AUTODESK UNIVERSITY 2013

Beyond Buildings: Advanced FEM Analysis and BIM in the Offshore Industry

Håkon Frank Bardarson – TDA AS

SE2086

This class covers the use of several software products, including Autodesk® Revit® Structure and Autodesk® Robot™ Structural Analysis Professional, for applications in the offshore energy and marine industry rather than the traditional building industry. These structures are often steel dominated and usually require advanced finite element method (FEM) analysis. We show examples from relevant projects and explain the workflow between the different software products used in the projects. We also show how the Robot API (application programing interface) can be useful for improved workflow between various analysis and modeling tools for complete detail engineering. You should be familiar with Robot, or another similar FEM analysis tool, and be open to API programming.

Learning Objectives

At the end of this class, you will be able to:

- Describe the usefulness of Revit Structure and Robot for special, non-standard structures
- List and define the possibilities of the Robot API
- Use different modeling tools to push the perceived boundaries
- Use the Revit Structure-Robot link and describe its limitations

About the Speaker

Hakon is a structural engineer with a Master degree from the University of Washington. He works as a structural engineer as well as the Autodesk product manager at TDA, a Norwegian engineering company serving the offshore industry with advanced engineering services for more than 30 years. He has been involved in various projects during his 14 years of working as a structural engineer ranging from sport stadium, hydropower plants, buildings, damns, bridges to fan wheels and lifting equipment.
At TDA he has taken the Autodesk BIM solution beyond mere buildings into a field with complex structures and demands for advanced FEM analysis. As the Autodesk product manager for a Norwegian Autodesk dealer he is responsible for classes and support for Autodesk® Revit® and Autodesk® Robot™ Structural Analysis Professional.

hfb@tda.as

Non standard structures in Revit and Robot

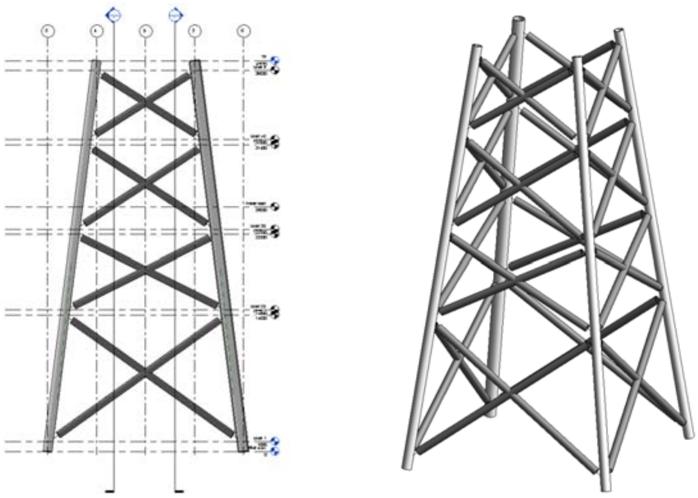
Modeling structures that are not typical buildings with straight vertical walls and horizontal floors is possible to make in to do in both Revit and Robot Structural Analysis even though at first glance both programs are very building focused.

Each program has his advantages and weaknesses and being able to move the model between different modeling tools is becoming more and more feasible as the programs are developed with this in mind. Sometimes a little bit of creativity is all that is needed to make the model.

What we look at here is a jacket foundation for an offshore wind turbine. These structures are made with steel profiles. Typically three or four legs with cross braces that transfer the horizontal load down to the seabed. Top piece of the jacket is a transition piece and on top of that is the tower holding the turbine.

Pictures from Ormonde Offshore Windfarm out from the Irish coast

Typically the jackets are many and the area that they are placed in has various depth and ground conditions. A few models are made with different heights and footprints. Revit is a good tool to start modeling this as using grid lines and levels it is easy to change the geometry and therefor easy to make many models from one base model. The braces can be connected to the legs and refer to the levels so changing levels and or footprint is easy and requires minimal remodeling.



Revit model of the jacket part

Revit to Robot link

Moving the model then to Robot is straight forward with the link inside Revit. The link will transfer the analytical model of standard beam, column, brace elements as well as wall and floor elements from the Revit model to Robot. Even though we can do all kinds of custom made objects in Revit we can not transfer them over to Robot unless they are based on the standard elements that contain the analytical lines or plans.

