



INTERACTIVE HPC WITH JUPYTERLAB

Training Course – SLURM provisioner

2024-04-22..23 | JENS HENRIK GÖBBERT
HERWIG ZILKEN

(J.GOEBBERT@FZ-JUELICH.DE)
(H.ZILKEN@FZ-JUELICH.DE)

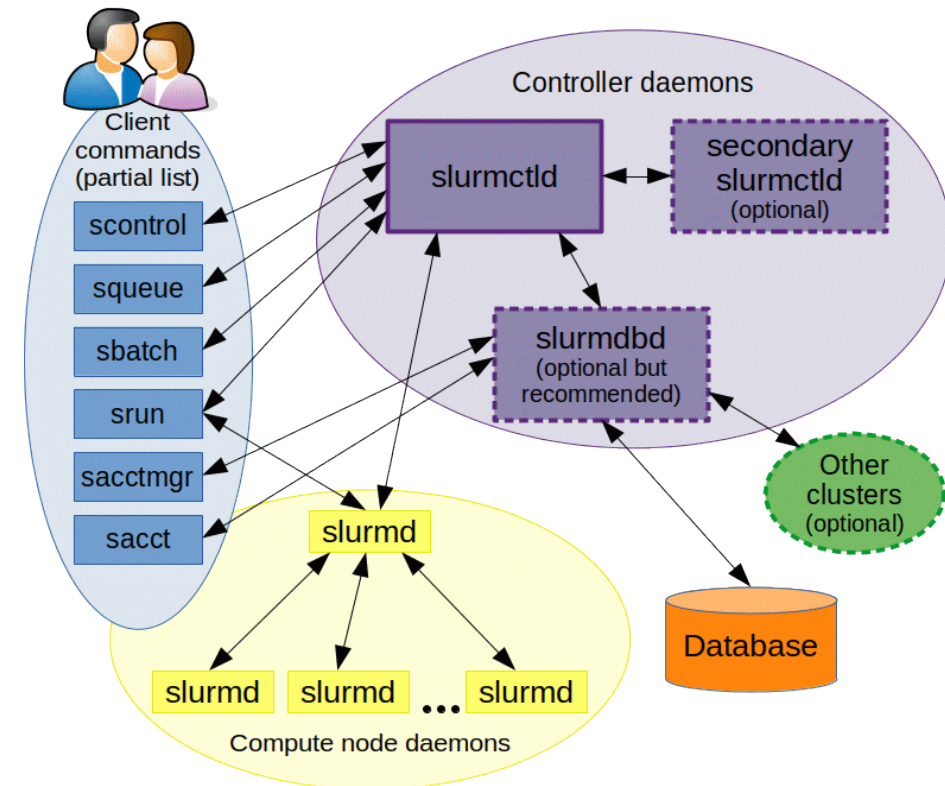
SHORT DIGRESSION:

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.

