INTERACTIVE HPC WITH JUPYTERLAB

Training Course – SLURM provisioner

2024-04-22..23 I JENS HENRIK GÖBBERT (J.GOEBBERT@FZ-JUELICH.DE)
HERWIG ZILKEN (H.ZILKEN@FZ-JUELICH.DE)

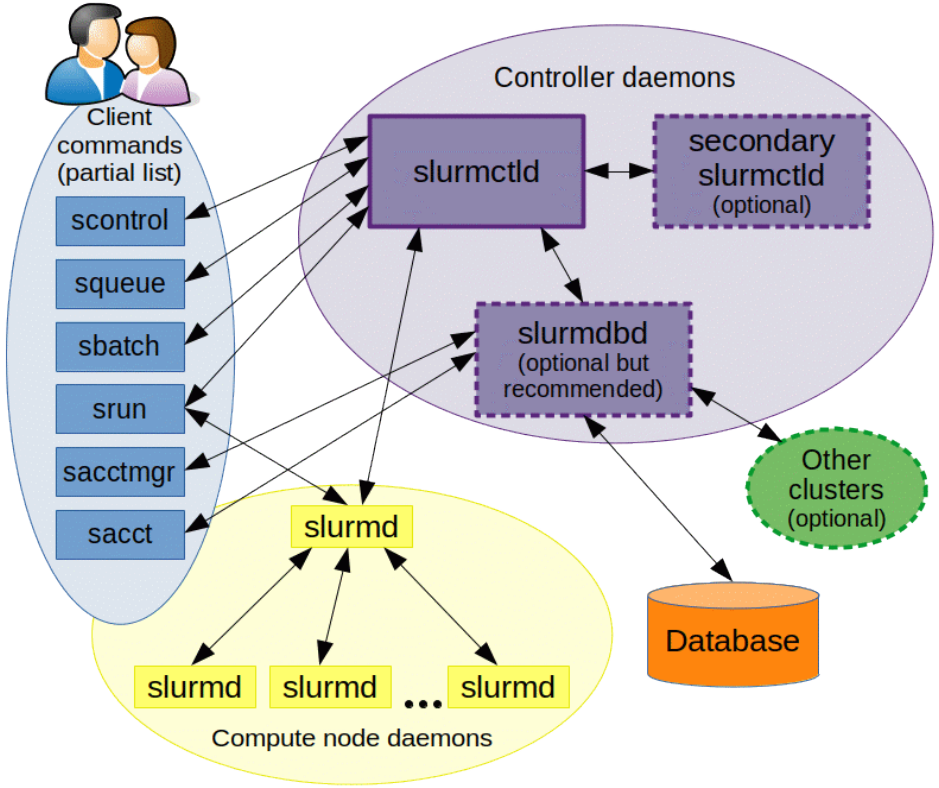
SHORT DIGRESSION:

Simple Linux Utility for Resource Management (SLURM)

Slurm is an

- open source,
- fault-tolerant, and
- highly scalable cluster management and
- job scheduling system

for large and small Linux clusters.

