OpenParEM2D
User Manual

v.1.1

Applications for OpenParEM2D

- The typical applications for OpenParEM2D are transmission line and waveguide design.
- The outputs from the simulations are propagation constants, attenuations, impedances (either modal or matrix), and field values at given points.
- OpenParEM2D is a full-wave 2D electromagnetic (EM) simulator using the finite element method (FEM).
- As a 2D solver, it can simulate any transmission line or waveguide for the dominant and higher-order modes.
- As a full-wave solver, solving Maxwell's equations including both electric and magnetic fields and their interactions, OpenParEM2D can solve to arbitrarily high frequencies.
- However, as a full-wave solver, it cannot solve to arbitrarily low frequencies due to the decoupling of Faraday's and Ampere's laws at low frequency.
  - The lower frequency limit primarily depends on the dimensions of the cross-section.
- As an FEM method, it can solve arbitrary 2D cross-sections.

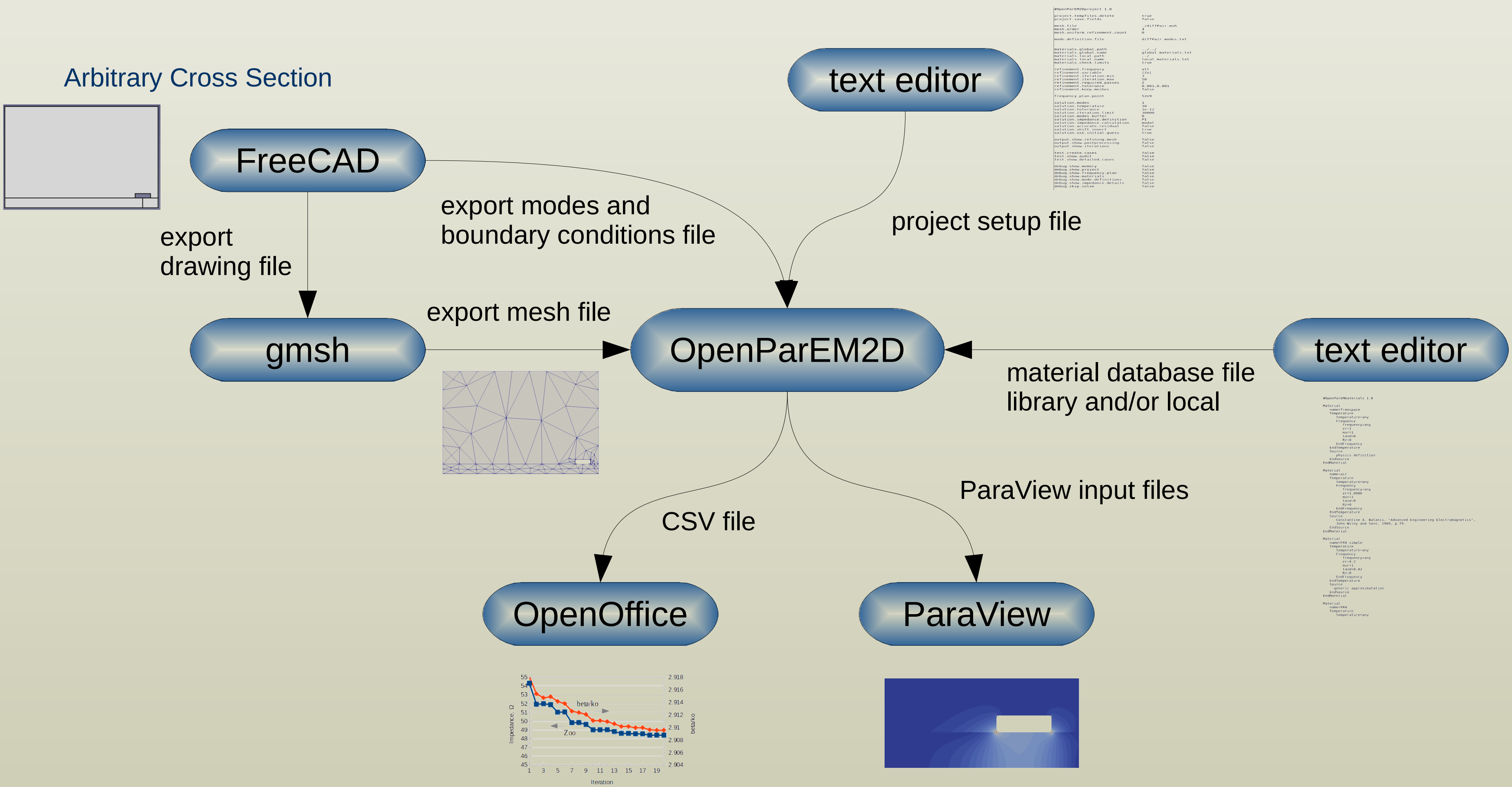
Steps to Solve a Problem

- Generate the input files for OpenParEM2D
  - Project control file
  - Finite element mesh file
  - Boundary conditions and modes file
  - Materials File
- Run OpenParEM2D
- View results in ParaView
- Plot results using a spreadsheet

- OpenParEM2D input files can be generated by hand or by various support scripts or programs.

