SESAM TUTORIAL

Spar-type Floating Offshore Wind

Parametric Modelling of Spar for GeniE and HydroD Workflow

Sesam Tutorial

Spar-type Floating Offshore Wind: Parametric Modelling of Spar for GeniE and HydroD Workflow

Date: April 2025

Valid from GeniE version 8.12, HydroD 7.2, Postresp 7.2.3, and Sesam Manager 6.7.

Prepared by: Digital Solutions at DNV

E-mail support: software.support@dnv.com

E-mail sales: digital@dnv.com

Table of contents

INTRODUCTION........ 1
Regarding JS code blocks 2

1 PANEL MODEL CREATION IN GENIE........ 3
1.1 Panel Model – Manual Modelling 3
1.1.1 Creating a new workspace 3
1.1.2 Creating properties 4
1.1.3 Creating guiding points 5
1.1.4 Creating the outer hull (side wall) 6
1.1.5 Creating the outer hull (bottom plate) 7
1.1.6 Applying mesh property 8
1.1.7 Applying wet surface property 8
1.1.8 Adding the plates to a named set 9
1.1.9 Creating analysis to generate panel model mesh 9
1.1.10 Creating a dummy hydro load case 10
1.1.11 Generating and exporting the panel model mesh 11
1.2 Panel Model – Automated Modelling 13
1.2.1 Design parameters 13
1.2.2 Defining user defined parameters 13
1.2.3 Pre-calculating variables 14
1.2.4 Modifying the manual modelling scripts 14
1.2.5 Running the parametric script and checking the panel model 15

2 HYDRODYNAMIC ANALYSIS IN HYDROD 17
2.1 Hydrodynamic Analysis – Manual Setup 17
2.1.1 Creating a new workspace 17
2.1.2 Creating a Wadam wizard 17
2.1.3 Defining environmental data 18
2.1.4 Defining a hydro model 20
2.1.5 Creating and executing Wadam analysis 22
2.1.6 Checking the results with Postresp 24
2.2 Hydrodynamic Analysis – Automated Setup 27
2.2.1 Design parameters 27
2.2.2 Defining user defined parameters 27
2.2.3 Pre-calculating variables 27
2.2.4 Modifying the manual setup scripts 27
2.2.5 Running the parametric script and checking the results 28

3 AUTOMATING GENIE AND HYDROD WITH SESAM MANAGER 29
3.1 Preparing the Parametric Scripts 30
3.1.1 Preparing the GeniE script 30
3.1.2 Preparing the HydroD script 30
3.1.3 Preparing the Postresp script 31
3.1.4 Preparing Sesam Manager Post Execute Script for Postresp 31
3.2 Setting Up Sesam Manager 32
3.2.1 Creating a new Job 32
3.2.2 Copying the parametric modelling scripts 32
3.2.3 Creating a Sequence folder 32
3.2.4 Creating a GeniE activity 33
3.2.5 Creating a HydroD activity 34
3.2.6 Creating a Postresp activity 35
3.2.7 Creating the rest of the Job 36
3.2.8 Running all analyses 37

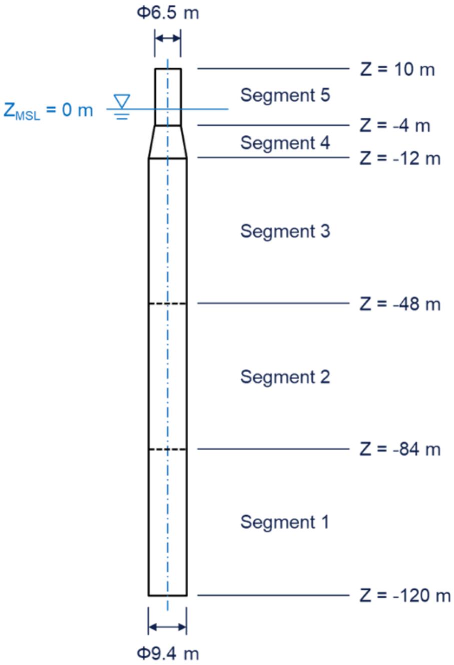
INTRODUCTION

This tutorial demonstrates how to create a spar-type floating offshore wind (FOW) floater and automate the process using GeniE's and HydroD's parametric modelling capabilities. The goal is to introduce a straightforward approach for converting a model into parametric scripts, enabling parametric studies for hydrodynamic analysis. The resulting data can then be used for motion analysis in Sima or Bladed.

Before starting this tutorial, you should already be comfortable working with GeniE and HydroD.

The approach presented here is intentionally simple: we will modify portions of the scripts that are automatically generated during manual modelling. However, note that there is no single, definitive way to perform parametric modelling. Many alternative approaches may also be valid—and in some cases, more efficient—than the one described here.

For this tutorial, the design is based on the spar-type floater shown below:



The tutorial is organized into three chapters:

1. **Panel Model Creation in GeniE** – Building the floater hull's panel model and exporting it for use in HydroD.
2. **Hydrodynamic Analysis in HydroD** – Importing the panel model from GeniE, setting up relevant parameters, and performing frequency-domain hydrodynamic analysis.
3. **Automation with Sesam Manager** – Combining GeniE and HydroD operations into a single automated workflow using Sesam Manager.

Each of the first two chapters is further divided into two sub-chapters:

1. **Manual Modelling** – Step-by-step instructions to manually set up the model and workspace.
2. **Automated Modelling** – Using JS scripts, recorded during the manual steps, to create parametric modelling scripts.

Disclaimer:

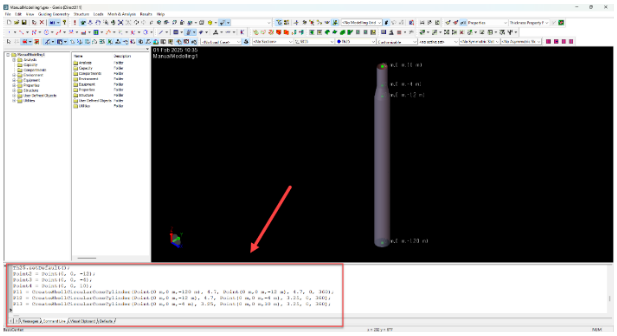
This tutorial is intended solely for training purposes to demonstrate how to use Sesam for automating the modelling of a FOW floater. It is not intended for use in real-world projects or decision-making. Users are responsible for validating all parameters and ensuring compliance with project-specific requirements and standards.

Regarding JS code blocks

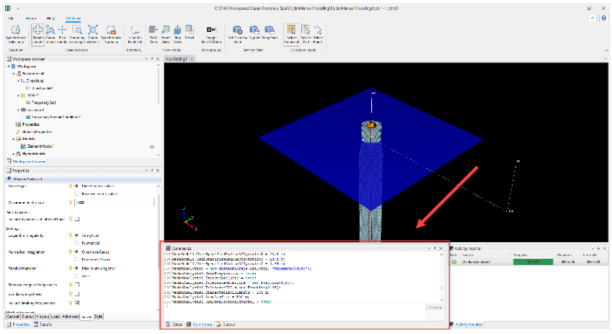
Each manual modelling step is accompanied by its corresponding JS script. For example:

```
Pl1 = CreateShellCircularConeCylinder(Point(0 m,0 m,-120 m), 4.7, Point(0 m,0 m,-12 m), 4.7, 0, 360);
Pl2 = CreateShellCircularConeCylinder(Point(0 m,0 m,-12 m), 4.7, Point(0 m,0 m,-4 m), 3.25, 0, 360);
Pl3 = CreateShellCircularConeCylinder(Point(0 m,0 m,-4 m), 3.25, Point(0 m,0 m,10 m), 3.25, 0, 360);
```

These JS scripts are automatically logged in the Command Line window of GeniE and HydroD when you perform modelling operations as shown below. Pay close attention to them, as they will be used in the Automated Modelling sub-chapter to build your parametric scripts.



GeniE



HydroD

You can also find the complete JS scripts after your manual modelling in the GeniE or HydroD workspace folders:

- If the workspace has been saved, the file name will be named **<workspace_name>.js**.
- If the workspace has not been saved, the file name will be **<workspace_name>_<datetime>.js.**

Note: The logged JS commands may vary depending on the order of operations or if any steps are skipped. Rather than copying them exactly, focus on understanding the core logic—this tutorial is about learning to build your own automation functions.

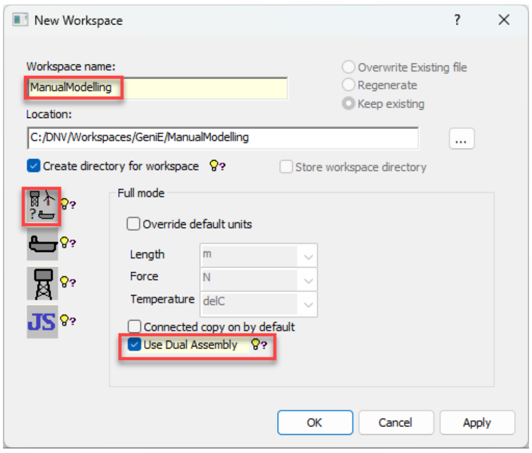
1 PANEL MODEL CREATION IN GENIE

1.1 Panel Model – Manual Modelling

In this first chapter, we will create the outer hull of the floater in GeniE. From the outer hull, we will export the panel model file (Panel_T1.FEM) to be used in the frequency-domain hydrodynamic analysis in HydroD.

1.1.1 Creating a new workspace

Create a new **GeniE** workspace with **Full Mode** and **Use Dual Assembly** selected.



The following JS commands will be logged in the **Command Line** window of GeniE (by default, located at the bottom of the screen):

```
GenieRules.Compatibility.version = "V8.12-03";
GenieRules.Tolerances.useTolerantModelling = true;
GenieRules.Tolerances.angleTolerance = 2 deg;
GenieRules.Meshing.autoSimplifyTopology = true;
GenieRules.Meshing.eliminateInternalEdges = true;
GenieRules.BeamCreation.DefaultCurveOffset = ReparameterizedBeamCurveOffset();
GenieRules.Geometry.AssemblyType = DualAssembly;
GenieRules.Transformation.DefaultConnectedCopy = false;
```

Note: The dual assembly format is recommended when modelling a floater substructure that combines shell and beam elements. Although this tutorial does not require beam elements for the panel model, they may be needed when developing a structural model later.

If you mistakenly create the workspace without selecting Use Dual Assembly, you can change the workspace type afterward using the following JS command:

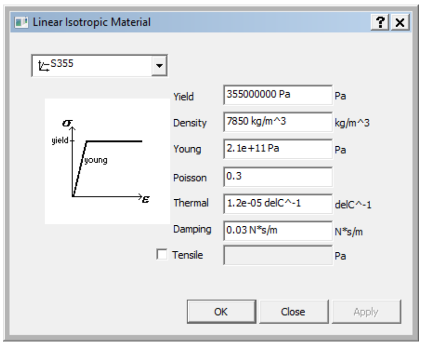
```
GenieRules.Geometry.AssemblyType = DualAssembly;
```

Note: If you are unable to copy text from this PDF, a copy of the JS script can be found in the **input_files** folder under: **Chapter 1\Ch1-1_PanelManualModelling.js**.

1.1.2 Creating properties

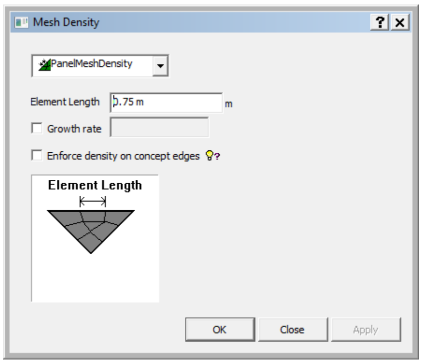
Create the following properties:

1. Create **material** property



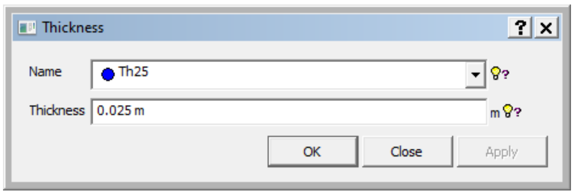

```
S355 = MaterialLinear(355000000 Pa, 7850 kg/m^3, 2.1e+11 Pa, 0.3, 1.2e-05 delC^-1, 0.03
N*s/m);
```

2. Create **mesh density** property



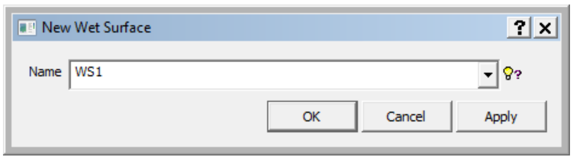

```
PanelMeshDensity = MeshDensity(0.75);
```

3. Create **thickness** property




```
Th25 = Thickness(0.025 m);
```

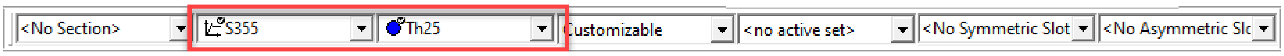
4. Create **wet surface** property




```
WS1 = WetSurface();
```

Hint: Right-click the appropriate folder in the browser and select New <PropertyType>. For example, to create a new material property, right-click the **Properties > Materials** folder and select **New Material**.

