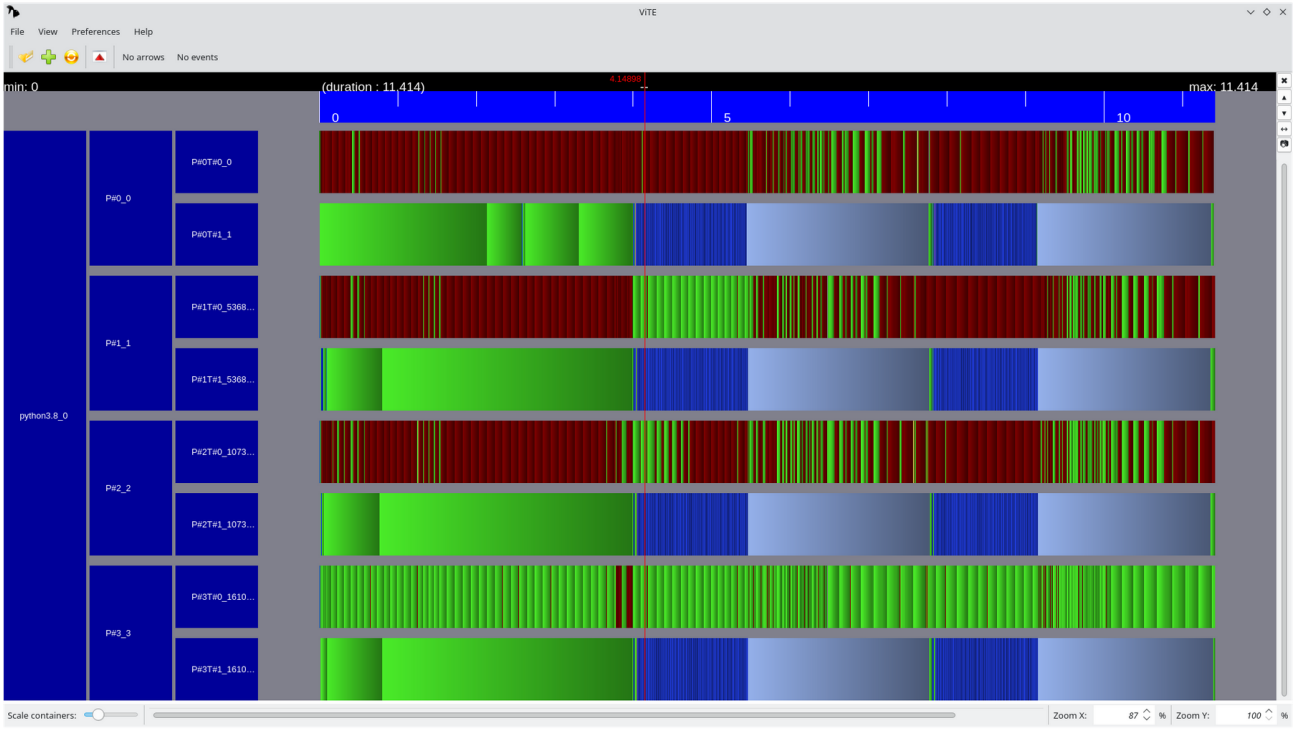
Web-based trace visualization

Web-based trace visualization

Performance analysis tools such as EZTrace generate traces that depict the execution of a parallel application. These traces can be visualized with tools such as ViTE:



Visualizing large traces is tedious because many events have to be represented. Typically ViTE fails to display traces larger than 500MB due to memory constraints. Additionnally, displaying large traces may require more pixels than the screen resolution.

