

Nautilus cluster



Dmitry Mishin

Applications Developer

San Diego Supercomputer Center / Calit2, UCSD

dmishin@ucsd.edu

Nautilus is a distributed hyperconverged cluster

● Multi-Institution

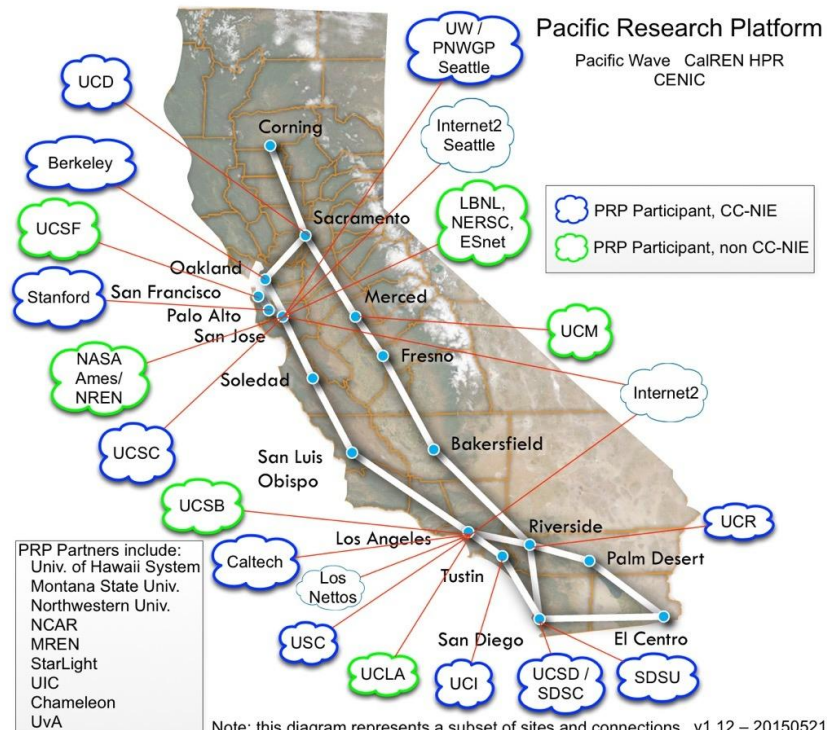
- .edu
- .gov
- .org
- .com

● Hyper-Converged

- CPU
- GPU
- Storage
- non von Neumann

● Global Federation

- CILogon.org Identity
- Namespace Isolation
- OIDC refresh token service



New NSF CHASE-CI Grant Creates a Community Cyberinfrastructure: Adding a Machine Learning Layer Built on Top of the Pacific Research Platform

**CI-New: Cognitive Hardware and Software Ecosystem
Community Infrastructure (CHASE-CI)**

For the Period September 1, 2017 – August 31, 2020

SUBMITTED – January 18, 2017



PI: Larry Smarr, Professor of Computer Science and Engineering, Director Calit2, UCSD
Co-PI: Tajana Rosing, Professor of Computer Science and Engineering, UCSD
Co-PI: Ken Kreutz-Delgado, Professor of Electrical and Computer Engineering, UCSD
Co-PI: Ilkay Altintas, Chief Data Science Officer, San Diego Supercomputer Center, UCSD
Co-PI: Tom DeFanti, Research Scientist, Calit2, UCSD

MSU, UCM, UCB, UCSC, UCI, UCR, UCSD, SDSU, Stanford, Caltech



NSF Grant for High Speed “Cloud” of 256 GPUs
For 30 ML Faculty & Their Students at 10 Campuses
for Training AI Algorithms on Big Data

JACOBS SCHOOL OF ENGINEERING

SDSC

UC San Diego



CHASE-CI

This project, called the **Cognitive Hardware And Software Ecosystem Community Infrastructure (CHASE-CI)**, will build a cloud of hundreds of affordable Graphics Processing Units (GPUs), networked together with a variety of neural network machines to facilitate development of next generation cognitive computing.

This cloud will be accessible by 30 researchers assembled from 10 universities via the NSF-funded Pacific Research Platform. These researchers will investigate a range of problems from image and video recognition, computer vision, contextual robotics to cognitive neurosciences using the cloud to be purpose-built in this project.