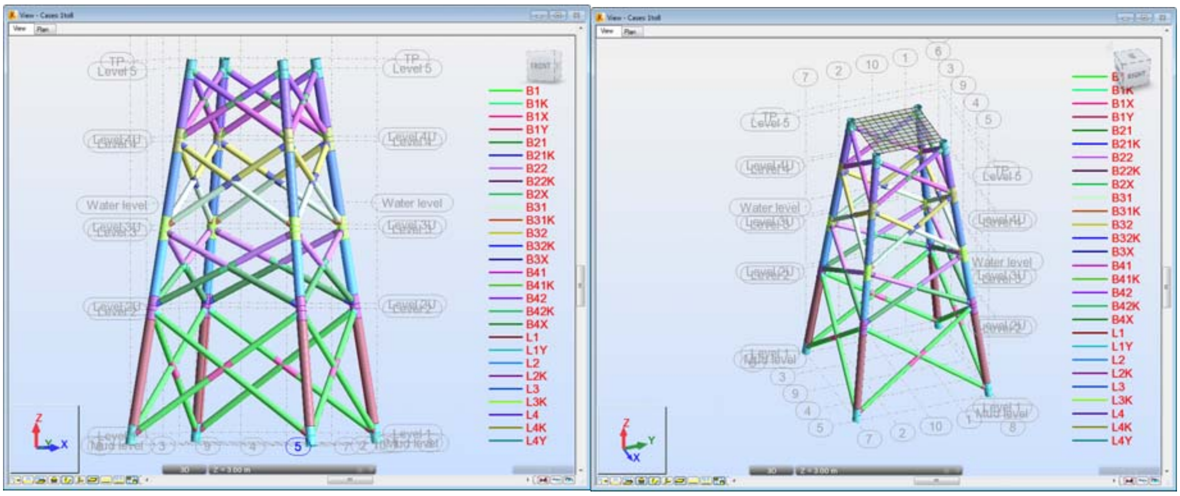
Normally a fair amount of work is needed to adjust the analytical model before it is ready to be transferred into Robot. Autodesk has made huge improvements in the latest versions of Revit on how we can manipulate the analytical model in Revit independently of the physical model. For a simple structure like we have here there is not much we need to do to the analytical model but it is necessary to view the analytical model to check if that is how we want our model to be as that is the model we are going to be using.

For these structures we are not using the Robot Structural Analysis to do any structural calculations because ANSYS ASAS does offer some features that Robot does not have and are needed for these structures. Two of these features are fatigue calculation and stochastic wave loading.

ANSYS ASAS software is a general-purpose structural finite element system containing industry-specific features to address the needs of offshore and marine engineers. ANSYS ASAS provides the capabilities required to perform global structural assessment of most types of marine structures, including jackets. ANSYS ASAS provides code checks as well as fatigue calculations for spectra, deterministic and time history.

ANSYS ASAS is not a modeling tool and normally the models are typically made in ANSYS Direct Modeling tool, ANSYS Structural Mechanical or FEMGV. The input in ANSYS ASAS is in a text file format and therefor it is easy to make them from tabulated data. Before using the Robot API to make the input files the standard method was to copy the tables out from Robot to excel and make the text files from them.

We use Robot here to prepare the model for ANSYS ASAS by splitting the elements at required locations naming the cross sections in a manner that is easy to size after and grouping different elements by using the bar type properties and assigning them to the bars.



Robot model after splitting the members and naming them.

What can we use the Robot API to do?

