Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

INFN

ICSC Centro Nazionale di Ricerca in HPC, Big Data and Quantum Computing

High-throughput analyses flagship updates

Tommaso Tedeschi (INFN Perugia)

Spoke2 INFN Meeting - 12 September 2024 - ZOOM

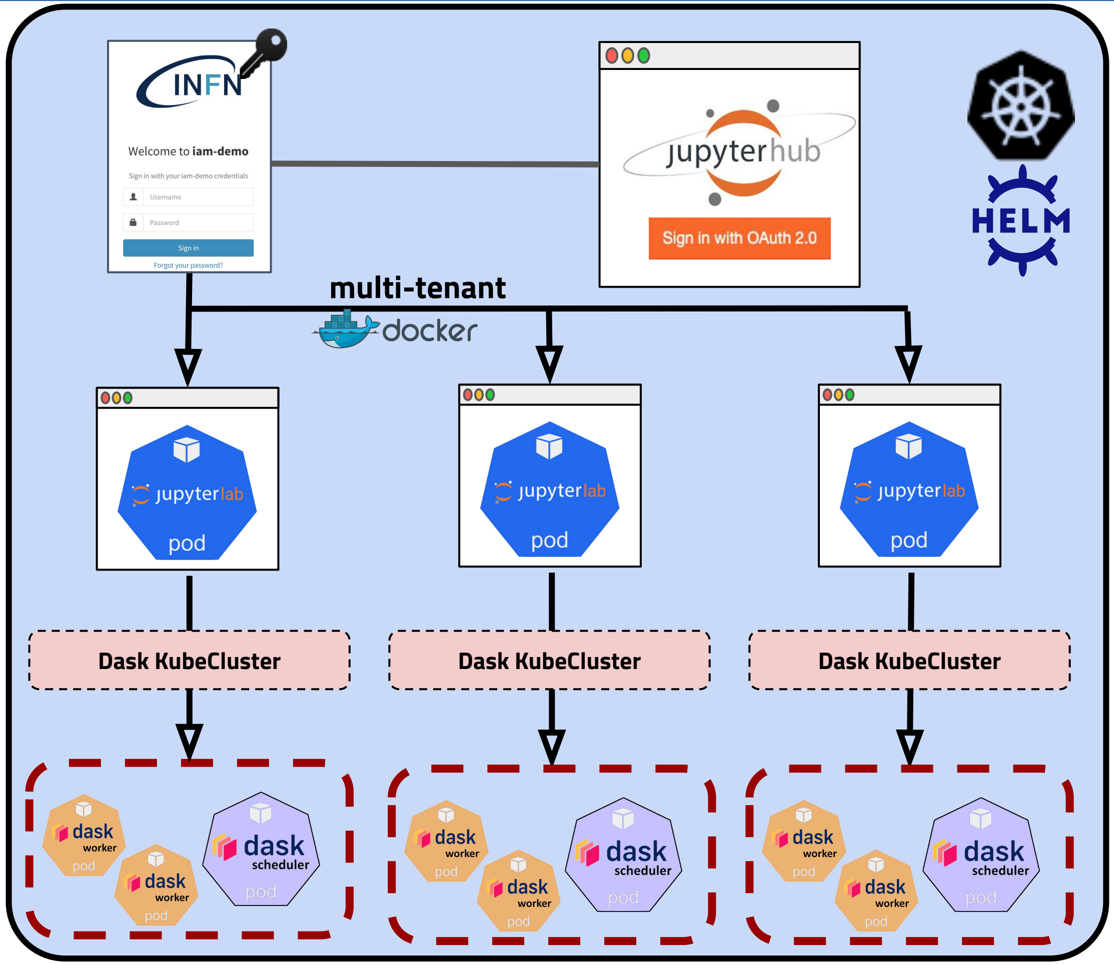
New deployment

As Spoke2 WP5 we are ready, using ICSC resources as per RAC allocation

A jupyterhub is deployed on a k8s cluster (**128 vCPUs and 258 GB**, divided into 8 nodes configured via rke2) via the official [Spoke2 JHub Helm repo]

- endpoint https://hub.131.154.98.51.myip.cloud.infn.it/
    - we should update docs https://icsc-spoke2-repo.github.io/HighRateAnalysis-WP5/sections/intro.html with the new endpoint
- still using iam-demo (highrate group) for authentication

