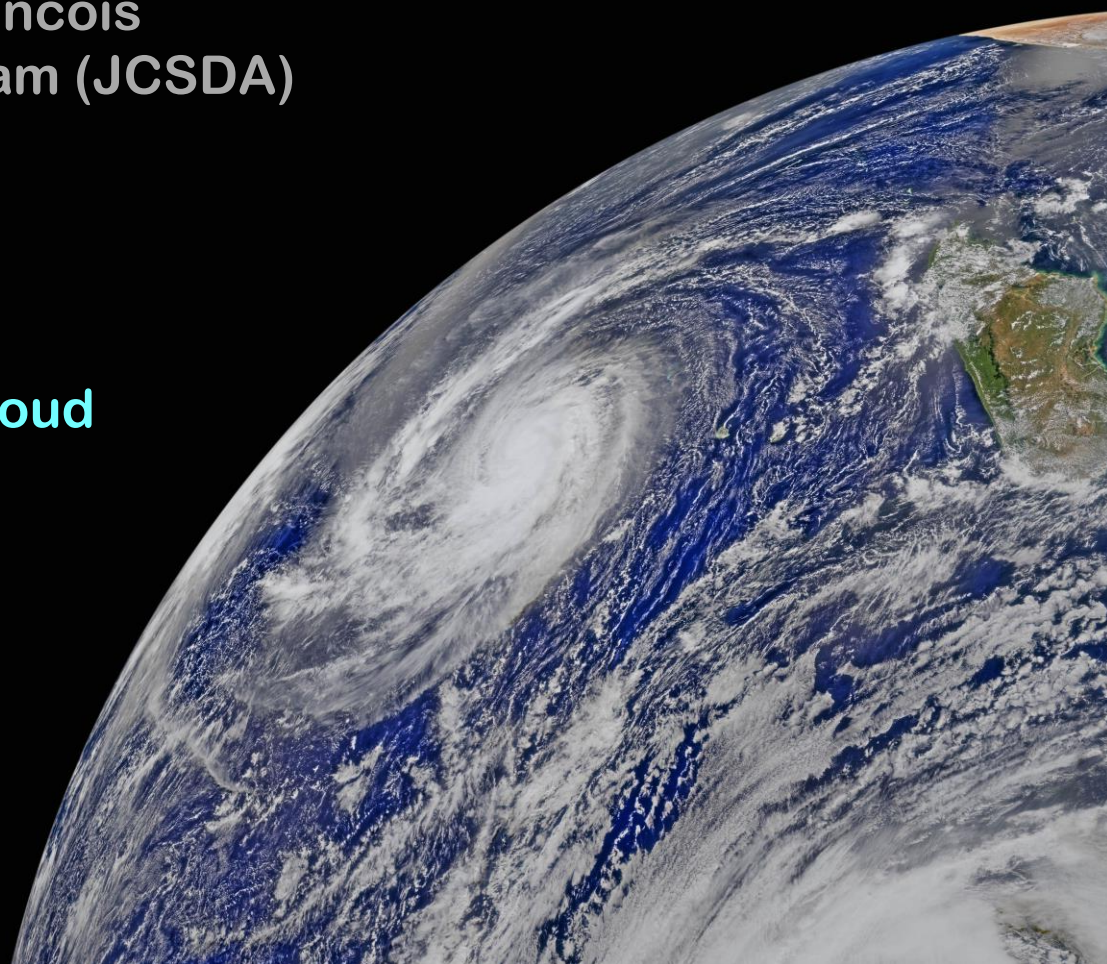
# JEDI in the Cloud

Mark Miesch, Maryam Abdi-Oskouei, Tom Auligne, David Hahn, Dan Holdaway, Steve Herbener, Mark Olah, Jim Rosinski, Travis Sluka, Yannick Tremolet, Francois Vandenberghe + the JEDI team (JCSDA)



**Virtual Workshop:**
**Weather and Climate in the Cloud**
**ECMWF**
**8-10 Feb, 2021**

U.S. AIR FORCE

# Outline

I) JEDI Overview

II) JEDI Development
- ✦Continuous Integration with AWS

III) HPC IaaS
- ✦AWS computing environment
- ✦JEDI container/cloud benchmarking
- ✦JEDI Academies / Training

