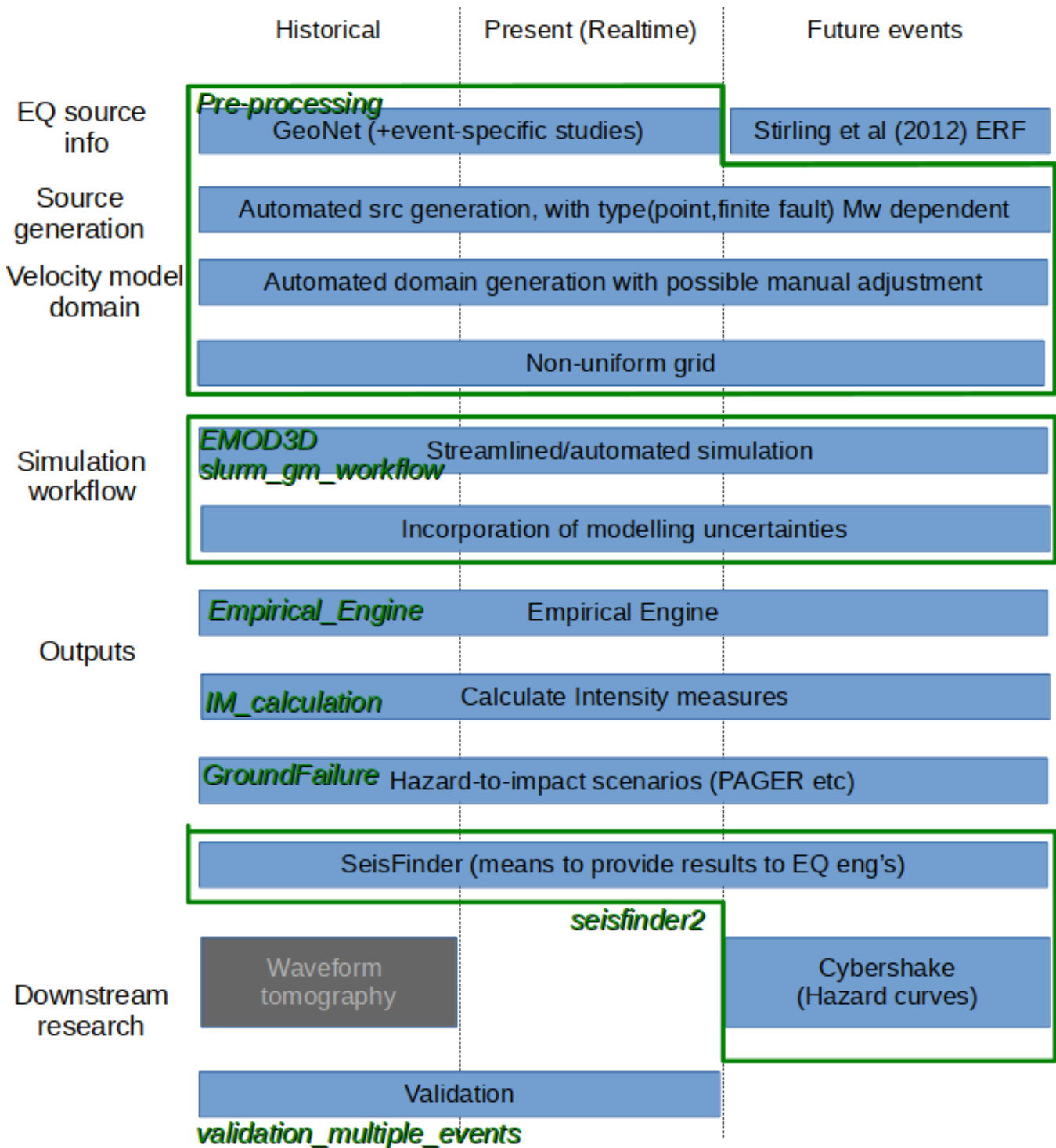


SW Team 2019 Q2 Planning

Structure of Codebase



(from [SW Codebase 2019](#) page : qcore, visualisation and GMSimViz are not listed.

