

# Using COMSOL to find the electric potential inside nanoparticles

Lischner nanoplasmonics group

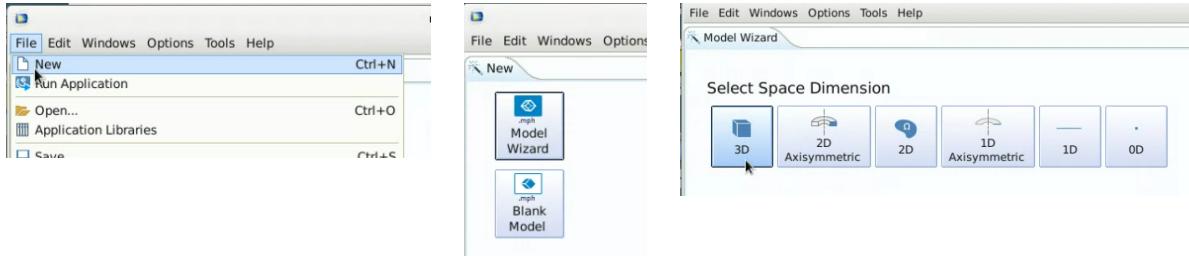
COMSOL Multiphysics is a general-purpose software for physics simulations. In this short tutorial, we will explain how to use it to find the electric potential inside nanoparticles.

## Getting started: physical interpretation

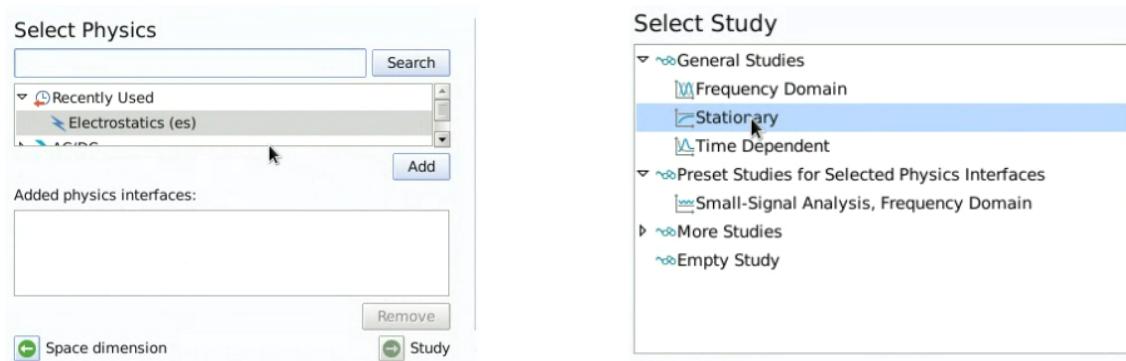
When a nanoparticle is subjected to an electric field, if it is made of a polarizable material, it will create an opposing electric field in reaction to the total electric field until an equilibrium has been reached. In general, the external electric field is not constant and so there is a frequency and a wavelength associated with the propagating electromagnetic wave. Finding the resulting electric field inside the nanoparticle would require solving Maxwell's equations in a complex time-dependent system. However, if the size of the nanoparticle is sufficiently small compared to the wavelength of the external electric field, then the external electric field can be considered as uniform inside the nanoparticle. Then, the quasistatic approximation can be employed, by considering the corresponding static electric potential and a frequency-dependent dielectric constant. At this point, the electric potential inside the nanoparticle can be calculated by solving Laplace's equation for the static potential instead of Maxwell's equations for the electric field. The external electric field is included implicitly by specifying the boundary conditions away from the nanoparticle. Since the electric field is expected to be uniform and pointing in the z direction away from the nanoparticle, the electric potential should be proportional to z away from the nanoparticle. The trick to enforce this is by constructing a cube (henceforth called boundary box) much larger than the nanoparticle in its center, with a specified potential equal to z along its faces. This completely determines the boundary conditions and ensures the potential is linear inside the boundary box but away from the nanoparticle, just as we wanted.

