A walk in the cloud: Facilitating climate research using Amazon Web Services

Natalie O'Leary
Aparna Radhakrishnan

CIMES

PRINCETON UNIVERSITY

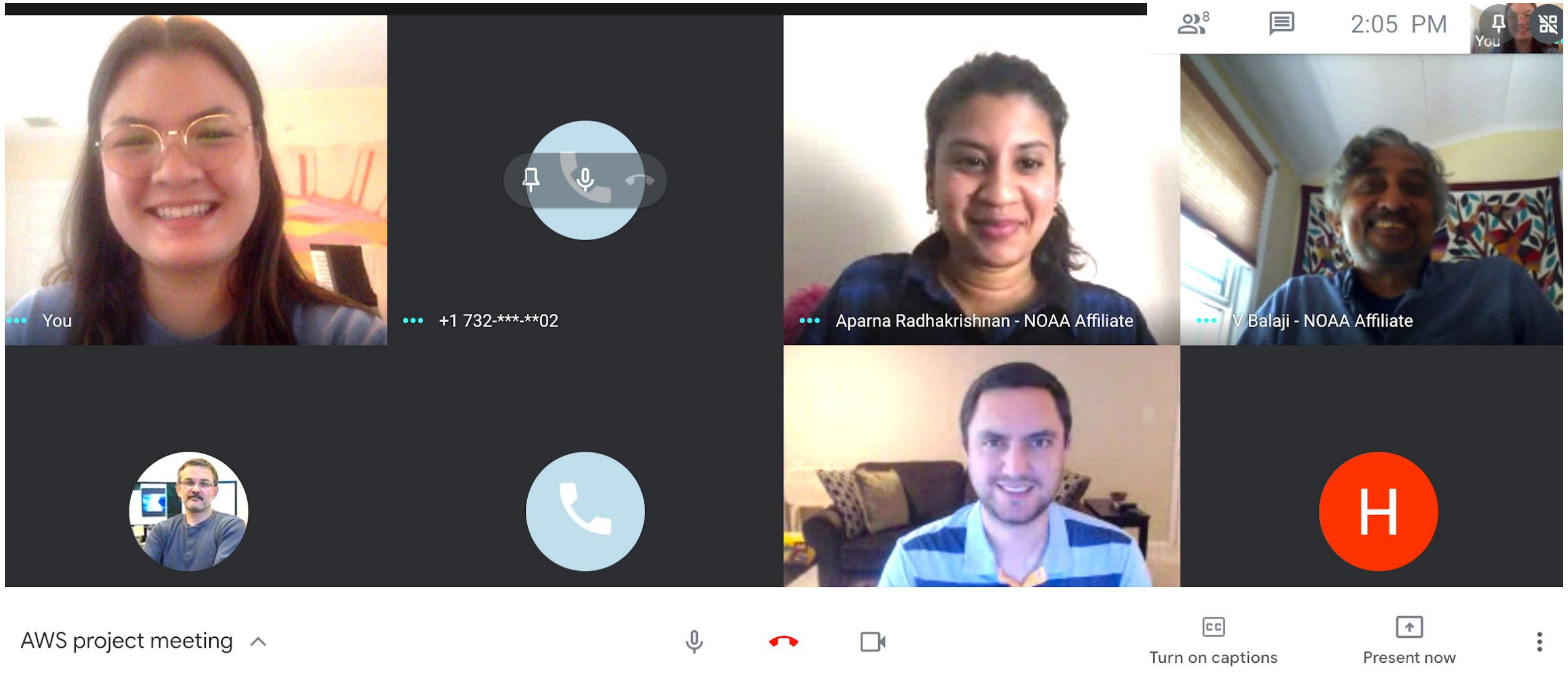
NOAA
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
U.S. DEPARTMENT OF COMMERCE

Special Thanks To:

V. Balaji, Zac Flamig, Serguei Nikonov, Kristopher Rand, Hans Vahlenkamp

Earth System Grid Federation

ASDI initiative from Amazon



Our team

Outline

- Data publication
  - AWS storage - S3
  - Cross Account Access
- Cloud Computing
  - JupyterHub Installation
  - Kubernetes
  - Dask
- Data Analysis Use Cases
  - CMIP6 Analysis Script
  - Speed Comparisons with Dask

