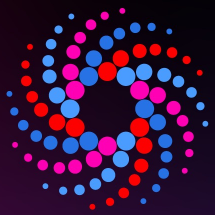


SC22
Dallas, TX | hpc accelerates.

Overview of European initiatives focusing on the orchestration of HPC, AI, and Big data

Workflows Community BoF

Alberto Scionti, ACROSS/LINKS - Jorge Ejarque, eFlows4HPC/BSC

EuroHPC JU and its projects

- The European High Performance Computing Joint Undertaking (EuroHPC JU) is a joint initiative between the EU, European countries and private partners to develop a World Class Supercomputing Ecosystem in Europe
  - Procuring and deploying pre-exascale and petascale systems in Europe
    - These systems will be capable of running large and complex applications demanding the composition of HPC, AI and data analytics
  - Support for research and innovation activities
    - Call on Jan 2020: EuroHPC-02-2019:
      High Performance Computing (HPC) and data driven HPC software environments and application-oriented platforms

ACROSS

eFlows4HPC

MICROCARD

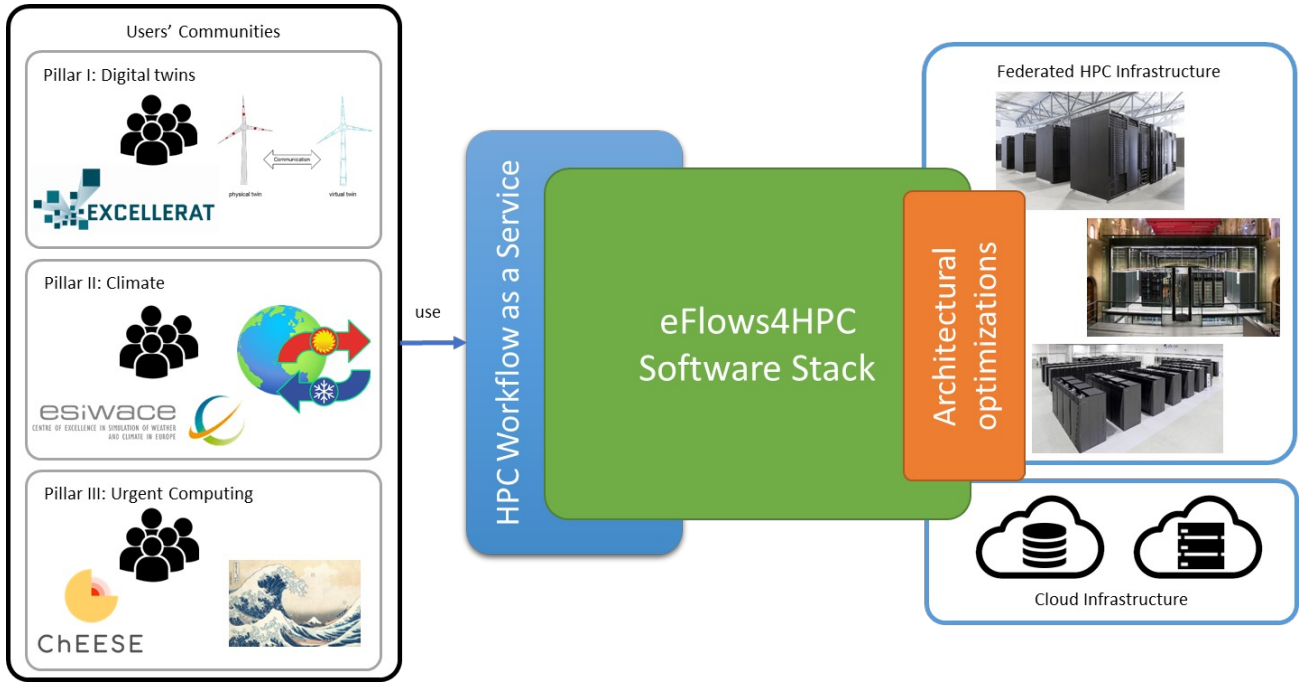
HEROES

REGALE
Open Architecture for Exascale Supercomputers

eFlows4HPC Project

eFlows4HPC objectives

- Software tools stack that make it easier the management of complex workflows
  - Combine different frameworks
    - HPC, AI + data analytics
  - Reactive and dynamic workflows
    - Automatic workflow steering
  - Full lifecycle management
    - Not just execution
    - Data logistics and Deployment
- HPC Workflows as a Service:
  - Mechanisms to make it easier the use and reuse of HPC by wider communities



