



National Research Platform (NRP) and Kubernetes for Education and Research

May 5, 2026

7NRP Workshop, La Jolla, CA

Mahidhar Tatineni, Mohammad Firas Sada, and Daniel Diaz

San Diego Supercomputer Center

Halicioğlu School of Data Science and Computing

University of California, San Diego

NRP NATIONAL RESEARCH
PLATFORM

SAN DIEGO
SUPERCOMPUTER
CENTER

UC San Diego

Outline

- Introduction to National Research Platform (NRP)
 - Motivation
 - Hardware resources
 - Storage Options
- Classroom usage
- Hands-On Exercises
 - Running using JupyterHub
 - Running using kubectl
 - Basics using simple pod examples
 - Persistent storage
 - Deployments
 - Batch jobs
 - Services, including public access setup over https
 - Resources including special GPUs

GitHub Repository for Today's Tutorials

<https://github.com/nrp-nautilus/7nrp>

Launch Tutorial Workspace

- We are going use a Jupyter workspace for all of today's hands-on content. Get started by clicking on the link in the github page
- You can authenticate using your institution credentials via CILogon

https://github.com/nrp-nautilus/7nrp

nrp-nautilus/7nrp

README

Quick start

Two ways to follow along during the workshop:

Option 1 — Training JupyterHub (recommended, zero install). The workshop hub at training.nrp-nautilus.io is pre-configured: every spawned JupyterLab pod already has `kubectl` installed and a kubeconfig wired up to the same identity, so you can open a terminal and run `kubectl` immediately. Click below to clone this repo straight into your JupyterLab session:

LAUNCH 7NRP TUTORIAL WORKSPACE ->

Option 2 — kubectl on your laptop. Install `kubectl` (Linux / macOS / Windows) and use the ready kubeconfig at [files/nrp-training.kubeconfig](#). It carries the `jupyterhub-sa` service-account token, cluster CA, and `nrp-training-k8s` as the default namespace; the embedded token is valid for the duration of 7NRP, through end-of-day Thursday, May 7, 2026. Step-by-step instructions live in [Tutorial 1 → Interacting with NRP](#).

Conventions

- Tutorials 1 and 2 share the `nrp-training-k8s` namespace. It already exists for the workshop; if you need to recreate it later: `kubectl create namespace nrp-training-k8s`.
- Tutorial 3 uses pre-created per-participant namespaces (`nrp-training-000` ... `nrp-training-099`).
- Replace `<username>` in any YAML or command with your NRP or GitHub username to avoid name collisions.
- Every workload must declare CPU and memory `requests` and `limits` — a cluster-wide Gatekeeper policy rejects pods without them.

Launch Tutorial Workspace

- **Do not request GPUs or Qualcomm AI devices in the form.** We will request and handle that via Kubernetes workloads in the sessions.

- Choose zero GPUs and Qualcomm AI Devices here



Contact admins in [Matrix](#).

Region
Any

GPUs
0

Qualcomm Cloud AI Devices
0

Cores
1

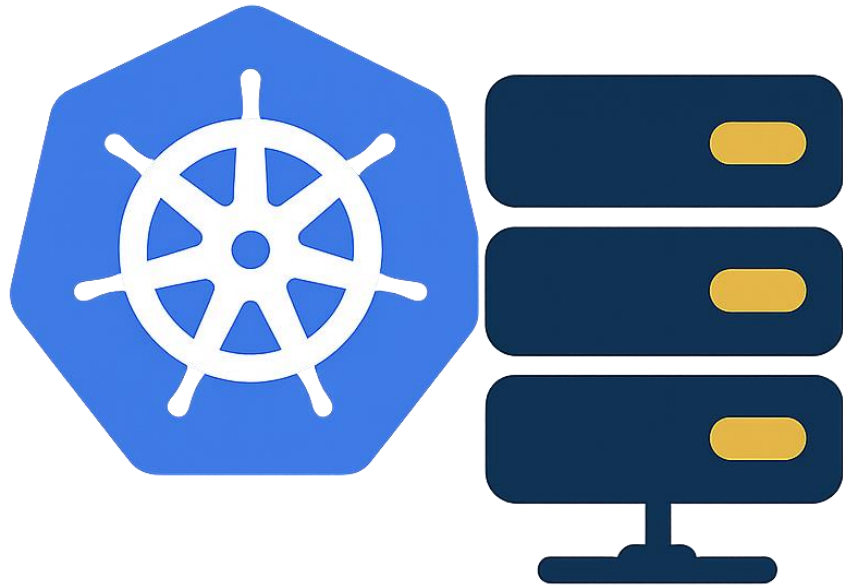
RAM, GB
8

FPGAs
0

GPU type
Any general

Refer to the [documentation](#) for images description.

Image
• NRP Deep Learning & Data Science Full, PyTorch



Introduction to the National Research Platform

NAIRR Pilot Workshop

Motivation: Current Challenges & Outlook

Current Challenges

- **Rising Demand:** Growing reliance on cyberinfrastructure (CI) for research and education.
- **Access Inequality:** Widening gap between institutions with and without CI resources
- **Architectural Fragmentation:** Post-Moore's Law era complicates domain science adoption.

Future Outlook

- **Equity Focus:** Prioritize CI access for *all* institutions.
- **Universal Connectivity:** Connect educational institutions to advanced CI.
- **Talent Empowerment:** Build environments for collaboration.
- **Global Leadership:** Invest holistically in CI to drive innovation and competitiveness.

The National Research Platform

- The National Research Platform is a partnership of more than **125** institutions.
- Led by researchers at **UC San Diego, University of Nebraska-Lincoln, and MGHPCC.**
- Includes contributions by the **National Science Foundation**, the **Department of Energy**, the **Department of Defense**, and many research universities and R&E networking organizations in the US and around the world.
- NRP operates a primary **Kubernetes** distribution cluster with hardware spanning across **3 continents**, called **Nautilus**.

