Development and implementation of an environmentally friendly Workload Management System in DiracX

WLCG Environmental Sustainability Workshop
12 December 2024

Henryk Giemza
Warsaw Technical University
National Centre For Nuclear Research

Disclaimer and about

- This presentation shows a very early stage of research that started just 2 months ago, and are the main focus of my PhD studies
- Motivation to submit the abstract was the email from the WLCG Office
- The studies are done in a form of co-opeartion between scientific institution and a company (special program run by the polish Ministry of Science and Higher Education).

Goals

Development of an optimized job allocation algorithm for the Workload Management System (WMS) in DiracX, aimed at reducing energy consumption and maximizing the utilization of renewable (low emission?) energy sources, maintaining high efficiency of calculations.

The solution is going to be deployed, tested and further optimized in the LHCb grid.

First look at the algorithm

The job allocation should be done in such way, that in overall:

Energy consumption is minimized
(Economic reason)

CO_2 emission is minimized
(Environmental reason)

Number of events processed is maximized
(Scientific reason)

Job can only be allocated to a resource center (RC) that have **all features** required to **successfully** run it
(Limitations)

Challenges

- Job allocation
  - Assignment of resource centers to all jobs, in a way that the cost is optimal, seems to be similar to problem described in [1] (i.e. NP-hard)
  - Depending on the VO constraints, optimization may be very hard/impossible, when the system is fully loaded (no free slots, waiting jobs in DIRAC)
- Data sources
  - Some data, like e.g. CO_2 emission per HEPScore are not available (yet)

Techniques

- Numerical optimizations
- Clustering
- Heuristics
- Machine Learning
  - Reinforced Learning

(Some) Factors to be considered in the algorithm

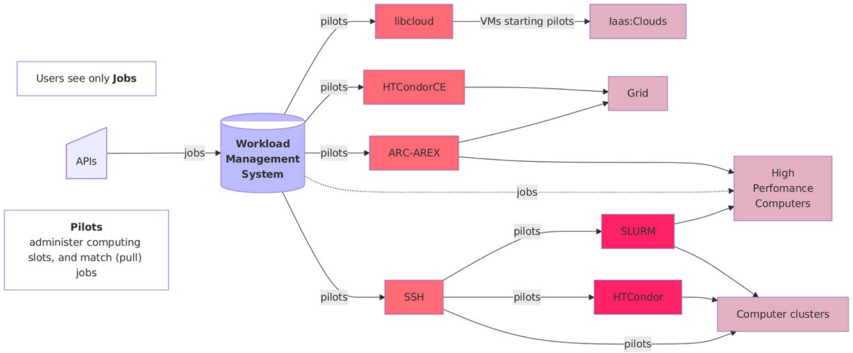
- Current CO_2 emission in given country/region
  - can be extended by forecasts, if available
- Resource Center features
  - Basic (number of slots, pledges, etc.)
  - PUE
  - Other
    - Local renewable energy sources
    - Dynamic downclocking (can it be enabled/disabled by sending a hint in a job? at later stage?)
- Renaissance of MJF?
- Use of less efficient processors when green energy is available
- Postpone jobs execution
- Rescheduling failed job at the best RCs

Simulations

- Simulations are going to be used to determine the efficiency of the new algorithm and compare it with the efficiency of the classical (current) method
- Option 1: Mini grid-like test environment(s)
  - A mixture of jobs and resource centers with different features will be created
- Option 2: Simulation in a real grid system
  - Some adjustments, like e.g. selection of CEs with homogeneous worker nodes may be needed
  - During periods of low activity

Implementation

- New version (**GreenWMS**) requires changes in multiple places:
  - WMS
  - Job and JobWrapper
  - Pilot
  - Matcher
- In case of adding the hint system, changes have to be supported by the RC (CE/Local queuing system)



Source: The neXt Dirac incarnation, F.Stagni, CHEP24

Evaluation

- We will start by enabling GreenWMS on a small part of the production system first
- Carefully monitor the behavior
- If no issues, go full scale
- Compare results with the results of WMS (which can be collected between the update of the monitoring and starting the GreenWMS)

Thank you!

References

[1] Environmentally-Conscious Cloud Orchestration Considering Geo-Distributed Data Centers, 1st International Workshop on Low Carbon Computing