LUG 2023 | MAY 1-4, 2023

Managing cloud HPC with infrastructure-as-code

Matt Vaughn

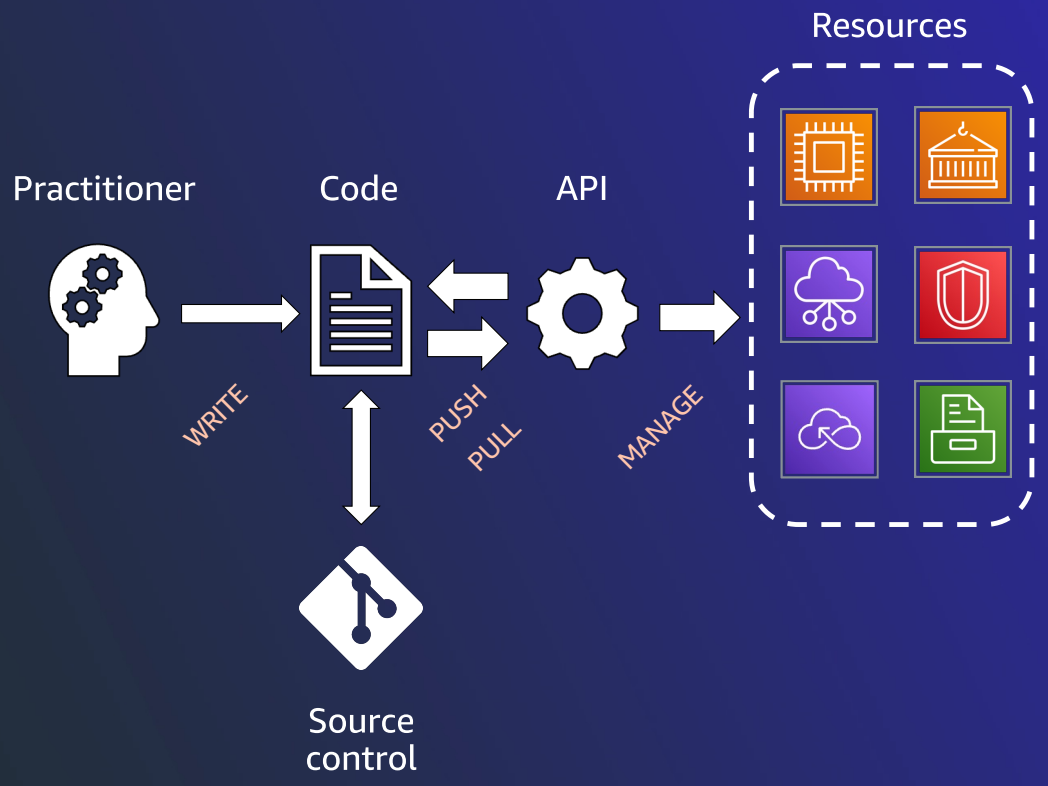
Principal Developer Advocate
HPC Engineering
Amazon Web Services

1. What is infrastructure as code?
2. Key IaC technologies
3. Interactive vs managed deployments
4. Infrastructure by composition
5. Exemplar HPC infrastructure as code
6. Use case 1: Complex compute environment
7. Use case 2: HPC-Ops
8. Summary and conclusion

What is Infrastructure as Code (IaC)?

Managing and provisioning of infrastructure through code instead of manual processes.

- Practitioner writes code
- Code managed under source control
- Push/pull to automation server
- Resources managed declaratively



Some key IaC technologies

AWS CloudFormation

AWS CDK

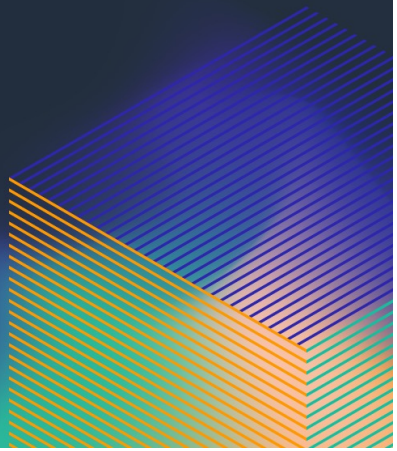
Terraform

CDKTF

Pulimi

Ansible/Chef/Puppet/Salt*

How does this apply to HPC?



