

Research and software development

Experimenting within a robust framework

Sander Hendrickx

Biophysics research group – Ghent University

OpenDGM

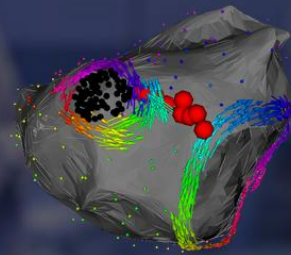
Framework to analyze electrophysiological mapping data – clinical and simulated

OpenDGM

Electrophysiological
Analysis Framework



[Get started](#) [Examples](#) [Code repository](#) [API Reference](#)



OpenDGM

A diagnostic software framework to - automatically - determine the source of a given arrhythmia. Using network theory on cardiac excitation, OpenDGM is capable of detecting all critical boundaries, to allow the optimal ablation line in each case.



Open source

OpenDGM is actively maintained and new features are added regularly. It is open source and free to use under the GPLv3. GNU GENERAL PUBLIC LICENSE version 3.



Intuitive

Each core object - mesh, graph, cycles, ... - has its own wrapper class for quick access, manipulation and visualization.



Modular

Our modular approach makes it easy to swap different parts in your pipeline to test different setups. The in- and output of a module stays the same. What happens in between is up to you!



Scalable

Modules make it easily scalable. Run and compare multiple cases or run your algorithms over a whole parameter space.

Research versus software development

Workflow and guidelines we implemented to get the best of both worlds

Research

- ▶ Experiment
- ▶ Fast prototyping
- ▶ Paper
- ▶ Personal use

Software development

- ▶ Stable
- ▶ Tested
- ▶ Documented
- ▶ General usage

First: one repository. Everybody worked in it, used it

- Wild growth of branches
- Merge conflicts
- Everything out of sync
- Very hard to maintain

Now: one general framework (OpenDGM, open source)
one repository per research topic

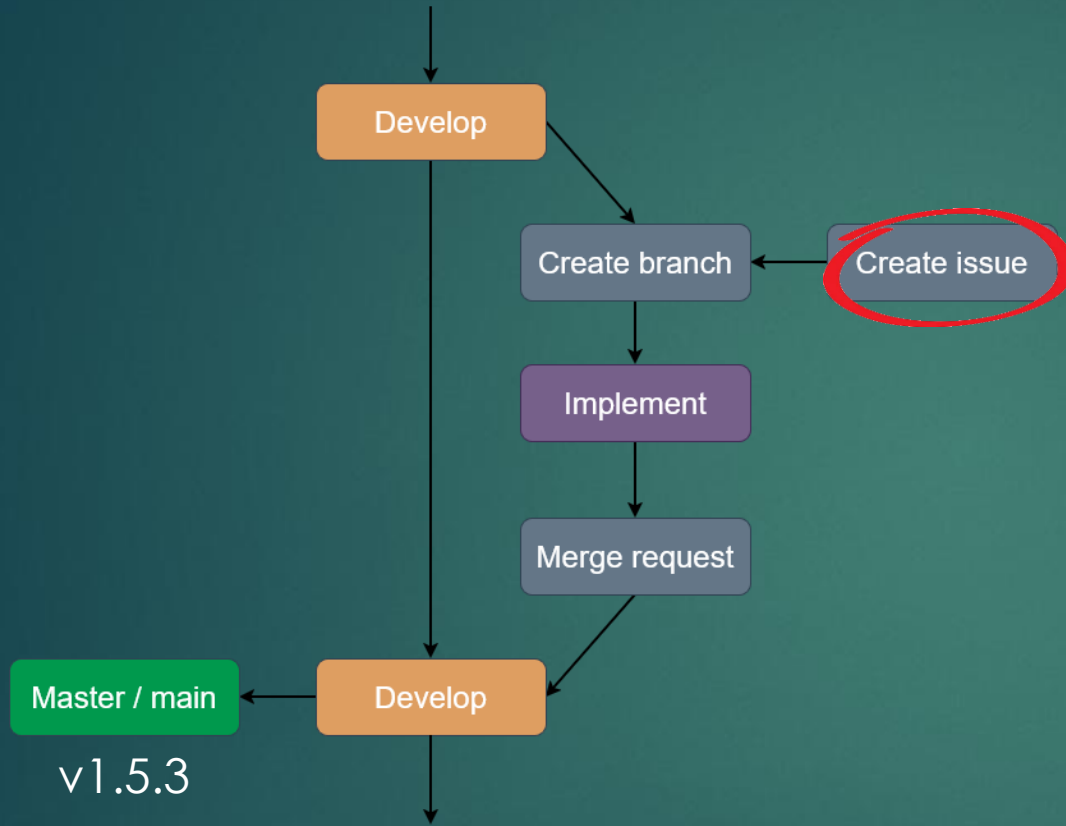


Use the framework to do research



Do research in the framework

Software development cycle



Master / main

- ▶ Nobody commits here
- ▶ Latest stable version

Develop

- ▶ Default branch for development
- ▶ Any feature branch starts here
- ▶ More up-to-date than master

