

PADT, Inc. – The Blog WE MAKE INNOVATION WORK

ANSYS Mechanical Scripting: HOWTO Part 3

by **Matt Sutton** 🛗 January 6, 2011 ② 6:34 am

Where We Left Off Last Time...

In the second part of this series I left you with a recipe for discovering ANSYS Mechanical functionality by using entities in the GUI itself to help guide your search through the ANSYS source code that is shipped with every ANSYS Mechanical installation. This technique is the first tool in our toolbox. In this blog post, I'll introduce you to a couple of the internal ANSYS Mechanical data structures that are associated with geometry in particular. I'll simply be introducing them in this blog without any discussion of how one might determine additional functionality associated with these structures. In a future blog post, I'll step you through the use of the most powerful tool available in our toolbox: the Visual Studio Script Debugger.

Introducing the ANSYS Mechanical Geometry Selection Manager and B-Rep Geometry Data Structure

CAD geometry is represented inside a computer using a Boundary Representation (B-Rep) data structure. This data structure falls into a category of data structures computer scientists like to call graphs. In general, a graph is a structure comprised of nodes connected to each other with arcs. A road map is a type of graph where cities represent nodes and the roads connecting them represent arcs. Sites like facebook and linkedin are giant graphs where people are represented as nodes and friendships or acquaintances are represented as arcs. As you can see, graphs are pretty general and you can repl lots of structured relationships with them.

Privacy - Terms

In CAD geometry, the nodes in our graph are comprised of various topological entities like vertices, edges, faces, shells and bodies. The arcs are the incident relationships that exist between these entities. So, for example, an edge will have a start vertex and an end vertex. These relationships can be represented by links between the particular edge and its start vertex or end vertex. Likewise, however, the particular start vertex of one edge may be the end vertex of another edge. So, this relationship can be represented by links between a given vertex and all of the edges to which it is attached. There are similar relationships between edges and faces. A series of edges will form the boundary for a given face and a series of faces might share a common edge. Shells represent a volume that is defined by a set of connected faces. (i.e faces that share common edges) A body is comprised of one or more shells. Why isn't a body the same as a shell? Well, it turns out that bodies with internal holes are more conveniently represented as comprising of two or more shells. One shell represents the outermost envelope and the other shells represent the internal voids. How, then, does one know if they are inside or outside of a body? Well, the faces that comprise a shell each have an associated outward normal vector that points to the "outside" of the body in question. The normal vector sense is defined using an ordering on the bounding edges of a given face. Typically as you traverse the list of edges comprising the boundary of a given face, you are actually walking counter-clockwise around the face. Using this assumption, one can determine an outward normal.

Lets discuss some of the specifics of the ANSYS Mechanical B-Rep. First, an individual B-Rep structure is associated with each part in a Mechanical session. So, if you have 15 parts you will have 15 different B-Rep structures. Second, each entity (vertex, edge, face, body) within a given B-Rep has an associated unique ID. (This ID is called the TopoID, or Topological ID) This ID is just an unsigned integer value, with one small twist. The top two bits of the ID encode the entity's topological type. That is, if you stripped off the top two bits, you would get one of the following four combinations:

Top Two Bits	Decimal Value	Topological Type
00	0	Vertex
01	1	Edge
10	2	Face
11	3	Body

The lower 30 bits actually encode the unique id for a given vertex, edge, face or body. So, what that means is that ANSYS Mechanical is limited to geometry with fewer than roughly one billion vertices, edges, etc... If you have a model with more than that, perhaps a geometry simplification is in order.