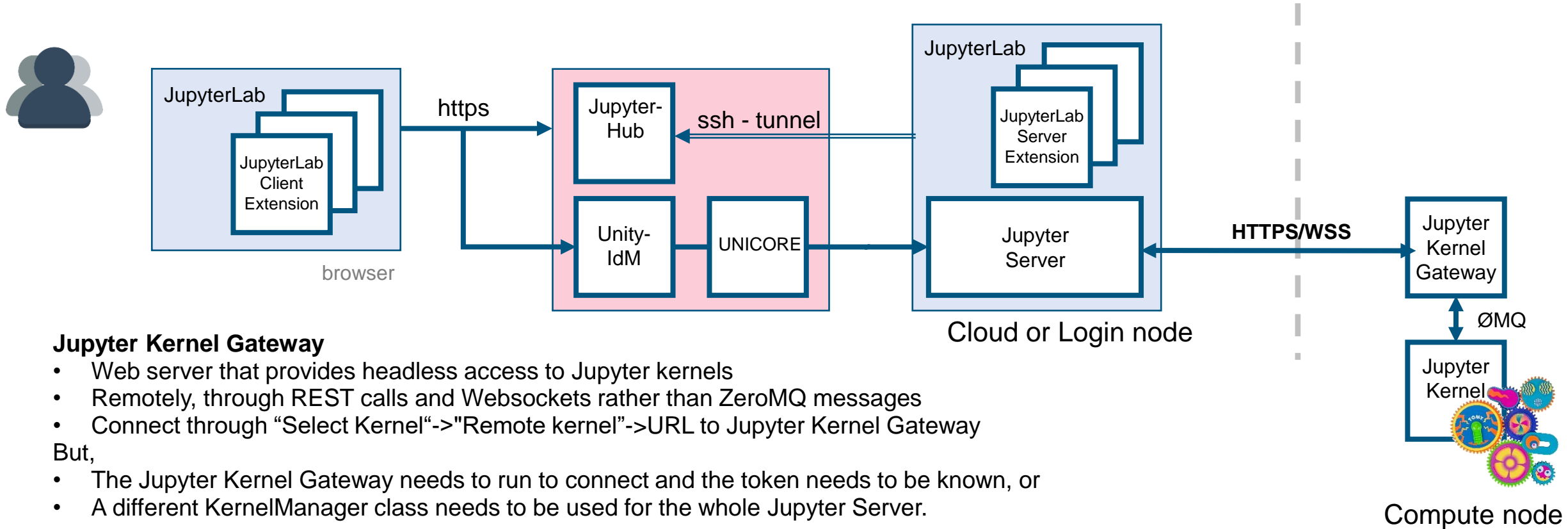


Source: <https://slurm.schedmd.com/overview.html>

SLURM WRAPPED KERNELS WITH SLURM-PROVISIONER

REMOTE JUPYTER KERNELS

Running multiple Jupyter kernels separate on the HPC system

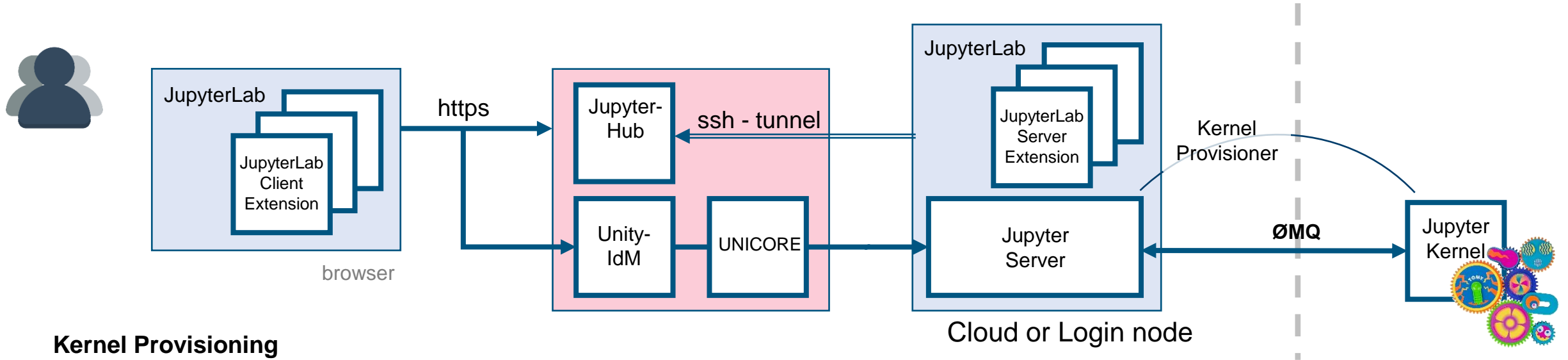


Both ways are non-intuitive and limit the user – especially as integration with the scheduler SLURM is missing.

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

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**.

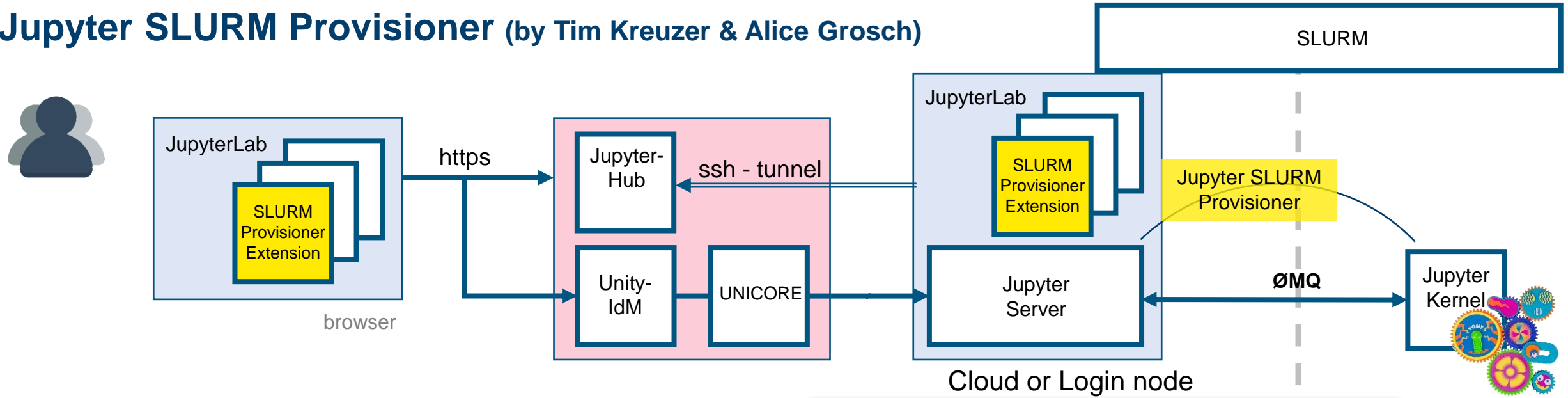
By implementing and configuring a *kernel provisioner*, third parties have the ability to **provision kernels for different environments**, typically managed by resource managers like Kubernetes, Hadoop YARN, Slurm, etc.