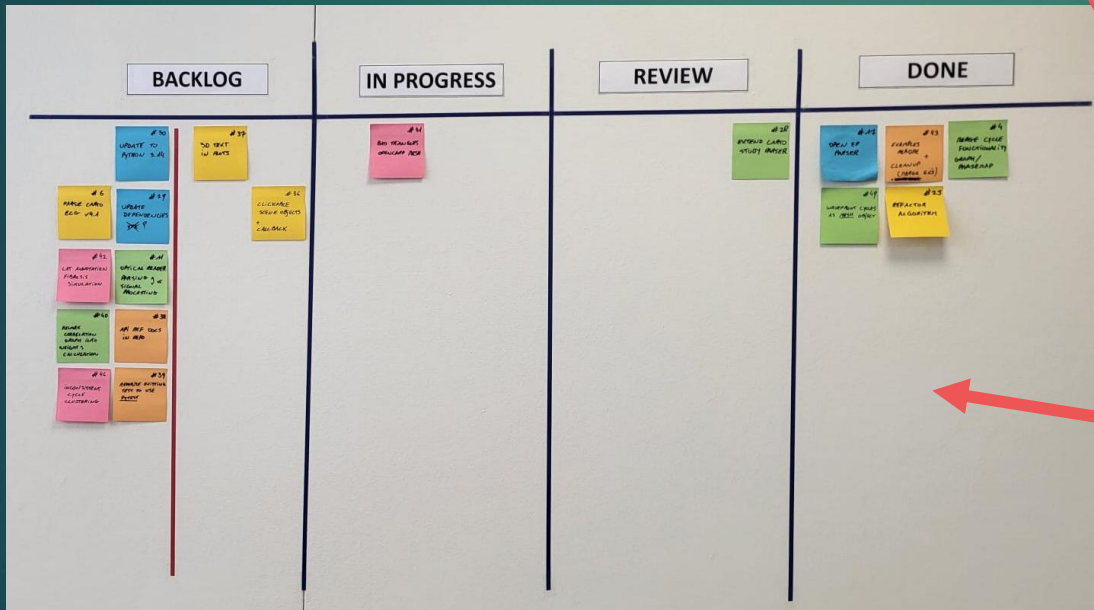
Issue → Feature branches

- ▶ Temporary
- ▶ Implementation of a new feature, solve a bug, enhance an existing feature, ...

Tools






from the software development world

```
graph\graph.py:1:0: W0611: Unused import logging (unused-import)
graph\graph.py:2:0: W0611: Unused deepcopy imported from copy (unused-import)
graph\graph.py:6:0: W0611: Unused pandas imported as pd (unused-import)
graph\graph.py:7:0: W0611: Unused pyvista imported as pv (unused-import)
***** Module opendgm.graph.nk_graph
graph\nk_graph.py:25:27: W0613: Unused argument 'directed' (unused-argument)
graph\nk_graph.py:1:0: W0611: Unused numpy imported as np (unused-import)
-----
Your code has been rated at 9.32/10 (previous run: 9.93/10, -0.62)
```



Git (github / gitlab)  

▶ Issues → branches

-  Feature
-  Enhancement
-  Suggestion
-  Bug
-  Documentation / testing

▶ Versioning + [CHANGELOG.md](#)
▶ CI

Code consistency

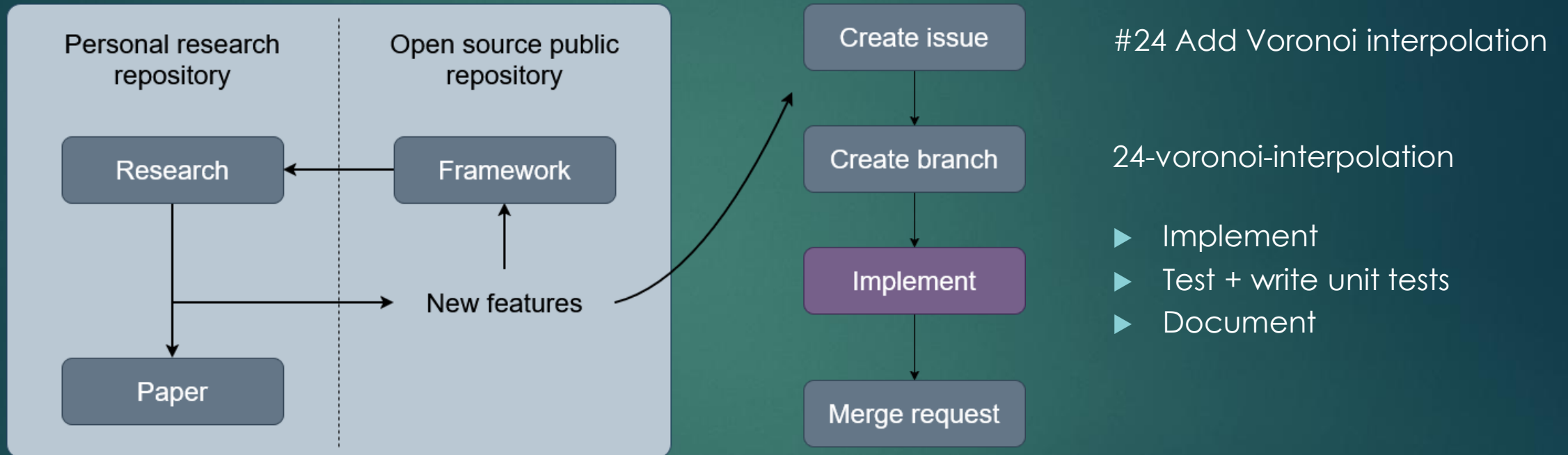
- ▶ pylint
- ▶ mkdocs – google style docstrings
[API reference guide](#)

Scrum board

- ▶ Keep track of progress
- ▶ Set priorities

Research cycle

how to fit our research into the development cycle



Ready to submit paper? Clean up repo, make public
Using OpenDGM v1.5.3

Benefits

- ▶ Reproducibility
public research repository
linked to framework version
- ▶ Clean separation
repositories for research itself ↔ framework with tools to do research
- ▶ Everyone contributes to and benefits from the same framework
- ▶ Easy to maintain

Thank you for your attention

Feel free to ask any questions