# Getting started: COMSOL

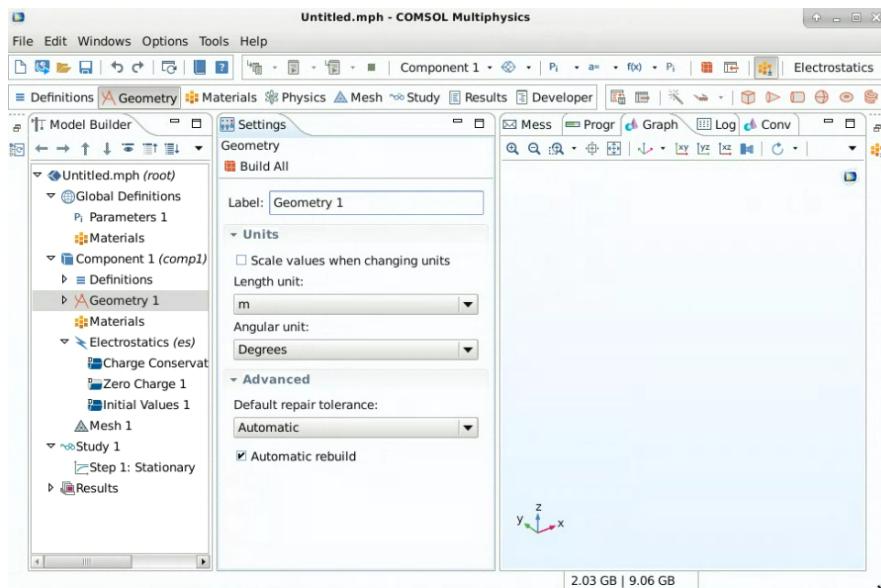
Since we want to solve Laplace's equation inside the nanoparticle under the quasistatic approximation, let's go ahead and open a fresh COMSOL session and select the correct physics simulator. Go to **File** -> **New** -> **Model Wizard** -> **3D**



Next, select the **Electrostatics (es)** physics simulator, press **Add** and then **Study**. Finally, under **General Studies**, double click **Stationary**



and you will be taken to the COMSOL interface to build the nanoparticle, which should look something like this



In the following sections, we will learn how to:

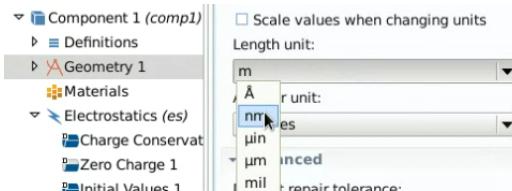
1. set up the nanoparticle and boundary box
2. set the boundary conditions
3. define the material and dielectric constant
4. build the mesh that will be used to solve Laplace's equation
5. solve Laplace's equation using the built-in algorithms
6. analyze the results
7. export the data into a file

## Setting up the shapes

In order to solve Laplace's equation we have to specify the units and the geometry of the objects we want to include. In this case, we want to work in units of nanometers, we want to work with a boundary box in the shape of a cube and a small nanoparticle with the shape of a square prism.

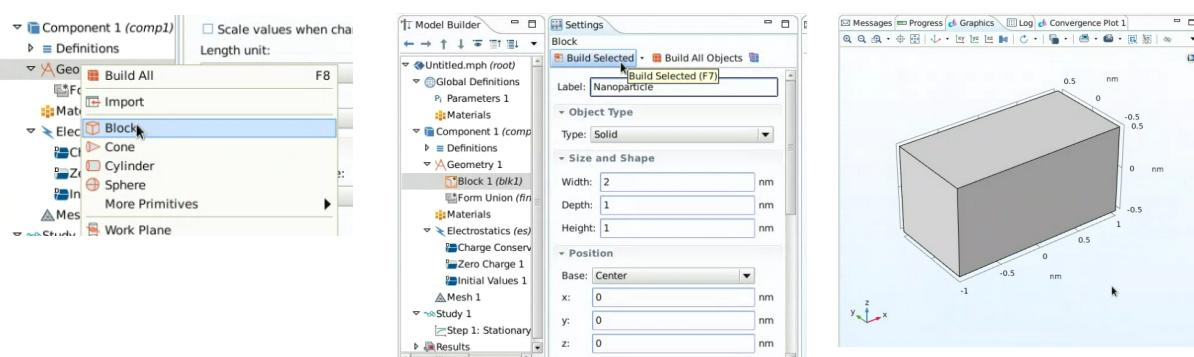
### Units:

Under **Geometry 1**, let's begin by specifying the length units to be nanometers



