Barracuda Virtual Reactor Users' Conference 2024 Advanced Training Workshop

Pramod Bangalore and Sam Clark

CPFD Software

June 26, 2026

Outline

Example Case Overview

Barracuda Virtual Reactor Feature Highlights

10:30 – 11:00 am: Break

Basic Post-Processing

Advanced Post-Processing

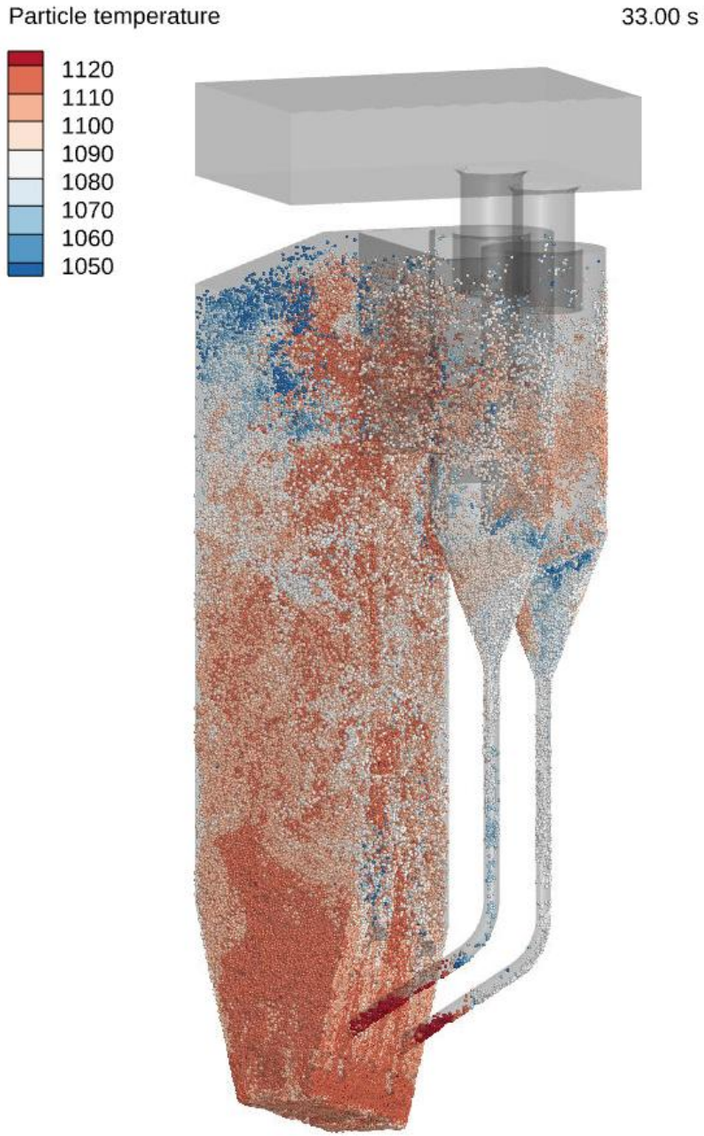
Q&A

Example Case Overview

Application Model: Industrial Scale Circulating Fluidized Bed Combustor was used as a starting point

Modifications made for this training workshop:

- Coarser grid definition to increase simulation speed
- Particle resolution distribution (PRD) enabled
- Flux planes added to analyze flow into and out of cyclone



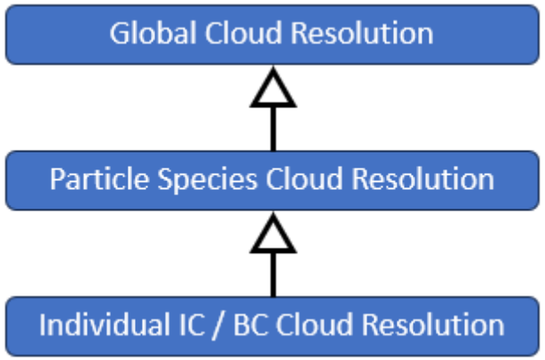
Reviewing an Already-Run Initial Case

Cloud Resolution Settings

Added in version 24.1

Inheritance-based model for specifying cloud resolution

- Easier to get consistent cloud resolution at ICs and BCs
- Simplified approach for setting up Virtual Reactor projects



Clouds per Cell Definition

In many simulations, *Clouds per Cell* is the most convenient basis for defining cloud resolution. It is based on the average cell volume in the system:

$$\text{Clouds per cell} = \frac{\text{Number of clouds at } \theta_p = 1}{\text{Cell of average volume}} \tag{19.1}$$

This definition is useful because it causes the cloud resolution to naturally scale with the cell resolution, i.e. models with higher real cell counts will automatically use higher numbers of clouds. A frequently used rule of thumb for determining a good value of *Clouds per cell* at an IC or BC is:

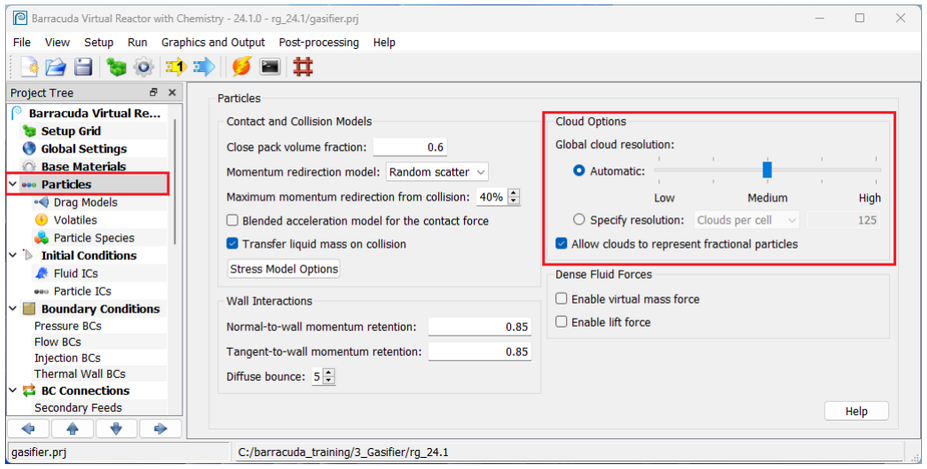
$$\text{Clouds per cell} = \frac{50}{\theta_p} \tag{19.2}$$

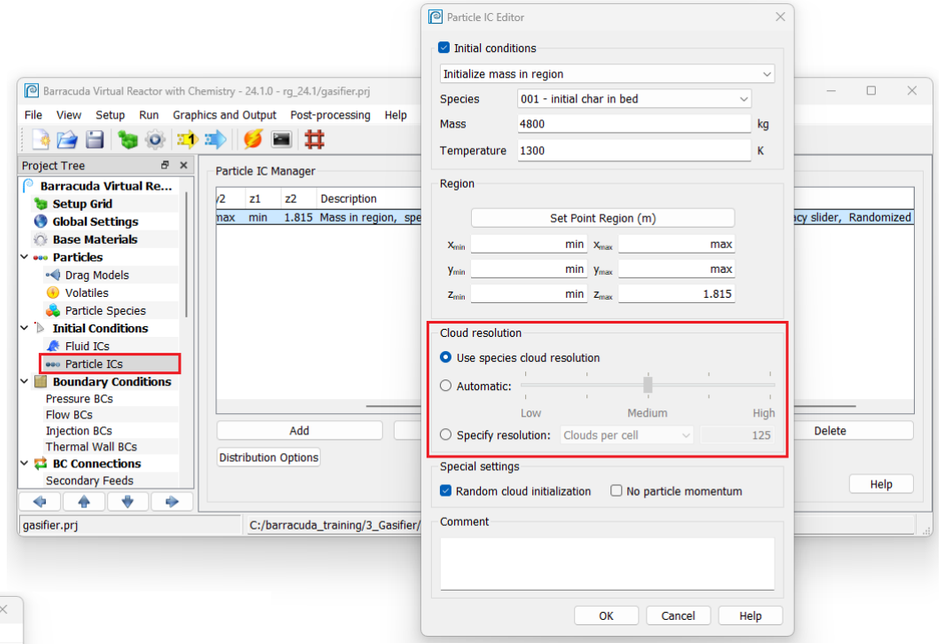
where:

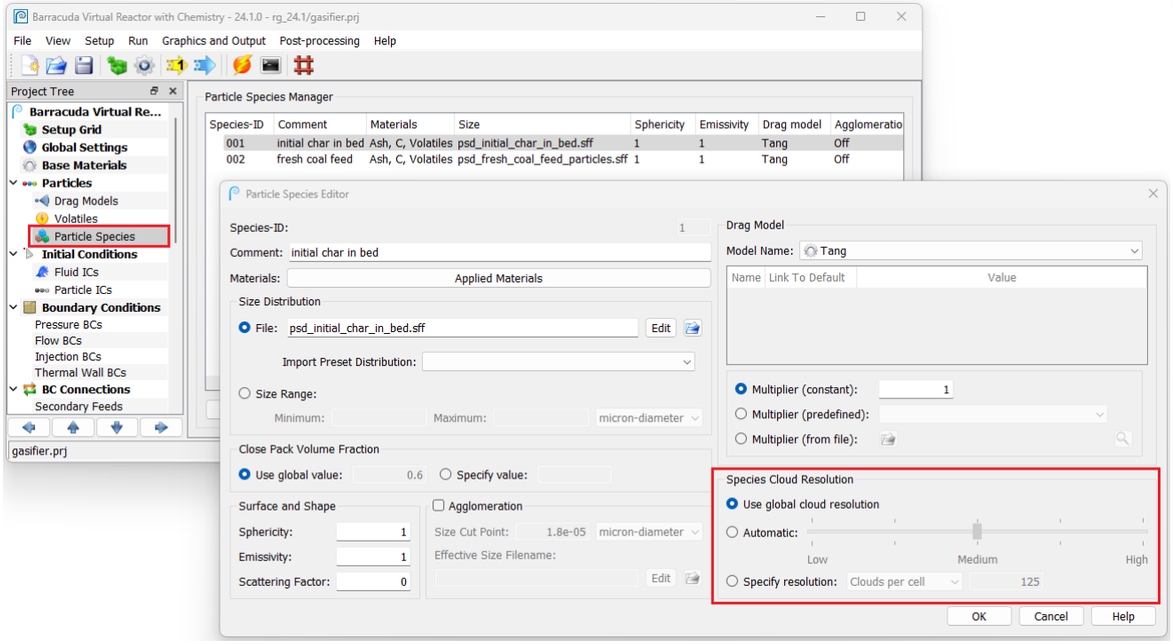
θ_p is the volume fraction of particles at the IC or BC being specified

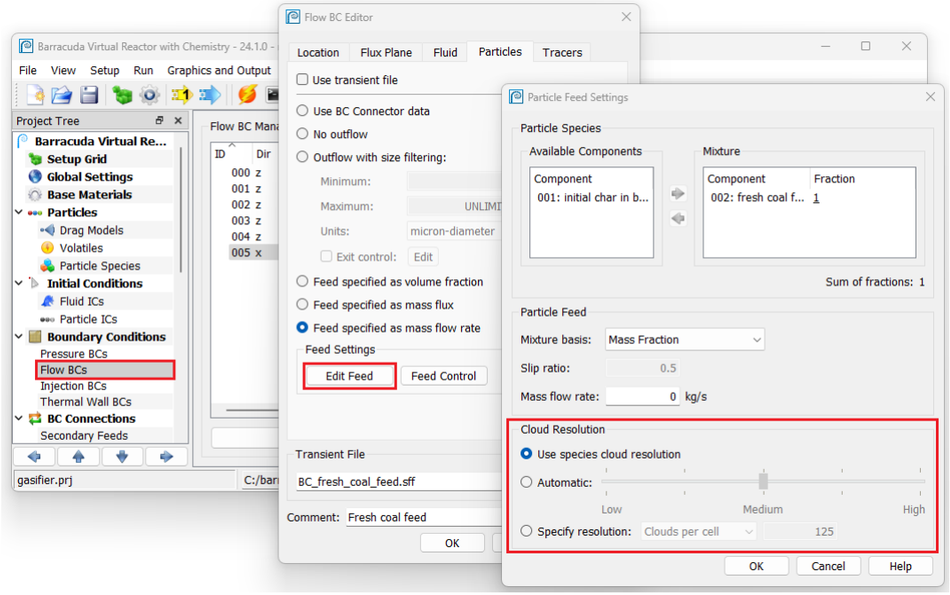
In all locations where the Cloud Resolution GUI pattern is used, the *Automatic* slider bar levels correspond to the following values of *Clouds per cell*:

Slider Bar Setting	*Clouds per Cell* Value	Corresponding θ_p in (19.2)
Low	50	1
Low-Medium	80	0.625
Medium	125	0.4
Medium-High	250	0.2
High	1000	0.05


Barracuda Virtual Reactor with Chemistry - 24.1.0 - rg_24.1/gasifier.prj
File View Setup Run Graphics and Output Post-processing Help
Project Tree
Barracuda Virtual Re...
Setup Grid
Global Settings
Base Materials
Particles
Drag Models
Volatiles
Particle Species
Initial Conditions
Fluid ICs
Particle ICs
Boundary Conditions
Pressure BCs
Flow BCs
Injection BCs
Thermal Wall BCs
BC Connections
Secondary Feeds
Particles
Contact and Collision Models
Close pack volume fraction: 0.6
Momentum redirection model: Random scatter
Maximum momentum redirection from collision: 40%
Blended acceleration model for the contact force
Transfer liquid mass on collision
Stress Model Options
Wall Interactions
Normal-to-wall momentum retention: 0.85
Tangent-to-wall momentum retention: 0.85
Diffuse bounce: 5
Cloud Options
Global cloud resolution:
Automatic:
Low Medium High
Specify resolution: Clouds per cell 125
Allow clouds to represent fractional particles
Dense Fluid Forces
Enable virtual mass force
Enable lift force
Help
gasifier.prj
C:/barracuda_training/3_Gasifier/rg_24.1

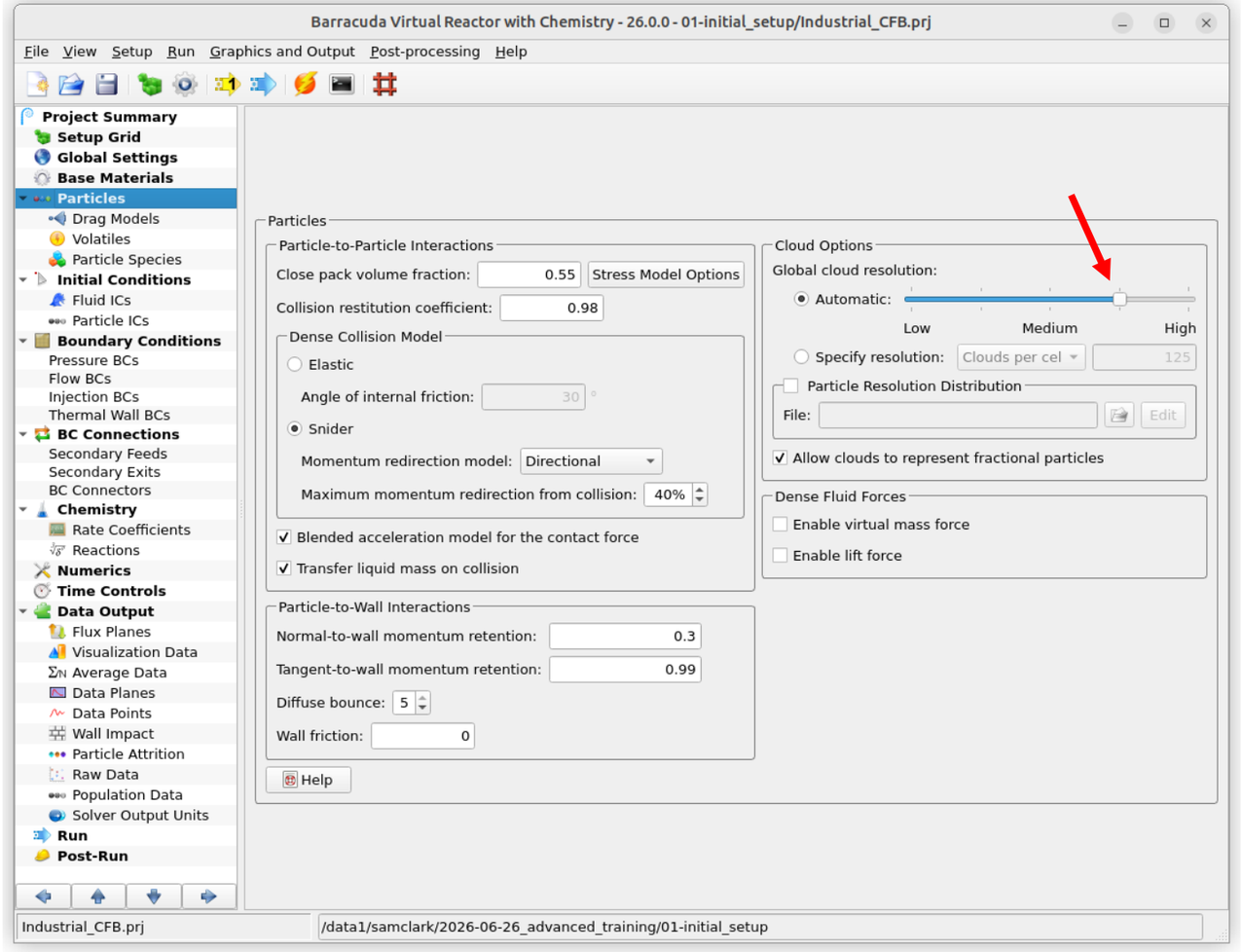

Particle IC Editor
Initial conditions
Initialize mass in region
Species 001 - initial char in bed
Mass 4800 kg
Temperature 1300 K
Region
Set Point Region (m)
Cloud resolution
Use species cloud resolution
Automatic:
Low Medium High
Specify resolution: Clouds per cell 125
Special settings
Random cloud initialization
No particle momentum
Comment
OK Cancel Help
Particle IC Manager
Add
Distribution Options
Delete
Help


Particle Species Manager
Species-ID Comment Materials Size Sphericity Emissivity Drag model Agglomeratio
001 initial char in bed Ash, C, Volatiles psd_initial_char_in_bed.sff 1 1 Tang Off
002 fresh coal feed Ash, C, Volatiles psd_fresh_coal_feed_particles.sff 1 1 Tang Off
Particle Species Editor
Species-ID: 1
Comment: initial char in bed
Materials: Applied Materials
Size Distribution
File: psd_initial_char_in_bed.sff Edit
Import Preset Distribution:
Size Range:
Minimum: Maximum: micron-diameter
Close Pack Volume Fraction
Use global value: 0.6 Specify value:
Surface and Shape
Sphericity: 1
Emissivity: 1
Scattering Factor: 0
Agglomeration
Size Cut Point: 1.8e-05 micron-diameter
Effective Size Filename:
Drag Model
Model Name: Tang
Name Link To Default Value
Multiplier (constant): 1
Multiplier (predefined):
Multiplier (from file):
Species Cloud Resolution
Use global cloud resolution
Automatic:
Low Medium High
Specify resolution: Clouds per cell 125
OK Cancel Help


Flow BC Editor
Location Flux Plane Fluid Particles Tracers
Use transient file
Use BC Connector data
No outflow
Outflow with size filtering:
Minimum:
Maximum:
Units: micron-diameter
Exit control: Edit
Feed specified as volume fraction
Feed specified as mass flux
Feed specified as mass flow rate
Feed Settings
Edit Feed Feed Control
Transient File
BC_fresh_coal_feed.sff
Comment: Fresh coal feed
OK
Particle Feed Settings
Particle Species
Available Components
Component
001: initial char in b...
Mixture
Component Fraction
002: fresh coal f... 1
Sum of fractions: 1
Particle Feed
Mixture basis: Mass Fraction
Slip ratio: 0.5
Mass flow rate: 0 kg/s
Cloud Resolution
Use species cloud resolution
Automatic:
Low Medium High
Specify resolution: Clouds per cell 125
OK Cancel Help

Cloud Resolution Settings for Current Example

Set to medium-high in the Particles main window because bed is fairly shallow and fluidization regime is turbulent



Particle Resolution Distribution (PRD) Settings

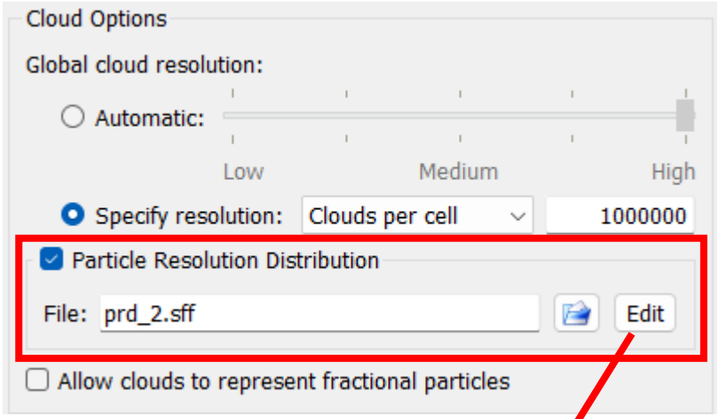
Added in version 26.0

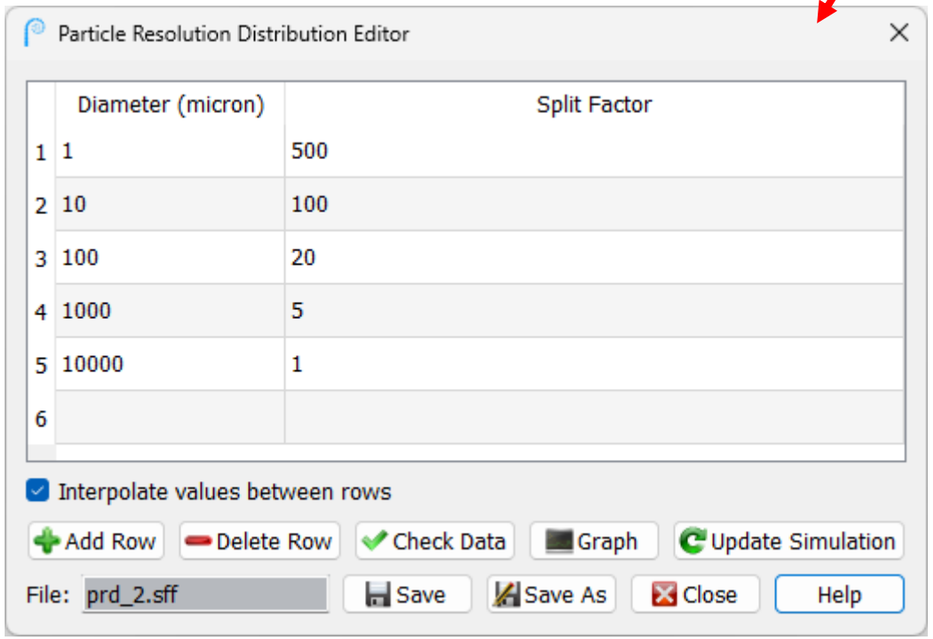
Enables control over how clouds are allocated based on particle size