Time Range **24h** 7d 30d

Updated 10:29:28 AM

SITES

129

Hosting NRP nodes



NODES

501

Registered in Kubernetes

**1.5K**

Across all nodes

**32.3K**

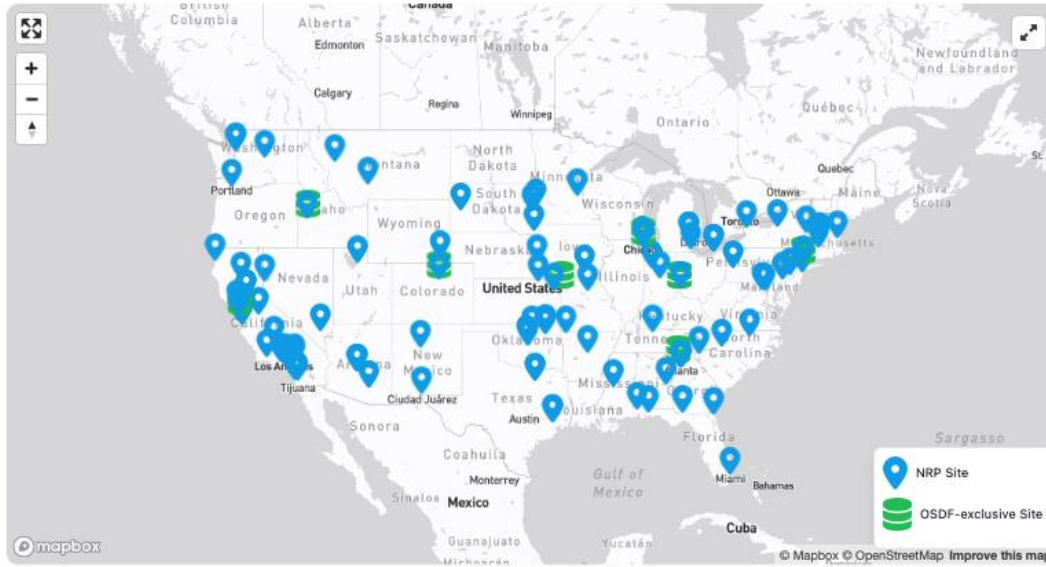
Across all nodes



OSDF NODES

31

Nodes serving OSDF cache

[View OSDF Dashboard →](#)

ABOUT NRP

NRP NATIONAL RESEARCH PLATFORM

The National Research Platform is a partnership of more than 50 institutions, led by researchers at UC San Diego, University of Nebraska-Lincoln, and Massachusetts Green High Performance Computing Center and includes contributions by the National Science Foundation, the Department of Energy, the Department of Defense, and many research universities and R&E networking organizations in the US and around the world.

SELECT SITE

Choose a site or click on the map

Select...

MAP CUSTOMIZATION

Select multiple sites to highlight

Select multiple sites...

Or filter by regex pattern

e.g., `chicago|boulder|^ucsd.*`

GPUS ALLOCATED

1,467

→ 0%

May 3, 2026 at 10 AM



May 2, 2026 at 10 AM

May 3, 2026 at 10 AM

RUNNING JUPYTER PODS

111

↗ 11%

May 3, 2026 at 10 AM



May 2, 2026 at 10 AM

May 3, 2026 at 10 AM

ACTIVE RESEARCH GROUPS USING GPUS

95

↘ 2%

May 3, 2026 at 10 AM



May 2, 2026 at 10 AM

May 3, 2026 at 10 AM



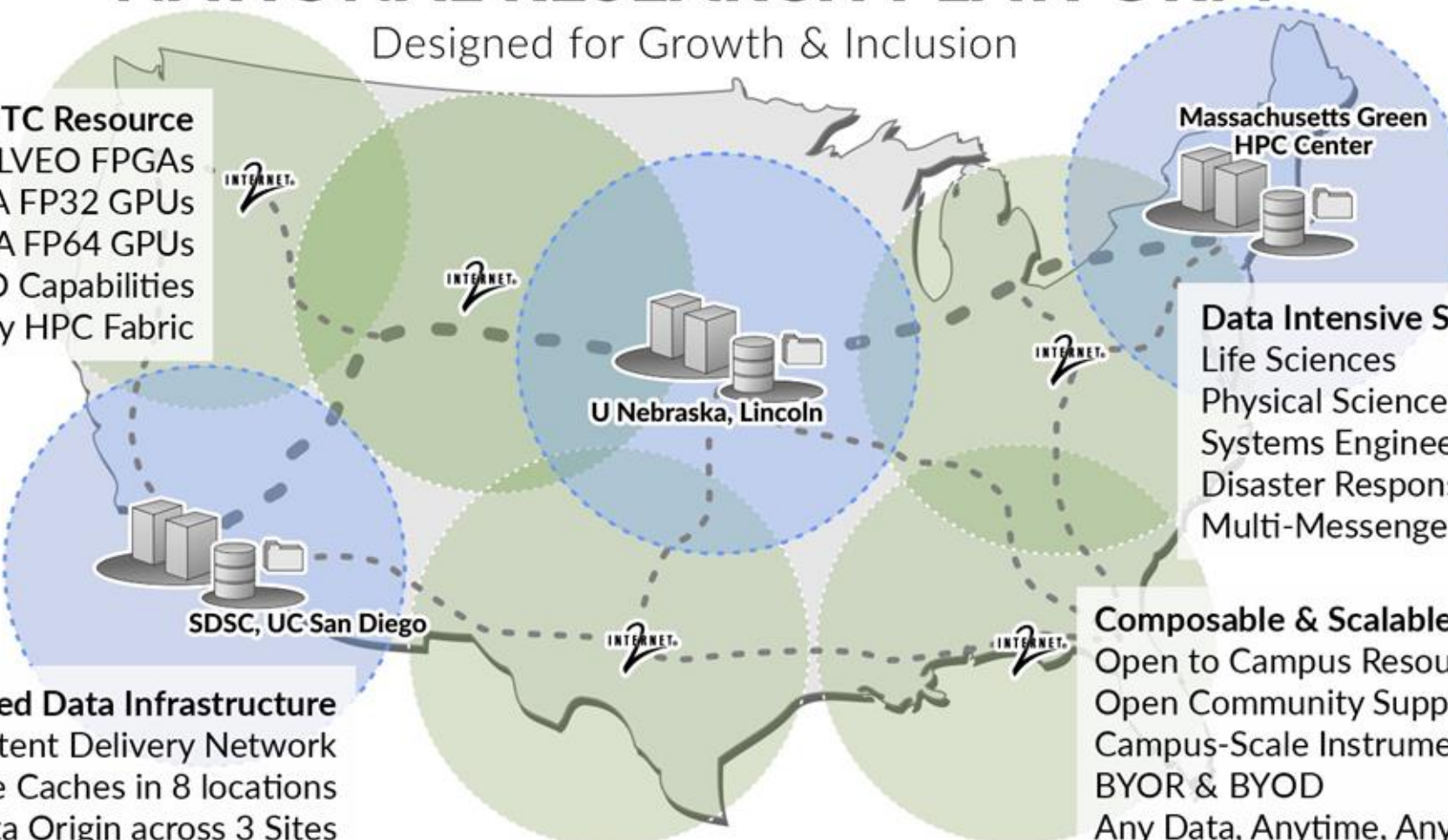
This work was supported in part by National Science Foundation (NSF) awards CNS-1730158, ACI-1540112, ACI-1541349, OAC-1826967, OAC-2112167, CNS-2100237, CNS-2120019.

