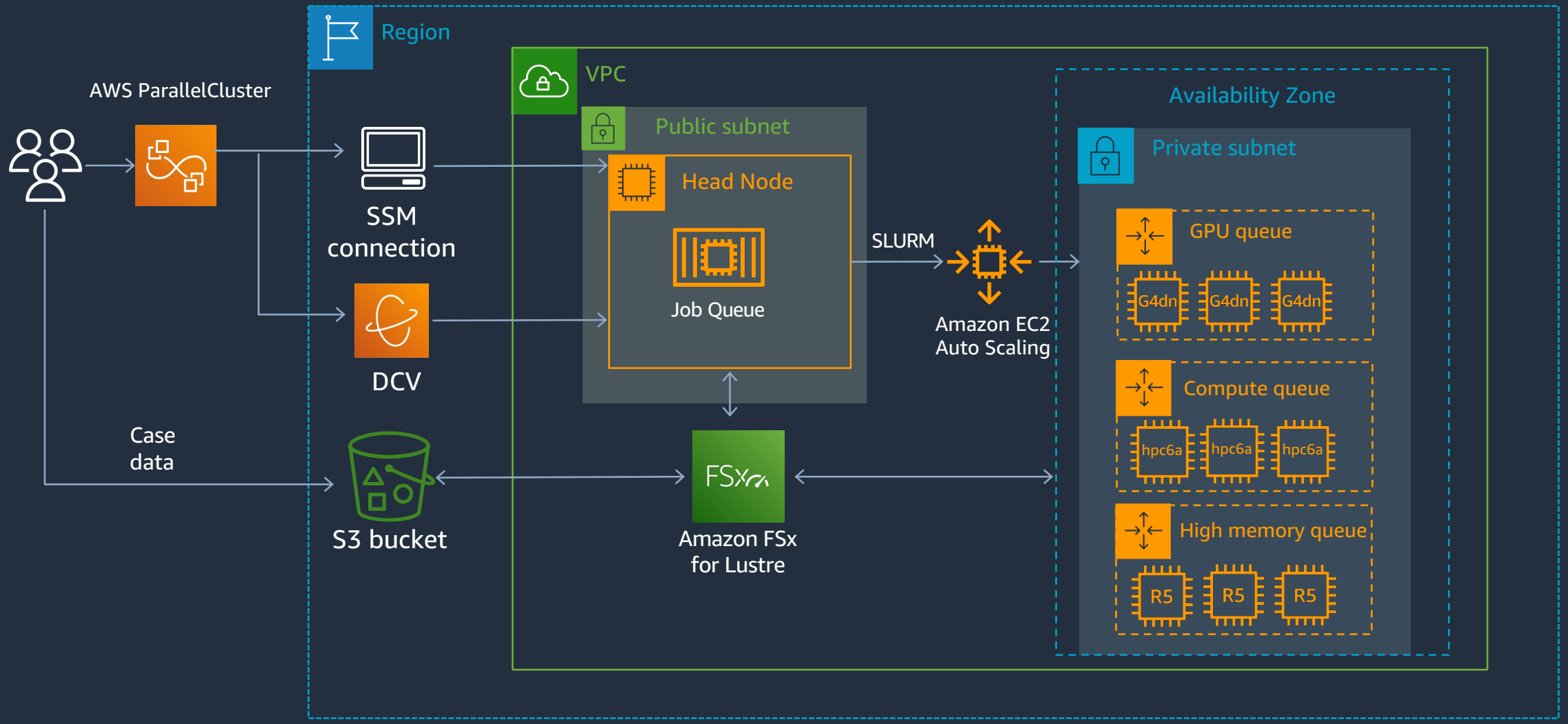
Scaling NWP workloads on AWS to achieve your research goals

Timothy Brown, Sean Smith
aws-hpc-weather@amazon.com

Agenda

1. AWS HPC Overview
2. Parallel Cluster Manager
3. Installing Applications
4. Application Performance Results

AWS ParallelCluster



x86 Instance Snapshot for Weather and Climate

NEW!

Hpc6a.48xlarge

- AMD Milan CPU, up to 3.6GHz frequency
- 96 cores/instance
- 384 GiB/instance
- 100 Gbps Networking Bandwidth w/ EFA

NEW!