The Robot API (Application Programming Interface) is basically how we can use other programs to use the features and functions in Robot. The Robot API can for example be used for:

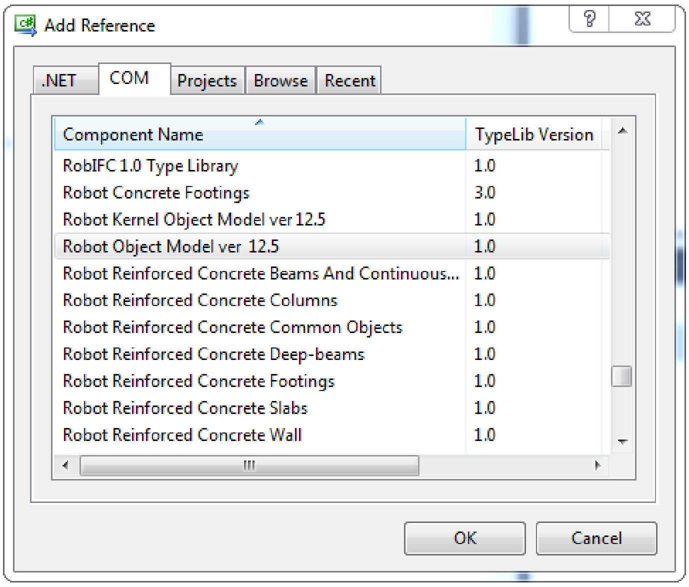
- Automating repetitive tasks in Robot program
- Creating applications that facilitate modeling of typical structures
- Analyzing projects, creating own reports and printouts
- Importing external data to Robot program
- Integrating other programs with Robot program
- Interpreting results and creating documentation

To use Robot API one does not need to be a programming expert. It is more important to know Robot and what you can do in Robot. Most operations that you can do via the user interface you can also do via the API.

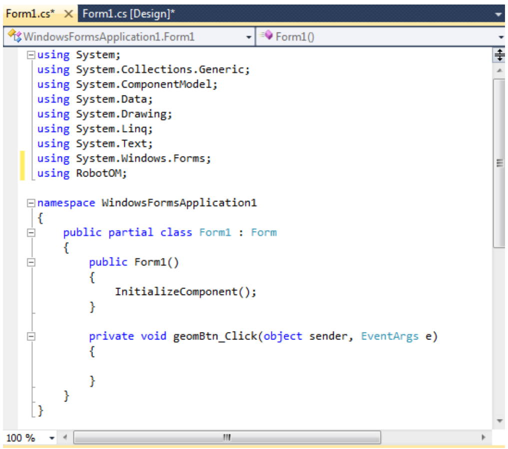
We use Microsoft Visual Studio Express 2010 to program a simple program that reads the data we want to use from the Robot model and writes the appropriate input files for ANYSIS ASAS. These input files are simple text files with a certain format that can be read into the program.

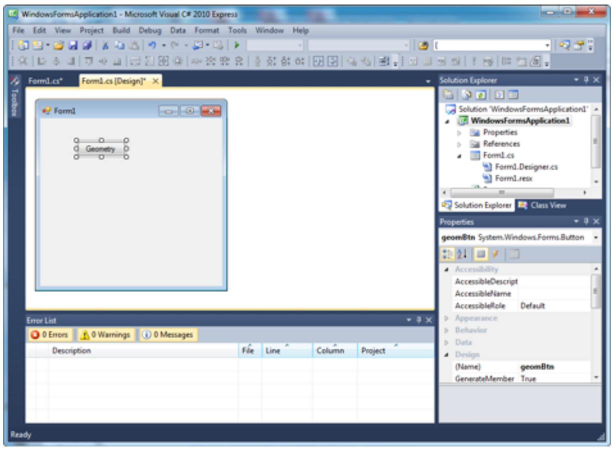
Microsoft Visual Studio Express is a free software that can be downloaded from the Microsoft website and can be used to program in several languages. To get started programming for Robot API a few simple actions need to be done.

First of all you need a copy of Robot on your computer. Then when you have defined a new project in Visual Studio you need to go to Project -> Add Reference and in the COM tap selecting the Robot Object Model.



