LA-UR-11-04369

Approved for public release;
distribution is unlimited.

Title: Creating A Merged Single-Part Abaqus Model For Use In MCNP6 (U)

Author(s): Kevin Marshall

Intended for: General release with MCNP6

Creating A Merged Single-Part Abaqus Model For Use In MCNP6

Kevin Marshall 07/20/2011

Abstract

This document is a quick-start guide to creating an Abaqus model that is intended for use in MCNP6, and contains only *one* active part in the instance of the Abaqus assembly.

1.Introduction

The overall process for creating a single merged part essentially involves creating many individual Abaqus parts, which represent the 'conceptual parts' or 'components', before merging them into one single Abaqus part, within which the 'conceptual parts/components' can be represented as Abaqus element sets (*statistic and material).* So, for example, if one were making an Abaqus model of a car, then one might make individual parts for the wheels, the engine, the hood, the roof, etc. These would then be merged together to make one single Abaqus part, within which the individual components/parts are *easily* selected as element sets, thus allowing these components to be assigned their own pseudo-cells.

This can be considered as a parallel approach to creating a model with individual Abaqus parts for each of the conceptual parts, and the user should consider which is most appropriate for their own problem in hand. In general, the approach described here whereby all of the initial parts are merged into a single part will guarantee that the mesh will be continuous over the entire geometry. This can be useful if the subsequent radiation transport data is to be used within another analysis code that requires a continuous mesh. On the other hand, creating an Abaqus model that contains multiple Abaqus parts can be useful as it allows meshes to be created that have discontinuities between each of the parts. In this situation Abaqus will create tie constraints between the the mesh on the two parts. Where the model to be created has very specific requirements on the mesh, creating multiple parts with tie constraints can sometimes be the only option.

2. Creating a simple car

All of the following was created in Abaqus version **6.9.2**

2.1 Create the individual parts.

The fist step is to create all of the individual Abaqus parts. This should be done as one would any normal Abaqus part (Figure 1). This model consists of 4 wheels, 4 tires, a chassis, a hood, a trunk, and a cab compartment. These should all be made individually, either by extruding, rotating or sweeping, whichever is appropriate. Parts that are to be duplicated, such as wheels, should only be created once at this stage.

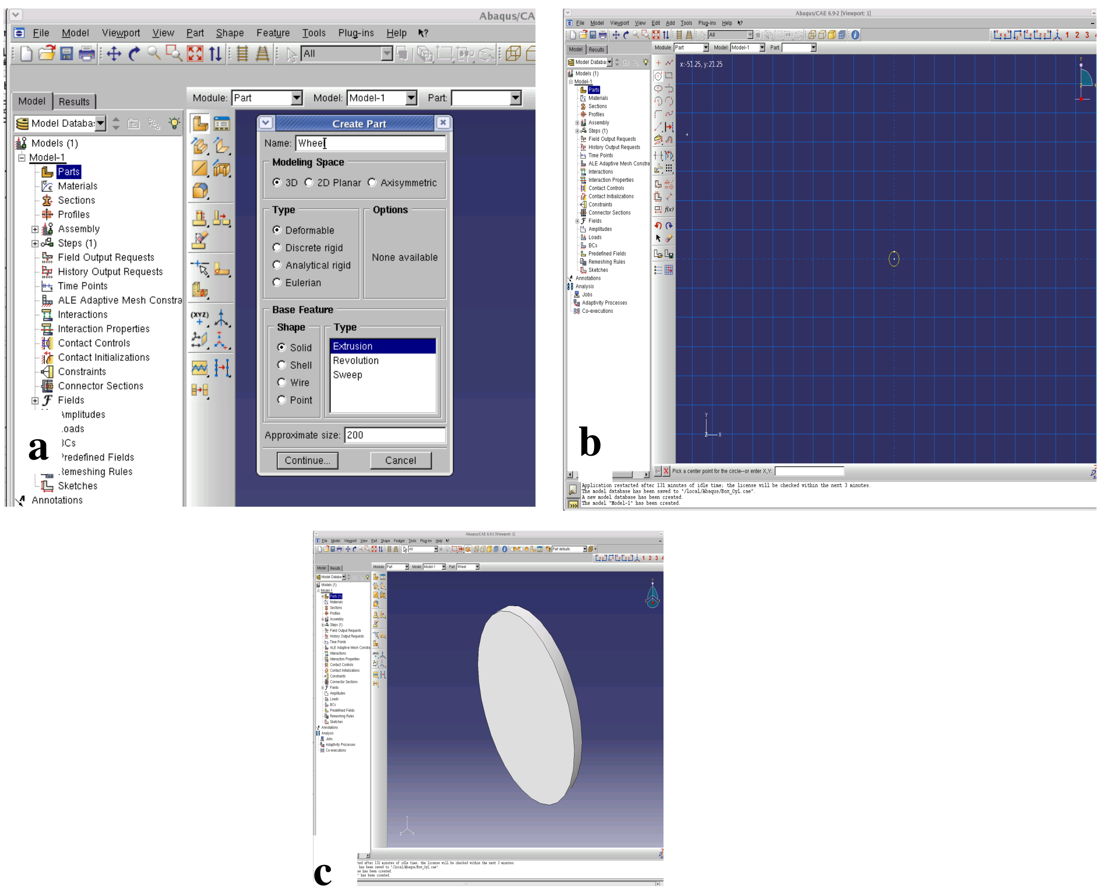


Figure 1. Process for creating a wheel. (a) Create the parts using extrude, revolve of sweep. (b) Draw the sketch for the part. (c) Extrude the part.

Once all of the parts have been created they will be visible in the part tree. Upon creation of all of the parts an assembly instance should be created that contains all of the parts. Any parts that occur multiple times, such as the wheels and the tires should be imported into the assembly the required number of times.

