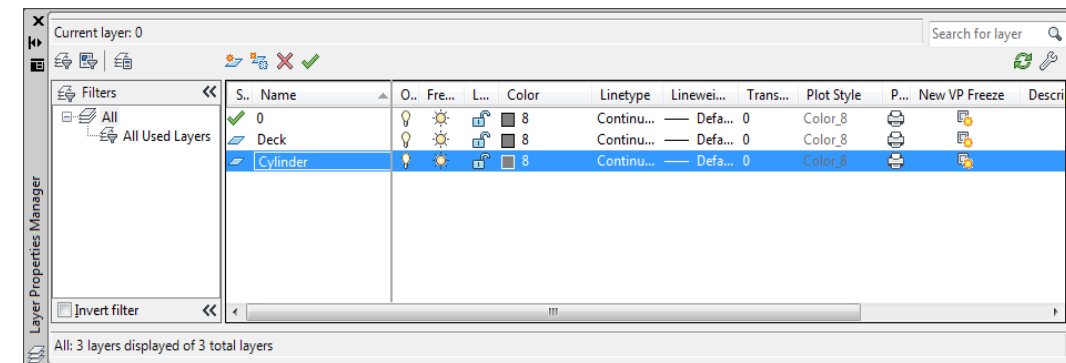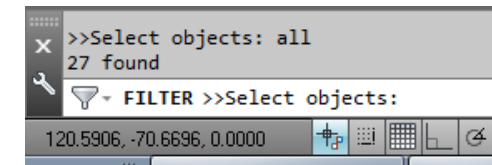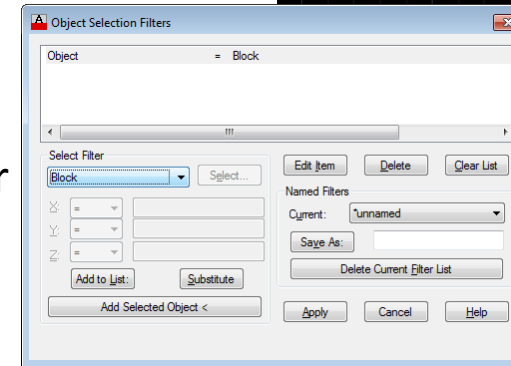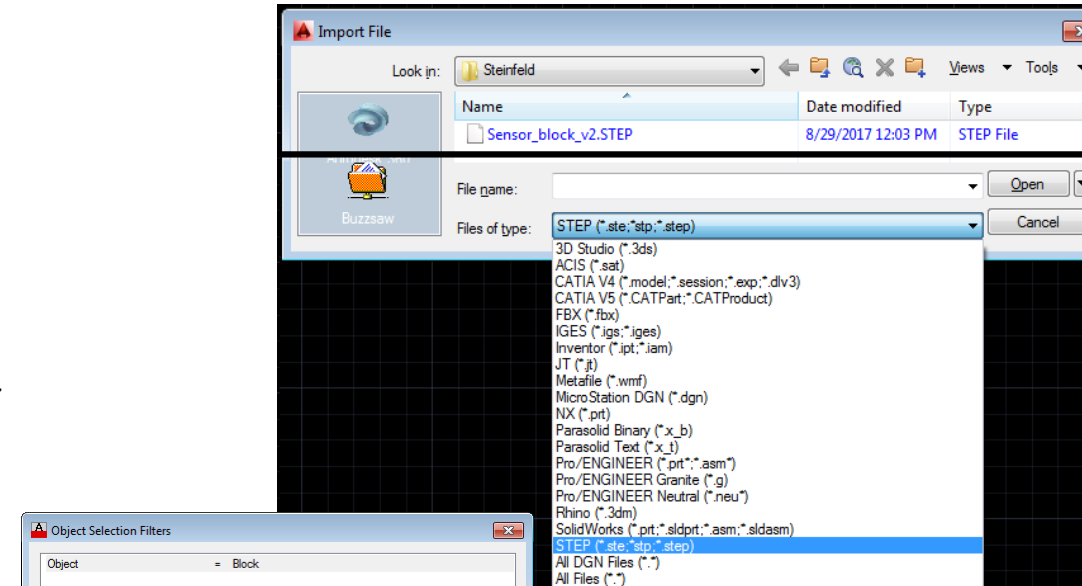
BUILD MODEL: Working with CAD

- Many functions help with using CAD:
  - ✓ ZOOM: enlarge and shrink image on screen
  - ✓ 3DFORBIT: rotate image in screen
  - ✓ OSNAP: allow user to select midpoint, endpoint, quadrant, intersection, node, center, perpendicular, nearest
  - ✓ UCS: define 2D working plane
  - ✓ MOVE: translate geometry
  - ✓ COPY: copy geometry within drawing
  - ✓ ROTATE3D: rotate geometry about specified axis by specified angle
  - ✓ MIRROR3D: create object that is mirror image of selected object about specified plane
  - ✓ 3DALIGN: align geometry based on 3 points in reference frame and destination frame
  - ✓ SLICE: cut solid objects by slicing plane
  - ✓ LINE: create line segment
  - ✓ DIVIDE: create N points along line segment
  - ✓ EXPLODE: convert assembly into selectable constituent parts
  - ✓ DIST: measure distance and angle between two points
  - ✓ ARRAYCLASSIC: create copies or an object as array (rectangular or polar)
  - ✓ PROPERTIES: change layer, color and linetype for selected objects
  - ✓ LAYER: allow visibility, color, linetype changes for specific layers
  - ✓ ORTHO: toggle if second point is orthogonal (along X or Y) to first point
  - ✓ FILTER: allow removal or inclusion of selection objects based on properties
  - ✓ FILLET: connect two, non parallel lines
  - ✓ ERASE: remove object from drawing database
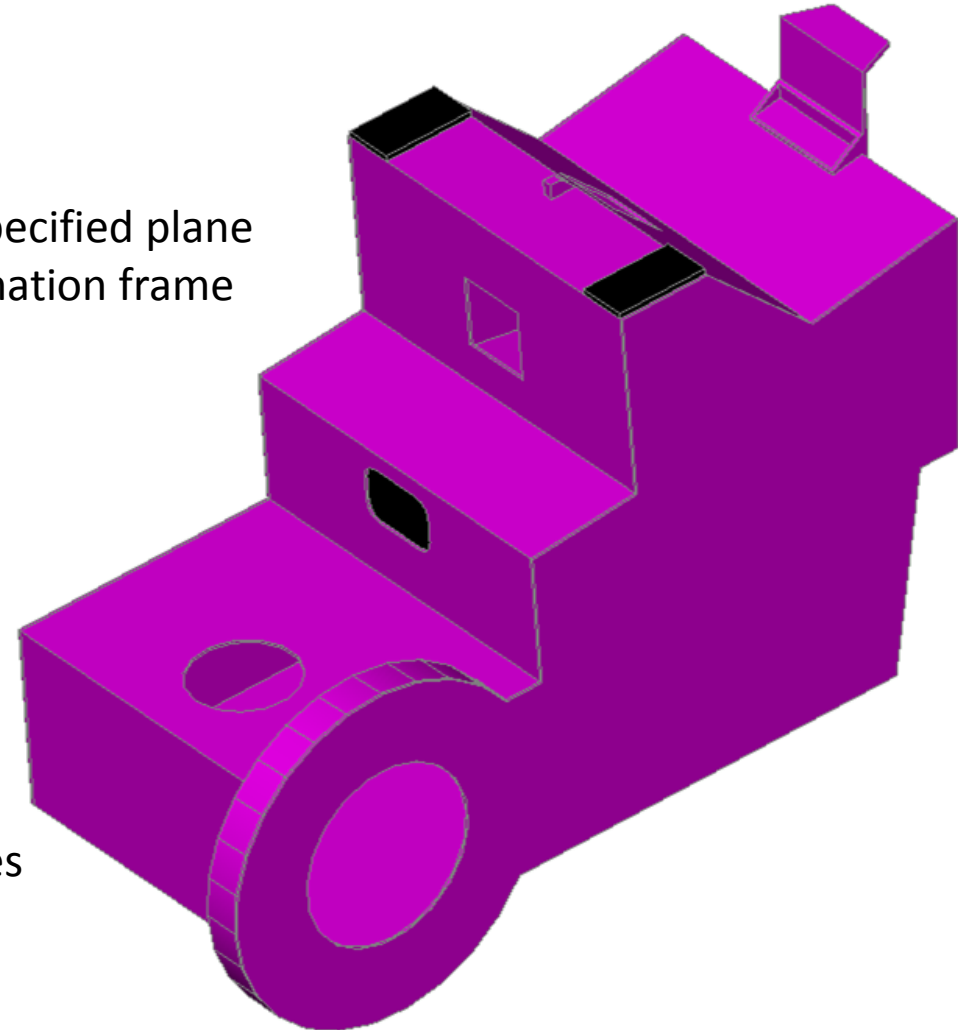
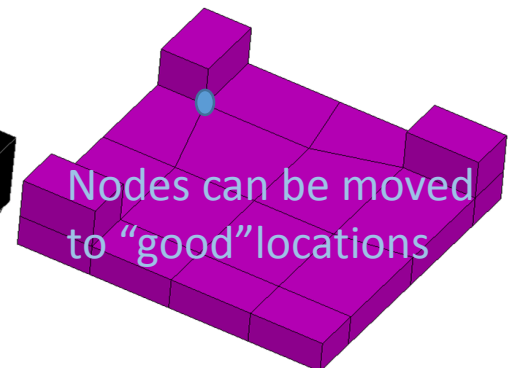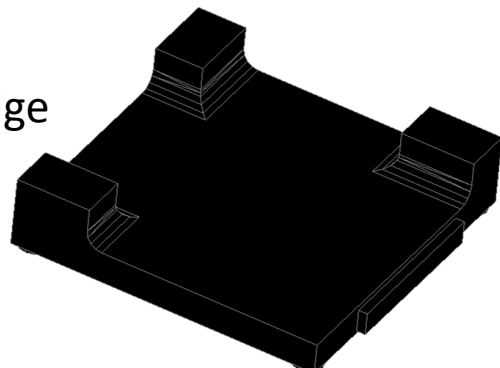# BUILD MODEL: Building Geometry

- Most models represent 3D CAD with 2D simplifications
- The representations include additional properties that make them thermal
  - Thermo-optical properties and radiatively active sides
  - Thermo-physical properties and thickness
  - Node assignments and subdivision
  - *The node assignments form the basis for a network thermal model which is used to compute temperatures and heater powers. It is the link between the geometric and thermal models*
- Some surface shapes follow naturally occurring geometry
  - Rectangle, Cylinder, Disk, Cone, Triangle, Sphere, Paraboloid, etc.
- Other 2D shapes do not conform to simple geometry
  - For these, the use of finite elements is recommended, which allows a user to more closely match CAD geometry
- Sometimes, it is necessary to model heat flows as three-dimensional
  - Solid Primitives (Box, Disc/Cylinder, Sphere, Cone) exist which provide the same properties as their 2D counterparts, but allow for a network based on 3D subdivision
  - For more complex shapes, 3D finite elements (Tetrahedron, Wedge, Brick, and Pyramid) are available. Note that these do not include radiation properties…
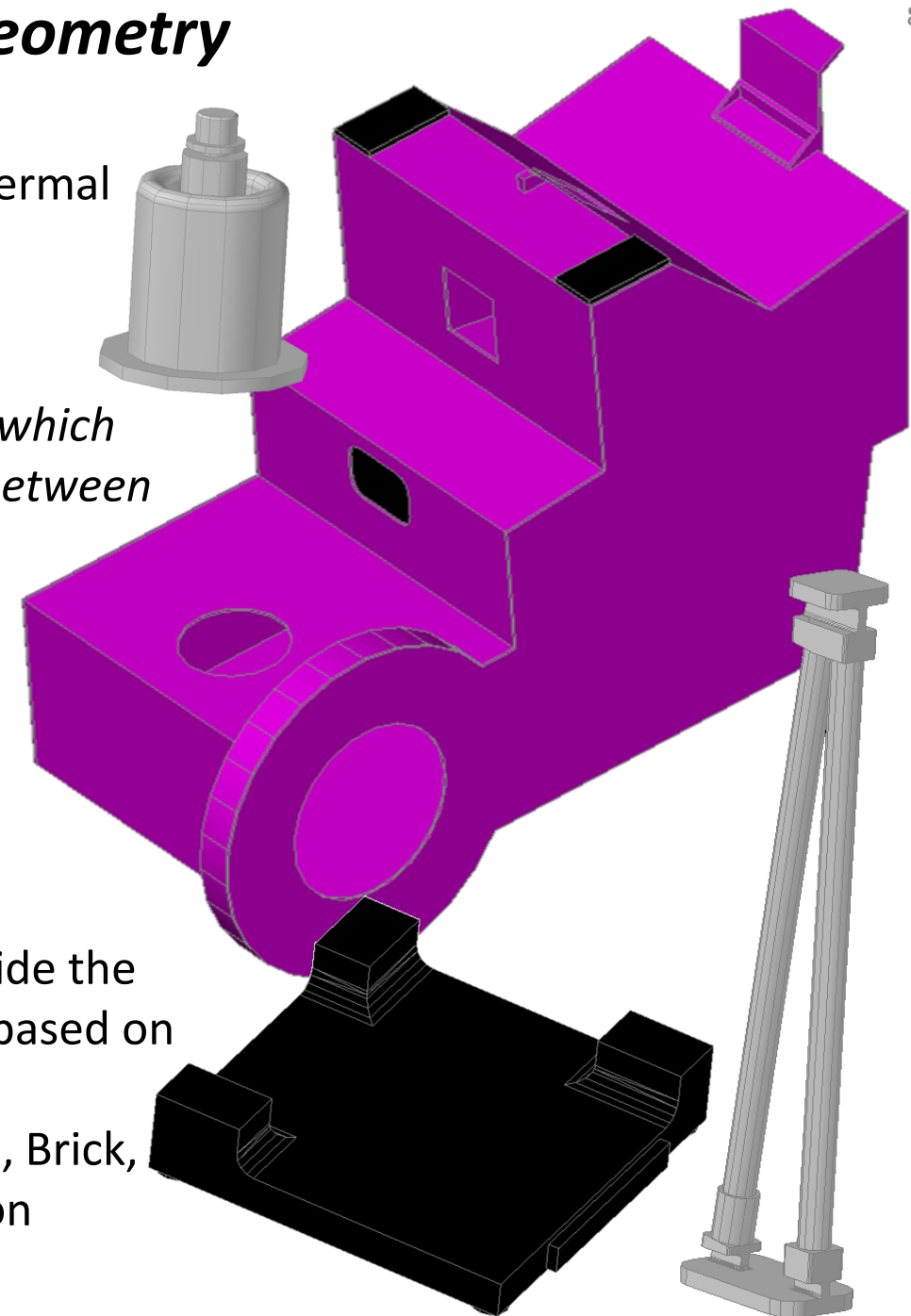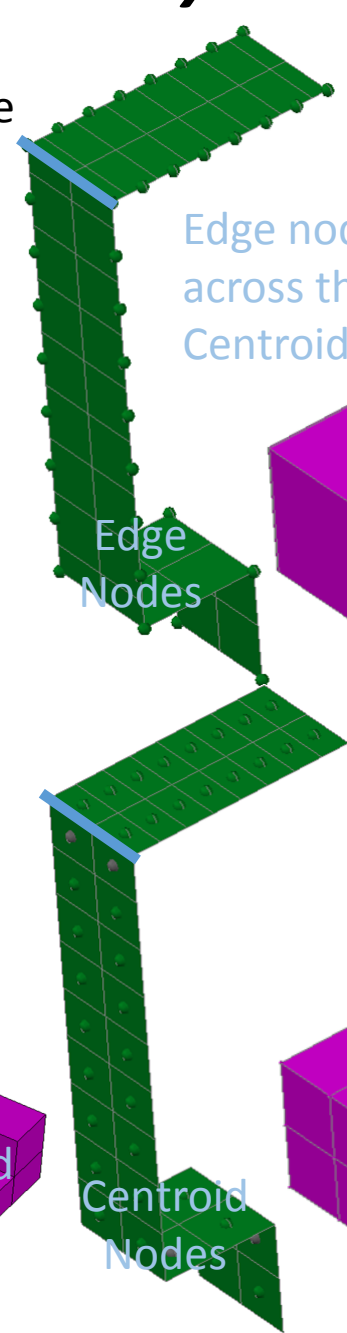
BUILD MODEL: *Meshing Geometry*

- Use of Edge Nodes recommended over centroid nodes
  - ✓ Centroid and edge nodes both account for conduction *within* a single surface
  - ✓ Edge nodes move nodes to edges of each sub-surface instead of centroid
  - ✓ Edge nodes that are merged allow conduction between surfaces via the shared edge nodes
  - ✓ Use of finite elements allows for more complex shapes to be modeled
  - ✓ Building elements by hand not as time consuming as may be thought
    - ➢ Define nodes first by placing user nodes at location on CAD geometry
    - ➢ Define element by selecting nodes in right hand rule order to define "top side"
  - ✓ Solid elements can also be defined and allow non-isotropic properties
    - ➢ Manually define by selecting nodes (one face then the opposite)
    - ➢ OR Revolve/Extrude Plate elements
  - ✓ Benefit of elements is ability to move nodes to desired locations
  - ✓ Elements cannot be moved…
    Elements defined by nodes…
    Move the nodes instead
  - ✓ Shell coat solid elements
  - ✓ Define surfaces as either edge Node or centroid node
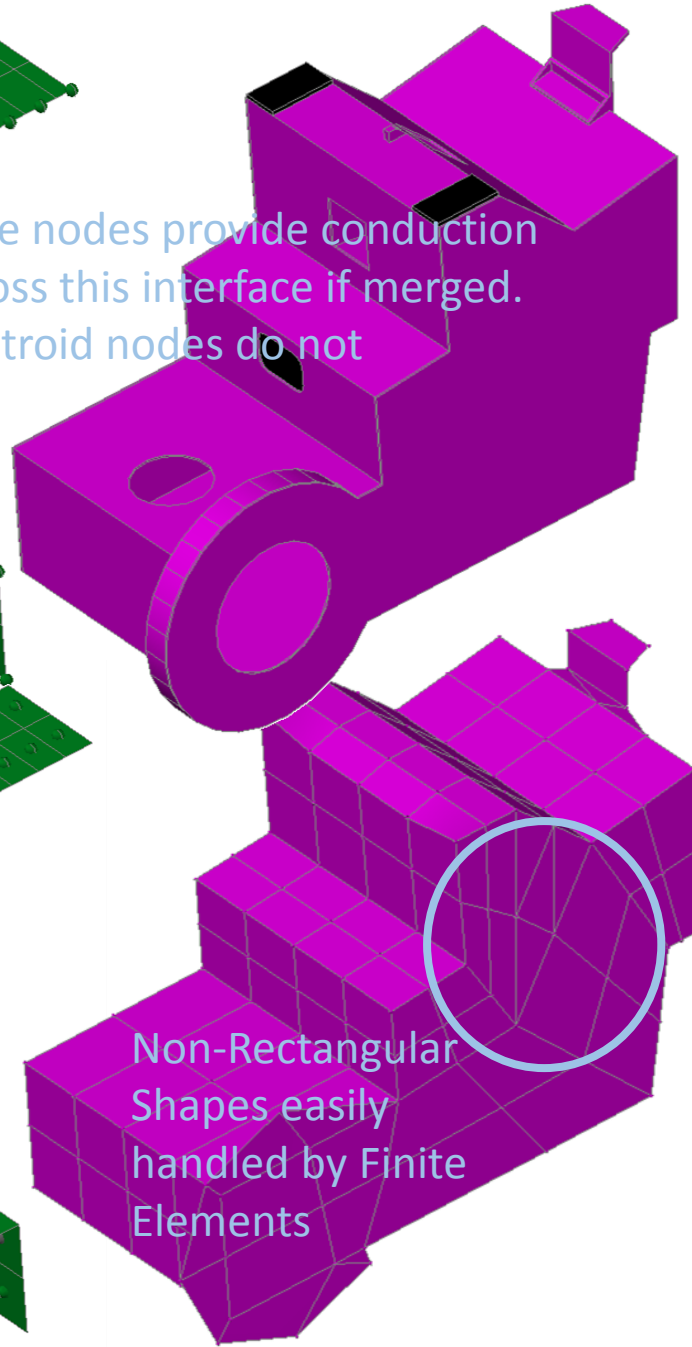 - ✓ **Merge nodes**
  - ✓ Assign rest of properties…
- *Extremely important!!!!*