Set **default material** to ***S355*** and **default thickness** to ***Th25***.




```
S355.setDefault();
Th25.setDefault();
```

1.1.3 Creating guiding points

Create the following guide points with **Guiding Geometry > Guide Point Dialog**:

- **0 0 -120 m**
- **0 0 -12 m**
- **0 0 -4 m**
- **0 0 10 m**

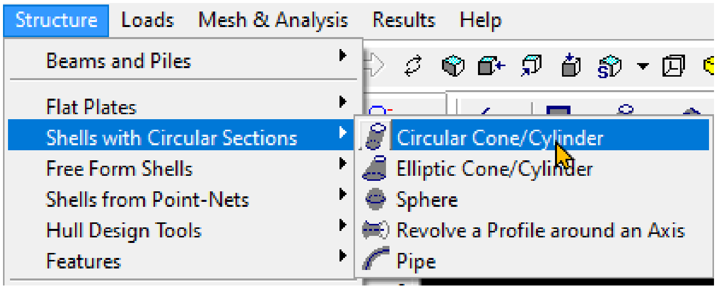

```
Point1 = Point(0, 0, -120);
Point2 = Point(0, 0, -12);
Point3 = Point(0, 0, -4);
Point4 = Point(0, 0, 10);
```

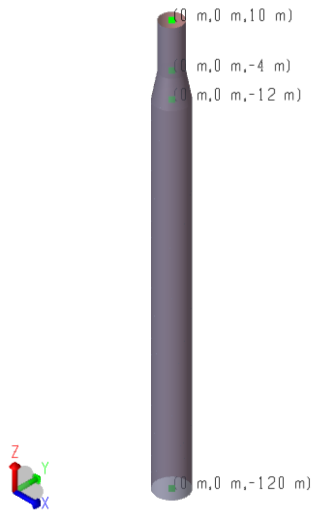
1.1.4 Creating the outer hull (side wall)

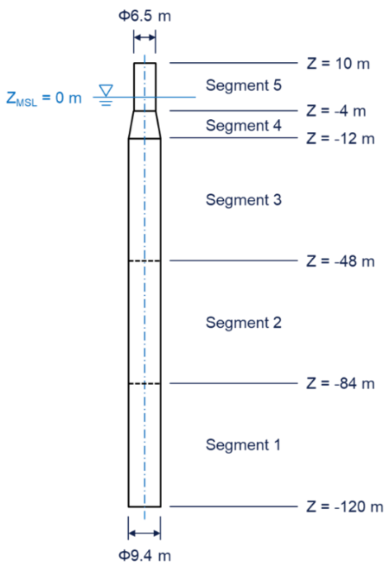
Activate **Reference Point Modelling** by clicking the icon on the toolbar. This enables command logging to refer to the object names rather than their specific values.

Create outer hull shell using **Structure > Shells with Circular Sections > Circular Cone/Cylinder**. When prompted, specify the following inputs in order:

1. Starting point
2. Radius at the starting point
3. Finish point
4. Radius at the finish point








```
Pl1 = CreateShellCircularConeCylinder(Point1, 4.7, Point2, 4.7, 0, 360);
Pl2 = CreateShellCircularConeCylinder(Point2, 4.7, Point3, 3.25, 0, 360);
Pl3 = CreateShellCircularConeCylinder(Point3, 3.25, Point4, 3.25, 0, 360);
```

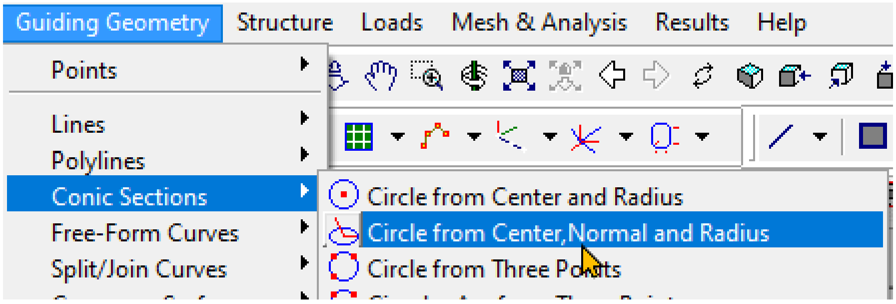
Note: If you did not activate the **Reference Point Modelling**, the logged commands will use explicit values like `Point(0 m,0 m,-120 m)` instead of referencing the object name, such as `Point1`.

1.1.5 Creating the outer hull (bottom plate)

Create a circle guiding curve with **Guiding Geometry > Conic Sections > Circle from Center, Normal and Radius**.

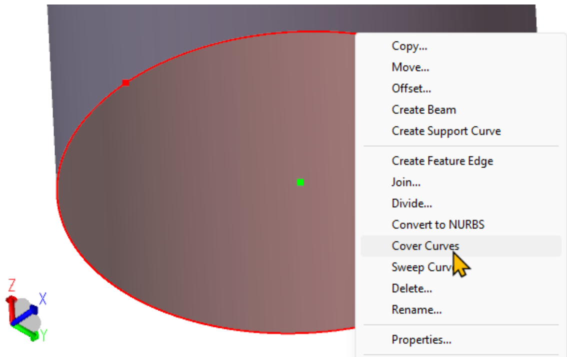
Click the **bottom guide point**, then the **top guide point** to define the normal vector.

Finally, enter the radius: **4.7 m**.



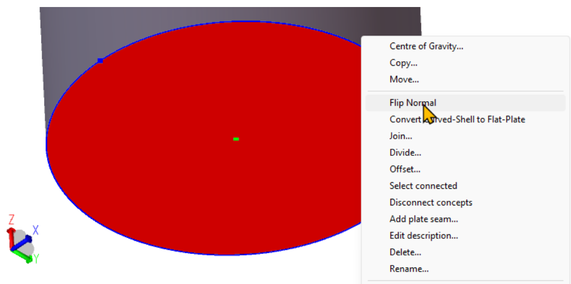

```
Curve1 = CreateCircleFromPlaneAndRadius(Point1, Vector3d(0 m,0 m,130 m), 4.7);
```

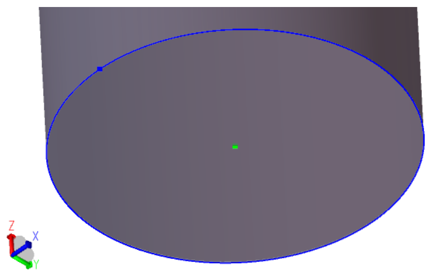
Select the circular guide curve, right-click it, and choose **Cover Curves** to create the bottom plate. Note that the *bottom* surface (displayed in red) is facing outward.




```
Pl4 = CoverCurves(Curve1);
```

Select the bottom plate, right-click it, and choose **Flip Normal**. This sets the *front* surface (shown in grey) as the wetted surface.

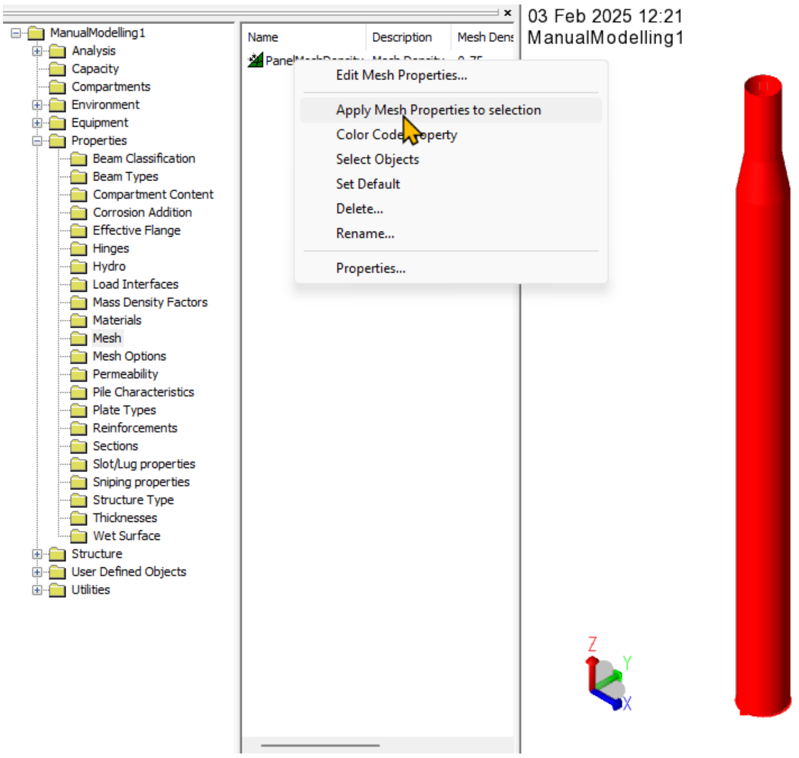




```
Pl4.flipNormal();
```

1.1.6 Applying mesh property

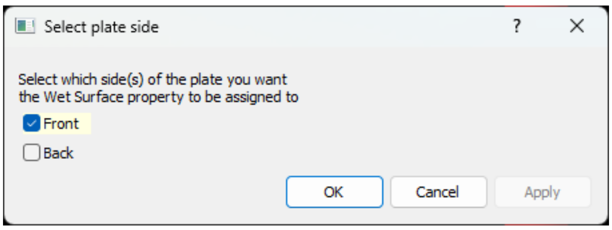
Select all the outer hull plates and apply the mesh property **PanelMeshDensity** as follows: Navigate to **Properties > Mesh**, right-click **PanelMeshDensity**, and choose **Apply Mesh Properties to Selection** while the plates are selected.




```
Pl1.meshDensity = PanelMeshDensity;
Pl2.meshDensity = PanelMeshDensity;
Pl3.meshDensity = PanelMeshDensity;
Pl4.meshDensity = PanelMeshDensity;
```

1.1.7 Applying wet surface property

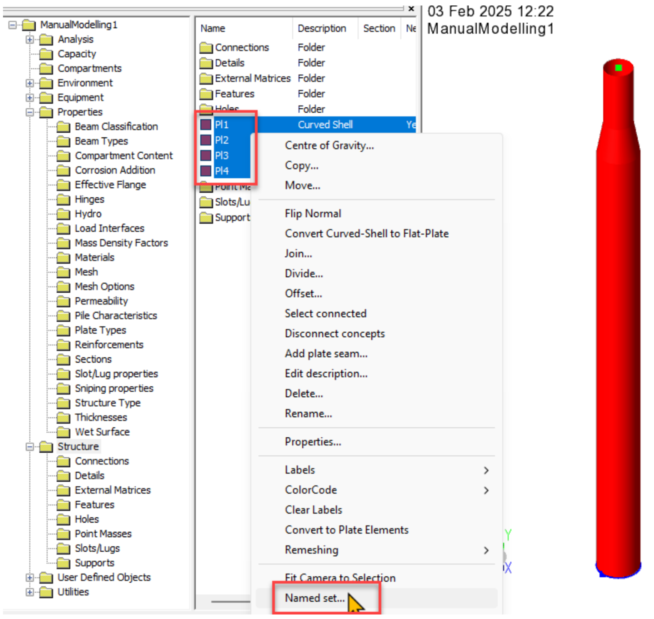
Apply the wet surface property **WS1** to the front side of all plates: Navigate to **Properties > Wet Surface**, right-click **WS1**, and select **Apply Wet Surface Properties to Selection** while the plates are selected. In the dialog, choose **Front**, then click **OK**.

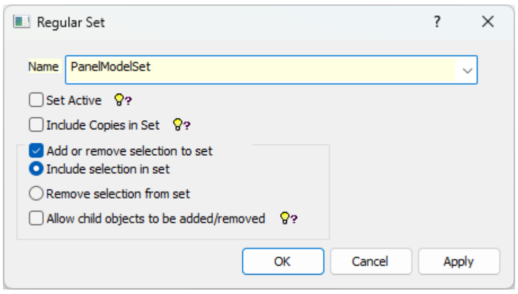



```
Pl1.front.WetSurface = WS1;
Pl2.front.WetSurface = WS1;
Pl3.front.WetSurface = WS1;
Pl4.front.WetSurface = WS1;
```

1.1.8 Adding the plates to a named set

Select *Pl1* to *Pl4*, right-click one of them and select **Named set**. Add them to a new regular set named **PanelModelSet**. This named set will be used to define the subset of the model for panel model meshing.

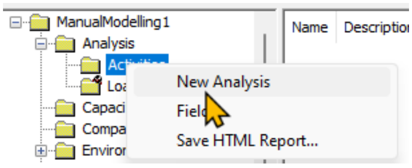




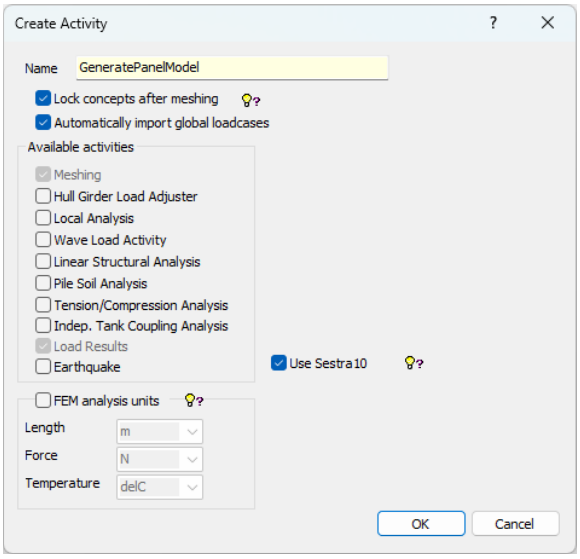

```
PanelModelSet = Set();
PanelModelSet.add(Pl1);
PanelModelSet.add(Pl2);
PanelModelSet.add(Pl3);
PanelModelSet.add(Pl4);
```

1.1.9 Creating analysis to generate panel model mesh

Create a new analysis by right-clicking **Analysis > Activities** and selecting **New Analysis**.



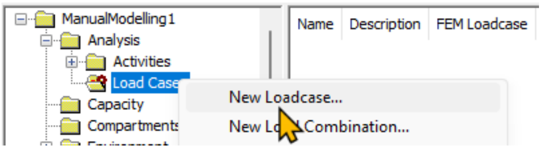
Name the analysis **GeneratePanelModel** and remove all activities, as we only need to mesh the concept model.



