Manual for WCT EL-IMG Package

Windows Version

Wave Computation Technologies, Inc.

March, 2017

Content

- Introduction
- Imaging Simulation Requirements
  - Transmitters & receivers
  - Measured signal
  - Material property in an imaging simulation
- Simulation Schemes
- Special notes on receiver
  - Receiver naming method
  - Field on receiver
- Case Setup in WCT Imaging Package
- Data file format

Content

Cont.

- Demo-Cases
  - I: 2D Imaging, detecting two objects with f_{max}=27 Hz, known background, point dipole source, Scheme II.
  - II: 2D Imaging, imaging a black box with limited known information, point monopole source, Scheme II.
    - Extension part of Case II: obtaining the signal on sensor
- Tutorial/demo package

Introduction

- Wavenology EL-IMG Package uses the Reversed Time Migration method to image a specified 3D region in a 3D simulation space.

- Wavenology EL-IMG imaging package can produce an image with three kinds of transmitters/receivers scan schemes, as shown in the following pages
  - Single simulation --- (support multiple source excitation at one time)
  - Separated transmitters array and receivers array --- (single source excitation at one time)
  - Switching transmitters and receivers array --- (single source excitation at one time)

- All three imaging schemes include at least two steps
  - A forward simulation
  - A backward simulation
  - For scan scheme II & III, the whole imaging procedure will be
    - iterating each source by
      - A forward simulation
      - A backward simulation

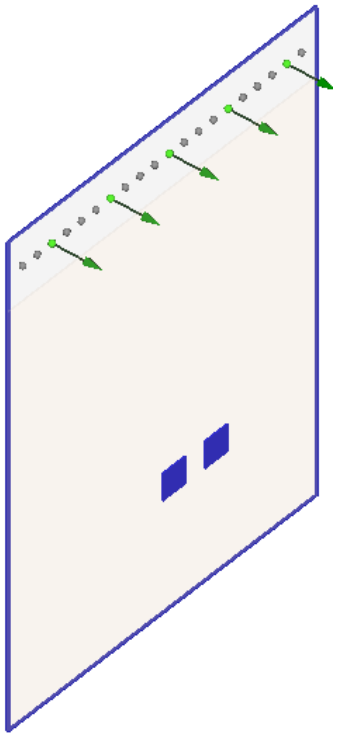
Requirements

- Transmitters
  - Point monopole
  - Point dipole
  - Moment tensor
- Receivers (Sensor)
  - Ideal point Receiver to receiving V
- Measured signal on receivers
  - transient V
- Material property in an imaging simulation
  - the material should be lossless, or the loss will be close to zero

Simulation Schemes

Scheme I: Single simulation

In this scheme, multiple sources can be simultaneously excited with individual pulse.

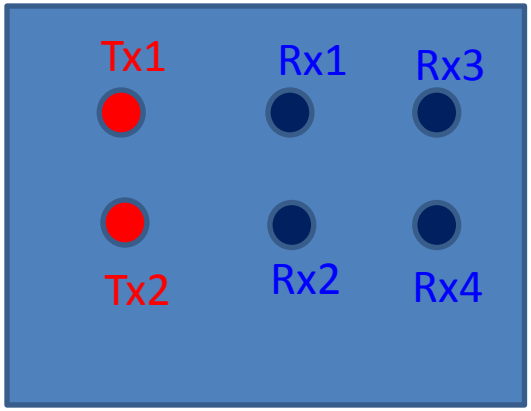


Here, there are 5 sources that will be simultaneously excited in one simulation

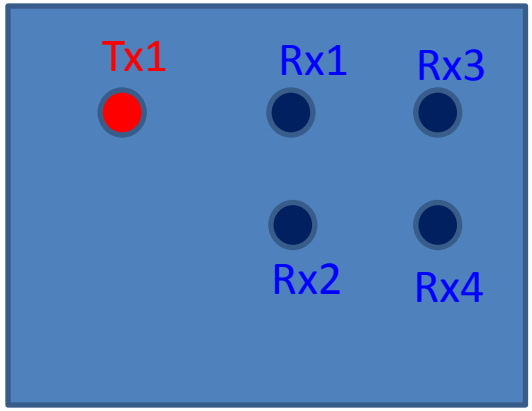
Scheme II: Separated transmitters array and receivers array

In this scheme,

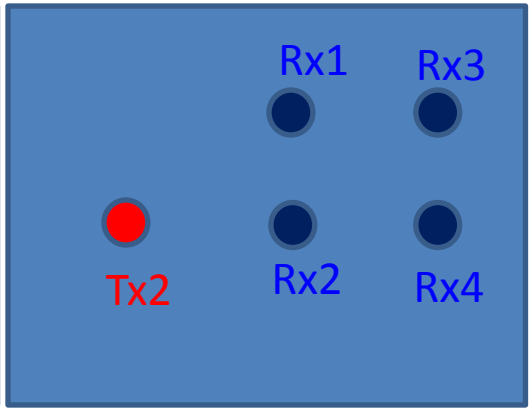
- user can define multiple sources, if we say it is N.
- user can define multiple receivers
- sources array is separated from receiver array
- the simulation will include N runs. Each will excite one source only, but the receiver array keeps the same in each run. Each source can use individual pulse.



T/R array in definition



Run1

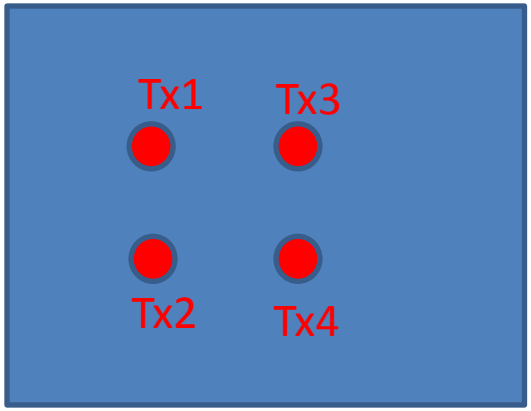


Run2

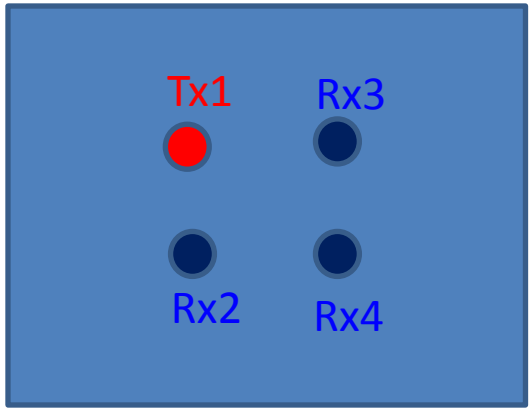
Scheme III: Switching transmitters/receivers array

In this scheme,

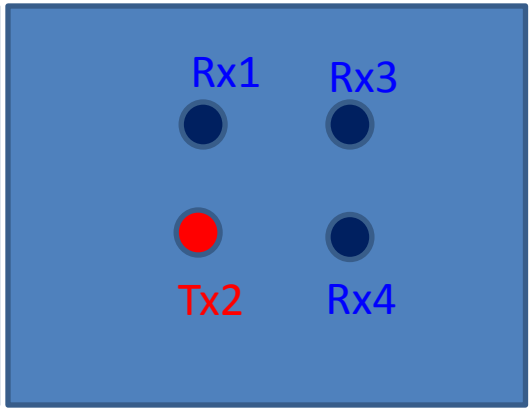
- user can define multiple receivers (or multiple sources), if we say it is N. (note: must be receivers only or source only, can not mix)
- the simulation will include N runs.
  - If define as receiver only, each run will convert one receiver to source and excite it only. Each source will use the same pulse, which is defined as the WCT project pulse.
  - If define as source only, each run will excite one source, other sources will be converted as receiver. Each source can use individual pulse.



T/R array in definition

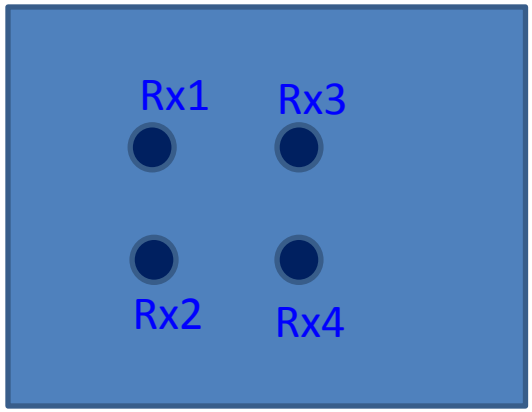


Run1

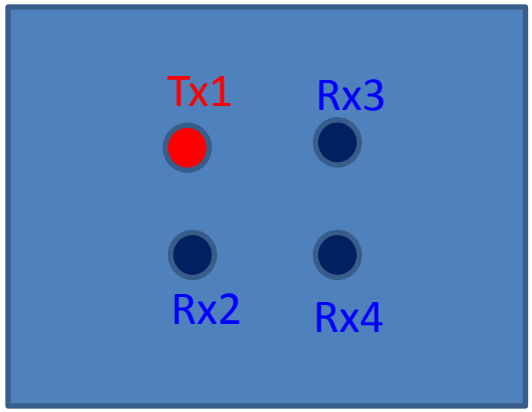


Run2 ……..

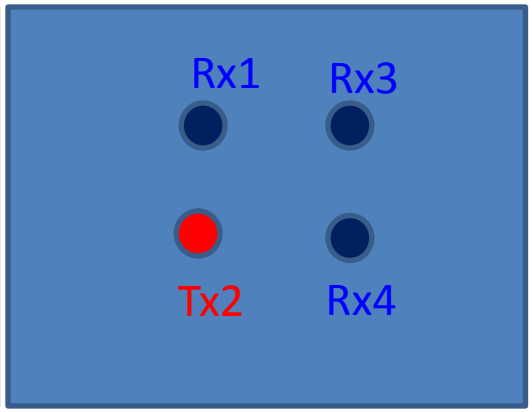
Or



T/R array in definition



Run1



Run2 ……..

Receiver Naming System

- In defining receiver's name in WCT imaging package, please make sure all receivers' name is following the ACSII sequence as following examples,
  - if the receiver number is < 10, user can define as: obv1, obv2, … obv9
  - if the receiver number is in the range 10-99, user can define as: obv01, obv02, … obv09, obv10, obv11, … obv99

We recommend this is due to WCT I/O the trace data file and mapping to receiver with a ASCII sequence. For a 3-obv system as obv1, obv2, obv10, not matter how to define the sequence of these 3 obv. In WCT GUI, the ASCII sequence is always: obv1, obv10, obv2. This will mess the trace sequence after loading and cause imaging problem.

Recorded Field on Receiver

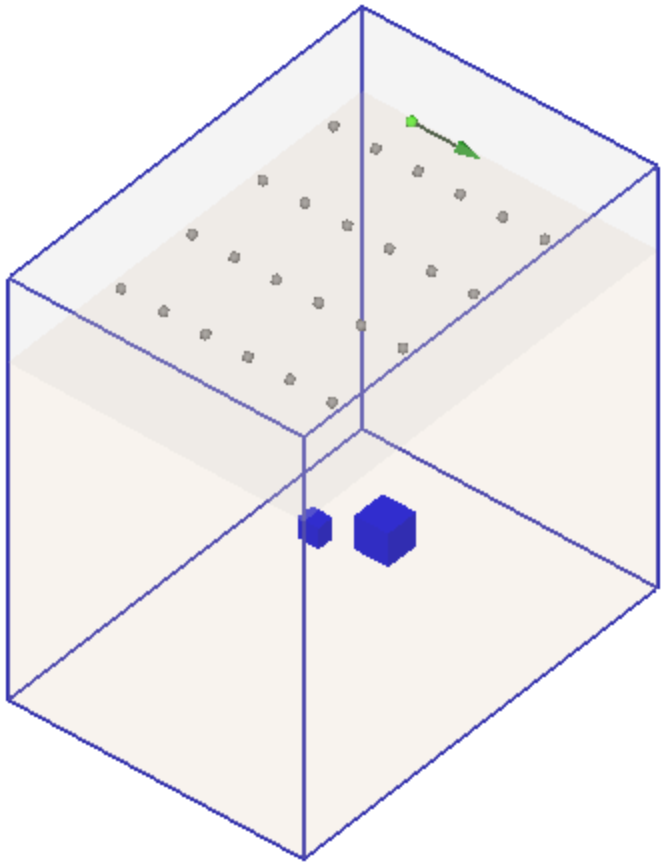
- In defining the receiver capturing field in imaging, it should be single component only, for example, Ex, or Ey, or Ez, or Hx, or Hy, or Hz only. Please do not combine two or more components.

- The reason is that, the current imaging code will convert receiver to source with a polarization. With more than one components, there is challenge in defining the source.

- If user want to use more than one components to imaging, the workaround is defining multiple observers at the same position, which captures single component only.

Setup an Imaging Simulation

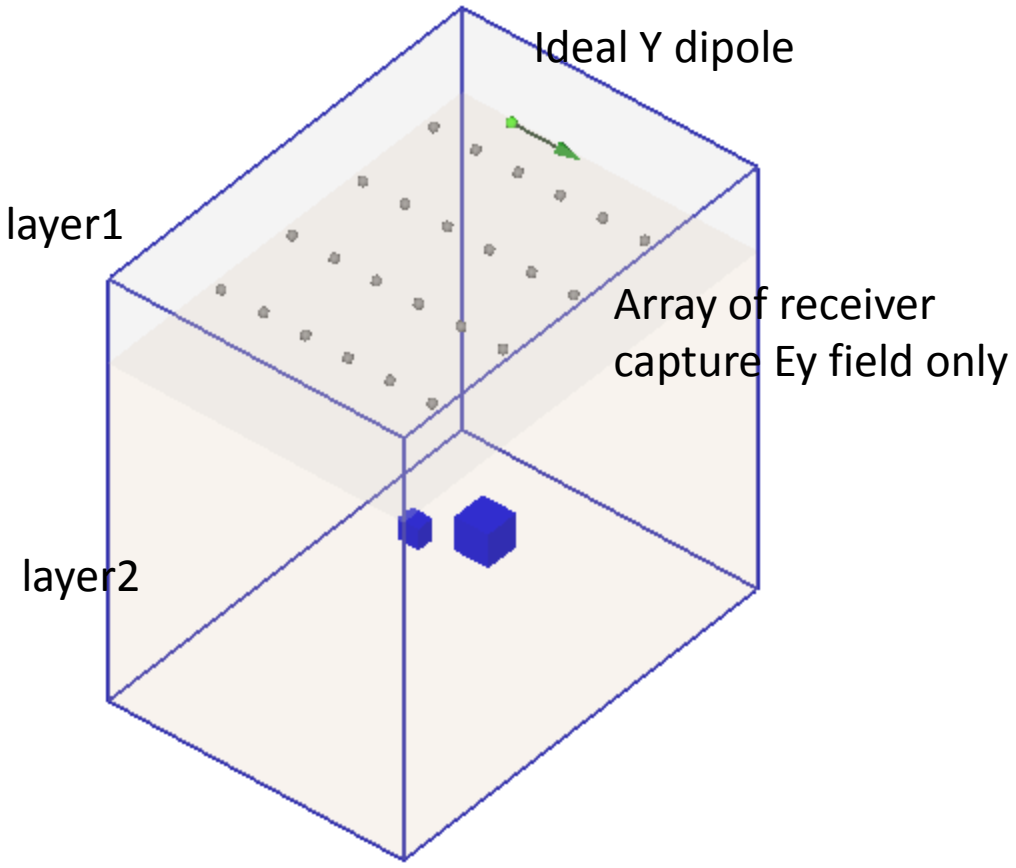
1. Define a WCT EL simulation case
   - With some kind of source
   - Array of receivers



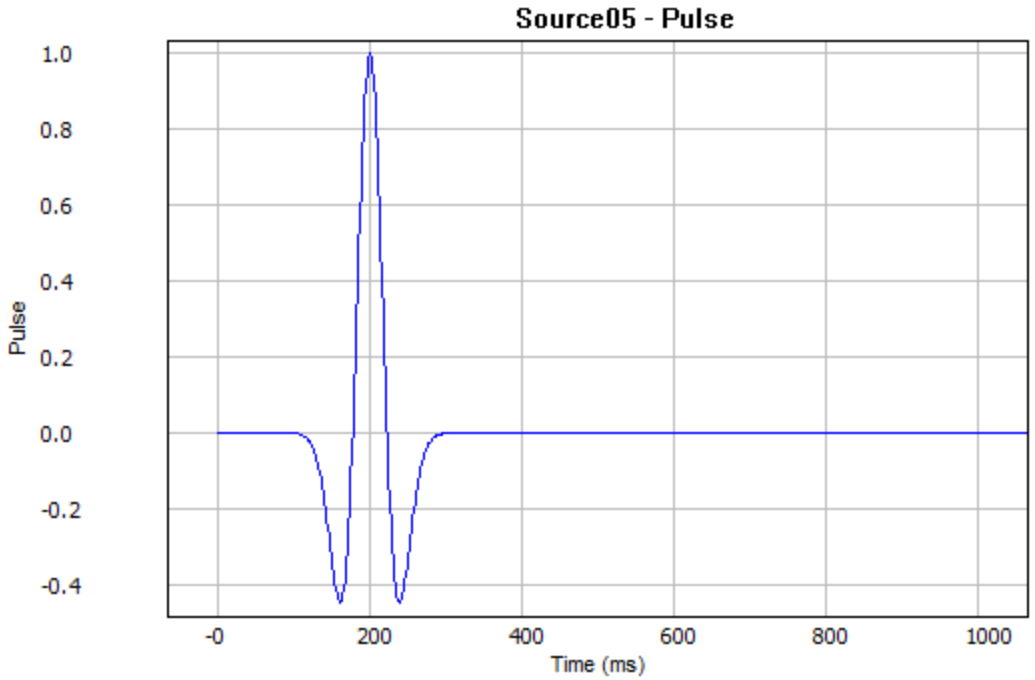
The right figure is a demo of a GPR case.

It has 2 layers media, one ideal point Y dipole source, and an array of receiver to capture Vy field only.

There is two objects in the bottom layer.

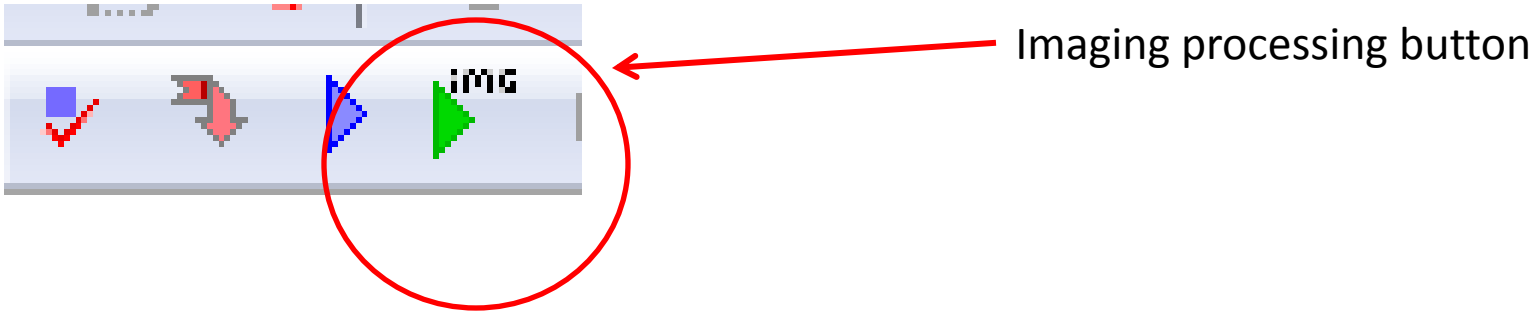


The source pulse in this case is a Ricker wave with fmax = 27 Hz, as shown in the right figure.



Imaging Processing

- After the case setup is finished, user can use Wavenology EM-IMG solver to image any region in the computational domain.



Imaging Processing Setup

Imaging scheme

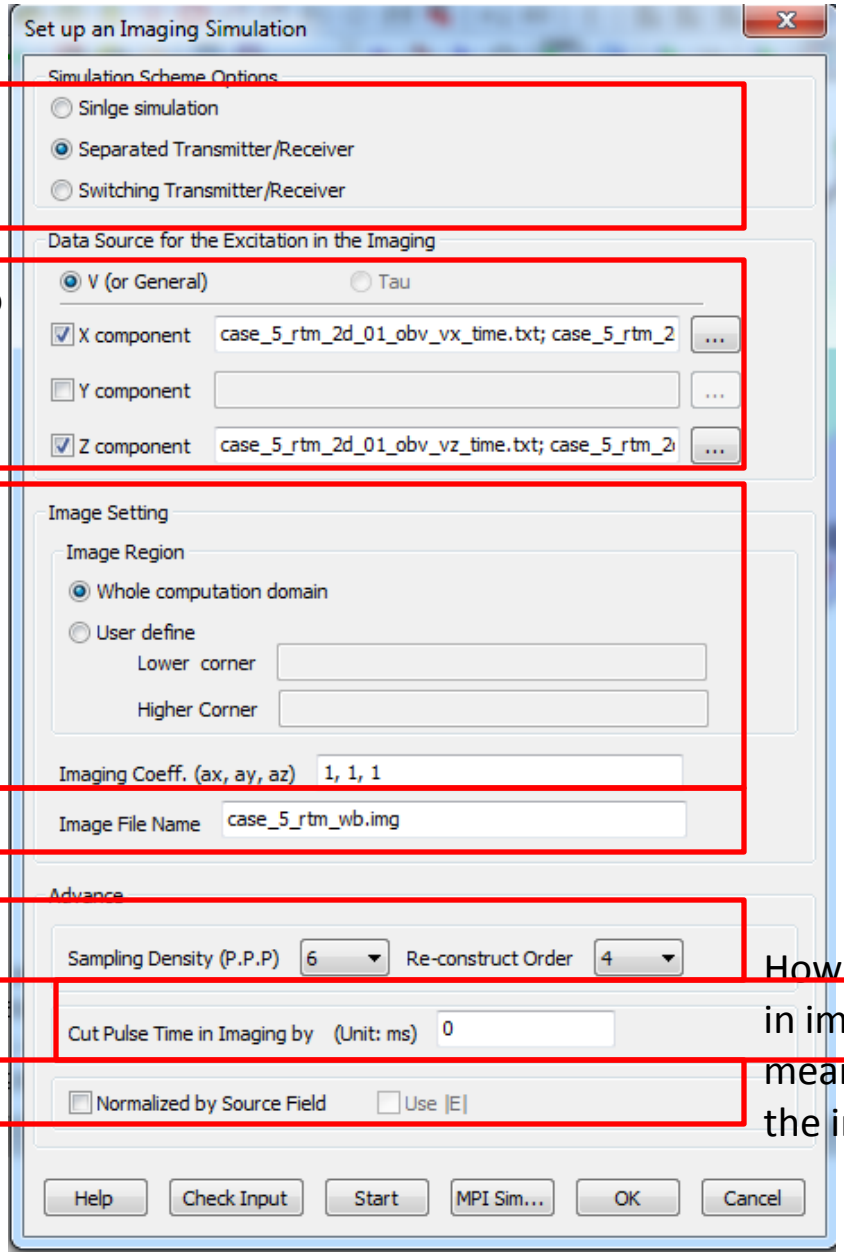
Signal on the receiver in the backward processing. If user want to use the signal outside the WCT EL package, it should follow the format as shown in the following slides.

Imaging region & the weights of V components used in imaging. a_i is for V_i component (i=x, y, z)

Image file name. The format will be provided in the following slides.

Advanced parameters in control imaging. Please do not change it.

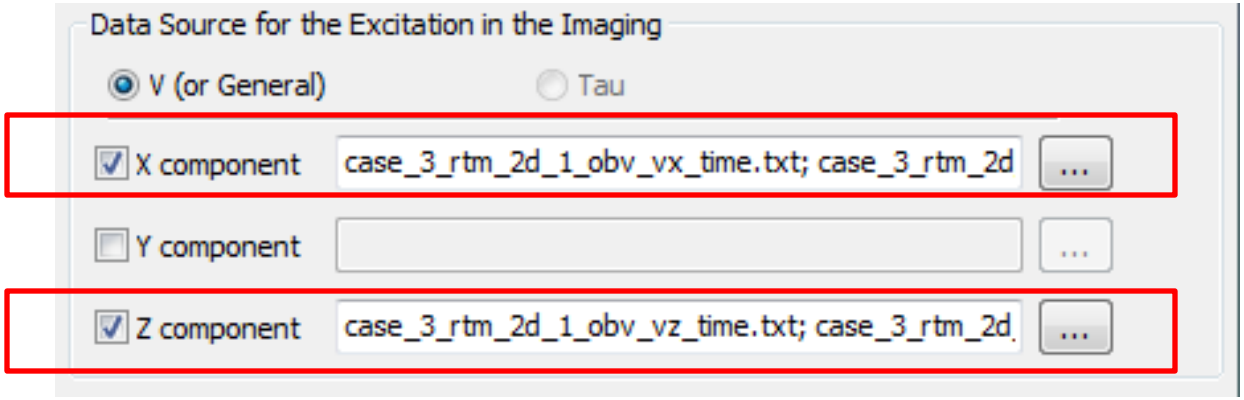
Whether need to normalize the final image, we recommend to **enable** this option



How to use the input signal in imaging. Default is 0, means that do not process the input signal.

Note on the measured data

For ideal point sensor with V field signal, user need to specify which component will be used for imaging, as following figure, we set the measured data as X & Z component

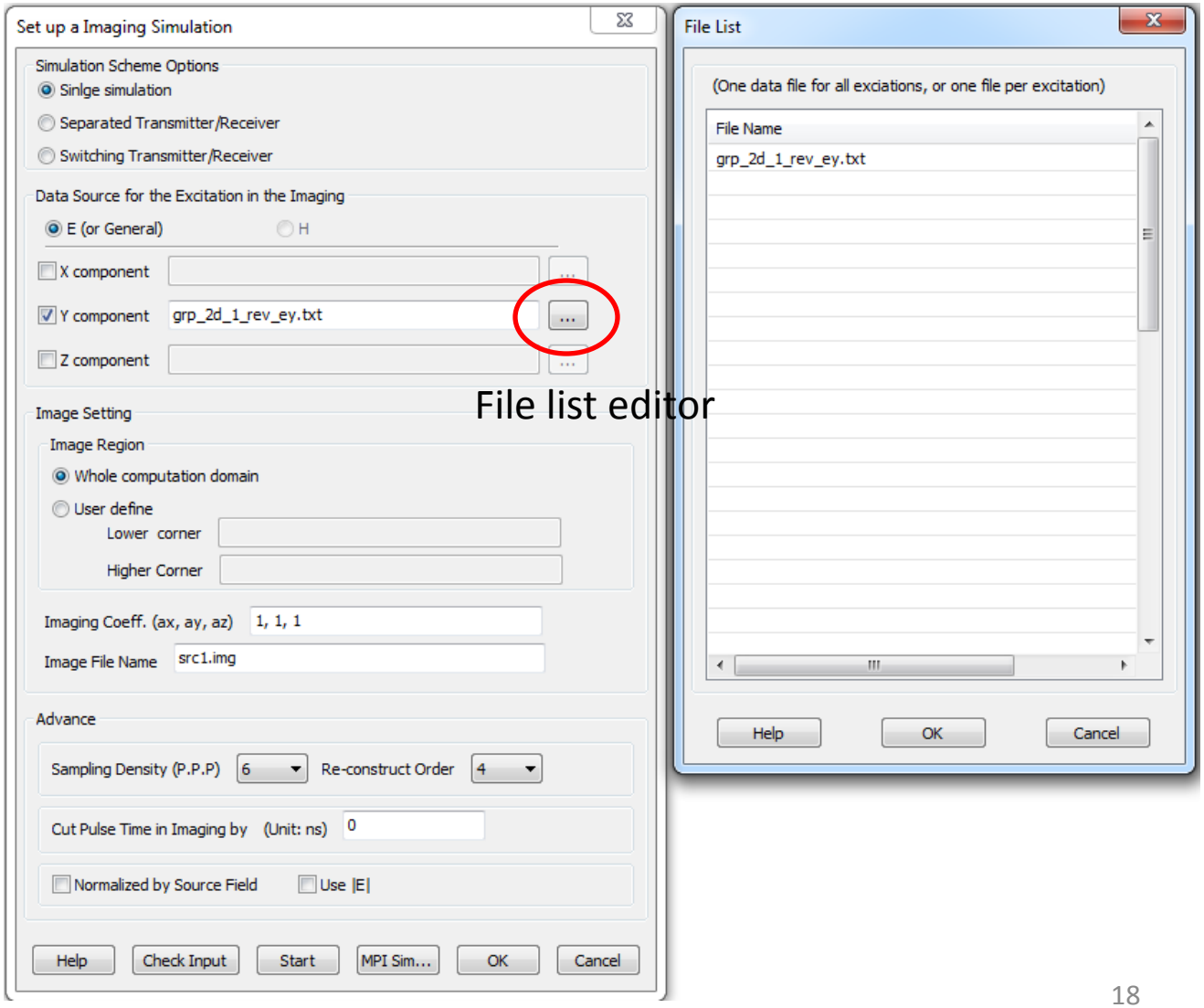


Imaging Processing Setup

Cont.

Signal files for backward simulation

In general, for scheme II & II, there are multiple runs in a simulation, and the signal for each source is stored in one file only. So, it requires multiple data files.

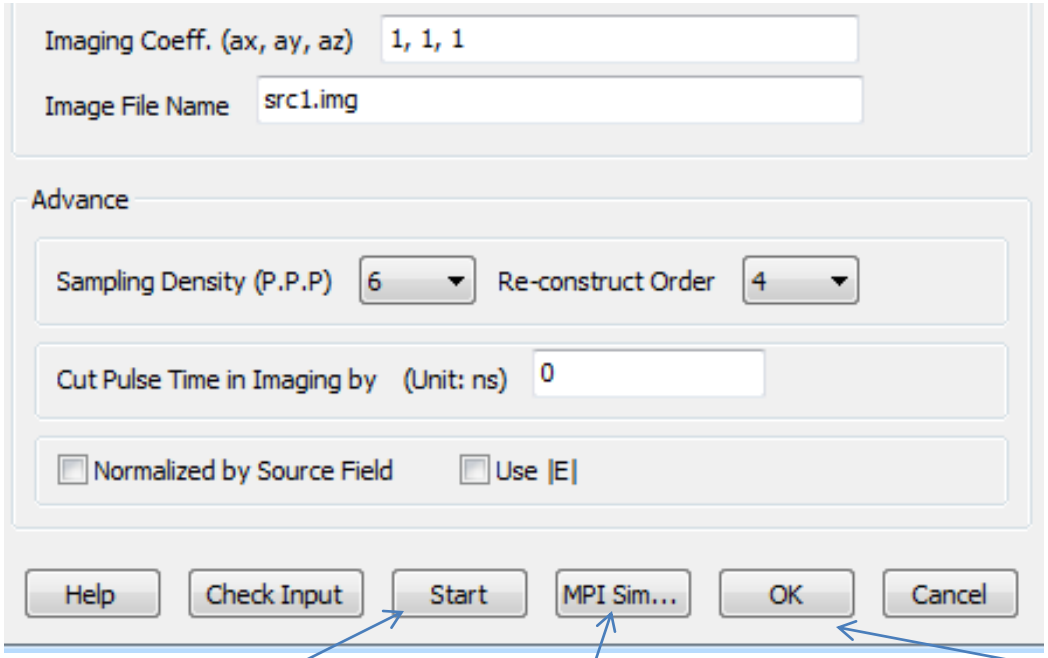


File list editor

Imaging Processing Setup

Cont.

After the imaging setup is finish, user have several action options



Start a "Forward+backward" processing to generate image.

This setup will be stored also for the future usage.

Generate a project file for this "Forward+backward" processing by WCT EM MPI solver.

This setup will be stored also for the future usage.

Save this setup for future usage. But do not make processing right now.

Log

Simulation is completed normally at 01/06/16 17:55:05. Total simulation time is 2 seconds (2 sec)

.....

Run 3 RTM backward simulation is completed ..... Start next run RTM forward simulation.....

.....

Validating the design

Body positons, layer positons, observer positons, user defined control points, source combination are verified.

Simulation has been started at 01/06/16 17:55:05 by Wavenology EM 1.9.0 (x64)

Preprocessing...

.....Begin to generate snapshot mesh...

.....end of snapshot mesh generation...

Domains: 1 x 1 x 1, Cells: 100 x 3 x 76, Delta time: 2e-011 sec, Mesher version: 1, CPU Time: 0.076 sec, Explicit solver is used. Single thread is used.

Time Stepping...

Postprocessing... Last time window: 20 ns, Number of time step: 1000, Time for time-stepping: 0.832 sec, Postprocessing: 0.011 sec

Simulation is completed normally at 01/06/16 17:55:06. Total simulation time is 1 second (1 sec)

.....

Run 4 RTM forward simulation is completed ..... Start RTM backward simulation.....

.....

Validating the design

Body positons, layer positons, observer positons, user defined control points, source combination are verified.

Simulation has been started at 01/06/16 17:55:06 by Wavenology EM 1.9.0 (x64)

Preprocessing...

.....Begin to generate snapshot mesh...

.....end of snapshot mesh generation...

Domains: 1 x 1 x 1, Cells: 100 x 3 x 76, Delta time: 2e-011 sec, Mesher version: 1, CPU Time: 0.075 sec, Explicit solver is used. Single thread is used.

Time Stepping...

Postprocessing... Last time window: 20 ns, Number of time step: 1000, Time for time-stepping: 1.875 sec, Postprocessing: 0.012 sec

~~Simulation is completed normally at 01/06/16 17:55:08. Total simulation time is 2 seconds (2 sec)~~

.....

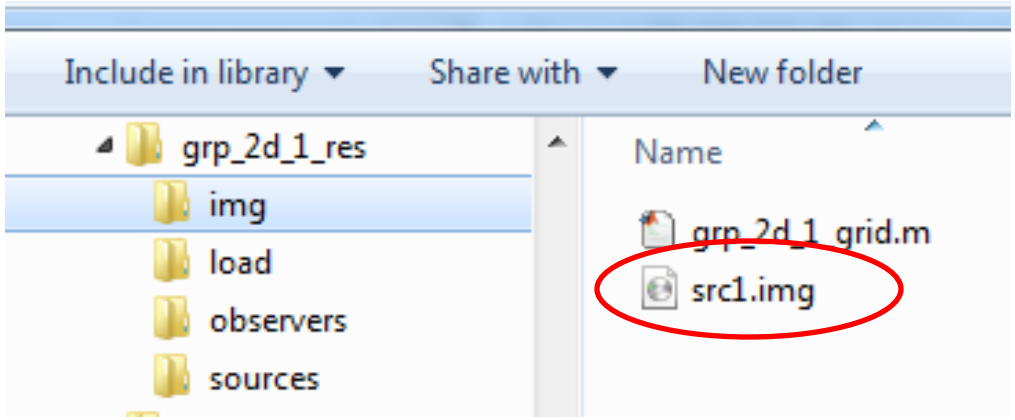
Run 4 RTM backward simulation is completed .....Whole RTM simulation is totally completed & temporary results are removed.

.....

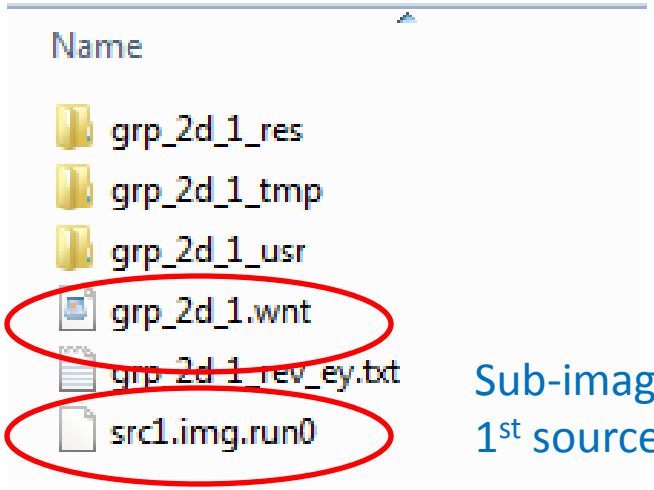
.....

The simulation log will report the status of each imaging run.

If user click “Start” button to generate image, if there is not error report in the processing and the Imaging can be finished successfully, a target file “src1.img” (as it is defined in the setup dialog) will be created in the image result folder: ***xxxx_res/img/*** , as shown in the following figure,



Meanwhile, the sub-image for each source will be placed in the project root folder as following:



Project file

Sub-image for the 1st source

The Signal File Format in the WCT EL-IMG Package

Signal on the receiver for the backward simulation

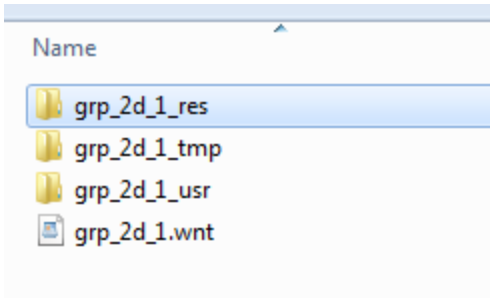
- ASCII TEXT file
- if user want to use the data directly from a WCT forward simulation, please use the data file: "forward_project_folder/projectname_res/observers/projectname_rev_componentname.txt"
  - it is better to copy this file to the root folder of the imaging project and set this file as the signal source in the imaging simulation

Line number	meaning
1-3	comment
4	number of frames in the file
5	comment
6	frame start time, unit: s
7	comment
8	frame end time, unit: s
9	comment
10	Frame time step, unit: s
11	comment
12	Length of each frame
13 : n0	Frame 1
n0+1 : n2	Frame 2

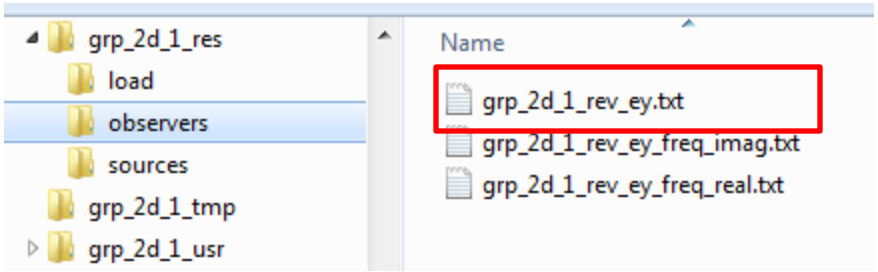
Example File

```
%Wave Computation Technologies simulation waveform data, version 1.0 ::
%Time (ns)
%frames number
31
%frame start
0
%frame end
1.98e-008
%frame step
3.6e-010
%frame length
56
0.0000000e+000
0.0000000e+000
0.0000000e+000
0.0000000e+000
-6.5508699e-033
-3.2401052e-027
-1.5777409e-023
3.9143431e-020
5.3799126e-017
1.5878495e-014
1.8202063e-012
1.0259642e-010
3.2743763e-009
6.5165523e-008
8.6780892e-007
8.1612652e-006
5.6583969e-005
2.9980612e-004
1.2526923e-003
4.2461860e-003
.....
```

Example File Folder for a simulation to obtain the signal on the receiver for the backward simulation



Project root folder for project: grp+2d_1



Sub-folder for project and the data file will be used for the backward simulation

The Image File Format in the WCT EL-IMG Package

Binary file

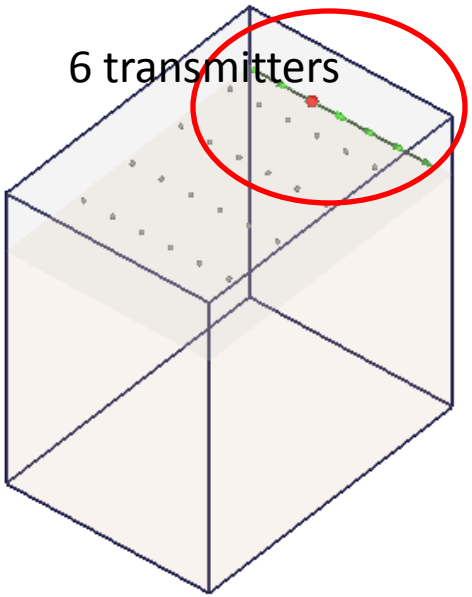
Meaning	Data type	Length (Bytes)	Comment
header	char	128	
version	int	4	sizeof(int), the value is: ***2***
array 3D start index (cell)	int	4 (int)* 3	x0, y0, z0
array 3D end index (cell)	int	4 (int)* 3	x1, y1, z1
array size	int	4 (int)* 3	x, y, z
array content	float	4*(nX*nY*nZ)	nK=k1-k0+1, (k=x,y,z) sequence as: inner(Z)->middle(Y)->outer(X)

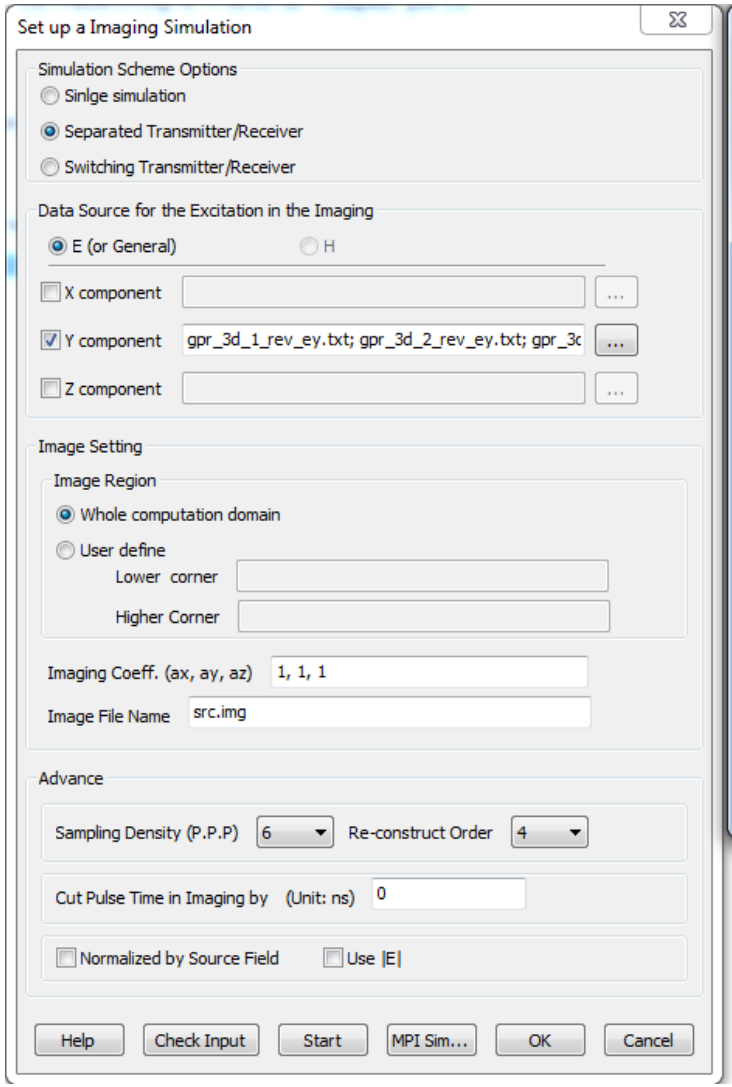
Attached is a Matlab code to load this image file and display the image. More details can be checked with the attached matlab code in each demo case.

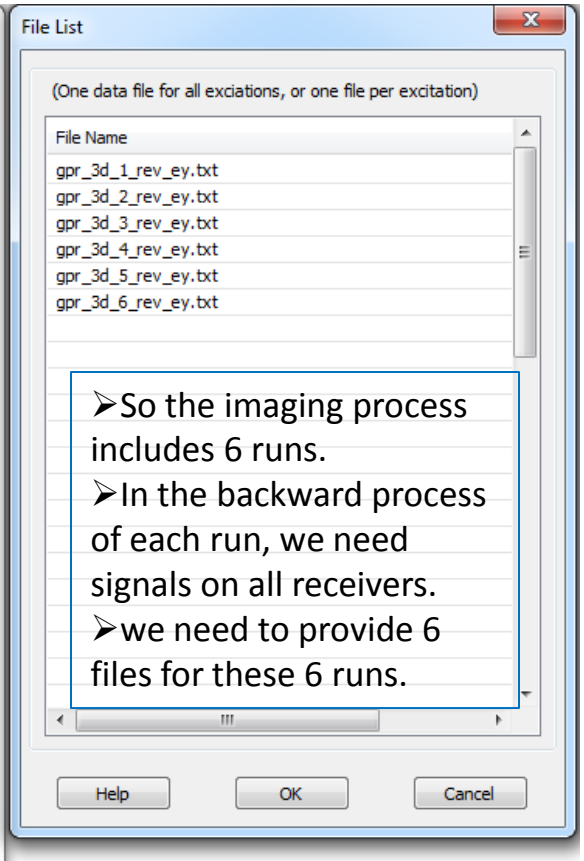
```
close all;

%%% define the data file name
sFile = 'a.img';

%%% open file as binary mode
fid = fopen( sFile, 'rb' ); % target file
if( fid == -1 )
   return;
end;

%% read 128 file header info
info = fread( fid, 128, '*char' );

%% file version number
version = fread( fid, 1, '*int' );

%% image grid range in the whole system, 6 numbers as [x0,y0,z0,x1,y1,z1]
img_range = fread( fid, 6, '*int' );

%% image size by cell number, 3 numbers as [nx,ny,nz]
img_sz = fread( fid, 3, '*int' );
sz = img_sz(1) * img_sz(2) * img_sz(3);

%% read whole array
my_img = fread( fid, double(sz), '*float' );

%% reshape this 1D data to 3D array
my_img = reshape( my_img, img_sz(3), img_sz(2), img_sz(1) );
                        %% the 3D array is ordered as [z, y, x]

%% close file
fclose( fid );

%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%  show image
slide_id = ceil(img_sz(1) / 2);

my_slide = my_img( :, :, slide_id );
my_slide = squeeze( my_slide );

figure;
imagesc( my_slide );
xlabel( 'X (cell)' );
ylabel( 'Z (cell)' );
```

File System for an Imaging Project with Imaging Scheme II

3D GRP Imaging with separated transmitter array and receiver array

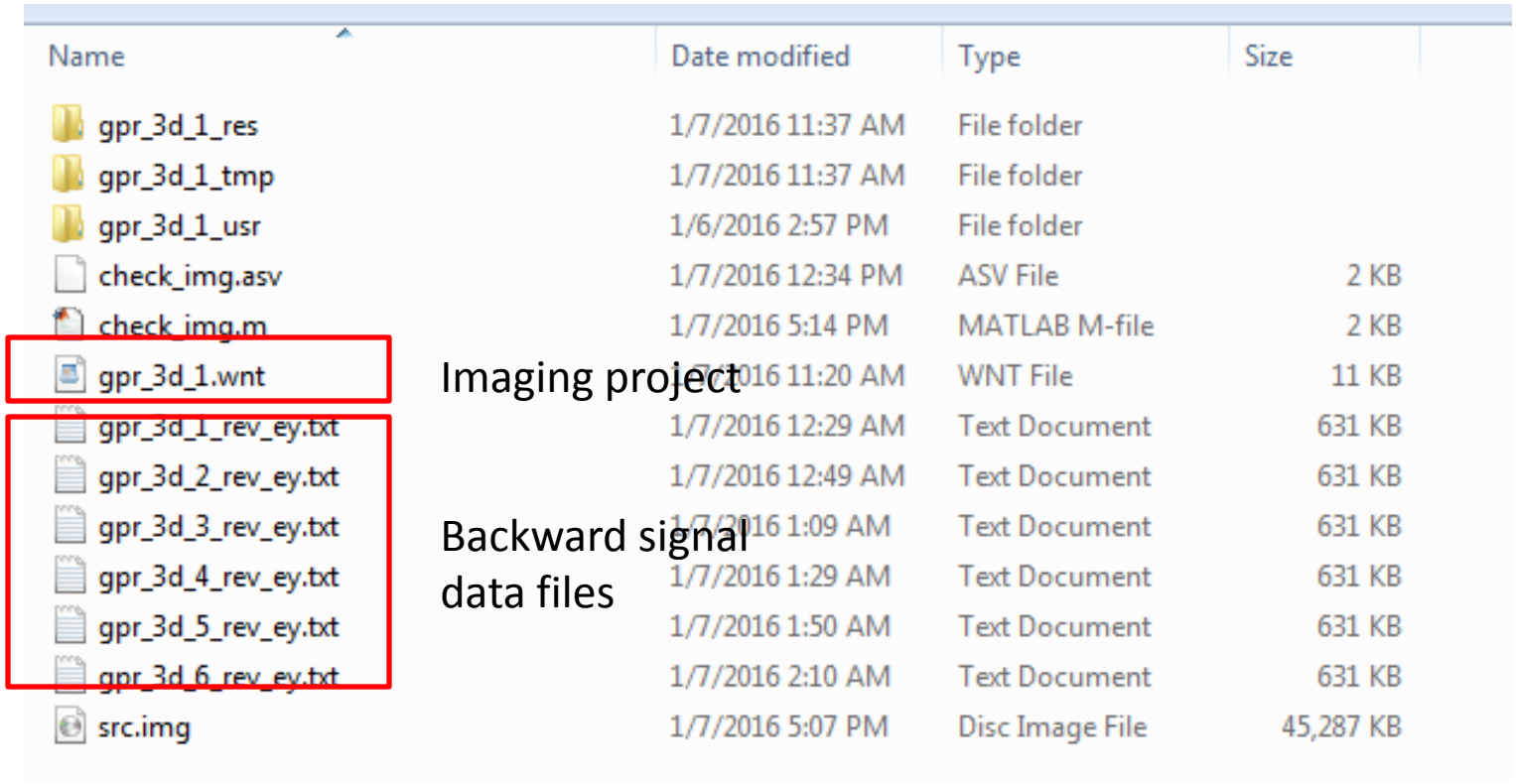






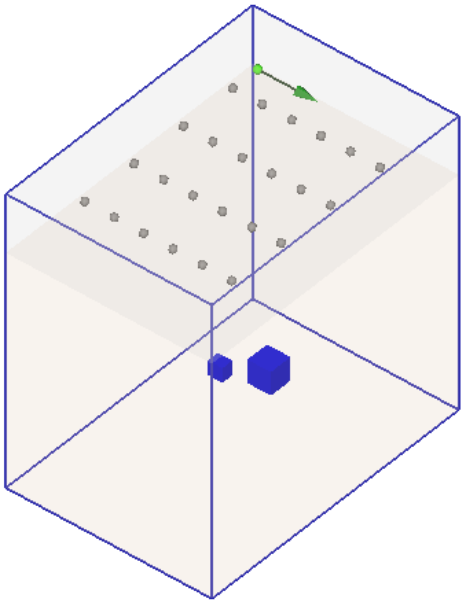
- So the imaging process includes 6 runs.
- In the backward process of each run, we need signals on all receivers.
- we need to provide 6 files for these 6 runs.

File system for this 3D imaging project

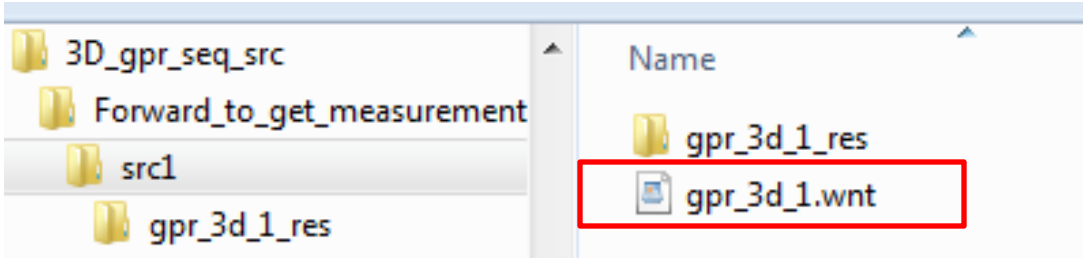


For example, data file “gpr_3d_1_rev_ey.txt”

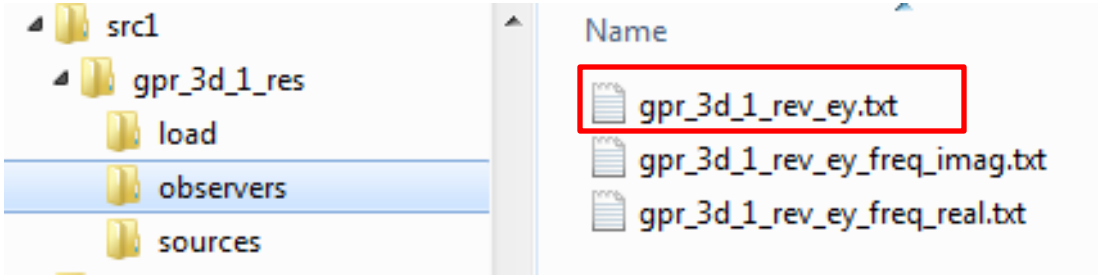
- can be user measurement but written as WCT format
- comes from a WCT EM project as following



Project



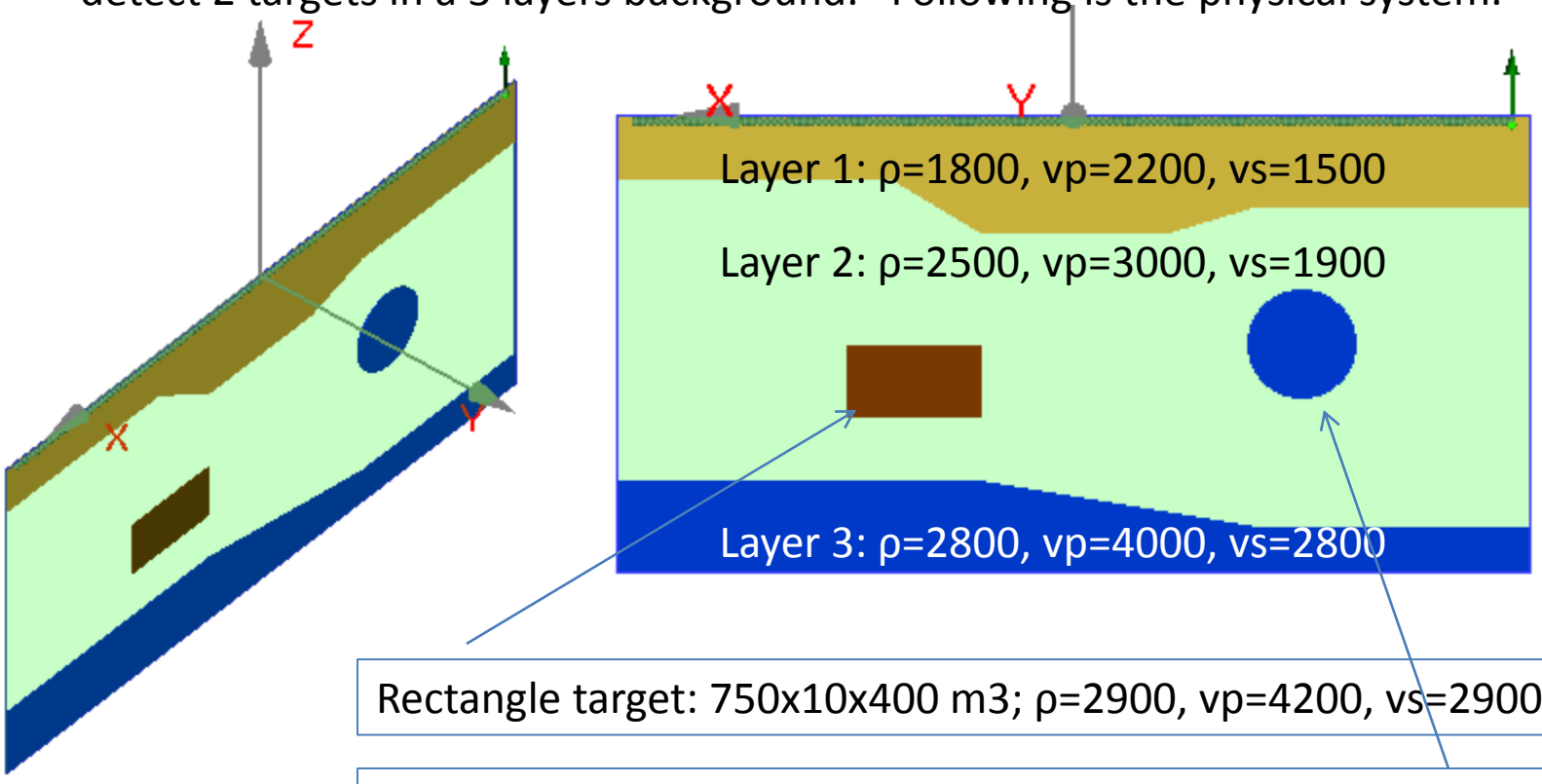
Received signal on receivers



Case I : 2D Imaging

Detecting Two objects in 3 Layers Media with Ricker Wave Source Pulse (f_{max}=27 Hz)

Here, we will demonstrate how to use WCT Cartesian EL imaging method to detect 2 targets in a 3 layers background. Following is the physical system.