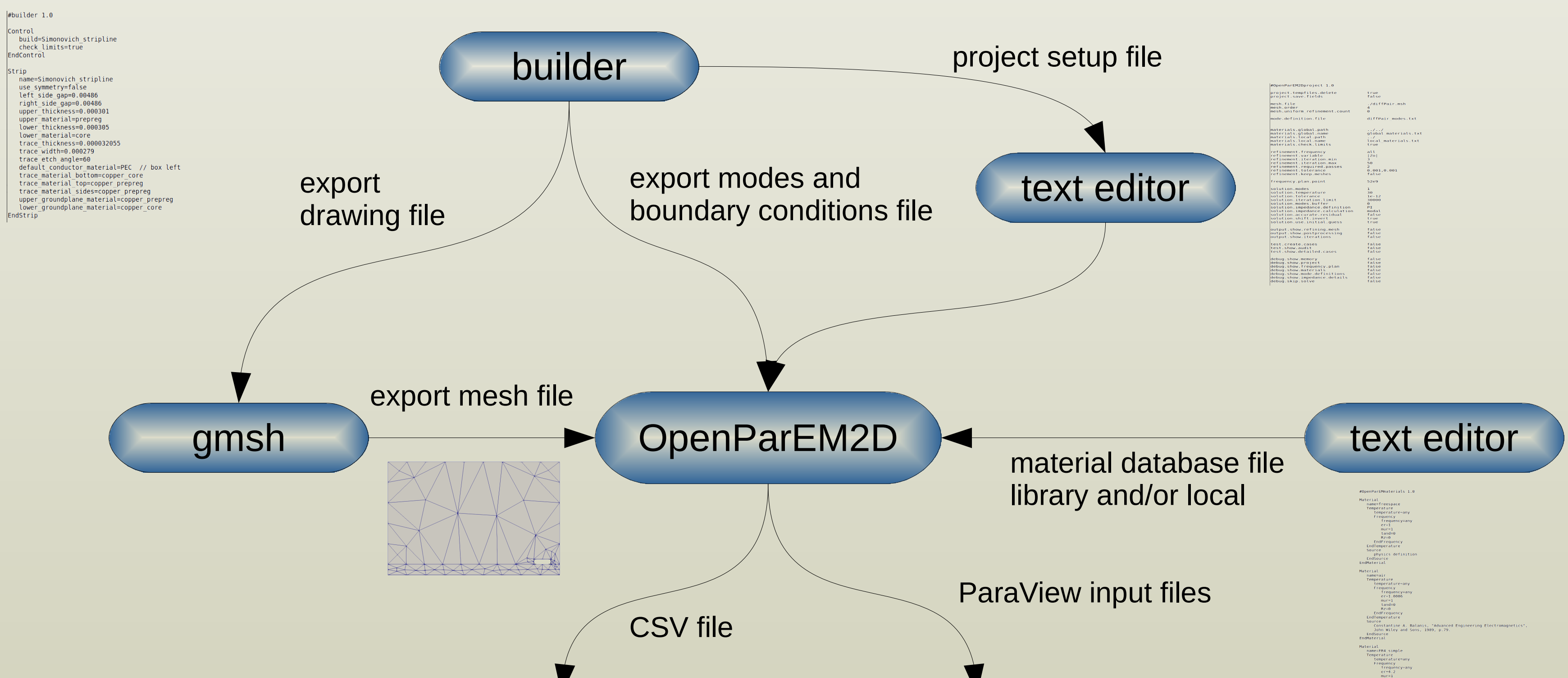
Workflow without "builder"

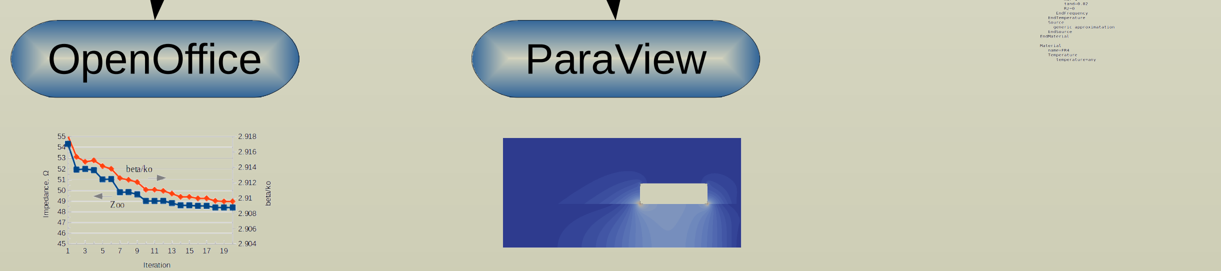
- OpenParEM2D is a command-line tool with text inputs and outputs
- Use with open source tools to create a complete workflow
- Workflow used for development:



Workflow with “builder”

- For transmission line and waveguide types supported by the front-end tool “builder”, the flow is greatly simplified.





Supported as of This Writing

- rectangular waveguide
- microstrip
- symmetric coupled microstrip
- asymmetric coupled microstrip
- stripline
- symmetric coupled stripline
- asymmetric coupled stripline

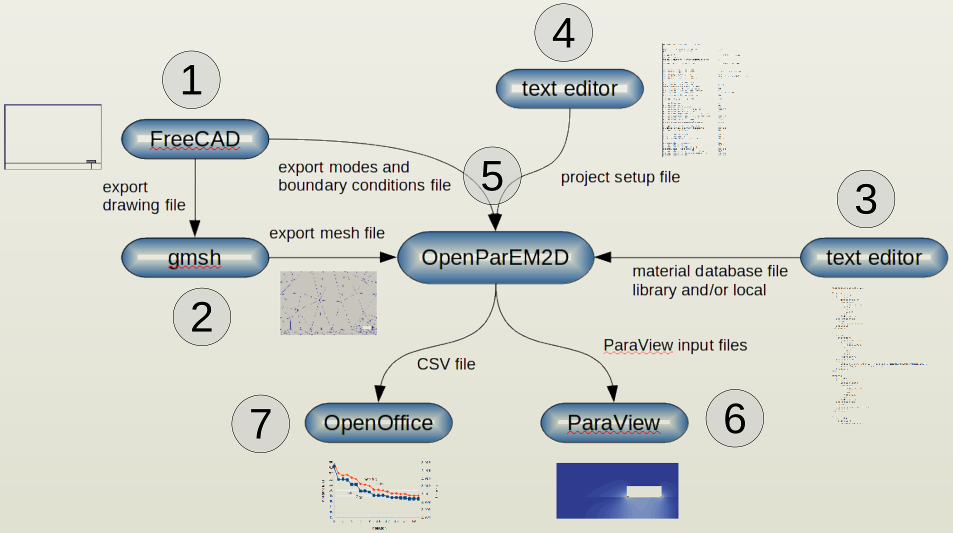
Workflow

without "builder"

Use to Solve Arbitrary Cross Sections

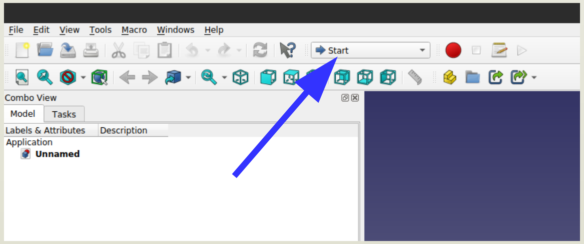
Workflow without "builder"