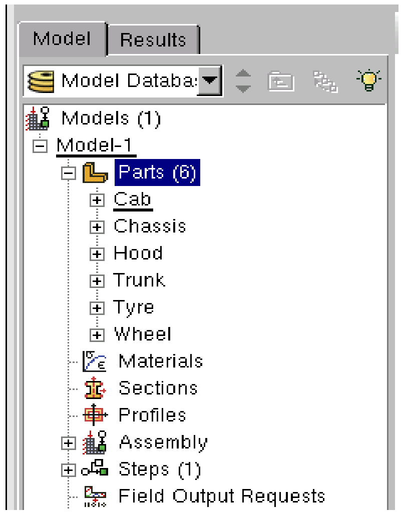


Figure 2. All individually created parts will be shown in the *part tree.*

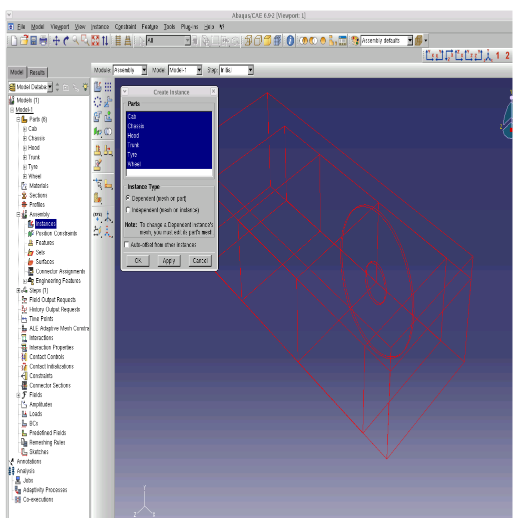


Figure 3. All parts should be imported as an instance into the assembly. Depending on how the parts were sketched, they may not initially be in the correct position.

2.2Assemble The Parts

2.2.1 Cut Overlapping Parts

Once the parts have been imported into the assembly they will initially be in their positions relative to the sketch origin. If they were sketched in their correct positions then this is fine. However, if they are not in the correct positions relative to each other they will need to be moved using the transform function, See Figure 4(a).

Once they are in the correct positions, as shown in Figure 4(b), any parts that overlap in space should be subtracted from each other. This subtraction acts only on parts that are selected when carrying out the operation. So, in this example the wheels and their tires overlap. Therefore the wheels should be subtracted from the tires using the boolean 'cut' operation, accessed via the 'Merge/Cut Instances' tool, Figure 5. When carrying out a 'cut' operation to separate two overlapping parts, the '*suppress*' option should be checked in the 'boolean' options box. Otherwise, once the operation is complete, the parts doing the cutting, i.e. the tire in this case, will be lost. The suppress option allows the part doing the cutting to be resumed. Here, the tire should be the part that is to be cut, and the wheel is the part to make the cut. So, once the 'wheel part' has cut the 'wheel hole' out of the tire it can be resumed back into that region. The boolean option box will ask for a name for the new 'cut' part that will be created, Figure 6(a). It is useful to give this new part a meaningful name, such as 'tire_cut' to distinguish it from the original uncut part. After the 'cut' operation there will therefore be three parts listed in the assembly, Figure 6(b); the wheel, the tire and the new part, which is the original tire with the wheel hole cut out, Figure 6(c). The wheel can now be resumed to fill the hole in the tire . This is simply done by right-clicking on the wheel in the assembly's part list and selecting *resume,* Figure 6(d). Normally the original uncut part should be left suppressed in the assembly's part list (shown with a red cross), although it can be resumed at any time if required by right-clicking over it and selecting 'resume'.

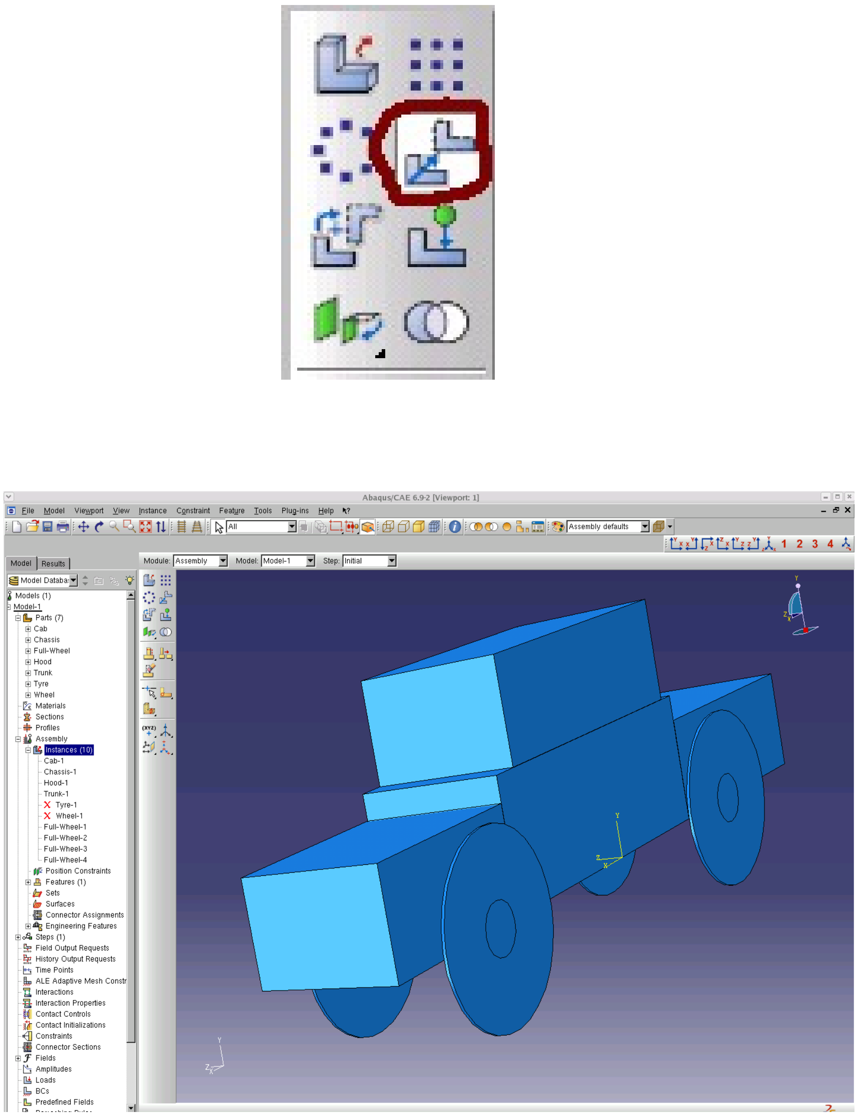


Figure 4(a) The translation tool. (b) The assembly after all of the individual parts have been translated into their correct positions.

