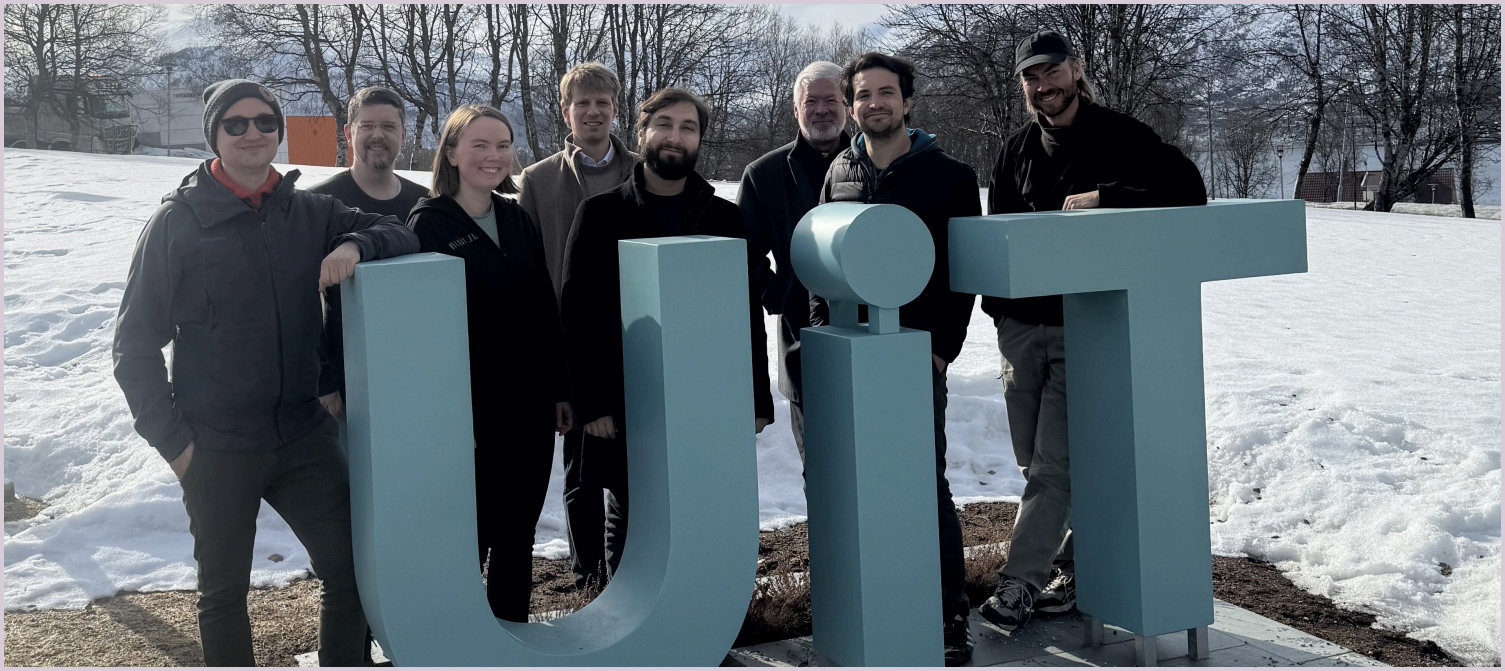


Research Software Engineering at UiT

RSE UiT

Software development is part of research

- Scientists typically spend **30%** of their time developing software
- 90% or more are primarily self-taught
- Many are unaware of tools and practices that would allow them to write **more reliable and maintainable code** with less effort

[Wilson, Greg, et al. "Best practices for scientific computing." PLoS biology 12.1 (2014): e1001745.]

Research software engineers

- … are people who combine professional software expertise with an understanding of research [https://researchsoftware.org/]
- Often people **who grew up in research** and liked computing and programming
- … or people **who come from software development** drawn towards meaningful and impactful work of academia