C6i.32xlarge

- Intel Icelake CPU, up to 3.5GHz frequency
- 64 cores/instance
- 256 GiB/instance
- 50 Gbps Networking Bandwidth w/ EFA

C5n.18xlarge

- Intel Skylake CPU, up to 3.5GHz frequency
- 36 cores/instance
- 192 GiB/instance
- 100 Gbps Networking Bandwidth w/ EFA

Hpc6a.48xlarge

- OD: 3c/core-hr
- 1Y AURI: 2.21/core-hr
- 3Y AURI: 1.41c/core-hr

C6i.32xlarge

- OD: 8.5c/core-hr
- 1Y AURI: 5.43c/core-hr
- 3Y AURI: 3.26c/core-hr

C5n.18xlarge

- OD: 10.8c/core-hr
- 1Y AURI: 6.35c/core-hr
- 3Y AURI: 3.41c/core-hr

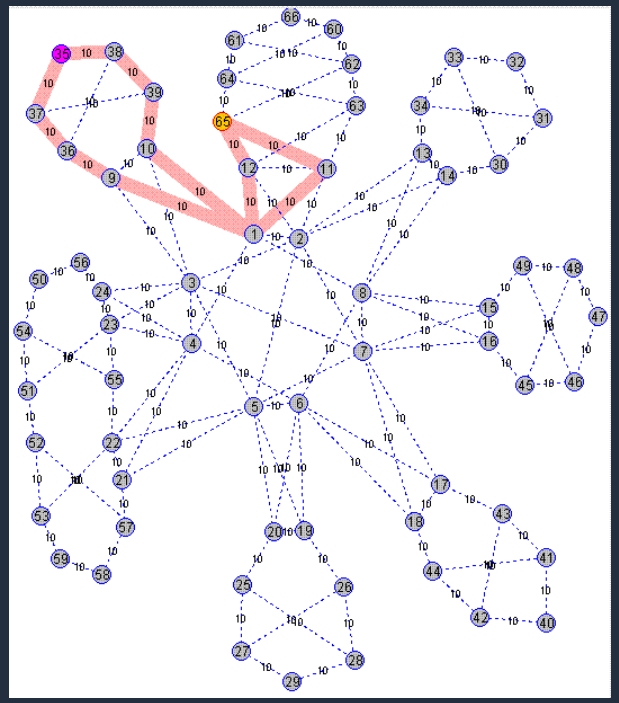
* OD == On-Demand * AURI == All Upfront Reserved Instances * All pricing info is for Ohio (us-east-2) Region

aws

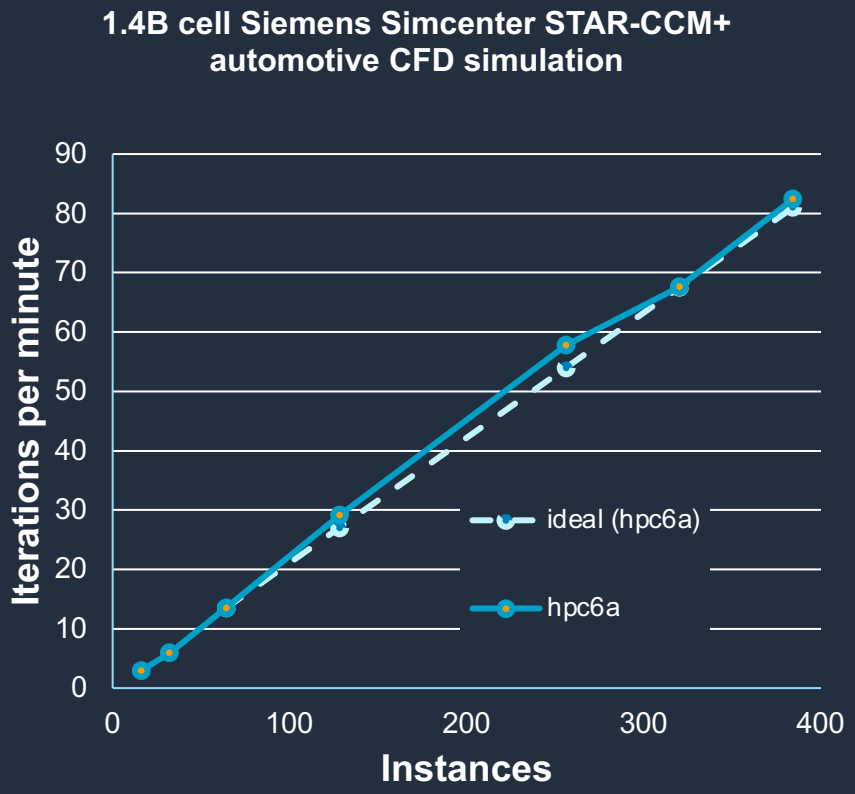
Elastic Fabric Adapter – Networks built to scale

Up to 400 Gbps networking bandwidth

- ✓ OS bypass
- ✓ GPUdirect and RDMA
- ✓ Libfabric core supports wide array of MPIs and NCCL