Publishing Data to the Cloud

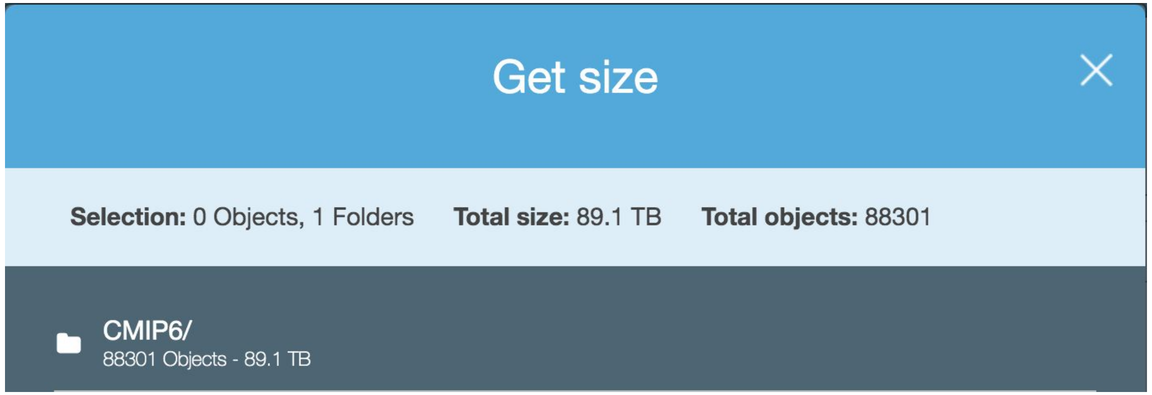
CMIP6 Data

- We have moved approximately 70 terabytes of GFDL data onto the cloud using Amazon Web Services
- AWS will soon announce the availability of this resource to the public, drawing inspiration from Pangeo’s open source big data climate analysis model
- The data is primarily CMIP6 data, a large scale, internationally unified data store designed to help model and understand the Earth system

The World Climate Research Programme's
Coupled Model Intercomparison Project

AWS Simple Storage Service (S3)

- Amazon Web Services is a more cost efficient and faster alternative to physical data storage using large computers
- We’ve enabled intelligent tiering, so the data is automatically stored in different locations based on how often they’re accessed
- Centralises computing and storage in one place on AWS



amazon S3

Cross Account Access

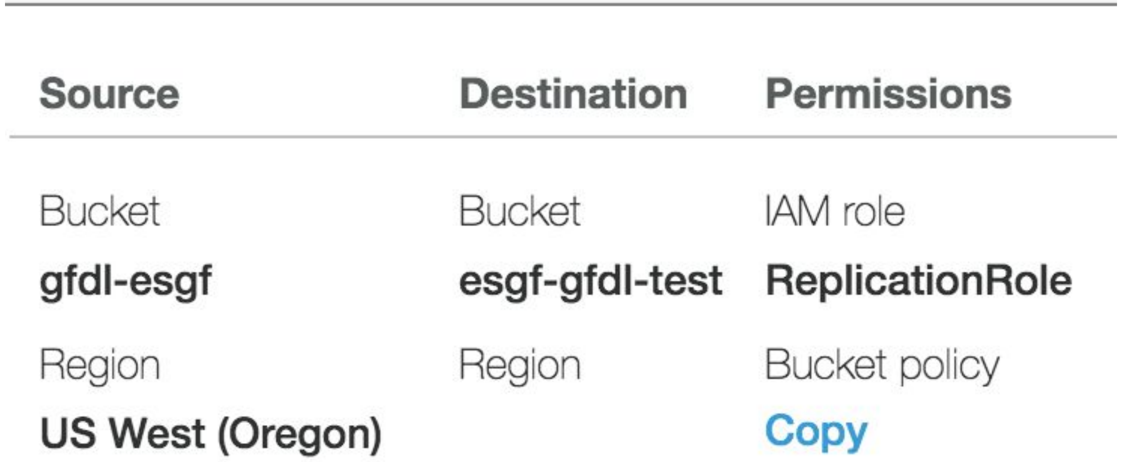
- We have 2 AWS awards from the Amazon Sustainability Data Initiative
- These awards are separated into two AWS accounts, one for analysis and one for data storage
- This created a need to provide access between the accounts

Photo Credit: ASDI

Options explored

1. AWS allows automatic replication which copies the data automatically to the new account as it is uploaded
   a. inefficient to store the data in two places
2. Role switching is another approach, which allows you to assume the role and permissions of another account
   a. This is cumbersome when it has to be done often