Edge nodes provide conduction across this interface if merged. Centroid nodes do not

Edge Nodes

Centroid Nodes

Nodes can be moved to "good" locations

Non-Rectangular Shapes easily handled by Finite Elements

BUILD MODEL: Working with Finite Elements

- Finite Elements are a powerful tool, but can be more difficult to work with than primitive surfaces
- Elements cannot be double sided and the node numbers are not able to be changed by the element. You can change the node number of the node itself and any elements that reference it will be updated, but you cannot change a node number from the element properties like you can for a primitive surface.
- Element shapes are defined by the nodes. Move a node to a different location and the shape changes.
- Material Orienter may be needed for solid element to define axes for anisotropic material property
- Options exist under Modeling Tools that are useful for working with elements
  - *Refine Elements*: subdivides elements to convert a quad element into 4 quads based on the midpoint of each edge. Similarly this can be applied to triangular elements and creates 4 tri elements.
  - *Reverse Connectivity*: At times, the top side of an element is defined incorrectly. The connectivity can be reversed to switch the top and bottom sides
  - *Shift Connectivity*: If using edge contactors with elements, it often assigns the first, second, etc. edge for making contact. The marker is displayed along the specified edge(s). However, instead of having to guess which edge is the first, the connectivity can be shifted to redefine the first edge. The marker will visible shift and this command should be repeated for each element until the markers line up along the desired location
  - *Convert FD to FE*: Elements cannot be subdivided. A rectangle defined with edge nodes can be converted to elements (FD/FEM Network), but a 3x5 edge node breakdown become 8 individual elements
- Options also exist under FD/FEM Network that are useful for working with elements
  - *Extrude/Revolve Planar Elements*: You can extrude a set of planar element along a line or revolve around an axis to define 3D elements. The number of elements along the extrusion can also be specified
  - *Surface Coat*: Solid elements do not radiate and cannot be used with a contactor. Therefore, to use these features, a plate element of zero thickness is needed. Surface Coating created zero thickness plate elements on all free faces of a group of solid elements. Free faces are defined as faces that are not shared between elements.

BUILD MODEL: Node Discretization

- Common question: **How many nodes should I use?**
- The answer depends on the design…
  - Thick, high conductivity material structure…fewer nodes
  - Thin OR low conductivity structure…some nodes
  - Thin AND low conductivity structure…many nodes
- Decide as an analyst your maximum allowable temperature between two nodes. (0.5 K, 2 K, 5 K? and where)
  - ✓ If two nodes are very close in temperature, they can be merged/eliminated
  - ✓ If two nodes are very far apart in temperature, you may need more in between
  - ✓ Do not force two nodes to be at nearly the same temperature with a very high coupling…merge instead
- If conduction is…
  - ✓ Mostly 1D, consider only discretizing in one direction (along strut, bar, or heatpipe)
  - ✓ Mostly 2D and planar (e.g. thin panel), you should discretize in two directions
  - ✓ Mostly 2D and non planar (cylinder, cone), consider uneven loading from environment or radiation when deciding on circumferential discretization
  - ✓ Mostly 3D, use solids or solid FE
- Often it also comes down to a matter of convenience for modeling conduction across an interface
- After you have run your model for the first time, investigate if the gradients warrant more or fewer nodes
  - Best to stay with surfaces and edge nodes if possible while determining this.
  - It is easy to renodalize a surface; it is far more difficult to remesh elements (unless you use the TD mesher)
  - Can Refine Mesh in TD which makes an FE Tri into 3 Tris and a Quad into 4 Quads…cannot go back though

BUILD MODEL: Assigning Properties

- Once surfaces/solids are geometrically defined, remaining properties need to be defined in ThermalDesktop
  - ✓ *Subdivision*: X,Y,Z breakdown and subdivision, Edge vs. Centroid*
  - ✓ *Numbering*: Submodel and Node Numbers (multiple submodels allowed)*
  - ✓ Radiation: Active Sides, ThermoOptical Properties, Overrides
    - ✓ Active sides defined for each Radiation Group
  - ✓ Cond/Cap: Thermophysical Material, Thickness
  - ✓ Insulation: Top and Bottom Sides, Insulation Material, Overrides
  - ✓ Contact: **Do Not Use**
  - ✓ *Surface*: Dimensions, Comment*
    - ✓ *Comment for Finite Elements (not Surface)*
  - ✓ *Trans/Rot: Additional rotations and translations*

* These properties not applicable to finite elements

- Each node is an independent calculation point in SINDA
  - ✓ Use duplicate numbers if you intend for the temperatures to be the same (e.g. @ I/F)
- If node numbers are duplicated and nearly coincident, best to merge them into single node entity
- Use submodels for organization to identify component associations (e.g. Opt_Bench, Scan_Mir, FPA)
  - ✓ Too many submodels gets cumbersome in the ModelBrowser…don't go overboard
- But what node numbers should be used?
  - – Good practice to number nodes according to their location within an assembly
  - ✓ Best to keep MLI offset at 100000 for one side and 100000/200000 for both
  - ✓ Number first portion of a component in the 1xxx range, the next in 2xxx, and so on…
  - – Could start with 1xxxx range, but run out of "blocks" sooner before the 100000 range
  - – If you have more than 1000 nodes in a component either skip ranges (e.g. 1000-2999, 3000-3999, 4000-4999, etc) or use a block of 10000 for the entire component
- Why bother?   Why not just renumber everything and let TD display the contours?
  - – Helpful for navigating model to have ranges…If 1000-1999 is the motor shaft, can turn its visibility on and off in the ModelBrowser by selecting that range
  - – Helpful for post processing…if all 100000 nodes are MLI, can likely ignore their temperatures. If 1000-1999 is the motor shaft, can apply limits to those nodes knowing what they represent
  - – Helpful for documentation…likely you will not be the last user of a model. The next person who picks it up and has to work with it will have an easier time learning the model if it is organized
- To propagate numbering scheme in Z direction…
  - ✓ Renumber all nodes to 100000
  - ✓ Renumber first layer of nodes to 1000
  - ✓ Copy 1st layer of nodes in z over top of 2nd layer
  - ✓ Merge nodes and keep Smallest Node ID
 - ✓ Renumber with Add Increment checked
  - ✓ Repeat
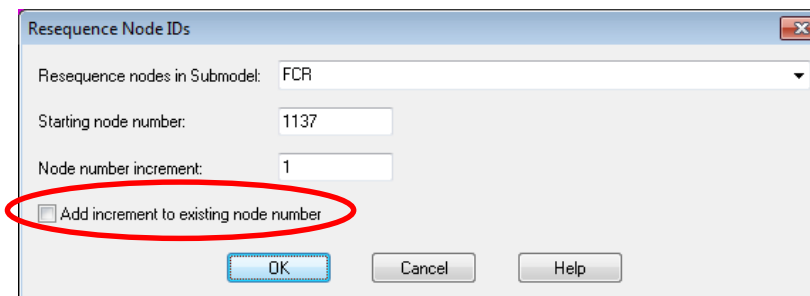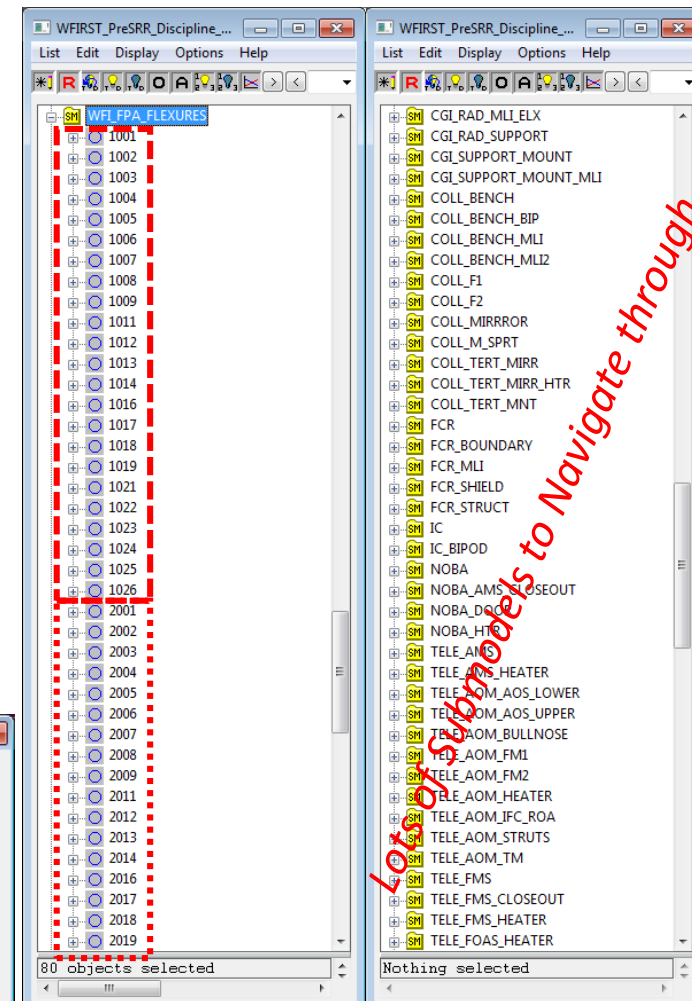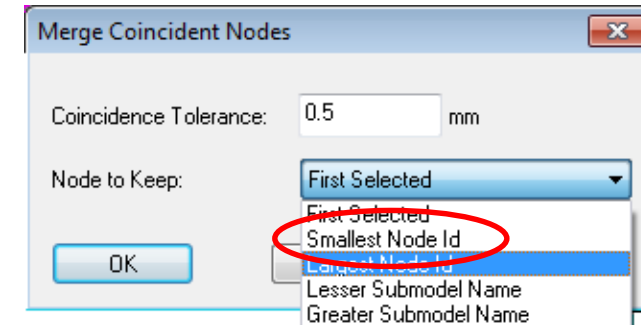
BUILD MODEL: Radiation Groups

- While Thermo-optical properties are specified for a surface, it does not necessarily mean that the surface participates in the radiative calculations
- Radiation Groups define collections of surfaces (i.e. enclosures). Only one Radiation Group is considered at a time during calculations
- For each Radiation Group, participation in calculations (activity) can be specified
- Multiple Radiation Groups can be used to define different configurations
- Radiation groups can be OR merged (If Top is active in on and Bottom is active in the other, OR merging makes Both sides active)
- They can also be copied to spawn a different configuration which may be used for changes in the geometry such as use of different instrument models or removal of a cover

✓ Define the Radiation Groups before assigning activity

✓ For each surface, assign whether the top side, bottom side, both sides, or neither side participate in the solution. Additionally, Not in Group can be used for surfaces that should be excluded completely from calculations

- Active None vs. Not in Group: Active none will still reflect and absorb energy; it just won't be tallied for the surface. Not in Group: surface is not even there

✓ Once activity is assigned, visualize using Model Checks-Display Active Sides. This will be for a specified Radiation Group. Colors indicate activity: Yellow ((Both Sides), Green (Visible Side), Light Blue (Not Visible Side), Dark Blue (Neither Side), Red (Not in Group)

✓ Be sure to check activity for every group that will be used in calculations…
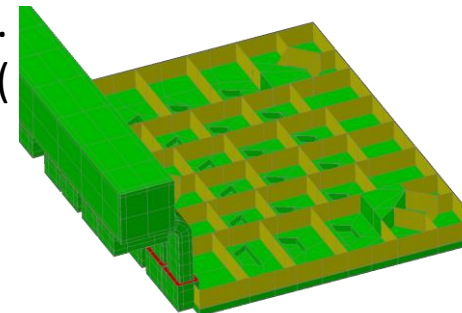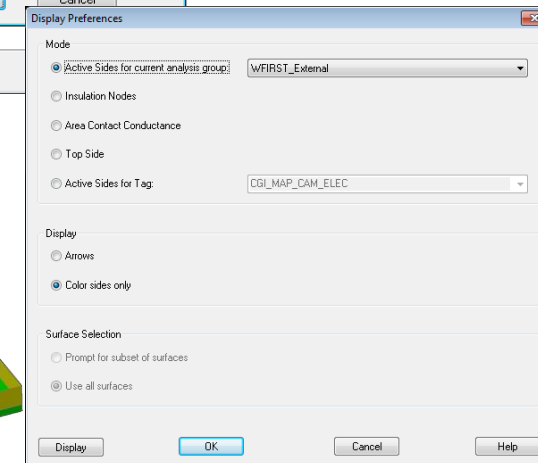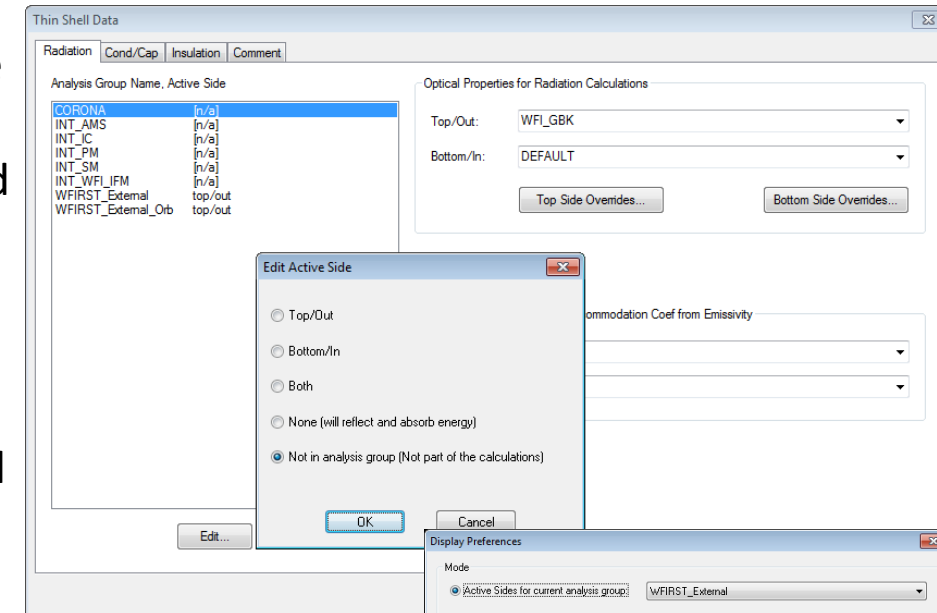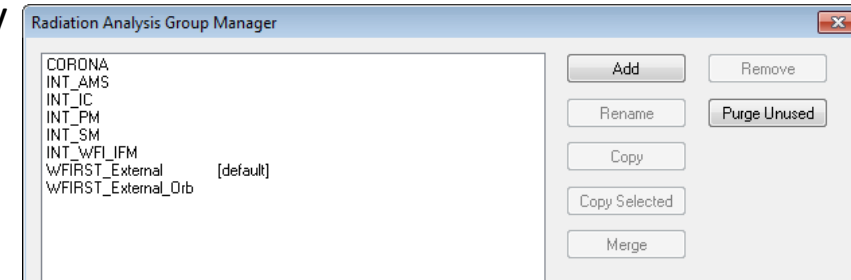
BUILD MODEL: Controlling Visibility