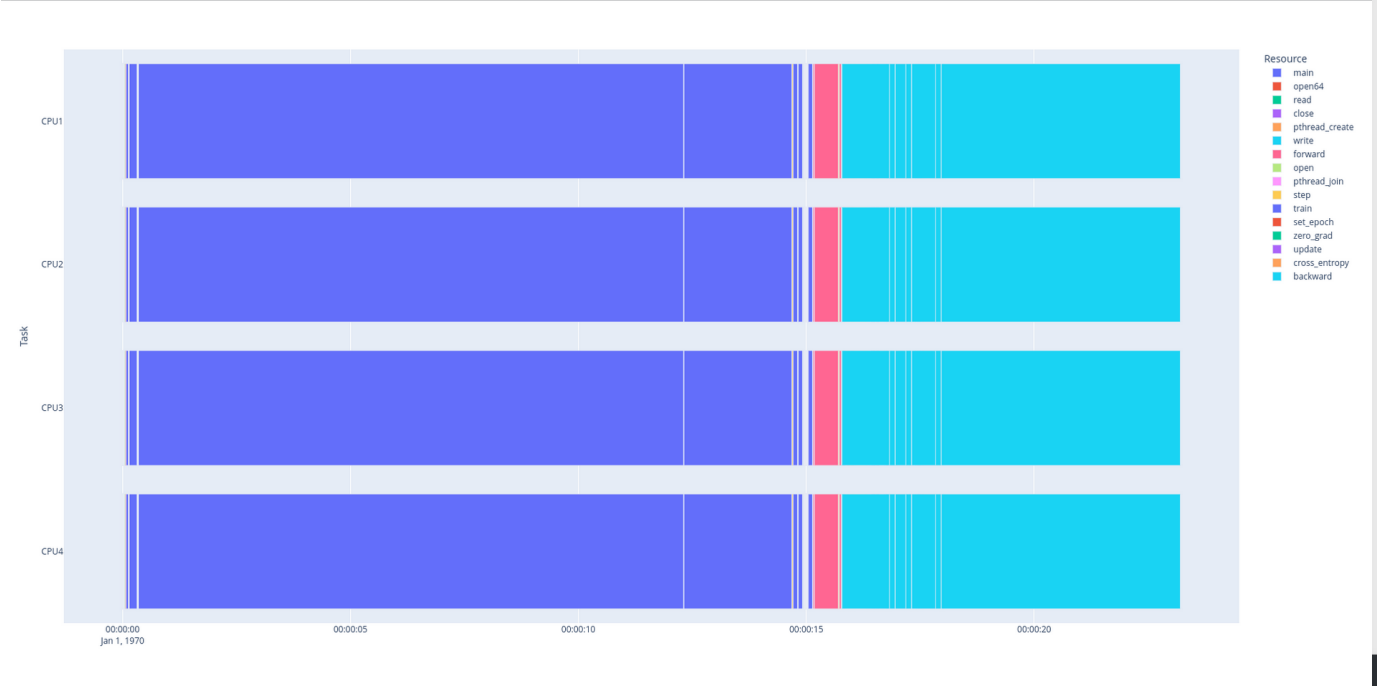
The Benagil group at Inria develops a new trace format named Pallas. This trace format detects patterns of events, and represents traces in a structured way:

```
Tag       Duration     Event
S7        2.00004882   E0_S S6 Ed_S
├─E0_S    0.00003671   THREAD_BEGIN()
├─S6      2.00000076   E1_E L1 Ec_L
│ ├─E1_E  0.00001130   Enter 1 (Working)
│ ├─L1    1.99998946   2 * S5 = L0 S4
│ │ ├─S5  0.09816532   E8_E E9_S Ea_S Eb_L
│ │ └─S5  1.90182414   E8_E E9_S Ea_S Eb_L
│ └─Ec_L  0.00001135   Leave 1 (Working)
└─Ed_S    THREAD_END()
```

This hierarchical view of a trace could be used for displaying a trace efficiently. Instead of displaying all the events of a trace, the visualization tool could display the top-level sequences, and only enter the lowest-level sequences when the user zooms in.

Objectives

The goal of this project is to design a tool for visualizing and exploring large Pallas traces. The main idea is to use the Plotly python library to represent the traces. In order to speed up the processing of large traces, the visualization tool would communicate with a trace parsing process in charge of reading the Pallas trace, filtering events (in order to limit the quantity of data to display), and transfer events.



The trace visualizer will consist of two parts.

- The trace reader is in charge of reading the hierarchical trace and summarize the events to display for a thread during a time period. Since the number of events to display may be high (typically millions of events), the trace reader will select a (small) subset of the events to display
- The trace rendering part will get the list of events to display, and render them in a web browser. The web page allows to navigate through the trace, and to zoom in a subset of the trace. When zooming in, the trace renderer asks the trace reader for a new list of events.

Technical details

During this project, you will learn:

- data visualization in Python with Plotly
- Interactive data visualization with live update
- Calling C++ code from a Python program

Contact

François Trahay francois.trahay@telecom-sudparis.eu