Cloudy With a Chance of Savings: User Resource and Cost Monitoring for JupyterHubs in the Cloud

Jenny Wong
Open Source Applications Engineer @2i2c.org
Tarashish Mishra
Software Engineer @DevelopmentSeed



#JupyterCon

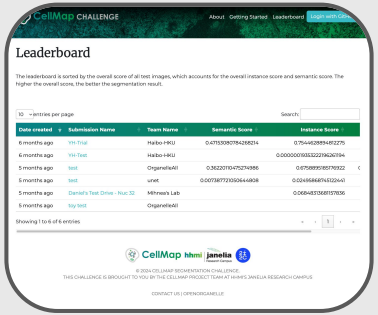
Disclaimer: There will be a lot of Cloudy with a Chance of Meatballs references in this presentation 🍝

> whoami

- 🥇 First time attending JupyterCon
- 💙 Joined 2i2c in 2023
- 👩‍🔬 Former RSE, fluid dynamics researcher, maths grad
- 🇬🇧 Based in Yorkshire, UK



catalystproject.cloud



cellmapchallenge.janelia.org



jupyterhub-cost-monitoring

> find ${MOTIVATION}

- JupyterHubs in the cloud are deployed on virtual machines
- Cloud providers bill you for the services rendered
- There is no cost transparency at the application level
- Admins want to know
  - How many people took part in a workshop we ran?
  - How much compute are our power users consuming?
  - Are we spending money responsibly or recklessly?
- FinOps 🤝 Observability



> which jupyterhub

- 2i2c is a global network of community hubs for interactive learning and discovery
- We manage Zero to JupyterHub distributions on Kubernetes for ~35 communities in research and education
- Majority of infrastructure is deployed on AWS public cloud
- We co-create with our communities and build on open source, making foundational and directed contributions upstream
- We enshrine the community's Right to Replicate their hub infrastructure