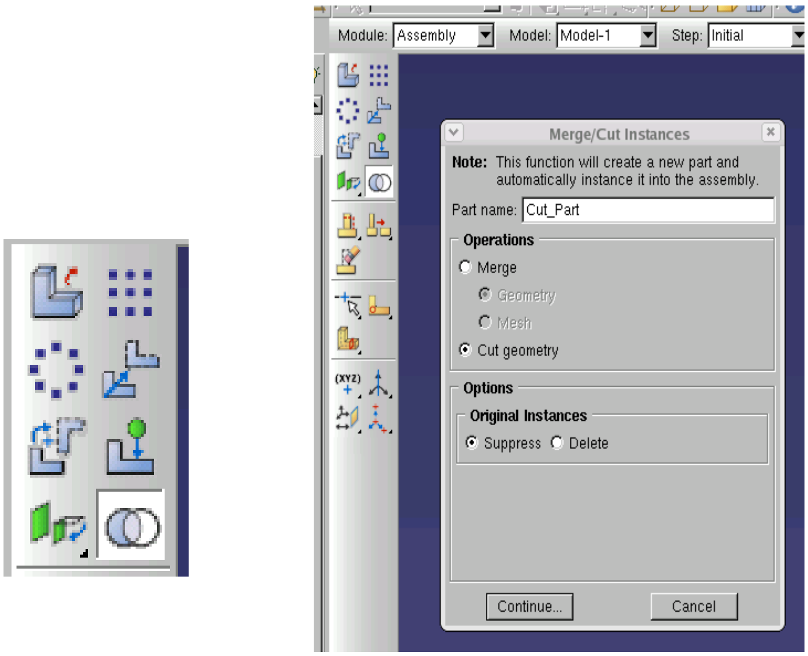


Figure 5. The Boolean 'cut' function in Abaqus. (a) The boolean 'Merge/Cut Instances' icon. (b) The boolean 'Merge/Cut Instances' dialogue box. For the processes described here the 'Cut geometry' option should be selected, along with the 'Suppress' option, as shown.

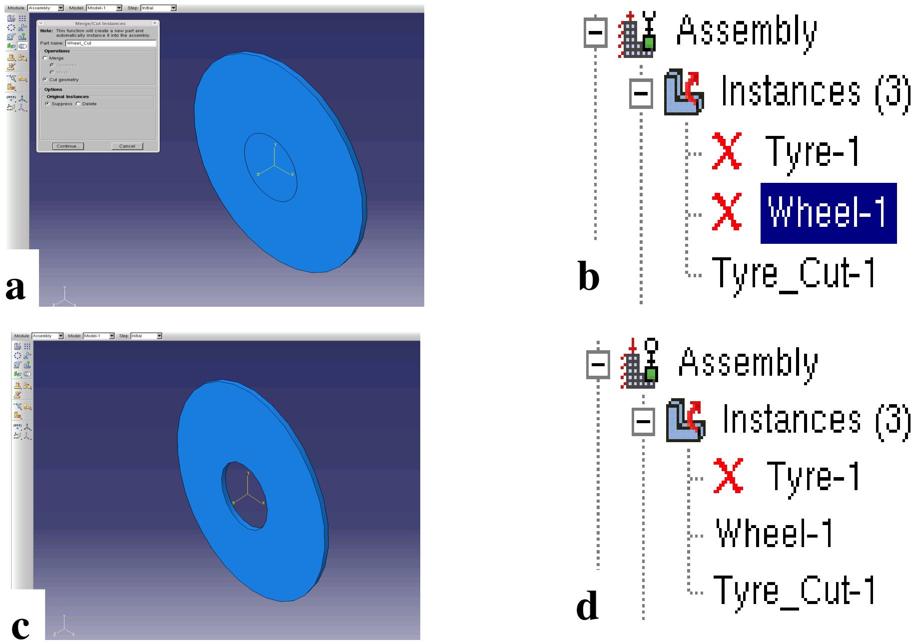


Figure 6. Cutting overlapping parts. (a) Within the boolean options box the 'cut' option should be selected, along with the 'suppress' option in the 'Original Instances' section. (b) After the 'cut' operation three parts will be listed in the assembly; the two suppressed original parts and the third cut part. (c) The newly created cut tire. (d) The part that was used to make the cut, i.e. the wheel, should be resumed

2.2.2Merging The Assembly

Once the assembly has been created such that the individual parts are in the correct positions and any overlapping parts have been cut then all of the parts can be merged to create a single merged part.

This merging is once again done using the boolean 'Merge/Cut Instances' tool, but this time the 'merge' option should be selected rather than the 'cut' option, as shown in Figure 7. In addition, the 'retain' option should be selected in the 'Intersecting Boundaries' section. This will allow information about the original parts to be retained within the final merged part, which can be useful for selecting element sets that correspond to the original parts.

The merging process will create a new part, which is the single merged part, with all other parts being shown as suppressed. This new part will also be created within the parts listed in the model tree as shown in Figure 8. Once again, within the 'Merge/Cut Instances' dialogue box it is sensible to provide a meaningful name for the new merged part that is to be created.