- Seldom is it useful to display the entire model. Most often, only one part of the model is being updated at a time
- Visibility control allow only parts of the model to be show. Three ways to control visibility: Layers, Global Entity Visibility, Object Level Visibility
- Layers: (layers command, native to AutoCAD)
  - If a layer is Off or Frozen (Blue Light Bulb or Snowflake) anything on that layer will not be displayed
  - Also can control from drop down along toolbar
  - HINT: If something isn't showing up in post processing, make sure the layer is not frozen in the VP too!
- Global Entity: (Thermal-Preferences)
  - If the visibility of a particular object type is unchecked, it will not be displayed
  - On/Off Toggles for common objects (Node, UserNode, Surface, Solid Element, Heat Load, Conductor, and Contactor) available on toolbar
- Object Level: (Blue and Yellow Lightbulb toolbar buttons)
  - Objects can be selected in Model Browser and visibility turned on or off (Blue and Yellow Lightbulb)
  - Objects can be selected from the screen and visibility turned off
  - Objects can be selected by group (even if currently off) and turned on
  - On/Off state can apply to solid CAD but hard to manage; use "all" for selection if needed
- *Best to use Layers for CAD objects. User choice for Thermal objects, as visibility can be controlled by Model Browser*
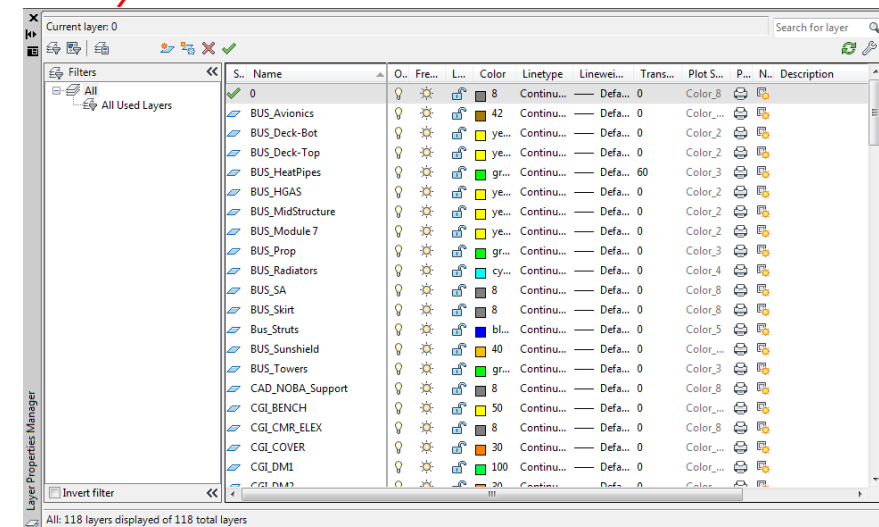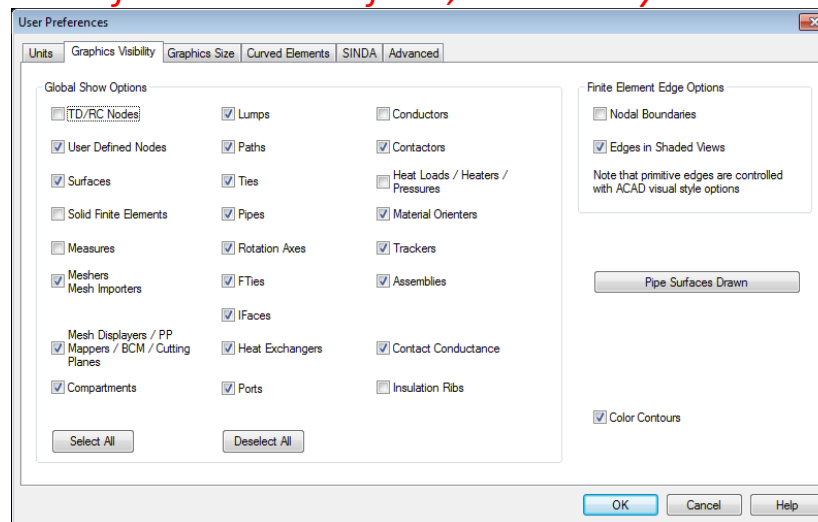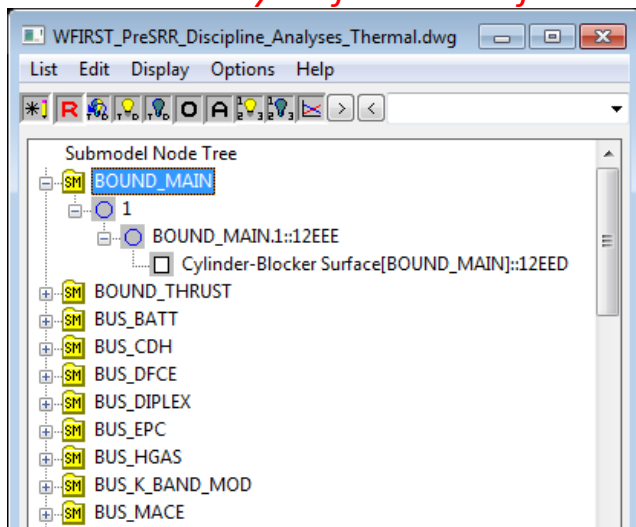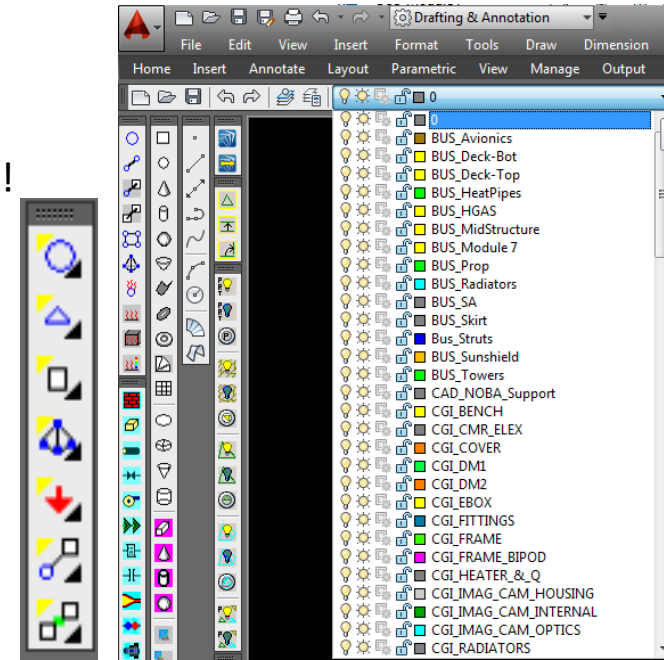
BUILD MODEL: Defining Variable Parameters

- Symbols can be used to control just about anything in ThermalDesktop
  - Dimensions (thickness, length, width, height)
  - State Flags (On/Off, Enabled/Disabled, Open/Close, Stow/Deploy)
  - Margin Factors (Heat Load Multiplier)
  - Other Values (Conductance, Heat Load, Setpoint, Rotations, etc)
- Symbols can
  - be single values (Integer or Real), expressions, or arrays of values
  - depend on other symbols
  - be overridden in the Case Set Manager for a particular analysis
- Symbols should
  - ✓ Be named intuitively. Use prefixes for thickness (thk_), heat (Q_), conductance (G_), Multipliers (Mult_), etc.
  - ✓ Be grouped together where logical (power, subsystem, etc)
  - ✓ Include a description or comment
- Symbols can be passed through to SINDA as REGISTERS
  - This is automatically done if the symbol is needed in SINDA
  - User can override to ensure symbol is passed as REGISTER if not detected by ThermalDesktop
- To use symbol, dbl click in textbox to access Expression Editor
- *Values can be set in Symbol Manager, Case Set Overrides, or user defined logic…be careful where you define things*
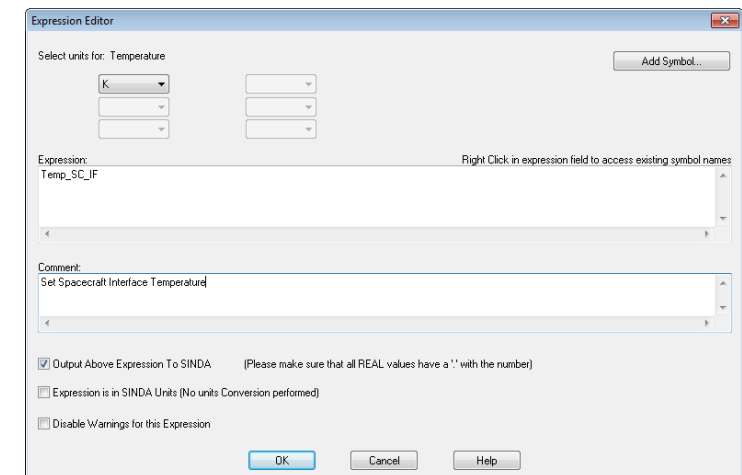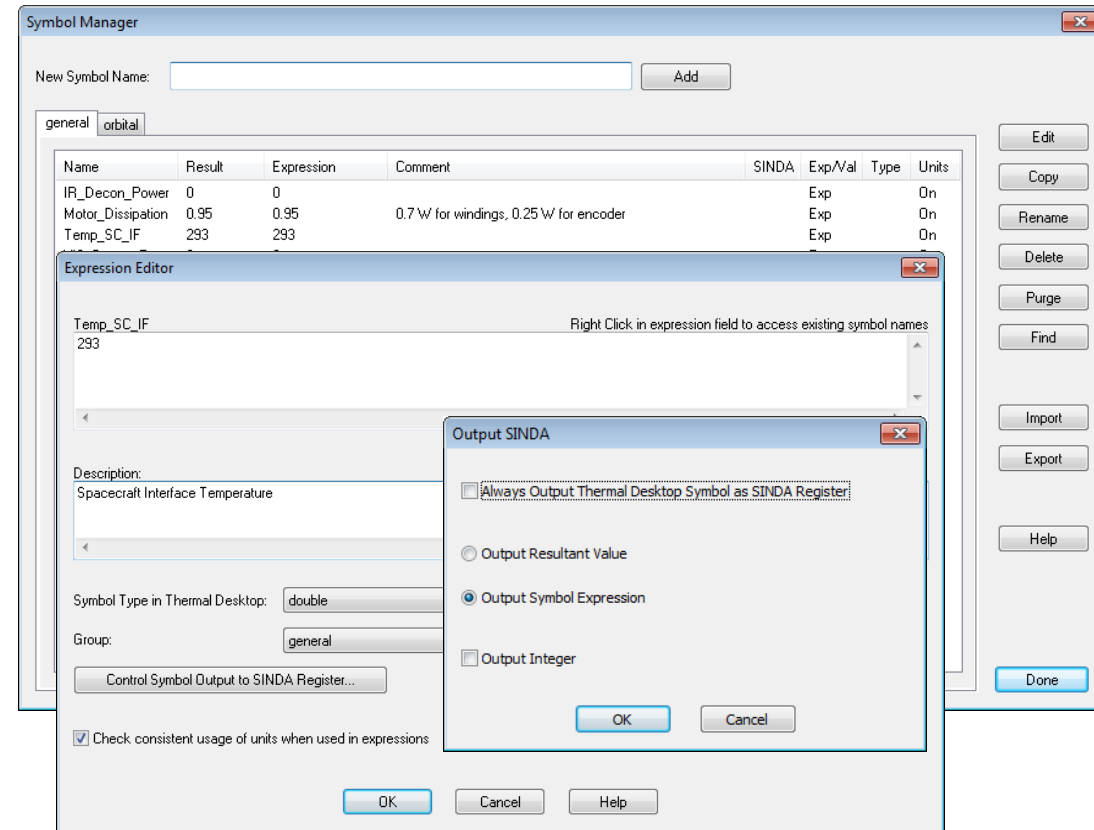
BUILD MODEL: Assigning Variable Parameters

- Once a symbol is defined in ThermalDesktop, double click in just about any textbox to access Expression Editor to use it
  - Values that are dependent on expressions are indicated in **Bold** font
  - Values with a description are highlighted in light blue
  - Illegal or incalculable expressions default to 1 with a red background (importing a model without importing symbols)
- Unit conversion can be performed
- Can output expression to SINDA (as REGISTERs)
- Conditional statements (***Not Valid in SINDA FORTRAN block…***)
  - (Heater_ON == 1) ? 10 : 0
  - If Heater_ON is equal to 1, set expression to 10 otherwise 0
- *interp* function can be used to access elements of symbol arrays
  - interp(CaseArray,PowerArray,CaseFlag)
  - Look in CaseArray for Case Flag and return entry at same index in PowerArray – ***Not Valid in SINDA FORTRAN block…***
- Example symbol usages include:
  - *Power Dissipation (Heat Load or Heater)*
  - *Conductance Values or Convection Coefficient*
  - *PID Gains*
  - *Assembly Rotation Angles or Translations (Open/Close)*
  - *Thickness*
  - *Geometric Distances*
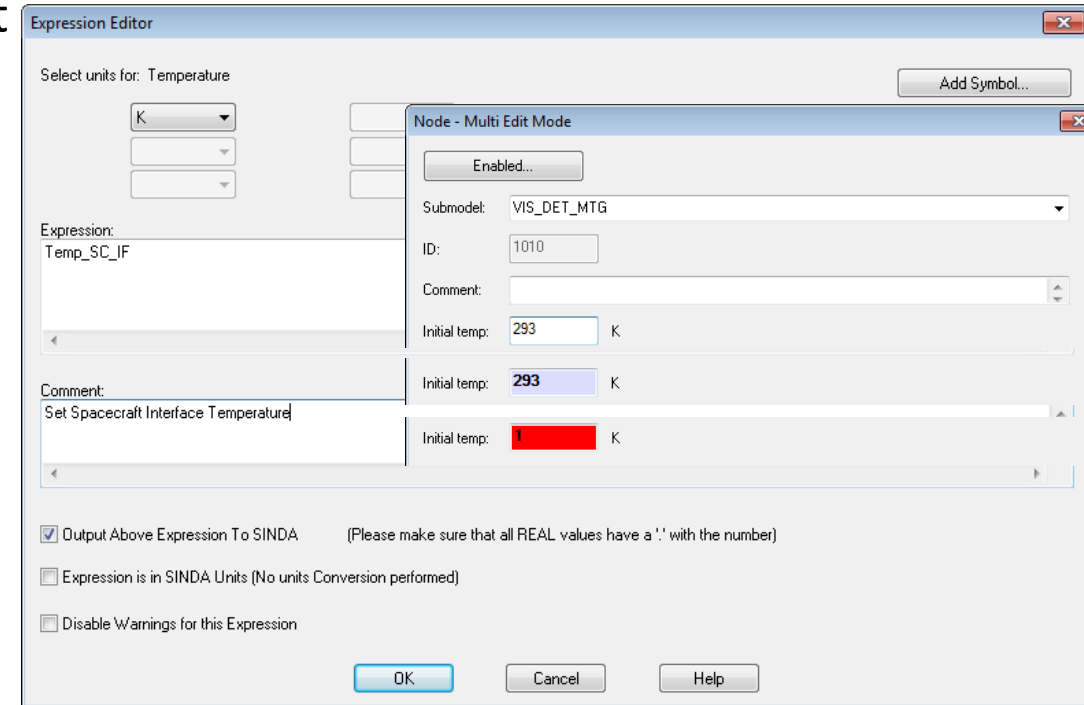
  - *Boundary Temperatures*
  - *Initial Temperatures*
  - *Setpoints*
  - *MLI Assignment*
  - *Number of Rays or Error for MCRT*
  - *Margin Factors/Multipliers*
  - *State Flag (On/Off)*
  - *Failure Flag (zero out value to disconnect or null)*
  - *Absorptivity/Emissivity/k/e*/etc*
 - *Etc…*

BUILD MODEL: Moving Geometry

- Two methods exist in ThermalDesktop for handling moving geometry: **Assemblies** and **Trackers**
  - For both, geometry must be attached
  - If using FE, nodes must be attached since nodes define elements
- Assemblies
  - Everything attached moves when assembly is moved
  - Be careful using AutoCAD move command; moving TD objects and the assembly makes everything attached to the assembly move twice
  - Useful for moving geometry based on user needs (e.g. door opening, shield deploying, lid closing, etc). Could be orbit position dependent.
- Trackers
  - Similar to Assemblies, but point specified axis towards target by rotating about second specified axis. Can be nested, but gets complex quickly
  - Useful for simple solar arrays or more complex spacecraft pointing
  - Utilized by HR and Articulating Radk calculations
- Fast Spin (many times in one orbit position)
  - Fast spinning capability exists for radiation calculations for scan mirrors, rotating drums, etc.
  - Use an assembly, assign spinning geometry, and assign rotation angle to symbol variable
  - In Radk or HR Spin Tab, specify start angle, end angle and increments
  - Be sure to use 0 error and max number of rays per *position*
- Slow Spin/Slew (<< one time in orbit position)
  - Orbit Rotation or slew to multiple of *hrMeanAnom*
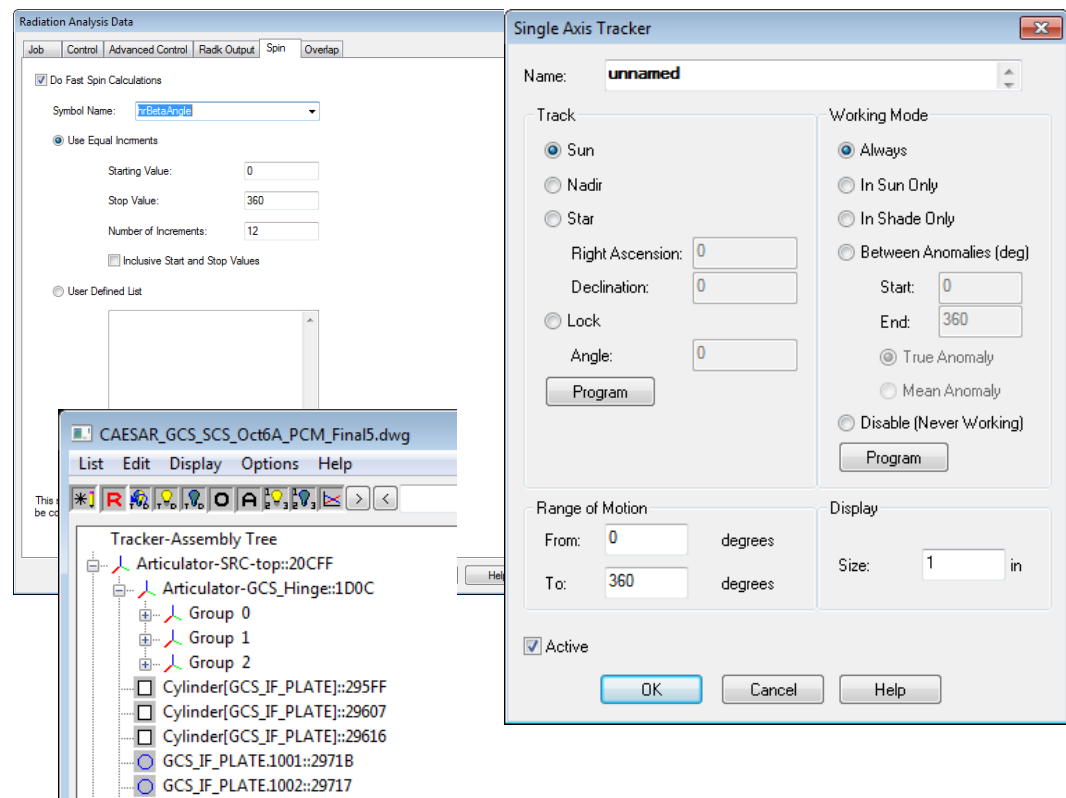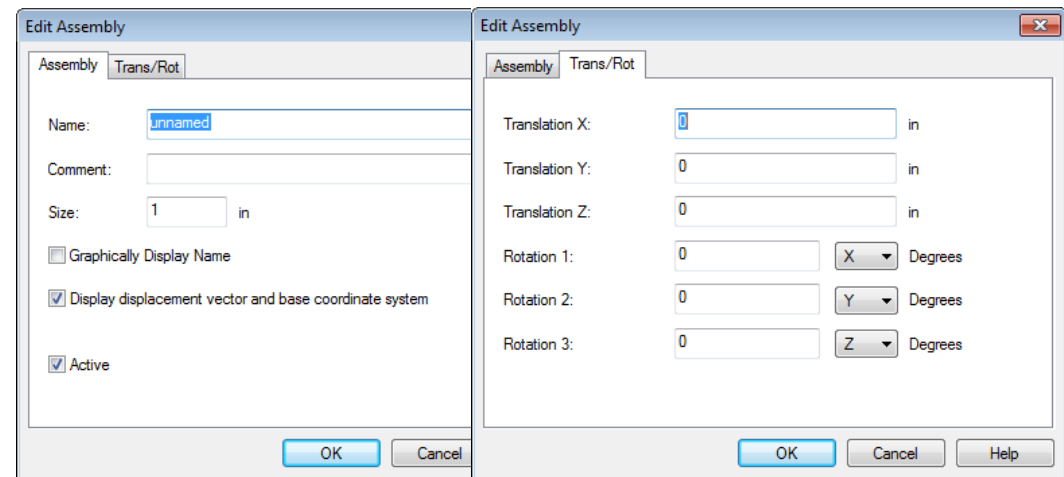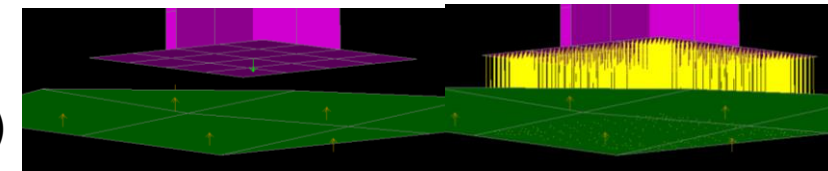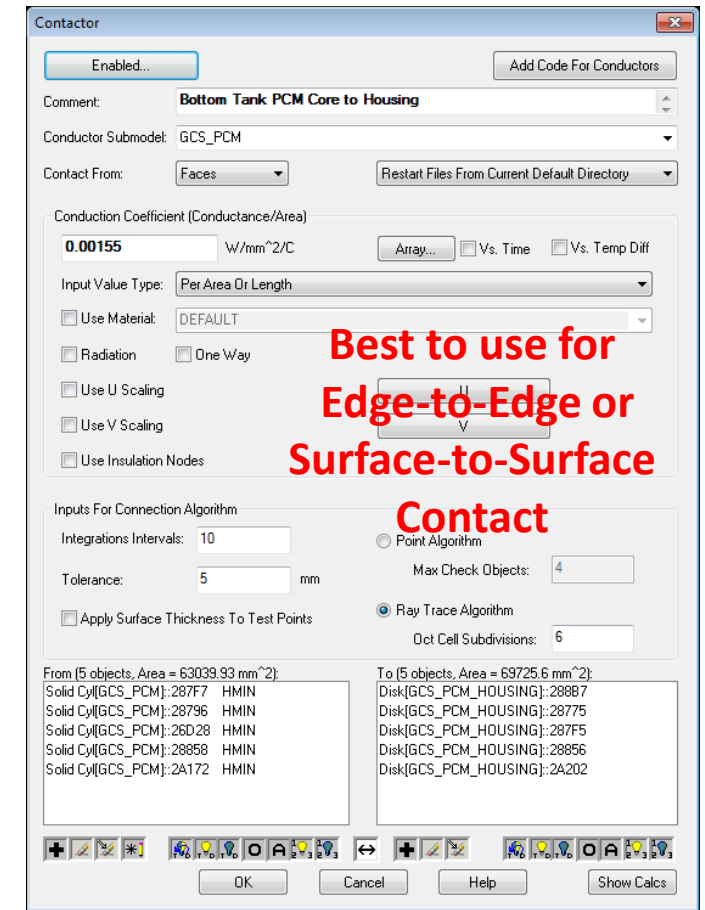
BUILD MODEL: Establishing Connectivity