READY TO BE TESTED



Questions

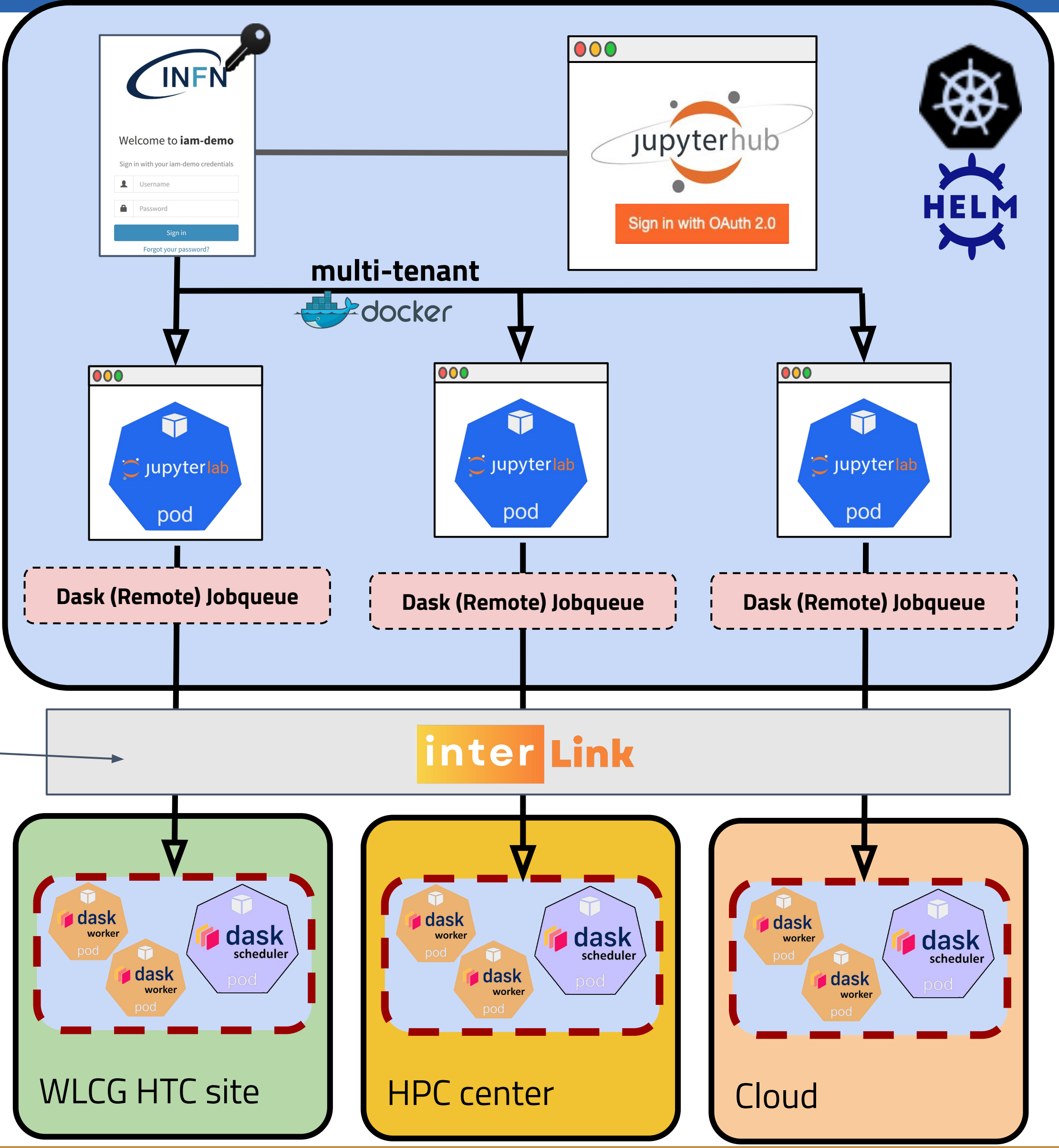
- Any more feature request from use cases side?
  - cvmfs?
  - base images based on alma8 at the moment, enough?