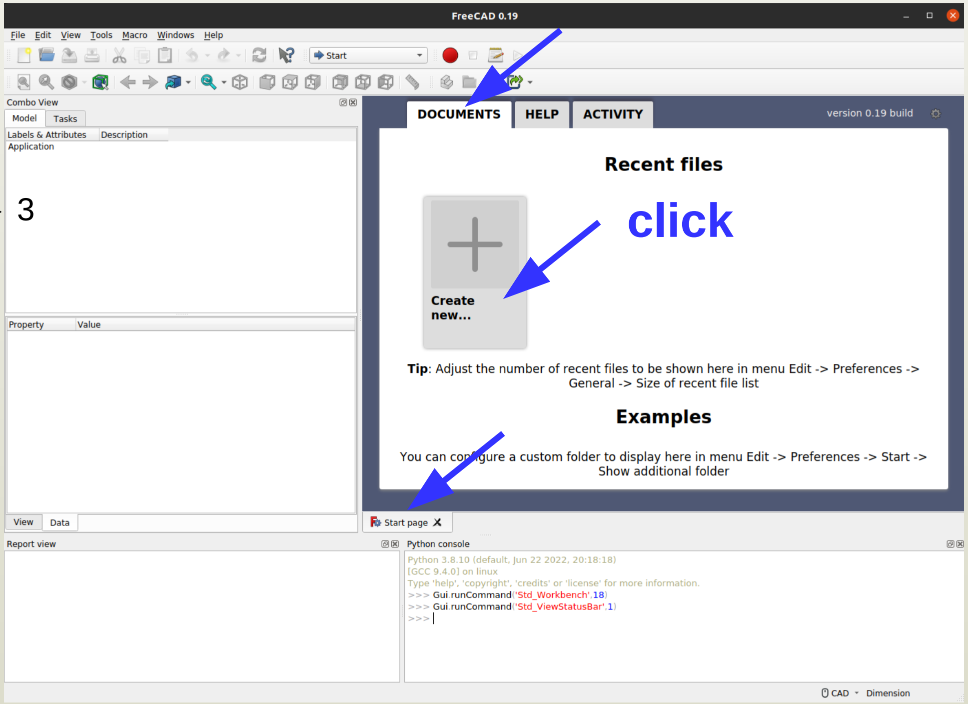
1 Run FreeCad
  - Draw the geometry
  - Add borders
  - Add ports
  - Run the macro "OpenParEM2D_save.py"
    - This saves a boundary/modes file.
  - save a brep file
2 Run gmsh
  - $ gmsh -format msh22
  - Setup and evaluate for the best mesh quality
  - Save the mesh.
3 Locate a global library file to use or write a local materials file
4 Write a project file
  - Pull in the boundary/modes file (from 1), the mesh file (from 2), and a material file or files (from 3)
  - Easiest to copy from a similar project and customize
5 Run OpenParEM2D
  - example: $ mpirun -np 5 OpenParEM2D project_file.proj
  - example: $ OpenParEM2D project_file.proj
6 Run ParaView
  - Requires `project.save.fields true`
  - Run the macro "field_plot" to generate a standard set of field plots
7 Run OpenOffice
  - Pull in csv output into a spreadsheet to make plots



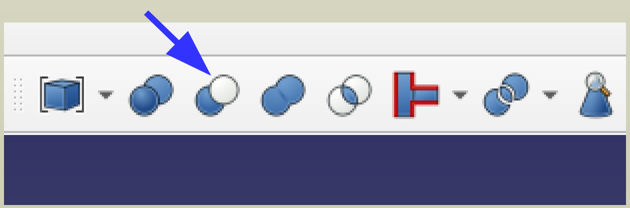
1 FreeCAD

- This document is using FreeCad 0.19
- start FreeCad: $ freecad &
- at the tab “Start page”, tab “DOCUMENTS”, click “Create new ...”
- Edit → Preferences → tab “Units” → MKS, “Number of decimals” → 3
- Switch mode from “Start” to “Draft”



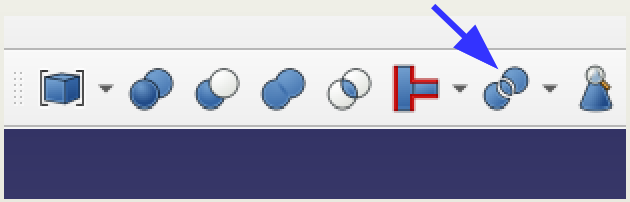


- Draw the structure.
  - Set "Make Face" as true as each primitive is drawn.
    - This is found in the properties section for each object.
    - This can be changed at any time after the element is drawn.
  - Name the primitives by editing the "Label" property.
    - Use the name of the material that represents the area.
    - This can be changed in the mesh file later, if needed.
- Switch mode from “Draft” to “Part”
  - Use the icon "Make a cut of two shapes" to eliminate overlaps and make cutouts for conductors.



Any surface that faces nothing or an unfilled object becomes a perfect electric conductor (PEC) by default in OpenParEM2D. Make a conductor by leaving a “hole” in the drawing. Any surface facing the outside of the drawing is also a PEC. Any surface can be marked to become a perfect magnetic conductor (PMC) or surface impedance boundary.

- When the drawing is done, the parts need to be unified into a single object for meshing.
  - Select all of the objects, then use the icon “Boolean fragments” to unify them.



  - This step enables gmsh to line up the mesh elements along boundaries.
  - If you need to go back to primitive objects, you can select the "Boolean Fragments" object, then right click "Delete", and the primitives are restored.
- Even though the units are set to MKS, on export, FreeCAD will likely output the object dimensions in mm instead of m. OpenParEM2D requires all physical dimensions in m, not mm. A work-around is to scale the drawing down by 0.001.
  - Switch the mode from “Part” back to “Draft”.
  - Select the boolean object, then Modification → Clone
  - Select the cloned object, then in the properties box, set Scale → x → 0.001 and Scale → y → 0.001.
  - Work with the scaled object from now on.
  - Tip: You can hide objects by selecting the object then pressing the space bar. Hiding the original unscaled boolean object is a good idea at this point.
- Add paths to enable the definitions of boundary conditions and modes.
  - Boundary conditions
    - Any surface facing the outside of the boolean object, which may be actually inside the object as a hole, is by default PEC. For a surface that needs another boundary condition, a path must be defined on that surface.
  - Modes
    - Voltages - For each voltage integration line, a path is required from one conductor to another.
    - Currents - For each current integration path, a closed path is required on the contour of the conductor.
- Process to add a path
  - Paths can be added by drawing a line, a polyline, or a rectangle, which ever is most convenient. Any drawn path does need to line up precisely with the boolean object, so be sure to use the snap option.
  - For each path, edit the Label in the properties box, and give it a unique name prepended by _P, so _Pname.

