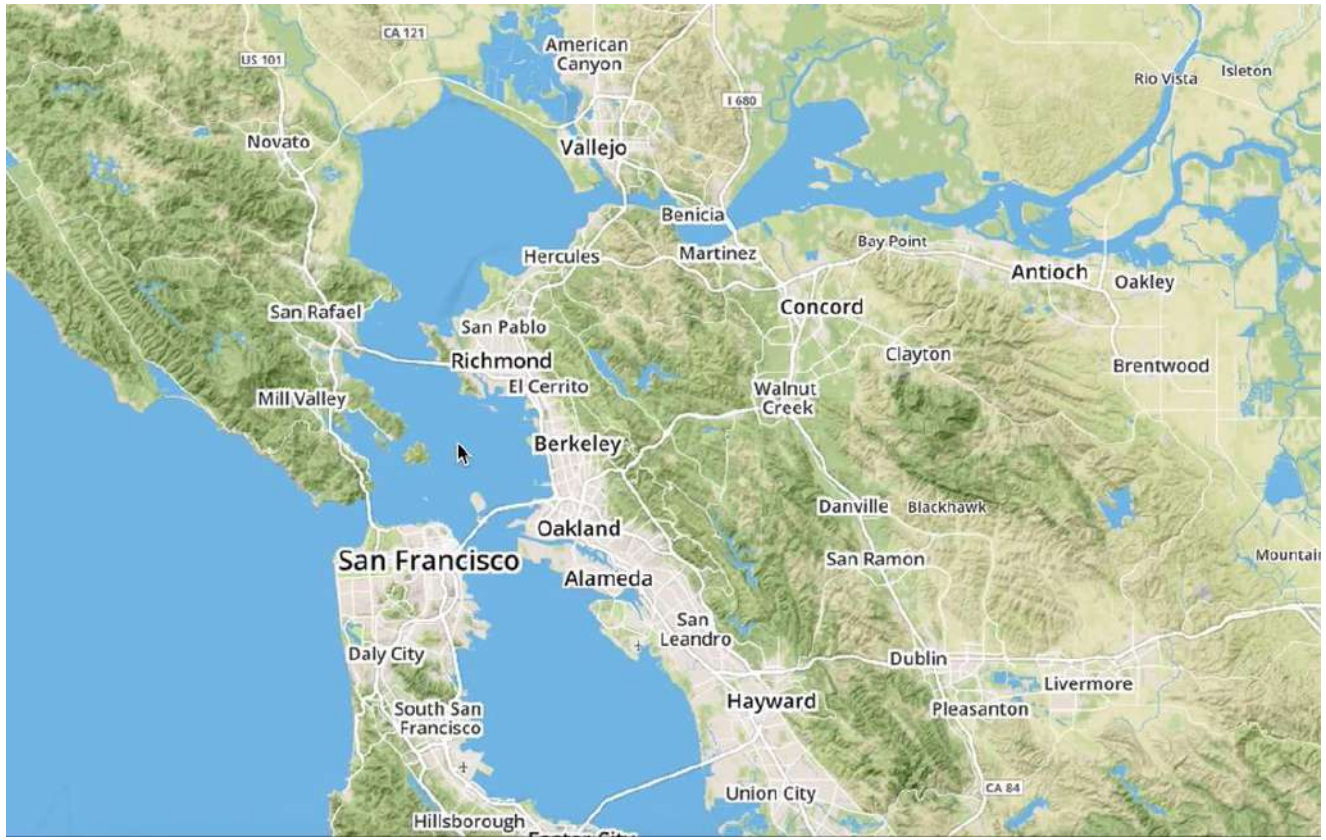
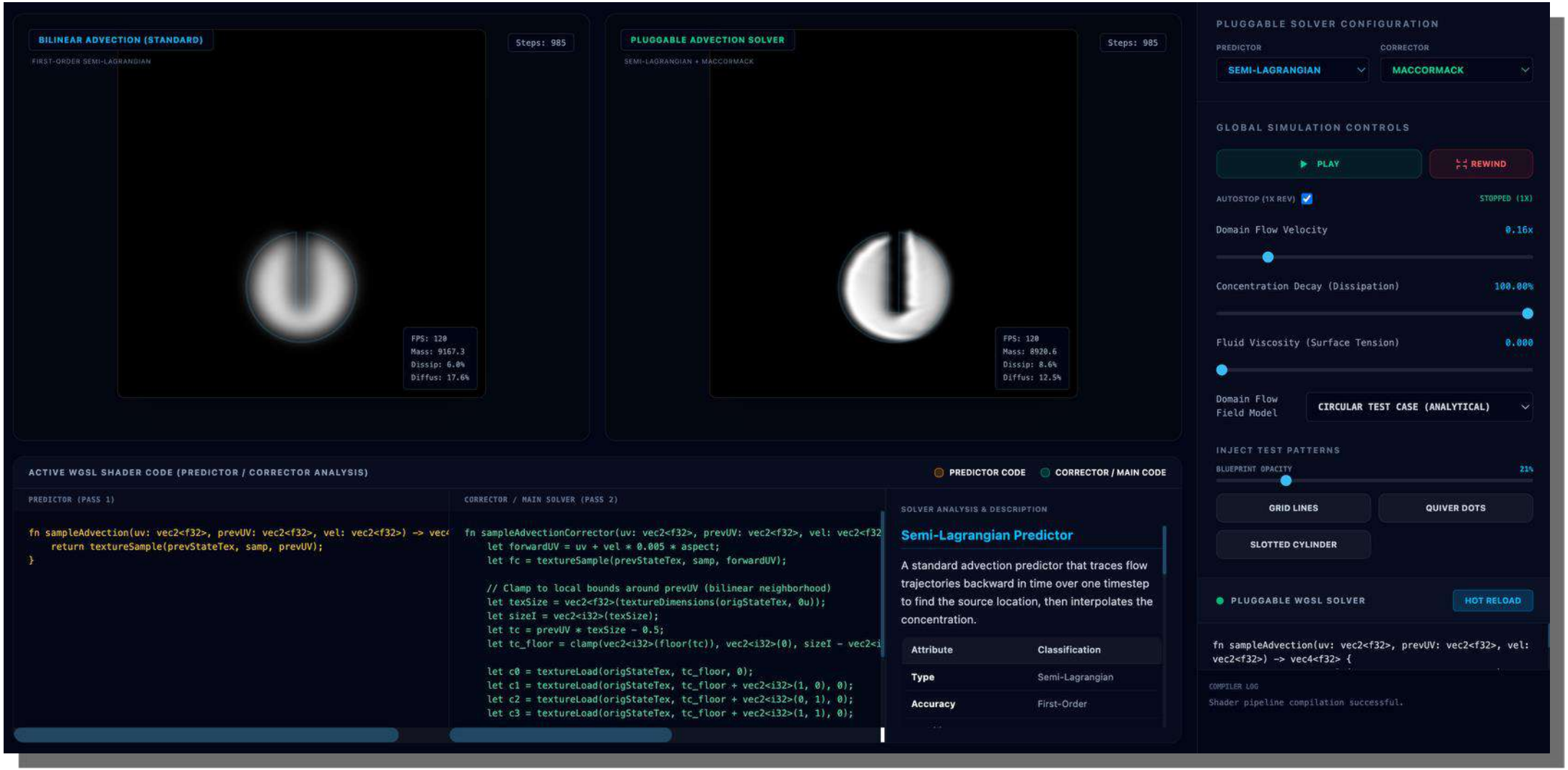
Experiment 3: Postprocessing tidal model