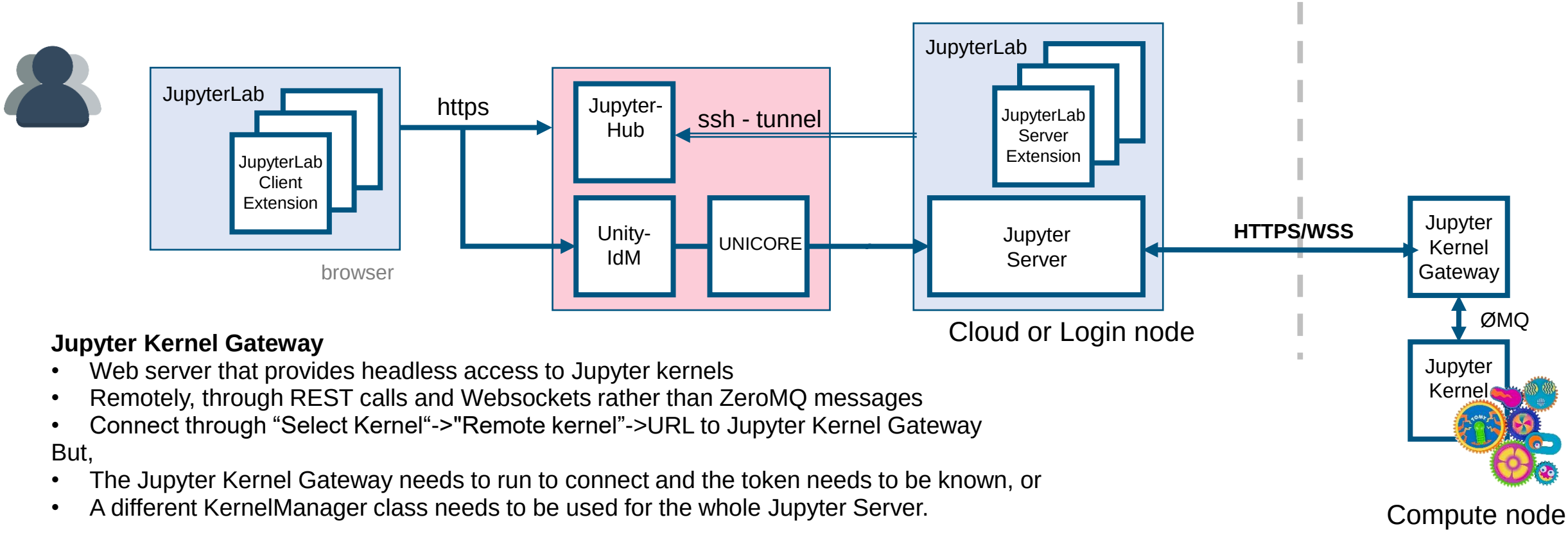


Source: https://slurm.schedmd.com/overview.html

SLURM WRAPPED KERNELS WITH SLURM-PROVISIONER

REMOTE JUPYTER KERNELS

Running multiple Jupyter kernels separate on the HPC system



Jupyter Kernel Gateway

- Web server that provides headless access to Jupyter kernels
- Remotely, through REST calls and Websockets rather than ZeroMQ messages
- Connect through “Select Kernel“->"Remote kernel”->URL to Jupyter Kernel Gateway

But,

- The Jupyter Kernel Gateway needs to run to connect and the token needs to be known, or
- A different KernelManager class needs to be used for the whole Jupyter Server.

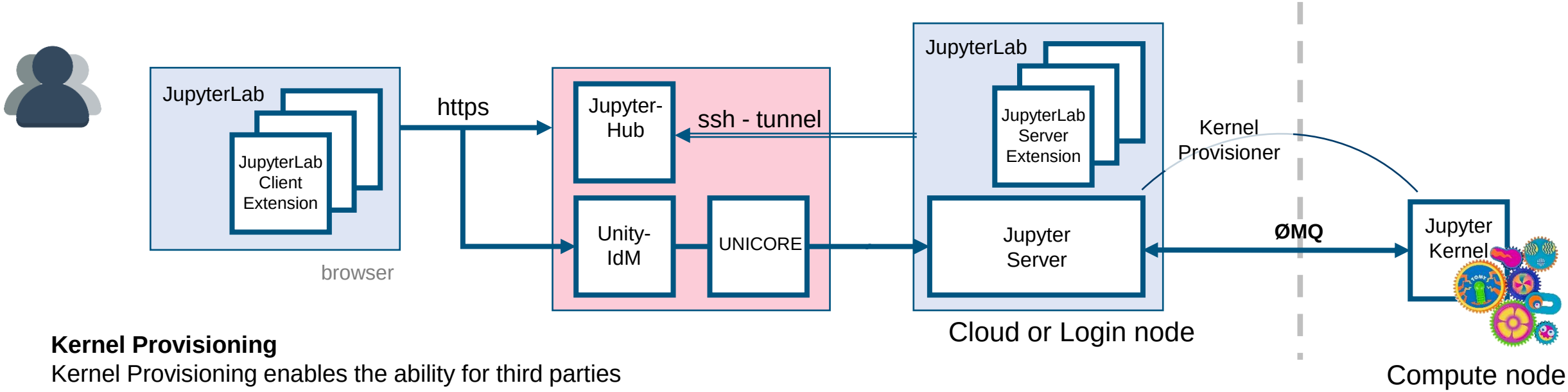
Both ways are non-intuitive and limit the user – especially as integration with the scheduler SLURM is missing.