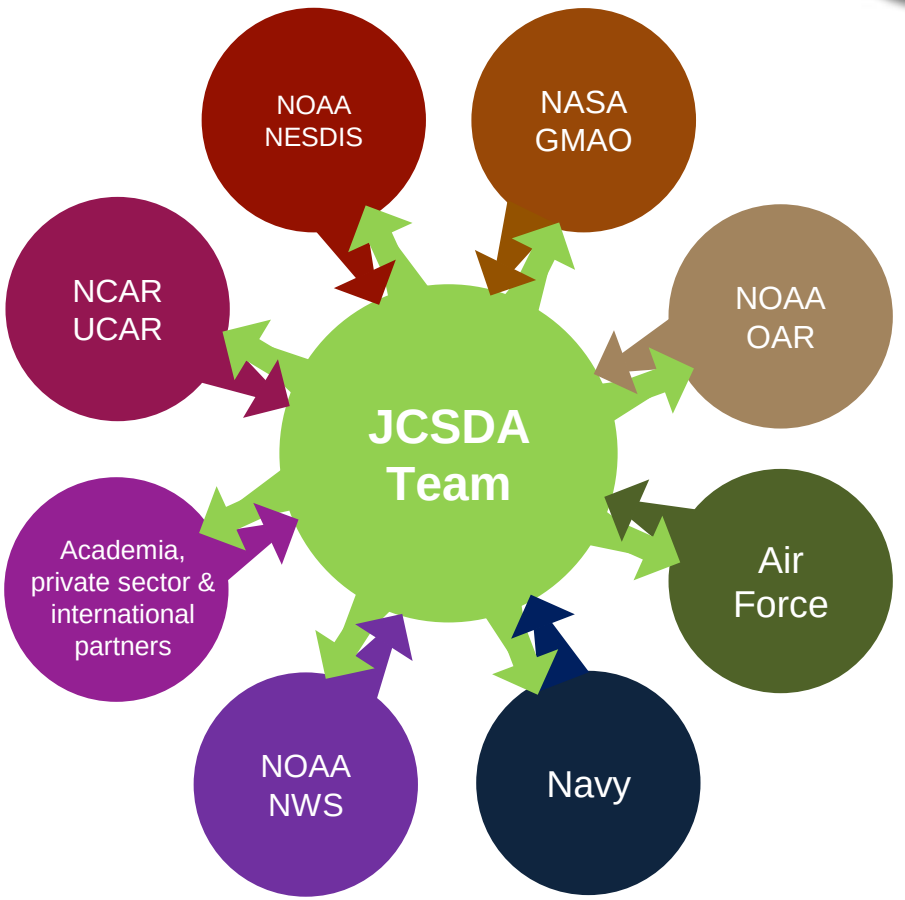
IV) Web Applications
- ✦NRT Observation monitoring

V) Summary and Outlook
- ✦Broader JCSDA context

# Joint Center for Satellite Data Assimilation

**JCSDA**

**A multi-agency research center created to improve the use of satellite data for analyzing and predicting the weather, the ocean, the climate and the environment.**




**WHO**
Distributed staff

**HOW**
Joint operating plan

**WHAT**
Critical path to operations

# JEDI

FV3 (GFS+GEOS)
(NOAA/NASA)

MPAS
(NCAR)

UM & LFRic
(UKMO)

SOCA
(JCSDA)

Toy models
(Lorenz 95, QG,
shallow water)

...

## Joint Effort for Data assimilation Integration

**JEDI**

**First Release**
**Oct, 2020**

A Next-Generation
Unified
DA System

Radiosondes

Radiance

Aircraft

Aerosols

GNSSRO

...