<https://dash.nrp-nautilus.io/>

NATIONAL RESEARCH PLATFORM

Designed for Growth & Inclusion

HPC/HTC Resource
32 ALVEO FPGAs
A10 288 NVIDIA FP32 GPUs
80GB A100 64 NVIDIA FP64 GPUs
Tbps WAN IO Capabilities
GigalO's Low Latency HPC Fabric



Massachusetts Green HPC Center

U Nebraska, Lincoln

SDSC, UC-San Diego

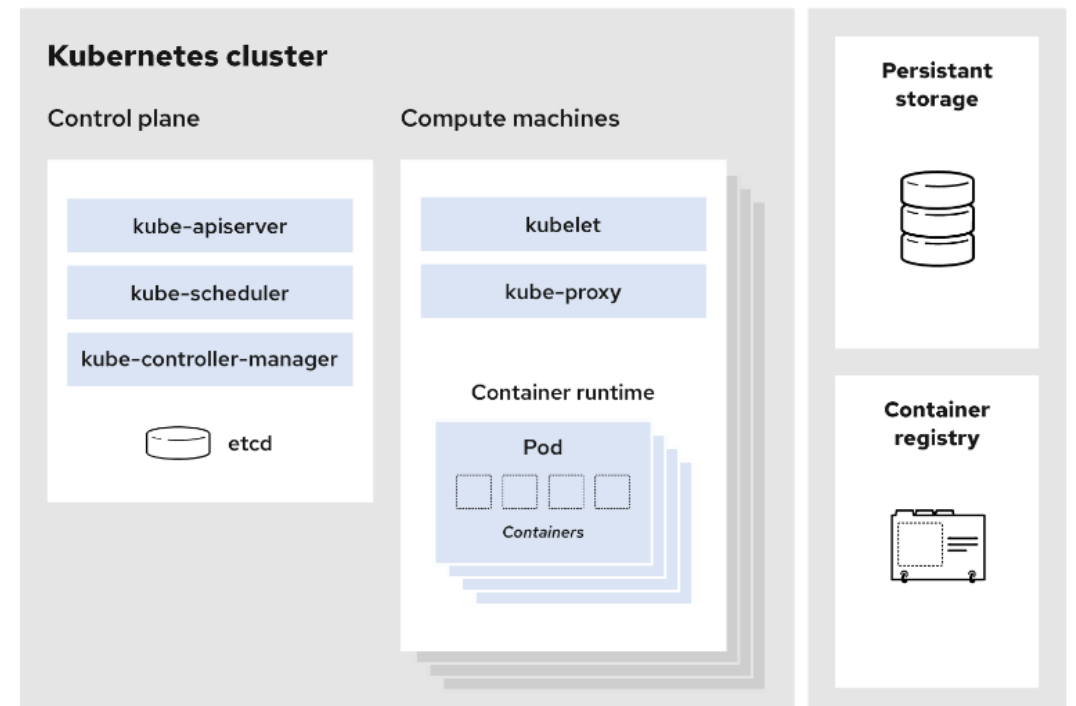
Data Intensive S&E
Life Sciences
Physical Sciences
Systems Engineering
Disaster Response
Multi-Messenger Astrophysics

Composable & Scalable Innovation
Open to Campus Resource Integration
Open Community Support Model
Campus-Scale Instrument integration
BYOR & BYOD
Any Data, Anytime, Anywhere

Category II: A Prototype National Research Platform (PNRP) Project (NSF OAC #2112167)
added significant compute and distributed storage resources to NRP
5-year project: \$5M for Acquisition and Deployment; \$7.25M for Operations and Maintenance
PI = Wuerthwein; Co-PIs: DeFanti, Rosing, Tatineni, Weitzel

The Nautilus Cluster

- Nautilus is a Kubernetes cluster
- Control plane: Manages the cluster and the worker nodes
- Worker Nodes: Maintain running pods and provide Kubernetes runtime environment
- Has been in continuous operation for **6 years**.
- Various storage options like **CephFS; CvmFS; S3**; advanced monitoring tools like **PerfSONAR; traceroute; Prometheus**.
- Diverse computation and data tools: **JupyterHub; WebODM; GitLab; Nextcloud; Overleaf**.
- Collaboration and communication: **Jitsi; EtherPad; Hedgedoc; SyncThing**.
- **500+ nodes; 1500+ GPUs; 50+ FPGAs; Bluefield 2 SmartNICs; SmartSSDs**; programmable **Tofino** switches....

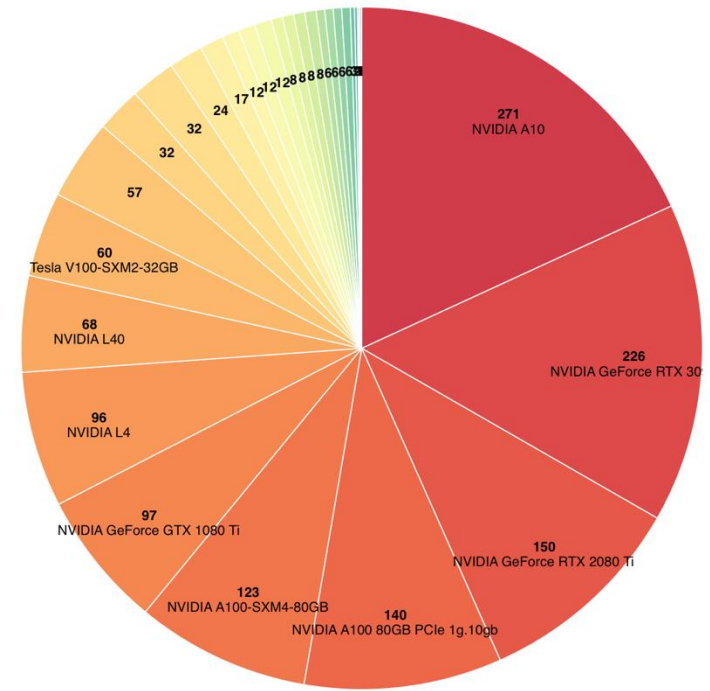


Hardware on Nautilus

- **ARM + x86_64 Nodes**
- **1500+ GPUs with various models**
 - Good for training most of the ML models
 - 4-16 GPUs / node: can request all of them