Then you need to write using Robot OM; in the top of your program file:





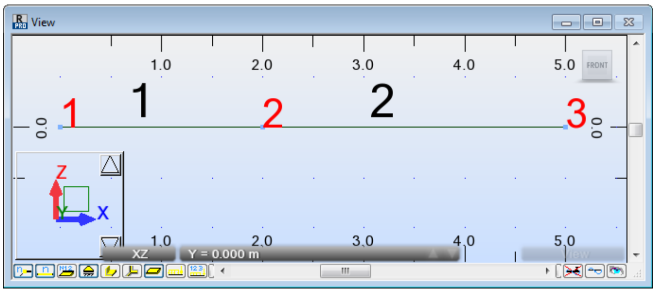
When this is done you can start writing the code that interacts with Robot. Writing code in Visual Studio is very easy and often you only need to write the first letter and then you can scroll up and down to find the correct variable or method that your are looking for.

The internet is full of information on how to write code in different programming languages, and if you have once learned one language it is very easy to start using other languages even though there have been a few years since you last wrote some code. The access to the information that you need to get started is just a click in the browser away.

To get you started here is a simple method that activates when pressing the geomBtn button and makes a new 2D frame project in Robot then creates 3 nodes and 2 bar elements between those nodes.

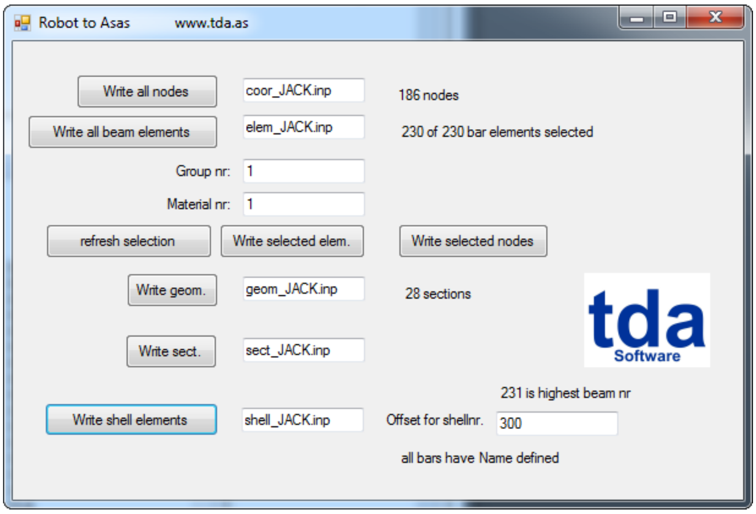
```
private void geomBtn_Click(object sender, EventArgs e)
{
    //defining a robot class in this method
    IRobotApplication robApp;
    // initiating a robot class
    robApp = new RobotApplication();
    // setting Robot visible and allow user interaction
    robApp.Interactive = 1;
    robApp.Visible = 1;
    //create a new project in 2 D frame
    robApp.Project.New(IRobotProjectType.I_PT_FRAME_2D);
    //generate a structure
    IRobotStructure str;
    str = robApp.Project.Structure;
    //generate some nodes
    str.Nodes.Create(1, 0, 0, 0);
    str.Nodes.Create(2, 2, 0, 0);
    str.Nodes.Create(3, 5, 0, 0);
    //generate some bars
    for (int i = 1; i < 3; i++)
    {
        str.Bars.Create(i, i, i + 1);
    }

    //end of actions for geomBtn_Click method
}
```

And this is how it looks in Robot after running the program:



If you install the Robot SDK (Software Development Kit) you will see a description of the variables and methods in the Robot API as well as some code examples that can also be helpful in getting you started. The Robot Software Development Kit can be installed from your product installation menu under Install Tools & Utilities.

As I mentioned earlier we used Microsoft Visual studio to write a small program that uses the Robot API to write the input files used in ANSYS ASAS. The program we wrote can write out five different files, which the user can define the name for. It can either write out all of the nodes/beam elements or the nodes/beam elements selected in Robot.