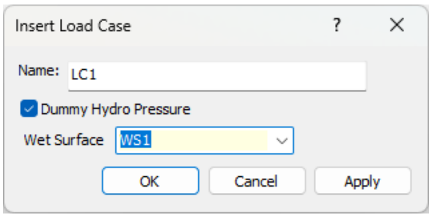

```
GeneratePanelModel = Analysis(true);
GeneratePanelModel.add(MeshActivity());
GeneratePanelModel.add(LoadResultsActivity());
GeneratePanelModel.setActive();
```

1.1.10 Creating a dummy hydro load case

Next, create a dummy hydro load case to allow HydroD to recognize the wetted part of the model. Right-click **Analysis > Load Cases** and select **New Loadcase**.



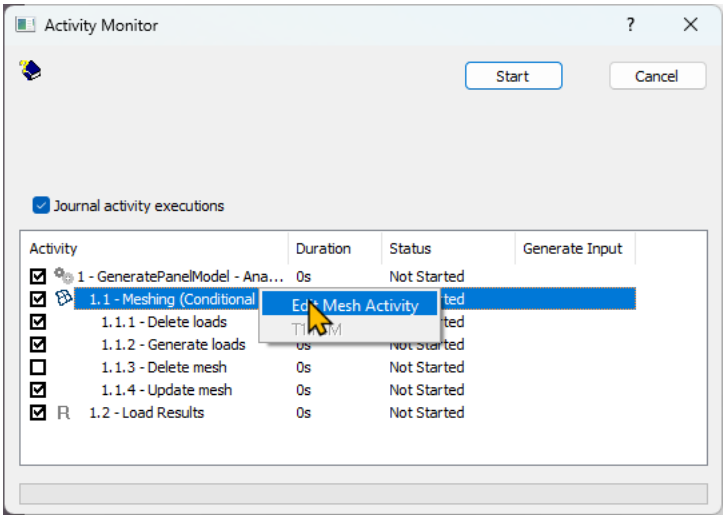
Select **Dummy Hydro Pressure** option and set **Wet Surface** to ***WS1***.



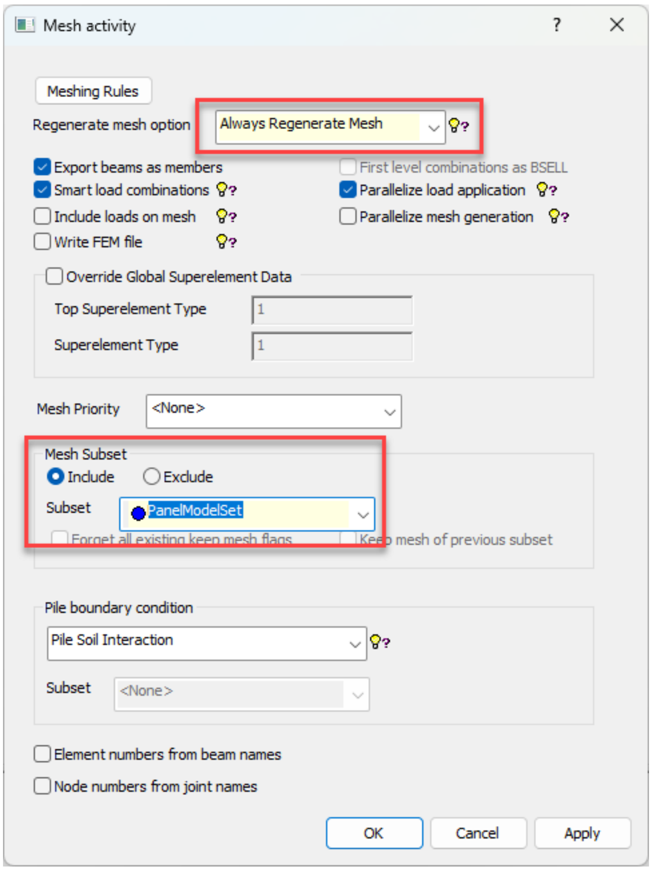

```
LC1 = DummyHydroLoadCase(WS1);
```

1.1.11 Generating and exporting the panel model mesh

Open activity monitor (**Alt + D**), right-click **Meshing activity**, and select **Edit Mesh Activity**.



Set the **Regenerate mesh option** to **Always Regenerate Mesh,** and select **Mesh Subset** to **Include** with the **PanelModelSet** selected as the subset**.**

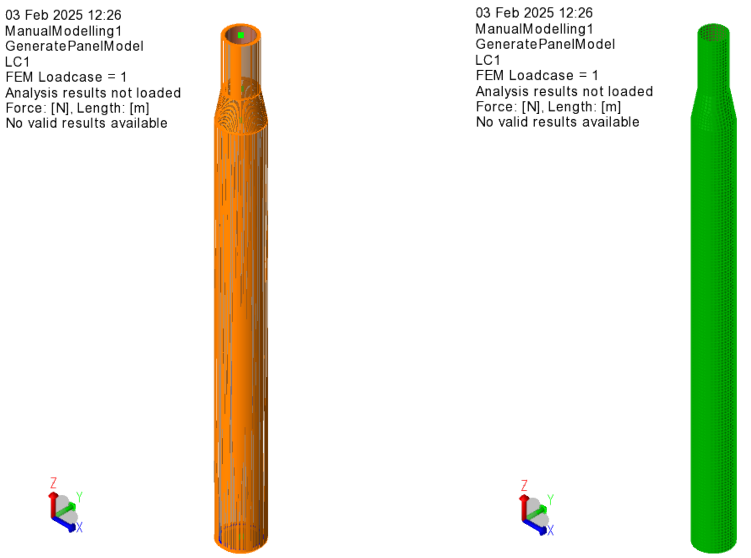



```
GeneratePanelModel.step(1).subset = PanelModelSet;
GeneratePanelModel.step(1).regenerateMeshOption = anAlwaysRegenerateMesh;
```

Tick the **Journal activity executions** checkbox and click **Start** to run the analysis to perform the meshing.

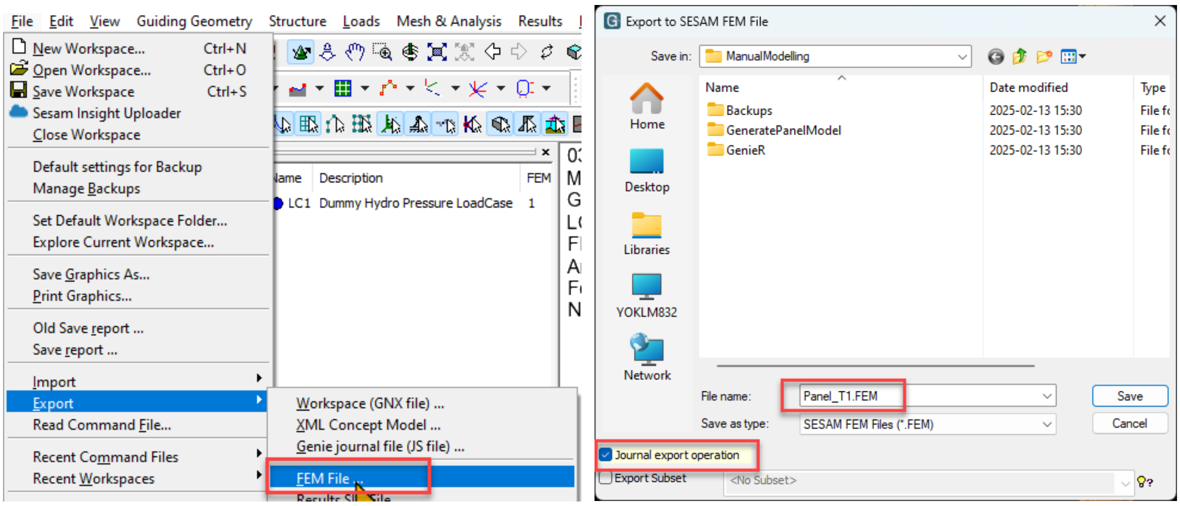
```
SimplifyTopology();
GeneratePanelModel.execute();
```

Check if the dummy hydro loads (displayed as orange vectors) are correctly applied to the front side of the outer hull plates. Also, inspect the mesh quality for any issues such as sharp edges, sliver elements, etc.



Finally, select **File > Export > FEM File,** tick **Journal export operation** option, and export the FEM file as **Panel_T1.FEM**.

The journal export operation logs the JS commands associated with the export process.




```
ExportMeshFem1 = ExportMeshFem();
ExportMeshFem1.DoExport("Panel_T1.FEM");
```

With this, the panel model file (*Panel_T1.FEM*) is created. Verify the location of this file and keep it for later use.

1.2 Panel Model – Automated Modelling

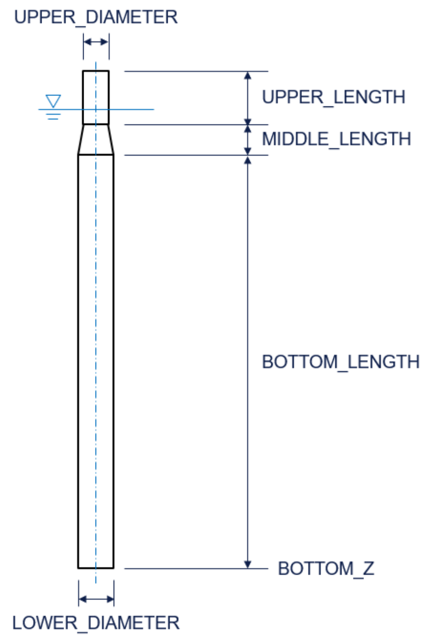
In this sub-chapter, we will modify the JS commands from the manual modelling sub-chapter to create a simple parametric script. This approach is one of the simplest ways to generate a parametric script.

You can find the complete JS script from your manual modelling session in its corresponding workspace folder. The script file is named after the workspace, for example, if you used the “ManualModelling” as the workspace name, the file will be located in the ManualModelling folder as **ManualModelling.js** or **ManualModelling_<datetime>.js**.

Note: If you are unable to locate it, you may use the **Ch1-1_PanelManualModelling.js** located in the **input_files\Chapter 1** folder.

1.2.1 Design parameters

The design parameters are as follows:



In addition, we will define the mesh density (**MESH_DENSITY**) and the panel model file name prefix (**PANEL_MODEL_PREFIX**) as additional parameters. The prefix helps differentiate between files and ensures that the file names do not conflict with each other.

1.2.2 Defining user defined parameters

First, we need to define the parameters as variables. Let's define them at the start of the script:

```
// USER DEFINED PARAMETERS
UPPER_LENGTH = 14;
MIDDLE_LENGTH = 8;
BOTTOM_LENGTH = 108;
UPPER_DIAMETER = 6.5;
BOTTOM_DIAMETER = 9.4;
BOTTOM_Z = -120;
MESH_DENSITY = 0.75;
PANEL_MODEL_PREFIX = "Panel1_";
```

1.2.3 Pre-calculating variables

To avoid recalculating the same values multiple times, we will pre-calculate the following two variables:

```
// PRE-CALCULATION OF DERIVED PARAMETERS
upperRadius = UPPER_DIAMETER/2;
bottomRadius = BOTTOM_DIAMETER/2;
```

1.2.4 Modifying the manual modelling scripts

Finally, take the scripts generated in the previous sub-chapter and make slight modifications. The modifications are highlighted in red, with the design parameters **bolded**.

```
// MAIN FUNCTIONS
GenieRules.Compatibility.version = "V8.12-03";
GenieRules.Tolerances.useTolerantModelling = true;
GenieRules.Tolerances.angleTolerance = 2 deg;
GenieRules.Meshing.autoSimplifyTopology = true;
GenieRules.Meshing.eliminateInternalEdges = true;
GenieRules.BeamCreation.DefaultCurveOffset = ReparameterizedBeamCurveOffset();
GenieRules.Geometry.AssemblyType = DualAssembly;
GenieRules.Transformation.DefaultConnectedCopy = false;
GenieRules.Geometry.AssemblyType = DualAssembly;
S355 = MaterialLinear(355000000 Pa, 7850 kg/m^3, 2.1e+11 Pa, 0.3, 1.2e-05 delC^-1, 0.03 N*s/m);
PanelMeshDensity = MeshDensity(MESH_DENSITY);
Th25 = Thickness(0.025 m);
WS1 = WetSurface();
S355.setDefault();
Th25.setDefault();
Point1 = Point(0,0,BOTTOM_Z);
Point2 = Point1 + Vector3d(0,0,BOTTOM_LENGTH);
Point3 = Point2 + Vector3d(0,0,MIDDLE_LENGTH);
Point4 = Point3 + Vector3d(0,0,UPPER_LENGTH);
Pl1 = CreateShellCircularConeCylinder(Point1, bottomRadius, Point2, bottomRadius, 0, 360);
Pl2 = CreateShellCircularConeCylinder(Point2, bottomRadius, Point3, upperRadius, 0, 360);
Pl3 = CreateShellCircularConeCylinder(Point3, upperRadius, Point4, upperRadius, 0, 360);
Curve1 = CreateCircleFromPlaneAndRadius(Point1, Vector3d(0,0,1), bottomRadius);
Pl4 = CoverCurves(Curve1);
Pl4.flipNormal();
Pl1.meshDensity = PanelMeshDensity;
Pl2.meshDensity = PanelMeshDensity;
Pl3.meshDensity = PanelMeshDensity;
Pl4.meshDensity = PanelMeshDensity;
Pl1.front.WetSurface = WS1;
Pl2.front.WetSurface = WS1;
Pl3.front.WetSurface = WS1;
```

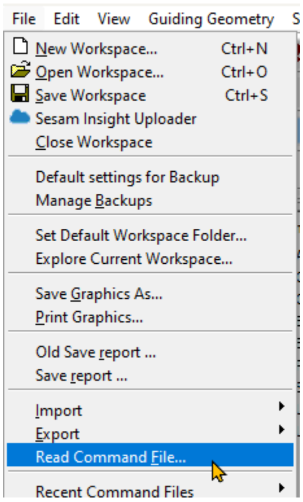
```
Pl4.front.WetSurface = WS1;
PanelModelSet = Set();
PanelModelSet.add(Pl1);
PanelModelSet.add(Pl2);
PanelModelSet.add(Pl3);
PanelModelSet.add(Pl4);
GeneratePanelModel = Analysis(true);
GeneratePanelModel.add(MeshActivity());
GeneratePanelModel.add(LoadResultsActivity());
GeneratePanelModel.setActive();
LC1 = DummyHydroLoadCase(WS1);
GeneratePanelModel.step(1).subset = PanelModelSet;
GeneratePanelModel.step(1).regenerateMeshOption = anAlwaysRegenerateMesh;
SimplifyTopology();
GeneratePanelModel.execute();
ExportMeshFem1 = ExportMeshFem();
ExportMeshFem1.DoExport(PANEL_MODEL_PREFIX + "T1.FEM");
```

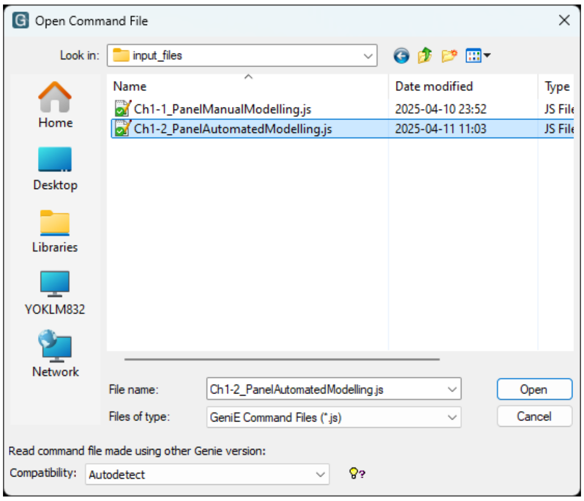
Save the modified parametric script file.