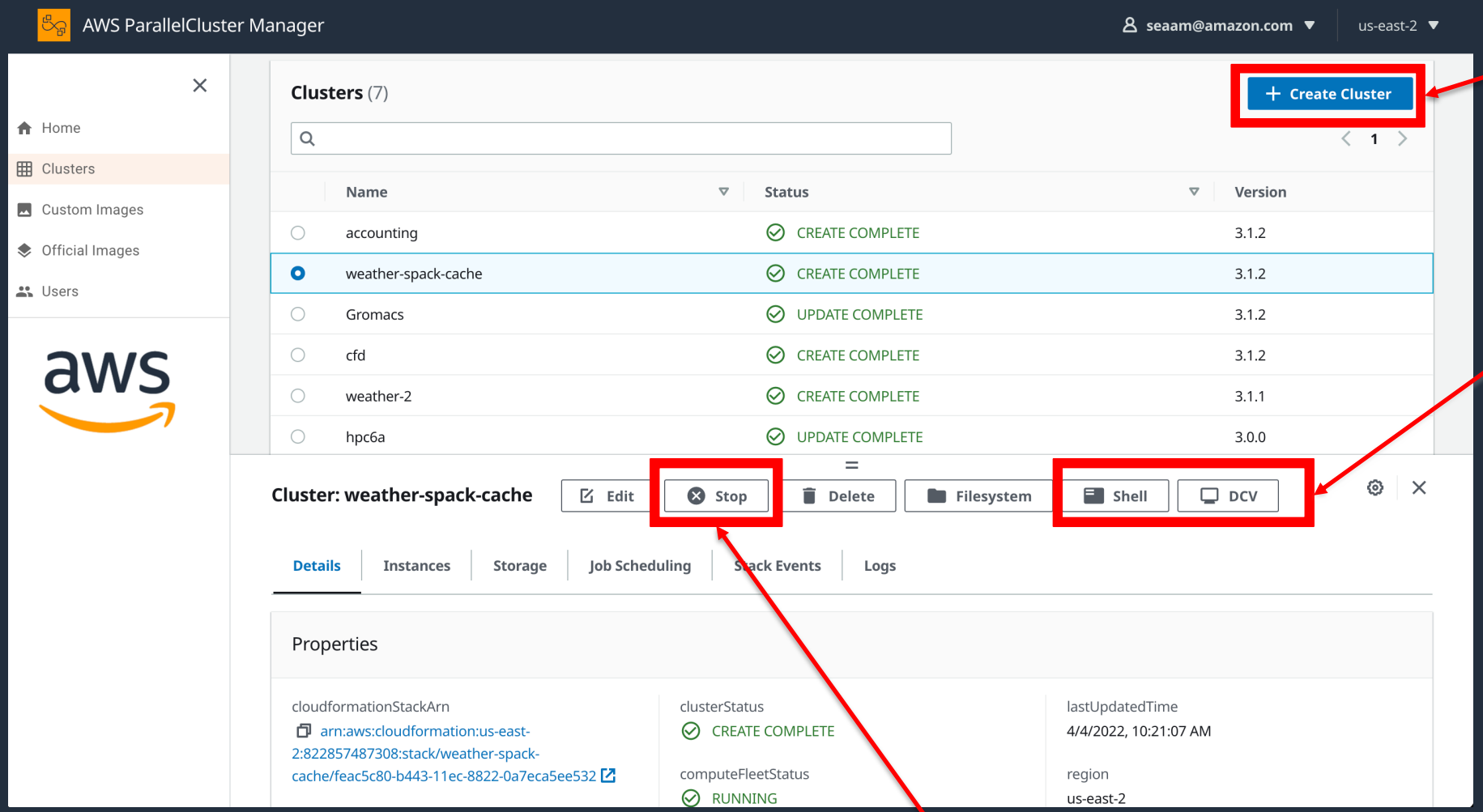
ECMP-enabled packet spraying and cloud-scale congestion control



At ~40,000 cores (400 nodes), Hpc6a+EFA shows nearly 100% scaling efficiency

Elastic Fabric Adapter

15-minutes to create a HPC Cluster



Create-Cluster Wizard

Connect via the browser (or SSH)

Stop a cluster (preserve all data)
Then resume later

Slurm

```
Scheduling:
  Scheduler: slurm
  SlurmQueues:
    - Name: general
      ComputeResources:
        - Name: hpc6a
          InstanceType: hpc6a.48xlarge
          MinCount: 0
          MaxCount: 200
          Efa:
            Enabled: true
      Networking:
        SubnetIds:
          - subnet-XXXXXXXXXXXX
        PlacementGroup:
          Enabled: true
```

Slurm is the built-in scheduler with ParallelCluster.