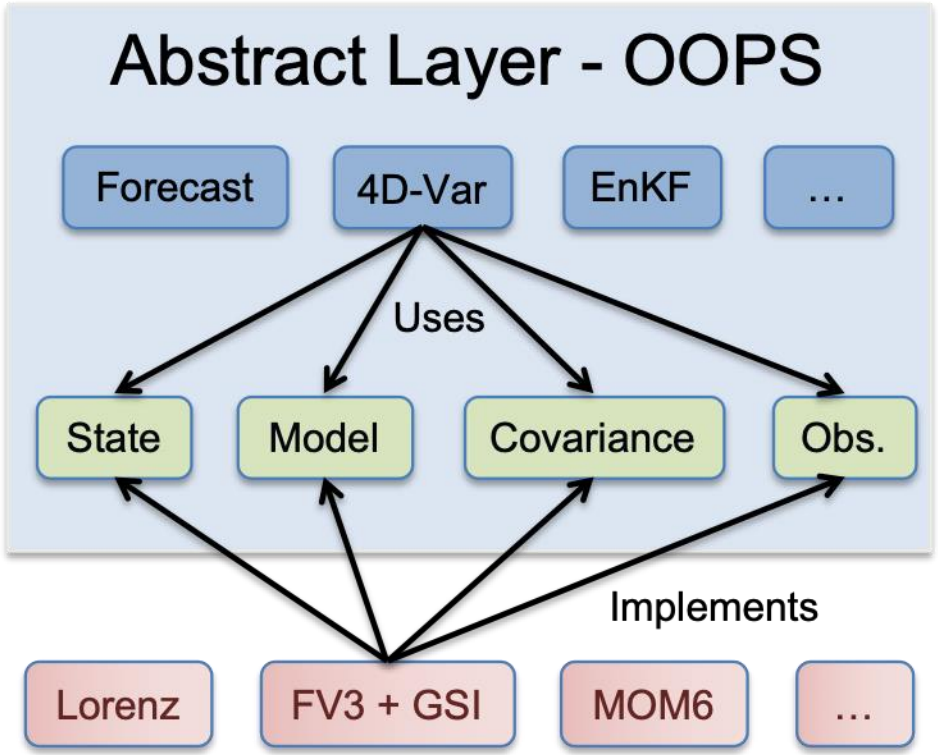
# JEDI

## Motivations

- Reduce duplication of effort among partners
  - New obs, DA algorithms
- Unified
  - same DA algorithms for atmosphere, ocean, coupled & toy models
- Promote R2O/O2R

## Approach

- Exploit object-oriented & generic programming
- Separation of concerns
- Agile, collaborative development environment




- C++ (high-level)
- Fortran (low-level)

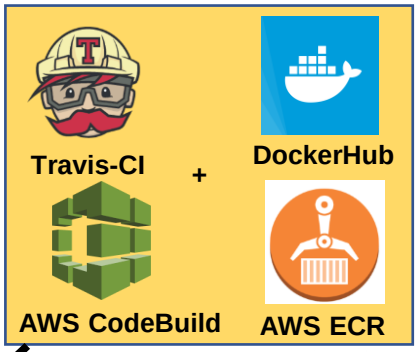
# II: JEDI Development: CI with AWS

**Automated testing tools**

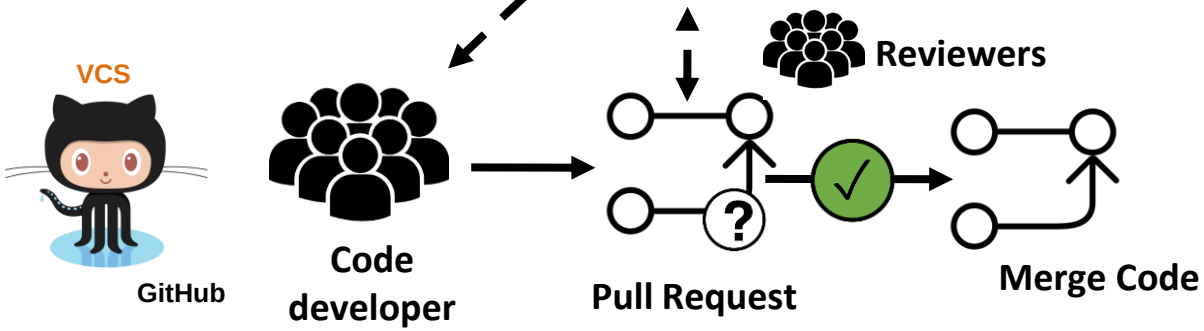
**Containers**

**Each PR tested with multiple compiler/mpi combinations**

- GNU / OpenMPI
- Clang / MPICH
- Intel / IMPI




- **AWS CodeBuild:**
  - scales easily with computational needs
  - run multiple configurations in parallel using **containers**




**Web hooks trigger CI pipelines, sync private & public repos, enforce branch naming conventions**

**CI also checks code coverage**

**AWS CodePipeline for comprehensive downstream testing of dependent repos**

**Higher-tier application-level testing to be handled through nightly/weekly cron jobs**

# III: HPC IaaS

- **Single Development node**
  - ✦ For development, optimization
  - ✦ Easy to use launch script
  - ✦ Can terminate/stop from EC2 console
  - ✦ Custom AMI
  - ✦ gnu-openmpi, intel-impi environment modules
  - ✦ Docker, Singularity, Charliecloud

```
18:34 $ jedinode.py --help
Usage: jedinode.py [OPTIONS]

Options:
  --key TEXT            ssh key   [required]
  --type TEXT           Instance type (default c5.4xlarge)
  --ncores INTEGER      Number of cores (you can omit this for most instance
                        types)
  --securitygroup TEXT  Security group id (default is virginia-default)
  --region TEXT         Region (default is us-east-1)
  --spot                spot market (default is False)
  --maxprice TEXT       Max Price (defaults to on-demand price; only used if
                        spot is set)
  -h, --help            Show this message and exit.
```

