

Requirements for an event to add to SeisFinder

Introduction

Creating MetaData

- Go you the simulation folder on Fitzroy
- Run the genCSV.py

```
python /nesi/projects/nesi00213/workflow/genCSV.py -location Christchurch -datetime 2010-09-04-04:30:00 -
exetime 201705301501 -mag 6.8
```

-datetime is the time when the earthquake happened, and must have the formate : %Y-%M-%D-hh-mm-ss
*if the event has not yet happened, use : -datetime Future
-exetime is when the simulation is ran, and must fit the formate: %Y%M%%Dhhmm, e.g. 201705301501

Note: In default , the meta data should be located under CSV folder. But if you are creating metadata for other user's simulation you can pass an extra argument to genCSV for it to save at specific place.

```
python /nesi/projects/nesi00213/workflow/genCSV.py -location Christchurch -datetime 2010-09-04-04:30:00 -
exetime 201705301501 -mag 6.8 -save ~/metadata/20100904
```

IMs CSV

One needs to provide a CSV file with the Intensity Measures (IMs) that we are going to serve from SeisFinder.

The following command will create a IMs . csv file that should be copied to the private/data/eventname/doc/ directory on the SeisFinder server. It does also create a number of single station csv on a directory called ./IMs_per_station/ that should be copied to private/data/eventname /seismo/ directory on the server, as they will be provided to the users on the zip file

```
pushd /home/nesi00213/post-processing/examples/
python export_IM_csv.py /home/nesi00213/RealTime/eventname/GM/Sim/Data/path_to/station_file.ll /home/nesi00213
/RealTime/eventname/GM/Sim/Data/BB/Vel /home/nesi00213/RealTime/eventname/GM/Sim/Data/IMs.csv full_csv
popd
```

Important: run 'source path.sh' before running other steps. If the running folder was not created follow the wiki page [Real-time Ground Motion Simulation Setup](#)

- **Source_map**

```
plot_srf_map.py src_model.srf
```

```
$plot_srf_map.py Ian02_s103245.srf
```

- **Source_square**

```
plot_srf_square.py src_model.srf
```

```
plot_srf_square.py Ian02_s103245.srf
```

- **Vs30StationMap**

note: this step requires observation generated, please follow the wiki page referring [How to create observation files](#)
*uses non_uniform_grid and prune out the virtual stations manually if you are making document for future event.

```
plot_vs30.py root_simulation_dir
```

```
plot_vs30.py /home/nesi00213/RealTime/20161213_Kaikoura_m7p8_201705301501
```

- **ObsTimeSeriesOnMap**

note: this step requires observation generated, please follow the wiki page referring [How to create observation files](#)
plot_obs.py sim_folder

```
plot_obs.py .
```

- **PNG_timeslices**

plot_ts.py processes

```
plot_ts.py 30
```

note: if files missing, try looking at the [realtime sim wikipage](#) referencing "plot_transfer.py auto"

```
plot_transfer.py auto folder_path_on_fitz local_path user_on_fitzroy  
$plot_transer.py auto /nesi/projects/nesi00213/RunFolder/baes/AlpineFault_Rupture_VMSI_v1p65_200m-  
h0p200_EM0Dv3p0p4_170531_s2n/ . ykh22
```

important: the original plot_ts.py will combine all ts.png into a mov file and delete all the png files. MUST modify the code so that it will not delete the time slices.

Comment out the rm line

```
# temporary files can be quite large  
rmtree(gmt_temp)  
#rmtree(png_dir)
```

*below is a example if you just want to plot specific ts without producing the movie file.
* you will need at least 10 time slices for the document

Plot specific TS

```
pool = mp.Pool(processes)  
#pool.map(render_slice, xrange(xyts.t0, xyts.nt - xyts.t0))  
pool.map(render_slice, [120,330,550,700,800,910,1000,1200,1500,1600])  
print('FINISHED TIMESLICE SEGMENT IN %.2fs' % (time() - ts0))  
print('AVERAGE TIMESLICE TIME %.2fs' % \  
      ((time() - ts0) / (xyts.nt - xyts.t0)))  
# images -> animation  
#make_movie('%s/ts%%04d.png' % (png_dir), \  
#          os.path.join(base_dir, 'animation.mov'), fps = 20)  
# temporary files can be quite large  
rmtree(gmt_temp)  
#rmtree(png_dir)
```