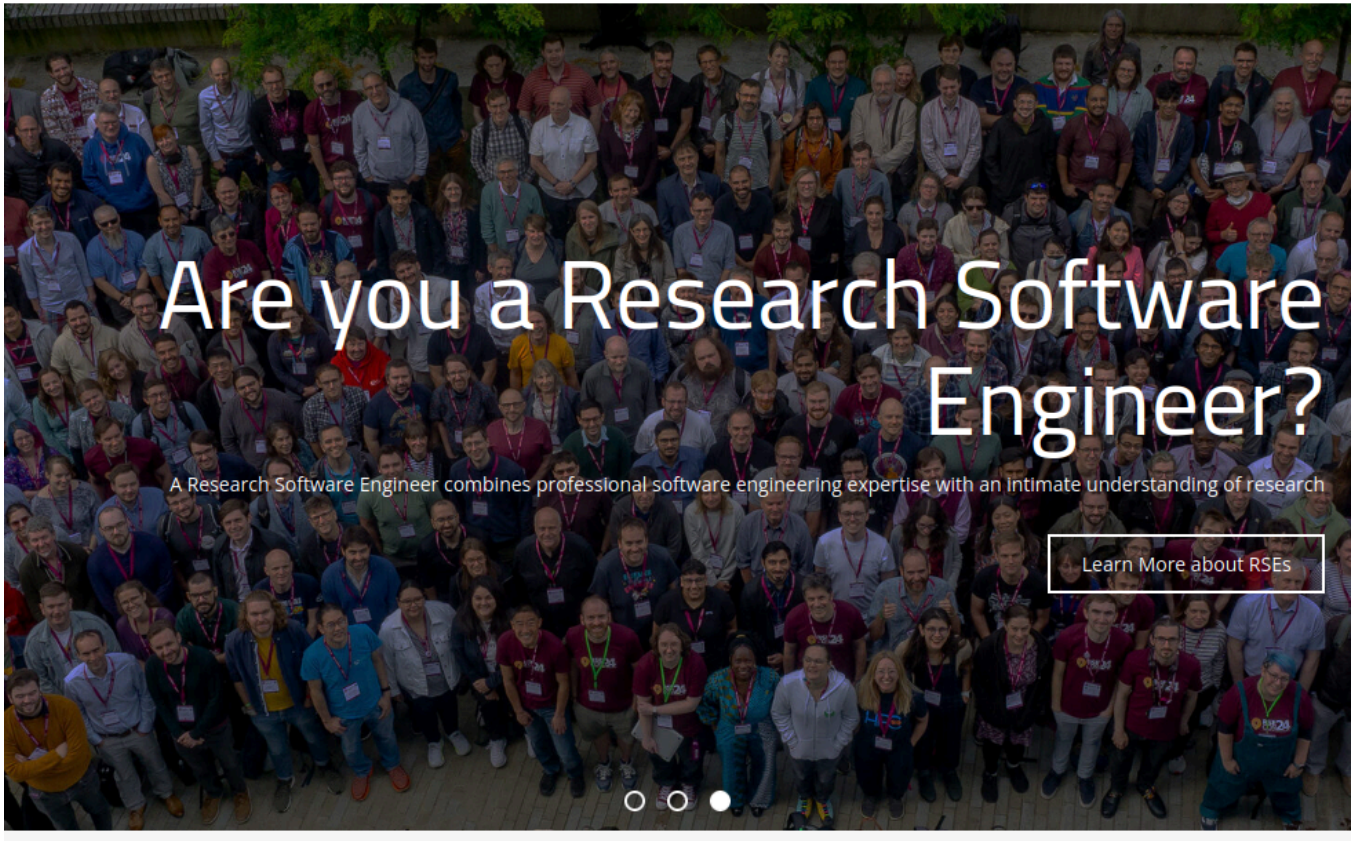
Resources

- Society of Research Software Engineering
- Recent conference: RSECon 2025
- https://nordic-rse.org/
- Nordic-RSE conference 2025

Society of Research Software Engineering

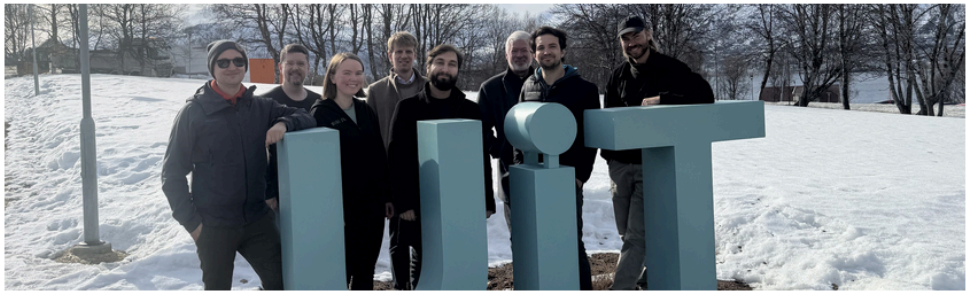
Founded on the belief that a world which relies on software must recognise the people who develop it.

About Community Events Careers News Resources Get involved Member Log In



Join the Society of Research Software Engineering

Research Software Engineering



We help all researchers and students at UiT to develop and improve their research software. This can span from simple R or Python scripts to complex GPU-enabled software packages.

Save time on your project by getting quick and comprehensive help by experts.

"I highly recommend this service to any researcher looking for reliable and efficient computational resources."

"Great support. Saved me weeks and weeks of time. Super helpful and friendly."

"RSE helped me adapt my analysis to run on the HPC computing cluster at UiT. This cut the time to run my analysis from many days down to hours."