AWS FSx Console

Create file system

File system details

File system name - optional Info

lugdemo

Maximum of 256 Unicode letters, whitespace, and numbers, plus + - = . _ : /

Deployment and storage type Info

Select a deployment type and storage type to fit your workload requirements

- (•) Persistent, SSD
- ( ) Persistent, HDD
  - [x] with SSD cache
- ( ) Scratch, SSD

Throughput per unit of storage Info

Throughput (MB/s) per unit of storage (TiB)

- ( ) 125 MB/s/TiB
- ( ) 250 MB/s/TiB
- ( ) 500 MB/s/TiB
- (•) 1000 MB/s/TiB

Storage capacity Info

240 TiB

Supported sizes: 1.2 TiB or increments of 2.4 TiB

Throughput capacity Info

Throughput capacity = Storage capacity (TiB) * Per unit storage throughput (MB/s)

240000 MB/s

Data compression type Info

Data compression reduces the physical disk space needed to store file data. Select LZ4 to enable data compression

LZ4

AWS CLI v2

```
% aws fsx create-file-system \
  --file-system-type LUSTRE \
  --storage-capacity 240000 \
  --storage-type SSD \
  --subnet-ids subnet-0123456789 \
  --security-group-ids sg-0123456a \
  --lustre-configuration \
    {“DeploymentType”: “PERSISTENT_2”,
     ”PerUnitStorageThroughput”: 1000,
     “DataCompressionType”: “LZ4”}
```

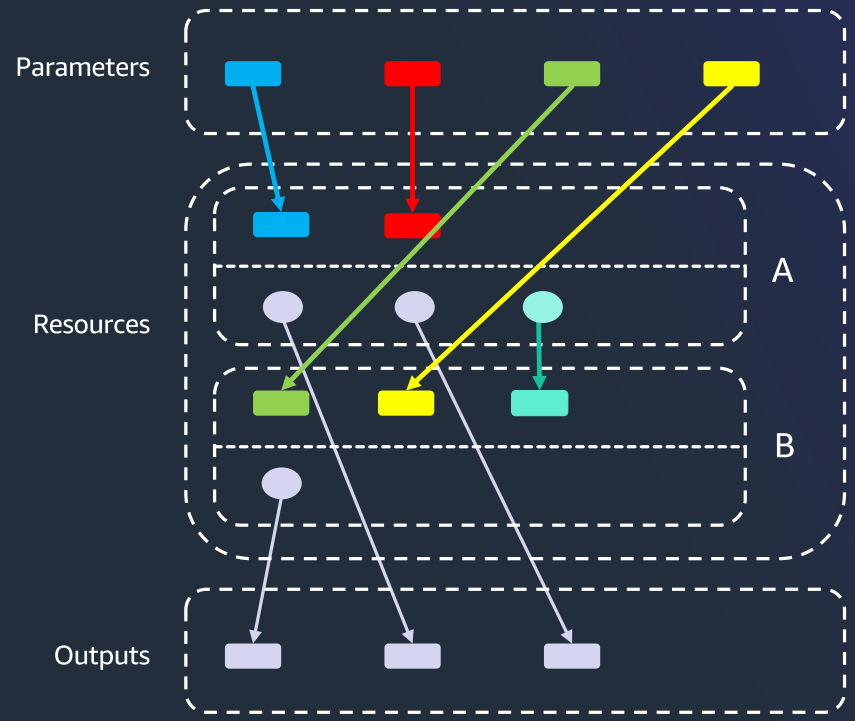
```
resource "aws_fsx_lustre_file_system" "lugdemo" {
    deployment_type = "PERSISTENT_2"
    per_unit_storage_throughput = 1000
    storage_type = "SSD"
    data_compression_type = "LZ4"
    storage_capacity = 240000
    subnet_ids = ["subnet-0123456789"]
    security_group_ids = ["sg-0123456a"]
}
```

Hashicorp Terraform

```
FSxLFilesystem:
    Type: AWS::FSx::FileSystem
    Properties:
        FileSystemType: LUSTRE
        FileSystemTypeVersion: "2.12"
        StorageType: SSD
        StorageCapacity: 240000
        SecurityGroupIds:
            - sg-0123456a
        SubnetIds:
            - subnet-0123456789
        LustreConfiguration:
            DataCompressionType: LZ4
            DeploymentType: PERSISTENT_2
            PerUnitStorageThroughput: 1000
```