Jupyter Enterprise Gateway is significantly richer in functionality,
but a service users can connect to and primarily made for a cloud.

https://jupyter-kernel-gateway.readthedocs.io
https://jupyter-enterprise-gateway.readthedocs.io

REMOTE JUPYTER KERNELS

Running multiple Jupyter kernels separate on the HPC system



Kernel Provisioning
Kernel Provisioning enables the ability for third parties
to **manage the lifecycle of a kernel's runtime environment**.

By implementing and configuring a *kernel provisioner*,
third parties have the ability to **provision kernels for different environments**,
typically managed by resource managers like Kubernetes, Hadoop YARN, Slurm, etc.

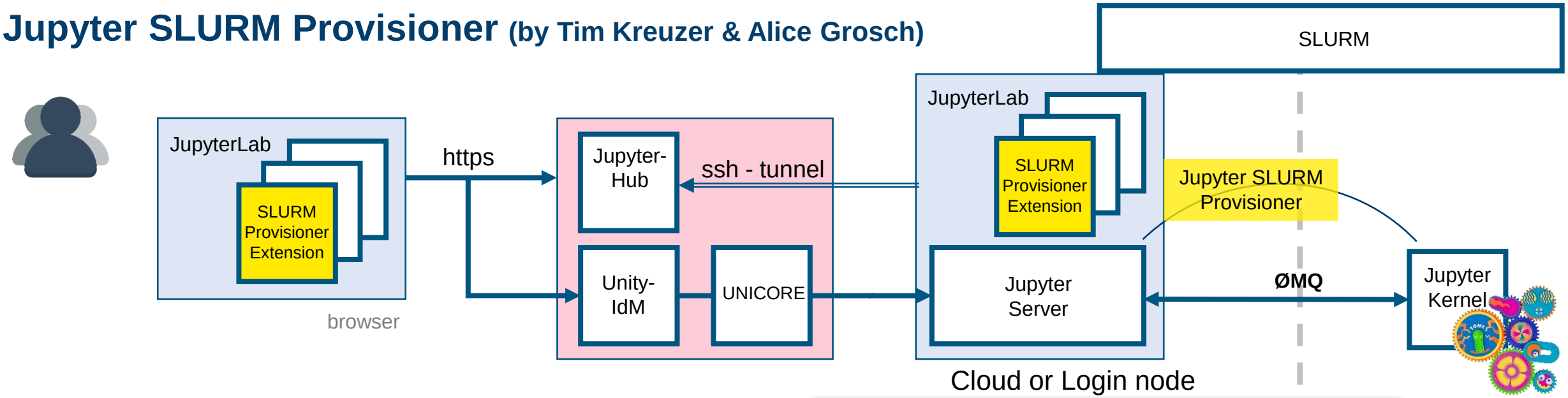
The kernel provisioner optionally extends the current **metadata stanza within the kernel.json**
to include the specification of the kernel provisioner name, along with an optional config stanza

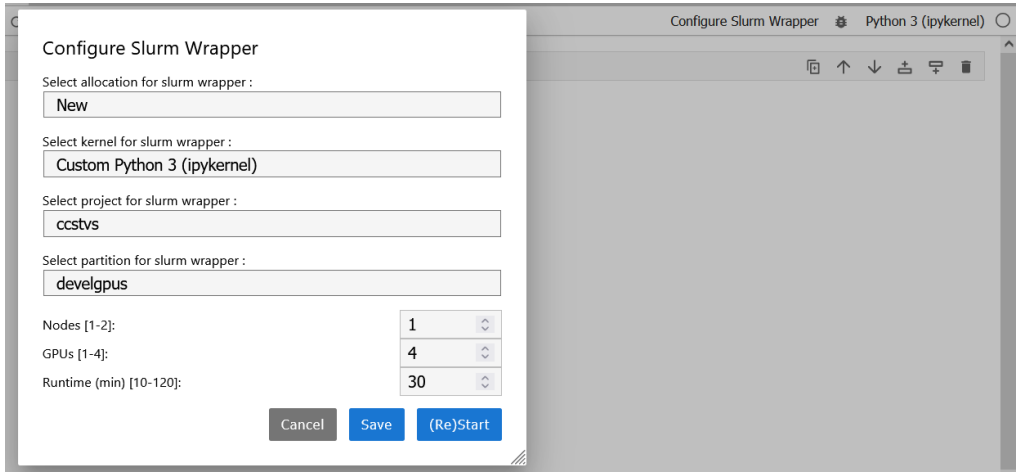
```
[..]
"metadata": {
    "kernel_provisioner": {
        "provisioner_name": "slurm-provisioner",
        "config": {
            "kernel_argv": "Python",
            "project": "zam",
            "partition": "batch",
            "nodes": 1,
            "runtime": 3600,
        }
    }
},
```

