



WEC-Sim Technical Training Course

for users and developers



9/8/2023

PRESENTED BY

Jeff Grasberger





WEC-Sim Visualization

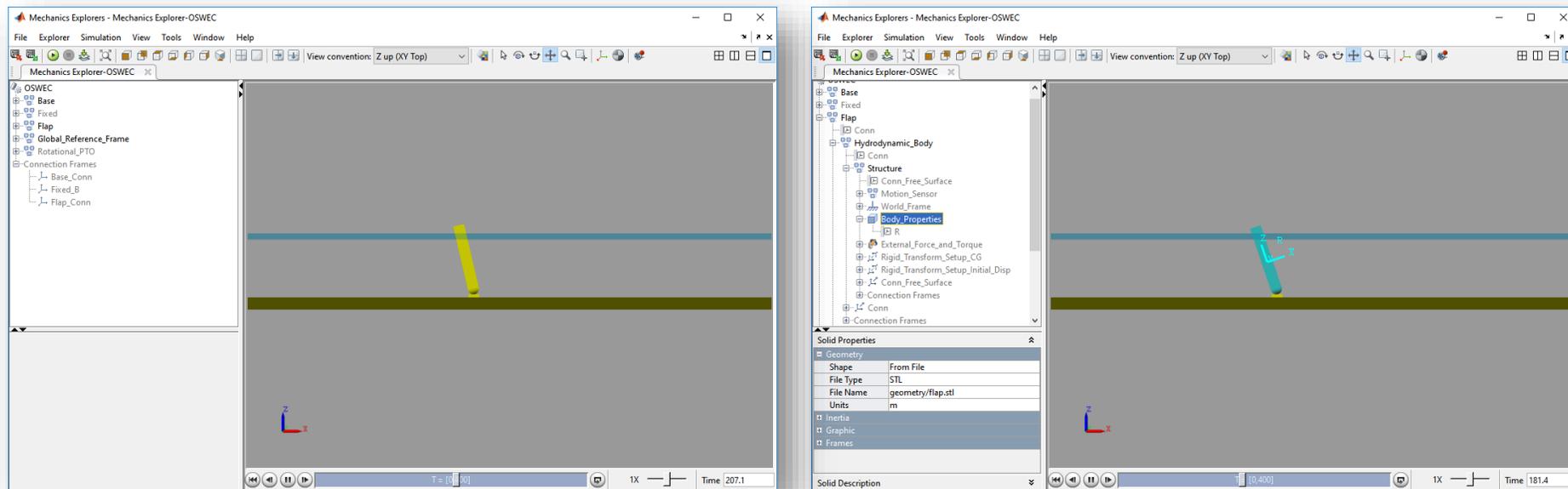
Visualization with Simscape

WEC-Sim **default** visualization uses Simscape

- **simu.explorer = 'on'** turns Mechanics Explorer on (default is on)
- Can't use Simscape visualization with accelerator or rapid-accelerator

Simscape provides animation of WEC-Sim run

- Shows rigid body motion, center of gravity, coordinates, etc.
- Can create video
- Does not show incident, radiated, or diffracted wave surface elevation