The format of the files is that one line typically defines an element and text after the * is a comment, therefore ignored by ANSYS ASAS. The first file is the node file which includes the node number, and coordinates in x, y and z.

coor_JACK.inp - Notepad

```
*2013-11-13
*coor
1 0.000 20.000 0.000
2 5.000 15.000 40.000
3 0.000 0.000 0.000
4 5.000 5.000 40.000
5 20.000 0.000 0.000
6 15.000 5.000 40.000
7 20.000 20.000 0.000
8 15.000 15.000 40.000
9 19.875 0.125 1.000
10 1.750 1.750 14.000
11 0.125 0.125 1.000
12 18.250 1.750 14.000
13 18.188 1.813 14.500
14 2.775 2.775 22.200
15 1.813 1.813 14.500
16 17.225 2.775 22.200
17 17.163 2.838 22.700
18 3.925 3.925 31.400
19 2.838 2.838 22.700
20 16.075 3.925 31.400
21 16.013 3.988 31.900
22 4.875 4.875 39.000
23 3.988 3.988 31.900
24 15.125 4.875 39.000
25 19.875 19.875 1.000
26 18.250 18.250 14.000
27 18.188 18.188 14.500
28 17.225 17.225 22.200
29 17.163 17.163 22.700
30 16.075 16.075 31.400
31 16.013 16.013 31.900
32 15.125 15.125 39.000
33 0.125 19.875 1.000
34 1.750 18.250 14.000
35 1.813 18.188 14.500
```

The second one is the bar element file which sorts the bars after the type parameter and includes the nodes that the bar connects to as well as the bar number and cross-section number. The text after the * is just a comment.

elem_JACK.inp - Notepad

```
*2013-11-13
grou 1
matp 1
tube 11 81 1 1 * L1Y Legs
tube 15 88 2 2 * L2K Legs
tube 9 82 1 3 * L1Y Legs
tube 13 92 2 4 * L2K Legs
tube 88 94 3 5 * L2 Legs
tube 25 83 1 7 * L1Y Legs
tube 27 90 2 8 * L2K Legs
tube 33 84 1 9 * L1Y Legs
tube 35 86 2 10 * L2K Legs
tube 84 85 5 11 * L1 Legs
tube 85 34 2 12 * L2K Legs
tube 37 98 6 15 * L3K Legs
tube 81 87 5 16 * L1 Legs
tube 87 10 2 17 * L2K Legs
tube 90 95 3 18 * L2 Legs
tube 83 89 5 19 * L1 Legs
tube 89 26 2 20 * L2K Legs
tube 92 96 3 23 * L2 Legs
tube 82 91 5 24 * L1 Legs
tube 91 12 2 25 * L2K Legs
tube 86 97 3 26 * L2 Legs
```

elem_JACK.inp - Notepad

```
tube 111 24 12 134 * L4Y Legs
tube 32 8 12 135 * L4Y Legs
tube 5 9 1 136 * L1Y Legs
tube 112 40 12 137 * L4Y Legs
tube 12 13 2 138 * L2K Legs
tube 113 22 12 139 * L4Y Legs
tube 16 17 6 140 * L3K Legs
tube 20 21 9 142 * L4K Legs
tube 24 6 12 144 * L4Y Legs
grou 2
matp 1
tube 11 45 4 6 * B1Y Brace
tube 9 43 4 13 * B1Y Brace
tube 25 42 4 14 * B1Y Brace
tube 25 48 4 21 * B1Y Brace
tube 46 58 7 22 * B1 Brace
tube 11 93 4 27 * B1Y Brace
tube 93 44 4 28 * B1Y Brace
tube 33 46 4 29 * B1Y Brace
tube 33 47 4 30 * B1Y Brace
tube 9 41 4 37 * B1Y Brace
tube 48 57 7 38 * B1 Brace
tube 41 60 7 40 * B1 Brace
tube 44 59 7 44 * B1 Brace
```

The third file is the file which includes the cross section number, the cross section name and the beta angle of the cross section.