Privacy - Terms

To access a given topological entity within a B-Rep object, one uses the following type of function call (*Note, I haven't introduced where the brep_object below comes from... Bear with me, we'll get to it soon.*)

```
var topo object = brep object.Cell(topo id);
```

The Cell function accepts a valid topoID and returns the associated object. The returned object is a rich object and has numerous fields that obviously differ depending on the topological type. So, for example, an edge object will have a StartVertex and EndVertex field that contains references to its start and end vertices in particular. However, all topological objects have one field in common. They all contain an ID field. So the following code would supply the topoID of the start vertex associated with a given edge:

```
var brep_object = /*Get B-Rep from somewhere */
var edge_id = /* Get Edge ID from somewhere*/
var edge = brep_object.Cell(edge_id);
var start_vertex_id = edge.StartVertex.Id;
```

You might gather that one can get the actual start vertex object with a subsequent line like:

```
var start_vertex = brep_object.Cell(start_vertex_id);
```

Now, each vertex object is going to contain a list of edges that are attached to this vertex. So, if we needed all the edges attached to the original edge for some reason, we could use this technique with the start vertex and the end vertex.

Walking the B-Rep data structure is perhaps interesting at best, but it currently doesn't do anything for us. What we need is to be able to select some piece of geometry, or pieces of geometry. Enter the ANSYS Mechanical Selection Manager. This object is closely associated with the B-Rep data structures. Before we discuss this object in some detail, consider starting each and every one of your scripts with the following code:

```
// The script code object is a reference to the javascript engine
// inside the mechanical application. This object has all of the
// that are defined in the ANSYS mechanical javascript files
```

```
var SC = DS.Script;

// The selection manager object is a reference to the ANSYS mechanical
// geometry selection tool. This is a complex object that contains
// references to the currently selected set of geometric entities. Mos
// importantly, it contains back pointers to entity ids within the
// Mechanical BREP structure.
var SM = SC.sm;
```

This code introduces two global variables into your script. The first object contains a reference to the actual javascript engine embedded in the Mechanical application. This object is derived from the DS object, which represents the fundamental Mechanical application object. You can think of the DS object as the mother of all objects. The ScriptCode or SC object stores all of the javascript code that ships with the ANSYS mechanical product. So, if you use the technique described in Part 2 of this series and find a particular function call, you would actually call it against this object. So, for example, we found the function doPrototypeInsertCommandEditor() in Part 2. If we were to actually call that function in our script, it would look like:

```
SC.doPrototypeInsertCommandEditor();
```

The second global variable shown above is the selection manager, or SM. You can see that this object is a child of the script code object. This object has numerous functions and properties and in a later post I hope to document as many as I can. However, a great deal of mileage can be gained by understanding the following two function calls, and the following few properties. I list them below in code form so you can see how they would typically be called inside a script:

```
/* FUNCTIONS OF THE SELECTION MANAGE
// Clear()
// This function resets the selection manager so that no geometry is cur
// selected.
SM.Clear();

// AddToSelection(partID, topoID, bShouldDraw)
// This function adds a particular topological entity (vertex, edge
// to the currently selected set. The first argument specifies the
// part this entity is associated with. You can think of this as relected
```

```
// particular BRep structure. The second argument specifies the topolog
// entities specific ID. The final argument is a boolean specifying whe
// not the selection manager should highlight the selected geometry in c
SM.AddToSelection(partID, topoID, bShouldDraw);
/* PROPERTIES OF THE SELECTION MANAG
// Vertices
// The number of currently selected vertices
var num selected vertices = SM.Vertices;
// Edges
// The number of currently selected edges.
var num selected edges = SM.Edges;
// Faces
// The number of currently selected faces.
var num selected faces = SM.Faces
// Parts
// The number of currently selected (bodies*)
// Note: I think this is where the SM and the BRep use different
// terms for a body. However, I may be wrong.
var num selected bodies = SM.Parts
// SelectedCount
// The total number of selected entities. (Sum of the above)
var num selected = SM.SelectedCount;
// SelectedEntityTopoID(i)
// The ith selected entity's topological id. Note you can use the
// top two bits of this id to determine the type of this selected
// object. i goes from 1 to SelectedCount.
var selected topoid = SM.SelectedEntityTopoID(i);
// SelectedPartID(i)
// The part id for the ith selected entity. For models with numerous
// parts this will tell you which part the currently selected entity is
// associated with. You use this to reference the proper BRep structure
// i goes from 1 to SelectedCount.
var selected partid = SM.SelectedPartID(i);
// PartMgr
// The Part Manager (PartMgr) object is a child of the selection me Privacy-Terms
```

```
// It also has numerous methods and properties, however, the one
// property that I have found most useful is the PartById(id).
// PartMgr.PartById(id)
// This returns the particular part object associated with a given part
var part = SM.PartMgr.PartById(partID);
// The part object returned by the PartMgr.PartById(partID) call has
// associated with it a given BRep data structure. Referencing this bre
// structure is how you get from the selection manager's selected entity
// id to a given piece of topology. So for example, assume the user
// has selected one face in the model. This is one way we can get to the
// face object.
var topo id = SM.SelectedEntityTopoID(1);
var part id = SM.SelectedPartID(1);
var part = SM.PartMgr.PartById(part id);
var brep = part.BRep;
var face = brep.Cell(topo id);
```