- When nodes are not merged to establish connectivity between two surfaces, a user must add this manually. Two methods exist in ThermalDesktop for this: **Conductors** and **Contactors**

Conductor: Tie a specified node to a surface, multiple nodes, or a single node

Best to use for Point-to-Point or Point-to-Surface Contact
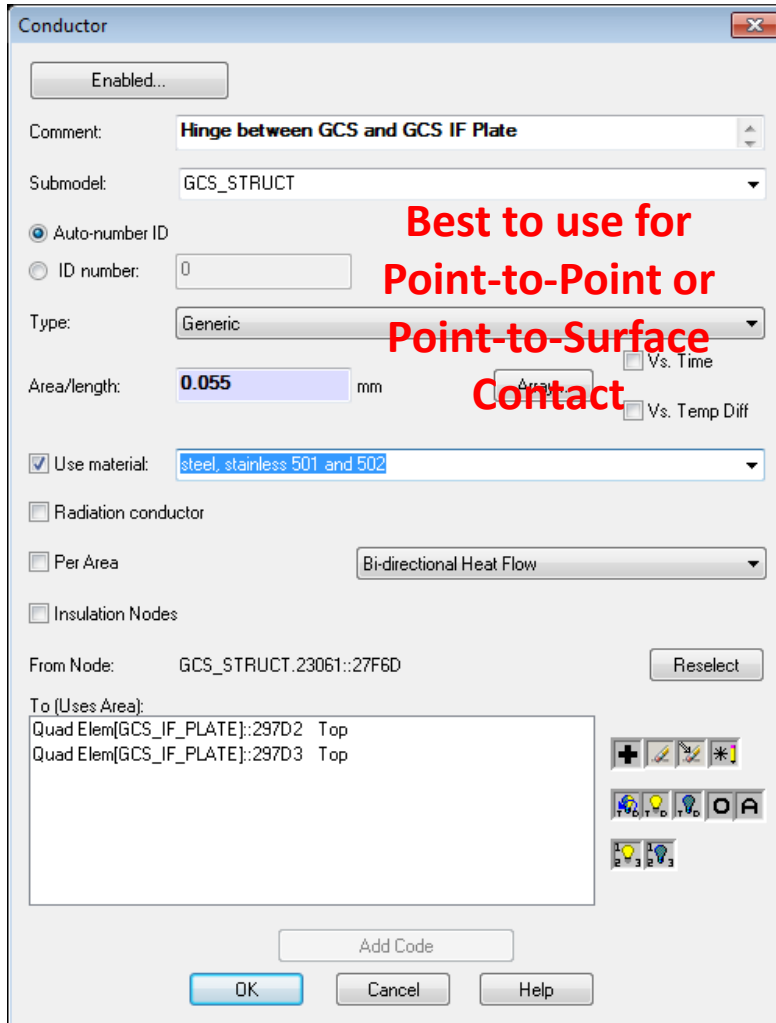
- Both
  - Can be material dependent, time or temperature dependent or constant value
  - Can be to insulation nodes, one way, or radiative conductors
- Conductors
  - Remeshing/renodalization can break this link
  - Can be per Area, Cannot mix types (i.e. cannot add a surface to a node to node conductor)
- Contactors
  - Renodalization does not break this link, unless new elements are created
  - ✓ Must specify Edge or Face contactor type and edges/face sides to make contact
  - ✓ Use Ray Trace for Face Contact
  - ✓ Specify reasonable tolerance
  - ✓ Preference is to use Per Area or Length over absolute conductance
  - ✓ Often, best not to Apply Surface Thickness
  - Contact From displayed as Green Arrow, To faces as Yellow Arrows
  - Can visually display contact made (or missed)

Contactor: Tie nodes associated with two surfaces or two edges together

Best to use for Edge-to-Edge or Surface-to-Surface Contact

BUILD MODEL: Establishing Boundary Conditions

- Boundary conditions must be applied to models. Typical boundary conditions for thermal analysis include: fixed temperatures and fixed heat loads
  - Boundary nodes defined by overriding the calculations by elements/surfaces. Need to also specify temperature
- Heat Loads
  - ✓ User specified application of heat to nodes, surfaces, or solids
  - ✓ Can be total load or flux
  - Can be temperature dependent or time dependent
  - ✓ Specify Value and assign appropriate submodel
- Heaters
  - ✓ User specifies application of heat location and sensing location
  - Can be applied to nodes, surfaces, or solids
  - ✓ Specify Available Power, On Temperature, and Off Temperature
  - ✓ Specify control approach (thermostatic or proportional)
  - ✓ Specify Steady State behavior (Proportional recommended)
  - ✓ Specify Append string for symbol output recognition
    - TD creates power, total power, on time and #cycle registers using this string
  - Can specify sensing method (defaults to area weighted average temperature, but could be min temp or user defined)
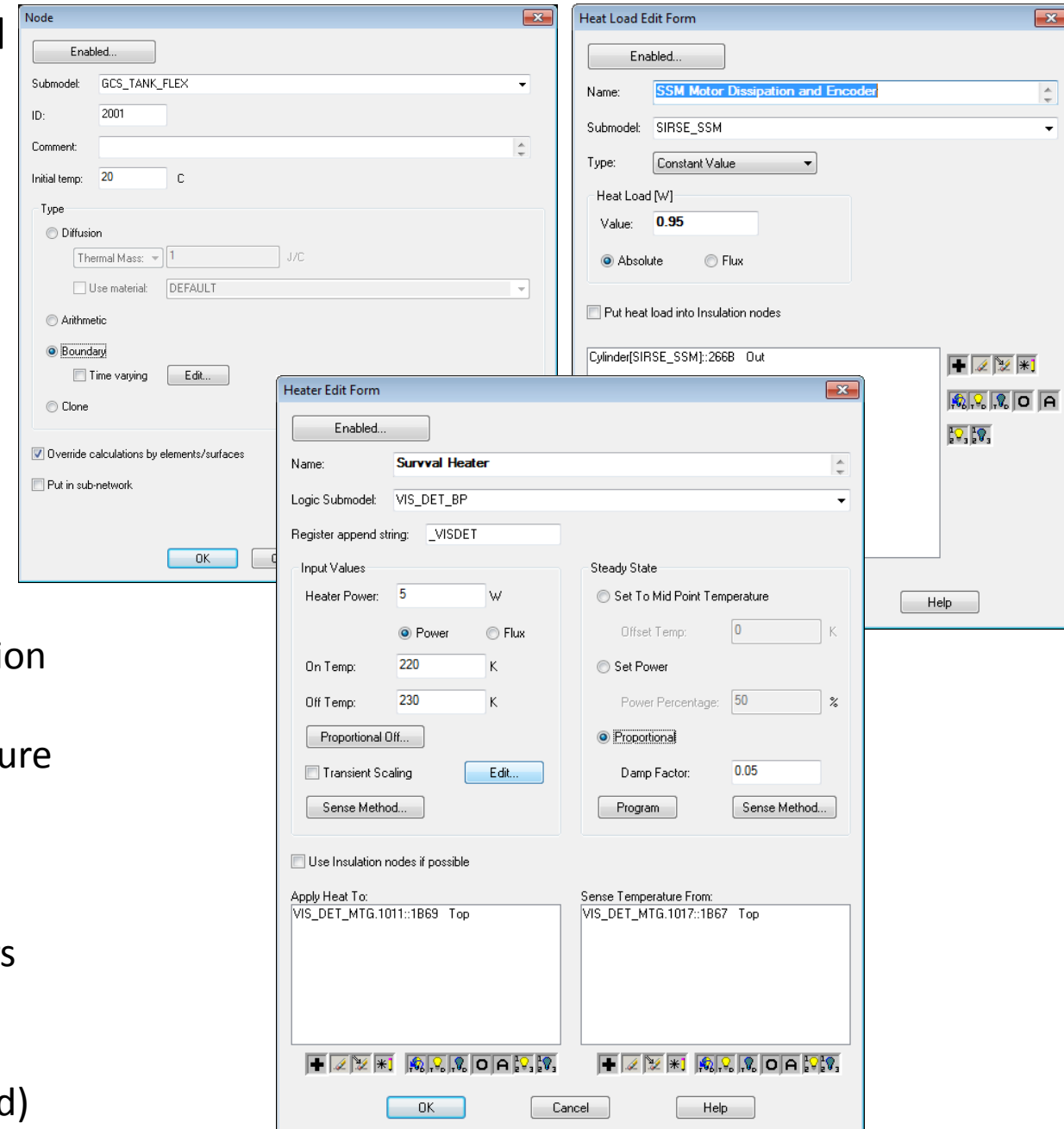
- The entities referenced by Heat Loads, Heaters, Conductors, and Contactors can be modified from those references when originally created
  - Add (+), Remove, Remove Selected, Edit
  - Switch Visibility, Turn on Visibility, Turn Off Visibility, Show Only Selected (O) or All (A), Turn on, Turn off Node Numbers
- Domain Tag Sets are groups of objects that can be used in numerous ways
- Create the Domain Tag Set
  - ✓ Specify the Name
  - ✓ Specify the type of object it will contain (cannot change this later!!)
  - ✓ Add/remove objects using buttons described above
- Domain Tag Sets show up in Model Browser and are useful for controlling component On/Off visibility
- Used as Groups for mapping temperatures to FEM
- Domain Tag Sets may be used as target objects for Conductors, Heat Loads, Contactors **_even if it does not contain any objects yet_**
- Useful for defining interfaces that may change on either side (dummy plate vs. SC deck)
- When prompted to select objects, type d to bring up list box. Only Domain Tag Sets of a valid type will be shown

BUILD MODEL: Simulating Thermal Hardware

- Every analyst will develop their own modeling style (i.e. how physical components will be represented in the analysis model)
- The following slides include suggestions on methods to model:

    - Honeycomb Panels

    - Insulation

    - Thermistors and Thermocouples

    - Constant Conductance Heat Pipe

    - Variable Conductance Heat Pipe

    - ThermoElectric or CryoCooler

    - Controller (most often for a heater)

    - Phase Change Material

    - Louvers

- Three ways exist to model Honeycomb Panels
  1. Neglect Core: use a single sided surface with 2x the facesheet thickness and the material of the facesheet
  2. Include Core through Conductivity: use a double sided surface with thickness and material specified for facesheet core, facesheet. Note: the in-plane spreading effect of the core material is not included using this method...
  3. Include full core effect: model the core as a solid element and the facesheets as plate elements using the same nodes as the top/bottom for each facesheet. If applying non-isotropic properties, must define material orienter for solids
- TIP: you can use the U Cond/V Cond multipliers increase the conductivity as a function of the core material to account for inplane spreading by the core

HC Core Material Orienter
(defines CS for anisotropic materials)

Method 1

Method 2

Method 2

Method 3

Method 3
Core as Solid Elements

Method 3
FS as Plate Elements
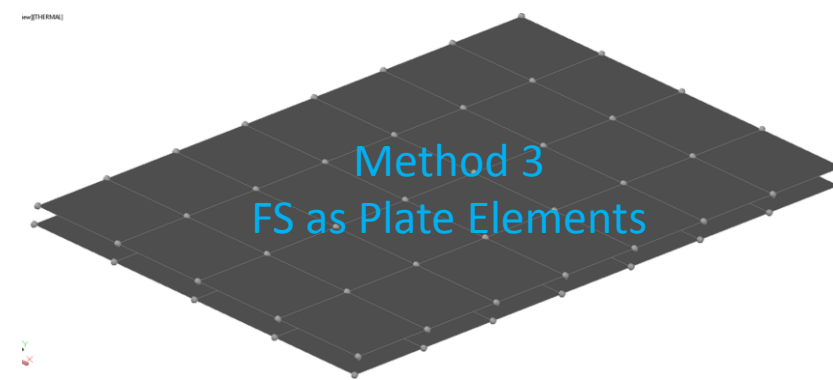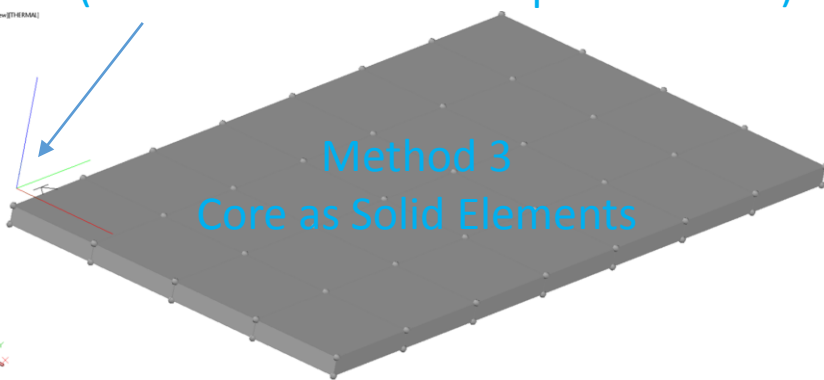
BUILD MODEL: Insulation Modeling