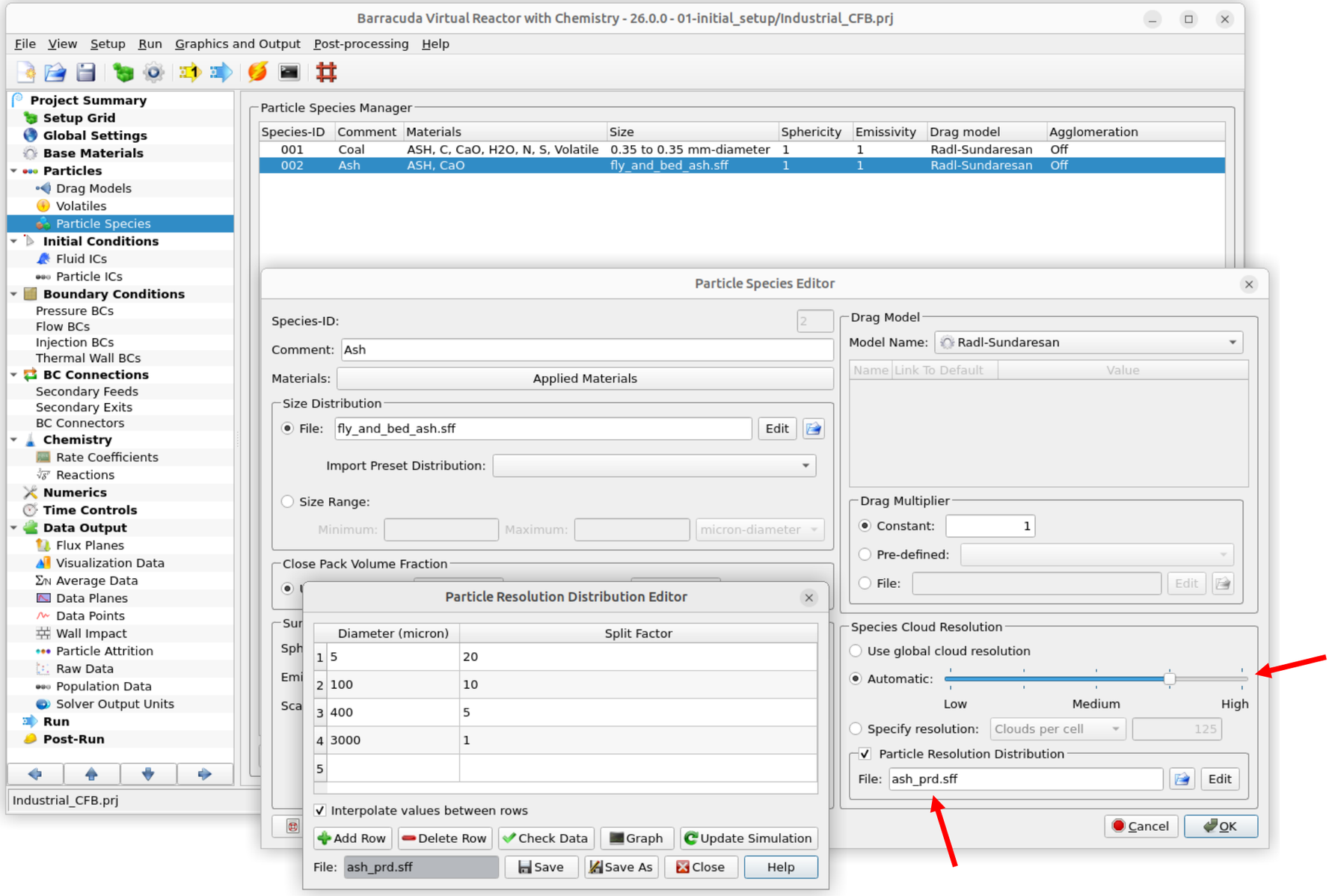
Most useful for particle species with wide particle size distributions (PSDs)

When PRD is used:

- Split factor controls relative number of clouds used for corresponding particle size
- PSDs are not changed
- Total number of clouds is not changed






Barracuda Virtual Reactor with Chemistry - 26.0.0 - 01-initial_setup/Industrial_CFB.prj
File View Setup Run Graphics and Output Post-processing Help
Project Summary
Setup Grid
Global Settings
Base Materials
Particles
Drag Models
Volatiles
Particle Species
Initial Conditions
Fluid ICs
Particle ICs
Boundary Conditions
Pressure BCs
Flow BCs
Injection BCs
Thermal Wall BCs
BC Connections
Secondary Feeds
Secondary Exits
BC Connectors
Chemistry
Rate Coefficients
Reactions
Numerics
Time Controls
Data Output
Flux Planes
Visualization Data
Average Data
Data Planes
Data Points
Wall Impact
Particle Attrition
Raw Data
Population Data
Solver Output Units
Run
Post-Run
Industrial_CFB.prj
Particle Species Manager
Species-ID Comment Materials Size Sphericity Emissivity Drag model Agglomeration
001 Coal ASH, C, CaO, H2O, N, S, Volatile 0.35 to 0.35 mm-diameter 1 1 Radl-Sundaresan Off
002 Ash ASH, CaO fly_and_bed_ash.sff 1 1 Radl-Sundaresan Off
Particle Species Editor
Species-ID: 2
Comment: Ash
Materials: Applied Materials
Size Distribution
File: fly_and_bed_ash.sff Edit
Import Preset Distribution:
Size Range:
Minimum: Maximum: micron-diameter
Close Pack Volume Fraction
Drag Model
Model Name: Radl-Sundaresan
Name Link To Default Value
Drag Multiplier
Constant: 1
Pre-defined:
File: Edit
Species Cloud Resolution
Use global cloud resolution
Automatic:
Low Medium High
Specify resolution: Clouds per cell 125
Particle Resolution Distribution
File: ash_prd.sff Edit
Cancel OK
Particle Resolution Distribution Editor
Diameter (micron) Split Factor
1 5 20
2 100 10
3 400 5
4 3000 1
5
Interpolate values between rows
Add Row Delete Row Check Data Graph Update Simulation
File: ash_prd.sff Save Save As Close Help

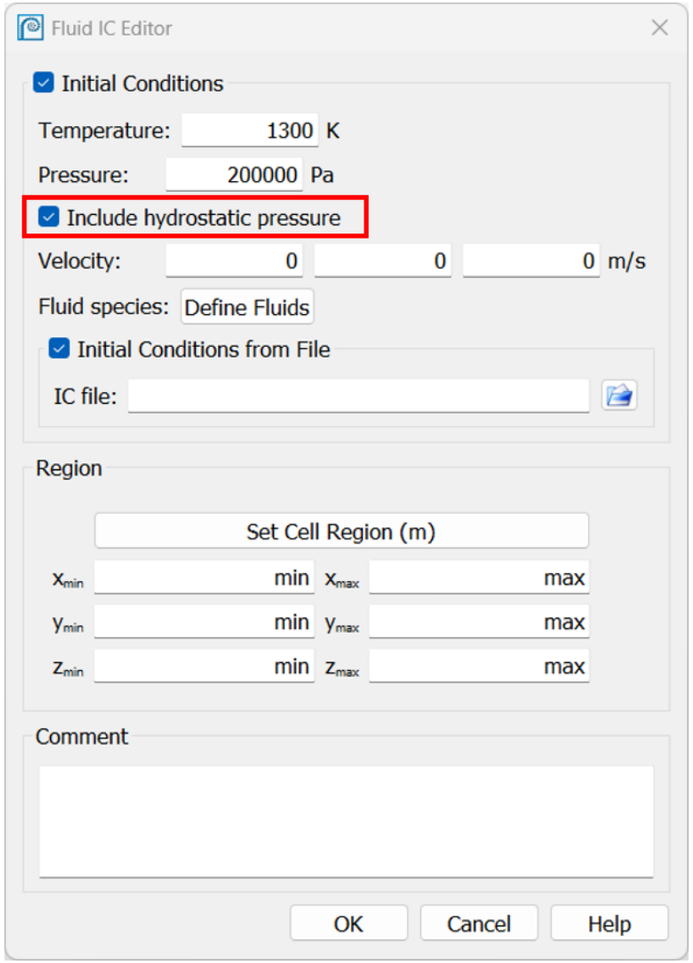
Hydrostatic Pressure Settings for Pressure BC and Fluid IC

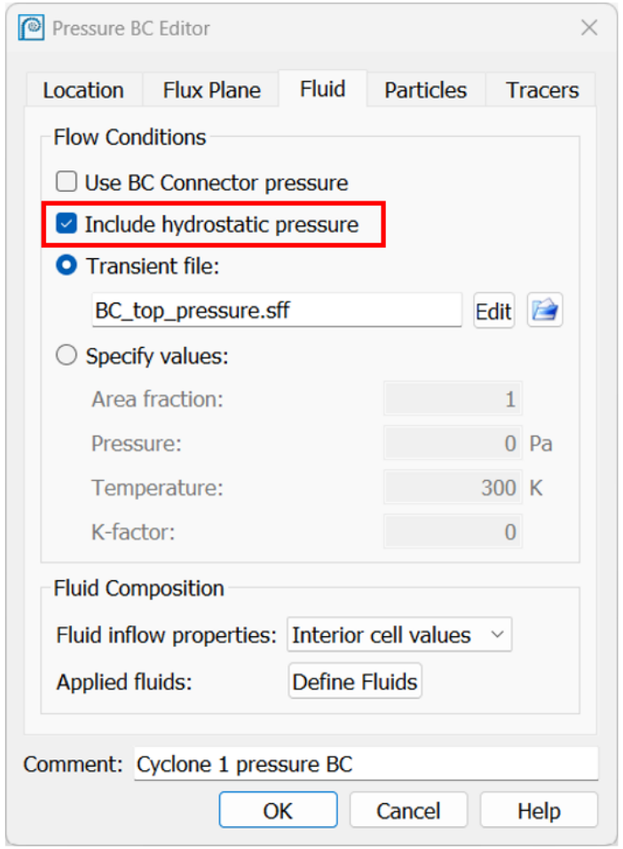
Added in release 25.0

For Fluid ICs enabling hydrostatic pressure can help to avoid large initial in- and out-flows of fluid

For Pressure BCs enabling hydrostatic pressure can help to avoid artificial recirculation at boundaries that are not normal to gravity

Highly recommended for liquid domain simulations





Thermal Wall BC with Heat Transfer Resistance

Added in release 24.1

Allows specification of outside ambient temperature in combination with heat transfer resistance

$$\mathrm{R} = \frac{\Delta T}{\phi_q}$$

where:

R is the resistance to heat transfer (R-value) in units of $\mathrm{m}^2 \cdot \mathrm{K}/\mathrm{W}$
ΔT is the temperature difference between the thermal wall surface and the external fluid in units of K
ϕ_q is the heat flux through the thermal wall in units of W/m^2

Example of Calculating a Thermal Wall BC Resistance Value

Continuing with the example of an FCC regenerator, there are two dominant components of resistance as heat is transferred from the inside of the vessel to the ambient air:

1. **Conduction through the vessel wall:** let us assume, for this example, that the vessel is constructed of steel and lined on the inside with 4 inches (0.1 m) of refractory. It is often acceptable to neglect the thermal resistance of the steel because it is typically much lower than the thermal resistance of refractory. Assuming a refractory thermal conductivity of 2 $\mathrm{W/(m \cdot K)}$, the conductive *Resistance* of the wall material itself can be calculated as:

$$\mathrm{R}_{conduction} = \frac{L}{k} = \frac{0.1\ \mathrm{m}}{2\ \mathrm{W/(m \cdot K)}} = 0.05 \frac{\mathrm{m^2 \cdot K}}{\mathrm{W}} \tag{19.4}$$

where:

L is the thickness of the refractory in units of m
k is the thermal conductivity of the refractory in units of $\mathrm{W/(m \cdot K)}$

2. **Convection to the ambient air:** for a vessel with primarily vertical wall surfaces, and assuming no wind movement, an R-value can be obtained from reference text books or online sources such as R-value (insulation) Sample Values:

$$\mathrm{R}_{convection} = 0.12 \frac{\mathrm{m^2 \cdot K}}{\mathrm{W}} \tag{19.5}$$

Analogous to electrical resistances in series, these two components of heat transfer resistance can be summed to calculate the total thermal resistance value to be used in *Virtual Reactor*:

$$\mathrm{R} = \mathrm{R}_{conduction} + \mathrm{R}_{convection} = 0.17 \frac{\mathrm{m^2 \cdot K}}{\mathrm{W}} \tag{19.6}$$

https://cpfd-software.com/user-manual/release_guide_24.1.0.html#rg-thermal-wall-bc-updates

Oka Erosion Wall Impact Model

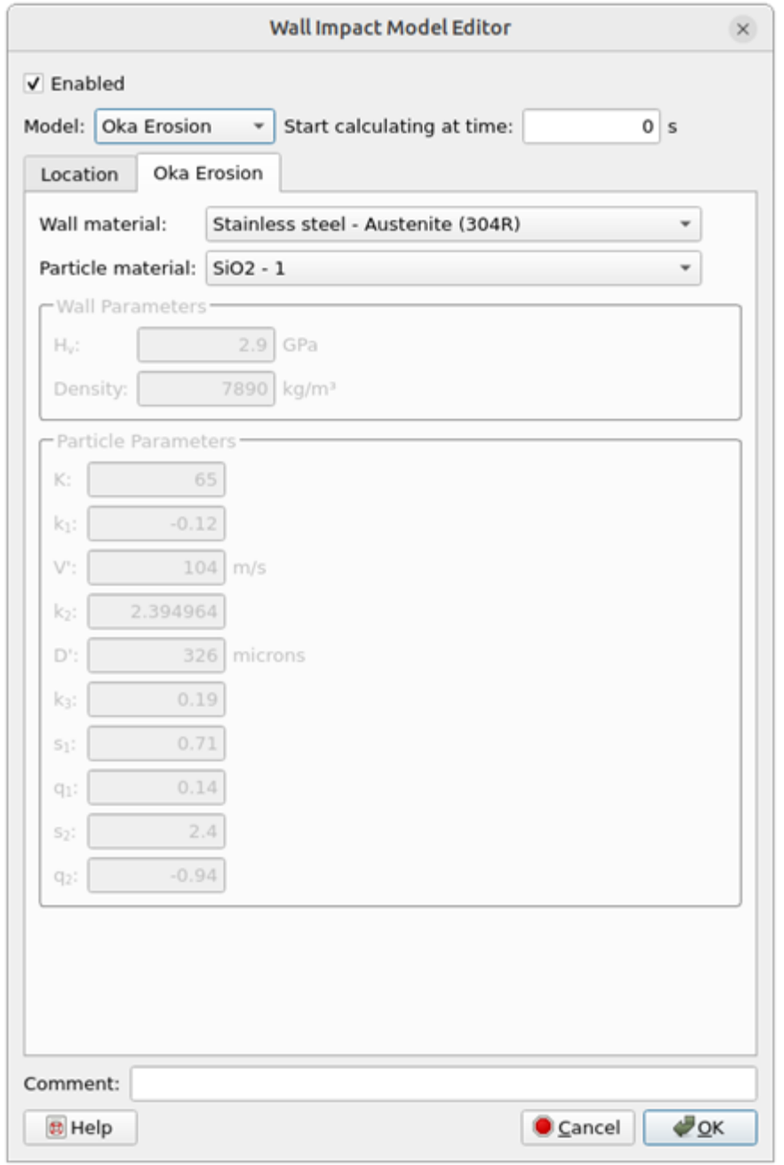
Added in release 25.0

Based on papers published by Oka et al

- https://doi.org/https://doi.org/10.1016/j.wear.2005.01.039
- https://doi.org/https://doi.org/10.1016/j.wear.2005.01.040

Quantitative calculation of wall material loss due to particle impact (kg/m^2/year)

Selection of pre-defined wall and particle materials is available, or user-defined parameters can be entered

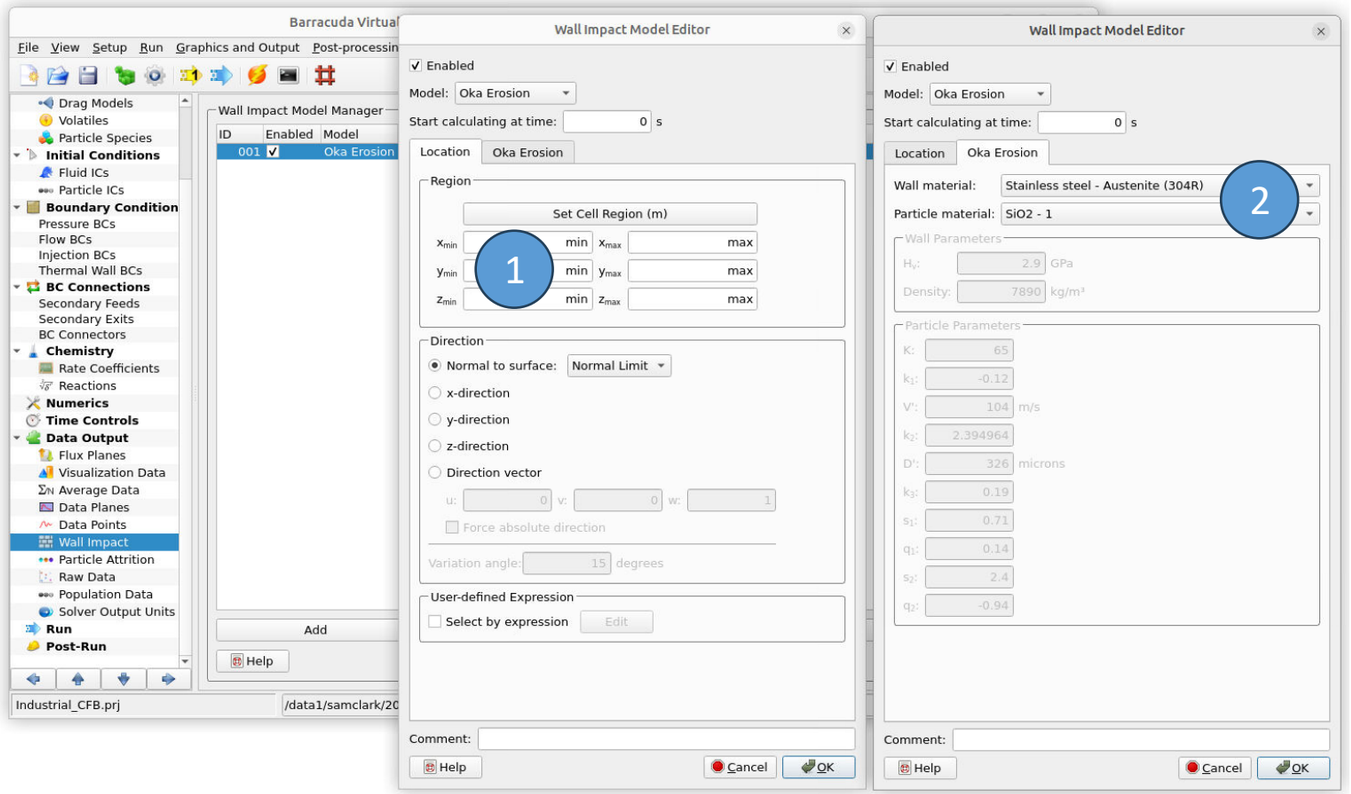


Wall Impact Model Settings

The initial case was configured with a single Wall Impact model, using built-in parameters for the Oka Erosion model

1. **Location tab: spatial region was set to min-max in all directions**

2. **Oka Erosion tab: wall material was set to Stainless steel – Austenite (304R), particle material was set to SiO2 - 1**