- Add a boundary. Repeat for additional boundaries.
  - Set the mode to “Draft” if the drawing is in another mode.
  - Add a Text object to the drawing.

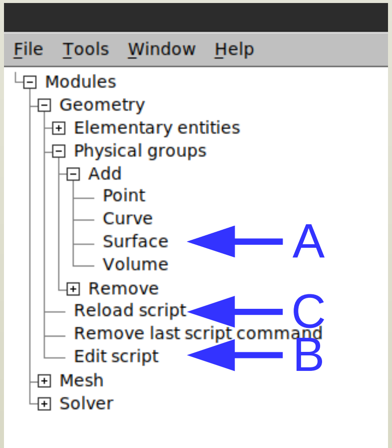
    - This is just a trick to get an object for which a unique label can be set.
    - Click anywhere outside the boolean object.
    - Set the text to anything, but a period is nice and small.
    - Click “Create text”.
  - Select the newly created text object, then edit its Label in the properties box.
    - The format for the label is _Bname(SI|PEC|PMC){[+|-]path,-path,+path,...}
    - “name” is a unique name for the boundary
    - Select the boundary type by keeping either SI, PEC, or PMC. Note that PEC is the default, so it does not need to be set. It remains an option here in case it comes in handy some day.
    - Choose the physical location for the boundary by listing previously defined paths by name, not including the _P prefix.
      - If more than one path is listed, they are chained together in order to create a more complex path.
      - With no sign or with a + sign, the path is included without reversing the direction.
      - Use a - sign to reverse the direction of the path.
- Add a port. Repeat for additional ports.
  - Add a text object using the procedure described for boundaries.
  - Select the newly created text object, then edit its label in the properties box.
    - The format for the label is _Mnumber(voltage|current){[+|-]path,-path,+path,...} or _Lnumber(voltage|current){[+|-]path,-path,+path,...}. Use _M if a modal impedance calculation is being done and _L for a lines calculation.
    - “number” is an integer indicating the mode. For N modes, N ports are needed numbered from 1, 2, …, N.
    - Select whether the port is for a voltage or current calculation.
  - Two ports can be defined for each mode number, one for voltage and one for current.

- Export the brep file.
  - Select the boolean fragment object, then File → Export.
  - Open the saved brep file in a text editor and scan the dimensions for correct scale. They should be in m. If they are not, then re-visit the scaling, correct as needed, and export a new brep file.
- Export the boundary/modes file.
  - Macro → Macros … → select OpenParEM2D_save.py → Execute
- Save the project and exit FreeCAD, if desired.

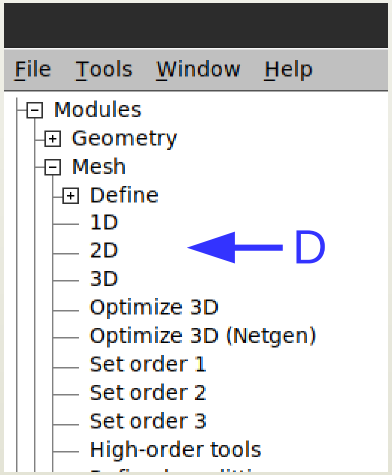
Note: This process is obviously not simple, although it does get easier after a couple of passes through it. Streamlining the process with additional scripting could certainly make it much easier to set up boundaries and ports. For example, being able to click on an edge then right-click to bring up a boundary definition box would be very convenient.