# III: HPC IaaS

- **ParallelCluster**
  - For applications, optimization, testing
  - Autoscaling: cluster size adjusts on demand
  - EFS, FSx for lustre
  - Intel 19 compilers/mpi
  - gnu-openmpi, intel-impi stacks
  - AWS-provided AMI; security patches, latest hardware support
  - Post-install script: Singularity, git-lfs…
  - Spot pricing or on demand
  - VPC (public master, private compute nodes) with subnets in us-east-1c (best availablity)
  - Dynamic placement group for collocated resources

**Unified approach to facilitate maintenance:**
**Intel compilers and environment modules (gnu-openmpi, intel-impi) provided by means of an external volume that is auto-mounted at boot time**

# Supercontainers

**JEDI cloud applications can be built and run either with environment modules or with software containers**

## Container benefits

- Portability
- Reproducibility
  - Version control (git)
- Bring your own environment
- Efficiency / workflow
  - Develop on laptops, run on HPC/cloud
  - Get new users up and running quickly

## JEDI Supercontainers

- Singularity
  - Development containers
  - Application containers (multi-stage build with intel runtime libraries)
- Enhanced components
  - Infiniband drivers
  - PMI
  - UCX, KNEM, XPMEM

# Container/Cloud Benchmarking

**Benchmark** FV3-GFS JEDI 3DVar Application

- Resolution c192
- ~9 of 12 million obs pass QC
- Inner loop: 30 iterations
- Outer loop: 2 iterations
- 864 MPI tasks (12x12x6)

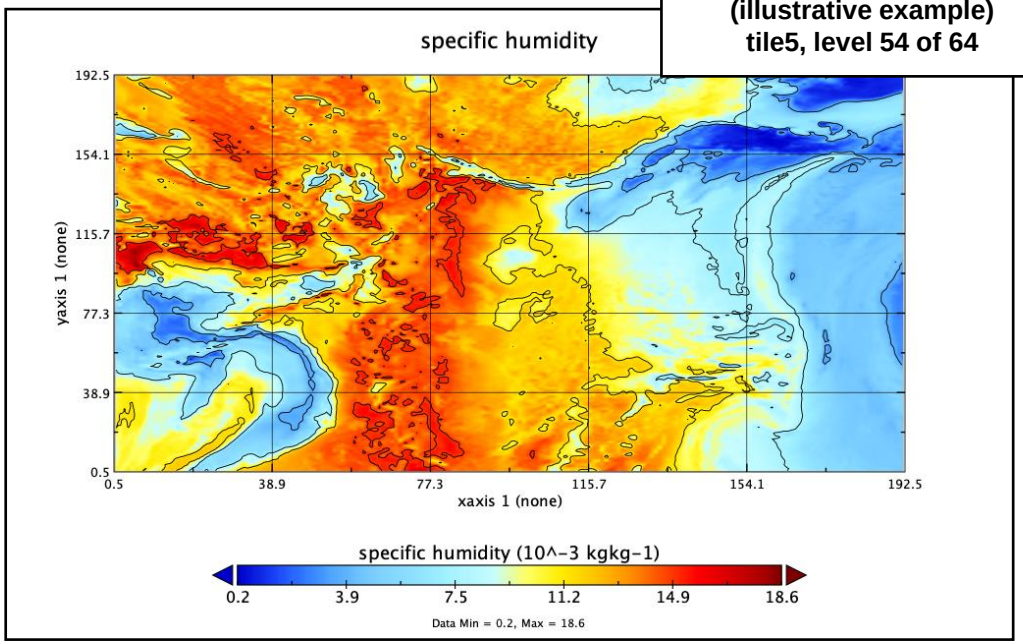
**~12 million obs**

Aircraft, Radiosonde, Rass, Satwind, Scatwind, Vadwind, AMSUA-NOAA19, AIRS-AQUA, IASI-METOPA, CRISFSR-NPP