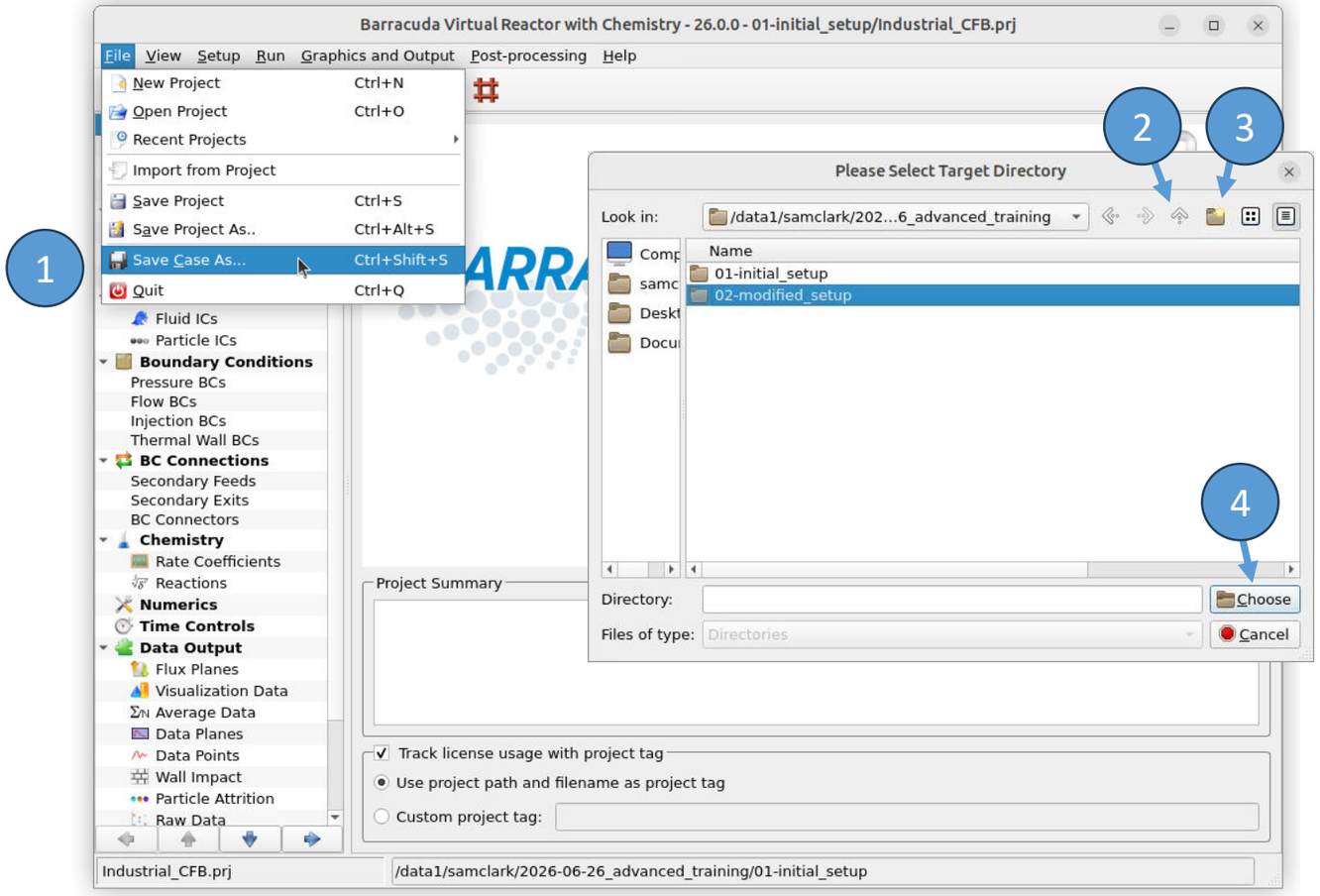


Starting a New Case from the Endpoint of the Initial Case

Save Case As…

Easiest way to clone an existing case when you want to make a few changes and start a new run

1. **File → Save Case As…**
2. **Go up a directory**
3. **Create new directory named: 02-modified_setup**
4. **Choose**



Include Final IC_#.#### from Initial case

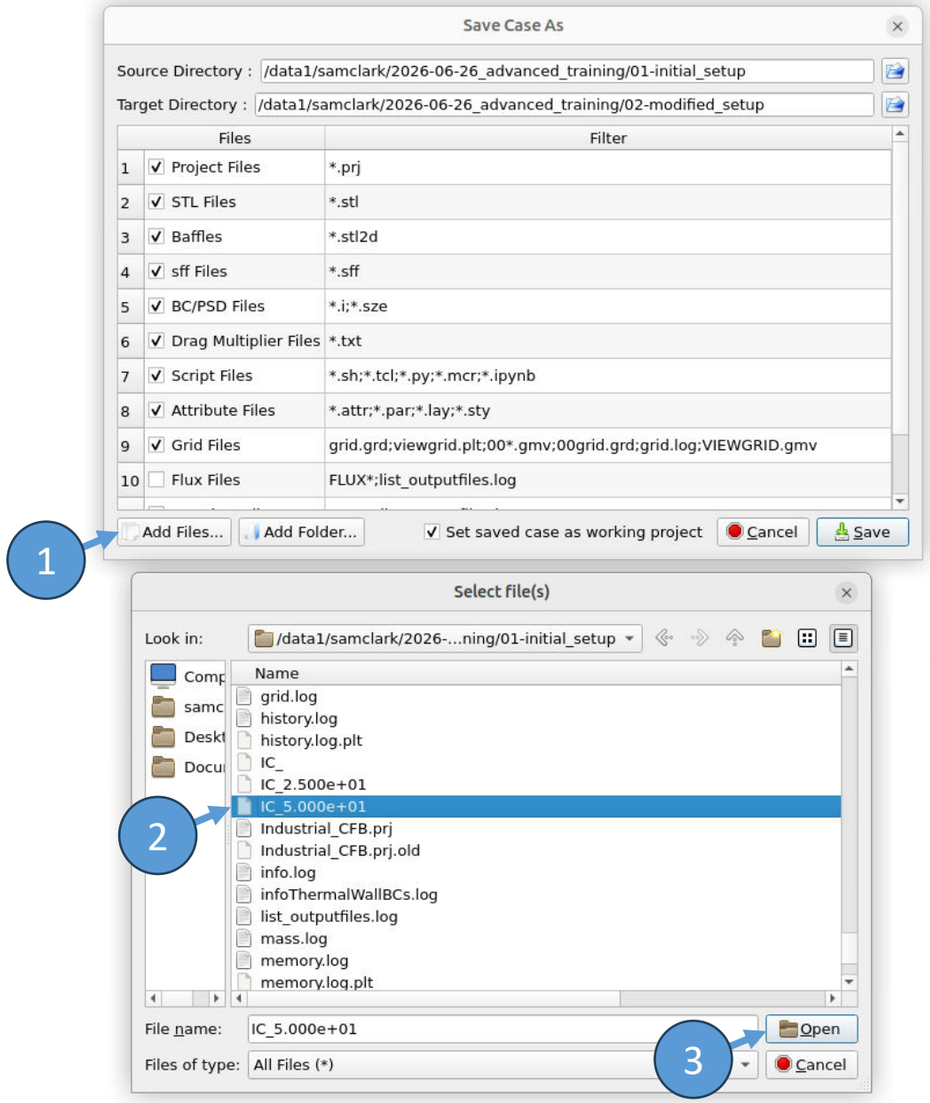
By default, rows 1-6 are selected

For this exercise, also select rows 7-9

In our new run, we want to initialize the fluid and particle phases based on the final state of the initial run

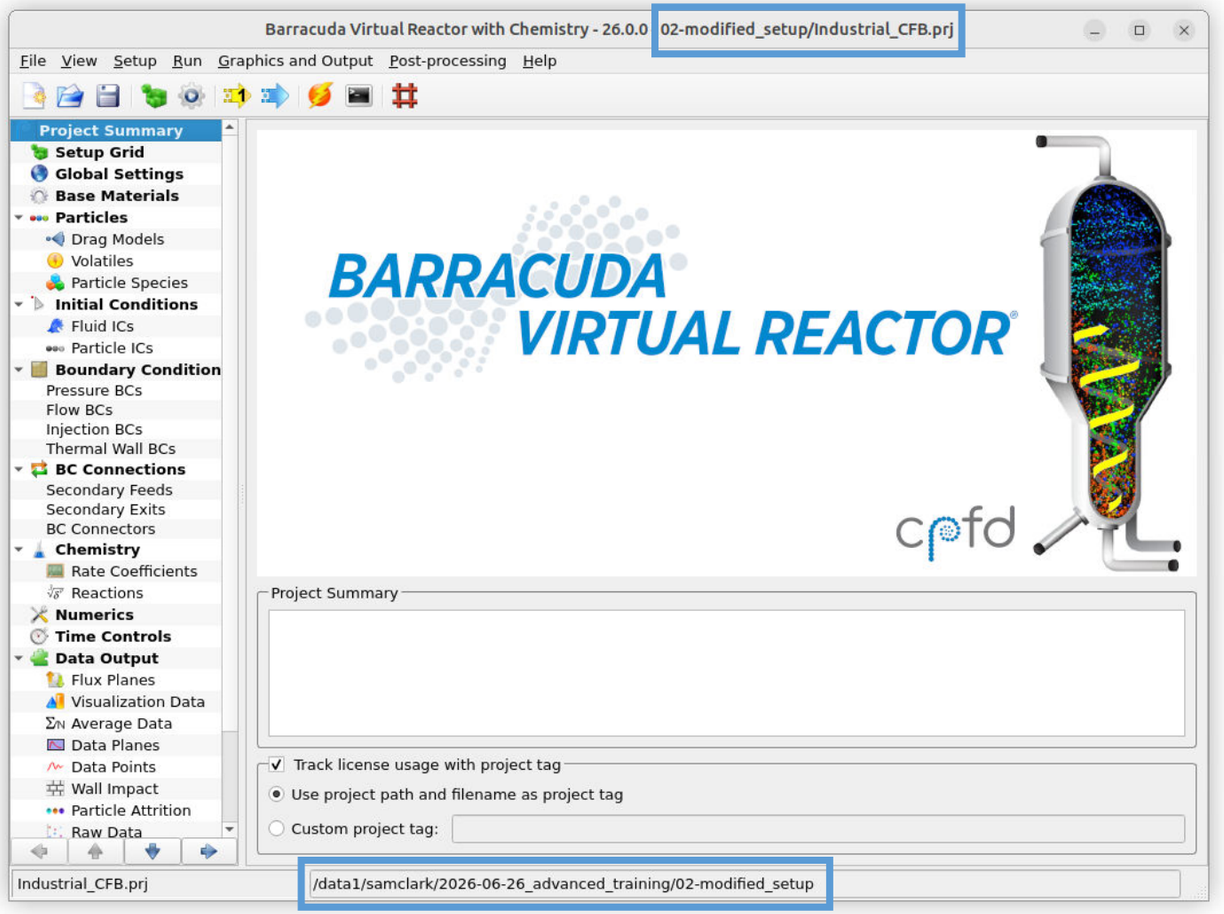
1. Add Files…
2. Select IC_5.000e+01
3. Open

Click "Save" to copy the selected files to the new case directory



Save Case As… Automatically Opens New Project

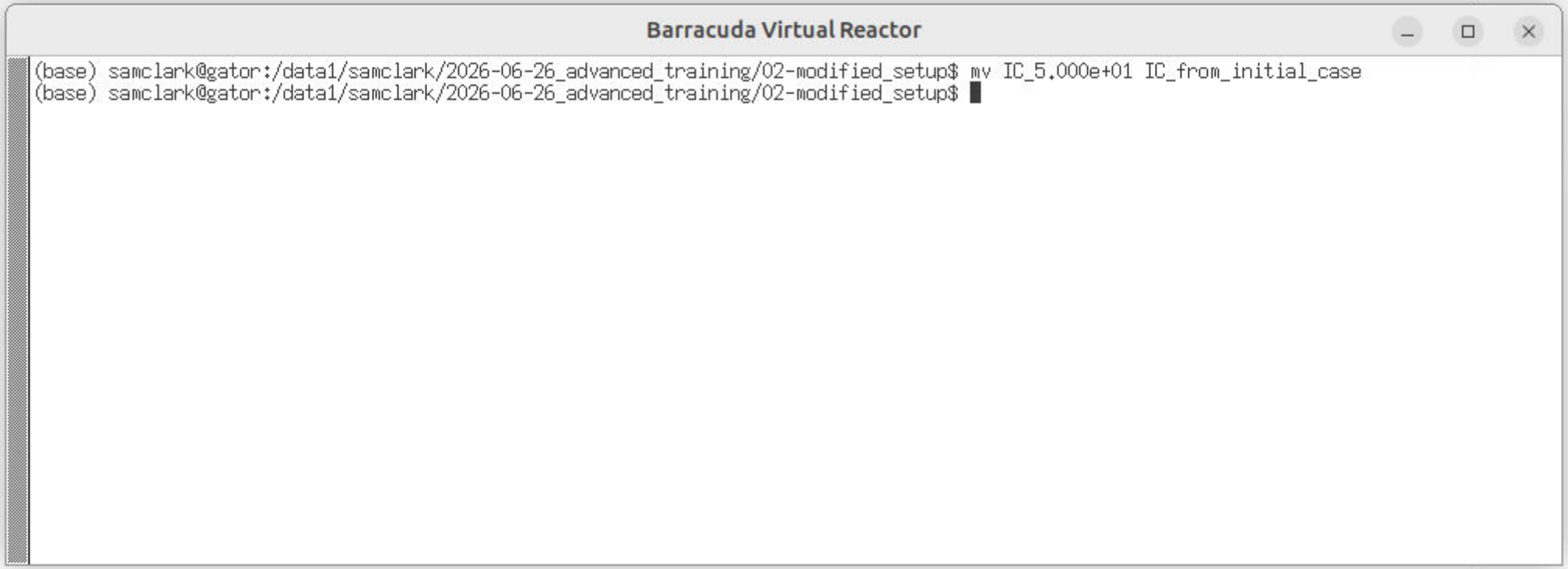
Be aware that, by default, the newly created project is automatically opened when you use Save Case As…



Rename IC_5.000e+01

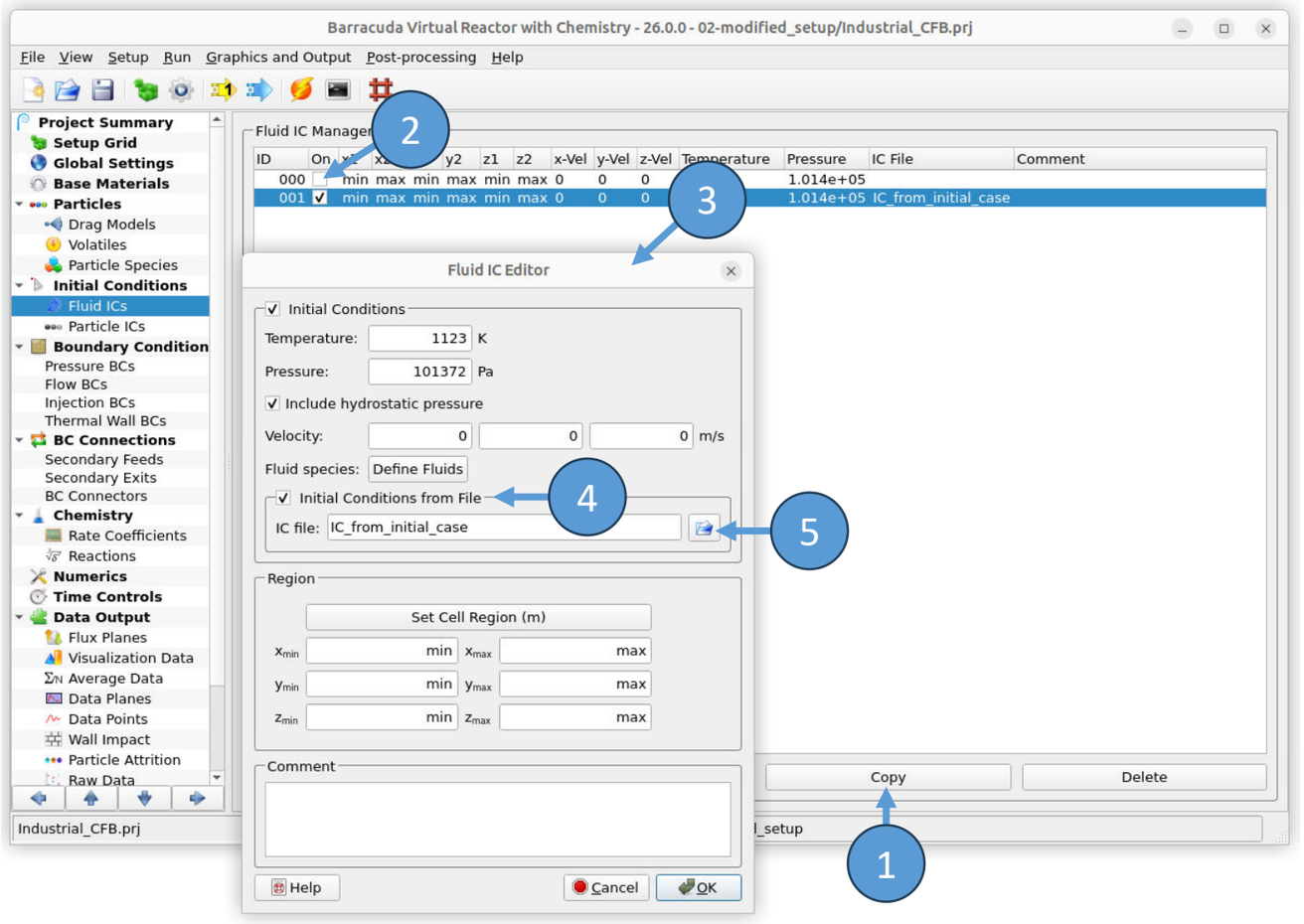
To avoid confusion, it is best practice to rename the IC file from the initial case

Open a terminal and run the following “mv” command:



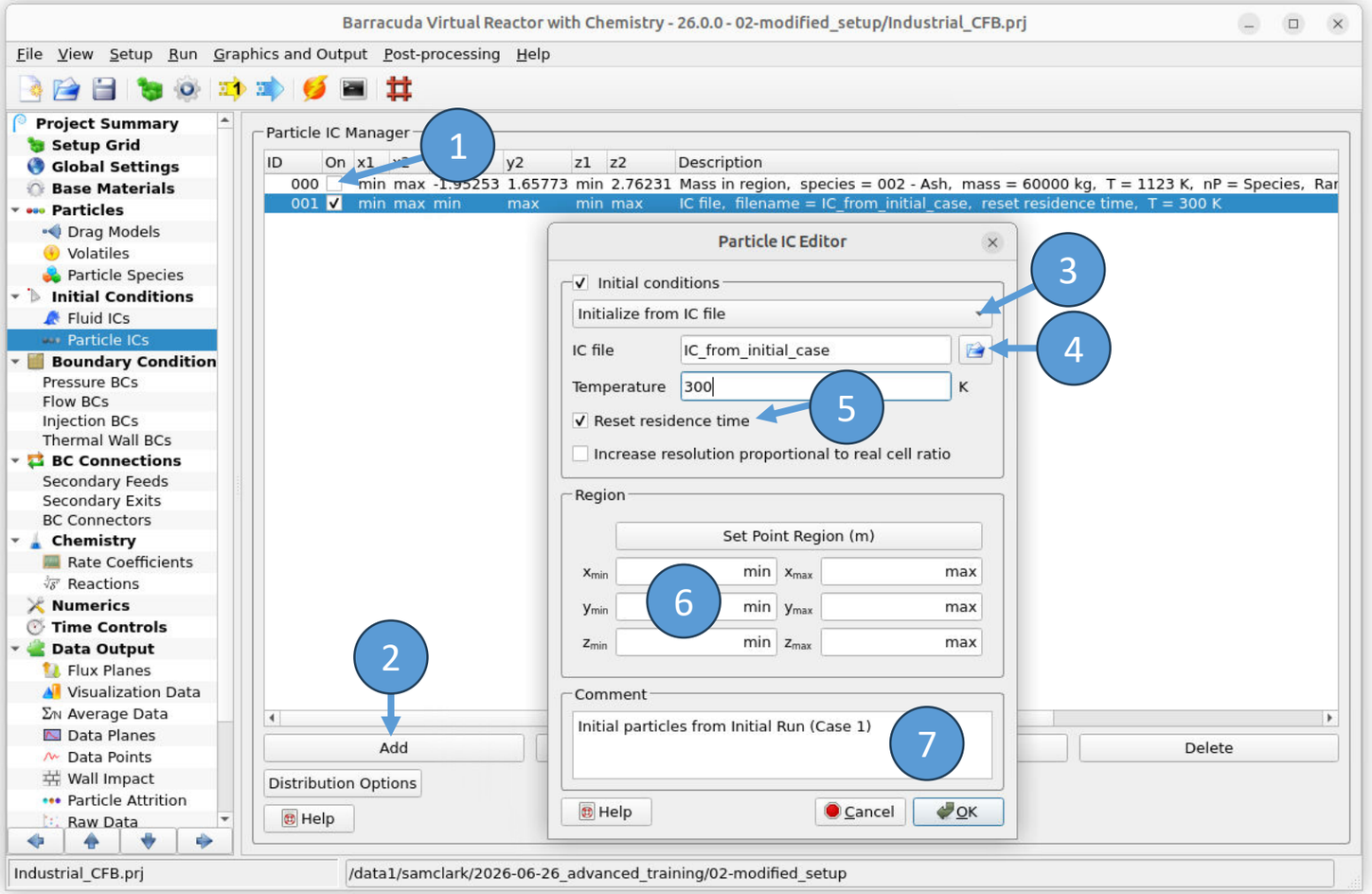
Define Fluid IC Using Final IC File from Case 1

1. **Copy Fluid IC 000**
2. **Disable Fluid IC 000**
3. **Edit Fluid IC 001**
4. **Enable "Initial Conditions from File"**
5. **Click the folder icon and browse for IC_from_initial_case**
    1. Verify that you are in the "02-…" directory