### Nanoparticle:

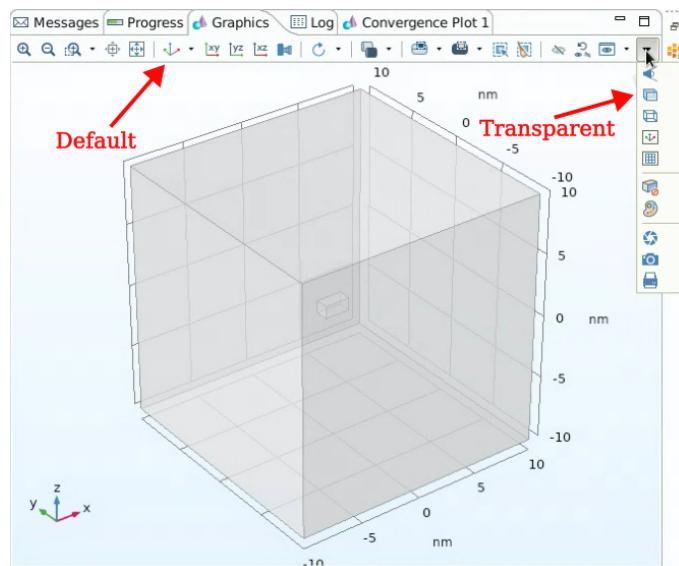
Right click on **Geometry 1** and select **Block**. This will create a “**Block 1 (blk1)**” entry with all the information about the block. Let's give it the name “**Nanoparticle**”, change its **Width** to 2 and change its **Base** to “**Center**”. Then, press “**Build Selected**”. This will produce a prism centered at (0,0,0) with **Width=2nm**, **Depth=1nm** and **Height=1nm**



The object can be visualized by rotating (left mouse button), zooming (middle mouse button) and panning (right mouse button)

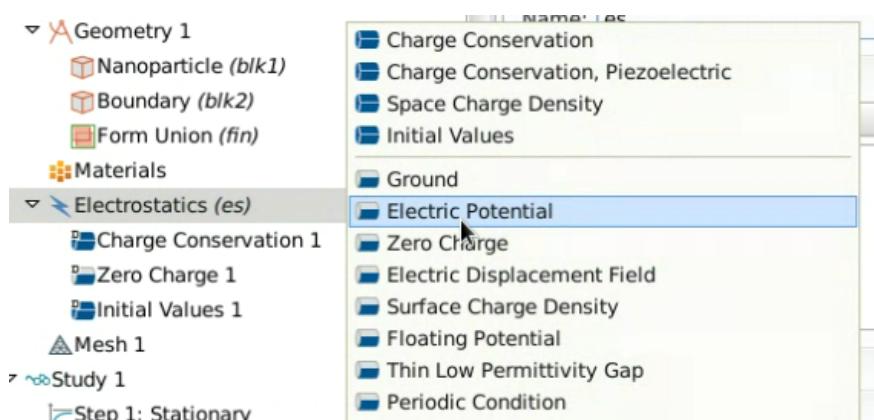
## Boundary box

Using the same procedure, create another block, also centered in **(0,0,0)** but with **Width=Depth=Height=20nm**, and let's call it “**Boundary**”. Press “**Build All Objects**”. Now you should have two objects generated: the large boundary box and the smaller nanoparticle inside it. In the “**Graphics**” tab, you can press “**Go to Default View**” in order to see the big picture. This will show the large boundary box but not the smaller nanoparticle which is hidden inside it. Using the “**Transparency**” view option solves this problem.



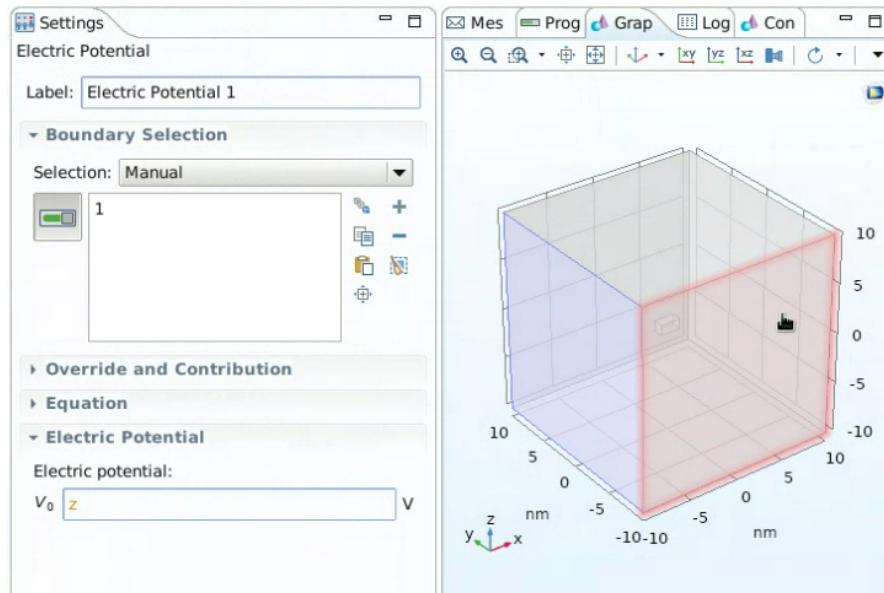
## Boundary conditions

The boundary conditions are specified by choosing the value of the electric potential along the faces of the large boundary box. Right click on **Electrostatics (es)** and select **Electric Potential**



Next, in the “**Electric Potential**” tab, under the “**Electric Potential**” menu, set it to **z** and click on the six faces of the larger cube to include them in the boundary. As they are

included, they will appear numbered in the large white selection box. Make sure all faces are included.