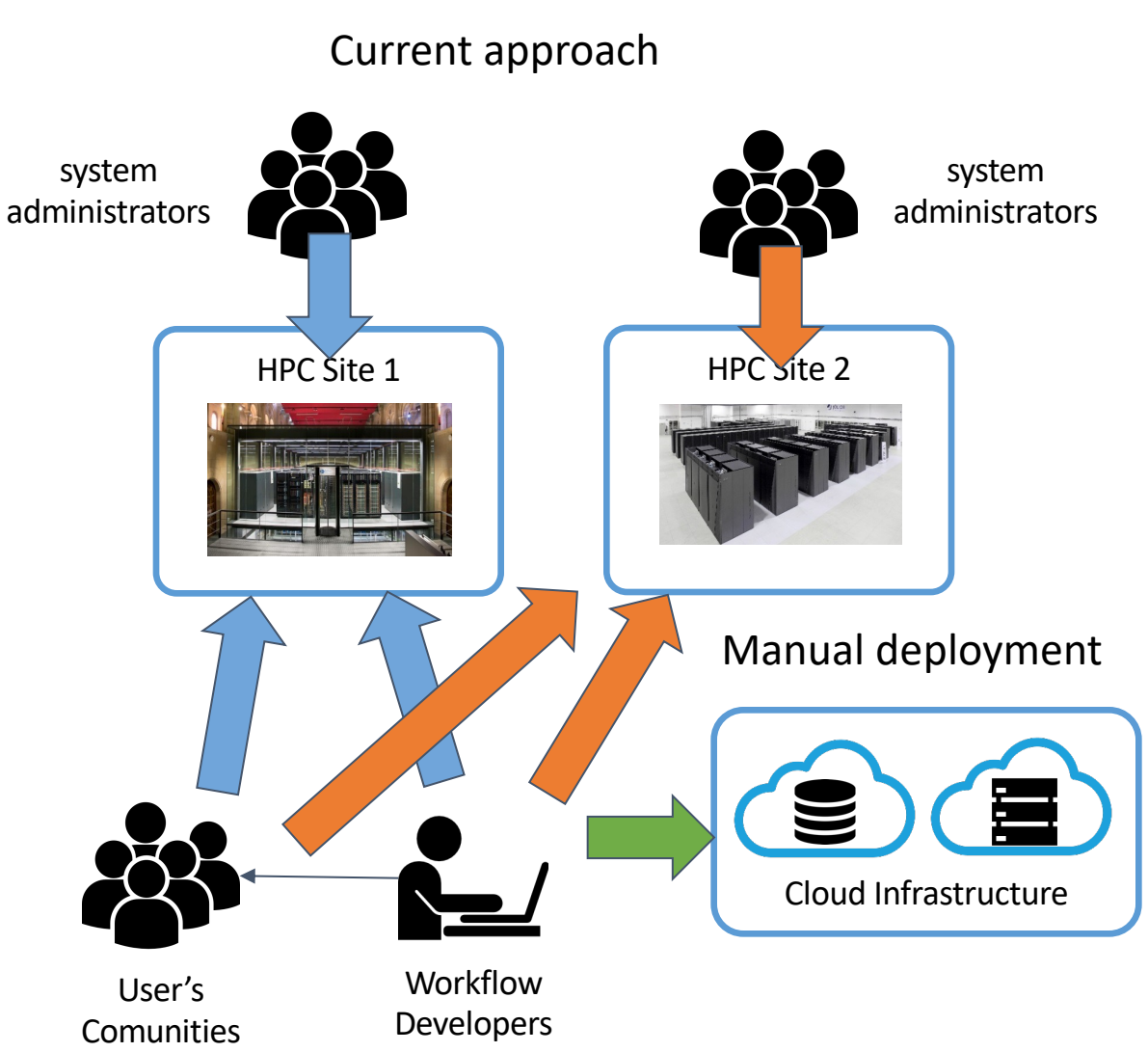
www.eFlows4HPC.eu

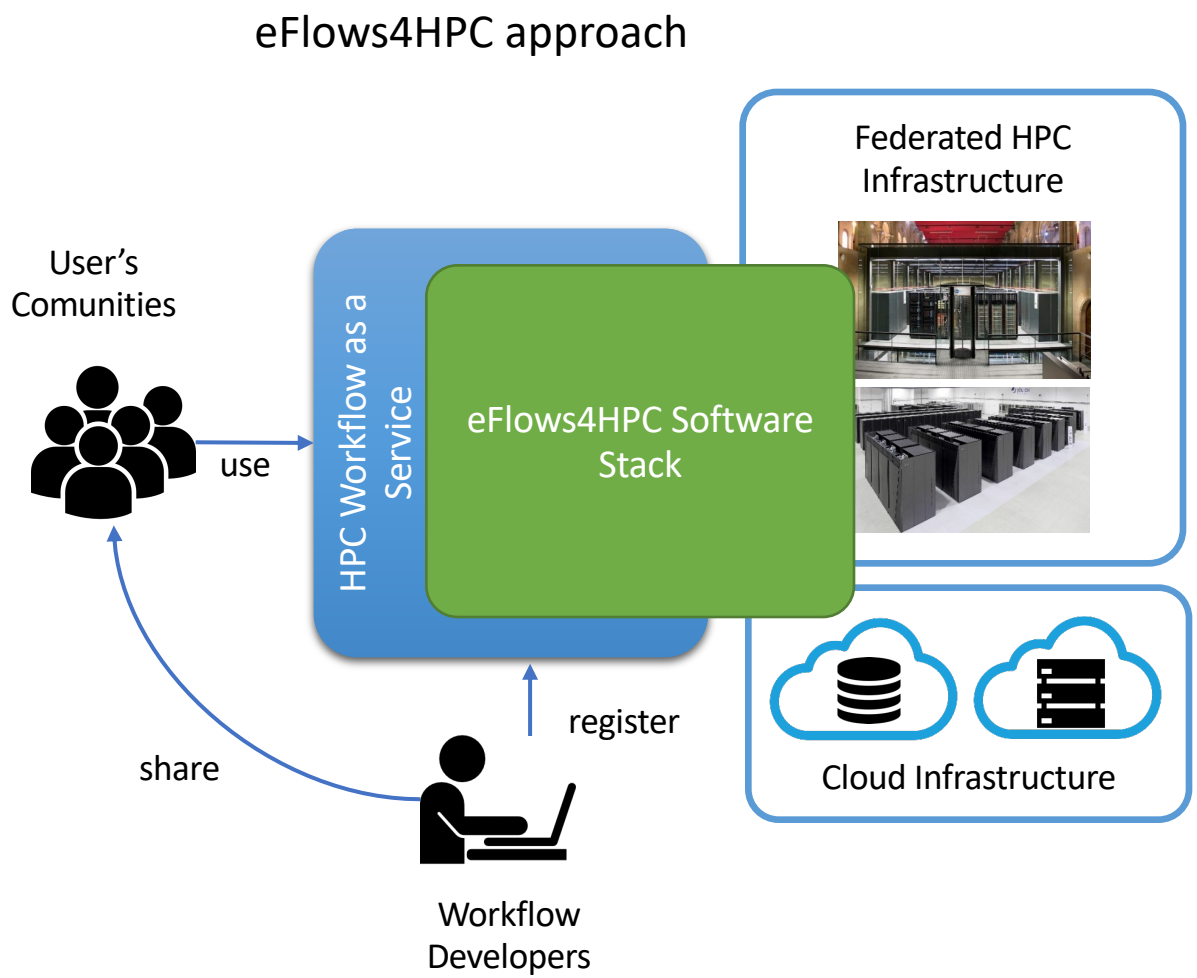
@eFlows4HPC