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,
    }
  }
},
```

REMOTE JUPYTER KERNELS

Jupyter SLURM Provisioner (by Tim Kreuzer & Alice Grosch)



The screenshot shows the 'Configure Slurm Wrapper' interface. It includes fields for selecting allocation, kernel, project, and partition. The 'Nodes' field is set to 1, 'GPUs' to 4, and 'Runtime (min)' to 30. The 'Save' button is highlighted.



```

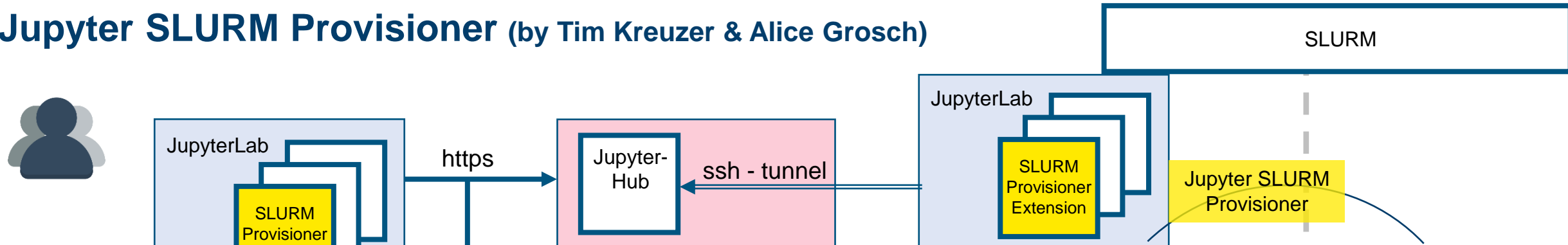
[4]: !hostname
jsfc080

[kreuzer1@jsf102 slurm-provisioner]$ hostname
jsf102.jusuf
[kreuzer1@jsf102 slurm-provisioner]$