~600 RTX/GTX “Gaming” GPUs

- **8 Qualcomm Cloud AI 100 Ultra Inference & Fine-tuning Cards:**
 - 1 card = 4 SoCs (System-on-Chip)
 - Each SoC capable of running 25B Param LLM with out-of-the-box config
 - Provisioned via k8s device plugin



<https://observablehq.com/collection/@nrp-nautilus/gpu>

ALL
GPUs



A100
GPUs



<https://grafana.nrp-nautilus.io/>

Hardware on Nautilus

The cluster has different types of resources including CPU, GPU, and FPGAs

- **CPUs:** 16-384 cores/node
- **GPUs:** max 16 in a node
- **FPGAs:** max 4 in a node
- **Memory (RAM):** 16GB – 2.16 TB/node
- **Disk Space (local scratch):** 77GB – 20TB /node

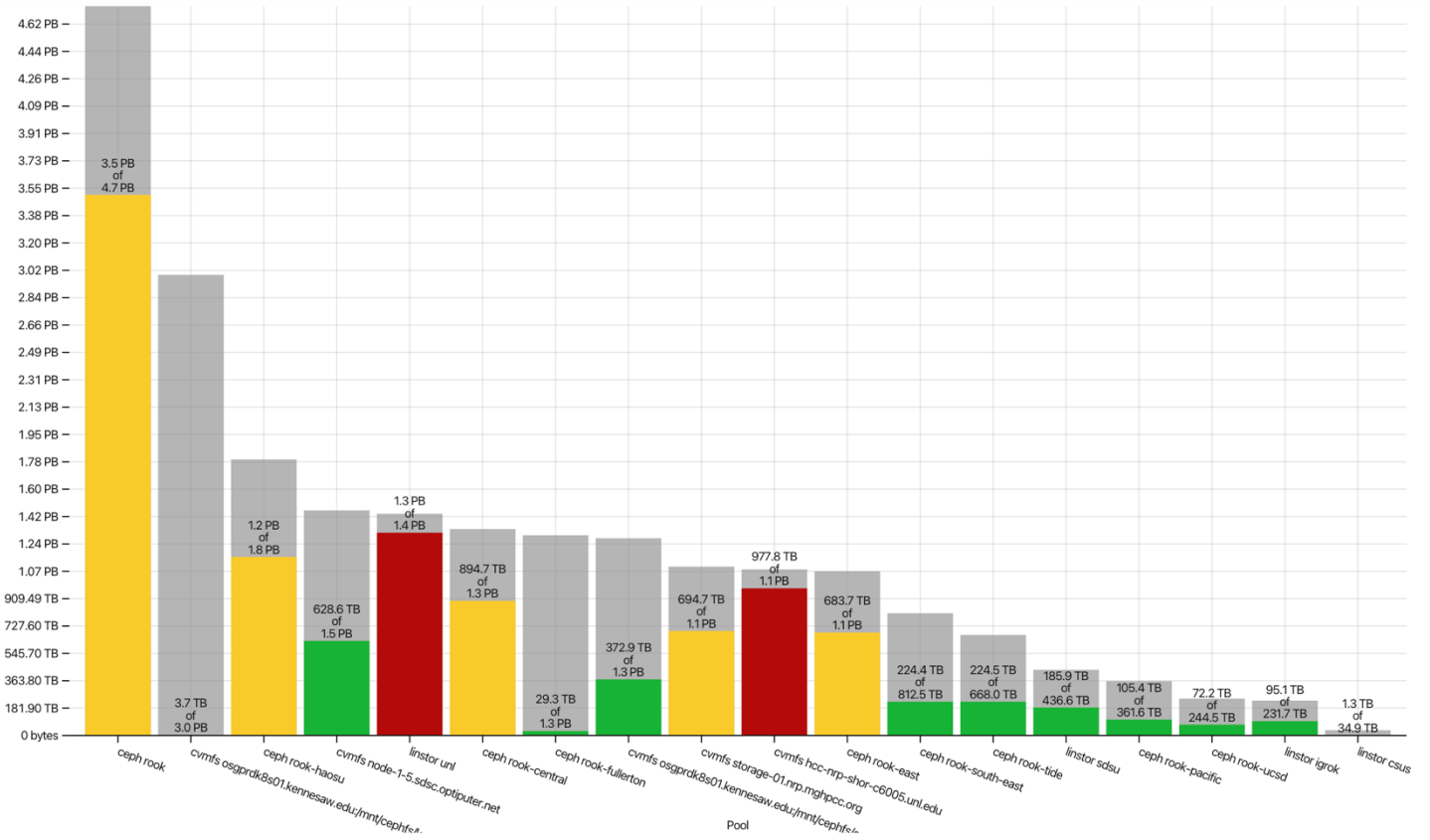
<https://nrp.ai/viz/resources>

- ConnectX-6 NICs, P4-programmable **Tofino** Switches
- **32** P4-programmable Xilinx Alveo U55C **FPGAs**
- **16** BitWare 520N-MX Stratix **FPGAs**
- **3** Xilinx Alveo SN1000 **FPGAs**
- **7** NVIDIA Bluefield 2 **DPU**s
- **2** Xilinx U.2 **FPGA**-powered **SmartSSDs**
- **Kubernetes** cluster with ***Multus***, ***Calico*** and ***ESnet SENSE*** support

Storage Options on Nautilus

- Ceph File System → Ceph FS → ReadWriteMany PersistentVolumes
- Ceph Rados Block Device → Ceph RBD → ReadWriteOnly PersistentVolumes
- Linstor Block → ReadWriteOnly PersistentVolumes
- S3 Compatible storage → Bucket Endpoints
- Open Science Data Federation → OSDF