We have specified 0 nodes running by default.

```
[ec2-user@ip-172-31-21-227 ~]$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
general*     up   infinite    200  idle~ general-dy-hpc6a-[1-200]
[ec2-user@ip-172-31-21-227 ~]$
```

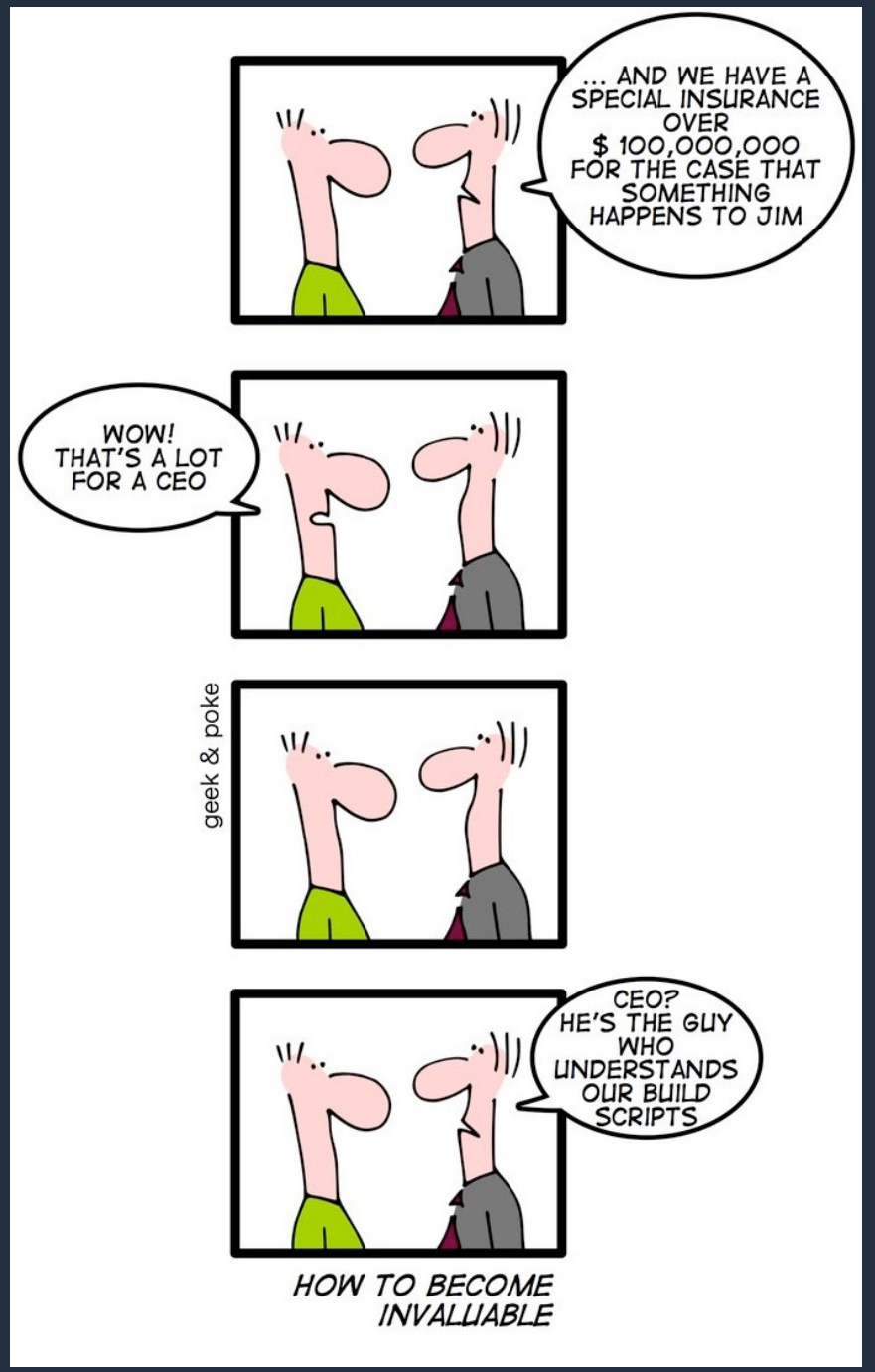
Use Slurm's power management.

aws

Installing Software

Common ways

- Custom Amazon Machine Image (AMI) via packer
- Package managers (Spack, Easybuild, …)
- By hand