Note: A completed JS file, **Ch1-2_PanelAutomatedModelling.js**, is provided in the **input_files\Chapter 1** folder.

1.2.5 Running the parametric script and checking the panel model

Create a new **GeniE** workspace and run the script by selecting **File > Read Command File**.
In the next window, browse for the parametric script (or the **Ch1-2_PanelAutomatedModelling.js**) and click **Open**.

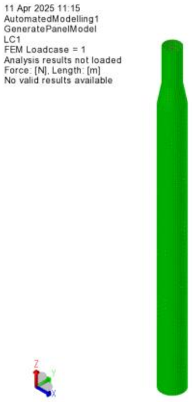




The model and the mesh will be created in sequence. After the operations are complete, verify the following:

- The dimensions
- The dummy hydro loads
- The mesh quality

Additionally, ensure that **Panel1_T1.FEM** has been outputted correctly.



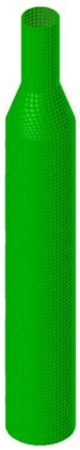
We can experiment by changing some design parameters and observing the results.

Table 1 Some design cases for experiment

Design Parameter	Case 1	Case 2	Case 3
UPPER_LENGTH	14	14	14
MIDDLE_LENGTH	8	8	8
BOTTOM_LENGTH	108	64.92667	46.81687
UPPER_DIAMETER	6.5	6.5	6.5
BOTTOM_DIAMETER	9.4	12	14
BOTTOM_Z	-120	-76.9267	-58.8169

Case 1

Case 2



Case 3

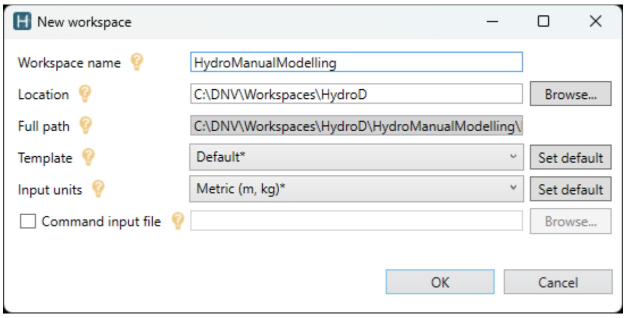
2 HYDRODYNAMIC ANALYSIS IN HYDROD

2.1 Hydrodynamic Analysis – Manual Setup

In this chapter, we will set up and execute the frequency-domain hydrodynamic analysis using the **Panel_T1.FEM** file generated in the previous chapter.

2.1.1 Creating a new workspace

Start **HydroD** and create a new workspace. To open the *New workspace* dialog, select **File > New**.

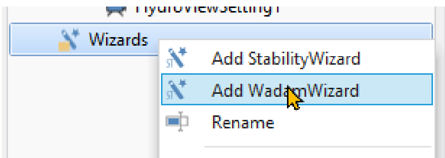


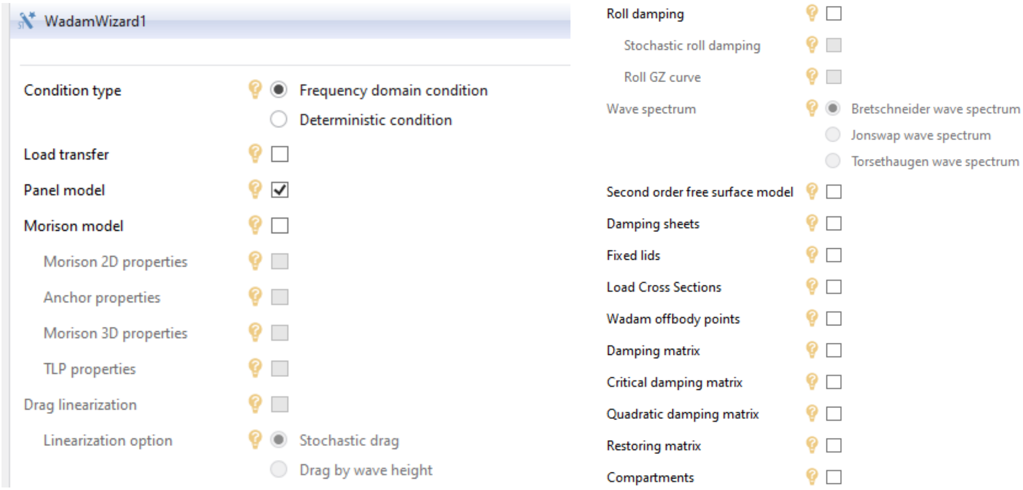
The following JS commands will be logged in the **Commands** window of HydroD (by default, located at the bottom of the screen, grouped with Output and Status tabs):

```
// HydroD V7.2-01 started YYYY-MM-DD hh:mm:ss
//
Workspace.AccelerationUnit = "m/s^2";
Workspace.AngleUnit = "deg";
Workspace.AngularFrequencyUnit = "deg/s";
Workspace.AreaUnit = "m^2";
…
Workspace = Reference("/Workspace");
Wizards = new WizardsFolder(Workspace, "Wizards");
Wizards.Description = null;
// HydroD V7.2-01 ended YYYY-MM-DD hh:mm:ss
// HydroD V7.2-01 started YYYY-MM-DD hh:mm:ss
```

2.1.2 Creating a Wadam wizard

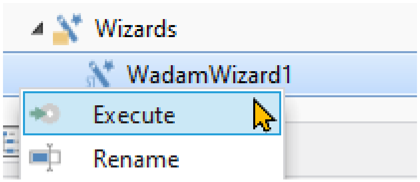
Create a **Wadam wizard** from browser with settings as shown below.

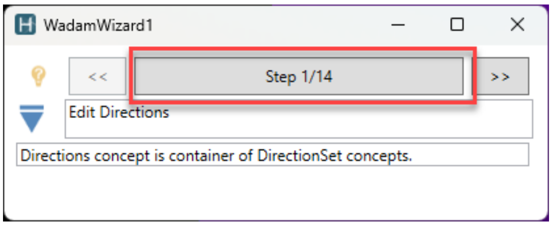





```
WadamWizard1 = new WadamWizard(Wizards, "WadamWizard1");
WadamWizard1.PanelModel = true;
```

Select the wizard in the browser and select **Execute** from the right-click menu. Click the **Step** button to create the relevant concept.

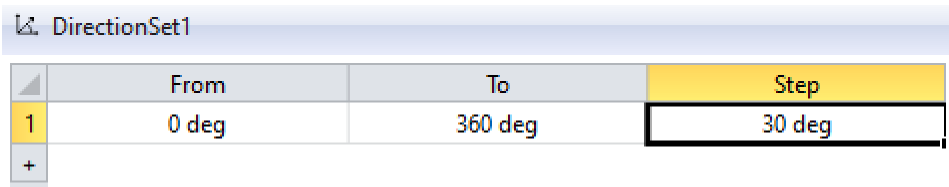




2.1.3 Defining environmental data

Step 1-2: Create a Directions folder and a DirectionSet.

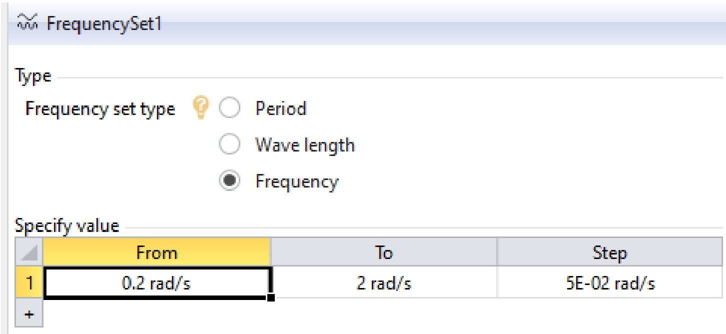
Give directions from **0 deg** to **360 deg** in steps of **30 deg**.




```
Directions1 = new Directions(Environment, "Directions1");
DirectionSet1 = new DirectionSet(Directions1, "DirectionSet1");
DirectionSet1.Items.Add(new AngleInterval());
DirectionSet1.Items[0].To = 360 deg;
DirectionSet1.Items[0].Step = 45 deg;
DirectionSet1.Items[0].Step = 15 deg;
DirectionSet1.Items[0].Step = 30 deg;
```

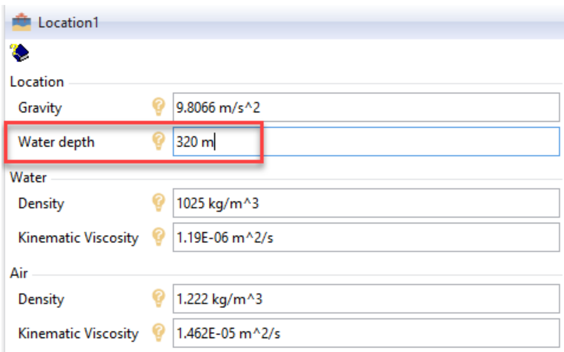
Step 3-4: Create a Water folder and a FrequencySet.

Select Frequency and specify from **0.2 rad/s** to **2 rad/s** in steps of **0.05 rad/s**.




```
Water1 = new Water(Environment, "Water1");
FrequencySet1 = new FrequencySet(Water1, "FrequencySet1");
FrequencySet1.WavePeriodItems.Add(new TimeInterval());
FrequencySet1.WavePeriodItems[0].To = 30 s;
FrequencySet1.WavePeriodItems[0].From = 5 s;
FrequencySet1.WavePeriodItems[0].Step = 5 s;
FrequencySet1.Type = FrequencySetType.Frequency;
FrequencySet1.WaveFrequencyItems.Add(new FrequencyInterval());
FrequencySet1.WaveFrequencyItems[0].From = 0.2 rad/s;
FrequencySet1.WaveFrequencyItems[0].To = 0.2 rad/s;
FrequencySet1.WaveFrequencyItems[0].To = 2 rad/s;
FrequencySet1.WaveFrequencyItems[0].Step = 5E-02 rad/s;
```

Step 5: In **Location**, set the water depth to **320 m**. Keep the default settings for the other items.




```
Location1 = new Location(Environment, "Location1");
Location1.WaterDepth = 320 m;
```

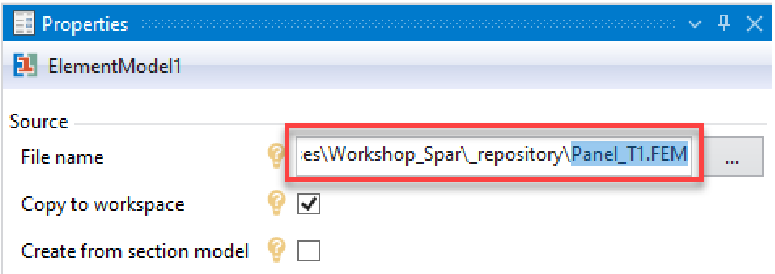
Step 6: Combine the *DirectionSet1* and the *FrequencySet1* into a **Frequency domain condition**.

```
FrequencyDomainCondition1 = new FrequencyDomainCondition(Location1,
"FrequencyDomainCondition1");
FrequencyDomainCondition1.DirectionSet = DirectionSet1;
FrequencyDomainCondition1.FrequencySet = FrequencySet1;
```

2.1.4 Defining a hydro model

Step 7-8: Create a **HydroModel**. Note that the values for AP and FP will not be used in this analysis.