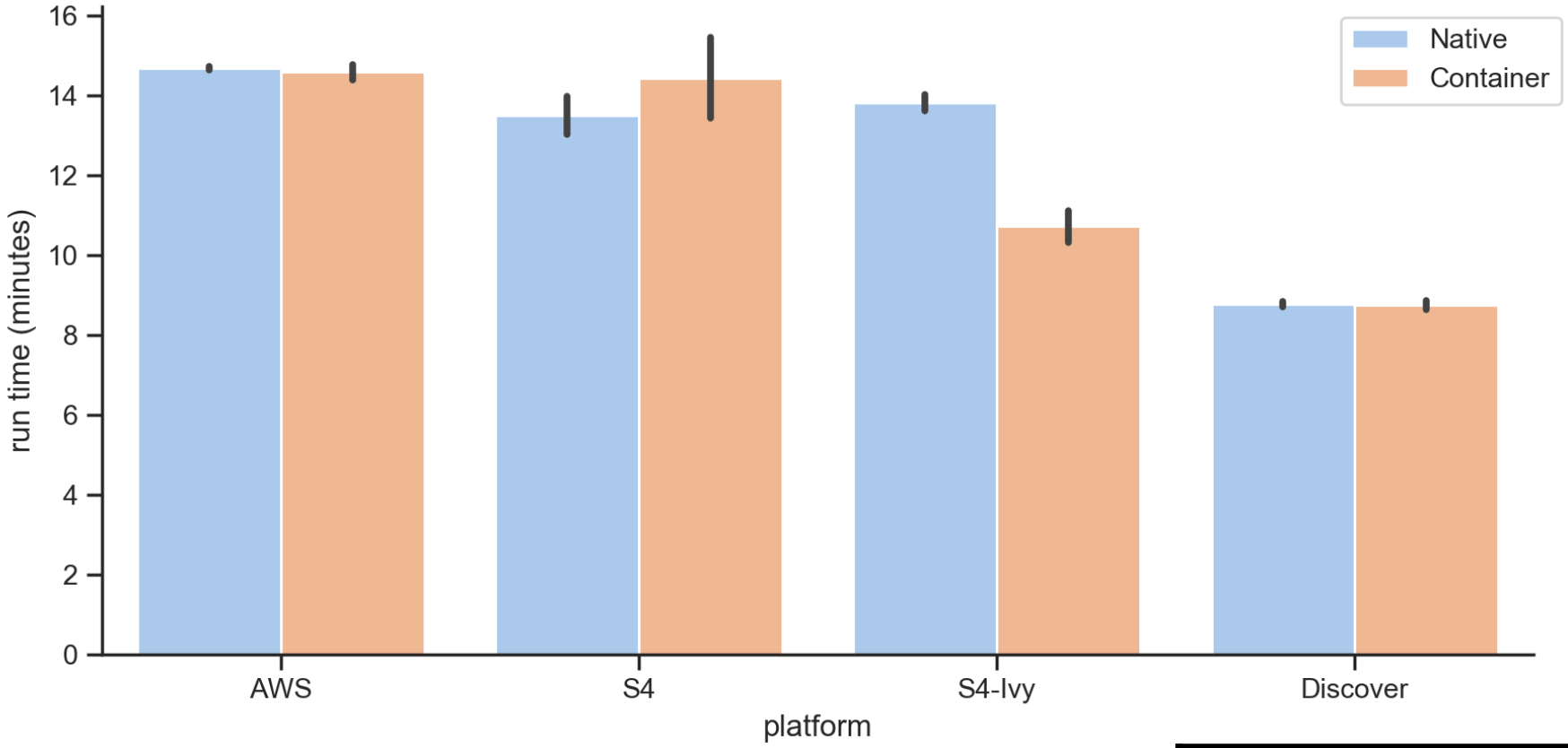
## Platforms

- Discover: NASA NCCS
- S4: SSEC/Univ. Wisconsin
- AWS
  - 24 c5n.18xlarge nodes
  - 36 cores/node
  - Elastic Fabric Adapter (EFA)

Analysis
(illustrative example)
tile5, level 54 of 64



# Container/Clloud Benchmarking




| Estimated AWS cost | |
|---|---|
| On demand | $23 |
| Spot | $7 |

**No overhead for running in the container**

# Container/Cloud Benchmarking

## Performance Tuning on AWS

**Mean run time on previous slide for AWS cluster:**
**14.59 min inside container, 14.69 min outside container**

- **~ 20 sec: FSx for Lustre instead of EBS**
- **~ 70 sec: MPI tuning**
- **~ 150 sec: mitigate memory fragmentation due to VM caching (sudo echo 3 > /proc/sys/vm/drop_caches)**

