NEST Updates

HIRSE

Live Meeting – Jülich 09/2025

Jessica Mitchell, Dennis Terhorst, Steffen Graber

Topics

- NEST Activities 2025/2026
- NEST PRs for code refactoring - sli2py with AI
- CI-bennch process
- HiRSE AI

NEST Activities

NEST 2025

Releases

- **NESTML** 8.1.0
- **NEST Desktop** 4.1.0
- **NEST Simulator** 3.9 (This week!)
    - 1193 commits by 24 developers since NEST 3.8 (2024-07-10)
- ✨ NEST **NEAT** (NEural Analysis Toolkit)
    - tool to simulate and simplify morphological neuron models

☑ all available in upcoming *EBRAINS* release 🎉

NEST 2025

Events

- Fortnightly open video conferences
- Quarterly Hackathons
- Yearly NEST conference
- Presence at conferences (Sweden NN, CNS symposium, …)
- FZ-Jülich open-day

NEST Simulator 2026

Goals

- Integrate CI for documentation
  - build Jupyter notebooks examples
  - build C++ docs in GitHub pages for developers
- *PyNEST-NG - “sli free NEST”*
- pip install nest
- integrate NEST GPU kernel

Refactoring by/with AI

NEST PRs for code refactoring - sli2py with AI

- refactoring of NEST language interface (SLI to Py) required language change of test suite
- Effort for whole community: O(1.5k) tests in NEST
    - mostly trivialy portable
    - not all relevant
    - not all trivial
- ⏩ Let’s do a port-a-thon!

Hey! A Port-a-thon

- … difficult to motivate community 😐
- … original authors moved on 🚶
- … no bandwidth provided from scientific side 📉
  (though sometimes required)

First results

- O(120) still left to be ported
  - O(15) LLM converted tests for trying
- Surprisingly complete ports
- Found a bug (original flawed, ported correct)
- Also reviews lack interest of community

Refactoring Efforts take-aways

- exhausting community effort
- requires dedication and specific “drivers”
- requires support from “higher levels”
- technically intersting target for application of AI

⏩ See use of AI platform hands-on session (later today)!

CI-bennch process

Continuous Benchmarking

- see performance improvement/degradation of the code
- validate machine updates (new stage/MPI/compiler/nodes/…)
- reference data for new installations (→ docs)

Approaches

- external tools (wrappers)
- pre-analyses (compiler, static analyses)
- *instrumentation in code* (models, simulator, …)
- pre-loaded introspection

Instrumentation in code

```
cmake -Dwith-detailed-timers=ON -Dwith-detailed-cycletimes=ON …
```

Different levels of complexity

normal timers
: (default on) no impact on performance expected, understandable by the user

detailed timers
: (default off) may have impact on performance, relevant to many developers

detailed xxx
: (default off) potentially strong impact on performance, only relevant for few specific feature developers

BeNNch

- Separate configs
  - simulator
  - model
  - machine
  - user
  - experiment
- first implemented as template workflow, now available as a gitlab pipeline