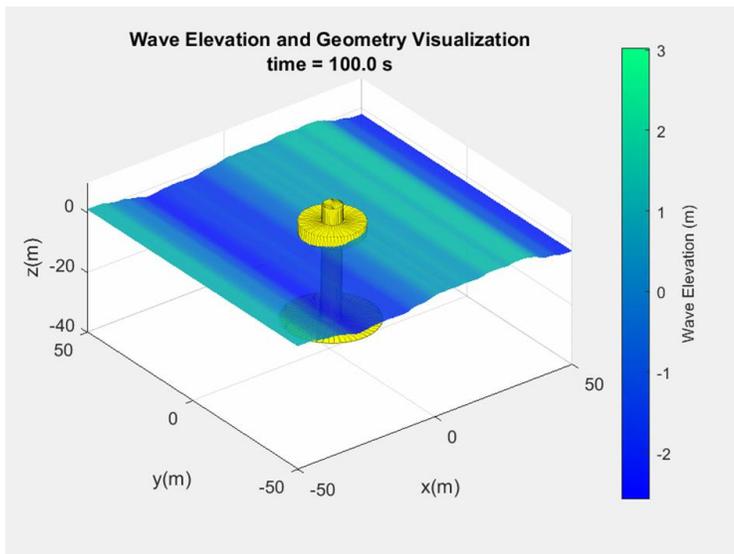


OSWEC Visualization in Simscape

WEC-Sim Visualization Overview

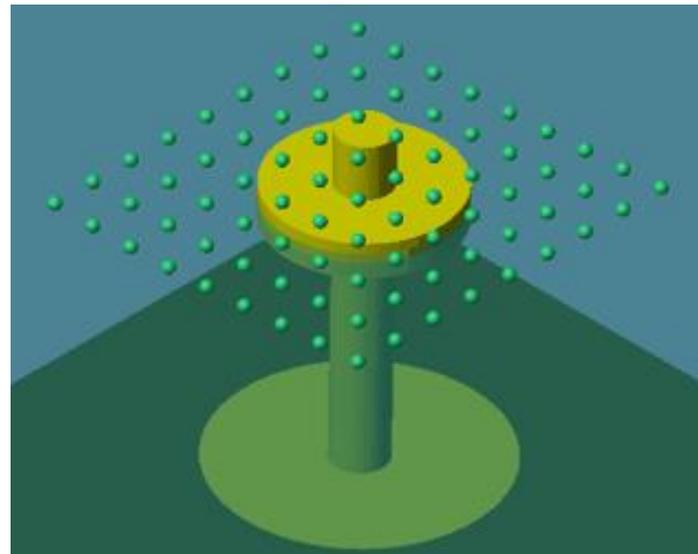
Save Visualization

- MATLAB 3D surface plot
 - Post-processing



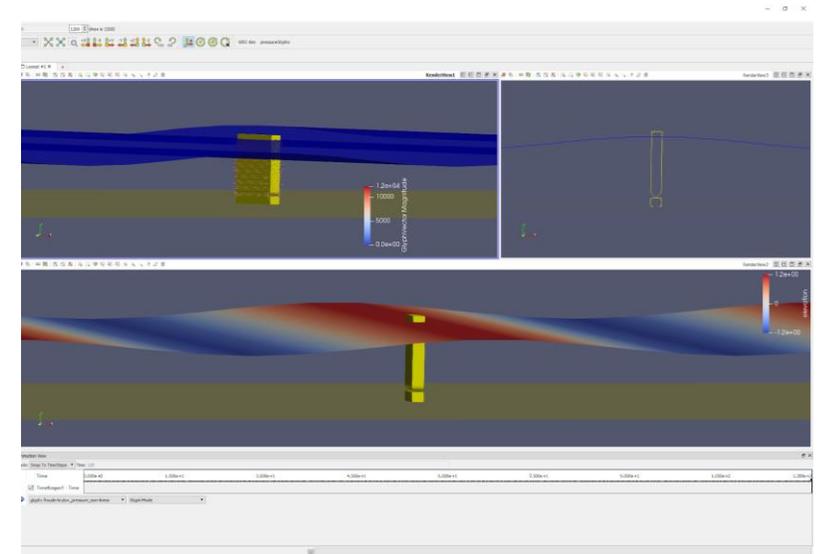
Mechanics Explorer Wave Markers

- In SimScape Mechanics Explorer



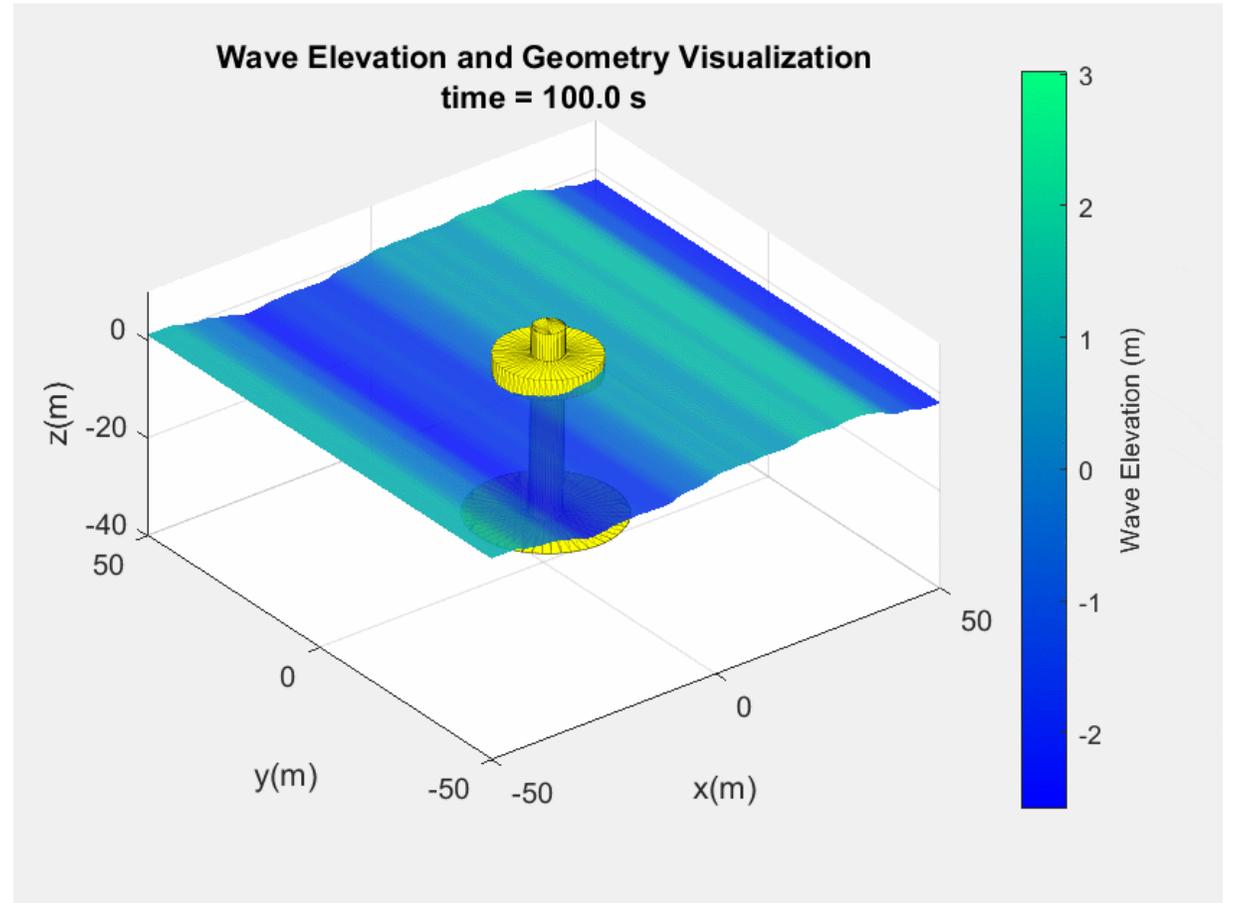
Paraview

- Post-processing
- Outside of WEC-Sim



5 Save Visualization Feature

- Response Class Function (*saveViz*)
- 3-dimensional plot of body(s) and wave motion
- Called by *userDefinedFunctions.m* script
- Shows incident but not radiated or diffracted wave surface elevation
- Options:
 - *timesPerFrame* – # of simulation timesteps per video frame
 - *axisLimits* – x, y, and z-bounds of figure axes
 - *startEndTime* – start and end time of video
 - *saveSetting* – option to save as AVI (0) or GIF (1) file



```
%Save waves and response as video
```

```
output.saveViz(simu,body,waves,...
```

```
    'timesPerFrame',5,'axisLimits',[-150 150 -150 150 -50 20],...
```

```
    'startEndTime',[100 125],'saveSetting',1);
```

HOME PLOTS APPS EDITOR PUBLISH VIEW

New Open Save Compare Print Go To Find Bookmark Refactor Analyze Profiler Run Section Run and Advance Run Step Stop

FILE NAVIGATE CODE ANALYZE SECTION RUN

C:\Users\jtgrasb\Documents\GitHub\WEC-Sim\examples\RM3

Current Folder

Name	Git
geometry	.
hydroData	.
new	.
output	.
slprj	.
vtk	.
elevationData.mat	●
RM3.slx	●
RM3.slx.autosave	.
RM3.slx	.
simulation.log	.
spectrumData.mat	●
userDefinedFunctions.m	●
waveViz.avi	.
wecSimInputFile.m	●

waveViz.avi (Video Clip)