<https://nrp.ai/viz/storageall/>



Classroom Support

- **Interactive access via Jupyter is a foundational feature of PNRP**
 - > 1000 users from 47 institutions used our JupyterHub
 - >100 namespaces have deployed their own JupyterHub instances
 - **Broader Impact & Open CI Vision & Jupyter → Education as a focus on PNRP**
- **NAIRR Classroom offering – over 25 allocations made on PNRP**
 - Students can login using CILogon with institutional credentials
 - Professor and TA will have admin status to create namespaces for classes
 - GPU nodes each with 8 A10 GPUs, 512GB of RAM, 2 AMD EPYC 7502 CPUs, and 8TB of NVMe
 - Support/Community interaction through Matrix channels
 - Direct ticketing system available for any issue needing longer follow ups

NAIRR Classroom Resources

- We are offering PNRP resources via NAIRR Classroom pilot allocations (<https://nairrpilot.org/opportunities/education-call>)
- **NAIRR Classroom allocations are already in progress (>25 allocations)**

NAIRR Pilot National Artificial Intelligence Research Resource Pilot

Get Started Opportunities Projects News/Events Learn/Get Help myNairrAllocation About

Home / Opportunities / Educational Resources

Preparing your NAIRR Pilot Classroom Request

Questions about this call or need help with allocation? [Submit a ticket](#)

To request access to NAIRR Pilot Classroom resources, you must prepare a description, no longer than three (3) pages, describing your course and requirements for computational resources available through this program. Do not include any proprietary information in proposals.

IMPORTANT: Note that NAIRR Classroom projects are only available on a select set of resource providers. Instructors should be sure that their classroom activities align with the hardware available on NAIRR Classroom resources and are suitably scaled for classroom work.

Eligibility

This call is open to proposals by US based educators and researchers who are teaching undergraduate or graduate courses or shorter duration training sessions to U.S. students in the subject matter in artificial intelligence and require that students use advanced computational resources as part of their coursework. Courses from any discipline are eligible. Courses and training sessions must not allow participants who are not US based.

[Ask me a question!](#)

Filters

AI Capabilities

- Model inference services
- Model training services (GPU)
- AI tools and support

Resource Category

- Federal agency systems
- Other private sector contribution
- Classroom Platform

Resource Status

- Active

Resource Type

- Cloud
- GPU Compute
- Service / Other
- Commercial Cloud

[Reset Filters](#)

Resources

- CloudBank Classroom
- Indiana Jetstream2 GPU
- NIH Cloud Lab
- Prototype National Research Platform (PNRP) Classroom**

Resource Type: Compute

Organization: NRP

Units: Number of Students

User Guide: [Link to User Guide](#)

Features Available:

- RP Allocated
- Federal agency systems
- Classroom Platform
- Service / Other
- Model training services (GPU)
- NAIRR Classroom
- Active

Resource Description

Prototype National Research Platform (PNRP) Classroom offers lecturers at non-profit degree granting higher education institutions the opportunity to request Jupyter resources for teaching a class of students as part of the "NAIRR Classroom" pilot. PNRP offers GPU resources for your students and a Jupyter platform for teaching a class that makes heavy use of AI/ML and. Our default platform has nodes each with 8 A10 GPUs, 512GB of RAM, 2 AMD EPYC 7502 CPUs, and 8TB of NVMe. Depending on your needs, we may be able to provision other GPU resources via the NAIRR pilot and make it available to your class via the PNRP Nautilus cluster (<https://docs.nationalresearchplatform.org>). PNRP provides your class with a set of containers you can customize for the students; an authentication system based on InCommon technology that authenticates your students based on their identity at your institution; and some storage space where you can store the collections of files that your students will need to process as part of your course.

[Ask me a question!](#)