showcase.2i2c.cloud

The 2i2c JupyterHub for 2i2c Showcase

The International Interactive Computing Collaboration

Operated by: **2i2c** | Funded by: **2i2c** | Designed by: **2i2c**

Log in to continue

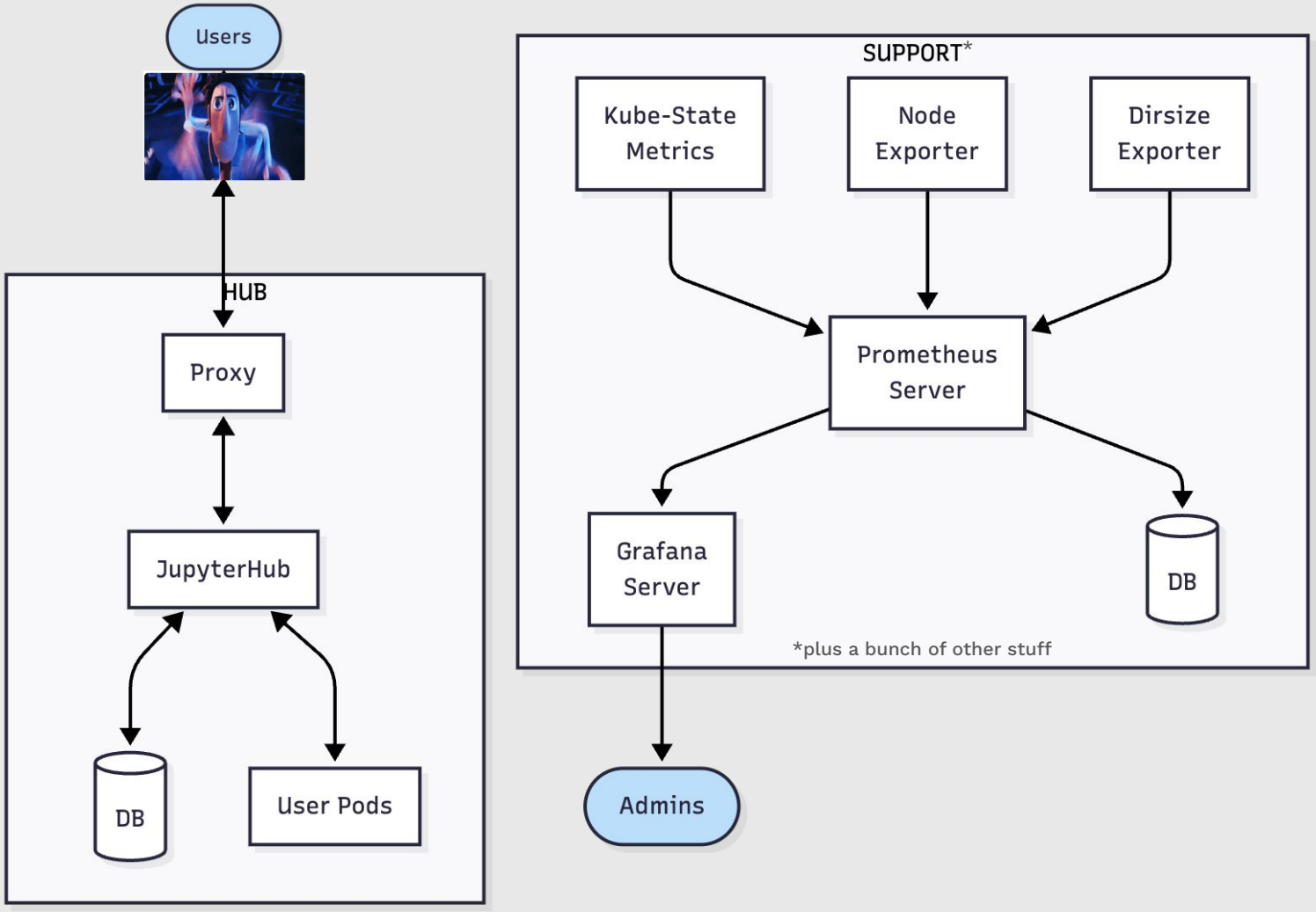
Welcome to the 2i2c Showcase 2i2c JupyterHub.

See 2i2c Team Compass - Showcase Hub for additional documentation. This is a pilot service running on open source infrastructure. See the 2i2c Pilot documentation for usage and deployment information.





Where can I learn more?


Users
HUB
Proxy
JupyterHub
DB
User Pods
SUPPORT*
Kube-State Metrics
Node Exporter
Dirsize Exporter
Prometheus Server
Grafana Server
DB
*plus a bunch of other stuff
Admins

> eval ${USER_COST}

- User-dependent costs can be split into compute and home storage components
- Estimate user cost per component by