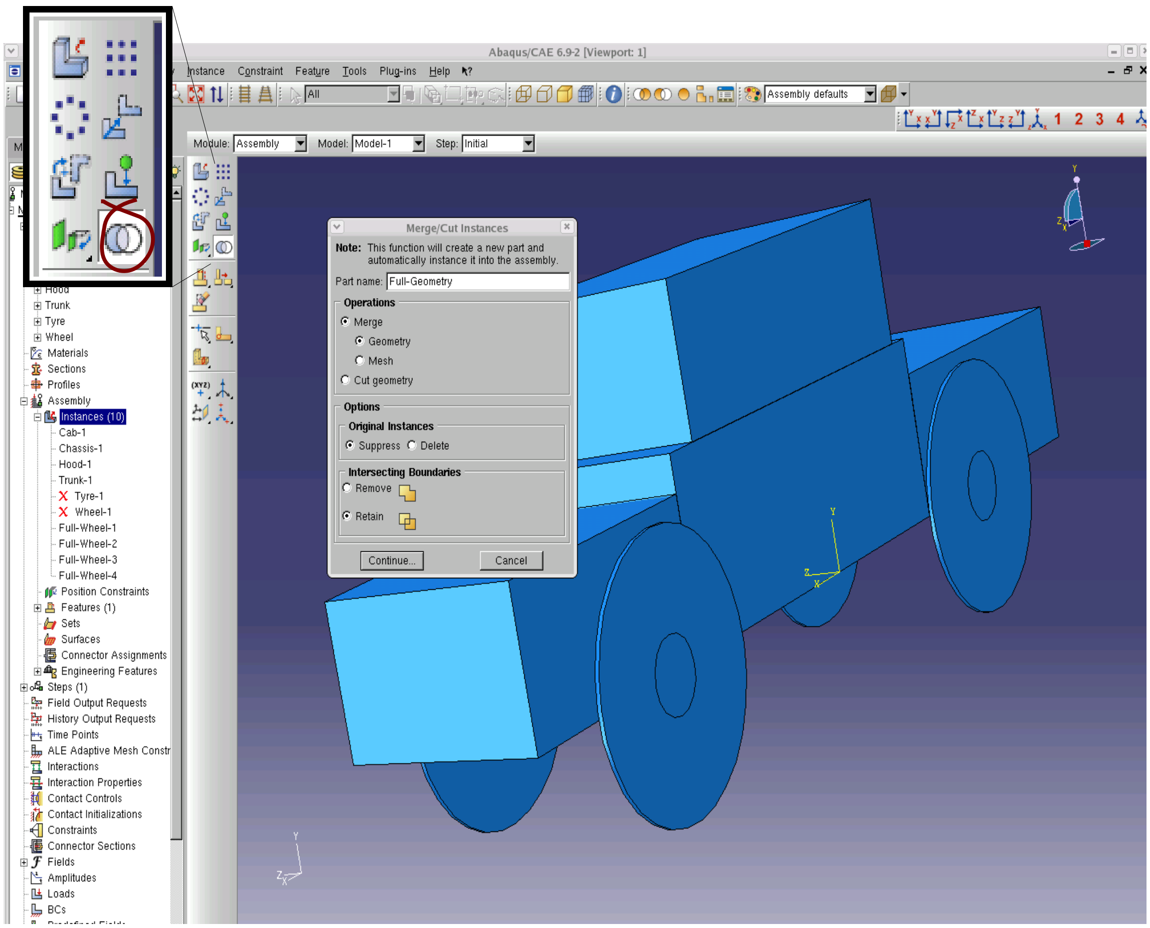


Figure 7. Merging the parts. Within the boolean 'Merge/Cut Instances' dialogue box the 'Merge' option should be selected along with 'retain'. All of the instances should then be selected for merging.

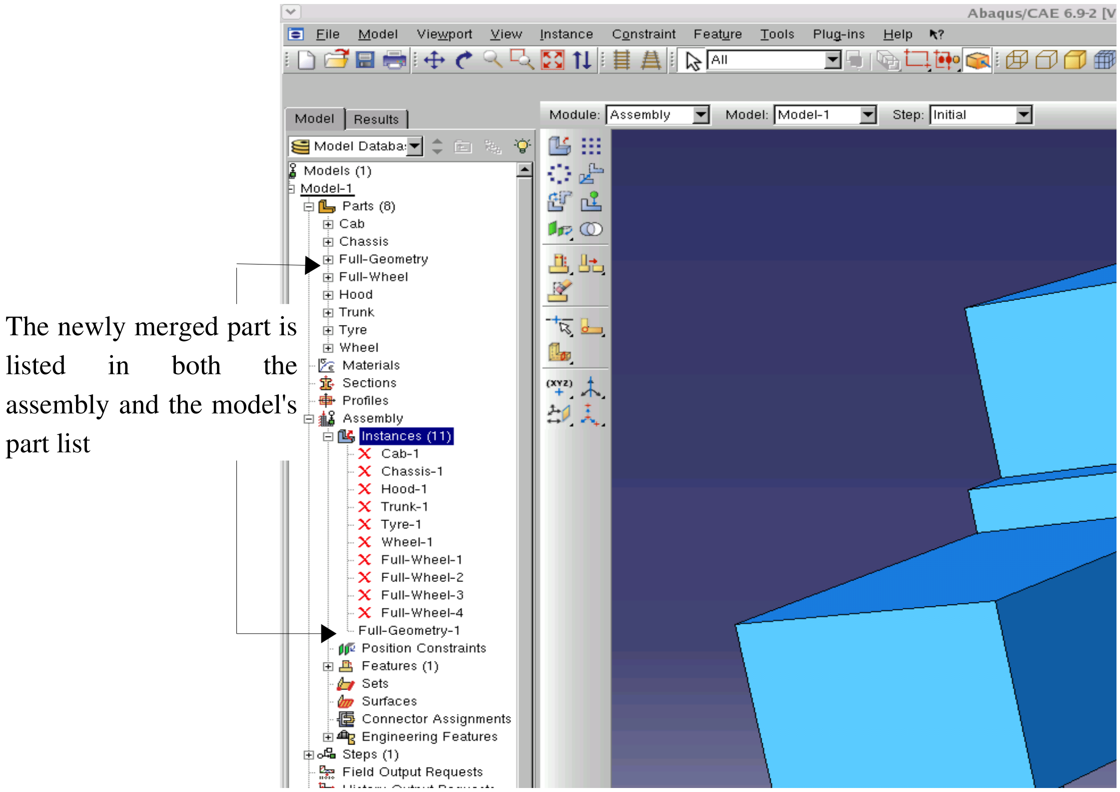


Figure 8. Once all of the parts have been merged into a single part only the merged part is shown as non-suppressed within the assembly's parts. The new merged part also appears within the model's part list and is the part on which the mesh should be generated.

2.3 Selecting Element Sets On individual Parts

It should be pointed out at this point that for any geometry to be passed to MCNP6 in an Abaqus format the '.inp' file must contain the appropriate element sets for each part. When creating an Abaqus geometry as described here there are two possible routes for doing this. Firstly, the element sets can be selected within each individual part before they are imported into the assembly. When the parts are merged the new merged part will contain all of the element set information from the parts that created it. However, any subsequent changes to the original parts in terms of their element sets will *not* be reflected in the merged part. This being the case, all element sets that are created in the original, unmerged parts should be created in advance of the merged part being created.

The second, alternative option is to create the element sets entirely on the merged part. This method is mentioned in the section below. Whether the element sets are created in the original parts or merged part is largely up to the preference of the user. However, for models that contain a large number of small parts, which comprise a complicated assembly, there may be a case for favouring creating the element sets on the original parts.