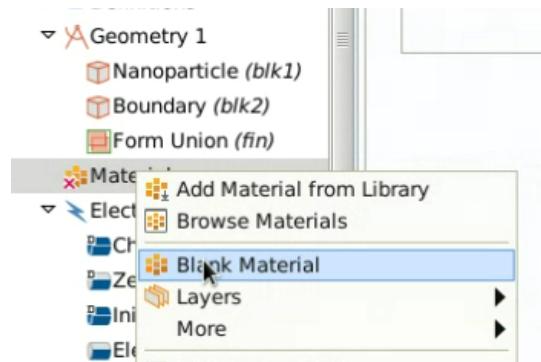
This will set the potential of the face at  $z=-10\text{nm}$  to be  $-10^{-9}\text{V}$ , the potential of the face at  $z=10\text{nm}$  to be  $10^{-9}\text{V}$ . The other four faces will have a position-dependent potential which goes from  $-10^{-9}$  to  $10^{-9}$  as a function of  $z$ .

## Material and permittivity

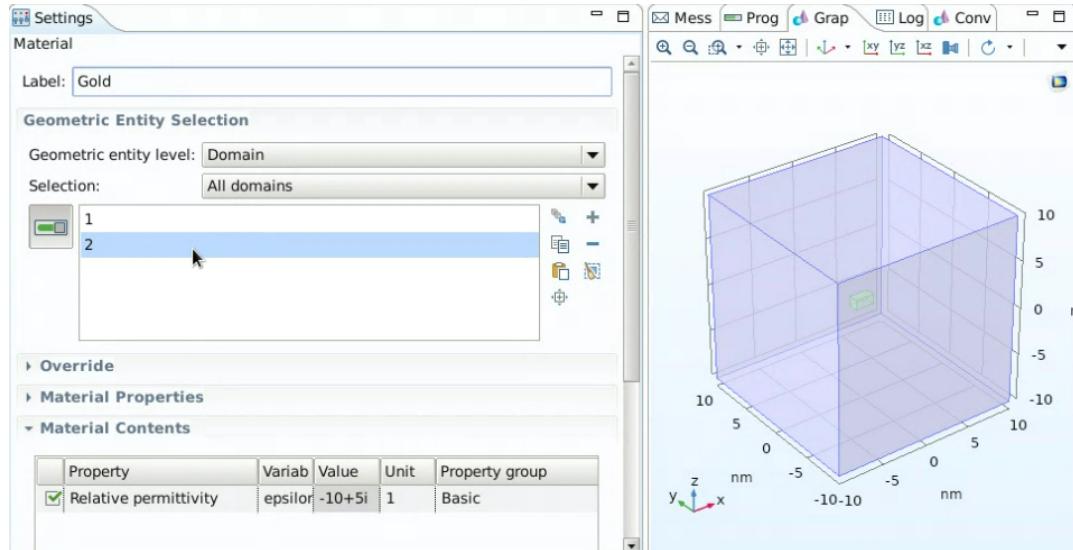
The only property of the materials that we care about in this context is their electric permittivity. This is the only property that is required to solve Laplace's equation in this setup.

## Nanoparticle

In the **Model Builder** menu, right-click on **Materials** and add a **Blank Material**.



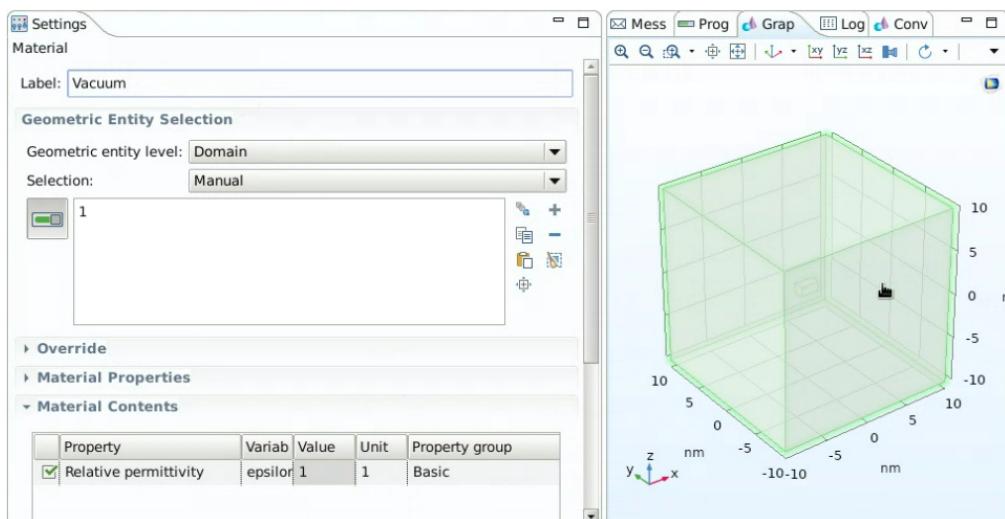
This will open the Material's settings, where you can specify the relative permittivity and which object gets what material. This material is assigned by default to both object 1 and 2. Let's begin by considering only the nanoparticle. Under "Geometric Entity Selection", by clicking on the numbers under "**Selection**", you can see which object has what number.