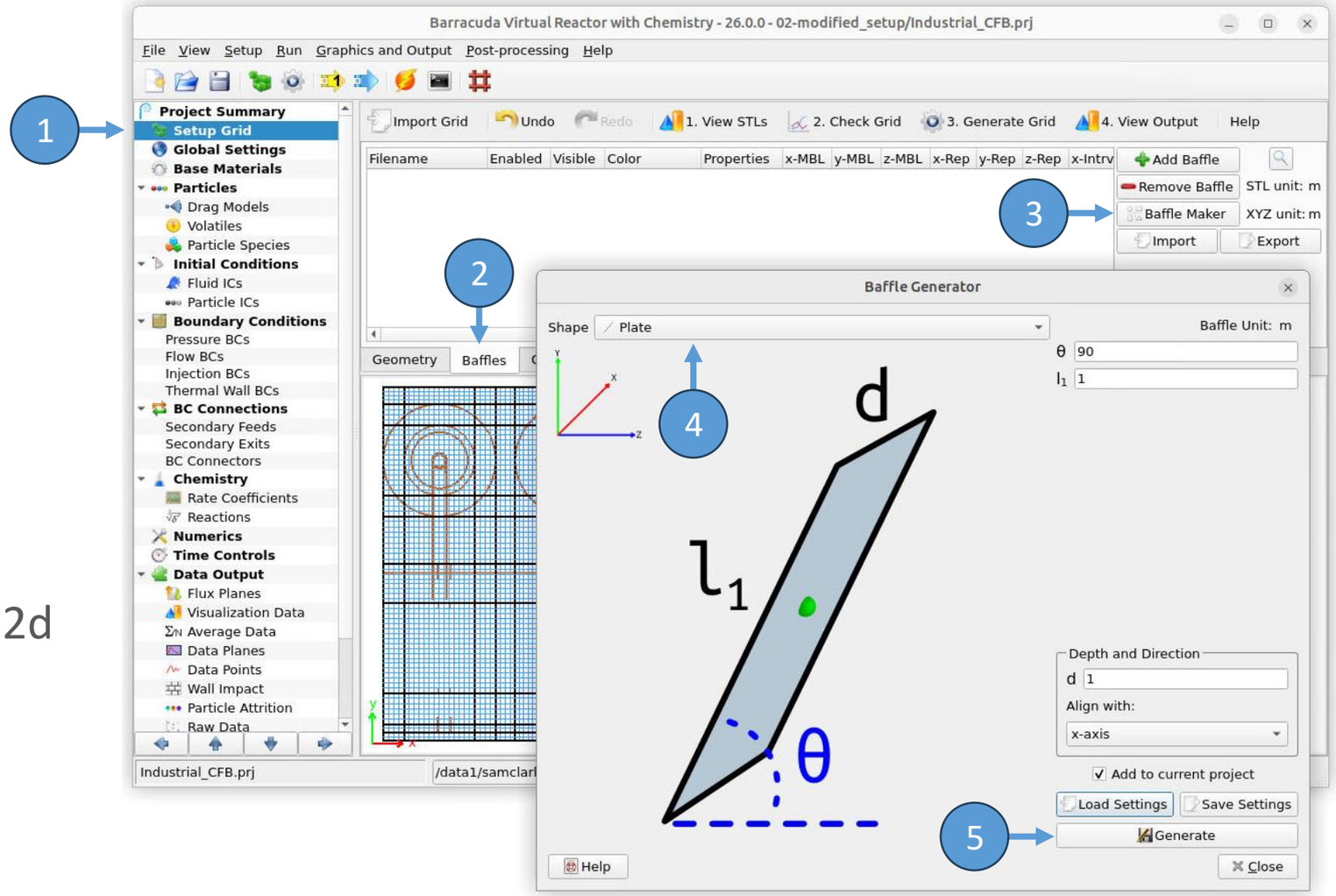
Define Particle IC Using Final IC File from Case 1

1. Disable Particle IC 000
2. Click “Add”
3. Select “Initialize from IC file”
4. Browse to select IC_from_initial_case
5. Enable “Reset residence time”
6. Set x-, y-, and z- limits to min and max
7. Add a comment



Add a Baffle with Particle Filtering by Size

1. **Setup Grid**
2. **Baffles tab**
3. **Baffle Maker**
4. **Shape: Plate**
5. **Generate**
   - Use name: plate_baffle.stl2d
   - Save
   - Ok



Position Baffle Near Top Outlet

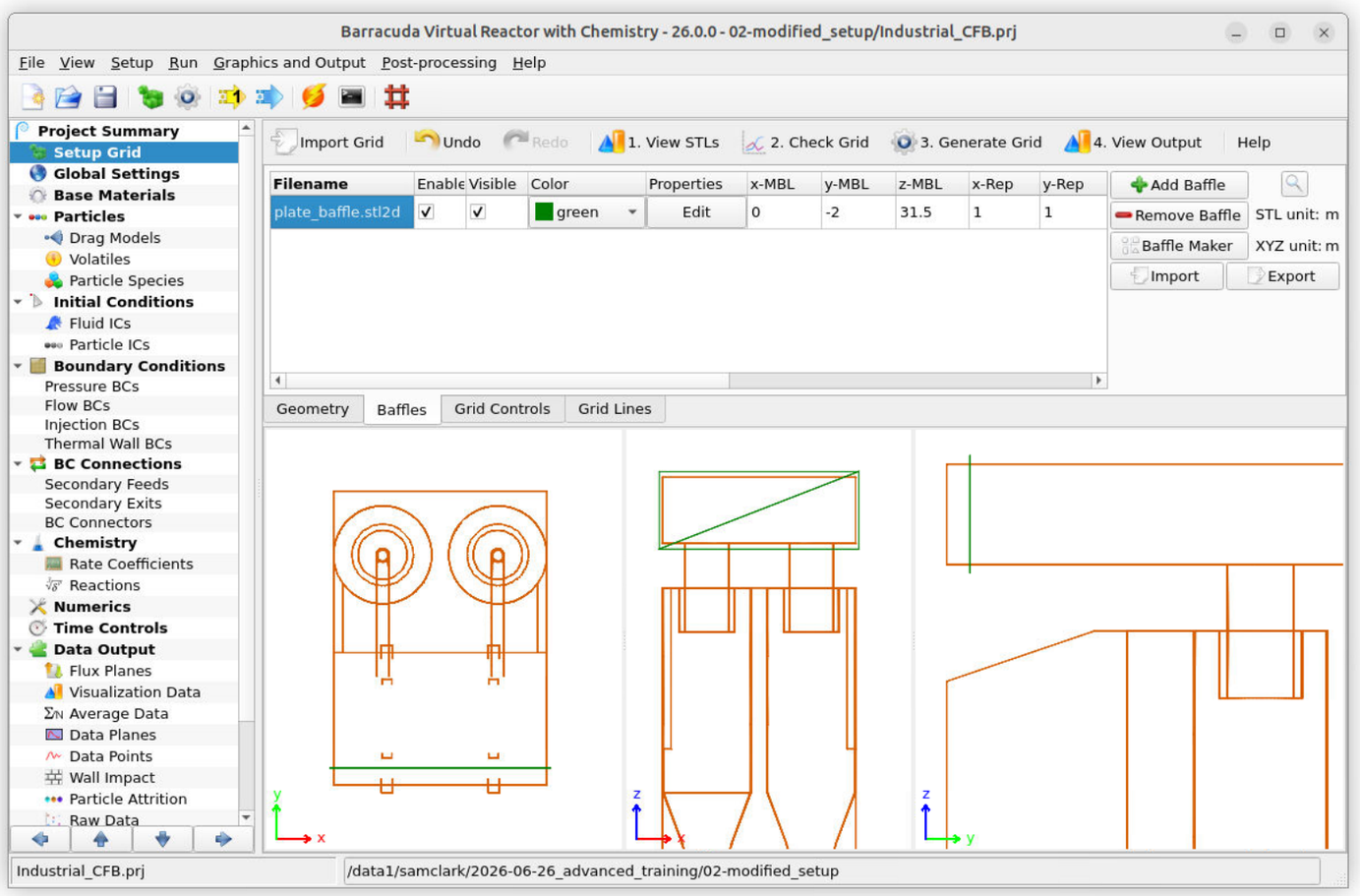
Baffle is initially centered at (0,0,0)

Grid Controls → Advanced Options:

- Pixel width for major grid lines = 0
- Pixel width for minor grid lines = 0

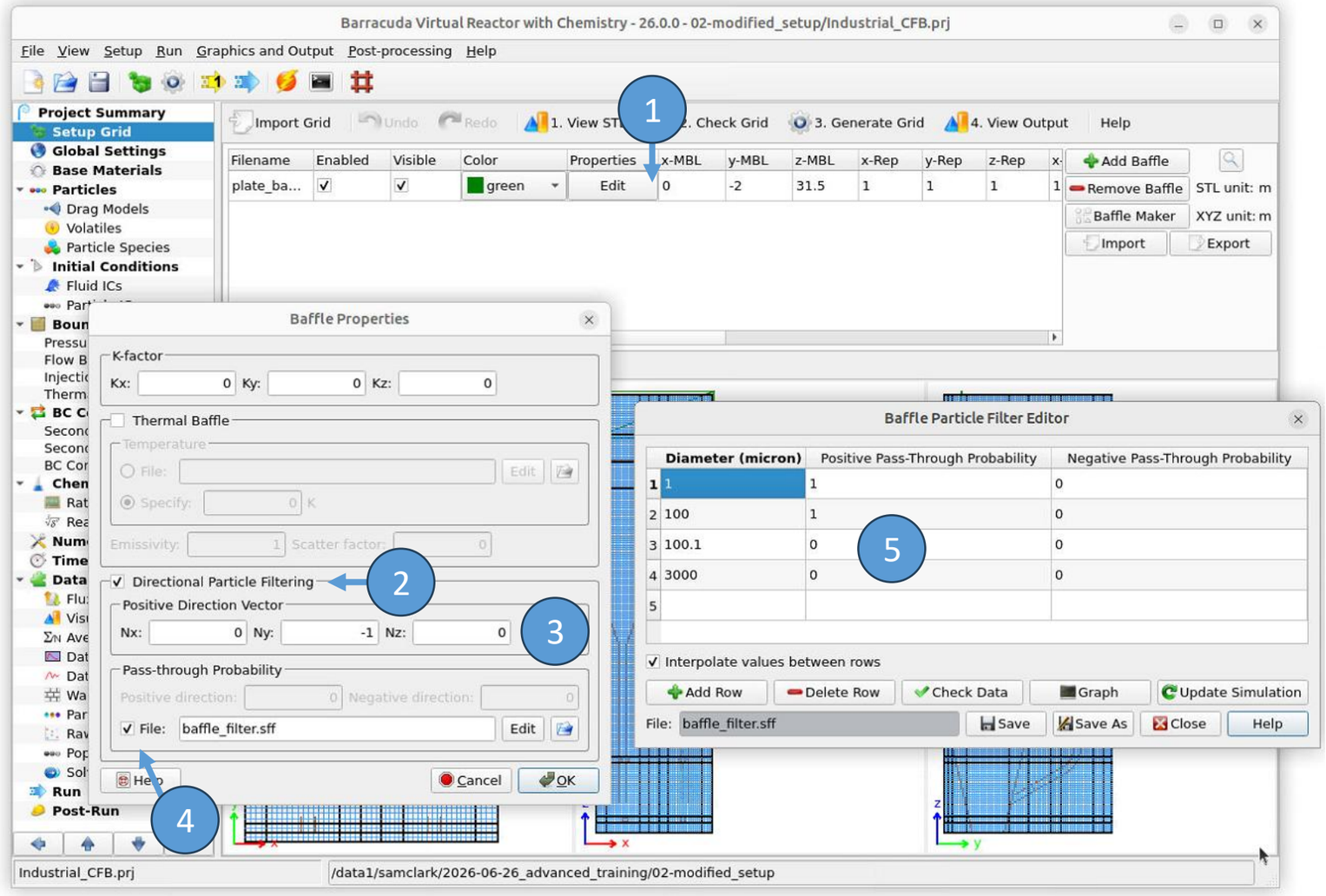
Baffle position settings:

- x-Rot = 90
- z-MBL = 31.5
- y-MBL = -2
- y-Scale = 3.5
- x-Scale = 9



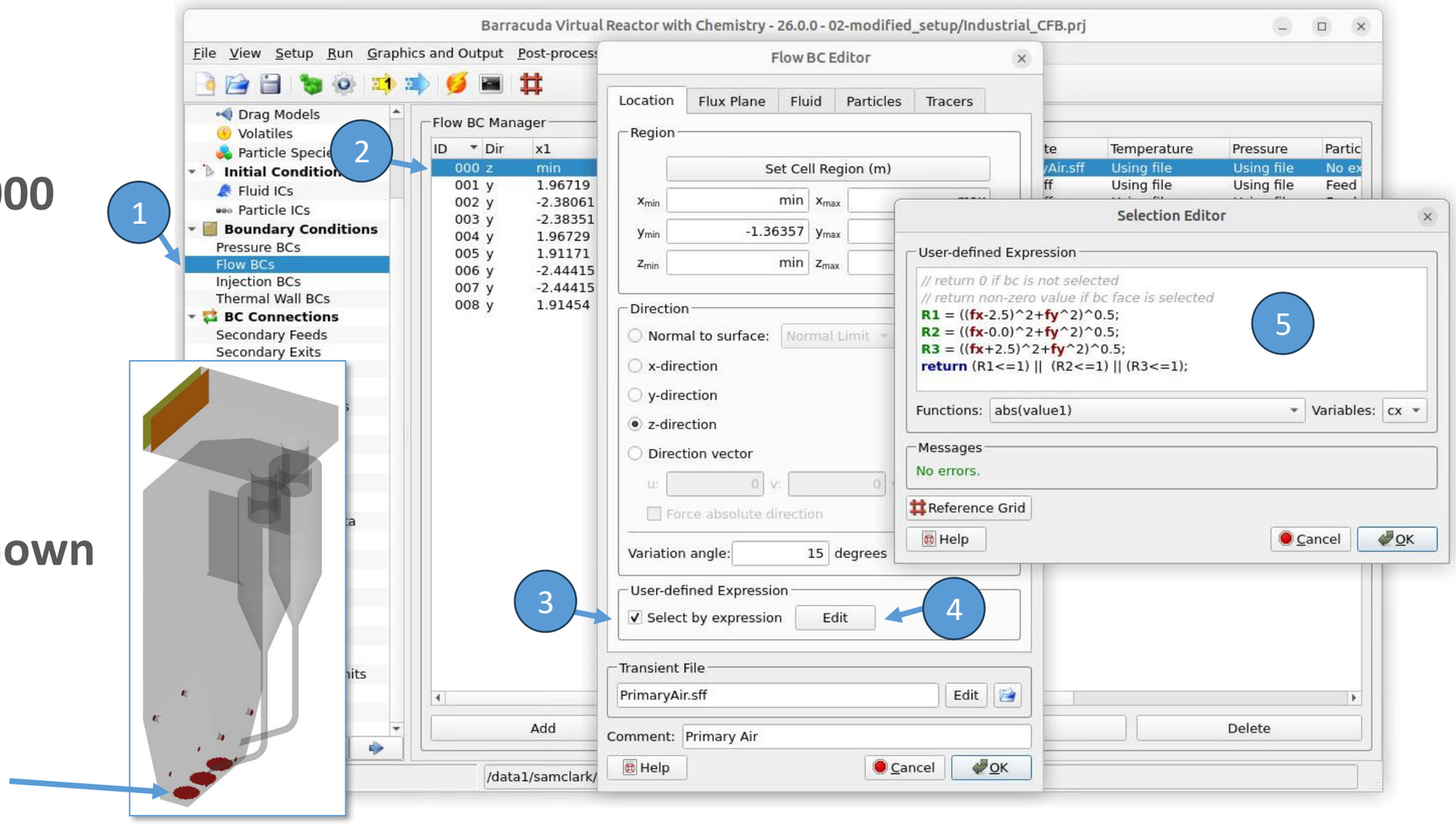
Set Baffle Filtering Properties

1. Edit
2. Directional Particle Filtering
3. Nx = 0, Ny = -1, Nz = 0
4. Enable “File” option
5. Set filter parameters as shown



User-defined Expression (UDE) for Flow BC

1. Flow BCs
2. Edit Flow BC ID 000
3. Enable Select by expression
4. Edit
5. Enter the UDE shown



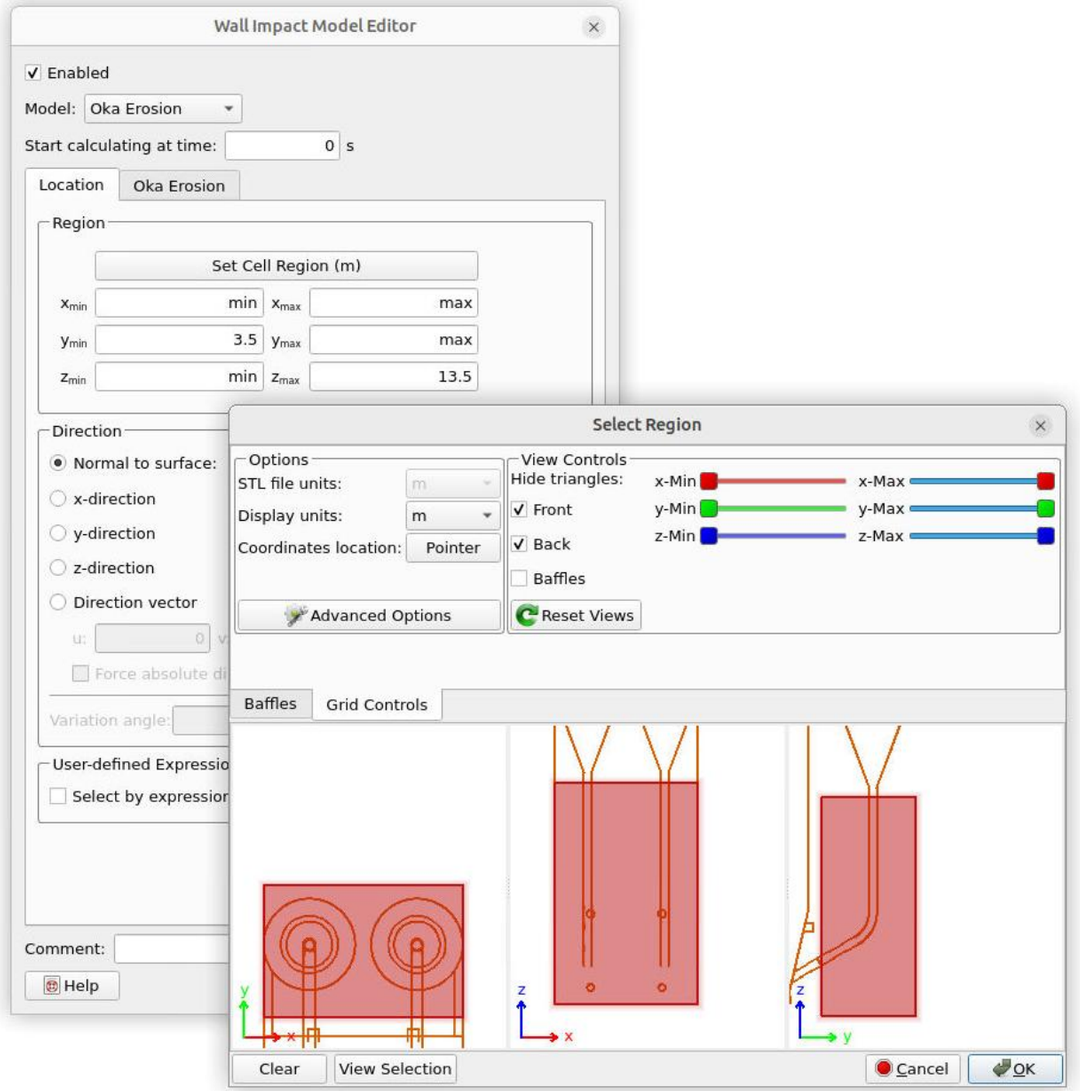
Wall Impact Model: Select Specific Spatial Regions

In systems with different materials present, it can be useful to isolate wall impact models to specific regions of space

For this example, let's assume the only bare steel surface is the inside of the cyclone diplegs

1. **Edit 001**

2. **Locations tab: use "Select Region" dialog to select cyclone diplegs**



Wall Impact Model: UDE to Select Spatial Regions

Further, let's assume that all other walls are refractory lined. We can use a UDE to select all walls that are *not* overlapping the region of the cyclone diplegs

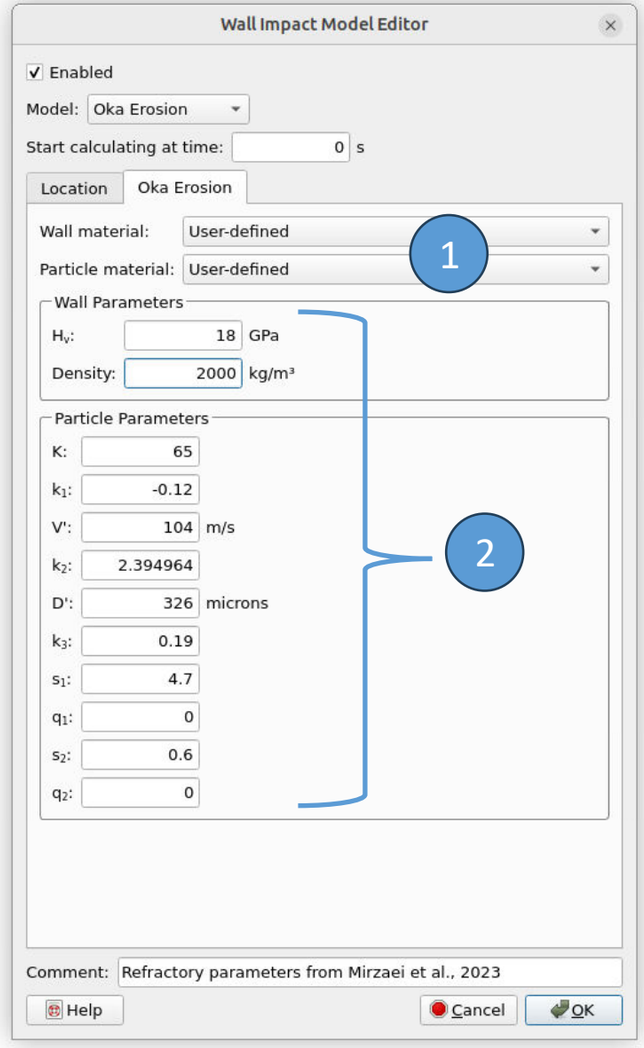
1. **Copy 001**
2. **Edit 002**
3. **Location tab → Enable User-defined Expression**
4. **Enter the expression shown**