- Insulation may be conductive or MLI (i.e. radiative)
  - ✓ Define insulation material
  - ✓ If material has conductivity <> 0, k and thickness will be used
  - ✓ If material has ε* defined, radiative will be used (thickness ignored)
  - ✓ Assigning kg/m3 of 0.6 and applying a thickness of 1 m will allow MLI mass to be estimated by model
  - – Possible to have both…
- Use intuitive node numbering
  - ✓ Keep all non MLI nodes below 100000
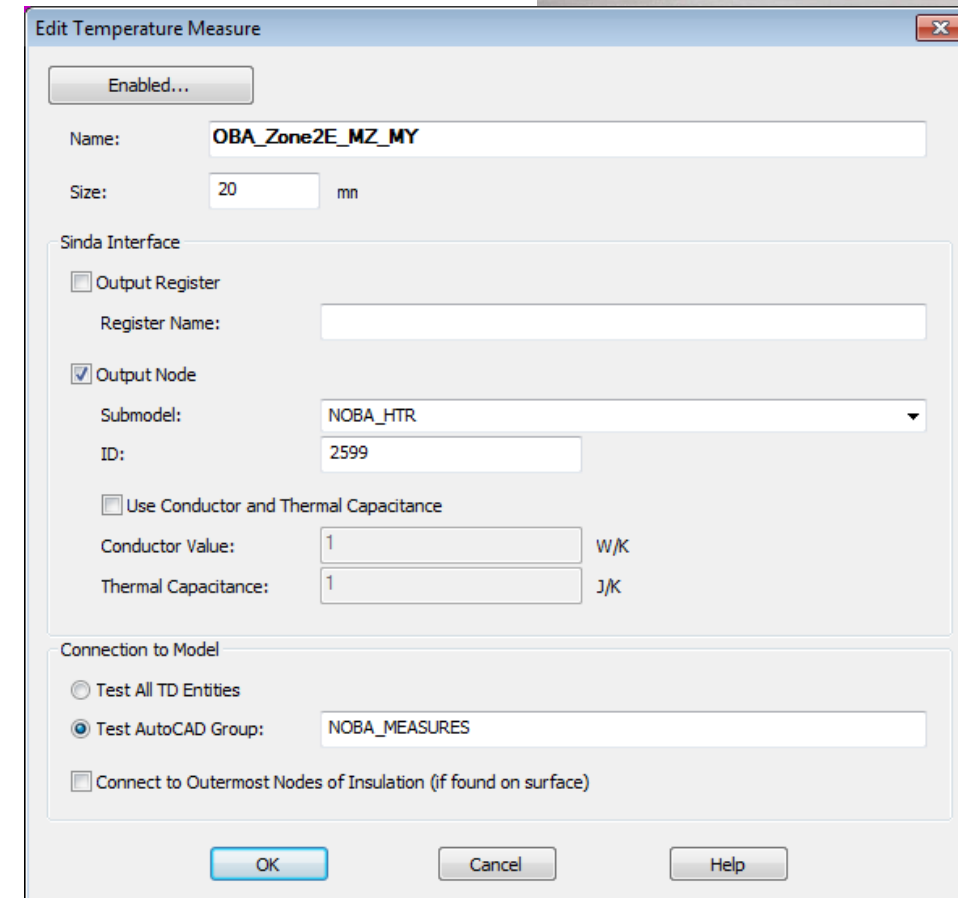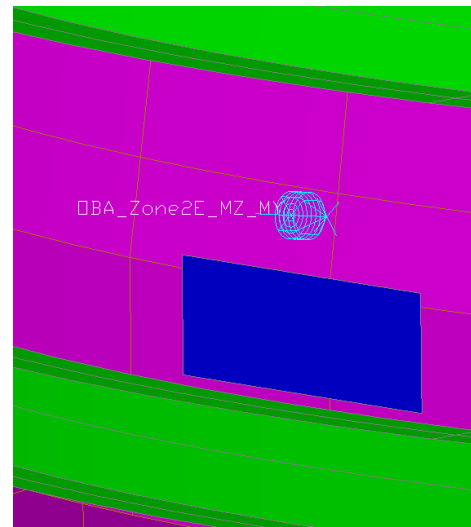 - ✓ Use Node Offset of 100000 (200000 if both sides have insulation)
  - – When post processing, any nodes above 100000 are MLI temps and can likely be disregarded
- If overrides are used (i.e. applying MLI to only some of the nodes), make sure to adjust the optical property overrides as well
  - ✓ Opt Prop overrides will need to reference MLI node number, not underlying surface
- MLI can be "programmed" to either be enabled or disabled using the P button at the top
- For "tented" MLI (offset by large gap from surface), best not to use low ε optical property for inner or outer layer – doubles isolation to have low ε and low ε* in series
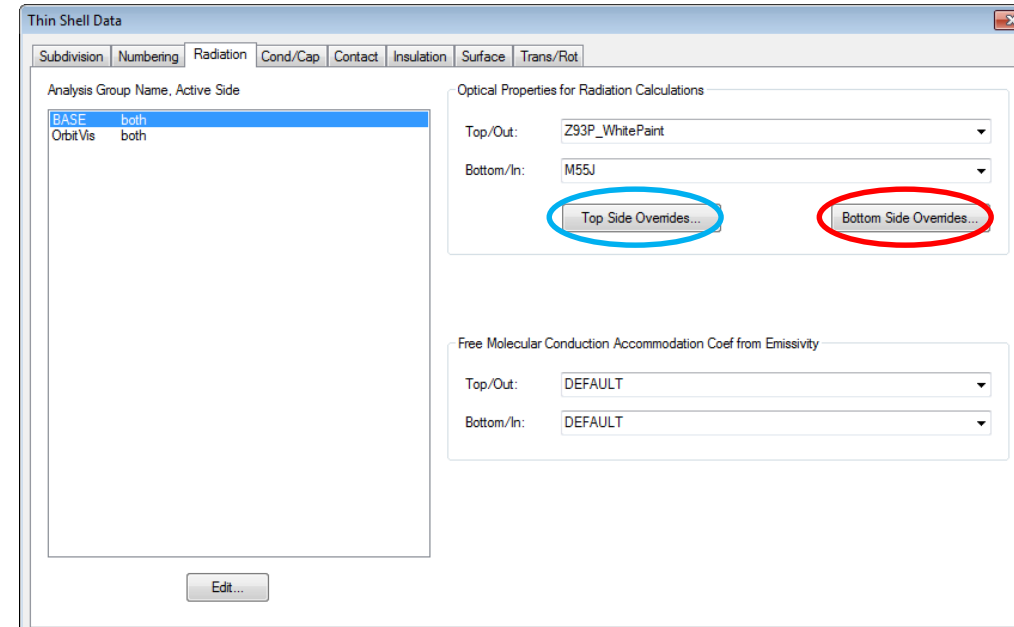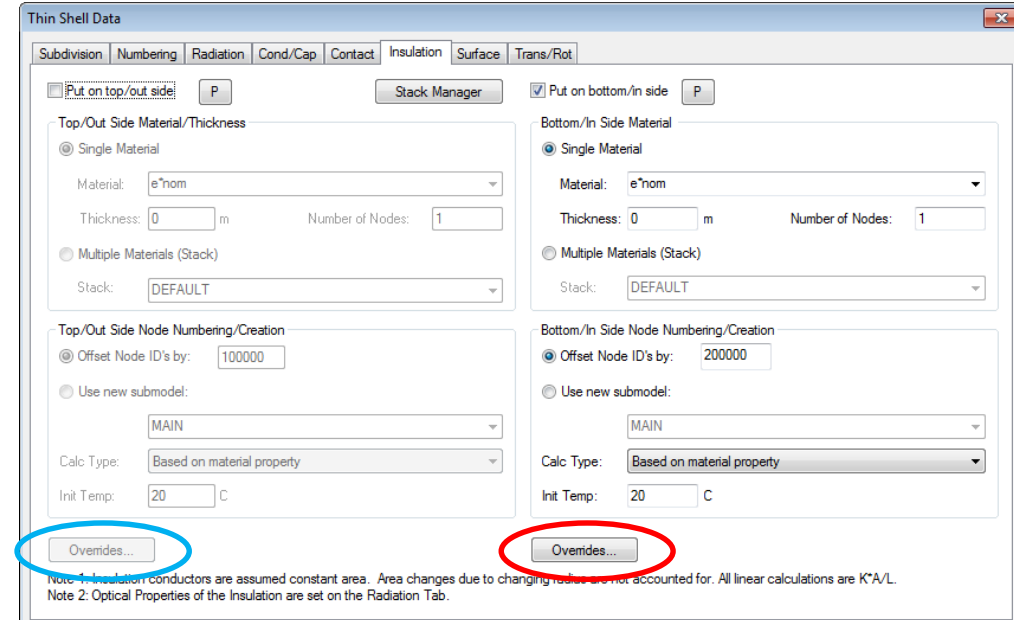
BUILD MODEL: Thermocouple and Thermistor Modeling

- Thermocouples, Thermistors, and other measurement devices are the tangible data readings visible to the thermal engineer based on the actual hardware
- ThermalDesktop has an object type called a Measure, which represents a spatial location for a temperature reading.
- Measures are displayed graphically with their Name and show the mapped location. They can be moved to the exact location desired by the user

✓ Define the Measure and place it correctly in space
✓ Set the size large enough to be visible when displayed
✓ Assign a meaningful name (e.g. Telemetry mnemonic)
✓ Determine if you wish to output measure value to an existing
  Register or a Boundary Node
✓ Good idea to Create groups to ensure Measure maps to
  reasonable entity
✓ Set tolerance
✓ Execute mapping
✓ Snap Measures to
  Mapped Entity
– Can copy measures from
  One drawing to another
– Can import from text file

Edit Temperature Measure

Enabled...

Name: OBA_Zone2E_MZ_MY

Size: 20   mn

Sinda Interface
☐ Output Register
  Register Name:

☑ Output Node
  Submodel: NOBA_HTR
  ID: 2599

☐ Use Conductor and Thermal Capacitance
  Conductor Value: 1   W/K
  Thermal Capacitance: 1   J/K

Connection to Model
○ Test All TD Entities
◉ Test AutoCAD Group: NOBA_MEASURES
☐ Connect to Outermost Nodes of Insulation (if found on surface)

OK   Cancel   Help

Variable Tolerance

Input one value per line, in units of mm

0
1e-006
2e-006
4e-006
1e-005
2e-005
4e-005

☐ Ignore Thickness when Gen Cond/Cap is not enabled
☐ Use Advanced Mapping

OK   Cancel

OBA_Zone2E_MZ_MY

BUILD MODEL: Constant Conductance Heatpipe Modeling

- Heatpipes are commonly used devices in thermal designs
  - They operate on an evaporation/condensation cycle that allows heat to be transported over large distances with a minimal temperature gradient
  - Heat enters the evaporator end of the pipe and vaporizes the working fluid
  - The vapor is driven to the condenser end of the pipe by a small pressure gradient
  - The vapor changes to liquid in the condenser end releasing the heat
  - The liquid is pumped to the evaporator by capillary forces through a wick structure
  - Unfortunately, the capillary forces that drive the fluid are very small compared to gravity which places constraints on HPs during ground testing
  - Device can operate in reverse depending on loads...
- HEATPIPE exists in FloCAD, but a simpler way exists to model a heatpipe that does not require a FloCAD license
  - ✓ Define surfaces based on length/shape of pipe
  - ✓ Set as edge nodes 1 x Z breakdown, assign nodes to increment from one end to the other (e.g. x001, x002, x003...) and merge at interfaces
  - ✓ Set width of surface to width of interface flange
  - ✓ Set Material to Aluminum (Assuming aluminum walls)
  - ✓ Set Thickness to effective thickness (HP Cross Sectional Area / Width)
  - ✓ Create Arithmetic node to represent vapor node and set to (x999)
  - ✓ Create a Node-to-Surface conductor from Vapor node to all HP surfaces
  - ✓ Check the Per Area option and set conductance value to h per linear inch / Flange Width (3.0 W/lin in. K typical for ammonia HP)
  - ✓ Add contactor from evaporator/condenser sections to structure/radiator
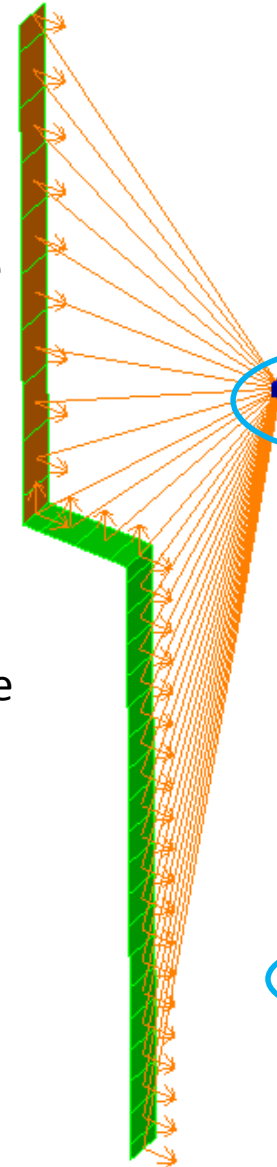
BUILD MODEL: *Variable Conductance Heatpipe Modeling*

- A Variable Conductance Heat Pipe (VCHP) is an extension of a CCHP
  - It includes a reservoir at the condenser end filled with Non-Condensable Gas (NCG)
  - This reservoirs is cold biased and includes a heater to maintain its temperature
  - When the reservoir heats, the NCG expands and partially fills the condenser section, blocking condensation and reducing the effective condenser length
 - When the reservoir cools, the NCG contracts and recedes toward the cold reservoir, increasing the effective condenser length
  - By heating and cooling of the reservoir, the evaporator end of the VCHP can be controlled to a given temperature under a variety of source and sink values
- HEATPIPE routine in SINDA can accommodate modeling a VCHP, but requires additional inputs: [Vapor Node, Array of Vap-to-Wall Cond Numbers, Array of Segment Lengths, Volume, HP Diameter, h_{cond}, h_{evap}, Res Volume, Res Temp, NCG charge, Working Fluid and NCG IDs]
  - ✓ Define a heatpipe as previously specified and ensure that the nodes increase incrementally from the Evap to the Condensor. Ensure the vapor node is arithmetic!
  - ✓ When defining the Vapor to Wall Conductors, be sure to specify the ID and not Auto-number
  - ✓ This ID will be a fixed value set by the user to be specified in the Vapor-to-Wall Cond Array
  - ✓ Be sure to add a geometric representation of the reservoir and its radiator and heater: heat should be applied at the Res Radiator and sensing at the evaporator
  - ✓ Define Arrays with Segment Lengths and Cond Nos in the Logic Manager
  - ✓ Define a VARIABLES 1 block in logic manager with a call to HEATPIPE and add in the appropriate variable and argument values
  - ✓ Refer to SINDA/FLUINT manual for specifics...*be careful with initial conditions*
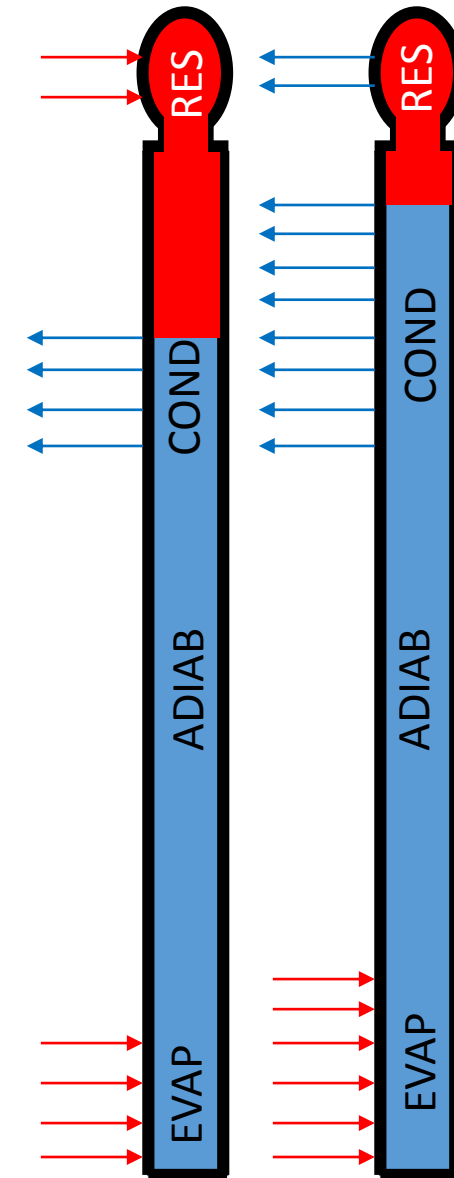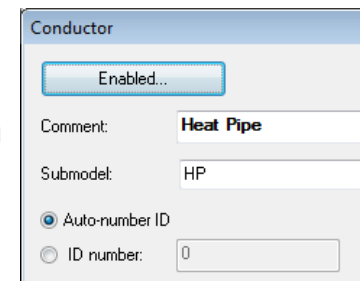
BUILD MODEL: TEC or Cryocooler Modeling

- ThermoElectric Coolers utilize the Peltier principal to induce a temperature gradient between two junctions when a current is applied to provide cooling. Typically about 6% efficient
- Performance based on 4 related variables specific to device. Knowing 3 can allow 4th to be calculated
  - Hot Side Temperature (can be retrieved from model)
  - Cold Side Temperature (usually the goal or setpoint)
  - Cold Side Load (can be calculated from model)
  - Hot Side Load (including input power, current, or voltage), often the independent control variable
- User must decide how to model controller…
  - Assume it can achieve control temperature and set cooling point to boundary temperature
  - OR Model controller with feedback and apply negative cooling load (see PID controller slide)
- If assuming controller can achieve temperature…
  - ✓ Need to extract heat removed from boundary cold side (QFLOW, QFLOWSET or HNQCAL)
  - ✓ Retrieve Hot Side Temperature from model
  - ✓ Determine cooling DT from T_{hot} - T_{cold}
  - ✓ Look up power needed for TEC to remove Q heat, and achieve DT for current T_{hot}
  - ✓ Apply power to Hot Side node
  - ✓ Routine developed for WFC3 to characterize TEC performance curves by 4th order polynomial
  - ✓ Could also use TRIVARIATE array with enough data points…
- If constant power TEC…
  - ✓ Determine heat removed based on input power, DT, and T_{hot}
  - ✓ Apply a negative heat load at cold side node
- Hot Side and Cold side usually not coupled unless TEC is off…
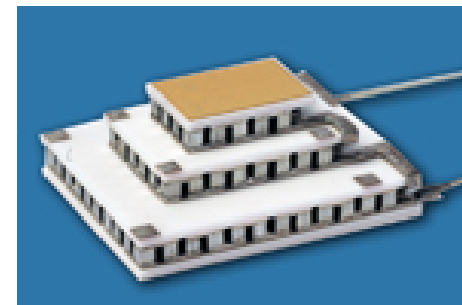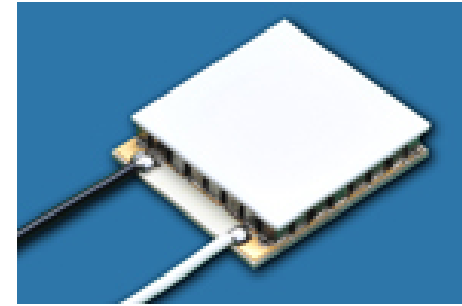
BUILD MODEL: Controller (Heater or Cooler) Modeling

