CS Internship: Modern Graphical User Interface for the SWEET PDE solver code

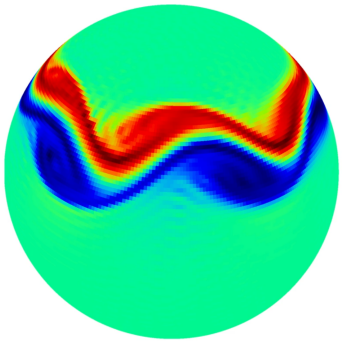
Martin Schreiber (UGA/LJK)

Location: Laboratoire Jean Kuntzmann, AIRSEA team, IMAG building, 700 avenue Centrale, 38401 Saint Martin d'Hères.

SWEET (`https://sweet.gitlabpages.inria.fr/sweet-www/`) is a PDE solver software that allows a fast exploration, investigation, and prototyping of time discretization methods for PDE solvers that use global spectral methods (Fourier for Cartesian systems & spherical harmonics on the sphere).

SWEET targets, in particular, to

- accelerate research around the development of (numerical) time integration methods
- investigate new ways to express parallelization
- do some early investigations of scalability of novel time integration methods (e.g., parallel-in-time).



Snapshot of a barotropic instability benchmark on the sphere computed with SWEET.

Rather than using operational software codes, in which it is hard to research different forms of time integration, SWEET allows to do rapid development due to a clear and powerful software design. Since it is developed with a co-design of applied mathematics and high-performance computing, HPC performance benchmarks, hence real speedup measurements, can show the potential impact once applied to operational codes.

Since its beginning, it has been successfully used to steadily pioneer the research on parallel-in-time methods targeting climate/weather simulations by investigating time integration for the horizontal parts as part of shallow-water equations, see publications at `https://sweet.gitlabpages.inria.fr/sweet-www/publications/`.

Although there are currently even more time integration approaches actively researched within SWEET, there is one drawback, as explained in the following section.

1 Overall goal

SWEET's current graphical user interface (GUI) directly allows to have a look at the simulation results, visualizes it in different ways, and even allows a certain steering of the simulation.

However, the current GUI is partly outdated in multiple ways: It's based on OpenGL, lacks certain features, requires a more modern object oriented approach, etc.

2 Objectives

1. Objective "**Vulkan**":
   Your first objective would be to make yourself (if required) familiar with the Vulkan 3D graphics API (`https://www.vulkan.org/`).

2. Objective "**SWEET**":
   In parallel to the previous objective, you'd make yourself familiar with the SWEET PDE solver.

3. Objective "**GUI development**":
   Finally, you will realize a more modern GUI approach which will support researchers in speeding up their development. As part of this, one possibility could be also to develop a GUI in Python and to steer the C++ SWEET PDE solver.

3 Prerequisites

In order to ensure a successful internship for you and us, please ensure that you fulfill all or at least most of the following prerequisites. Please point out your qualifications in the application letter (see below).

- Basic understanding of partial differential equations (Fourier-based analysis).
- Familiar with 3D rendering APIs (Vulkan is beneficial).
- Familiar with Python, C++, git.

4 Application

To apply to this internship, send an Email to martin.schreiber@univ-grenoble-alpes.fr, including your CV, a transcript including all your grades (Bachelor + Master), and a brief explanation of why you are interested in this project.