Role Switching Console Photo Credit: AWS

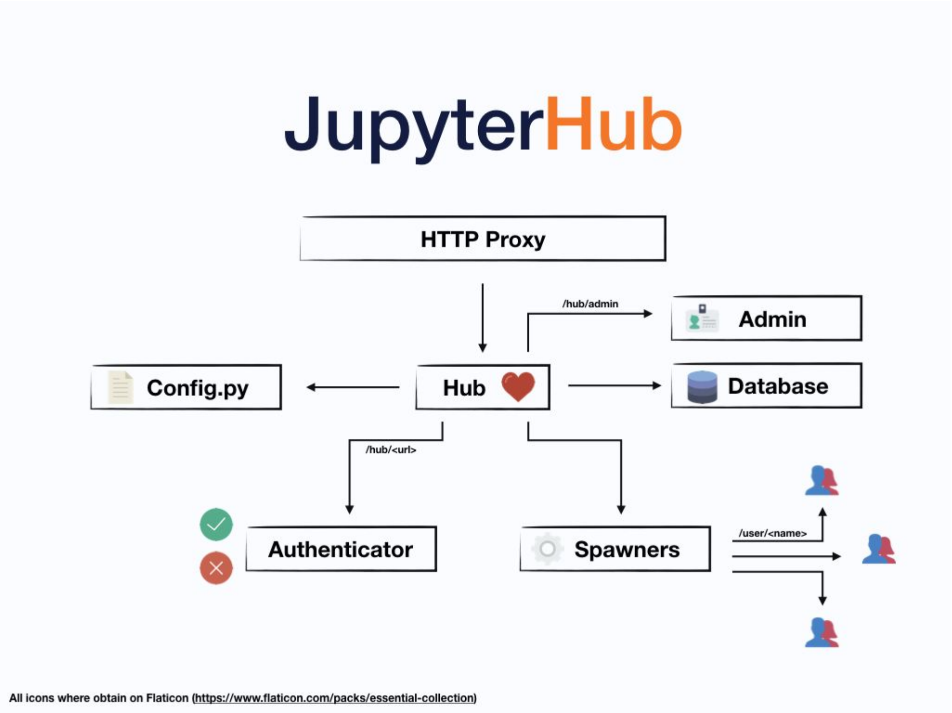


Replication Policy Photo Credit: AWS S3

Cross Account Permissions

- Working with members of the GFDL AWS team, we finally settled on granting cross account permissions
- This essentially gives users in the second account the credentials to access and change bucket in the data account with the same permissions as the original account
- Documenting the above efforts was one of the accomplishments since cross-account access is key to using AWS services, now and in the future.

```
~> aws s3 ls s3://gfdl-esgf/CMIP6/ --profile 4489
                           PRE AerChemMIP/
                           PRE C4MIP/
                           PRE CDRMIP/
                           PRE CFMIP/
                           PRE CMIP/
                           PRE DAMIP/
                           PRE FAFMIP/
                           PRE GMMIP/
                           PRE HighResMIP/
                           PRE LUMIP/
                           PRE OMIP/
                           PRE RFMIP/
                           PRE ScenarioMIP/
2020-06-16 17:00:50          0
```



JupyterHub Architecture Photo Credit: JupyterHub

Enabling Cloud Computing

What is JupyterHub?

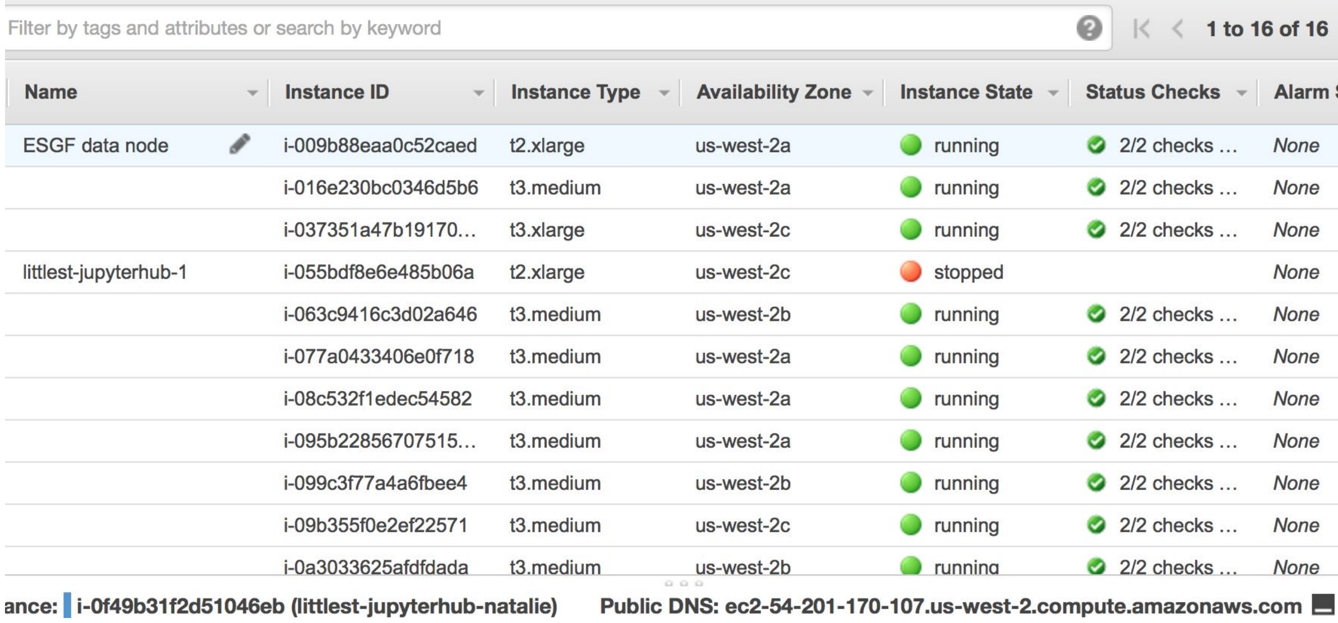
- JupyterHub is an online resource that can be used to write and execute code in different environments, facilitating faster analysis
- ideal for executing scripts on large amounts of data stored in the cloud, – no data or computation is taking up physical space
- Lets scientists spend less time on setting up the infrastructure for data analysis and provides documentation to reproduce the analysis.