In this case, object 2 is the nanoparticle and object 1 is the boundary box. Remove the boundary box from the list, since we only want to assign this material to the nanoparticle. Under **Material Contents**, set the **value** of the **Relative permittivity** to  $-10+5i$  (remember that this is a complex value).

## Boundary box

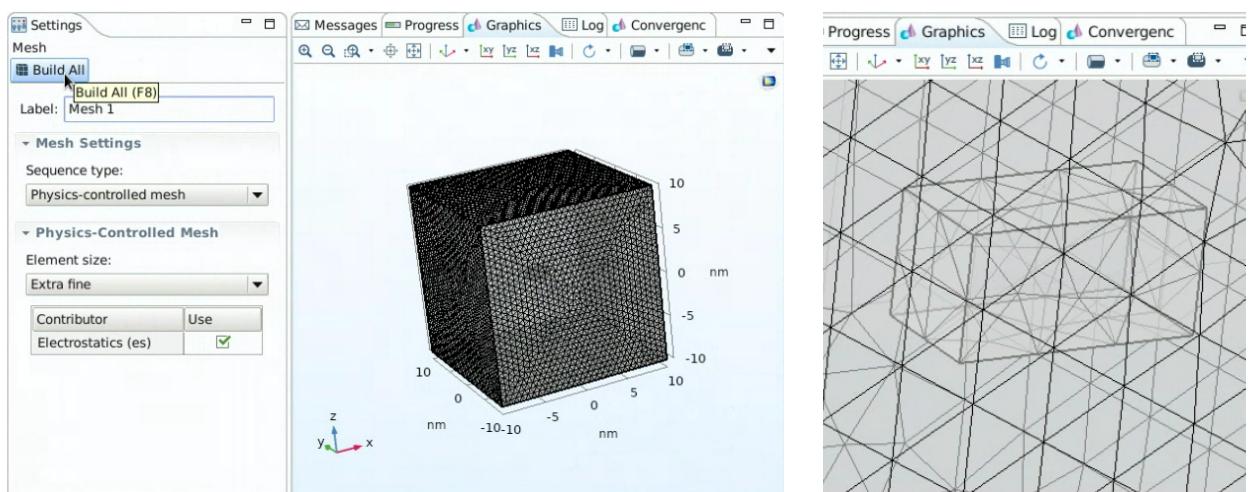
Finally, we want to specify the relative permittivity of the material inside the boundary box, that is the material surrounding the nanoparticle. This is going to be vacuum, which has relative permittivity of 1. Repeat the previous process by selecting the boundary box and setting the **value** of the **Relative permittivity** to 1.



# Meshing

Laplace's equation is going to be solved numerically, so we need to specify a mesh of points where this equation is to be solved.

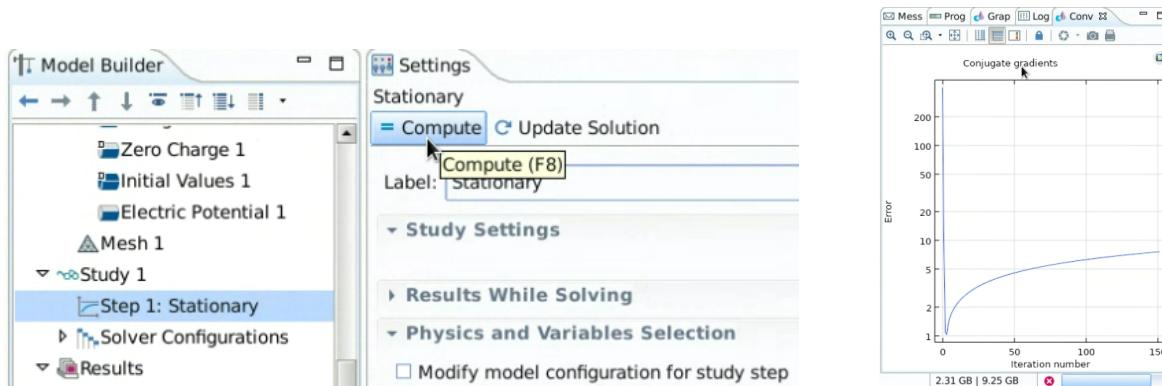
In the Model Builder, select the “**Mesh 1**” menu. This will bring up some generic options to build a mesh for the whole system. Under “**Element size**”, select “**Extra fine**”. This controls how many points will be considered for the numerical solution of Laplace's equation. Press “**Build All**” to produce the mesh. It is important to make sure that the nanoparticle has a sufficient number of points, since most of the large variations in potential will occur around it. You can zoom into the nanoparticle to check this. For now, let's keep it like this, but keep this point in mind in case the resulting potential doesn't make sense.