The above code represents a small portion of the interface of the ANSYS Mechanical selection manager and also the geometry B-Rep structure. As you can see, the two structures are coupled through the notion of a part id, which selects a given B-Rep structure, and a selected entity id, which represents a piece of topology within the specified B-Rep structure. Once a given topological entity is identified and retrieved from the B-Rep structure, one can navigate the connectivity information contained in the B-Rep to find other topological entities nearby. By using a combination of the Clear() function and the AddToSelection(...) function associated with the selection manager, one can programmatically control which topological entities are currently selected.

You can see this technique in action by perusing the Example in the previous posting.

Privacy - Terms



Related

The 10 Coolest New Features in R15 of ANSYS Mechanical

December 3, 2013 In "The Focus"

An ANSYS Mechanical Scripting Example January 5, 2011

January 5, 2011 In "The Focus"

ANSYS Mechanical Scripting: HOWTO Part 1

November 10, 2010 In "The Focus"

You	must	log	in	to	post a	comment	
100	mast	IUS		ιO	post o		٠.

< An ANSYS Mechanical Scripting Example

New Year, New Articles, And New Features on The Focus Blog >

Contact PADT 1-800-293-PADT & info@padtinc.com

_	_	_	_	_	
C		Λ	п.		ш.
7	г.	Δ	ĸ		п.

Search



LINKS

PADT Website

Contact PADT

PADTMarket.com

Manage PADT Subscriptions

SUBSCRIBE TO OUR BLOG

Enter your email address to subscribe to this blog and receive notifications of new posts by email.

Join 1,804 other subscribers

Email Address

Subscribe

UPCOMMING EVENTS

01/19/2022 - 01/21/2022 Arizona Photonics Days

04/04/2022 - 04/07/2022 37th Space Symposium Arizona Space Industry Booth

04/12/2022 - 04/14/2022 D&M West | MD&M West

3D PRINTING GLOSSARY



INTRODUCTION TO THE ANSYS PARAMETRIC DESIGN LANGUAGE



Learn APDL with this workshop based book written by PADT's Technical Support Team. The new and improved Second Edition contains additional chapters on APDL Math and APDL in ANSYS Mechanical.

Now available on Kindle as well as paperback.

More...

SIMULATION SERVICES

PADT's simulation engineers are true experts in virtual prototyping. Trust the people you come to for ANSYS expertise to handle your simulation outsourcing needs.

Structural, Thermal, Fluid, Electromagnetic, and Systems.



LOOKING FOR ANSYS TRAINING?



Let the people who write "The Focus" train you and your team. Check out our **schedule** or **contact us** to set up a class at your place or something custom.