AWS CloudFormation

Infrastructure via composition

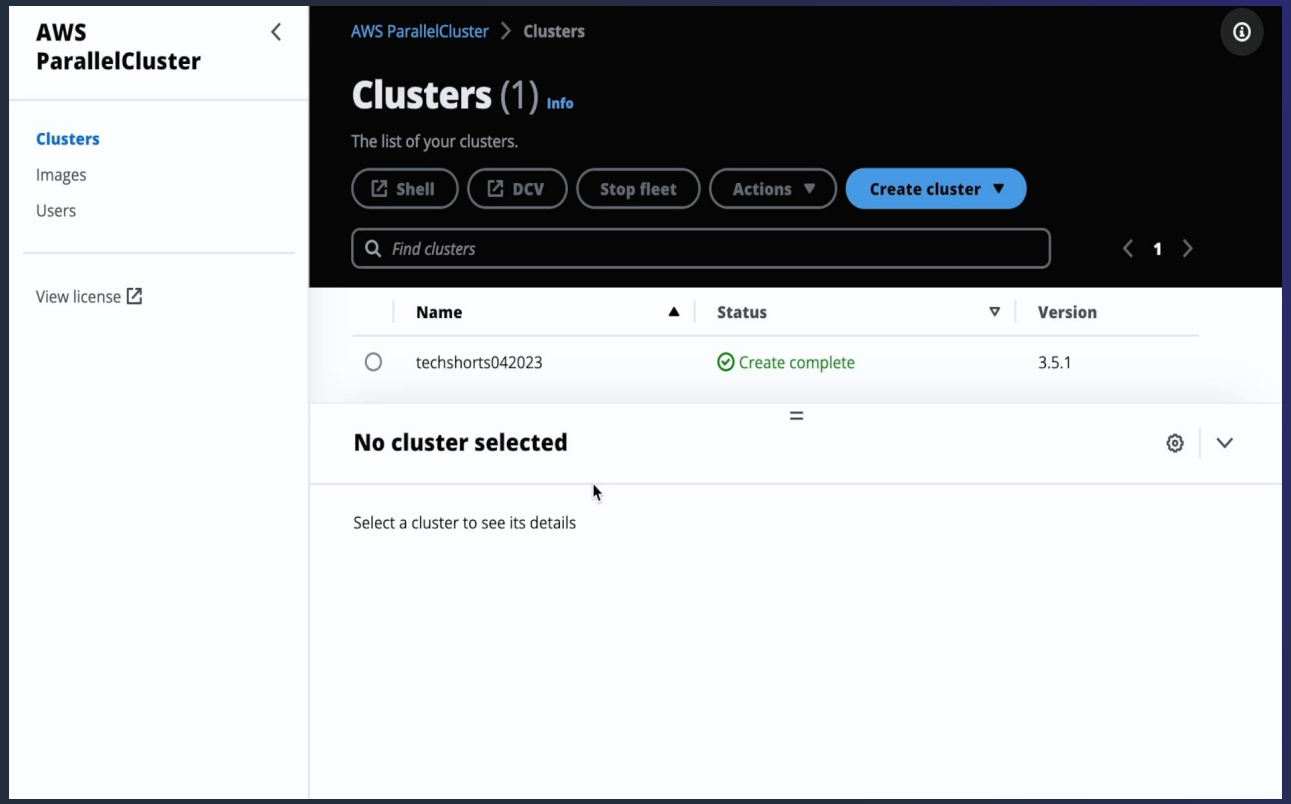


CloudFormation Template

- A CloudFormation template deploys a ***stack***
- Each template has Parameters, ***Resources***, Outputs, Mappings, & Conditionals
- Resources are AWS (and other) types, including other CloudFormation stacks
- Resources have Properties which can be defined by parameters or by properties of other resources
- Outputs from nested stacks can be as parameters and property values for dependent resources
- CloudFormation service handles order of execution, drift detection, etc.
- CloudFormation templates are DAGs

aws

But… how does this apply to HPC?