eFlows4HPC Project

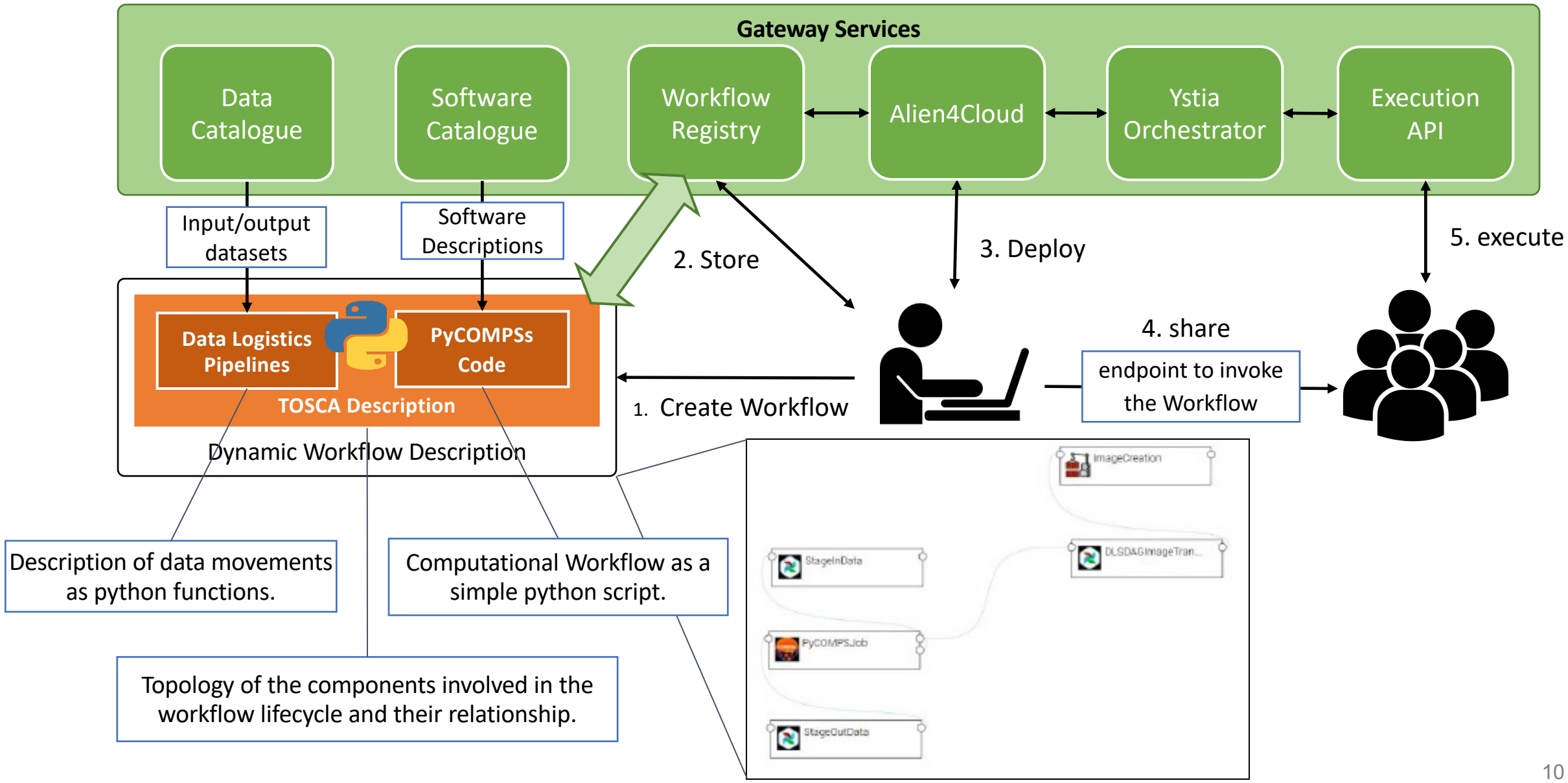
This project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 955558. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Spain, Germany, France, Italy, Poland, Switzerland, Norway.