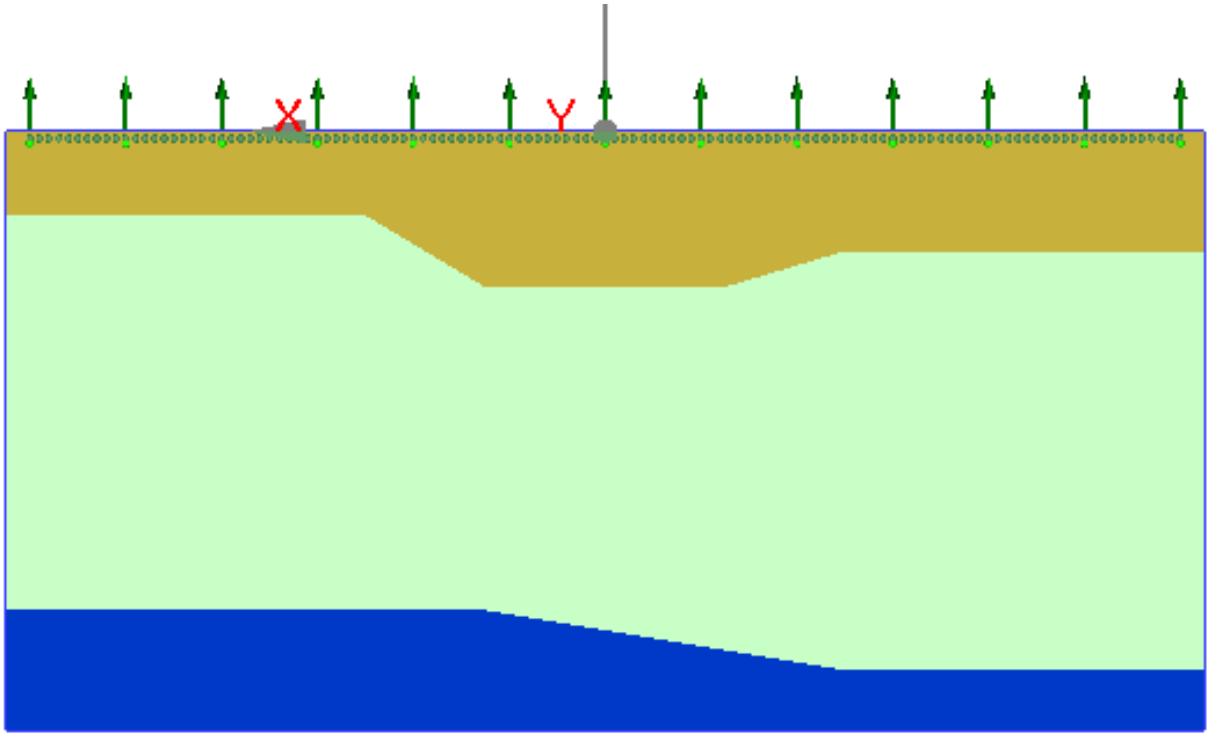


Rectangle target: 750x10x400 m3; ρ=2900, vp=4200, vs=2900

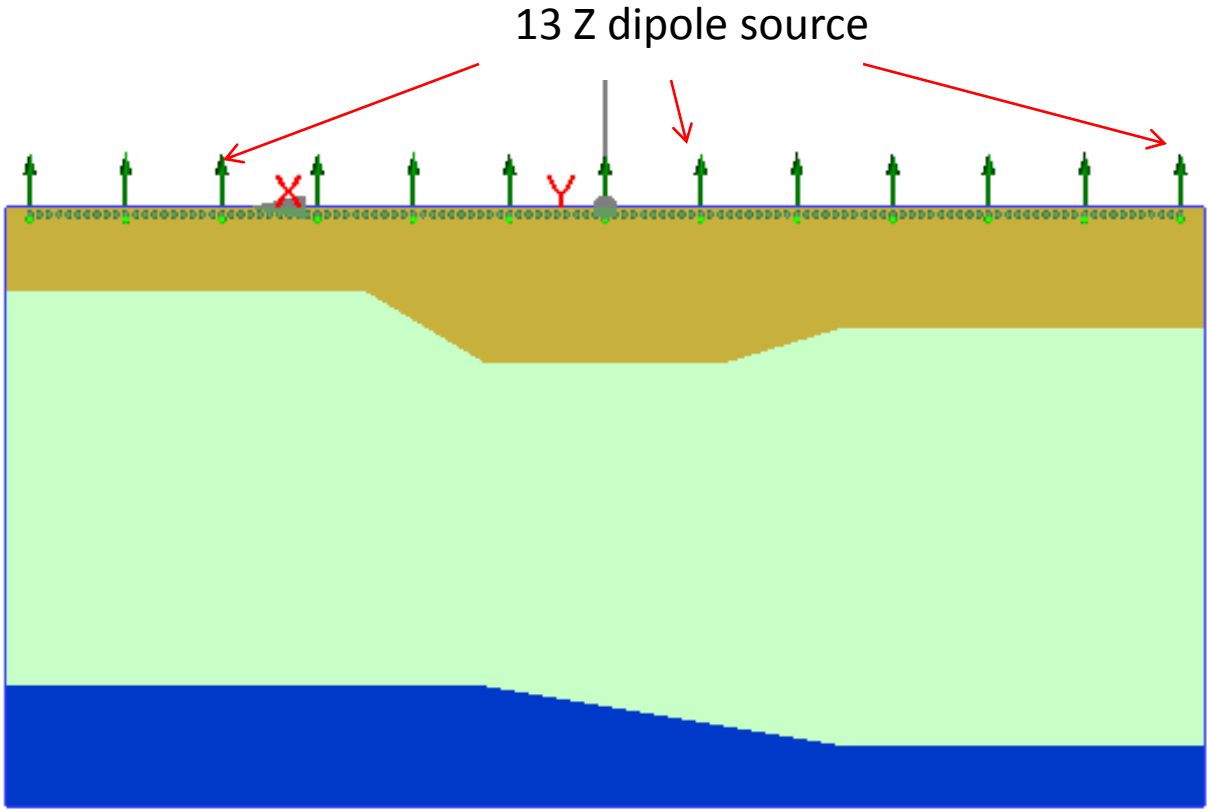
Circle target: r=300 m; ρ=3000, vp=4500, vs=3000

The Known Before Imaging

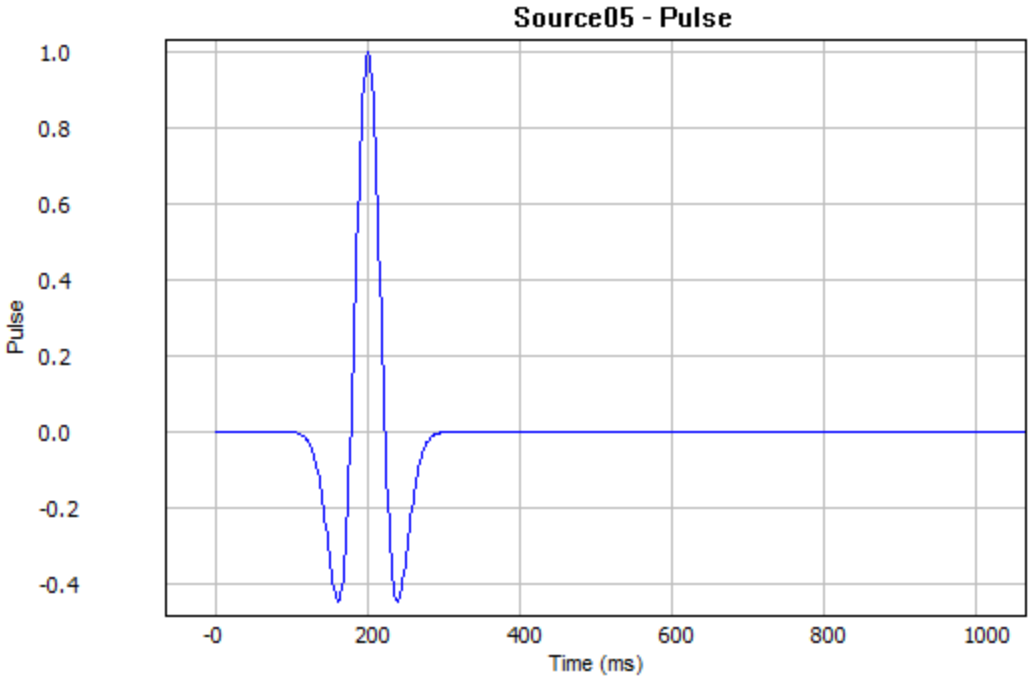
1. a known 3 layers background, as shown



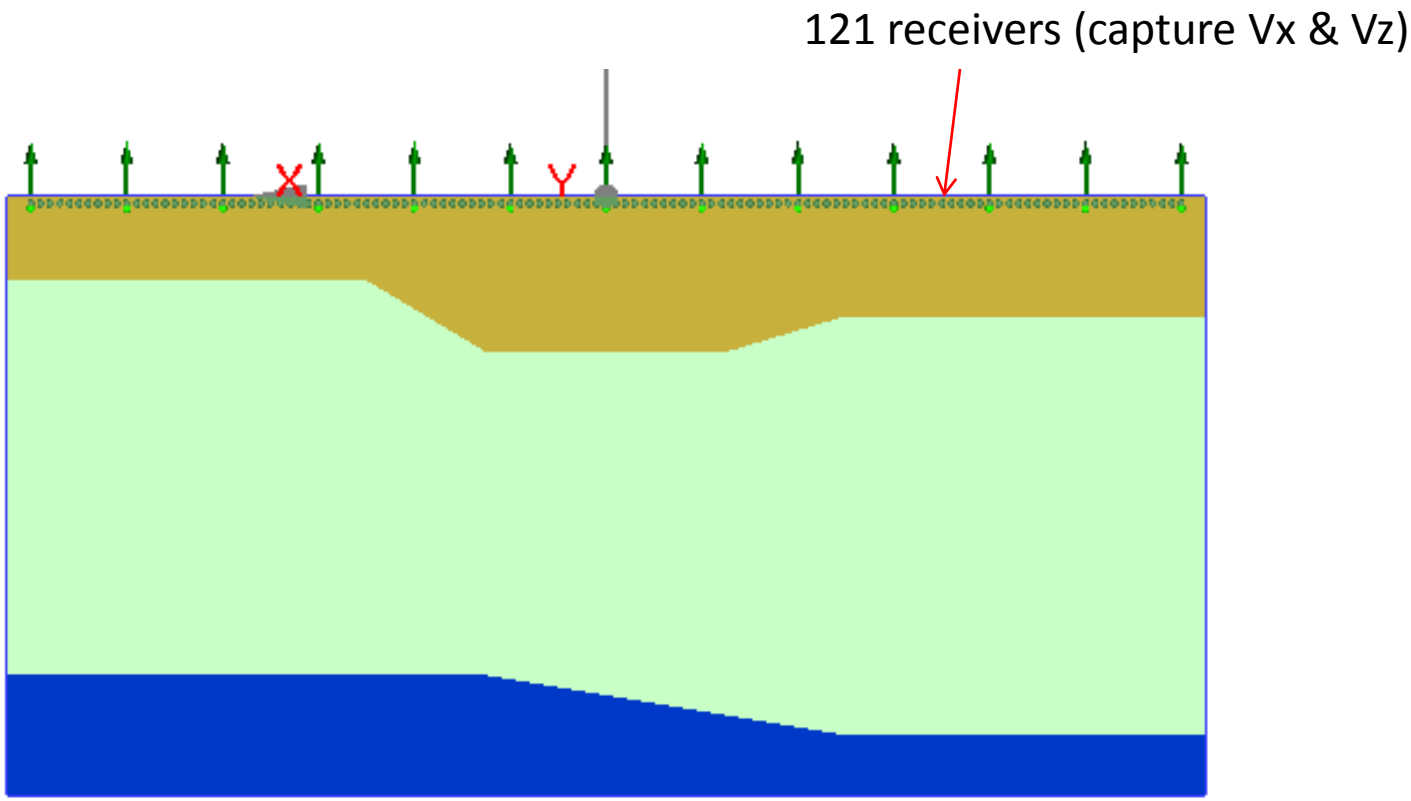
2. the position and the pulse of 13 point dipole sources, the source pulse is known as the Ricker wave at f_{max}=27 Hz



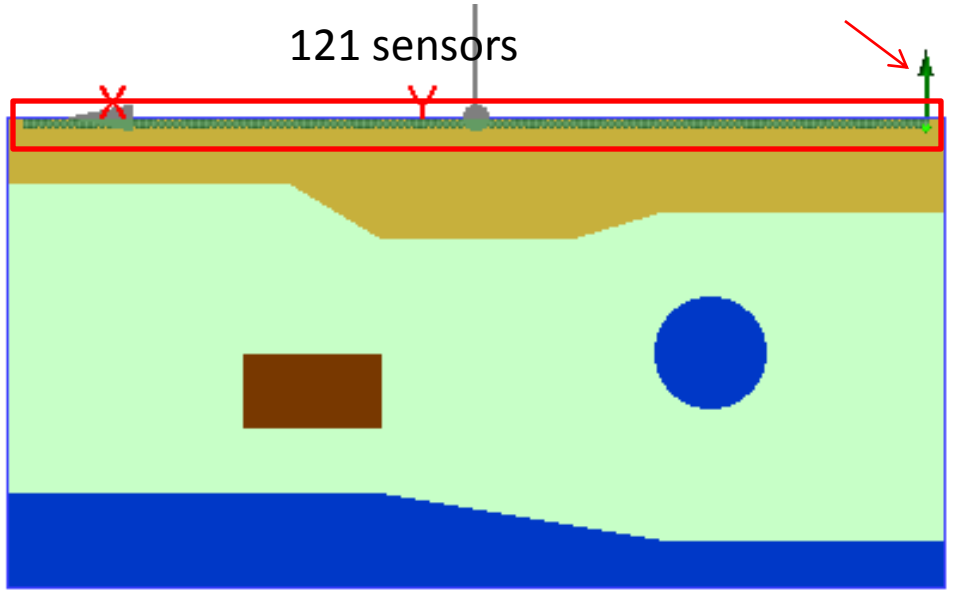
The Ricker Wave with fmax=27 Hz



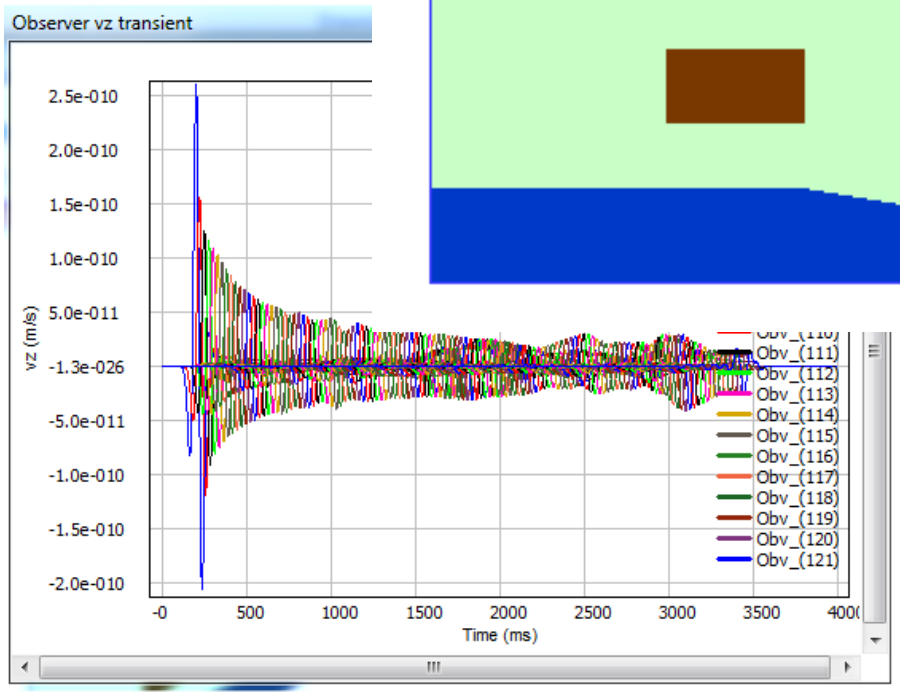
3. the position of 121 sensors to record Vx & Vz field



4. the detected signal Vx & Vz (measurement data) on these 121 sensors for each sources

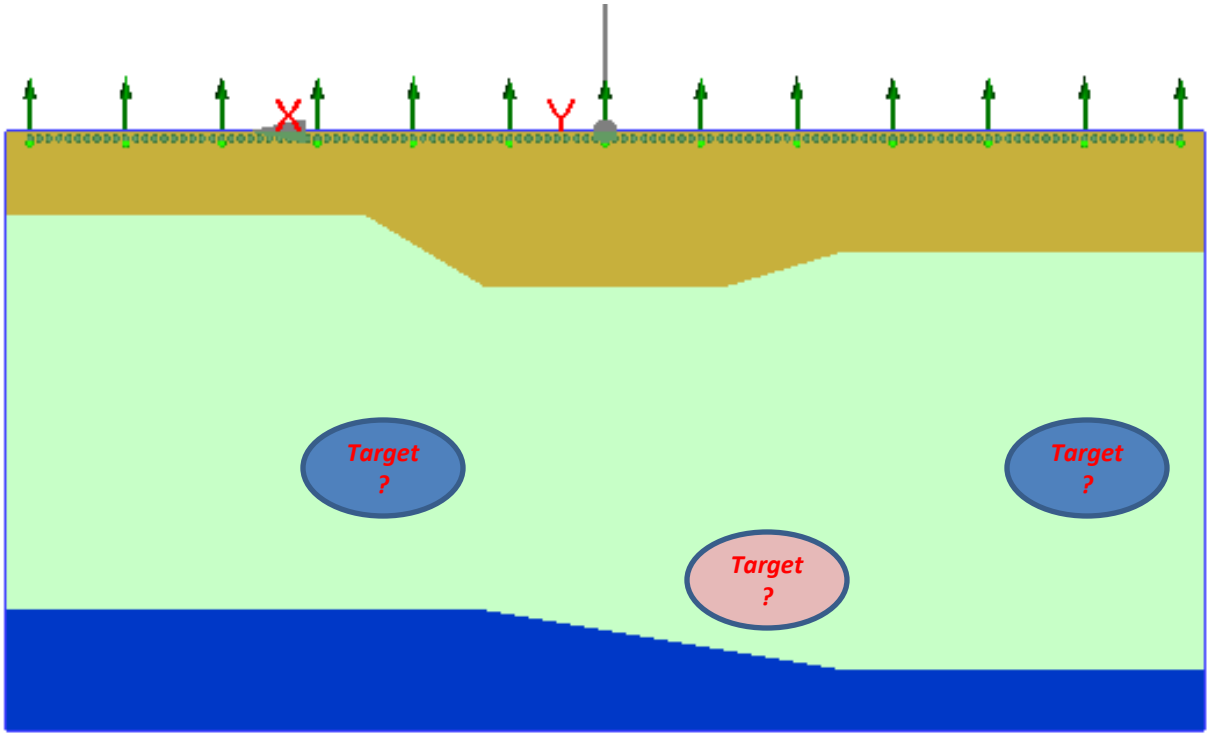


the signal on all 121 sensors from source #1



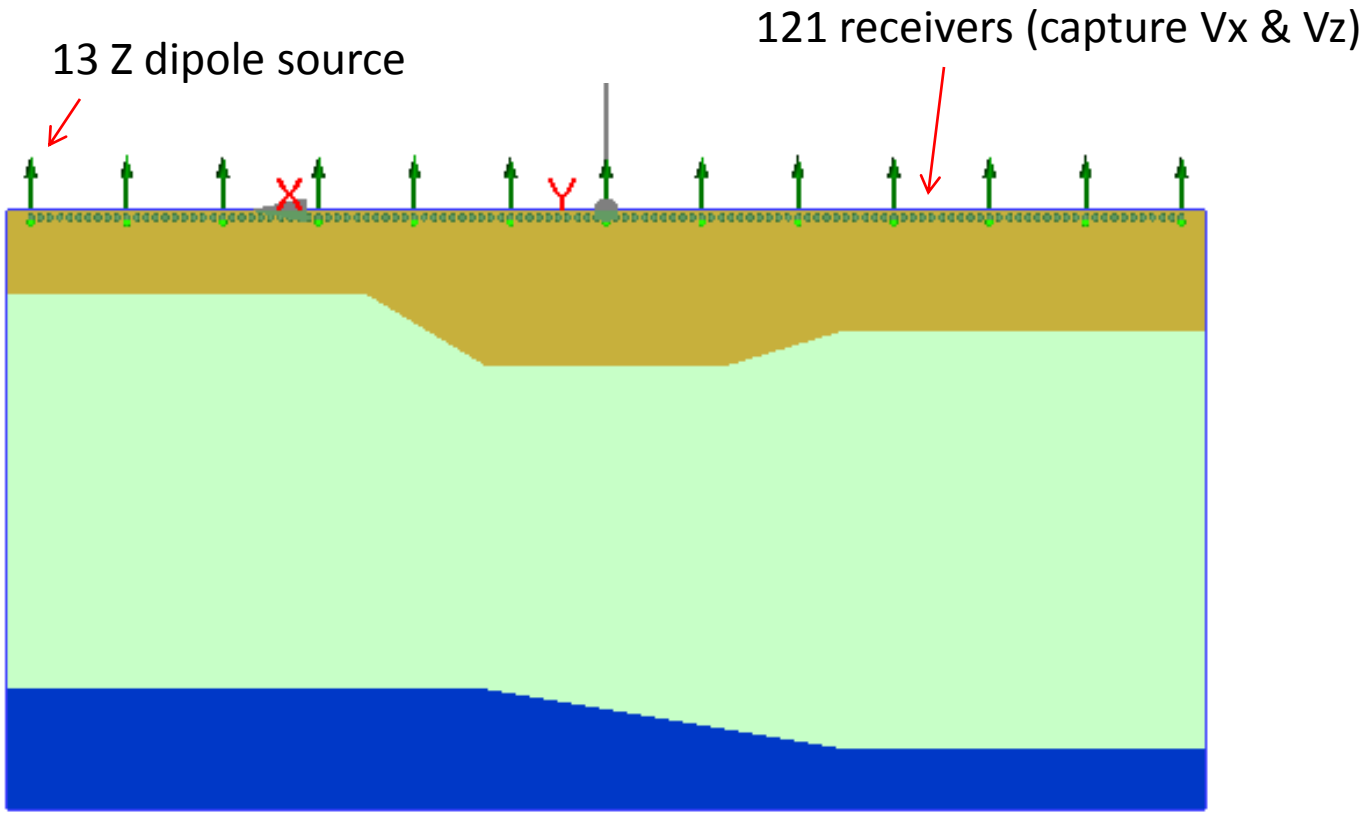
The Unknown in the System Before Imaging

What we don't know is whether there is anything in this 3 layers background, and where they are?



Goal of this Case

- We will
  - sweep 13 sources in a known 3 layers background with measurement data, as following,
    - To imaging
    - To check whether we can find something in the simulation space from the measurement data

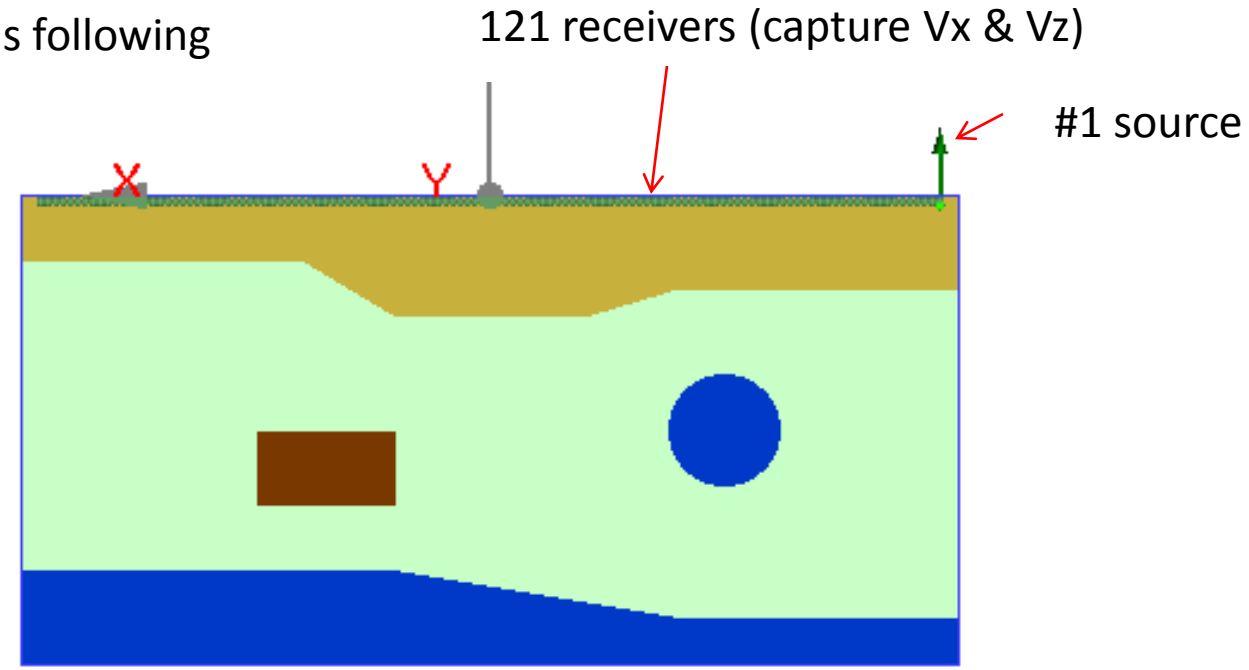


Step 1: Obtaining the Measurement Data

➢ it can be obtained from real measurement or ***forward simulation***

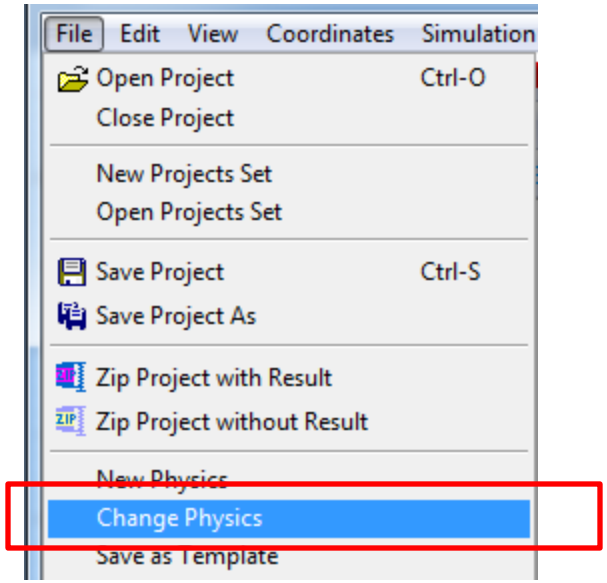
For example, if user want to use WCT EL solver to get the signal on sensors for source #1.

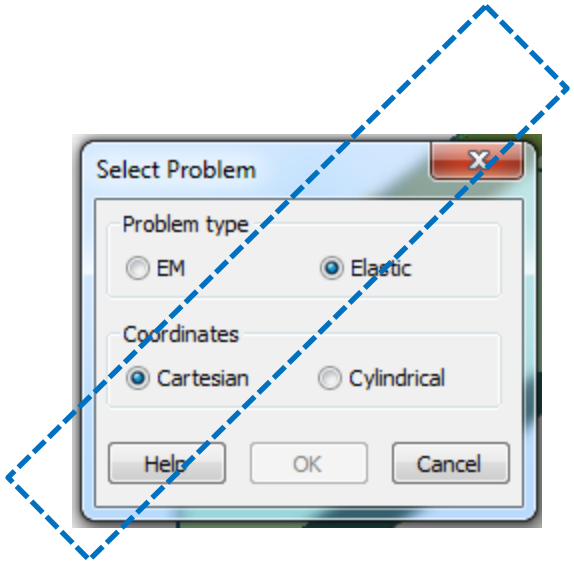
Setup a case as following



I. Setup WCT Cartesian EL project

Note: This case is already build in the demo package as "xxxx\2D\sequential_t\13_sources\forward\ case_4_rtm_2d_01.wnt"

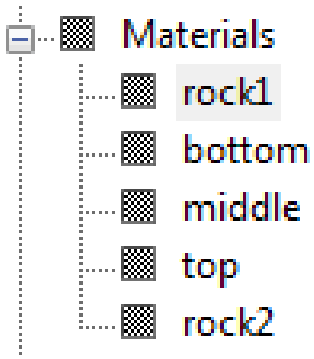


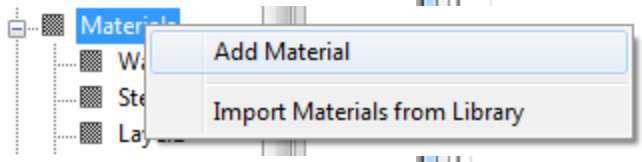


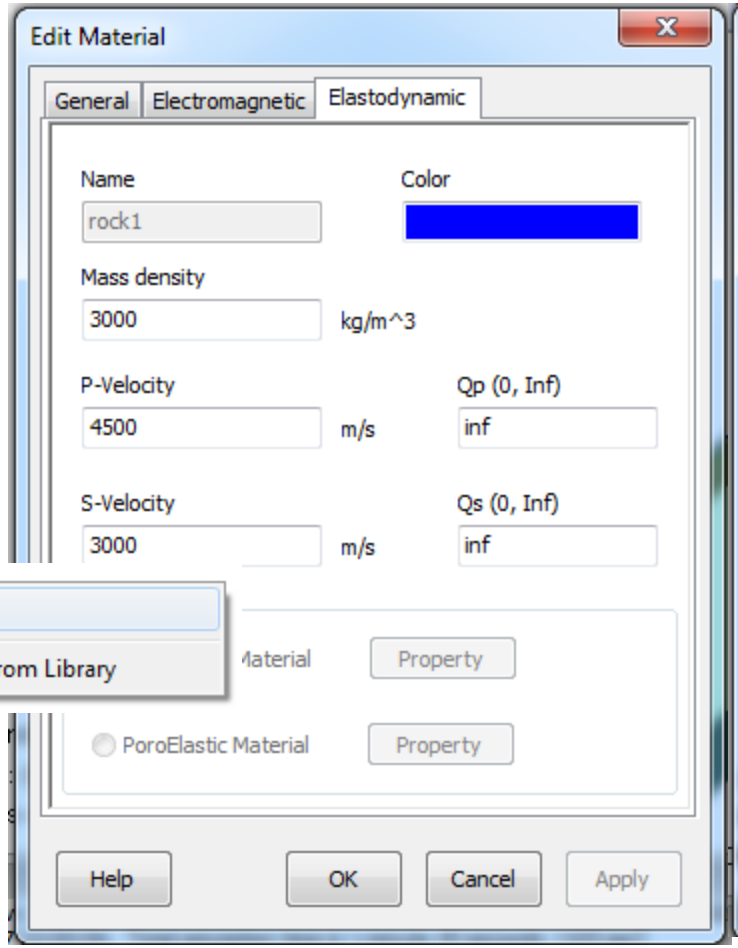
Then save as "case_4_rtm_2d_01.wnt"

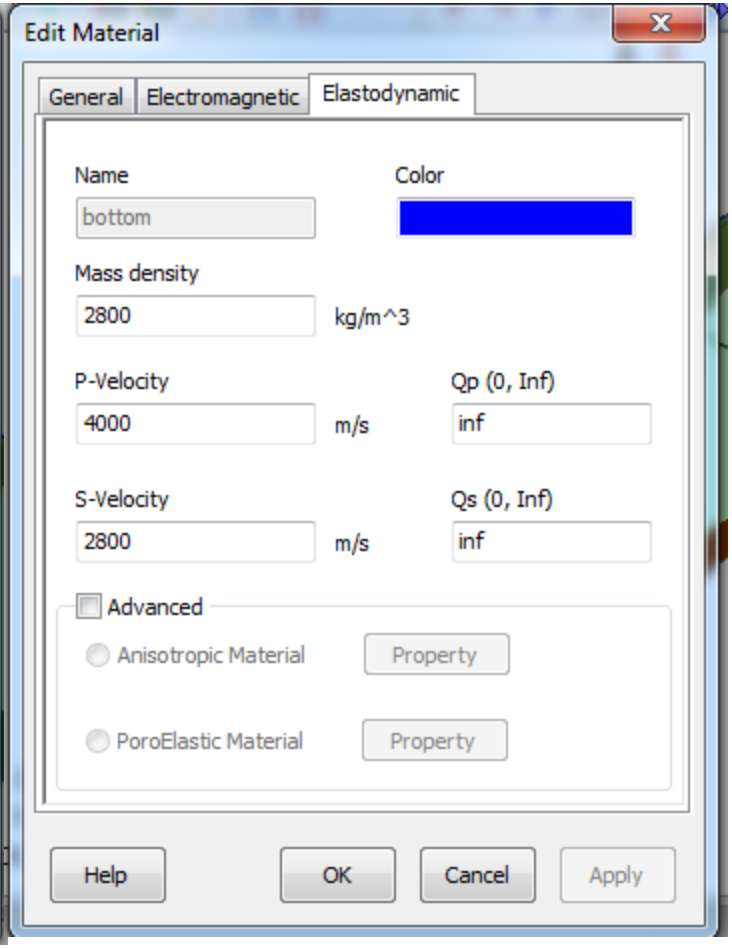
II. Setup materials

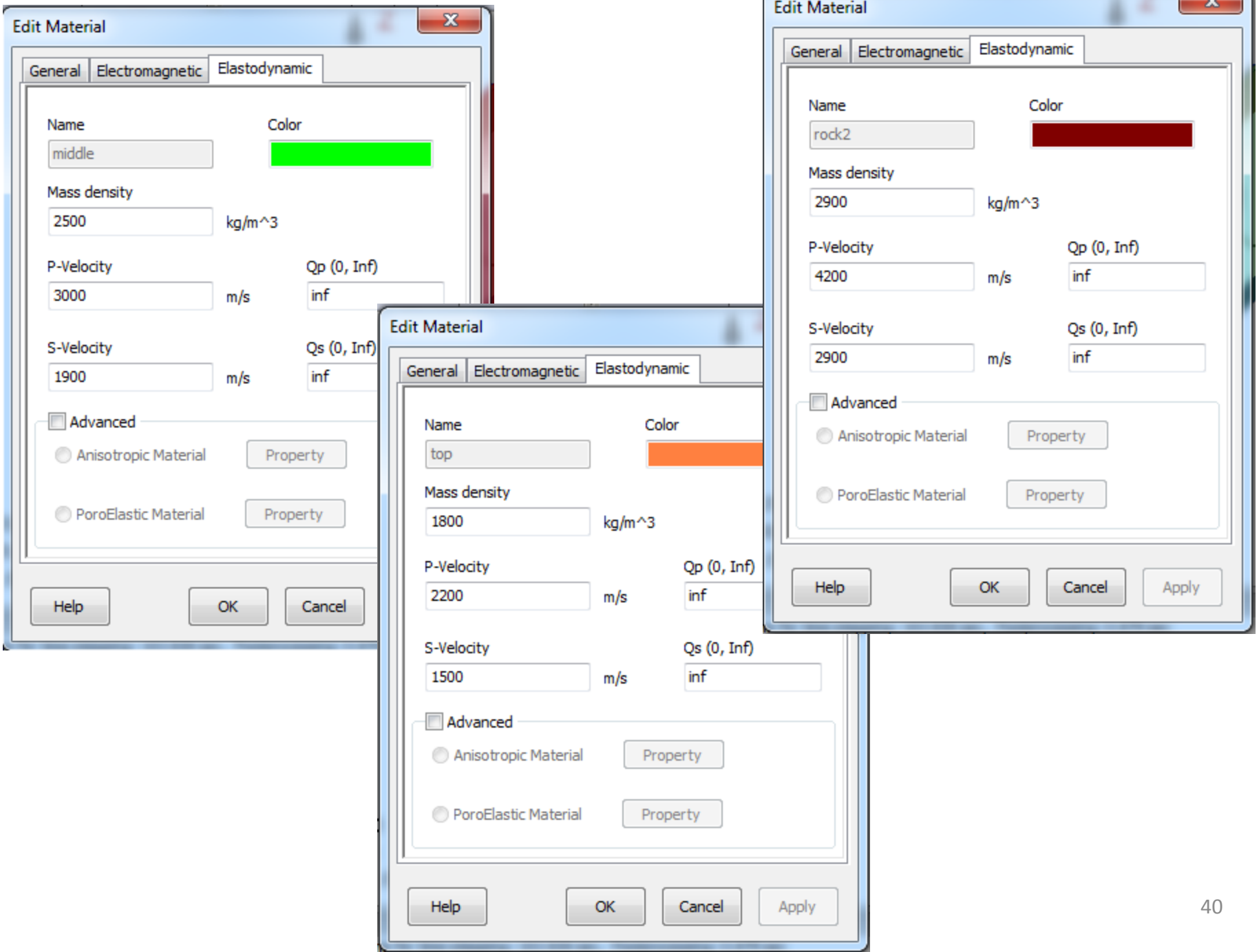
Materials



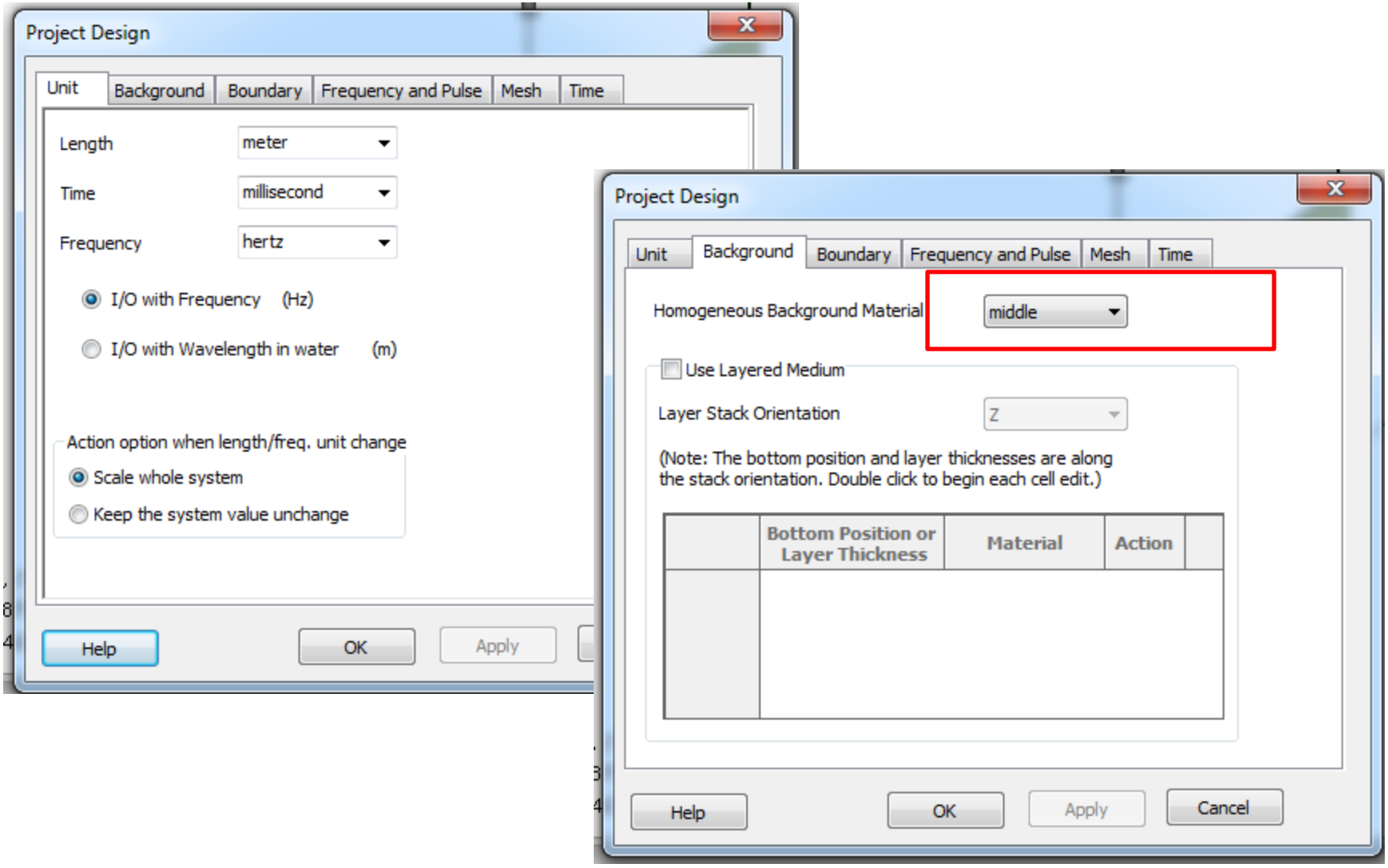


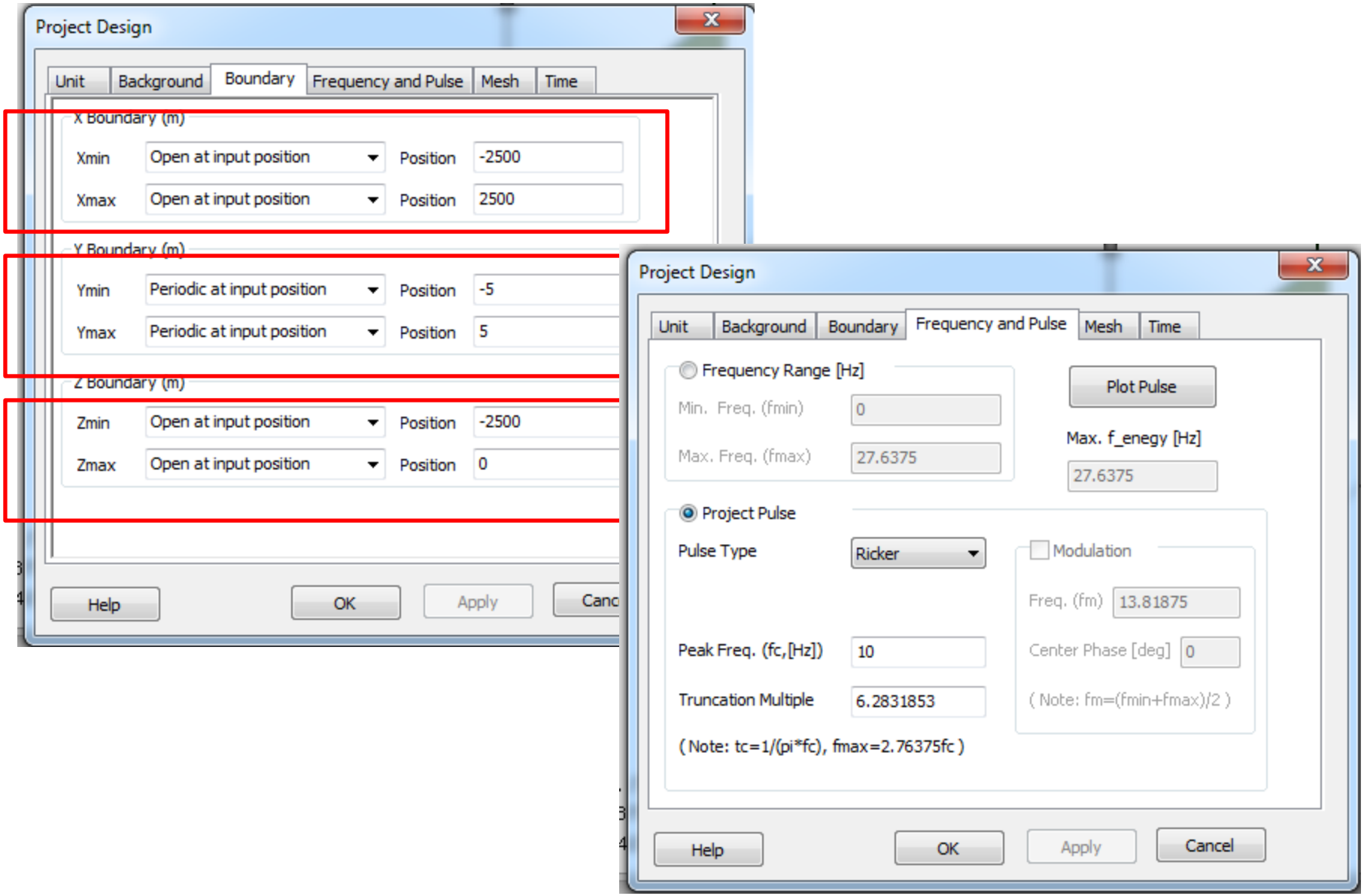


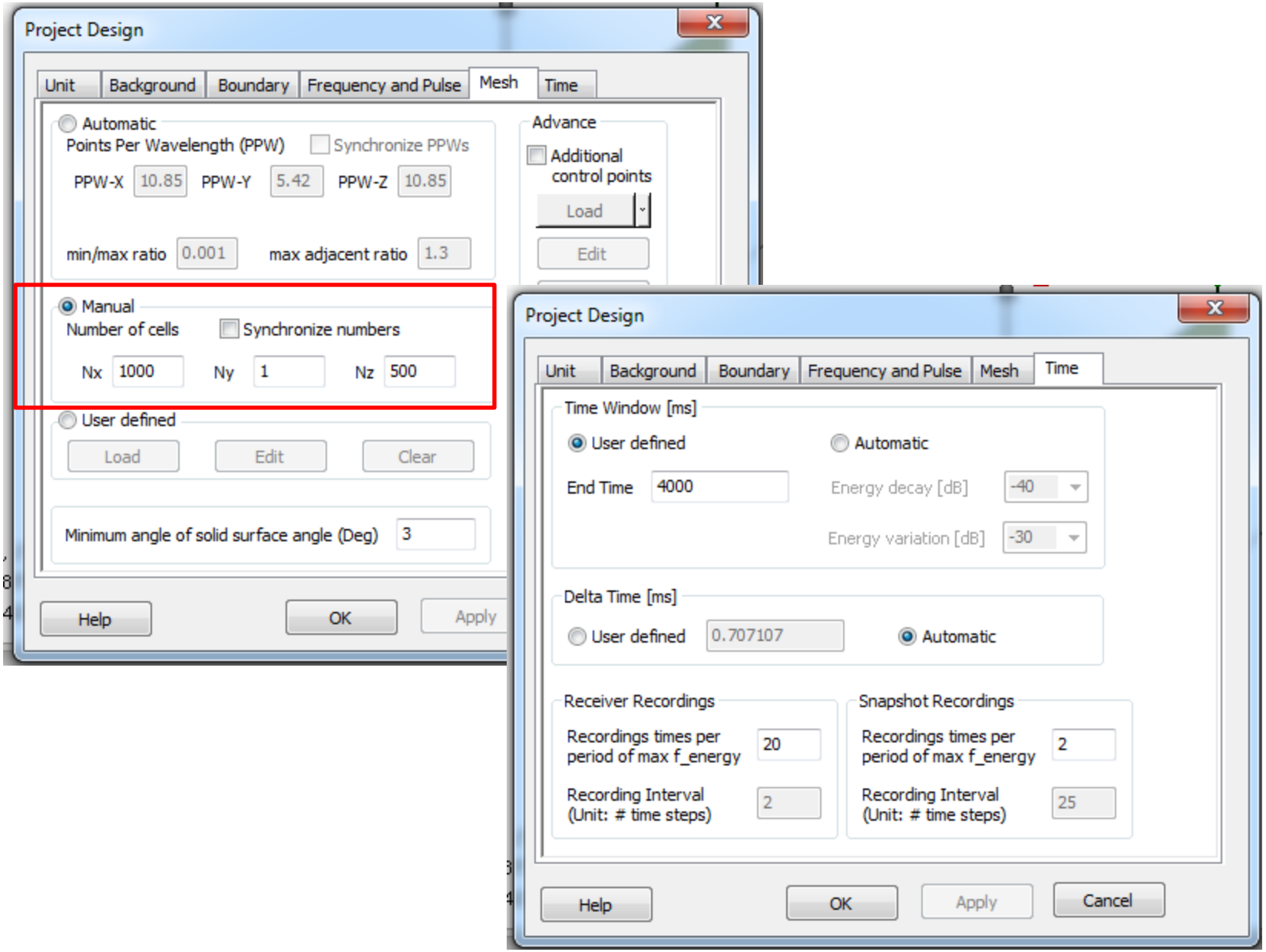



Edit Material
General
Electromagnetic
Elastodynamic
Name
middle
Color
Mass density
2500
kg/m^3
P-Velocity
3000
m/s
Qp (0, Inf)
inf
S-Velocity
1900
m/s
Qs (0, Inf)
inf
Advanced
Anisotropic Material
Property
PoroElastic Material
Property
Help
OK
Cancel
Edit Material
General
Electromagnetic
Elastodynamic
Name
top
Color
Mass density
1800
kg/m^3
P-Velocity
2200
m/s
Qp (0, Inf)
inf
S-Velocity
1500
m/s
Qs (0, Inf)
inf
Advanced
Anisotropic Material
Property
PoroElastic Material
Property
Help
OK
Cancel
Apply
Edit Material
General
Electromagnetic
Elastodynamic
Name
rock2
Color
Mass density
2900
kg/m^3
P-Velocity
4200
m/s
Qp (0, Inf)
inf
S-Velocity
2900
m/s
Qs (0, Inf)
inf
Advanced
Anisotropic Material
Property
PoroElastic Material
Property
Help
OK
Cancel
Apply

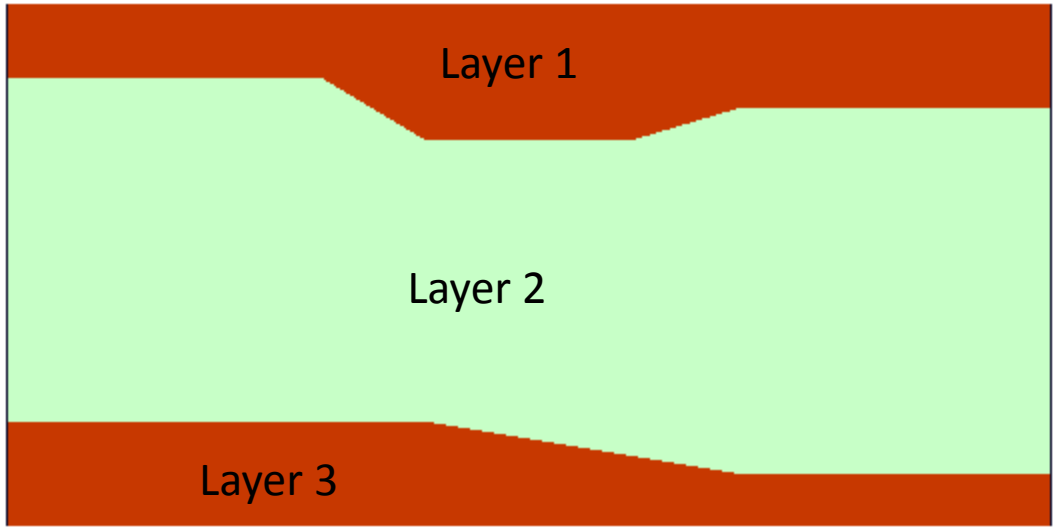
III. Setup project background, pulse, mesh & time system




Project Design
Unit
Background
Boundary
Frequency and Pulse
Mesh
Time
X Boundary (m)
Xmin
Open at input position
Position
-2500
Xmax
Open at input position
Position
2500
Y Boundary (m)
Ymin
Periodic at input position
Position
-5
Ymax
Periodic at input position
Position
5
Z Boundary (m)
Zmin
Open at input position
Position
-2500
Zmax
Open at input position
Position
0
Help
OK
Apply
Canc
Project Design
Unit
Background
Boundary
Frequency and Pulse
Mesh
Time
Frequency Range [Hz]
Min. Freq. (fmin)
0
Max. Freq. (fmax)
27.6375
Plot Pulse
Max. f_enegy [Hz]
27.6375
Project Pulse
Pulse Type
Ricker
Modulation
Freq. (fm)
13.81875
Peak Freq. (fc,[Hz])
10
Center Phase [deg]
0
Truncation Multiple
6.2831853
(Note: fm=(fmin+fmax)/2)
(Note: tc=1/(pi*fc), fmax=2.76375fc)
Help
OK
Apply
Cancel


Project Design
Unit
Background
Boundary
Frequency and Pulse
Mesh
Time
Automatic
Points Per Wavelength (PPW)
Synchronize PPWs
PPW-X 10.85
PPW-Y 5.42
PPW-Z 10.85
min/max ratio 0.001
max adjacent ratio 1.3
Advance
Additional control points
Load
Edit
Manual
Number of cells
Synchronize numbers
Nx 1000
Ny 1
Nz 500
User defined
Load
Edit
Clear
Minimum angle of solid surface angle (Deg) 3
Help
OK
Apply
Project Design
Unit
Background
Boundary
Frequency and Pulse
Mesh
Time
Time Window [ms]
User defined
Automatic
End Time 4000
Energy decay [dB] -40
Energy variation [dB] -30
Delta Time [ms]
User defined 0.707107
Automatic
Receiver Recordings
Recordings times per period of max f_energy 20
Recording Interval (Unit: # time steps) 2
Snapshot Recordings
Recordings times per period of max f_energy 2
Recording Interval (Unit: # time steps) 25
Help
OK
Apply
Cancel

IV. Setup 3 layers background

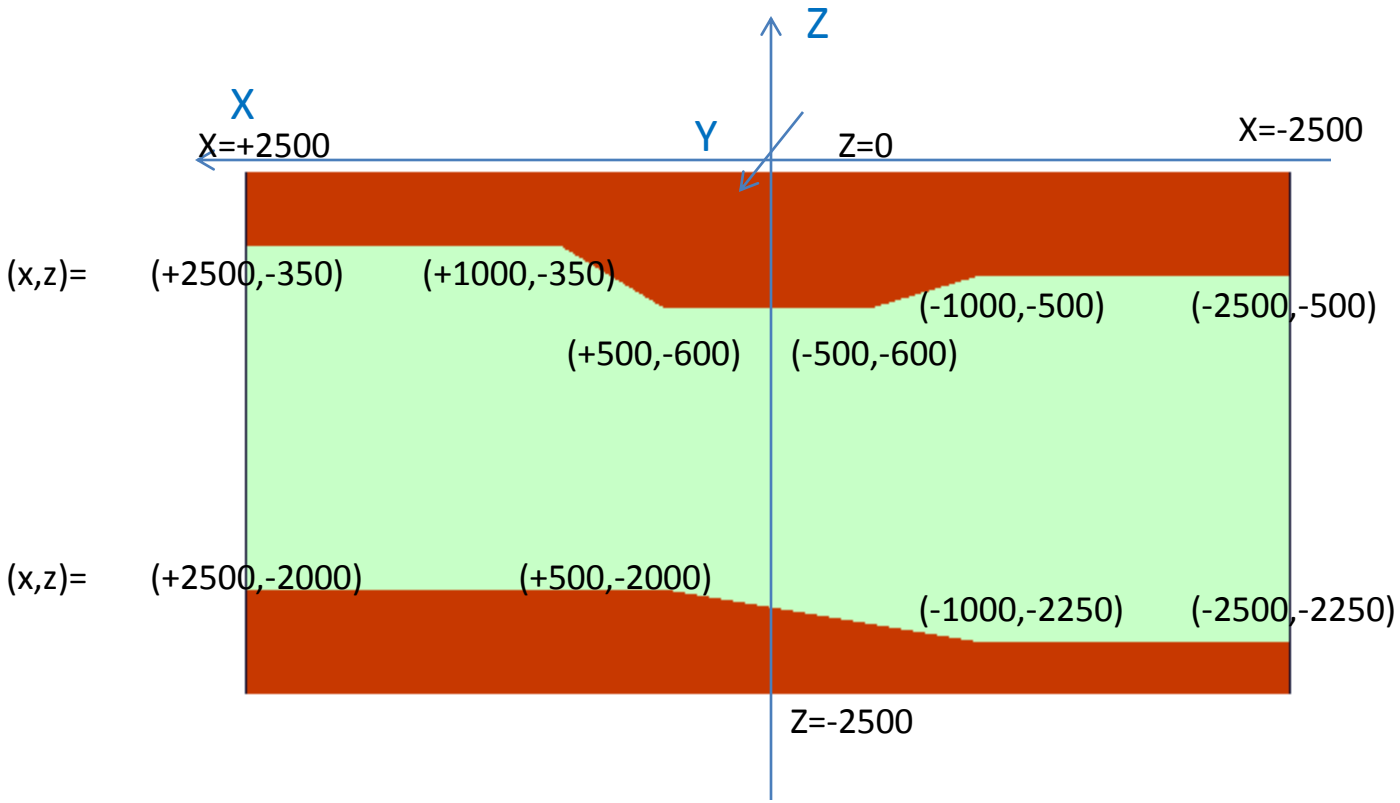


We can use following 3D geometry system to simplify the modeling

- homogeneous background by the material of layer 2
- top layer is modeled as polygon cylinder
- bottom layer is modeled as polygon cylinder also

❑ due to the region outside the top and bottom layer will be the background, the layer 2 will be built automatically after top & bottom layers are built

Firstly, we need to know the vertexes of all layers

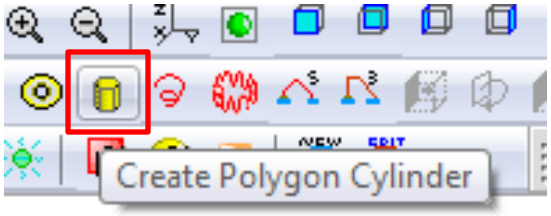


➢ Build top layer

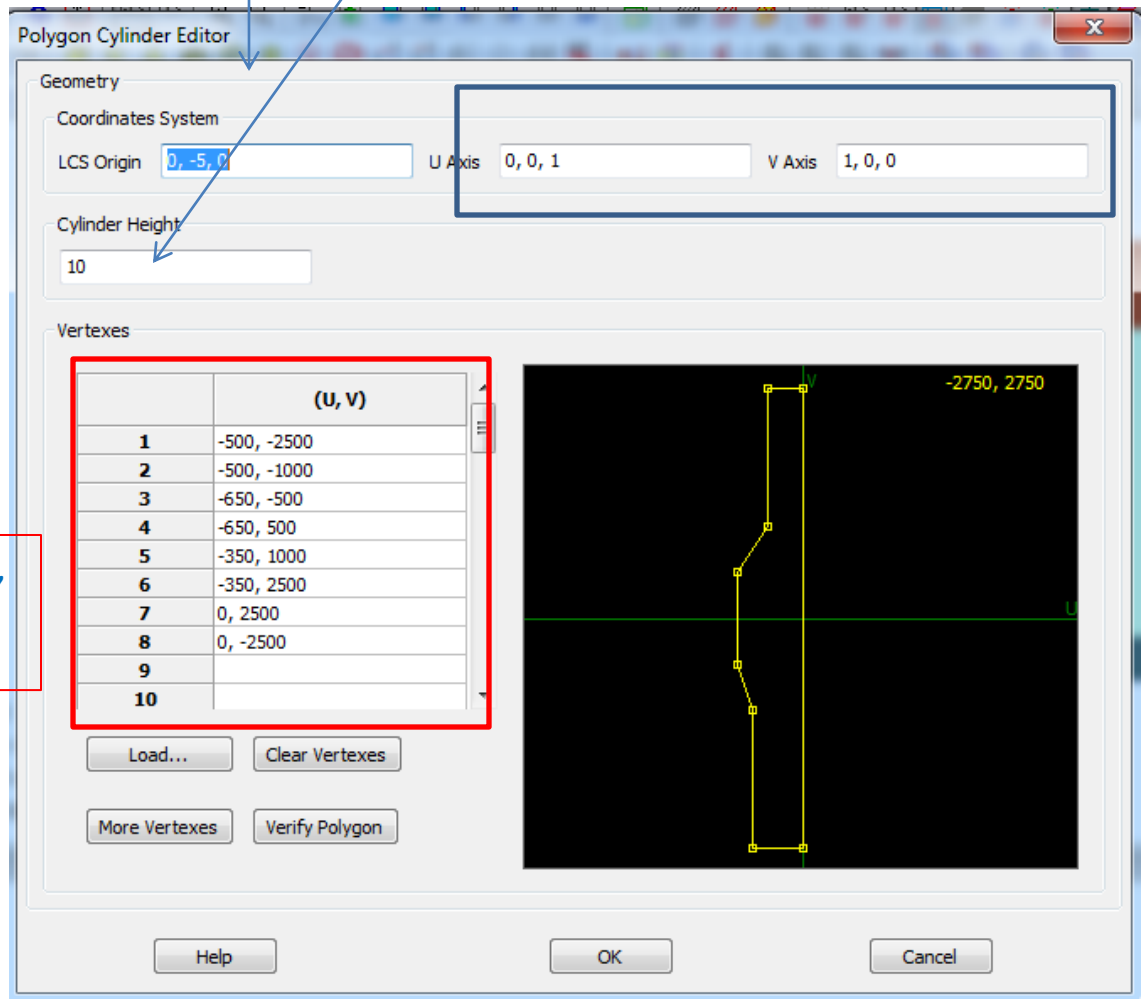
The cylinder will start from y=-5, so, enter (0,-5,0) as the LCS origin. Height is 10.