(more testimonials)

Contact information

Email us at rse@uit.no

Learn more about Research Software Engineering:

- Nordic Research Software Engineers Association
- Society of Research Software Engineering
- A History of Research Software Engineers

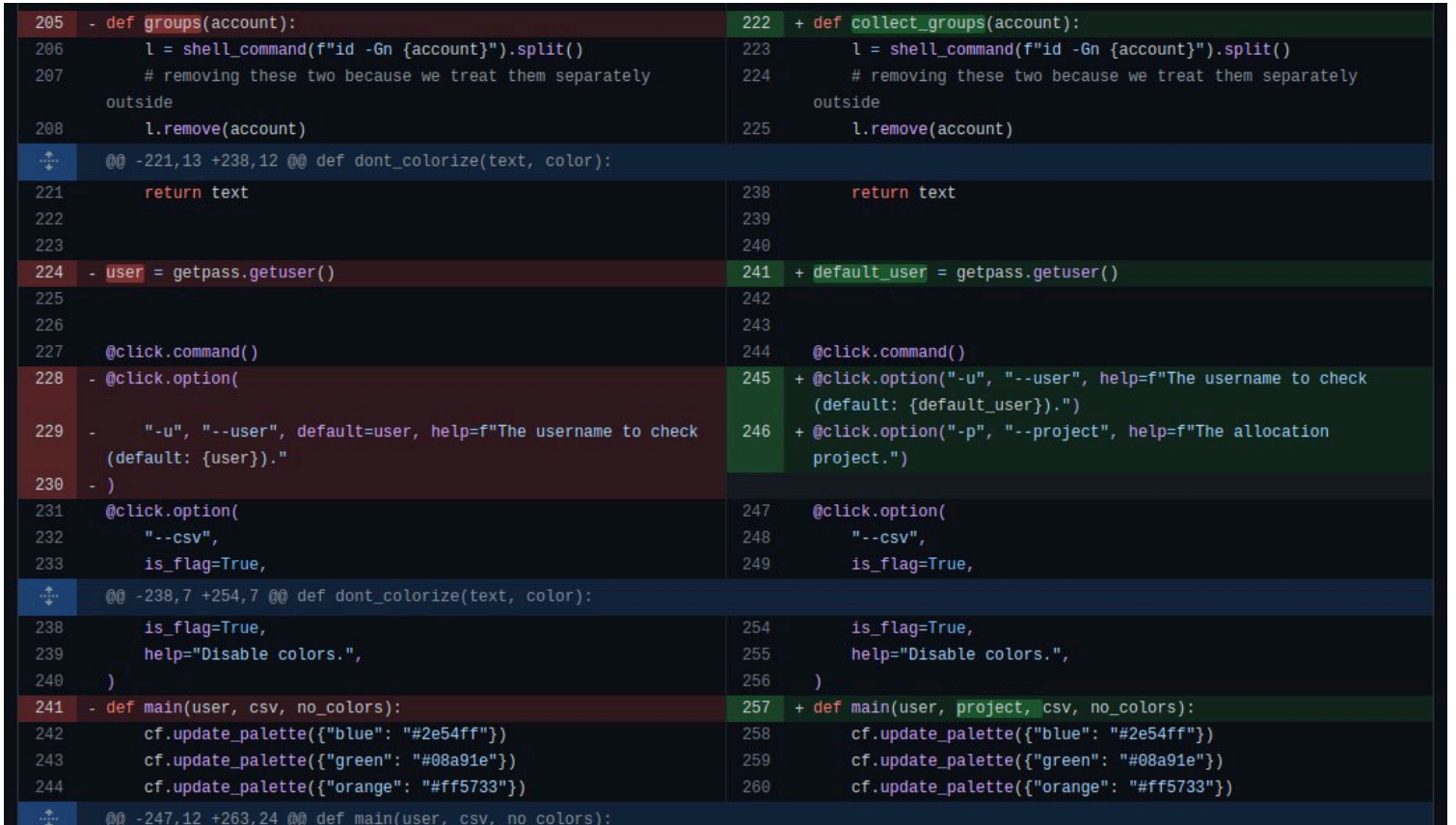
Nordic RSE

UiT's RSE group will be hosting the 2026 Nordic RSE Conference on 9-10 June. Click here for details.

Type: Facility
Unit: Department of Information

Help with improving your scripts/code

- Collaborative code review (we discuss code in a constructive way)
- Making code cleaner, faster, and more reusable
- Best practices for documentation