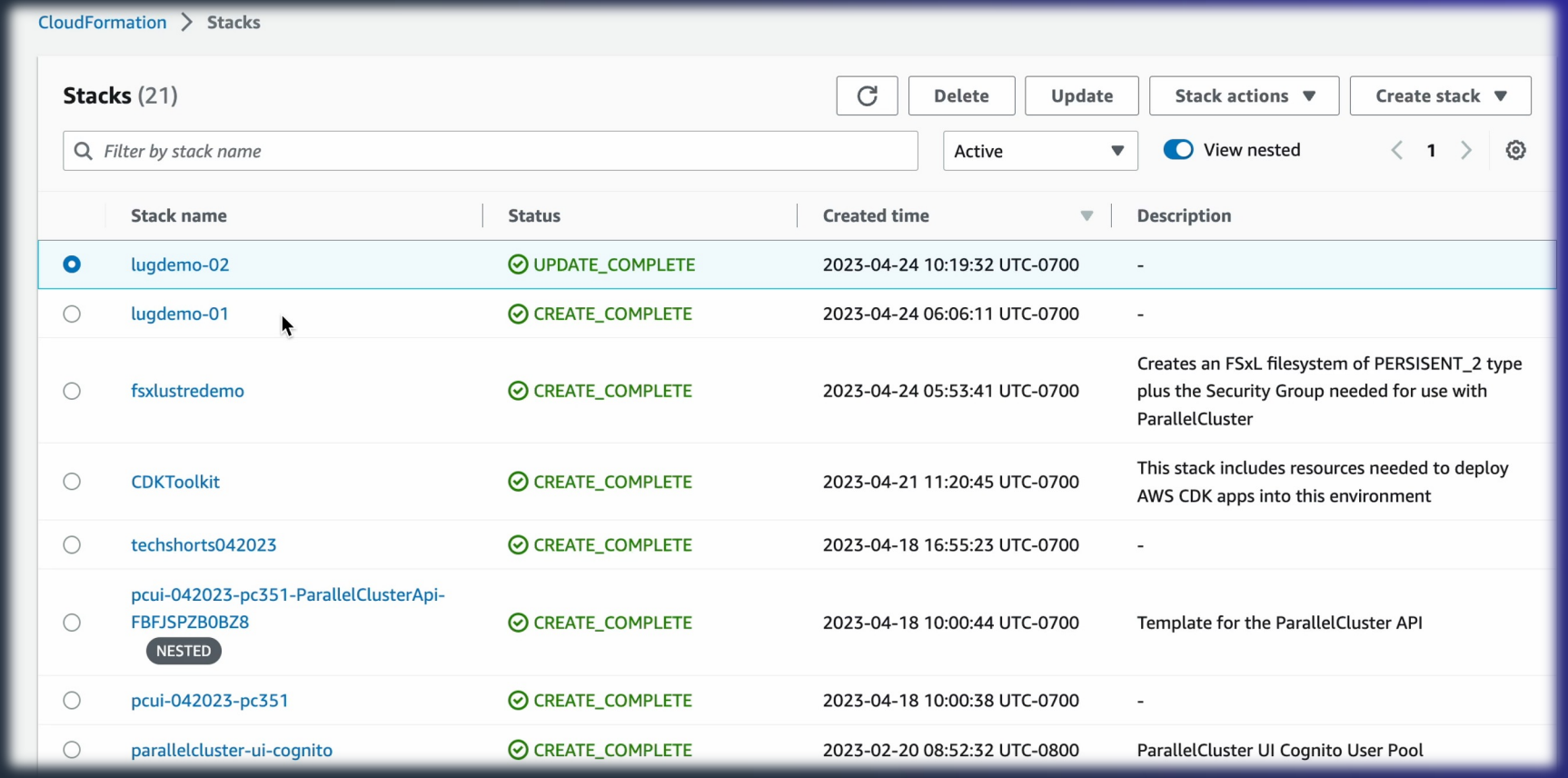


pcluster create-cluster –n lugdemo-2 –c demo-config.yml

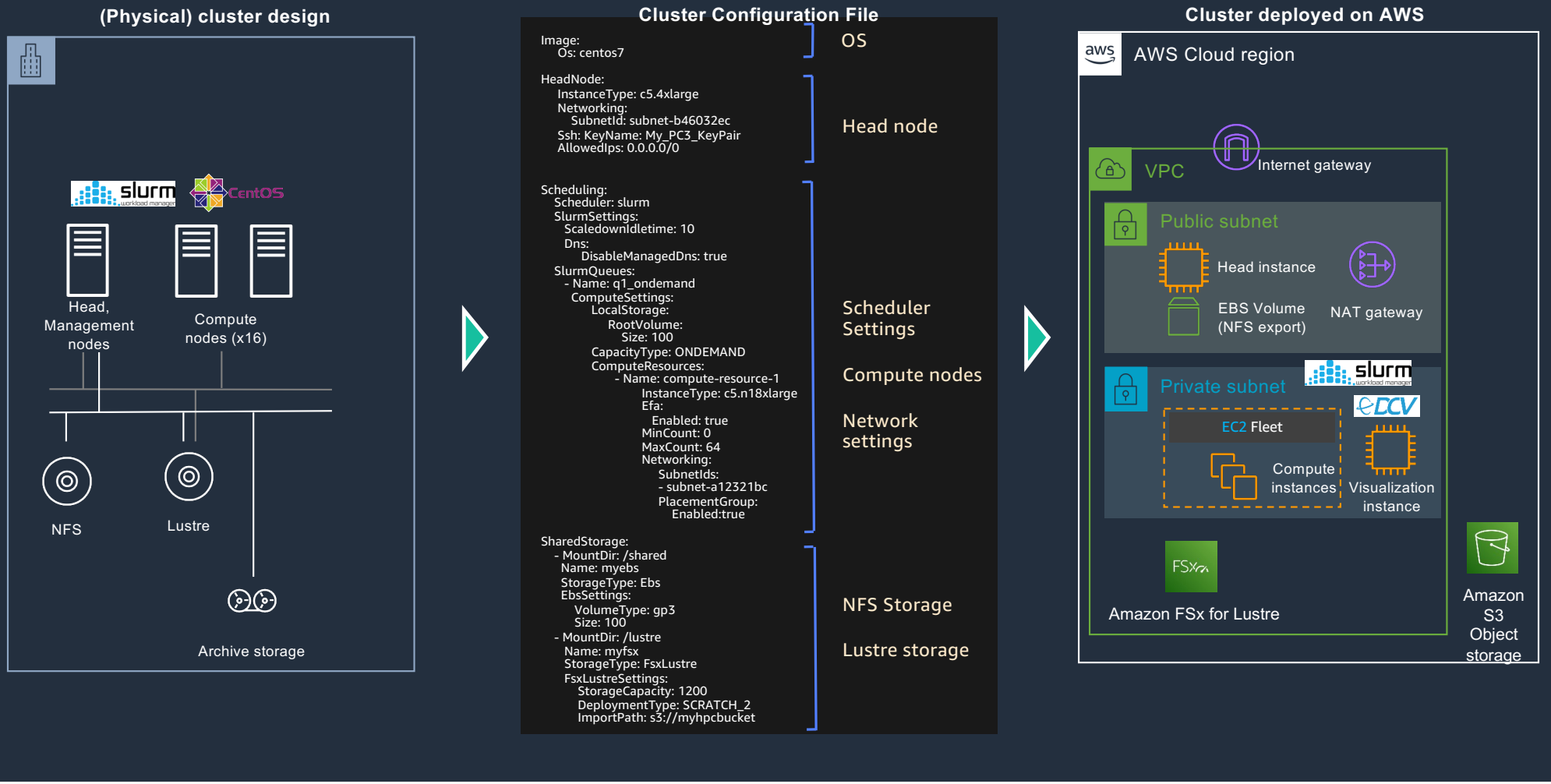
ParallelCluster is a first-class example of Infrastructure as Code