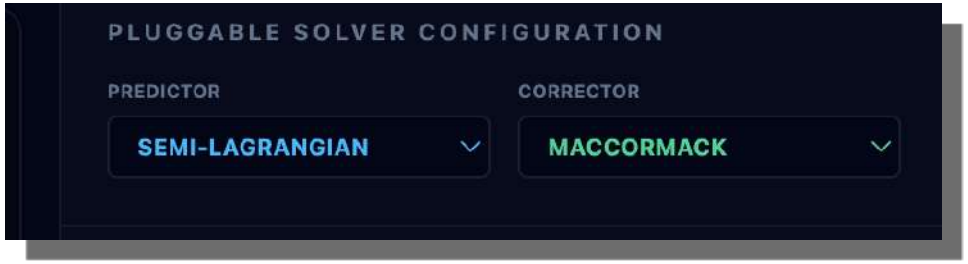
- Rebuild old visualization tool
- New model (GTSM / DCSM)
- New language WebGL1 – WebGPU
- Fix numerical issues (mass conservation, dissipation)
- User experience

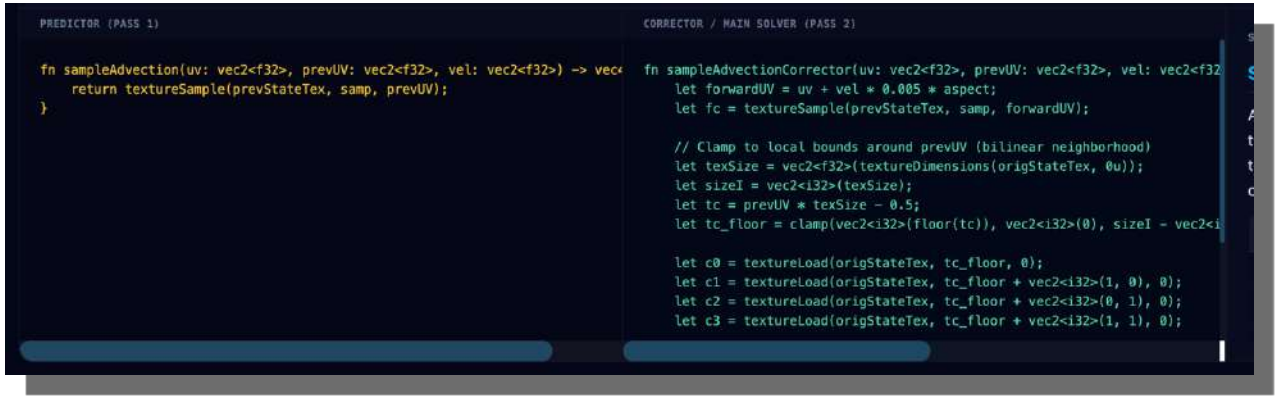


Numerical testbed

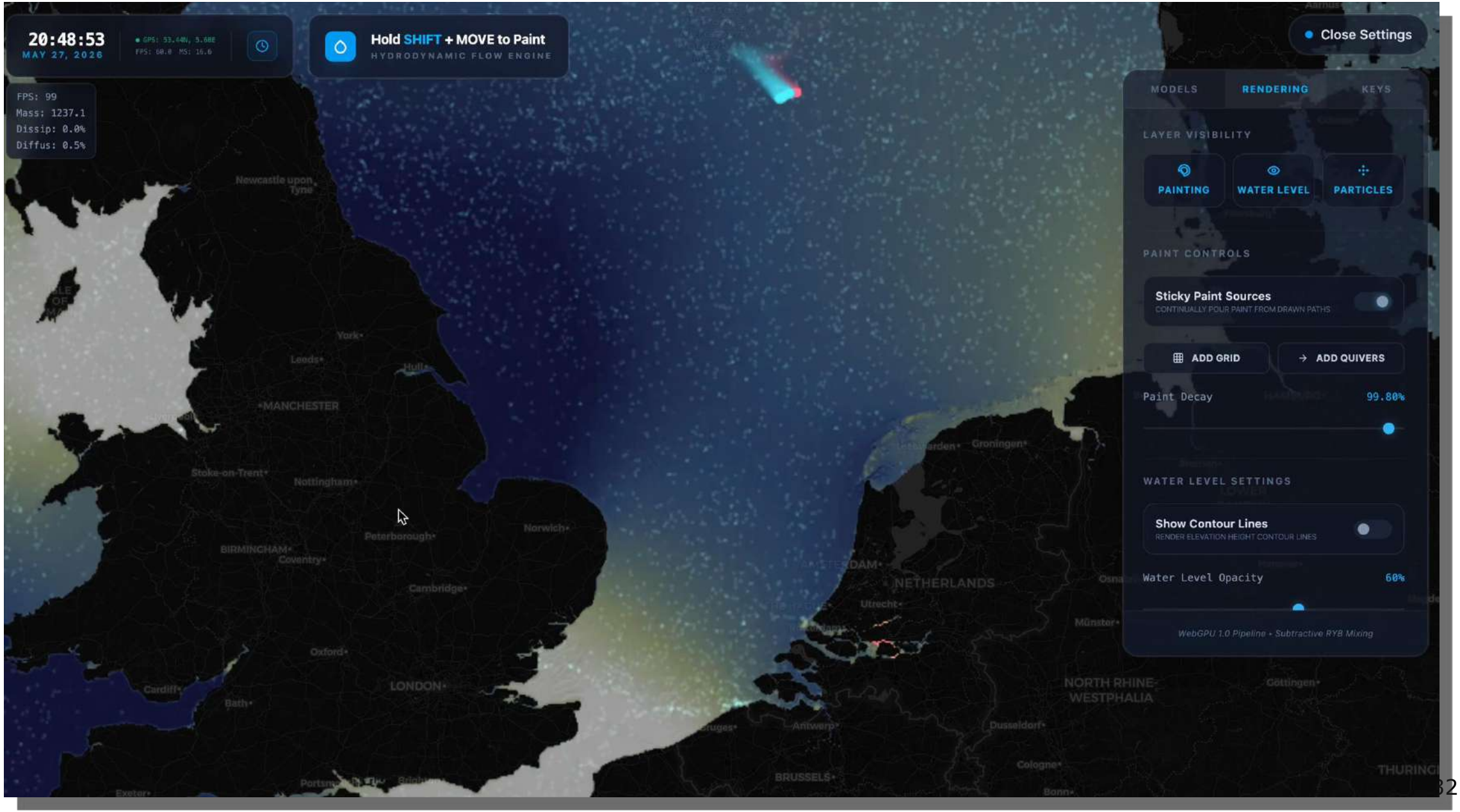


Separate numerical improvement task



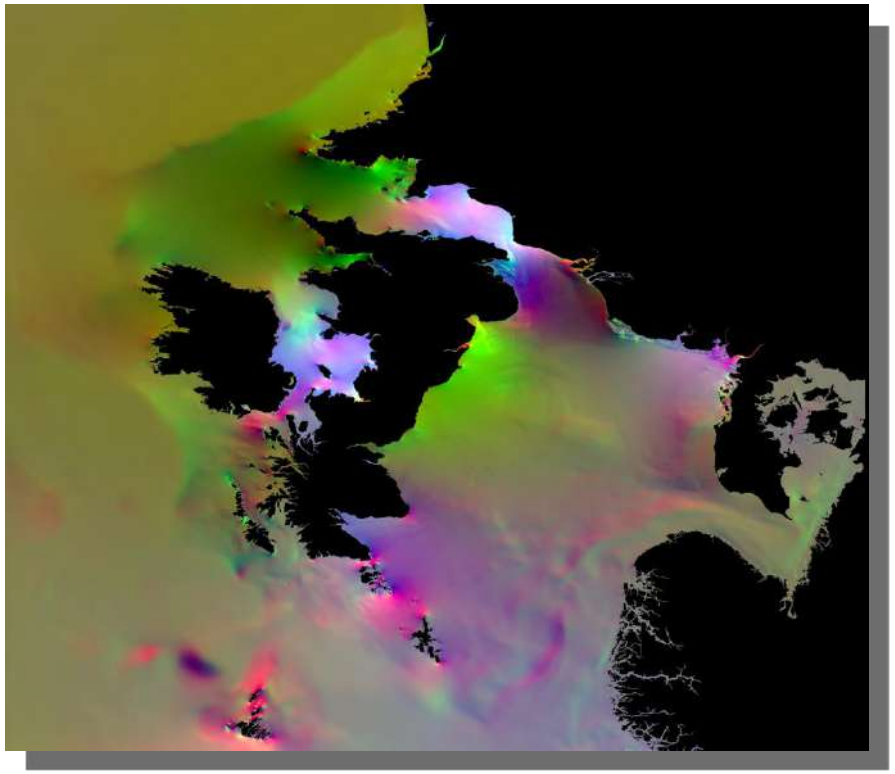


- Hoop based: A-B testing: gradual improvements
- Literature research (gemin research)
- Composable shaders
- Offline validation
  - Rust WGPU + Python validator


20:48:53
MAY 27, 2026
Hold SHIFT + MOVE to Paint
HYDRODYNAMIC FLOW ENGINE
Close Settings
FPS: 99
Mass: 1237.1
Dissip: 0.0%
Diffus: 0.5%