PODCAST: ALL THINGS ANSYS



ANSYS ACADEMIC PROGRAM



RSS SUBSCRIBE:
RECENT POSTS:
Surprise – 2021 Turned out to Be a Lot Like 2020
All Things Ansys 102: Electronics Reliability Updates in Ansys 2021 R2
Simulating an Electro-Permanent Magnet (EPM) using Ansys Maxwell
Ansys Pro - Premium - Enterprise Electronics Licensing Adjustments
All Things Ansys 101: Additive & Structural Optimization Updates in Ansys 2021 R2
CATEGORIES
Additive Manufacturing
Ansys
ANSYS Discovery
ANSYS Energy Innovation Campaign
ANSYS R 18
Carbon
Education
Events

Privacy - Terms

Flownex

Fun

Contact PADT 1-800-293-PADT & info@padtinc.com ⊠

THINKING ABOUT A BLOG FOR YOUR COMPANY?



Where do you start? How do you keep it going? Where do I get ideas for posts? Should I use humor?

These questions and many others are answered in the book that was inspired by the success of PADT's blog:

Better Blogging for your Business

Available now as a Kindle book or softcover on Amazon.

PADT EMAIL

Manage how PADT Emails You

Want to receive Emails from PADT? Please select any of the basic topics below to tell us what you are interested in. We promise to keep it simple sharing news, opportunities, and useful information. You can come back and change your settings whenever you need to.

Email Email
State/Province
Company
Email Lists
PADT Additive & Advanced Manufacturing Email List
☐ PADT General Information Email List
☐ PADT Product Development and Testing Email List
☐ PADT Simulation Email List
By submitting this form, you are consenting to receive marketing emails from: Phoenix Analysis and Design Technologies, 7755 S. Research Dr., Suite 110, Tempe, AZ, 85284, US, http://www.padtinc.com. You can revoke your consent to receive emails at any time by using the SafeUnsubscribe® link, found at the bottom of every email. Emails are serviced by Constant Contact.
Sign Up!

Privacy - Terms

Contact PADT 1-800-293-PADT & info@padtinc.com ⊠

SEARCH

LINKS

PADT Website

Contact PADT

PADTMarket.com

Manage PADT Subscriptions

ANSYS STARTUP PROGRAM

LOOKING FOR ANSYS TRAINING?



Let the people who write
"The Focus" train you
and your team. Check
out our **schedule** or **contact us** to set up a
class at your place or
something custom.

TRYING TO FIND A COMPUTER BUILT FOR SIMULATION?



Workstations, Servers, and Clusters designed by PADT specifically for

simulation users.

Learn More

INTRODUCTION TO THE ANSYS PARAMETRIC DESIGN LANGUAGE



Learn APDL with this workshop based book written by PADT's Technical Support Team. The new and improved Second Edition contains additional chapters on APDL Math and APDL in ANSYS Mechanical.

Now available on Kindle as well as paperback

More...

RECENT POSTS

Surprise – 2021 Turned out to Be a Lot Like 2020

All Things Ansys 102: Electronics Reliability Updates in Ansys 2021 R2

Simulating an Electro-Permanent Magnet (EPM) using Ansys Maxwell

Ansys Pro - Premium - Enterprise Electronics Licensing Adjustments

All Things Ansys 101: Additive & Structural Optimization Updates in Ansys 2021 R2

CATEGORIES

Additive Manufacturing	
Ansys	
ANSYS Discovery	
ANSYS Energy Innovation Campaign	
ANSYS R 18	
Carbon	
Education	
Events	
Flownex	
Fun	
Getting To Know PADT	
News	
Nimbix	
PADT Medical	
PADT Startup Spotlight	
Podcast	
Product Development	
Publications	
Startups	
Stratasys	
Stratasys Marketing	
The Focus	
Uncategorized	Privacy - Terms

Connect with PADT

© 2021 PADT, Inc. – The Blog • Privacy Policy for the PADT Blog • Powered by WordPress • Theme: Gillian



ü