Due to the cylinder will grow along +Y in WCS, the U, V axis of LCS is as following, so, the W axis is WCS (0,1,0) based on U, V

Use this button to enter Polygon Cylinder Editor



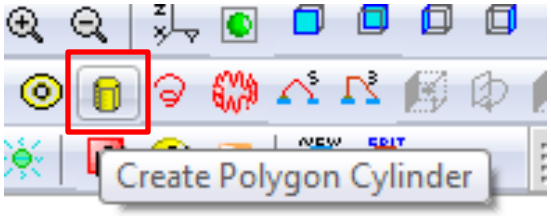
The vertices of polygon, here, as we know, U is WCS's Z, V is WCS's X



Set top layer using material "top"

➢ Then bottom layer

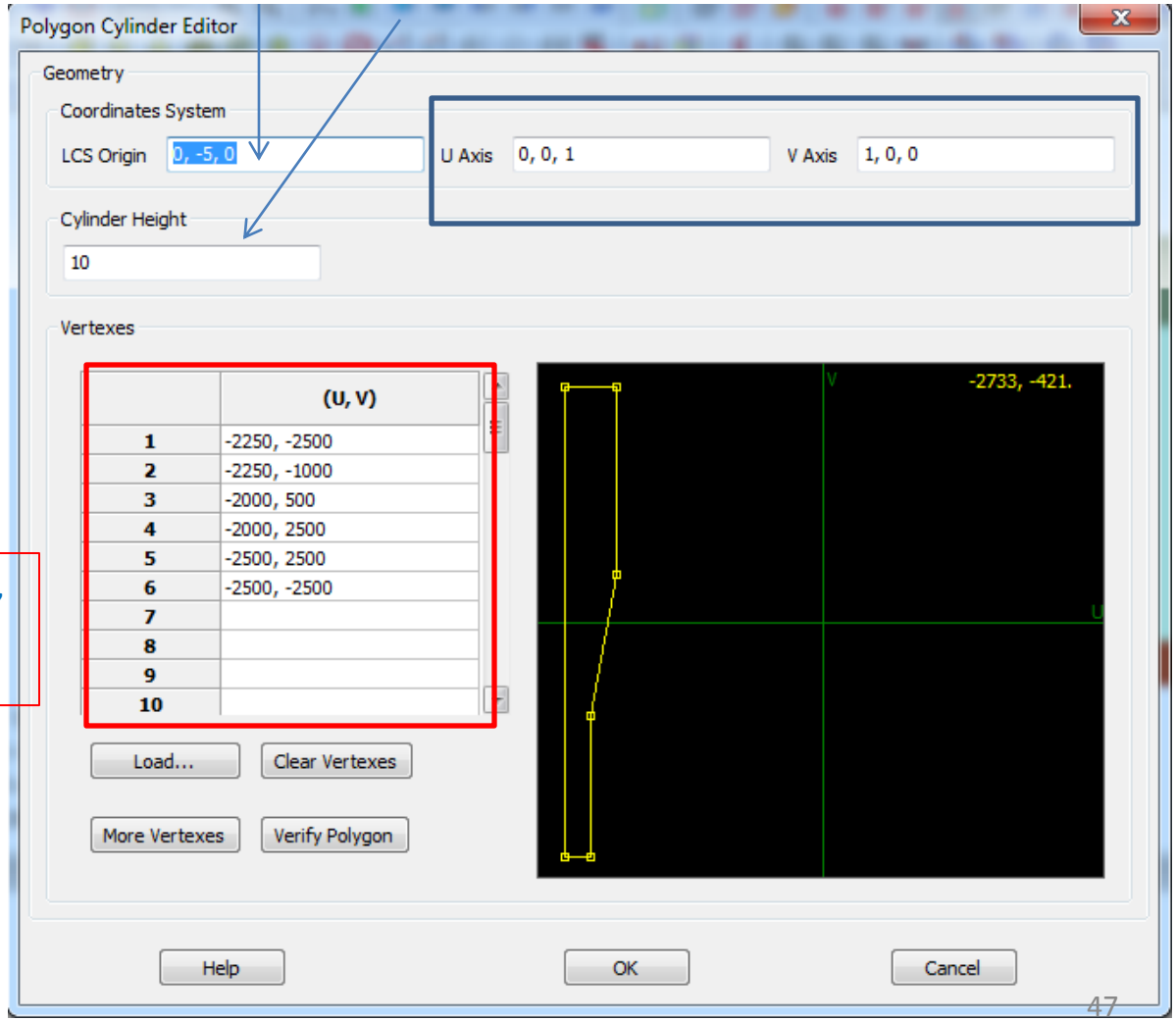
Use this button to enter Polygon Cylinder Editor



The cylinder will start from y=-5, so, enter (0,-5,0) as the LCS origin. Height is 10.

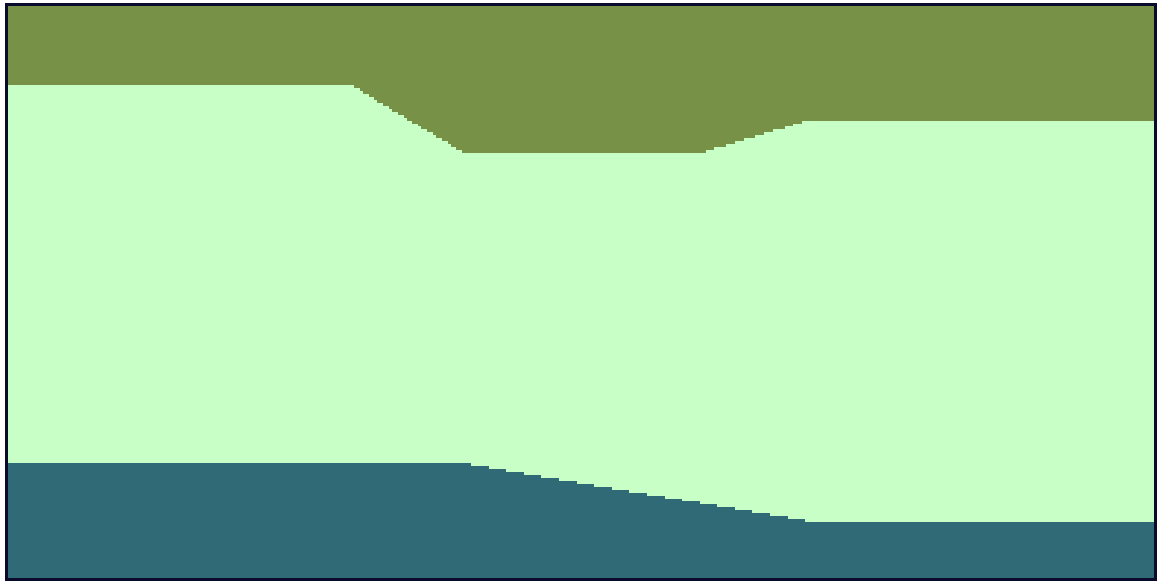
Due to the cylinder will grow along +Y in WCS, the U, V axis of LCS is as following, so, the W axis is WCS (0,1,0) based on U, V

The vertices of polygon, here, as we know, U is WCS's Z, V is WCS's X

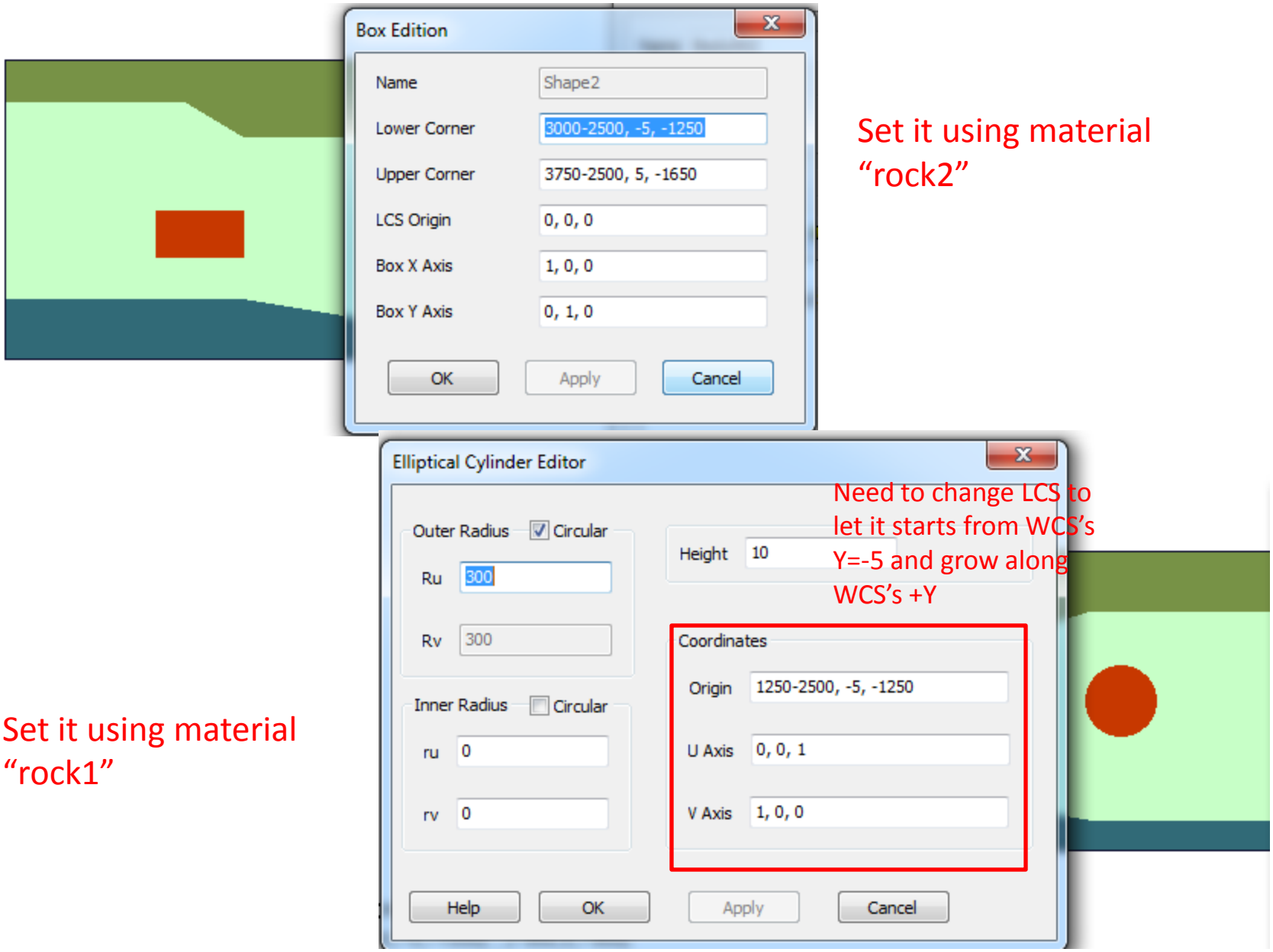


Set bottom layer using material "bottom"

- After the material for each layer is set, we get this

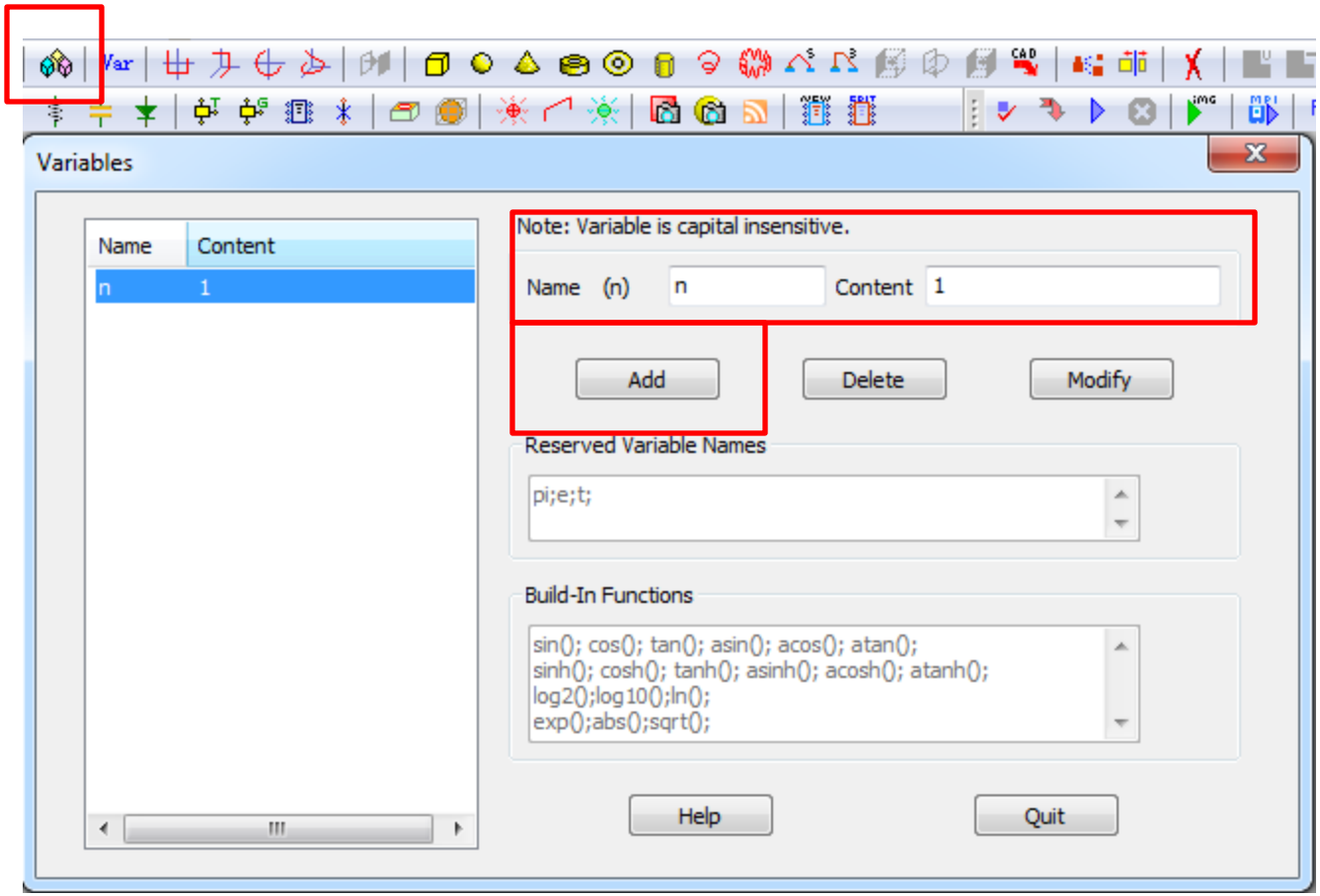


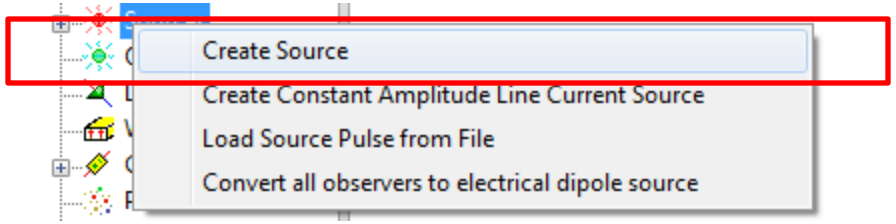
V. Define two targets & the material for them



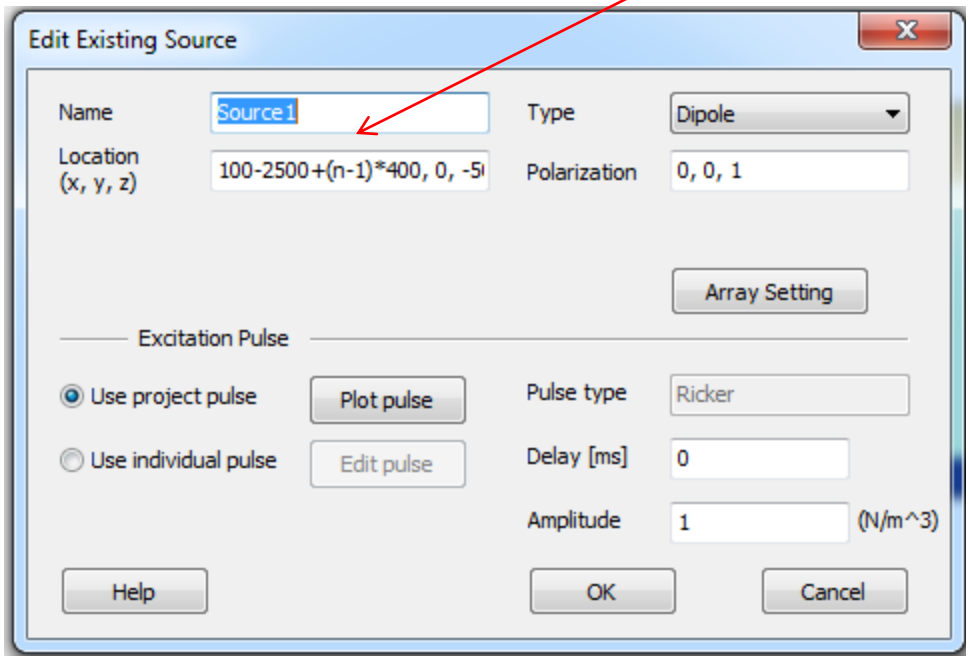
V. Define the source through ***Variable system***

Define a variable ***n***, default value is ***1***

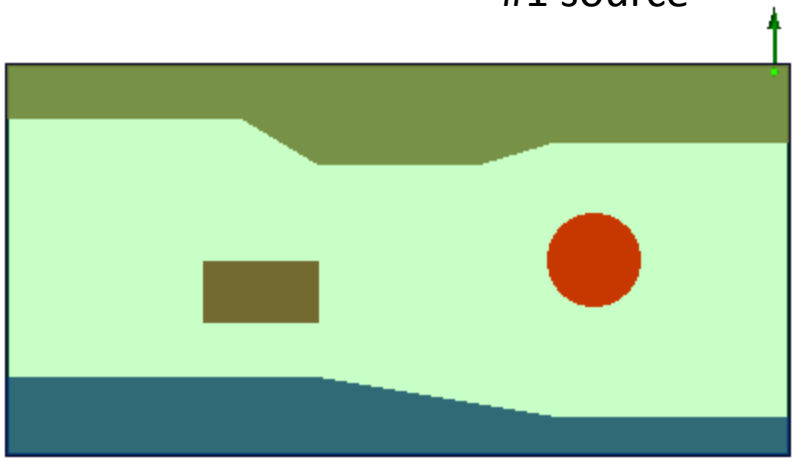




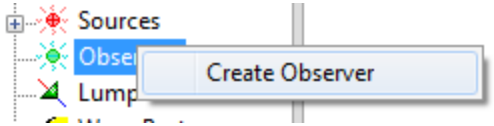
Define a Z dipole source, the position of the source uses variable ***n***



#1 source



VI. Define 121 receivers to record Vx & Vz

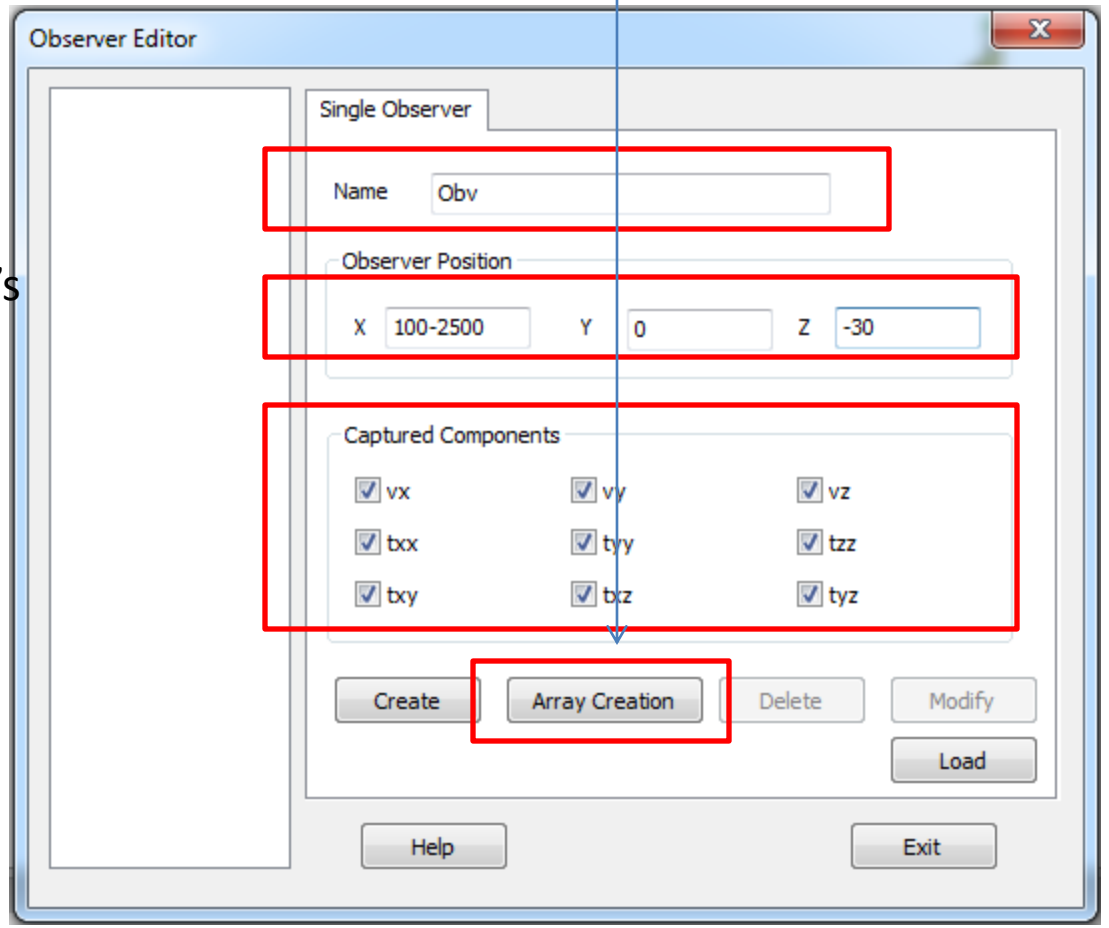


We will use ***Array-Creation*** to create 121 receivers in one step

Base name

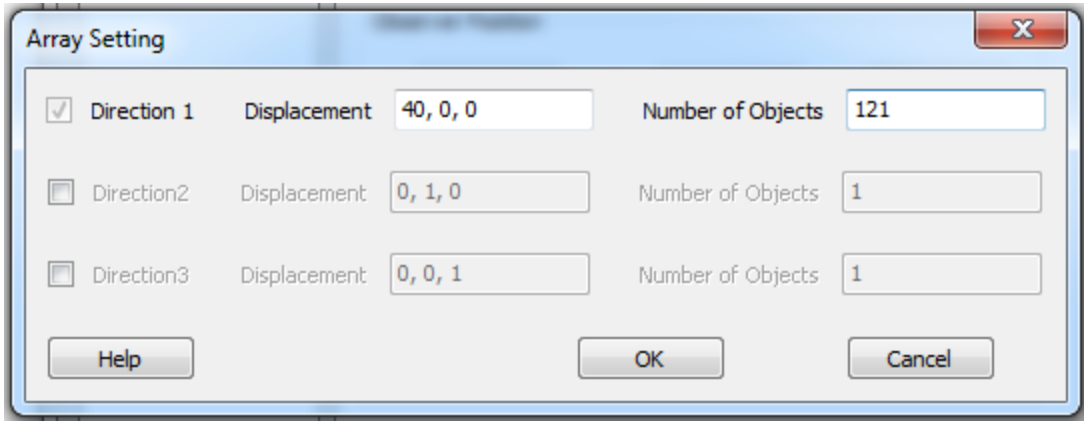
The 1st receiver's position

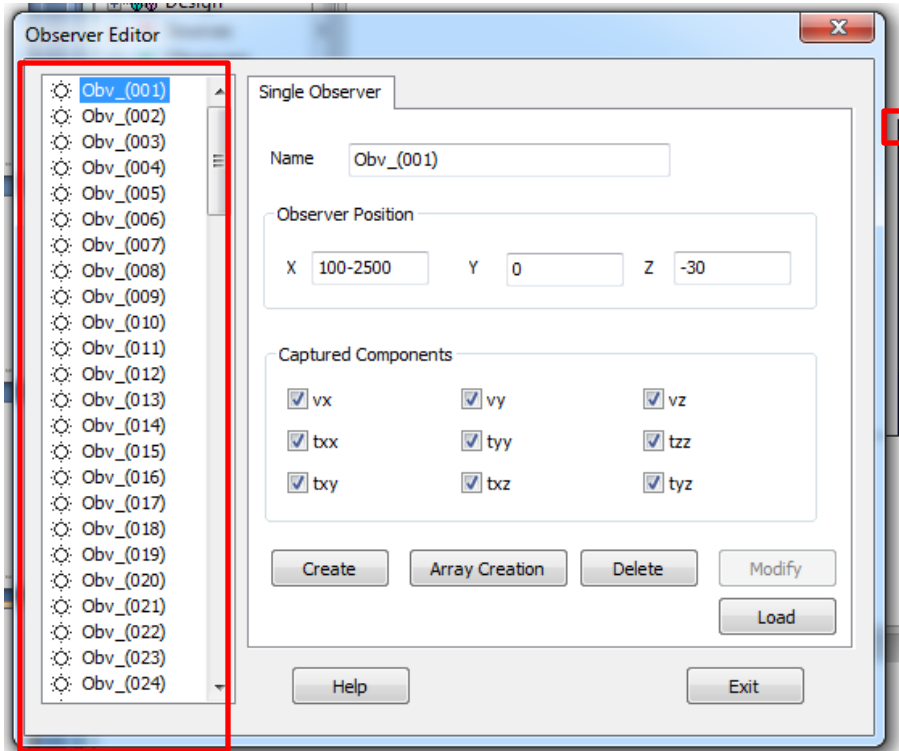
Receiver recording components



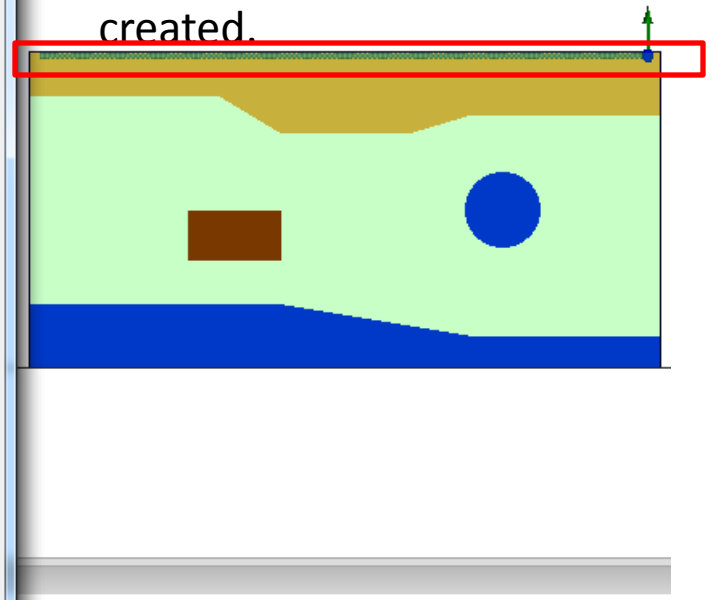
The receivers in the array has a distance of (40, 0, 0)

Totally 121 receivers in this creation

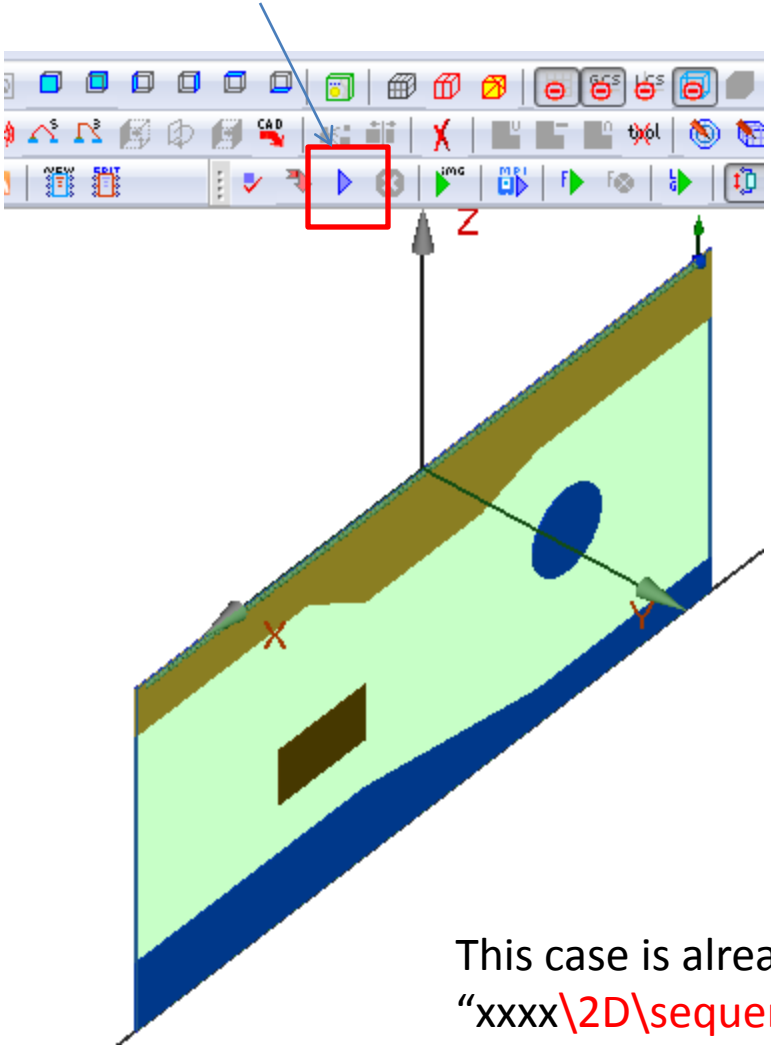




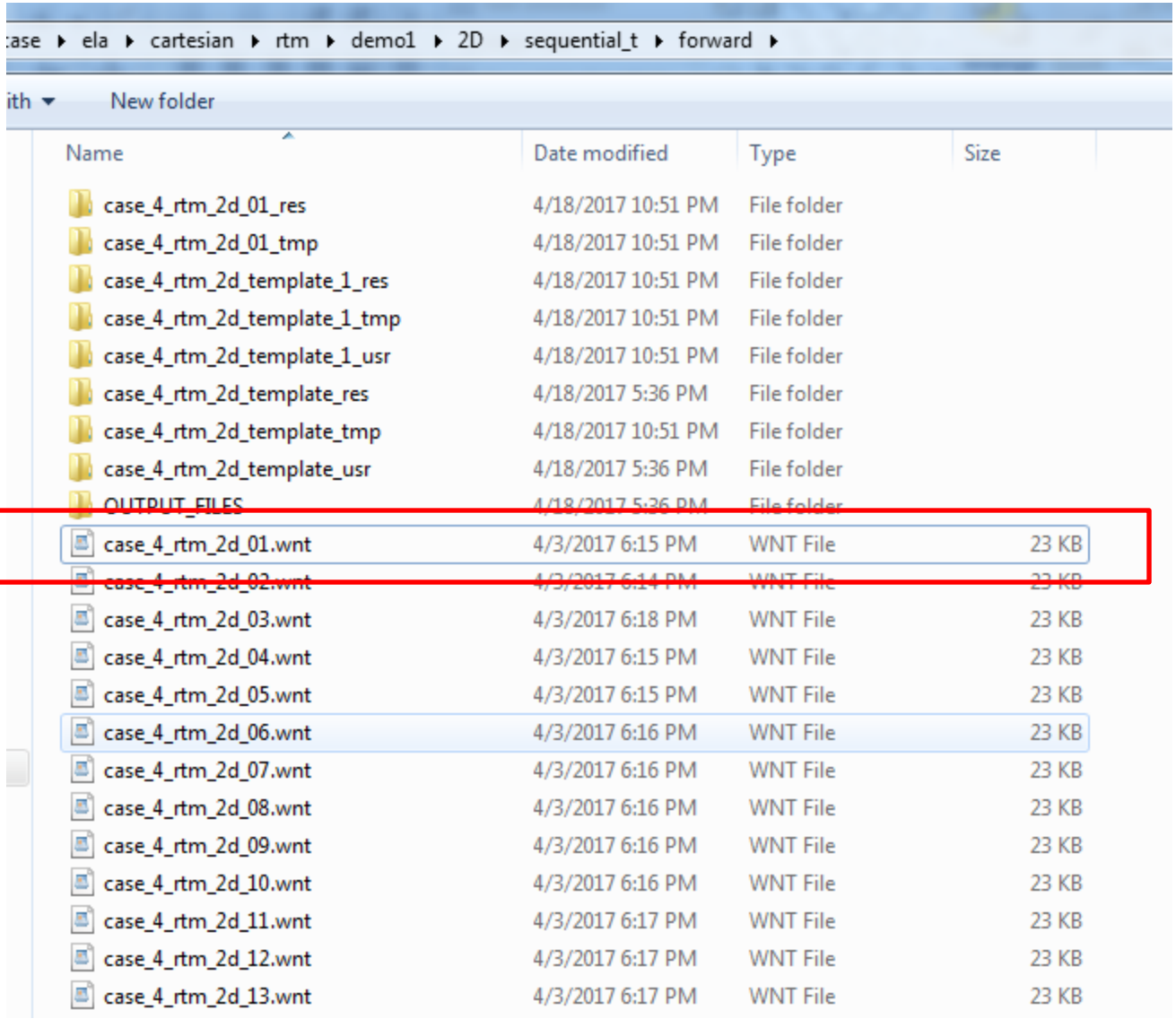
After OK, 121 receivers are created.



VII. After all are set, we can start to simulate this project to get the Vx & Vz on all sensors for source #1



This case is already build in the "xxxx\2D\sequential_t\ 13_sources \forward\ case_4_rtm_2d_01.wnt", as next page

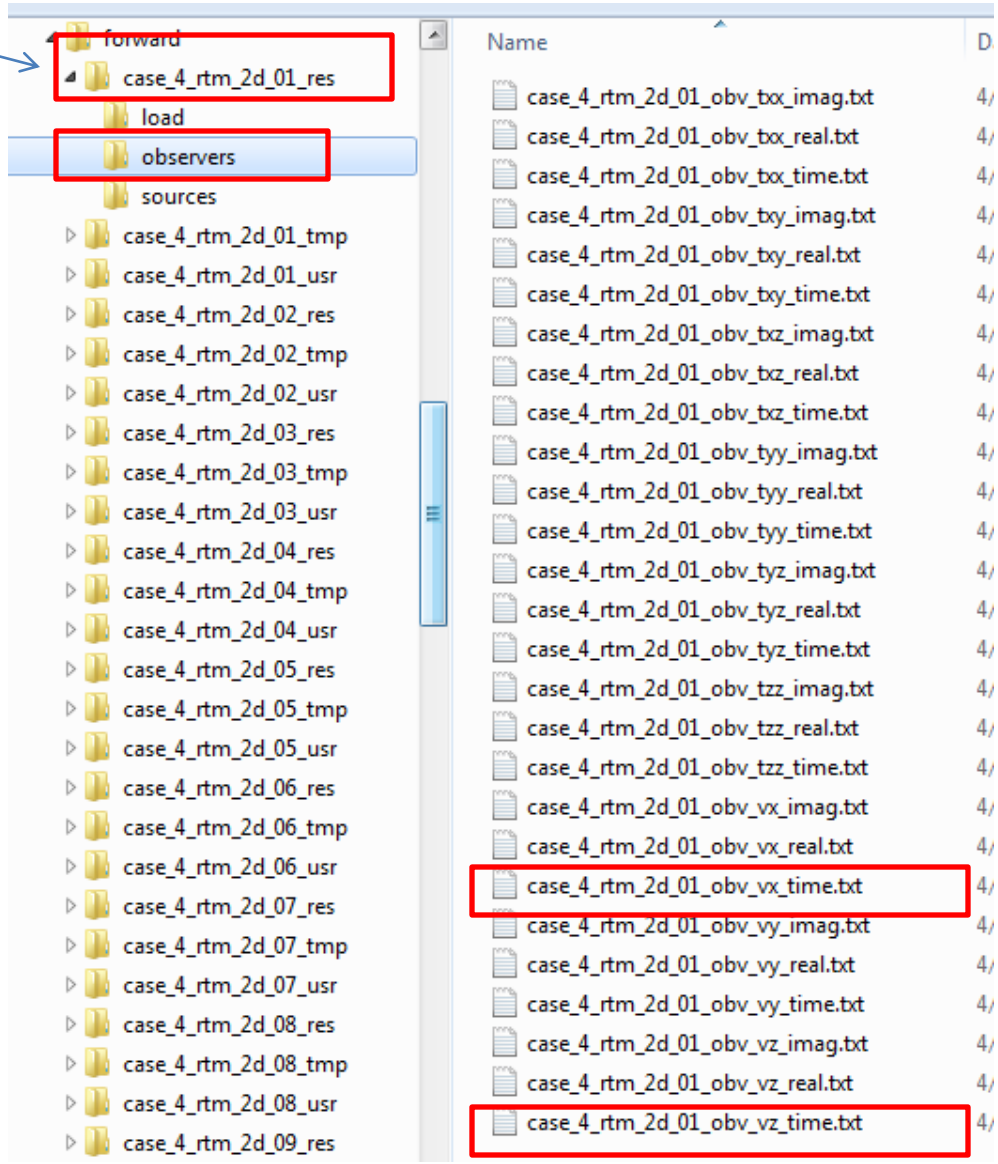

case ▸ ela ▸ cartesian ▸ rtm ▸ demo1 ▸ 2D ▸ sequential_t ▸ forward ▸
ith ▾ New folder
Name
Date modified
Type
Size
case_4_rtm_2d_01_res 4/18/2017 10:51 PM File folder
case_4_rtm_2d_01_tmp 4/18/2017 10:51 PM File folder
case_4_rtm_2d_template_1_res 4/18/2017 10:51 PM File folder
case_4_rtm_2d_template_1_tmp 4/18/2017 10:51 PM File folder
case_4_rtm_2d_template_1_usr 4/18/2017 10:51 PM File folder
case_4_rtm_2d_template_res 4/18/2017 5:36 PM File folder
case_4_rtm_2d_template_tmp 4/18/2017 10:51 PM File folder
case_4_rtm_2d_template_usr 4/18/2017 5:36 PM File folder
OUTPUT_FILES 4/18/2017 5:36 PM File folder
case_4_rtm_2d_01.wnt 4/3/2017 6:15 PM WNT File 23 KB
case_4_rtm_2d_02.wnt 4/3/2017 6:14 PM WNT File 23 KB
case_4_rtm_2d_03.wnt 4/3/2017 6:18 PM WNT File 23 KB
case_4_rtm_2d_04.wnt 4/3/2017 6:15 PM WNT File 23 KB
case_4_rtm_2d_05.wnt 4/3/2017 6:15 PM WNT File 23 KB
case_4_rtm_2d_06.wnt 4/3/2017 6:16 PM WNT File 23 KB
case_4_rtm_2d_07.wnt 4/3/2017 6:16 PM WNT File 23 KB
case_4_rtm_2d_08.wnt 4/3/2017 6:16 PM WNT File 23 KB
case_4_rtm_2d_09.wnt 4/3/2017 6:16 PM WNT File 23 KB
case_4_rtm_2d_10.wnt 4/3/2017 6:16 PM WNT File 23 KB
case_4_rtm_2d_11.wnt 4/3/2017 6:17 PM WNT File 23 KB
case_4_rtm_2d_12.wnt 4/3/2017 6:17 PM WNT File 23 KB
case_4_rtm_2d_13.wnt 4/3/2017 6:17 PM WNT File 23 KB

After the simulation finish, the transient Vx and Vz can be obtained as

Assume this simulation has name: case_4_rtm_2d_01

Simulation result folder

Result for observer

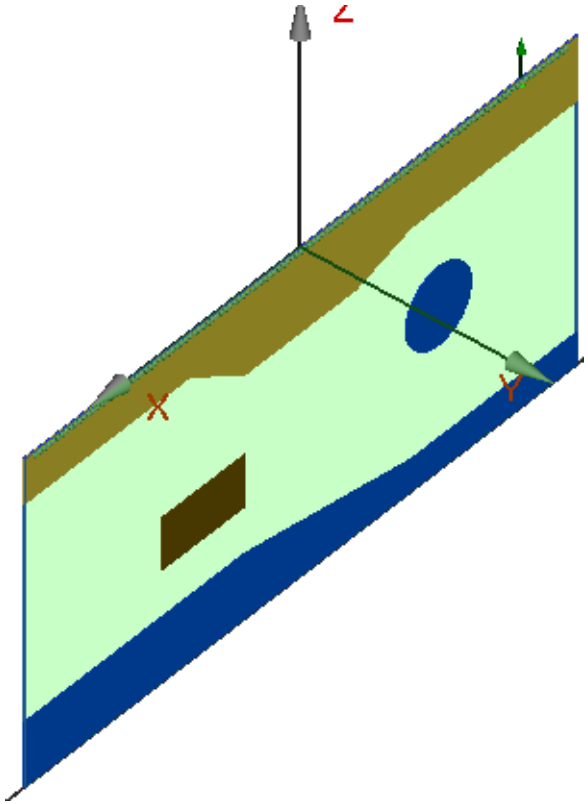


Data File format is listed in page 23.

Vx on all receivers

Vz on all receivers

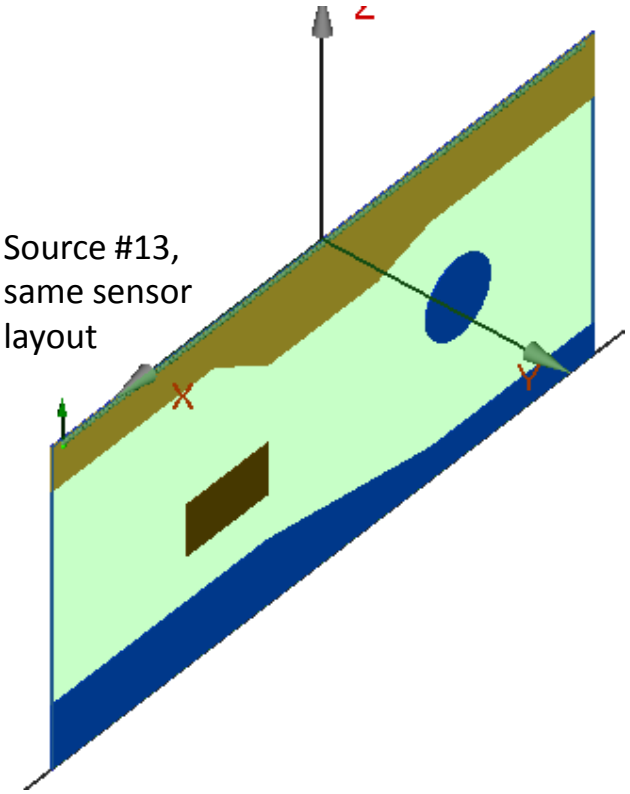
Then, setup other cases for source #2 - #13, to obtain the measurement data files. (Same environment, only source position and project name are different)



Source #2, same sensor layout

.....

Source #13, same sensor layout

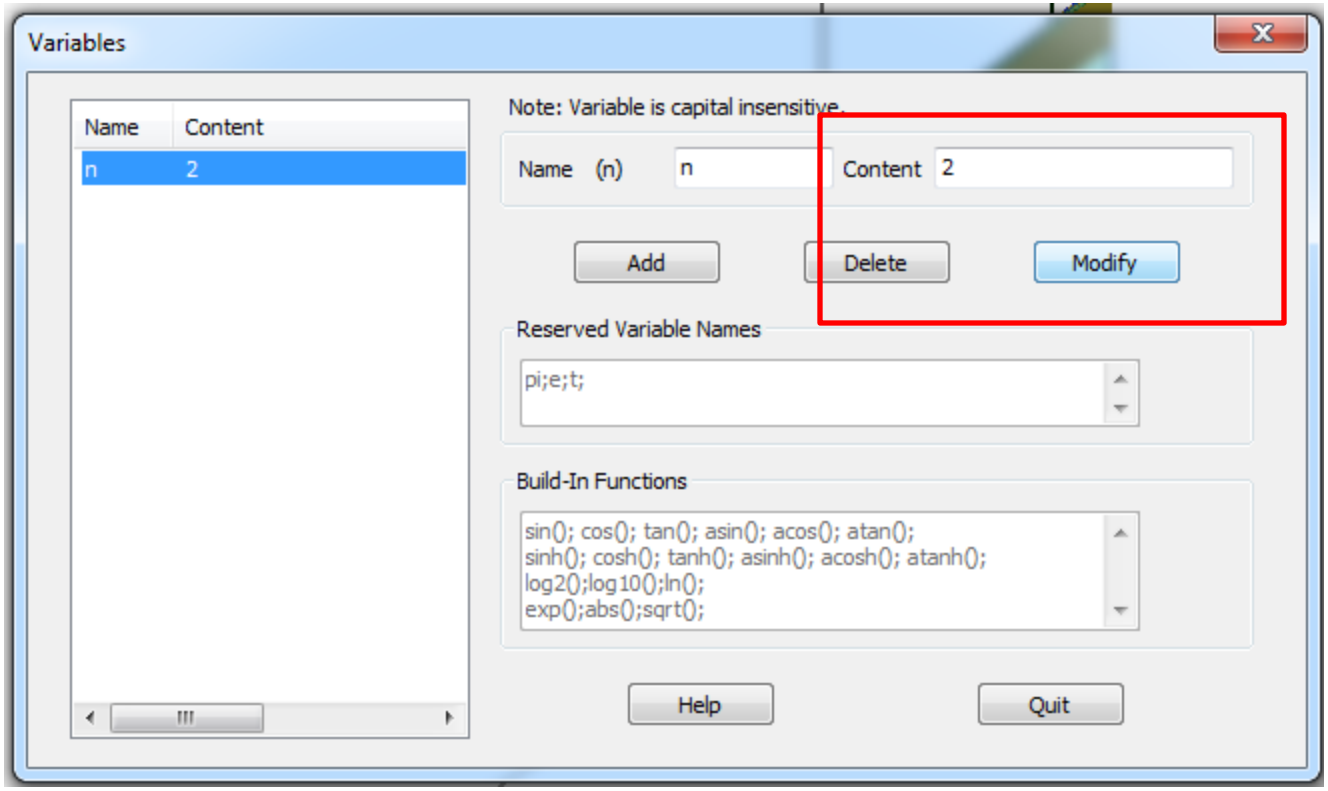


For example, we define the project name for source #XX as: case_4_rtm_2d_XX

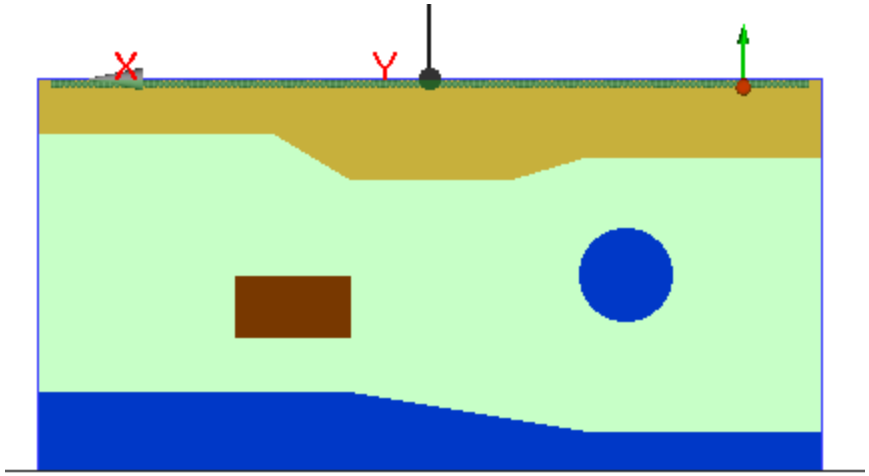
For example, for the case of #2 source.