256 FP16/32 GPUs in 32 2U Intel Scalable Dual 12 core with Optane Memory

FIONA8: a FIONA with 8 GPUs

Supports PRP Data Science Machine Learning--4M GPU Core Hours/Week

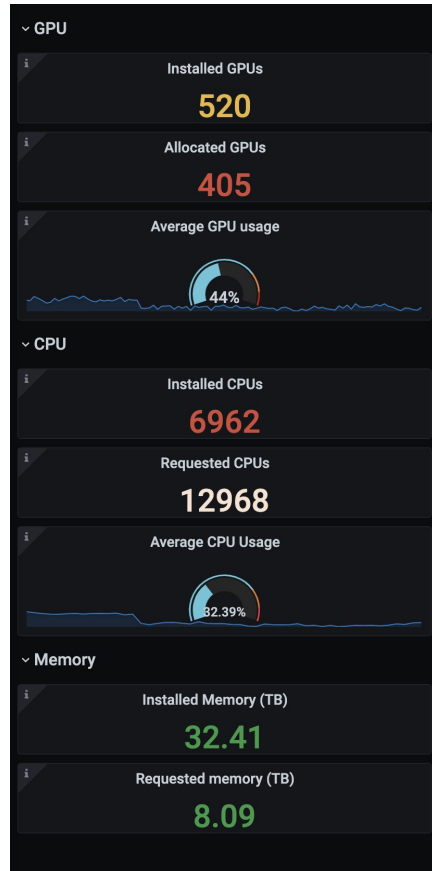


24 CPU Cores, 32,000 GPU cores, 96 GB RAM, 2TB SSD, Dual 10Gbps ports

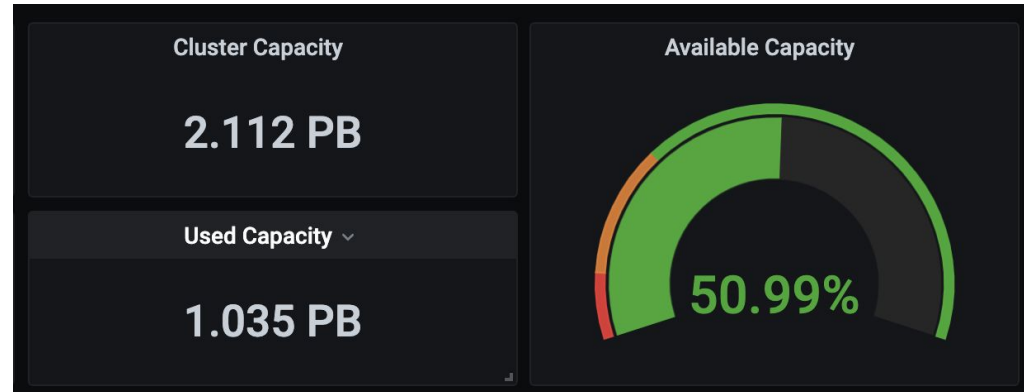
3" High; ~\$16,000



Nautilus is a distributed hyperconverged cluster



Ceph storage:

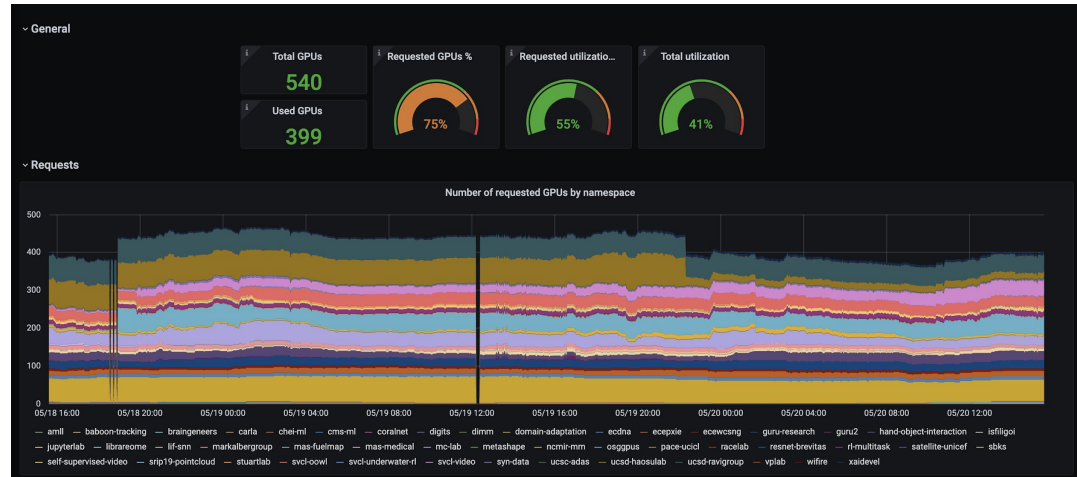


Typical use of Nautilus

Admins - Network monitoring tools
(PerfSonar, sflow, tstat, artemis)

Users - Mostly Machine Learning, GPUs + CPUs
(Tensorflow, PyTorch, etc)

Additional services - Jupyter, GitLab,
Nextcloud, WebODM, RocketChat



Federated Identity



Nautilus requests access to the following information. If you do not approve this request, do not proceed.

- Your CILogon username
- Your name
- Your email address
- Your username and affiliation from your identity provider

Select An Identity Provider:

Goldsmiths, University of London
Geological Survey of Slovenia
Gonzaga University
Google

Search:

Remember this selection:

Log On

By selecting "Log On", you agree to [CILogon's privacy policy](#).

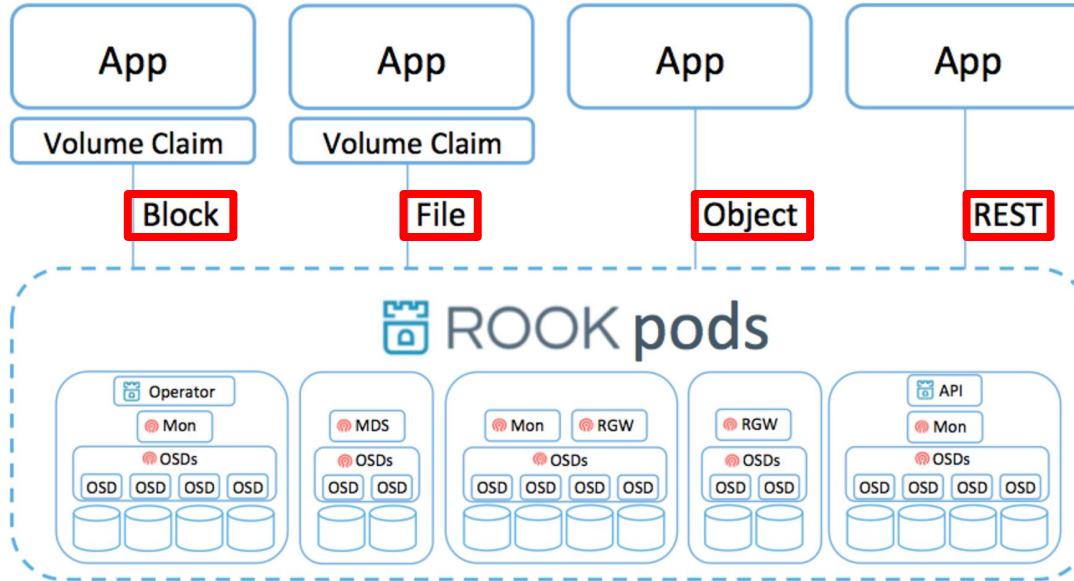
For questions about this site, please see the FAQs or send email to help @ cilogon.org.
Know your responsibilities for using the CILogon Service.
See acknowledgements of support for this site.