Backup

Extending the Resources

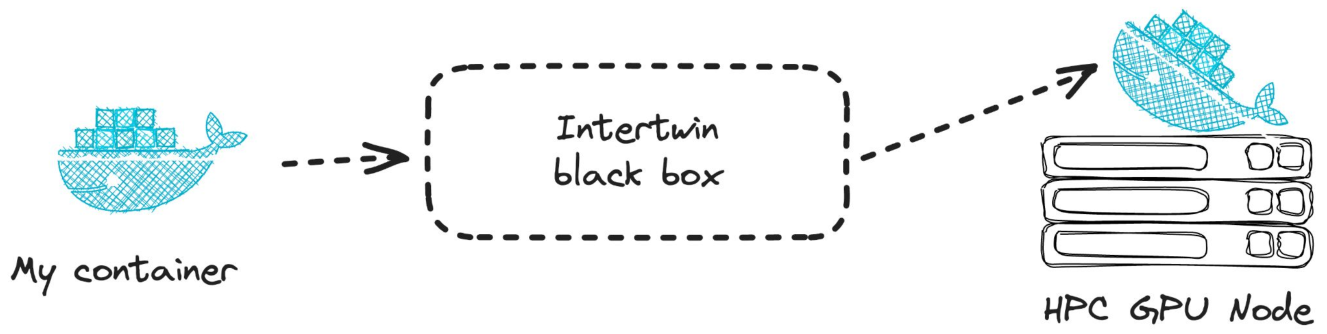
Aim: enable the platform to **dynamically exploit all kinds of resources (HTC, HPC, Cloud)** transparently for the user

- looking for a synergy with active developments in this context, to delegate container execution on remote resources while keeping the very same user interface
- Solution: InterLink, which provides execution of a Kubernetes pod on almost any remote resource
  - Resources visible to the user thanks to an HTCondor overlay



What is offloading?

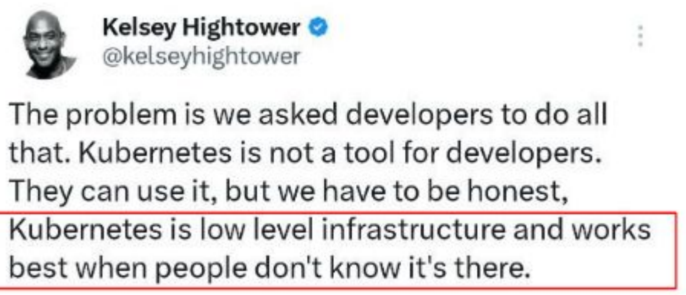
- **Delegate the execution of a container/workflow on remote resources while keeping the user interface unchanged.**
- Example:
  - "I have my own ML training container, and I want to run it on a node with 4 A100s"



How can we implement offloading?

We want a NATIVE integration with the Kubernetes primitives, acting underneath as a virtual node.

N.B. We aim to use Kubernetes as the workhorse for the "offloading", NOT as the user interface though



Kelsey Hightower
@kelseyhightower

The problem is we asked developers to do all that. Kubernetes is not a tool for developers. They can use it, but we have to be honest, Kubernetes is low level infrastructure and works best when people don't know it's there.

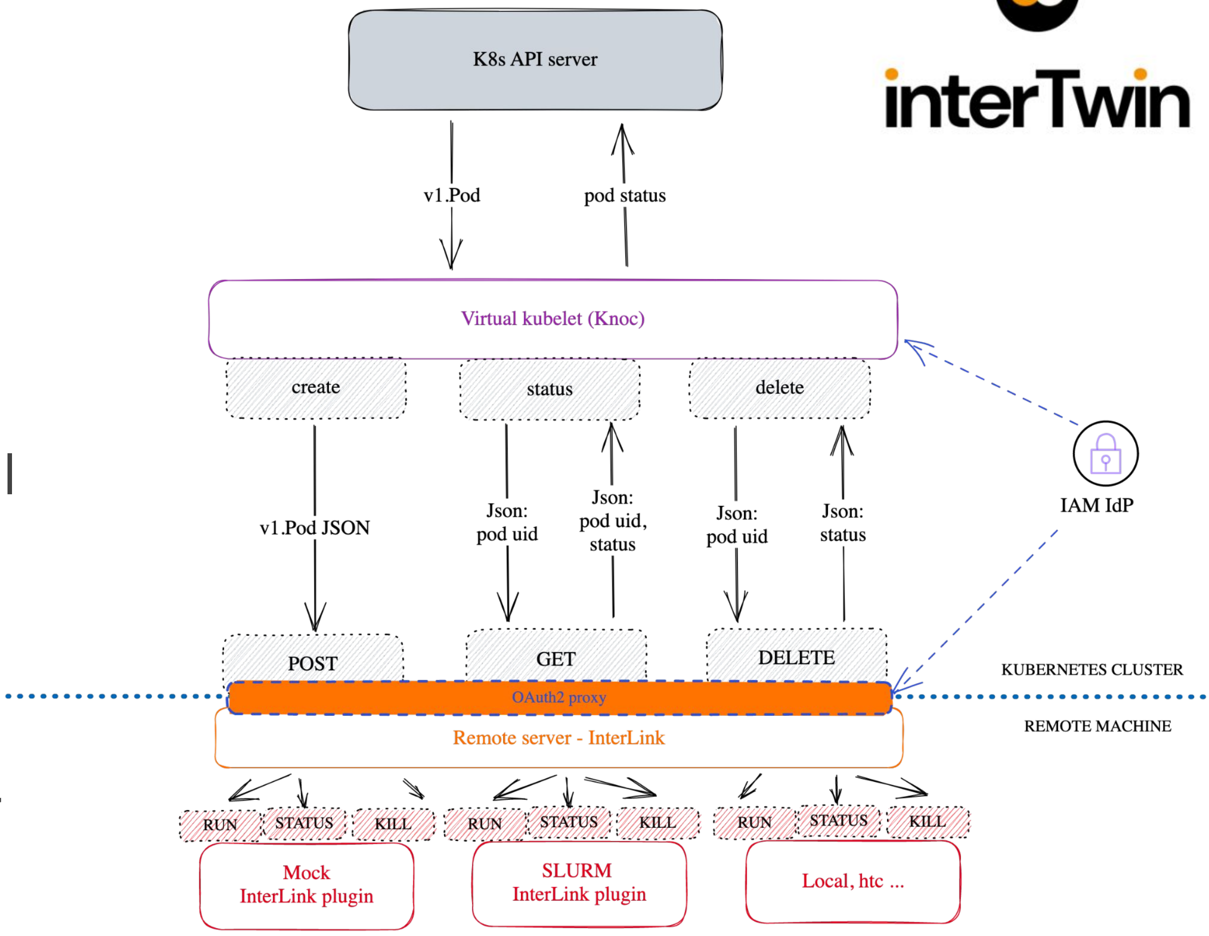
Offloading should be transparent for the users