Workspace

Name	Value
------	-------

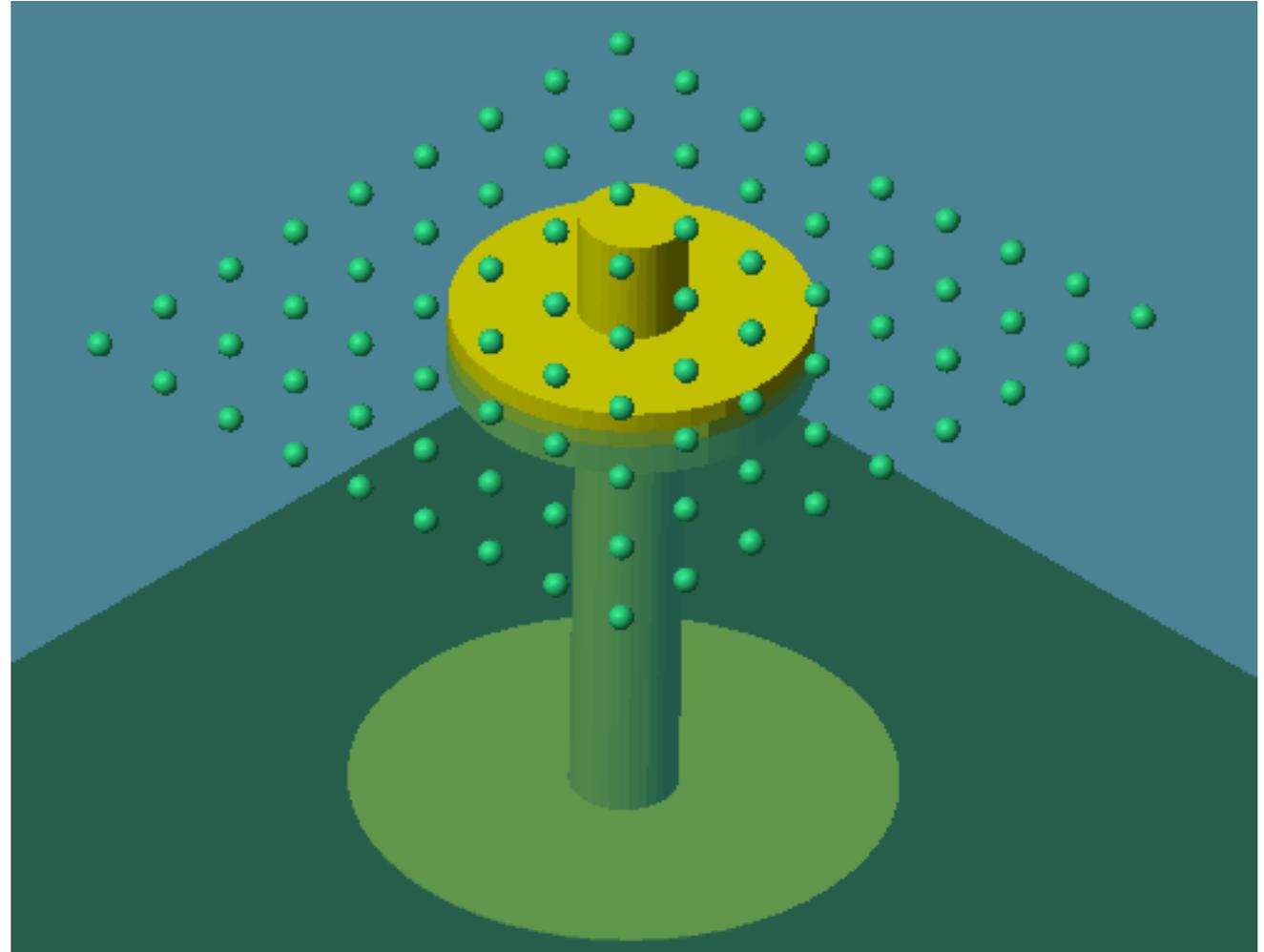
```
Editor - C:\Users\jtgrasb\Documents\GitHub\WEC-Sim\examples\RM3\userDefinedFunctions.m
wecSimInputFile.m userDefinedFunctions.m
1 %Example of user input MATLAB file for post processing
2
3 %Plot waves
4 waves.plotElevation(simu.rampTime);
5 try
6     waves.plotSpectrum();
7 catch
8     end
9
10 %Plot heave response for body 1
11 output.plotResponse(1,3);
12
13 %Plot heave response for body 2
14 output.plotResponse(2,3);
15
16 %Plot heave forces for body 1
17 output.plotForces(1,3);
18
19 %Plot heave forces for body 2
20 output.plotForces(2,3);
21
22 %Save waves and response as video
23 % output.saveViz(simu,body,waves,...
24 % 'timesPerFrame',5,'axisLimits',[-150 150 -150 150 -50 20],...
25 % 'startEndTime',[100 125]);
26
```

Command Window

fx >>

Mechanics Explorer Wave Markers

- Global Reference Frame block adds wave markers to simulation
- Wave markers are specified in the *wecSimInputFile.m*
- Appear in the Simscape Mechanics Explorer
- Shows incident but not radiated or diffracted wave surface elevation
- Options:
 - *waves.marker.location = [X, Y]*
 - location of markers
 - *waves.marker.style = [1], [2], [3]*
 - marker type: 1) sphere, 2) cube, 3) frame
 - *waves.marker.size = [#]*
 - specify marker size in Pixels





Current Folder: C:\Users\jtgrasb\Documents\GitHub\WEC-Sim\examples\RM3

Name	Git
geometry	.
hydroData	.
new	.
output	.
slprj	.
vtk	.
elevationData.mat	●
RM3.slx	●
RM3.slxc	●
simulation.log	.
spectrumData.mat	●
userDefinedFunctions.m	●
waveViz.avi	.
wecSimInputFile.m	●

workspaceViz.avi (Video Clip)