➢ Save “case_4_rtm_2d_01.wnt” as “case_4_rtm_2d_02.wnt” in the same folder

➢ Modify the variable ***n*** to ***2***, as



Based on the source position definition (100-2500+(n-1)*400, 0, -50), the source will move to a new place as



Other parameters in the project will be kept the same

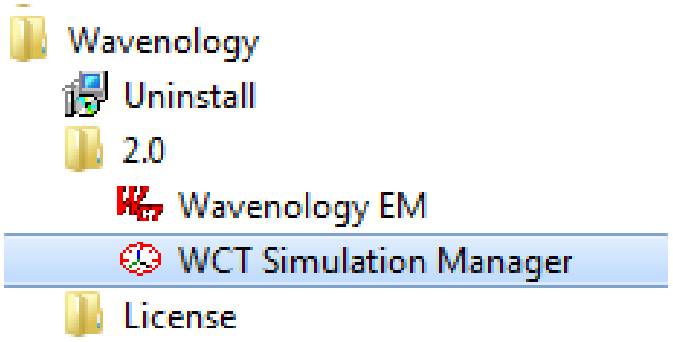
Simplify the procedure to obtain measurement data

- In the previous pages, each case is built and then run to get result. This procedure need to repeat 13 times with user monitoring

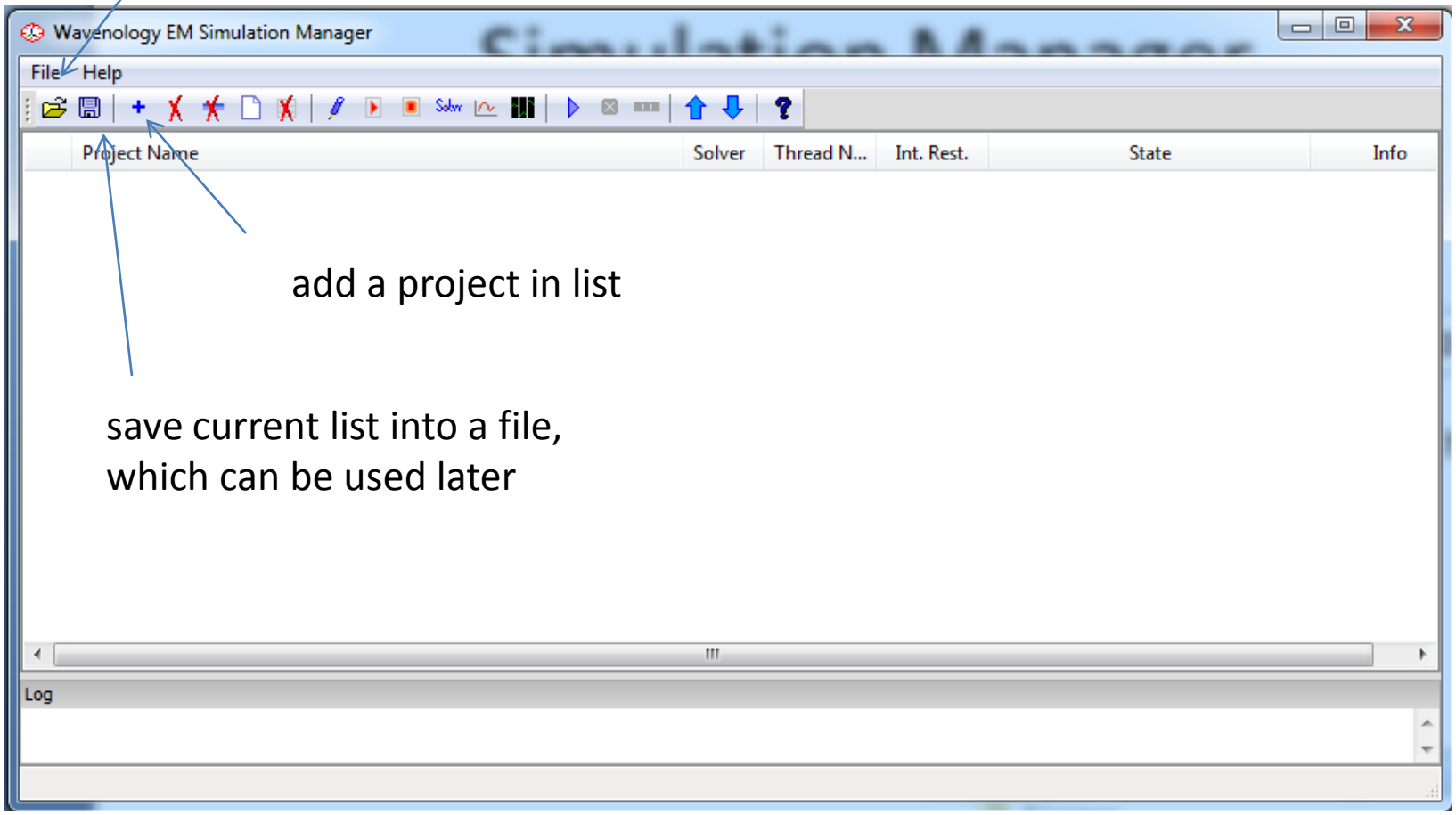
- there is a better way to simplify this procedure through ***WCT simulation manager***
  - build previous 13 cases without simulation, this procedure will be very simple, just change the value of variable ***n***, then “Save as” to a new project. Repeat 13 times
  - load these 13 cases in WCT simulation manager and let the simulation manager batch simulate these 13 cases without monitoring

In WCT GUI, use this button to start a simulation manager

Or, in Windows menu, expand here to start a simulation manager

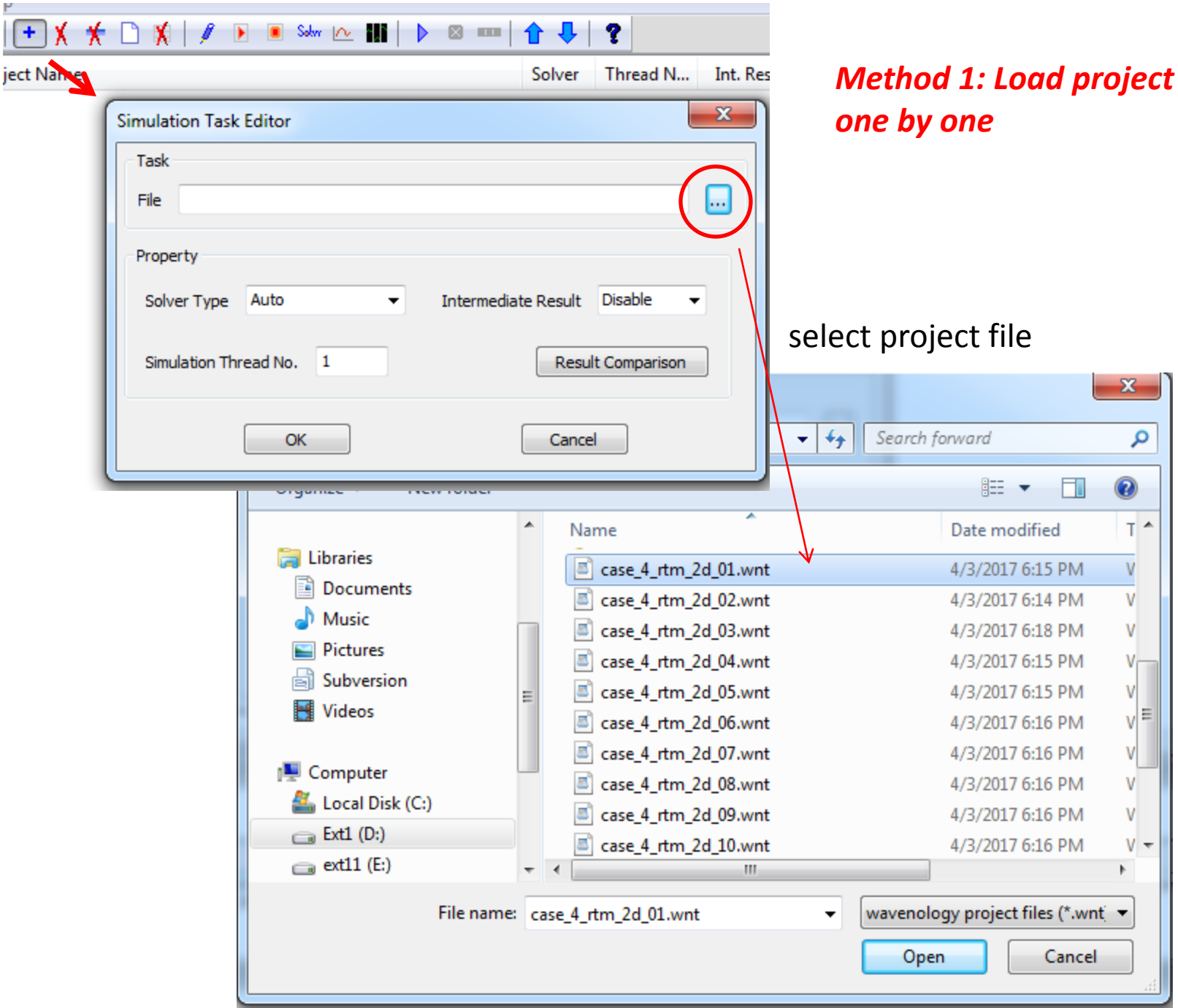


load saved list file or batch-link file



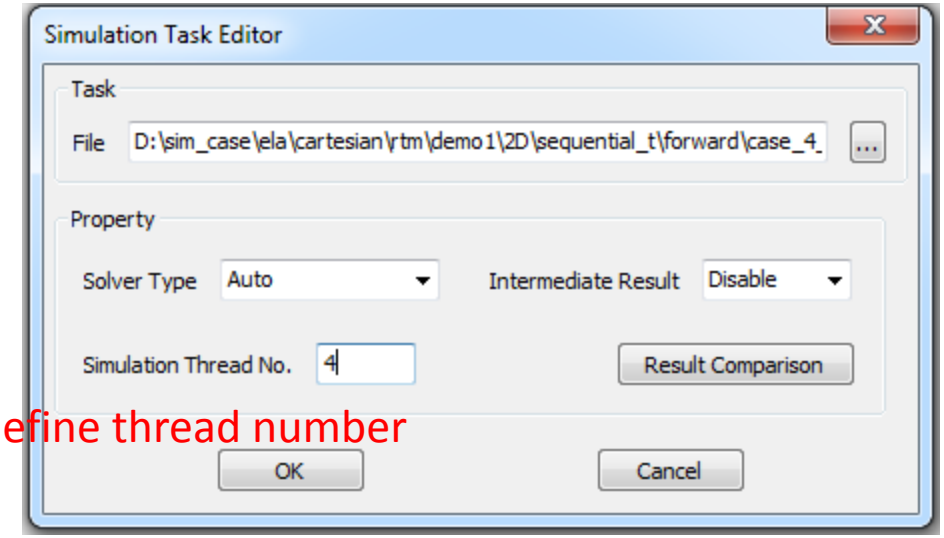
add a project in list

save current list into a file,
which can be used later



Method 1: Load project one by one

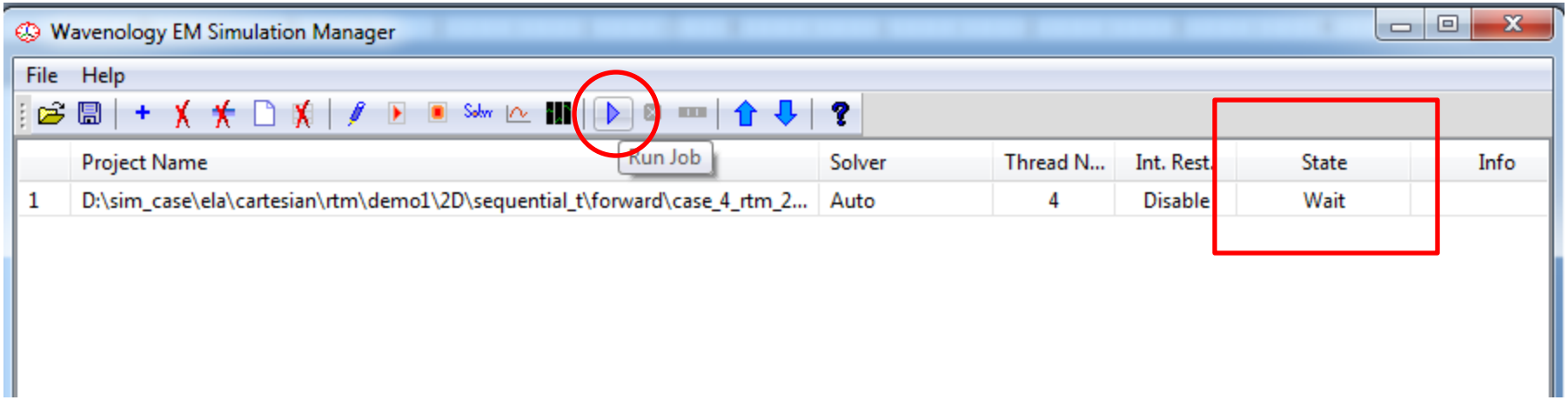
select project file



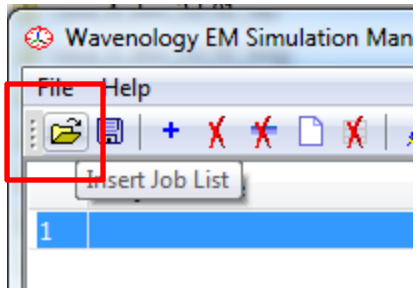
define thread number

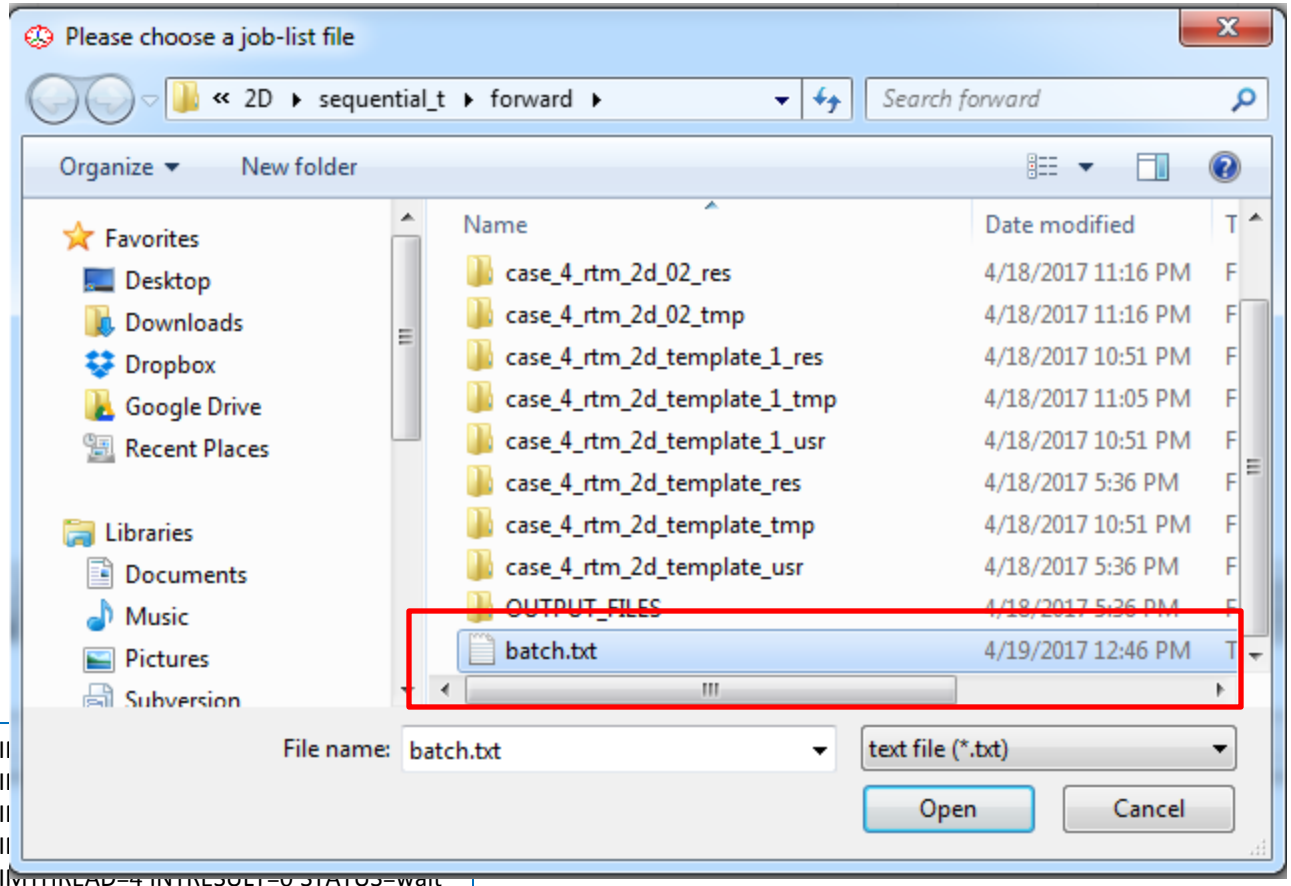
Click here to run the project in the list

The project status is listed here to shown whether it is finished or not



Method 2: Load 13 projects through a batch job file



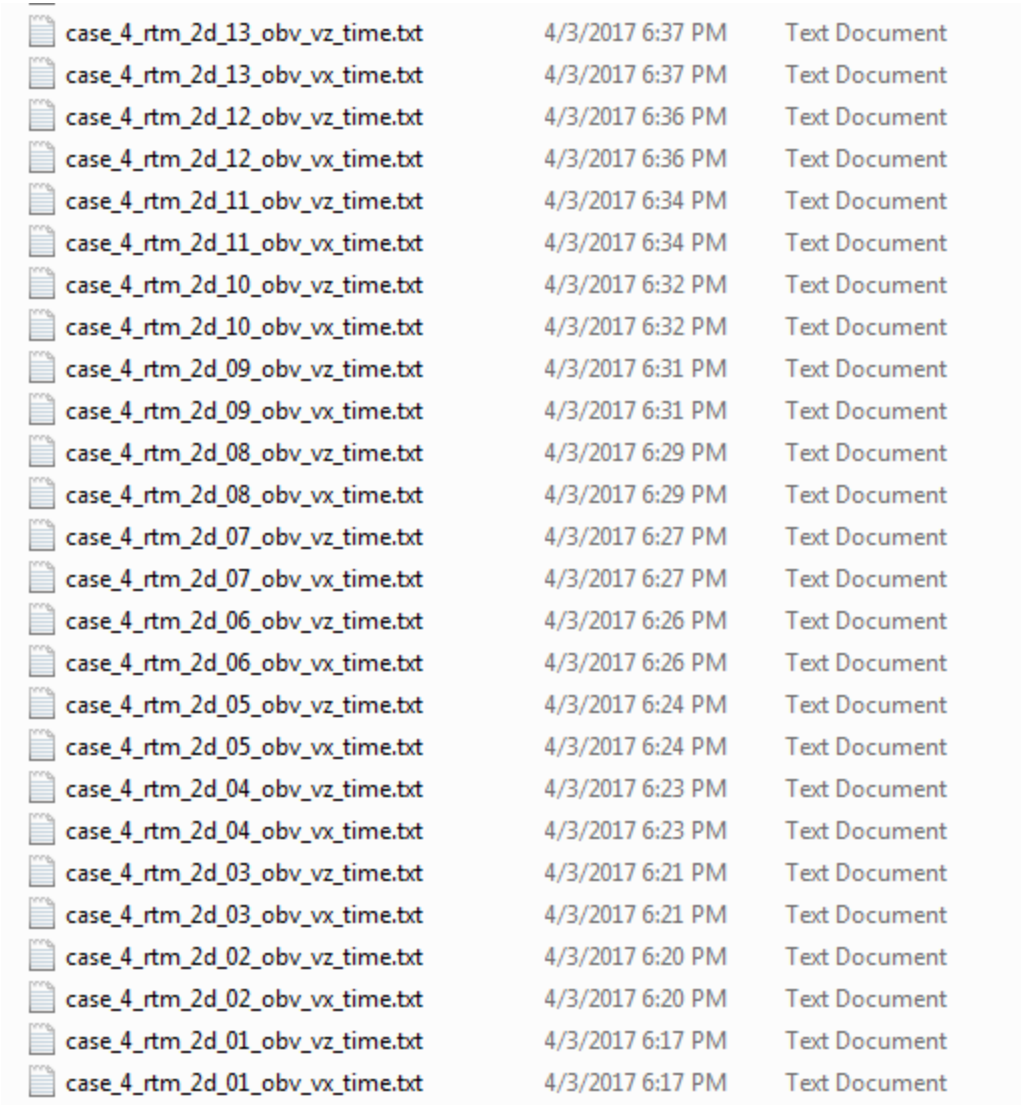


batch.txt

```
"case_4_rtm_2d_01.wnt" SIMTYPE=auto SI
"case_4_rtm_2d_02.wnt" SIMTYPE=auto SI
"case_4_rtm_2d_03.wnt" SIMTYPE=auto SI
"case_4_rtm_2d_04.wnt" SIMTYPE=auto SI
"case_4_rtm_2d_05.wnt" SIMTYPE=auto SIMTHREAD=4 INTRESULT=0 STATUS=wait
"case_4_rtm_2d_06.wnt" SIMTYPE=auto SIMTHREAD=4 INTRESULT=0 STATUS=wait
"case_4_rtm_2d_07.wnt" SIMTYPE=auto SIMTHREAD=4 INTRESULT=0 STATUS=wait
"case_4_rtm_2d_08.wnt" SIMTYPE=auto SIMTHREAD=4 INTRESULT=0 STATUS=wait
"case_4_rtm_2d_09.wnt" SIMTYPE=auto SIMTHREAD=4 INTRESULT=0 STATUS=wait
"case_4_rtm_2d_10.wnt" SIMTYPE=auto SIMTHREAD=4 INTRESULT=0 STATUS=wait
"case_4_rtm_2d_11.wnt" SIMTYPE=auto SIMTHREAD=4 INTRESULT=0 STATUS=wait
"case_4_rtm_2d_12.wnt" SIMTYPE=auto SIMTHREAD=4 INTRESULT=0 STATUS=wait
"case_4_rtm_2d_13.wnt" SIMTYPE=auto SIMTHREAD=4 INTRESULT=0 STATUS=wait
```

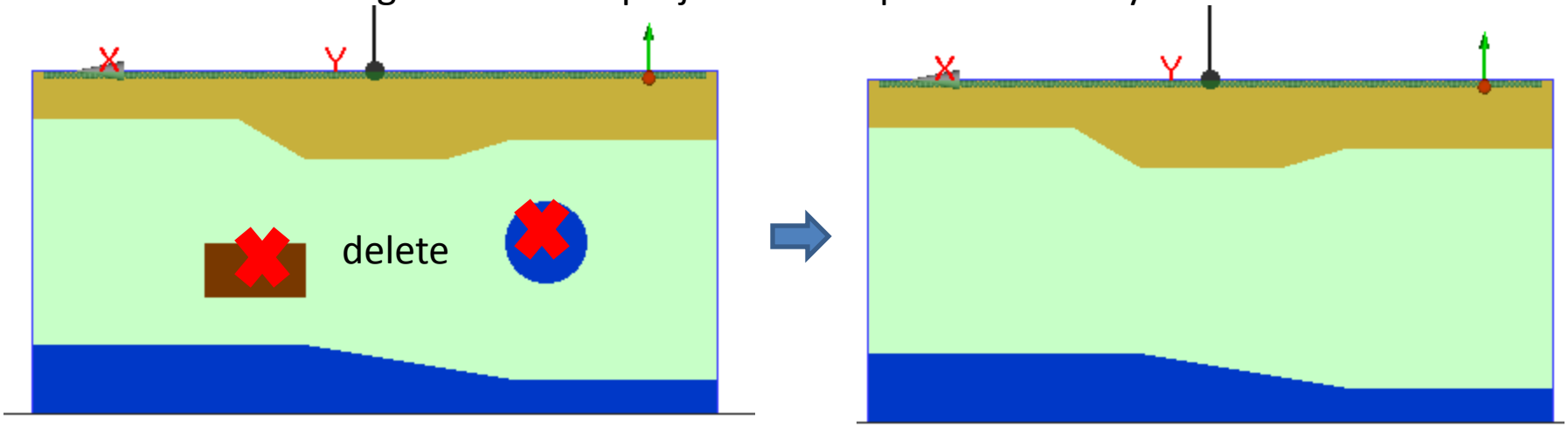
For more detail about WCT batch-link file format, please refer to WCT Cartesian EL manual

After all 13 source cases are built and simulated successfully, copy all data files to imaging project root folder as



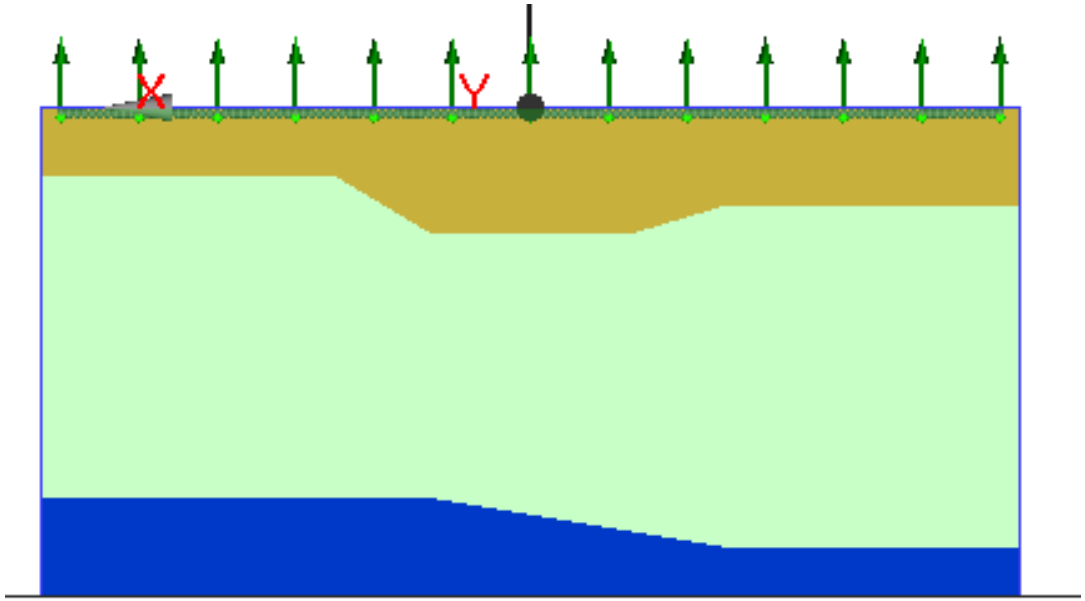
Step 2: Setup the imaging project

- We can save "case_4_rtm_2d_01" as "case_4_rtm_2d_03" to the imaging project root folder, this case_4_rtm_2d_03 will be the name of our imaging project.
  - in the demo package, this case is the "case_4_rtm_2d_03.wnt" in the ***backward*** folder.
- Delete two targets from this project to set up a known 3 layers

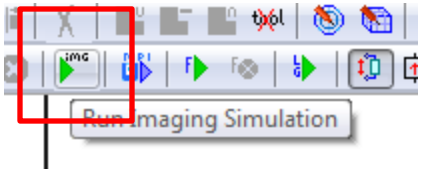


➢Then, setup all 13 sources in the project. Each source should be the same position that in "case_4_rtm_2d_01.wnt" to "case_4_rtm_2d_13.wnt".

- ➢ that is, #1 source should be same as that in "case_4_rtm_2d_01.wnt"
- ➢ that is, #2 source should be same as that in "case_4_rtm_2d_02.wnt"
- ➢ ….



- ➢ 121 receivers are the same as that in "case_4_rtm_2d_xx.wnt"
- ➢ Setup WCT EL imaging system as next page →



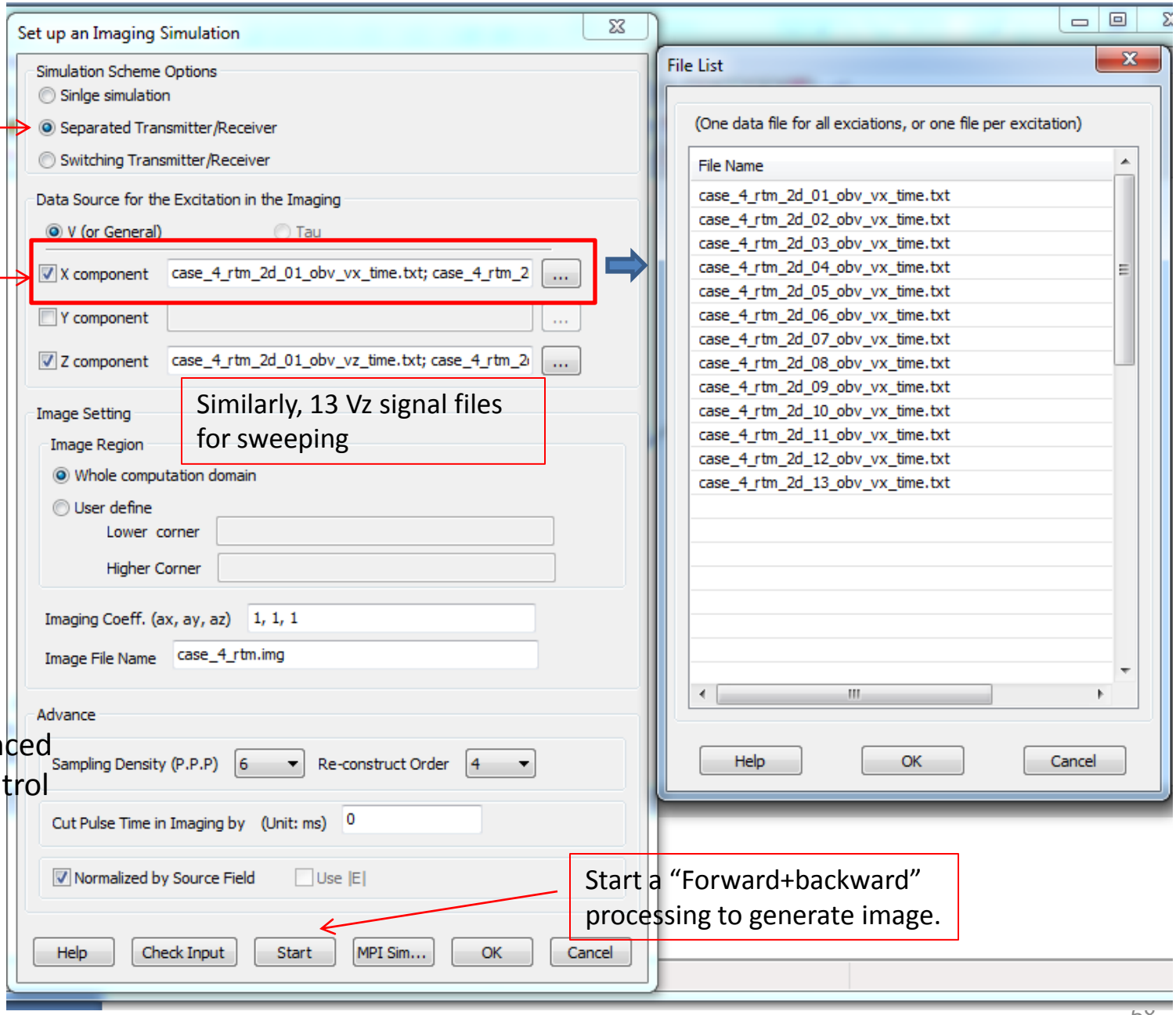
We will sweep 13 source to make the image

For each source, there are 121 Vx traces in one file, so, totally 13 Vx trace file in sweeping

Image file name & imaging region

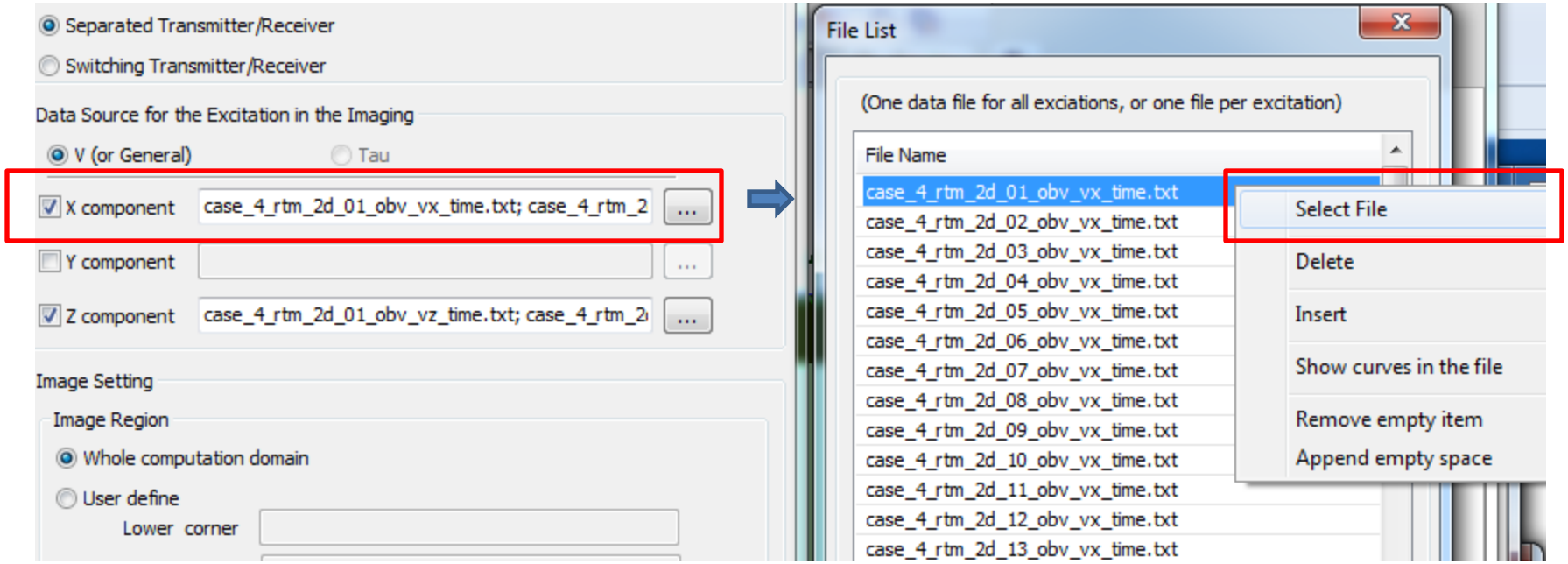
Use default Advanced parameters in control imaging.

enable normalization

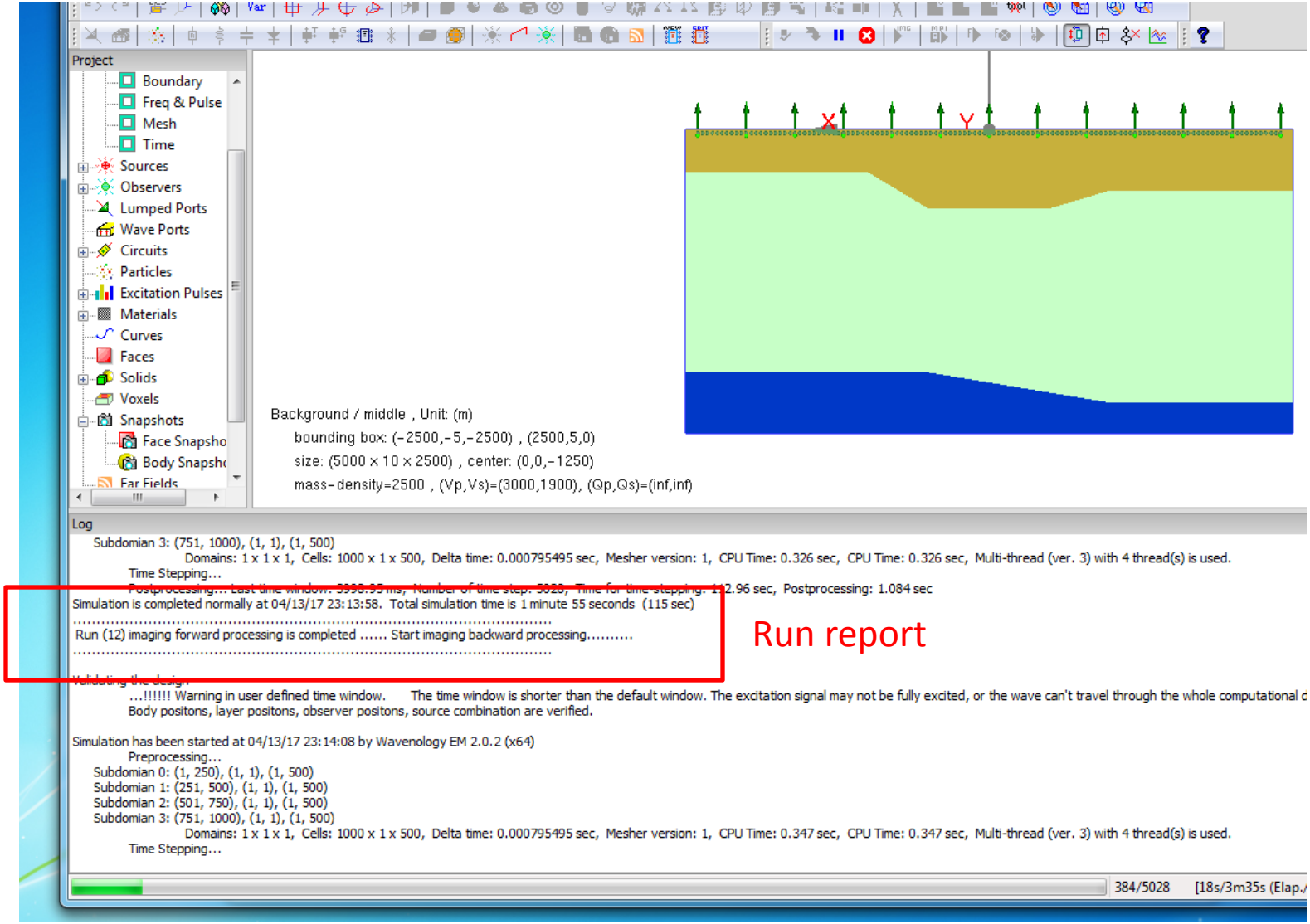


Define the data files that will be used in imaging

High-light any item, right click mouse to popup a menu

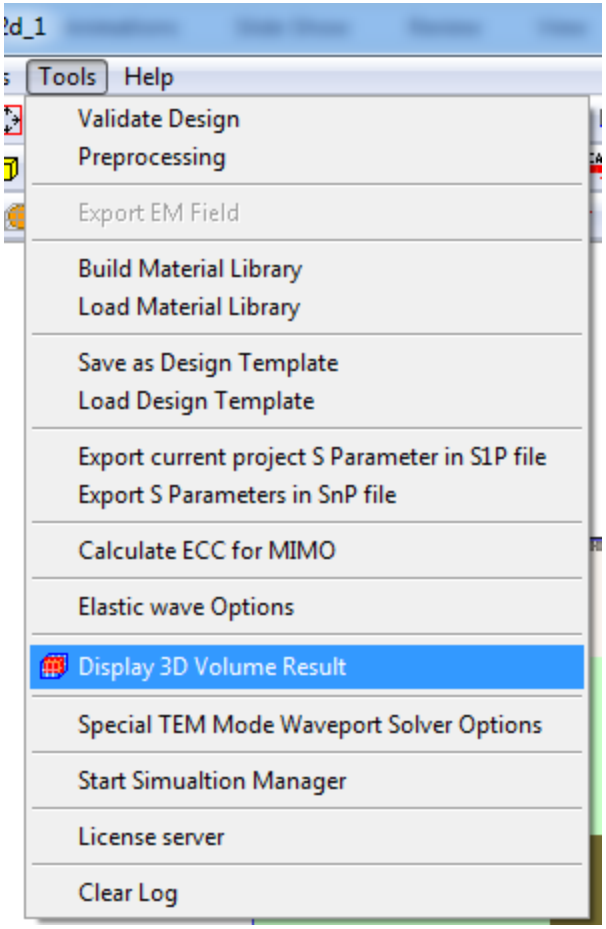


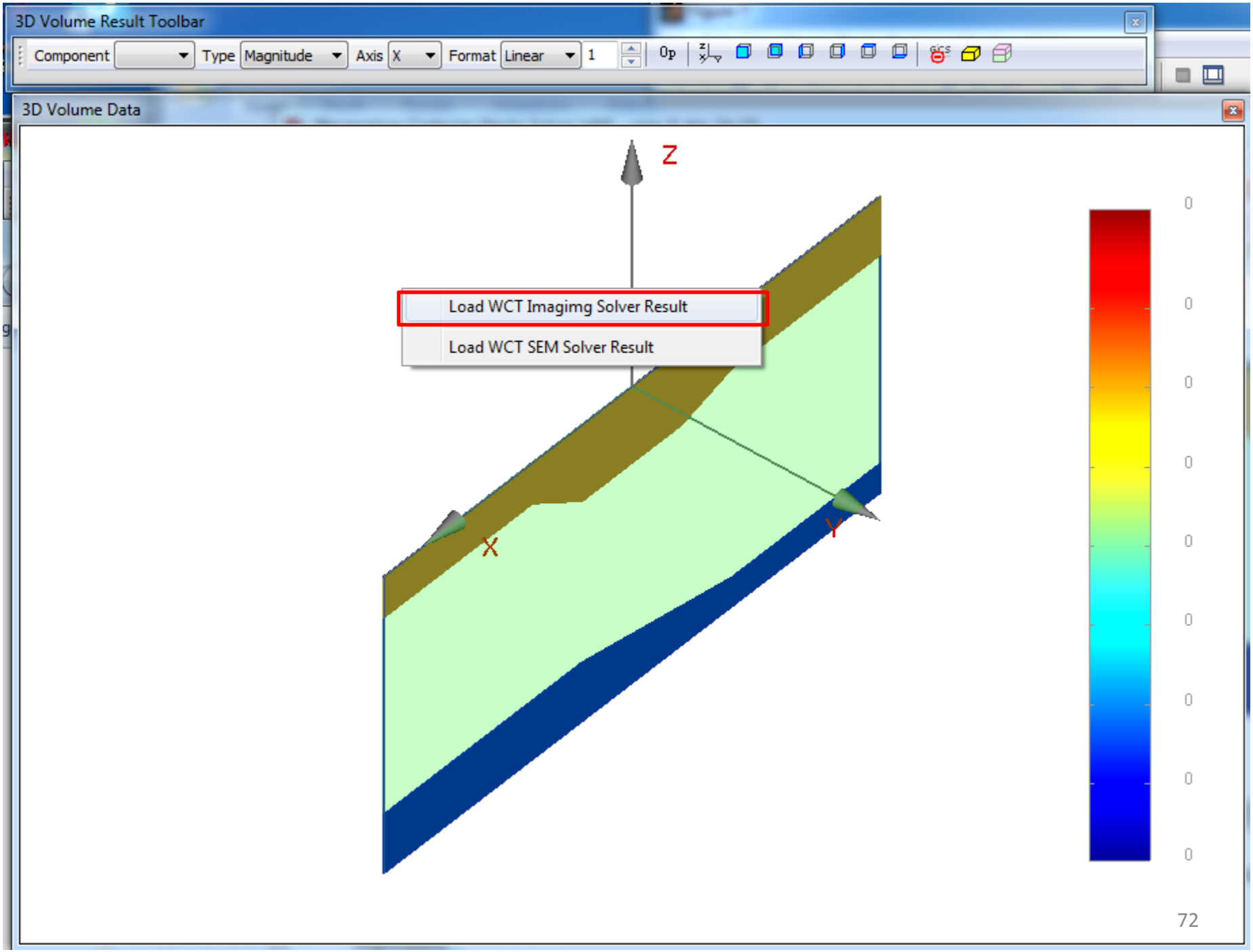
After the imaging procedure start, wait until the sweeping finish

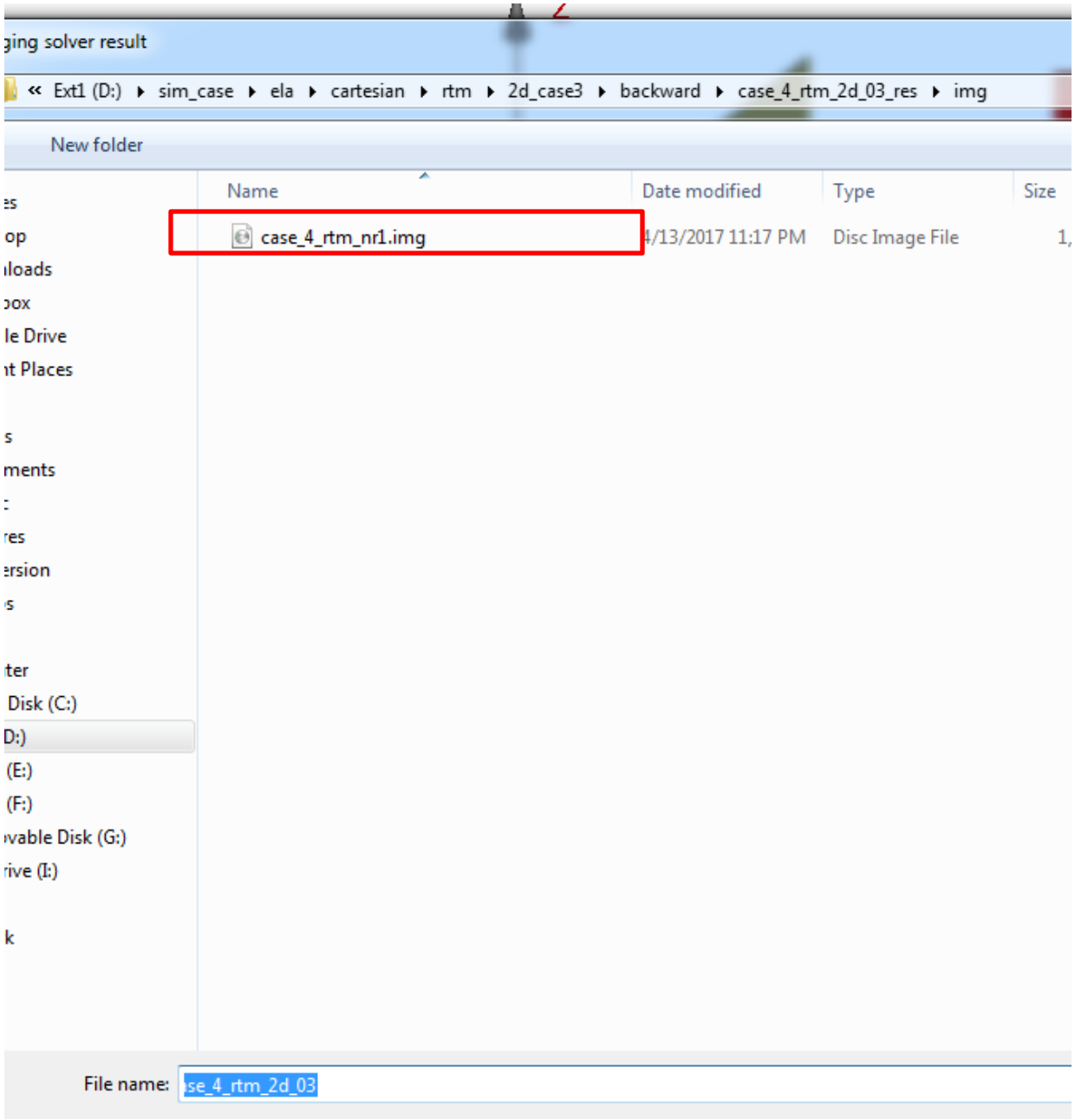


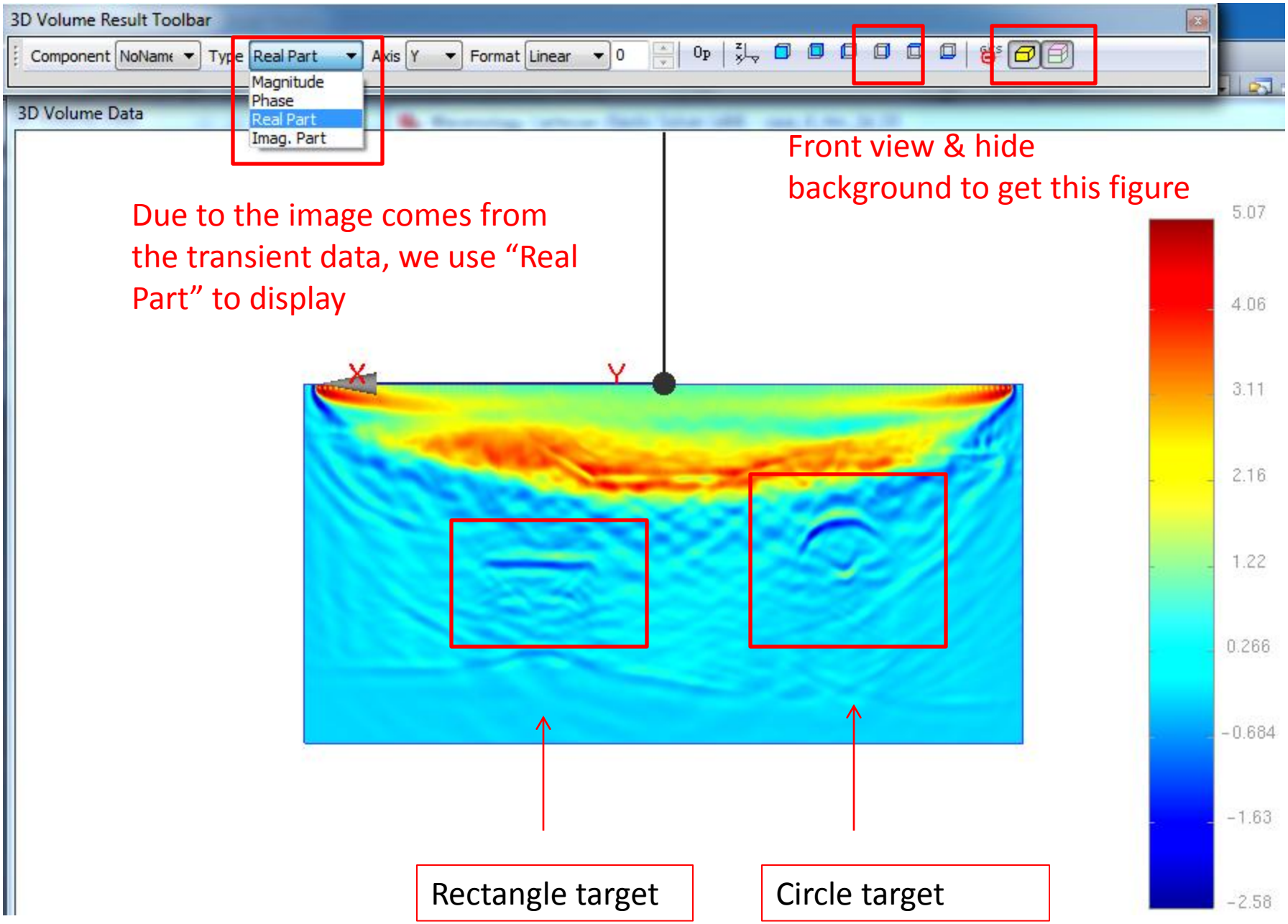
Run report

After simulation successfully, load the image

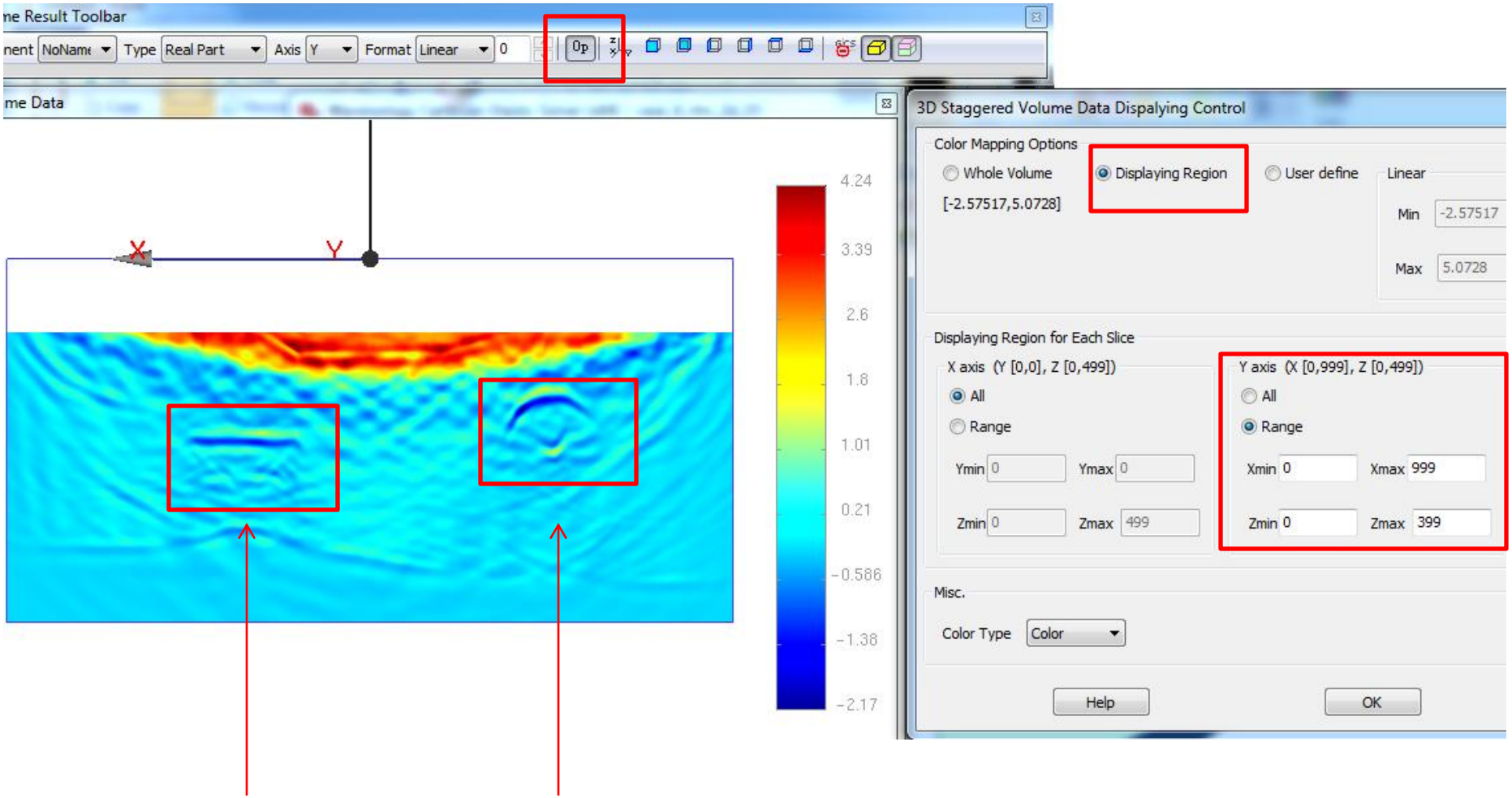



3D Volume Result Toolbar
Component
Type
Magnitude
Axis
X
Format
Linear
1
3D Volume Data
Z
Load WCT Imagimg Solver Result
Load WCT SEM Solver Result
X
Y
0
0
0
0
0
0
0
0
0
0


ging solver result
« Ext1 (D:) ▸ sim_case ▸ ela ▸ cartesian ▸ rtm ▸ 2d_case3 ▸ backward ▸ case_4_rtm_2d_03_res ▸ img
New folder
Name
Date modified
Type
Size
case_4_rtm_nr1.img
4/13/2017 11:17 PM
Disc Image File
1,
Disk (C:)
D:)
(E:)
(F:)
File name:
se_4_rtm_2d_03


3D Volume Result Toolbar
Component
NoName
Type
Real Part
Axis
Y
Format
Linear
0
Magnitude
Phase
Real Part
Imag. Part
3D Volume Data
Front view & hide background to get this figure
Due to the image comes from the transient data, we use “Real Part” to display
X
Y
5.07
4.06
3.11
2.16
1.22
0.266
-0.684
-1.63
-2.58
Rectangle target
Circle target

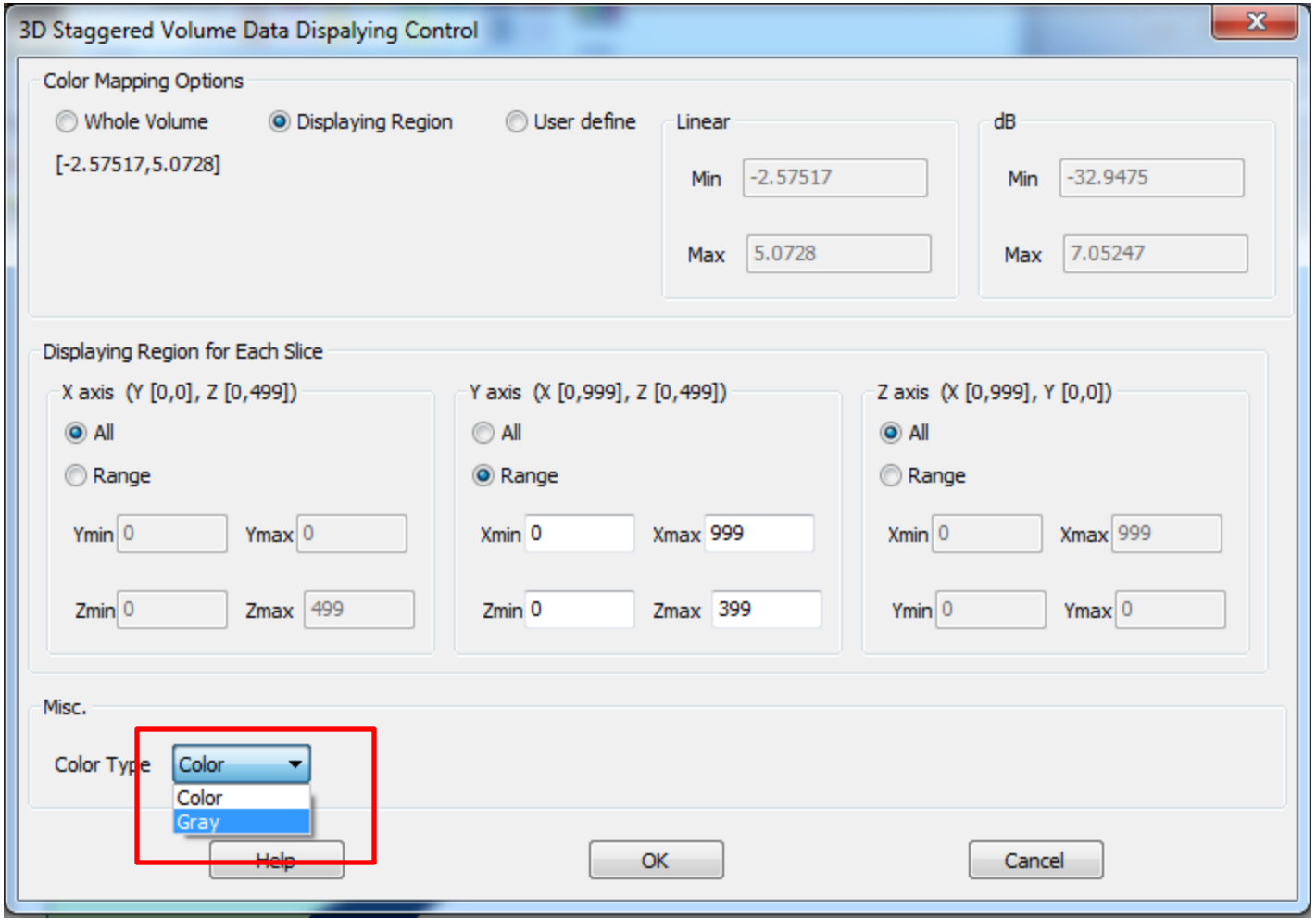
Use option to change the image displaying & color range

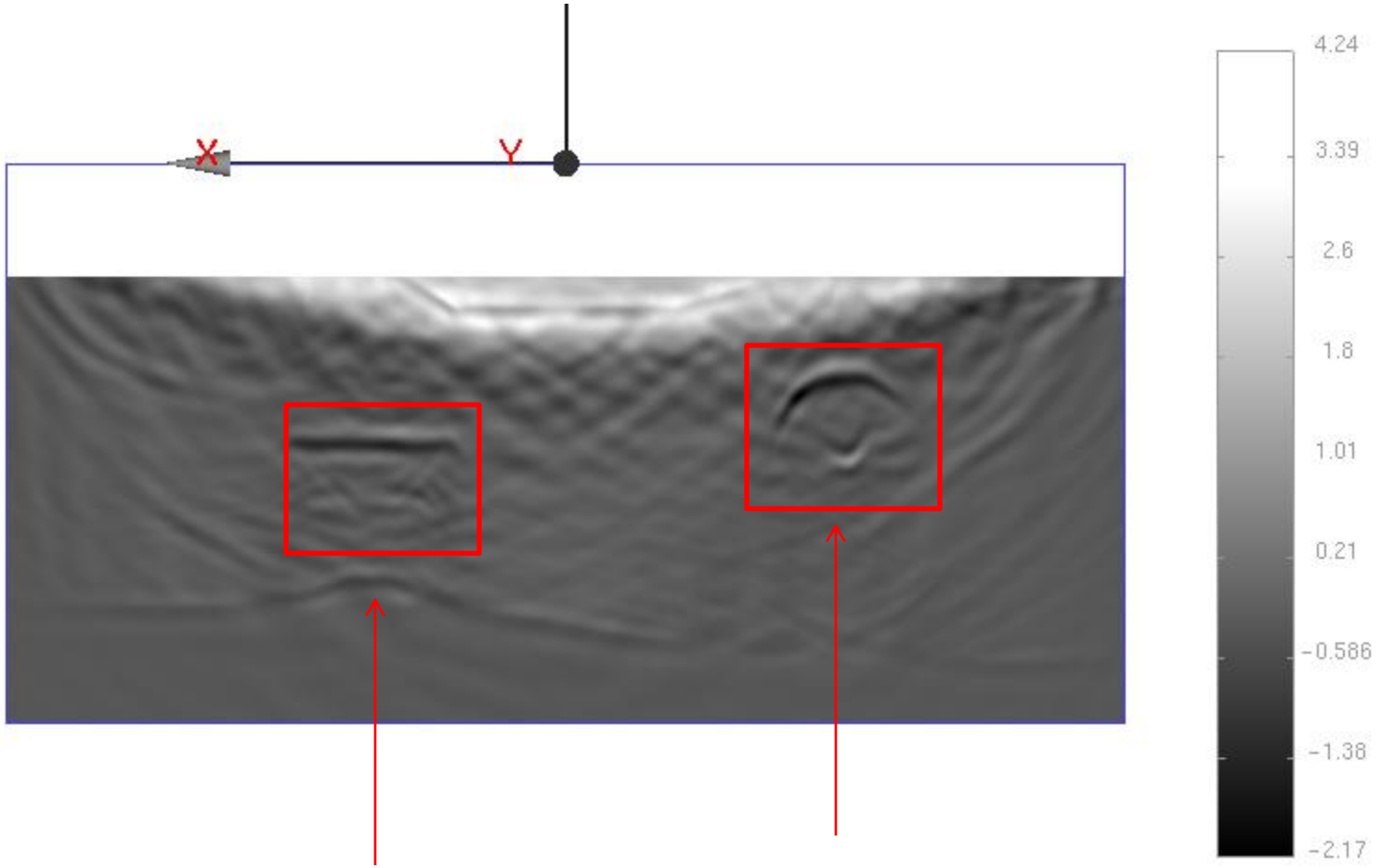


Rectangle target

Circle target

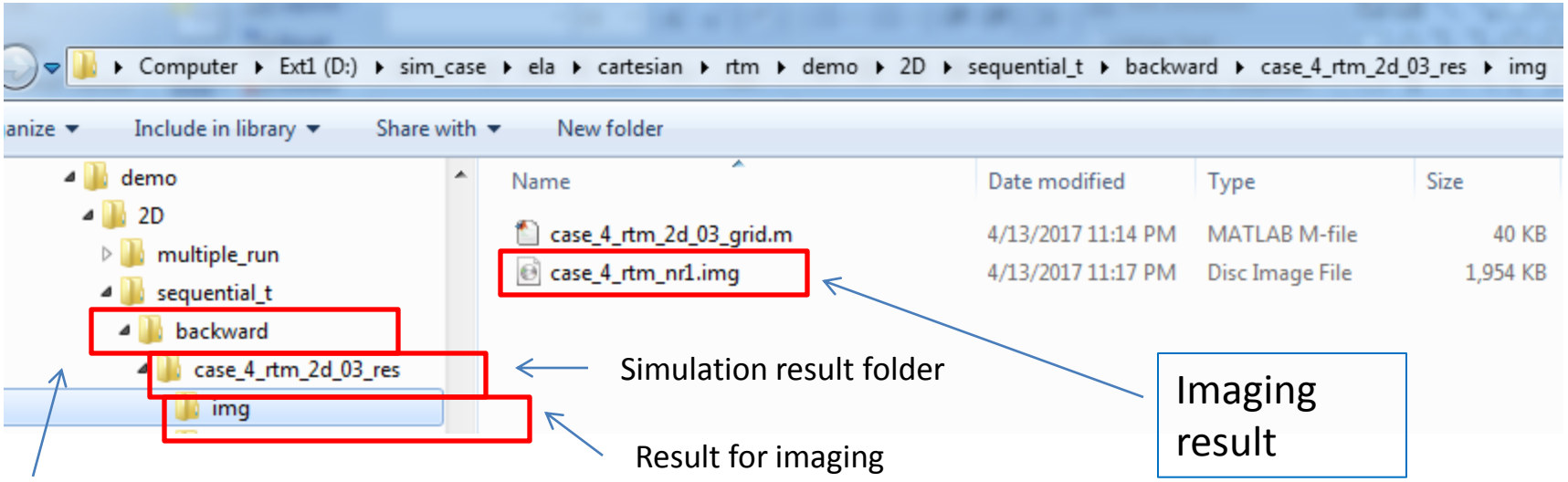
switch to Gray displaying



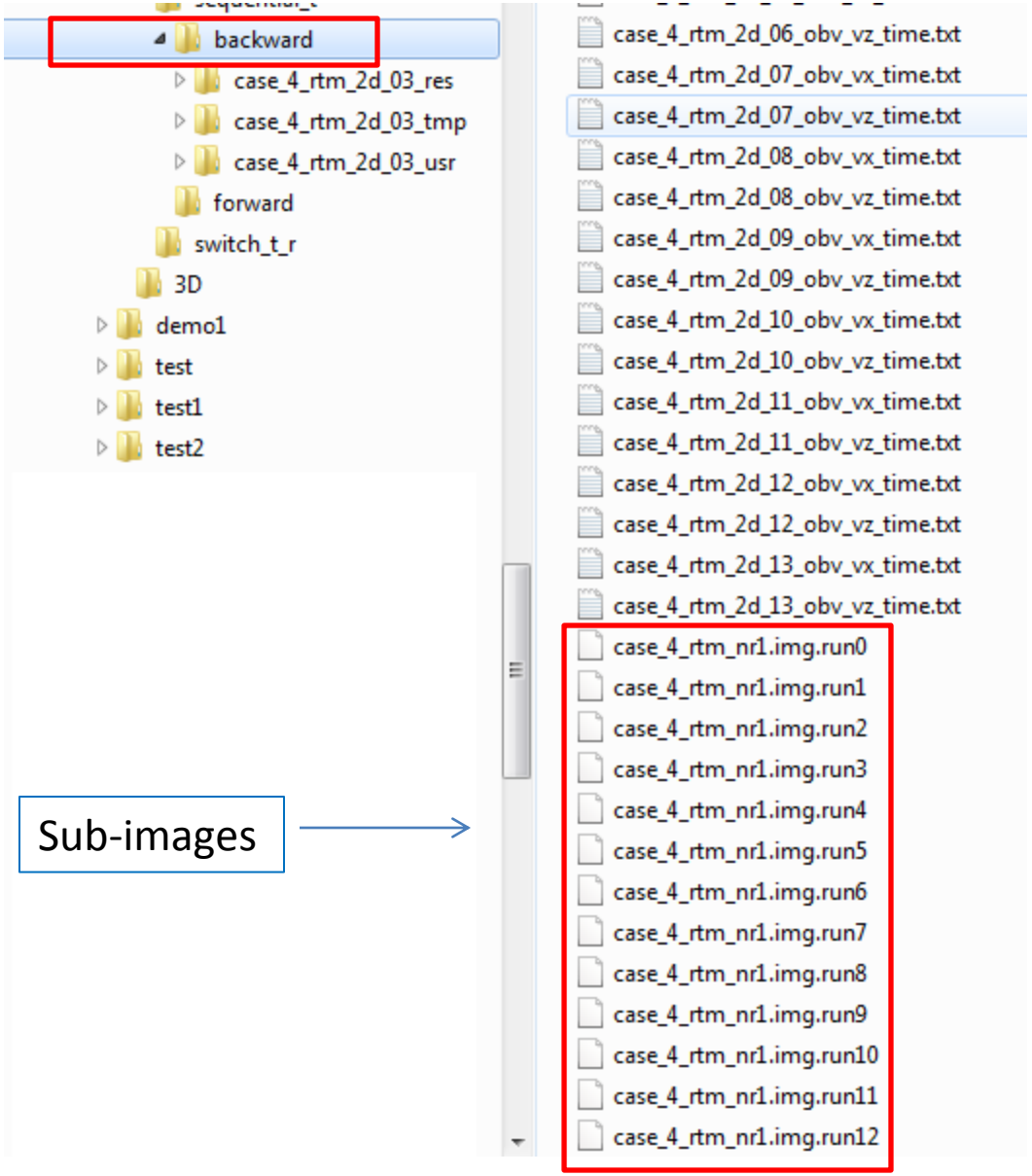

X
Y
4.24
3.39
2.6
1.8
1.01
0.21
-0.586
-1.38
-2.17
Rectangle target
Circle target

Simulation results

- A final image data file is placed in:
project_res\img***image_file_name***
- N sub-image data file is place in project root folder as:
  image_file_name.run#, here # is source index in sweep
    - each sub-image is the image created by each source, the final image is the sum of all sub-images
- Following is the example of image files



Project root folder →



Sub-images →

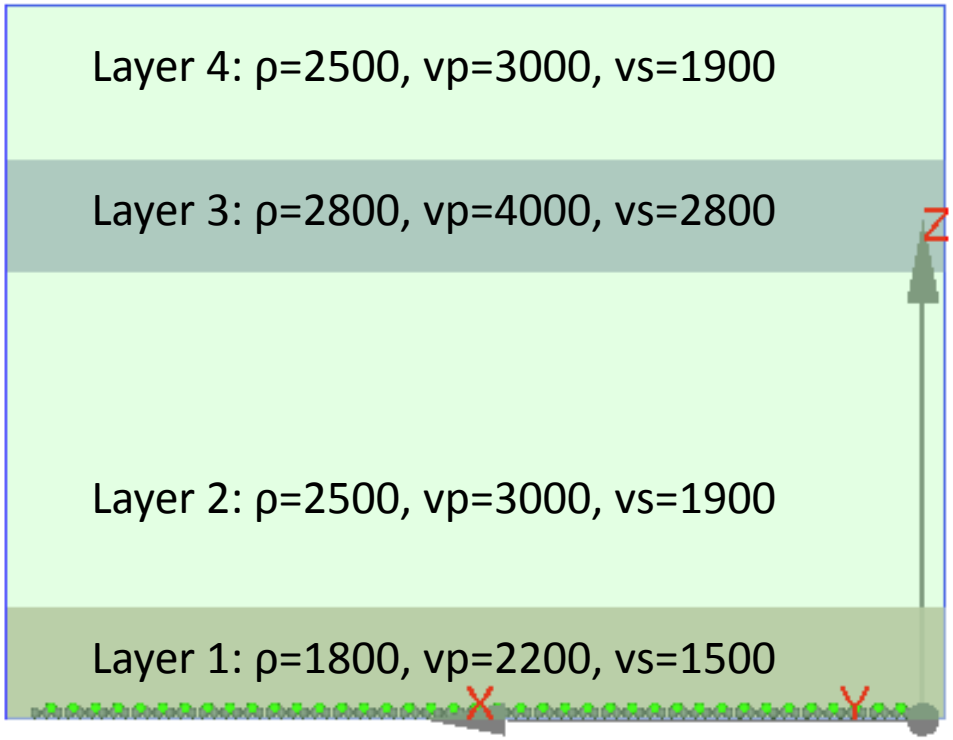
Case II : 2D Imaging

Imaging a black box with Ricker Wave Source Pulse (f_{max}=27 Hz)

Here, we will demonstrate how to use WCT Cartesian EL imaging method to imaging an almost black box environment to check whether we can use limited information to reconstruct the real space

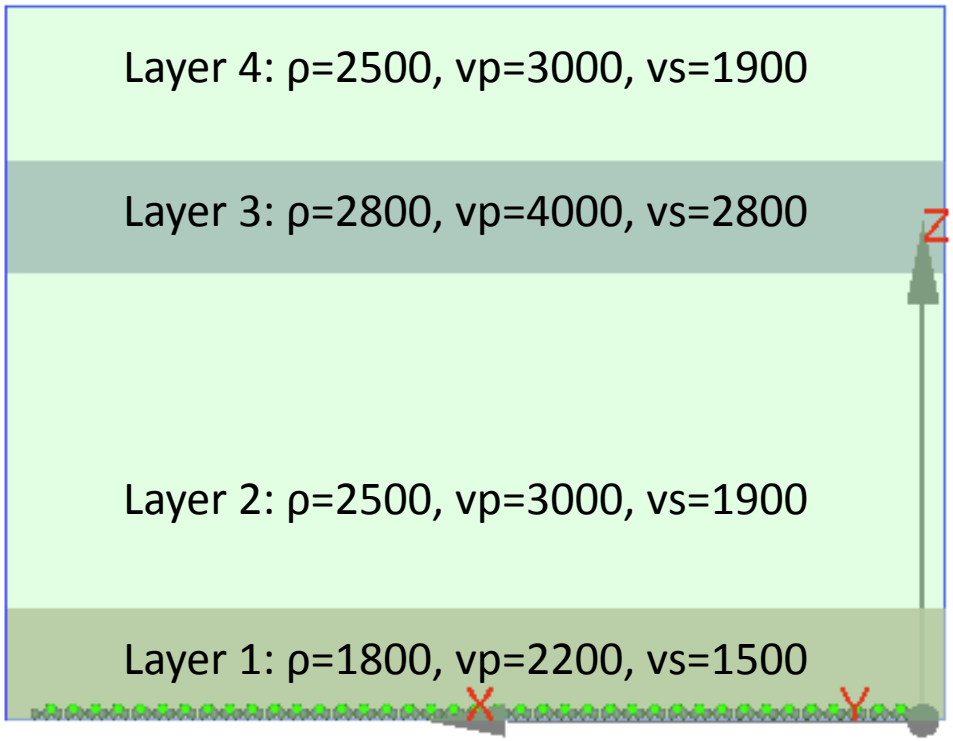
What we know is there are estimated 4 layers in the space, but we don't know the exact shape of each layer and the position.