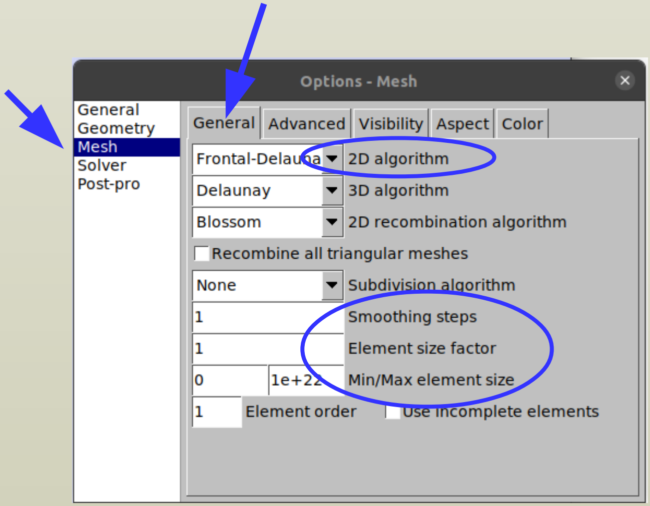
2 gmsh

- This document is using gmsh version 4.8.4
- Start gmsh: $ gmsh -format msh22 &
- Load the brep file exported from FreeCAD
- Add physical groups for the dielectric regions.
  - Geometry->Physical groups->Add->Surface [see selection A in the figure]
  - Name the surface in the pop-up, click the surface, then press the key "e" to end.
    - The name of the surface is the material name, and the material name must appear in a materials file called out in the project file.
  - For the first region, gmsh needs to create a geo file.
    - click “Yes” on the “Create new 'geo' file” button.
  - Repeat for each surface, where a name is entered, the region is clicked, and the key “e” is pressed.
  - When all regions have been entered, press “q” to quit.
- Fix the region numbering.
  - Geometry->Edit script [see selection B in the figure]
  - For each region, set the number next to the surface name to match the number in the curly brackets.
    - Note that the numbers in the curly brackets must start with 1 and go sequentially to higher numbers. If this is not the case, then go back and try again.
  - Save and exit the text editor.
  - Geometry->Reload script [see selection C in the figure]
    - The changes do not take effect until the script is re-loaded.



- Mesh the layout
  - Mesh → 2D [see selection D in the figure]
  - Check mesh alignment at boundaries
    - Visually check the mesh to make sure that the meshes line up at the boundaries. If they do not line up, then the boolean fragments step was not properly executed in FreeCAD. In this case, delete the brep and geo files, go back to FreeCAD to the boolean fragments step, fix the drawing, then re-export a brep file.
- Improve mesh quality, if needed
  - Visually evaluate the mesh.
    - Ideally, the mesh consists only of equilateral triangles.
    - If the mesh has only nice, fat triangles, then proceed to exiting gmsh.
  - Long, thin triangles degrade the condition number of the matrices when solving the eigenvalue problem. As the mesh becomes worse, the iterative eigenvalue solution takes more iterations to find a solution. At some point, poor mesh quality can cause the eigenvalue solver to fail to find a solution.
  - Bring up general options
    - Tools → Options.
  - Show meshing options
    - Select "Mesh"
    - Select the "General" tab
  - Try out the circled options to improve the mesh quality.
    - After each change, reload the script [C from the prior slide], then click Mesh → 2D.
    - The general goal is the smallest mesh with the best mesh quality.
  - When the mesh looks good, proceed to saving the mesh.
- Save the mesh
  - File → Save Mesh
  - File → Quit [if desired]





3 Material Database

- Ideally, a materials file is available as a site-specific materials library.
- If a materials file does not exist, one must be written that meets the specifications for materials files in the file “specifications.pdf”.
  - A good starting point is the file “global_materials.txt” in the regression directory of the src distribution.

4 Project File

- OpenParEM2D is controlled through a text file called the “project file”.
- The user must create the project file.
  - The specification for the project file can be found in file “specifications.pdf”.
  - The easiest way to create a project file is to copy an existing project file then customize it.
- The project file pulls in the boundary/mode file exported from FreeCAD, the mesh file exported from gmsh, and the materials file, either referencing a library file or freshly created [or both].

```
#WR90_order_3_refinement_no_mesh_reuse/WR90.proj

#OpenParEM2Dproject 1.0
project.save.fields           true
mesh.file                     ../WR90_waveguide.msh
mesh.order                    3
mode.definition.file          ../WR90_waveguide_modes_no_Zs.txt
refinement.frequency          all
refinement.variable           |gamma|
refinement.iteration.max      20
refinement.tolerance          1e-8
materials.global.path         ../../../
materials.global.name         global_materials.txt
materials.local.path          ./
materials.local.name
frequency.plan.linear         9e+09,1e+10,1e+09
solution.modes                5
solution.temperature          20.
solution.tolerance            1e-12
solution.iteration.limit      10000
solution.modes.buffer          4
solution.impedance.definition PV
```

mesh file

boundary/mode file

materials file

to run this example:
$ cd regression/WR90_rectangular_waveguide/WR90/WR90_order_3_refinement_no_mesh_reuse
$ mpirun -np 5 OpenParEM2D WR90.proj

- An example project file from the regression suite is shown to the left
  - defaults are removed with the utility “simplify”
- Other inputs are custom for this project

As a description, this project solves a WR90 waveguide for the first 5 modes using 3rd order finite elements at 9 and 10 GHz. The mesh is adaptively refined at both frequencies starting from the initial input mesh. The tolerance is set fairly tight at 1e-12. The power-voltage definition is used to calculate the impedance. Four extra modes are calculated to ensure that the requested 5 non-null modes are found.