MODELS
RENDERING
KEYS
LAYER VISIBILITY
PAINTING
WATER LEVEL
PARTICLES
PAINT CONTROLS
Sticky Paint Sources
CONTINUALLY POUR PAINT FROM DRAWN PATHS
ADD GRID
ADD QUIVERS
Paint Decay
99.80%
WATER LEVEL SETTINGS
Show Contour Lines
RENDER ELEVATION HEIGHT CONTOUR LINES
Water Level Opacity
60%
WebGPU 1.0 Pipeline + Subtractive RYB Mixing
Newcastle upon Tyne
York
Leeds
Hull
MANCHESTER
Stoke-on-Trent
Nottingham
Peterborough
Norwich
BIRMINGHAM
Coventry
Cambridge
Oxford
LONDON
Cardiff
Bath
NETHERLANDS
Utrecht
Groningen
Münster
NORTH RHINE-WESTPHALIA
Dusseldorf
Cologne
Bonn
Antwerp
BRUSSELS
Exeter

Experience

- Reduce burden of coding in WGPU
- Numerical improvement (kept bilinear – bicubic)
- Postprocessing (reimplemented rgba video texture encoding, but forgot to check in code)
- Understood all visualisation hints, e.g. "use crameri", "add quiver", "use ink style mixing"



```
fn cubicWeight(x: f32) -> vec4<f32> {
    let x2 = x * x;
    let x3 = x2 * x;
    let w0 = 0.5 * (-x3 + 2.0 * x2 - x);
    let w1 = 0.5 * (3.0 * x3 - 5.0 * x2 + 2.0);
    let w2 = 0.5 * (-3.0 * x3 + 4.0 * x2 + x);
    let w3 = 0.5 * (x3 - x2);
    return vec4<f32>(w0, w1, w2, w3);
}

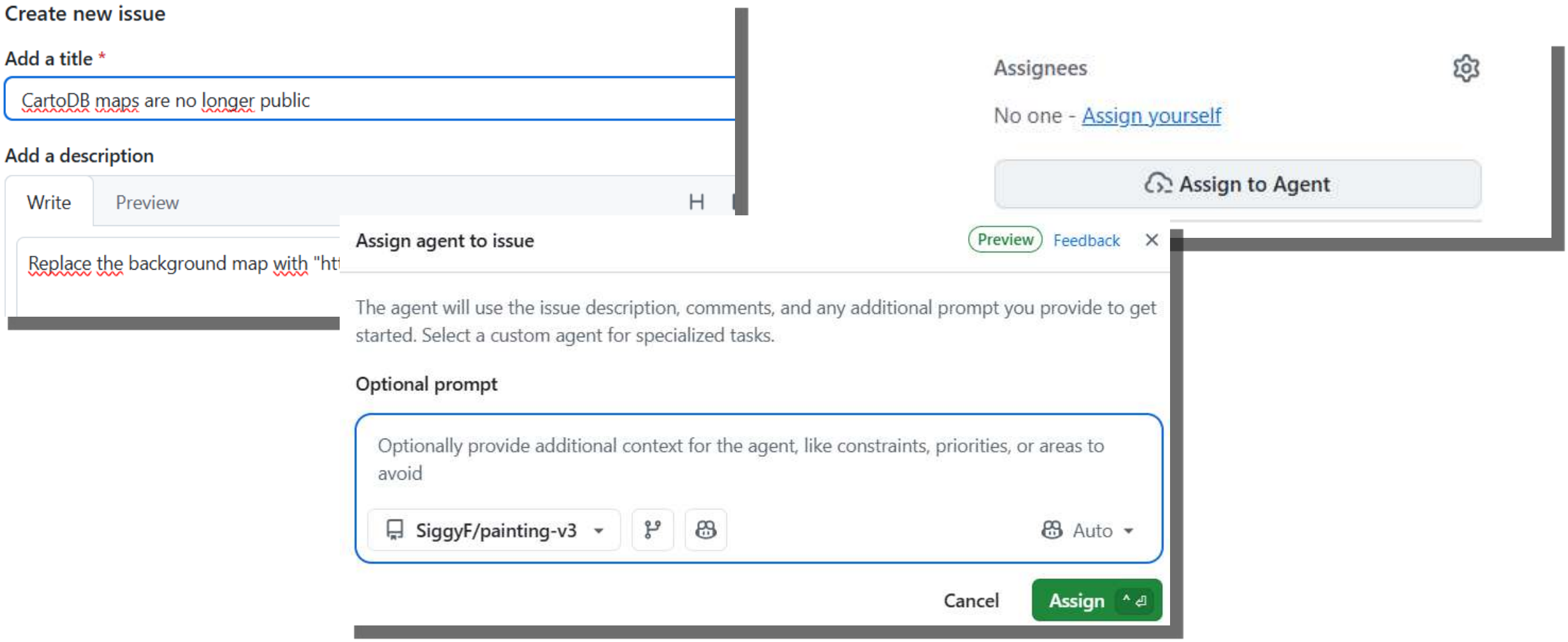
fn sampleAdvection(uv: vec2<f32>, prevUV: vec2<f32>, vel: vec2<f32>) ->
    let texSize = vec2<f32>(textureDimensions(prevStateTex, 0u));
    let texelSize = 1.0 / texSize;
```