https://geek-and-poke.com/geekandpoke/2010/5/14/how-to-become-invaluable.html

Spack – Package Manager

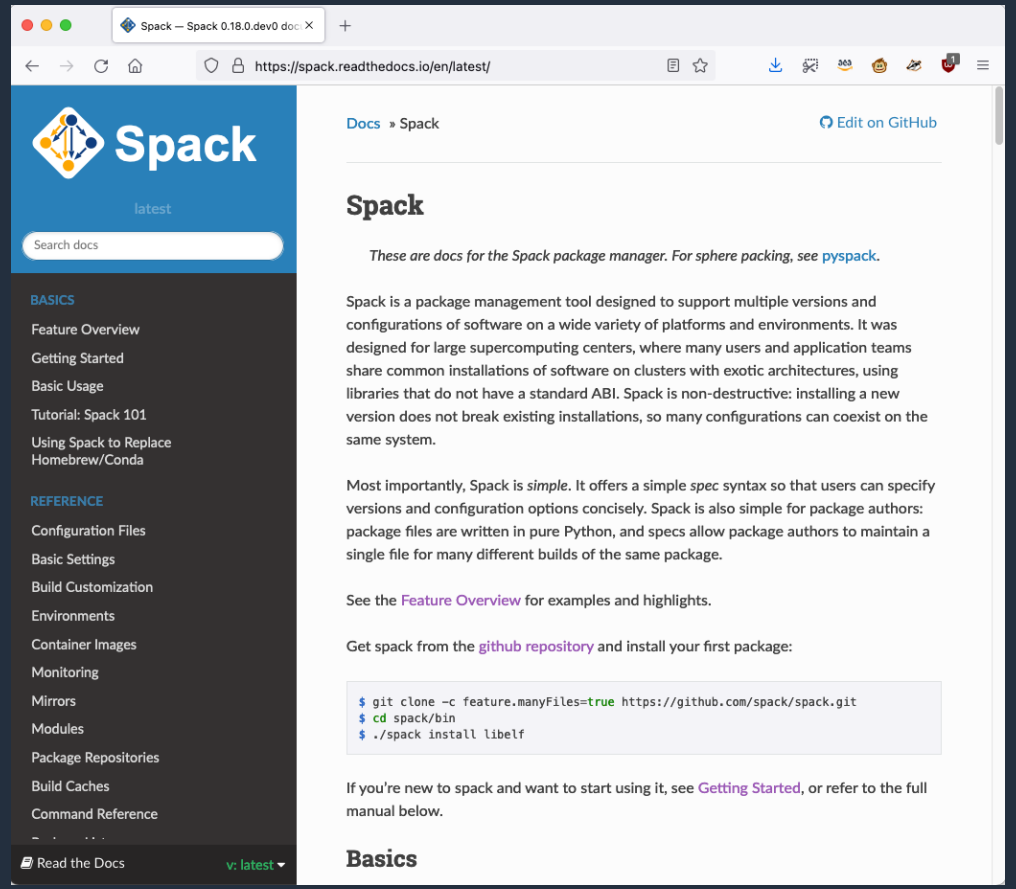
We are going to use Spack[1] to install all our packages.

```
$ export SPACK_ROOT=/shared/spack
$ git clone -c feature.manyFiles=true https://github.com/spack/spack $SPACK_ROOT
$ echo "export SPACK_ROOT=/shared/spack" >> $HOME/.bashrc
$ echo "source \$SPACK_ROOT/share/spack/setup-env.sh" >> $HOME/.bashrc
$ source $HOME/.bashrc
```

Spack binary cache

https://binaries.spack.io/develop

```
$ spack mirror add binary_mirror https://binaries.spack.io/develop
$ spack buildcache keys --install --trust
```



1. Todd Gamblin, Matthew P. LeGendre, Michael R. Collette, Gregory L. Lee, Adam Moody, Bronis R. de Supinski, and W. Scott Futral. The Spack Package Manager: Bringing Order to HPC Software Chaos. In *Supercomputing 2015 (SC'15)*, Austin, Texas, November 15-20 2015. LLNL-CONF-669890.

aws

Spack External Packages

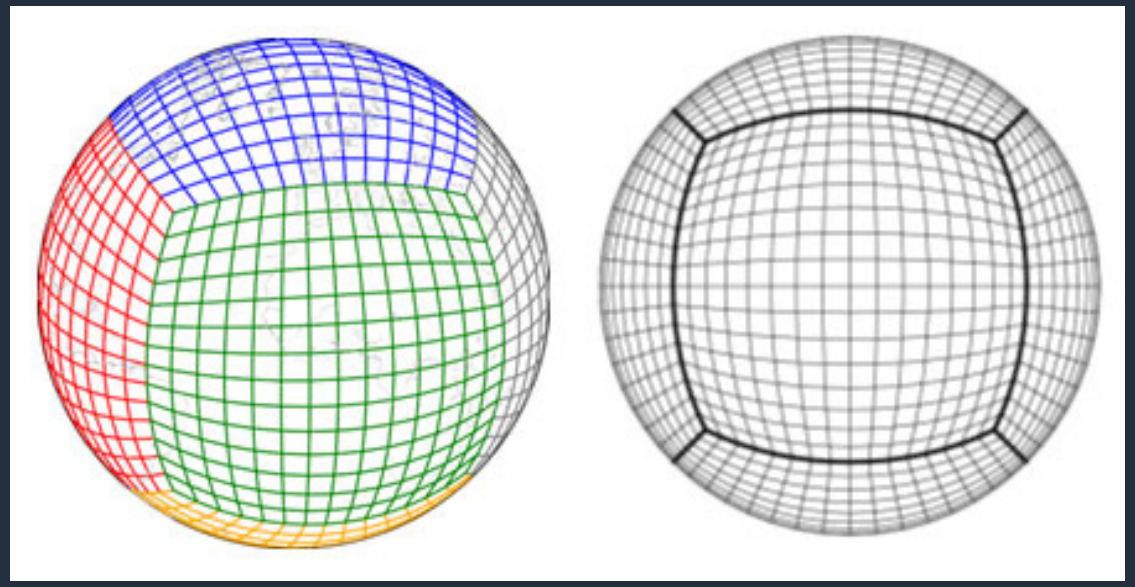
We define the external packages in $SPACK_ROOT/etc/spack/packages.yaml

```
packages:
    libfabric:
        variants: fabrics=efa,tcp,udp,sockets,verbs,shm,mrail,rxd,rxm
        externals:
        - spec: libfabric@1.16.0 fabrics=efa,tcp,udp,sockets,verbs,shm,mrail,rxd,rxm
          prefix: /opt/amazon/efa
        buildable: False
```