  `user cost = usage fraction * total cost`

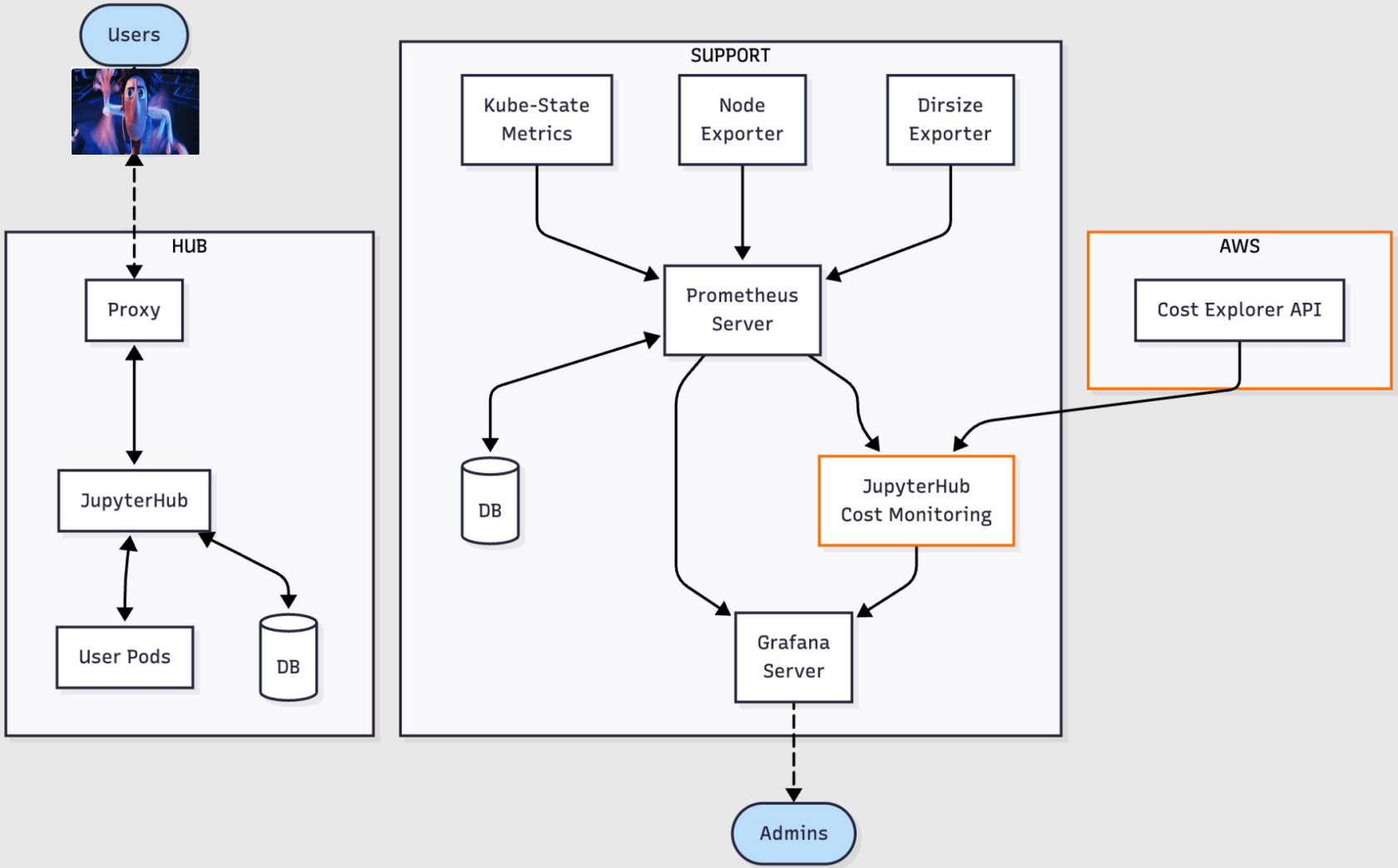


- For compute, the usage fraction is

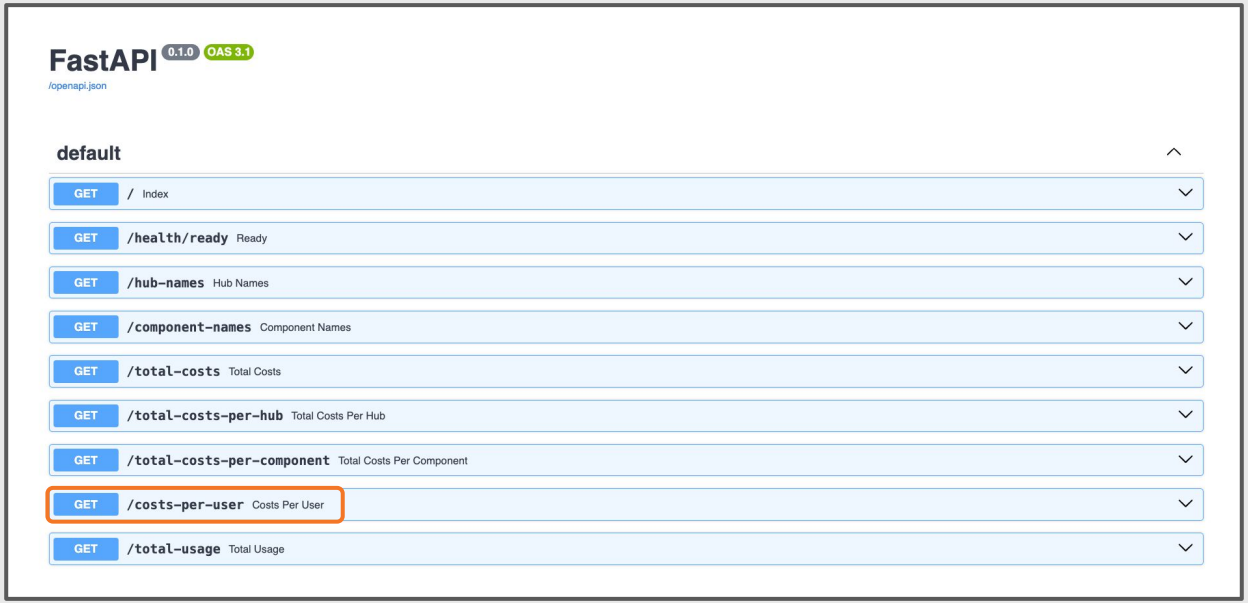
  `memory requested (user) / memory requested (all)`