geom_JACK.inp - Notepad

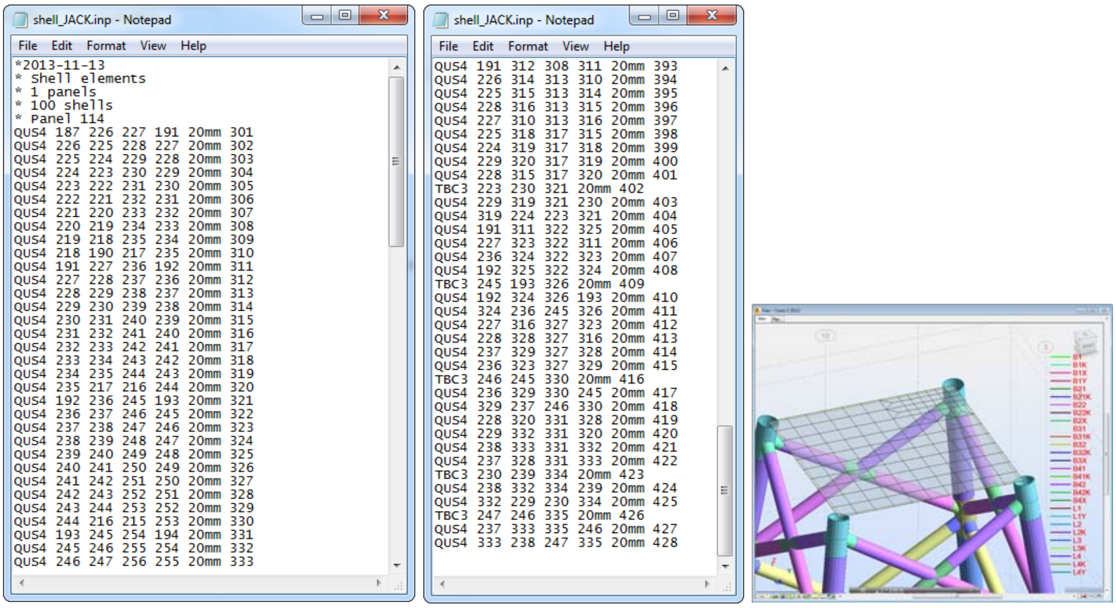
```
*2013-11-13
1 tube L1Y beta 0.0
2 tube L2K beta 0.0
3 tube L2 beta 0.0
4 tube B1Y beta 0.0
5 tube L1 beta 0.0
6 tube L3K beta 0.0
7 tube B1 beta 0.0
8 tube L3 beta 0.0
9 tube L4K beta 0.0
10 tube B1K beta 0.0
11 tube B21K beta 0.0
12 tube L4Y beta 0.0
13 tube L4 beta 0.0
14 tube B1X beta 0.0
15 tube B21 beta 0.0
16 tube B22 beta 0.0
17 tube B32 beta 0.0
18 tube B31K beta 0.0
19 tube B3X beta 0.0
20 tube B31 beta 0.0
21 tube B41K beta 0.0
22 tube B41 beta 0.0
23 tube B42 beta 0.0
24 tube B4X beta 0.0
25 tube B2X beta 0.0
26 tube B22K beta 0.0
27 tube B32K beta 0.0
28 tube B42K beta 0.0
```

The fourth file is the section file which includes information about the cross sectional properties of the pipes Outer diameter and thickness. This we change manually when running the code checks and fatigue calculations.

sect_JACK.inp - Notepad

```
*2013-11-13
L1Y tub xsec D0 t0
L2K tub xsec D1 t1
L2 tub xsec D2 t2
B1Y tub xsec D3 t3
L1 tub xsec D4 t4
L3K tub xsec D5 t5
B1 tub xsec D6 t6
L3 tub xsec D7 t7
L4K tub xsec D8 t8
B1K tub xsec D9 t9
B21K tub xsec D10 t10
L4Y tub xsec D11 t11
L4 tub xsec D12 t12
B1X tub xsec D13 t13
B21 tub xsec D14 t14
B22 tub xsec D15 t15
B32 tub xsec D16 t16
B31K tub xsec D17 t17
B3X tub xsec D18 t18
B31 tub xsec D19 t19
B41K tub xsec D20 t20
B41 tub xsec D21 t21
B42 tub xsec D22 t22
B4X tub xsec D23 t23
B2X tub xsec D24 t24
B22K tub xsec D25 t25
B32K tub xsec D26 t26
B42K tub xsec D27 t27
```

