Distributed Computing on GENI:
Hadoop in a Slice

Paul Ruth, pruth@renci.org
RENCI / UNC Chapel Hill

RESEARCH ENGAGEMENT INNOVATION

Outline

- Background
  - Cloud, HPC, and Distributed Computing
- GENI for Distributed Computing
  - Specifying slices for specific applications
  - Previous examples
- Tutorial (Hadoop, Jacks)
  - Create a Hadoop Slice
  - Example application: HDFS file sort

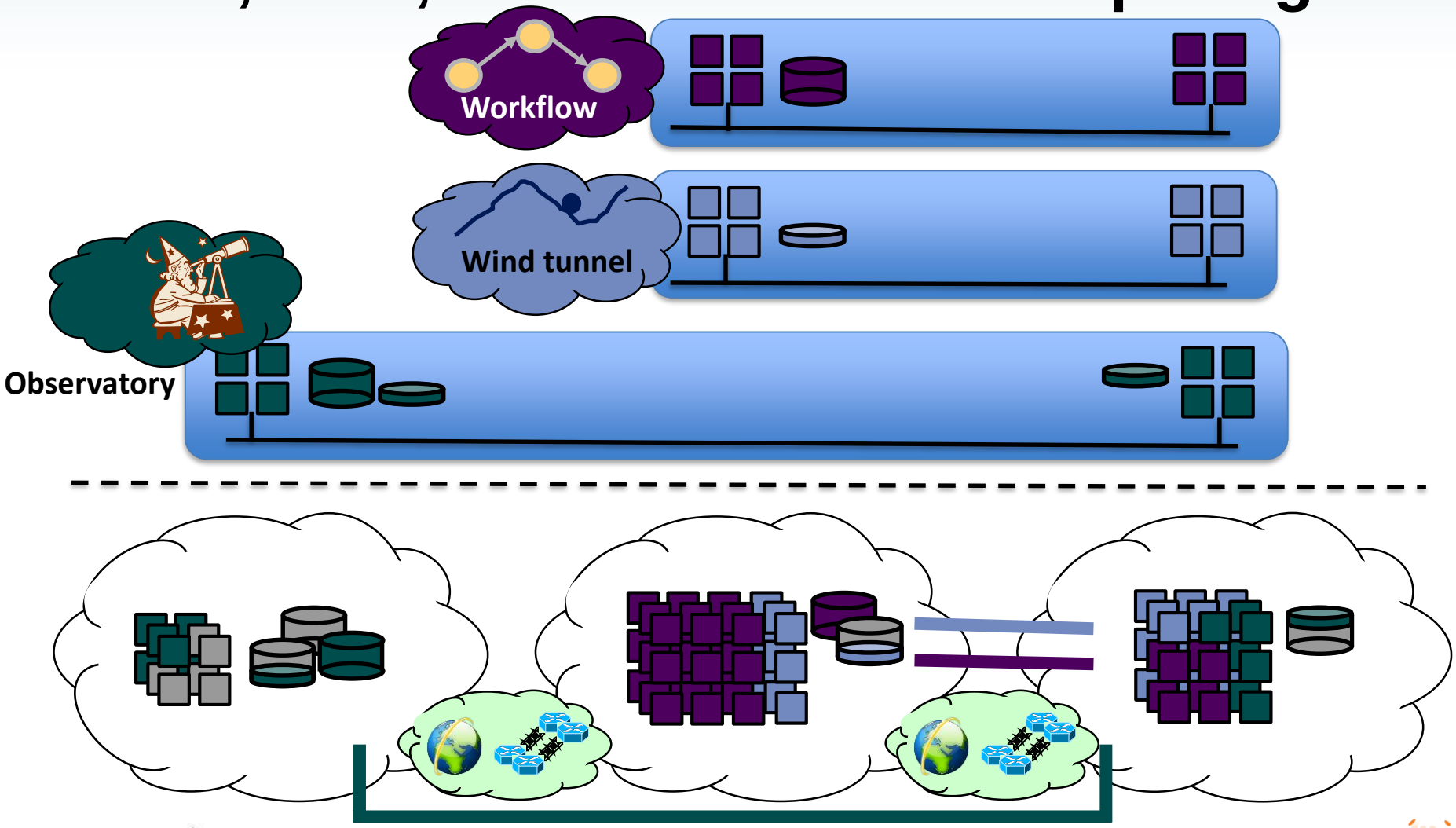
Cloud, HPC, and Distributed Computing

- Combine multiple resources
  - Compute, Network, Storage
- Challenges
  - Distribute tasks
  - Synchronize tasks
  - Handle failure
- Management systems/models
  - PBS/Torque, MPI, HTCondor, Hadoop, Slirm
  - Tightly or loosely coupled
- GENI as a Cloud
  - Allocate compute, network, and storage
  - GENI can allocate WAN resources