Using Nautilus and NRP



JupyterHub



Coder



Kubernetes

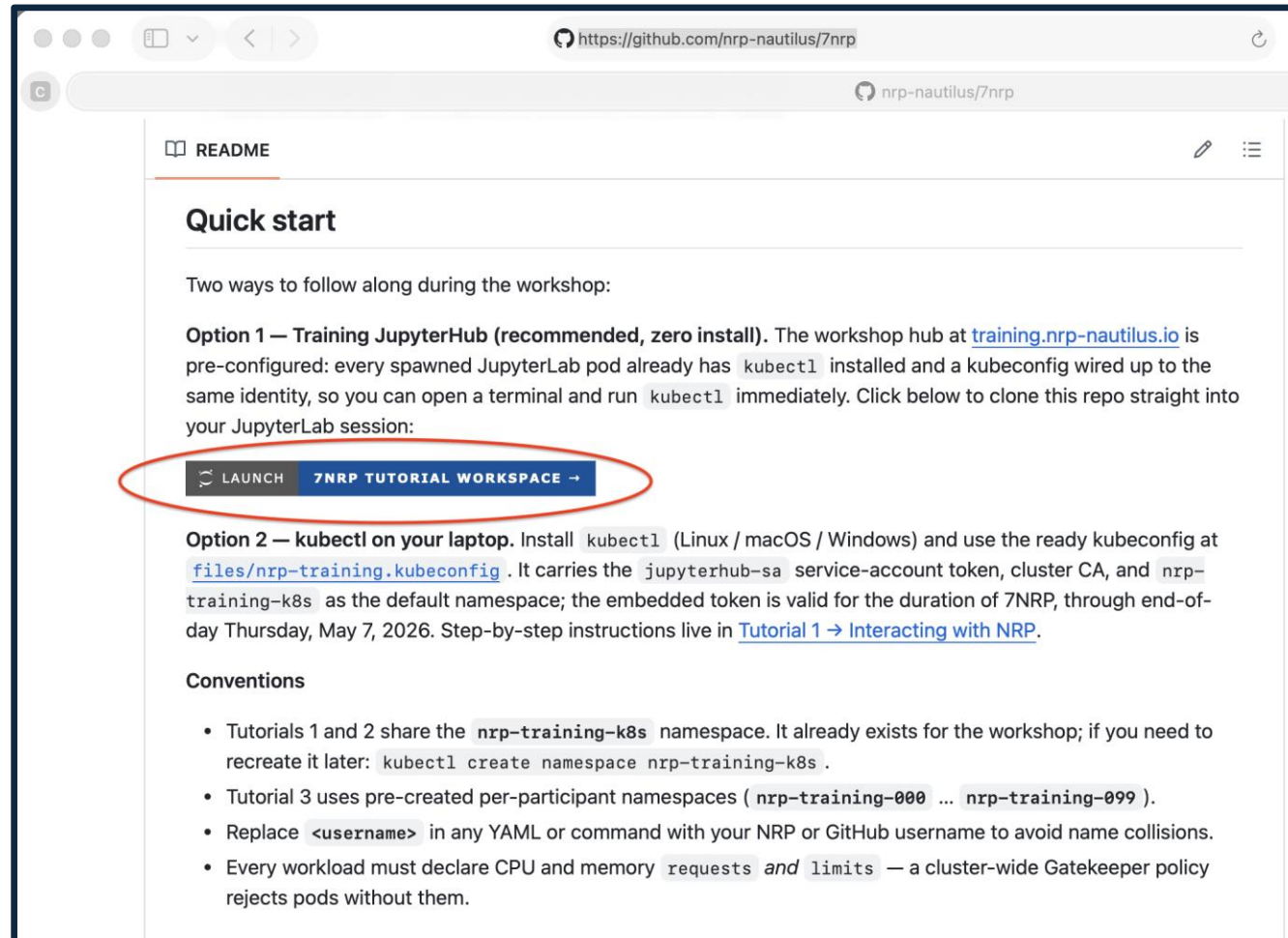
GitHub Repository for Today's Tutorials

<https://github.com/nrp-nautilus/7nrp>

Let's do some hands-on work!

Launch Tutorial Workspace

- We are going use a Jupyter workspace for all of today's hands-on content. Get started by clicking on the link in the github page
- You can authenticate using your institution credentials via CILogon



The screenshot shows a web browser window displaying the GitHub README for the repository 'nrp-nautilus/7nrp'. The page title is 'README' and the repository name is 'nrp-nautilus/7nrp'. The main heading is 'Quick start'. Below this, there is a section titled 'Two ways to follow along during the workshop:'. The first option is 'Option 1 — Training JupyterHub (recommended, zero install)'. It describes a pre-configured JupyterLab environment and includes a red circle around a button that says 'LAUNCH 7NRP TUTORIAL WORKSPACE →'. The second option is 'Option 2 — kubectl on your laptop', which describes installing kubectl and using a ready kubeconfig. Below the options is a 'Conventions' section with a bulleted list of instructions.

Quick start

Two ways to follow along during the workshop:

Option 1 — Training JupyterHub (recommended, zero install). The workshop hub at training.nrp-nautilus.io is pre-configured: every spawned JupyterLab pod already has `kubectl` installed and a kubeconfig wired up to the same identity, so you can open a terminal and run `kubectl` immediately. Click below to clone this repo straight into your JupyterLab session:

LAUNCH 7NRP TUTORIAL WORKSPACE →

Option 2 — kubectl on your laptop. Install `kubectl` (Linux / macOS / Windows) and use the ready kubeconfig at [files/nrp-training.kubeconfig](#). It carries the `jupyterhub-sa` service-account token, cluster CA, and `nrp-training-k8s` as the default namespace; the embedded token is valid for the duration of 7NRP, through end-of-day Thursday, May 7, 2026. Step-by-step instructions live in [Tutorial 1 → Interacting with NRP](#).

Conventions

- Tutorials 1 and 2 share the `nrp-training-k8s` namespace. It already exists for the workshop; if you need to recreate it later: `kubectl create namespace nrp-training-k8s`.
- Tutorial 3 uses pre-created per-participant namespaces (`nrp-training-000` ... `nrp-training-099`).
- Replace `<username>` in any YAML or command with your NRP or GitHub username to avoid name collisions.
- Every workload must declare CPU and memory `requests` and `limits` — a cluster-wide Gatekeeper policy rejects pods without them.

Launch Tutorial Workspace

- **Do not request GPUs or Qualcomm AI devices in the form.** We will request and handle that via Kubernetes workloads in the sessions.

- Choose zero GPUs and Qualcomm AI Devices here



Contact admins in [Matrix](#).