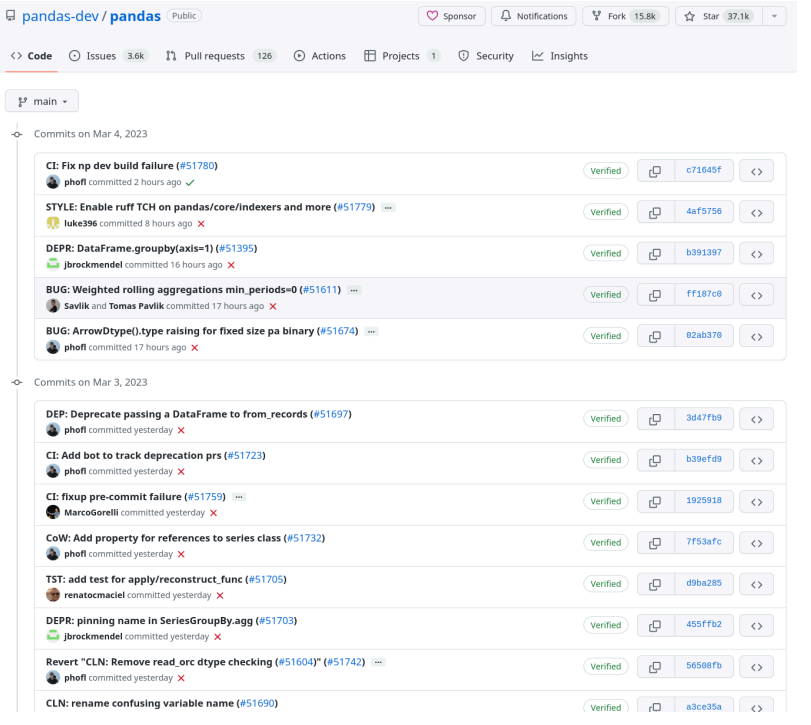


```
205 - def groups(account):
206       l = shell_command(f"id -Gn {account}").split()
207       # removing these two because we treat them separately
    outside
208       l.remove(account)
    @@ -221,13 +238,12 @@ def dont_colorize(text, color):
221       return text
222
223
224 - user = getpass.getuser()
225
226
227   @click.command()
228 - @click.option(
229 -     "-u", "--user", default=user, help=f"The username to check
    (default: {user})."
230 - )
231   @click.option(
232       "--csv",
233       is_flag=True,
    @@ -238,7 +254,7 @@ def dont_colorize(text, color):
238       is_flag=True,
239       help="Disable colors.",
240   )
241 - def main(user, csv, no_colors):
242       cf.update_palette({"blue": "#2e54ff"})
243       cf.update_palette({"green": "#08a91e"})
244       cf.update_palette({"orange": "#ff5733"})
    @@ -247,12 +263,24 @@ def main(user, csv, no_colors):

222 + def collect_groups(account):
223       l = shell_command(f"id -Gn {account}").split()
224       # removing these two because we treat them separately
    outside
225       l.remove(account)
238       return text
239
240
241 + default_user = getpass.getuser()
242
243
244   @click.command()
245 + @click.option("-u", "--user", help=f"The username to check
    (default: {default_user}).")
246 + @click.option("-p", "--project", help=f"The allocation
    project.")
247   @click.option(
248       "--csv",
249       is_flag=True,
254       is_flag=True,
255       help="Disable colors.",
256   )
257 + def main(user, project, csv, no_colors):
258       cf.update_palette({"blue": "#2e54ff"})
259       cf.update_palette({"green": "#08a91e"})
260       cf.update_palette({"orange": "#ff5733"})
    @@ -263,24 +281,24 @@ def main(user, csv, no_colors):
```

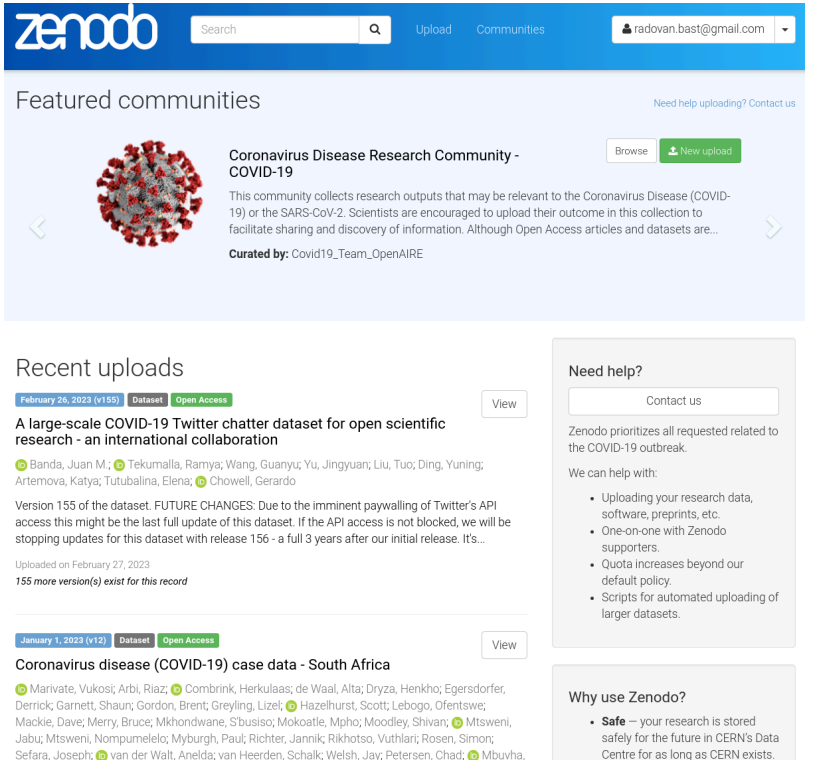
Help with organising your code

- Git, GitHub, and GitLab
- Moving your work/project/code/data to Git
- Modularizing your code
- Organization of reusable Python/R notebooks