Cloud, HPC, and Distributed Computing

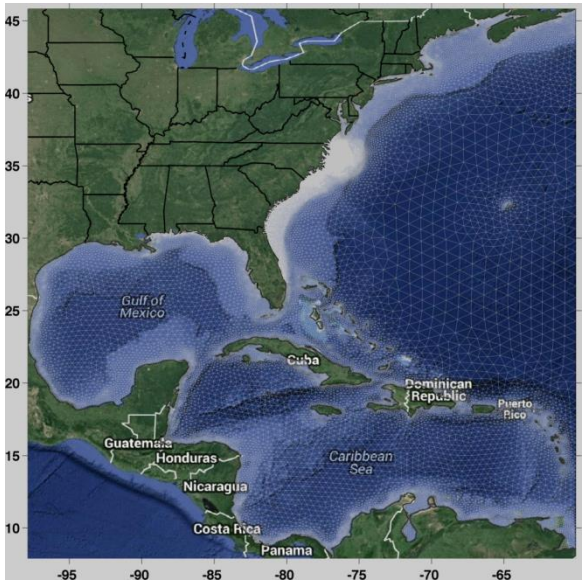
- Why Cloud / Distributed Computing?
  - Performance
    - More CPU, memory, storage, etc.
  - Locality
    - Compute close to the data and/or end users
  - Dynamic resource allocation
    - Only allocate resources when they are needed
  - Reliability
    - Failover to alternate resources

