Use of the HPC Containers in the way towards exascale

Jan Meizner¹, Marian Bubak¹,², Jan Kapała¹, Piotr Nowakowski¹, Patryk Wójtowicz¹

¹ACC Cyfronet AGH, Kraków, Poland
²Department of Computer Science, AGH University of Science and Technology

http://dice.cyfronet.pl/
• Motivation and objectives
• Project Use Cases
• Road Towards Exascale
• Why Singularity?
• PROCESS Platform Architecture
• Singularity Step Implementation
• Conclusions and Further Plans
The PROCESS project aims to:

• Pave the way towards exascale by providing scalable platform
• Enable deployment of services on heterogeneous infrastructures
• Support different domains of science and business

Our objective is to:

• Build the Container based platform based on Singularity
• Integrate the HPC resources across multiple countries
• Provide effortless user experience via the WebUI
Project Use Cases

Health
- Content-based search and classification
- HAUTE ECOLE SPECIALISEE DE SUISSE OCCIDENTALE (CH)

Astronomy
- Square Kilometre Array SKA
- STICHTING NETHERLANDS E-SCIENCE CENTER (NL)

Risk Management
- Supporting innovation based on global disaster risk data
- LUDWIG-MAXIMILIANS-UNIVERSITAET MUENCHEN (DE)

Airlines Operations
- Ancillary pricing for airline revenue management
- LUFTHANSA SYSTEMS GMBH & CO KG (DE)

Agriculture
- Agro-Copernicus
- LUDWIG-MAXIMILIANS-UNIVERSITAET MUENCHEN (DE)
Road Towards Exascale

- No single HPC system capable of exceeding exaflop for generic computations
- Fastest known: < 0.2 Eflop/s
- 1-5 still way bellow 1 Eflop/s
- We may need to pool resources of multiple Centers
- Has been done for years but we want to:
  - Enable running single computation on multiple sites
  - Provide support for non-IT scientists to prepare/run codes on such massive scale
Why Singularity?

- Why container?
  - Small footprint
  - Less overhead
  - Quick launch
  - Manageable images

- Why Singularity?
  - Built for the HPC
  - Integrated with SLURM
  - Unprivileged / secure
  - Support for MPI, GPU, ...
• User accesses WebUI
• Service layer is used to:
  • select inputs
  • choose code version
  • prepare and run computations
• Computations are scheduled on the HPC via RIMROCK
• Computations may be in form of classical scripts or Singularity (new)
Singularity Step (1/5)

1. SIGN-IN
2. DASHBOARD
3. CREATE PROJECT
4. CREATE PIPELINE
5. RUN COMPUTATION
Singularity Step (2/5)

1. SIGN-IN
2. DASHBOARD
3. CREATE PROJECT
4. CREATE PIPELINE
5. RUN COMPUTATION
Singularity Step (3/5)

1. **SIGN-IN**
2. **DASHBOARD**
3. **CREATE PROJECT**
4. **CREATE PIPELINE**
5. **RUN COMPUTATION**
Singularity Step (4/5)

1. SIGN-IN
2. DASHBOARD
3. CREATE PROJECT
4. CREATE PIPELINE
5. RUN COMPUTATION
Singularity Step (5/5)

1. SIGN-IN
2. DASHBOARD
3. CREATE PROJECT
4. CREATE PIPELINE
5. RUN COMPUTATION
Conclusions and Further Plans

• Conclusions:
  • New type of the pipeline step based on the Singularity containers was created for the Interactive Execution Environment
  • It enables execution of user provided applications on the HPC
  • Tested on the HPC system at Cyfronet (CPU and GPU)

• Future plans:
  • Extending solution for other Clusters in PROCESS (in Germany, Slovakia and the Netherlands)
  • Enable running same computation on multiple sites
  • Handle site-to-site communication
  • Add support for the private and public clouds
See more at our sites

http://dice.cyfronet.pl

http://www.process-project.eu

PROCESS H2020 Project 777533