







Integration of ICSC provisioned resources hosted at INFN Naples into the high rate platform via InterLink

T. Tedeschi¹, D. Ciangottini¹, A. Doria¹, S. Pardi¹, B. Spisso¹, S.M. Stellacci¹ on behalf of WP5

¹ Istituto Nazionale di Fisica Nucleare



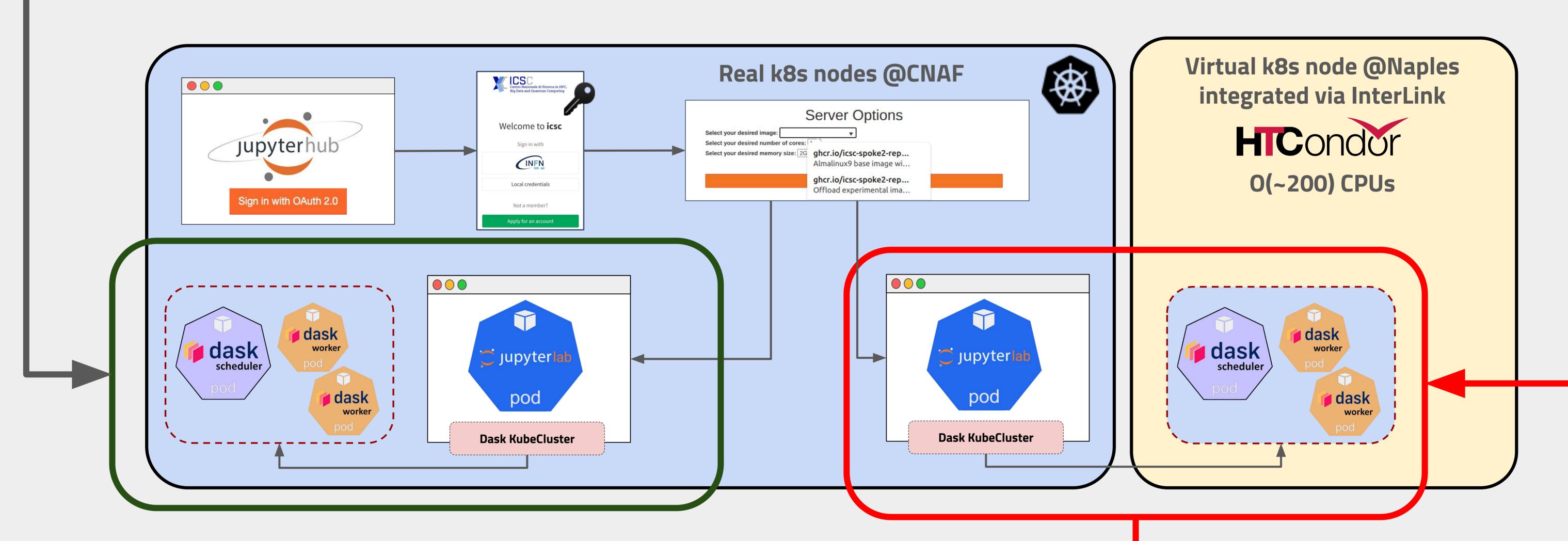
The **High Rate Analysis platform** that has been implemented in the context of the Working Group 5 of ICSC Spoke2 offers a general purpose environment where analyzers can scale up their computations via the python Dask library.

A jupyterhub is deployed on a k8s cluster (128 vCPUs and 258 GB) hosted at CNAF via the official Spoke2 JHub Helm repoertupe endpoint is here and ICSC Indigo-IAM is used for authentication

Users choose JupyterLab image to deploy and after deployment completion, they get access to a full Integrated Development Environment (persistent storage, terminal, notebooks, editors)

Initially, users' only option to scale up their python-based computation, using Dask library, was scaling within the Kubernetes cluster

Thanks the presented integration work, users can now scale up their python-based computation, using Dask library, offloading to external ICSC-provided resources hosted in Naples



- 3. InterLink communicates with the Virtual Kubelet and translates the pod request into API quests to the HTCondor InterLink plugin
- 2. The scheduler pod is scheduled on the Naples Kubernetes Virtual Node
- 1. The user deploys a Dask KubeCluster via the Labextension or their application
- 7. The Dask worker group is modified accordingly and the cluster can then be scaled via the same submission mechanism (worker jobs submitted to HTCondor pool now know the scheduler private IP address and can connect to
- HICondor **HTC**ondor Virtual InterLink Kubelet Plugin 000 💆 jupyterl<mark>a</mark>t dask worker pod **Dask KubeCluster** k8s SSH forwarder service Port SSH-tunnel 31022
- 4. The HTCondor InterLink plugin, using service credentials, submits the Dask scheduler singularity job to Naples HTCondor pool (exploiting images hosted at unpacked.infn.it via cvmfs and shipping with it JHUB username and access token)
- 5. Once running, the Dask scheduler sets an ssh-tunnel connecting back to the k8s cluster (authenticating with the JHUB access token), exposing scheduler, dashboard, and info about its private IP

6. An ssh-forwarder process running in a pod authenticates and forwards these connections into the k8s cluster network.