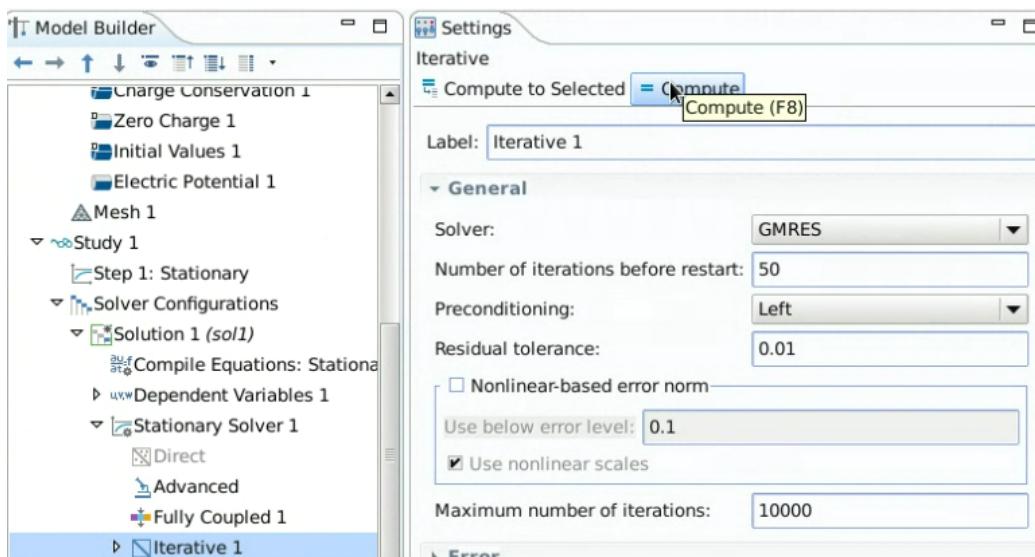
# Solving Laplace's equation

Laplace's equation is solved with built-in algorithms implemented in COMSOL, which use the previously specified mesh of points.

In the “**Model Builder**” menu, go to “**Study 1**” and choose “**Step 1: Stationary**”. This will open up the menu with basic options for solving Laplace's equation. Press “**Compute**” to use the default options. You will notice that the calculation might take a while. In fact, if you go to the “Conv” tab on the right side of the screen, you can see how the algorithm is performing by looking at the error as a function of the iteration number. Clearly, in this case, it is not converging and we need a better solution.



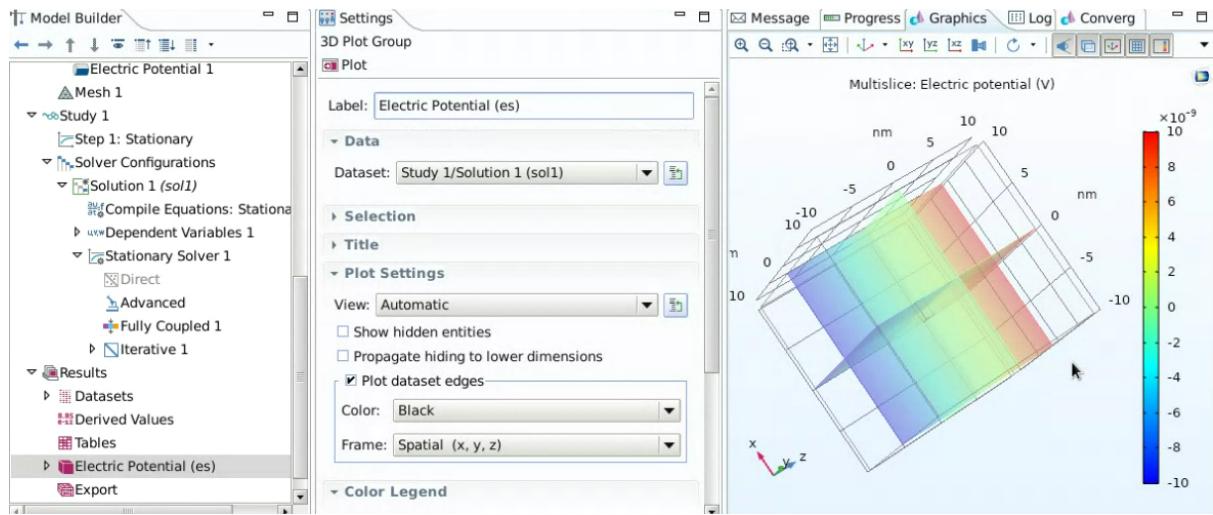
Fortunately, COMSOL provides a plethora of options to solve this problem. End the simulation by pressing the red circle in the bottom right corner of the screen. Go to “**Solver Configurations**” -> “**Solution 1**” -> “**Stationary Solver 1**” -> “**Iterative 1**” to find more advanced options. Under “**General**”, you can change the “**Solver**” from “**Conjugate gradients**” (which is the default) to “**GMRES**” (Generalized minimal residual method). Now select “**Compute**” and the solver should converge in a couple of seconds!



