
Ben Cotton
Senior Support Engineer
We believe utility access to technical computing power accelerates discovery & invention
The Innovation Bottleneck:

Scientists/Engineers forced to size their work to the infrastructure their organization bought
Our slogan

• Better Answers. Faster.
• We want our customers to get the resources they need when they need them.
Better Answers...
Measure woody biomass on the southern Sahara at 40-50 CM scale

- NASA project in partnership with
  - Intel
  - Amazon Web Services
  - Cycle Computing
Project goals

• Estimate carbon stored in trees and bushes in arid and semi-arid south Sahara
• Establish a baseline for future CO$_2$ studies of the region
Zone of Study

Desert
Sahara

Savanna
Sahel
(Acacia spp., Mimosaceae)

Dry Woodland
Sudan
(Sclerocarya birrea, Anacardiaceae)

Moist Deciduous Forest
Guinea
(Kigelia africana, Bignoniaceae)

Rainforest
Congo
(Aucoumea klaineana, Burseraceae)

Africa Biomes
The input data

- Images collected from satellites
- ~20 terabytes total
The workflow

• Pleasantly parallel
• Each task takes 20-30 minutes
• 0.5 million CPU hours total
The workflow

• Tasks have two parts
  – Orthorectification and cloud detection
  – Feature detection
  • Uses 2-20 GB of RAM
AWS setup

- Spot instances
- C3 and M3 instance families
- Data staged into S3
Job Submission

- DataMan uploads data from local Lustre filer to S3
- When transfers complete, DataMan creates a record in CycleCloud
- CycleCloud batches records and builds HTCondor submit files
Job Submission

• Easy for the scientist
What’s next?

- Proof-of-concept is wrapping up
- Operational project expected to take approximately 1 month
Faster....
Improve hard drive design

- HGST runs an in-house drive head simulation suite
- In-house grid engine cluster runs the simulations in 30 days
- ~620K compute hours
We can make this faster!

• On Wednesday: “Hey, guys! Can we have this done by this weekend?”
We can make this faster!

- Un-batch the sweeps: 1.1M jobs
- 5-10 minute per-job runtime
Enter the cloud

- Used 10 AWS availability zones, spanning 3 regions
- Spot instances from the m3, c3, and r3 families
Pool setup

- One pool per availability zone
- Two schedulers per pool
How we did it

• CycleCloud autoscaled multiple instance types and multiple availability zones
• CycleServer spread jobs across multiple schedulers/pools based on load
• Used Amazon S3 instead of a shared filer
HTCondor configuration

• Very little!
• NEGOTIATOR_CYCLE_DELAY and NEGOTIATOR_INTERVAL set to 1
• CLAIM_WORKLIFE set to 1 hour
• *_QUERY_WORKERS set to 10
HTCondor configuration

- SHADOW_WORKLIFE set to 1 hour
- JOB_START_COUNT set to 100
- Disabled authentication
We did it!

- Went from 0 to 50k cores in 23 minutes
- Peaked at ~ 70K cores from 5689 instances
- Simulation completed in 8 hours
- Infrastructure cost: $5,594
Where do we go from here?
Better-er answers. Faster-er.
If you build it, they will come

• Large financial institution actuarial modeling
  – Originally just wanted to do Federal Reserve stress tests
  – Then month-end actuarial runs
  – Now regularly use 8000 cores in AWS
Coming concerns

- Data movement
- Multi-provider cloud usage
- Seamless burst to cloud
We write software to do this...

Cycle Computing easily orchestrates workloads and data access to local and Cloud technical computing

- Scales from 100 - 100,000’s of cores
- Handles errors, reliability
- Schedules data movement
- Secures, encrypts and audits
- Provides reporting and chargeback
- Automates spot bidding
- Supports Enterprise operations
Does this resonate with you?

We’re hiring software developers, HPC engineers, sales, etc.

jobs@
cyclecomputing.com