Help with sharing your code

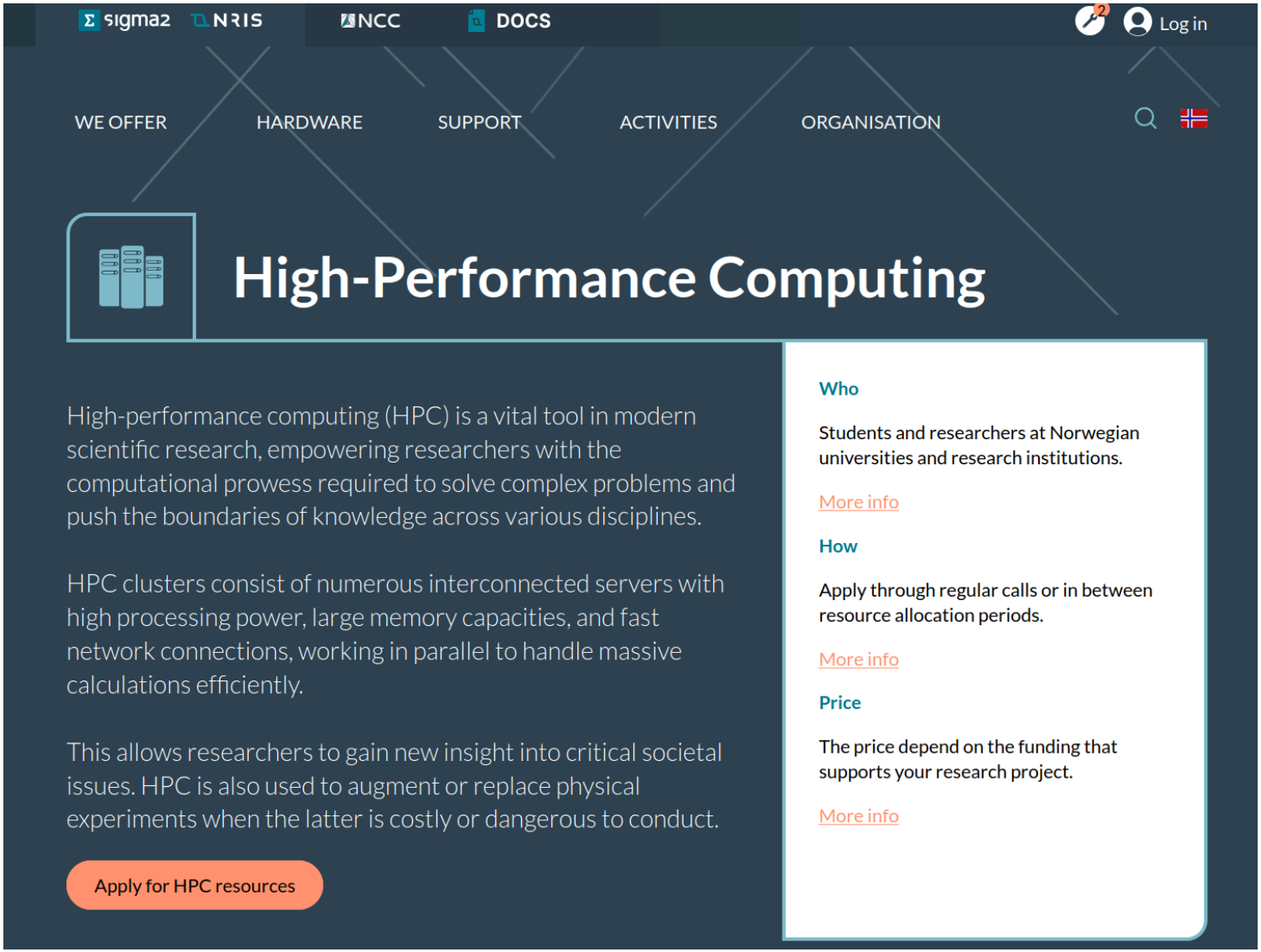
- Help with software licenses and open sourcing
- Publishing code
- Packaging and sharing software
- Containerization (Singularity, Docker)
- PyPI and Conda
- Journal of Open Source Software (JOSS)



Getting results sooner

- Improving scaling, CPU, and memory optimization
- Porting to GPU
- Moving from local computer to cloud or high-performance computing clusters
- Helping with running independent steps in parallel

[Midjourney, CC-BY-NC 4.0]