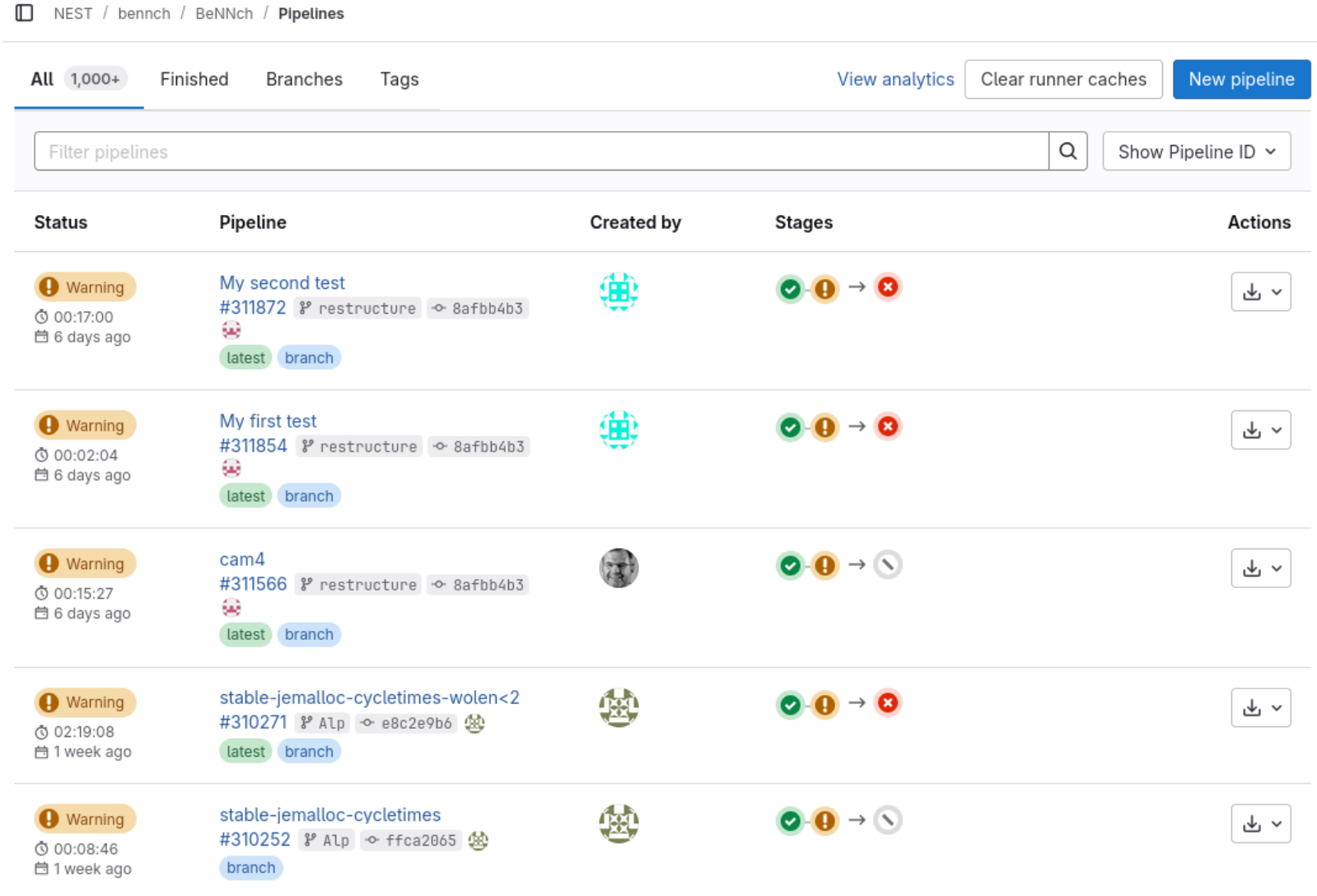
Success

- onboarding became matter of hours
  - accounts & links
  - repository overview & fork
  - start default pipeline
  - 🎉 results.