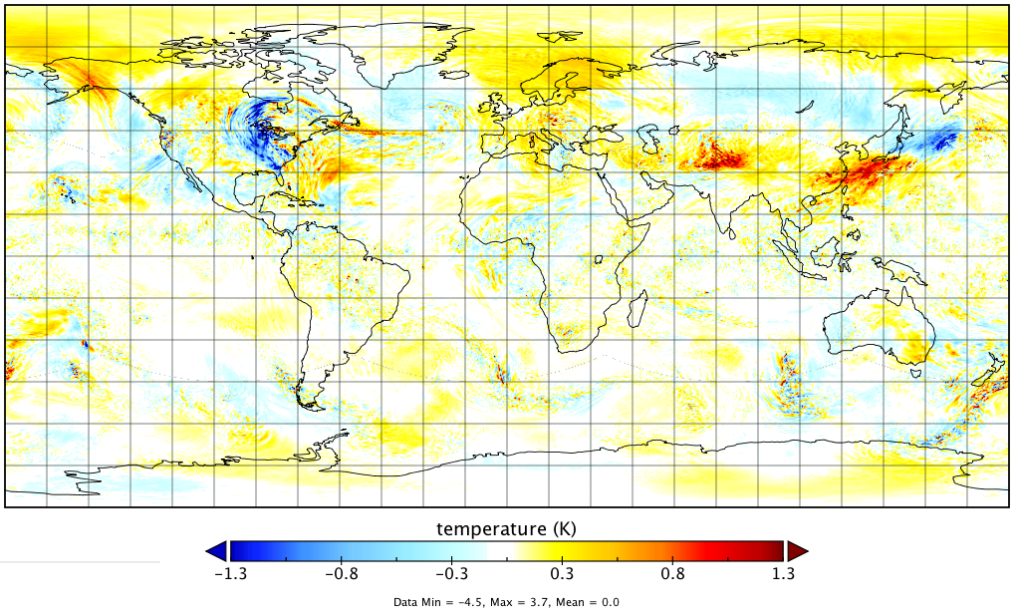
**Now running about:**
**~ 10.6 min outside container**

**May be possible to get it under 10 min**
**Recall Discover ~ 8.8 min**

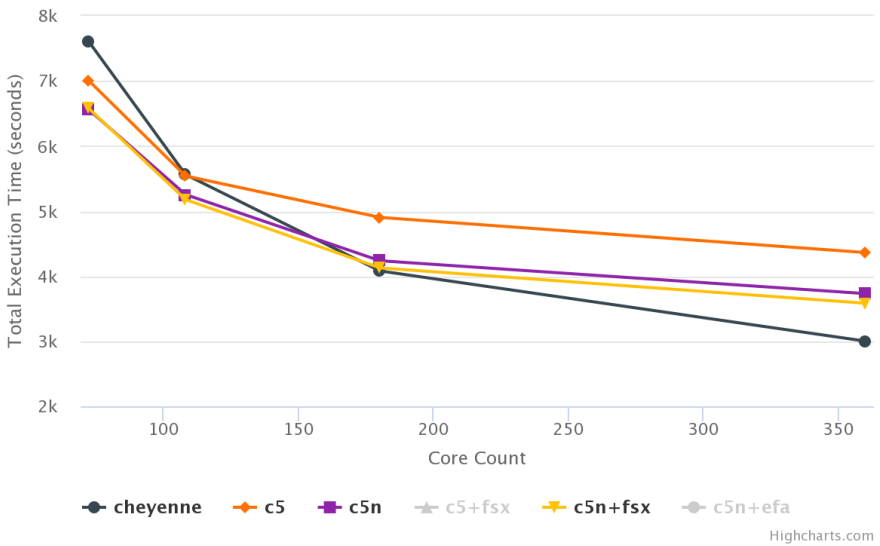
# Forecast Model Benchmarking

- UFS (pre-)operational C768 (13km) global configuration
- 10-day forecast with NEMSfv3gfs on 48 AWS c5.x18large nodes
  - 36 cores, 144GB mem, 25GB/s network, Intel 19 with iMPI
  - 1728 cores in total
- model run takes 7 min 22s real time per simulated day

JEDI temperature increment propagated with
24hr UFS forecast valid at 00z on April 16, 2018, near 200 hPa




Operational GFS v15




**Results prior to EFA (Early 2019)**

WRF Performance
on AWS (vs. Cheyenne)