1. Cluster architecture defined with IaC
2. Under the hood, ParallelCluster uses CloudFormation to implement cluster components
    1. Networking/Security/IAM
    2. Storage (EFS, FSx, EBS, S3)
    3. Compute/Head Node resources
    4. (A lot of other stuff)

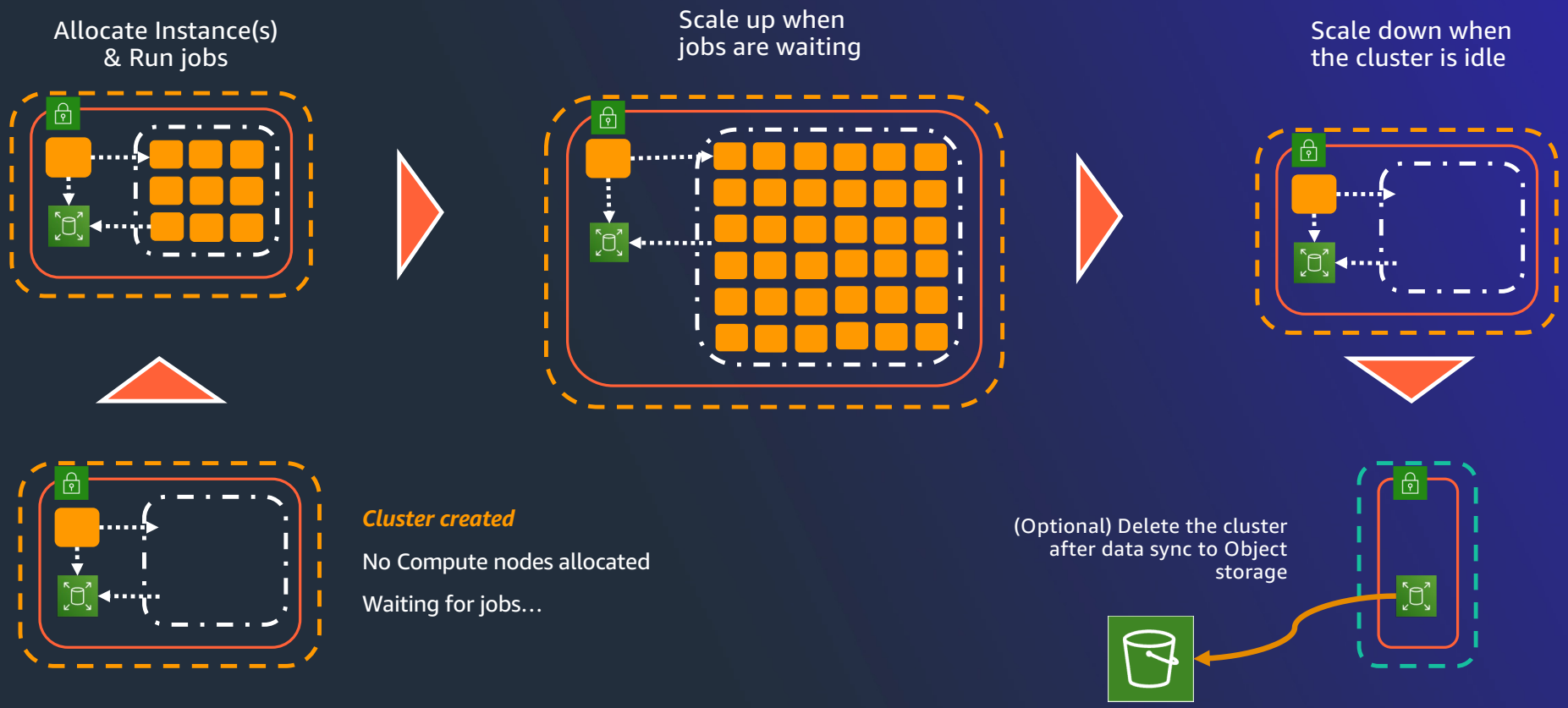
ParallelCluster clusters as CloudFormation stacks