We just have a initial guess as right figure.

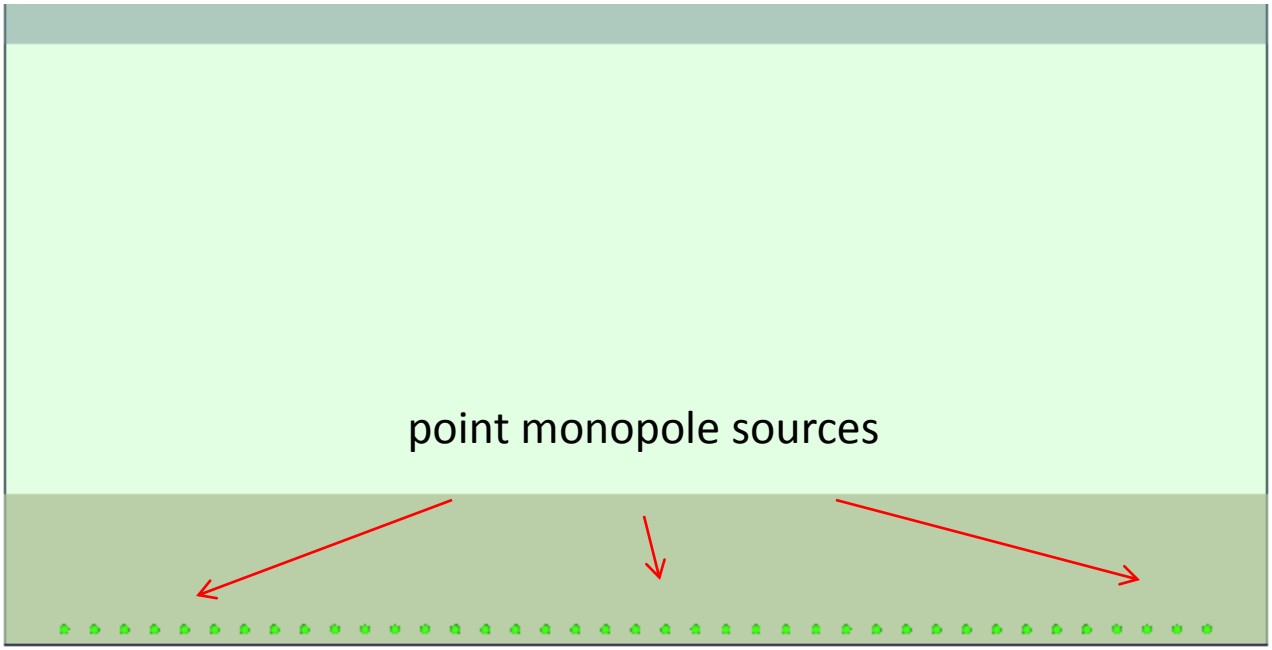


The Known Before Imaging

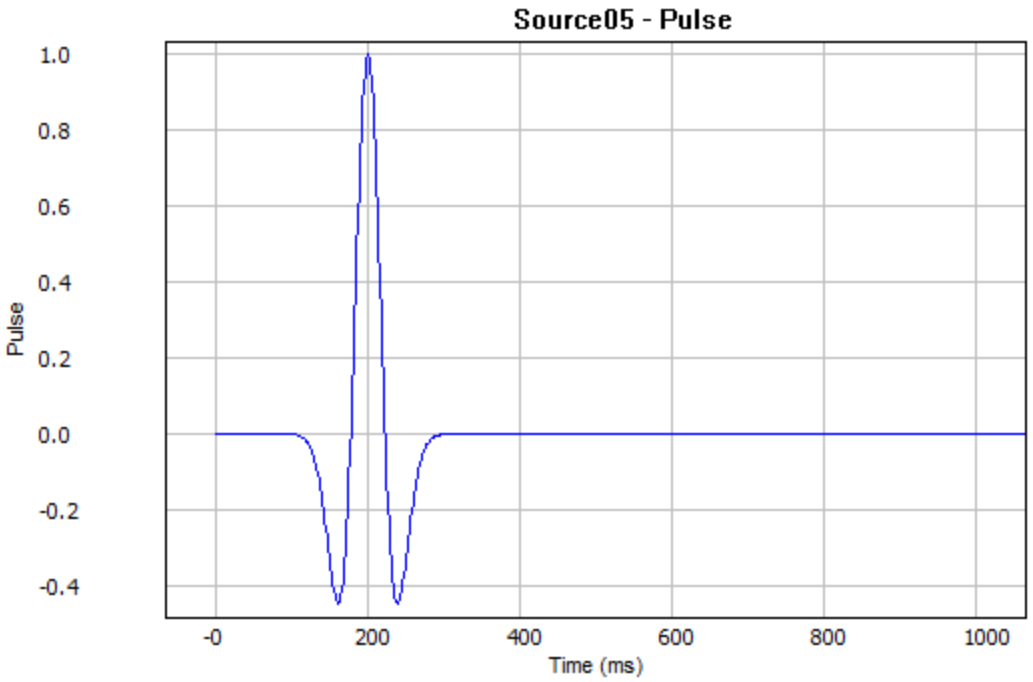
1. an estimated 4 layers in the space, each layers material is known



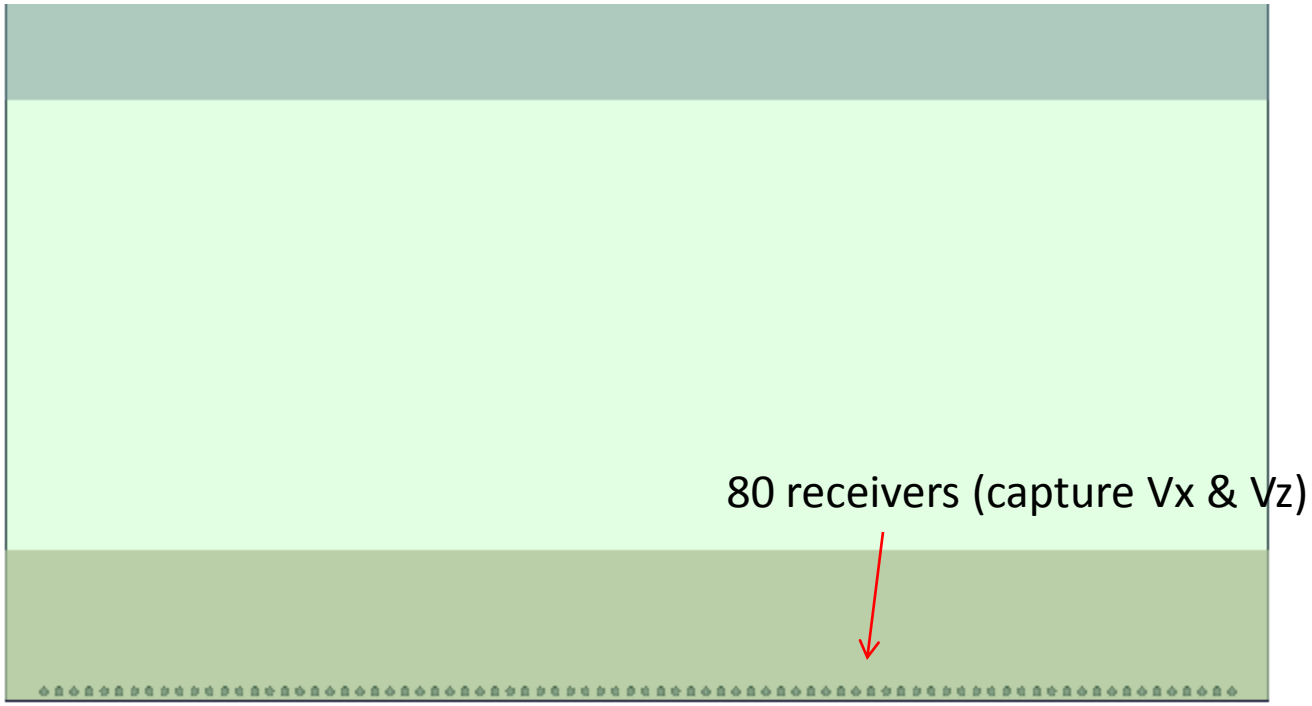
2. the position and the pulse of 39 point dipole sources, the source pulse is known as the Ricker wave at f_{max}=27 Hz



The Ricker Wave with fmax=27 Hz

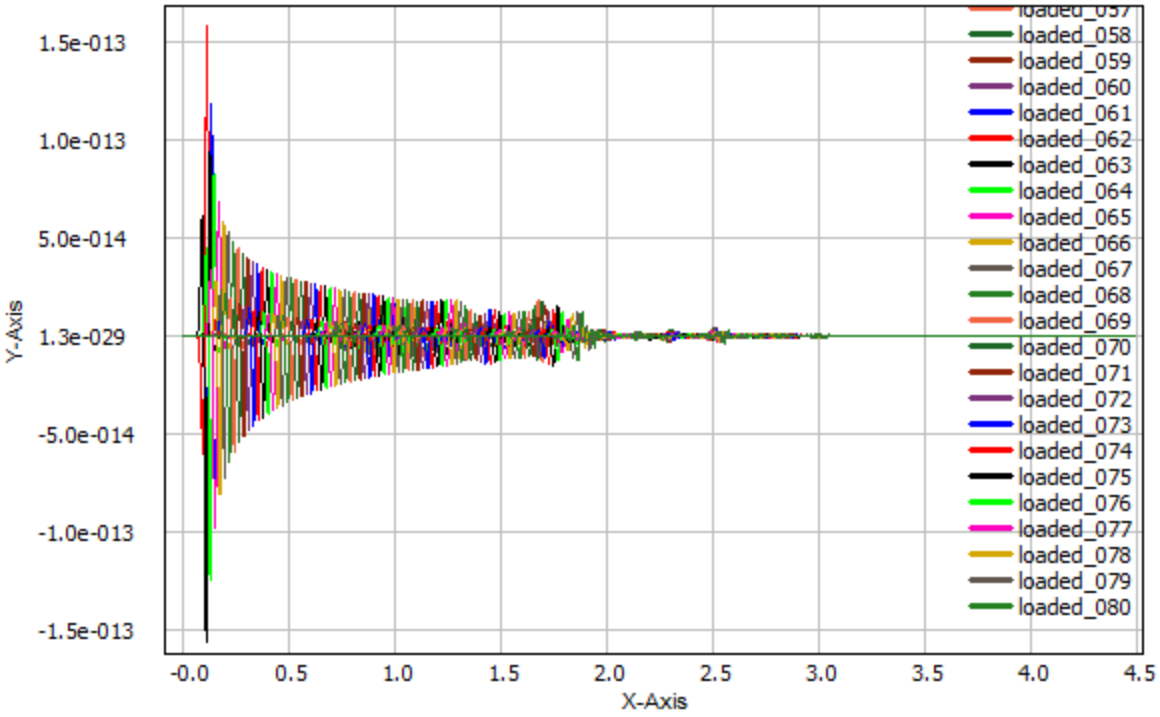


3. the position of 80 sensors to record Vx & Vz field



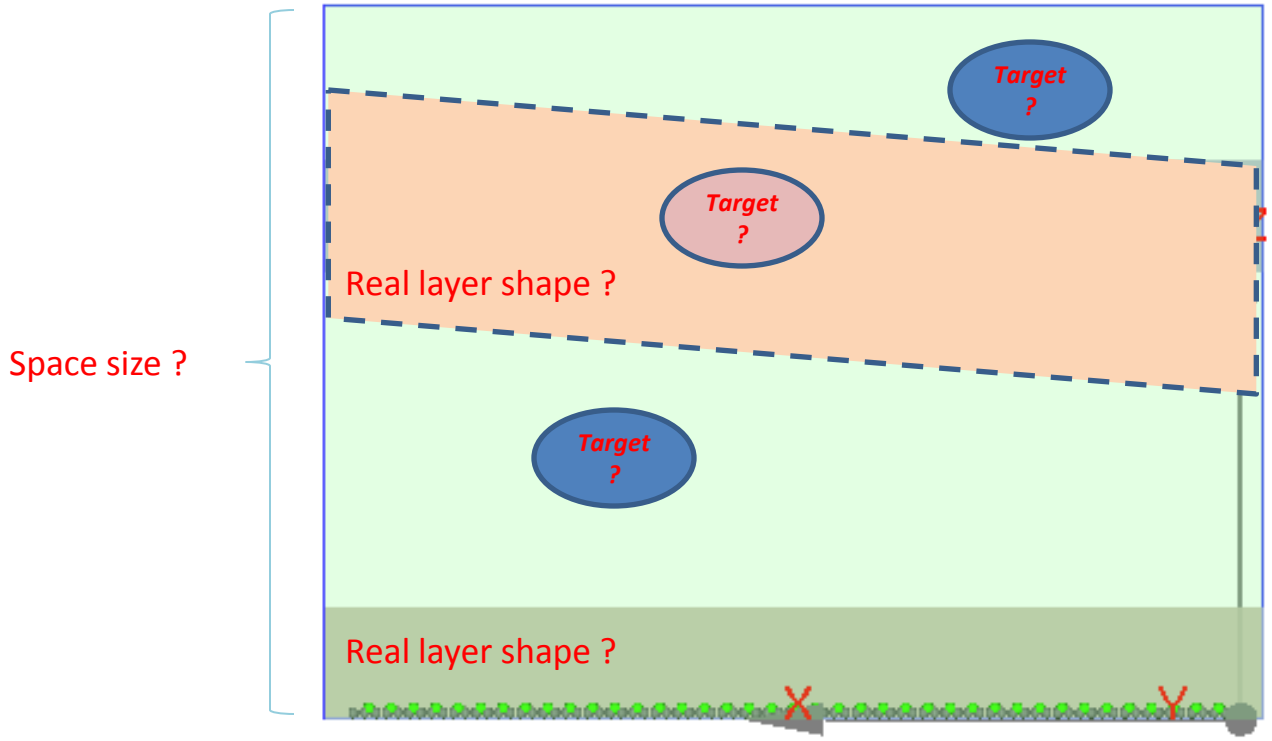
4. the detected signal Vx & Vz (measurement data) on these 80 sensors for each source (which can be obtained from the “Extension part of Case II”)

The Vx signal on all 80 sensors from source #1



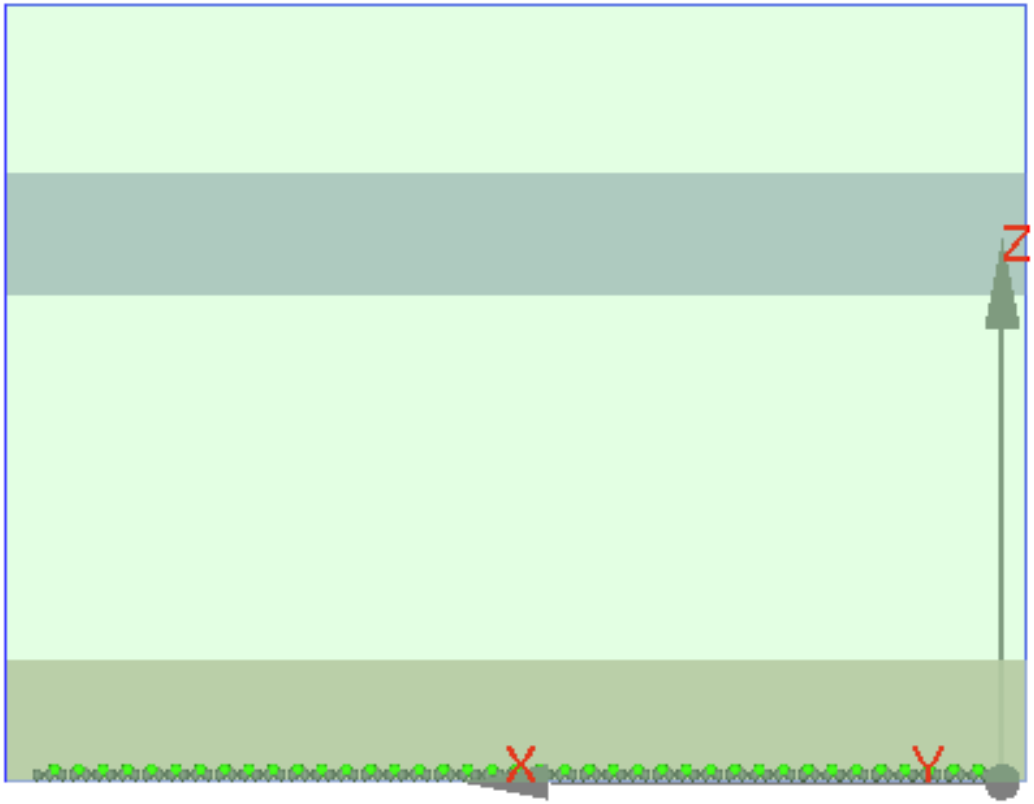
The Unknown in the System Before Imaging

- ***we don't know what is the exact shape of each layer***
- ***we don't know where the detected Vx & Vz signal come from, which means even the signal comes from simulations, we don't know the original setting, for example, space size, simulation mesh, Δt, etc.***
- ***whether there is objects in the real space***



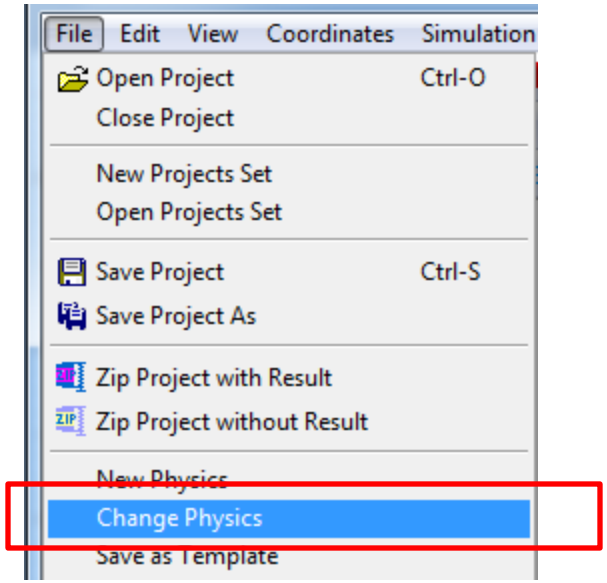
Goal of this Case

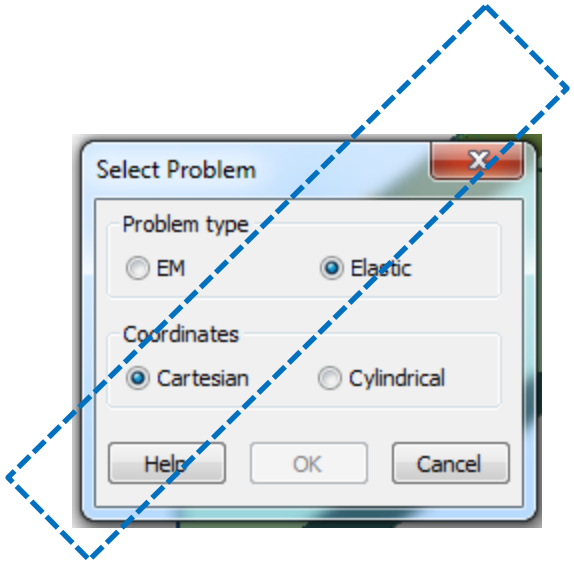
- We will
  - sweep 39 sources in a guessed background with detected signal, as following,
    - to reconstruct the space



I. Setup WCT Cartesian EL project

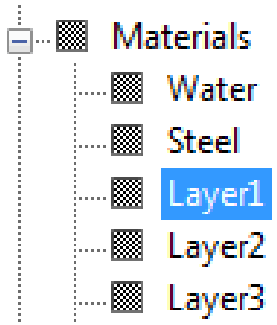
Note: This case is already build in the demo package as "xxxx\2D\sequential_t\ 39_sources_80recv_small_obj_v3 \Imaging\ Imaging_Layered_2D.wnt"

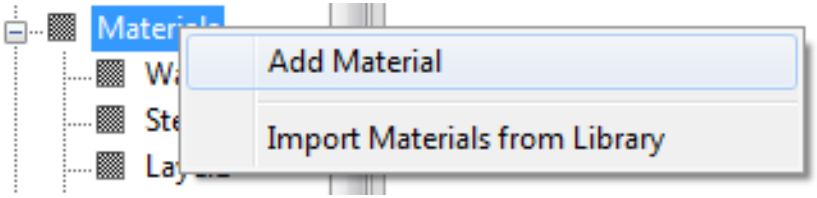


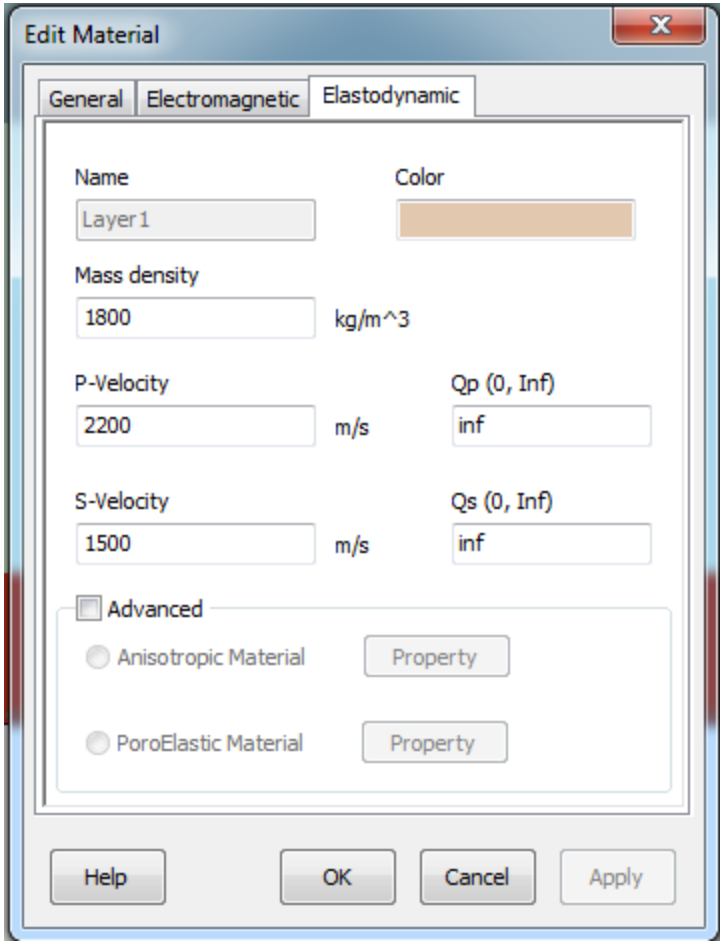


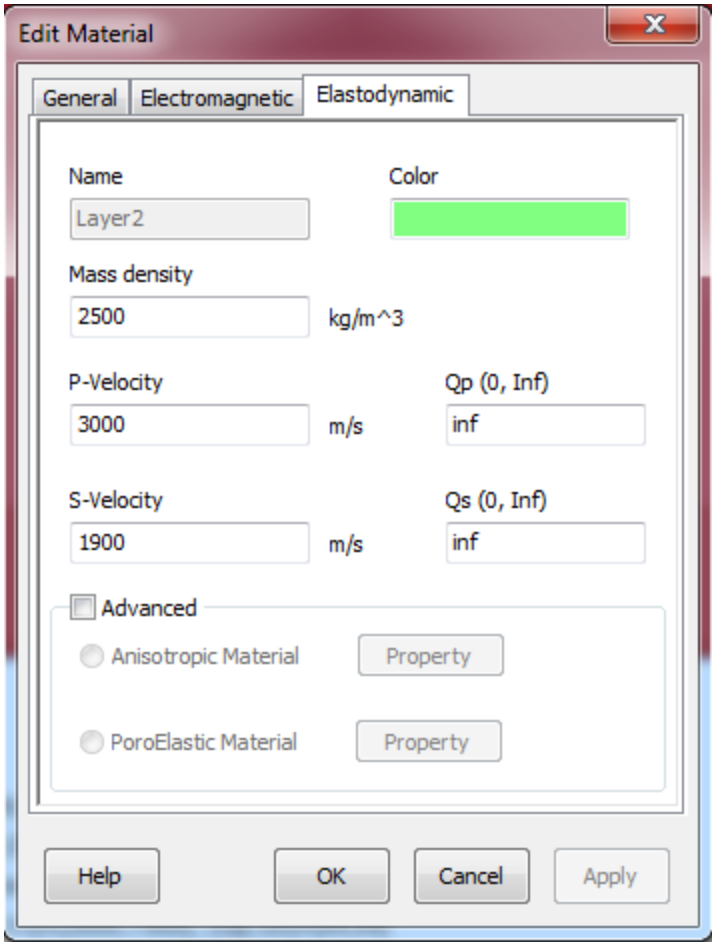
Then save as "Imaging_Layered_2D.wnt"

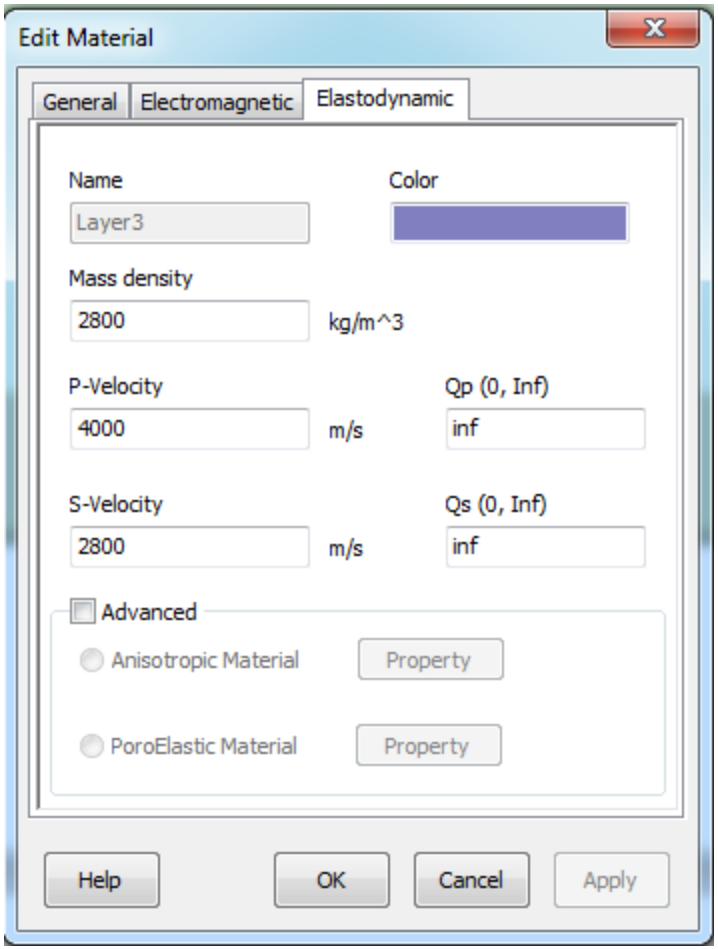
II. Setup materials



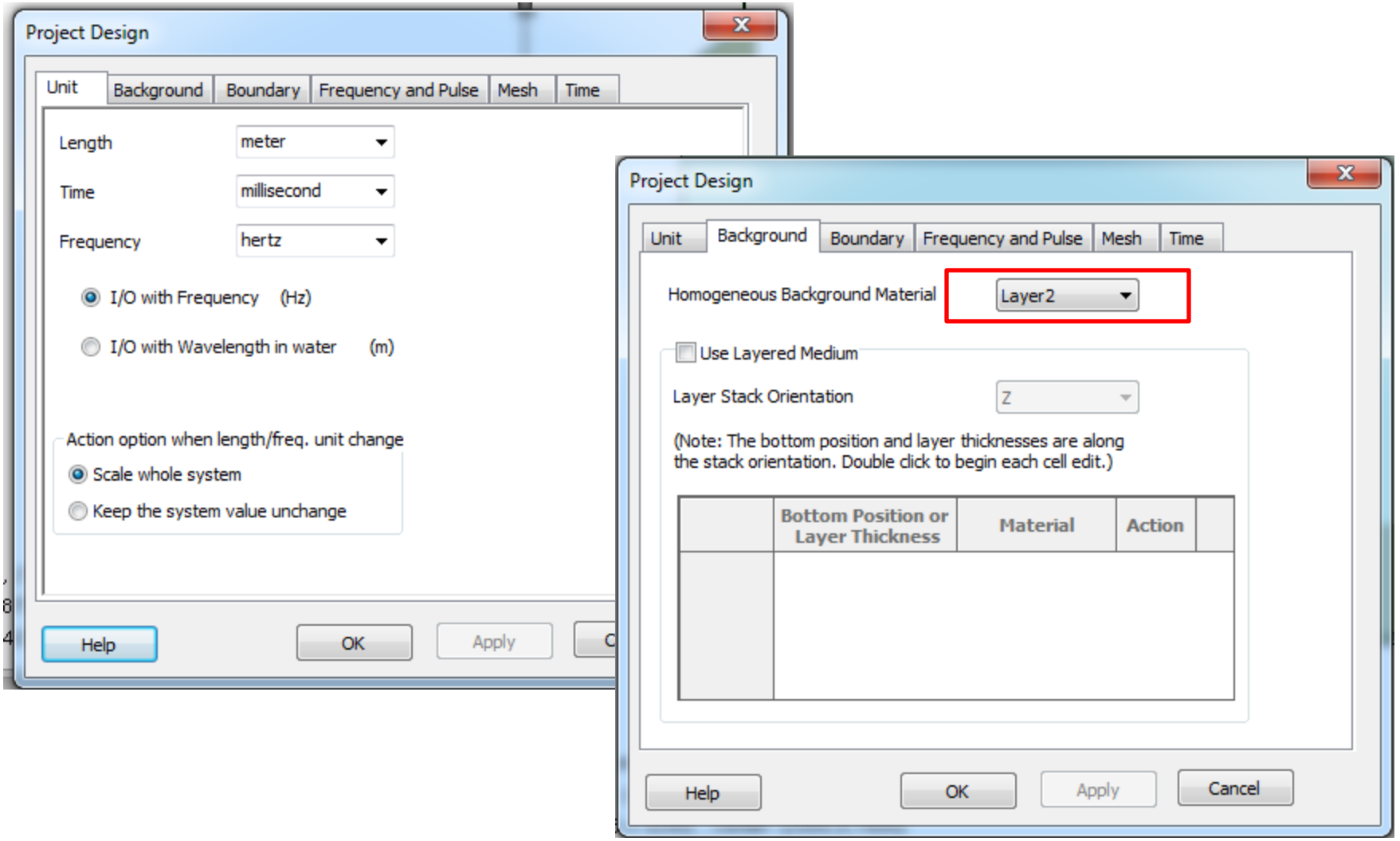


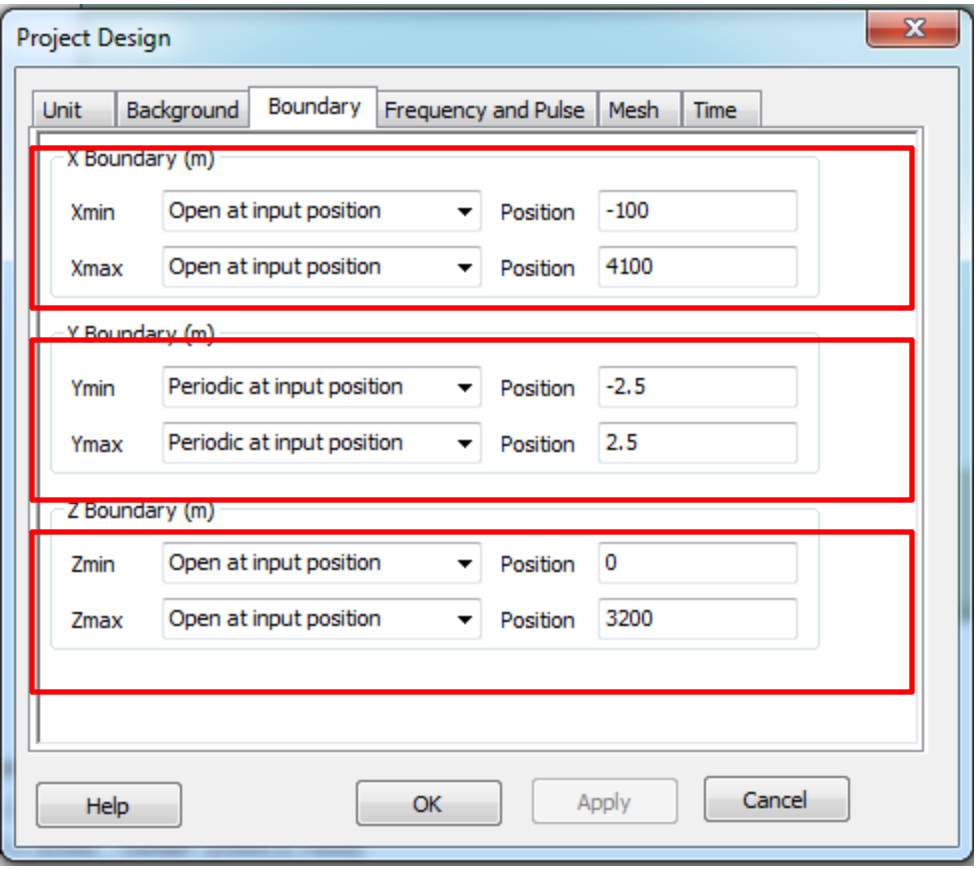



Edit Material
General
Electromagnetic
Elastodynamic
Name
Layer2
Color
Mass density
2500
kg/m^3
P-Velocity
3000
m/s
Qp (0, Inf)
inf
S-Velocity
1900
m/s
Qs (0, Inf)
inf
Advanced
Anisotropic Material
Property
PoroElastic Material
Property
Help
OK
Cancel
Apply


Edit Material
General
Electromagnetic
Elastodynamic
Name
Layer3
Color
Mass density
2800
kg/m^3
P-Velocity
4000
m/s
Qp (0, Inf)
inf
S-Velocity
2800
m/s
Qs (0, Inf)
inf
Advanced
Anisotropic Material
Property
PoroElastic Material
Property
Help
OK
Cancel
Apply

III. Setup project background, pulse, mesh & time system



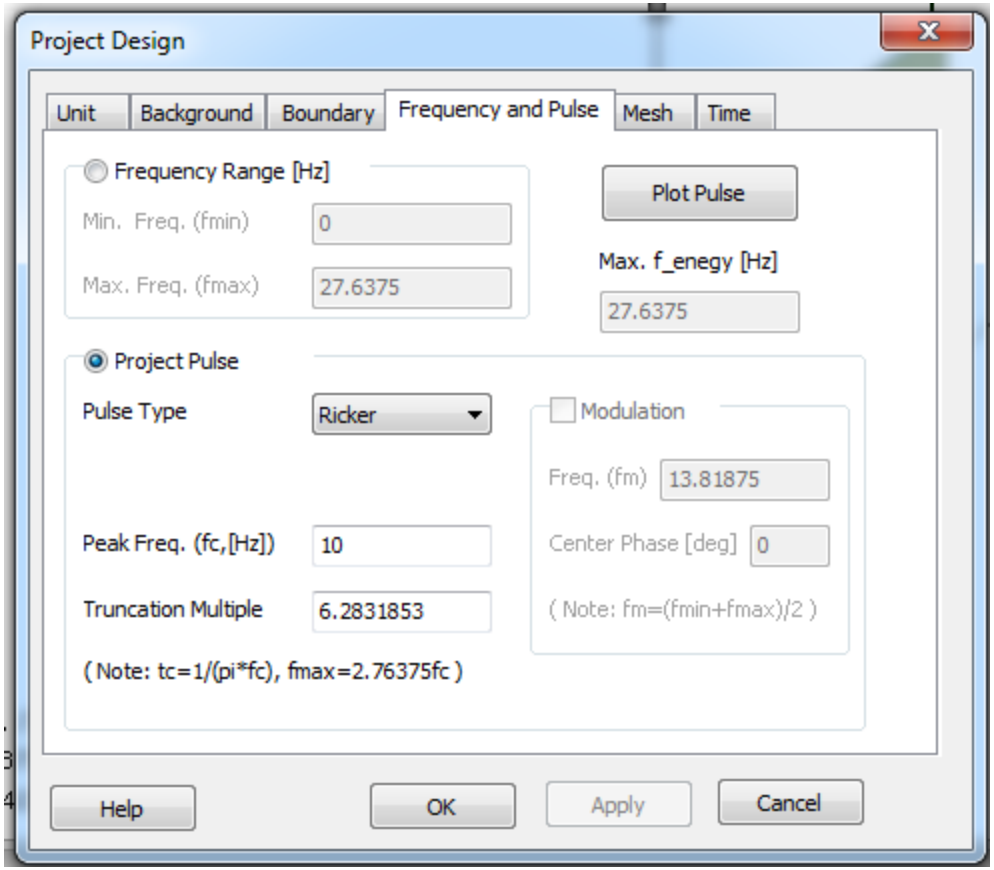


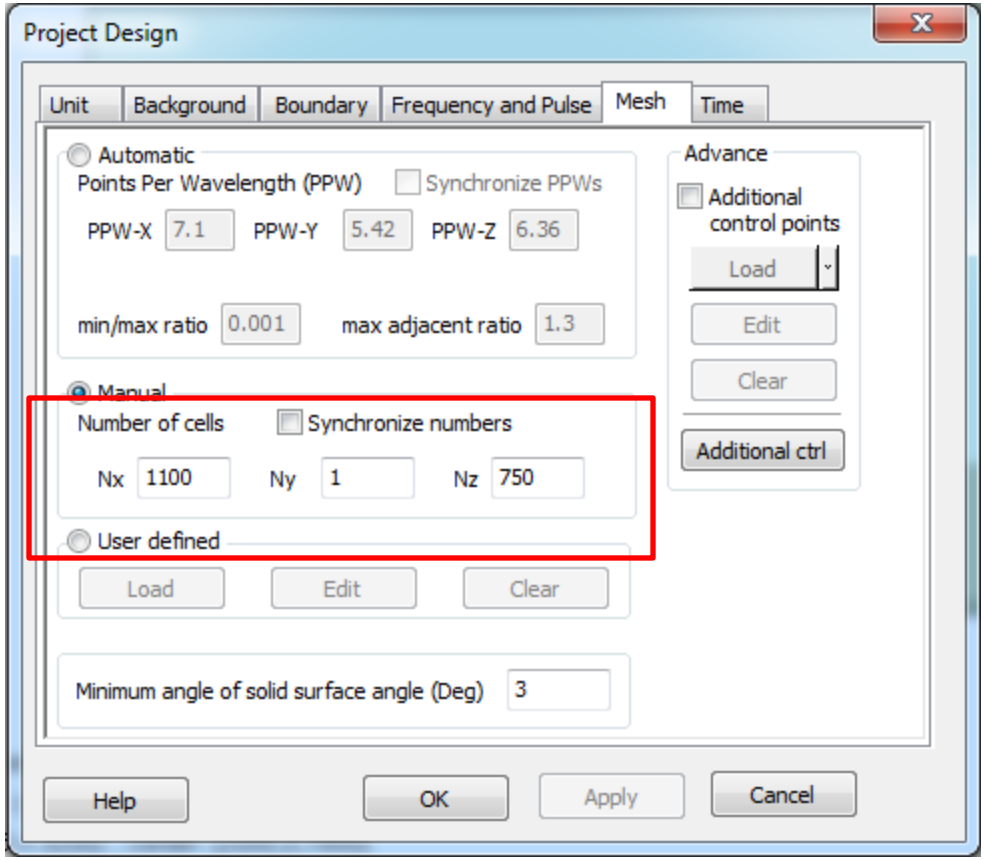
Note: due to the imaging size is determined by us, we can only define it by known signal time window and source, sensor positions

- the source + sensor have a X range in [20, 3970]m, so, we can let the X range of the simulation space as [-100, 4100] m

- the signal time window is 4.5 s, based on the slowest Vs=1900 m/s for all background materials. So, the travel distance will be around 8500 m with this Vs. In order to make sure the source signal can travel the whole simulation space, we can define the Z range of space as [0, 3200] m.

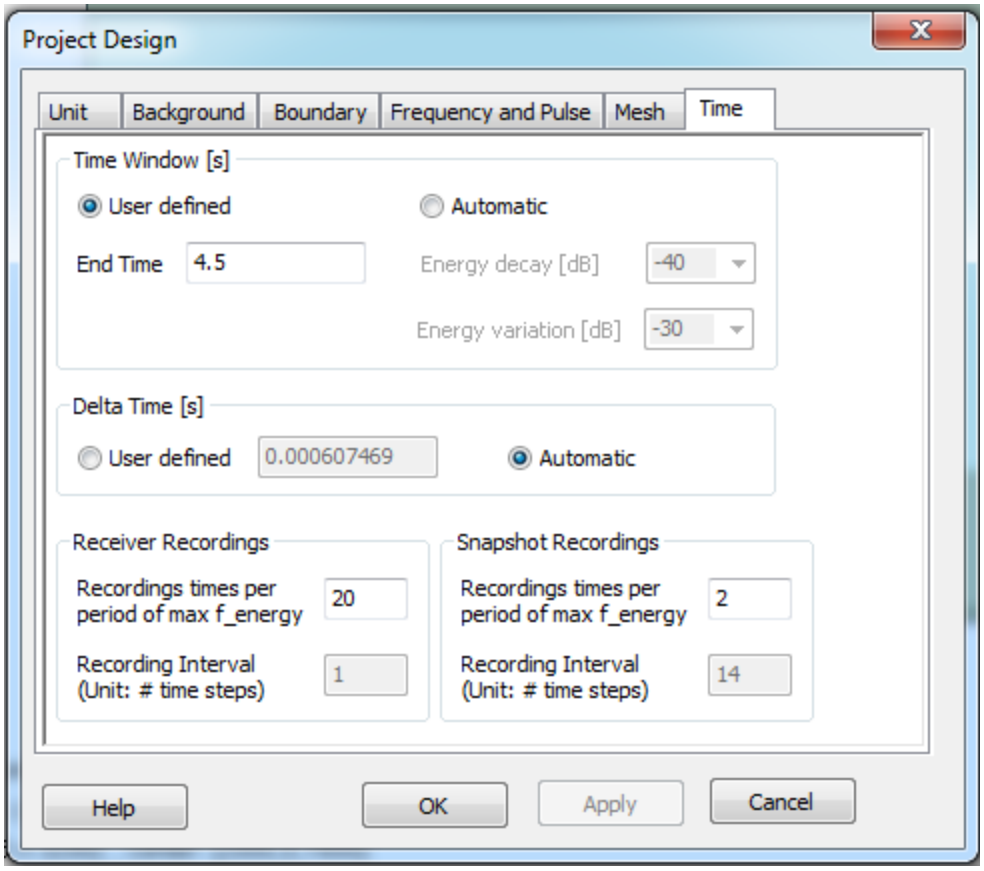
Source pulse is the Ricker wave with fc=10 Hz (eq. fmax=27.6 Hz)



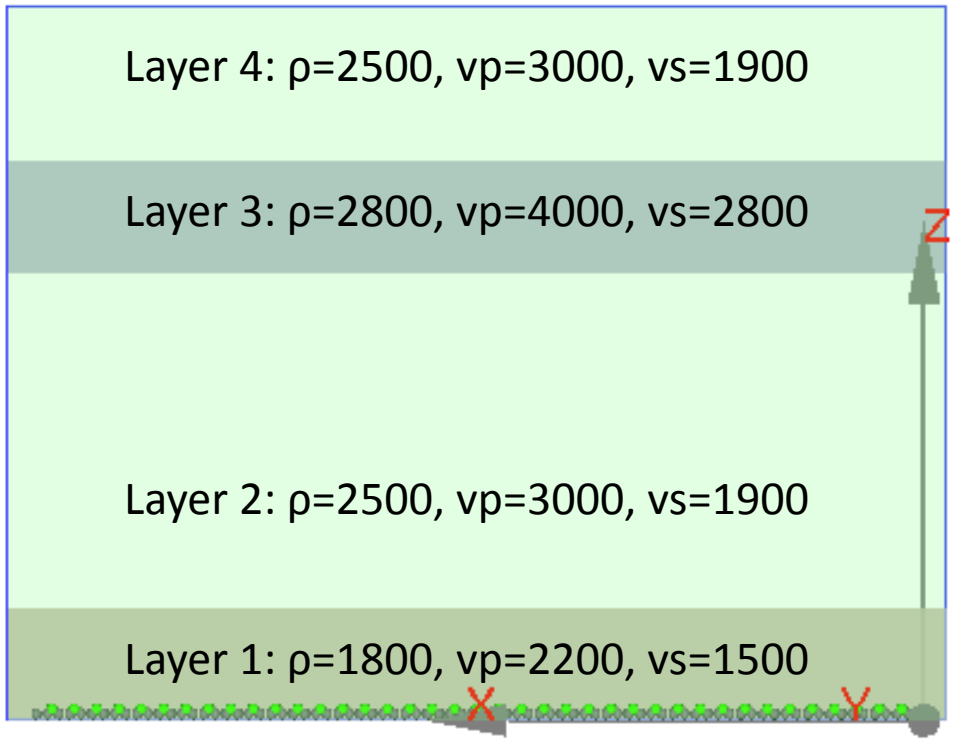


- due to we don't known too much information about the real space, we can define the simulation mesh as [1100, 1, 750]
  - with this setting, we can see the space sampling density will be around ppw_x=7.1, ppw_z=6.4 (this is 2D case in Y, ppw_y can be ignored). Due to this is an imaging procedure, we don't need very accurate result, so, make sure ppw > 6 is enough.
    - higher ppw will get better result, but need to take longer imaging time.

Define the time windows as detected signal's time window.
Then use automatic Δt.