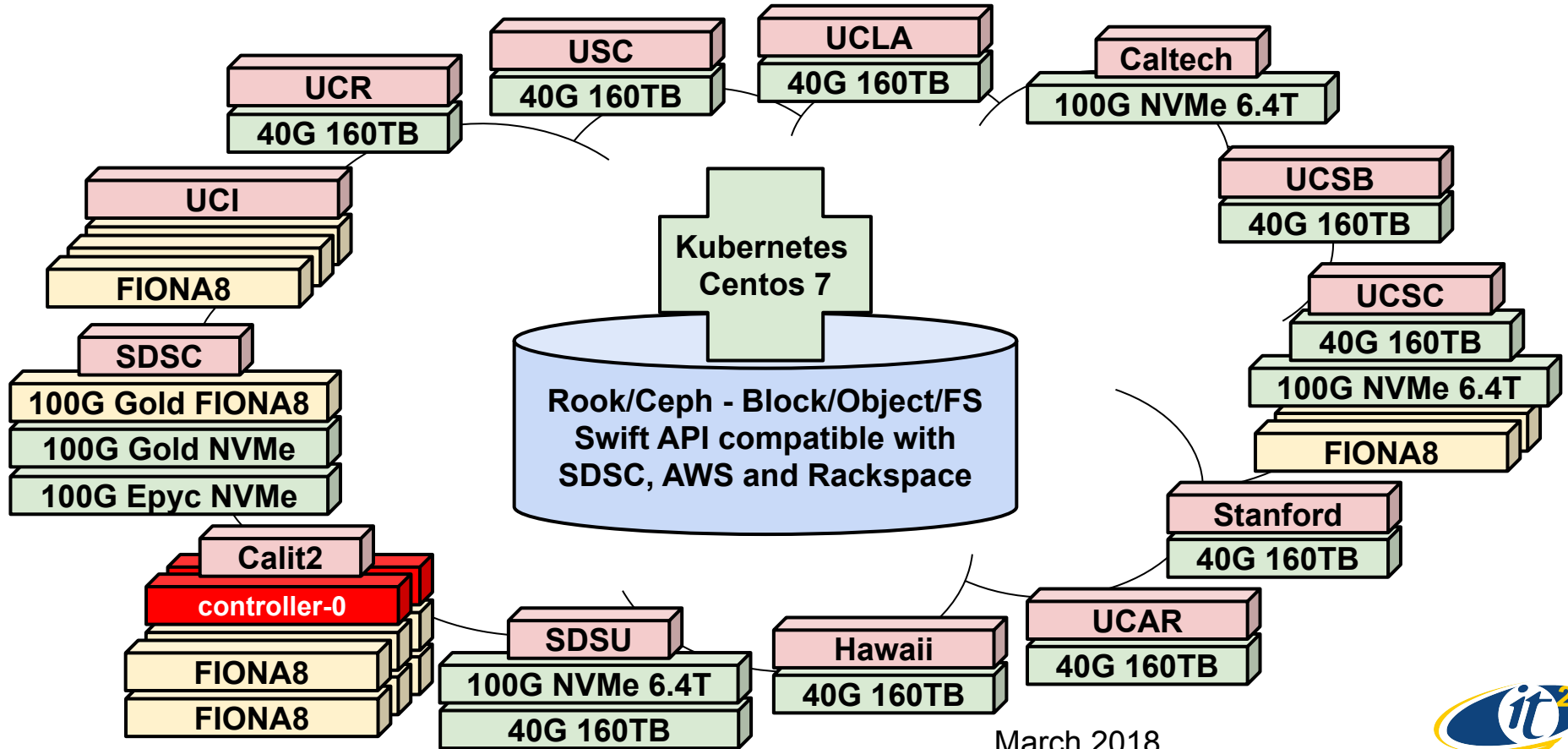
Rook is Cloud Native Ceph in our Hyper-converged cluster

Design

With Rook running in the Kubernetes cluster, Kubernetes applications can mount block devices and filesystems managed by Rook, or can use the S3/Swift API for object storage. The Rook operator automates configuration of the Ceph storage components and monitors the cluster to ensure the storage remains available and healthy. There is also a REST API service for configuring the Rook storage and a command line tool called `rook`.



Nautilus 2PB+ Storage upgrade



March 2018



Schedule FPGAs



The Xilinx FPGA device plugin for Kubernetes is a Daemonset deployed on the kubernetes(a.k.a k8s) cluster which allows you to:

Discover the FPGAs inserted in each node of the cluster and expose info of the FPGAs such as quantities, DSA(shell) type and timestamp, etc

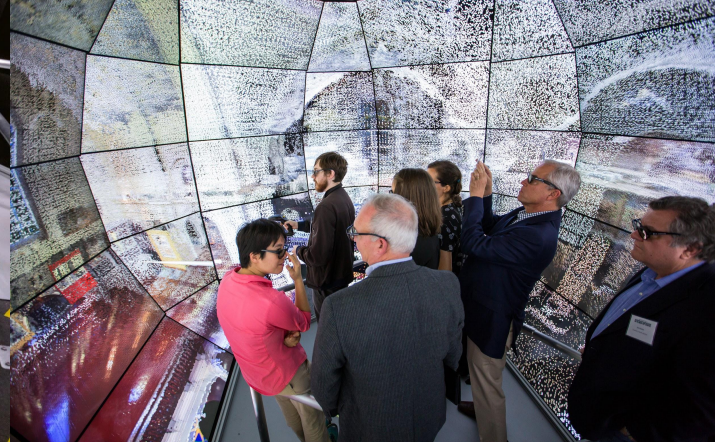
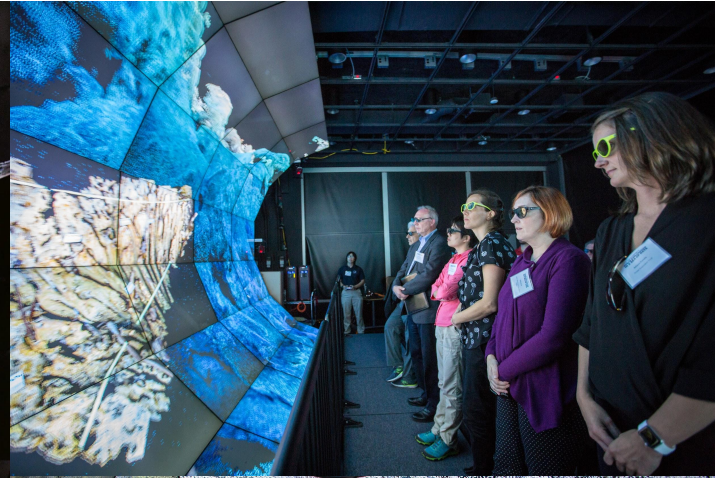
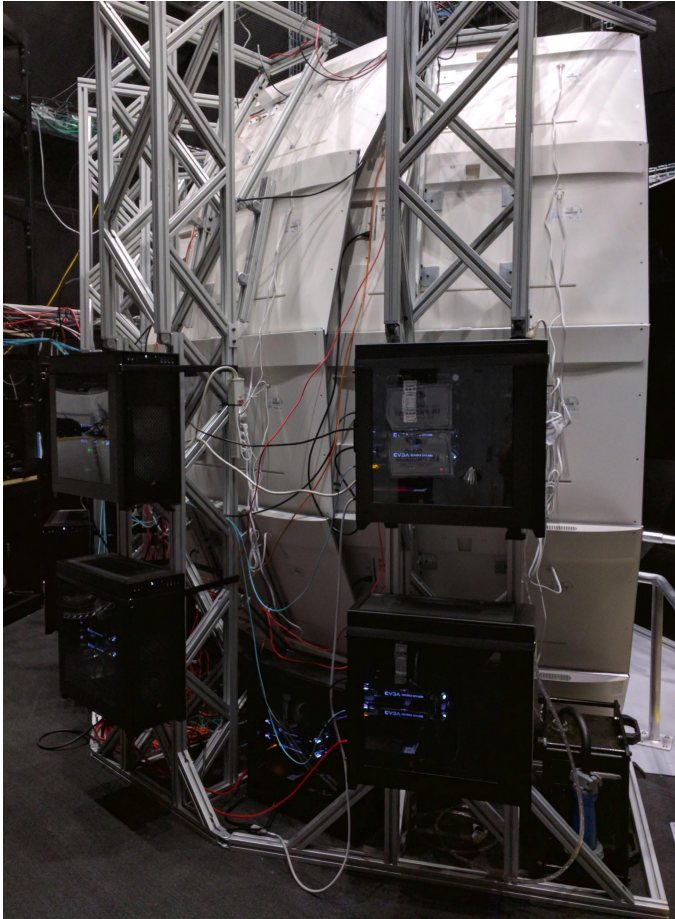
Run FPGA accessible containers in the k8s cluster