Cluster definition in a single file



Dynamic compute resource scaling



CloudFormation for cluster deployments

New in ParallelCluster 3.6

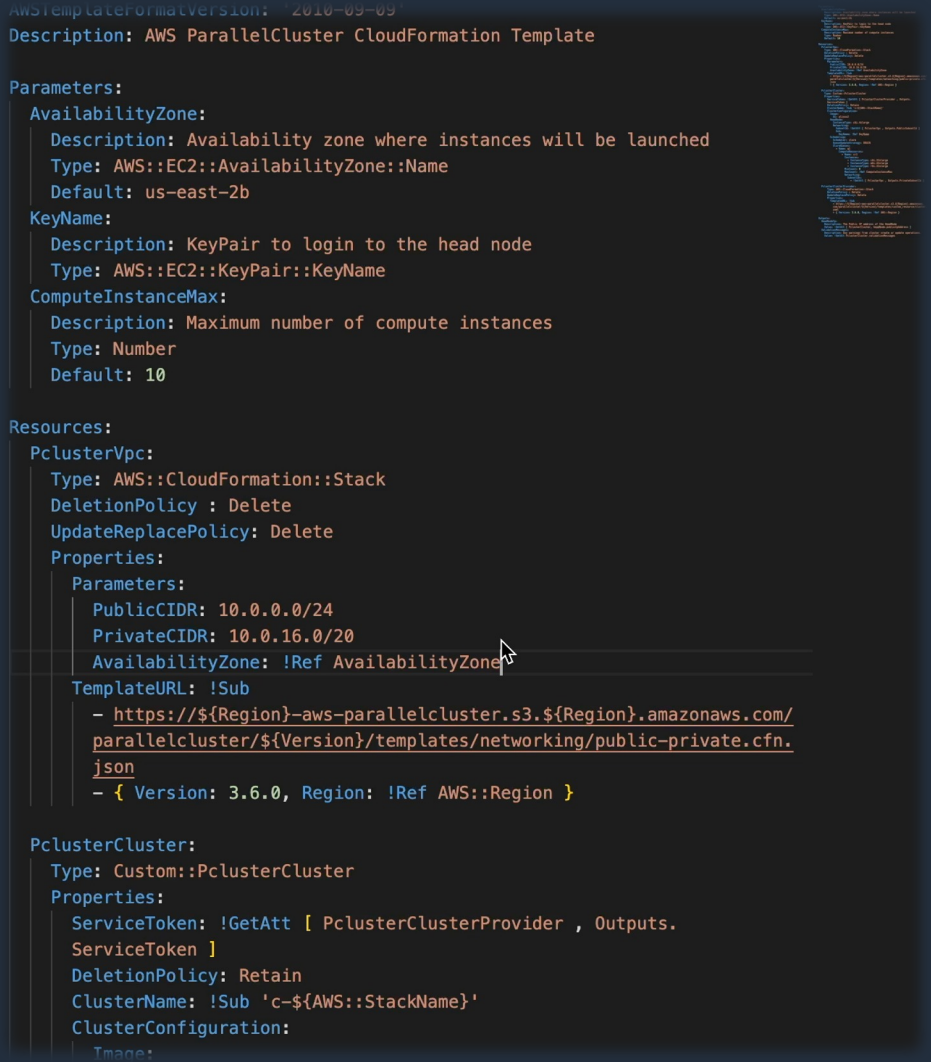
Define your cluster as a CloudFormation template