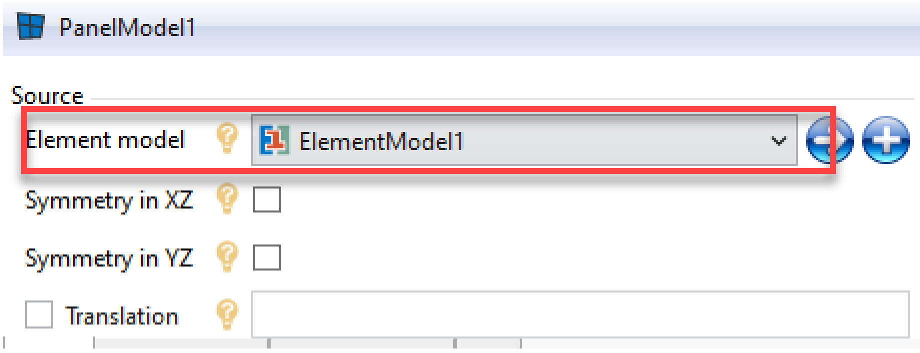
In the **ElementModel1**, browse to the **Panel_T1.FEM** file outputted in Chapter 1.




```
HydroModel1 = new HydroModel(HydroModels, "HydroModel1");
ElementModel1 = new ElementModel(Models, "ElementModel1");
ElementModel1.FileName =
"C:\\DNV\\Workspaces\\Workshop_Spar\\_repository\\Panel_T1.FEM";
```

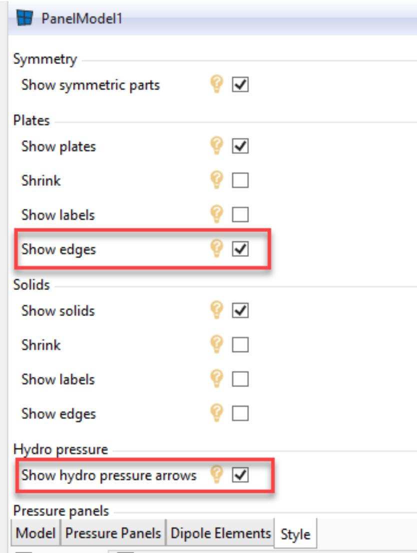
Note: The location of the Panel_T1.FEM file may be different.

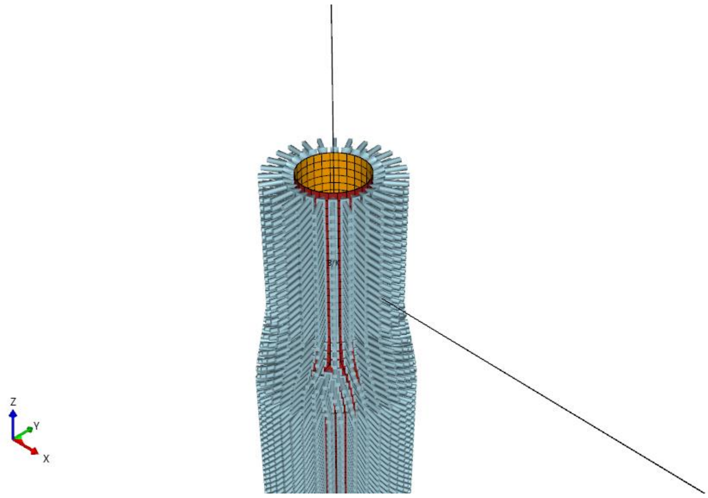
Step 9: Select **ElementModel1** as the panel model.




```
PanelModel1 = new PanelModel(HydroModel1, "PanelModel1");
PanelModel1.ElementModel = ElementModel1;
```

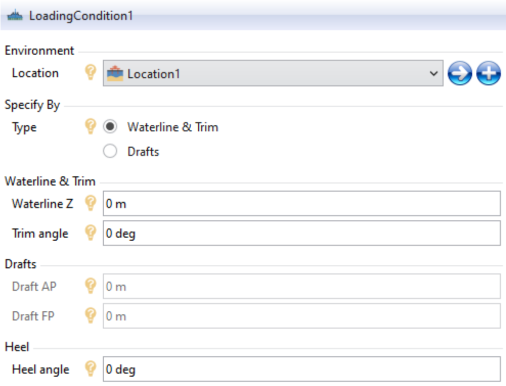
In the **Style** tab, check the boxes for **Show edges** and **Show hydro pressure arrows**. Verify that the hydro pressure arrows are pointing inward, indicating the wetted surface is defined properly.





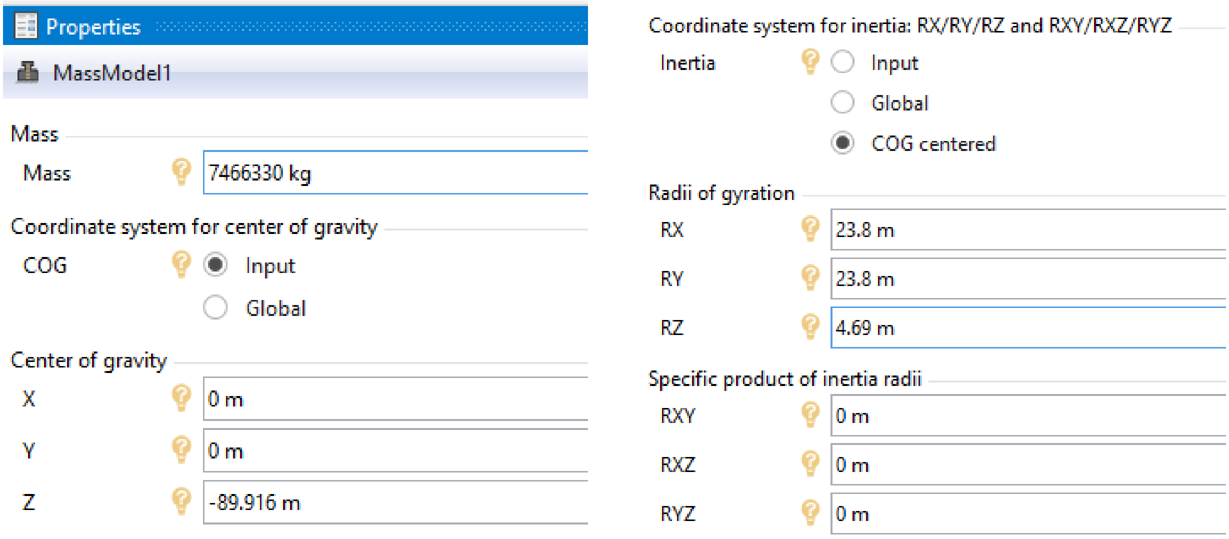
```
PanelModel1.ShowPlateEdges = true;
PanelModel1.ShowHydroPressureArrows = true;
```

Step 10-11: Create a **LoadingConditions** folder and a **Loading condition**. Specify 0 m as waterline and zero trim and heel angles.




```
LoadingConditions1 = new LoadingConditions(HydroModel1, "LoadingConditions1");
LoadingCondition1 = new LoadingCondition(LoadingConditions1, "LoadingCondition1");
LoadingCondition1.Location = Location1;
```

Step 12: Create a **Mass Model**. Select **User specified** and enter the following parameters.




```
MassModel1 = new MassModel(LoadingCondition1, "MassModel1");
MassModel1.UserSpecifiedMass = 7.46633E+06 kg;
MassModel1.UserSpecifiedCenterOfGravityX = 0 m;
MassModel1.UserSpecifiedCenterOfGravityY = 0 m;
MassModel1.UserSpecifiedCenterOfGravityZ = -89.9155 m;
MassModel1.UserSpecifiedRadiusOfGyrationX = 23.8 m;
MassModel1.UserSpecifiedRadiusOfGyrationY = 23.8 m;
MassModel1.UserSpecifiedRadiusOfGyrationZ = 4.69 m;
```

Note: This mass model is not yet balanced with the floater's displacement. For a first-order wave hydrodynamic analysis, an unbalanced condition is acceptable, as it does not affect the first-order wave excitation forces, the added mass coefficients, or the wave damping coefficients. However, achieving a balanced condition is crucial for accurately computing the second-order wave loads.

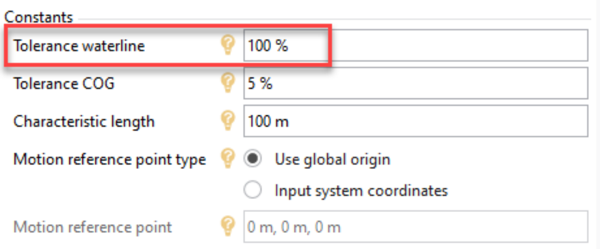
2.1.5 Creating and executing Wadam analysis

Step 13: Create a Wadam analysis. Please note the following commands are truncated to save space.

```
WadamAnalysis1 = new WadamAnalysis(Analyses, "WadamAnalysis1");
WadamAnalysis1.DataPreparation = true;
…
WadamAnalysis1.EnvironmentCondition = FrequencyDomainCondition1;
WadamAnalysis1.UseStochasticLinearization = false;
WadamAnalysis1.UseRollGZCurve = false;
```

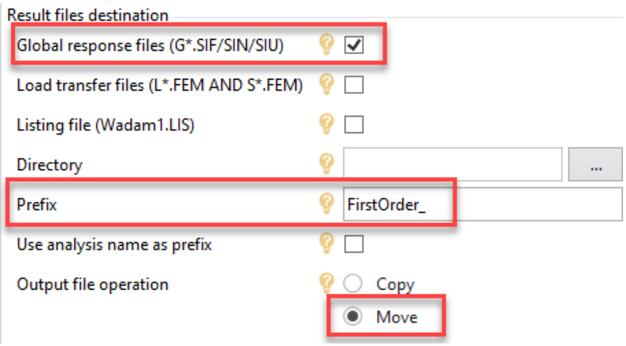
Modify the following options:

In **General** tab, change the **tolerance waterline** to **100%**. This adjustment is necessary to allow Wadam to calculate an unbalanced condition without stopping the analysis.



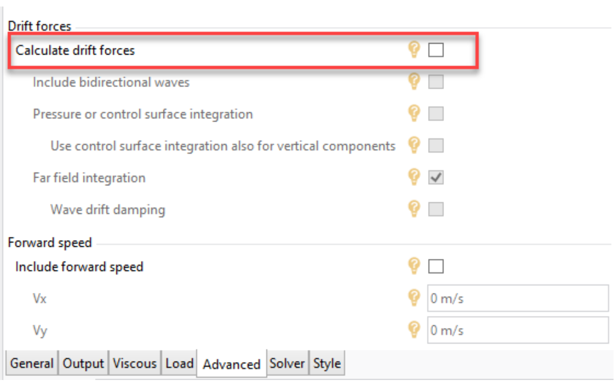

```
WadamAnalysis1.ToleranceWaterline = new Fraction(1);
```

In the **Output** tab, check the box for **Global Response Files (G*.SIF/SIN/SIU)**.
Set the **Prefix** to **"FirstOrder_"**, and select **Move** as the **Output File Operation**.
This will save the resulting **G1.SIF** file as **FirstOrder_G1.SIF**.



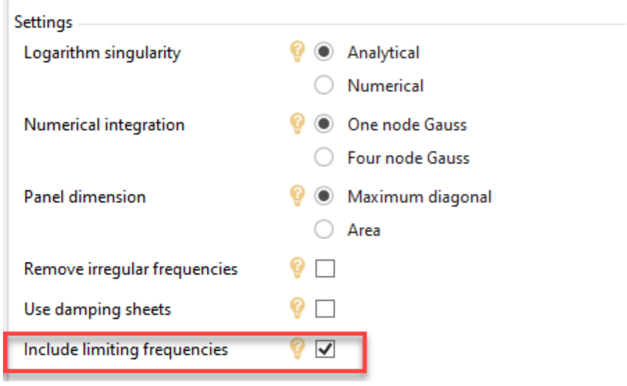

```
WadamAnalysis1.GlobalResponseFiles = true;
WadamAnalysis1.Prefix = "FirstOrder_";
WadamAnalysis1.OutputFileOperationType = OutputFileOperationType.Move;
```

In the **Advanced** tab, uncheck the **Calculate drift forces**.




```
WadamAnalysis1.CalculateDriftForces = false;
```

In the **Solver** tab, check **Include limiting frequencies**.



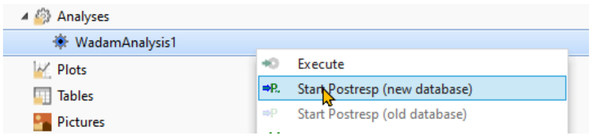

```
WadamAnalysis1.IncludeLimitingFrequencies = true;
```

Step 14: Execute the Wadam analysis and wait for it to finish.

```
WadamAnalysis1.Execute();
```

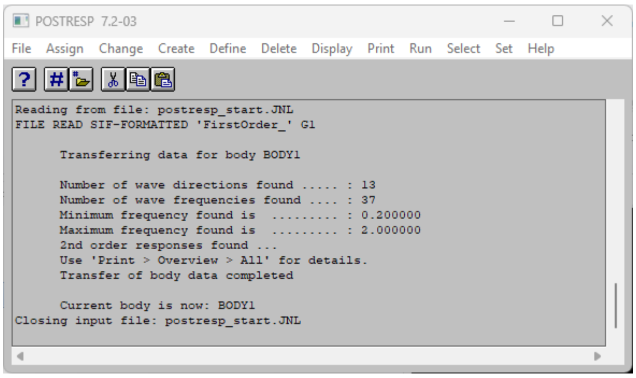
2.1.6 Checking the results with Postresp

Right-click **WadamAnalysis1** and select **Start Postresp (new database)**.




```
WadamAnalysis1.StartPostrespNewDatabase();
```

The **FirstOrder_G1.SIF** file created by Wadam will be read automatically.

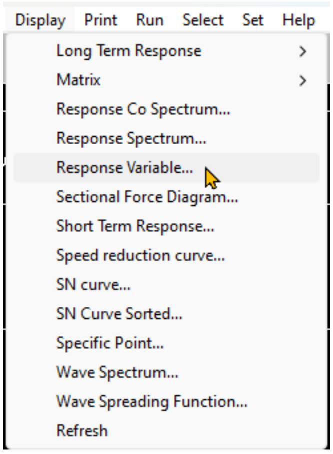


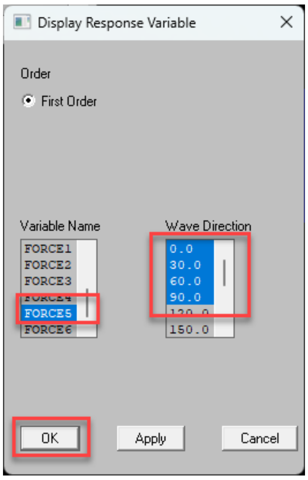
Display the wave transfer functions, for example FORCE5 (first-order wave moment in pitch direction).