More info about k8s device plugin, please refer to

<https://kubernetes.io/docs/concepts/extend-kubernetes/compute-storage-net/device-plugins/>

https://github.com/Xilinx/FPGA_as_a_Service

Walls, Caves and Waves



UCSD Adding >350 Game GPUs to Data Sciences Cyberinfrastructure - Devoted to Data Analytics and Machine Learning

UC San Diego

IT SERVICES

88 GPUs
for Students

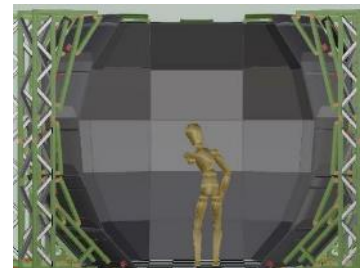
UCSD Cognitive Science

JACOBS SCHOOL OF ENGINEERING

 Open Science Grid

48 GPUs for
OSG Applications

SDSC 



SunCAVE 70 GPUs
WAVE + Vroom 48 GPUs

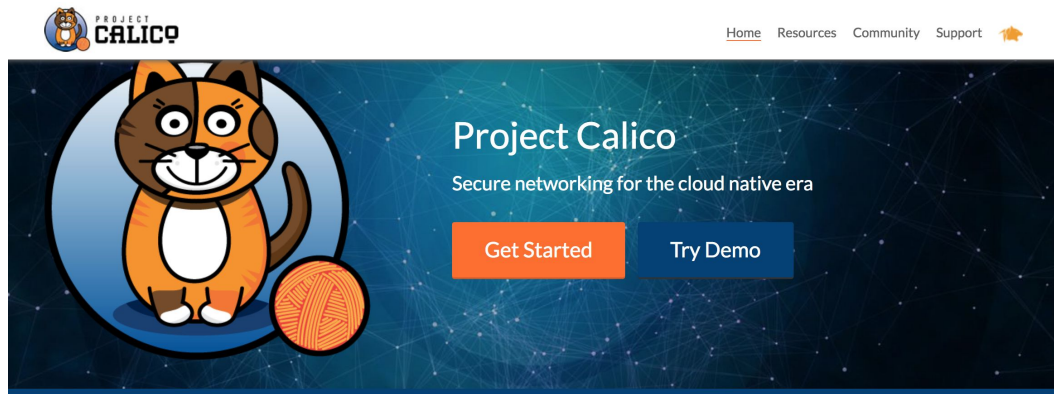


FIONA with
8-Game GPUs

CHASE-CI Grant Provides
96 GPUs at UCSD
for Training AI Algorithms on Big Data



Project Calico



Why Calico?

Free and open source, Project Calico is designed to simplify, scale, and secure cloud networks



Simple

Let's remove the complexity

Traditional SDNs are complex, making them hard to deploy and troubleshoot. Calico removes that complexity, with a simplified networking model designed for the demands of today's cloud-native applications.



Scalable

From dev/test to enterprise deployment

Unlike SDNs that require a central controller, limiting scalability, Calico is built on a fully distributed, scale-out architecture. So it scales smoothly from a single developer laptop to large enterprise deployments.