IV. Setup 4 layers background

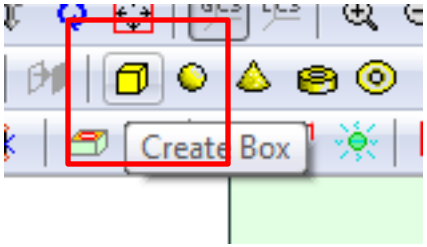


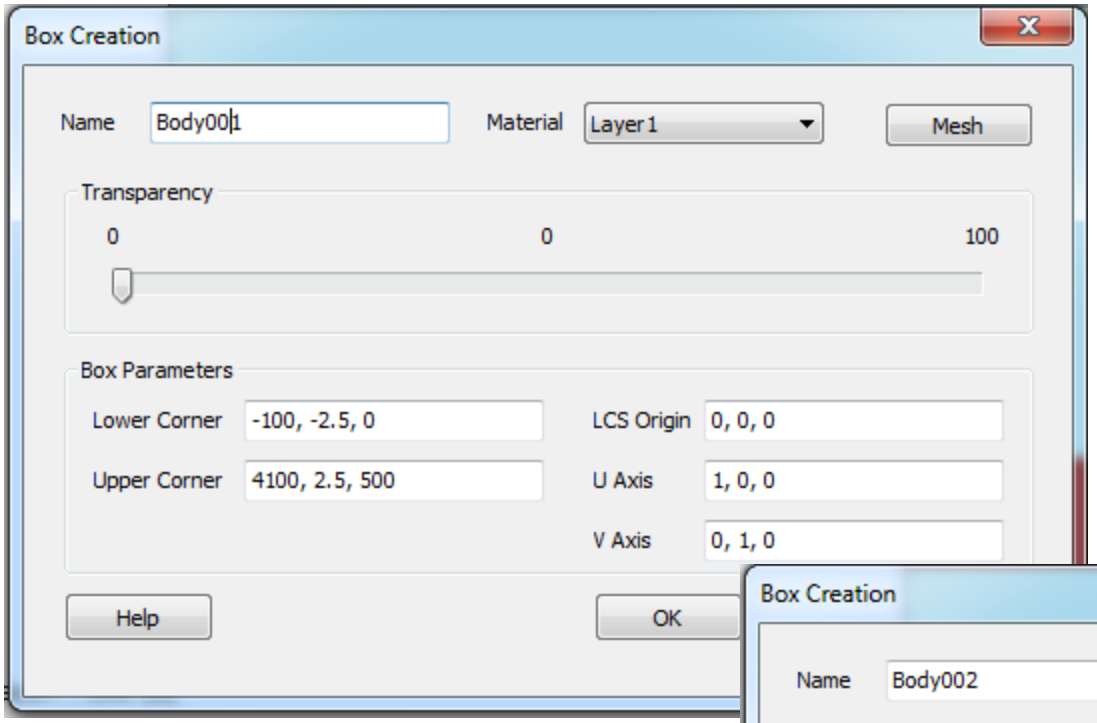
We can use following 3D geometry system to simplify the modeling

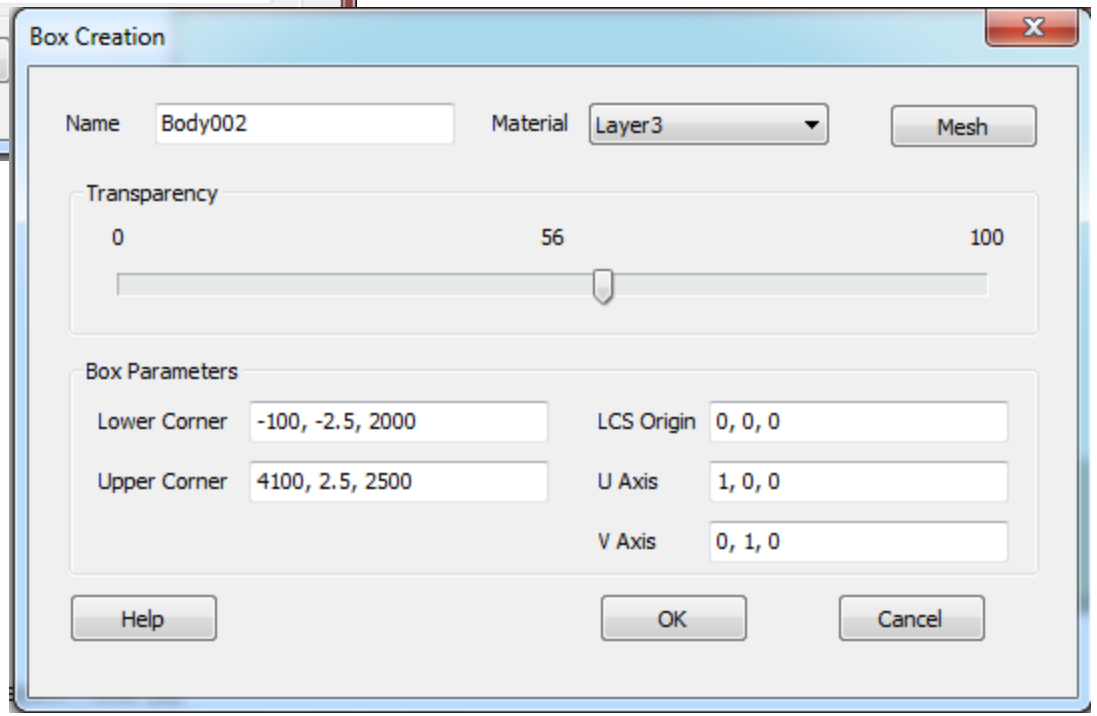
- ➢ homogeneous background by the material of layer 2
- ➢ layer 1 is modeled as a box
- ➢ layer 3 is modeled as box also

❑ due to the region outside the layer 1 & 3 will be the background, the layer 2 & 4 will be built automatically after layer 1 & 3 are built

Click “Create Box” to build layer 1 & 3

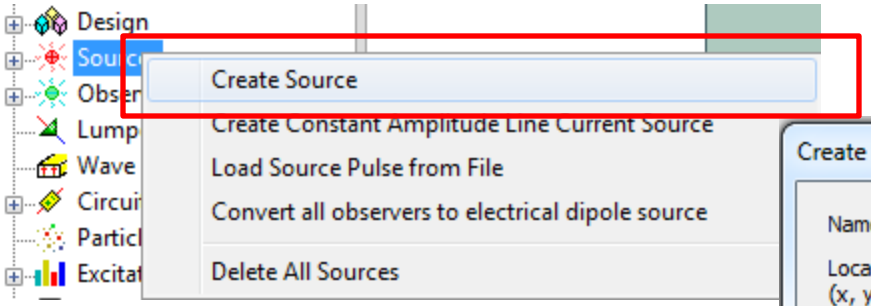


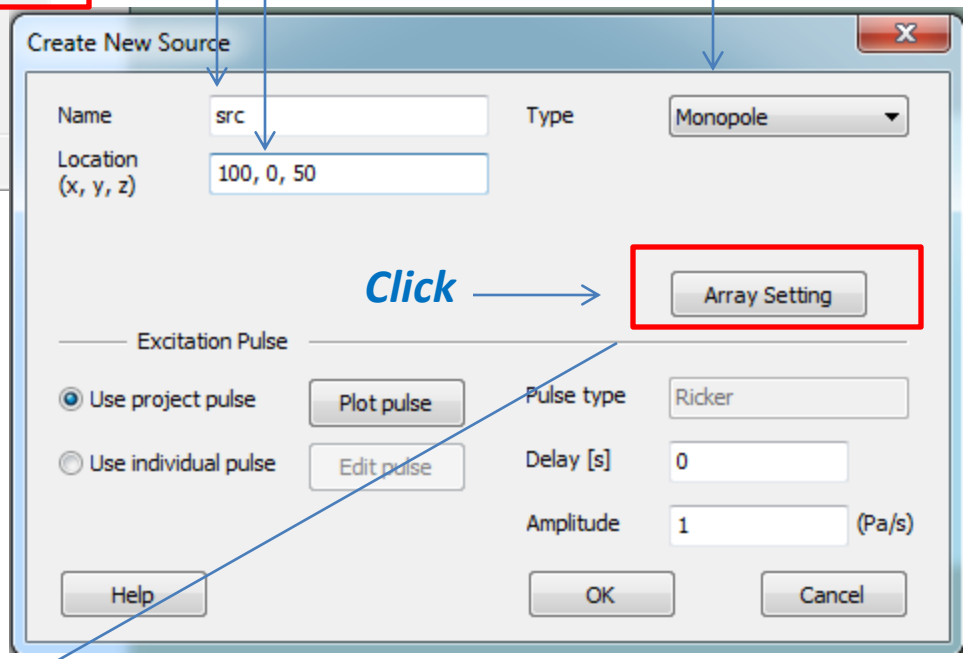




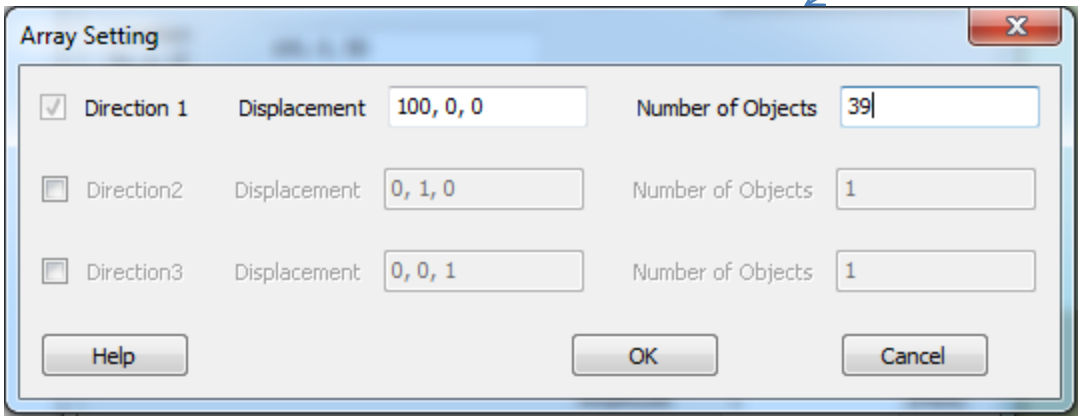
V. Setup 39 sources by “Array Creation”

Input
- the source name prefix
- first source position
- source type

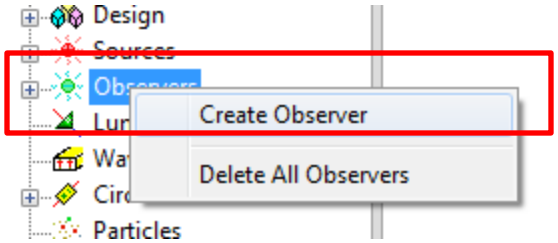




39 sources with a X distance as 100 m

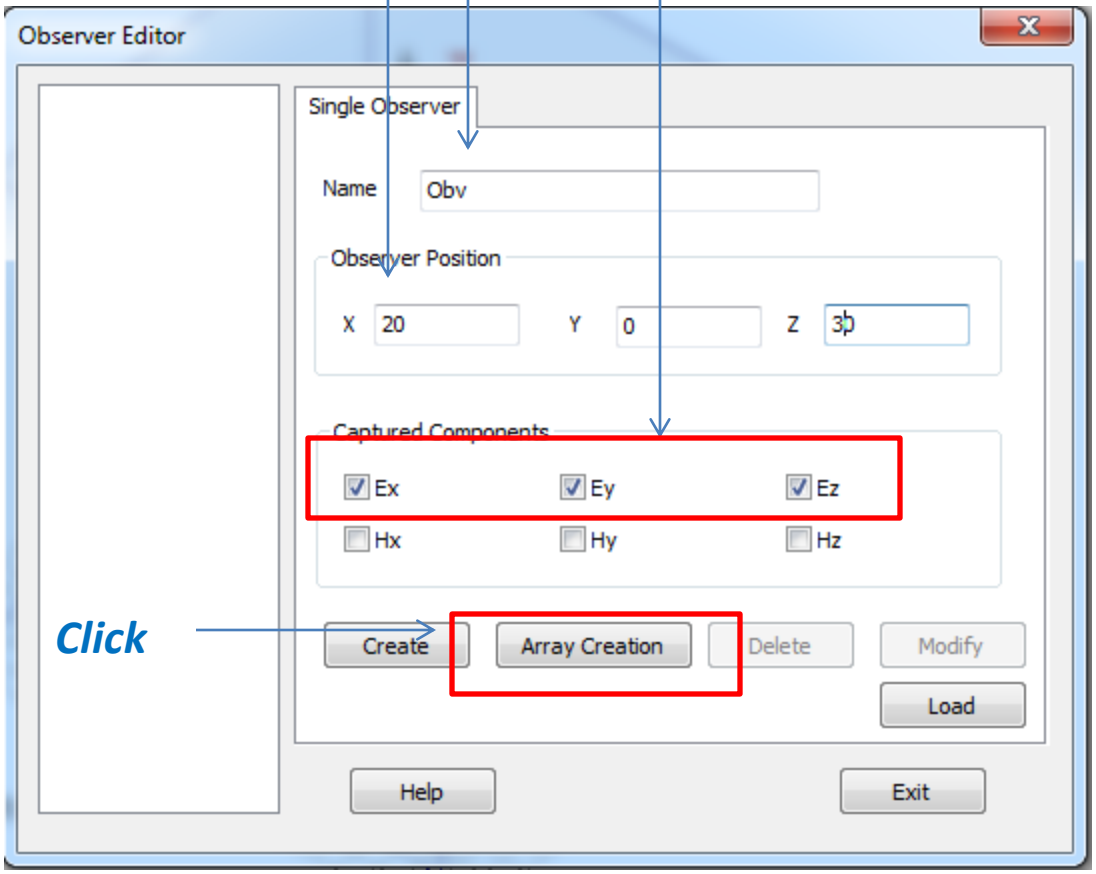


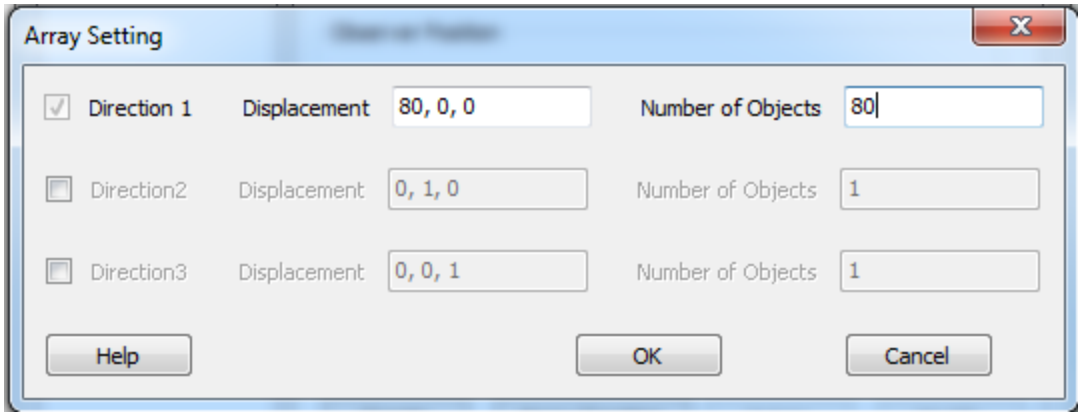
VI. Setup 80 receivers by “Array Creation”



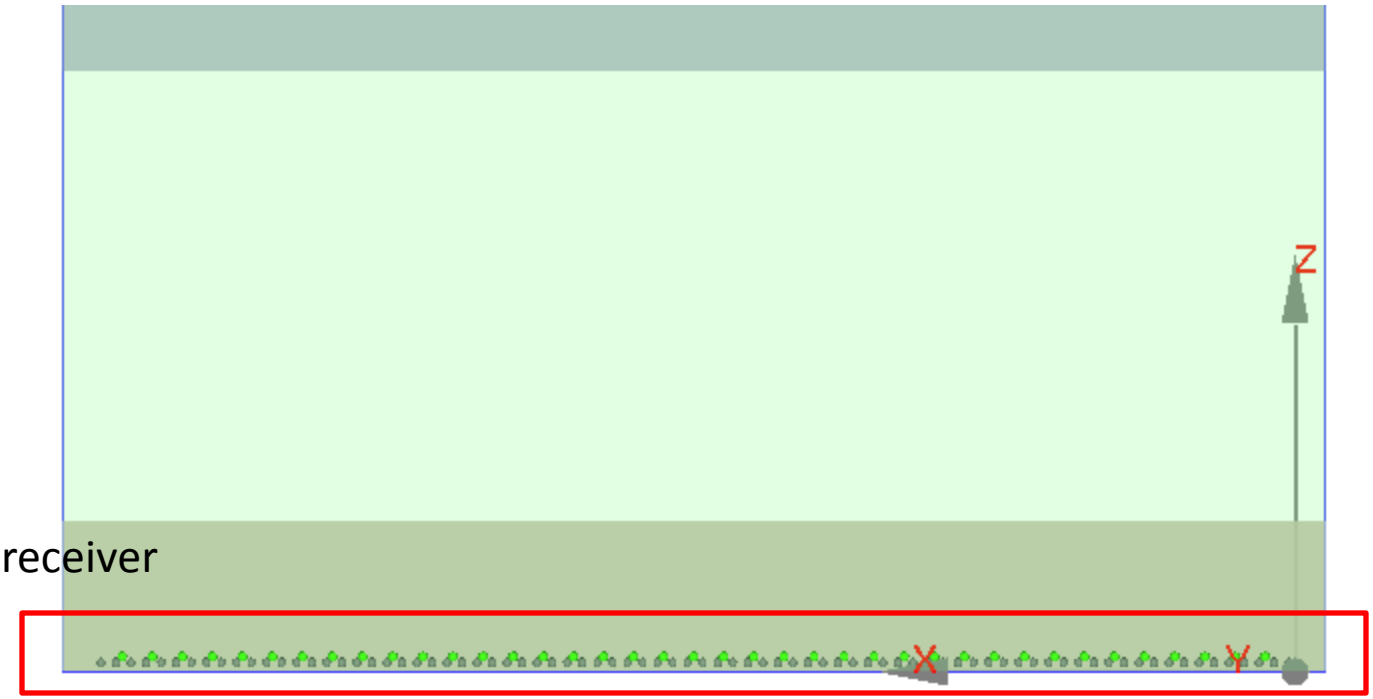
Input

- the receiver name prefix
- first receiver position
- record component



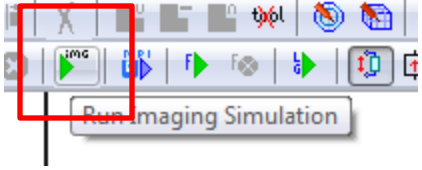


After source & receiver are created, the project layout will be as following



source & receiver

VII. Setup WCT EL imaging system



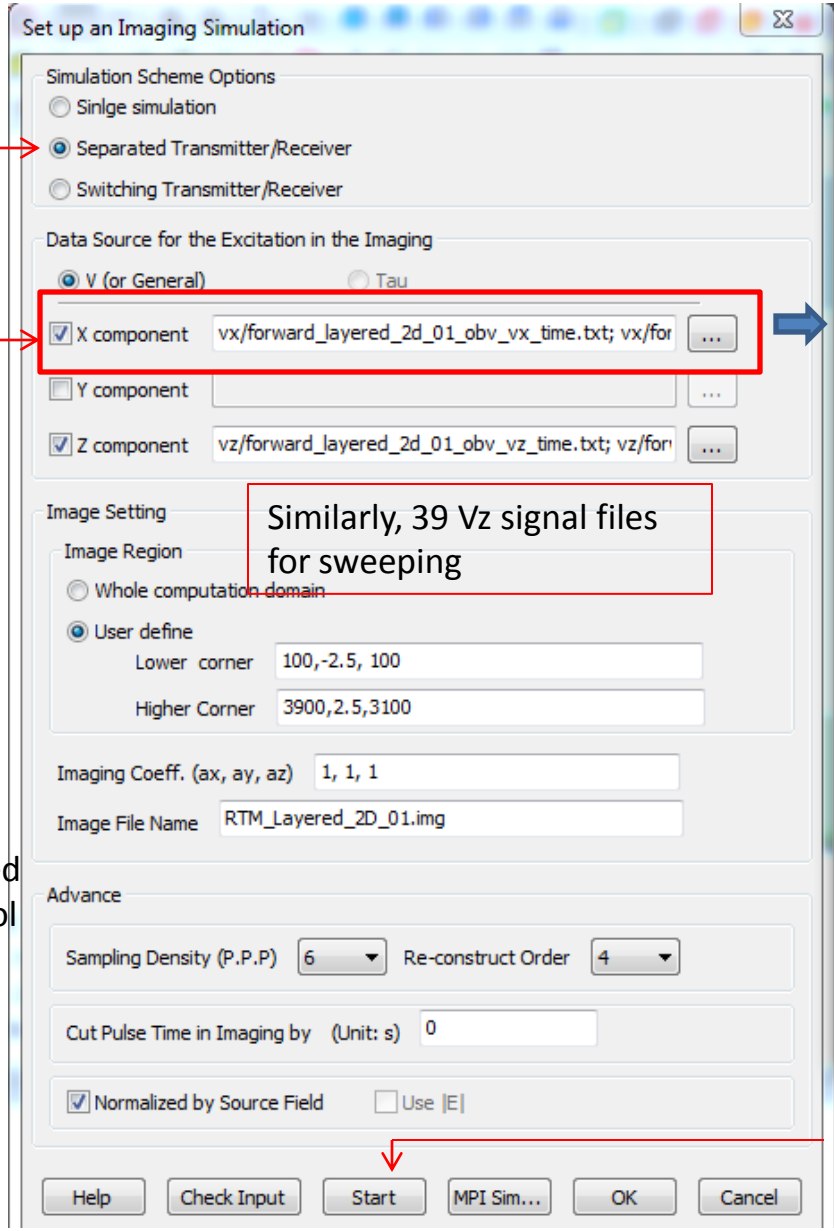
We will sweep 39 source to make the image

For each source, there are 80 Vx traces in one file, so, totally 39 Vx trace file in sweeping

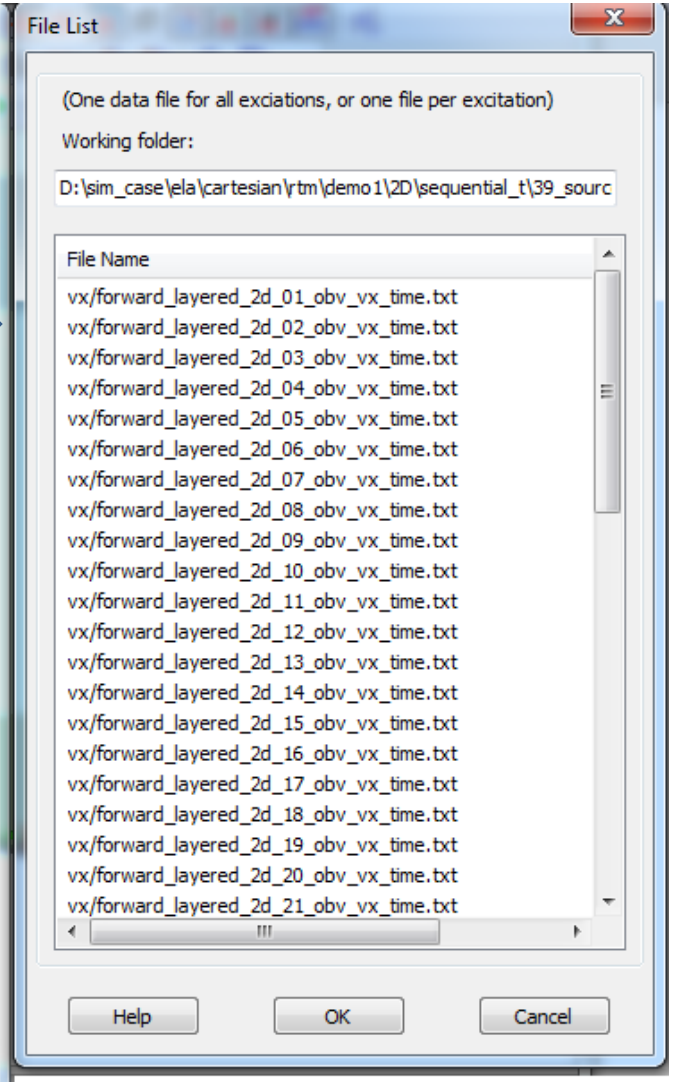
Image file name & imaging region

Use default Advanced parameters in control imaging.

enable normalization



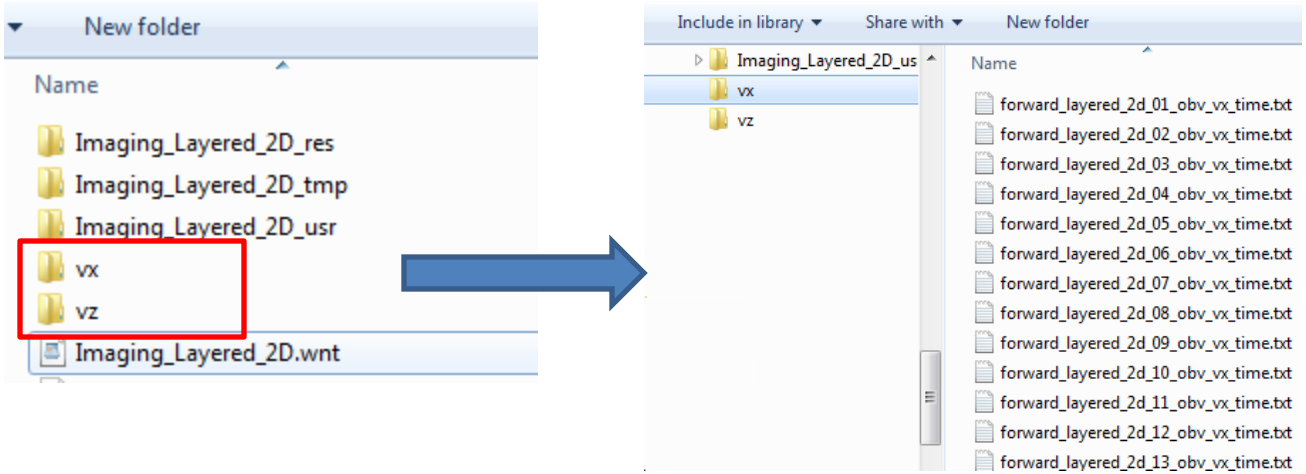
Similarly, 39 Vz signal files for sweeping

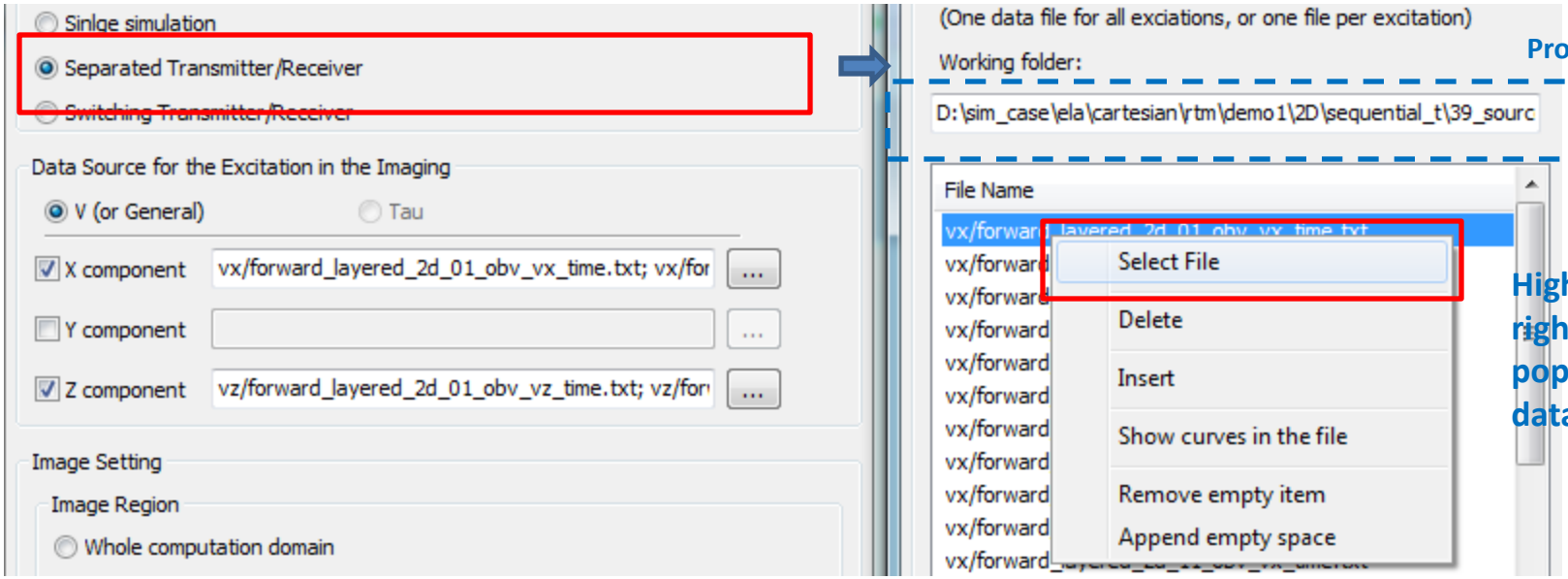


Start a “Forward+backward” processing to generate image.

the data files that will be used in imaging

- because there are 39 files for Vx & Vz, specifically, 78 files in one folder will make the management difficult
- here, we can put all Vx, Vz data files in a sub-folder “Vx” & “Vz” separately as

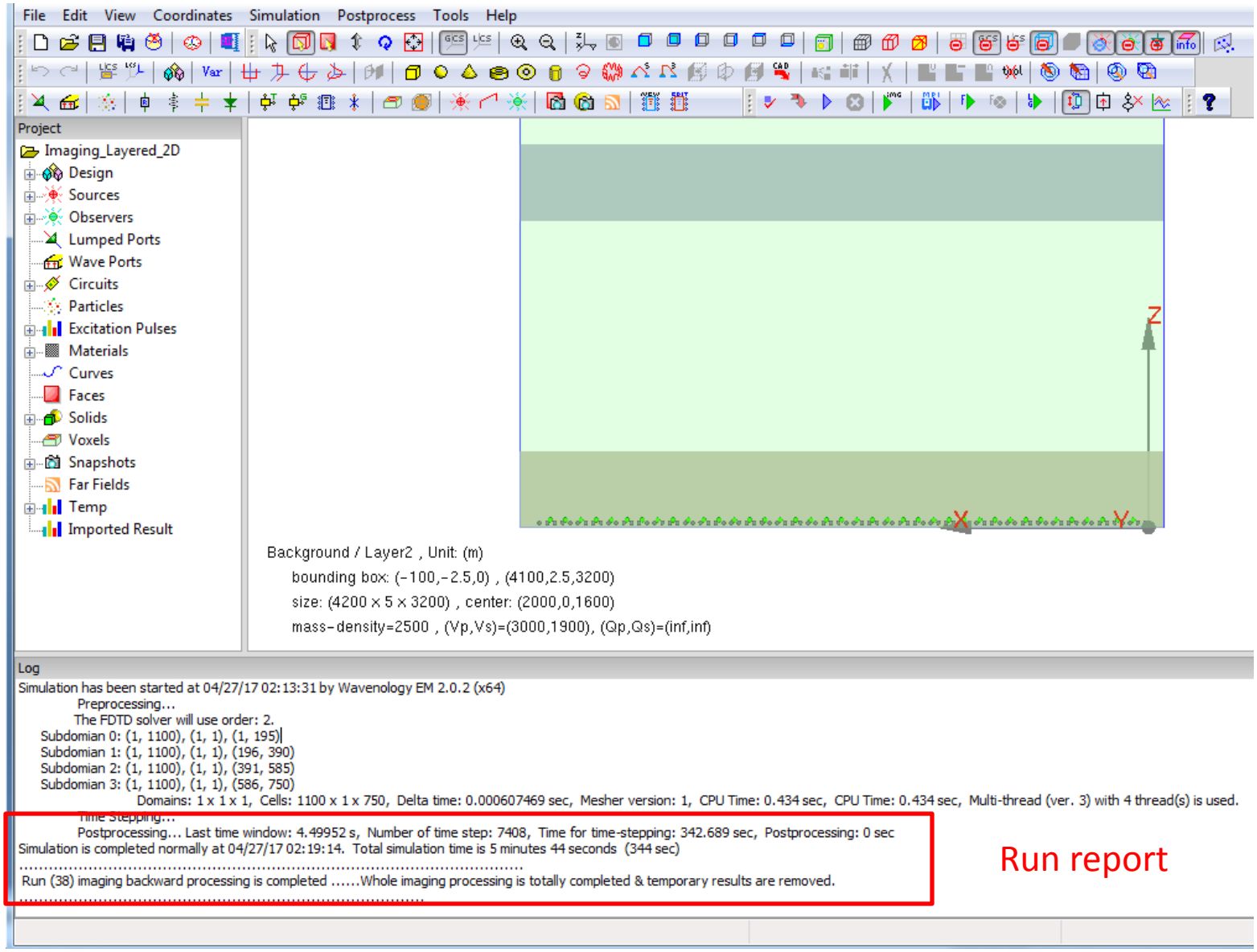




Project working folder

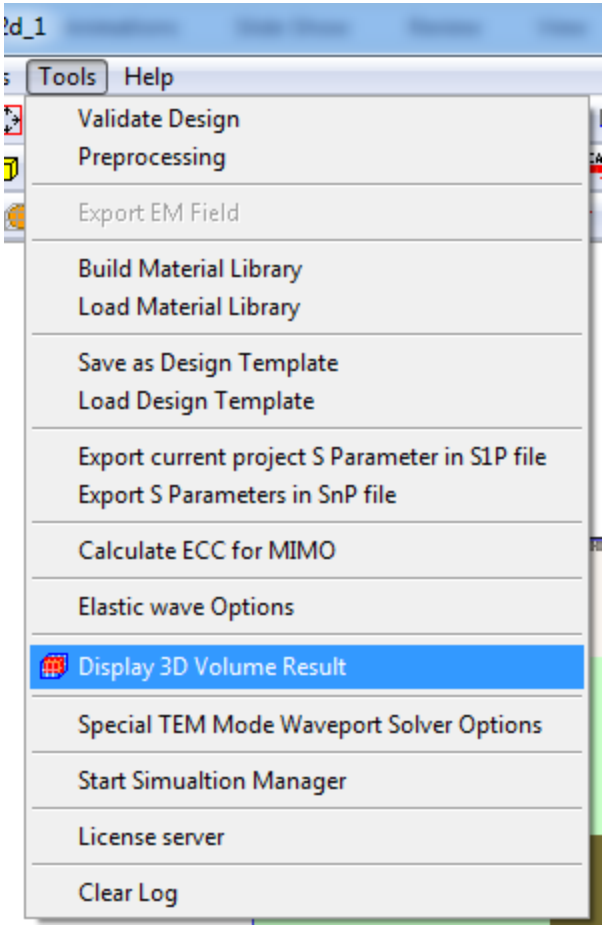
High-light any item, right click mouse to popup a menu, load data files

After the imaging procedure start, wait until the sweeping finish

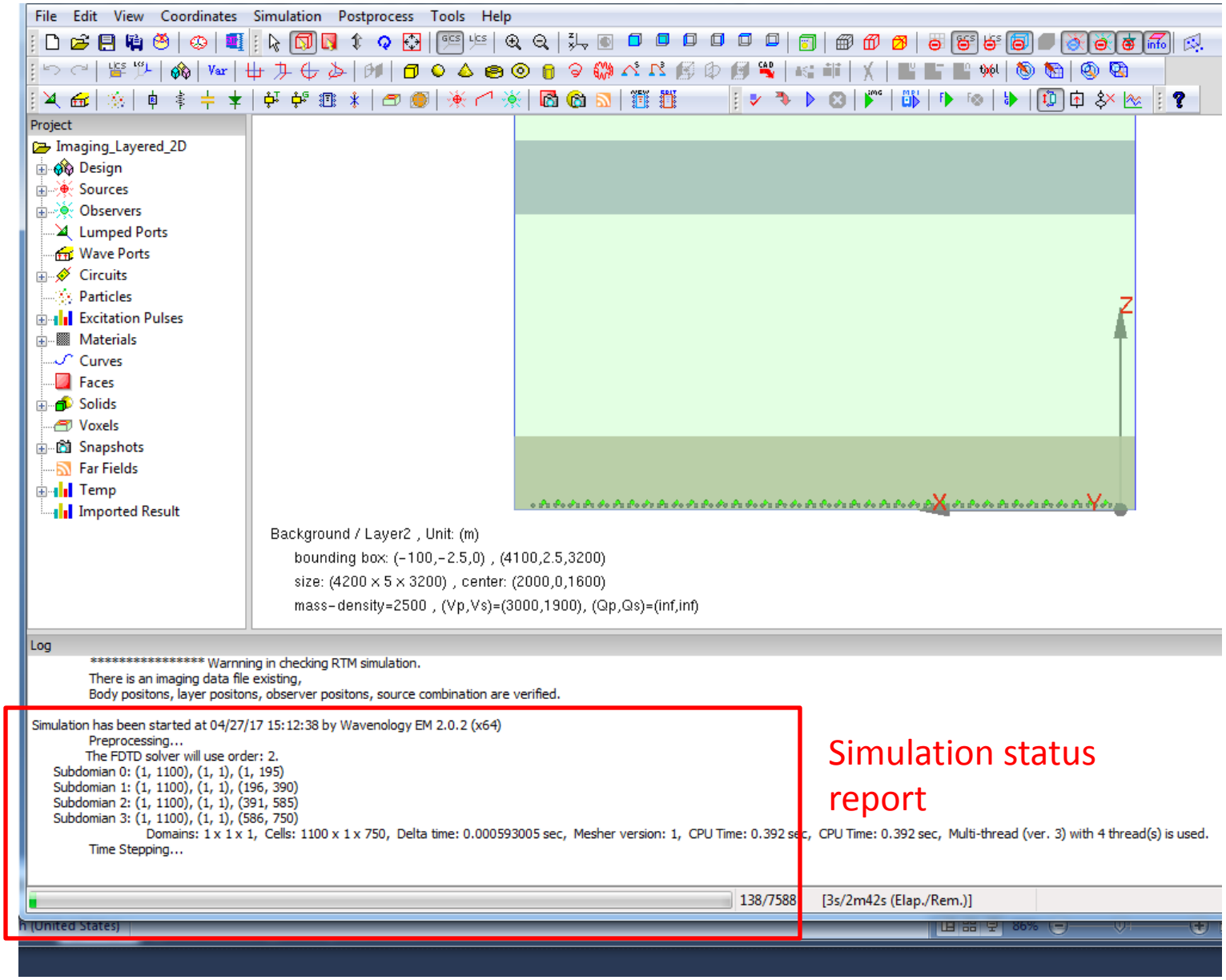


Run report

After simulation successfully, load the image

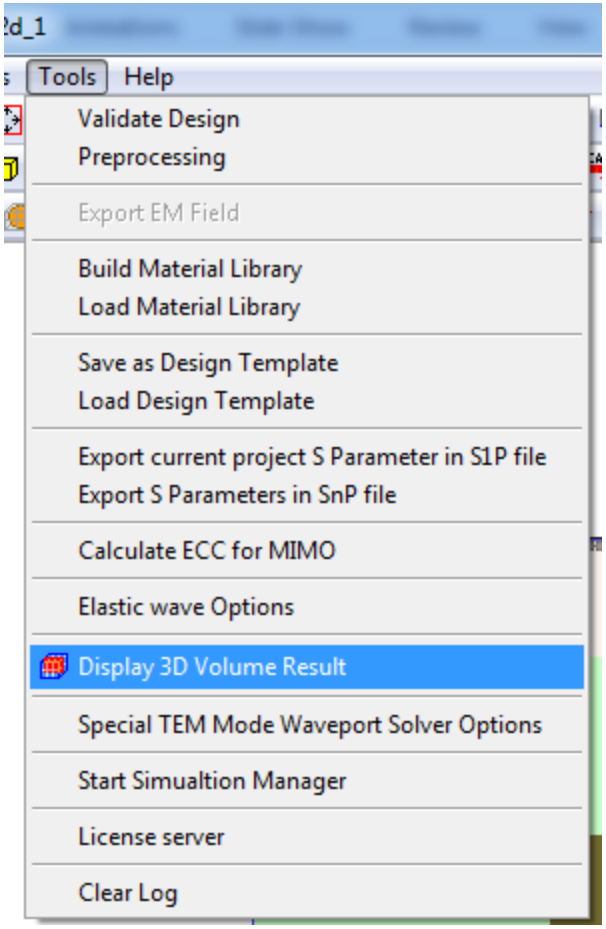


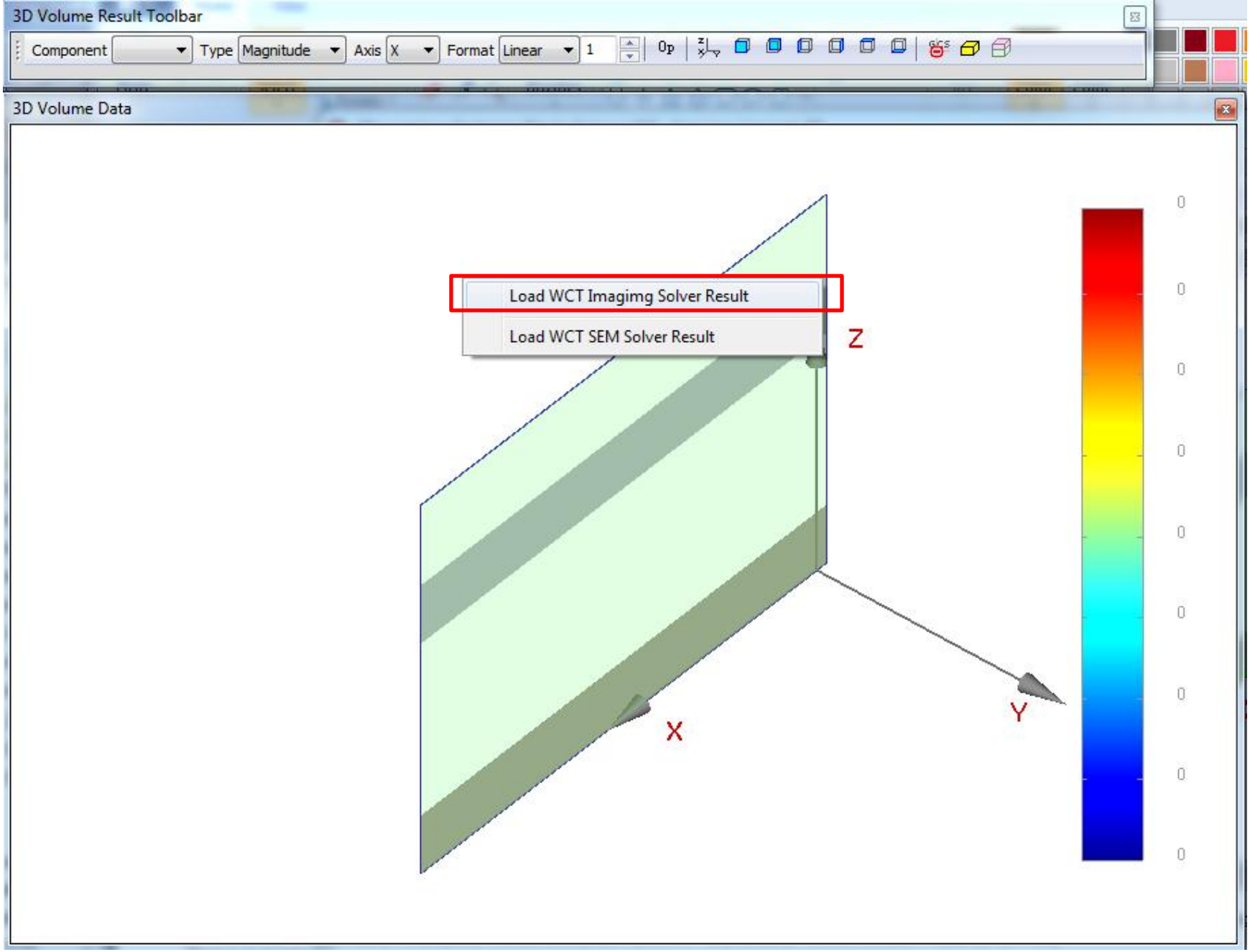
After the imaging procedure start, wait until the sweeping finish

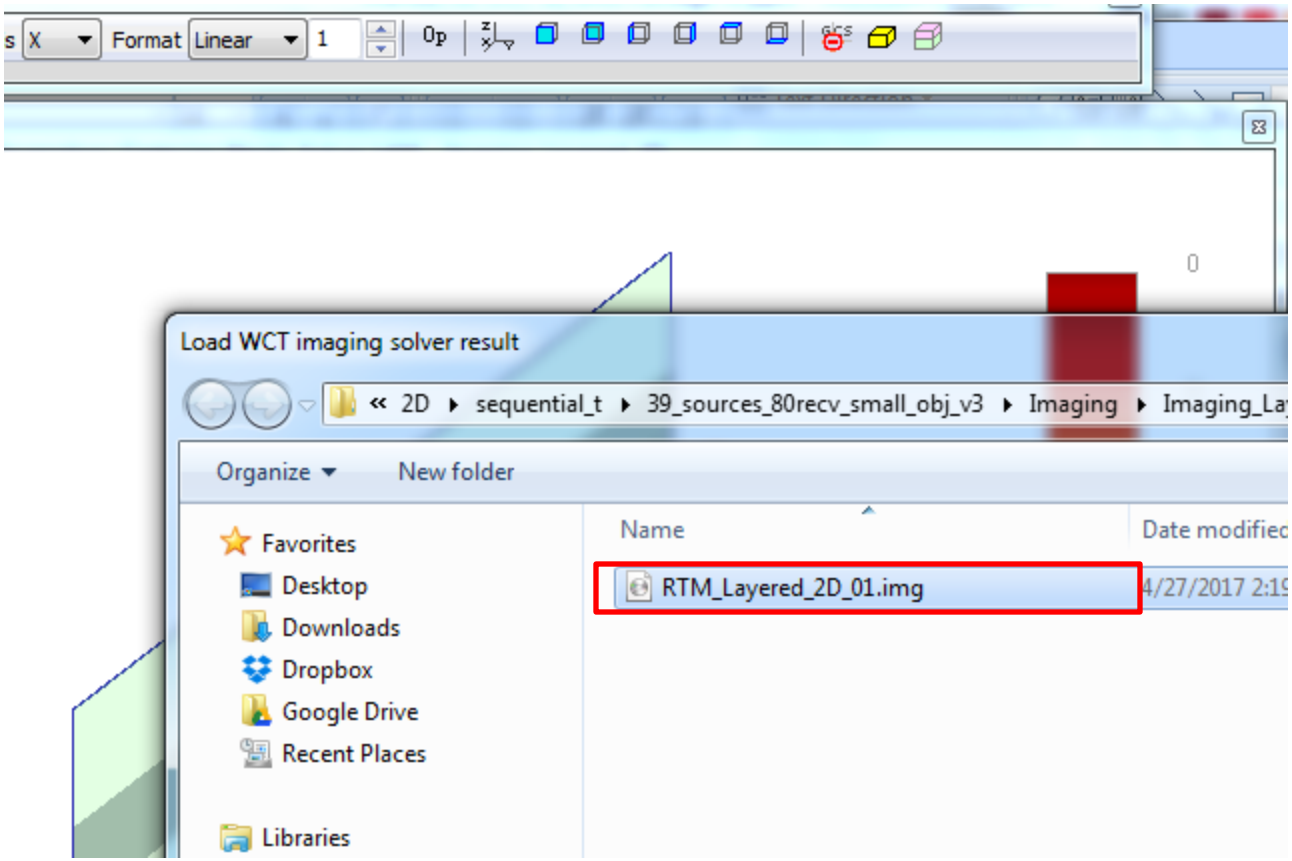


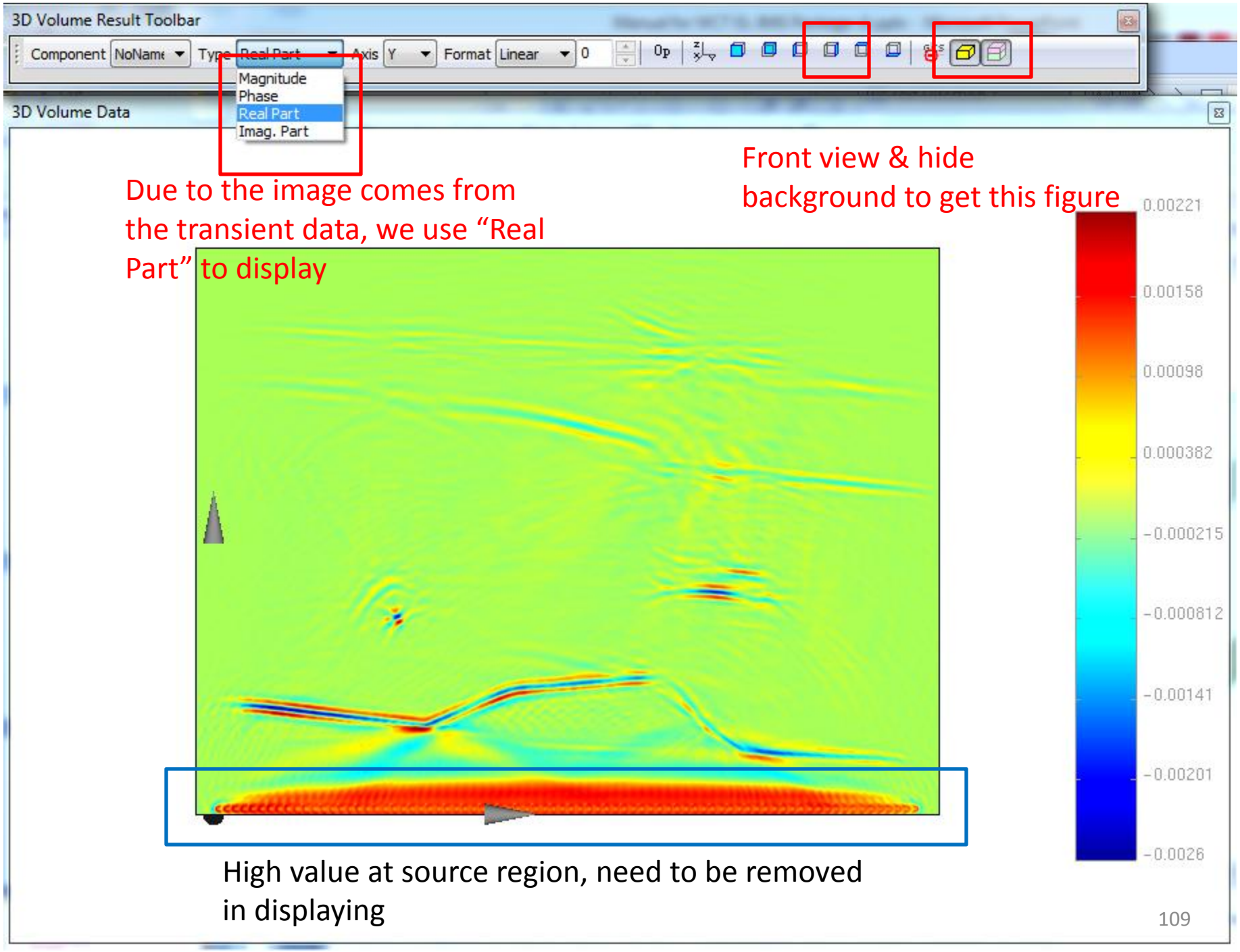
Simulation status report

After simulation successfully, load the image

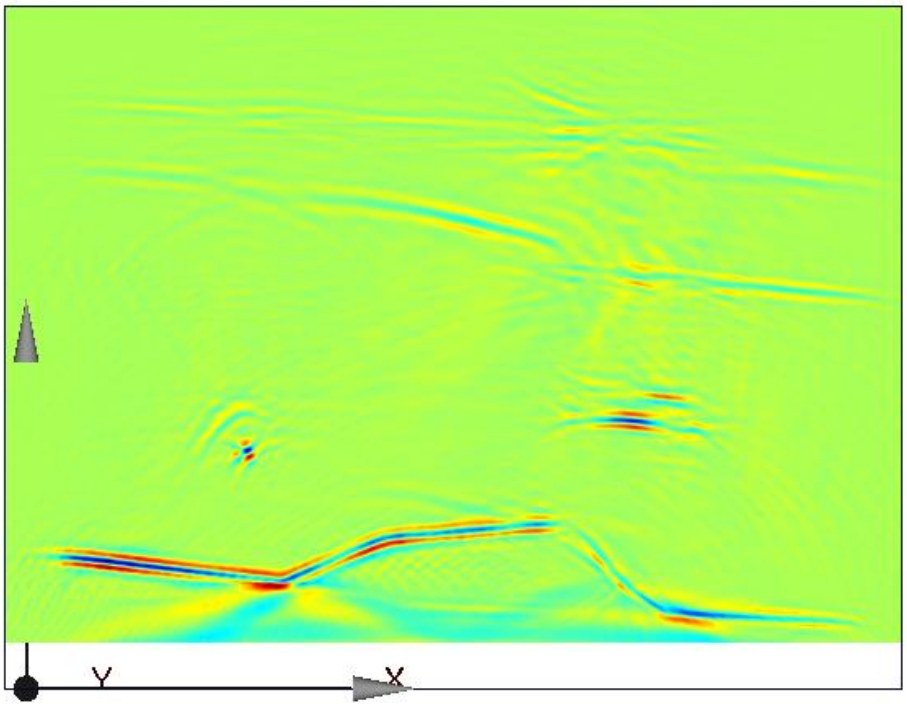


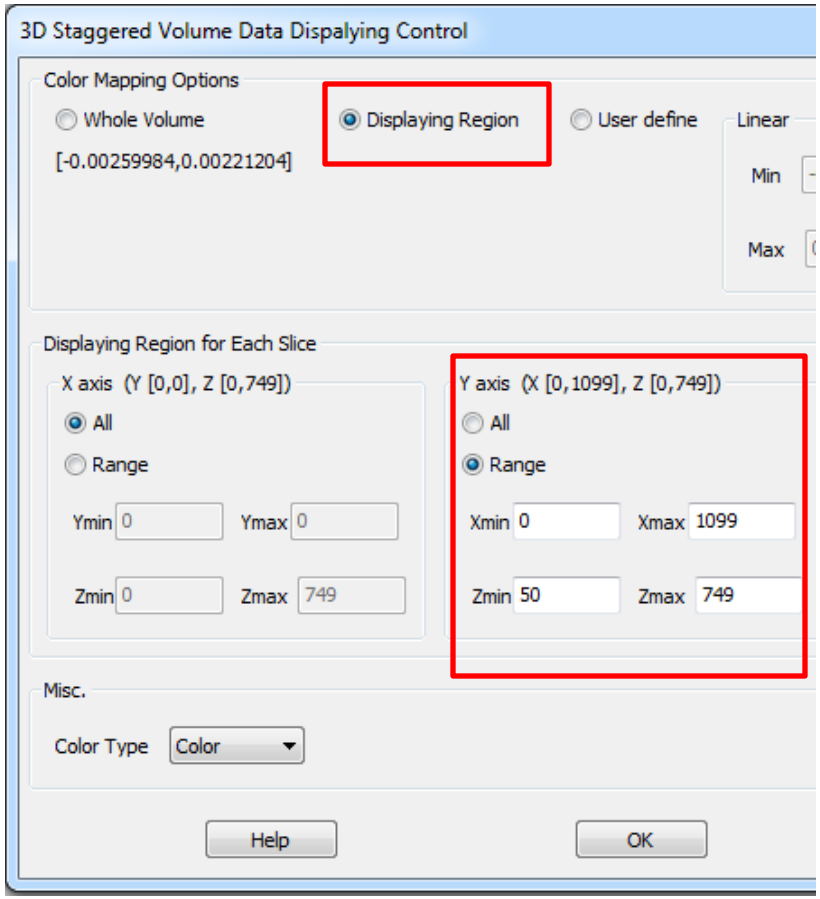

3D Volume Result Toolbar
Component
Type
Magnitude
Axis
X
Format
Linear
1
3D Volume Data
Load WCT Imagimg Solver Result
Load WCT SEM Solver Result
Z
X
Y
0
0
0
0
0
0
0
0
0


X
Format
Linear
1
0
Load WCT imaging solver result
« 2D ▸ sequential_t ▸ 39_sources_80recv_small_obj_v3 ▸ Imaging ▸ Imaging_La
Organize
New folder
Favorites
Desktop
Downloads
Dropbox
Google Drive
Recent Places
Libraries
Name
Date modifie
RTM_Layered_2D_01.img
4/27/2017 2:19

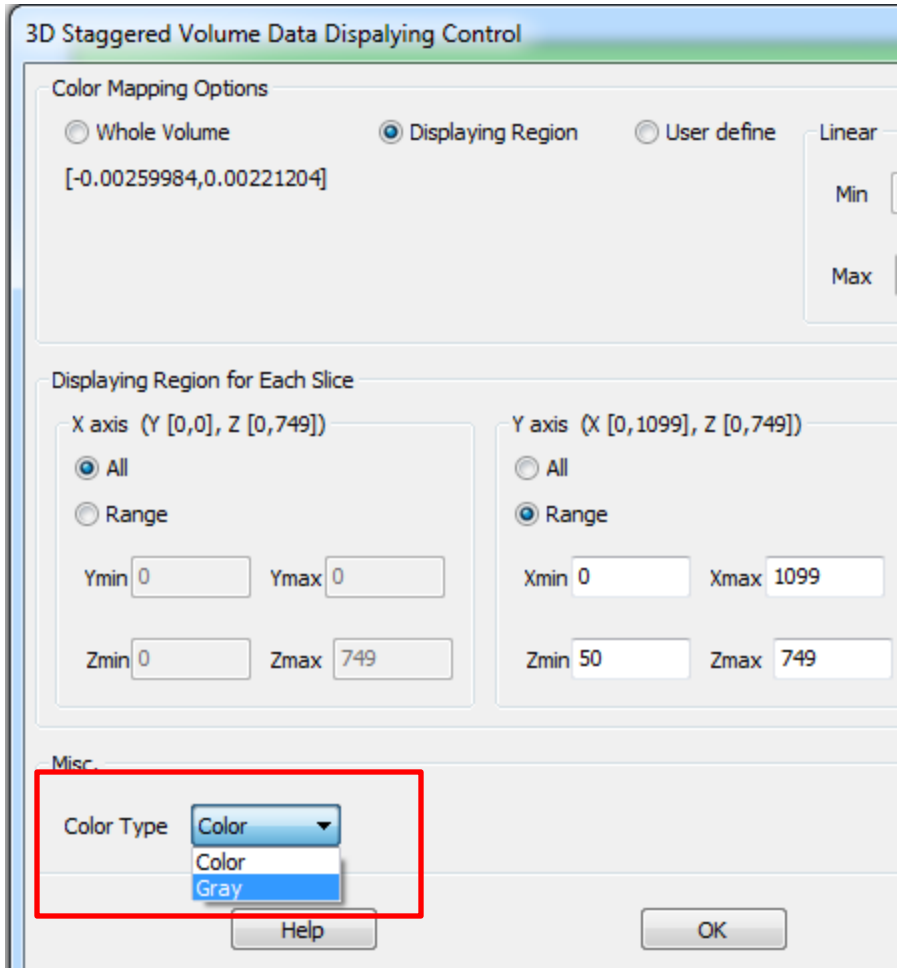

3D Volume Result Toolbar
Component
NoName
Type
Real Part
Axis
Y
Format
Linear
0
Magnitude
Phase
Real Part
Imag. Part
3D Volume Data
Due to the image comes from the transient data, we use “Real Part” to display
Front view & hide background to get this figure
0.00221
0.00158
0.00098
0.000382
-0.000215
-0.000812
-0.00141
-0.00201
-0.0026
High value at source region, need to be removed in displaying

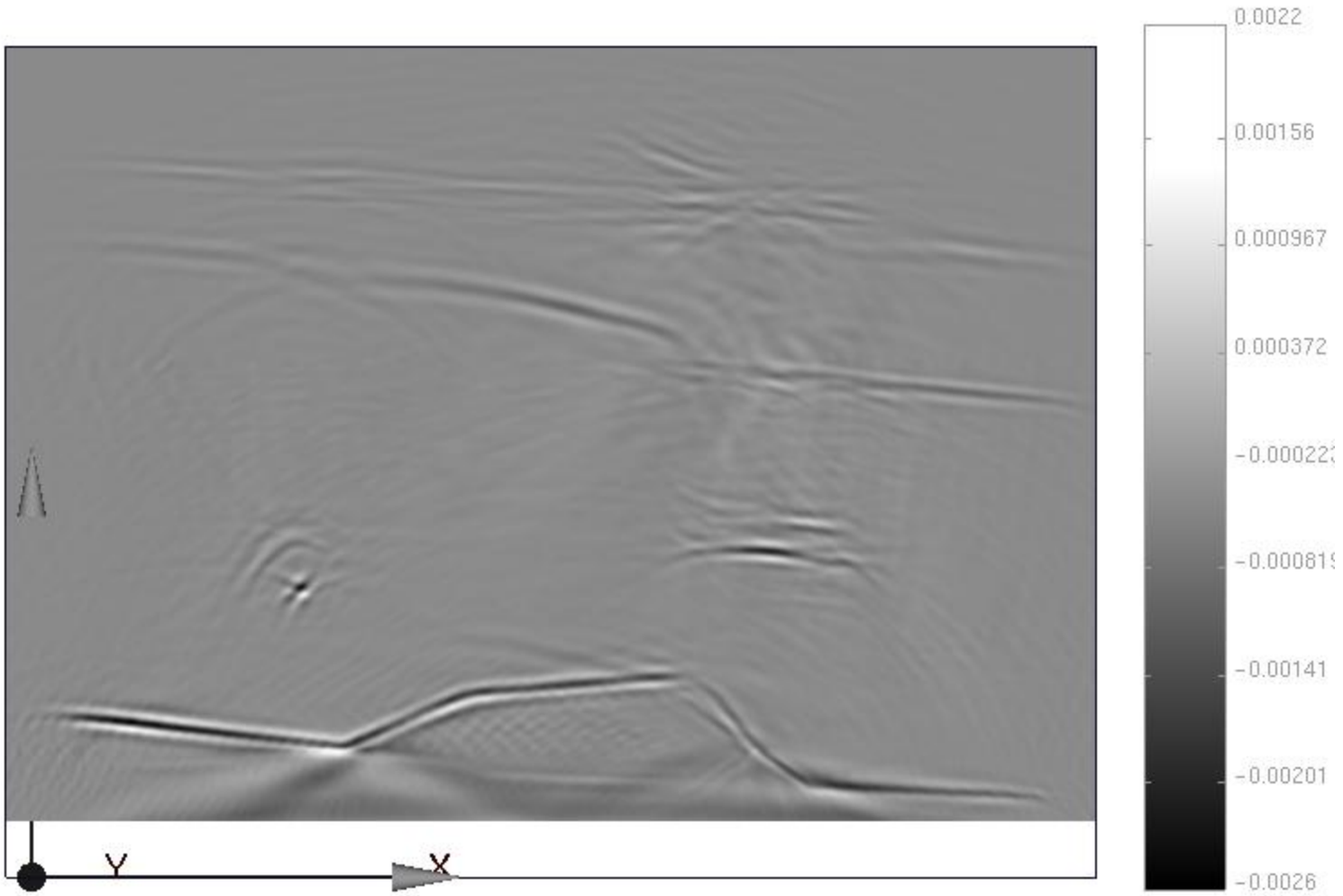
Use option to change the image displaying & color range