5 OpenParEM2D

- Execute OpenParEM2D with one of the following commands:
  - $ mpirun -np N OpenParEM2D project_file.proj
    - Run using N processors
  - $ mpirun -np 1 OpenParEM2D project_file.proj
    - Run using 1 processor
  - $ OpenParEM2D project_file.proj
    - Run using 1 processor
- OpenParEM2D runs and provides errors, if any, plus status and results to the screen.
  - Re-direct the output to save the screen-printed results as a log file.
  - The output includes the line “mesh worst element aspect ratio: X <> target: 5”. If X is much greater than 5, perhaps greater than ~10, then revisiting the mesh quality in gmsh could be in order.
- A lock file is created at the start of execution, and the lock file is removed on exit. If the job dies without a clean exit, the lock file will prevent another execution. To remove the lock file:
  - rm .project_file.lock
- As the job runs, intermediate results are saved
  - project_file_results.csv - see the commented header for the data included
  - project_file_fields.csv [if field.point is included in the project file] - see the commented header
  - Paraview files
- On clean exit, the intermediate results are final.
- Review the information printed to the screen for errors, warnings, and expected behaviors and results.

6 ParaView

- Requires `project.save.fields true` in the project file.
- Run paraview
  - $ paraview &
- Open a file to view a result
  - Navigate to the directory “ParaView_project_file”.
  - There are results for each mode at each frequency.
  - Select result to view and open.
- Run the macro “field_plot” to generate a standard set of field plots.
  - Note that the macro cannot properly set the scaling for the vector plots. Re-scale as needed.
- OpenParEM2D always scales the results such that the tangential electric field is real at its maximum value.
  - So generally, view the real electric field and the real magnetic field for propagating modes and the imaginary magnetic field for cutoff modes.

7 CSV file

- OpenParEM2D always produces a csv file with results for propagation and impedance along with some stats about the run.
  - See the header for what is included. The header is self-explanatory, except for:
    - “converged” is boolean, so 0 means that the simulation did not converge while performing iterative mesh refinement, while 1 means that it did converge.
      - If an adaptive simulation is not requested, then converged is entered as 0.
    - “modal impedance calculation” is boolean, so 1 means that the ports are set up for a modal impedance calculation, and 0 means that they are set up for a line impedance calculation. See the presentation “impedance_guide.pdf” for details on these two setups.
  - These results can be pulled into other tools for further analysis, graphing, etc.
- If the project file specifies any field.point keyword/value pairs, then a second csv file is produced with the computed field results.
  - See the header for what is included.

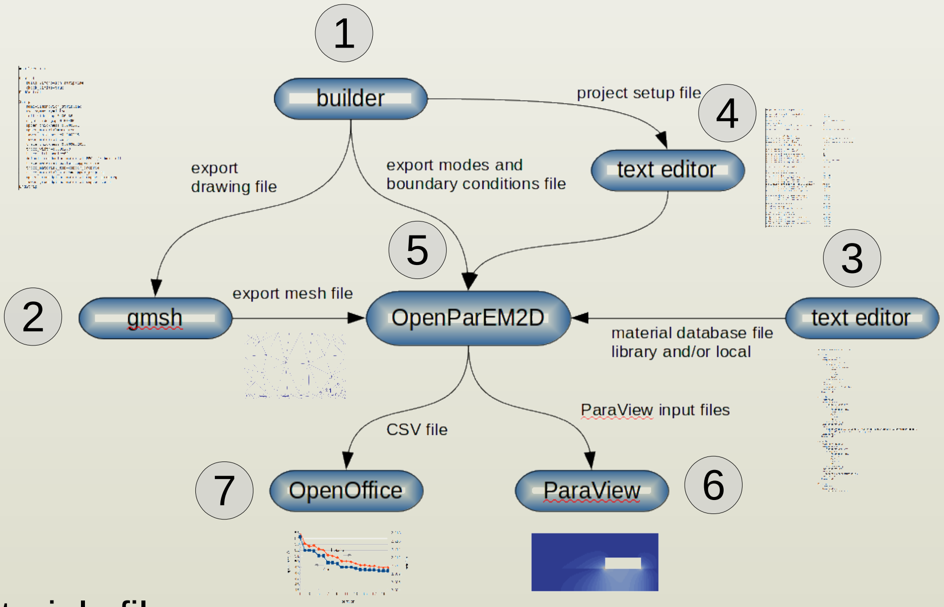
Workflow

with “builder”

Use to Solve Transmission Line and Waveguide Types Supported by builder.

