Introduction to Cloud Computation

Or how I learned to get over my fears and learn love the cloud

John Massey, ASRC Federal

Overview

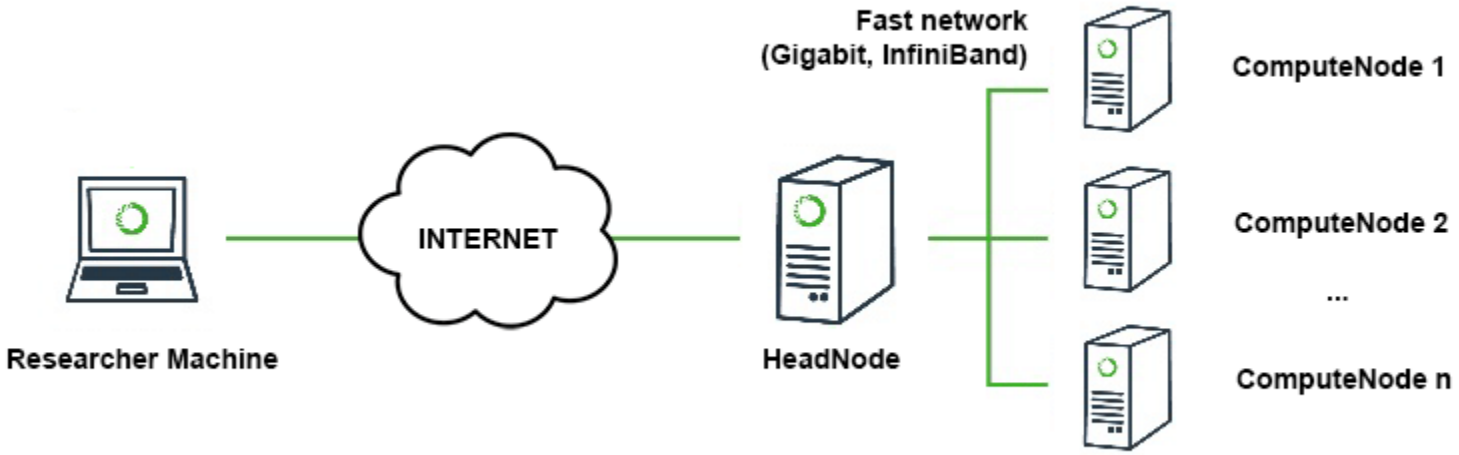
- Typical Model Execution
- What is a High Performance Cluster Anyway?
- Benefits of Cloud Processing
- Cloud Providers and the CBPO
- HPC Implementation at CBPO
- Cloud Computing Costs
- How we could do it better
- Cloud Providers and You
- Getting started in AWS

Typical Model Execution

- What is a model to a computer?
  - Simply put, to a computer a model is nothing more then another piece, or several pieces, of software to run.
- Can I run a model locally or on a single server?
  - Yes, well maybe or maybe not
  - Do you have days? Weeks? Or maybe even months to wait for results?
- How about on a High Performance Cluster?
  - A HPC or High Performance cluster is ideal for large jobs when combines with a job scheduler like SLURM or Torque, OpenLava, etc….

What is a High Performance Cluster Anyway?

"High-Performance Computing," or HPC, is the application of "supercomputers" to computational problems that are either too large for standard computers or would take too long. A desktop computer generally has a single processing chip, commonly called a CPU. A HPC system, on the other hand, is essentially a network of nodes, each of which contains one or more processing chips, as well as its own memory. – The National Institute for Computational Sciences



Typical Model Execution HPC Edition

- Benefits of HPC Computation / Parallel Processing
  - Break large jobs into smaller tasks that can be handled by individual processing cores.
  - Run several Large processing tasks simultaneously.
  - Perform multiple modeling steps that could perform all tasks from initial data loading through processing, and potentially even post processing.
  - Thanks to lower processing times, multiple result sets can be generated much quicker.
  - Job queues allow for unattended processing.