“Bring analysis to data”(6)

Amazon Elastic Compute Cloud - EC2

Amazon
EC2

- EC2 provides compute capacity on the cloud, so computation is not using up local resources
- Its buildable nature makes it highly scalable
- One EC2 instance can give you the power to launch a simple JupyterHub
- With the combined power of many EC2 instances you can really speed workflow



Filter by tags and attributes or search by keyword — 1 to 16 of 16

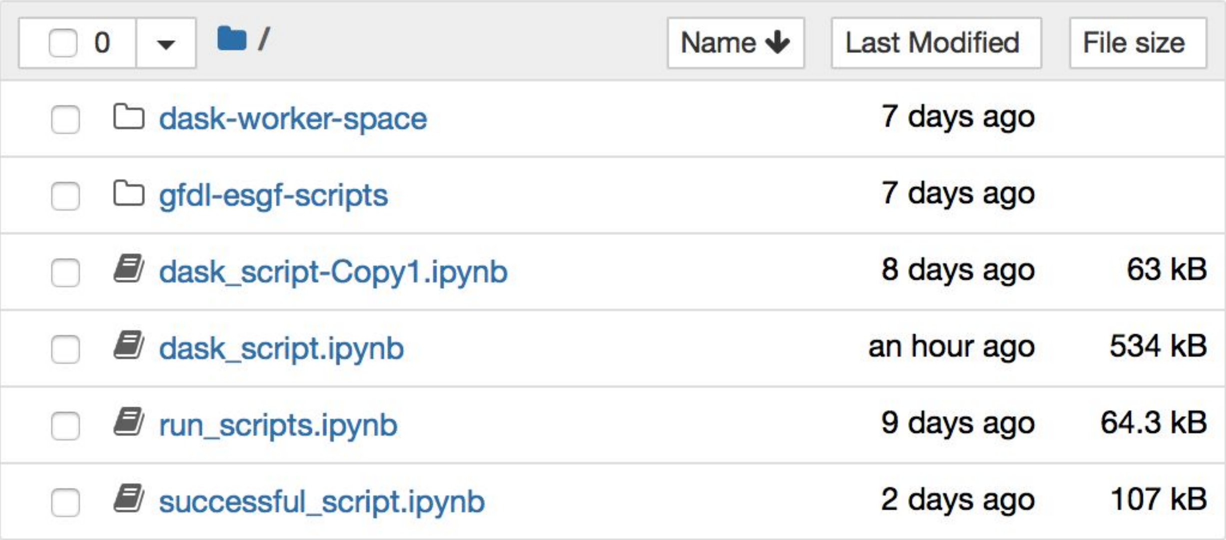
Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm
ESGF data node	i-009b88eaa0c52caed	t2.xlarge	us-west-2a	running	2/2 checks ...	None
	i-016e230bc0346d5b6	t3.medium	us-west-2a	running	2/2 checks ...	None
	i-037351a47b19170...	t3.xlarge	us-west-2c	running	2/2 checks ...	None
littlest-jupyterhub-1	i-055bdf8e6e485b06a	t2.xlarge	us-west-2c	stopped		None
	i-063c9416c3d02a646	t3.medium	us-west-2b	running	2/2 checks ...	None
	i-077a0433406e0f718	t3.medium	us-west-2a	running	2/2 checks ...	None
	i-08c532f1edec54582	t3.medium	us-west-2a	running	2/2 checks ...	None
	i-095b22856707515...	t3.medium	us-west-2a	running	2/2 checks ...	None
	i-099c3f77a4a6fbee4	t3.medium	us-west-2b	running	2/2 checks ...	None
	i-09b355f0e2ef22571	t3.medium	us-west-2c	running	2/2 checks ...	None
	i-0a3033625afdfdada	t3.medium	us-west-2b	running	2/2 checks ...	None

ance: i-0f49b31f2d51046eb (littlest-jupyterhub-natalie) Public DNS: ec2-54-201-170-107.us-west-2.compute.amazonaws.com

EC2 Instances Photo Credit: AWS EC2

Littlest JupyterHub

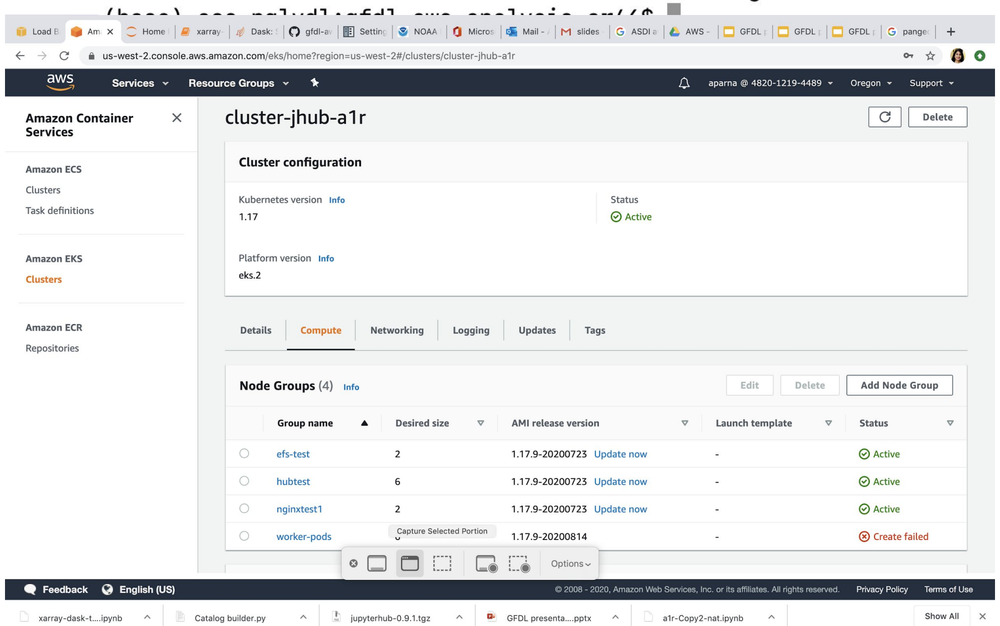
- The littlest JupyterHub was our first sojourn into cloud computing
- It is can be installed directly through the creation of an Amazon EC2 Instance which gives it compute power
- The Littlest JupyterHub allowed us to run analysis scripts in the cloud, but was ultimately too slow with its limited compute power