aws

UFS Weather Model

- Version 2.0.0
- Global C768 with 65 vertical levels
- 2 day forecast
- Model timestep is 150 seconds



C768 → 768 cells per direction per face → 13 km resolution

aws

Building UFS Weather Model With Spack

spack install –j 12 ufs-weather-model%intel^intel-oneapi-mpi+external-libfabric

Spack Arguments and Flags	Description
install	Install a package.
-j 12	Parallel build with 12 cores.
ufs-weather-model	The package to install.
%intel	Specify the Intel compiler (icc and ifort).
^intel-oneapi-mpi	Use the Intel OneAPI MPI library.
+external-libfabric	Use an external fabric for MPI.

aws

Running UFS Weather Model

I_MPI_OFI_LIBRARY_INTERNAL must be set to 0 before loading Intel MPI

Specify EFA as the fabric adaptor

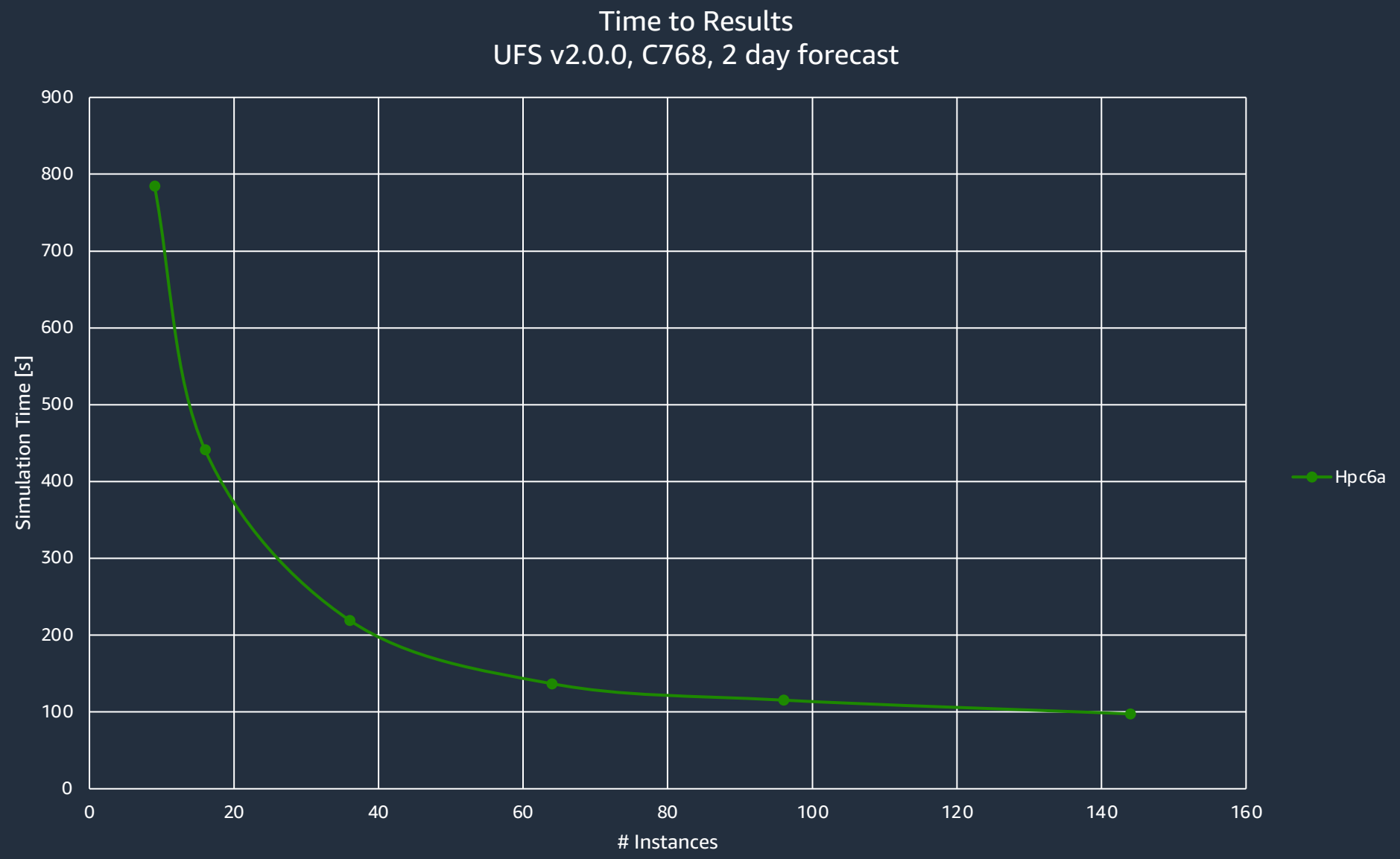
```
#!/bin/bash

#SBATCH --job-name=UFS
#SBATCH --output=slurm-ufs-%j.out
#SBATCH --nodes=150
#SBATCH --ntasks-per-node=24
#SBATCH --exclusive

export I_MPI_DEBUG=4
export I_MPI_FABRICS=ofi
export I_MPI_OFI_LIBRARY_INTERNAL=0
export I_MPI_OFI_PROVIDER=efa
export I_MPI_PIN_DOMAIN=omp
export KMP_AFFINITY=compact
export OMP_NUM_THREADS=4


spack load intel-oneapi-mpi
spack load ufs-weather-model
module load libfabric-aws
set -x
ulimit -s unlimited
ulimit -a

time mpiexec.hydra ufs_weather_model
```

