# CAE of Sensors and Actuators
# Electrostatic Field Example

In this example, we would like to simulate the electric field and the capacitance of a **cylindrical** capacitor, including the surrounding air. The inner electrode consists of a centered rod and the outer electrode is the metallic coat. Between inner and outer electrode a voltage of 1 V shall be applied.

The dimension are as follows

- ◦ Radius of inner electrode = 1 mm
- ◦ Radius of outer electrode = 6 mm
- ◦ Thickness of outer electrode = 1 mm
- ◦ Height of cylinder = 100 mm
- ◦ Size of additional boundary = 10 mm

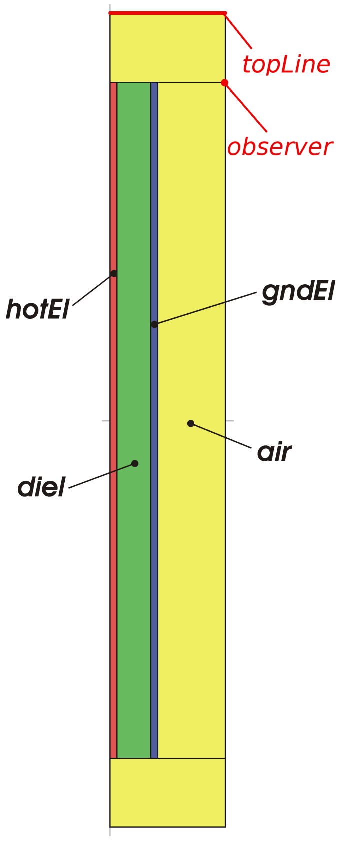


Figure 1: Domain regions of the capacitor model

This example is divided into three parts:

- Part I. Meshing with Ansys
- Part II. Simulation with NACS
- Part III. Postprocessing with Gid

## Part III. Postprocessing with Gid

**Tutorial: October 21st, 2014**
**Further information:**

- `postprocessBasics.pdf` - postprocessing with Gid, Matlab and Gnuplot

### Postprocessing with Gid

- **Preparations**
  Simulate the electrostatic model (→ see Part II) and make sure that a gid output file called `results_gid/nacsTutorial.post` exists.
- **Start Gid-Postprocessing**
  To start postporcessing with Gid you have two possibilities:
  1. start Gid by typing `gid-failsafe` into linux command line, switch to postprocessing via `Files -> Postprocess` and then load your file results file by opening it via `Files -> Open`
  2. right-click onto your results file (e.g. `nacsTutorial.post.bin`) and select `Open With ... -> Custom Command Line -> gid-failsafe` (in that case Gid will automatically switch to postprocessing)
- **Display results with Gid**
  There are various possibilities to display your simulation results. Those can be found and selected
  1. under `View results` → enables direct access to the postprocessing functions
  2. under `Window -> View results` → opens dialog window where one can select not only the functions but also the current timestep or frequency (only relevant for **transient** and **harmonic** simulations)
  3. by clicking on the corresponding icon in the icon bar → quick access, but icons need to be known

  A detailed overview of the most important functions can be found in `postprocessBasics.pdf`. For this electrostatic example we want to visualize both the electric potential and the electric field. From the postprocessing point of view, we have to distinguish between:

  - Scalar fields
  - Vector fields

  The electric potential is a scalar field, i.e. at each node there is only one unknown. Those kind of fields can be visualized either as **contour lines** or as **contour fill**. To create e.g. a contour fill of the electric potential go to
  `View results -> Contour Fill -> elecPotential`.
  To show the minimum and maximum select
  `View results -> Show Min Max -> elecPotential`.
  The electric field on the other hand is a vector field, so at each node exists a vector which stores for each coordinate a value. To visualize those vector fields you can choose one of the following:

  - visualize only one component as scalar field
  - visualize the magnitude as scalar field
  - visualize the vector field (or one of its components) with vectors

To show the electric field with vector arrows go to
`View results -> Display Vectors -> elecFieldIntensity`. The apperance of the vectors can further be adjusted under `Options -> Vectors`. For example you can select that all vectors have the same length (under `Options -> Vectors -> Size`).