A possible solution: InterLink

InterLink aims to provide an abstraction for the execution of a Kubernetes pod on any remote resource capable of managing a container execution lifecycle.

The project consists of two main components:

- **A Kubernetes Virtual Node**: based on the VirtualKubelet technology. Translating request for a kubernetes pod execution into a remote call to the interLink API server.
- **The interLink API server**: a modular and pluggable REST server where you can create your own container manager plugin (called sidecar), or use the existing ones: remote docker execution on a remote host, singularity Container on a remote SLURM or **HTCondor batch system**, etc...



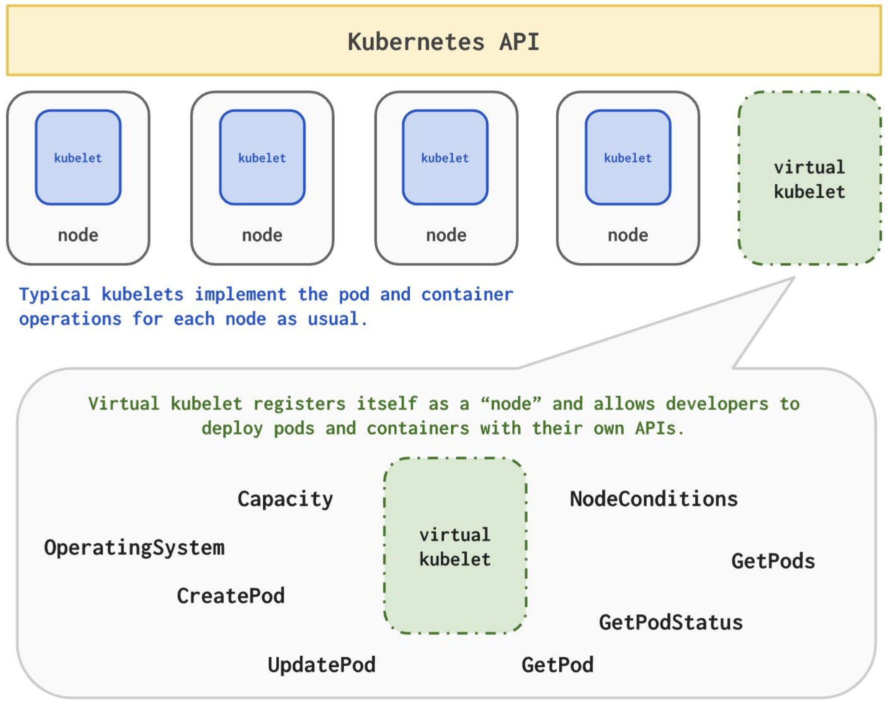
interTwin

https://github.com/interTwin-eu/interLink

Components: VK

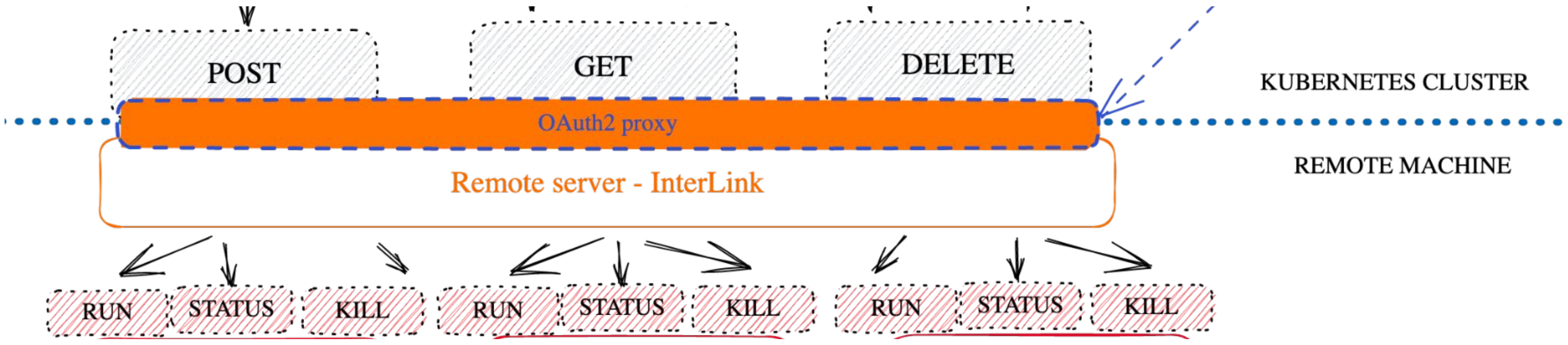
https://virtual-kubelet.io/

- **Virtual kubelet (VK)**:
  - "Open-source Kubernetes kubelet implementation that masquerades as a kubelet. This allows Kubernetes nodes to be backed by Virtual Kubelet providers"
- Can be imagined as a translation layer:
  - "I take your pod and run your container wherever I want"
- Registers virtual node and pulls work to run
- The pod lifecycle is managed via interlink rest calls