Confirm EFA is used, when I_MPI_DEBUG=4
(slurm job output text)

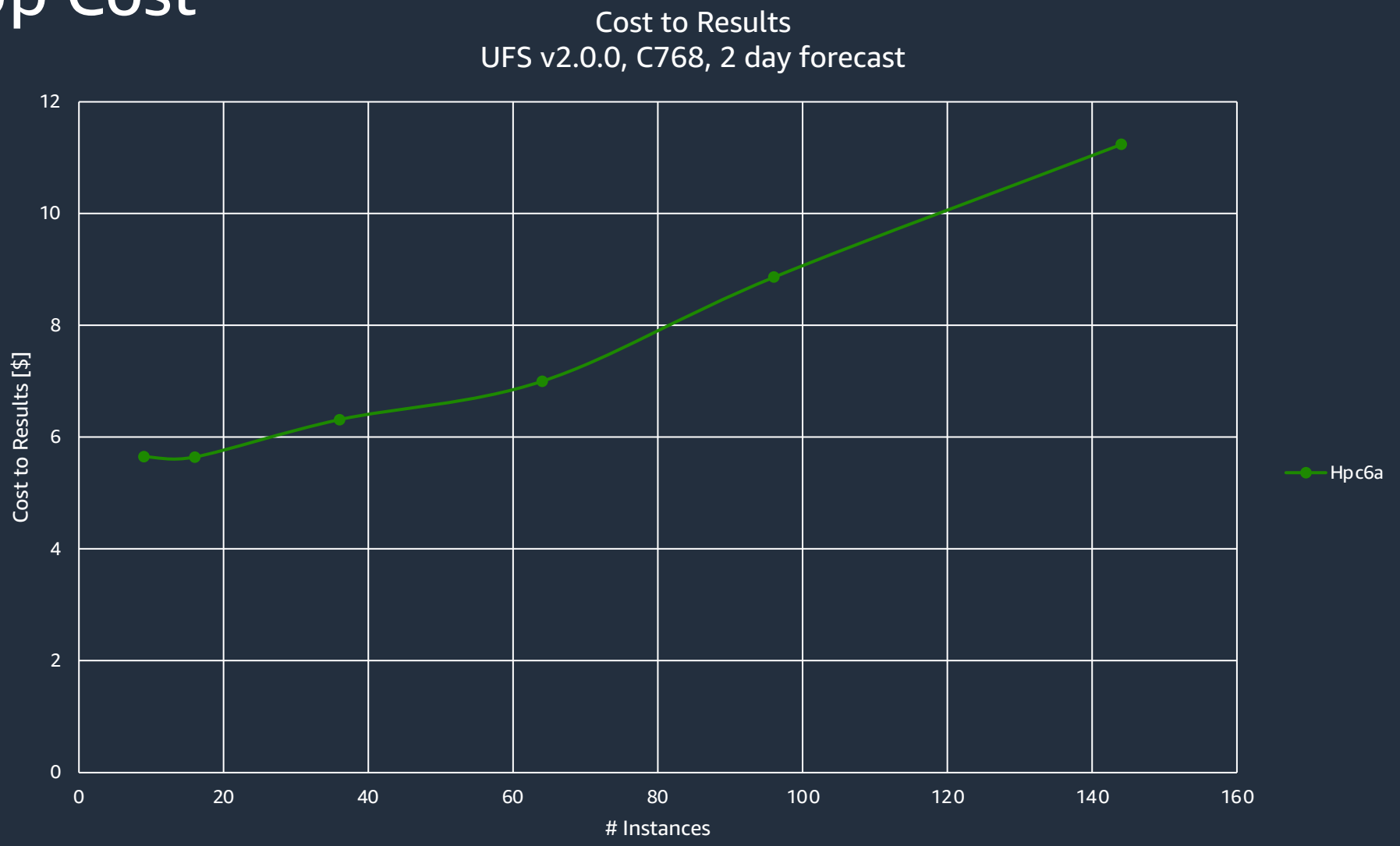
```
[0] MPI startup(): libfabric version: 1.16.0~amzn3.0
[0] MPI startup(): libfabric provider: efa
```