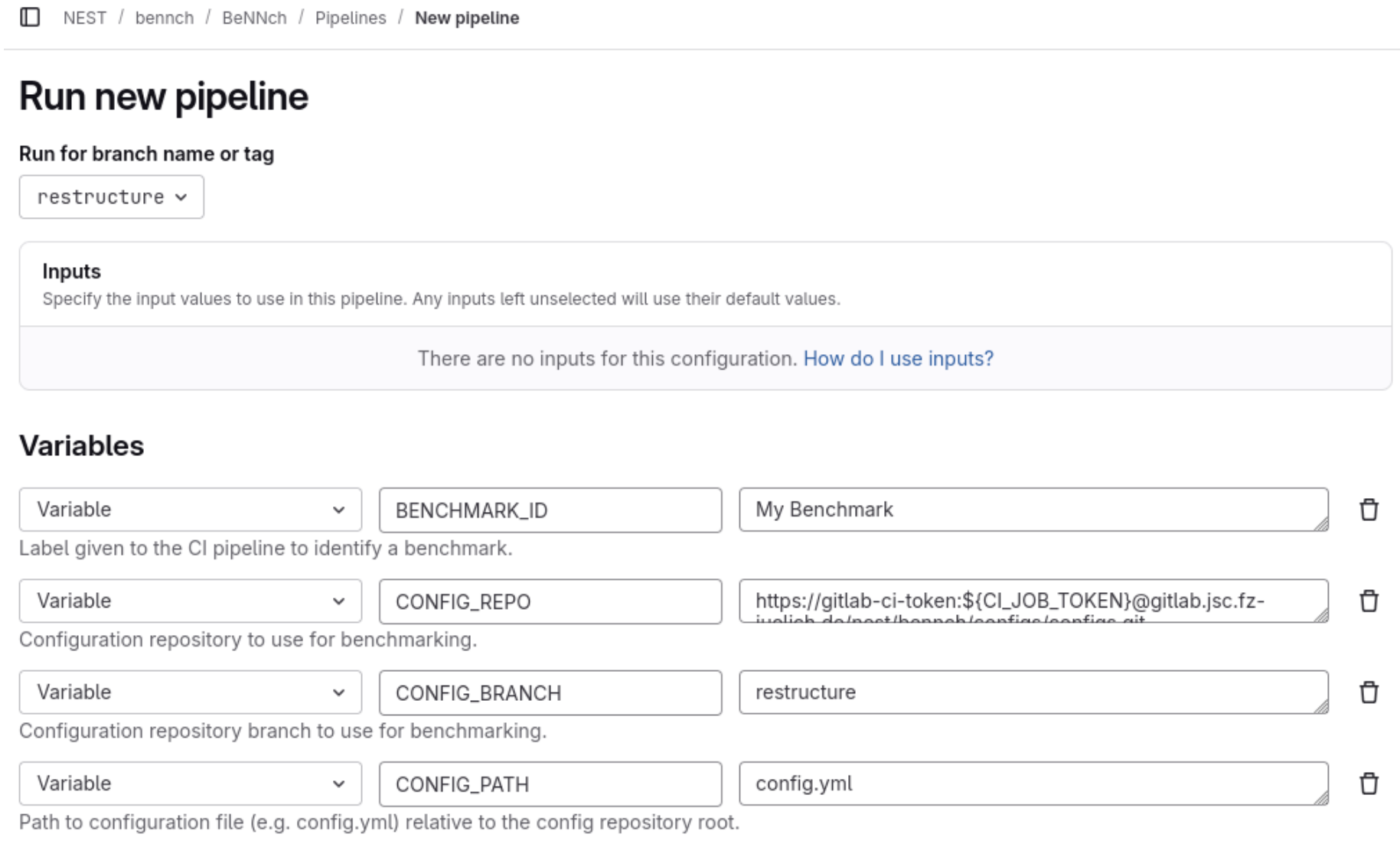
Success

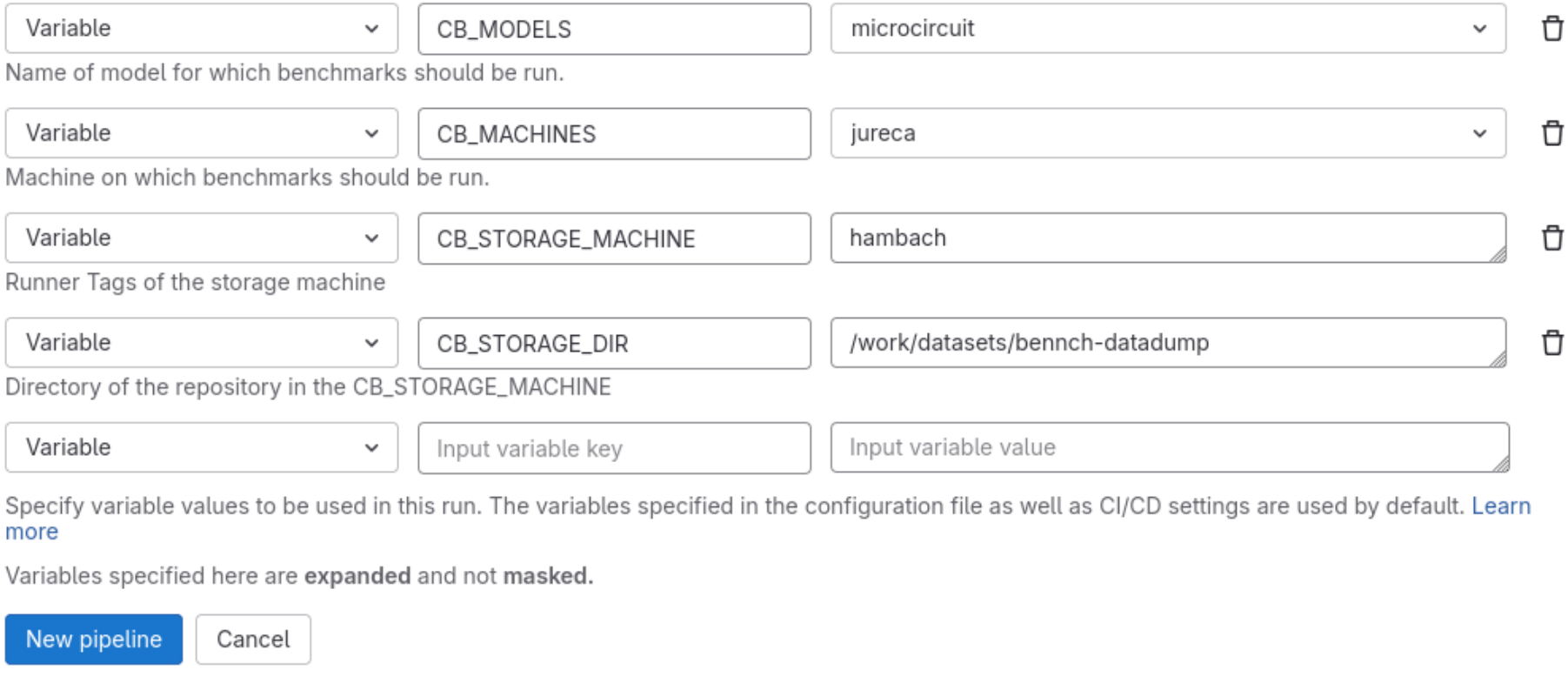
- facilitates sharing of
  - models
  - simulators (branches, features, flags, …)
  - analysis technologies
- makes work visible in pipeline view
  - who is currently doing benchmarks?
  - how was this done?
  - what were the results?

Impressions – Pipelines



Impressions – New pipeline





- Setting overaching variables
- Workflow configuraiton

Tangible results

- 12 people from ~7 groups using the same code simulateously
- 1500 pipelines since Mar 6, 2024
- 📰 paper in progress

Open points

- exploratory phase → many branches, many parallel developments
- new technologies difficult to re-integrate, as it requires
  - accept changes by others
  - review changes of others
  - willingness to learn from each other

→ usage of resources has to be coupled to proper code integration.

HiRSE AI platform

A comprehensive RAG platform for scientific code analysis, documentation generation, and intelligent querying using the Helmholtz Blablador API

✍️ Developed with Cursor

Get started

Prerequisites

- Docker and docker compose
- Helmholtz Blablador API key (Helmholtz Codebase GitLab)

See: https://helmholtz-blablador.fz-juelich.de/
Go to “documentation” link on bottom

Setup

Clone repository: https://codebase.helmholtz.cloud/j.mitchell/hirse-ai-platform
Create the .env file from template in README.md and set environment variable OPENAI_API_KEY with your Helmholtz blablador API key.

Start the platform: docker compose up --build
Access
Frontend: http://localhost:3001

Goal: Create an adaptable, useful and modular AI interface for scientific codes

Current Features

🌍 Multi-language Code and Documentation Analysis

💬 RAG-powered Query System

🤔 Script Troubleshooting

📑 Project Evaluation

🔠 Advanced Query Configuration

🐘 Large Repository Handling

✏️ Query Logging

📖 Free and Open Source

---

tags: `HiRSE` `RSE` `workshop` `report` `slides` `2025`