Wall Impact Model: Custom Parameters for Oka Erosion

Custom Oka Erosion parameters can be used to represent refractory-lined walls. For this example, we will use parameters based on Mirzaei et al., 2023 (https://doi.org/10.1016/j.powtec.2023.118424)

1. **Oka Erosion tab → User-defined for both Wall material and Particle material**

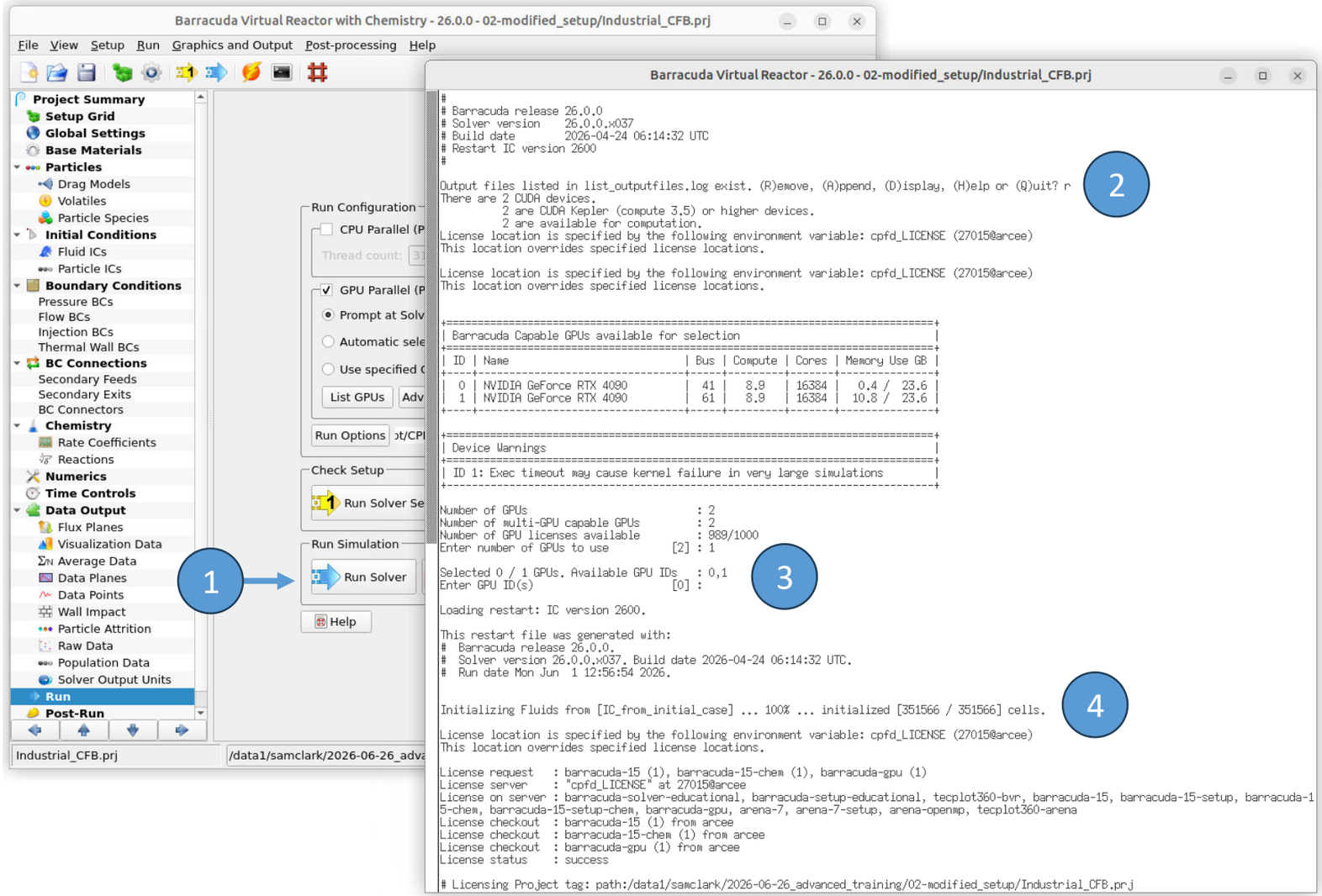
2. **Enter the values shown**



Start Simulation from t=0

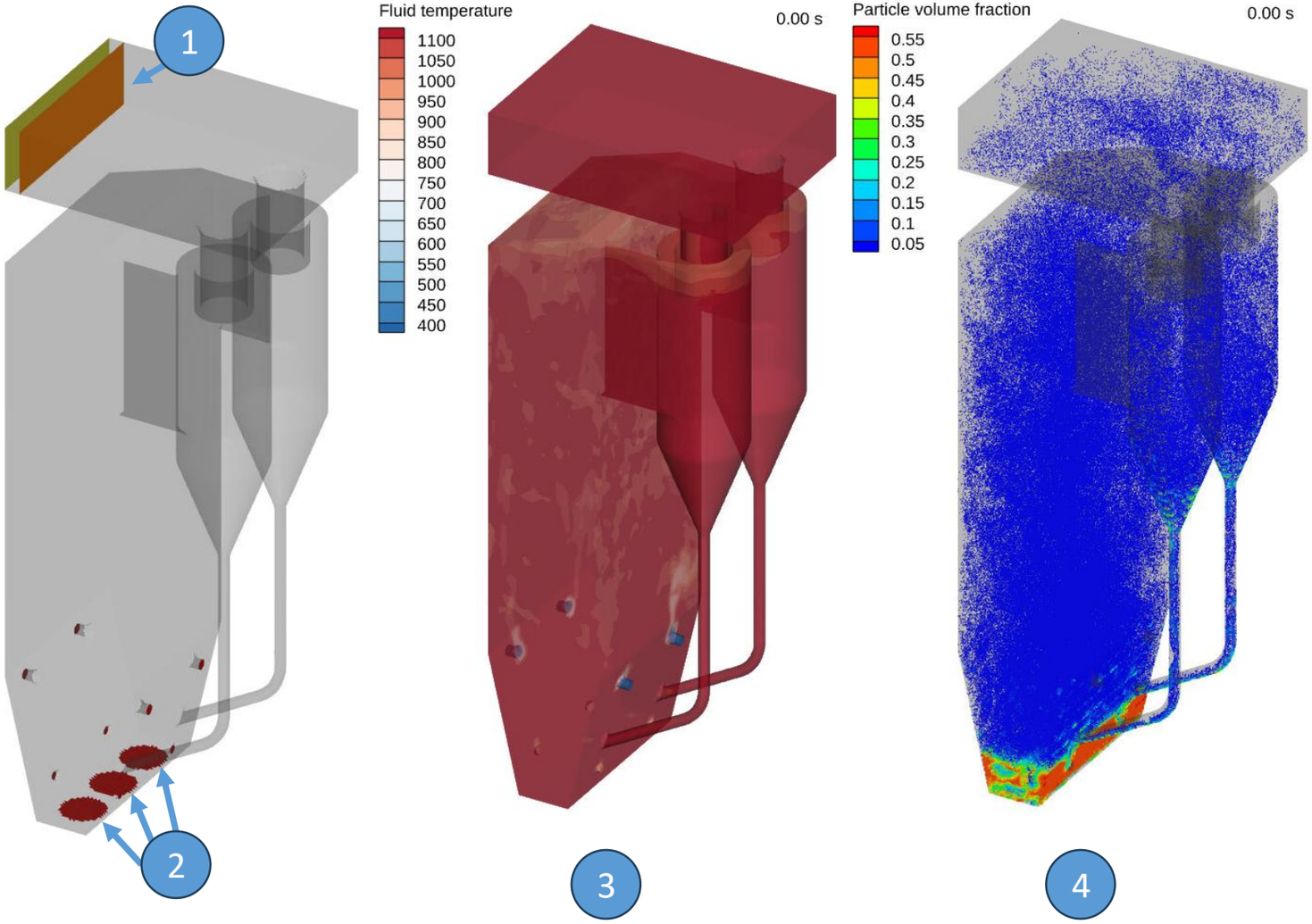
Remember: this is a new simulation, not a restart

1. **Run → Run Solver**
2. **Remove any files when prompted to start solver**
3. **Select the GPU ID**
4. **Note that fluid and particles are initialized from IC_from_initial_case**



Use "View Setup" to Verify Changes to Project

1. Baffle has been added
2. Flow BCs for primary air have been changed to three circular regions
3. Fluid has been initialized from previous case
4. Particles have been initialized from previous case



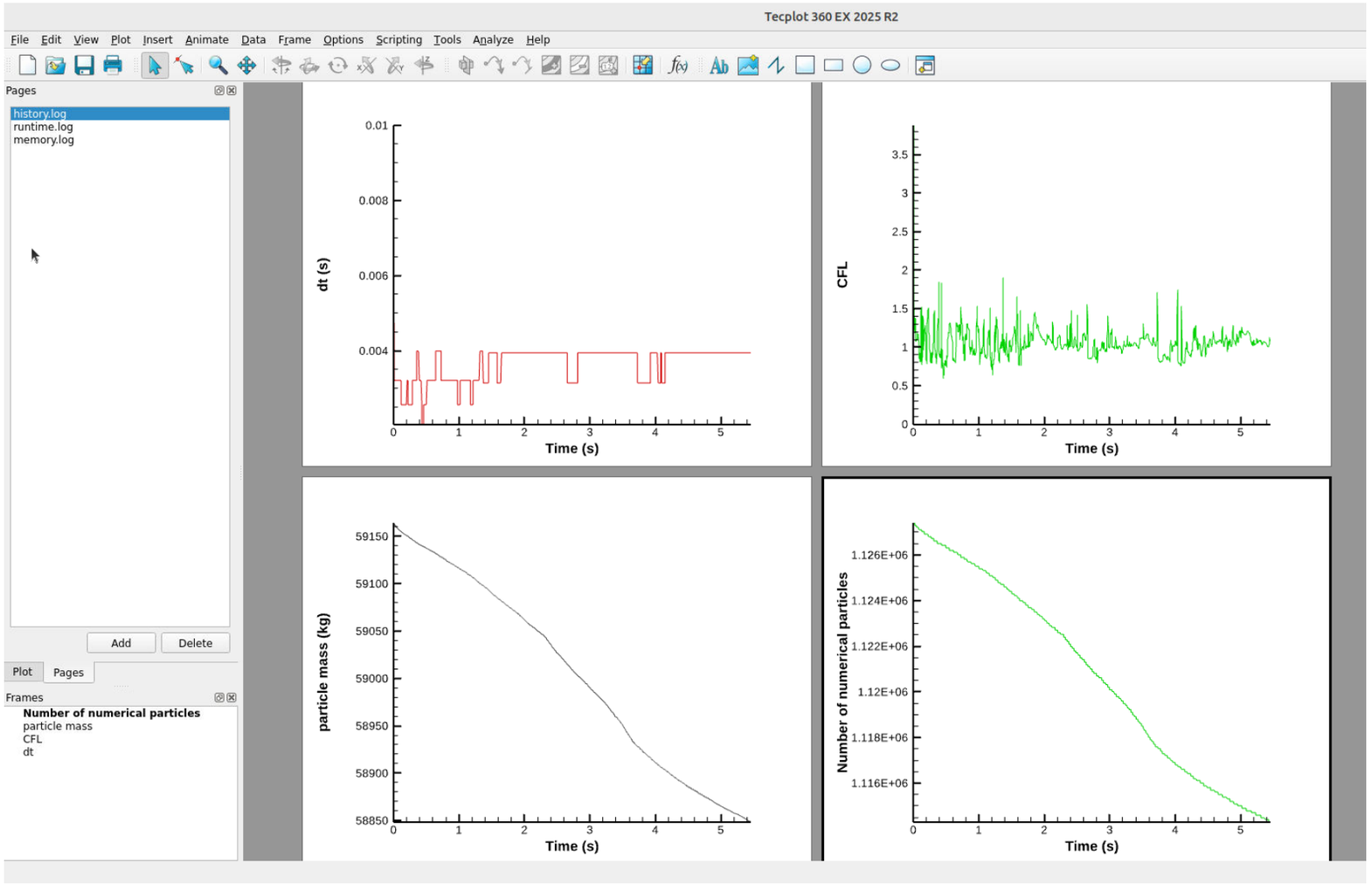
Post-Processing Examples and Exercises

Monitor Run – history.log Page

Good to do first

Keep an eye on:

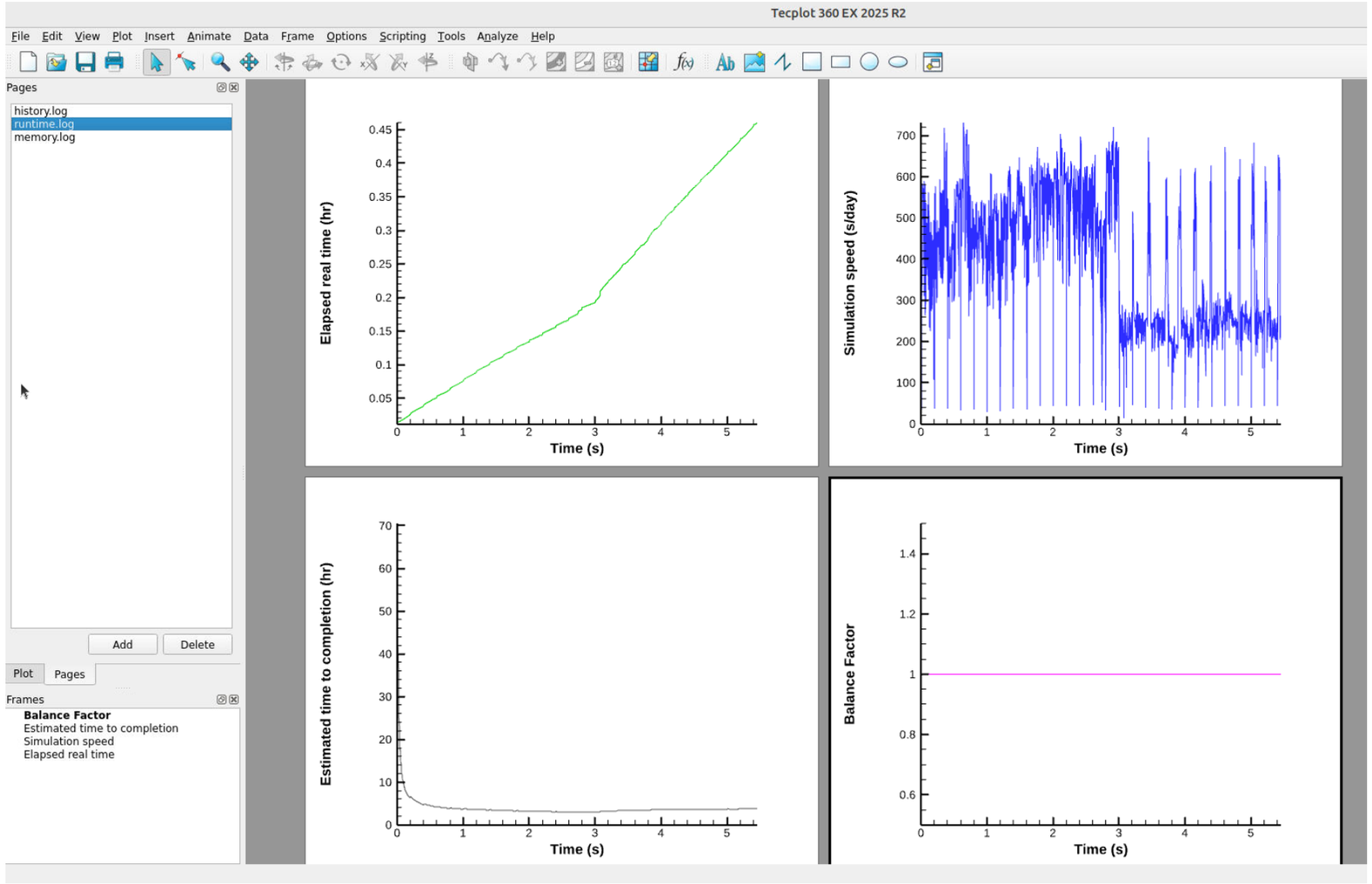
- Time step: is it staying at a reasonable value? You don't want it to drop too low because that is slow.
- CFL: is it staying around 1?
- Total mass and total particle count: are these increasing or decreasing too much?



Monitor Run – runtime.log page

Most useful plots are often:

- Elapsed real time: to answer “how long did the simulation take to run”, especially at the end
- Simulation speed: to know how fast the simulation is running and if it is slowing down over time



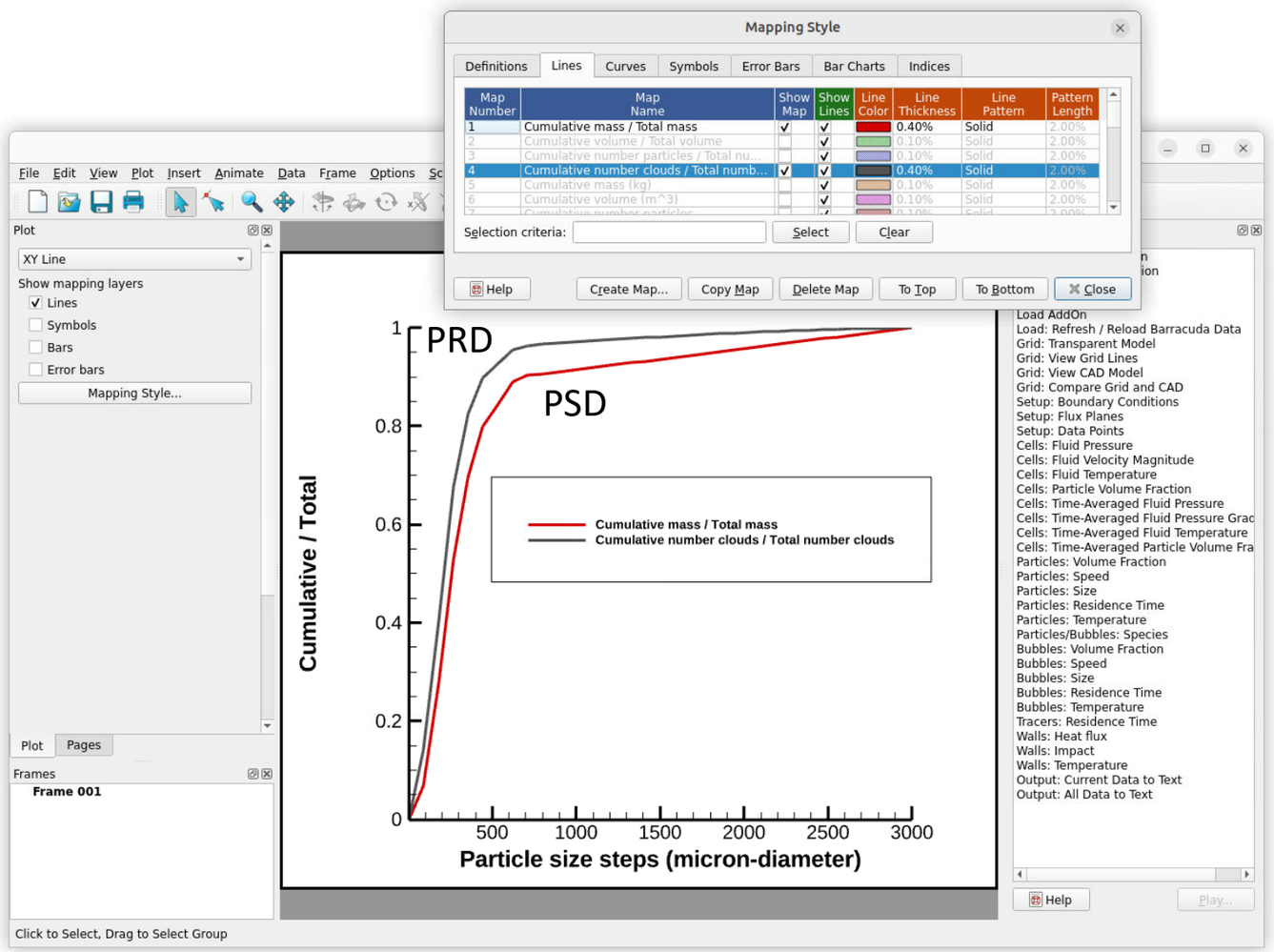
Plotting PRD Curves from POPUL*log Files

Launch Tecplot

File → Load Barracuda Data → Load Data Files → POPUL_0.000e+00.log

Useful to quantify PRD vs PSD, to show how much resolution is allotted based on particle size