aws

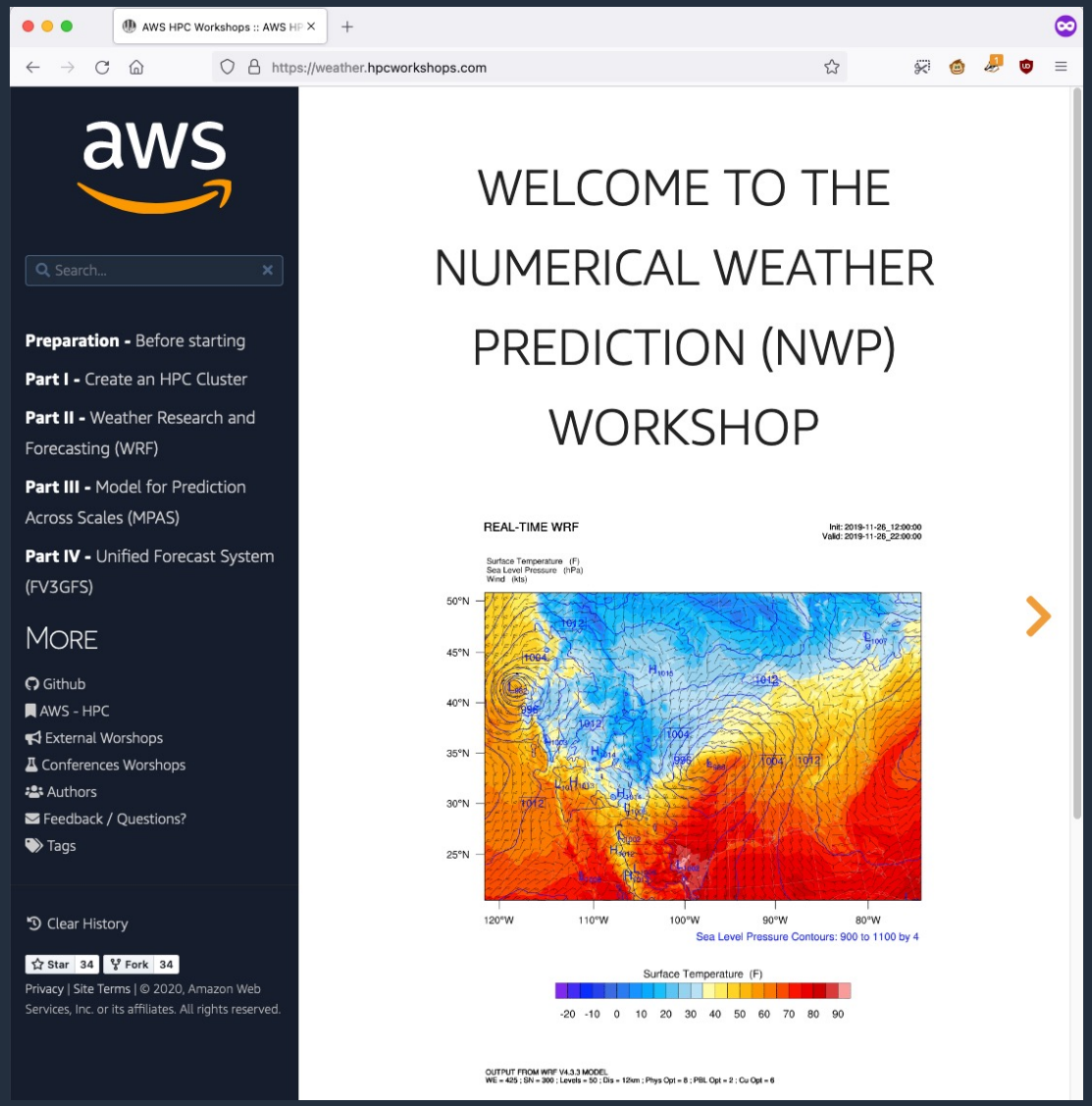
Scale Up Performance



Scale Up Cost



Weather Workshops https://weather.hpcworkshops.com



Thank you

If you have any additional questions, please feel free to reach out.

Timothy Brown, Sean Smith
aws-hpc-weather@amazon.com

aws

Weather Models Run On AWS

- WRF ARW
- MPAS Atmosphere
- UFS Atmosphere (FV3GFS)
- Unified Model
- Harmonie
- ICON

aws