Contribution ID: **59** Type: **Talk (15min + 5min)**

The Helmholtz Analytics Toolkit (Heat) and its role in the landscape of massively-parallel scientific Python

Wednesday, March 6, 2024 4:30 PM (20 minutes)

When it comes to enhancing exploitation of massive data, machine learning methods are at the forefront of researchers'awareness. Much less so is the need for, and the complexity of, applying these techniques efficiently across large-scale, memory-distributed data volumes. In fact, these aspects typical for the handling of massive data sets pose major challenges to the vast majority of research communities, in particular to those without a background in high-performance computing. Often, the standard approach involves breaking up and analyzing data in smaller chunks; this can be inefficient and prone to errors, and sometimes it might be inappropriate at all because the context of the overall data set can get lost.

The Helmholtz Analytics Toolkit (Heat) library offers a solution to this problem by providing memory-distributed and hardware-accelerated array manipulation, data analytics, and machine learning algorithms in Python. The main objective is to make memory-intensive data analysis possible across various fields of research —in particular for domain scientists being non-experts in traditional high-performance computing who nevertheless need to tackle data analytics problems going beyond the capabilities of a single workstation. The development of this interdisciplinary, general-purpose, and open-source scientific Python library started in 2018 and is based on collaboration of three institutions (German Aerospace Center DLR, Forschungszentrum Jülich FZJ, Karlsruhe Institute of Technology KIT) of the Helmholtz Association. The pillars of its development are…

- to enable memory distribution of n-dimensional arrays,
- to adopt PyTorch as process-local compute engine (hence supporting GPU-acceleration),
- to provide memory-distributed (i.e., multi-node, multi-GPU) array operations and algorithms, optimizing asynchronous MPI-communication (based on mpi4py) under the hood, and
- to wrap functionalities in NumPy- or scikit-learn-like API to achieve porting of existing applications with minimal changes and to enable the usage by non-experts in HPC.

In this talk we will give an overview on the current state of our work. Moreover, focussing on the research software engineering perspective we will particularly address Heats role in the existing ecosystem of distributed computing in Python as well as technical and operational challenges in its further development.

Slot length

Primary authors: COMITO, Claudia (Forschungszentrum Jülich, Jülich Supercomputing Centre); Dr GÖTZ, Markus (Karlsruhe Institute of Technology); GUTIRREZ HERMOSILLO MURIEDAS, Juan Pedro (SCC); HAGEMEIER, Björn (Forschungszentrum Jülich GmbH); HOPPE, Fabian (DLR - Institut für Softwaretechnologie - HPC); KNECHTGES, Philipp (DLR); KRAJSEK, Kai (Forschungszentrum Jülich GmbH); RUETTGERS, Alexander (German Aerospace Center (DLR)); TARNAWA, Michael (Forschungszentrum Jülich)

Presenter: HOPPE, Fabian (DLR - Institut für Softwaretechnologie - HPC)

Session Classification: Parallelization and HPC Infrastructure

Track Classification: HPC and Massive Data: Workflowmanagement for Parallel Computing