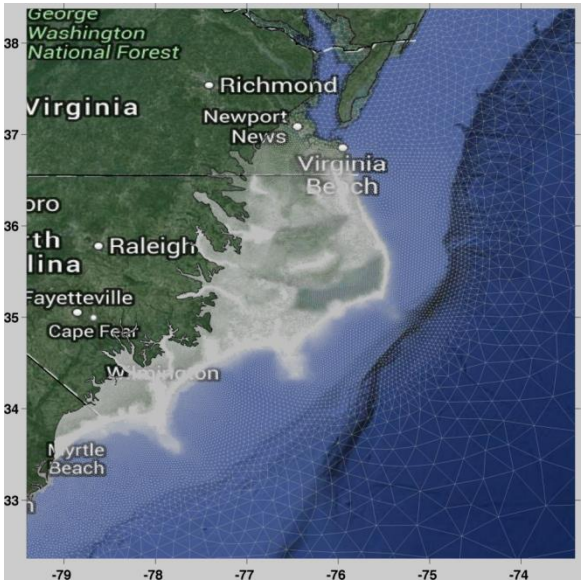
Cloud, HPC, and Distributed Computing

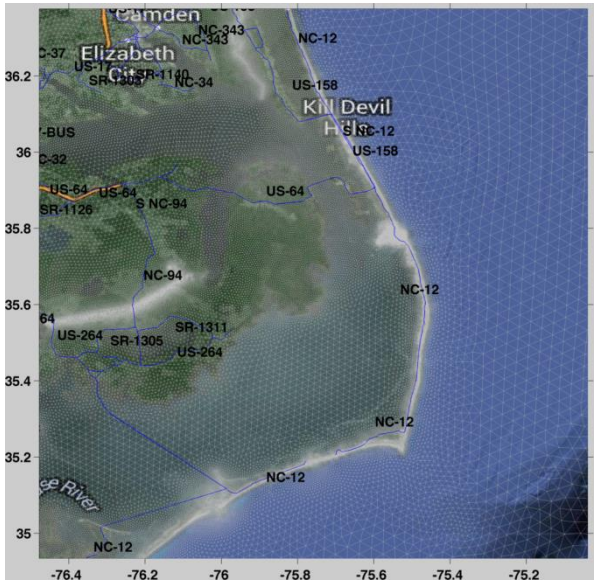


Example: ADCIRC Storm Surge Model

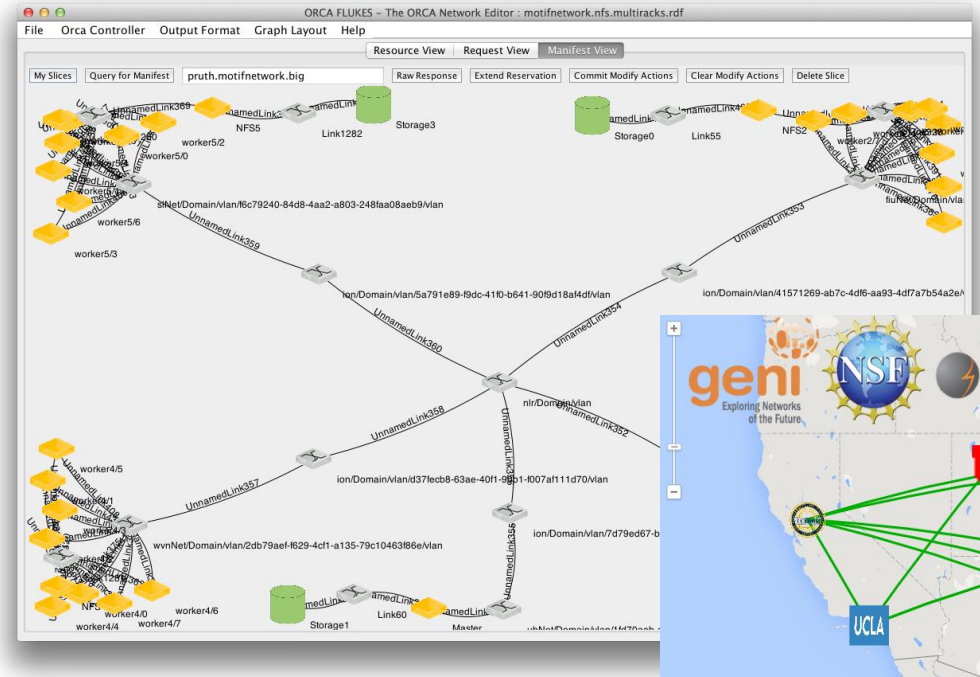
- Finite Element
- Very high spatial resolution (~1.2M triangles)
- Efficient MPI implementation, scales to thousands of cores
- Typically use 256-1024 cores for forecasting applications
- Used for coastal flooding simulations
  - FEMA flood insurance studies, Forecasting systems, Research applications
- Urgent computing
  - 3 hour turnaround requirement



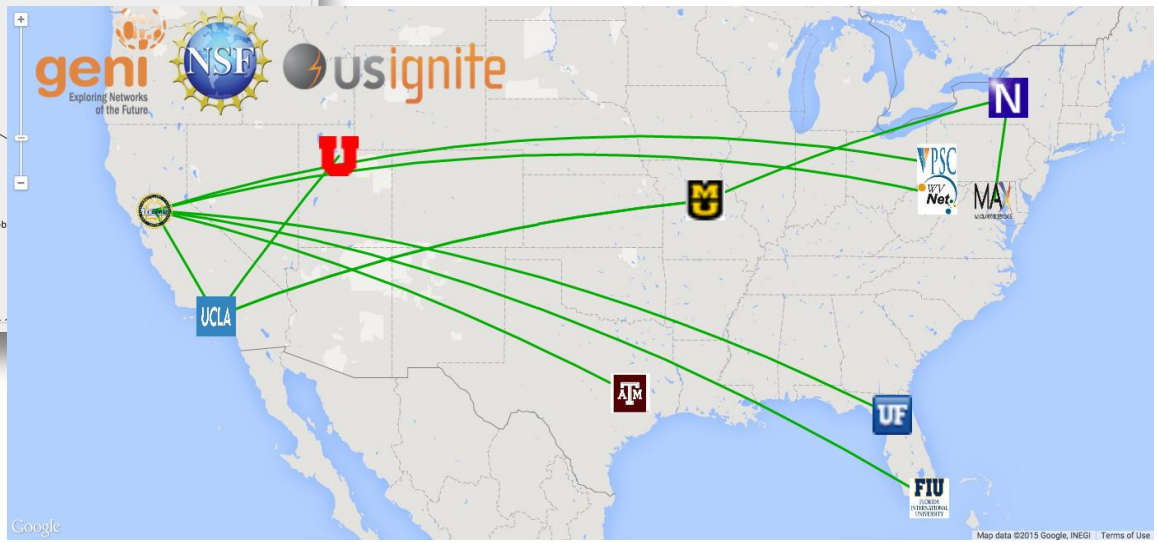




Previous Scaling Efforts



Both GENI Testbeds (ExoGENI & InstaGENI)



Hadoop

- Idea from Google
  - Google File System (GFS)
  - MapReduce
- Implementation by Apache (Open Source)
  - MapReduce framework
  - Hadoop Distributed File System (HDFS)
- Philosophy
  - Distribute data and work on small chunks (map)
  - Combine results (reduce)
- Failure handling
  - Re-map task automatically

Hadoop Distributed File System

- Basic functionality
  - Write/Read files
  - Create/Destroy directories
  - Not POSIX
- Differences from regular file system
  - Distributed
  - Multiple Nodes
  - Replicated data
  - Custom map/reduce functionality
- Why Hadoop file system?
  - Reliable
  - Fast
  - Large files

Tutorial

- Design Slice
  - Create/submit request for resources
- Execute Experiment
  - Observe the properties of the slice
  - Hadoop file system “Hello GENI Word” example
  - Hadoop sort example
- Finish
  - Shut your slice down! Other users need the resources.

Tutorial