JupyterHub and Kubernetes

- Kubernetes is a container orchestration system designed to facilitate containers and virtual machines for an application
- Kubernetes cluster in conjunction with JupyterHub can be configured to use multiple EC2 instances, which give a much larger compute power than our first attempt

```
(base) aos-pqlvdl:gfdl-aws-analysis ar46$ kubectl --namespace jhub get pods
NAME                                  READY   STATUS    RESTARTS   AGE
continuous-image-puller-58x82         1/1     Running   0          10d
continuous-image-puller-6vwg4         1/1     Running   0          10d
continuous-image-puller-8h6gm         1/1     Running   0          28h
continuous-image-puller-j4qhl         1/1     Running   0          10d
continuous-image-puller-m2r2x         1/1     Running   0          27h
continuous-image-puller-m9s65         1/1     Running   0          28h
continuous-image-puller-mvjft         1/1     Running   0          7d23h
continuous-image-puller-r97mw         1/1     Running   0          10d
continuous-image-puller-rwlb2         1/1     Running   0          28h
continuous-image-puller-xh9zl         1/1     Running   0          28h
continuous-image-puller-xvsq5         1/1     Running   0          7d23h
hub-dfbc895cf-xrp5z                   1/1     Running   0          10d
jupyter-jupyter-5fadmin               1/1     Running   0          97m
proxy-75d76c4c64-bhs94                1/1     Running   0          10d
user-scheduler-5d97fdd64f-dxd6n       1/1     Running   0          10d
user-scheduler-5d97fdd64f-sm2rf       1/1     Running   1          10d
```




```
(base) aos-pqlvdl:gfdl-aws-analysis ar46$ helm list
NAME           REVISION   UPDATED                    STATUS     CHART            APP VERSION   NAMESPACE
appt-release   1          Tue Aug 18 14:23:55 2020   DEPLOYED   dask-4.3.1       2.23.0        default
jhub           1          Sun Aug  9 00:43:22 2020   DEPLOYED   jupyterhub-0.9.0 1.1.0         jhub
(base) aos-pqlvdl:gfdl-aws-analysis ar46$
```

Amazon Elastic Kubernetes Service - EKS

- Amazon has a built in way to run Kubernetes services called EKS
- EKS can utilize multiple EC2 instances to create a cluster and provide compute power to multiple Kubernetes pods

Dask

- Allows computation work to be split between several workers
- These processes work asynchronously and concurrently, so it can increase analysis speed dramatically
- Provides a comprehensive dashboard to interpret and understand what each worker is doing

Dask Dashboard

Executing Analysis Scripts

Sample analysis used in testing time-average plot of sea_water_potential_temperature/thetao

Code adapted from Modular Ocean Model:

https://mom6-analysiscookbook.readthedocs.io/en/latest/01a_setting_up_dask_jobqueue.html

https://github.com/aradhakrishnanGFDL/enes2020

Data Acknowledgment:

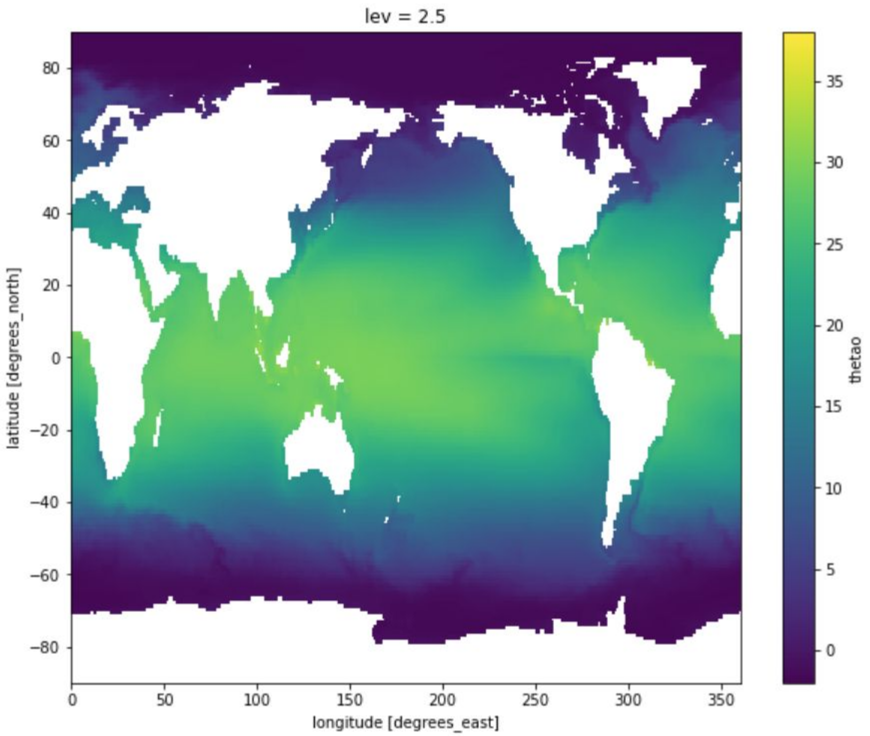
Adcroft, Alistair; Blanton, Chris; McHugh, Colleen; Nikonov, Serguei; Radhakrishnan, Aparna; Zadeh, Niki T.; Anderson, Whit; Bushuk, Mitchell; Dufour, Carolina O; Dunne, John P.; Griffies, Stephen M.; Hallberg, Robert; Harrison, Matthew; Held, Isaac M; Jansen, Malte F; John, Jasmin G; Krasting, John P.; Langenhorst, Amy; Legg, Sony; Liang, Zhi; Reichl, Brandon G; Rosati, Anthony; Samuels, Bonita L.; Shao, Andrew; Stouffer, Ronald; Winton, Michael; Wittenberg, Andrew T.; Xiang, Baoqiang; Zhang, Rong (2018). NOAA-GFDL **GFDL-CM4 model output prepared for CMIP6** OMIP. Earth System Grid Federation. https://doi.org/10.22033/ESGF/CMIP6.1403

