On the use of containers for machine learning and visualization workflows on JUWELS

Bing Gong, Jan Vogelsang, Amirpasha Mozaffari, Martin Schultz

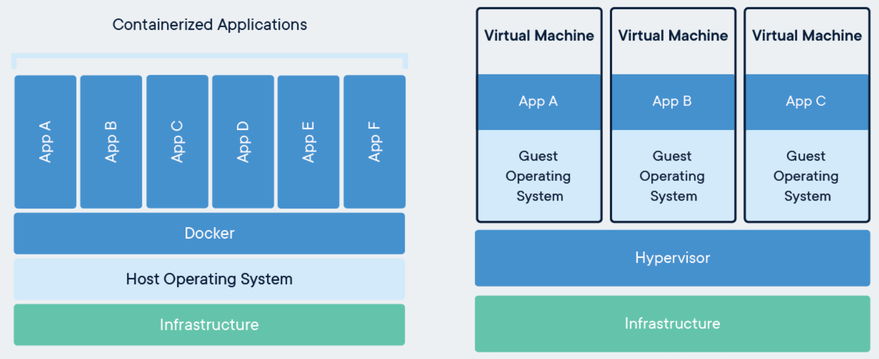
Earth System Data Exploration (ESDE), Jülich Supercomputing Centre (JSC)

Contact : b.gong@fz-juelich.de

Motivation

Containers stock a single code along with its dependencies so it can run reliably and efficiently in different computing environments. They promise the same level of isolation and security as a virtual machine and a higher degree of integration with the host operating system (OS).

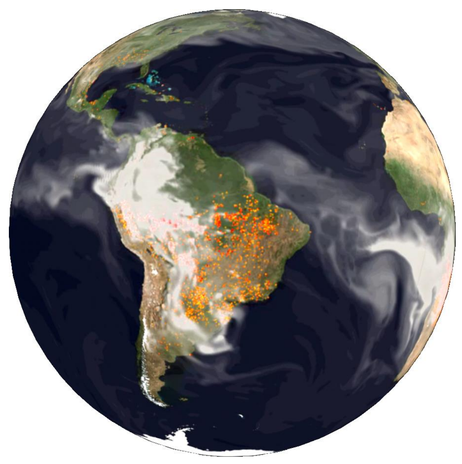
The main benefits of containers are, from a user perspective: greater software flexibility, reliability, ease of deployment, and portability. Containers have become very popular on cloud systems, but they have not been used much in HPC environments. In this study, we have tested the use of containers and measured the performance of the containerized workflow for two separate applications in the HPC system.



left) containerized applications docking into the system OS and hardware
right) applications on a virtual machine require additional layers
(source: Docker.com)

1. Visualization workflow with Paraview

This workflow contains the visualization process of global wildfire activity and the resulting “smoke” plumes from earth system model simulations. The motivation for this workflow was to expedite the process of visualizing new fire situations without having to engage several people along the workflow from data extraction, data transformations, and the actual visualisation.



a.Data

a. The different data sets provided by Copernicus Atmosphere Monitoring System (CAMS) had to be harmonized to visualize them together. Therefore, the geographical and temporal dimensions were adjusted to match each other.

b.Workflow

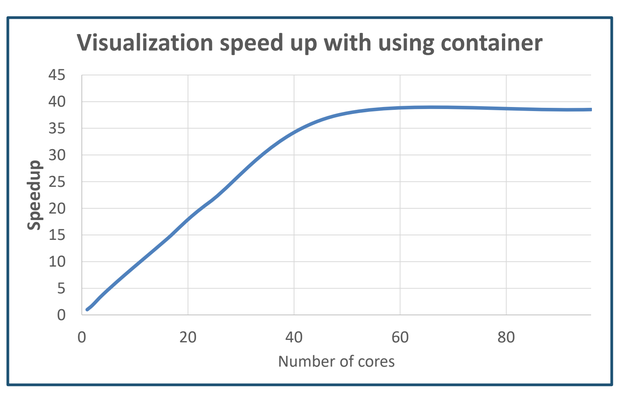
b. The visualization workflow utilizes Paraview to first create 2D textures for each timeframe. They are subsequently mapped onto a sphere creating the 3D-animation.