- For home storage, the usage fraction is

  `home storage (user) / home storage (all)`


Users
HUB
Proxy
JupyterHub
User Pods
DB
SUPPORT
Kube-State Metrics
Node Exporter
Dirsize Exporter
Prometheus Server
DB
JupyterHub Cost Monitoring
Grafana Server
AWS
Cost Explorer API
Admins

> whatis jupyterhub-cost-monitoring



FastAPI backend talks to AWS Cost Explorer and Prometheus and applies calculations to output user cost information

GET `/costs-per-user` Costs Per User

Endpoint to query costs per user by combining AWS costs with Prometheus usage data.

This endpoint calculates individual user costs by:

1. Getting total AWS costs per component (compute, storage) from Cost Explorer
2. Getting usage fractions per user from Prometheus metrics
3. Multiplying total costs by each user's usage fraction

Query Parameters: from (str): Start date in YYYY-MM-DD format (defaults to 30 days ago) to (str): End date in YYYY-MM-DD format (defaults to current date) hub (str, optional): Filter to specific hub namespace component (str, optional): Filter to specific component (compute, home storage) user (str, optional): Filter to specific user usergroup (str, optional): Filter to specific user group limit (int, optional): Limit number of results to top N users by total cost.

Returns: List of dicts with keys: date, hub, component, user, value (cost in USD) Results are sorted by date, hub, component, then value (highest cost first)

Parameters Cancel

Name	Description
from `string | (string | null)` *(query)*	Start date in YYYY-MM-DDTHH:MMZ format 2025-10-01T00:00:Z
to `string | (string | null)` *(query)*	End date in YYYY-MM-DDTHH:MMZ format 2025-10-31T00:00:Z
hub `string | (string | null)` *(query)*	Name of the hub to filter results showcase
component `string | (string | null)` *(query)*	Name of the component to filter results compute
user `string | (string | null)` *(query)*	Name of the user to filter results jnywong

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8080/costs-per-user?from=2025-10-01T00%3A00%3AZ&to=2025-10-31T00%3A00%3AZ&hub=showcase&component=compute&user=jnywong' \
  -H 'accept: application/json'
```

Request URL

```
http://localhost:8080/costs-per-user?from=2025-10-01T00%3A00%3AZ&to=2025-10-31T00%3A00%3AZ&hub=showcase&component=compute&user=jnywong
```

Server response

Code	Details
200	Response body