Motivation

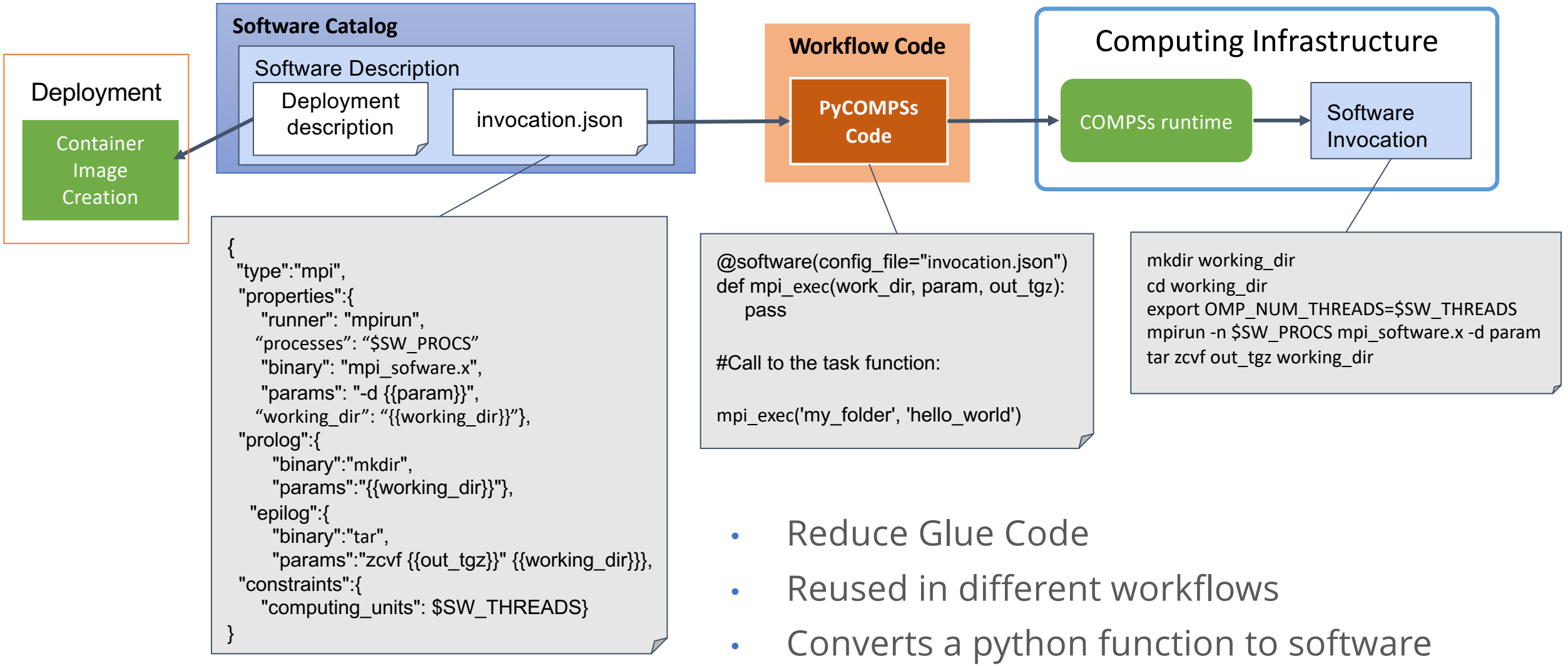

Current approach
system administrators
system administrators
HPC Site 1
HPC Site 2
Manual deployment
Cloud Infrastructure
User's Comunities
Workflow Developers


eFlows4HPC approach
Federated HPC Infrastructure
User's Comunities
use
HPC Workflow as a Service
eFlows4HPC Software Stack
register
share
Cloud Infrastructure
Workflow Developers