- More thermal models are asking for controller modeling: input value is mathematically evaluated to predicts output value of controller (e.g. Input: Temperature, Output: Heater Power)
- Common example is thermostatic or proportional heater:
  - ✓ Define On and Off Temperatures and Max power to apply
  - ✓ Define Location to Apply Heat and Location to Sense Temperature
  - ✓ Use Proportional for Steady State behavior
  - ✓ Determine if Proportional is needed for transient
    - ✓ Often needed for tighter stability
    - ✓ **_Widen_** Range between Toff and Ton for better control
- For a full PID controller, use a PID controller under Logic Manager
  - ✓ Define Proportional, Integral, and Derivative Gains
  - ✓ Define Setpoint: what you are trying to achieve
  - ✓ Define Sensing Variable: what it actually is
  - ✓ Define Control Variable what you are actually changing (e.g. power)
    - ✓ Done as a Register to be used elsewhere in TD
  - ✓ Define upper/lower bounds on Control Variable (e.g. max and min power)
    - ✓ Generally a good idea to _Prevent Integral Windup_
  - ✓ Apply output variable to some TD object, usually a heat load
    - ✓ Make sure to output as expression so that changes in process variable as SINDA runs are applied correctly
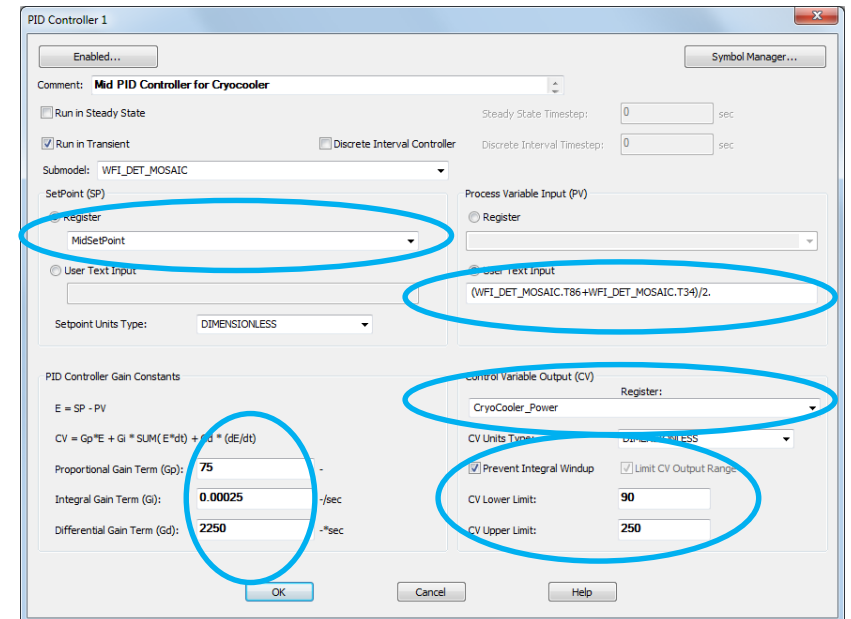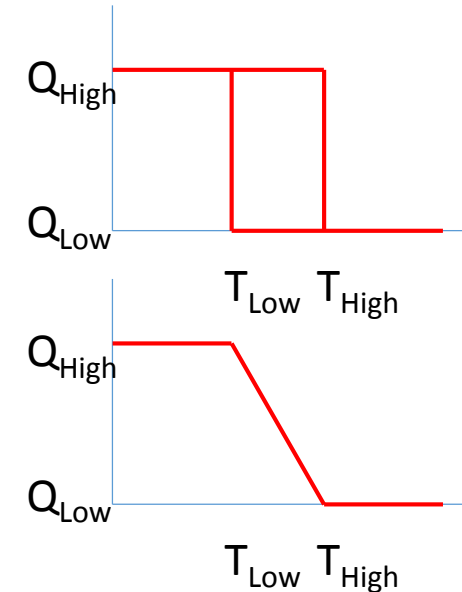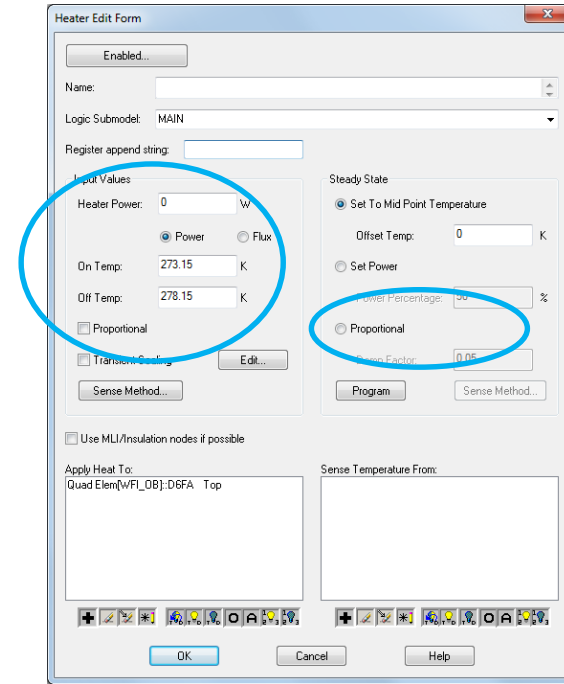
- Phase change happens over a constant temperature
  - PCMs (paraffins) are generally very poor conductors of heat
  - PCM assembly is generally a hermetically sealed aluminum housing to which a core for uniform spreading is bonded
  - Gaps in between core is filled with PCM
  - Core may contact multiple faces for improved through conductivity (aka vias)
  - Melting point is material specific. Paraffins typically used with Solid to Liquid phase change
- ThermalDesktop includes capabilities to model PCMs via FUSION function
  - ✓ Define conductivity of material as the core (typically 10% or AL1100 for plane of fins, and 1% or AL1100 for out of plane)
  - ✓ Define Specific Heat of material as FUSION
  - ✓ Specify Melt Pointing and Heat of Fusion
  - ✓ Majority of mass of Core+PCM is in the paraffin for density

Alkane	Formula	MP (°C)	h_{FUSION} (kJ/kg)
Decane	$C_{10}H_{22}$	-29.6	202.3
n-Undecane	$C_{11}H_{24}$	-25.6	142.9
n-Dodecane	$C_{12}H_{26}$	-9.6	214.6
n-Tridecane	$C_{13}H_{28}$	-5.4	155.6
n-Tetradecane	$C_{14}H_{30}$	5.9	224.2
n-Pentadecane	$C_{15}H_{32}$	10.0	161.9
n-Hexadecane	$C_{16}H_{34}$	18.2	226.9
n-Heptadecane	$C_{17}H_{36}$	22.0	165.4
n-Octadecane	$C_{18}H_{38}$	28.2	240.0

Data from http://webbook.nist.gov/chemistry/
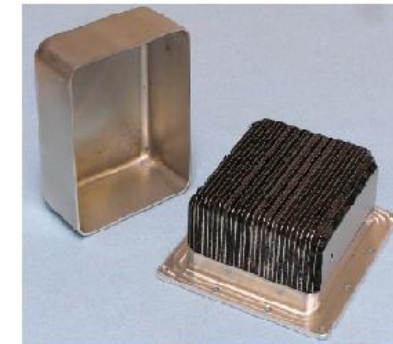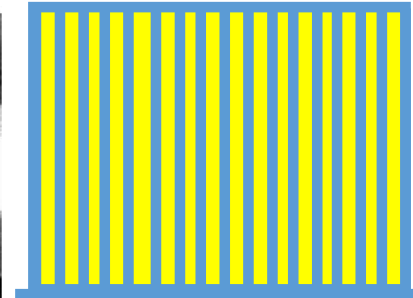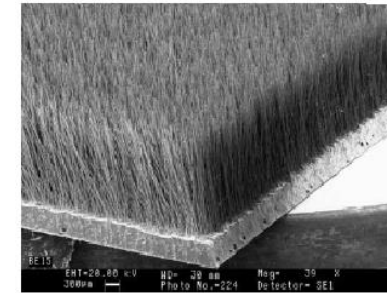
Core material helps to spread heat uniformly throughout assembly. Void areas of core are filled with PCM. PCM generally stays in contact with core
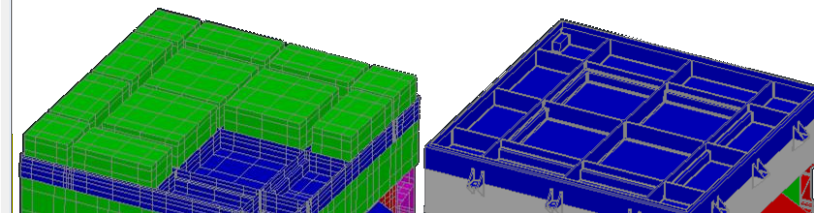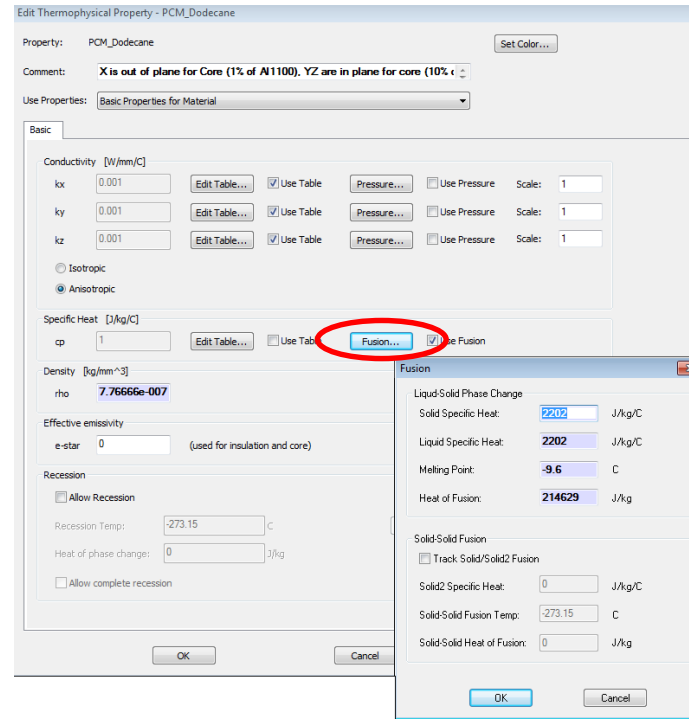
Model Housing as 2D Rectangles, PCM+Core as Solid with PCM Material assigned. Contactor between Core/Housing for bond

BUILD MODEL: Louver Modeling

- Louvers are devices which passively activate as a function of temperature. They are often used to reduce the view of a radiator to space when they get colder and increase the view as they get warmer to conserve heater power.
- They function when a bi-metallic spring changes its length and spring constant as a function of temperature, rotating a set of highly reflective, parallel blades
- These blades change the view factor from an underlying radiator its environment
- Since Radiation couplings are generally for a fixed geometry and properties, the typical Monte Carlo Ray Trace does not account for this variability nor does it know the temps…
- To model this, it is necessary to modify primarily the Radk to Space and environmental loads in the thermal model. Radiation Model should be run with fully open properties…
- Account for louver in SINDA model (to first order at least)…
  - ✓ In OPERATIONS, loop from 1 to NGTOT for each Louver Node. For each *Index…*
    - ✓ CALL CONDAT(GETGMOD(*Index*),GETGNUM(*Index*),JTEST,KTEST,ZTEST)
    - ✓ Check if NDNAM(ITEST)=SPACE, NDINT(ITEST)=SpaceNode, NDNAM(JTEST)=Louver Sub, NDINT(JTEST)=Louver Node
    - ✓ Check if NDNAM(JTEST)=SPACE, NDINT(JTEST)=SpaceNode, NDNAM(ITEST)=Louver Sub, NDINT(ITEST)=Louver Node
    - ✓ If either condition is met, then flag *Index* as a Louver G and store Cond number for later reference
    - ✓ Store Fully Open conductor value for this G for future modification
  - ✓ In VARIABLES1, for each Louver node (G Index and Fully open conductance should have been determined and stored)
    - ✓ Determine the scale factor for partial closure based on temperature and Louver Performance curves: SCL (1 to >0)
    - ✓ Set G(Louver Index) = G Fully Open * SCL
    - ✓ Adjust Heat Load by SCL (If Nadir pointing, Tot Env - Eclipse Env = UV Env, Eclipse Env = IR Env, adjust based on α,ε)
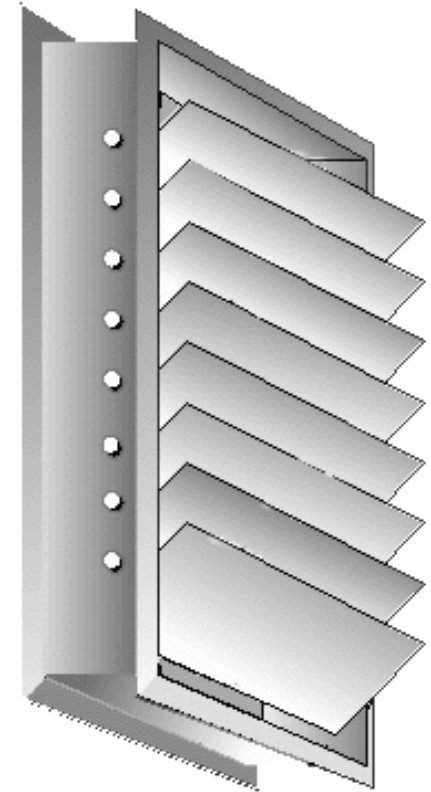
BUILD MODEL: Integrating Delivered Models

- Models delivered from other organization should have already been checked out. Delivered models (PARTs) and assembly models (ASSY) often need to be prepared for integration
  - ✓ *Remove boundary conditions from PART model that represent interfaces: boundary nodes, blocker surfaces. Keep conductors/contactors to boundary nodes to be able to connect to non-boundary interface. All that should remain is what needs to be imported into next higher assembly. Ensure that Radiation Group names are predefined or do not conflict with those that exist in the ASSY file. This should be a clean PART file for import*
  - ✓ *Remove previous instance of PART from the ASSY model if necessary or consider if better to keep old PART in ASSY and rename submodels with OLD_ prefix to have previous integration template. Save this as a clean ASSY file for integration*
  - ✓ *Ensure both PART and ASSY are using the same units*
  - ✓ *Use AutoCAD's INSERT command to insert PART file. Explode and place at correct location and orientation. Could use Copy/Paste if no Domain Tag Sets need to be preserved*
  - ✓ *If you need to move the entire inserted PART, be careful with any nodes/surfaces that may be attached to assembly. Moving everything will move the attached entities twice (relative to assy and then the assy itself)*
- ✓ Ensure that all PART submodels added have the same number of Nodes, Surfaces, Contactors, Heat Loads, etc. in the PART file and the new ASSY file
- ✓ Import Optical and Material Properties (Model Browser will show *Property Not Found* for undefined properties)
- ✓ Import Symbols (Model Browser will show Symbol values as *[]* when referenced but not defined)
- ✓ Import Logic Object Manager objects and PART file node correspondence if needed
- ✓ Verify that Radiation Groups are correct. Can merge imported PART groups with existing ASSY groups
- ✓ Ensure Domain Tag Sets from PART are included in ASSY
- ✓ Determine if any symbol or property alias over rides from PART case set are necessary in ASSY case sets
- ✓ Reconnect contactors/conductors to ASSY side entities and remove any temporary PART boundary conditions. Merge any nodes that may be coincident between PART and ASSY prepared files
- ✓ Add Tab to Notes (Utilities-Notes-right Click Tabs) and paste text documentation (text only, no images, tables, etc)

CHECK MODEL: General Model Checks

The following list can be used for your own model or models you received from others:

- ✓ **Ensure Material and Optical properties have correct values defined**

- ✓ **Ensure all surfaces have material and optical properties assigned**

- ✓ **Verify thicknesses** *(Model Checks-Color by Property Value-Thickness)*

- ✓ **Check Mass of Submodels** *(Model Checks-Calculate Mass)*

- ✓ **Check Free Edges of Elements** *(Model Checks-Show Free Edges)*

- ✓ **Check for Duplicate Nodes** *(Model Checks-List Duplicate Nodes)*

- ✓ **Check for Coincident Nodes** *(FD/FEM Network-Merge Coincident Nodes)*

- ✓ **Active Sides for Radiation Group** *(Model Checks-Display Active Sides)*

- ✓ **MLI Assignments** *(Model Checks-Active Display Preferences)*

- ✓ **Verify All Contactors make contact** *(Model Checks-Show Contactor Markers)*

- ✓ **Verify User Defined nodes are connected** *(Model Browser)*

- ✓ **Verify heater setpoints and Steady State behavior**

- ✓ **Verify proper behavior of assemblies/trackers** *(Orbit-Display Current Orbit)*

- ✓ **Verify proper associations of Symbols** *(Model Browser)*

- ✓ **Verify reasonable values for conductors, heaters, contactors, etc (units check and avoid large couplings)**

Submodel ->	Submodel A	Submodel ...	Submodel Z
Build Geometry	X	X	X
Shell Coat	X		
Nodal Subdivision	X		
Merge Nodes	X		
Submodel	X		
Node Numbering	X		
Thickness	X		
Material	X		
Material Orienters			
Active Sides			
Opt Props			
Insulation			
Comment			
Assembly/Tracker			
IF Contactors			
IF Conductors			
Heat loads			
Heaters			

If building a lot of new geometry, it can be helpful to have a per submodel checklist to keep track of whether you have fully assigned everything needed

