A thousand and one file

(Or: Data-Parallel Computations at UZH)

Riccardo Murri, <riccardo.murri@uzh.ch>

DSI Data Day, July 4 2018

A typical problem...

From: some.one@uzh.ch
Subject: simulations

Hello :)

I'm a postdoc in the department of *[...]*.
I have simulations written in MATLAB and Python 2.7 scripts. My simulation is computationally intensive.

Therefore I would like to use possibly hundreds of CPUs, but each run may only take minutes to a few hours.

Many thanks,
Some One

Assumptions

1. Data to process is scattered
   over a (large) number of files.
2. Each file can be processed independently.

Each file can be processed independently.

Computations can be distributed over several interconnected computers.

("pleasantly parallel")

Local computation

Wrap your computations in a "`for`" loop and let your computer grind it for days.

- \+ Easiest solution
- – Limited in processing power and disk storage
- – No easy path to share and re-use

Only sensible for one-off very short computations.

Batch-queuing cluster

Enqueue all jobs and let the cluster scheduler take care of them.

- \+ Easy to implement “pleasantly parallel” workloads
  - Easy things are easy – complex things are possible but not necessarily easy
  - Traditional tools that make this easy (`bash`, GUI portals) quickly become a mess to maintain and change
- \+ Shared filesystem (hence, no data distribution issues)
  - Watch out for performance and scaling issues, though!
- – Environment may be unfriendly to GUI users
- – Shared resource, so need to stick to policy
  - Turnaround time might not be what you want/need
  - Other users may be affected if you screw up

Virtual cluster on Science Cloud

Deploy your own batch-queuing cluster on Science Cloud.

- \+ Easy to implement “pleasantly parallel” workloads
- \+ Shared filesystem (hence, no data distribution issues)
- \+ ~~Shared resource, so need to stick to policy~~
- \+ Build process of cluster completely reproducible
  - Can rebuild the same environment on public cloud providers
- – Building a cluster takes time
- – Size-limited compared to UZH-wide cluster
- – Science Cloud not built for performance
- – Environment may be unfriendly to GUI users

See also: https://elasticluster.readthedocs.io/

Dynamic provisioning on Science Cloud

Use a GC3Pie driver script to provision resources and run computations.

- \+ Easy cases are easy, complex cases are not so difficult
- \+ Building and running (almost) completely reproducible
  - Can redo the same on public cloud providers
- – Much harder to write a driver script *(but we're here to help!)*
- – Largest overhead: only sensible when each computation run takes several minutes.
- – Environment totally unfriendly to GUI users

See also: https://gc3pie.readthedocs.io/

Build your own . . .

Science Cloud lets you build your own parallel processing infrastructure, using virtualized computing components, and tailor it to your needs.

This is much, *much*, ***much*** harder than any other solution — *but we're here to help!*

(Hardly in the scope of "processing a 1'000 files", though.)

Thanks!

(Any questions?)