No user-installed CLI or web UI needed

Should also work with AWS CDK

Embed HPC in complex IT systems

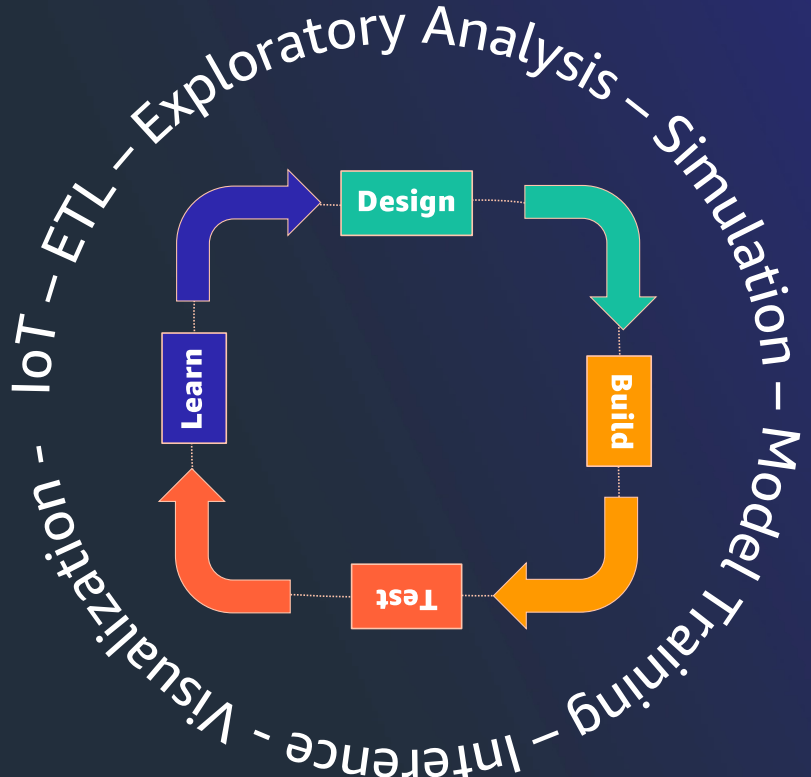
- Interoperability between compute environments
- *Ops-models for HPC workloads



- Choose Availability Zone for instances
- Select SSH key name
- Specify max # instances
- Provision VPC + subnets
- Provision cluster in VPC & subnets

```
aws cloudformation create-stack \
    --stack-name lugdemo-3 \
    --template-body file://demo.yml
```

A real-world complex computing environment



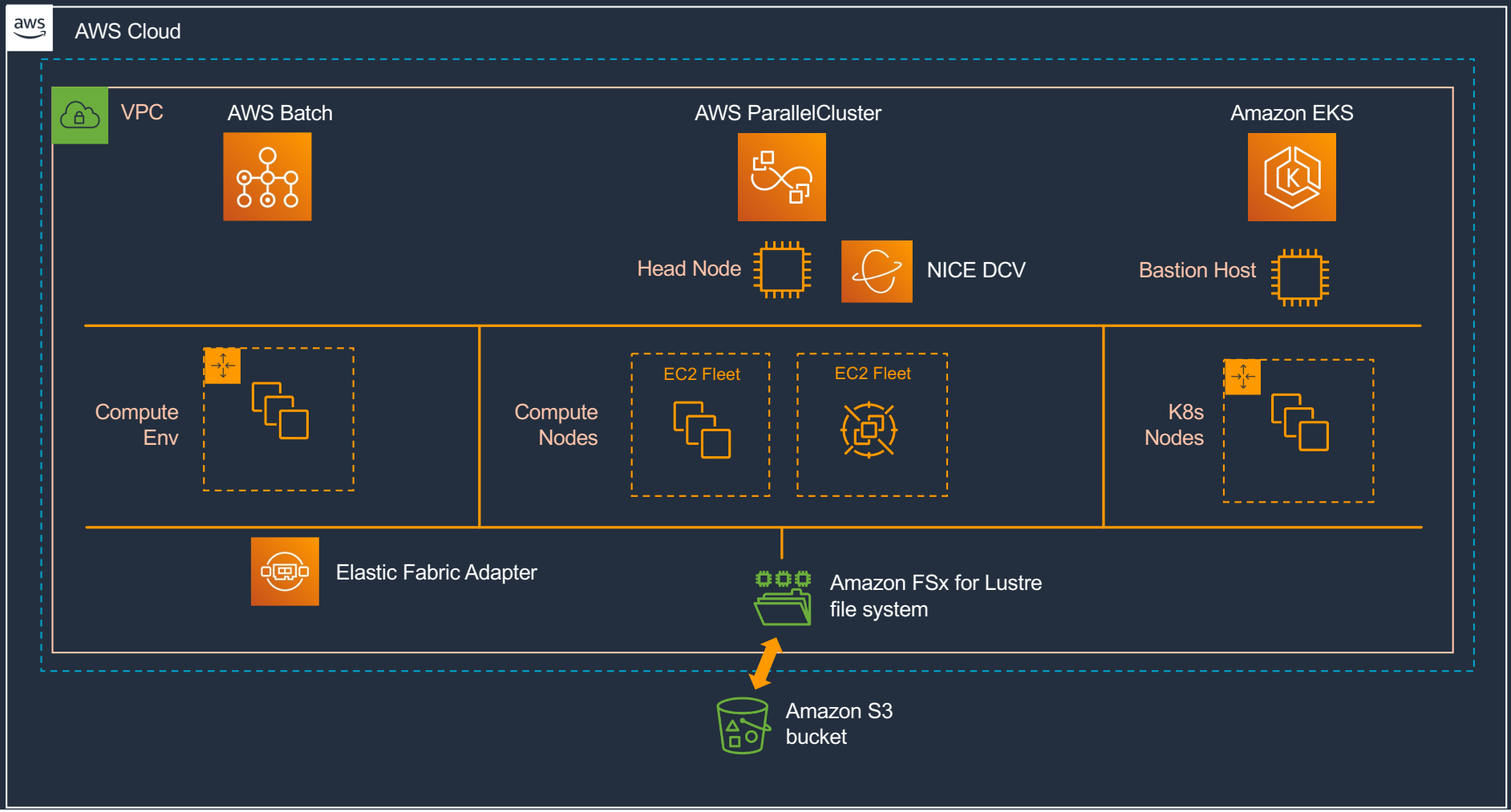
Option 1 - Fit everything in one computing paradigm