Name	Value
body	1x2 bodyClass
constraint	1x1 constrain...
e	1x1 MSLExcep...
output	1x1 response...
outputFile	'C:\Users\jtr...
pto	1x1 ptoClass
simu	1x1 simulatio...
waves	1x1 waveClass

```

wecSimInputFile.m
1 %% Simulation Data
2 simu = simulationClass(); % Initialize Simulation Class
3 simu.simMechanicsFile = 'RM3.slx'; % Specify Simulink Model File
4 simu.mode = 'normal'; % Specify Simulation Mode ('normal','accelerator','rapid-accelerator')
5 simu.explorer = 'off'; % Turn SimMechanics Explorer (on/off)
6 simu.startTime = 0; % Simulation Start Time [s]
7 simu.rampTime = 100; % Wave Ramp Time [s]
8 simu.endTime = 400; % Simulation End Time [s]
9 simu.solver = 'ode4'; % simu.solver = 'ode45' for fixed step & simu.solver = 'ode45' for variable step
10 simu.dt = 0.1; % Simulation time-step [s]
11
12 %% Wave Information
13 %% noWaveCIC, no waves with radiation CIC
14 waves = waveClass('noWaveCIC'); % Initialize Wave Class and Specify Type
15
16 %% Regular Waves
17 waves = waveClass('regular'); % Initialize Wave Class and Specify Type
18 waves.height = 2.5; % Wave Height [m]
19 waves.period = 8; % Wave Period [s]
20
21 %% Regular Waves with CIC
22 waves = waveClass('regularCIC'); % Initialize Wave Class and Specify Type
23 waves.height = 2.5; % Wave Height [m]
24 waves.period = 8; % Wave Period [s]
25
26 %% Irregular Waves using PM Spectrum
27 waves = waveClass('irregular'); % Initialize Wave Class and Specify Type
28 waves.height = 2.5; % Significant Wave Height [m]
29 waves.period = 8; % Peak Period [s]
30 waves.spectrumType = 'PM'; % Specify Wave Spectrum Type
31
32 %% Irregular Waves using JS Spectrum with Equal Energy and Seeded Phase
33 waves = waveClass('irregular'); % Initialize Wave Class and Specify Type
34 waves.height = 2.5; % Significant Wave Height [m]
35 waves.period = 8; % Peak Period [s]

```

Command Window

```

List of Constraint(s): Number of Constraints = 1

***** Constraint Name: Constraint1 *****

Simulating the WEC device defined in the SimMechanics model C:\Users\jtgrasb\Documents\GitHub\WEC-Sim\examples\RM3\RM3.slx...
Elapsed time is 0.146083 seconds.
Elapsed time is 4.647332 seconds.

Post-processing and saving...
Elapsed time is 0.466586 seconds.
fx >>

```

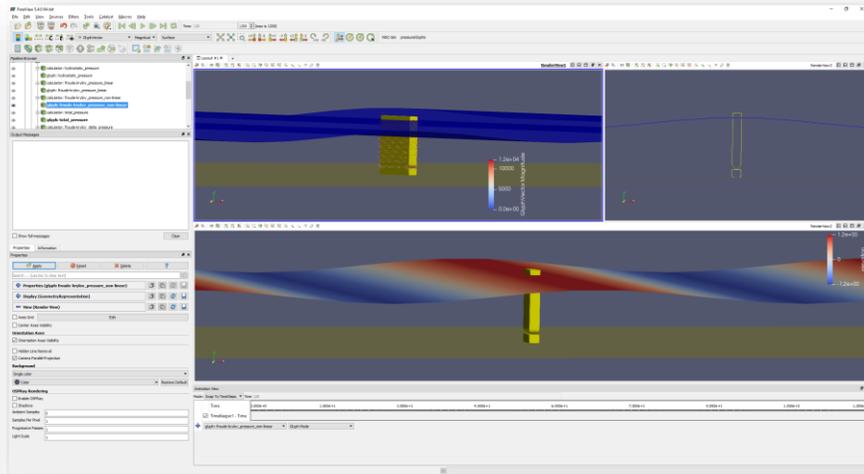
Visualization with ParaView (v5.11.1)

ParaView is an open-source, data analysis and visualization application.

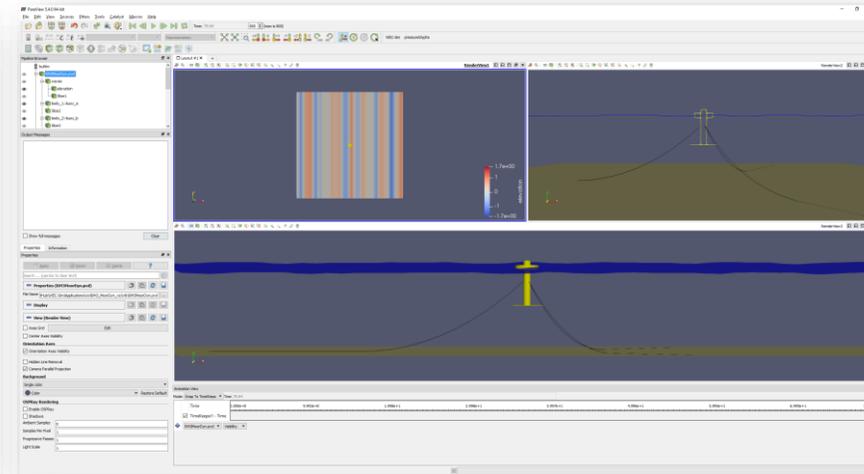
- Pro: WEC-Sim uses ParaView to create videos, visualize wave field, cell-by-cell non-linear forces, and other features.
- Con: Takes *a lot* more time to run

Examples using ParaView for WEC-Sim data visualization available on the applications repository:

https://github.com/WEC-Sim/WEC-Sim_Applications



**OSWEC Nonlinear Viz
with ParaView**



**RM3 MoorDyn Viz
with ParaView**

Installing ParaView

Download and Install ParaView

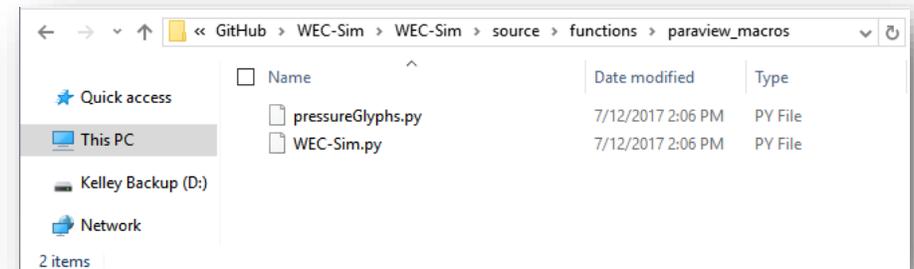
- For ParaView visualization:
<https://www.paraview.org/>
- WEC-Sim is compatible with v5.11.1