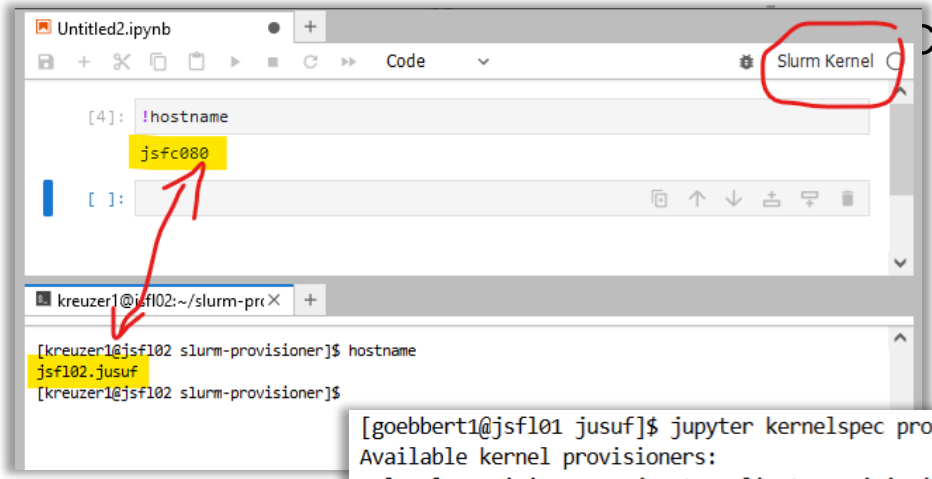
https://jupyter-client.readthedocs.io/en/stable/provisioning.html

REMOTE JUPYTER KERNELS

Jupyter SLURM Provisioner (by Tim Kreuzer & Alice Grosch)





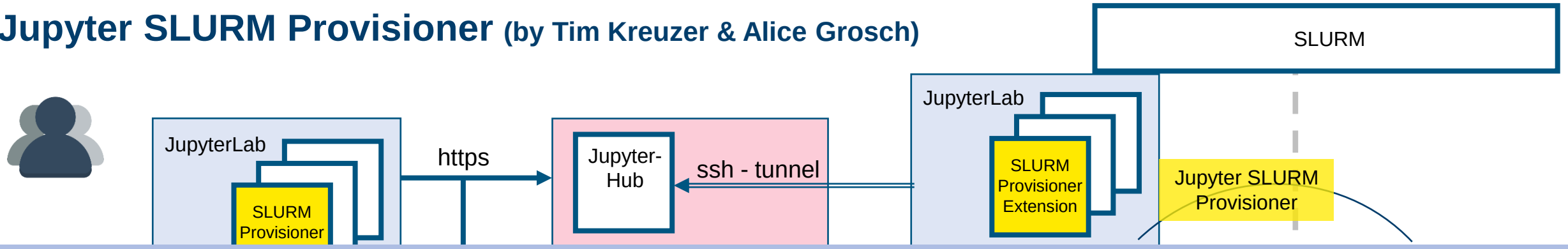



```
[goebbert1@jsfl01 jusuf]$ jupyter kernelspec provisioners
Available kernel provisioners:
  local-provisioner    jupyter_client.provisioning:LocalProvisioner
  slurm-provisioner    jupyter_slurm_provisioner:SlurmProvisioner
```

https://github.com/FZJ-JSC/jupyter-slurm-provisioner
https://github.com/FZJ-JSC/jupyter-slurm-provisioner-extension