- **Entity selection, coloring and mesh view**
  In some cases a given result shall be visualized only on some single region (e.g. only the electric potential in the dielectricum is of interest). For such cases one can manually select which regions are shown under `Window -> View style ....` Here one can switch off the visualization of single regions by clicking on the small light bulbs in column **I/O**.
  The other columns allow further functions like
  - coloring regions and groups → column **C**
  - visualizing only mesh, mesh with results, only results, only nodes, ... → column **St**
  - selecting the order of overlapping regions → column **Tr**
- **Changing the result scale**
  The easiest way to change the result scale is by using the icons in the icon toolbar. There you can find for example icons to change the upper limit (red button numbered with 21 and 17), to set the lower limit (blue button numbered with -7 and -9) or to change to logarithmic scale (colorbar with log).
- **Visualize deformation (only for mechanics)**
  For mechanic simulations Gid offers not only the possibility to visualize the deformation in terms of colors but also as actual deformed geometry. Under `View results -> Deformation` this feature can be switched on.
- **Setup camera view**
  If your simulation model consists of multiple parts or even is three dimension, you will at some point need to setup the camera view, e.g. for zooming in to a region of interest or to rotate your geometry. All options regarding the camera view are placed under `View`.
- **Creating screenshots**
  One very important function, not only for the assignments, is to create screenshots of the results. Screenshots of the current view can be created by clicking on the small camera icon in the top icon bar.
- **Creating animations (only for transient and harmonic simulations)**
  Under `Window -> Animate ...` you can find the Gid animation functionality. It allows visualizing the results and their change over time or frequency. The animation functionally can be combined with any displaying method mentioned above, so you can for example animate the change of the electric potential over time.

**Bonus: Postprocessing with Gnuplot**

- **Preparations**
  Simulate the electrostatic model (→ see Part II) and make sure that text results exist in the `history` folder.
- **Start Gnuplot-Postprocessing**
  To start a gnuplot terminal type `gnuplot` into linux command line.
- **Plot column wise data from file**
  Plotting data files with gnuplot is very easy and gives a quick overview at the simulated results. To show simulation results which are for example stored in a file called `sim_results.hist` type
  `plot 'sim_results.hist' [options]`
  into the gnuplot terminal. Thereby `[options]` is a list of optional tags like

  - `w l` → plot is shown as continuous line
  - `w lp` → plot is shown as continuous line with marker points
  - `u M:N` → plots data of column `N` over values of column `M`
  - `title "legend_string"` → data is labeled with `"legend_string"`

  To plot multiple files at once you can pass a list of the corresponding filenames separated by a `,` to the `plot` command, e.g.:
  `plot 'sim_results_1.hist' [options_1], 'sim_results_2.hist' [options_2]`

- **Tab completion**
  Gnuplot supports `tab`-completion, i.e. by pressing `tab` the current string will be completed so far as the name is uniquely determined. If you for example have only a file called `'sim_results.hist'` in your directory and you type `s` and `tab`, gnuplot will complete the whole string to `'sim_results.hist'`. If you have two files called `'sim_results_1.hist'` and `'sim_results_2.hist'` gnuplot will complete the string to `'sim_results_'`.

- **Changing the result scale**
  You can change the axis of your plot to logarithmic scale by typing `set logscale x` or `set logscale y` into gnuplot terminal. By `unset logscale x` and `unset logscale y` you can reverse this effect.

- **Labeling the plot**
  To label your plot you can use the `set` command followed by `xlabel`, `ylabel`, `title` and a string in `'` or `"`. Examples:

  - `set xlabel 'time t in s'`
  - `set ylabel 'f(t) in N'`
  - `set title "test function f(t) plotted over time t"`

  To disable the labels again use `unset` plus one of `xlabel`, `ylabel`, `title`.

- **Create screenshots / save your plot**
  To save your plot you can do one of the following:

  - Create a screenshot using the `Copy the plot to clipboard` button in the upcoming plot window. Then go to the graphic editor of your choice, paste the screenshot and save the image.
  - Export the plot directly to a file by using the following commands before plotting the results:
    ```
    set terminal jpeg
    set output "filename.jpg"
    ```