- HPC?
- Batch?
- K8s?

Option 2 –Integrate computing paradigms

- HPC
- Batch
- K8s

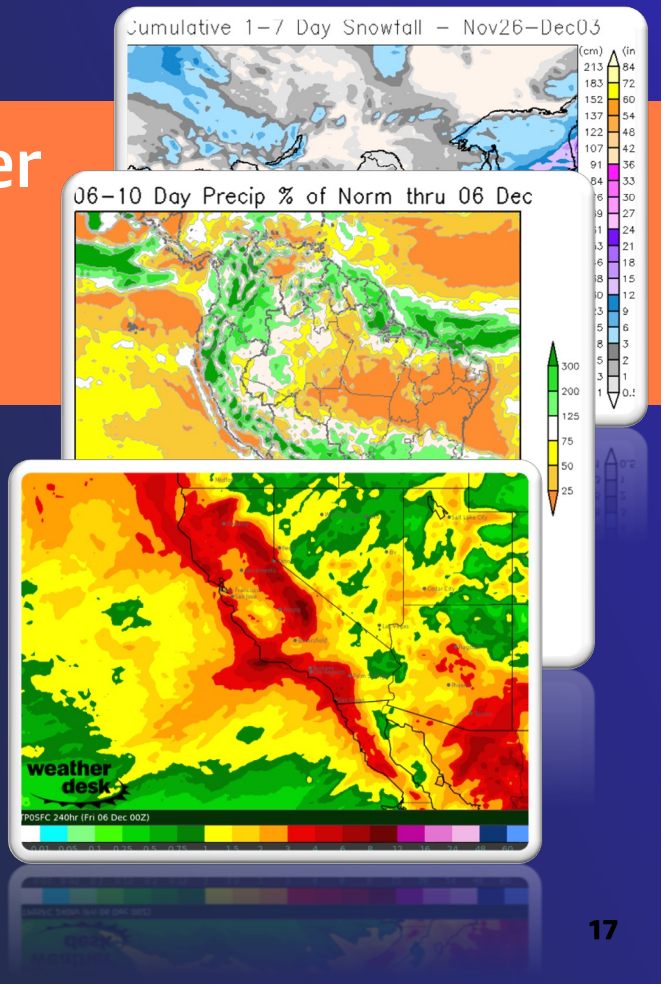
Unifying multiple compute systems with FSx for Lustre



HPC-Ops – an emerging urgent computing paradigm made easier with IaC

MAXAR

58% Faster than NOAA Supercomputer
45% Lower Compute Cost
~ 5600 Cores



Each forecasting job is run on an ephemeral HPC cluster

Charges incurred for ~45min

aws

Conclusions

- Infrastructure as code is a powerful approach for modeling and managing complex resource deployments
- AWS ParallelCluster uses IaC to deploy and manage dynamic, autoscaling HPC
- AWS ParallelCluster 3.6 supports cluster deployment directly with CloudFormation
- This allows more sophisticated IT integrations with HPC
- It also makes it easier to implement DevOps, MLOps, DataOps, etc.

Questions

Supplementary Resources

- HPC Workshops
  - https://www.hpcworkshops.com/
  - https://workshops.aws/categories/HPC
- Media:
  - AWS HPC Blog : https://aws.amazon.com/blogs/hpc/
  - HPC Tech Shorts YouTube: https://www.youtube.com/c/hpctechshorts
  - Community Site: https://day1hpc.com/