Workflow development overview



Integrating Software in workflow



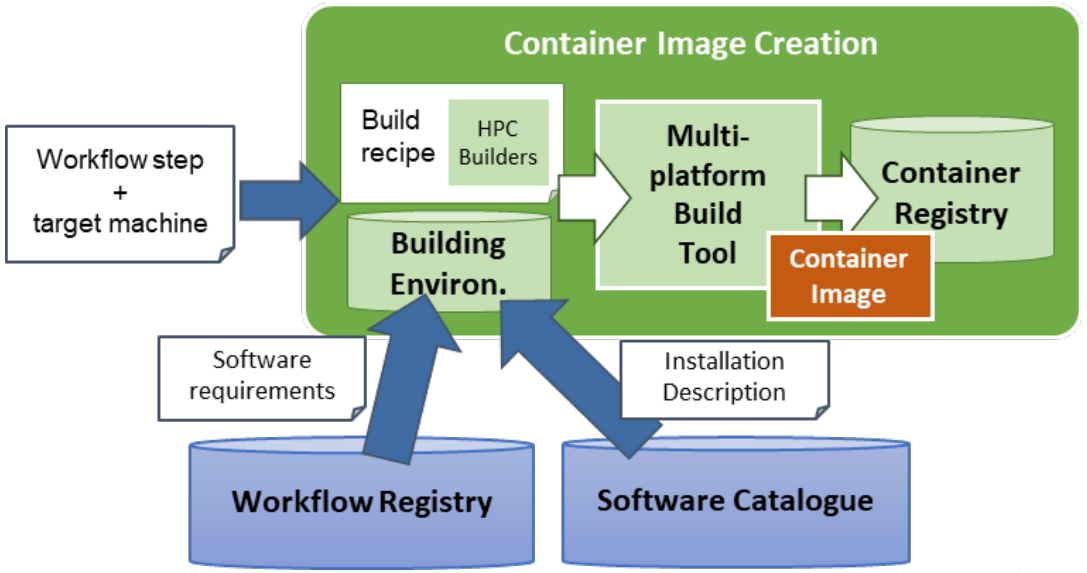
- Reduce Glue Code
- Reused in different workflows
- Converts a python function to software invocation as a PyCOMPSs task

Deployment and Execution Overview

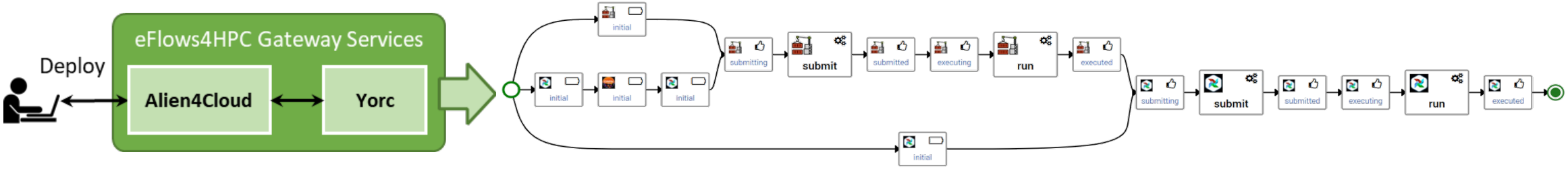
Deployment through Containers for specific systems

- Standard Containers
  - Generic compilations
  - More portable, but less performance
- HPC Ready containers
  - Compilation with Architecture Optimizations
  - Device specific compatibility (MPI/GPUs)
  - Less portable, but performance similar to bare metal

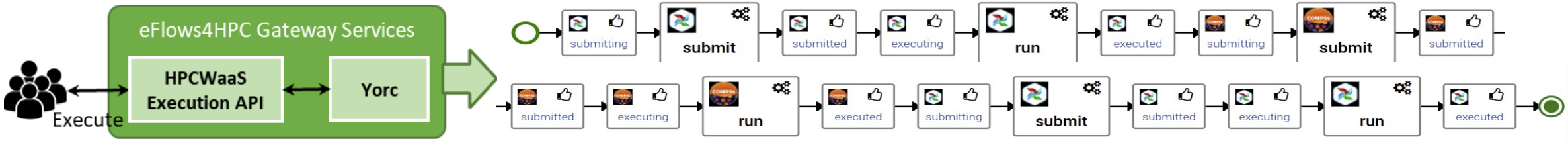
Service to generate tailored images



Application deployment workflow (once per HPC site)