Benefits of Cloud Processing

- Lower cost of ownership*
  - Pay only for what you use.
  - Low or no up front cost.
- Flexibility
  - Compute Resources
    - Allows for multiple configurations of systems.
    - Can be temporary or permanent.
  - Storage
    - Very large file storage at varying costs.
    - Provides for multiple data access types.
- Different Resource types
  - Software as a service
  - Infrastructure as a service
  - Platform as a service

Cloud Providers and the CBPO

- CBPO utilizes cloud resources provided by UMCES through an ongoing grant.
- After a review of all the major cloud providers (Google/Microsoft/Amazon) UMCES choose Amazon to serve as our cloud provider.
  - At the time AWS was the most mature.
  - Offered the largest array of service options.

HPC Implementation at CBPO

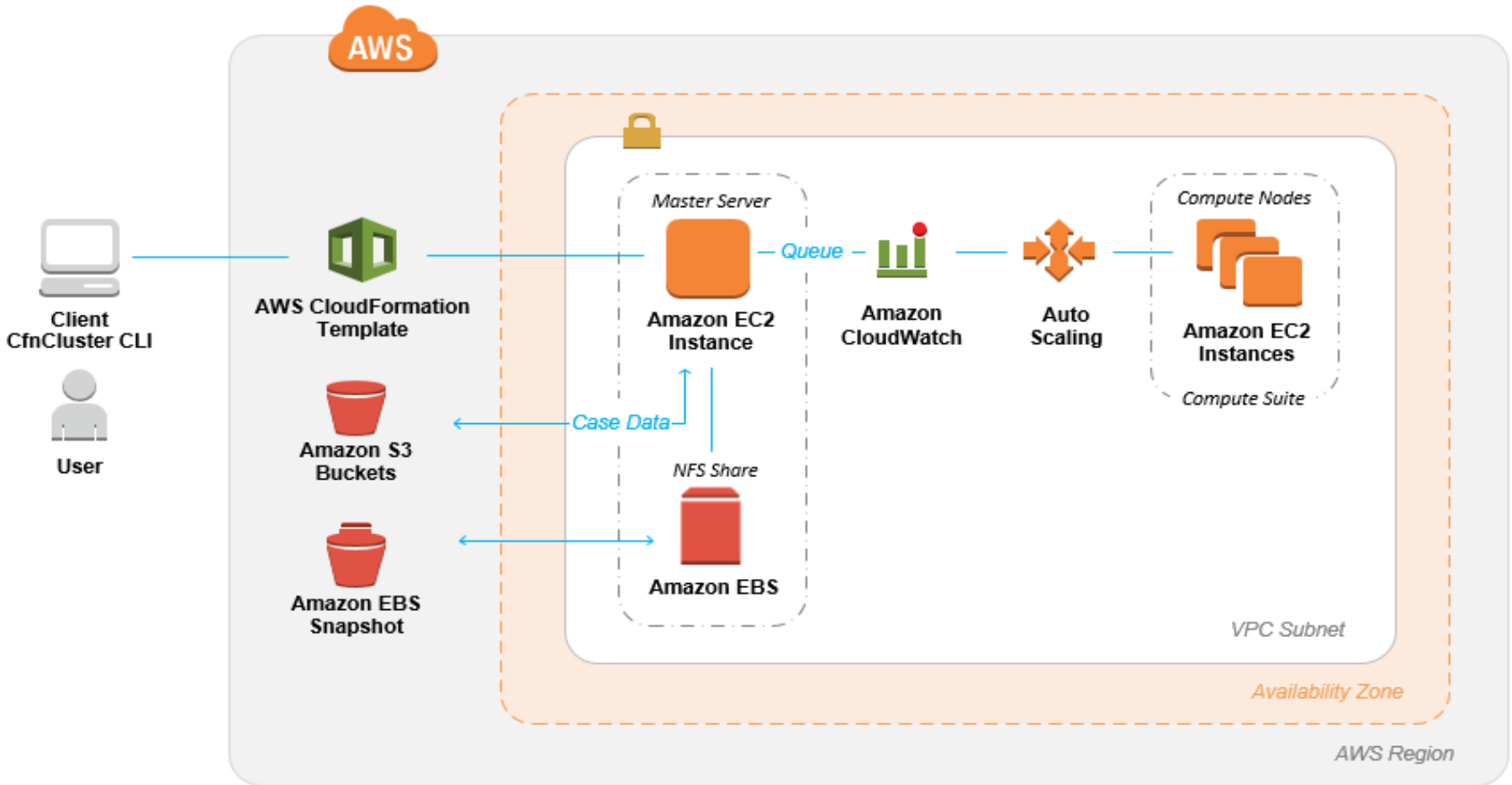
- Cloud Formation Cluster (CFNCluster) was used to create a base HPC cluster
  - Easily repeatable system creation.
  - Create resources from CLI application (Users or Administrators)
  - Dynamic compute node scaling based on number of tasks submitted to the cluster, or static if necessary.
  - Choice of Operating System and Job Scheduler.
  - All instances automatically customized using a ‘post install’ shell script
    - Allows for full customization of the master and compute nodes using standard bash scripting.
      - User accounts
      - Project Specific software
      - File Permissions, etc…
    - Can have different configuration sets for master/client nodes.
  - Ability to customize running cluster from AWS console, or CFNCluster application.
  - Largely a “Fork Lift” migration of the CBPO’s previous HPC environment to AWS.