# JEDI Training

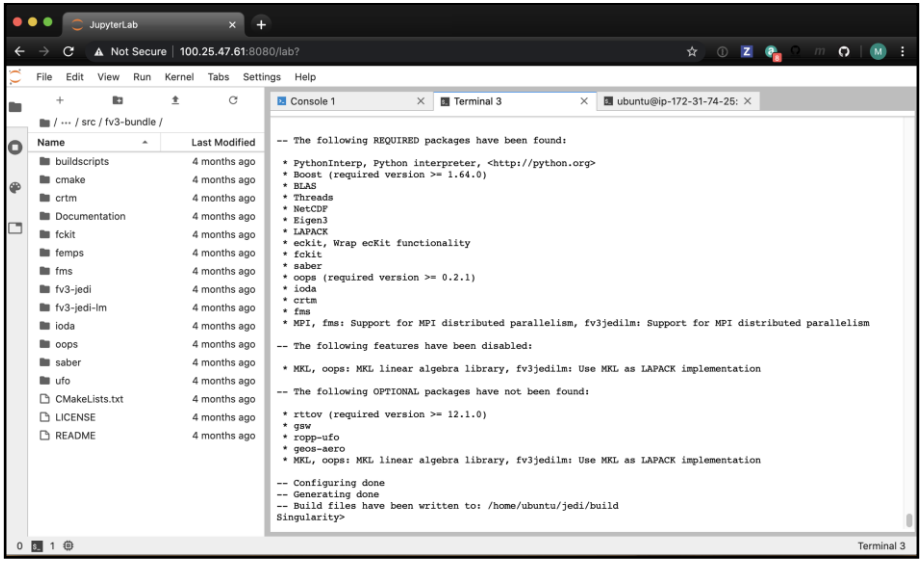
## JEDI Academy

- Biannual (disrupted by COVID)
- 4-5 days each, varying locations
- ~40 participants
- Mix of lectures and practicals
- Virtual in Nov-Dec 2020

Jupyterlab interface includes one or more ssh consoles, a python interpreter, a navigation panel, and an image viewer

## Activities use AWS

- Participants divided into 20-40 groups, each with their own AWS EC2 instance
- EC2 instances accessed via JupyterLab web interface
- Groups pull singularity container and build, run JEDI applications
- Estimated cost ~ $100/day




**Self-paced online tutorials also made available through a jcsda-tutorial public AMI and containers [https://jedi-docs.jcsda.org/en/latest/learning/tutorials/](https://jedi-docs.jcsda.org/en/latest/learning/tutorials/)**

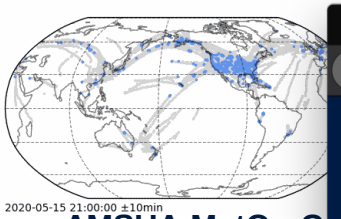
# IV) NRT Observation Monitoring


**http://nrt.jcsda.org**

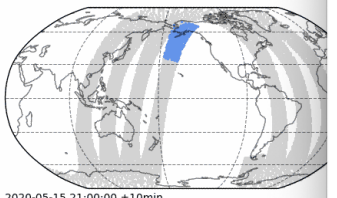
**Testbed for development & validation of observation operators**

# NRT Observation Monitoring

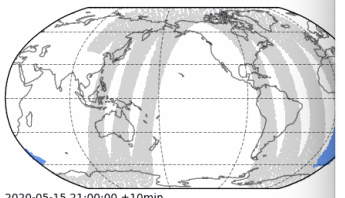
Aircraft | AIRS-Aqua | AMSUA Aqua | AMSUA MetOp-A | AMSUA MetOp-B