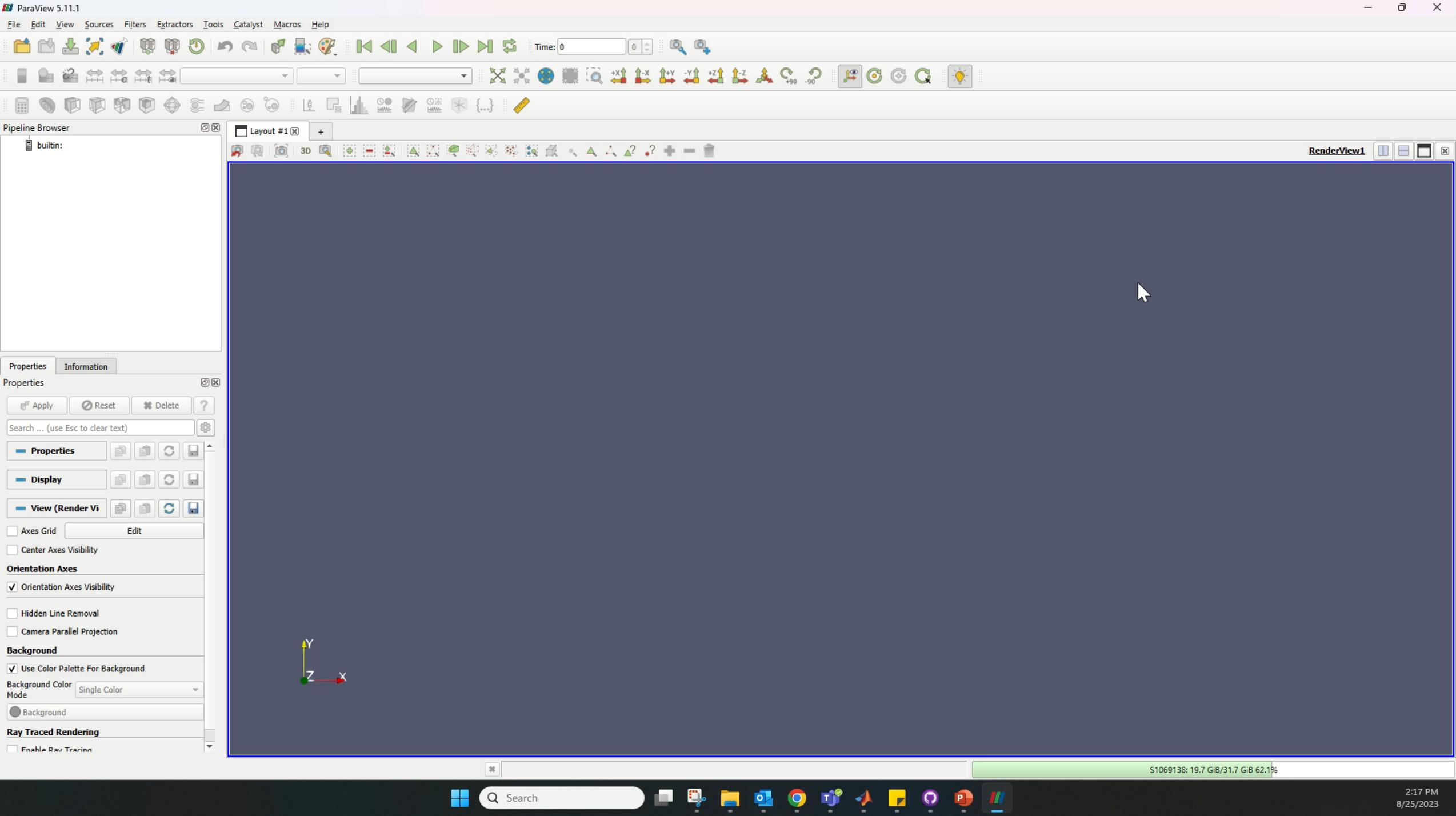
Download and Install Python

- For ParaView macros:
https://www.paraview.org/Wiki/ParaView_and_Python

Install the WEC-Sim macros in ParaView

- Open ParaView
- Click on **'Macros => Add new macro'**
- Navigate to the ***WEC-Sim/source/functions/paraview*** directory
- Select ***WEC-Sim.py*** macro file and click 'OK'
- Select ***pressureGlyphs.py*** macro file and click 'OK'





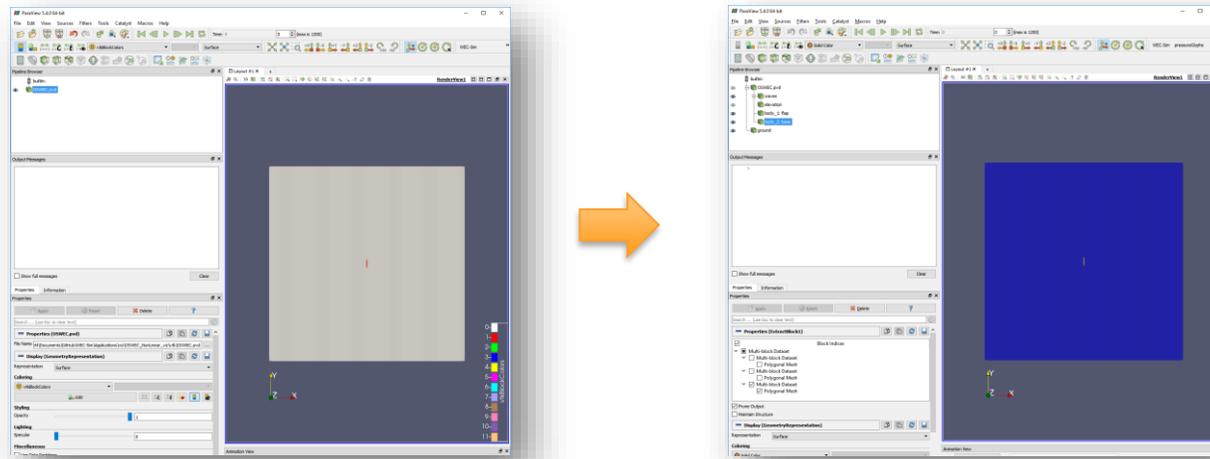
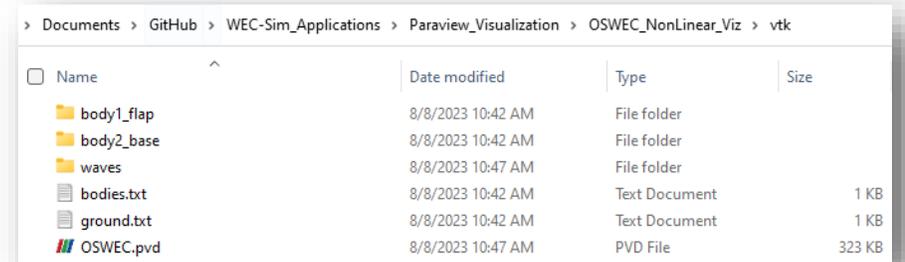
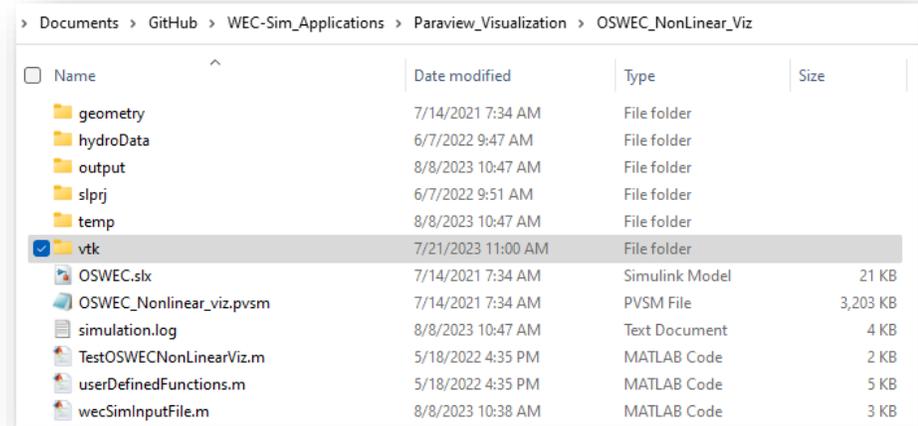
WEC-Sim ParaView Visualization