Region
Any

GPUs
0

Qualcomm Cloud AI Devices
0

Cores
1

RAM, GB
8

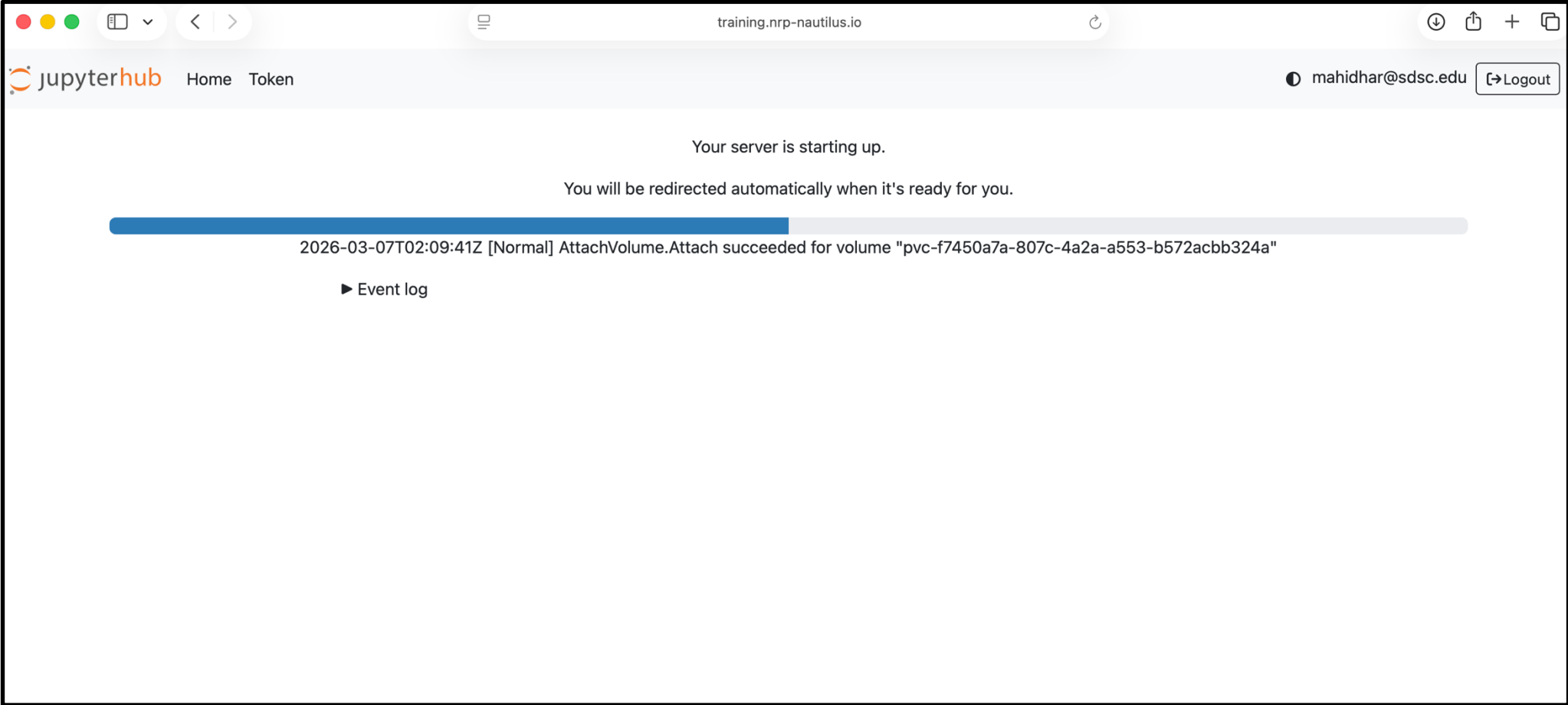
FPGAs
0

GPU type
Any general

Refer to the [documentation](#) for images description.

Image
• NRP Deep Learning & Data Science Full, PyTorch

Jupyter server startup process



Jupyter instance with clone of our repository

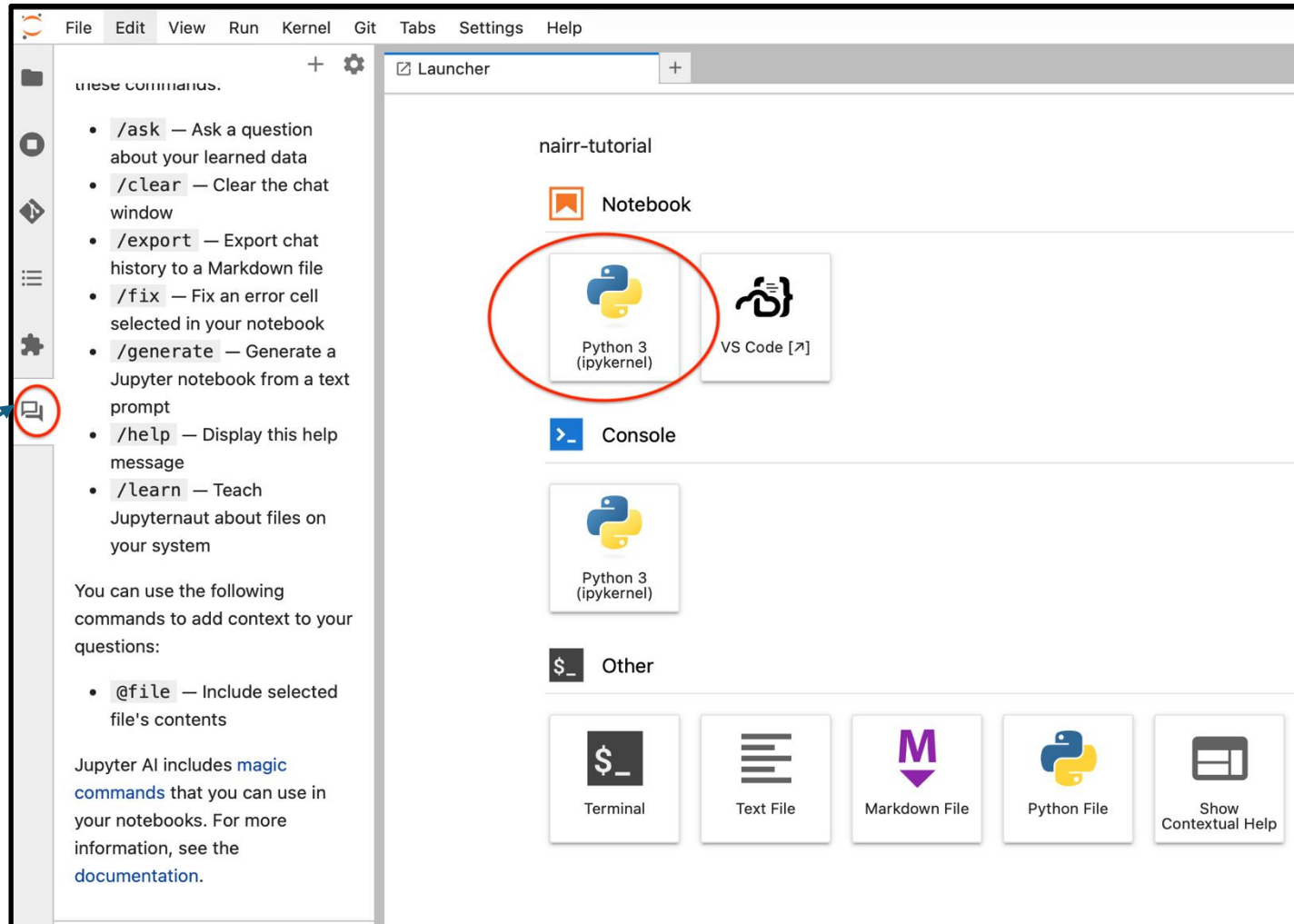
Click to get launcher pane

Clone of our git repo

Click on Python3 notebook icon so we can test things out. Later examples will use the notebooks on the left.

The screenshot displays the JupyterLab interface in a browser window. The browser address bar shows 'training.nrp-nautilus.io' and 'JupyterLab'. The interface includes a menu bar (File, Edit, View, Run, Kernel, Git, Tabs, Settings, Help) and a toolbar with icons for file operations. On the left, a file browser shows a directory '7nrp' containing several files and folders. A red circle highlights the file browser area. On the right, a launcher pane shows a 'Notebook' section with a 'Python 3 (ipykernel)' icon circled in red. Below it are 'Console', 'Python 3 (ipykernel)', and 'Other' sections. A notification banner at the bottom right asks 'Would you like to get notified about official Jupyter news?' with 'Open privacy policy', 'Yes', and 'No' options.