Current state (after 3months)

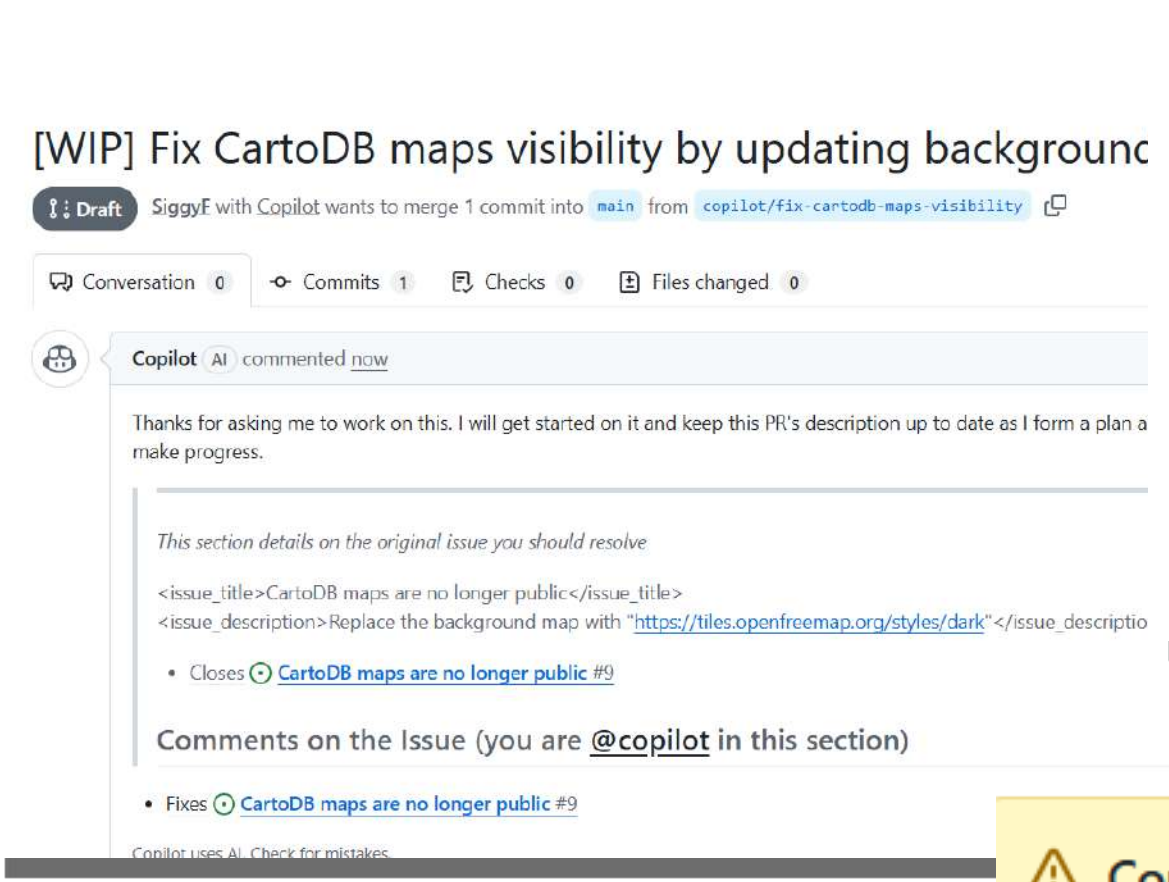


https://siggyf.github.io/painting-v3

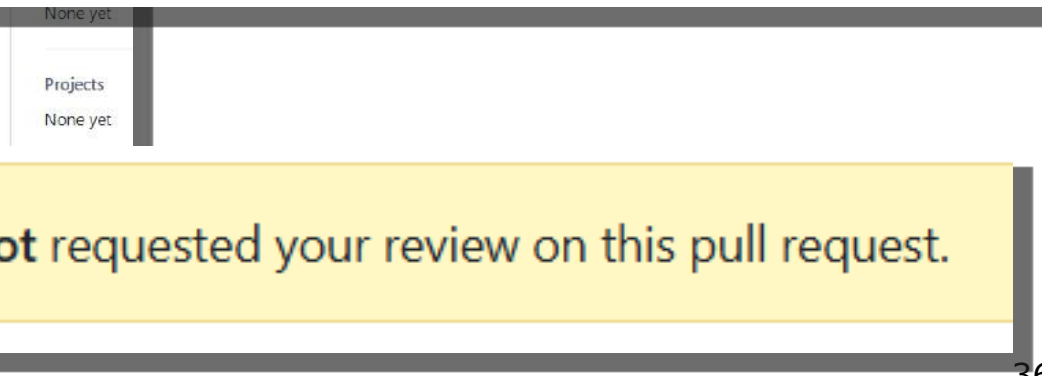
Workflow: report issue



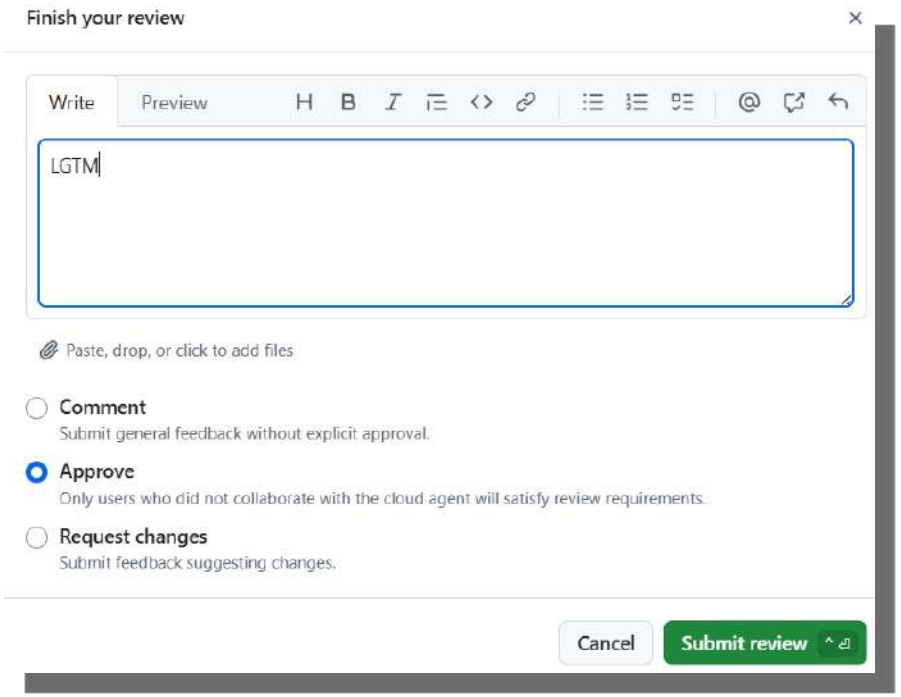
Workflow: vibe loop

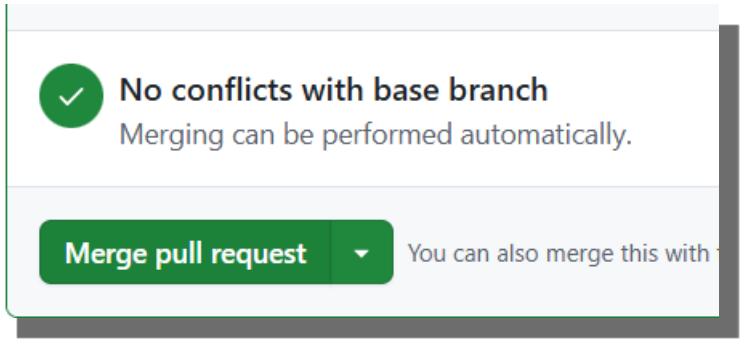


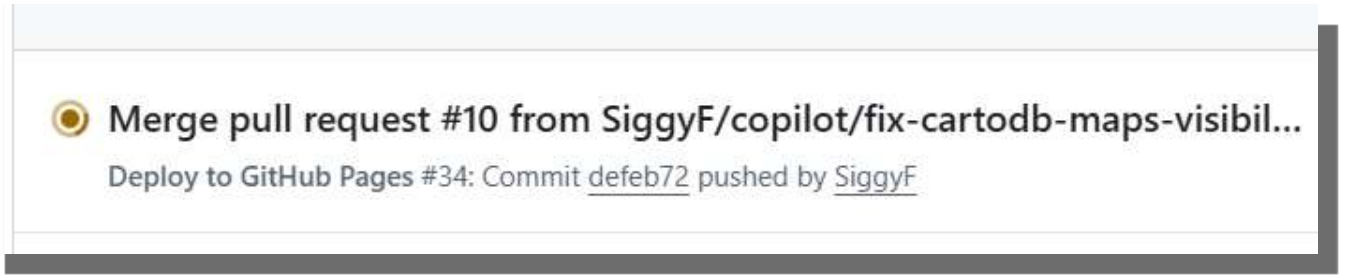




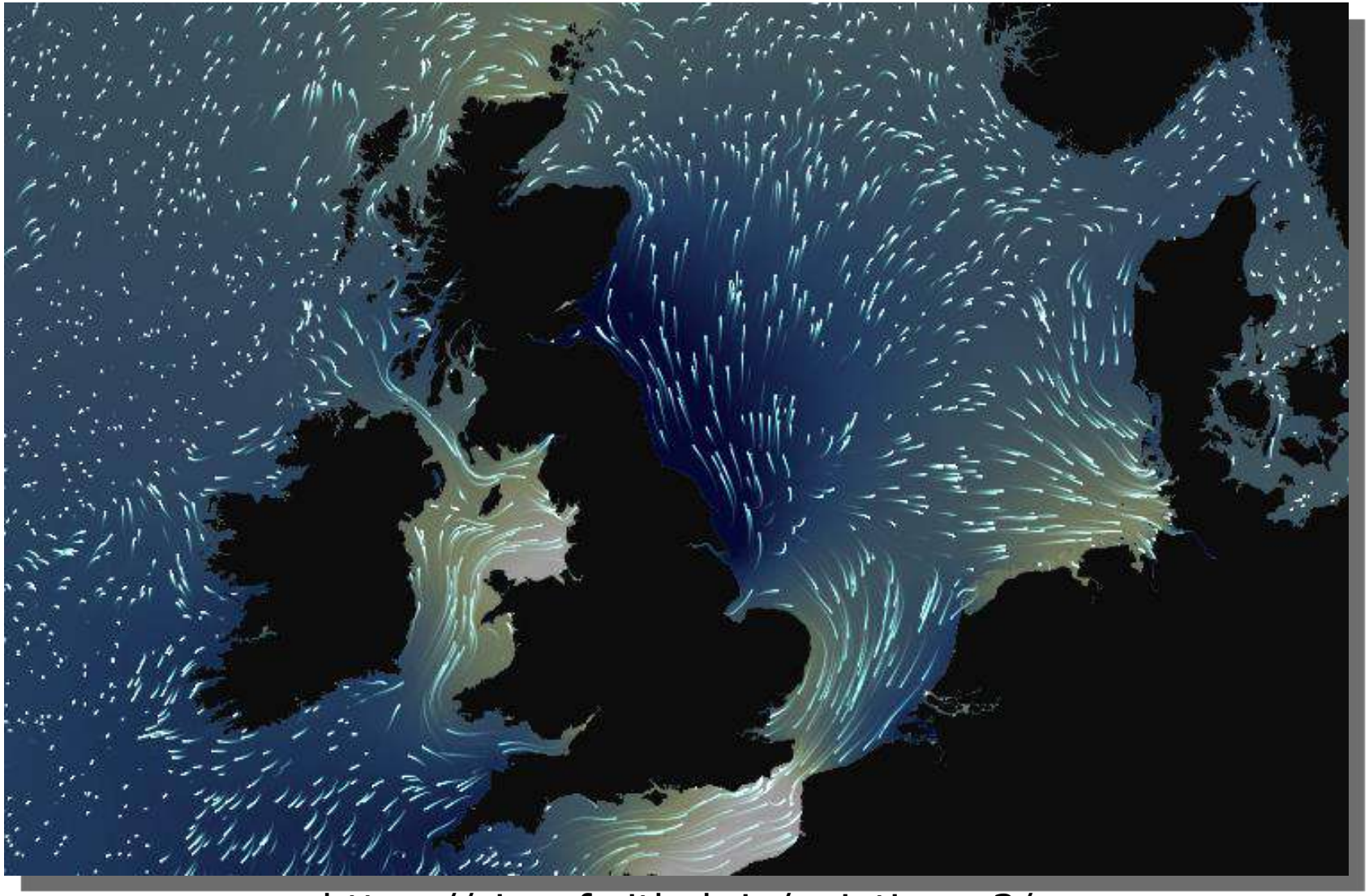
Workflow: Human in the loop





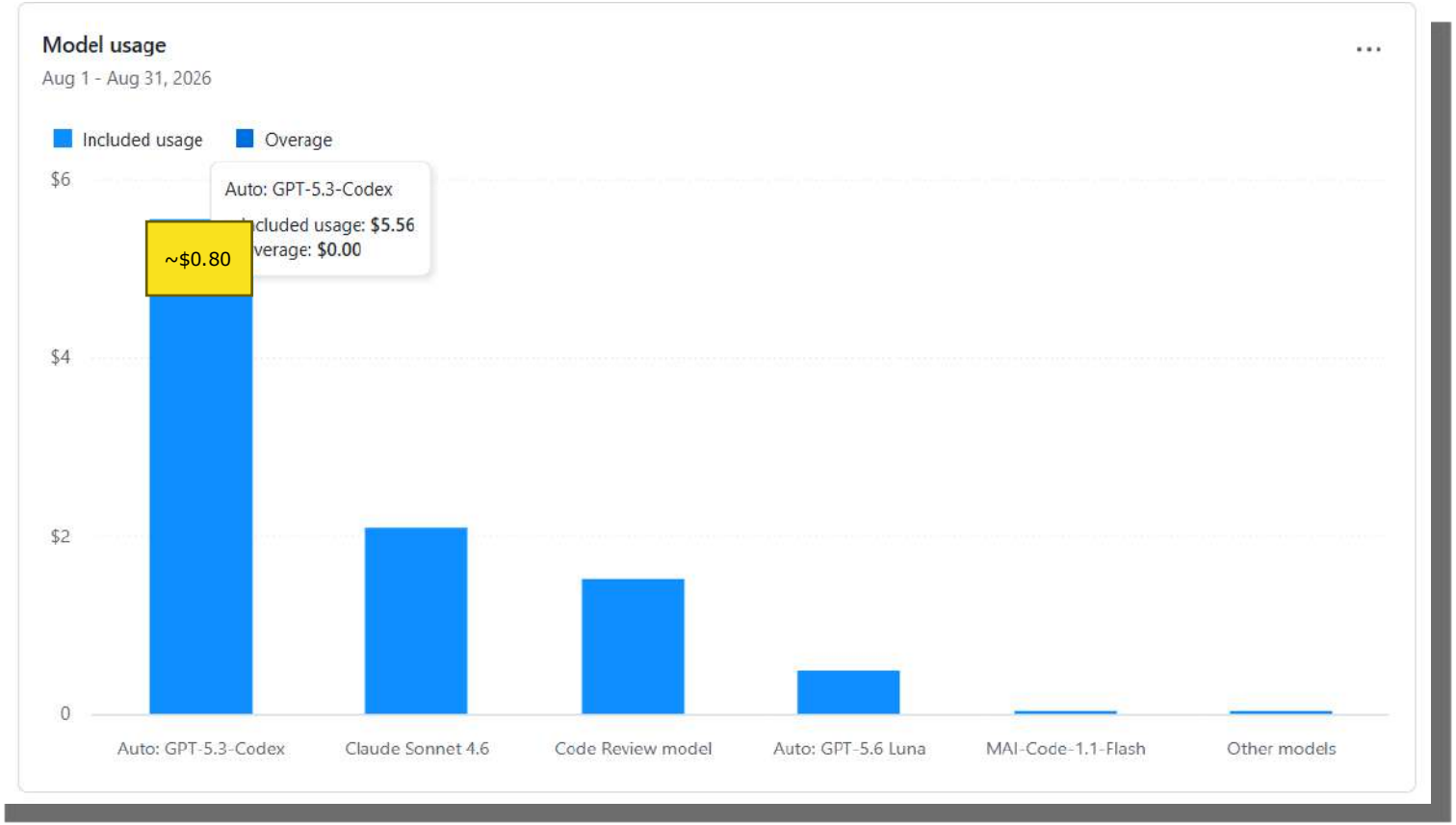


Bug fixed



https://siggyf.github.io/painting-v3/

Cost: bug fix (5min reporting, 5min PR checking + $0.80)



Guidelines

- You are responsible
  - Hoops, tests, validation