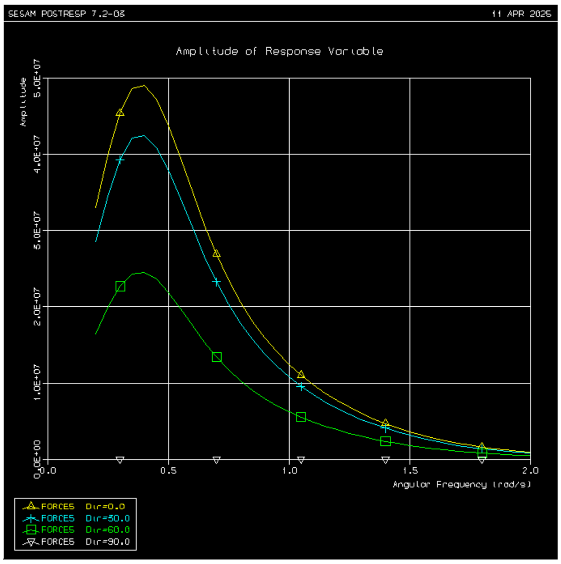
1. Click **Display > Response Variable**.

2. Select **FORCE5** as Variable Name and **0.0 – 90.0** as Wave Direction.

3. Click **OK**.



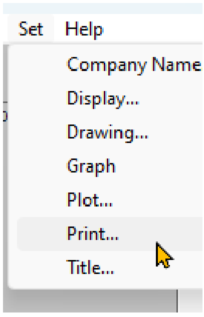


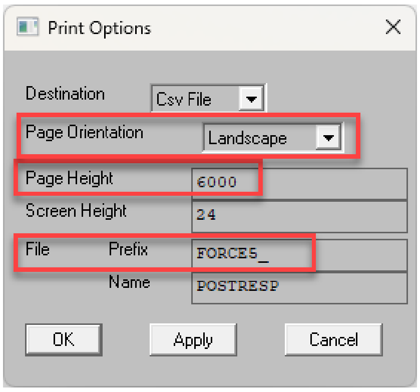


Let's print one of the response variables as a CSV file.

1. Click **Set > Print**.

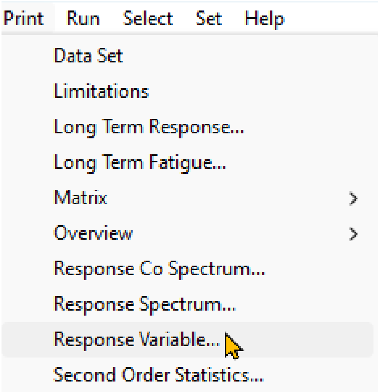
2. Set the print options as shown below:

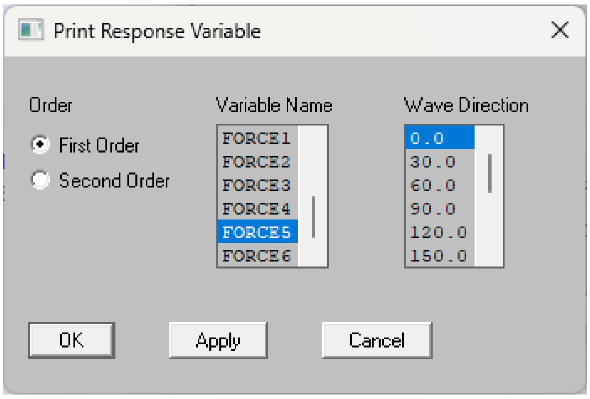




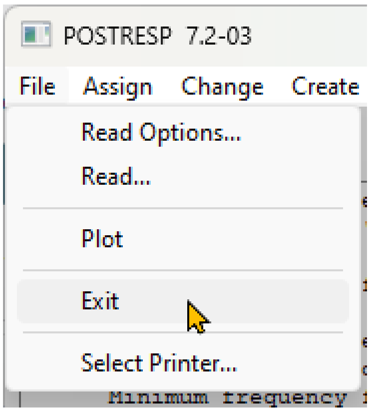
3. Click **Print > Response Variables**.

4. Select **First Order**, **FORCE5** as Variable Name, and **0.0** as Wave Direction.





Exit Postresp by clicking **File > Exit**.



The corresponding Postresp JNL commands for the above operations can be found in the analysis folder of the Wadam analysis as the **POSTRESP.JNL** file. Below are the commands logged in the file:

```
FILE READ SIF-FORMATTED 'FirstOrder_' G1
DISPLAY RESPONSE-VARIABLE ( ONLY FORCE5 ) ( ONLY 0.0 30.0 60.0 90.0 )
SET PRINT DESTINATION CSV-FILE
SET PRINT PAGE-HEIGHT 6000
SET PRINT SCREEN-HEIGHT 24
SET PRINT FILE FORCE5_ POSTRESP
SET PRINT PAGE-ORIENTATION LANDSCAPE
PRINT RESPONSE-VARIABLE ( ONLY FORCE5 ) ( ONLY 0.0 )
```

We can use this JNL file to operate Postresp.

2.2 Hydrodynamic Analysis – Automated Setup

We will use the same approach here, modifying part of the manual setup scripts generated in the previous sub-chapter.

You can find the complete JS script from your manual modelling session in its corresponding workspace folder. The script file is named after the workspace, for example, if you used the "HydroManualModelling" as the workspace name, the file will be located in the HydroManualModelling folder as **HydroManualModelling.js** or **HydroManualModelling_<datetime>.js**.

Note: If you are unable to locate it, you may use the **Ch2-1_HydroManualSetup.js** located in the **input_files\Chapter 2** folder.

2.2.1 Design parameters

We will define the design parameters as follows:

- **COG_Z**: The substructure's center of gravity Z position
- **PANEL_MODEL_FILEPATH**: The panel model file path
- **G1_PREFIX**: The prefix for the G1.SIF file

You may also add other parameters, such as the frequency range or the location's depth, but for simplicity, we will focus on these parameters for now.

2.2.2 Defining user defined parameters

The user-defined parameters are as follows:

```
// USER DEFINED PARAMETERS
COG_Z = -89.9155 m;
PANEL_MODEL_FILEPATH = "Panel1_T1.FEM";
G1_PREFIX = "FirstOrder1_";
```

2.2.3 Pre-calculating variables

We do not need to pre-calculate any variable this time.

2.2.4 Modifying the manual setup scripts

Take the scripts generated in the previous sub-chapter and add slight modifications. The modifications are highlighted in red, with the design parameters **bolded**. Some parts of the code that remain unmodified are removed to save space. Note that comments from the manual modelling scripts have been removed for brevity.

```
// MAIN FUNCTION
Workspace.AccelerationUnit = "m/s^2";
Workspace.AngleUnit = "deg";
Workspace.AngularFrequencyUnit = "deg/s";
Workspace.AreaUnit = "m^2";
Workspace.DensityUnit = "kg/m^3";
Workspace.Description = null;
…
ElementModel1.FileName = PANEL_MODEL_FILEPATH;
PanelModel1 = new PanelModel(HydroModel1, "PanelModel1");
```

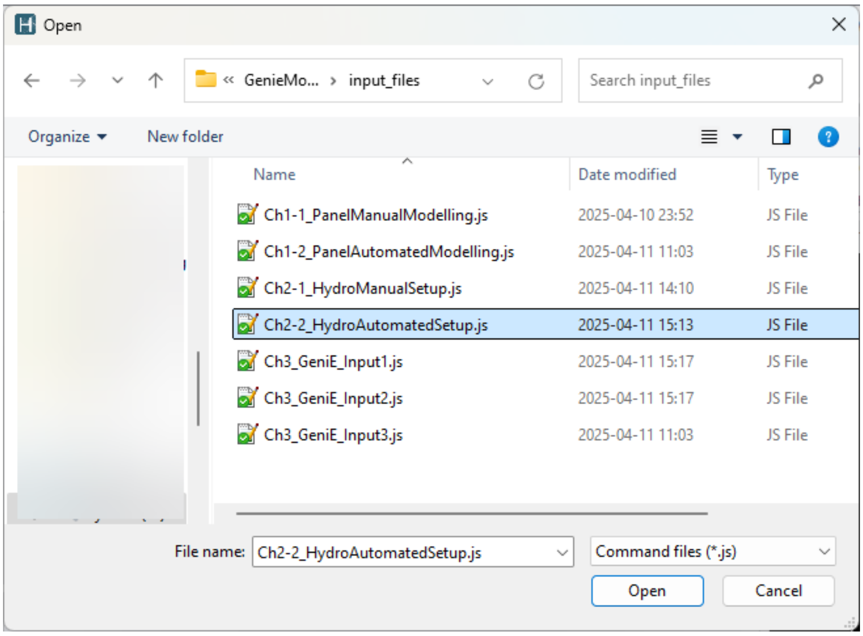
```
PanelModel1.ElementModel = ElementModel1;
PanelModel1.ShowPlateEdges = true;
PanelModel1.ShowHydroPressureArrows = true;
…
MassModel1.UserSpecifiedCenterOfGravityZ = COG_Z;
MassModel1.UserSpecifiedRadiusOfGyrationX = 23.8 m;
MassModel1.UserSpecifiedRadiusOfGyrationY = 23.8 m;
MassModel1.UserSpecifiedRadiusOfGyrationZ = 4.69 m;
…
WadamAnalysis1.Prefix = G1_PREFIX;
WadamAnalysis1.OutputFileOperationType = OutputFileOperationType.Move;
WadamAnalysis1.CalculateDriftForces = false;
WadamAnalysis1.IncludeLimitingFrequencies = true;
WadamAnalysis1.Execute();
~~WadamAnalysis1.StartPostrespNewDatabase();~~ //deleted because it is not needed
```

Note: A completed JS file, **Ch2-2_HydroAutomatedSetup.js**, is provided in the **input_files\Chapter 2** folder.

2.2.5 Running the parametric script and checking the results

Create a new **HydroD** workspace and run the script by going to **File > Read script**.

In the next window, browse for the parametric script and click **Open**.



The workspace will be populated, and the Wadam analysis will be run. After the operations are finished, verify the model and the hydro arrows are displayed correctly.

Also, check if the **FirstOrder1_G1.SIF** is outputted successfully.

In addition, you may open Postresp afterward to review and verify the analysis results.

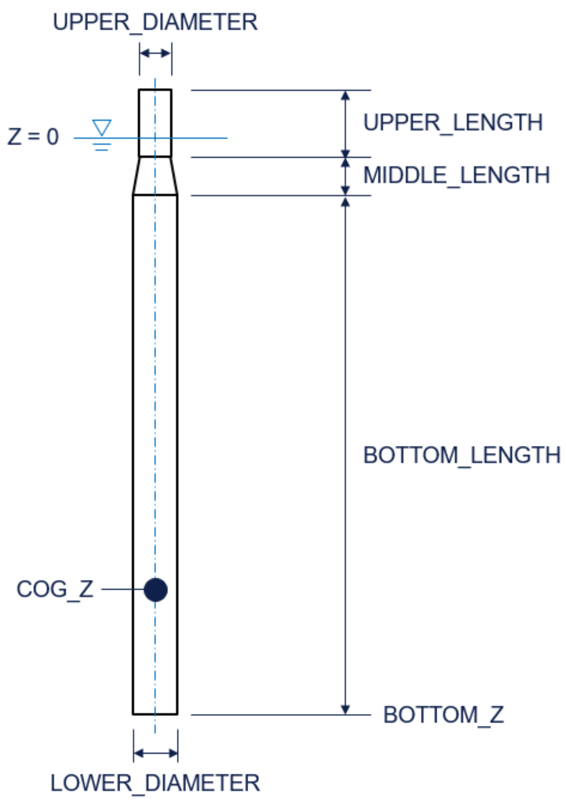
3 AUTOMATING GENIE AND HYDROD WITH SESAM MANAGER

We will use **Sesam Manager** to manage multiple cases of our analyses. **Sesam Manager** is a free module within **Sesam** that helps in managing analyses. In this chapter, we will perform a simple comparative study of the following spar-type FOW floaters, all with the same displaced volume.

Note: We will focus on how to set up the analyses rather than comparing the results themselves. Additionally, in an actual project, the user must ensure that the analyses have successfully converged before moving on to the next steps in the design workflow.

Table 2 Design cases to be inspected

Design Parameter	Case 1	Case 2	Case 3
UPPER_LENGTH	14	14	14
MIDDLE_LENGTH	8	8	8
BOTTOM_LENGTH	108	64.92667	46.81687
UPPER_DIAMETER	6.5	6.5	6.5
BOTTOM_DIAMETER	9.4	12	14
BOTTOM_Z	-120	-76.9267	-58.8169
COG_Z	-89.9155	-56.0	-40.0
Prefix	Case1_	Case2_	Case3_



3.1 Preparing the Parametric Scripts

3.1.1 Preparing the GeniE script

Prepare three separate **GeniE** parametric modelling scripts, one for each case. The only difference between the files is the value of the user-defined parameters. Copy the JS script you used in Chapter 1.2 (or use the provided **Ch1-2_PanelAutomatedModelling.js**) and name the copied files as follows:

- **Case 1:** Ch3_GeniE_Input1.js
- **Case 2:** Ch3_GeniE_Input2.js
- **Case 3:** Ch3_GeniE_Input3.js

Populate each file's **user defined parameters** according to Table 2. For example, for **Case 3**:

```
// USER DEFINED PARAMETERS
UPPER_LENGTH = 14;
MIDDLE_LENGTH = 8;
BOTTOM_LENGTH = 46.81687075;
UPPER_DIAMETER = 6.5;
BOTTOM_DIAMETER = 14;
BOTTOM_Z = -58.81687075;
MESH_DENSITY = 0.75;
PANEL_MODEL_PREFIX = "Case3_";
```

Note: For each case, the panel model will be saved as **Case#_T1.FEM** file. We need to reflect this in the HydroD scripts.