- The Model Browser is a powerful capability to view the relationships between all the ThermalDesktop objects and manipulate what is displayed in AutoCAD
  - Many options for listing and controlling display
  - Shows relationship hierarchy (e.g. node-surface, Conductor-node)
- Submodel: for model/node organization
- Non Graphical: Orbits, Properties, Case Sets
- Analysis Group for Active Side Visualization
- Optical and Thermo Props for property assignments
- Surfaces/Solids for Conduction Submodels
- Contact/Contactor for displaying associated surfaces
- Assemblies/Trackers for showing assembly hierarchy
- Conductors for user defined conductors
- Heaters/Heatloads for used defined dissipations
- Orienters for anisotropic conduction
- Fluid objects
- Meshers: TD Mesh definitions
- Mesh Displayers: FEM Mesh Mappers (STOP)
- Symbols for showing object dependencies
- Groups/Domain Tag Sets for user defined collections
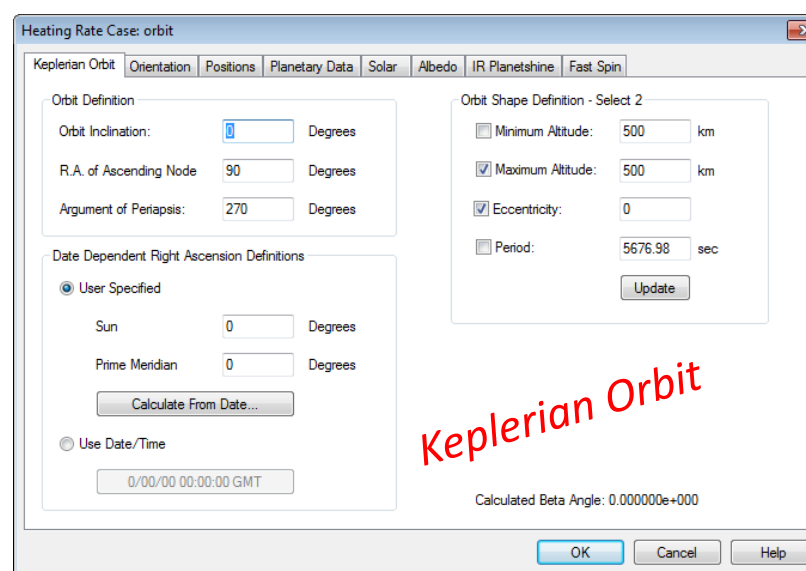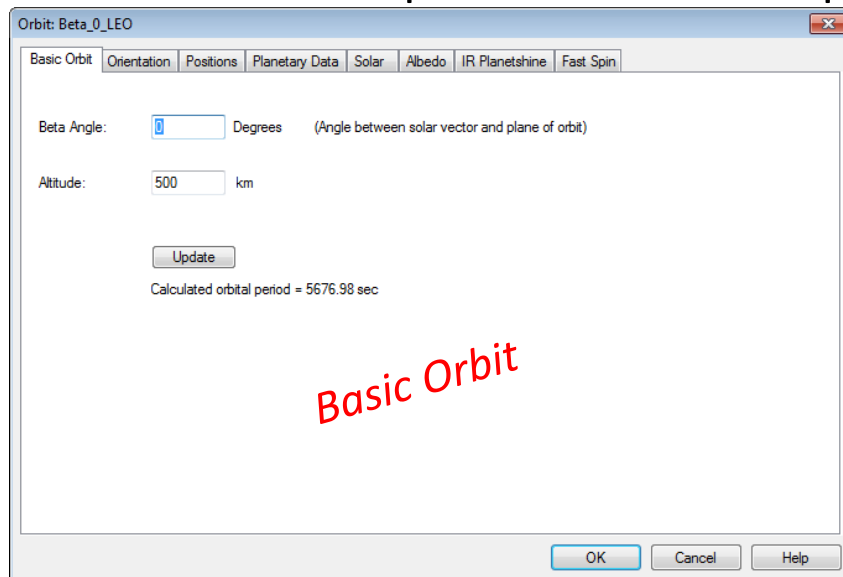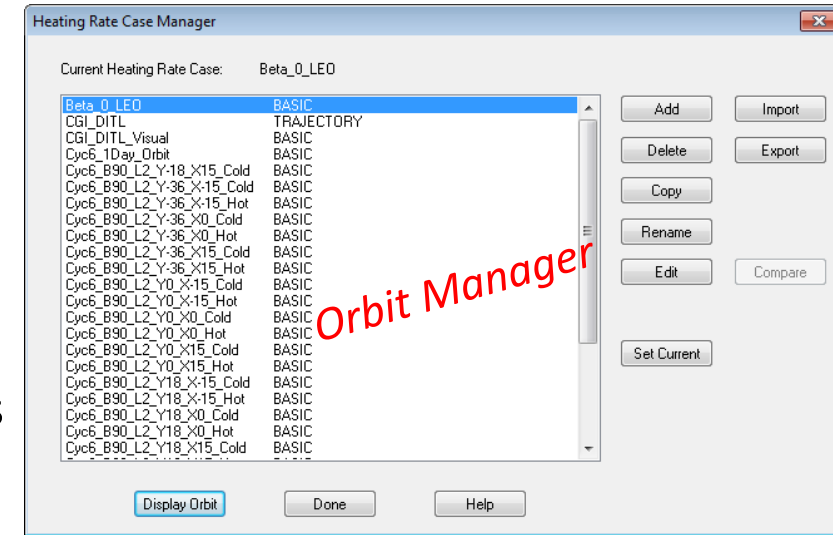- Layers for AutoCAD layers for visibility control
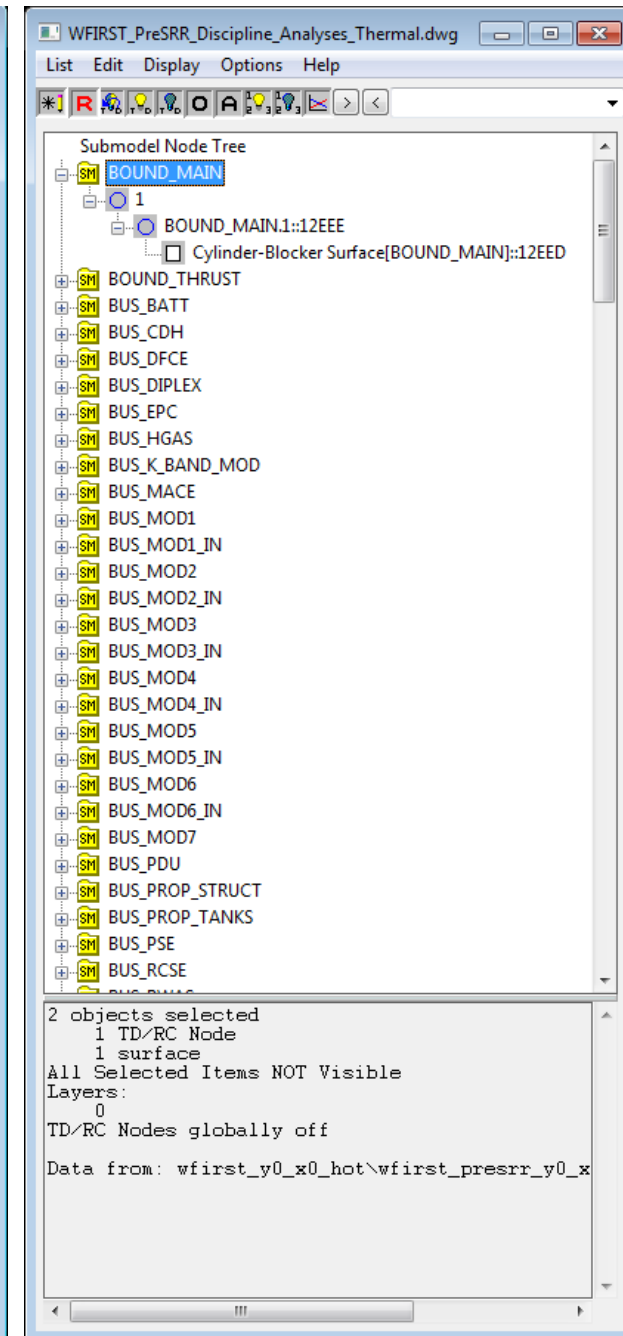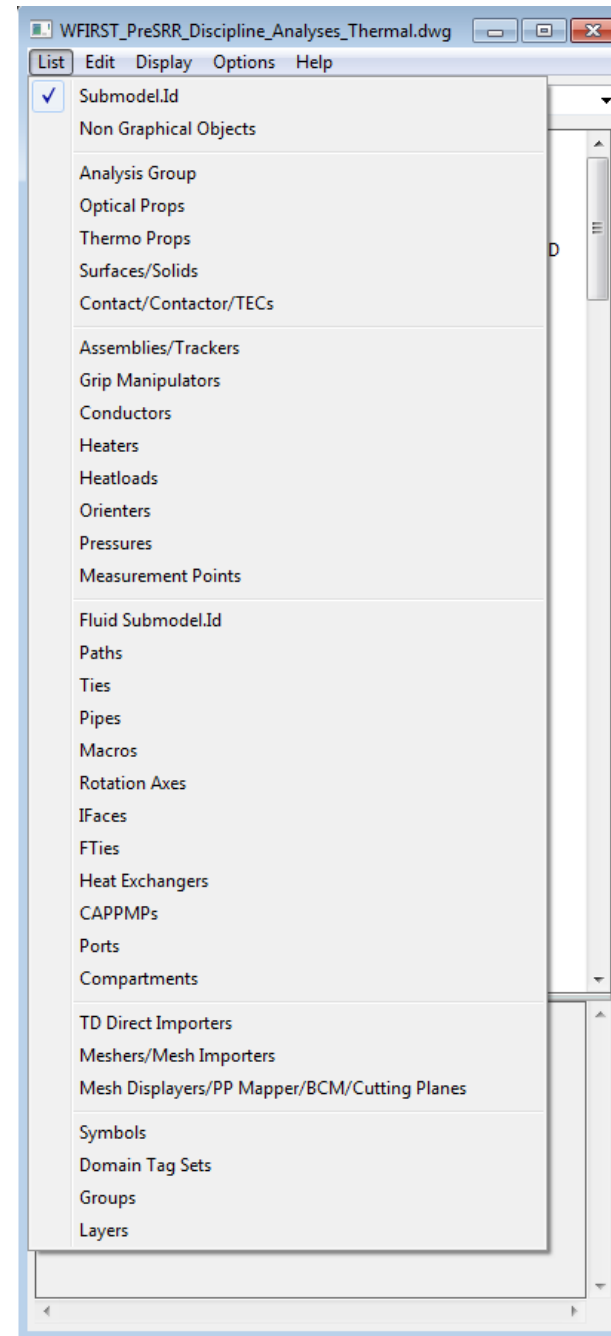
EXECUTE MODEL: *Defining Analysis Cases*

- ✓ **Define Orbits**

- ✓ **Define Case Set name and File Names**

- ✓ **Define Radiation Tasks (Radiation Groups, Radk, HR, Articulating Radks, Orbits)**

- ✓ **Symbol Overrides**

- ✓ **Property Overrides**

- ✓ **Operations Block (TIMEND, Steady, Transient, etc)**

- ✓ **Verify Control Cases**

- ✓ **Define Outputs and Intervals**

- ✓ **Add Comments to document Case Set**

- ✓ **Temporarily Set all Symbols, Aliases, and Property DataBases to case set and verify model is as intended. Restore back when done verifying...**

- ✓ **Run Case(s)**

- ✓ **Post Process Results**

Orbit Definition

Radiation Task Definition

Case Set Definition

Property Overrides

Symbol Overrides
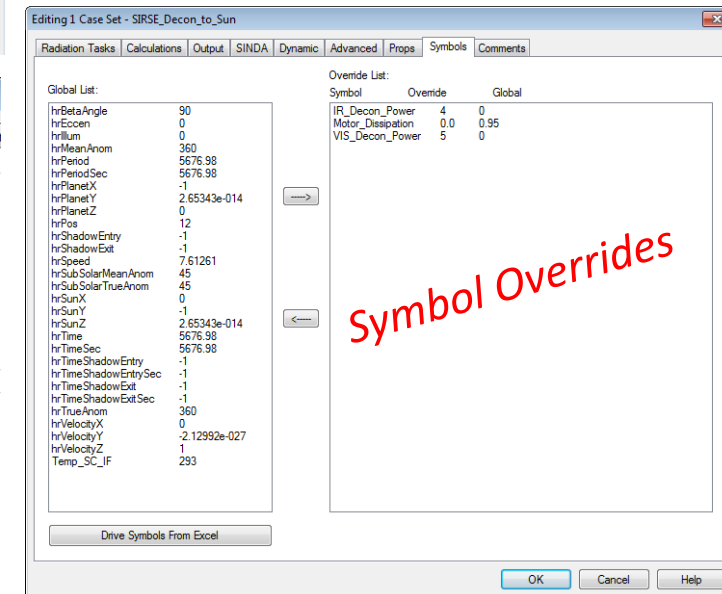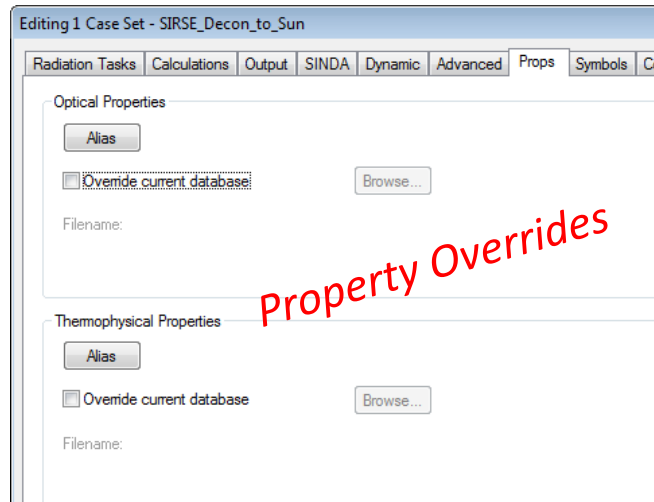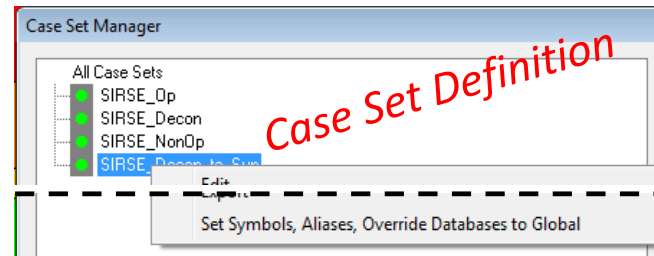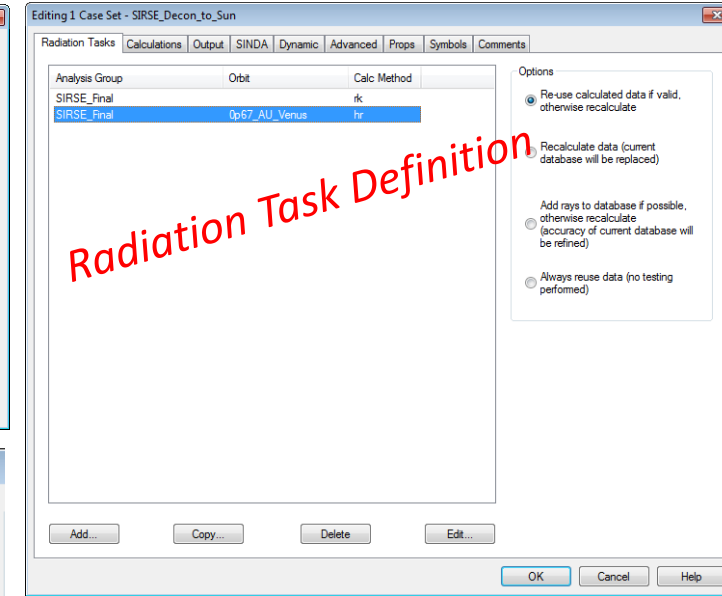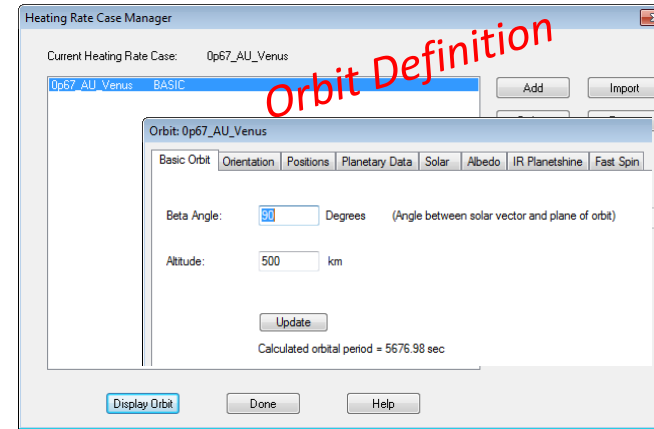
EXECUTE MODEL: Defining Orbits

- Orbits are defined to specify the variability of heating based on orbiting a celestial object (Planet, Sun, Moon, etc)
- Sources typically include: Direct Solar, Planetary Heating, and Albedo (Solar Reflections off Planet)
- Orbits can be specified by Altitude, Planetary Data (Size, Mass, Temperature, Fluxes, etc) or by Vector List (Good for small body orbits (comet, asteroid)). Orbits internally decompose to vector list for calculations
  - *Can use Beta = 90 with no Earth/Albedo for Lagrange orbits*
- Orbit Manager allows you to define multiple orbits and specify the current orbit for visualization
- Orbits are referenced in Radiation Tasks in the Case Set Manager
- Orbits can be imported from and exported to other models