3. Meshing a simple merged part

The newly merged part will be created in the list of parts in the model tree (Figure 8) and it is this part that should be meshed. Within the mesh mode of the 'merged part' the mesh type for all of the regions should be set. Depending on the geometry and the type of mesh desired, it can be useful, if possible, to set the entire part to be of one mesh-type as Abaqus cannot introduce tie constraints within a single part. However, individual regions corresponding to the original parts can easily be individually selected for assigning different mesh types (setting the selection filter to 'Cells', as shown in Figure 9, is particularly useful when doing this).

Once the mesh type has been assigned, the desired seeding can also be set and the part meshed. Element sets can also be selected before or after meshing. If the selection filter is set to 'Cells' (Figure 9) then the regions corresponding to the original parts can easily be selected as element sets.

Once all of the required element sets have been selected then the model is ready to be written to the '.inp' file. As the only part not suppressed in the Assembly is the merged part, only the information pertaining to this part will be contained within the '.inp' file.

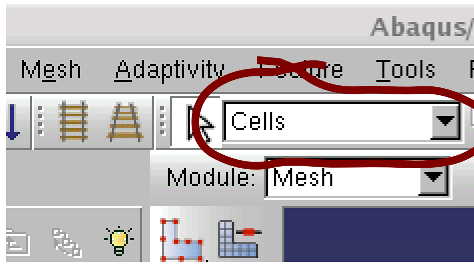


Figure 9. Setting the selection filter to 'Cells' allow only cells to be selected.

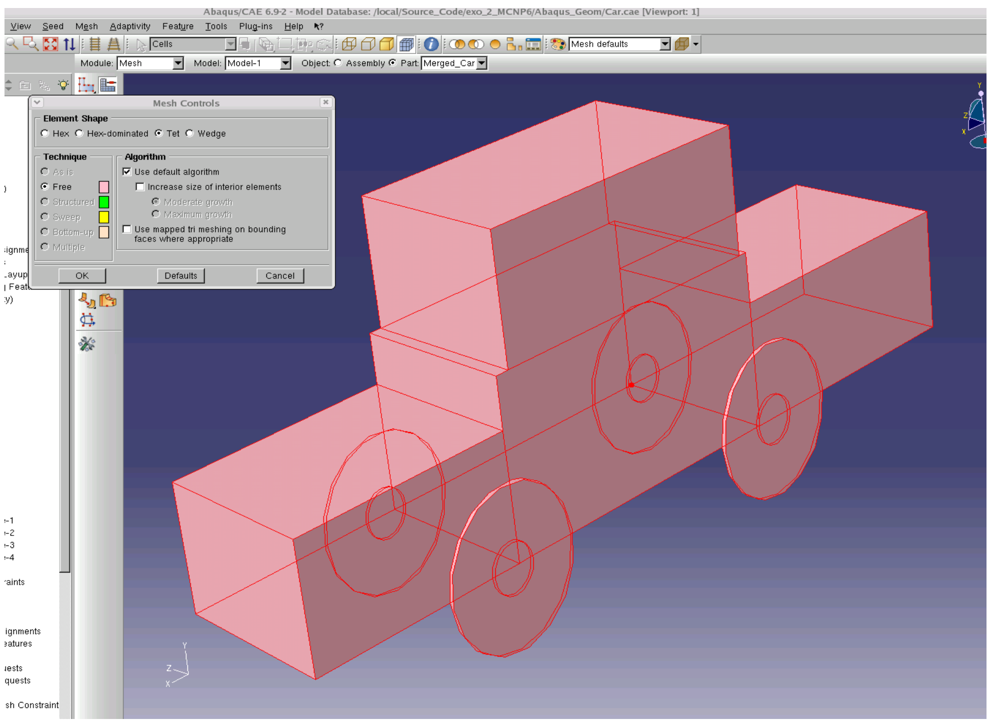


Figure 10. Within the mesh-mode of the merged part it is advisable to set all of the regions to the same mesh type.