Pangeo https://pangeo.io/

```
CPU times: user 7min 5s, sys: 2min 41s, total: 9min 46s
Wall time: 24min 49s
```

```
/opt/conda/lib/python3.7/site-packages/dask/array/numpy_compat.py:41: Runtim
ue_divide
  x = np.divide(x1, x2, out)
```

```
<matplotlib.collections.QuadMesh at 0x7eff18f5cdd0>
```



Speed Comparison

- Using JupyterHub to run the analysis test without Dask using Littlest JupyterHub takes 25 minutes
- With Elastic Kubernetes Cluster EKS and Dask, the work is distributed between 4 workers and shows a performance increase by a factor of 5
- We could use many more than just 4 workers to accelerate computation

```
CPU times: user 7min 55s, sys: 1min 21s, total: 9min 16s
Wall time: 25min 45s
```

```
CPU times: user 1.79 s, sys: 64.1 ms, total: 1.86 s
Wall time: 5min 8s
```

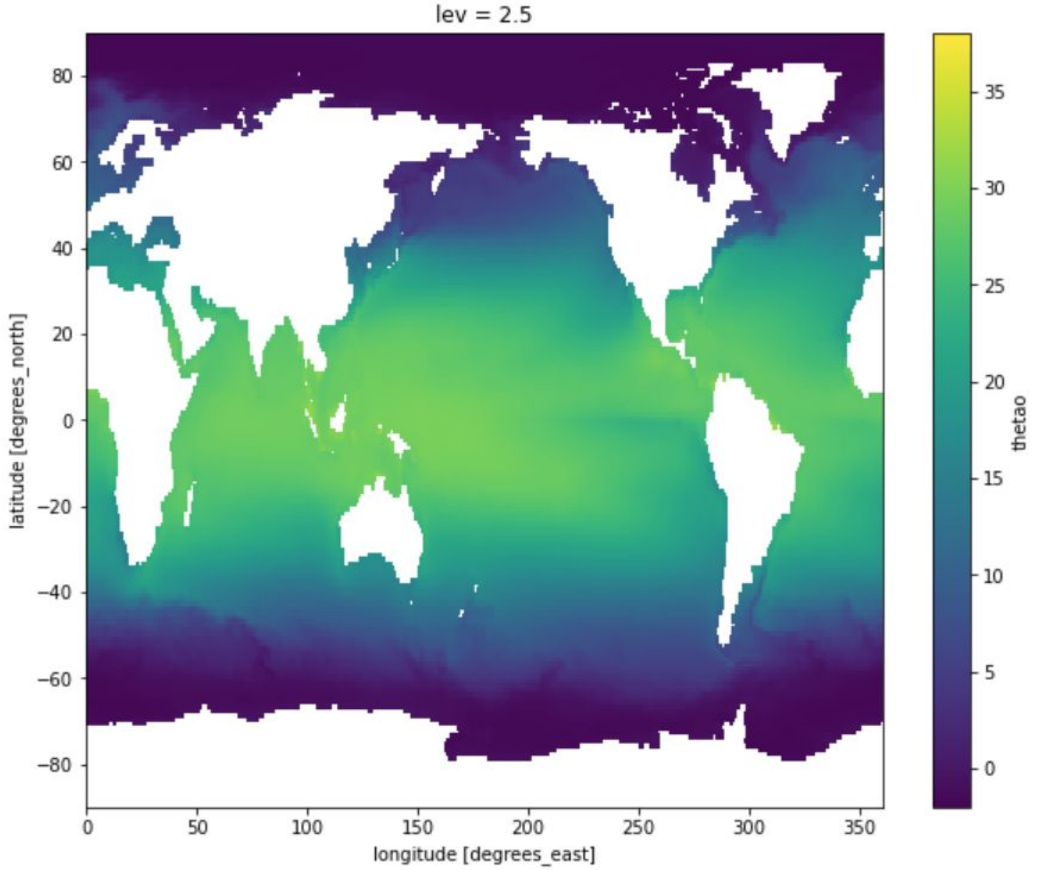
https://github.com/aradhakrishnanGFDL/gfdl-aws-analysis/blob/master/xarray-dask-testing-on-EKS-JupyterHub.ipynb

Side By Side Comparison

```
CPU times: user 7min 5s, sys: 2min 41s, total: 9min 46s
Wall time: 24min 49s
```