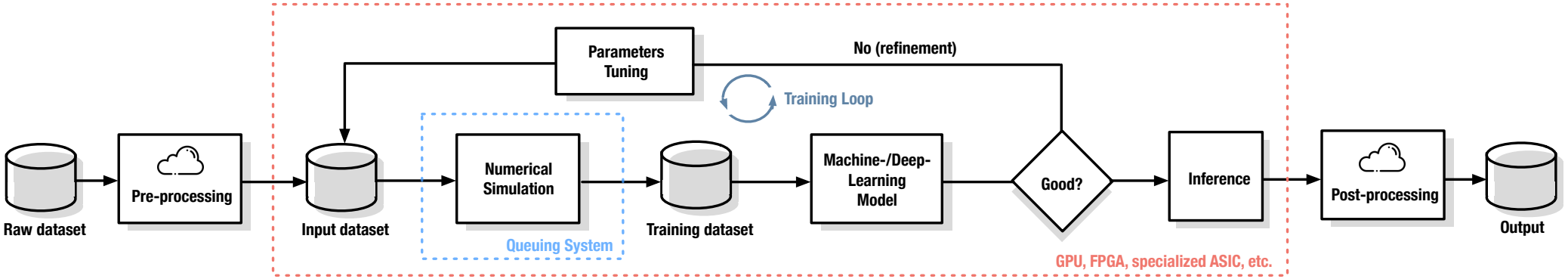
End-User workflow (multiple executions)



ACROSS Project

Motivations

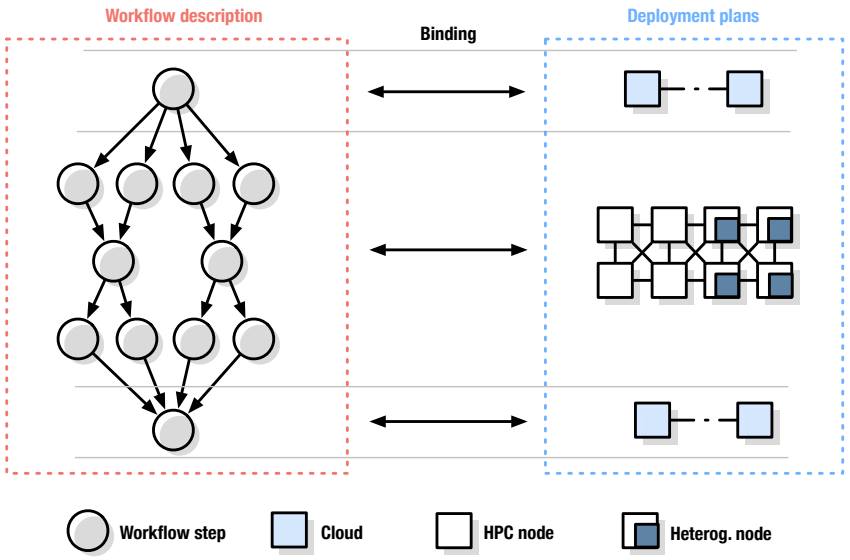
- Complex heterogeneous application workflows:
  - Mixing (large) numerical simulations, machine learning, deep learning, HPDA, etc.
- Heterogeneous infrastructural resources:
  - Cloud computing, HPC, edge computing, etc.
  - Hardware specialization (GPUs, FPGAs, AI-tailored architectures, specialized interconnections, storage hierarchy)
- Energy efficiency in the loop



Need to revise the way orchestration is done to catch/tackle such application complexity

Workflow management

- Decoupling workflow description (which are the steps composing the workflow) from deployment plans (where to execute each step)[1]:
  - Management of loops and streaming
  - Binding workflow steps with their specific deployment plans (execution environment)
  - Deployment plan statically defined at the beginning by the user:
    - Precluding possible (workflow-aware) optimizations



[1] *StreamFlow: https://streamflow.di.unito.it*

Need to introduce a more dynamic way of defining deployment plans (i.e., resource allocation)