'**simu.paraview.option = 1**' turns Paraview on

- creates a */vtk* directory in the WEC-Sim case directory
- saves ParaView data files

Open the ***.pvd** file to view data visualization in ParaView and Click '**Apply**'

Select model in pipeline and run the '**WEC-Sim**' macro to import default WEC-Sim colors and orientation



Nonlinear Hydro, videos, etc

For nonlinear hydro, select body in pipeline and run the **'pressureGlyphs'** macro, adds glyphs for:

- Hydrostatic Pressure
- Linear Froude-Krylov pressure
- Non-linear Froude-Krylov pressure
- Total pressure (hydrostatic + non-linear Froude-Krylov)
- Froude-Krylov delta (non-linear minus linear)

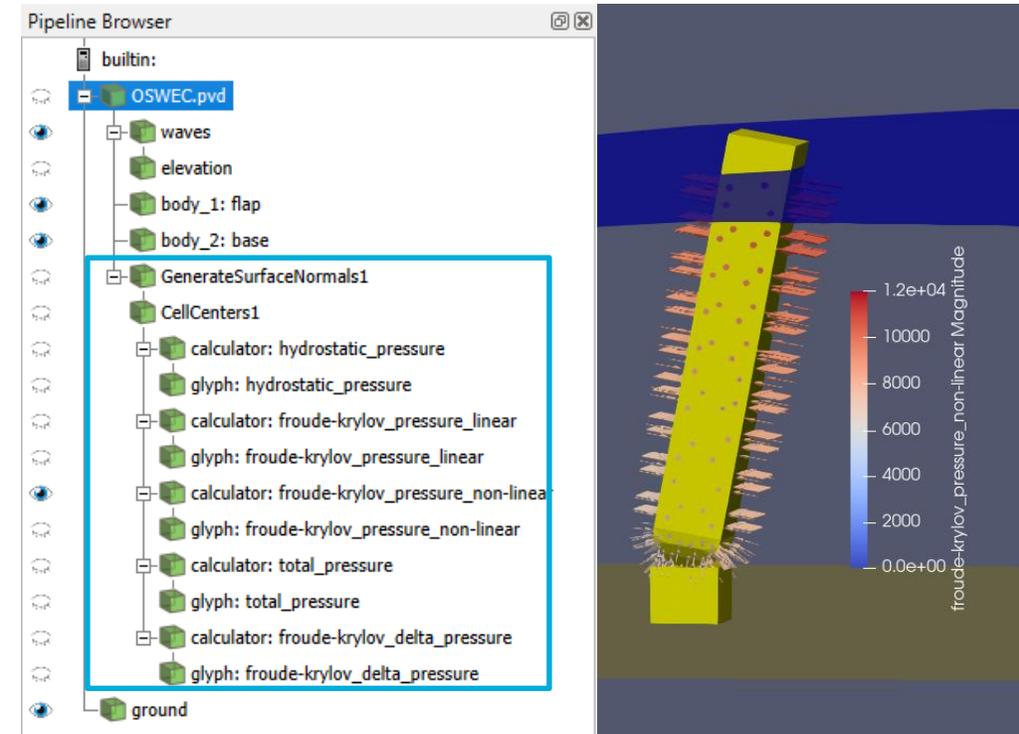
Add multiple views, slices and data filters

Save State to recreate viz for many runs

- 'File => Save State', saves state as a **.pvsm*

Create video file

- 'File => Save Animation', saves video as **.avi*



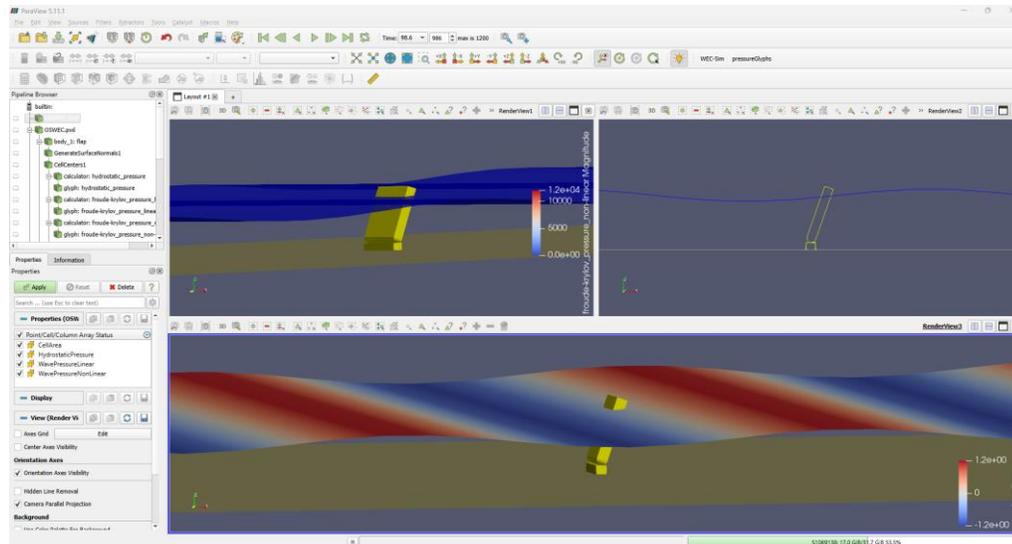
Visualization with ParaView Examples

Examples using ParaView for WEC-Sim data visualization available on the Applications repository:

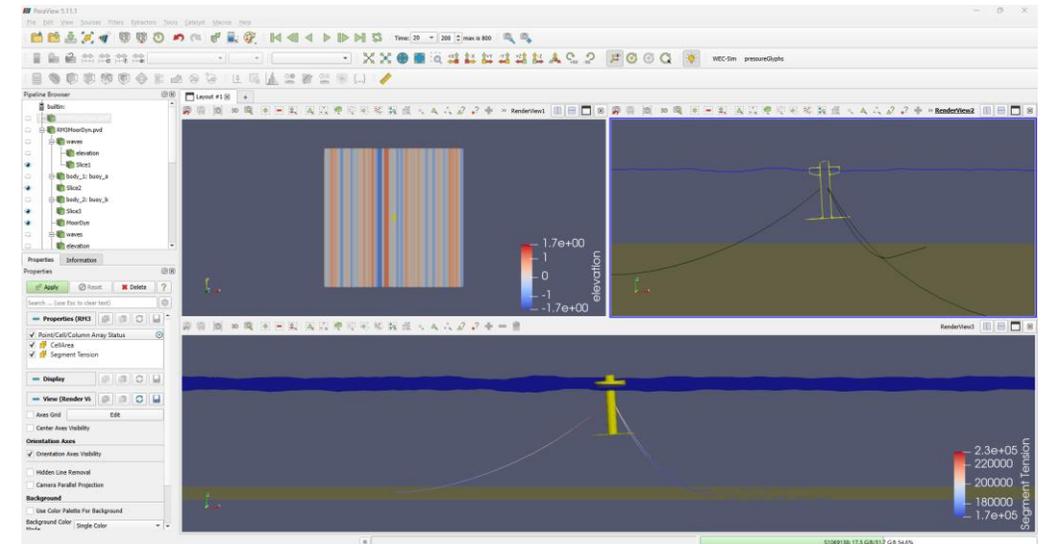
https://github.com/WEC-Sim/WEC-Sim_Applications

'Viz' Examples include

- OSWEC with non-linear hydro
- RM3 coupled with MoorDyn



OSWEC_Nonlinear_Viz



RM3_MoorDyn_Viz

ParaView for RM3 coupled with MoorDyn

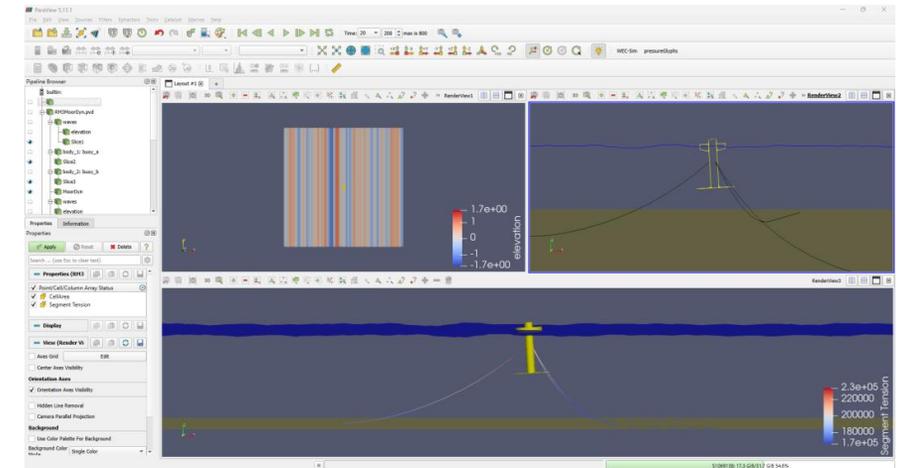
```
wecSimInputFile.m +
1 % Simulation Data
2 simu = simulationClass();
3 simu.simMechanicsFile = 'RM3MoorDyn.slx'; % Location of Simulink Model File with MoorDyn
4 simu.mode='accelerator';
5 simu.explorer = 'off';
6 simu.rampTime = 0;
7 simu.endTime=80;
8 simu.dt = 0.01;
9 simu.dtOut = 0.1; % Specifies output time-step
10 simu.cicDt = 0.05;
11 simu.solver = 'ode45'; % Runs WEC-Sim with variable time-step
12 simu.paraview.option = 1; % Saves data to *.vtp for Paraview
13 simu.domainSize = 300; % Changes default domain size
14
15 %% Wave Information
16 % Irregular Waves using PM Spectrum with Convolution Integral Calculation
17 waves = waveClass('irregular'); % Create the Wave Variable and Specify Type
18 waves.height = 2;
19 waves.period = 8;
20 waves.spectrumType = 'JS';
21 waves.bem.option = 'Traditional';
22 waves.viz.numPointsX = 1000;
23 waves.viz.numPointsY = 2;
24
25 %% Body Data ***
39 %% PTO and Constraint Parameters ***
50 %% Mooring
51 % Moordyn
52 mooring(1) = mooringClass('mooring'); % Initialize mooringClass
53 mooring(1).moorDynLines = 3; % Specify number of lines
54 mooring(1).moorDynNodes = [21 21 21]; % Specify number of nodes per line
55
```