Secure








Policy-based micro-segmentation

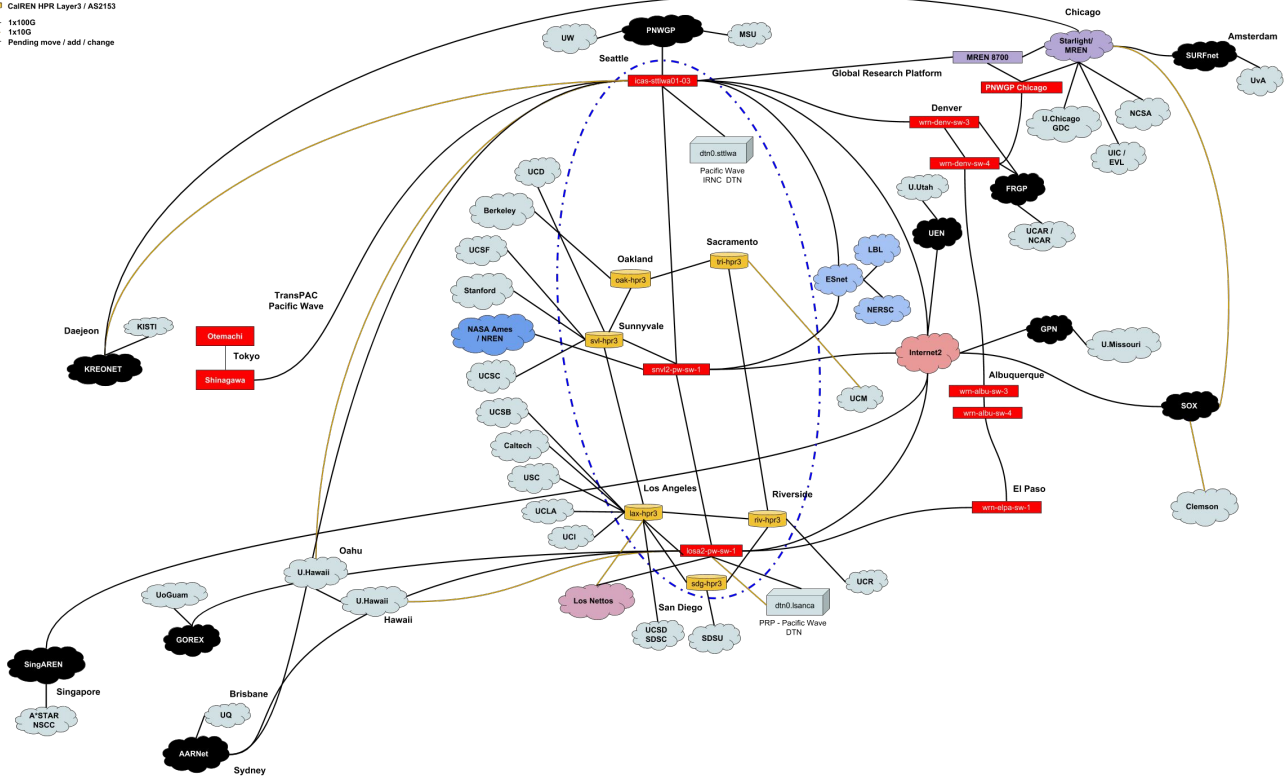
Defining secure network policy used to be reserved for skilled network engineers. Calico's powerful micro-segmentation capabilities build on a simple policy language that naturally expresses the developer's intent.

- CNI plugin
- Full cluster network security
 - a. Nodes firewall
 - b. Namespaces isolation

Pacific Research Platform

<http://pacificresearchplatform.org>

-  PRP Participant
-  PRP Layer2 / AS395889 (in development)
-  Pacific Wave
-  CalREN HPR Layer3 / AS2153
-  1x100G
-  1x10G
-  Pending move / add / change



NOTE: this diagram represents a subset of sites, devices, and connectors

v0.40.16
20180716

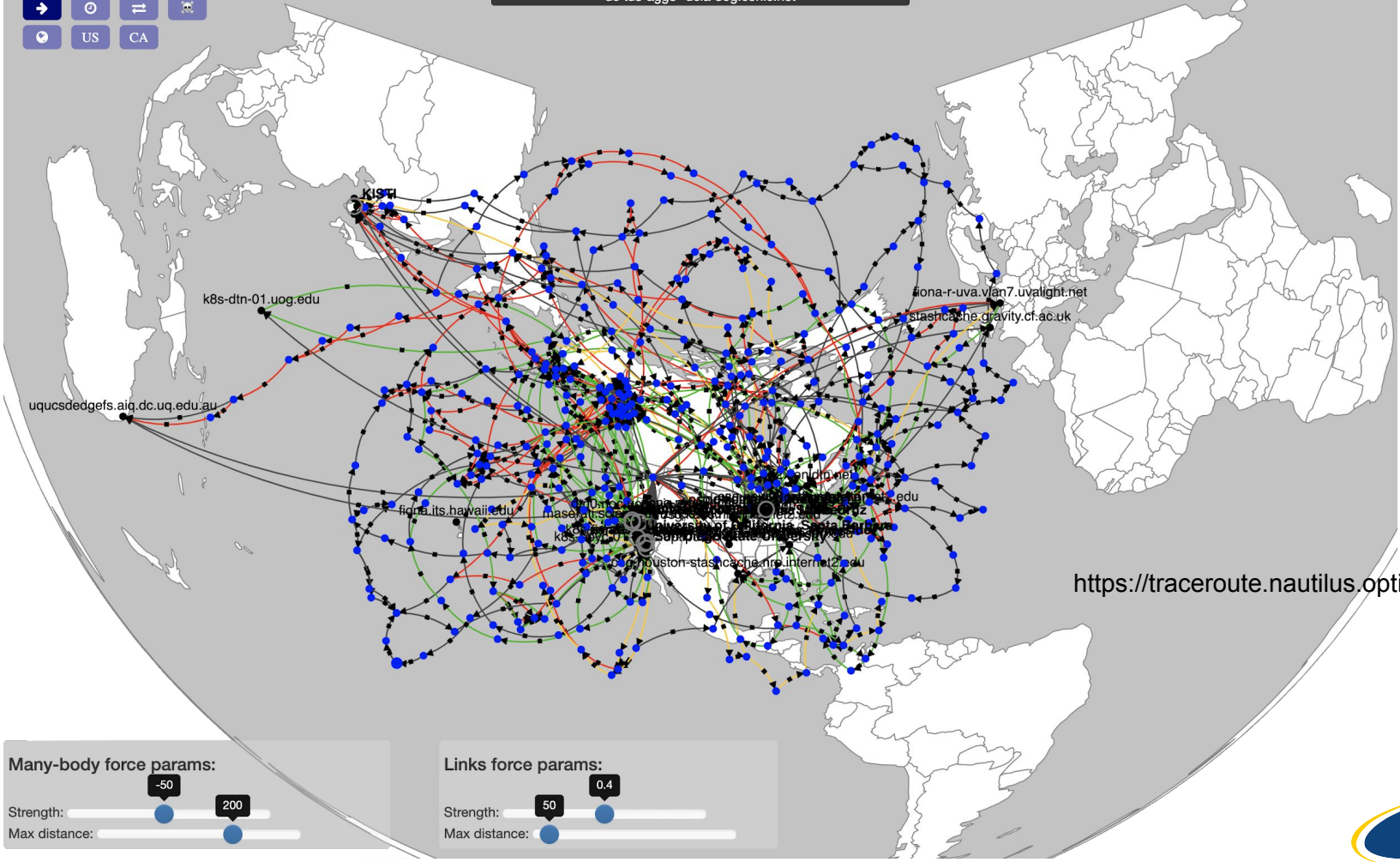


PerfSonar for network monitoring

AUTOMATION !!!