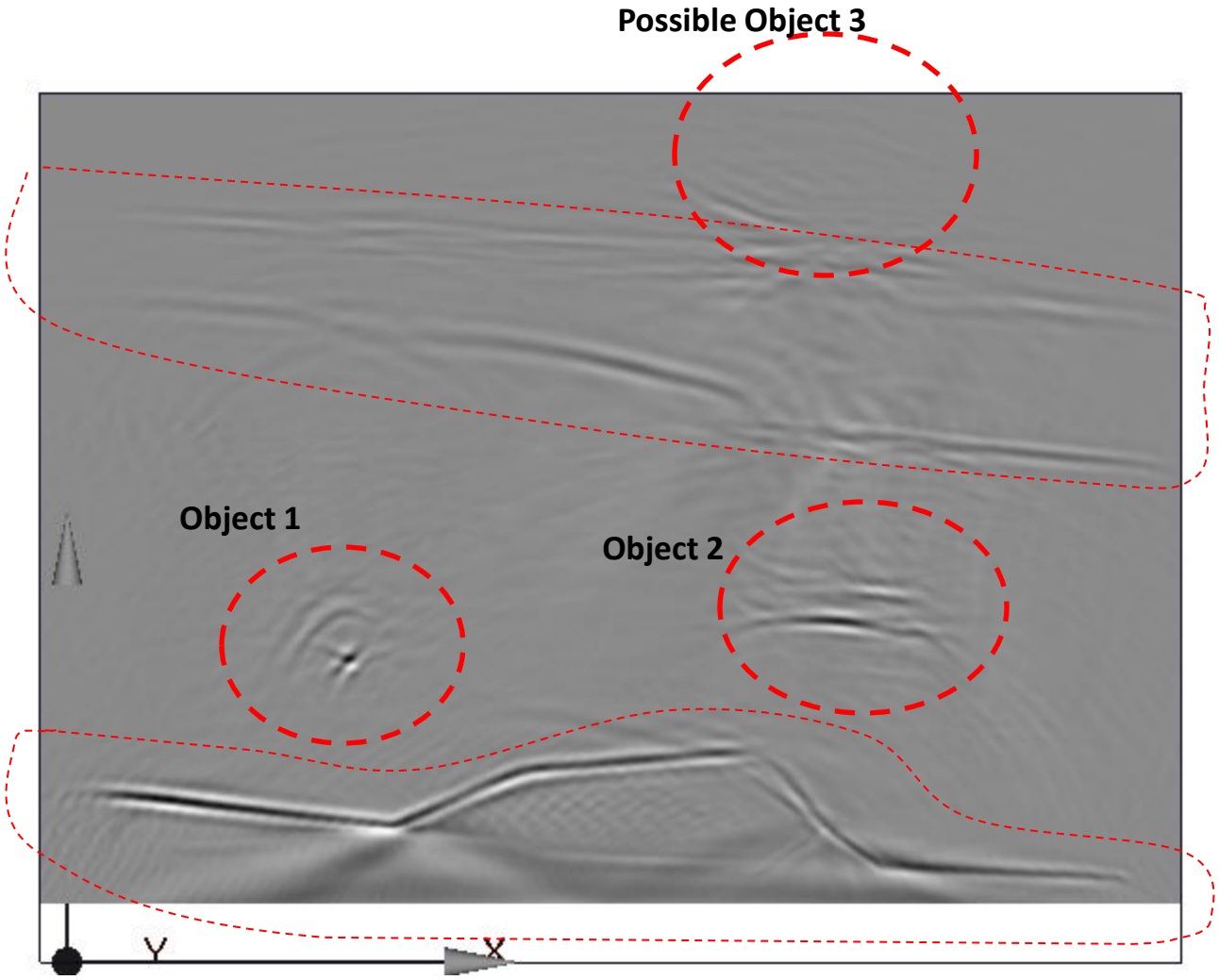




switch to Gray displaying




0.0022
0.00156
0.000967
0.000372
-0.000223
-0.000819
-0.00141
-0.00201
-0.0026
Y
X


Possible Object 3
Object 1
Object 2
Interface of layer 3
Interface of layer 1
Y
X

Compare the image with the ground true space, we can see the contour of layers & objects can be shown correctly

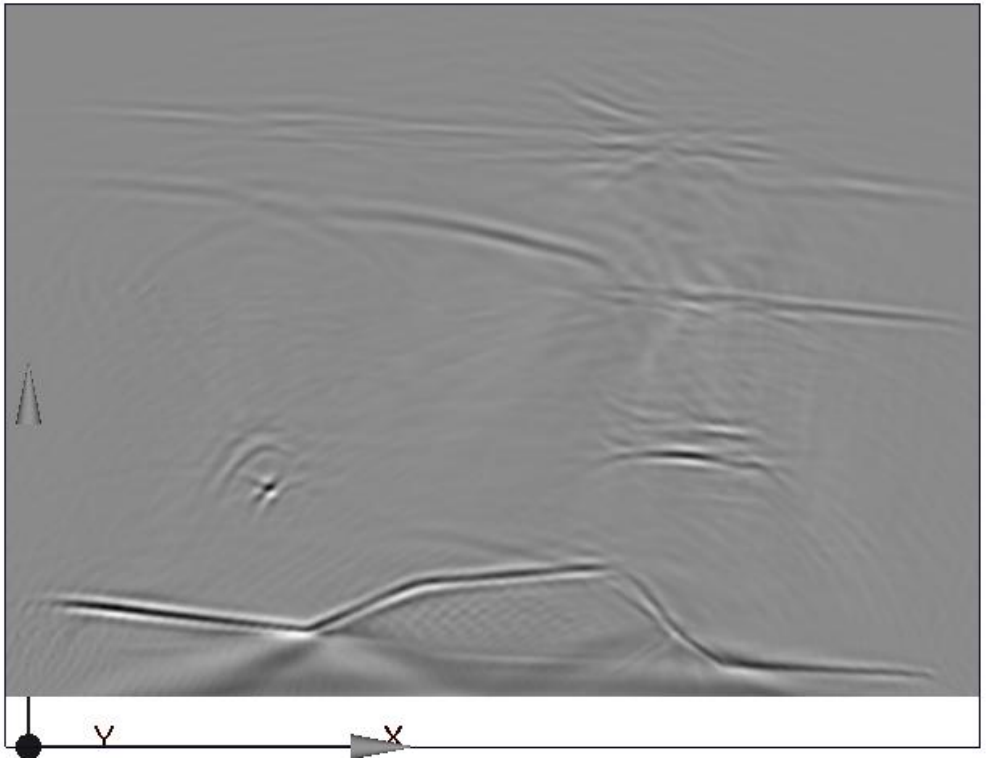
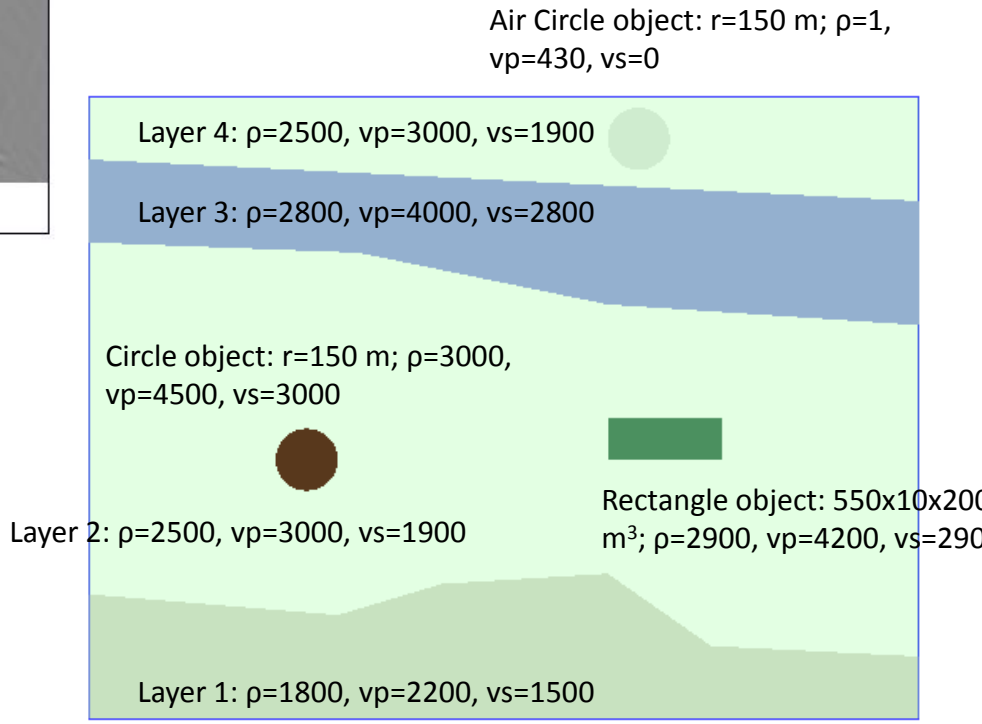


Image from WCT EL IMG solver

the ground true



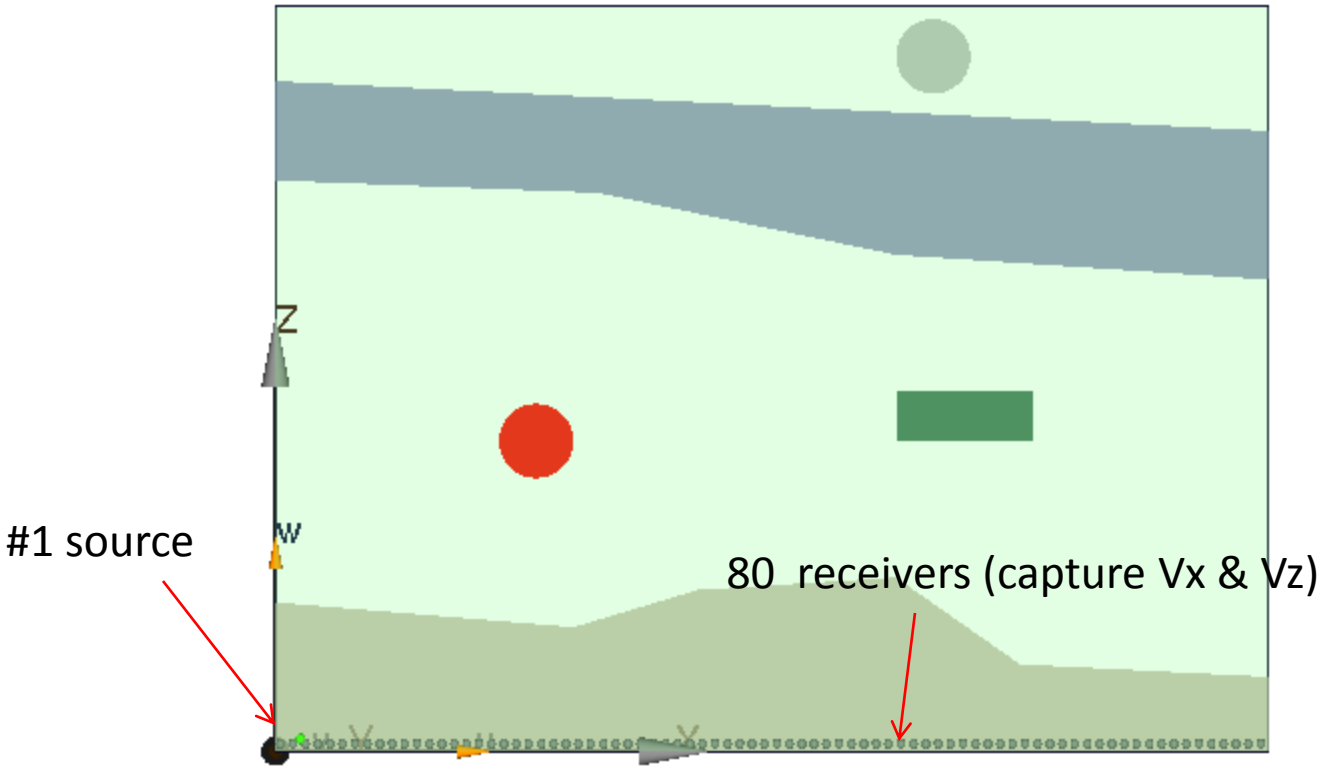
Extension part of Case II

Obtaining the detected Vx & Vz on sensor from the true space

➢ it can be obtained from real measurement or ***forward simulation***

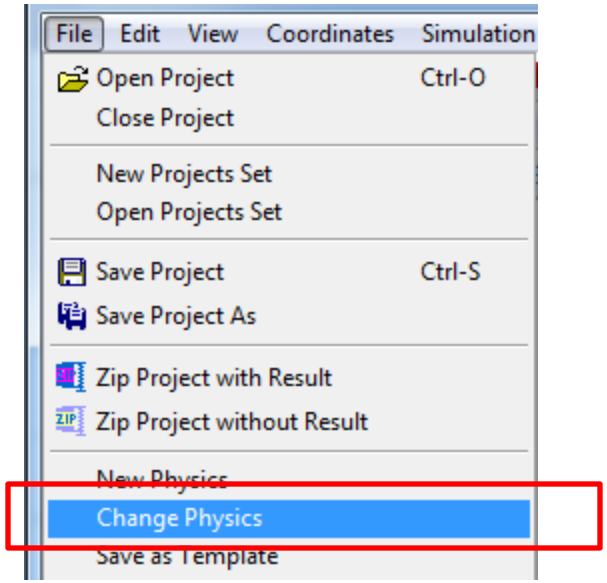
For example, if user want to use WCT EL solver to get the signal on sensors for source #1.

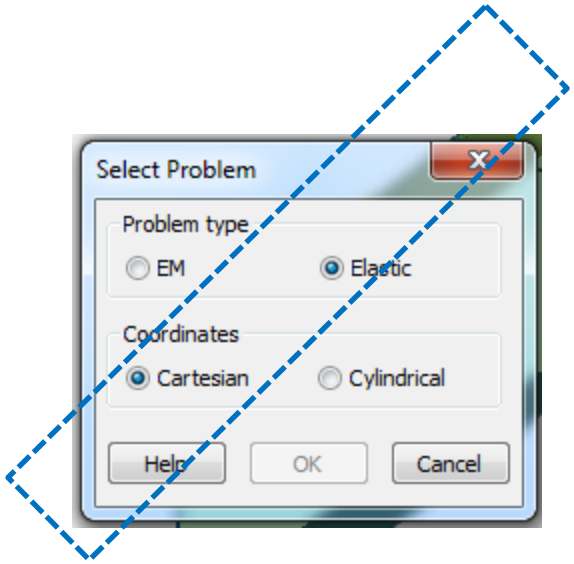
Setup a case as following



I. Setup WCT Cartesian EL project

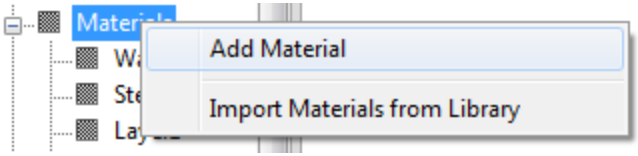
Note: This case is already build in the demo package as “xxxx\2D\sequential_t\39_sources_80recv_small_obj_v3\Forward\ Forward_Layered_2D_01.wnt”

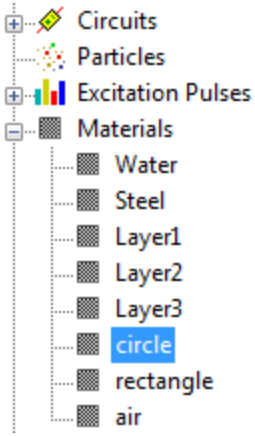


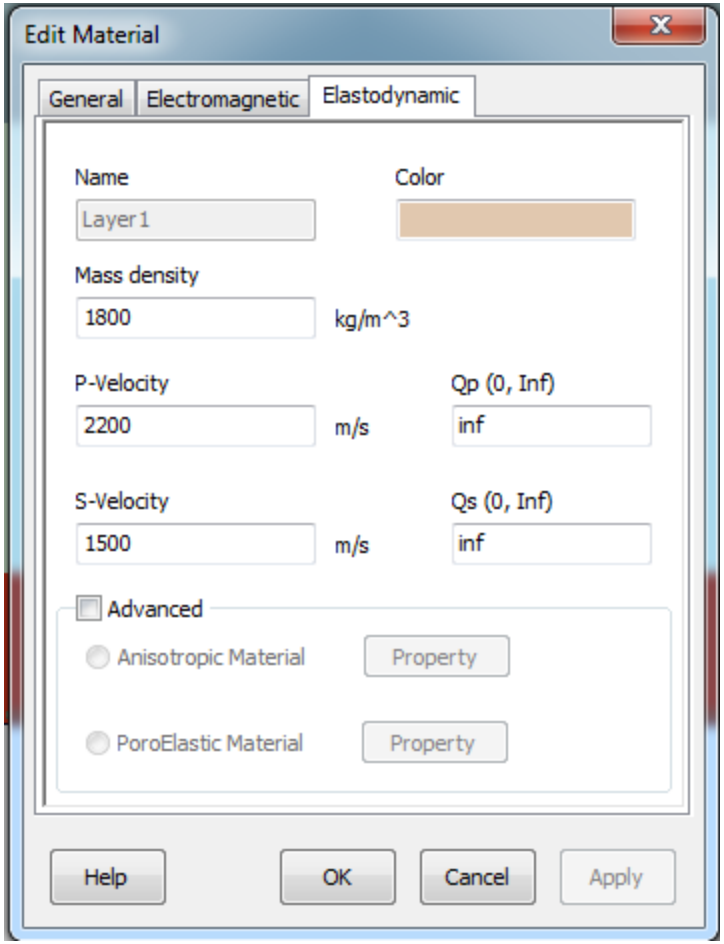


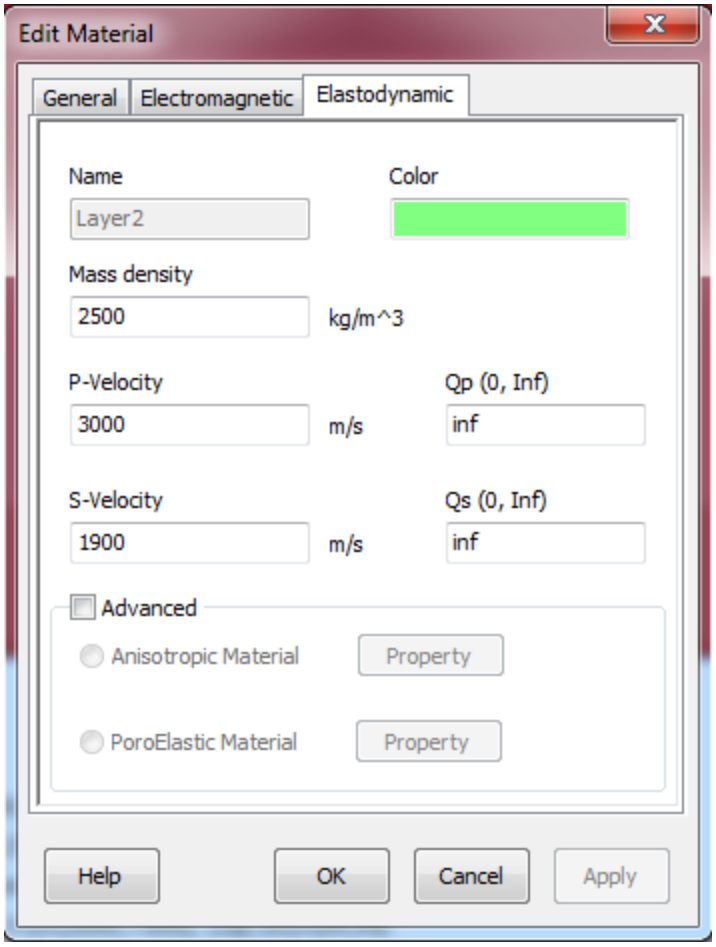
Then save as “Forward_Layered_2D_01.wnt”

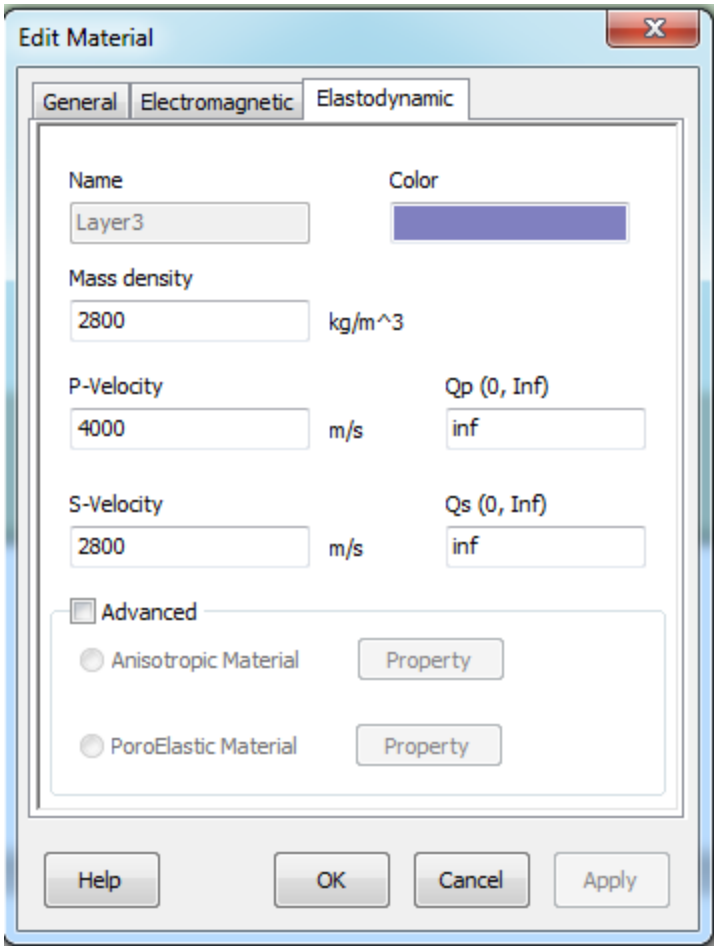
II. Setup materials

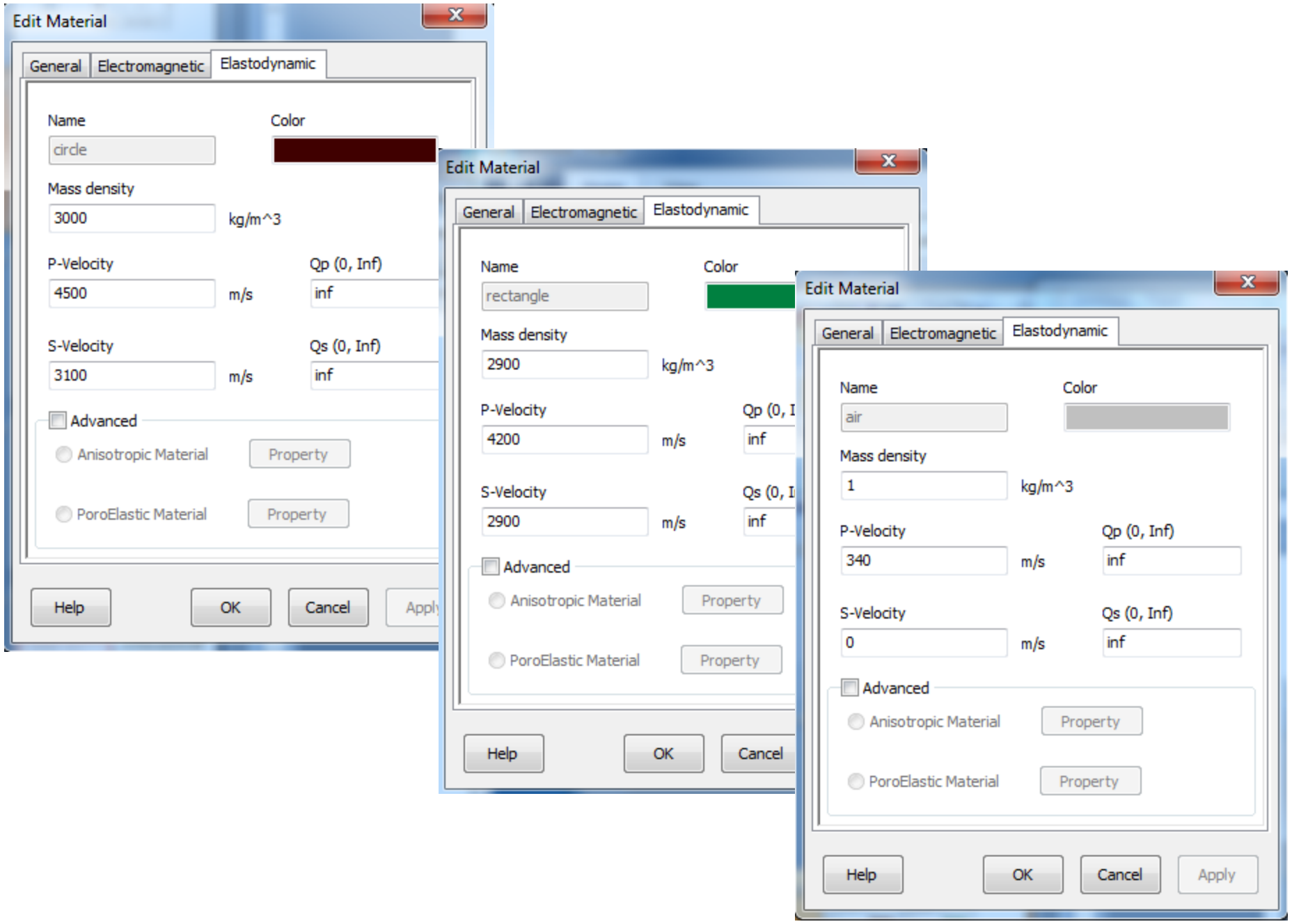




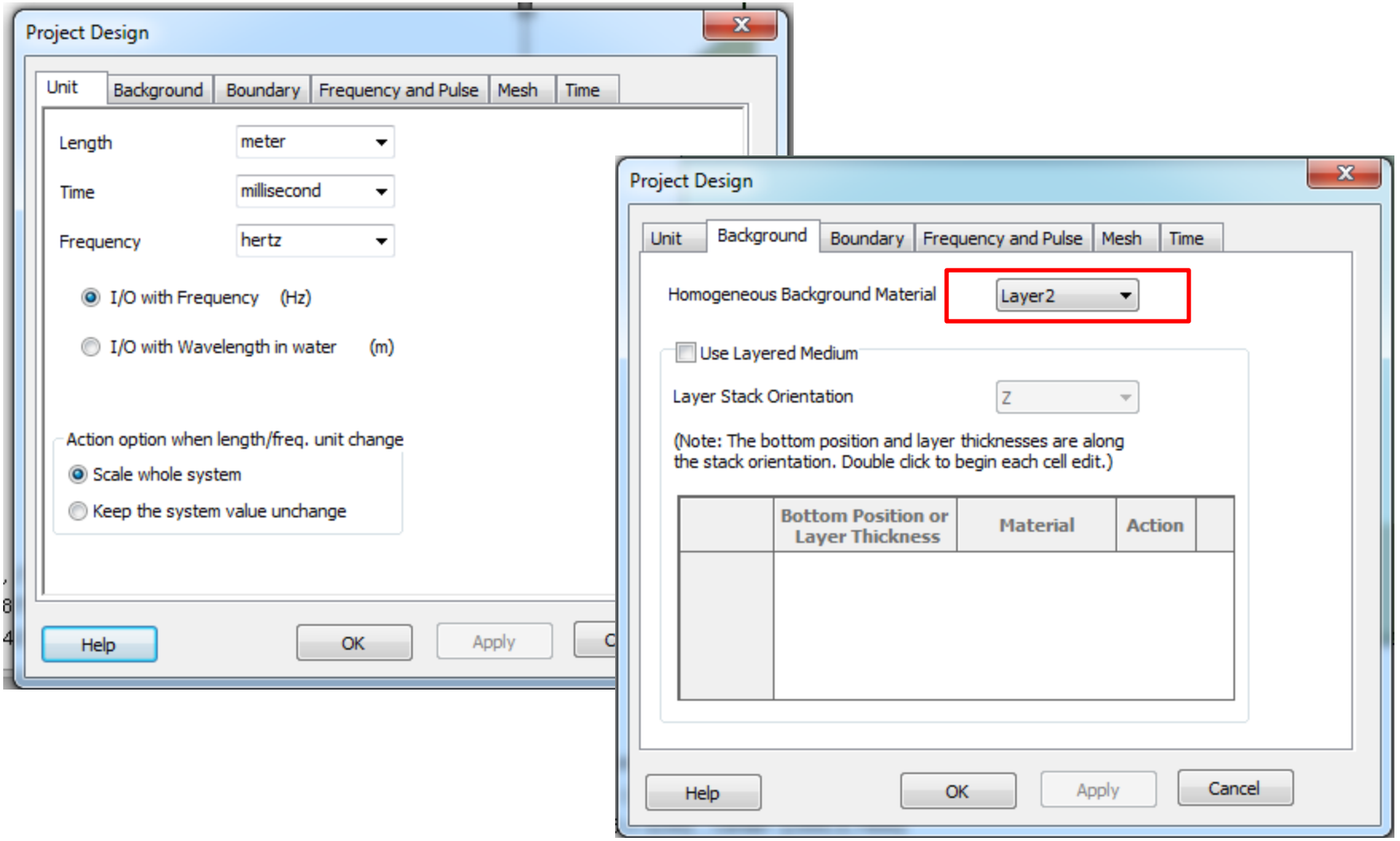


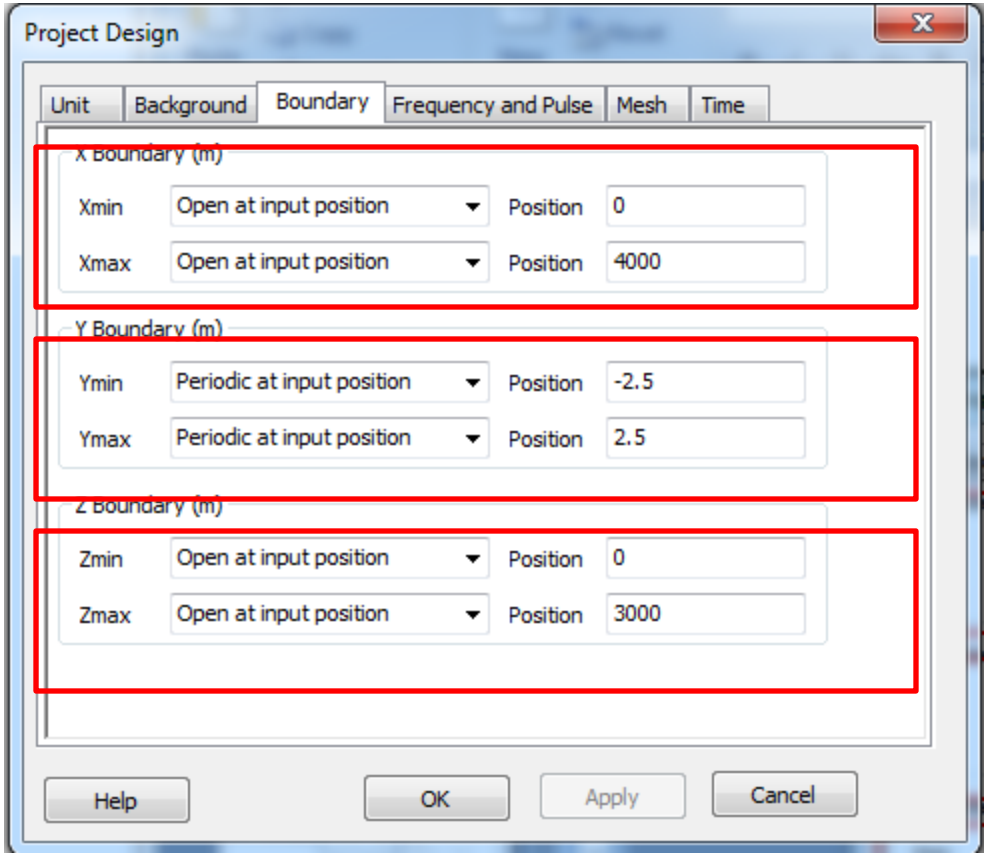

Edit Material
General
Electromagnetic
Elastodynamic
Name
Layer2
Color
Mass density
2500
kg/m^3
P-Velocity
3000
m/s
Qp (0, Inf)
inf
S-Velocity
1900
m/s
Qs (0, Inf)
inf
Advanced
Anisotropic Material
Property
PoroElastic Material
Property
Help
OK
Cancel
Apply


Edit Material
General
Electromagnetic
Elastodynamic
Name
Layer3
Color
Mass density
2800
kg/m^3
P-Velocity
4000
m/s
Qp (0, Inf)
inf
S-Velocity
2800
m/s
Qs (0, Inf)
inf
Advanced
Anisotropic Material
Property
PoroElastic Material
Property
Help
OK
Cancel
Apply

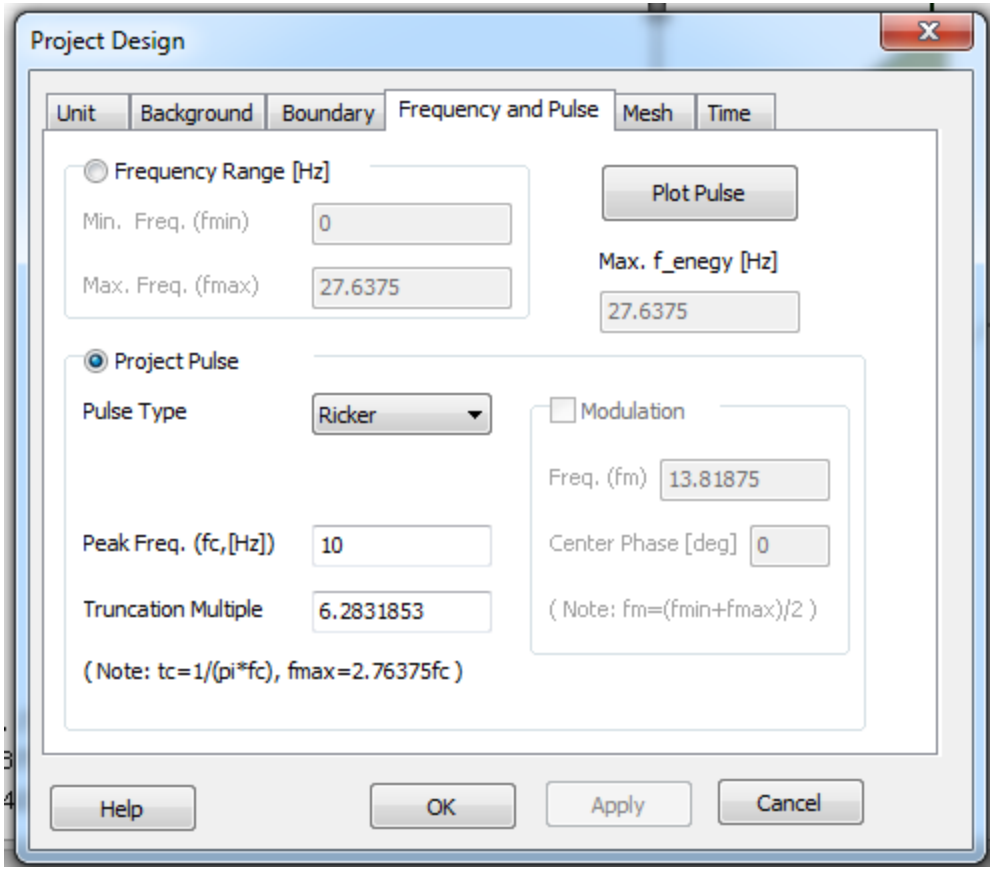

Edit Material
General
Electromagnetic
Elastodynamic
Name
circle
Color
Mass density
3000
kg/m^3
P-Velocity
4500
m/s
Qp (0, Inf)
inf
S-Velocity
3100
m/s
Qs (0, Inf)
inf
Advanced
Anisotropic Material
Property
PoroElastic Material
Property
Help
OK
Cancel
Apply
Edit Material
General
Electromagnetic
Elastodynamic
Name
rectangle
Color
Mass density
2900
kg/m^3
P-Velocity
4200
m/s
Qp (0, I
inf
S-Velocity
2900
m/s
Qs (0, I
inf
Advanced
Anisotropic Material
Property
PoroElastic Material
Property
Help
OK
Cancel
Edit Material
General
Electromagnetic
Elastodynamic
Name
air
Color
Mass density
1
kg/m^3
P-Velocity
340
m/s
Qp (0, Inf)
inf
S-Velocity
0
m/s
Qs (0, Inf)
inf
Advanced
Anisotropic Material
Property
PoroElastic Material
Property
Help
OK
Cancel
Apply

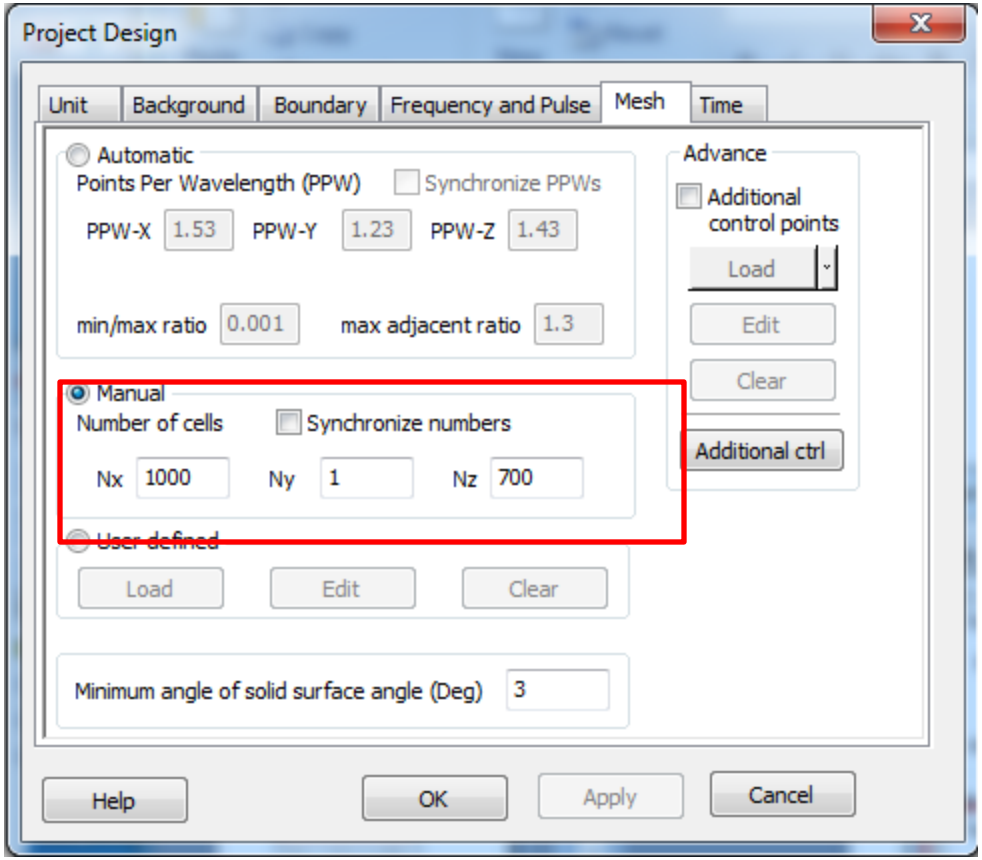
III. Setup project background, pulse, mesh & time system




Project Design
Unit
Background
Boundary
Frequency and Pulse
Mesh
Time
X Boundary (m)
Xmin
Open at input position
Position
0
Xmax
Open at input position
Position
4000
Y Boundary (m)
Ymin
Periodic at input position
Position
-2.5
Ymax
Periodic at input position
Position
2.5
Z Boundary (m)
Zmin
Open at input position
Position
0
Zmax
Open at input position
Position
3000
Help
OK
Apply
Cancel

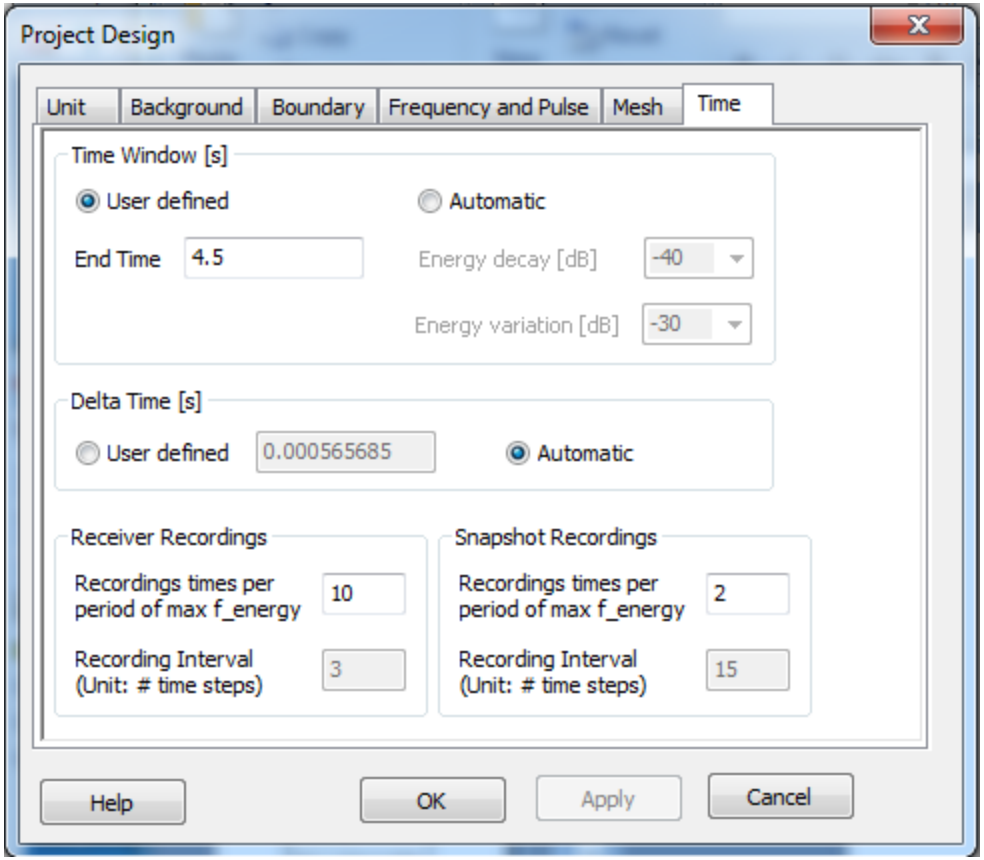
Source pulse is the Ricker wave with fc=10 Hz (eq. fmax=27.6 Hz)





➢ due to the slow Vp of air, the ppw in air object will be low. However, due to the air object is small compared to the whole space, we still consider this simulation sampling density is good because for most other V, the ppw will be larger than 7.

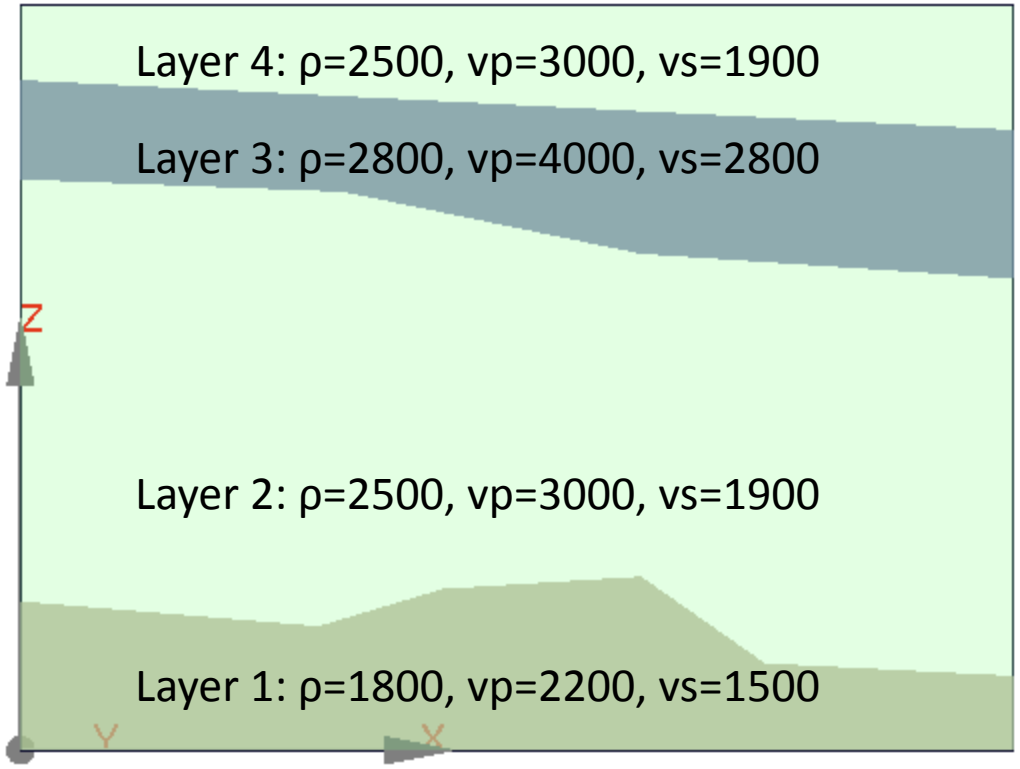
Define the time windows as 4.5 s and use automatic Δt.



From these project setups, we know the forward simulation has many difference from the imaging simulation setup

- project size
- mesh
- simulation Δt

IV. Setup 4 layers background



We can use following 3D geometry system to simplify the modeling

- ➢ homogeneous background by the material of layer 2
- ➢ layer 1 is modeled as a polygon cylinder
- ➢ layer 3 is modeled as a polygon cylinder also

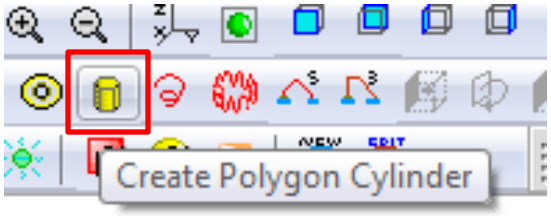
❑ due to the region outside the layer 1 & 3 will be the background, the layer 2 & 4 will be built automatically after layer 1 & 3 are built

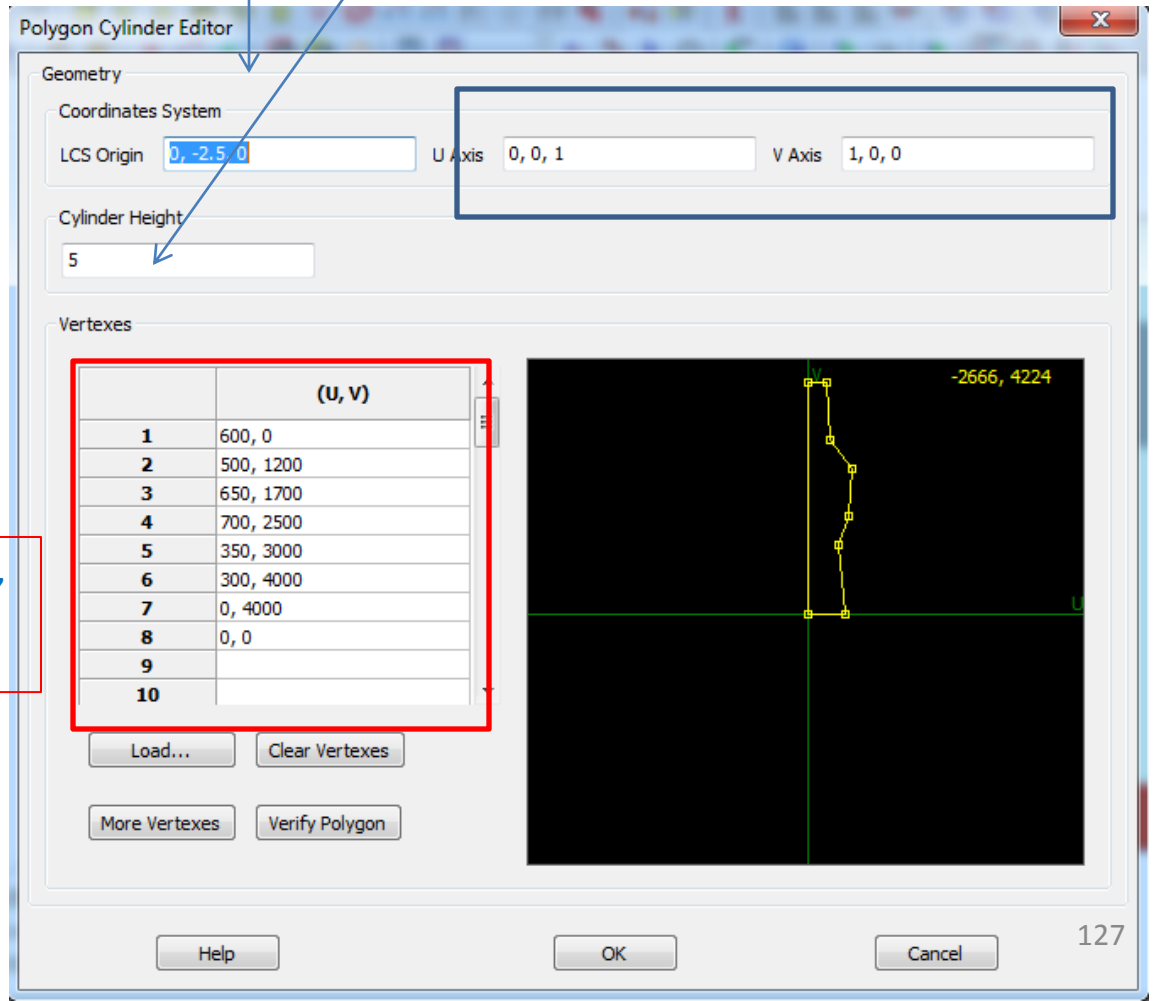
➢ Build layer 1

The cylinder will start from y=-2.5, so, enter (0,-2.5,0) as the LCS origin. Height is 5.

Due to the cylinder will grow along +Y in WCS, the U, V axis of LCS is as following, so, the W axis is WCS (0,1,0) based on U, V

Use this button to enter Polygon Cylinder Editor



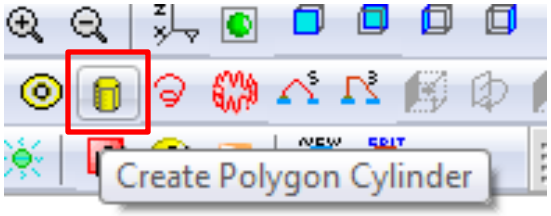


The vertices of polygon, here, as we know, U is WCS's Z, V is WCS's X

Set layer 1 using material "layer1"

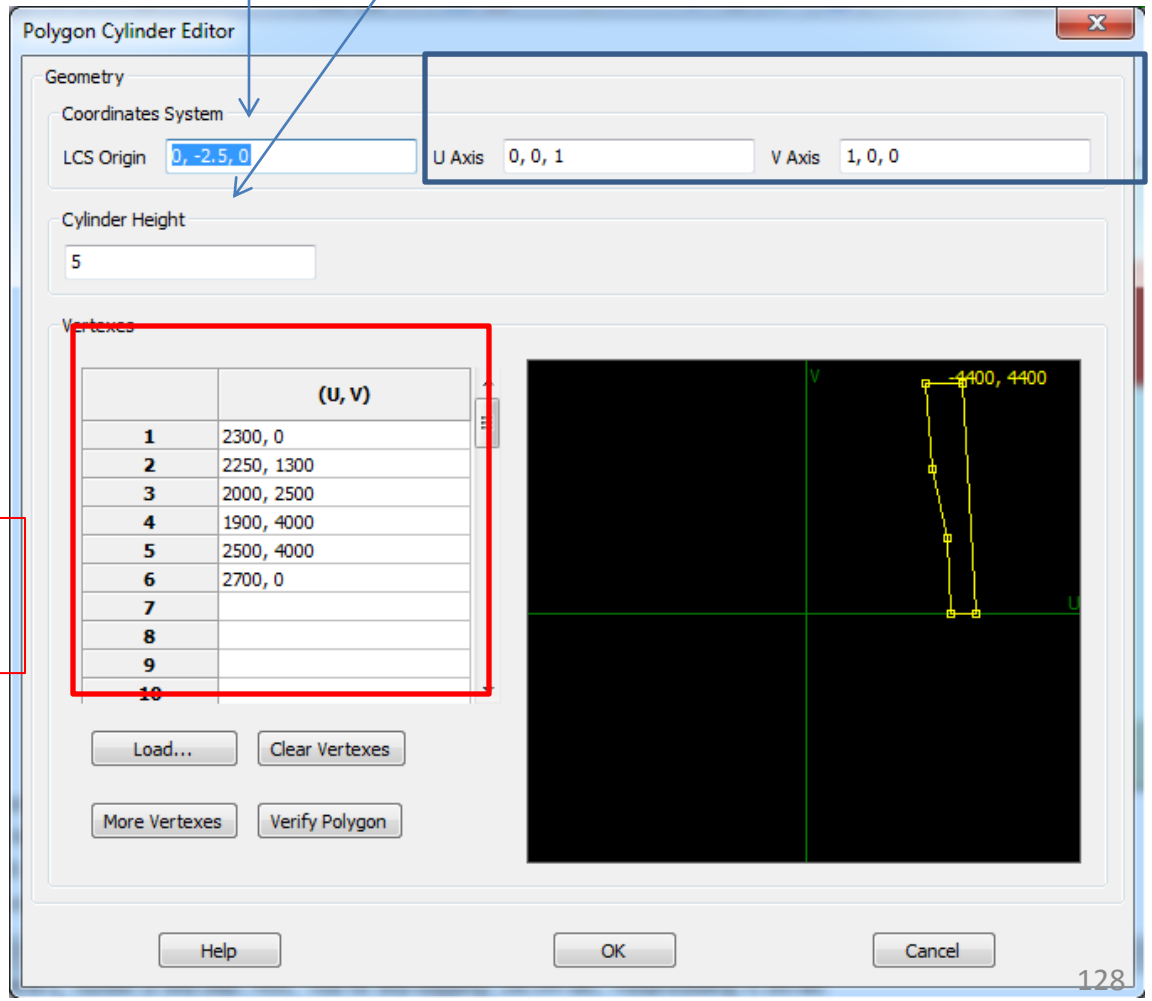
➢ Then layer 3

Use this button to enter Polygon Cylinder Editor



The cylinder will start from y=-2.5, so, enter (0,-2.5,0) as the LCS origin. Height is 5.