If no PRD used → PSD curve equals PRD curve

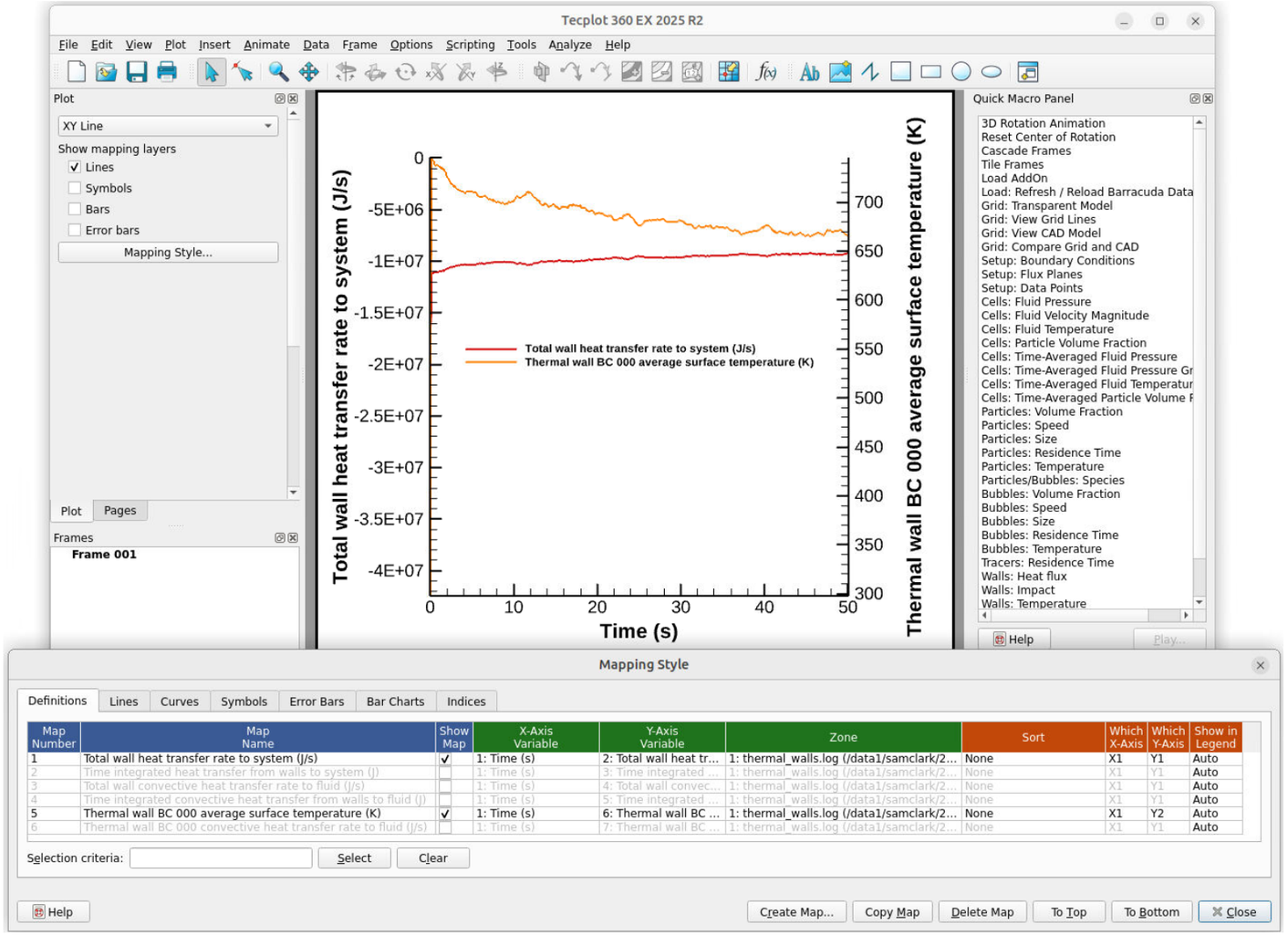


https://cpfd-software.com/tecplot-for-barracuda-creating-xy-plots/

Plotting Data from thermal_walls.log

In this example, it's interesting to plot heat transfer along with average surface temperature (since we specified an ambient temperature and a resistance)

In other simulations, plotting conductive, convective, and radiative transfer is often useful



https://cpfd-software.com/tecplot-for-barracuda-creating-xy-plots/

Visualizing Particles Stopped by Baffle

Quick macro: Particle Size

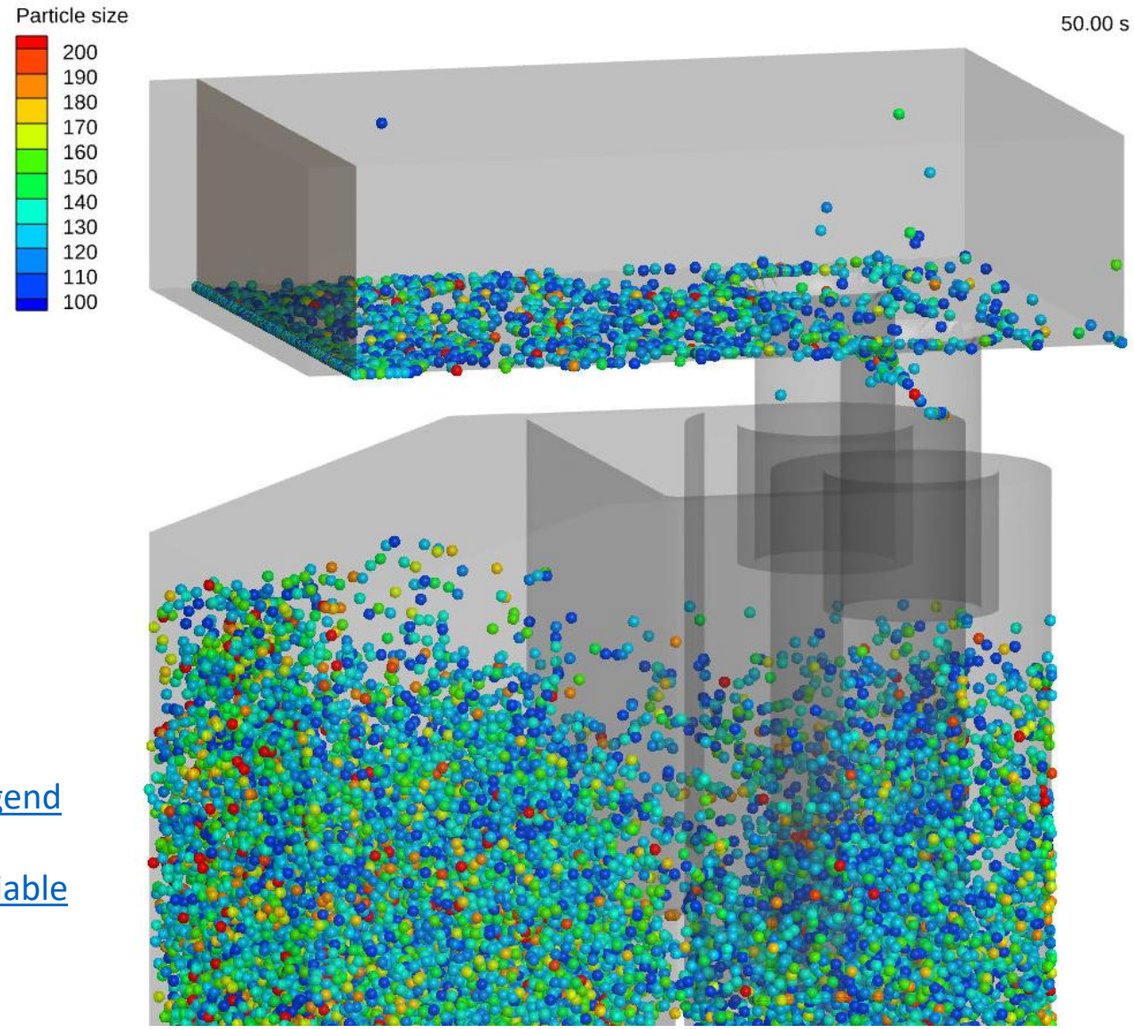
Zone style: Show baffle

Zone style: Use spheres for scatter

Blanking: Hide particles <= 100.1 microns diameter

Contour legend: 100 – 200 microns range

https://cpfd-software.com/tecplot-for-barracuda-adjusting-the-contour-legend
https://cpfd-software.com/tecplot-for-barracuda-using-value-blanking
https://cpfd-software.com/tecplot-for-barracuda-scaling-particles-by-a-variable



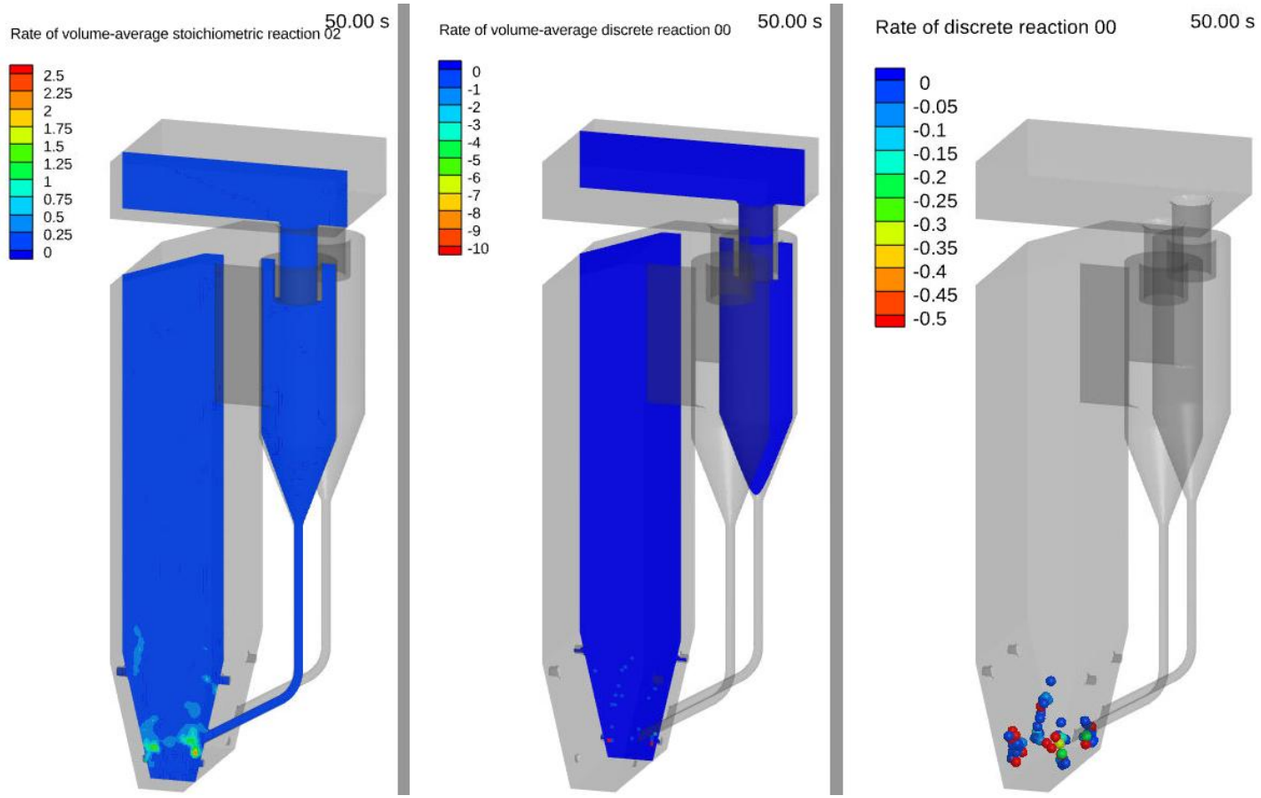
Visualizing Chemical Reaction Rates

Slices to show "volume-average stoichiometric" and "volume-average discrete" reaction rates

- Animate to find regions of interest
- Adjust contour legend
- Both use selected "Volume-Average Chemical Reaction Rate" units from Solver Output Units window (see next slide)

Particles to show discrete reaction rate

- Blanking to isolate only particles of interest based on the specific reaction you're looking at
- Uses selected "Discrete Chemical Reaction Rate" units from Solver Output Units window (see next slide)



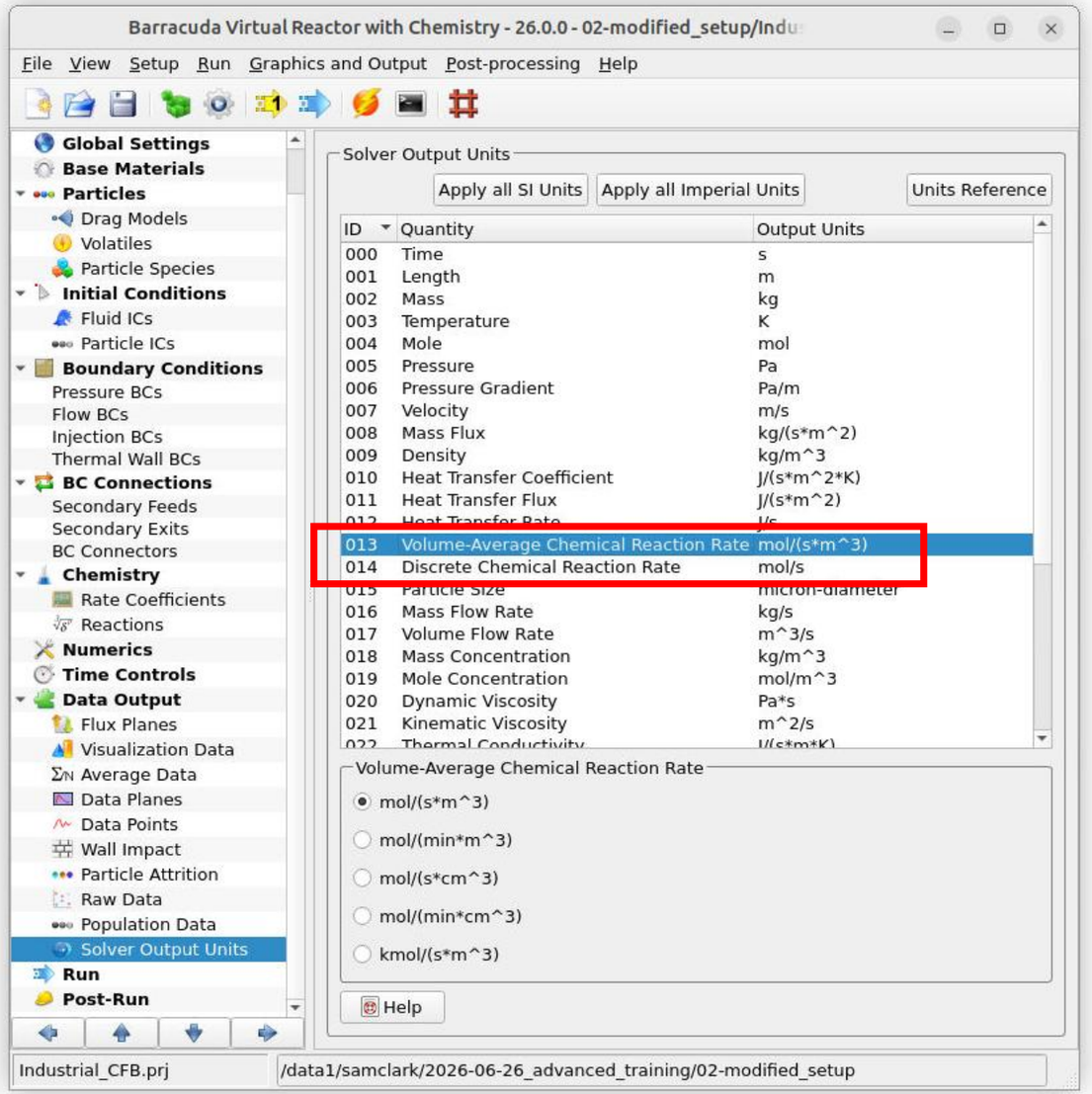
https://cpfd-software.com/tecplot-for-barracuda-adjusting-the-contour-legend
https://cpfd-software.com/tecplot-for-barracuda-using-value-blanking
https://cpfd-software.com/tecplot-for-barracuda-using-frames
https://cpfd-software.com/tecplot-for-barracuda-scaling-particles-by-a-variable

Visualizing Chemical Reaction Rates: Units

All chemical reaction rate output data has units based on the user's selections in the "Solver Output Units" window

You can use any combination of units when inputting chemical reaction rates, they will always be converted to the selected output units for all data output

"Volume-average discrete reaction rates" have the same units as "Volume-average stoichiometric" reaction rates



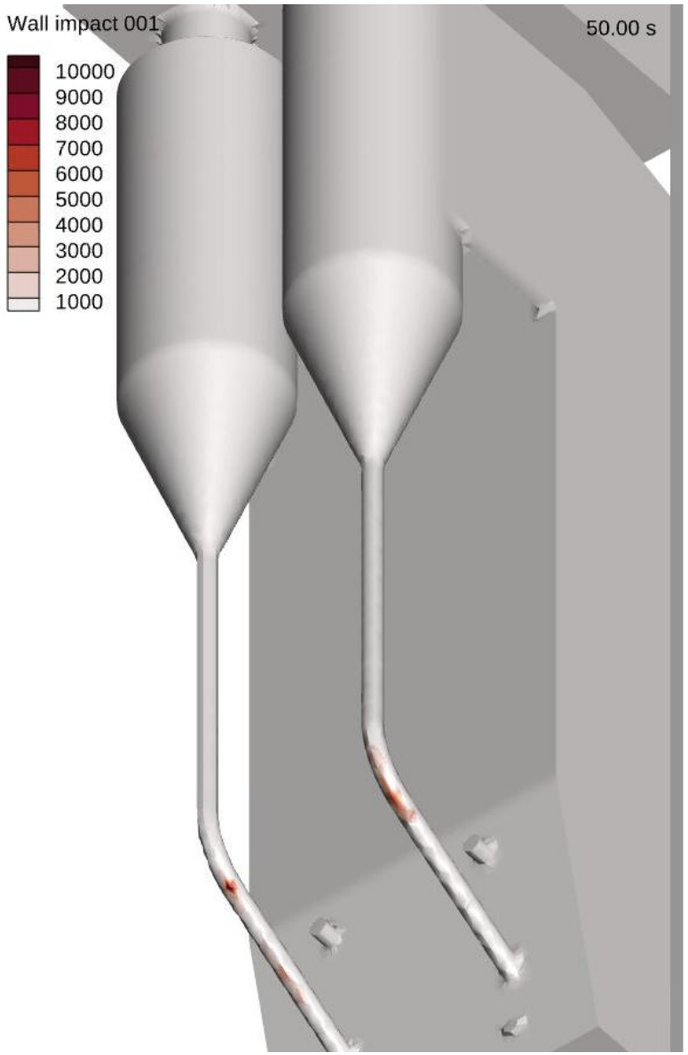
Examine Wall Impact Results

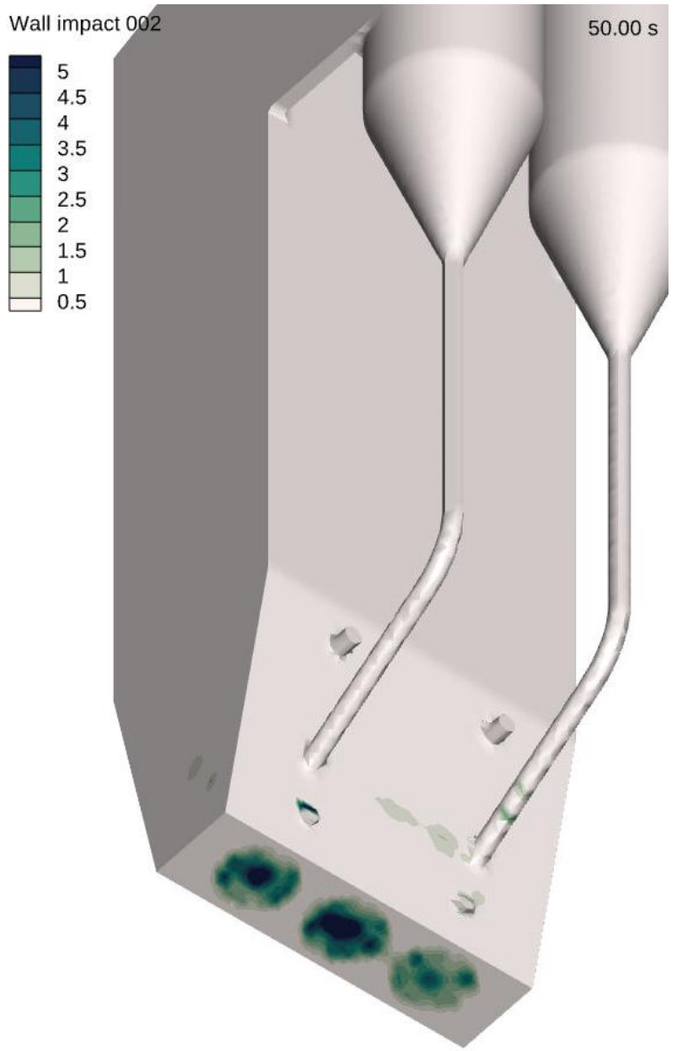
Use multiple frames to show wall impact results in specific regions based on project setup

- Wall impact 001 was configured to represent bare steel
- Wall impact 002 was configured to represent refractory lining

Units for wall impact:

- kg/m^2/year





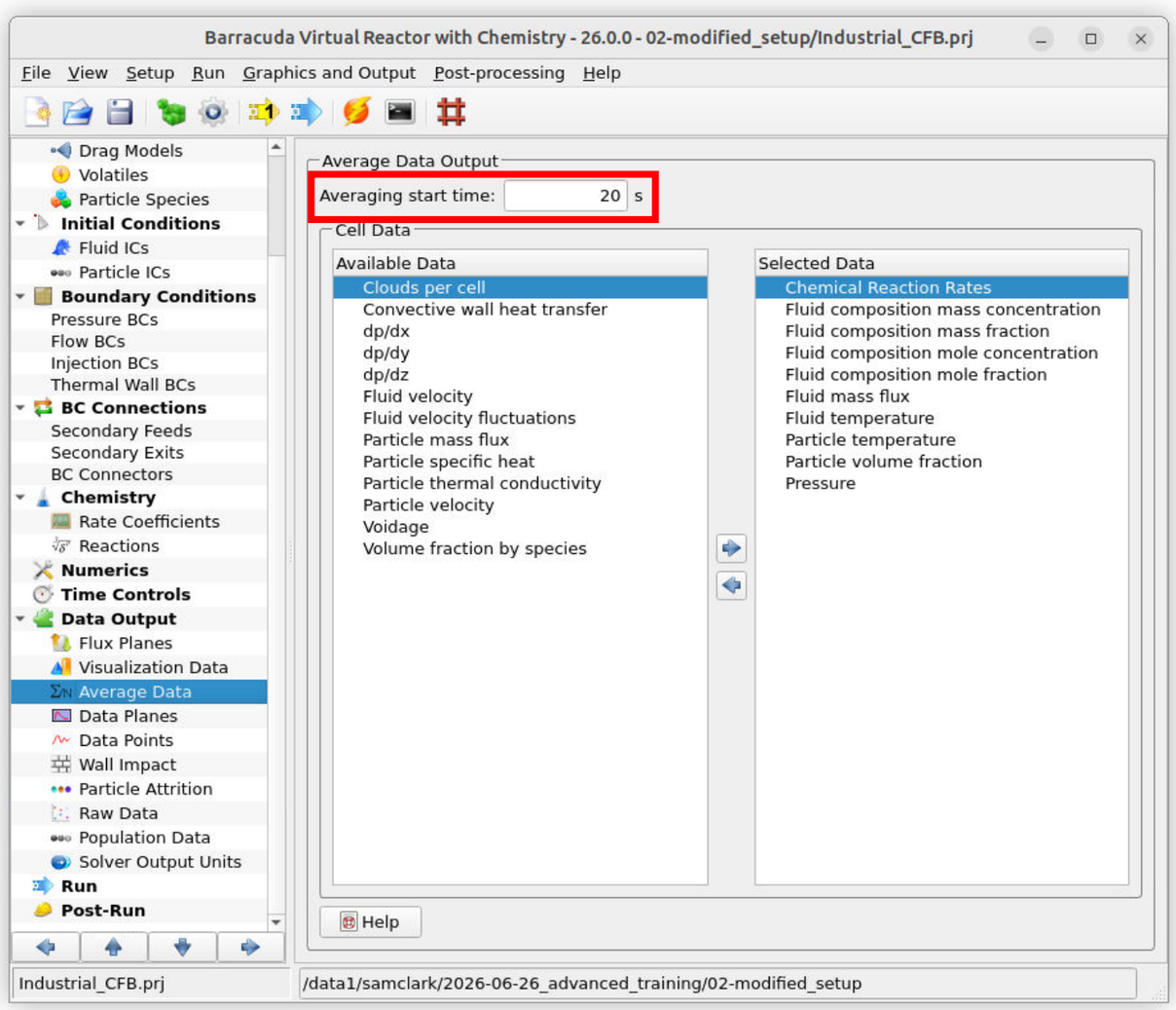
PyTecplot: Calculate Time-Average Over Custom Period

Barracuda's built-in Average Data:

- Starts at a user-specified time
- Is a cumulative average from that time

This default behavior often works well

However, sometimes it is useful to calculate a time-average over a different period of time → we can use PyTecplot to do this



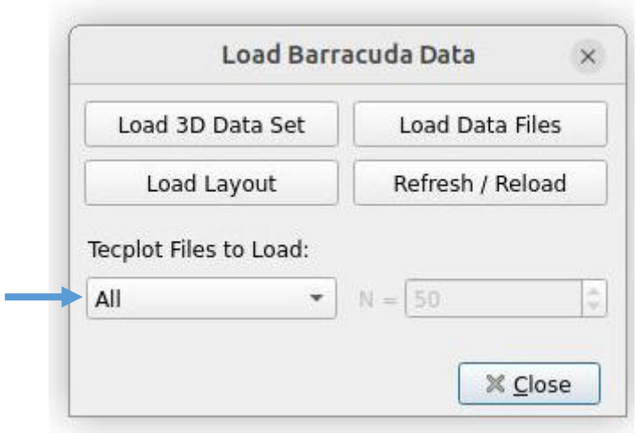
PyTecplot: Calculate Time-Average Over Custom Period

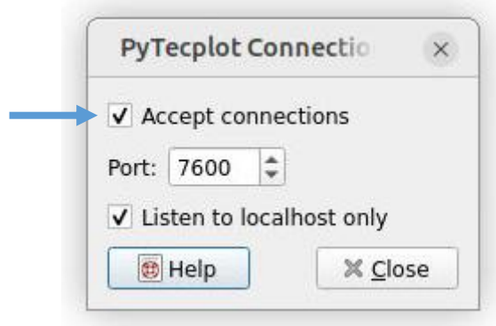
The necessary scripts are already in your directory for this example

- Can also be downloaded from https://cpfd-software.com/tecplot-for-barracuda-calculating-average-values-over-a-custom-time-period/

Typical steps for running the scripts:

- In Barracuda GUI:
  - Open terminal
  - View Results to open Tecplot and load latest results file
- In Tecplot:
  - File → Load Barracuda Data → All → Refresh / Reload
  - Scripting → PyTecplot Connections → Accept connections

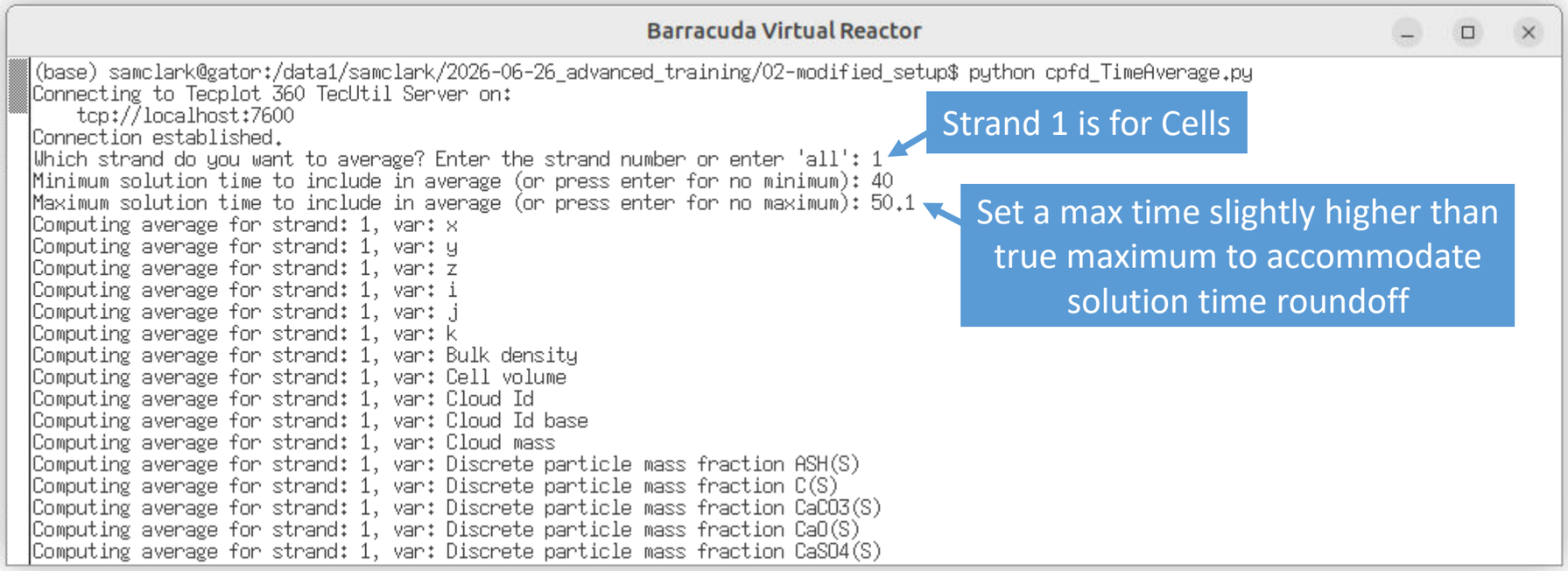




PyTecplot: Calculate Time-Average Over Custom Period

In the terminal you opened from the Barracuda GUI, run the command:

- python cpfd_TimeAverage.py

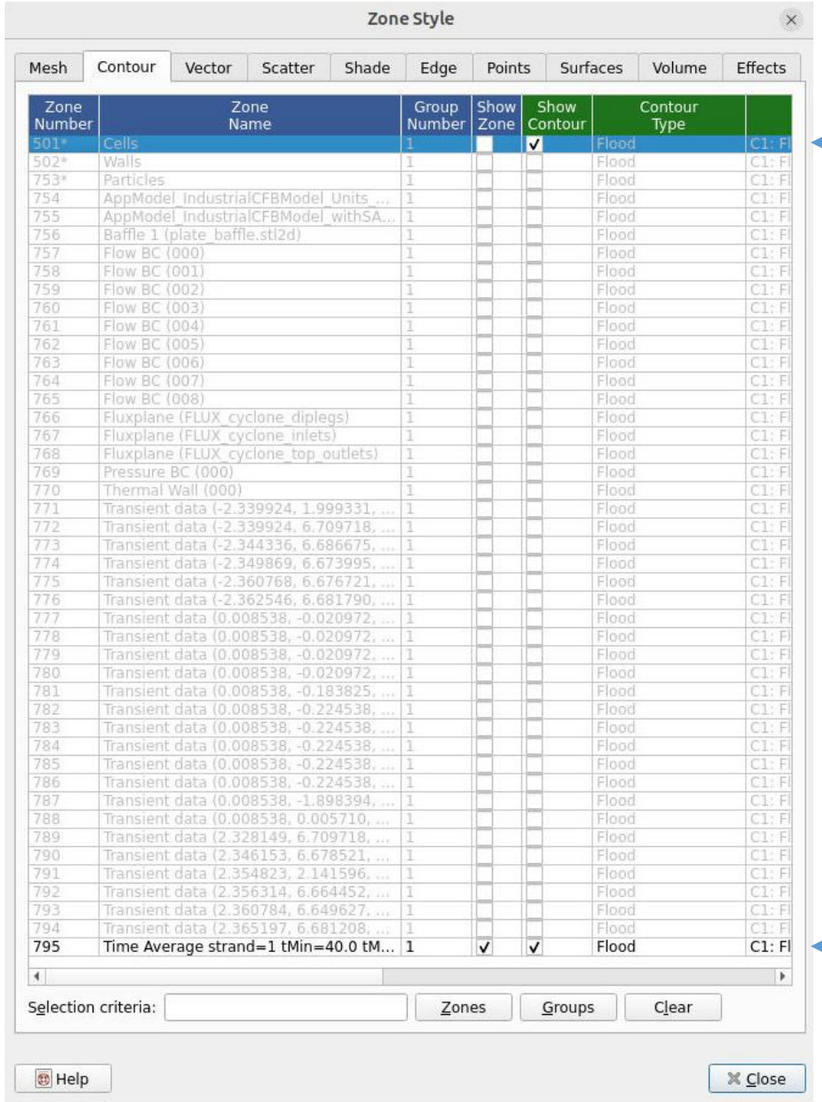


PyTecplot: Calculate Time-Average Over Custom Period

In the Tecplot GUI:

1. Quick Macro Panel → Cells: Fluid Velocity Magnitude
2. Zone Style dialog → Uncheck "Show Zone" for Cells
3. The last Zone in the list should be the only one enabled → this is the calculated time average zone

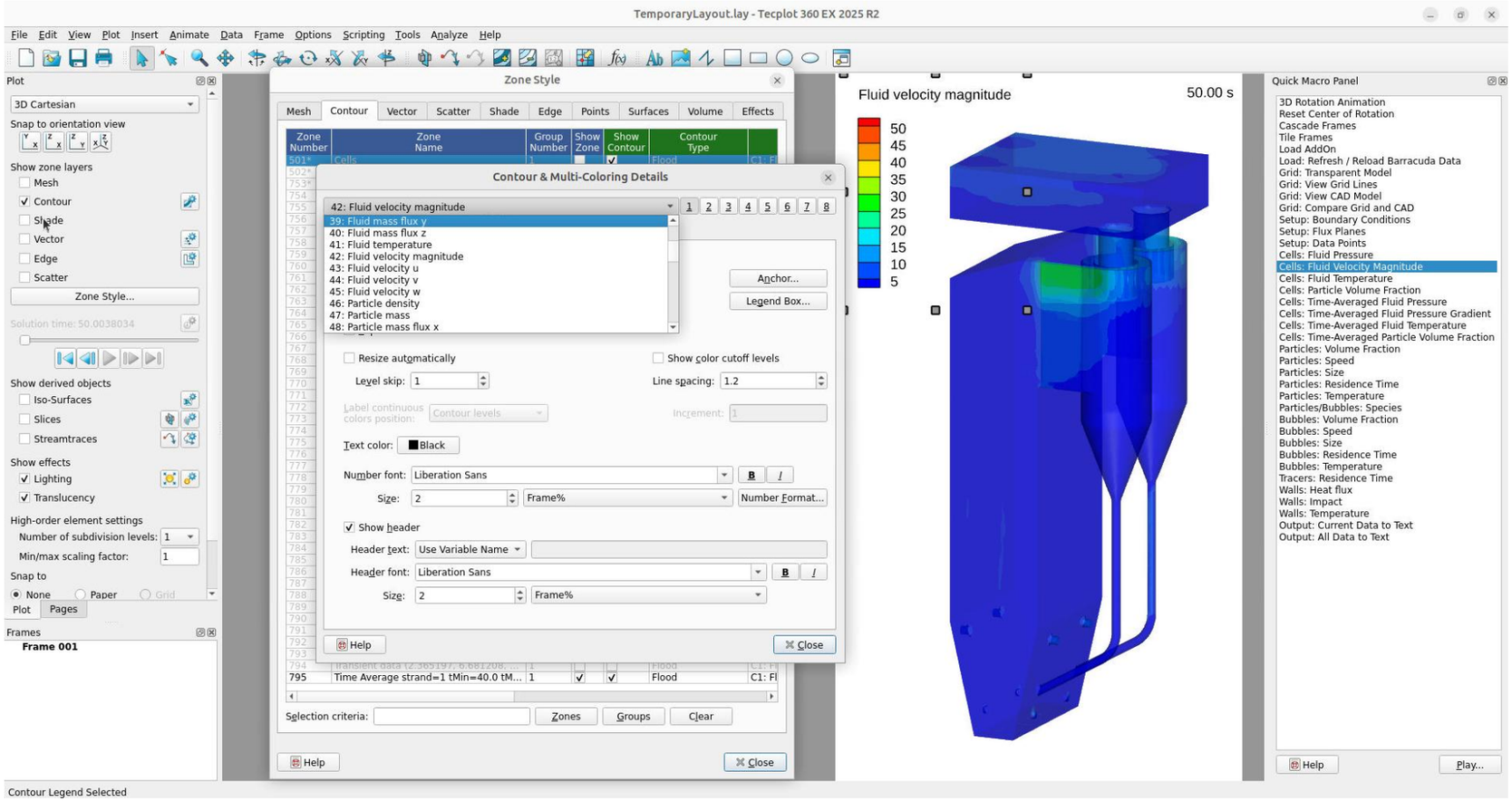
Any cell-based data you plot will now by from this zone, which is the time average from 40 to 50 s



PyTecplot: Calculate Time-Average Over Custom Period

Double-click the Contour Legend to select different variables

All cell-based variables will represent the time average values over the custom time period



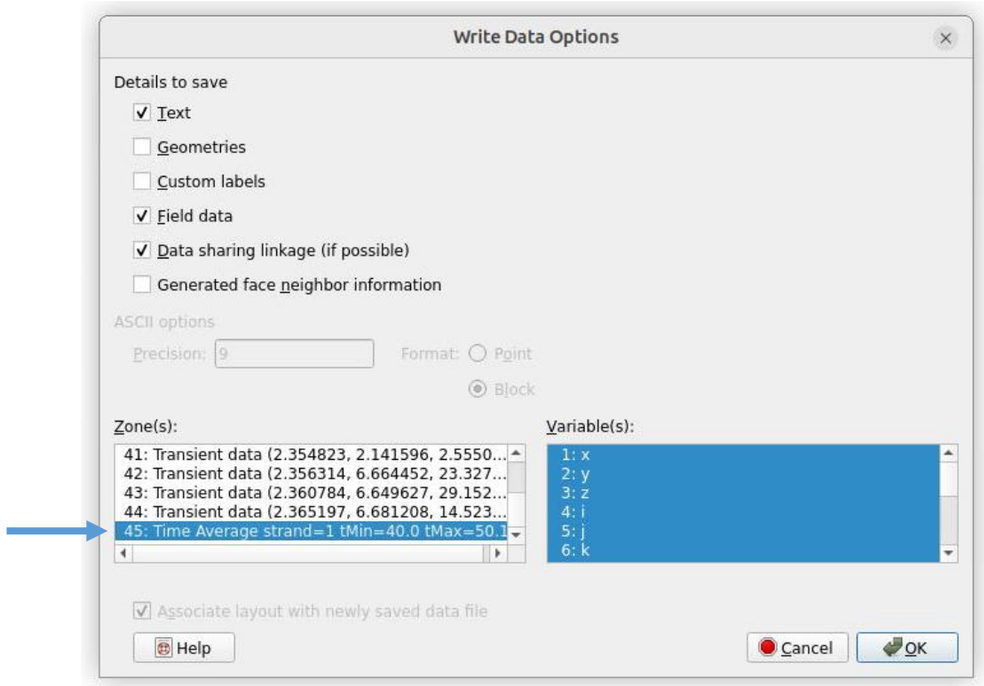
PyTecplot: Calculate Time-Average Over Custom Period

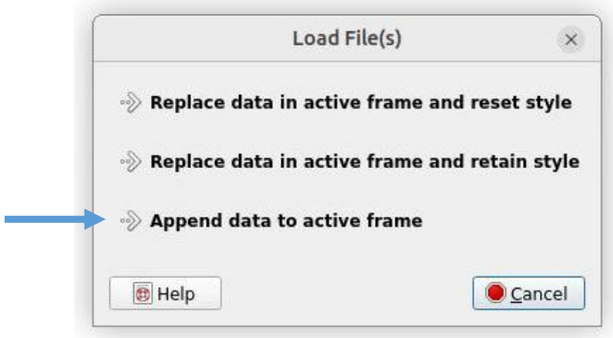
To save calculated time average results:

- File → Write Data…
- Enter a file name
- Select only the time average zone
- Save

To load calculated time average results:

- From Barracuda GUI: View Results
- File → Load Data…
- Select the time average data file that was previously saved
- Append data to active frame





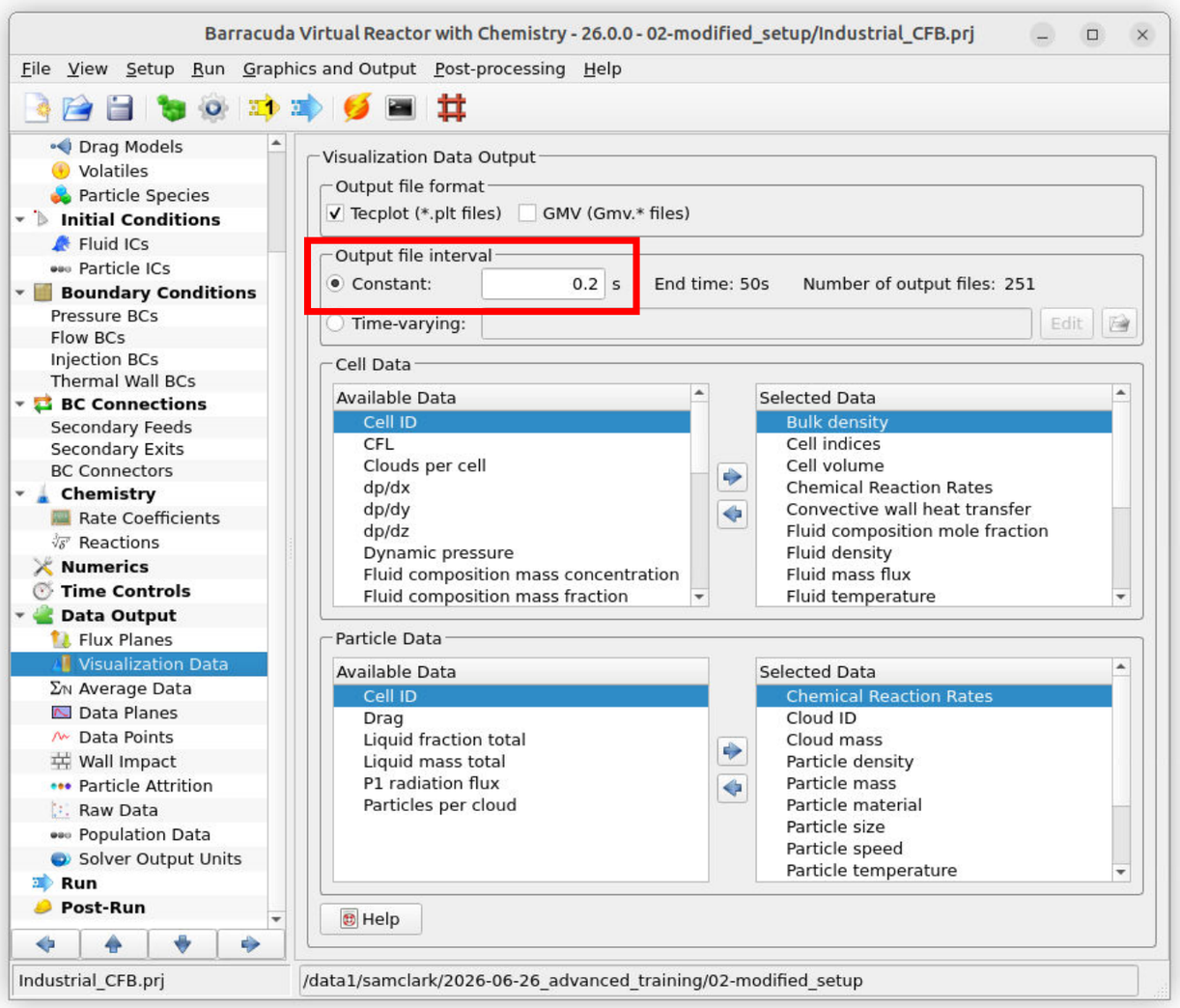
PyTecplot: Calculate Time-Average Over Custom Period

Advantages:

- Possible to calculate time average over any desired time period
- Useful if you forgot to enable time average data before you ran a simulation

Disadvantages:

- Averaging resolution is limited to the output file interval for Tecplot files



PyTecplot: Calculate Time-Average Flow Uniformity

Gas flow uniformity can be a useful metric for quantifying fluidization quality

- GMV-focused post-processing steps were presented at the 2018 European User Group Meeting in the Advanced Training session
- Today we'll cover how to calculate gas flow uniformity using PyTecplot

Calculating Gas Uniformity Index

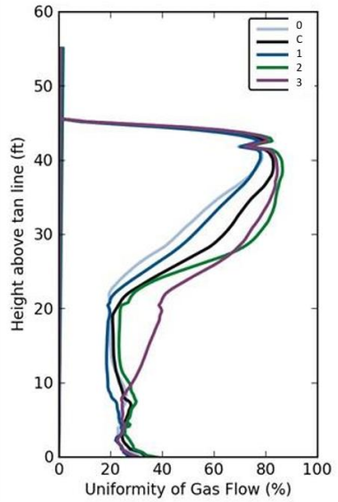
Quantification of time-averaged gas flow shown (Uniformity Index*)

$$U = \frac{Cross\text{-}sectional\ area\ being\ used\ for\ gas\ flow}{Total\ cross\text{-}sectional\ area}$$

Calculated using:

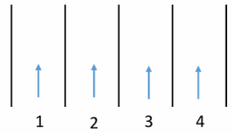
- GMV output data, converted to text format (gmv2txt.py)
- Python (Jupyter notebook)

* See Fletcher, R. et. al., "Identifying the Root Cause of Afterburn in Fluidized Catalytic Crackers", AFPM 2016 Annual Meeting, AM-16-15.