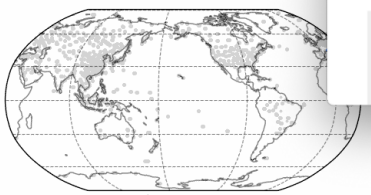

AMSUA MetOp-C

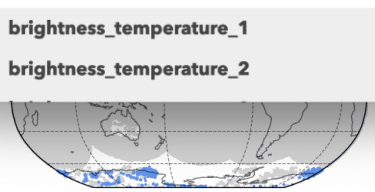



HIRS4 MetOp-B

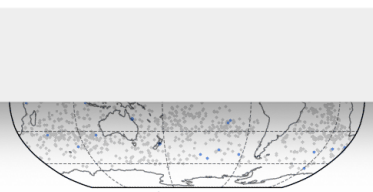



Radiosonde





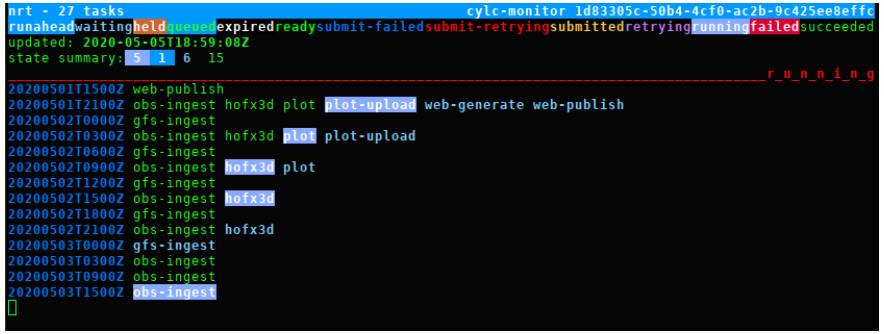




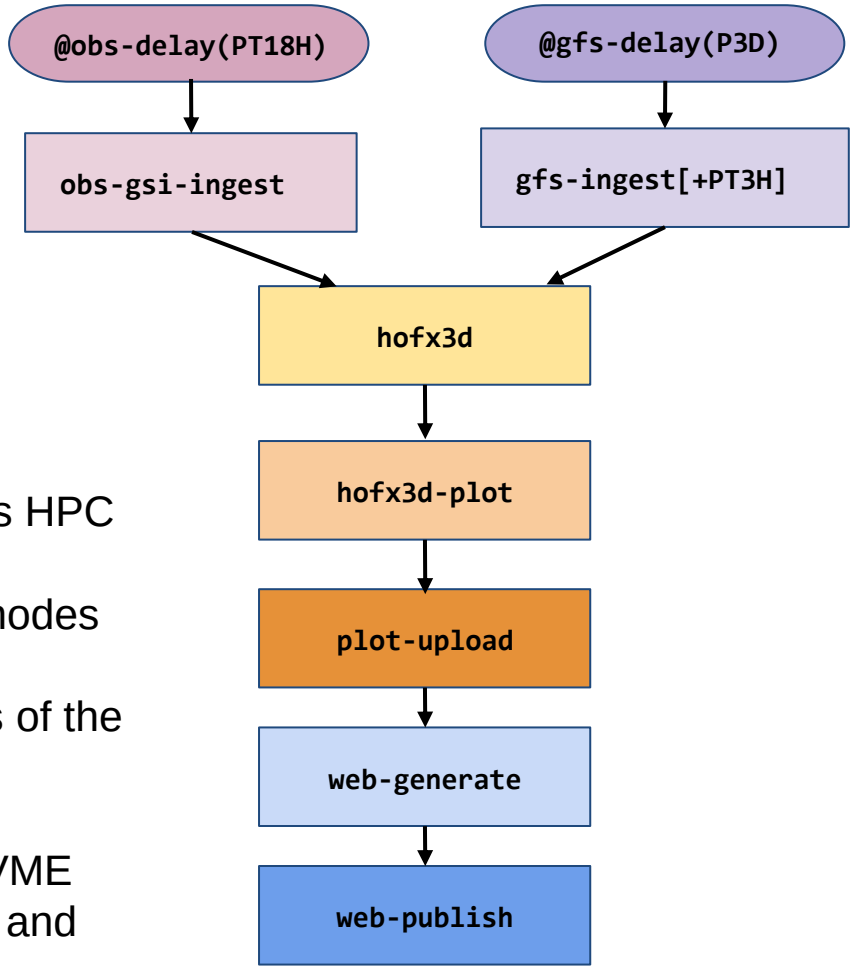


from public S3 bucket.

# JEDI NRT Unified Workflow




- JCSDA workflow software is portable across HPC and Cloud resources.
- Rapidly initialize and deploy EC2 compute nodes based on prepared AMI images.
- Depending on resource costs different parts of the workflow can be run on different machine instances.
- Data is moved from S3 to EBS and local NVME storage depending on access requirements and machine resources



# V: Summary & Outlook

- **JEDI Cloud Computing**
  - CI / development
  - HPC IaaS: JEDI Applications, optimization
  - JEDI Training: Academies, Tutorials
  - https://nrt.jcsda.org NRT H(x)

- **JCSDA**
  - More web Apps
    - Observation Impact: https://ios.jcsda.org (serverless FSOI diagnostics on-demand)
    - NRT Marine DA: https://soca.jcsda.org
  - R2D2
    - Research Repository for Data & Diagnostics
  - Machine Learning