[goebbert1@jsf101 jusuf]$ jupyter kernelspec provisioners
Available kernel provisioners:
  local-provisioner      jupyter_client.provisioning:LocalProvisioner
  slurm-provisioner      jupyter_slurm_provisioner:SlurmProvisioner
```


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.

The screenshot shows the 'Select allocation for slurm wrapper' dialog in JupyterLab. It includes fields for 'New', 'Select kernel for slurm wrapper' (set to 'Custom Python 3 (ipykernel)'), 'Select project for slurm wrapper' (set to 'ccstvs'), and 'Select partition for slurm wrapper' (set to 'develgpu'). There are also input fields for 'Nodes [1-2]' (set to 1), 'GPUs [1-4]' (set to 4), and 'Runtime (min) [10-120]' (set to 30). Buttons for 'Cancel', 'Save', and '(Re)Start' are at the bottom.



The screenshot shows a JupyterLab terminal window. The top part shows a JupyterLab interface with a code cell containing `hostname` and a kernel named `jsf102`. Below the code cell, a terminal window shows the command `hostname` being executed, resulting in `jsf102.jusuf`. A red arrow points from the kernel name in the JupyterLab interface to the terminal output. Below the terminal window, a command prompt shows the command `jupyter kernelspec provisioners` being executed, resulting in the output: `Available kernel provisioners: local-provisioner jupyter_client.provisioning:LocalProvisioner slurm-provisioner jupyter_slurm_provisioner:SlurmProvisioner`.

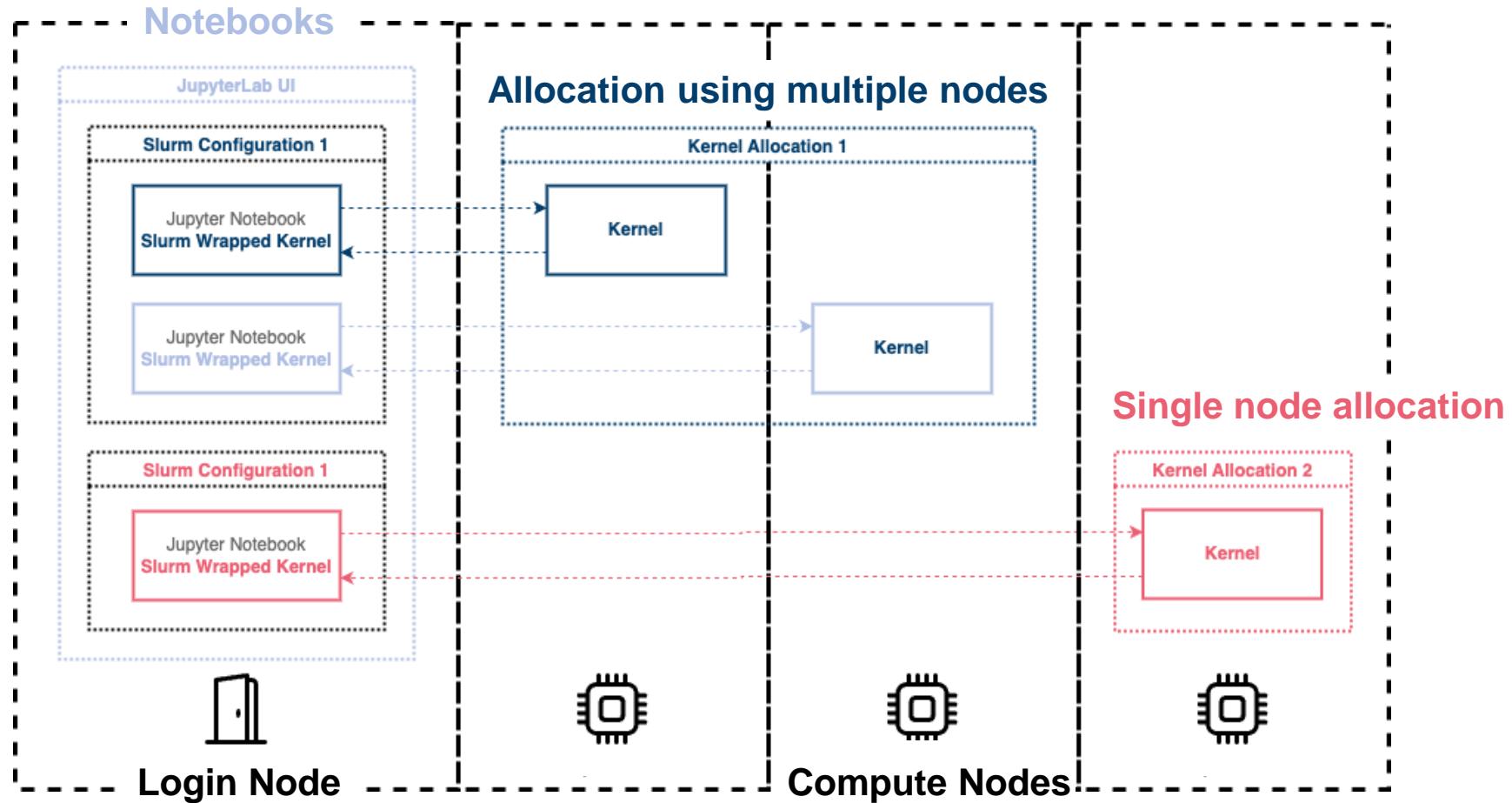
REMOTE JUPYTER KERNELS

Jupy

The screenshot shows the Jupyter-JSC web interface in a browser window. The URL is `https://jupyter-jsc.fz-juelich.de/hub/home`. The page features the Jülich Forschungszentrum logo and a sidebar with navigation links: JUPYTER, JupyterLabs, LINKS, Documentation, JSC Service Status, Jupyter, and JSC. A 'Configuration' modal is open, displaying settings for a kernel named 'jusuf_slurmwrapper'. The modal has a sidebar with options: Service, Options, Resources, Reservation, and Kernels and Extensions. The 'Name' field is set to 'jusuf_slurmwrapper' and the 'Version' dropdown is set to 'JupyterLab - 3.4'. At the bottom of the modal are 'Cancel' and 'Start' buttons. In the background, the 'Actions' section shows 'Open' and 'Stop' buttons. A 'Compute node' label is visible on the right. A 'slurm-provisioner' extension is listed at the bottom right. A 'RECORDED WITH SCREENCAST MATIC' watermark is present in the bottom left corner of the interface.

JUPYTER SLURM PROVISIONER

Different kernel allocations



QUESTIONS?