Set data dtOut for ParaView viz

Saves data for ParaView viz

Wave field viz options

Sets ParaView Viz lines and nodes



RM3_MoorDyn_Viz

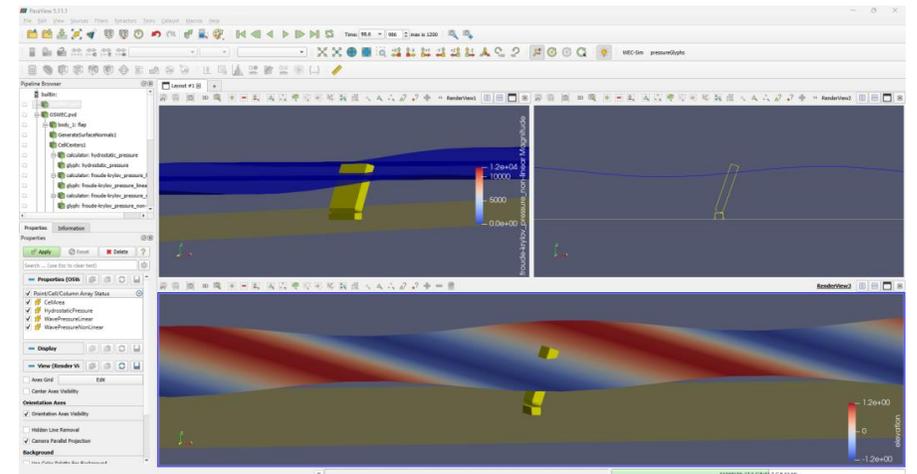
ParaView for OSWEC with nonlinear hydrodynamics

```
wecSimInputFile.m x +
1 %% Simulation Data
2 simu = simulationClass();
3 simu.simMechanicsFile = 'OSWEC.slx';
4 simu.mode='accelerator';
5 simu.explorer = 'off';
6 simu.startTime = 0;
7 simu.rampTime = 8*5;
8 simu.endTime=8*15;
9 simu.dt = 0.1;
10 simu.pressure = 1; % Saves pressures data for Paraview
11 simu.paraview.option = 1; % Saves data to *.vtp for Paraview
12
13 %% Wave Information
14 % Regular Waves
15 waves = waveClass('regular');
16 waves.height = 2.5;
17 waves.period = 8;
18 %waves.viz.numPointsX = 1000; % wave plane discretization: # X points [default 50]
19 %waves.viz.numPointsY = 100; % wave plane discretization: # Y points [default 50]
20
21 %% Body Data
22 % Flap
23 body(1) = bodyClass('hydroData/oswec.h5');
24 body(1).geometryFile = 'geometry/flap.stl';
25 body(1).mass = 127000;
26 body(1).inertia = [1.85e6 1.85e6 1.85e6];
27 % body(1).viz.color = [1 1 0]; % [RGB] body color (default [1 1 0])
28 % body(1).viz.opacity = 1; % body opacity (default 1)
29 body(1).nonlinearHydro = 2; % Turns non-linear hydro on
30
31 % Base (Non-hydro Body)
32 body(2) = bodyClass(''); % Initialize bodyClass without an *.h5 file
33 body(2).geometryFile = 'geometry/base.stl'; % Geometry File
34 body(2).nonHydro = 1; % Turn non-hydro body on
35 body(2).name = 'base'; % Specify body name
36 body(2).mass = 999; % Specify Mass
37 body(2).inertia = [1 1 1]; % Specify MOI
38 body(2).centerGravity = [0 0 -10.9]; % Specify Cg
39 body(2).centerBuoyancy = [0 0 0]; % Specify Cb
40 body(2).volume = 0; % Specify Displaced Volume
41
42 %% PTO and Constraint Parameters
```

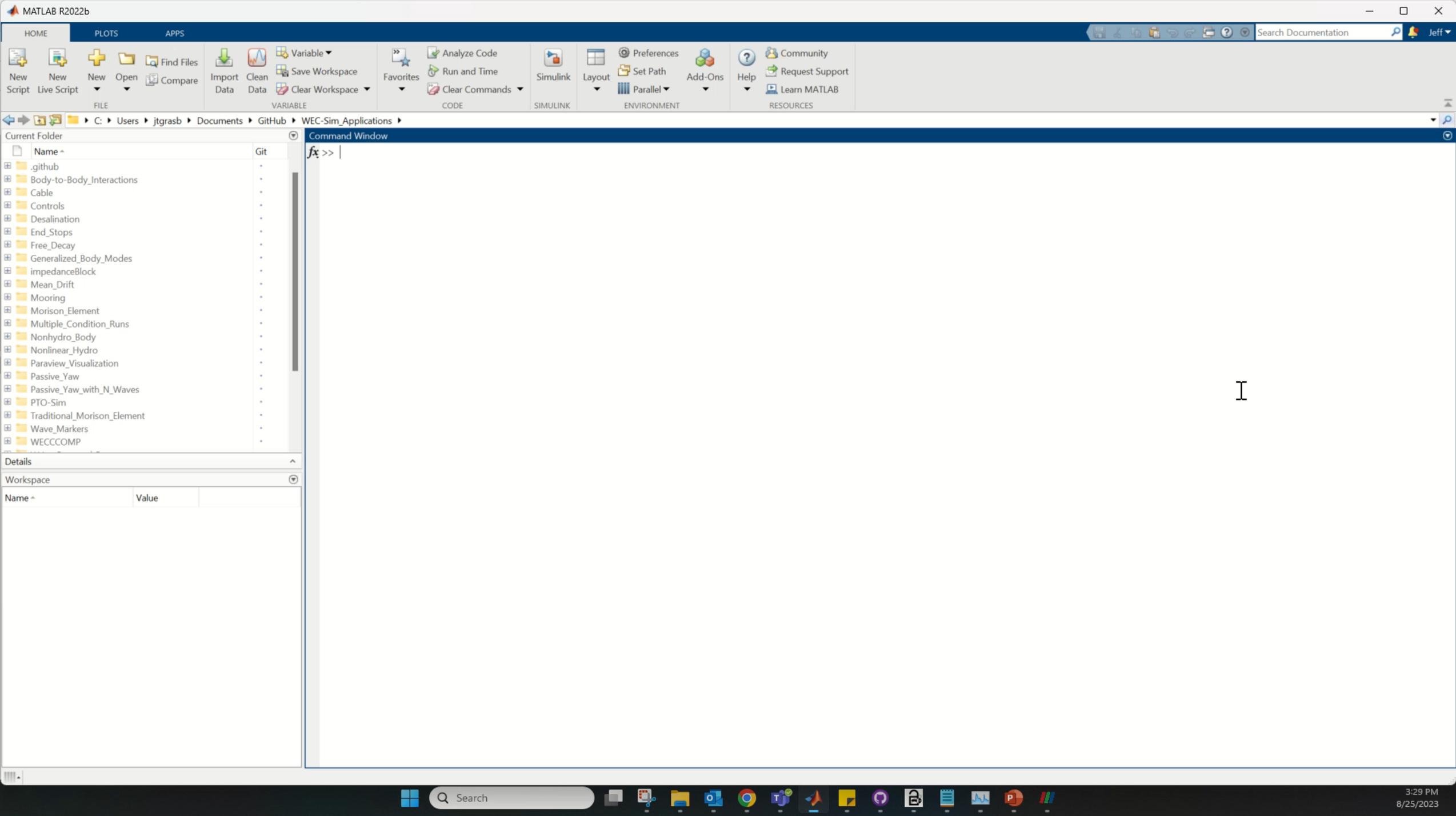
Saves nonlinear force data for ParaView viz
Saves data for ParaView viz

Wave field viz options

Body viz options



OSWEC_Nonlinear_Viz



Thank you

For more information please visit the WEC-Sim website:

<http://wec-sim.github.io/WEC-Sim>

If you have questions on this presentation please reach out to any of the WEC-Sim Developers on GitHub:

<https://github.com/WEC-Sim/WEC-Sim>



Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

This work was authored in part by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308.

Funding provided by the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Water Power Technologies Office. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