CBPO Cloud “Hardware”

- 1 Master node – c4.8xlarge – 36 Core + 60GB Memory
  - 200 GB for OS and Home Directories
  - 16TB /Modeling directory
  - 2 Archive disks totaling 22TB local “cool” storage
    - Single disk Max for EC2 Local disk is 16TB
- 0-8 Compute nodes - c4.8xlarge – 36 Core + 60GB Memory
  - 15GB for OS
  - /Home, /Modeling and /opt shared from Master to Compute
- External “Cold” Storage
  - Multiple S3 ‘Buckets’ totaling ~130TB

Deploy an Elastic HPC Cluster

Access on-demand, scalable resources for your High-Performance Computing (HPC) workloads



AWS Reference Architectures

amazon web services

Cloud Computing Costs

- Cost can vary depending on workload
  - Full Price 100% compute running Per Month
  - Full Price 30% compute utilized Scaling On Demand
  - Actual usage over time in CBPO – AWS Console Reports
- Reservations cost less than on-demand systems depending on usage time.
  - Reservations can span multiple systems
    - Reservations are time, not host based.
- Disk storage is not-reservable
  - Use lower priced storage whenever possible to contain costs.
  - Disk storage can quickly eat through a budget.

How we could do it better

- Running your own cluster isn’t the only way.
  - Leverage S3 for source/result data storage
    - Lowers cost of data storage
    - Multiple access methods
      - HTTP/HTTPS
      - API
      - CLI utilities
      - C/Python/Perl/Java libraries.
    - Ability to use this file system directly as part of processing depends on the processing needs/method.
  - AWS Batch
    - Amazon service utilizing Docker instances to perform HPC tasks without a master server.
    - Job Scheduler service used to send tasks to the dockerized compute containers.
    - Minimal changes to existing modeling code.
  - AWS Redshift
    - Compute cluster most normally associated with Python/Spark.
    - Simpler setup
    - Would require a full Model re-write

Cloud Providers and You

- Do you already have access to an HPC System?
- Does your organization already have a cloud provider?
- Can your budget scale with your workload?

Getting started with Amazon Web Services

- Free access to basic resources for 12 Months: https://aws.amazon.com
- Install CFNCluster: https://aws.amazon.com/getting-started/projects/deploy-elastic-hpc-cluster/
- Create or copy a CFNCluster configuration file
- Run CFN Cluster
  - Login via SSH using connection string provided at end up setup.
- Work through the 30Box model, or any code base of your choosing.
- Don’t forget to clean up / shutdown resources when you are finished.