- **PGV and MMI**

The below command needs to be run from the same folder as params_base.py is in. It is normally located in \$root_dir/GM/Sim/Data

plot_ts_sum.py

```
plot_ts_sum.py
```

Once this is done, please reduce the size of the PGV.png to a new file called pgv-small.png. This is the front page image and should be in GM/.
The code using convert is:

```
convert PGV.png -resize 20x20% pgv-small.png
```

- **Post-processing related plots**

note: this step requires observation generated, please follow the wiki page referring [How to create observation files](#)

1.Create (or copy from template) a config file: post-processing_config.cfg

post-processing_config.cfg

```
[files]
outputDirectory = /home/nesi00213/RealTime/YYYYMMDD_EVENT_NAME_mXpX_YYYYMMDDHHmm/GM/Validation/
srffFile = /home/nesi00213/RealTime/YYYYMMDD_EVENT_NAME_mXpX_YYYYMMDDHHmm/Src/Model
/YYYYMMDD_EVENT_NAME_mXpX/t2/Ian02_s103245.srf
stationFile = /home/nesi00213/RealTime/YYYYMMDD_EVENT_NAME_mXpX_YYYYMMDDHHmm/Stat
/YYYYMMDD_EVENT_NAME_mXpX/YYYYMMDD_EVENT_NAME_mXpX.ll
velSimDir = /home/nesi00213/RealTime/YYYYMMDD_EVENT_NAME_mXpX_YYYYMMDDHHmm/GM/Sim/Data/BB/Vel/
dbLocation = /home/nesi00213/RealTime/YYYYMMDD_EVENT_NAME_mXpX_YYYYMMDDHHmm/GM/Validation/database.db
siteClassFile = /nesi/projects/nesi00213/Pre-processing/RegionalSeismicityTectonics/Sept42010GmMetadata.
ll
vs30File = /home/nesi00213/RealTime/YYYYMMDD_EVENT_NAME_mXpX_YYYYMMDDHHmm/Stat/YYYYMMDD_EVENT_NAME_mXpX
/YYYYMMDD_EVENT_NAME_mXpX.vs30
velObsDir = /home/nesi00213/RealTime/RealTime/YYYYMMDD_EVENT_NAME_mXpX_YYYYMMDDHHmm/GM/Obs/Data/velBB
runName = RealTime/YYYYMMDD_EVENT_NAME_mXpX_YYYYMMDDHHmm

[plotGMPE]
DistMin = 1.0
DistMax = 100
nval = 51
vs30Default = 250.0,
rake = ERROR:PLEASE_SET
Rtvz = 0
Mw = ERROR:PLEASE_SET
Ztor = ERROR:PLEASE_SET
model = 1
dip = ERROR:PLEASE_SET

[plotSeismogram]
normalize = True
magnify = 3.0

[calcIM]
plotGMPE = True
recalculateGMPE = False
recalculateIMS = False
calculateRRups = True
calcIM = 3

[plotTimeSeries]
gapStations = 5
nLimit = 5
tmax = 200
stationsToPlot = []
subdir = waveforms
gapObsSim = 2

[plotIM]
ImName = PGA, pSA, PGV, Ds575, Ds595, AI, CAV
pSAStations = []
pSAPlotCommonAxisLimits = False
unknown = D
pSAPeriod = 0.1,0.2,0.5,1.0,3.0,10.0
siteClassMarkers = False
```

2.Edit all the file directory to the correct folder. – it will error if you don't change all of the paths above.

3a) If you are running a RealTime simulation:

```
python /home/nesi00213/post-processing/examples/run_default_plots.py path_to_config_file
```

```
python /home/nesi00213/post-processing/examples/run_default_plots.py /home/nesi00213/RealTime/20161213_Kaikoura_m7p8_201705301501/GM/Validation/post-processing_config.cfg
```

3b) If you are running a future event simulation:

```
python /home/nesi00213/post-processing/examples/run_simulation_plots.py path_to_config_file
```

```
python /home/nesi00213/post-processing/examples/run_simulation_plots.py /home/nesi00213/RealTime/20161213_Kaikoura_m7p8_201705301501/GM/Validation/post-processing_config.cfg
```

* important: must edit the stationsToPlot if simulation used non-uniform grid. Otherwise it will plot all the virtual stations

* for future events, which mean there is not observation data

- **Spatial Distribution**

note: this must be done after post-processing

1. cd GM/Validation/Data. There should exist two .ll files PGV and pSA

2. plot the files using plot_stations.py

```
plot_stations.py stat_ratios_PGV_20161213_Amberley_m7p8_201705301501.ll IMs # this will write the images to IMs relative directory
plot_stations.py stat_ratios_pSA_20161213_Amberley_m7p8_201705301501.ll pSA # this will write the images to pSA relative directory
```

Note: new format of config will have IM instead of PGV, 6 file instead of 1. pick the first one and ignore the rest.