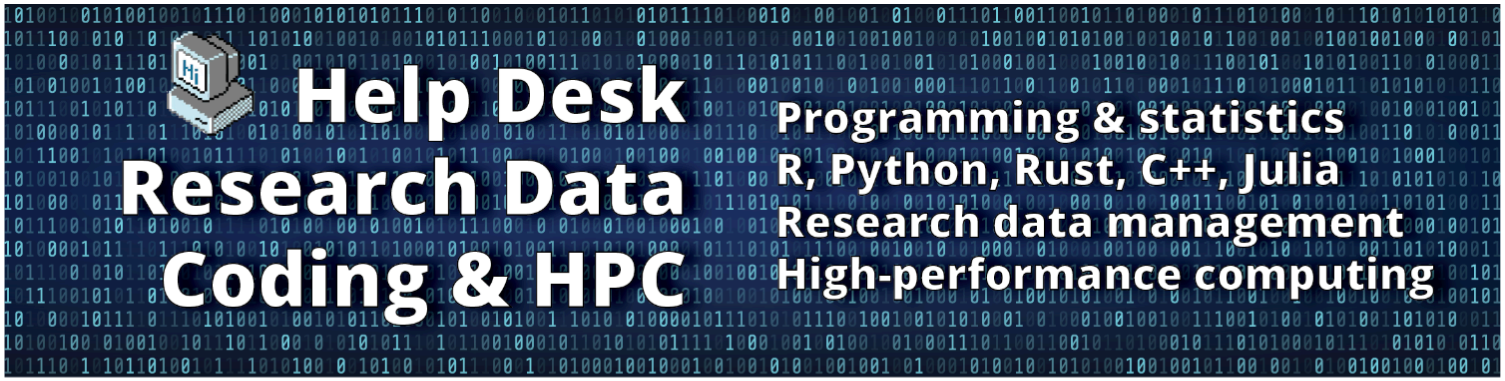

sigma2
NRIS
NCC
DOCS
Log in
WE OFFER
HARDWARE
SUPPORT
ACTIVITIES
ORGANISATION
High-Performance Computing
High-performance computing (HPC) is a vital tool in modern scientific research, empowering researchers with the computational prowess required to solve complex problems and push the boundaries of knowledge across various disciplines.
HPC clusters consist of numerous interconnected servers with high processing power, large memory capacities, and fast network connections, working in parallel to handle massive calculations efficiently.
This allows researchers to gain new insight into critical societal issues. HPC is also used to augment or replace physical experiments when the latter is costly or dangerous to conduct.
Apply for HPC resources
Who
Students and researchers at Norwegian universities and research institutions.
More info
How
Apply through regular calls or in between resource allocation periods.
More info
Price
The price depend on the funding that supports your research project.
More info

Work with us

- **RSE Help Desk:**
  2 hours on (almost) every Wednesday at the UiT Library (UB 338)
  FREE (first come/first serve)

- **Individual Consultations:**
  One-on-one with an RSE engineer
  Initial consultation free, afterwards 600 kr/hr (5-hour minimum)

- **Extended Collaborations:**
  Part-time or full-time contracts with the RSE group
  Include us in your grant applications! - 600kr/hr



Wednesdays 14:00–16:00

UB 338 (University library)

RSE
UiT Research Software Engineering

Get Superflink with data
UiT Research Data Support

NRIS
High-performance computing group at UiT

Example 1: rewrite instead of buying a 30 GB hard disk

Problematic if data is 30 GB big

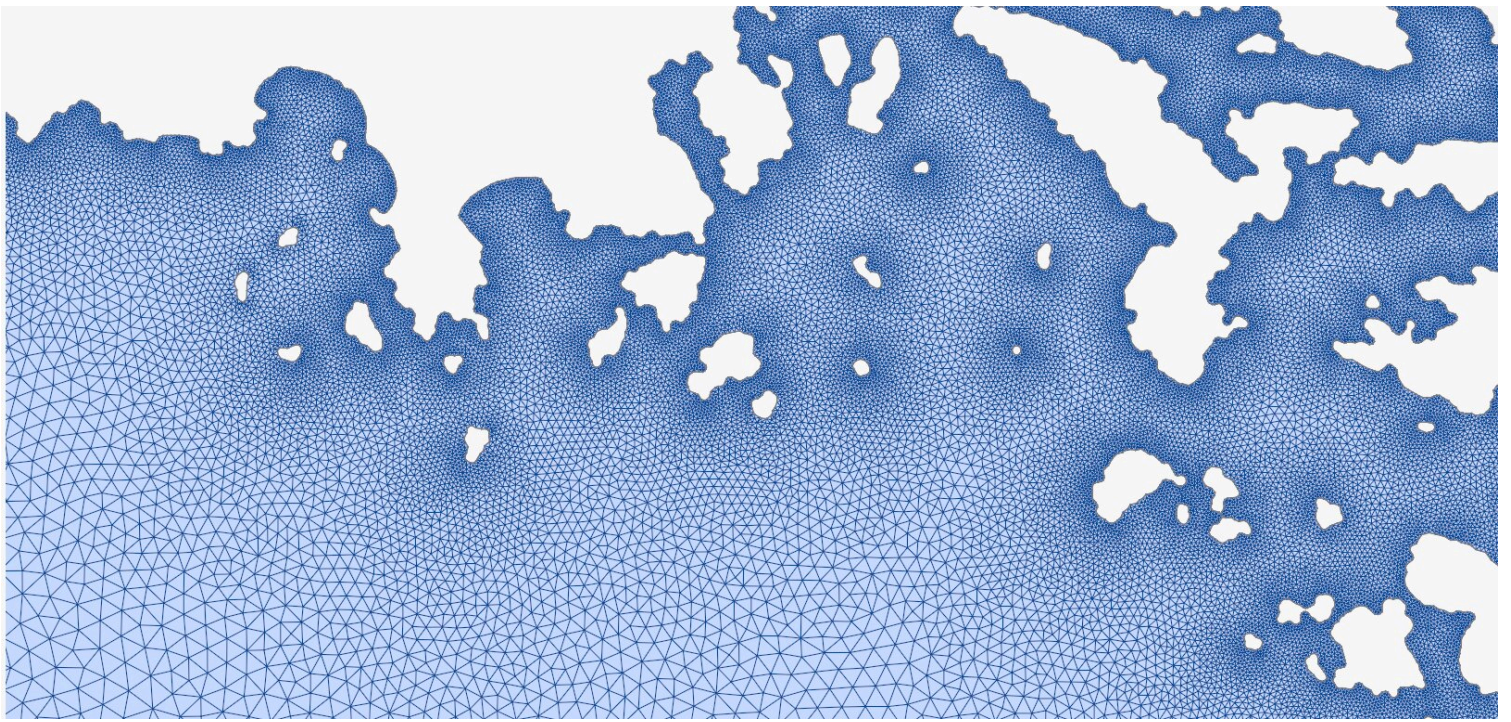
```
result = 0.0
with open("data.txt", "r") as f:
    lines = f.readlines()
    for line in lines:
        result += analyze(line)
```