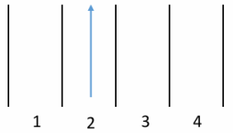
Simple Example of Uniformity Calculation

Consider a very simple system with 4 channels



Channel	Flux	Area	A*F	A*F^2
1	2	1	2	4
2	2	1	2	4
3	2	1	2	4
4	2	1	2	4
Sum		4	8	16
Uniformity =			100%	

*Uniformity = (sum(A*F))^2/sum(A*F^2)/sum(A)*



Channel	Flux	Area	A*F	A*F^2
1	0	1	0	0
2	8	1	8	64
3	0	1	0	0
4	0	1	0	0
Sum		4	8	64
Uniformity =			25%	

*Uniformity = (sum(A*F))^2/sum(A*F^2)/sum(A)*

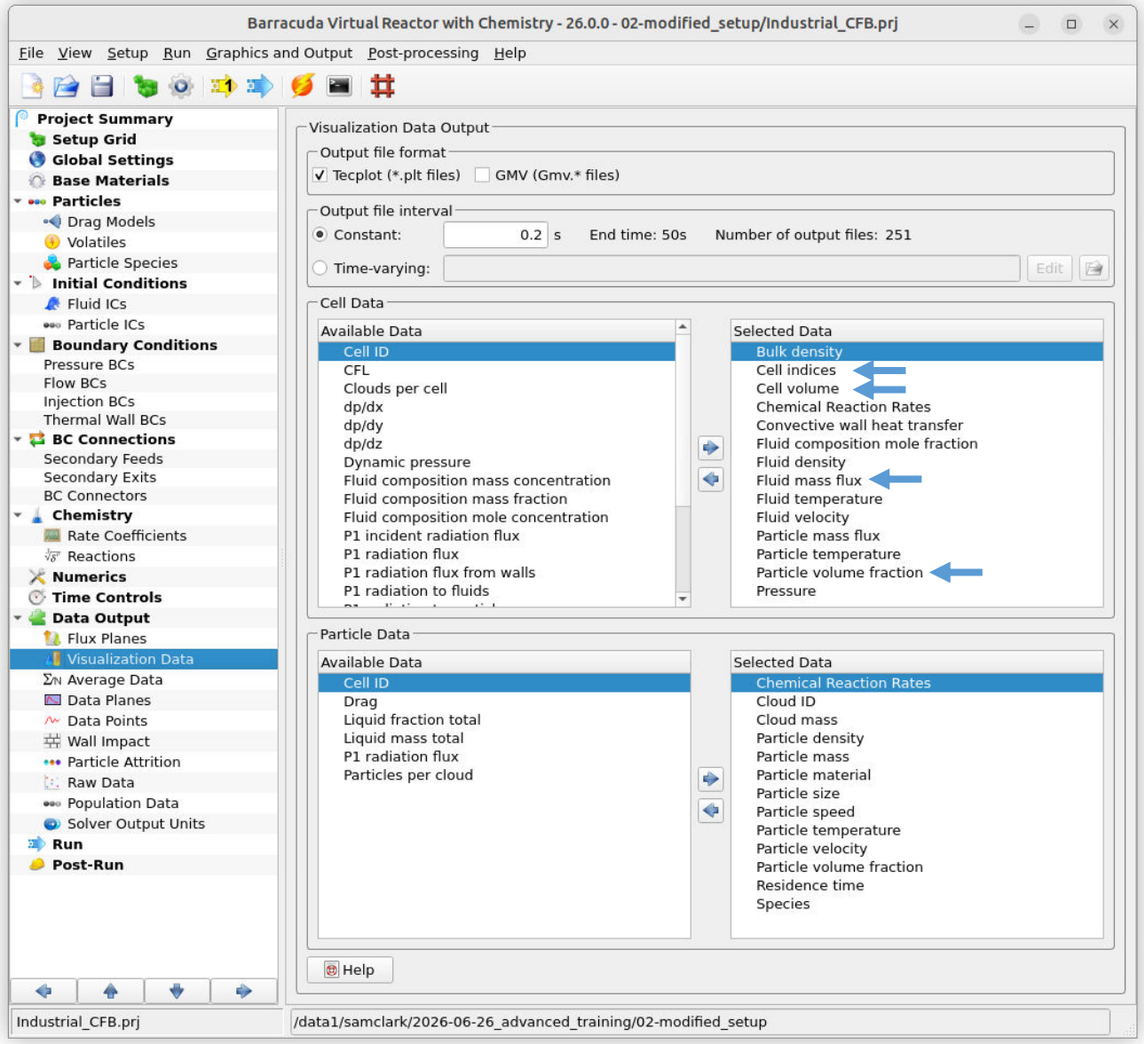
PyTecplot: Calculate Time-Average Flow Uniformity

To calculate instantaneous flow uniformity using this method, you must select the following Visualization Data outputs for <u>Cell Data</u>:

- Cell indices, Cell volume, Fluid mass flux, Particle volume fraction

To calculate time-average flow uniformity, select the following Average Data:

- Fluid mass flux, Particle volume fraction



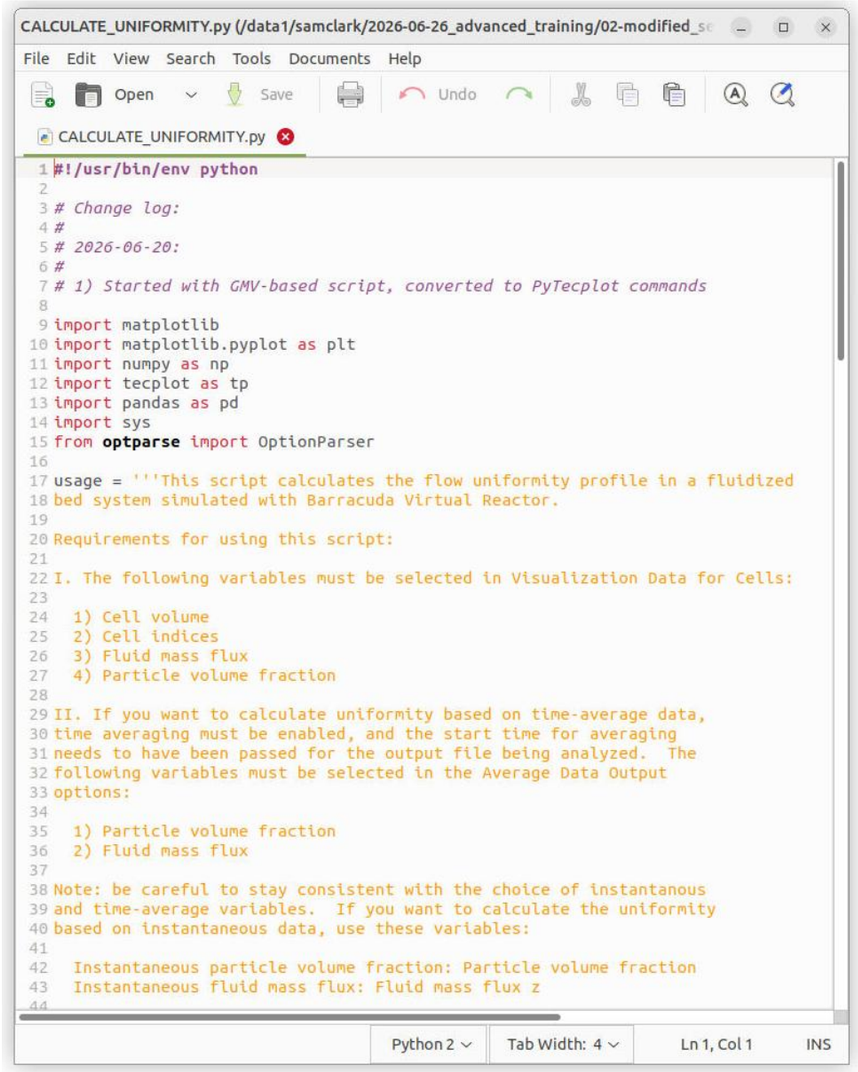
PyTecplot: Calculate Time-Average Flow Uniformity

A Python script is present in your simulation directory:

- CALCULATE_UNIFORMITY.py
- This script will also be posted on CPFD's website for the Users' Conference

You can view the script's contents on the cloud-based training machine by using the command:

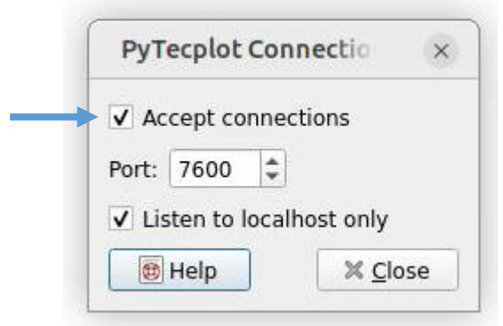
- `gedit CALCULATE_UNIFORMITY.py`

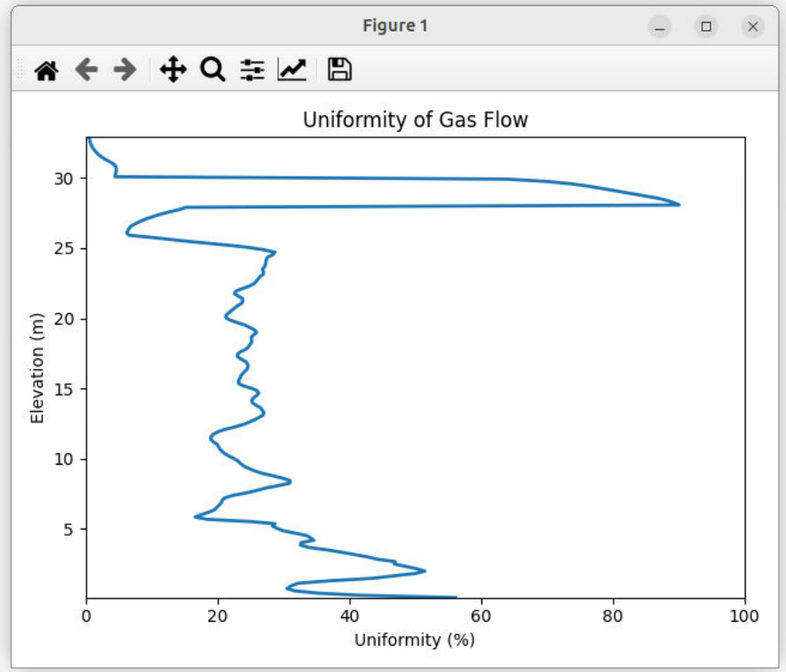


PyTecplot: Calculate Time-Average Flow Uniformity

Run the CALCULATE_UNIFORMITY.py script:

- In Barracuda GUI:
  - Open terminal
  - View Results to open Tecplot and load latest results file
- In Tecplot:
  - File → Load Barracuda Data → All → Refresh / Reload
  - Scripting → PyTecplot Connections → Accept connections
- In the terminal opened from Barracuda GUI:
  - **`python CALCULATE_UNIFORMITY.py`**
  - A plot of gas flow uniformity vs height should pop up





PyTecplot: Calculate Time-Average Flow Uniformity

The default behavior of the CALCULATE_UNIFORMITY.py script is to analyze the entire simulation domain

For our current example, we have upflow in the riser section of the CFB and downflow in the cyclone barrels and diplegs → it would probably be better to isolate just the regions with upward gas flow

In the terminal, recall your last command (up arrow) and add the following arguments to the end:

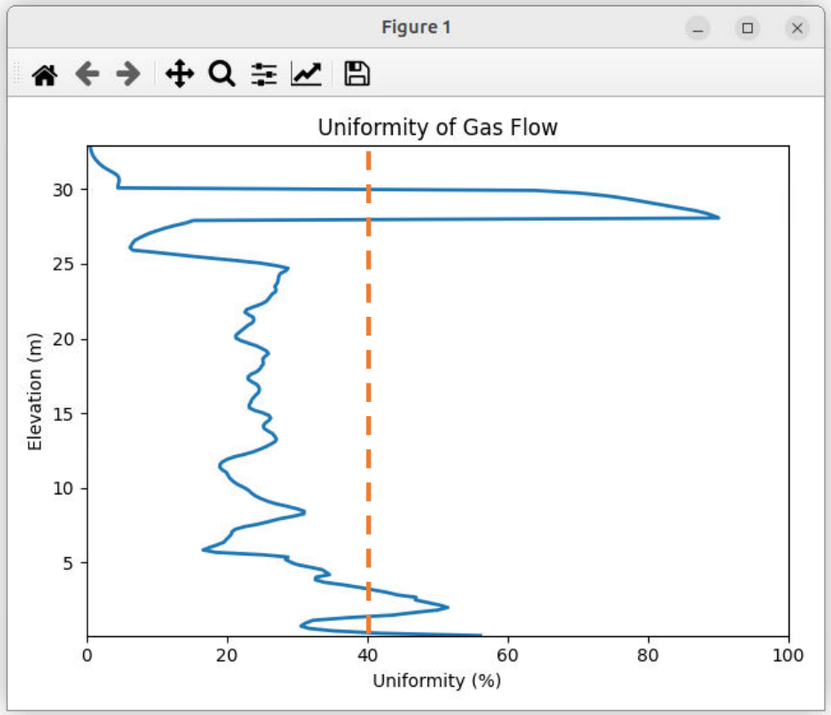
- `python CALCULATE_UNIFORMITY.py --yMax 2.75 --zMax 28`

Options to limit region of space for calculating gas flow uniformity

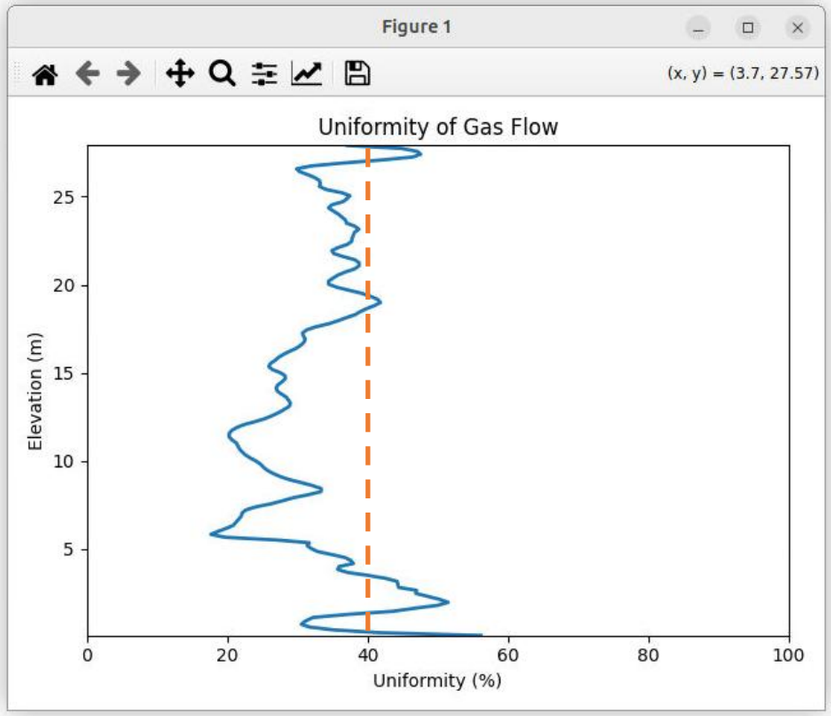
PyTecplot: Calculate Time-Average Flow Uniformity

Run the command with spatial limiters and compare the calculated flow uniformity profile with the original full-domain results

Full domain uniformity results



Spatially limited uniformity results



Exporting Calculated Python Results to Excel Format

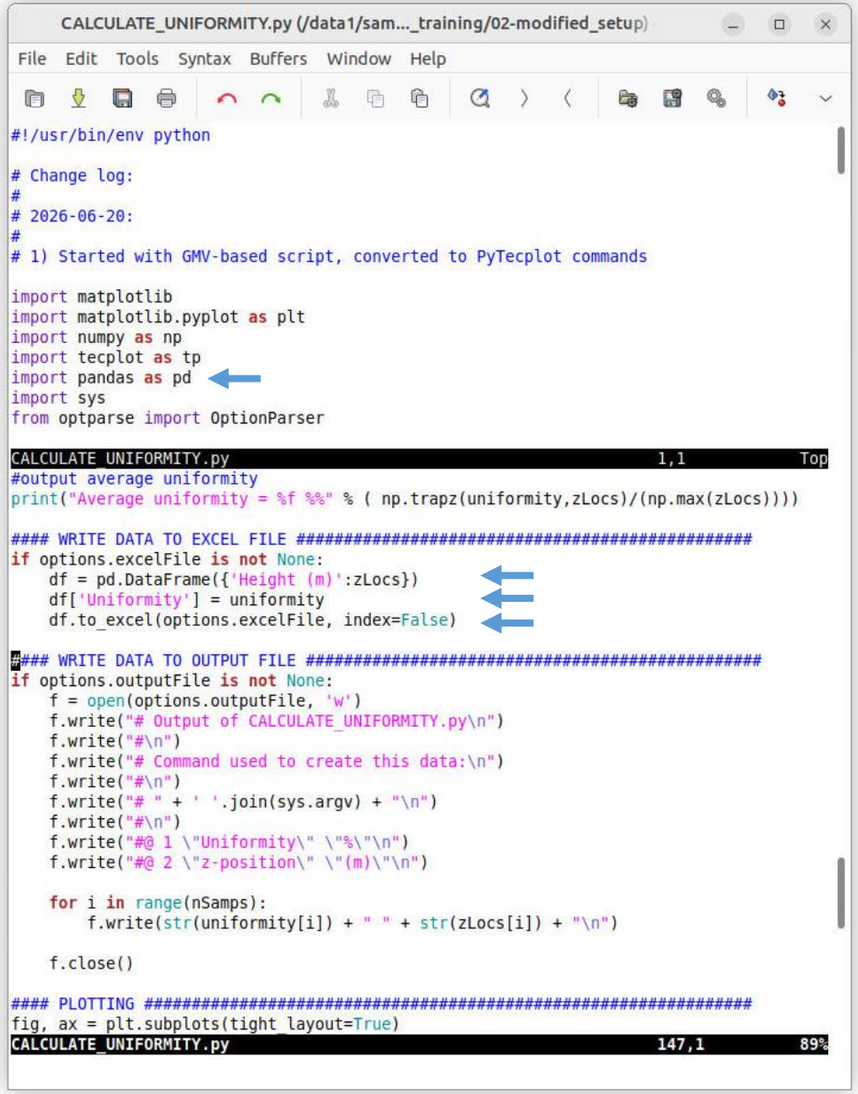
It can often be useful to export data from Python (Numpy) arrays to Excel for further processing

The Pandas library provides an easy method

```
import pandas as pd
# ... create some Python Numpy arrays ...

# Define a Pandas dataframe and add arrays to it
df = pd.DataFrame({'Time (s)':t})
df['Time step (s)'] = dt
df['CFL'] = cfl
# ... and more lines if you have more arrays

# Output the DataFrame to Excel format
df.to_excel('my_filename.xlsx', index=False)
```



Open Discussion and Q&A

Thank you for attending the Advanced Training Workshop

Questions?

- … about material covered today
- … about any other Barracuda Virtual Reactor features

We're here to support you. Contact us at support@cpfd-software.com