```
[
  {
    "date": "2025-10-09",
    "user": "jnywong",
    "hub": "showcase",
    "component": "compute",
    "value": 2.72,
  },
  {
    "date": "2025-10-10",
    "user": "jnywong",
    "hub": "showcase",
    "component": "compute",
    "value": 3.65,
  },
  {
    "date": "2025-10-14",
    "user": "jnywong",
    "hub": "showcase",
    "component": "compute",
    "value": 1.3,
  },
  {
    "date": "2025-10-14",
    "user": "jnywong",
```

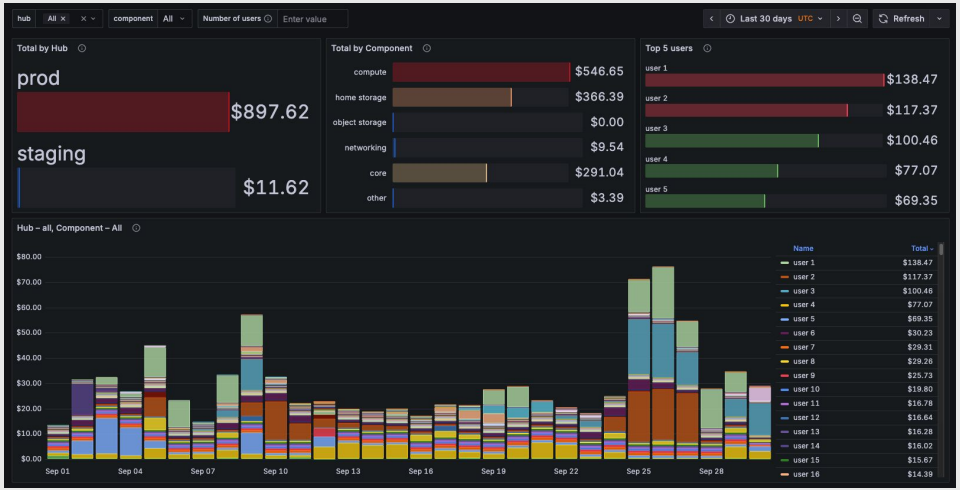

Download

Response headers