Better

```
result = 0.0
with open("data.txt", "r") as f:
    for line in f:
        result += analyze(line)
```

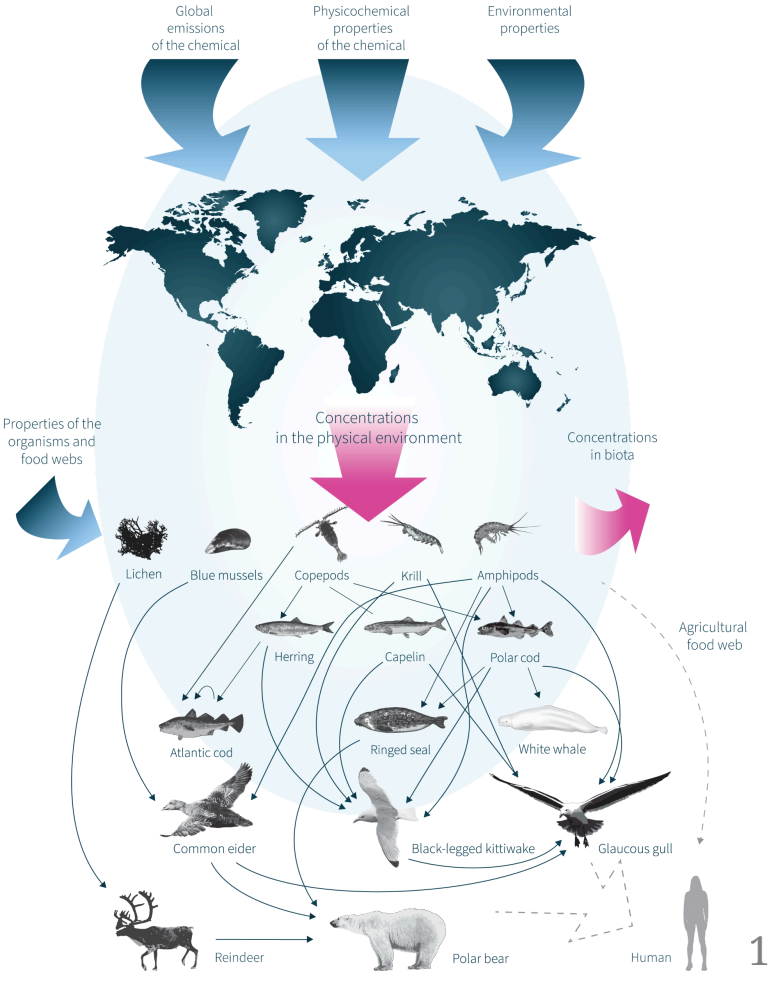
Example 2

Speed-up of grid mesh generation for oceanography code from days to seconds by a code rewrite from Matlab to Python+Rust using a more optimal algorithm



Example 3

- Bioaccumulation model for organic contaminants developed for arctic ecosystems
- Translated 10k lines of 20+ year old Visual Basic code to Python



Example 4

README AGPL-3.0 license

HeterogeneousArrays.jl

This repository introduces a type called HeterogeneousVector, a segmented array designed to hold elements of different concrete types while maintaining efficient, type-stable broadcasting.

The goal of this project is to generalize the prototype from COMBAT.jl into a more versatile and reusable package.

Key Features

- **Type-stable heterogeneous state**: Hold fields of different types in a single efficient container
- **Unitful integration**: Enforce physical correctness at compile-time
- **Zero-overhead broadcasting**: Fused broadcast operations maintain type stability
- **ODE-friendly**: Drop-in compatible with SciML solvers (DifferentialEquations.jl)
- **Physical safety**: Catch dimensional inconsistencies before they propagate

Installation

The package can be installed with the Julia package manager. From the Julia REPL type `]` and run:

```
pkg> add HeterogeneousArrays
```

Then use it in your code:

```
using HeterogeneousArrays
```

+ 1 release

Deployments 14

github-pages 5 days ago

+ 13 deployments

Packages

No packages published
Publish your first package

Contributors 3

gregordecristoforo Gregor Decri...

yaccos Jakob Peder Pettersen

Jullcifer Julia Mikhailova

Languages

Julia 100.0%

Example 5

SpeedyWeather.jl

An atmospheric model for research: friendly, interactive, extensible, and built for speed.