3.1.2 Preparing the HydroD script

Similarly, prepare three separate **HydroD** scripts using the JS script you used in Chapter 2.2 (or use the provided **Ch2-2_HydroAutomatedSetup.js**). Name the copied files as follows:

- **Case 1:** Ch3_HydroD_Input1.js
- **Case 2:** Ch3_HydroD_Input2.js
- **Case 3:** Ch3_HydroD_Input3.js

Populate each file's **user defined parameters** according to Table 2. For example, for **Case 3**:

```
// USER DEFINED PARAMETERS
COG_Z = -40.0 m;
PANEL_MODEL_FILEPATH = "Case3_T1.FEM";
G1_PREFIX = "Case3_";
```

3.1.3 Preparing the Postresp script

Prepare Postresp scripts by creating them manually for each case:

- **Case 1:** Ch3_POSTRESP1.JNL
- **Case 2:** Ch3_POSTRESP2.JNL
- **Case 3:** Ch3_POSTRESP3.JNL

Enter the following commands based on the operations performed in Chapter 2.1.6. Be sure to update the FILE READ section to match the prefix of the G1.SIF file generated by HydroD (highlighted red). For example, for **Case 3**:

```
FILE READ SIF-FORMATTED 'Case3_' G1
SET PRINT DESTINATION CSV-FILE
SET PRINT PAGE-HEIGHT 6000
SET PRINT SCREEN-HEIGHT 24
SET PRINT FILE CASE1_F5_ POSTRESP
SET PRINT PAGE-ORIENTATION LANDSCAPE
PRINT RESPONSE-VARIABLE ( ONLY FORCE5 ) ( ONLY 0.0 )
FILE EXIT
```

Note: The DISPLAY command has been omitted intentionally, as we do not require any graph output when running the script in the background. Additionally, make sure to include a FILE EXIT command at the end of the script. This is necessary to properly close Postresp and ensure that the analysis completes successfully.

3.1.4 Preparing Sesam Manager Post Execute Script for Postresp

The final file, **PostrespActivity_post.js**, is used to copy all the CSV files in the **Postresp** analysis folder to the **_repository** folder in the **Sesam Manager** workspace. It only contains one line:

```
Copy(CurrentActivity.Workspace + "*.csv", "_repository",
CurrentActivity.StartExecutionTime, FileSearchOption.AllDirectories);
```

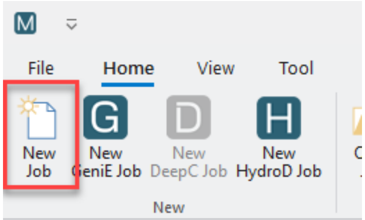
You may refer to Sesam Manager's User Manual for more information about the Post Execute Script functionality.

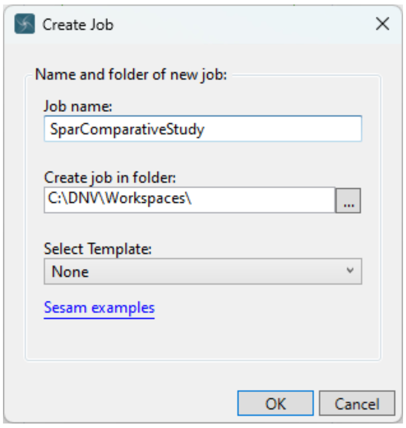
Note: Completed scripts are provided in the **input_files\Chapter 3\Scripts** folder.

3.2 Setting Up Sesam Manager

3.2.1 Creating a new Job

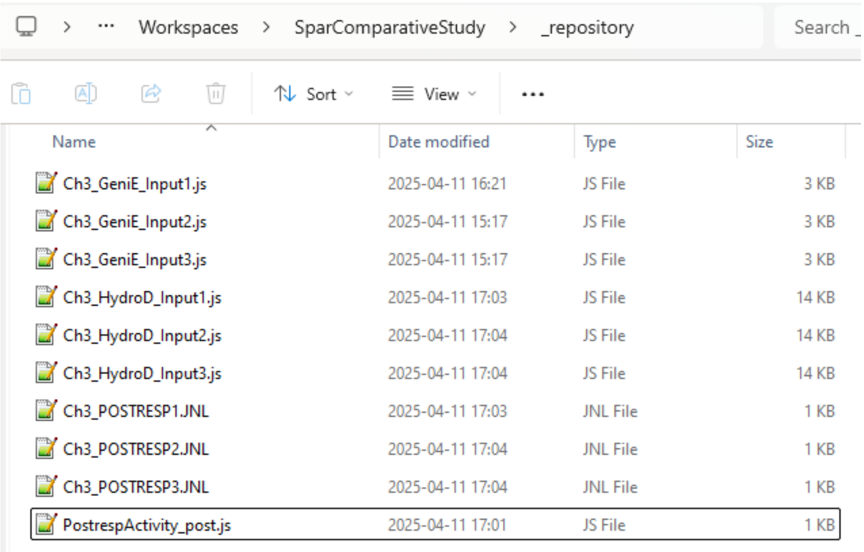
Open Sesam Manager 6.7 and click **New Job** icon.





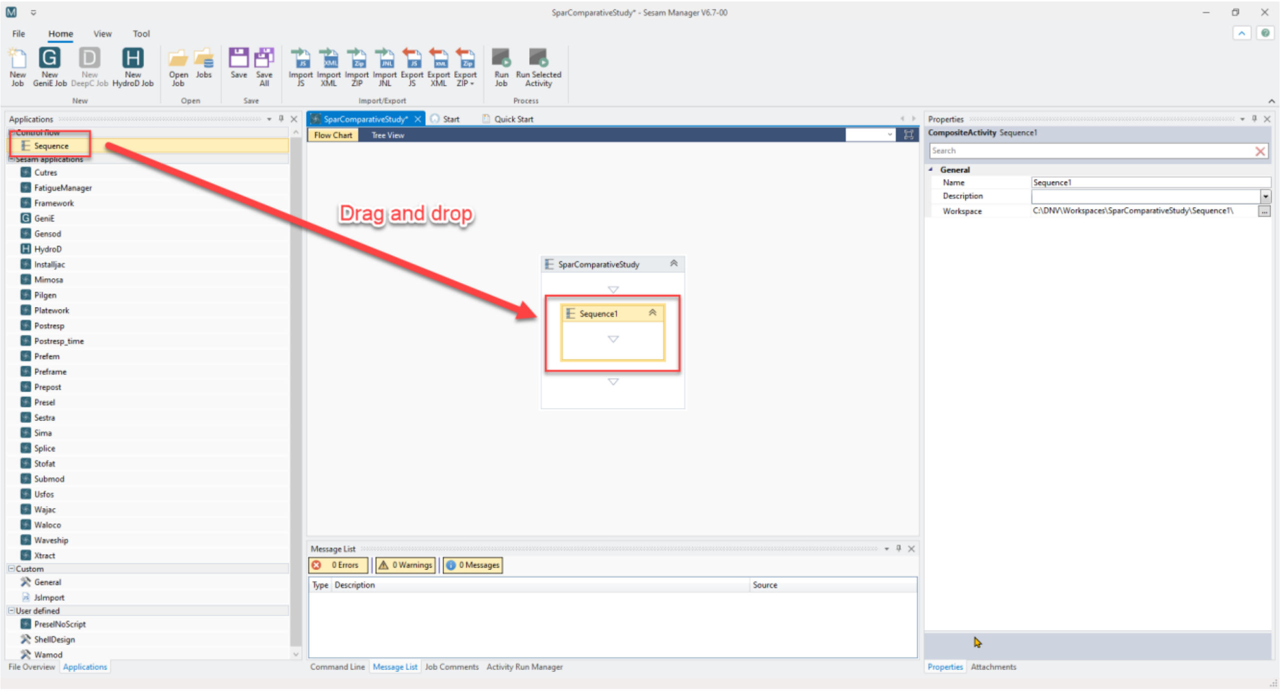
3.2.2 Copying the parametric modelling scripts

Copy the prepared scripts into the _**repository** folder inside the Sesam Manager workspace folder.



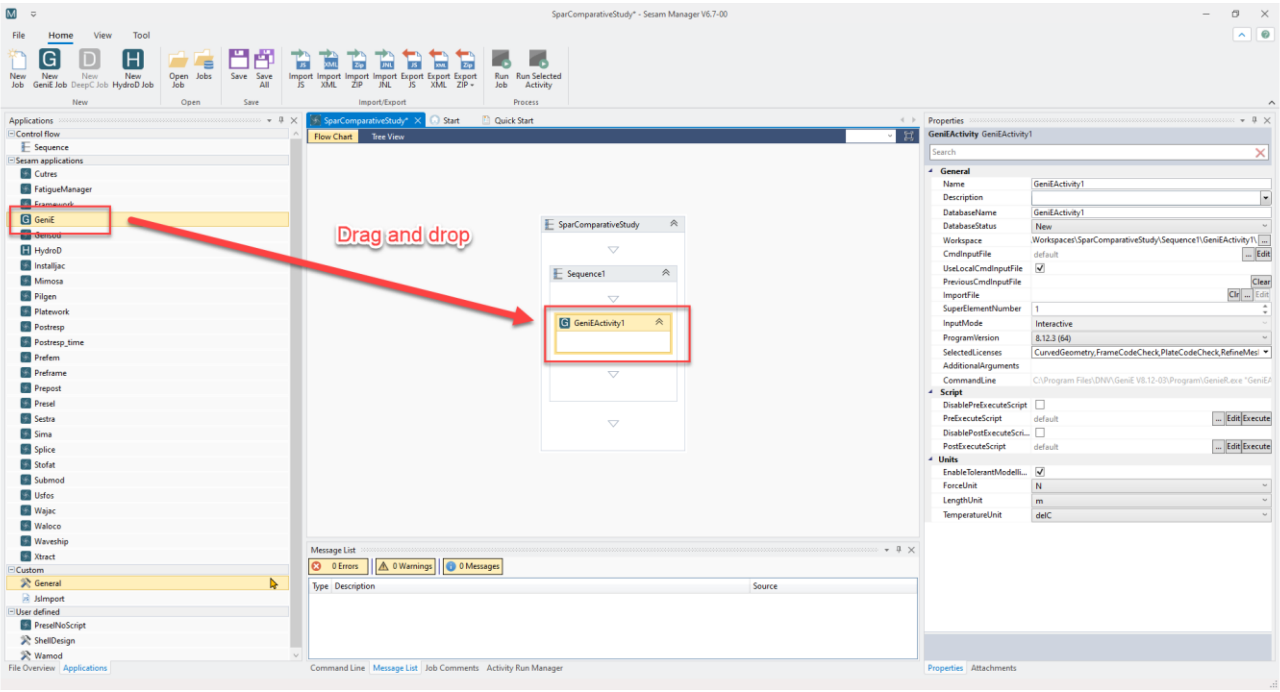
3.2.3 Creating a Sequence folder

Drag and drop a **Sequence** from the **Applications** pane (on the left) to the **Flow Chart** pane (at the centre of the window).



3.2.4 Creating a GeniE activity

Drag and drop **GeniE** from the **Applications** pane (on the left) to **Sequence1**.

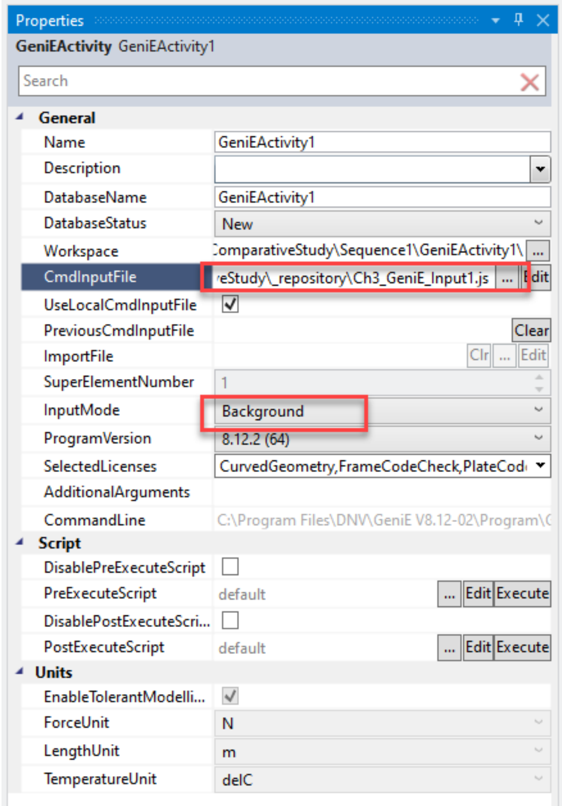


Click on the **GeniEActivity1** and look at the **Properties** pane (on the right).

Change the **CmdInputFile** by clicking the Browse button ....