Running publish

- cd GM/Sim
- create of copy the config file: publish.cfg

publish.cfg

```
[event]
title:13 Nov 2016 Mw7.8 Kaikoura
location:/home/nesi00213/RealTime/20161213_Amberley_m7p8_201705301501
csv:/home/nesi00213/RealTime/20161213_Amberley_m7p8_201705301501/GM/Sim/MetaData
output: 20161213_Amberley_m7p8_201705301501

[image]
SrfMapView:Src/Figures/source_map.png
srfSlipRiseTimeRake:Src/Figures/source_square.png
#
Vs30StationMap:VM/Figures/Vs30StationMap.png
ObsTimeSeriesOnMap:GM/Obs/Figures/ObsTimeSeriesOnMap.png
#
timesliceLocation:GM/Sim/Figures/PNG_timeslices/
timesliceFiles:ts0120.png,ts0330.png,ts0550.png,ts0700.png,ts0800.png,ts0910.png,ts1000.png,ts1200.png,
ts1500.png

validationLocation:GM/Validation/Figures/IM/IM_vs_source_site_distance/
validationFiles:PGA_comp_geom_with_Rrup_with_GMPE_with_Station_ClassesVel.png,pSA(0p2)
_comp_geom_with_Rrup_with_GMPE_20161213_Amberley_m7p8_201705301501.png,pSA(0p5)
_comp_geom_with_Rrup_with_GMPE_20161213_Amberley_m7p8_201705301501.png,pSA(1)
_comp_geom_with_Rrup_with_GMPE_20161213_Amberley_m7p8_201705301501.png,pSA(3)
_comp_geom_with_Rrup_with_GMPE_20161213_Amberley_m7p8_201705301501.png,pSA(10)
_comp_geom_with_Rrup_with_GMPE_20161213_Amberley_m7p8_201705301501.png

pSAWithPeriod:GM/Validation/Figures/IM/spectra_ratio
/pSAWithPeriod_compgeom_20161213_Amberley_m7p8_201705301501.png
#missing
spatialLocation:GM/Validation/Figures/ratios
spatialFiles:PGV/p000.png,pSA/p000.png,pSA/p001.png,pSA/p002.png,pSA/p003.png,pSA/p004.png
#
simPgv:GM/Sim/Figures/PNG_tssum/PGV.png
simMmi:GM/Sim/Figures/PNG_tssum/MMI.png
#
responseSpectraLocation:GM/Validation/Figures/IM/PSA
responseSpectraFiles:pSA_comp_geom_vs_Period_Vel_AMBC.png,pSA_comp_geom_vs_Period_Vel_CMHS.png,
pSA_comp_geom_vs_Period_Vel_DALS.png,pSA_comp_geom_vs_Period_Vel_FTPS.png,
pSA_comp_geom_vs_Period_Vel_GLWS.png,pSA_comp_geom_vs_Period_Vel_INGS.png,
pSA_comp_geom_vs_Period_Vel_KOWC.png,pSA_comp_geom_vs_Period_Vel_MCAS.png,
pSA_comp_geom_vs_Period_Vel_NCBS.png,pSA_comp_geom_vs_Period_Vel_WVFS.png
#
waveformLocation:GM/Validation/Figures/waveforms
waveformFiles:velts_ADCSAKSSAMBCAPPSARKS_2016Nov13_Ian02_s103245_VMCant_Amberly_200m-
h0p200_EMODv3p0p4_161221.png,velts_SMTCSOCSSOMSSPRSSTAS_2016Nov13_Ian02_s103245_VMCant_Amberly_200m-
h0p200_EMODv3p0p4_161221.png
```

- Edit all path to the correct directory
- Run the publish script

```
python /home/nesi00213/gm_publish/publish.py publish.cfg
```