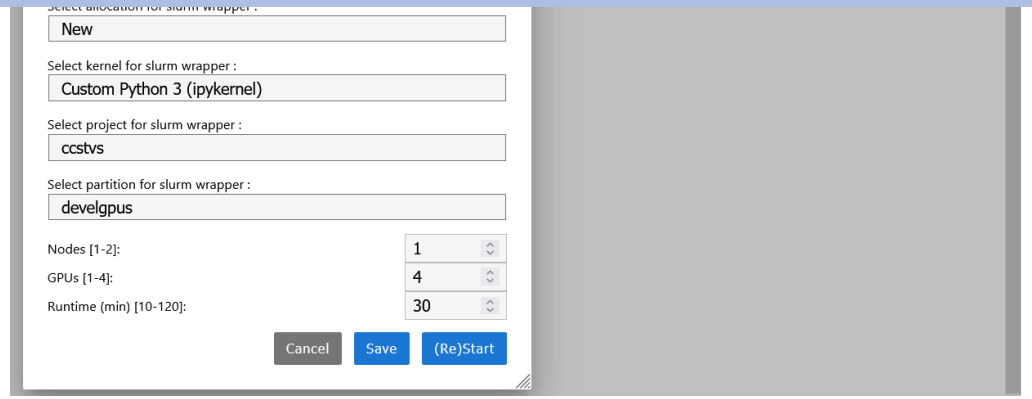
REMOTE JUPYTER KERNELS

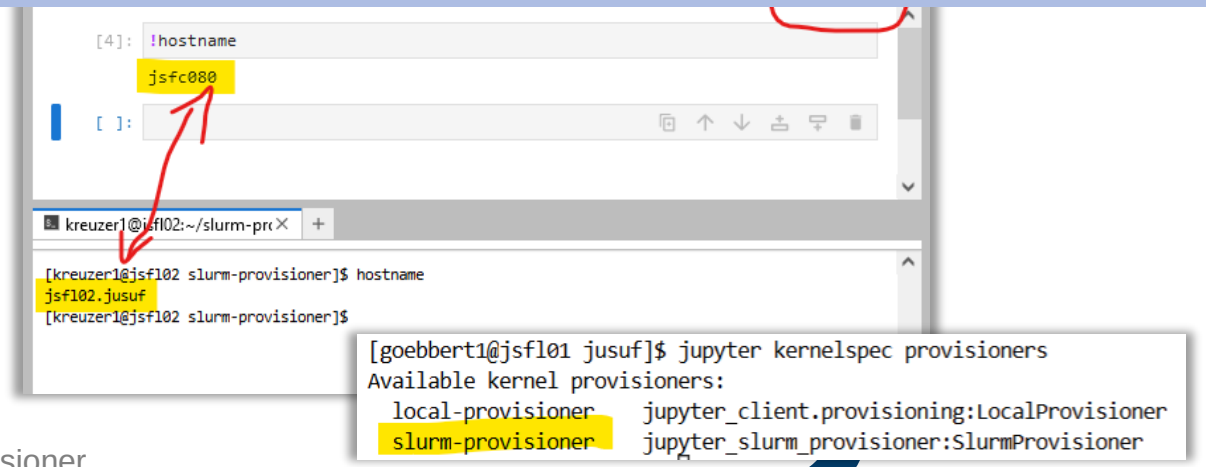
Jupyter SLURM Provisioner (by Tim Kreuzer & Alice Grosch)



Slurm wrapped kernels allow you to run kernels on compute nodes while your Jupyter Server runs on a login node.

This has the advantage that when your allocation on the compute node(s) ends, **only the kernel is stopped**, but your JupyterLab server keeps running. You will only have to restart the kernel, not your entire JupyterLab instance.