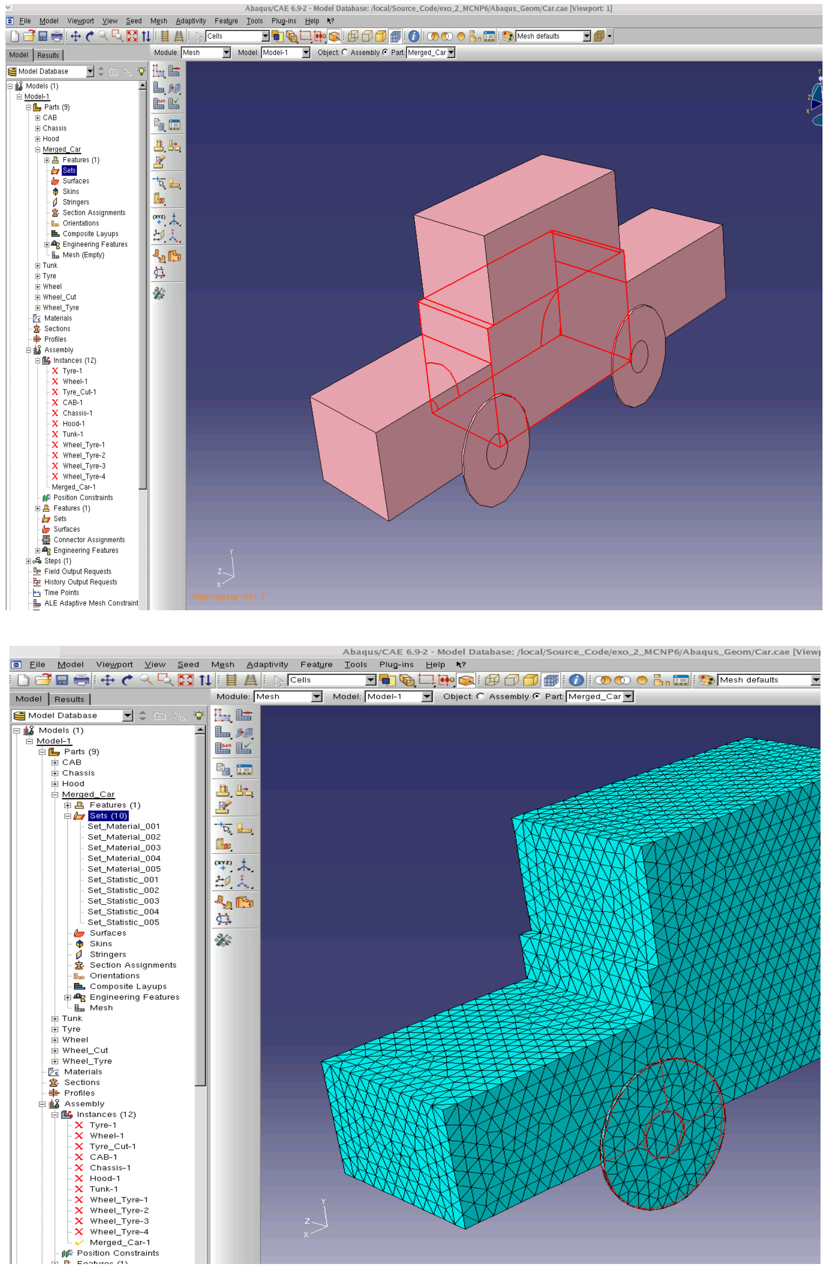


Figure 11. Picking the element sets. If the selection filter is set to 'cell' then it is easy to pick the element sets based on regions corresponding to the original parts. This can be done either before (a) or after (b) meshing.

4. Quick-Start Guide To Creating A Single Part From Multiple Original Parts

1) Individually create all of the parts that make-up the overall model.

2) Import all of the parts into the assembly. (Where there are multiple instances of an individual part import it multiple times and translate to the correct position)

3) Use the boolean 'Merge/Cut Instances' tool to cut away any overlaps between two parts. Ensure that the 'suppress' option is selected.

4) 'Resume' any of the parts that were used to make cuts and that are still required.

5) Merge the entire assembly using the boolean 'Merge/Cut Instances' command. Ensure that the 'retain' option is selected.

7) Observe that the only part not now suppressed within the assembly is the newly merged part, and that this part has been added to the list of parts within the model.

8) Now acting on the newly merged part within the model, set the mesh type for the regions of the part to the desired type.

9) Assign the appropriate element sets to the regions of the model. If the selection filter is set to 'Cells' then the regions corresponding to the original constituent parts can be selected.

10) Set the mesh seeding and mesh the part.

11) Create the job file, and write the '.inp'

12) The '.inp' file will contain a single part, with multiple element sets.

Appendix A

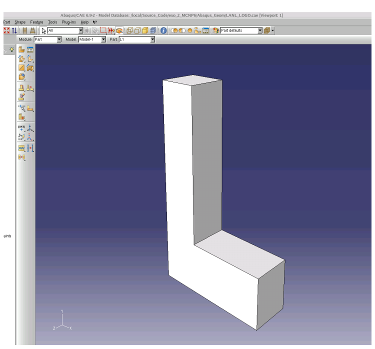
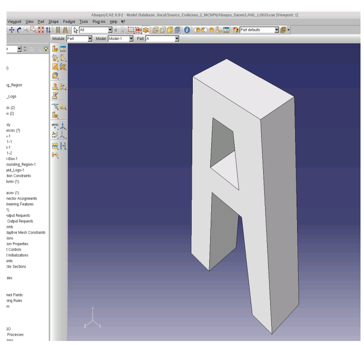
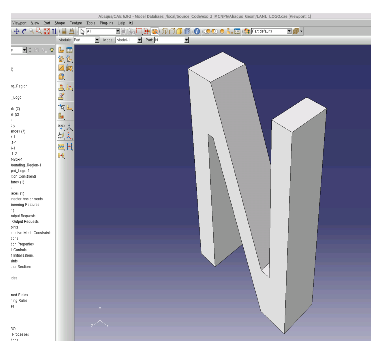
Creating A Model For The Abaqus To MCNP5 Translation Utility Developed By The Radiation Transport Team At AWE (UK)

The principle for creating a model for use in the Abaqus to MCNP5 conversion utility is largely the same as the process described in the main body of this report. However there are two main differences; the need for a bounding-box region, and the method for the picking of element sets.

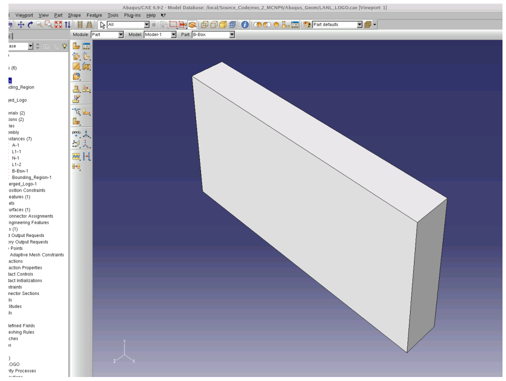
A1. Creating a bounding box.

This section briefly describes the process for creating a bounding box which is integrated into a single merged part. The bounding box is required in order for the translator to easily created a single surface that will allow the zero-importance external region to be defined relative to that, rather than the large number of 'arb' surfaces that make up the actual model.

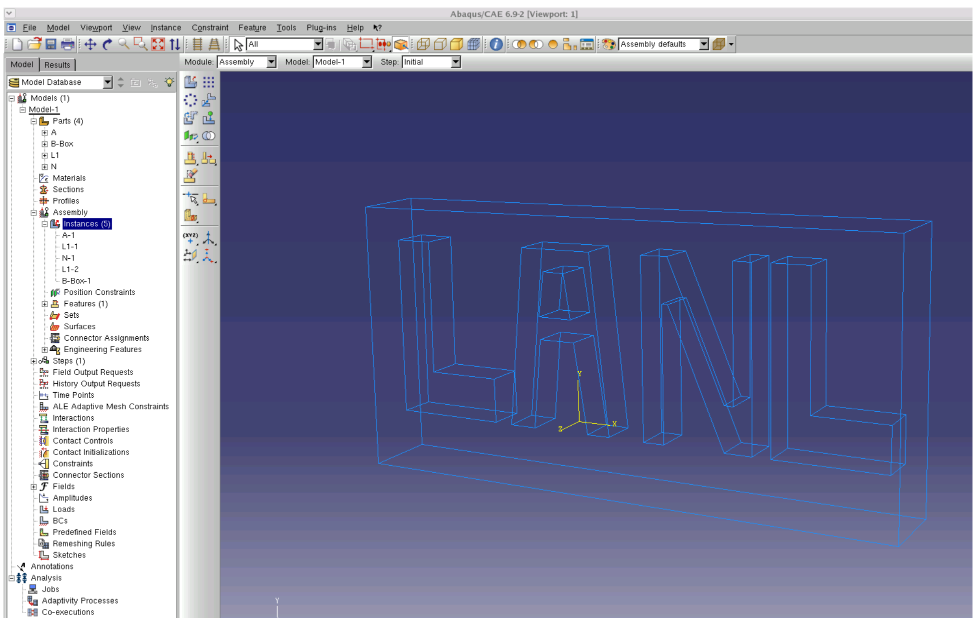
1) Create the individual parts that make-up the model.