- Perfsonar deploys automatically as new nodes are joined to the cluster.
- No Human is needed to configure the MaDDash.
- The Meshconfig is handled by a golang webservice.





<https://traceroute.nautilus.optiputer.net>

Many-body force params:



Links force params:



ElastiFlow sflow visualization

Overview | Top-N | Threats | Flows | Geo IP | AS Traffic | Exporters | Traffic Details | Flow Records

Client/Server | Src/Dst | AS



Flow Exporter: Select...
Source AS: Select...
Destination AS: Select...
Service: Select...

Source AS (flow records)



- The Regents of the U...
- San Diego Supercom...
- University of Califom...
- University of Califom...
- University of Califom...
- University of Califom...

Destination AS (flow records)

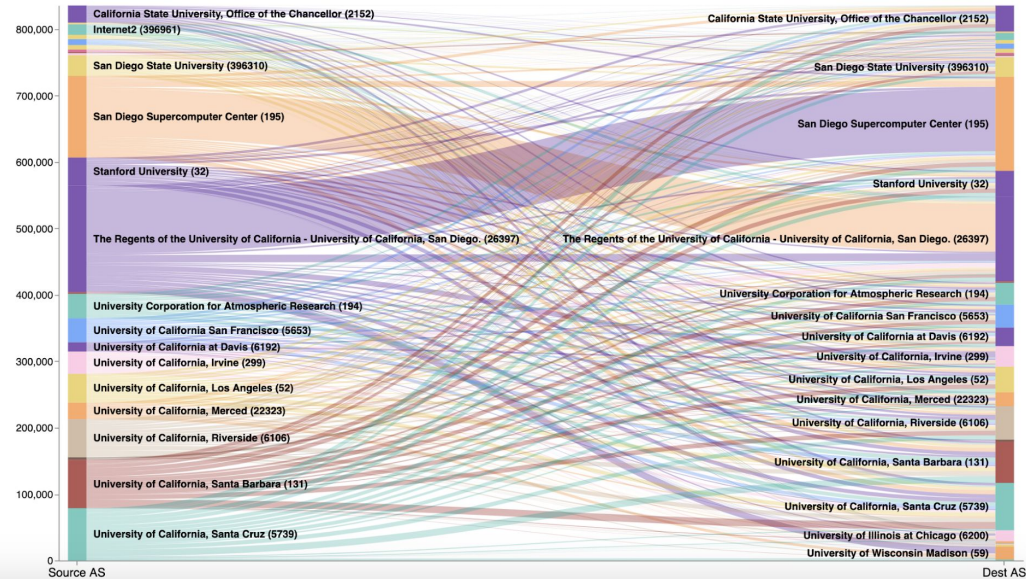


- The Regents of the U...
- San Diego Supercom...
- University of Califom...
- University of Califom...
- University of Califom...
- California State Univ...

Services (flow records)



- apsolab-tags (TCP/54...
- sun-sr-https (TCP/64...
- UDP/10400
- https (TCP/443)
- http (TCP/80)
- UDP/10401



ElastiFlow™ provides network flow data collection and visualization using the Elastic Stack (Elasticsearch, Logstash and Kibana). It supports Netflow v5/v9, sFlow and IPFIX flow types (1.x versions support only Netflow v5/v9).



JupyterLab running as a service

The screenshot displays the JupyterLab interface. On the left, there is a sidebar with sections for 'Files', 'Running', 'Commands', 'Cell Tools', and 'Tabs'. The 'Files' section shows a directory structure under 'brian2 > tutorials' with files like '1-intro-to-brian-neuro...', '2-intro-to-brian-synap...', '3-intro-to-brian-simul...', and 'brian.png'. The 'Running' section shows the same files with their last modified times. The main area is a notebook titled '1-intro-to-brian' with a Python 3 kernel. The notebook content includes a title 'Multiple neurons', a text block 'So far we've only been working with a single neuron. Let's do something interesting with multiple neurons.', and a code cell with the following code:

```
In [23]: start_scope()

N = 100
tau = 10*ms
eqs = '''
dv/dt = (2-v)/tau : 1
'''

G = NeuronGroup(N, eqs, threshold='v>1', reset='v=0', method='exact')
G.v = 'rand()'

spikemon = SpikeMonitor(G)

run(50*ms)

plot(spikemon.t/ms, spikemon.i, '.k')
xlabel('Time (ms)')
ylabel('Neuron index');
```

Below the code cell is a raster plot showing the activity of 100 neurons over time. The y-axis is labeled 'neuron index' and ranges from 0 to 100. The x-axis is labeled 'Time (ms)' and ranges from 0 to 50. The plot shows a dense pattern of black dots representing spikes, indicating that many neurons are active simultaneously.

At the bottom of the interface is a terminal window titled 'Terminal 1' showing the following output:

```
Building wheels for collected packages: brian2, py-cpuinfo
Running setup.py bdist_wheel for brian2 ... done
Stored in directory: /home/jovyan/.cache/pip/wheels/bd/f1/b4/cd2fa2e4cb808f22d4d94ee0c9b6
94c3f14f45c274e7b5643
Running setup.py bdist_wheel for py-cpuinfo ... done
Stored in directory: /home/jovyan/.cache/pip/wheels/a0/7a/be/03a05b26f48a7c49fdce56f34fdde
3c3043feed6ed8afde3c
Successfully built brian2 py-cpuinfo
Installing collected packages: py-cpuinfo, brian2
Successfully installed brian2-2.1.2 py-cpuinfo-3.3.0
jovyan@jupyter-046f34e74d10322b014d783335b00692-40ucsd-2eedu:~$
```

Kubernetes provides Jupyter Notebooks direct access to the GPU.

WebODM

Drone Data Processing Summary

05 March, 2018

Treasure Island, Flight 1, High Quality Preset, No resize

Treasure Island (urban area), nadir grid mapping mission with 80/70 overlap, 250 ft, 107 images, 17 acres.