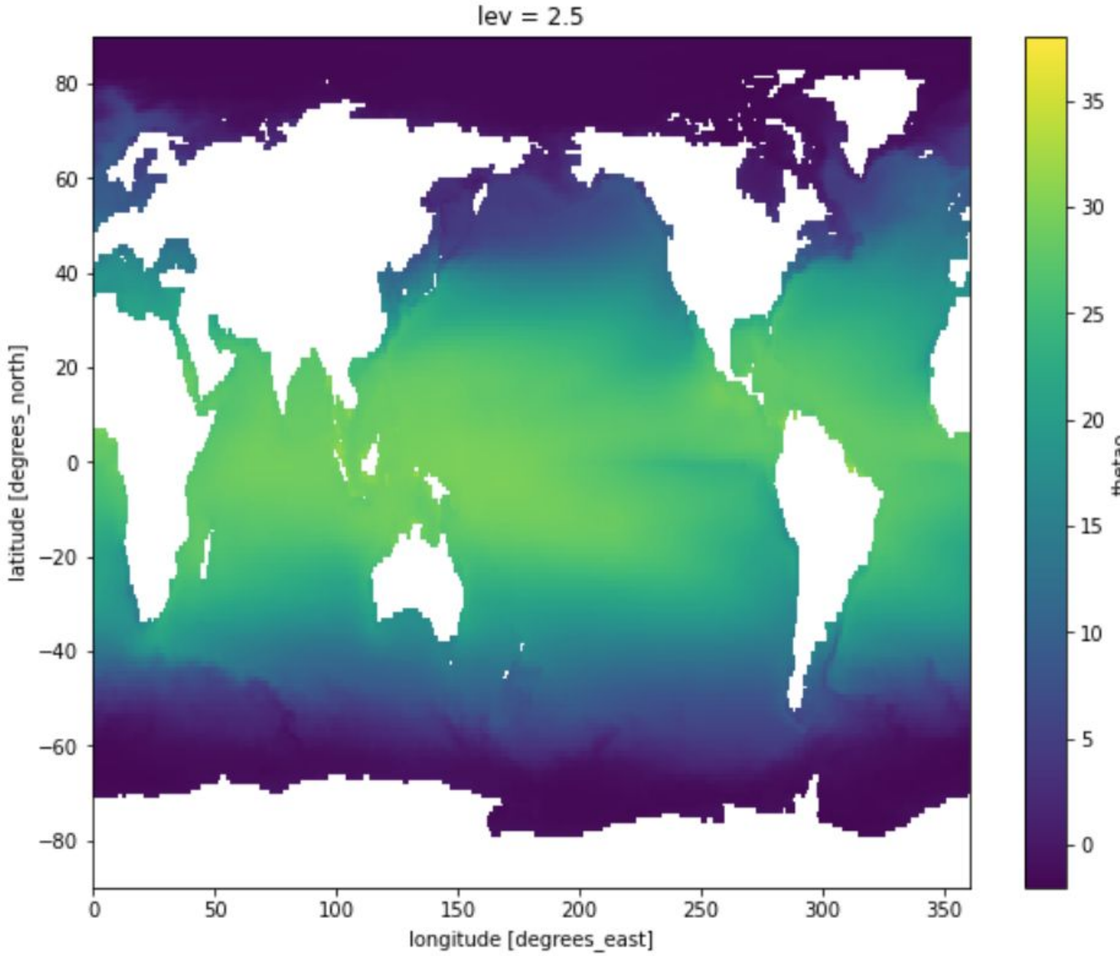
```
/opt/conda/lib/python3.7/site-packages/dask/array/numpy_compat.py:41: Runtim
ue_divide
  x = np.divide(x1, x2, out)
```

```
: <matplotlib.collections.QuadMesh at 0x7eff18f5cdd0>
```