```
content-length: 846
content-type: application/json
date: Wed,29 Oct 2025 15:13:24 GMT
server: uvicorn
```

Responses

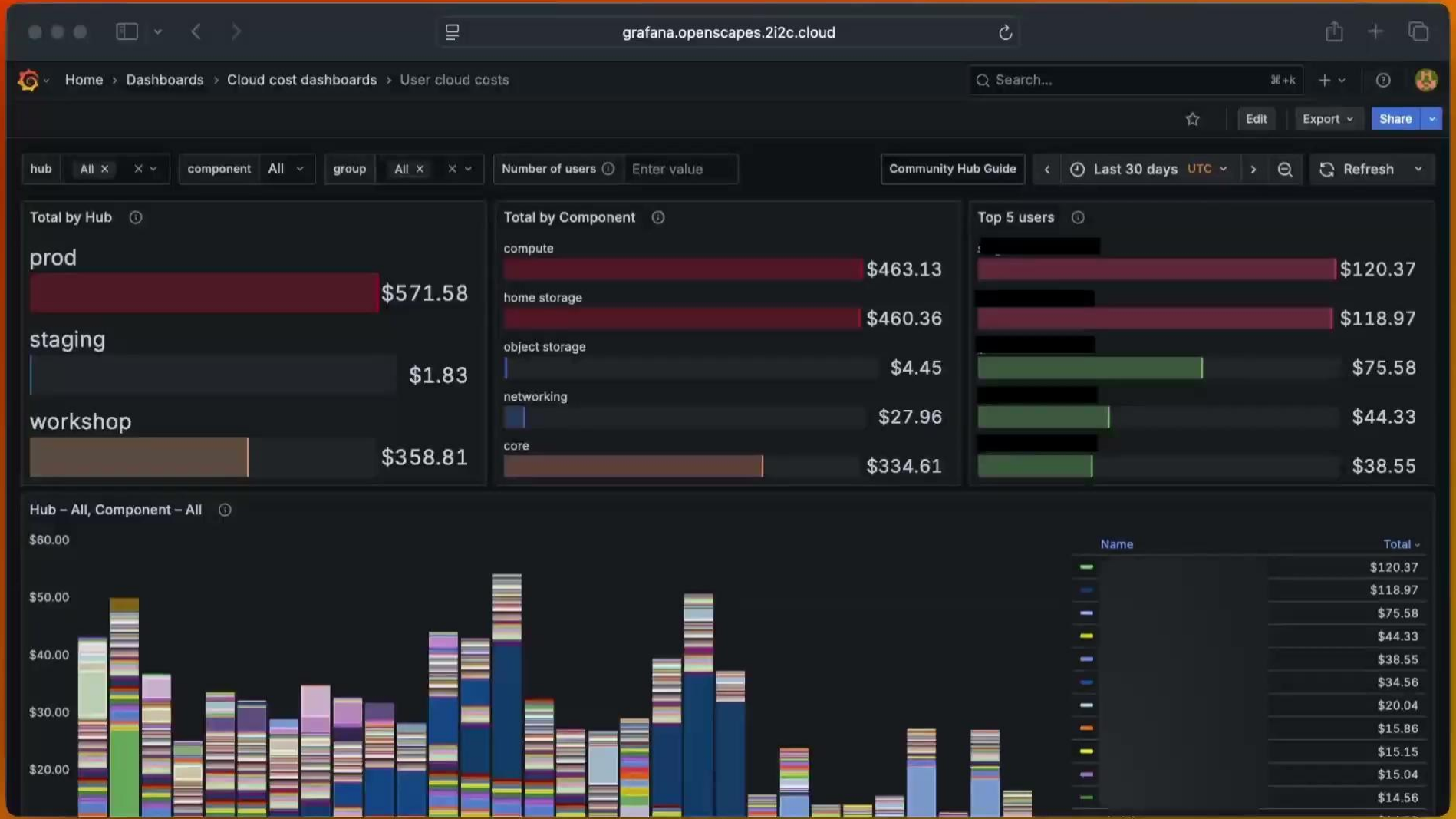
> vi sualization



- jupyterhub-cost-monitoring can be queried and visualized by Grafana
- Dashboard designs are encoded with `grafonnet` for reproducibility
- JupyterHub admins can interactively explore cloud cost dashboards and download `csv` to generate reports

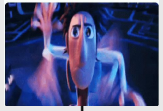
> watch demo

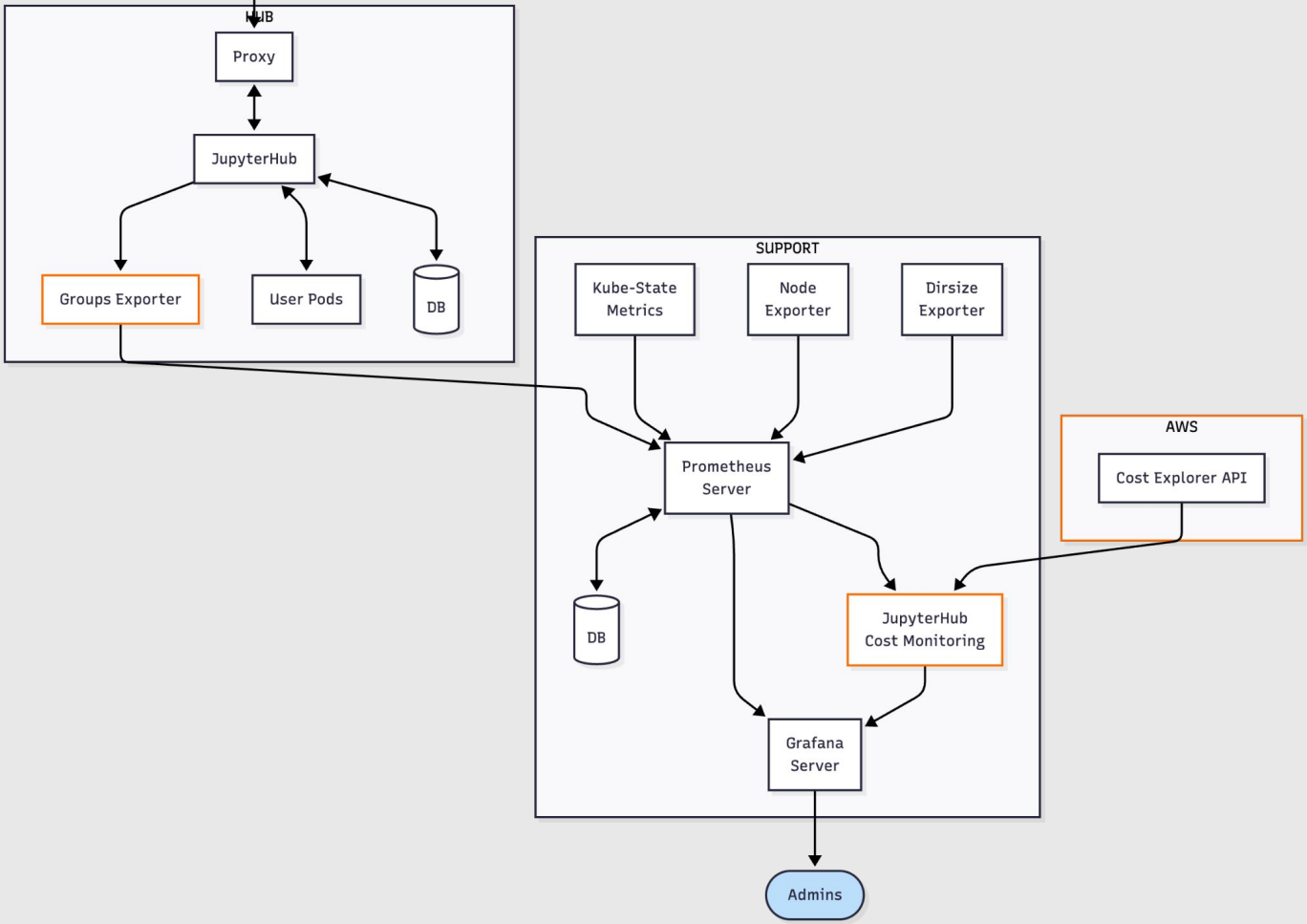



grafana.openscapes.2i2c.cloud
Home › Dashboards › Cloud cost dashboards › User cloud costs
Search...
Edit
Export
Share
hub All
component All
group All
Number of users Enter value
Community Hub Guide
Last 30 days UTC
Refresh
Total by Hub
prod $571.58
staging $1.83
workshop $358.81
Total by Component
compute $463.13
home storage $460.36
object storage $4.45
networking $27.96
core $334.61
Top 5 users
$120.37
$118.97
$75.58
$44.33
$38.55
Hub – All, Component – All
$60.00
$50.00
$40.00
$30.00
$20.00
Name
Total
$120.37
$118.97
$75.58
$44.33
$38.55
$34.56
$20.04
$15.86
$15.15
$15.04
$14.56

> groups plz

- Cloud costs for user groups are a natural extension to individual user costs, especially for hubs with a large user base
- JupyterHub stores user group memberships in its database and we can fetch this from its API
- Memberships are generally managed by admins via authentication method, e.g. GitHub groups, OpenID
- jupyterhub-groups-exporter is a custom Prometheus exporter that exposes user group info as a metric
- User group info can be joined with usage metrics




HUB
Proxy
JupyterHub
Groups Exporter
User Pods
DB
SUPPORT
Kube-State Metrics
Node Exporter
Dirsize Exporter
Prometheus Server
DB
AWS
Cost Explorer API
JupyterHub Cost Monitoring
Grafana Server
Admins

> groups plz



- Upstream JupyterHub usage dashboards 👉 [jupyterhub/grafana-dashboards]
- PromQL for memory requests by user group