Flight Info

Location: Treasure Island
Date: 2018-02-17
Start time: 10:13
End time: 10:19
UAV: Phantom III Standard
Sensor: Phantom III Standard

Capture Settings

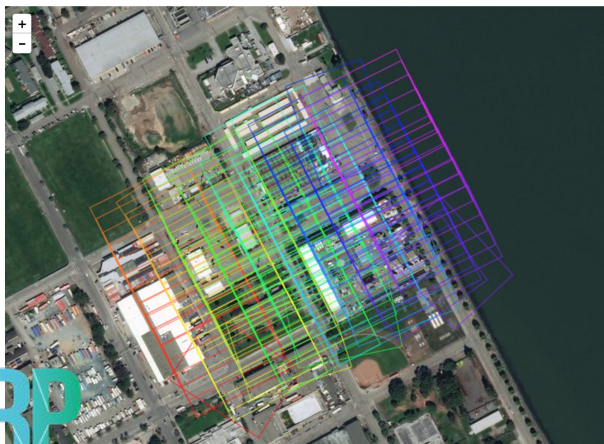
Flight Control App: Pix4Dcapture
Mission Type: grid
Num Images: 107
Front overlap: 80%
Side overlap: 70%
Altitude (ft): 250
Image GSD (in): 1.31
Drone Speed: fast
Trigger mode: fast
White balance: sunny
Exposure: auto

Processing Settings

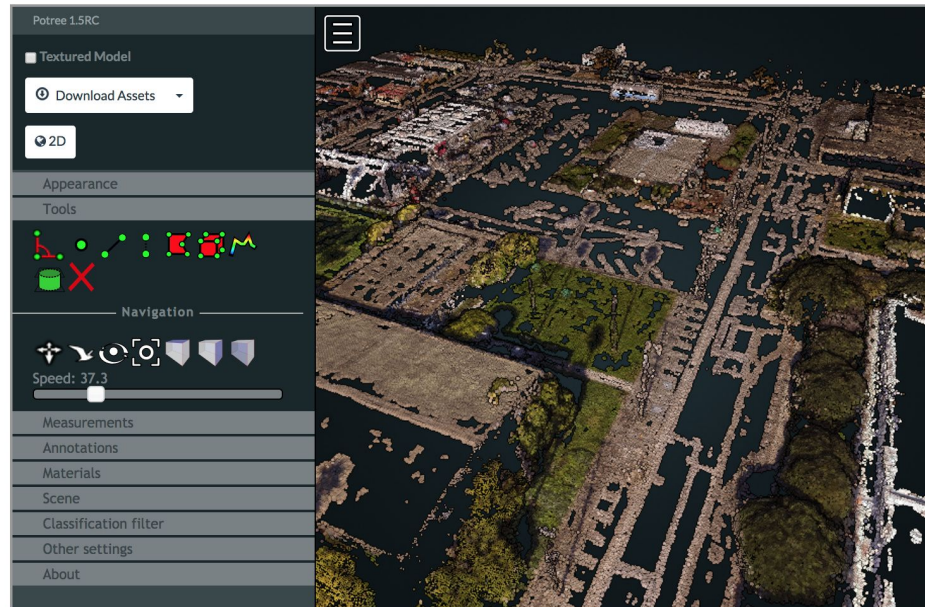
Platform: WebODM
StitchID: ti_2018-02-17_#f01_stch01
Parameter Preset: High Quality
Processing Time (min): 192
Description: High Quality Preset, no resize

Images

Locations Footprints



WebODM Results



Full screen

Assessment: Significant warping of cars and building tops. Mild warping of parking lot lines.



GitLab code and container repo

Projects Groups More

Search or jump to...

6

7






New project

Projects

Your projects Starred projects Explore projects

Filter by name... Last updated

All Personal

	prp / nautilus-cluster Owner	★ 0 Updated 1 hour ago
	prp / rook Owner	★ 0 Updated 1 hour ago
	prp / perfsonar Owner	★ 0 ✓ Updated 1 hour ago
	calit2-caves / plugin_devel Owner a place to build calvr plugins, old and new.	★ 1 ✓ Updated 5 hours ago
	prp / elastiflow Owner	★ 0 ✓ Updated 17 hours ago

<https://gitlab.nautilus.optiputer.net>

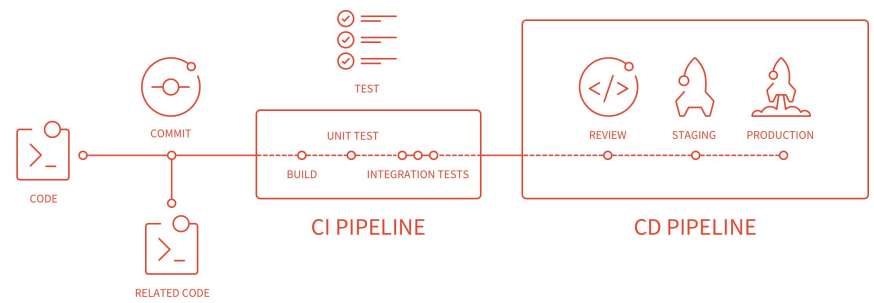




```

1 image: gitlab-registry.nautilus.optiputer.net/prp/golang-docker-gitlabci
2
3 before_script:
4   - docker login -u gitlab-ci-token -p $CI_BUILD_TOKEN gitlab-registry.nautilus.optiputer.net
5
6 stages:
7   - build-and-push
8
9 build-and-push-perfsonar:
10  stage: build-and-push
11  tags:
12  - build-as-docker
13  only:
14    changes:
15      - perfsonar-toolkit/**/*
16      - perfsonar-toolkit/*
17  script:
18  - cd perfsonar-toolkit
19  - docker build -t gitlab-registry.nautilus.optiputer.net/
20  - docker tag gitlab-registry.nautilus.optiputer.net/${CI
21  - docker push gitlab-registry.nautilus.optiputer.net/${
22
23 build-and-push-testpoint:
24  stage: build-and-push
25  only:
26    changes:
27

```





RocketChat for community support

The screenshot shows a RocketChat interface with a sidebar on the left containing a list of channels and conversations. The main area displays a chat history for the #federation channel. The chat includes messages from users like jed, jhess, dimm, john.graham, and adrienjt, along with system messages and a link to a GitHub repository. A search bar is visible at the bottom.