The fifth file is a file to write out the planar finite elements with nodal numbers, cross section name and element number.



```
*2013-11-13
* shell elements
* 1 panels
* 100 shells
* Panel 114
QUS4 187 226 227 191 20mm 301
QUS4 226 225 228 227 20mm 302
QUS4 225 224 229 228 20mm 303
QUS4 224 223 230 229 20mm 304
QUS4 223 222 231 230 20mm 305
QUS4 222 221 232 231 20mm 306
QUS4 221 220 233 232 20mm 307
QUS4 220 219 234 233 20mm 308
QUS4 219 218 235 234 20mm 309
QUS4 218 190 217 235 20mm 310
QUS4 191 227 236 192 20mm 311
QUS4 227 228 237 236 20mm 312
QUS4 228 229 238 237 20mm 313
QUS4 229 230 239 238 20mm 314
QUS4 230 231 240 239 20mm 315
QUS4 231 232 241 240 20mm 316
QUS4 232 233 242 241 20mm 317
QUS4 233 234 243 242 20mm 318
QUS4 234 235 244 243 20mm 319
QUS4 235 217 216 244 20mm 320
QUS4 192 236 245 193 20mm 321
QUS4 236 237 246 245 20mm 322
QUS4 237 238 247 246 20mm 323
QUS4 238 239 248 247 20mm 324
QUS4 239 240 249 248 20mm 325
QUS4 240 241 250 249 20mm 326
QUS4 241 242 251 250 20mm 327
QUS4 242 243 252 251 20mm 328
QUS4 243 244 253 252 20mm 329
QUS4 244 216 215 253 20mm 330
QUS4 193 245 254 194 20mm 331
QUS4 245 246 255 254 20mm 332
QUS4 246 247 256 255 20mm 333
```

```
QUS4 191 312 308 311 20mm 393
QUS4 226 314 313 310 20mm 394
QUS4 225 315 313 314 20mm 395
QUS4 228 316 313 315 20mm 396
QUS4 227 310 313 316 20mm 397
QUS4 225 318 317 315 20mm 398
QUS4 224 319 317 318 20mm 399
QUS4 229 320 317 319 20mm 400
QUS4 228 315 317 320 20mm 401
TBC3 223 230 321 20mm 402
QUS4 229 319 321 230 20mm 403
QUS4 319 224 223 321 20mm 404
QUS4 191 311 322 325 20mm 405
QUS4 227 323 322 311 20mm 406
QUS4 236 324 322 323 20mm 407
QUS4 192 325 322 324 20mm 408
TBC3 245 193 326 20mm 409
QUS4 192 324 326 193 20mm 410
QUS4 324 236 245 326 20mm 411
QUS4 227 316 327 323 20mm 412
QUS4 228 328 327 316 20mm 413
QUS4 237 329 327 328 20mm 414
QUS4 236 323 327 329 20mm 415
TBC3 246 245 330 20mm 416
QUS4 236 329 330 245 20mm 417
QUS4 329 237 246 330 20mm 418
QUS4 228 320 331 328 20mm 419
QUS4 229 332 331 320 20mm 420
QUS4 238 333 331 332 20mm 421
QUS4 237 328 331 333 20mm 422
TBC3 230 239 334 20mm 423
QUS4 238 332 334 239 20mm 424
QUS4 332 229 230 334 20mm 425
TBC3 247 246 335 20mm 426
QUS4 237 333 335 246 20mm 427
QUS4 333 238 247 335 20mm 428
```

Here the program can write out both 3 noded or 4 noded shell elements with the appropriate definition for ANSYS ASAS.

Pushing the boundary of the tools

When we use a certain modeling tool we are in some way limited to the options available in that program. By being able to use different modeling tools and moving the models between them we are able to use the strengths of one program to make for example the geometry and then use another program to perform the calculations. In the previous example we used Revit and Robot to make the geometric model while doing the finite element calculation in ANSYS programs.

Another example we can demonstrate is a finite element modeling of ship geometry for doing strength and lifting analysis in Robot Structural Analysis. Robot does not offer the option of modeling surfaces that have curvature in two plans like a typical ship structure has. But Robot can mesh such surfaces if they can be imported into Robot. Making the surfaces in another modeling tool such as ANSYS SpaceClaim Direct Modeler and then importing them to Robot makes it possible to do the analyses and code checks in Robot.