Supported as of This Writing
- rectangular waveguide
- microstrip
- symmetric coupled microstrip
- asymmetric coupled microstrip
- stripline
- symmetric coupled stripline
- asymmetric coupled stripline

Workflow with "builder"



1 Run builder
- Edit a text file to describe the geometry
- Run builder
- builder generates 3 of the needed 4 text files
  - geometry file (*.geo)
  - boundary/modes file (*_modes.txt)
  - template project file (*.proj).

2 Run gmsh
- $ gmsh -format msh22
- Import the *.geo file produced by builder
- Generate and save a 2D mesh

3 Locate a global library file to use or write a local materials file

4 Customize the project file produced by builder (from 1)
- Point to the boundary/modes file (from 1), the mesh file (from 2), and a material file or files (from 3)
- Customize the finite element order, convergence critieria, frequencies, etc.

5 Run OpenParEM2D
- example: $ mpirun -np 5 OpenParEM2D project_file.proj
- example: $ OpenParEM2D project_file.proj

6 Run ParaView
- Requires `project.save.fields true` [output by default from builder]
- Run the macro "field_plot" to generate a standard set of field plots

7 Run OpenOffice
- Pull in csv output into a spreadsheet to make plots

1 builder

- The frontend tool “builder” is included in the OpenParEM2D source tree and compiles/installs along with OpenParEM2D.
  - builder supports a number of transmission line and waveguide types. See the specification for the currently supported types.
- Edit a text file to create an input file for builder.
  - See the specifications document for the keyword/pair format for describing a geometry.
  - Typically, an existing deck is copied as a starting point. See the regression directory tree for examples.
- Run builder
  - $ builder builder.txt
- builder produces 3 output files
  - *.proj - A project file that can be quickly customized.
  - *_modes.txt - The required boundary/modes file. Editing is not required.
  - *.geo - The geometry input file required by gmsh.

Example builder Input File

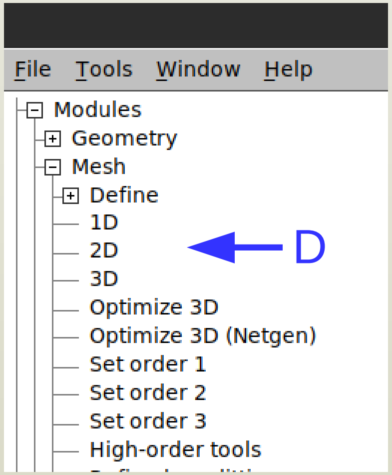
```
#builder 1.0

Control
   build=Simonovich_stripline
   check_limits=true
EndControl

Strip
   name=Simonovich_stripline
   use_symmetry=false
   left_side_gap=0.00486
   right_side_gap=0.00486
   upper_thickness=0.000301
   upper_material=prepreg
   lower_thickness=0.000305
   lower_material=core
   trace_thickness=0.000032055
   trace_width=0.000279
   trace_etch_angle=60
   default_conductor_material=PEC
trace_material_bottom=copper_core
   trace_material_top=copper_prepreg
   trace_material_sides=copper_prepreg
   upper_groundplane_material=copper_prepreg
   lower_groundplane_material=copper_core
EndStrip
```

2 gmsh

- This document is using gmsh version 4.8.4
- Start gmsh: $ gmsh -format msh22 &
- Load the *.geo file exported from builder
- Mesh the layout
  - Mesh → 2D [see selection D in the figure]
  - Check the mesh for visual quality.
  - Apply gmsh features to improve the quality, if needed. See the “2 gmsh” slide in the section “Workflow without 'builder'”.
- Save the mesh
  - File → Save Mesh
  - File → Quit [if desired]



3 Material Database

- Unchanged from the “3 Material Database” slide in the section “Workflow <u>without</u> 'builder'”

4 Project File

- OpenParEM2D is controlled through a text file called the “project file”.
- Edit the template project file produced by builder.
- Customize as needed.
  - Adjust file names and paths for the mesh, boundary/modes, and materials files.
  - Finite element order
  - Convergence criteria
  - Frequencies
  - etc.

5 OpenParEM2D

- Unchanged from the “5 OpenParEM2D” slide in the section “Workflow without 'builder'”

6 ParaView

- Unchanged from the “6 ParaView” slide in the section “Workflow <u>without</u> 'builder'”

7 CSV file

- Unchanged from the “7 CSV file” slide in the section “Workflow without 'builder'”