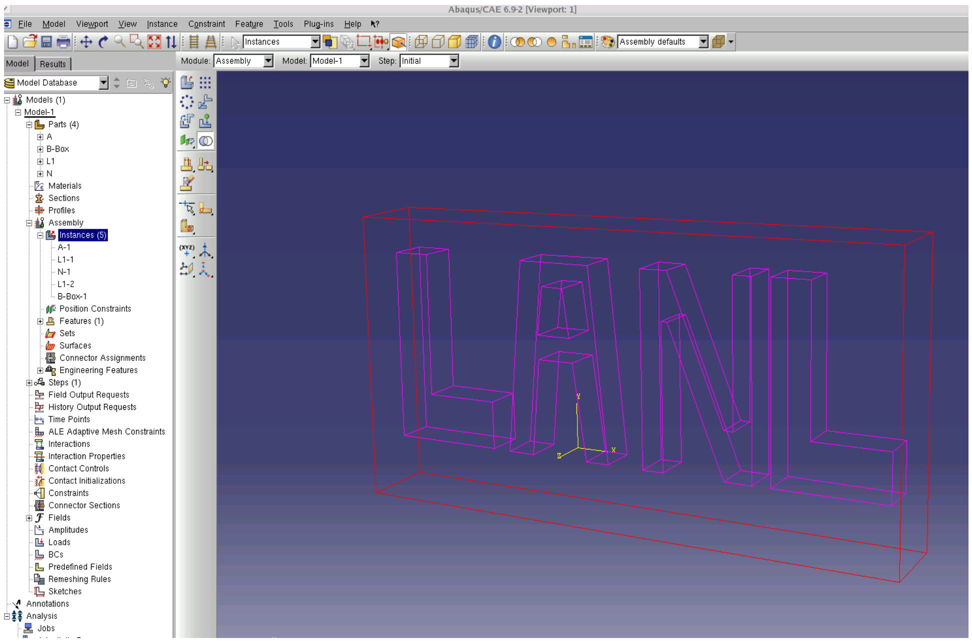
2) Create a suitable bounding box, i.e. that will encompass all of the parts when they are arranged correctly in the assembly. The box *must* be rectangular and *aligned with the main axes*.



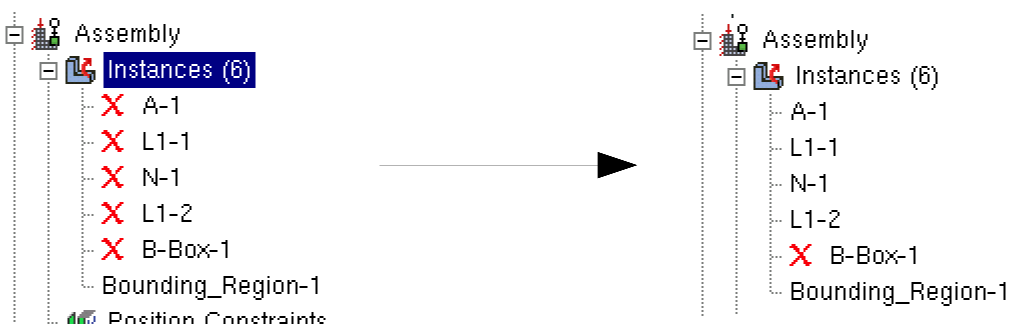
3) Import all of the parts, including the bounding box, into the assembly and translate as appropriate, I.e so that the parts are their correct relative positions and encompassed by the bounding box.



4) Cut all of the parts (apart from the bounding box) the from the bounding box using the boolean 'Merge/Cut Instances' tool and with the 'suppress' option selected. (Remember to give the resulting cut box that will be created a meaningful name).



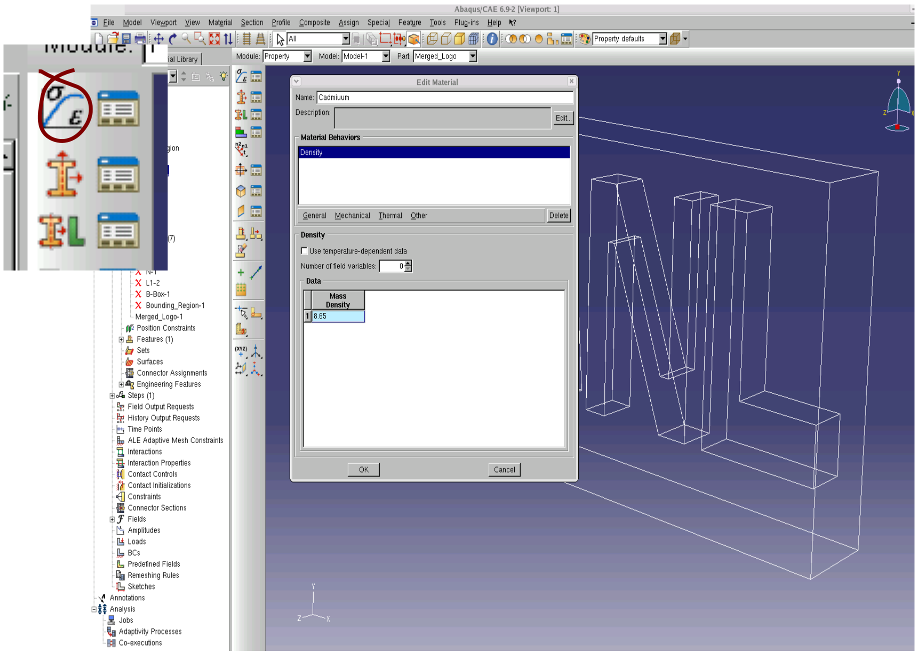
5) Resume all of the parts that were suppressed, apart from the original bounding box (this has now been replaced by a bounding box with space for the interior parts cut out). Only the original bounding box should be shown as suppressed after this.