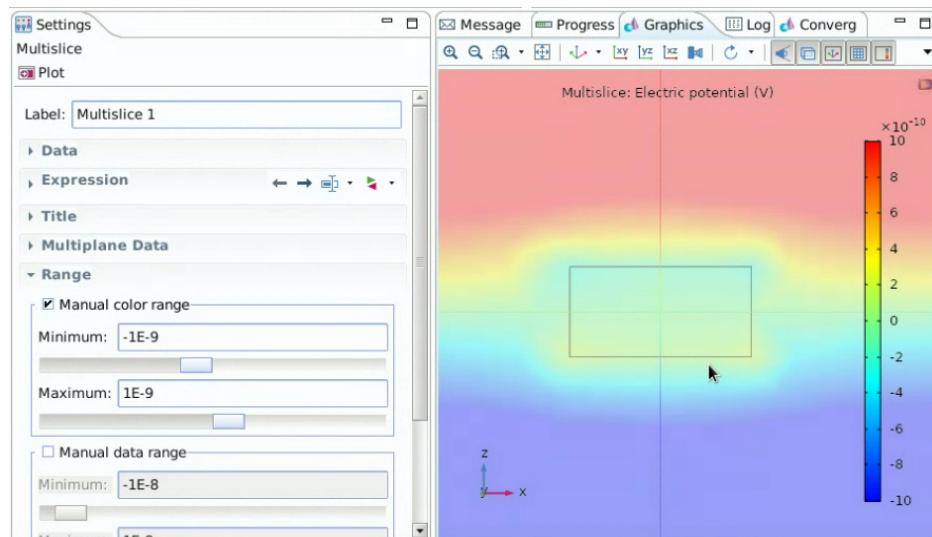
# Analyzing the results

Once the solver has converged, we can now look at the solution it provides: the (complex) electric potential as a function of position inside the whole boundary box, including the nanoparticle.

By default, after converging, COMSOL will immediately take you to the “Results” menu with a multislice visualization of the electric potential



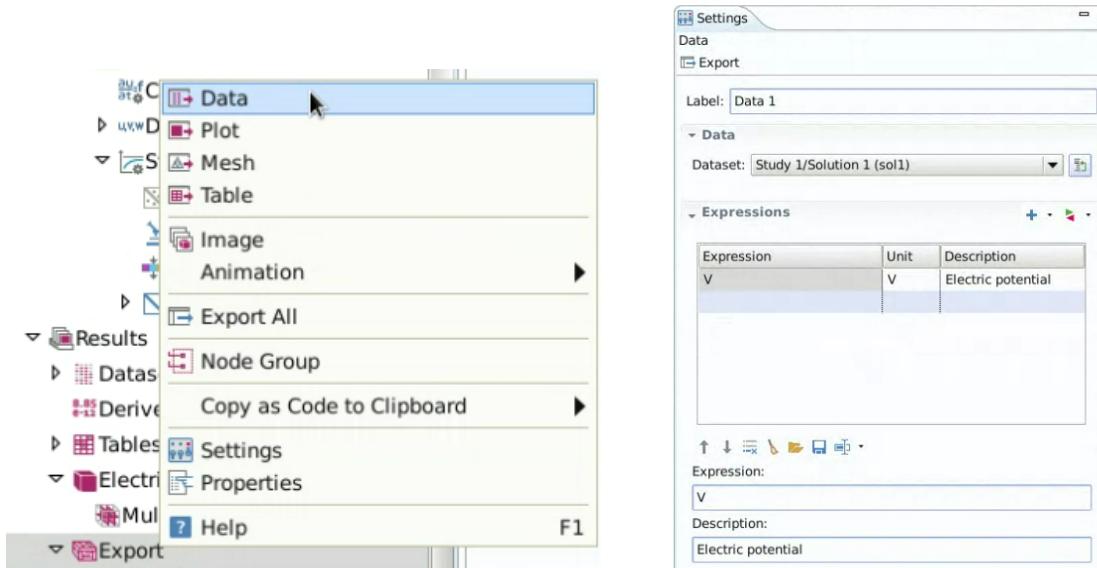
The colorbar ranges from  $-10^{-9}$  (dark blue) to  $10^{-9}$  (bright red). As expected from our choice of boundary conditions, the boundary at  $z=10\text{nm}$  is red and the boundary at  $z=-10\text{nm}$  is blue. Zooming in to the nanoparticle position, we can see how the electric potential is changed due to its presence. This is not easy to see with the default color scale, so we can manually change the color scale to go from  $-10^{-9}$  to  $10^{-9}$  to make it more evident.