- Security: confine, restrict
- Workflow: loop, guide, split

Typische problemen

"Stop replacing my files!!!!"

Context: Handmatig een afbeelding gecorrigeerd en bijgesneden omdat de AI niet goed had afgeknipt. Direct daarna draaide de AI een geautomatiseerd script dat deze handmatig herstelde bestanden stilzwijgend weer overschreef met een eigen gegenereerde versie.

Les: De agent overschrijft blind bestanden. Backup, versiebeheer, read only input.

- ***"Why are you looking around on my hard drive? Stop scanning my computer."***

Context: Een dataset op Zenodo ontbrak lokaal. In plaats van te falen of te downloaden, begon de AI met shell-commando's in ~ en willekeurige mappen te zoeken.

Les: Strikte zandbak (devcontainer, gelimiteerde tokens)

"Check of we nog meer fallbacks hebben. Dat willen we niet. Waarom lees je 10x dezelfde file?"

Context: De agent bleef fallbacks toevoegen (zoek bestand A, dan B, dan A, dan B) en las in een doelloze cyclus steeds hetzelfde bestand opnieuw in.

Les: Nieuwe infinite loop. Externe loop voor progress monitoring.

- ***"KAP NOU!!! … Je blijft maar uvw noemen. We stopppen ermee."***

Context: Tijdens het maken van een document bleef de agent hardnekkig refereren aan een onofficiële mirror (uvw.bestuurlijkeinformatie.nl) in plaats van open.overheid.nl. Na meerdere toezeggingen van de AI (*"Geen zorgen, we gebruiken uitsluitend open.overheid.nl"*) dook het domein weer op.

Les: alleen instructies zijn vaak niet voldoende als context besmet is. Reset sessie (compact / clear).

"We don't want to fix any other bugs. I did not ask for that."

Context: Tijdens het opstellen van een Pull Request voor een extern package (datashader) begon de AI spontaan omliggende bestanden te refactoren en extra bugs te fixen.

Les: Beperk scope en inspanning (simpele taken, lage thinking depth, simpelere modellen werken vaak beter)

Discussion