c. Containerization

c. For easy and flexible usage, we built a container using the Singularity container platform which pre-processes an arbitrary CAMS dataset as first step of the workflow.

d. HPC

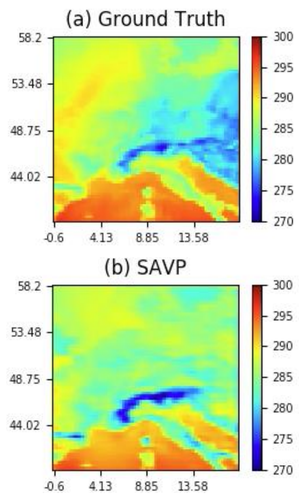
d. Preliminary results on the JUWELS system in the Jülich Supercomputing Center (JSC) show a satisfactory scaling of the application across multiple nodes.



Visualization speed up across multiple cores using container on JUWELS.

2. Video frame prediction for weather forecasting

One similarity between the deep learning tasks of video prediction and weather forecasting is that they both explore the spatio-temporal patterns from historical data to generate and forecast future scenes. This study explores the adaptation of a state-of-the-art video prediction approach based on cutting-edge deep learning architectures to predict surface temperatures over Europe for up to 24 hours.

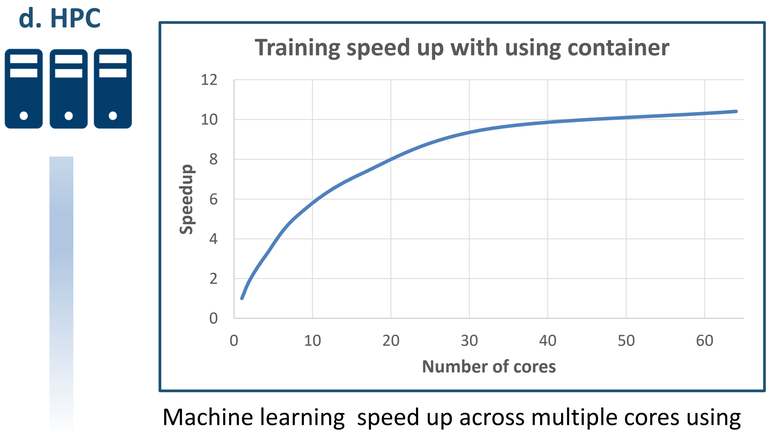


a. ERA5 reanalysis data provided by the European Centre for Medium Range Weather Forecast (ECMWF) was extracted to generate “images” of 64 by 64 pixels. The variables temperature, geopotential, and surface pressure were selected as the three image channels.

b. The workflow contains the entire lifecycle of temperature prediction that requires immense data processing, training of the deep neural network, performance evaluation, and visualization. Using TensorFlow and Horovod the workflow can run in parallel on several nodes.

c. We used the docker and Sarus technologies to containerize this workflow application and to port it to different HPC machines at CSCS.

d. The containerized application has been set up on 64 nodes (i.e. 64 GPUs) on the Piz Daint HPC system at CSCS. Preliminary results show a satisfactory level of scalability.



Machine learning speed up across multiple cores using container on Piz Daint

Conclusion and Outlook

We have implemented the containerization of two different workflows and have successfully deployed them on two different tier-0 massive parallel machines as proof of concept. The containers run with good performance and scale well on both systems. Nevertheless, at this point it is questionable if containers really provide advantages over straightforward implementation of the workflows, because they depend to a great extent on the underlying software stack – much more so than in typical cloud environments.

IntelliAQ is funded by the EU’s ERC programme, Grant Agreement 78576.

DeepRain is funded by the Bundesministerium für Bildung und Forschung (BMBF) under grant Agreement 01 IS18047A-E.