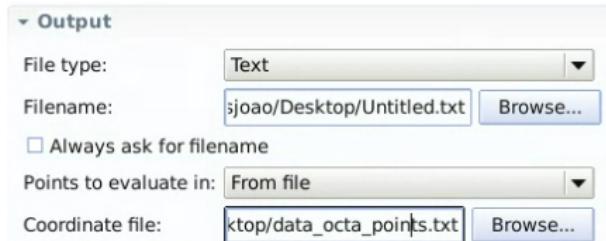
Now it becomes clear that the polarization is so strong that the potential decreases as  $z$  increases inside the nanoparticle!

# Exporting the data

In the “**Model Builder**” tab, inside the “**Results**” menu, right click “**Export**” and select “**Data**”. This will open the export settings which can be used to save the potential into a file. For that purpose, we need to specify which quantity is to be exported. In the “**Expressions**” section, write “**V**” under “**Expression**” and COMSOL will autocomplete the units and description.



Ultimately, we want to find the value of the potential at the atomic positions belonging to the nanoparticle, so we need to provide COMSOL with a list of these positions. Under the “Output” section, in the line “**Points to evaluate in:**”, change to “**From file**” and add the file with the atomic positions in “**Coordinate file**”. You will also need to add the filename to which the final result will be written to, in the line “**Filename**”.



Finally, press “Export” (on the top) and COMSOL will export the value of the potential at each of the specified points. The files should be formatted as such:

```
Open positions... Save Open Untitled.txt
---Desktop --- Save
% Model: octa2.mph % Model: cubecube.mph
% Version: COMSOL 5.5.0.359 % Version: COMSOL 5.5.0.359
% Date: Sep 29 2022, 16:37 % Date: Dec 1 2022, 11:49
% Dimension: 3 % Dimension: 3
% Nodes: 3134 % Nodes: 132651
% Expressions: 1 % Expressions: 1
% Description: Electric potential % Description: Electric potential
% Length unit: nm % Length unit: m
% x y z % x y z V (V)
-2.000 -2.000 -2.000 -2 -2 -1.964124887408478+0.0092706278073073651
-2.000 -2.000 -1.920 -2 -2 -1.88622775122696822+0.00931433981491891
-2.000 -2.000 -1.840 -2 -2 -1.86622005676418+0.00941508180435451581
-2.000 -2.000 -1.760 -2 -2 -1.72624769537170364+0.0094511916881348711
-2.000 -2.000 -1.680 -2 -2 -1.6463544007818691+0.00947345941961041
-2.000 -2.000 -1.600 -2 -2 -1.5665421289981378+0.009477307724123656
-2.000 -2.000 -1.520 -2 -2 -1.4868330001268045+0.009486366304991
-2.000 -2.000 -1.440 -2 -2 -1.4072725532384559+0.00949638844591
-2.000 -2.000 -1.360 -2 -2 -1.3227053245115+0.0095029957911615951
-2.000 -2.000 -1.280 -2 -2 -1.248425811239704+0.00919862643311871
-2.000 -2.000 -1.200 -2 -2 -1.1692432053030603+0.00909202621004416841
-2.000 -2.000 -1.120 -2 -2 -1.088019741904969354+0.008889953432108664
-2.000 -2.000 -1.040 -2 -2 -1.011329595836664+0.00854348107837733061
-2.000 -2.000 -0.960 -2 -2 -0.9329683523378478+0.008200395187046601
-2.000 -2.000 -0.880 -2 -2 -0.8541321640803338+0.007840646738194321
-2.000 -2.000 -0.800 -2 -2 -0.7755696185124529+0.0074409987823230191
-2.000 -2.000 -0.720 -2 -2 -0.6974739366606348+0.006668775174953099
-2.000 -2.000 -0.640 -2 -2 -0.619478224846981+0.006420639335016249
```