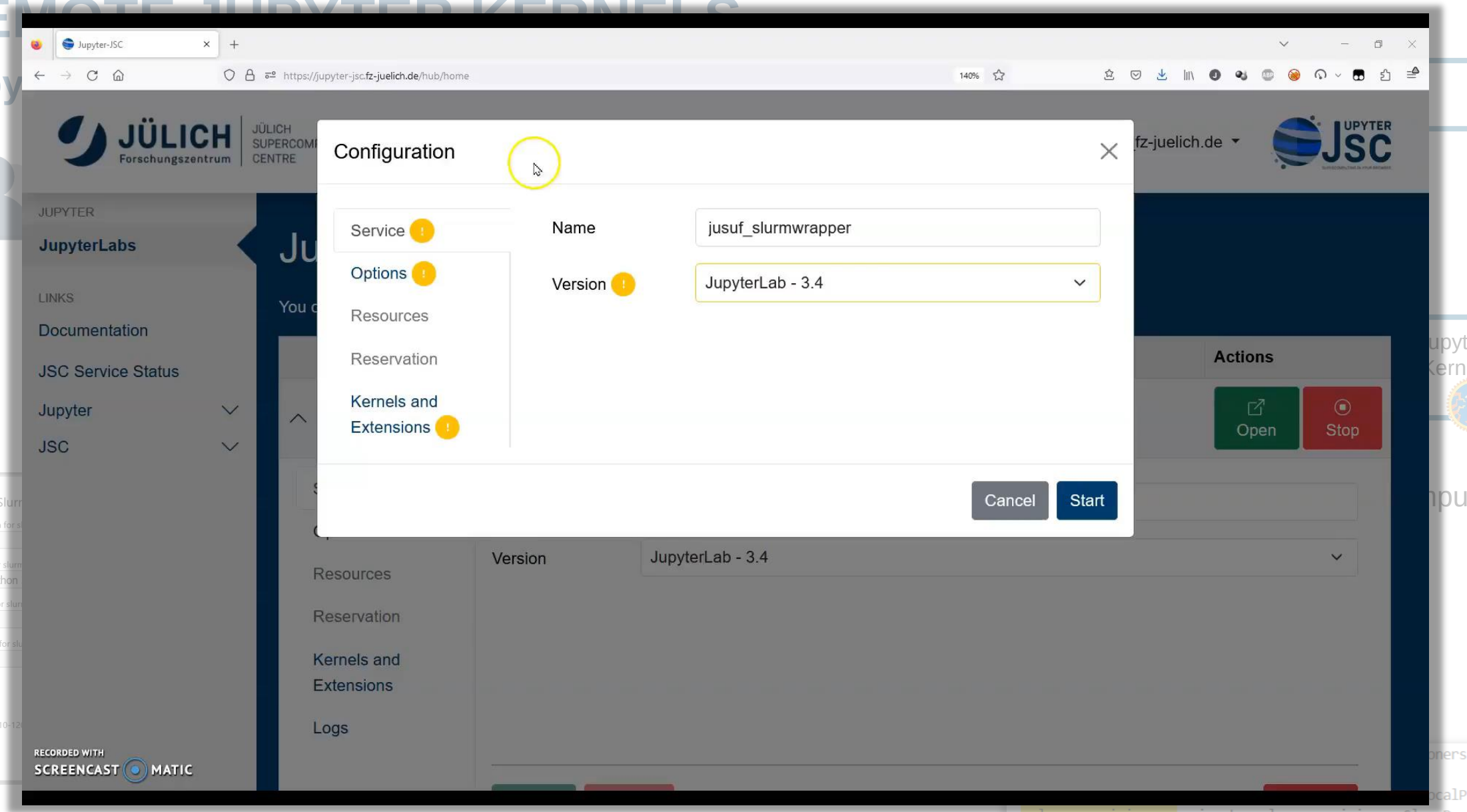




https://github.com/FZJ-JSC/jupyter-slurm-provisioner
https://github.com/FZJ-JSC/jupyter-slurm-provisioner-extension