Discussions

Conversations

- # nautilus-alerts 3:34 PM rocket.cat: Traffic Sentinel Events
- # federation 2:37 PM jim.chen: Hello, I am am following u...
- jed 2:29 PM You: Fixed, plz test
- # general 2:12 PM You: If you force http, it will break a...
- # prometheus-alerts 8:58 AM rocket.cat sent an attachment
- # news yesterday You: File uploads in rocketchat are ...
- joel.polizzi 12:57 PM You: Thank!!
- ar-noc 12:06 PM yjungk: Thank you again!
- # nautilus-ops 10:39 AM You: john.graham Did you run a sle...
- MichaelZhang 7:35 AM I will try to run my script again.
- dtn-as-a-service Monday You: It does, but in my opinion it's t...
- osg Monday isfiligoi: Much better! Thanks
- john.graham Monday You: ok!
- # spare-parts-list Monday john.graham: cool
- iperez Monday You: yup
- spowell Monday You: ("user"; "spowell"; "access_k...
- isfiligoi Monday ok, let me look into it
- # haosu-cluster Saturday mohammad.shafiei: I'm using arou...
- hnwren-nns Saturday

ROCKET.CHAT

#federation

adrienjt online Chat Now

k8s-gen4-05.calit2.optiputer.net	NotReady	<none>	38d	v1.18.1
ludo.calit2.optiputer.net	Ready	master	38d	v1.18.1

dimm Admin 5:36 PM
Yeah 😊
Maybe run some personars?
👍 1
Actually that's a good use case

john.graham Admin 5:36 PM
cool idea

dimm Admin 5:37 PM
I think that was the plan for federation in the beginning.

May 20, 2020

jed 8:10 AM Has joined the channel.

jhess Admin 8:51 AM
dimm, adrienjt has the "0.8.1" code dropped in the admiralty repo? atm the repo appears to list App Version 0.8.0

dimm Admin 8:53 AM
There's a helm chart referencing the right one...

jhess Admin 8:55 AM
helm install multicluster-scheduler admiralty/multicluster-scheduler --namespace admiralty --version 0.8.0 -f values.yaml
?
with the values.yaml contents posted last week?

adrienjt Leader 9:12 AM
I'm going to skip 0.8.1 and release 0.8.2 today. In the meantime, you can use the 0.8.0 chart, with 0.8.2-rc.2 image tag overrides.

jhess Admin 9:12 AM
okay -- cool

following destructions / prereq's here: <https://github.com/admiraltyio/multicluster-scheduler/tree/master/charts/multicluster-scheduler>

github.com
admiraltyio/multicluster-scheduler
A system of Kubernetes controllers that intelligently schedules workloads across clusters. - admiraltyio/multicluster-scheduler

👍 2

shawnw 2:27 PM User jim.chen added by shawnw. unread messages

jim.chen 2:37 PM
Hello, I am am following up from SDX call discussion on Federation Demonstration for SC20, JohnH indicated support for such demo, JohnH indicated interest to participate, anyone will be interest to lead such project?


Message

Currently:
>1K users
>300K messages

<https://rocket.nautilus.optiputer.net>

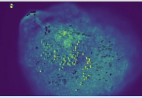


Namespaces as a collaboration environment



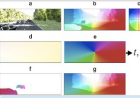
adalab

University of California, San Diego: Advanced Data Analytics




braingeneers

UC Santa Cruz & UC San Francisco: Reinforcement Learning with the recordings derived from cortical organoids



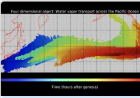
carl-uci

University of California, Irvine: Reinforcement learning and motion decomposition




chei-ml

University of California, San Diego and SIO: Deep learning for coral species segmentation



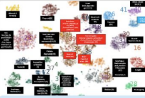
connect

University of California, San Diego: Machine Learning in Earth Sciences



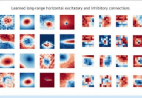
coralnet

University of California, San Diego: Computer Vision and Machine Learning for Coral Ecology



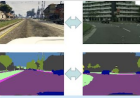
deepgtex-prp

Clemson University: Deep Learning in Oncogenomics




desalab

University of California, San Diego: Biologically plausible deep learning for vision



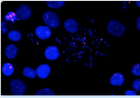
domain-adaptation

University of California, San Diego: Transfer and multitask learning




dreiman

University of California, Santa Cruz: Deep learning research with applications in cosmology, extragalactic astronomy and astrophysics



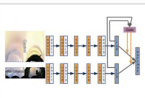
ecdna

University of California, San Diego: Deep Learning for medical imaging



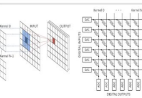
ecewcsng

University of California, San Diego: Deep Learning for sensor data processing



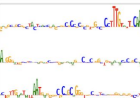
guru-research

University of California, San Diego: Applications of



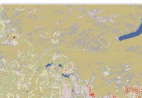
kube-environment

University of California, San Diego: ReRAM-based



kundajelab

Stanford University: Deep Learning in Genomics



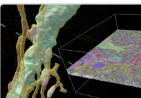
mas-fuelmap

University of California, San Diego: Generating



mesl

University of California, San Diego: Convolutional



ncmir-mm

University of California, San Diego: Image

Currently:
>300 namespaces

<https://ucsd-prp.gitlab.io/nautilus/namespaces/>



Federation

Scheduling

- Schedule pods in a remote cluster, control execution, watch the state

Network

- Be able to communicate to pods in remote cluster

Storage

- Be able to store data in remote cluster

AARCH64 IoT node

Also used as AARCH64 Gitlab runner for building IoT container images



Thanks to Our Support:

- **US National Science Foundation (NSF) awards**
 - **CNS 0821155, CNS-1338192, CNS-1456638, CNS-1730158, ACI-1540112, & ACI-1541349**
- **University of California Office of the President CIO**
- **UCSD Chancellor's Integrated Digital Infrastructure Program**
- **UCSD Next Generation Networking initiative**
- **Calit2 and Calit2 Qualcomm Institute**
- **CENIC, PacificWave and StarLight**
- **DOE ESnet**