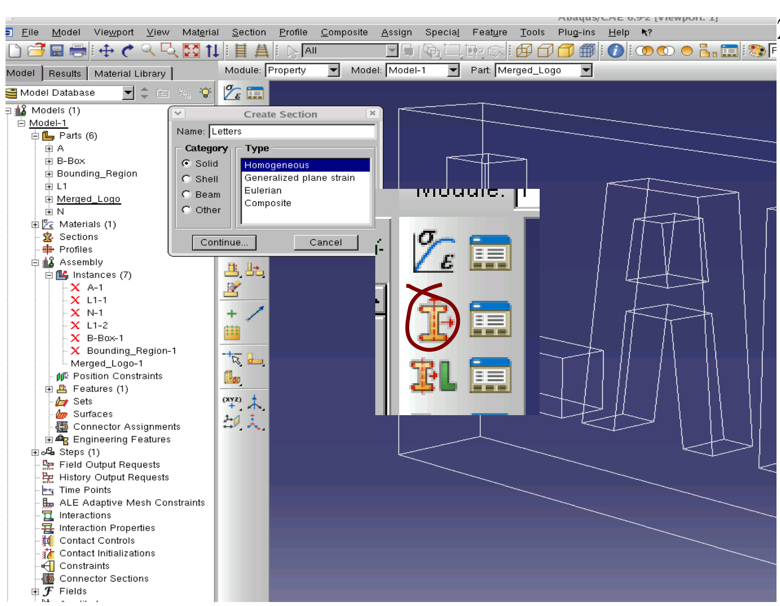


6) Merge all of the parts including the new bounding box, to create a single merged part. This new part should now be shown in the model's list of parts. The part can now be meshed as described in the main body of the report. The next section describes how to create 'pickedsets', which are required for translation into MCNP5 format.

A2. Assigning 'Pickedsets'.

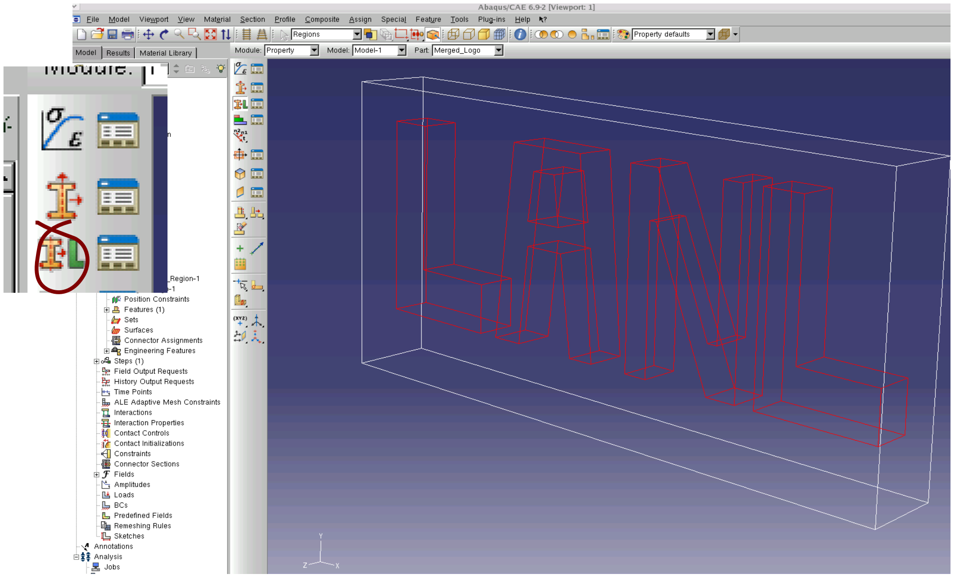
1) From within the 'property' module of the merged part create as many materials as are relevant to the problem. Each material should be given a negative density in g/cc.



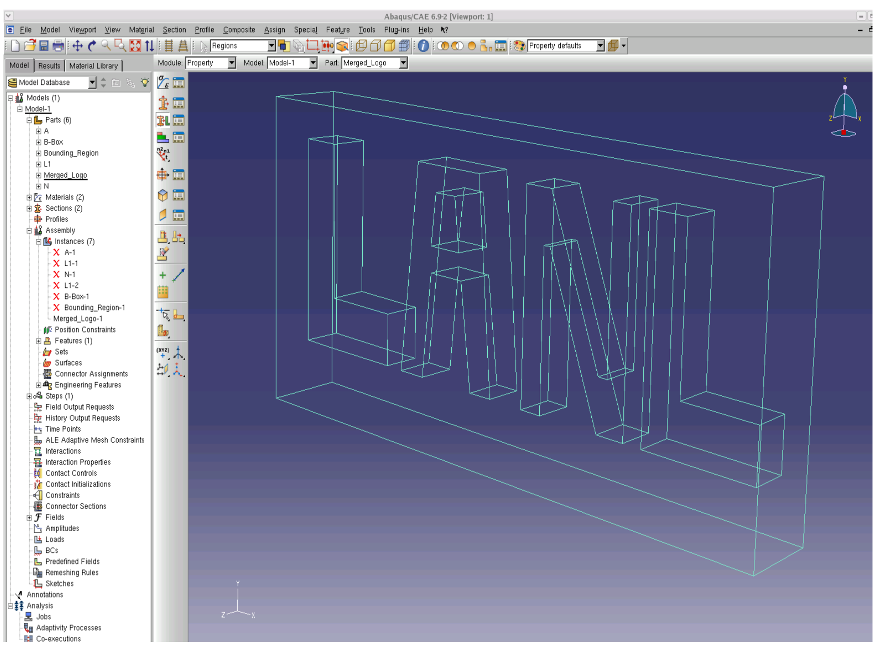


2) Create a 'Section' for each of the conceptual parts of the model. Assign a material to each of the 'Sections'.

3) Assign each of the sections to its associated region of the part.



4) While still in the property mode, once all regions of the part have been assigned a section all of the part will be shown as green.



5) A mesh should be created on the part, which is then now ready to be written to the '.inp' file.