Due to the cylinder will grow along +Y in WCS, the U, V axis of LCS is as following, so, the W axis is WCS (0,1,0) based on U, V

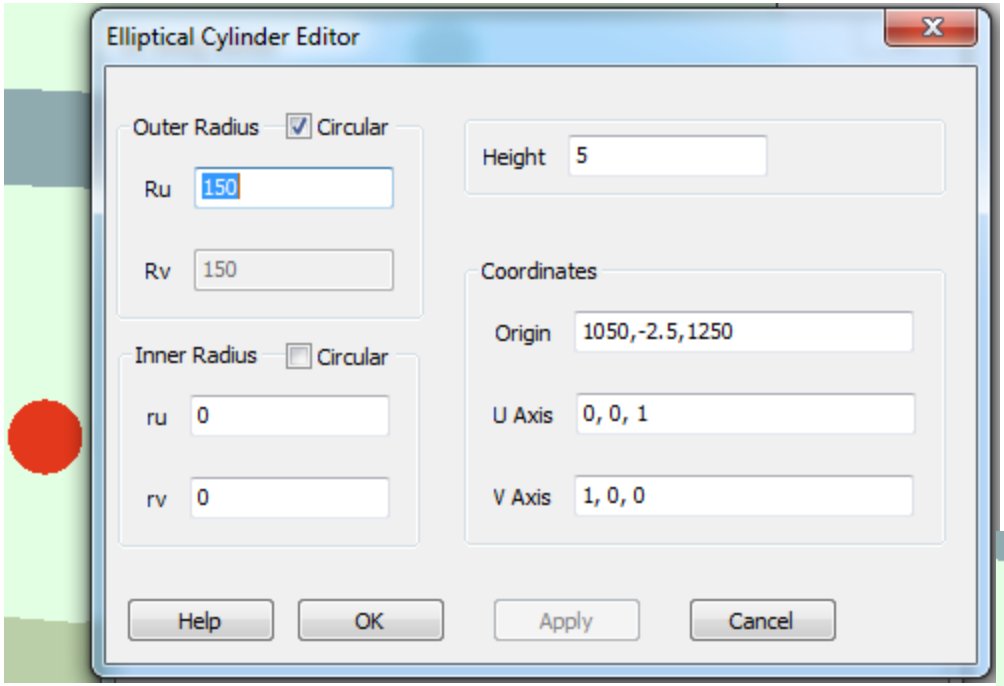


The vertices of polygon, here, as we know, U is WCS's Z, V is WCS's X

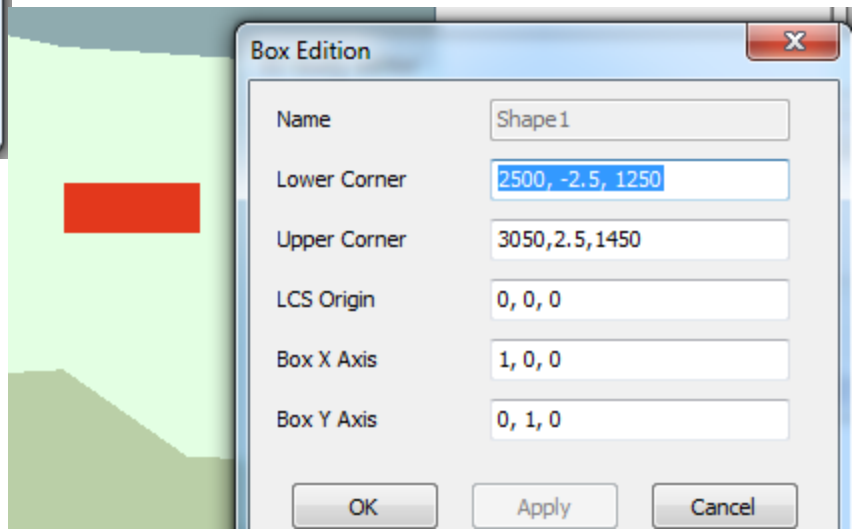
Set layer 3using material "layer3"

V. Build 3 small obejcts

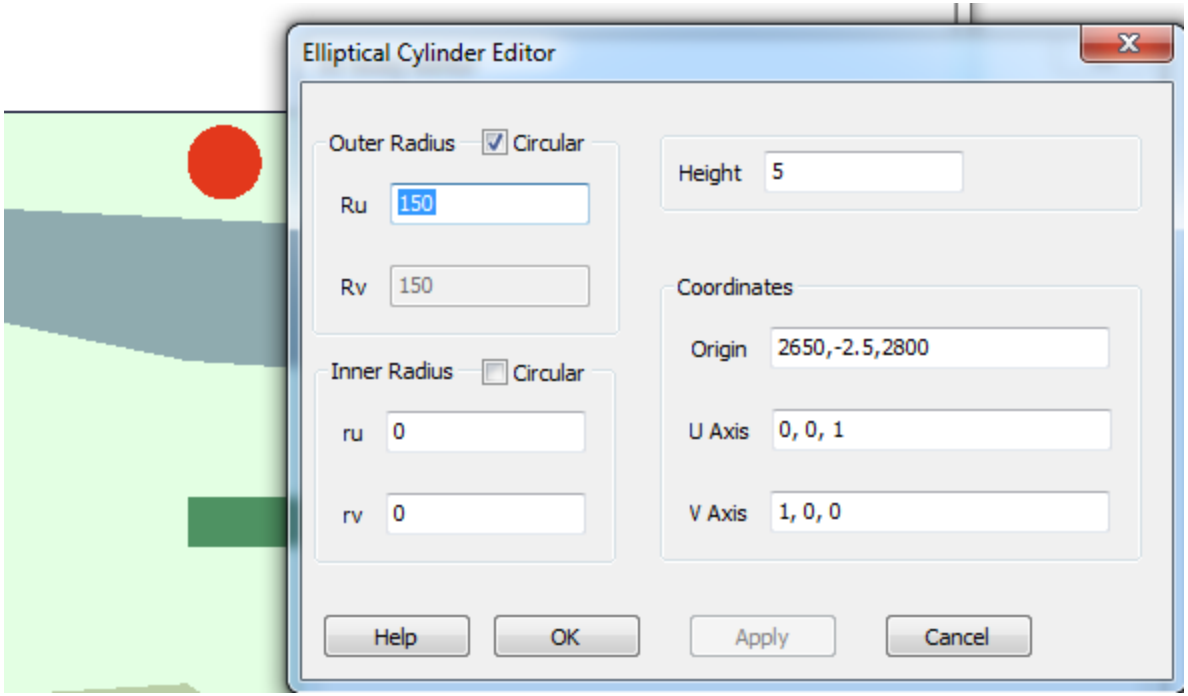
Use material “cirle”



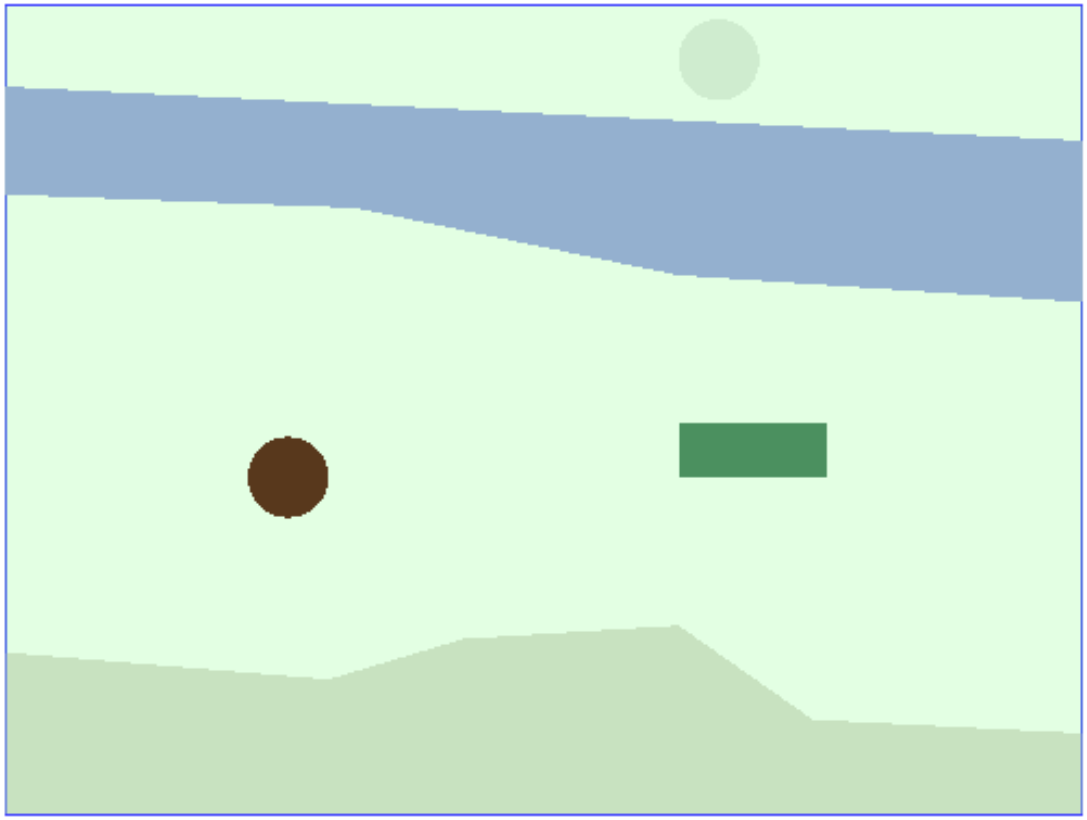
Use material “rectangle”



Use material “air”

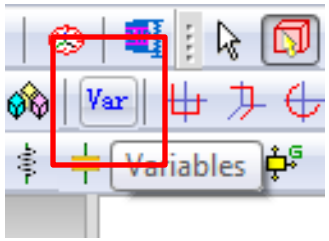


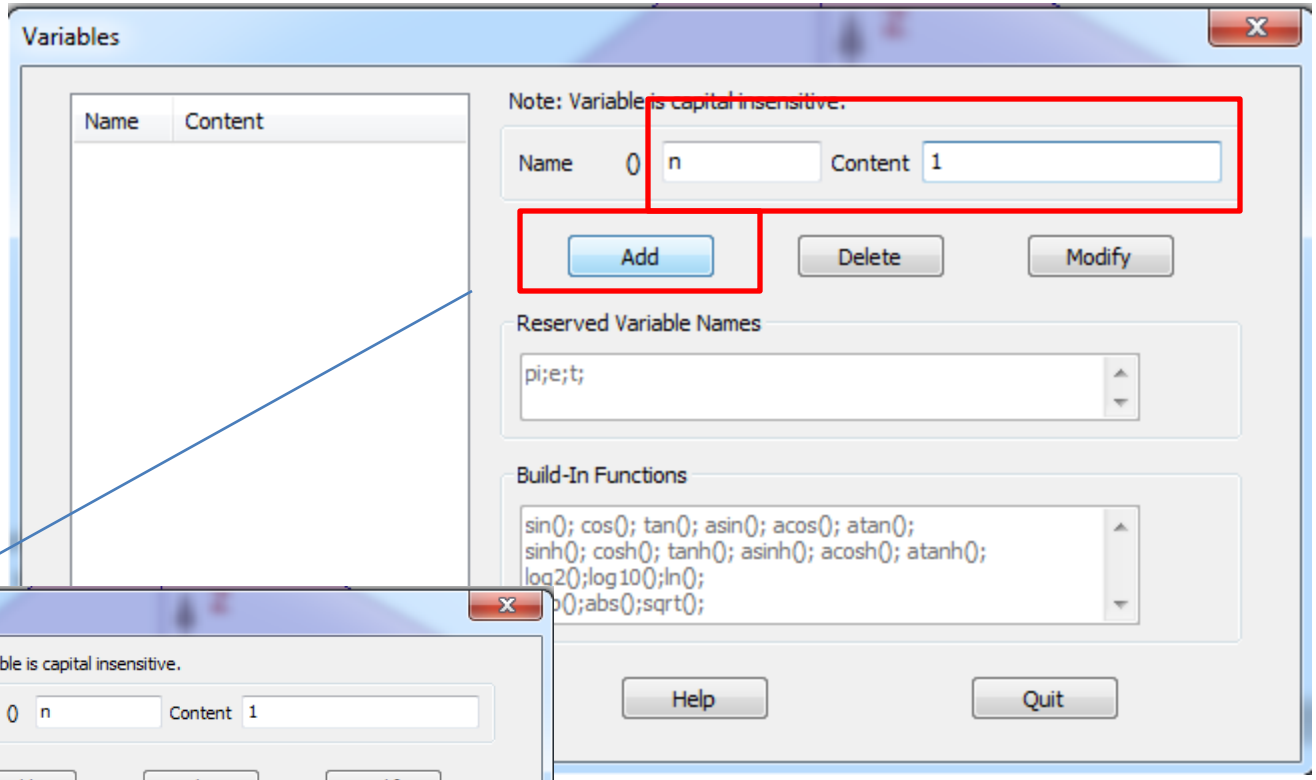
After geometry system are finished, the system layout will look like following

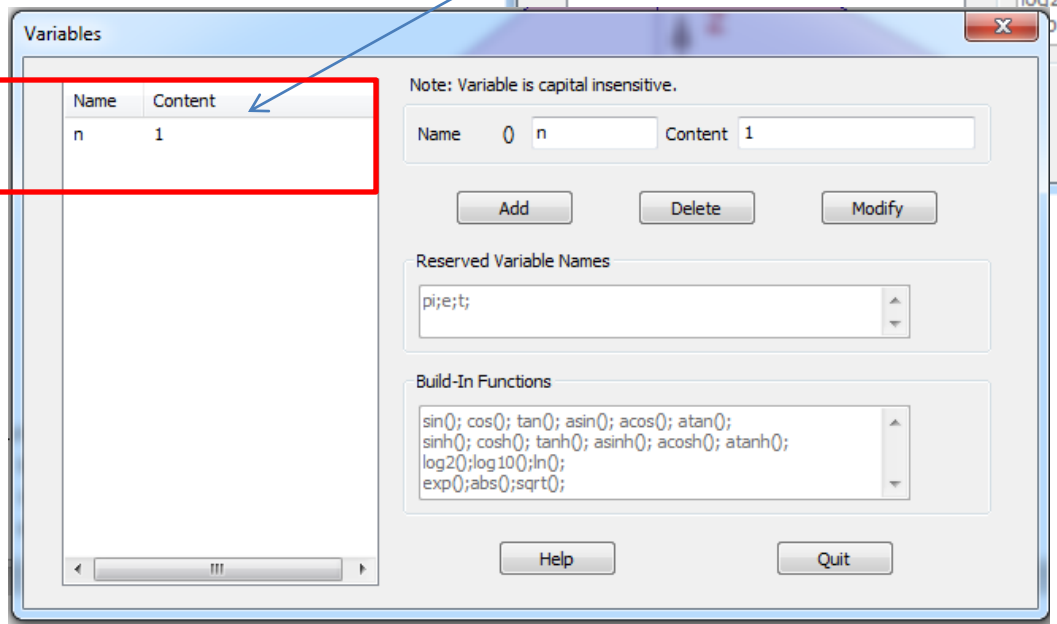


VI. Setup sources with variable

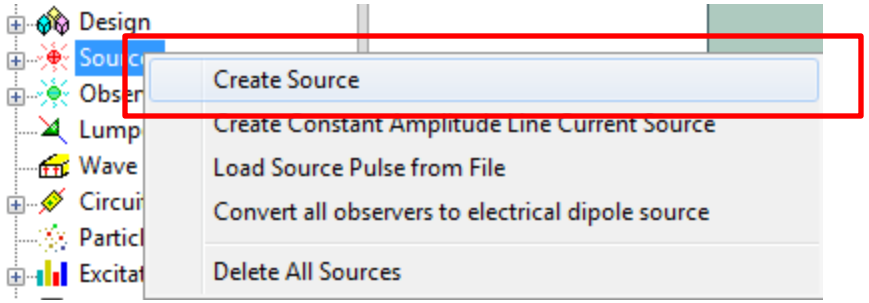
Define a variable ***n***





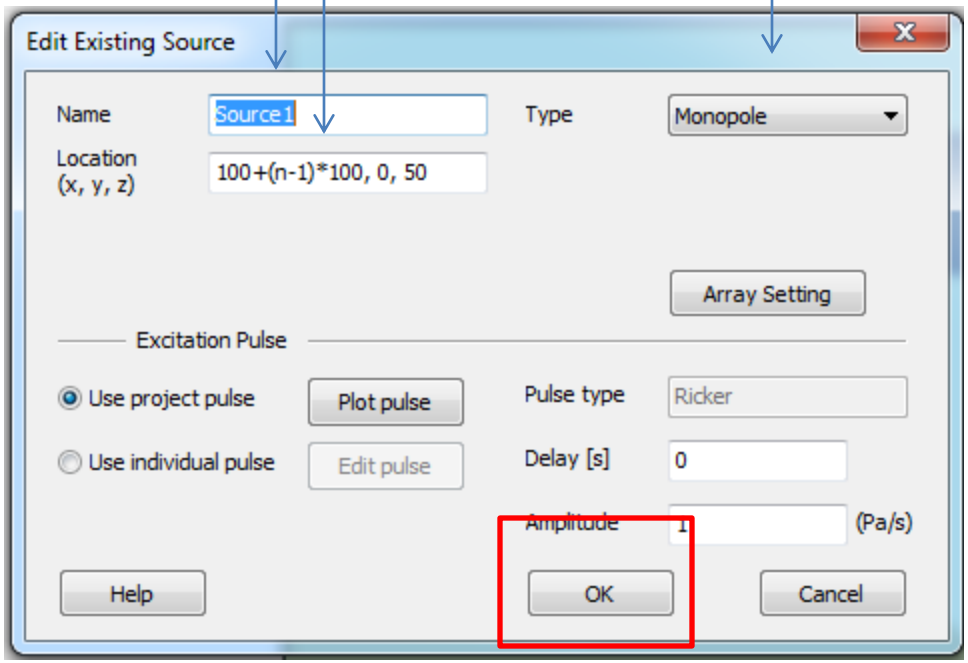


Defien sources with variable “n”

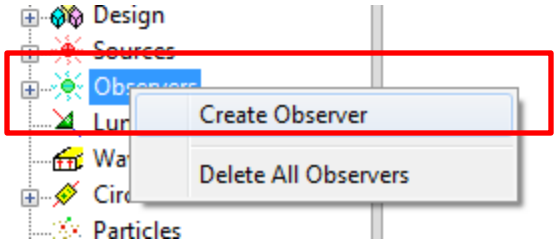


Input

- the source name
- source position
- source type

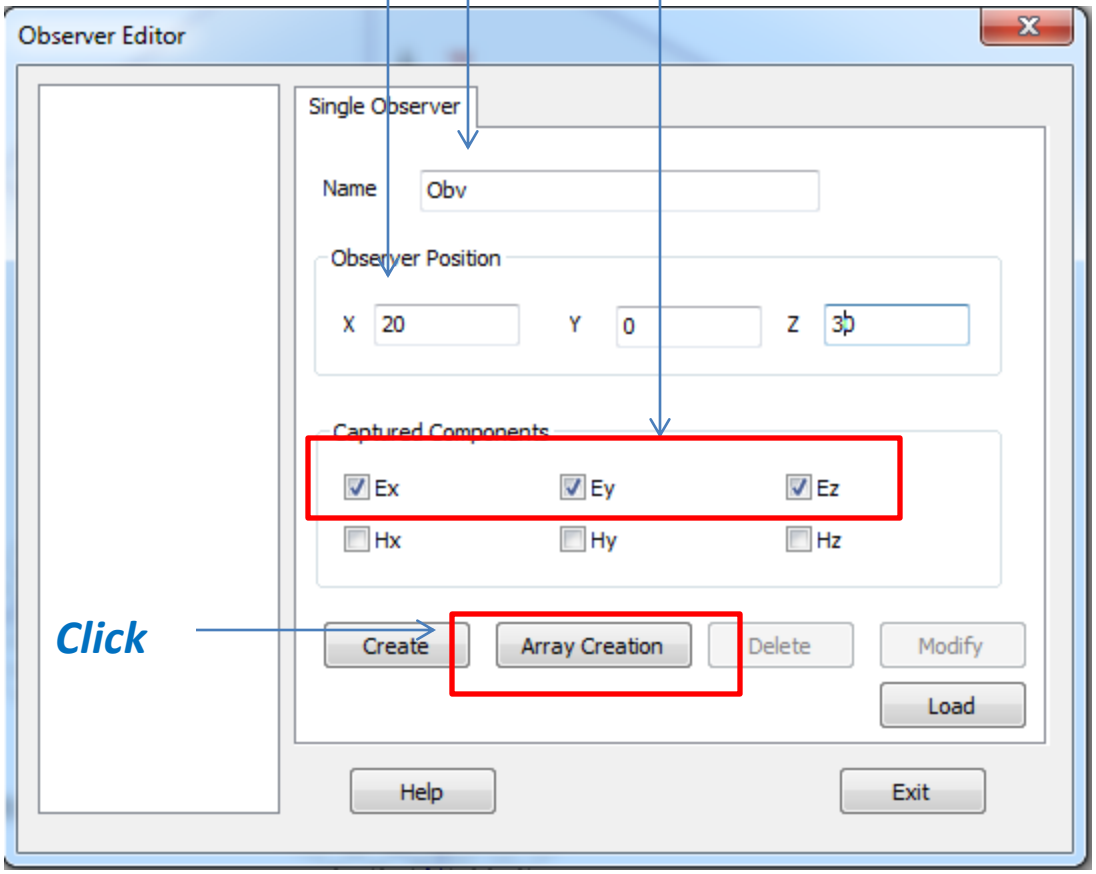


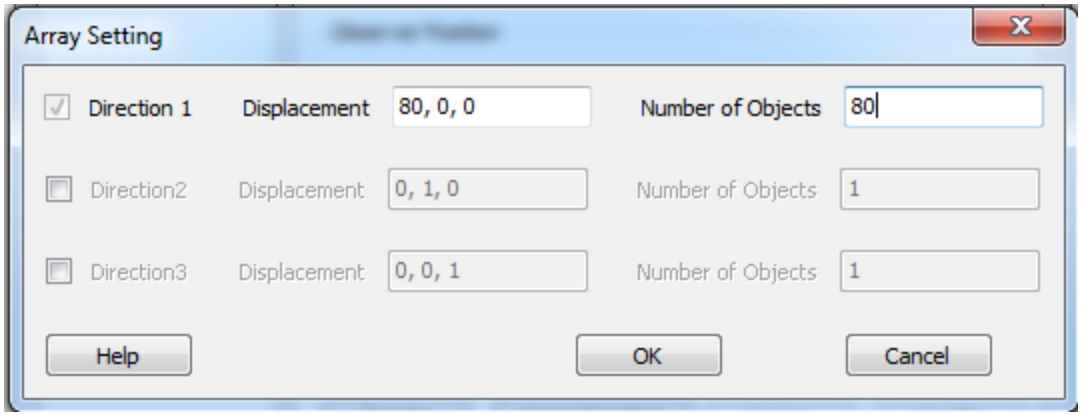
VI. Setup 80 receivers by “Array Creation”



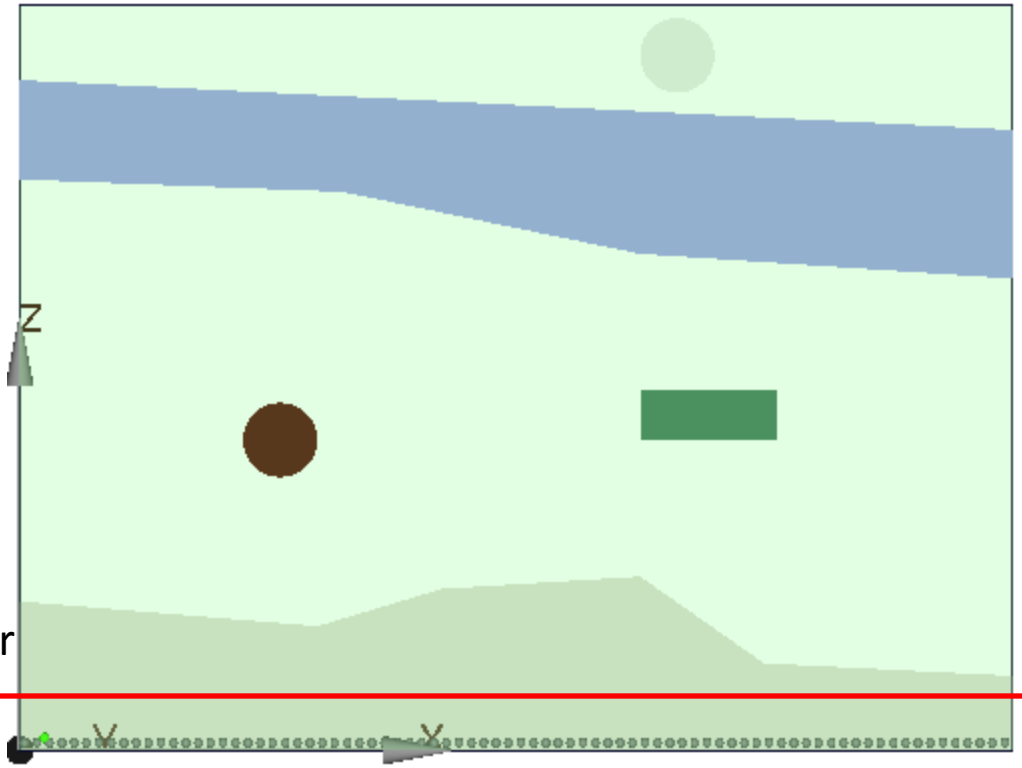
Input

- the receiver name prefix
- first receiver position
- record component





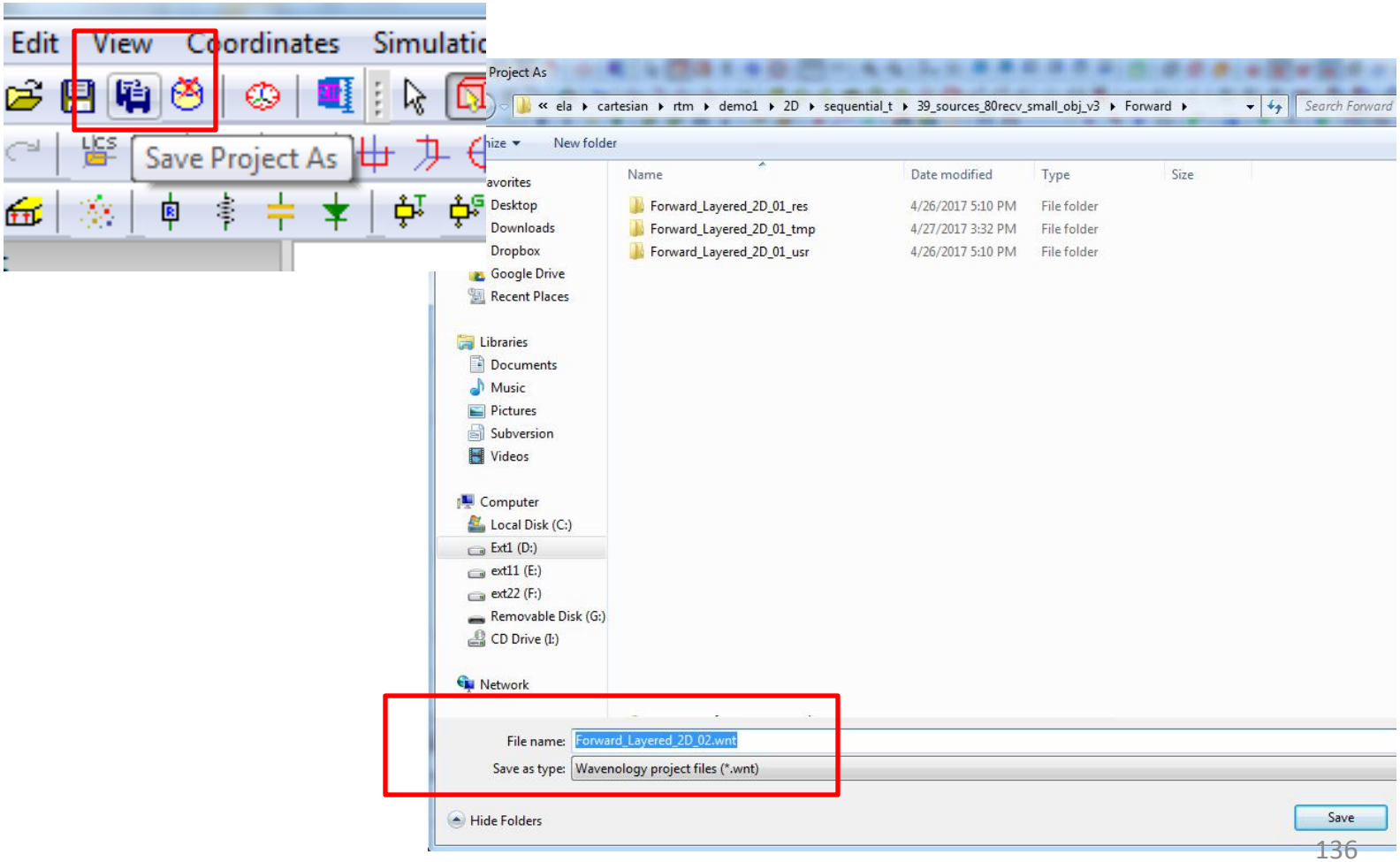
After source & receiver are created, the project layout will be as following



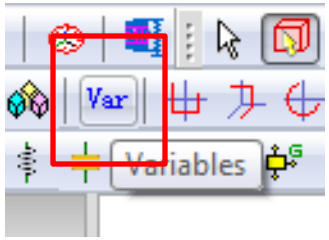
source & receiver

VII. Build other project for source #2 to #39

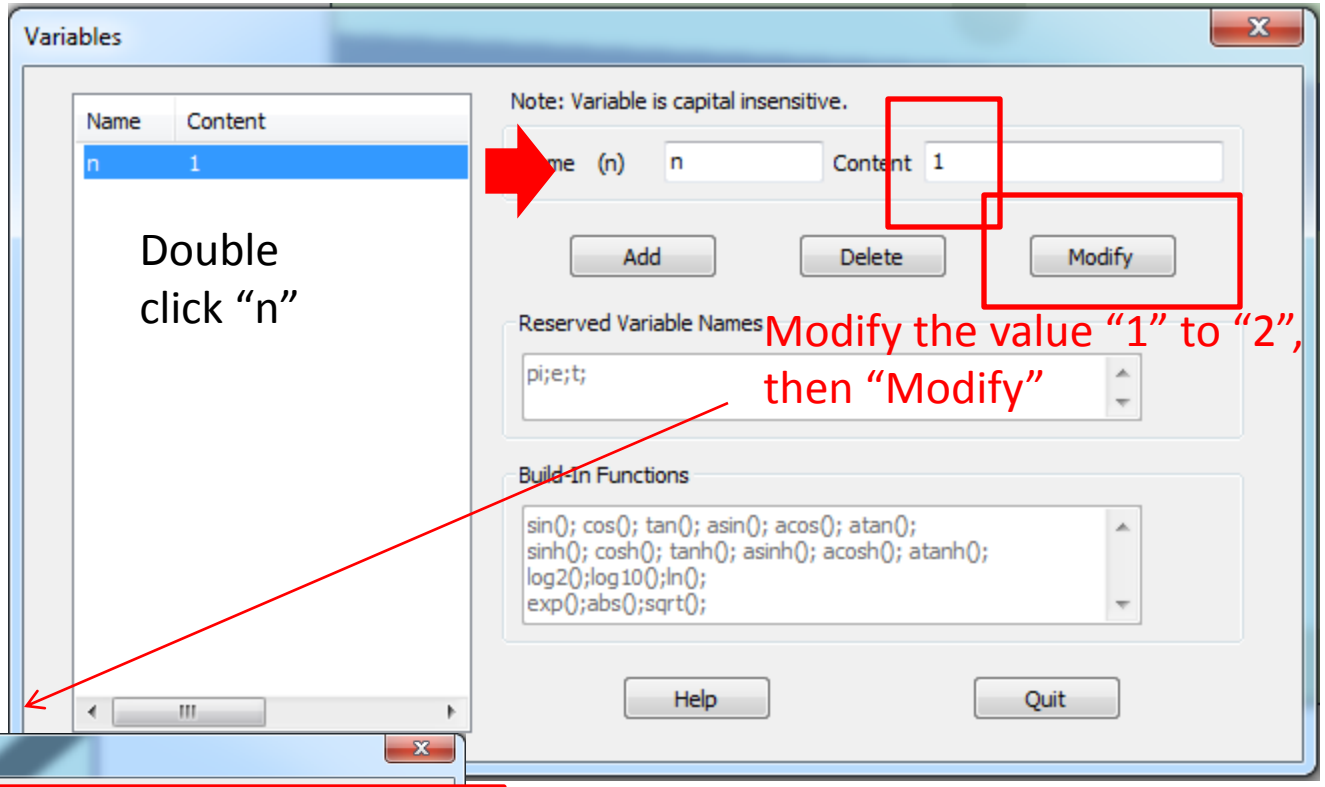
- save current project “Forward_Layered_2D_01.wnt” as “Forward_Layered_2D_02.wnt”

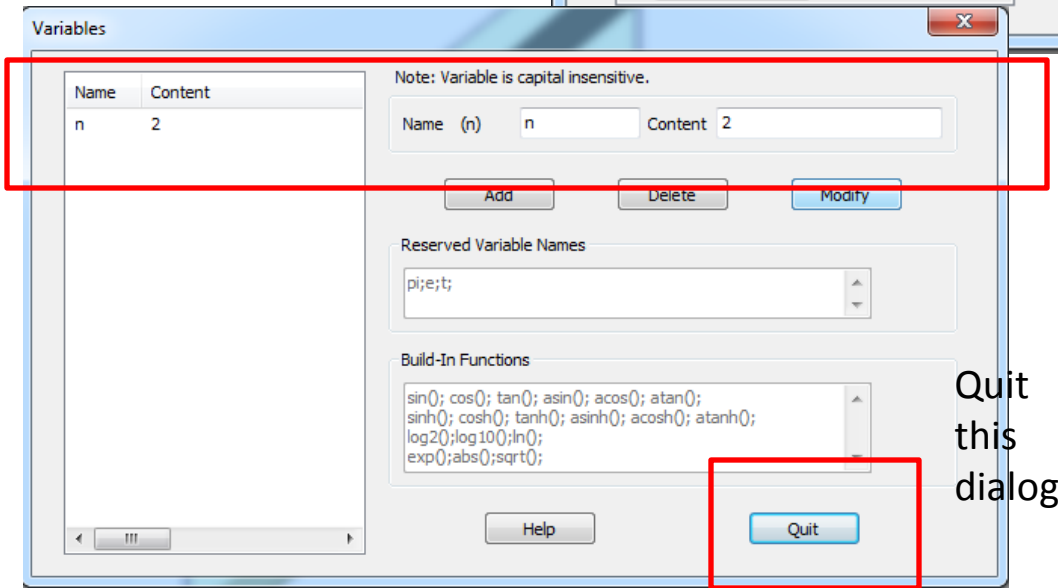


Modify the value for variable ***n***



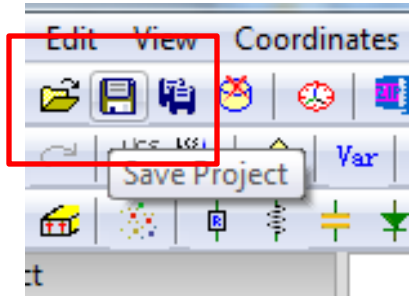
Let “n” shown here



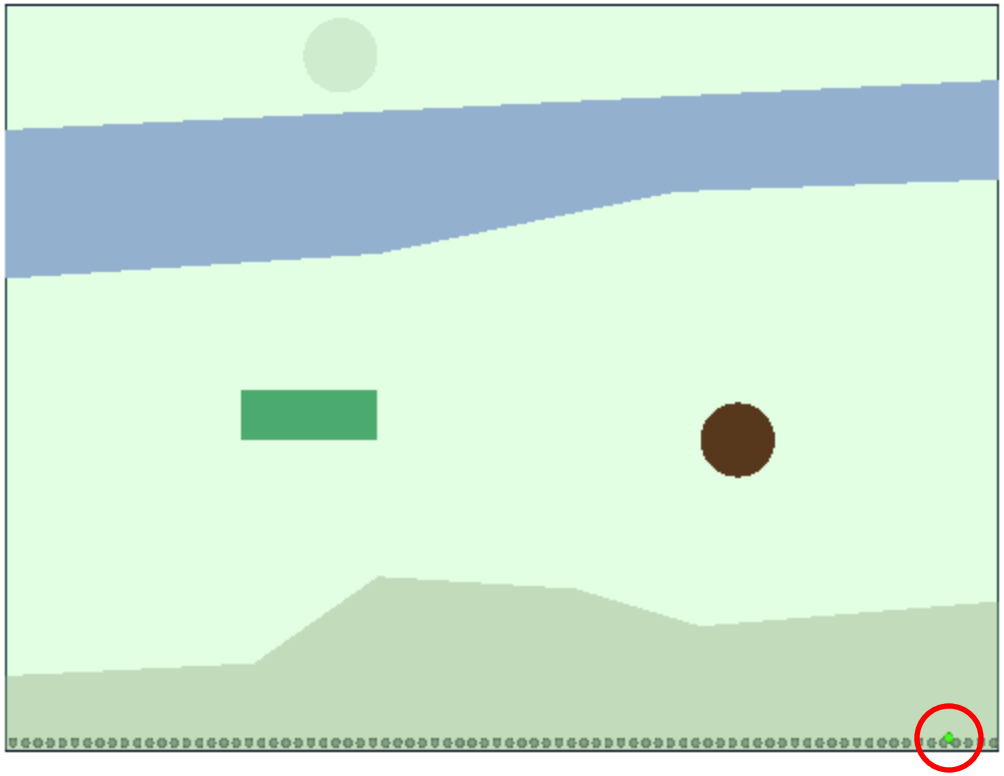


Quit this dialog

Save this project



You will see the source position is moved to x=200 m

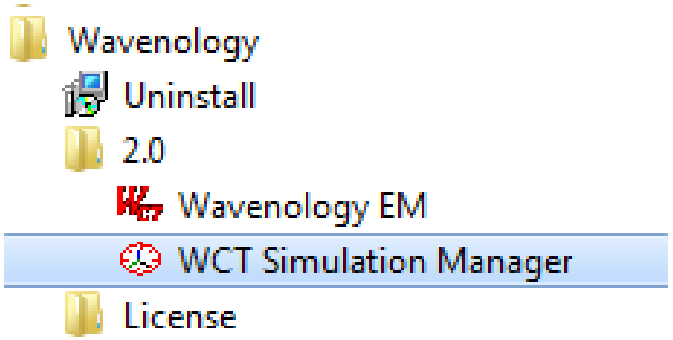


Repeat this process to create all 39 projects for 39 source positions.

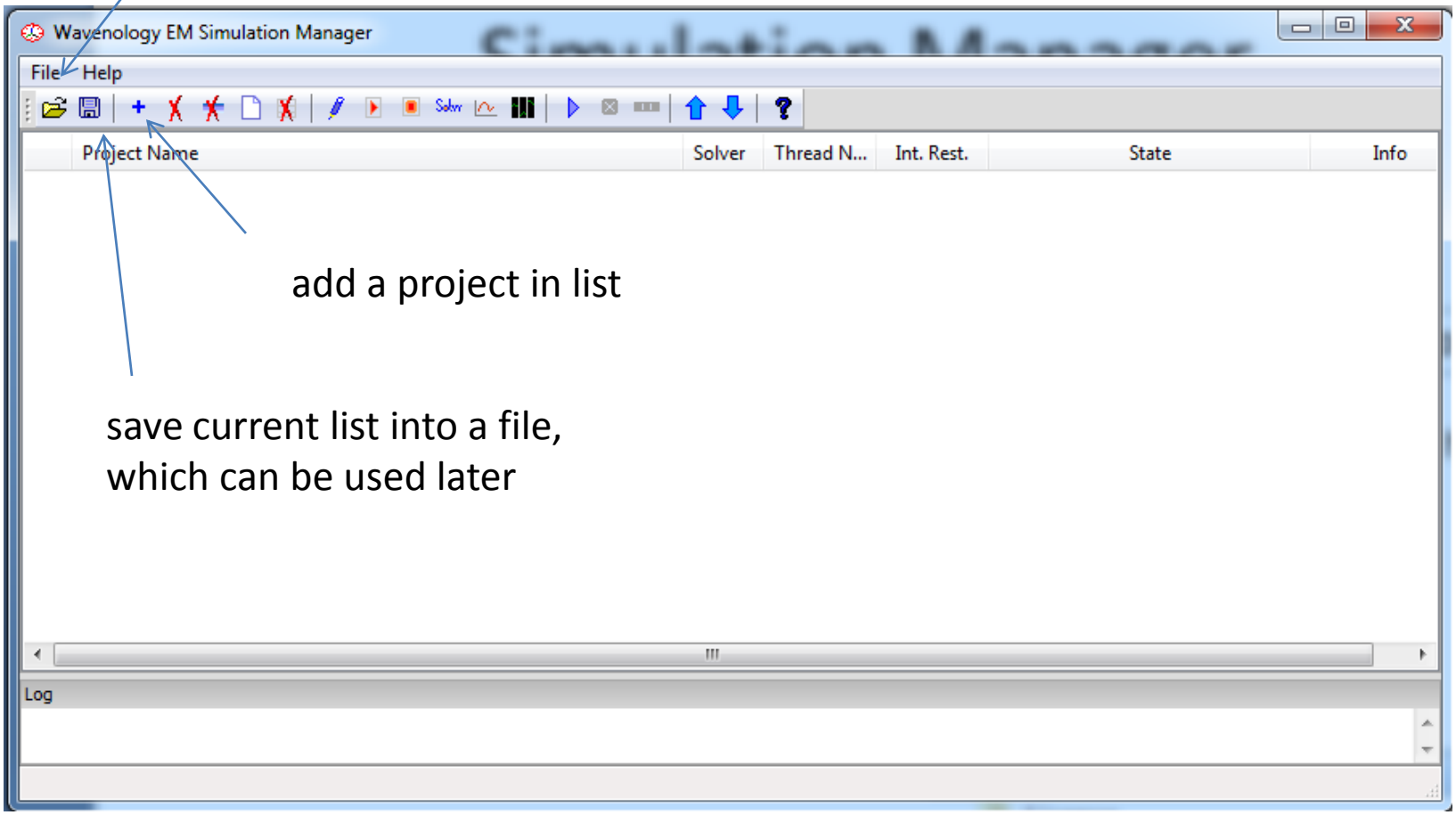
➢ simulation these 39 projects through ***WCT simulation manager***

In WCT GUI, use this button to start a simulation manager

Or, in Windows menu, expand here to start a simulation manager

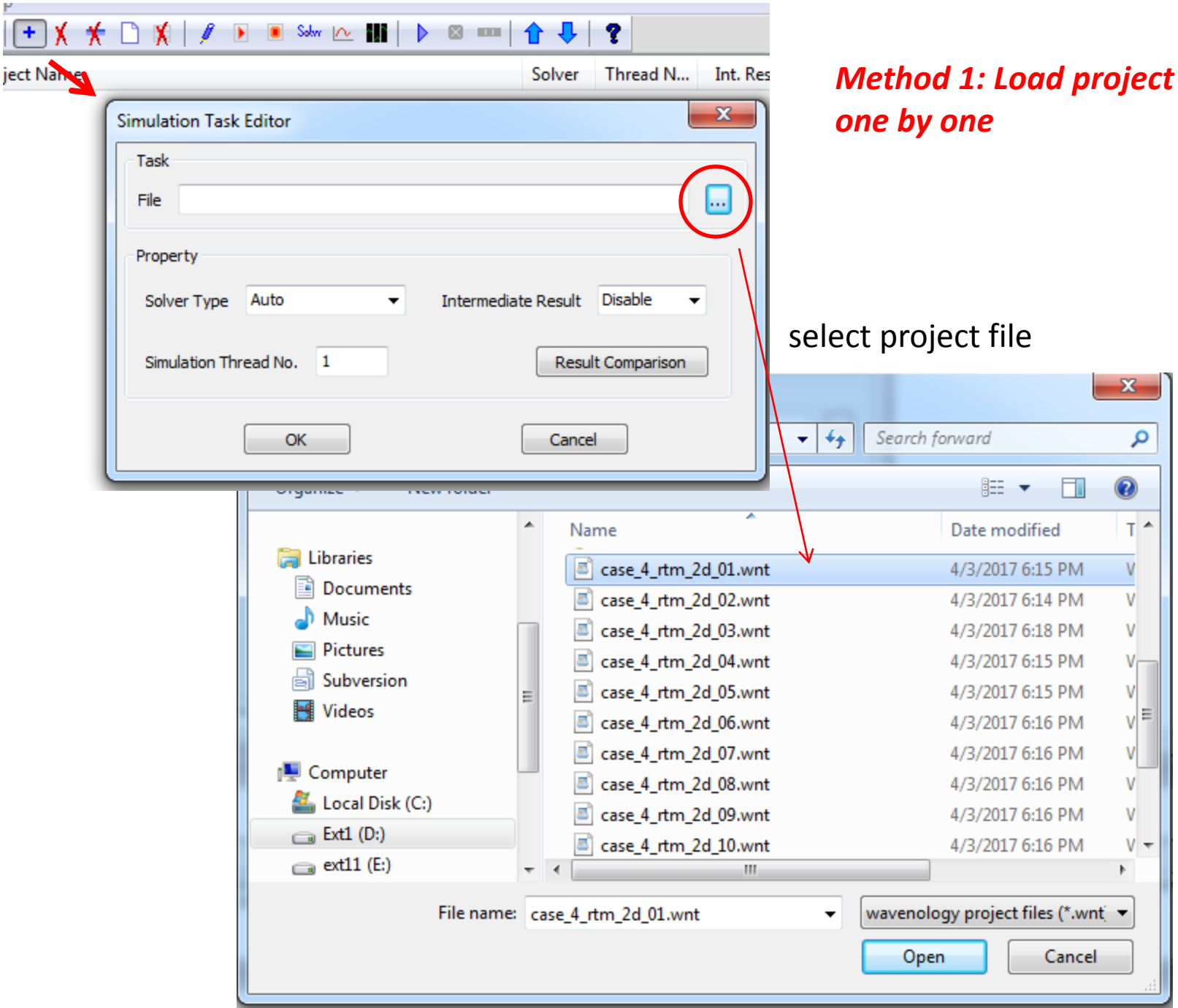


load saved list file or batch-link file



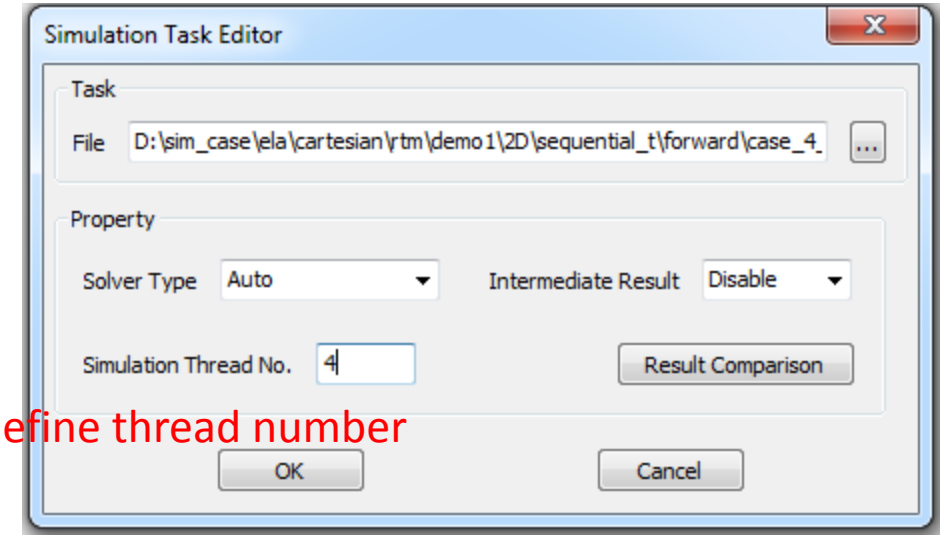
add a project in list

save current list into a file,
which can be used later



Method 1: Load project one by one

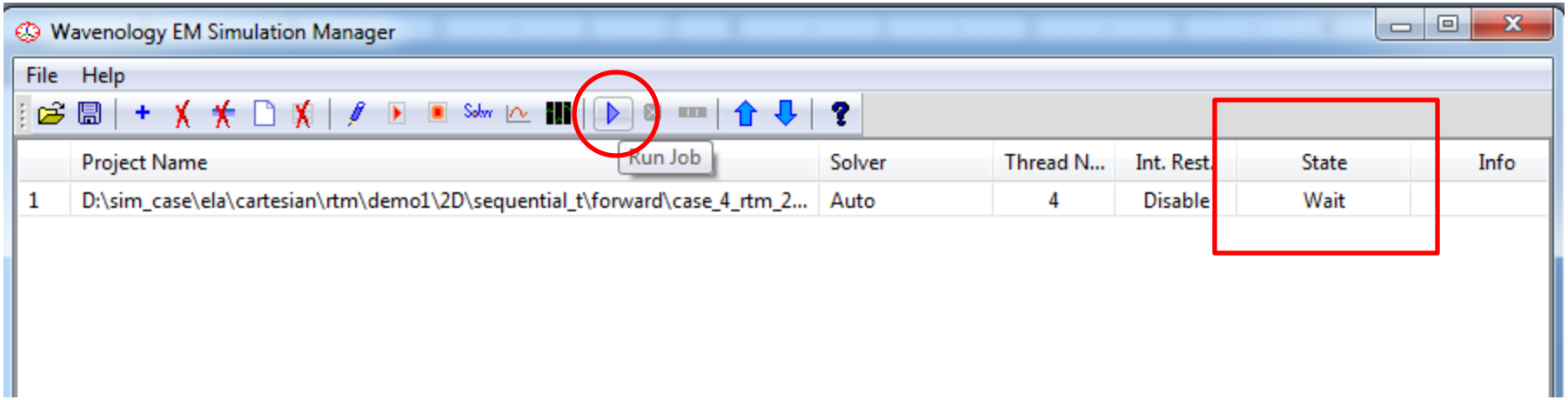
select project file



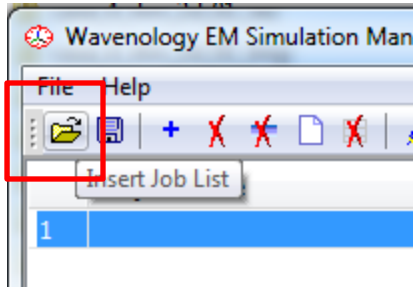
define thread number

Click here to run the project in the list

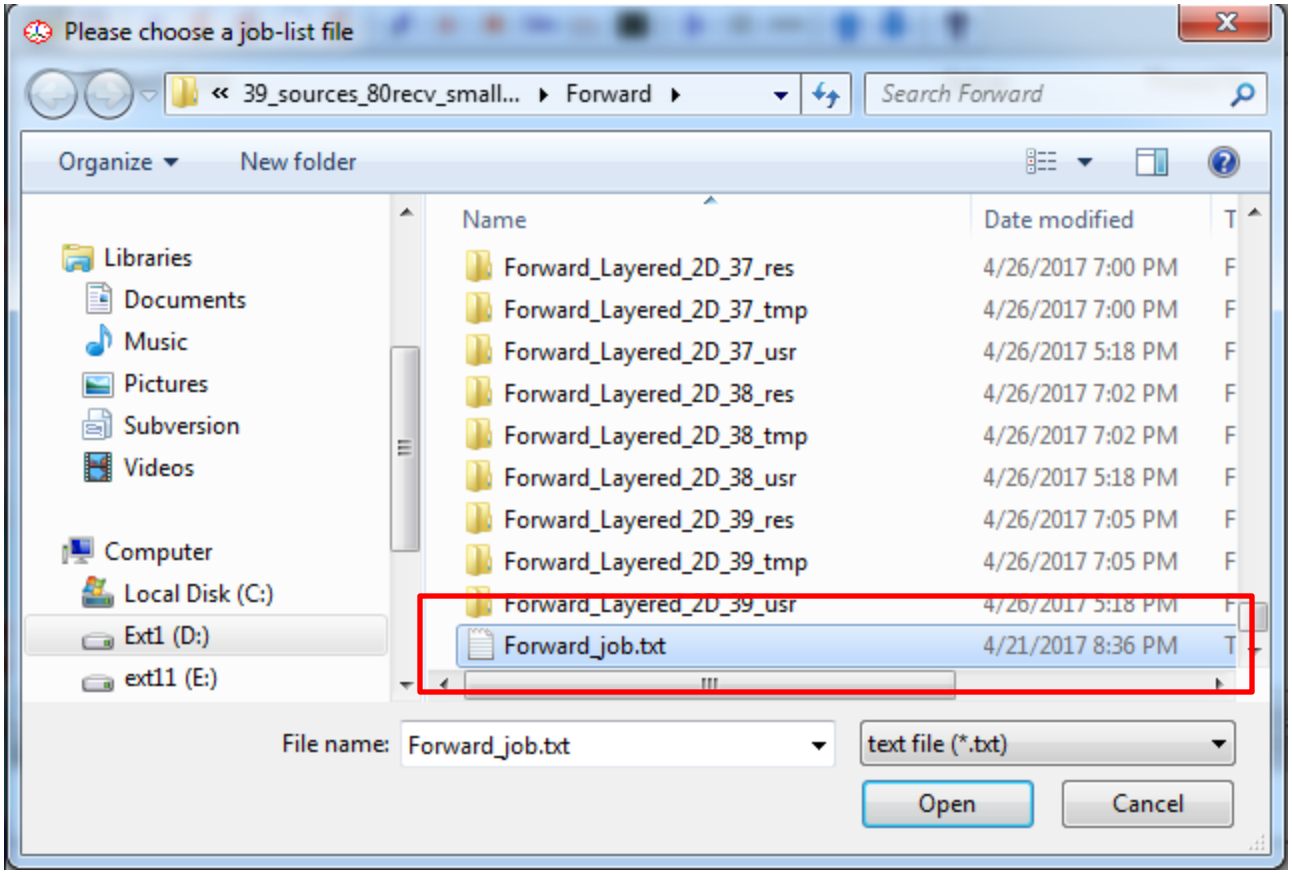
The project status is listed here to shown whether it is finished or not



Method 2: Load all 39 projects through a batch job file and Run



There is already a batch file in demo folder: as “xxxx\2D\sequential_t\39_sources_80recv_small_obj_v3 \Forward\Forward_job.txt”

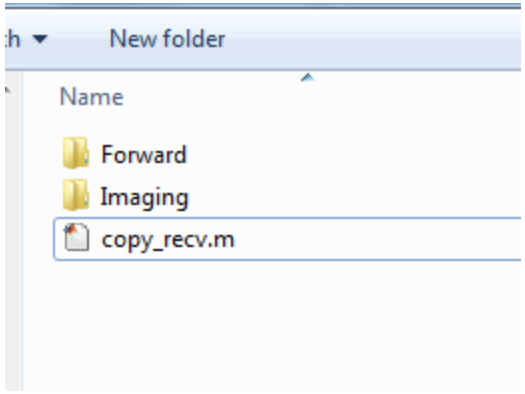


For more detail about WCT batch-link file format, please refer to WCT Cartesian EL manual

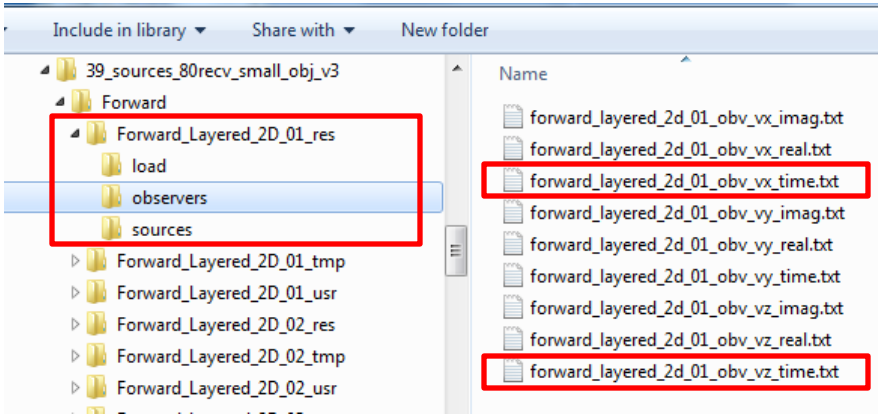
Forward_job.txt

```
[WCT Simulation Task List, ver=1.0]

wct_task_list_default_folder=[]

"Forward_Layered_2D_01.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_02.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_03.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_04.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_05.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_06.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_07.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_08.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_09.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_10.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_11.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_12.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_13.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_14.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_15.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_16.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_17.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_18.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_19.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_20.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_21.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_22.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_23.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_24.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_25.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_26.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_27.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_28.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_29.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_30.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_31.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_32.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_33.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_34.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_35.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_36.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_37.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_38.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
"Forward_Layered_2D_39.wnt" SIMTYPE=auto  SIMTHREAD=4 INTRESULT=0 STATUS=wait
```

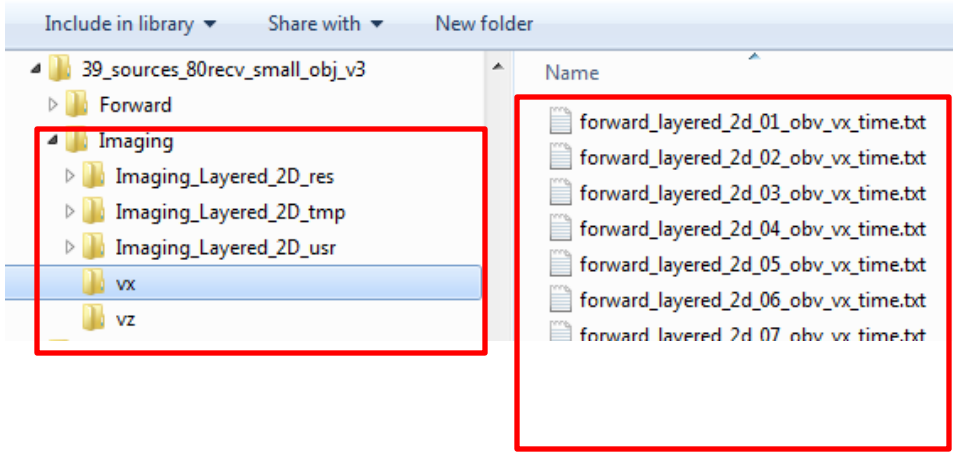
After all 39 projects are simulated successfully, copy all Vx & Vz results to the Imaging project's sub-folder "vx" & "vz", we already provide a Matlab code "copy_recv.m" to work this purpose. In order to let the code work correctly, user need to create sub-folder "vx" & "vz" in the imaging project root folder firstly.



Vx & Vz signals for source #1



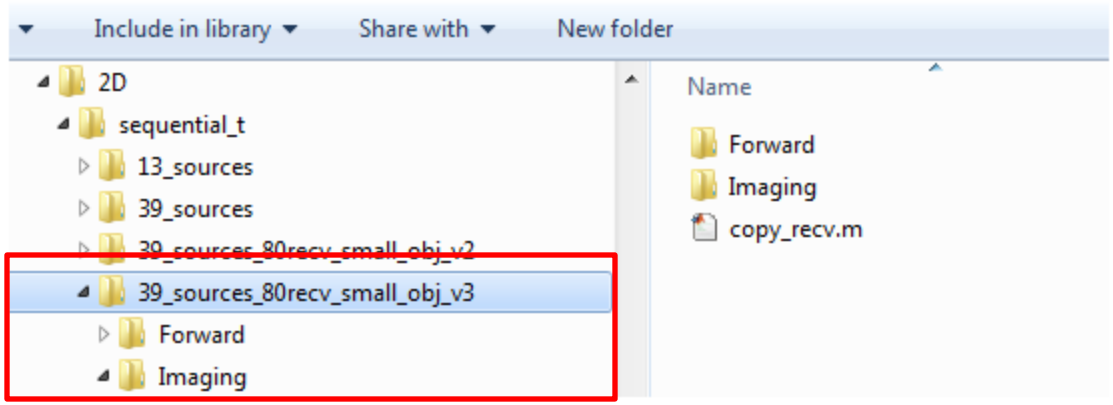
All Vx data files in the imaging project



Demo Package

- Case I: 2D Imaging, detecting two objects with f_{max}=27 Hz, known background, point dipole source, Scheme II.
  - ➢ sub-folder “forward” has the 13 cases to obtain measurement data
  - ➢ sub-folder “backward” is the project to generate the final image

- Case II: 2D Imaging, imaging a black box with limited known information, point monopole source, Scheme II.
  - sub-folder “forward” has the 39 cases to obtain measurement data
  - sub-folder “Imaging” is the project to generate the final image



END