```
sum(
      kube_pod_container_resource_requests{resource="memory", namespace=~"$hub_name",
      pod=~"jupyter-.*"}  * on (namespace, pod)
      group_left(annotation_hub_jupyter_org_username)
      group(kube_pod_annotations{namespace=~"$hub_name",
      annotation_hub_jupyter_org_username=~".*"})
      by (pod, namespace, annotation_hub_jupyter_org_username)
      * on (namespace, annotation_hub_jupyter_org_username) group_left(usergroup)
      group(label_replace(jupyterhub_user_group_info{namespace=~"$hub_name",
      username=~".*", usergroup=~"$user_group"}, "annotation_hub_jupyter_org_username",
      "$1", "username", "(.+)")) by (annotation_hub_jupyter_org_username, usergroup,
      namespace)
) by (usergroup, namespace)
```

grafana.openscapes.2i2c.cloud

Home › Dashboards › Cloud cost dashboards › User cloud costs

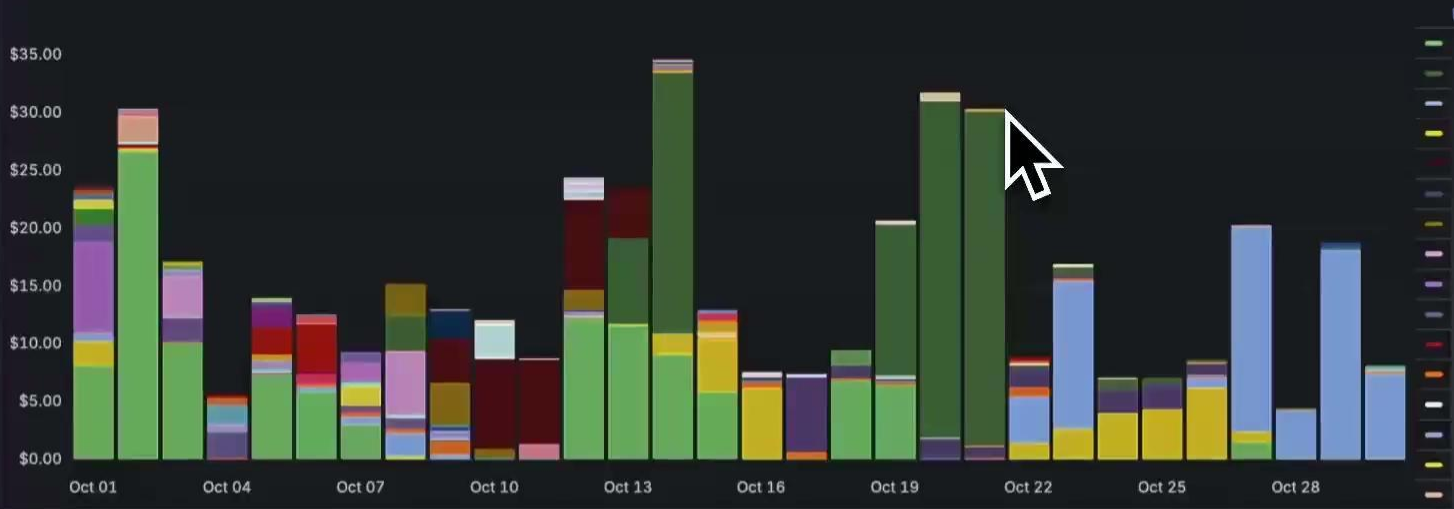
Search... ⌘+k

Edit | Export | Share

hub: All × prod × workshop × | component: compute | group: All × | Number of users: Enter value | Last 30 days UTC | Refresh

Community Hub Guide

Hub – All, Component – compute



Name	Total
	$114.20
	$107.05
	$69.34
	$34.76
	$31.22
	$18.15
	$8.97
	$8.87
	$7.69
	$7.35
	$6.80
	$6.53
	$3.20
	$3.03
	$2.51

Hub – prod, Component – compute



Total
$108.87
$102.40
$65.29

> wait

- Account for under-utilization
  - Cost for unused disk space is attributed to proportional usage
  - Would be a nice way to raise awareness of wasted resources and improve efficiency
  - `jupyterhub-home-nfs` ⚡ talk today @ 3.45pm by Yuvi (2i2c)
- Support for GPU compute and BinderHubs
- Plug in cost data from more cloud providers
- Generalise config and contribute to upstream projects

> touch base.py



- Let me know what you think! QR code 🤳
- Come say hi any time ☕
- Special thanks to
  - Tarashish @Development Seed
  - DSE Team @NASA ODSI
  - NASA VEDA and MAAP,
  - Earthscope, CryoCloud, Openscapes
  - 2i2c
  - You :)