- Oauth2 via service token kept "refreshed"


Kubernetes API
kubelet
node
kubelet
node
kubelet
node
kubelet
node
virtual kubelet
Typical kubelets implement the pod and container operations for each node as usual.
Virtual kubelet registers itself as a "node" and allows developers to deploy pods and containers with their own APIs.
Capacity
OperatingSystem
CreatePod
UpdatePod
virtual kubelet
NodeConditions
GetPods
GetPodStatus
GetPod

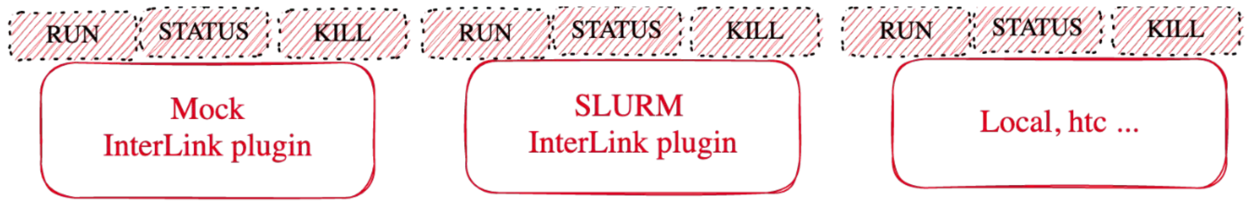
Components: Interlink + Oauth2 proxy

- Oauth2 proxy: authN with IAM and authZ configurable on aud and groups
- "Digests" and manipulates calls from VK to the sidecar
- Self contained binary, distributable on all OS without dependencies



Components: Sidecar/Plugin

- Agent that must expose a REST with defined specs, but which can be implemented in the language and with the methods you prefer:
  - creation of the pod: run local docker or submit a job on htc, slurm etc
  - collect the execution states
  - collect and forward logs upon request
  - kill
- Existing plugins: local Docker (Go), Slurm (Go), HTCondor (python), ARC (python), Kubernetes (python), Kueue (python)


RUN
STATUS
KILL
Mock InterLink plugin
RUN
STATUS
KILL
SLURM InterLink plugin
RUN
STATUS
KILL
Local, htc ...