Workflow management

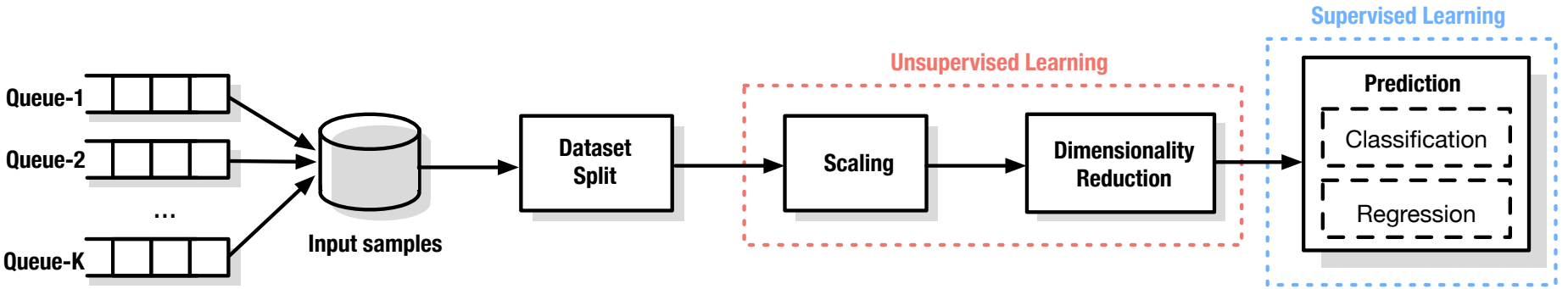
- Exploring optimization opportunities:
  - Better composition of resources in the (HPC) deployment plan
  - (Global) overview over the workflows:
- Improve workflow execution by shortening the overall makespan:
  - Need more deterministic allocation of resources and workflow steps execution

Two approaches:

- Smart job submission to the queuing system
- ‘On-demand’ resource reservation

Smart job submission

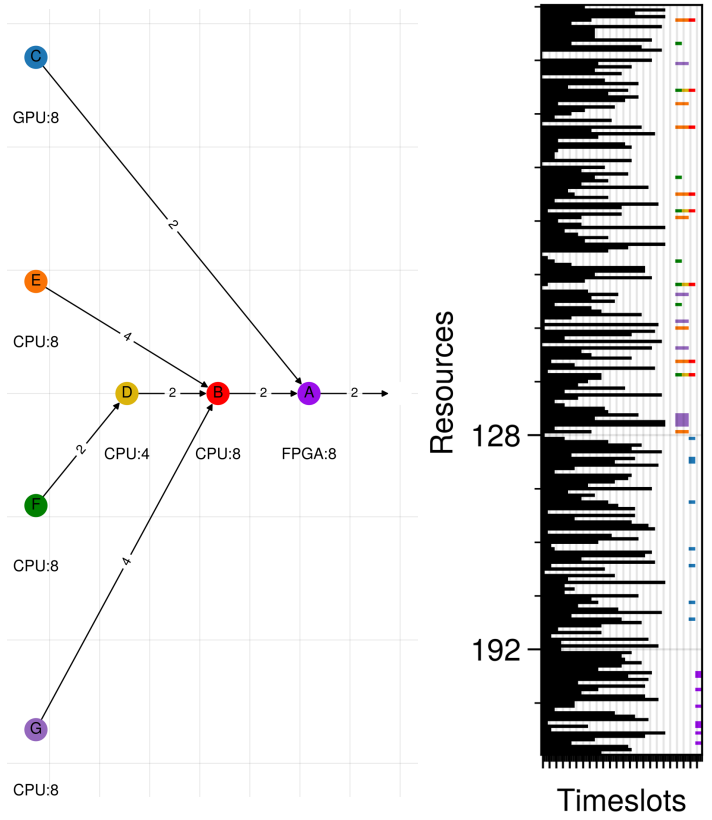
- Predicting job features to ‘guess’ the best point in time when to perform the job submission:
  - Machine learning (ML) / Deep learning (DL) in the loop:
    - Capture job features that are not easy to translate into scheduling rules
    - Require to collect and access to a large bunch of (training) data
    - Strong dependency from the infrastructural settings
    - Strong temporal dependency among jobs
    - Require to (often) retrain the models



Exploitable but not enough mature

On-demand resource reservation

- Smart way of making on-demand reservations of (HPC) resources:
  - Not breaking the underlying management policies already in place in the queuing system
  - Provide a more controllable way of introducing determinism
  - User-space mechanism
  - Making the deployment plan setting dynamic
  - Provide room for inter-workflow resource allocation:
    - Combinatorial problem definition
  - Easy to combine with a mechanism to provide fine grain resource allocation (i.e., less than a single entire node)



Under development of the EuroHPC-JU ACROSS project