Get Started | View on Github | API Reference

SpeedyWeather.jl is a global atmospheric model developed as a research playground with an everything-flexible attitude as long as it is speedy. Technically it is a climate model with simple, yet interactive representations of ocean, land and sea-ice. It is easy to use and easy to extend, making atmospheric modelling an interactive experience – in the terminal, in a notebook or conventionally through scripts.

- Ported to AMD GPUs
- Currently working on increasing performance using CUDA graphs and compiling into MLIR and XLA

- Version control
- Collaboration using Git
- Testing
- Documentation
- Notebooks
- Modular code development
- Reproducible research
- Software licensing
- How to share and publish code

Typical format: 6 half-days, twice per year, online, free, live-streamed, recorded, archived asynchronous Q&A in collaborative document

Lessons and recordings: https://coderefinery.org/lessons/

Teaching at UiT

PhD course at UiT: FYS-8805 Collaborative Coding

- One week of lectures + home exam
- 5 credit points
- Lecture material is public: fys-8805-collaborative-coding.github.io/lecture-material
- Next course: 25th–27th May & 11th–12th June

Involved in Bio-3027 / Bio-8027 and Bio-3032 / Bio-8032

Development of further teaching material

RSE projects in 2026

- Gabriel works 50% for Hylleraas Center
- Gregor works for Fusenow project one day per week
- Heterogeneous arrays project: 1 month of FTE
- SpeedyWeather: 3 month of FTE
- Svalbard Shiny app: > 6 months of FTE
- Heterogeneous arryas Part II: ca 4 months of FTE
- BOUT++ (upcoming project): 3 months of FTE
- Smaller projects: 2 month of FTE
- CodeRefinery: 1.5 months of FTE
- Teaching courses: 1 month of FTE

- **Enough projects for 3 full positions**

Our challenges

- Visibility at UiT (despite a lot of progress over the last 2 years)

- finding time for RSE projects and administering the RSE group

- **we need more people!**

Har du lyst til å være med i det neste RSE-prosjektet?

Hvis ja, ta kontakt enten personlig eller via rse@uit.no

UiT organizes the next Nordic RSE conference

Welcome

Join us for the Nordic-RSE Conference 2026, bringing together Research Software Engineers from across the Nordic region and beyond for two days of learning, networking, and collaboration.

Dates: June 9-10, 2026
Location: Auditorium Cerebrum, UiT - The Arctic University of Norway, Tromsø
Times: 9:00 - 16:00 each day

This community-driven conference is for anyone working at the interface between research and digital technologies - from software developers and data scientists to system administrators and research supporters.

more info at nordic-rse.org/nrse2026

Invited Speakers

Dr. Mihaela Duta

Principal Research Software Engineer at the University of Oxford

Mihaela Duta leads the service provision for OxRSE and contributes to shaping research software strategy across the University of Oxford. She specialises in data science and applied software engineering, with particular expertise in interactive visualisation dashboards, data processing pipelines, mobile and web applications, and instrument control. She is also the co-founder of OxEd and Assessment, work that was Highly Commended in the 2022 Vice-Chancellor's Innovation and Engagement Awards for making a positive difference to society.

Mihaela also helped professionalize the RSE group at UiT by providing valuable advice on the group's organisational structure, workflows and outreach activities.

Assoc. Prof. Dr. Elisabeth Wetzer

Associate Professor in Machine Learning at UiT The Arctic University of Norway

Elisabeth Wetzer is a principal investigator at the Centre for Research-based Innovation, SFI Visual Intelligence, and the Centre of Excellence, SFF Integreat. Her research spans biomedical applications, including microscopy, spatial biology, and medical imaging techniques such as dynamic PET, mammography, and echocardiography.

Together with Assoc. Prof. Dr. Kristoffer Wickstrøm, she started a Research Software Engineering course (FYS-8805 Collaborative Coding and Reproducible Research) for PhD students at UiT.

Dr. Radovan Bast

Senior Scientist at Oceanbox

Radovan Bast is a theoretical chemist by education and research software engineer by heart who has been jumping a bit between research, computation, programming, teaching, academia, and industry, and enjoys working somewhere in-between all of these with focus on reproducibility and modular code development.

Currently he is working at an oceanography start-up and having fun with computational geometry and computational fluid dynamics, Rust, Python, WebAssembly, and algorithms.

During his time at UiT The Arctic University of Norway, where he worked with the High Performance Computing Group, Radovan founded UiT's Research Software Engineering Group and started the CodeRefinery project. He is one of the founding members of the Nordic Research Software Engineer community.

RSE Help Desk:

Wednesday 14:00-16:00

https://research-software.uit.no/contact/

Email: rse@uit.no