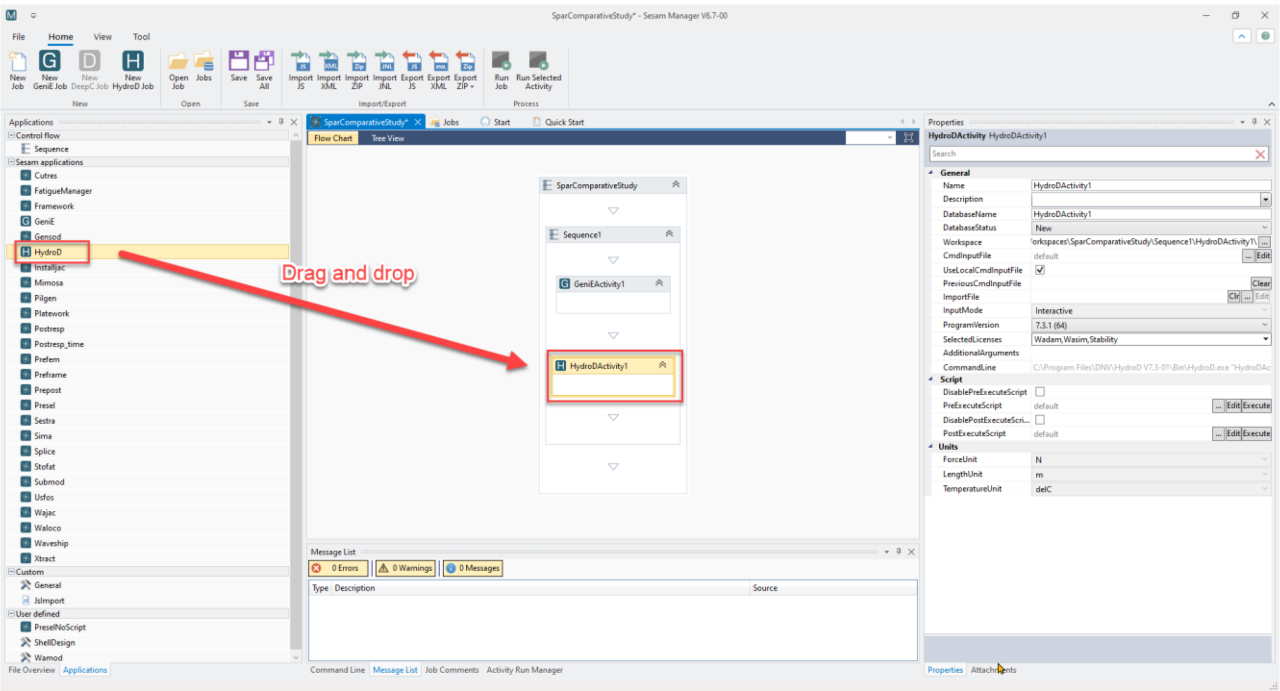
Set it to **Ch3_GeniE_Input1.js.**

Change the **InputMode** to **Background**.



3.2.5 Creating a HydroD activity

Drag and drop **HydroD** from the **Applications** pane to **Sequence1**, under the **GeniEActivity1**.

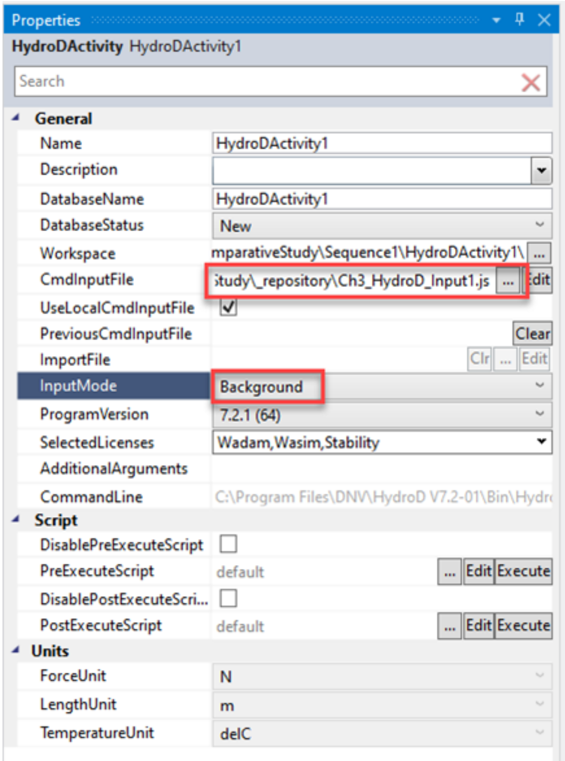


Select the newly created **HydroDActivity1**.

Change the **CmdInputFile** by clicking the Browse button ⋯.

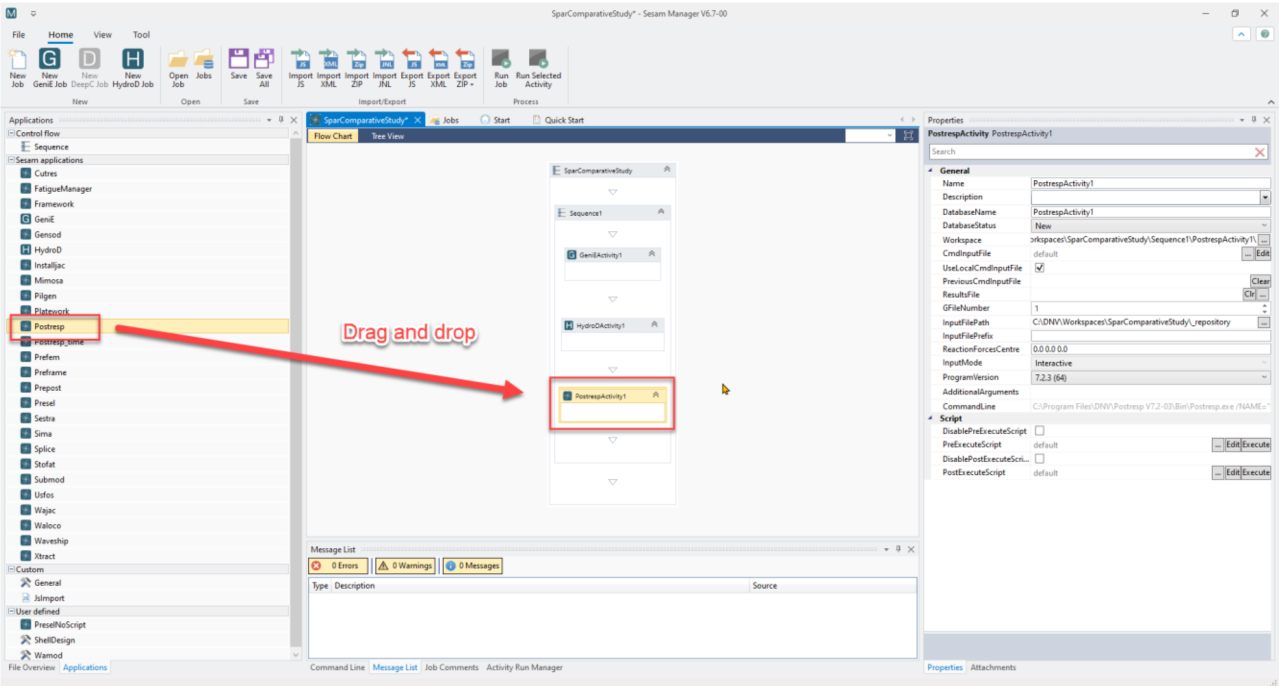
Set it to **Ch3_HydroD_Input1.js.**

Change the **InputMode** to **Background**.



3.2.6 Creating a Postresp activity

Similarly, drag and drop **Postresp** from the **Applications** pane to **Sequence1**, under the **HydroDActivity1**.



Select the newly created **PostrespActivity1**.

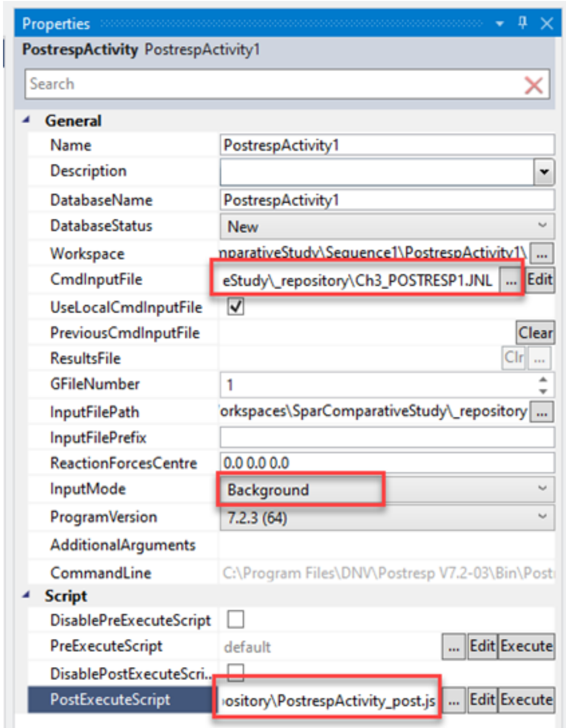
Change the **CmdInputFile** by clicking the Browse button .

Set it to **Ch3_POSTRESP1.JNL.**

Change the **InputMode** to **Background**.

Then, change the **PostExecuteScript** by clicking the Browse button .

Set it to **PostrespActivity_post.js**.

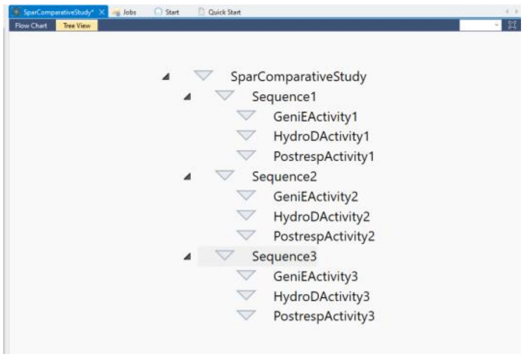


3.2.7 Creating the rest of the Job

Repeat the process to create **two more Sequences** for **Case 2** and **Case 3**.
Make sure each Sequence contains the correct:

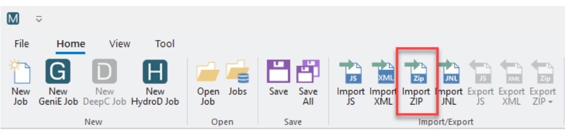
- **GeniE script** (Ch3_GeniE_Input2.js, Ch3_GeniE_Input3.js)
- **HydroD script** (Ch3_HydroD_Input2.js, Ch3_HydroD_Input3.js)
- **Postresp script** (Ch3_POSTRESP2.JNL, Ch3_POSTRESP3.JNL)

Once completed, your Job should contain three Sequences, each with the full set of activities.



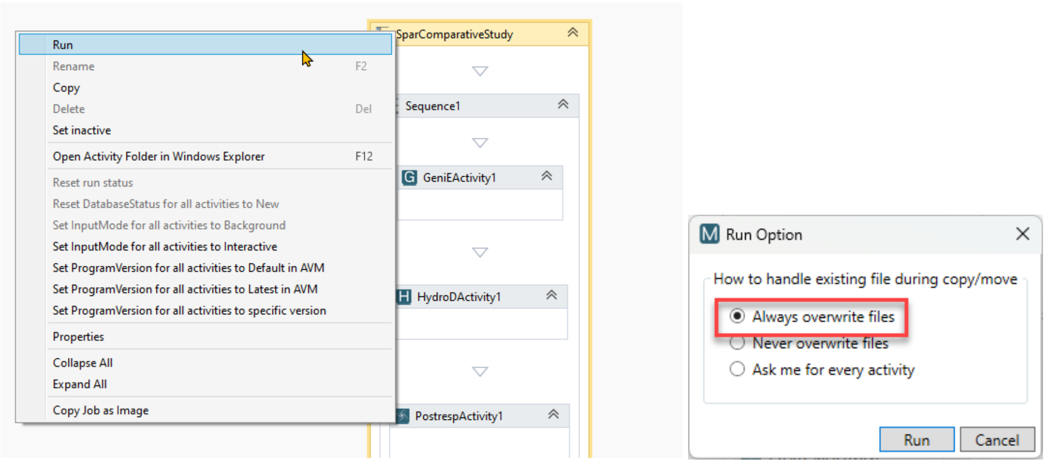
Tip: Switch to the **Tree View** tab to review the full structure clearly, as shown above.

Note: You can click Import ZIP and browse the **Ch3_SesamManagerJob.zip** provided in the input files folder to import the finished Sesam Manager Workspace.



3.2.8 Running all analyses

To run the analyses, right-click on the **SparComparativeStudy** container and select **Run**. When prompted, choose **Always overwrite files** and click **Run**.



The activities will run sequentially. After all analyses are finished, the results will be saved into the _**repository** folder:

Name	Date modified	Type	Size
CASE1_F5_POSTRESP.csv	2025-04-11 18:01	Microsoft Excel C...	9 KB
Case1_G1.SIF	2025-04-11 18:00	SIF File	431 KB
Case1_T1.FEM	2025-04-11 17:55	FEM File	4,732 KB
CASE2_F5_POSTRESP.csv	2025-04-11 18:06	Microsoft Excel C...	9 KB
Case2_G1.SIF	2025-04-11 18:05	SIF File	431 KB
Case2_T1.FEM	2025-04-11 18:01	FEM File	4,024 KB
CASE3_F5_POSTRESP.csv	2025-04-11 18:09	Microsoft Excel C...	9 KB
Case3_G1.SIF	2025-04-11 18:09	SIF File	431 KB
Case3_T1.FEM	2025-04-11 18:06	FEM File	3,609 KB

You can check each activity to ensure the analyses were set up correctly by running them individually, or review the CSV results containing the first-order wave moment in the pitch direction for each floater.

This concludes the Sesam tutorial. We hope it has inspired you to create more sophisticated parametric modeling methods tailored to your design workflow.

About DNV
DNV is an independent assurance and risk management provider, operating in more than 100 countries. Through its broad experience and deep expertise DNV advances safety and sustainable performance, sets industry standards, and inspires and invents solutions.

Digital Solutions
DNV is a world-leading provider of digital solutions and software applications with focus on the energy, maritime and healthcare markets. Our solutions are used worldwide to manage risk and performance for wind turbines, electric grids, pipelines, processing plants, offshore structures, ships, and more. Supported by our domain knowledge and Veracity assurance platform, we enable companies to digitize and manage business critical activities in a sustainable, cost-efficient, safe and secure way.

DNV | digital@dnv.com | dnv.com/digital