REMOTE JUPYTER KERNELS

Jupy



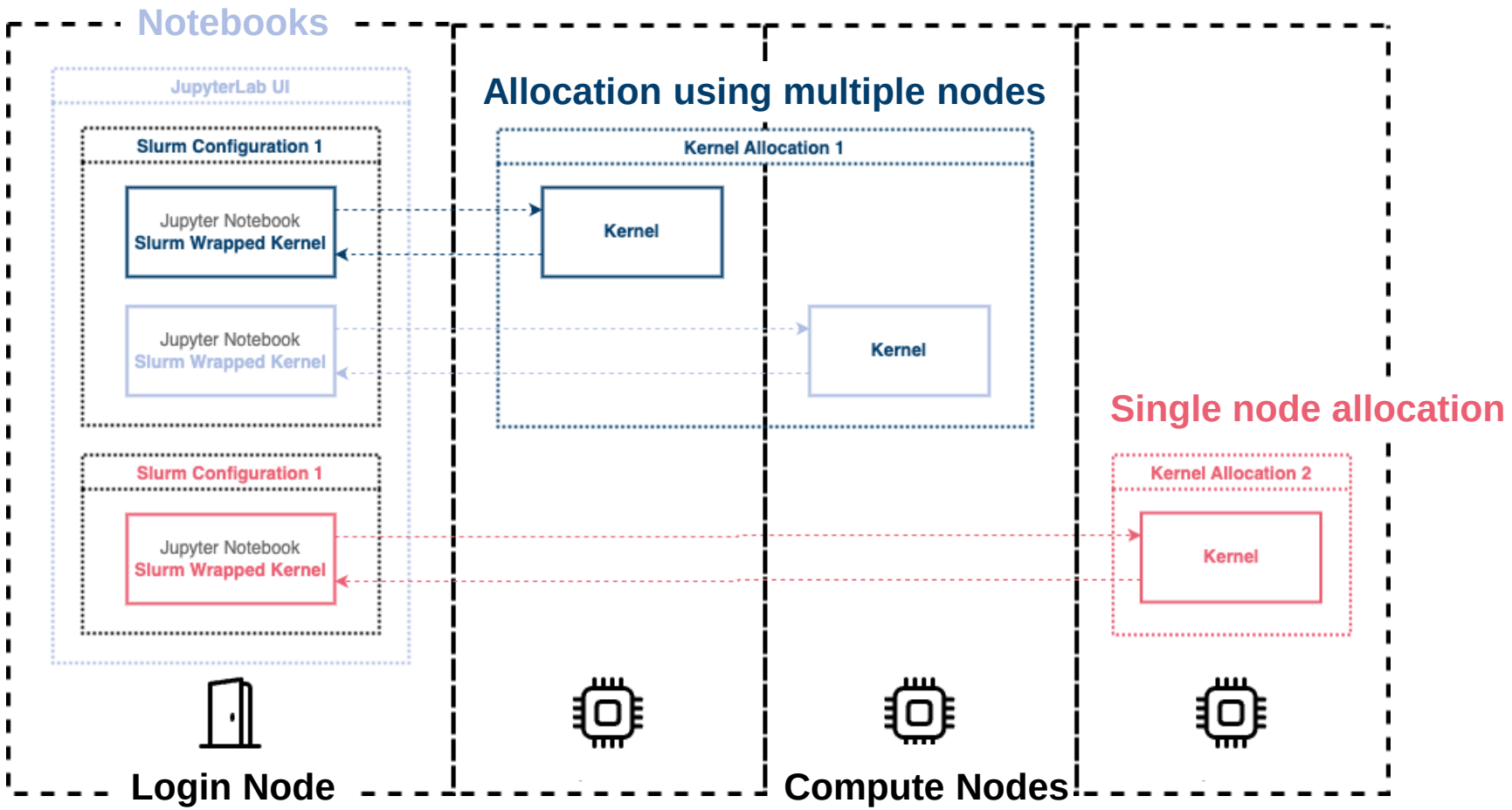
upyter
Kernel

mpute node

https://github.com/FZJ-JSC/jupyter-slurm-provisioner
https://github.com/FZJ-JSC/jupyter-slurm-provisioner-extension

JUPYTER SLURM PROVISIONER

Different kernel allocations



https://github.com/FZJ-JSC/jupyter-slurm-provisioner
https://github.com/FZJ-JSC/jupyter-slurm-provisioner-extension

QUESTIONS?