This page is somewhat outdated, but contains some useful background)

Repository Specific Improvements

Repo	Improvement	Note
------	-------------	------

slurm_gm_workflow https://github.com/ucgmsim/slurm_gm_workflow	Outstanding issues <ul style="list-style-type: none"> • DB Issue: Fix the lock issue caused by excessive access • Remove/update legacy code & parameters and accomodate new environment • Deprecate cybershake.json Improvements <ul style="list-style-type: none"> • Separate repos: workflow automation vs calculation • Better logging • Automated verification/testing • Integrate model (srf/vm) into the workflow (with an option to stop before simulation) • Estimation performance optimization • Automated Visualisation • Error handling • Realisation name change: AlpineF2K_HYP01-47_S1244 to AlpineF2K_REL01 • Site specific binary workflow 	
qcore https://github.com/ucgmsim/qcore	Remove unneeded code/functions More coherent structure with related functions kept in the same file Consistent comment styles using doctoring and API doc Expand automated unit test coverage (less than 10%) (lat.lon).csv grid.xml currently not used. Plan for PAGER?	
Pre-processing https://github.com/ucgmsim/Pre-processing	Better estimation for model generation Repo restructure : GMSim_model, NonUniformGrid and archive unused legacy code Incorporate model generation into management DB (See slurm_gm_workflow) Automated testing for model generation NonUniformGrid code has minor issues (but low priority, run yearly)	
seisfinder (ver.1/ver.2) https://github.com/ucgmsim/seisfinder2	Regression tests (after scientific validation) GM selection Login and user management Missing ver.1 features: <ul style="list-style-type: none"> • validation document (using gm_publish) • custom name • PGV map (upon the selection of an event) • All im .csv files into one .csv 	
Visualisation https://github.com/ucgmsim/visualization	Clean up Python 3 Refactor plot_stations.py	
empirical_engine https://github.com/ucgmsim/Empirical_Engine	Integrate into hazard workflow (replacing OpenPSHA, no new functionality, but can streamline empdb creation)	
ground failure https://github.com/ucgmsim/GroundFailure	Clean up	
validation https://github.com/ucgmsim/validation	Mixed effect regression workflow to be version-controlled Add automation Improve the code quality	

GMSimViz https://github.com/ucgmsim/GMSimViz	Specifying regions of interest	
gm_publish https://github.com/ucgmsim/gm_publish	Decide if seisfinder2 needs this	
IM_calculation https://github.com/ucgmsim/IM_calculation	Include just .000 and .090 for geom only (33% speed up) Calculate RTVZ and RX Replace Cython spectra with better Python code	
Velocity_Model https://github.com/ucgmsim/Velocit-Model	-	
EMOD3D https://github.com/ucgmsim/EMOD3D	-	

Common Improvements

- Template for README : Amalgamate README, Codebase wiki page and repo maturity page, and put everything in README.
- Python 3 and coding & comment style
- Automated testing and Continuous integration

Stable Release

A repository that satisfies the following criteria will have an official stable release.

- All the planned functionalities have been developed and tested
- Codebase has been cleaned up
- Good comments & documentation (README)
- Automated testing coverage over 80%

Ideally, we can have the package installed via "pip" command:

```
pip install qcore
```

The repositories we should aim to produce stable releases are (ordered by impact/risk analysis)

- IM_calculation
- Pre-processing
- Qcore
- Slurm_gm_workflow

Moving Forward: Mid April - End of June

1. Stable release of IM_calc and Pre-processing repos : Initially small team of 2~3 piloting, polishing up the process
 - a. IM_Calc:
 - i. Clean up
 - ii. Finish automated testing
 - iii. Comment style,
 - iv. README.md
 - v. Release
 - b. Pre-processing:
 - i. Better estimation,
 - ii. Split the repo : Model_gen, Grid_gen
 - iii. Clean up
 - iv. Automated testing
 - v. Comment style
 - vi. README.md
 - vii. Release
2. Slurm_gm_workflow & Qcore restructuring

- a. Slurm_gm_workflow:
 - i. Restructure, split the repo. make all .yaml compliant
 - ii. Remove/update legacy code
 - iii. Integrate Pre-processing into automated workflow
 - iv. Logging
 - v. Automated verification
 - vi. Estimation performance
 - vii. Visualisation
 - viii. Error handling
 - ix. Realisation name change
 - b. Qcore
 - i. Restructure
 - ii. Clean upj
 - iii. Comments in Numpy style
 - iv. Increase automated unit test coverage
 - v. API doc
 - vi. README
3. SeisFinder2: Feature completion (3months time frame)
- a. Integration of GM selection
 - b. Verification and regression testing etc.
 - c. Lite version(?)
4. README template : Example <https://gist.github.com/PurpleBooth/109311bb0361f32d87a2> : TODO, Changelog etc.
5. Standard for comment and API doc : [NumPy vs Google](#)
6. Cybershake
- a. Include subduction
 - b. Check the performance of new VM and run 1 cycle of Cybershake
 - c. HF changes (inc. path duration)
 - d. Deagg and determines more relevant faults, rerun Cybershake with less fault, higher res.