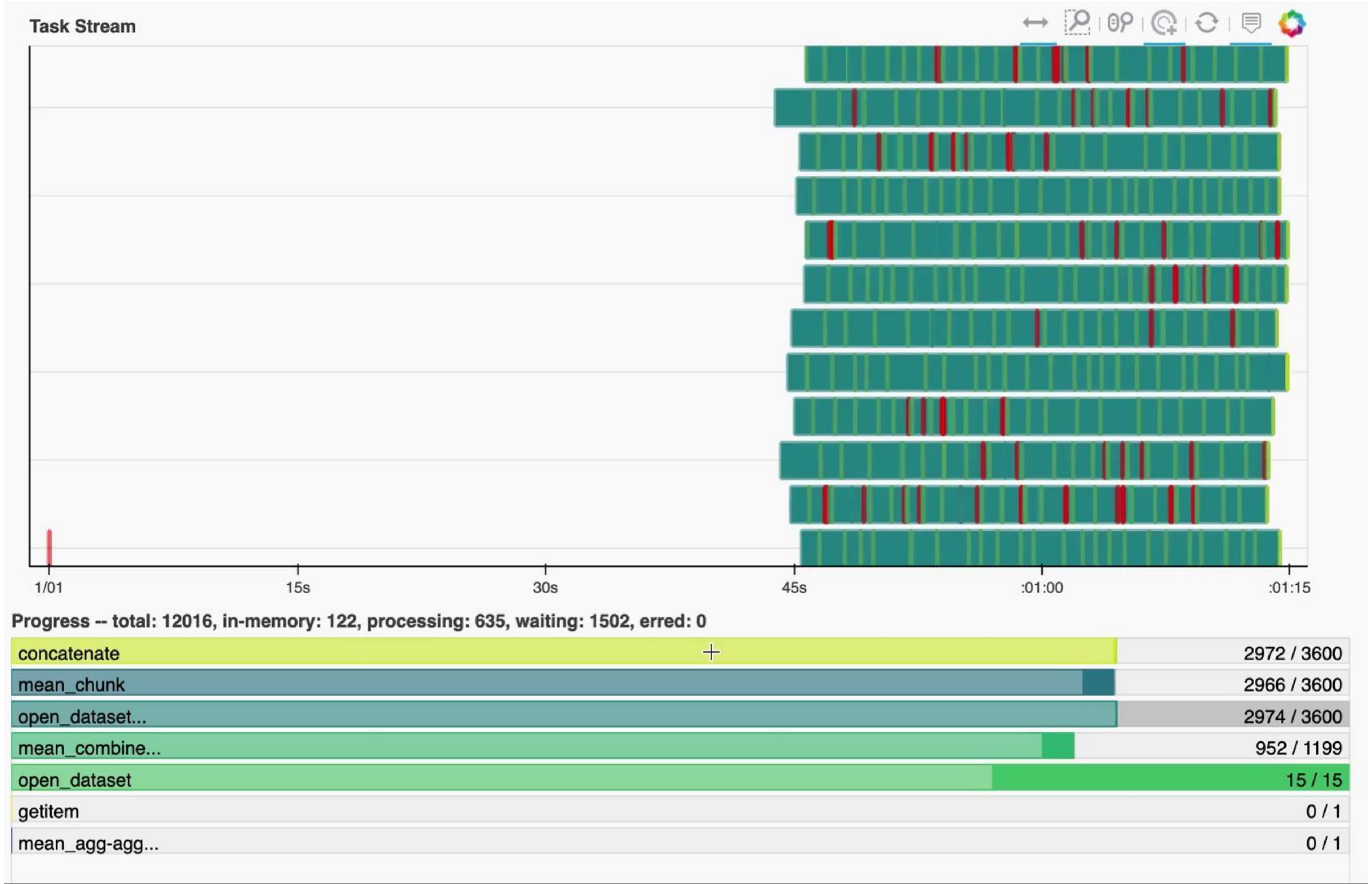



```
CPU times: user 1.77 s, sys: 53.2 ms, total: 1.82 s
Wall time: 5min 46s
```

```
Out[7]: <matplotlib.collections.QuadMesh at 0x7fd5076a5390>
```



Demo


Task Stream
1/01
15s
30s
45s
:01:00
:01:15
Progress -- total: 12016, in-memory: 122, processing: 635, waiting: 1502, erred: 0
concatenate
2972 / 3600
mean_chunk
2966 / 3600
open_dataset...
2974 / 3600
mean_combine...
952 / 1199
open_dataset
15 / 15
getitem
0 / 1
mean_agg-agg...
0 / 1

What next?

1. Public announcement of CMIP6 data in the AWS Cloud (S3 buckets) upon final testing
2. Deployment of ESGF data publication containerized workflow in Amazon cloud.
3. Federation of cloud-based ESGF node with the global network of nodes
4. Further performance testing of JupyterHub and exploring other cloud-optimized data formats in S3, flexible and persistent user environment configurations, securing the services, opening it for user testing
5. Technical and User Documentation

Bibliography

1. https://mom6-analysiscookbook.readthedocs.io/en/latest/01b_setting_up_dask_localcluster.html
2. https://mom6-analysiscookbook.readthedocs.io/en/latest/01a_setting_up_dask_jobqueue.html
3. https://github.com/aradhakrishnanGFDL/gfdl-aws-analysis/blob/master/xarray-dask-testing-on-EKS-JupyterHub.ipynb
4. Adcroft, Alistair; Blanton, Chris; McHugh, Colleen; Nikonov, Serguei; Radhakrishnan, Aparna; Zadeh, Niki T.; Anderson, Whit; Bushuk, Mitchell; Dufour, Carolina O; Dunne, John P.; Griffies, Stephen M.; Hallberg, Robert; Harrison, Matthew; Held, Isaac M; Jansen, Malte F; John, Jasmin G; Krasting, John P.; Langenhorst, Amy; Legg, Sony; Liang, Zhi; Reichl, Brandon G; Rosati, Anthony; Samuels, Bonita L.; Shao, Andrew; Stouffer, Ronald; Winton, Michael; Wittenberg, Andrew T.; Xiang, Baoqiang; Zhang, Rong (2018). NOAA-GFDL **GFDL-CM4 model output prepared for CMIP6** OMIP. Earth System Grid Federation. https://doi.org/10.22033/ESGF/CMIP6.1403
5. Pangeo https://pangeo.io/
6. Deploying user-developed scientific analyses on federated data archives V Balaji, S Ansari, A Radhakrishnan AGUFM 2011, IN41C-01, 2011

   From" Inspiration-driven" Research to" Industrial-strength" Research: Applying User-developed Climate Analytics at Large scale A Radhakrishnan, EE Mason, AR Langenhorst, V Balaji, S Nikonov AGUFM 2014, IN53E-04, 2014