Orbit Manager

Basic Orbit

Keplerian Orbit

Vector List

EXECUTE MODEL: Defining Orbits

- ✓ Open Orbit Manager and select Add…
- ✓ Define Type of orbit and Specify Name
- ✓ Enter Basic orbit information (e.g. Alt., Beta Angle, Incl., RAAN, Vector List, etc)
- ✓ Specify orientation of SC wrt celestial objects (e.g. Sun/Nadir/Zenith Pointing)
- ✓ Specify Euler Rotations to orient SC. Can use hrMeanAnom to do slow spin…
- ✓ Define number of orbit positions (default is 12: 15 positions with beginning and end + Eclipse Entry and Exit)
- ✓ Specify environmental heating (Solar, Albedo, IR Planetshine) Albedo and Planetshine can be Lat/Long dependent
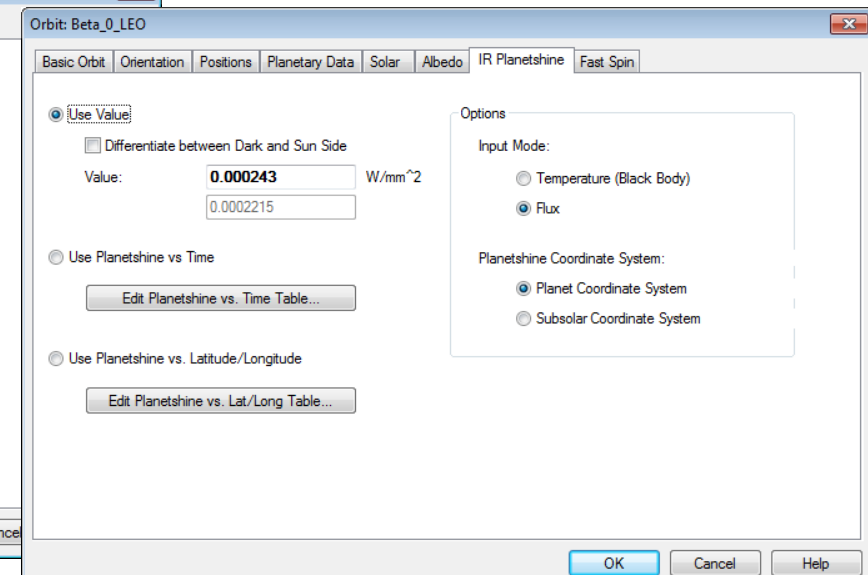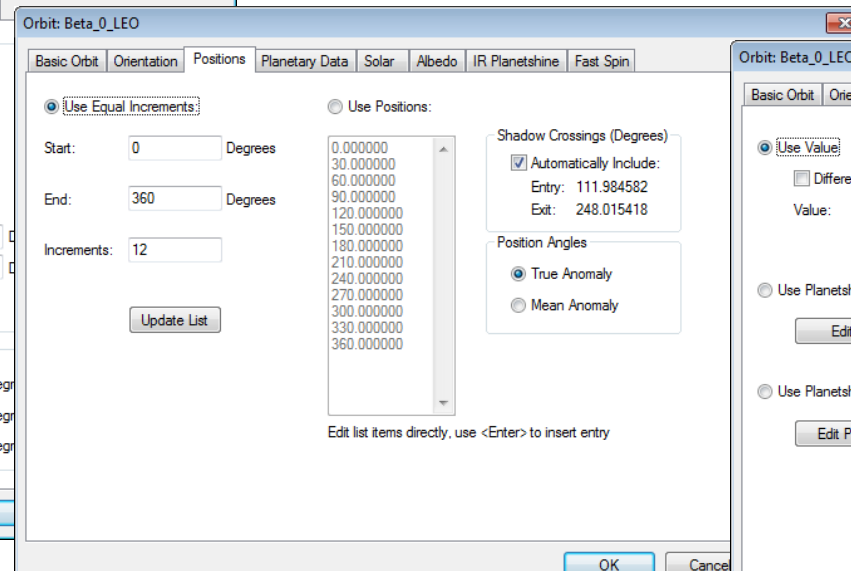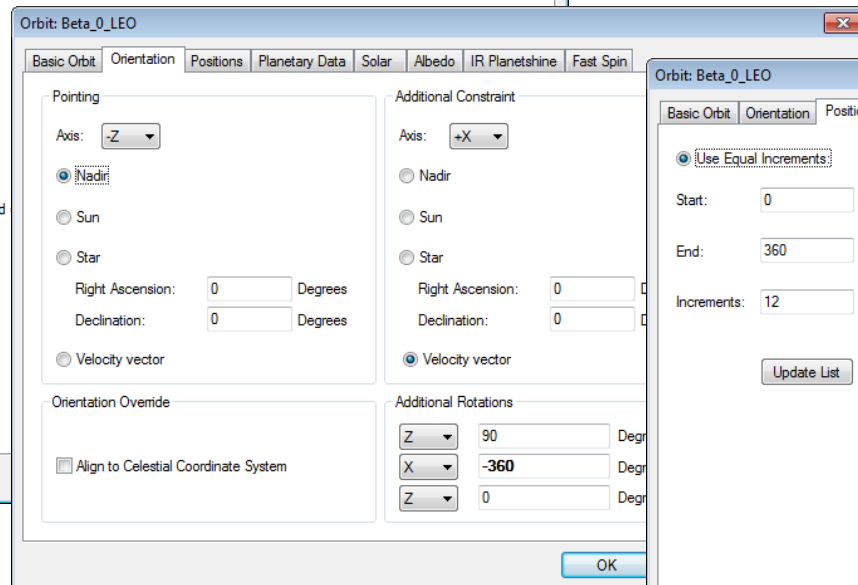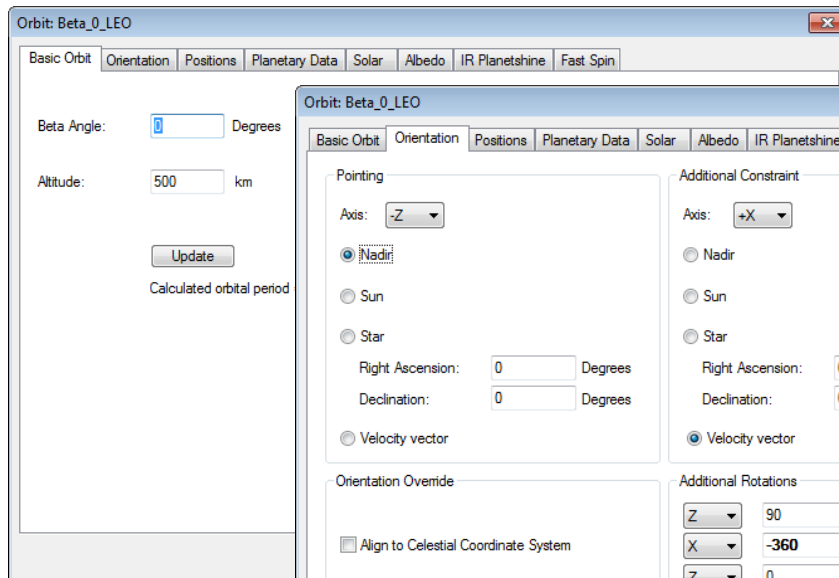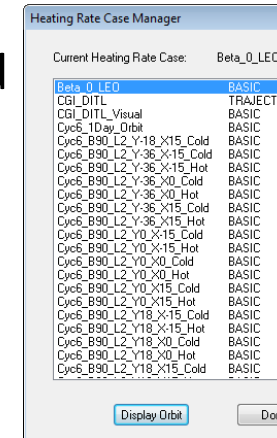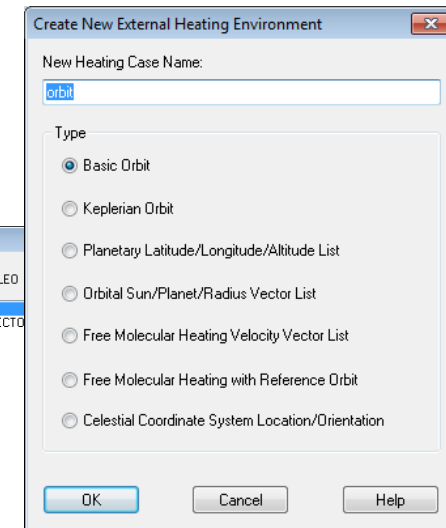
EXECUTE MODEL: *Defining Radiation Tasks*

- Radiation Tasks are a subset of a Case Set. The results of these Tasks serve as inputs to the thermal model and generally include internal radiation within an enclosure, external radiation, and environmental heating. Radiation can be for moving geometry.
- Each radiation task generates a file that is included along with the Cond/Cap calcs to form the SINDA model for temperature solution
- Results from Radiation Tasks may be used by multiple case sets. TD is generally fairly smart at evaluating if anything has changed since the last time a radiation task was run and determining if the data is still valid. User can over ride this…
- Each radiation task has its own control parameters including Max Rays to fire, Acceptable Error Criteria, Output File and Submodel, Oct Cell Subdivision, etc
- Fast Spin Capability also exists to vary a spin angle multiple times within one orbit position. Averaging these results (Radk/HR) together over multiple spin positions in one orbit position can simulate a fast spinning object (Scan Mirror, Reflector, etc). To effectively utilize this, an assembly should be created with the spinning geometry attached and the rotation based on the symbol representing the spin angle. **Use 0 Error!**
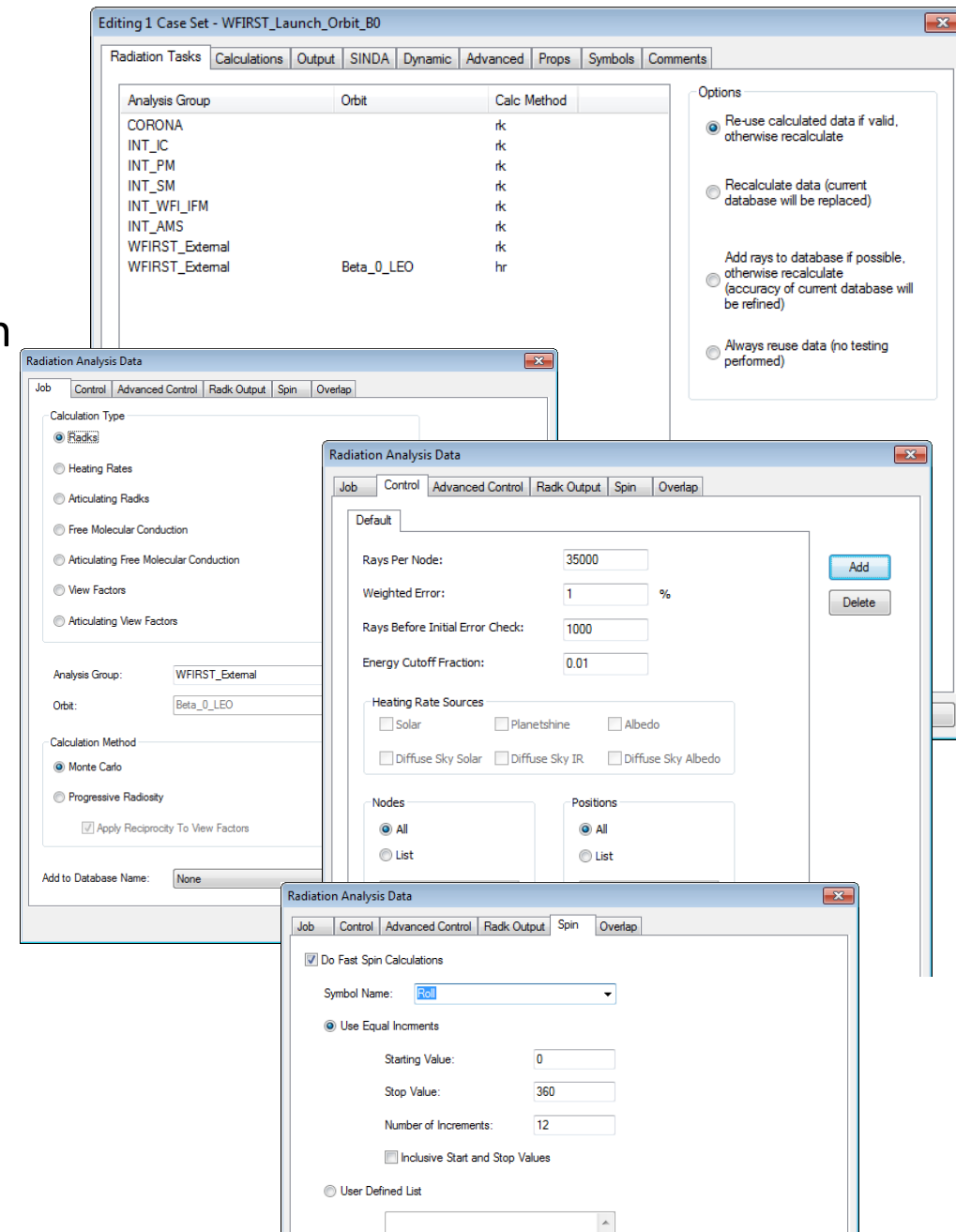
EXECUTE MODEL: Defining Radiation Tasks

✓ Open Case Set, Go to *Radiation Tasks* Tab, Click Add…

✓ Define Type of Task (Radk, Heating Rates), Analysis Group, and Orbit if needed

✓ *Add to Database Name* if you don't want to overwrite similar
   – Name of Database is RadGroup-Orbit-PropFile.rch or RadGroup-PropFile.rck

✓ Specify Max Rays, Error, and Energy Cutoff under *Control*

✓ Specify Levels and Objects on *Advanced Control* (Objects > 8)

✓ Set output filename and submodel…***DO NOT USE DEFAULTS!!***

✓ Set Bij Cutoff and Sum and disable Radks to space if internal radiation
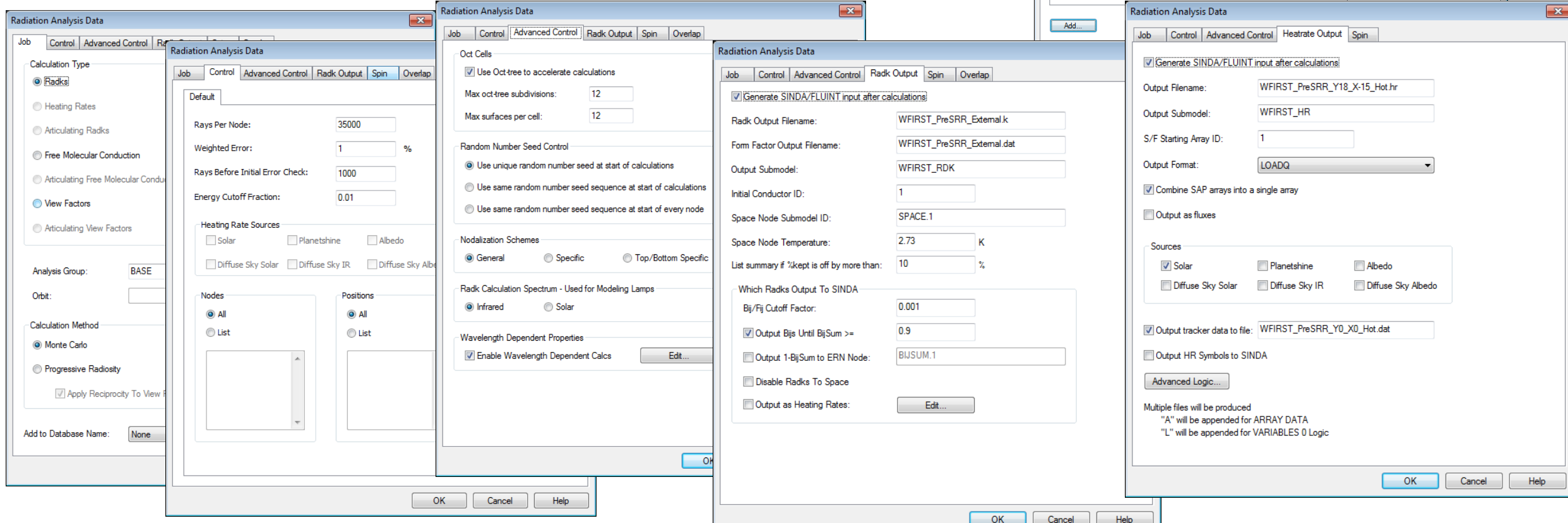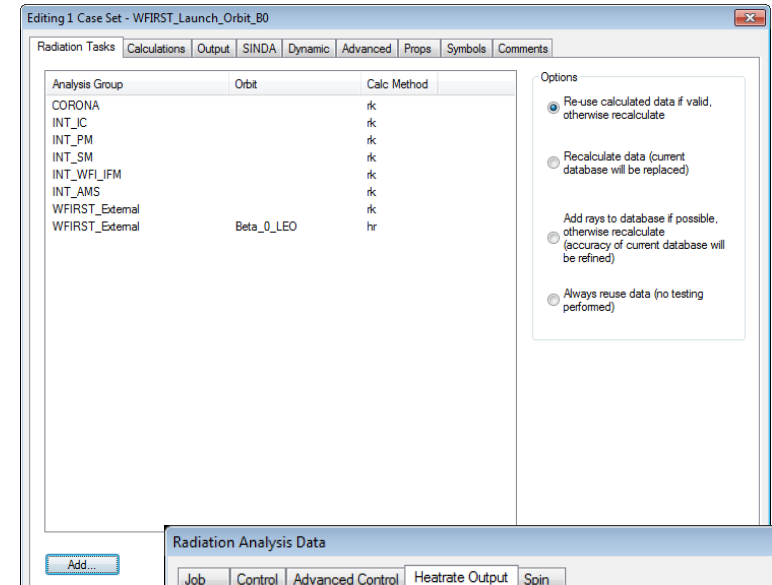
EXECUTE MODEL: Radiation Calculations

- **Monte Carlo Ray Trace**
  - *Begin with Surface 1 and find a random point and direction to "fire" the ray with 100% energy*
  - *Determine if the ray intersects any surface*
  - *Oct Cells subdivide model to reduce number of intersection tests*
  - *If ray intersects a surface, determine how much energy is absorbed by the surface (α, ε)*
  - *If some energy remains, determine if the energy is reflected specularly (Angle of Incidence = Angle of Reflection) or diffusely (Random direction)*
  - *Continue propagating ray until energy is below extinction threshold at which point it is either completely absorbed or completely reflected*
  - *Once extinguished, select new random point and direction for next ray*
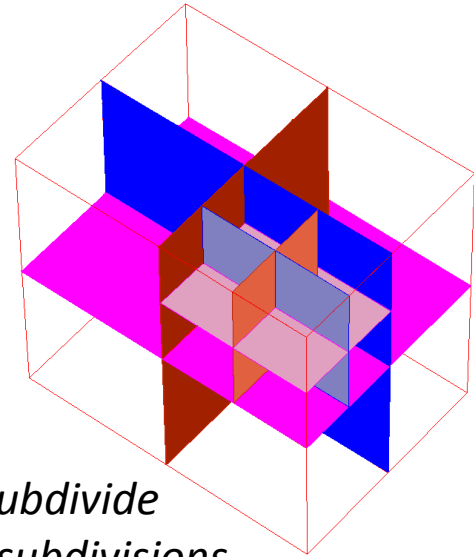 - *Once finished with Surface 1, move to next surface*
- **How Many Rays?**
  - *Typically start with 35000. (2.65/1.76% error for 0.1/0.2 B_{ij})*
  - *Is the run time tolerable?*
  - *Spending time calculating small couplings that are eliminated?*
  - *Does it affect temperatures? By how much?*
  - *In the end it comes down to run time vs. accuracy…*
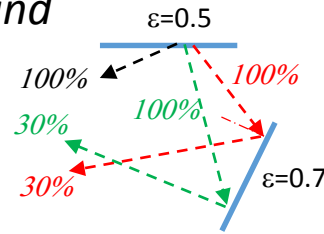
$\varepsilon=0.5$
100%   100%
30%   100%
30% $\varepsilon=0.7$

- **Oct Cells**
  - *Find bounding box around entire model*
  - *Divide by midplanes in all three directions to form 8 smaller boxes*
  - *Determine #surfaces in each of the 8 cells*
  - *If #surfaces is > specified, then subdivide cell again. Continue until max #subdivisions or max #surfaces is reached*
  - *Consider 8 surfaces meeting at point (i.e. ribs). Lower #surfaces can never be met…*
- **Error Calculations**
  - *Statistically, the error can be estimated with 90% confidence by:*

$$Error_{ij} = 1.65 \sqrt{\frac{1 - B_{ij}}{N_{rays} \, B_{ij}}} \times 100$$

  - *Weighting the Error for every node yields a single term per surface to determine if sufficient rays have been fired for that surface*

$$Weighted \; Error_i = \frac{\sum_{j=0}^{n}(B_{ij} * Error_{ij})}{\sum_{j=0}^{n}(B_{ij})}$$

VERIFY RESULTS

- Verify Results refers specifically to the process of confirming that the model and predictions are consistent with the physics of the design.
- Model correlation is a measure of how accurately the model predicts actual measured values under the same conditions as tested…this happens well after a model is built, much later in a program
- The first thing to check are temperatures. A contour plot is useful for this as it will highlight any gradients
- If the contours look reasonable (what should be cold is cold, what should be hot is hot), the next thing to compare is the temperatures against design limits. XY plots are good to see the local max and min values over an orbit, but it is hard to do this for every node of importance. Comparison of the hottest temperature and coldest temperature of a component (or group of node) over the orbit to the design limit is best accomplished using a spreadsheet and is often represented in a tabular form.
- XY plots are also useful to display the nodal stability and compare against any stability requirements
- For heaters, it is important to confirm that the heater power is sufficient. No more than 70% of the available heater power should be used per GSFC GOLD rules. The required amount of power to maintain a component within limits does not depend on heater size, but duty cycle does…
- A variety of external tools exist for post processing. TD offers XY, Contour Plots, Animated Contour Plots and methods for: querying Max/Mins, writing output text files, and calculating sinks and heat flows post-solution
- ***Lastly, to really understand a design, there is no substitute for investigating heat flows. Heat flows identify areas of a design where better coupling is needed, better isolation is needed, or unexpected heat inputs or leaks are present. This takes a fair bit of time to set up, but the understanding of a model/design is far better after having gone through the effort.***