Jupyter AI Chat interface



Click here for chat interface

Jupyter AI Chat interface

The screenshot shows a JupyterLab interface with a code editor and a sidebar. The code editor contains the following Python code:

```
[1]: import random

def calculate_pi_monte_carlo(iterations=1000000):
    inside_circle = 0
    for _ in range(iterations):
        x, y = random.random(), random.random()
        if x**2 + y**2 <= 1:
            inside_circle += 1
    return 4 * inside_circle / iterations

pi = calculate_pi_monte_carlo(1000000)
print(f"Pi ≈ {pi}") # Pi ≈ 3.1415... (varies slightly)

Pi ≈ 3.140464

[ ]:
```

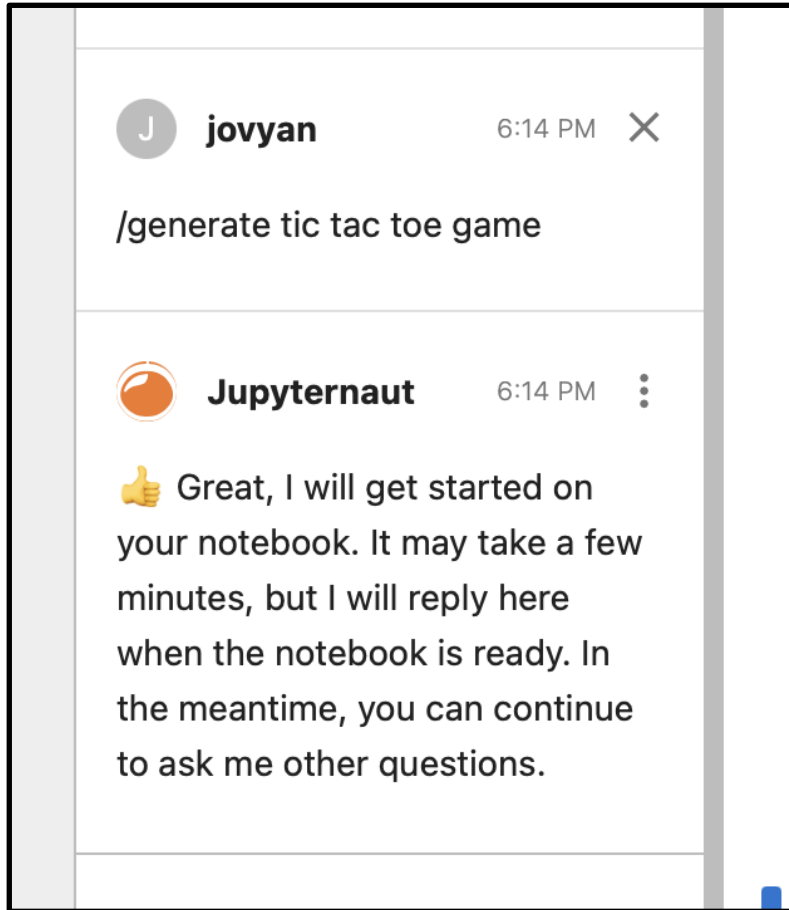
The sidebar on the left contains a section titled "3. Monte Carlo Method" with a description: "Using random sampling to estimate Pi (visualize as random points in a square and quarter circle):". Below this is a code block for the Monte Carlo method. A blue arrow points to a chat icon in the sidebar with the text "Click here for chat interface".

At the bottom right of the interface, there is a notification box that says: "Would you like to get notified about official Jupyter news?" with buttons for "Open privacy policy", "Yes", and "No".

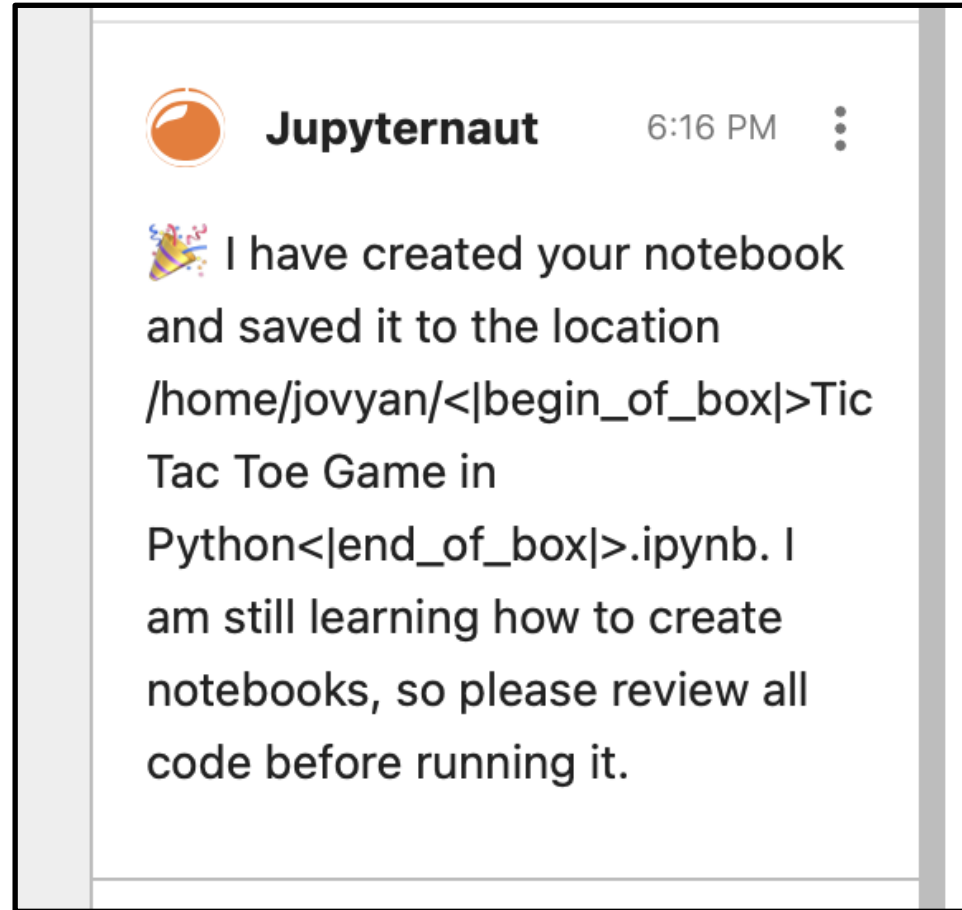
Click here for chat interface

Jupyter AI Chat interface – Generate notebooks

Request in Chat Interface



After a few minutes ... a notebook!



GitHub Repository for Today's Tutorials

<https://github.com/nrp-nautilus/7nrp>

Hand over to Daniel for k8s hands-on