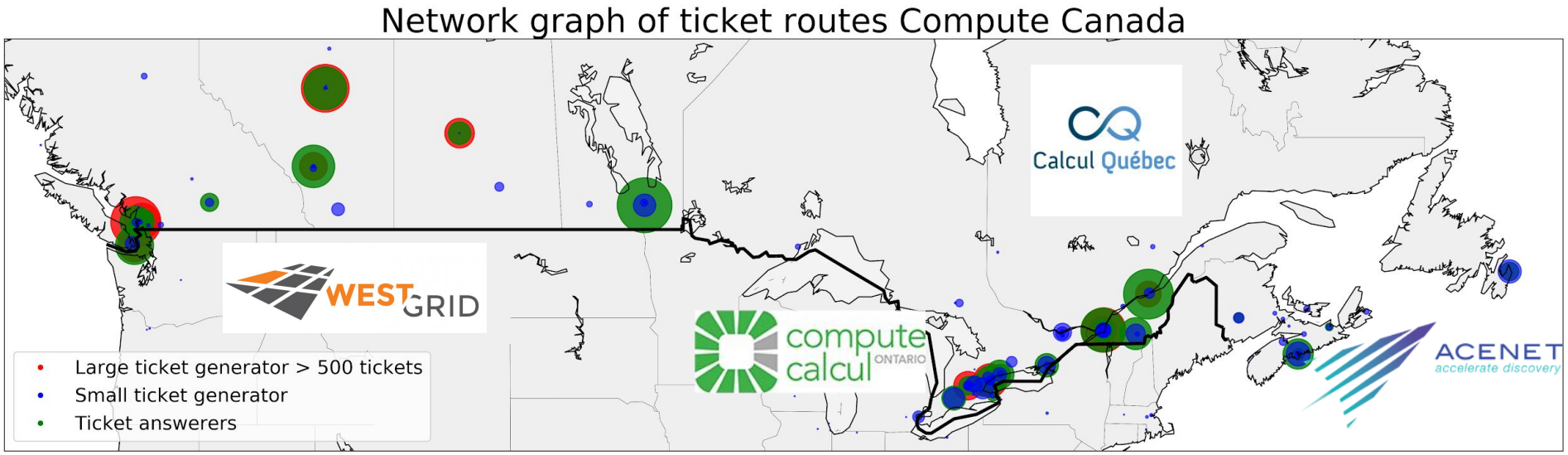
computecanada

Providing A Unified User Environment for Canada's National Advanced Computing Centers

Maxime Boissonneault, Bart Oldeman, Ryan Taylor
on behalf of
Compute Canada Research Support National Team

This presentation:
https://rebrand.ly/computecanada-sc19

Compute Canada



- 4 regional consortia
- 35 member institutions
- ~200 technical staff
- ~15,000 user accounts
  - 20% growth per year

System	Type	Network	Production
Arbutus	Cloud	10 GbE	2016 H2
Cedar	General	OPA	2017 H1
Graham	General	EDR IB	2017 H1
Niagara	Large MPI	EDR IB	2018 H1
Béluga	General	EDR IB	2019 H1

5 major national systems
~15 legacy systems
200K cores, 22 PF
70 PB disk, 180 PB tape

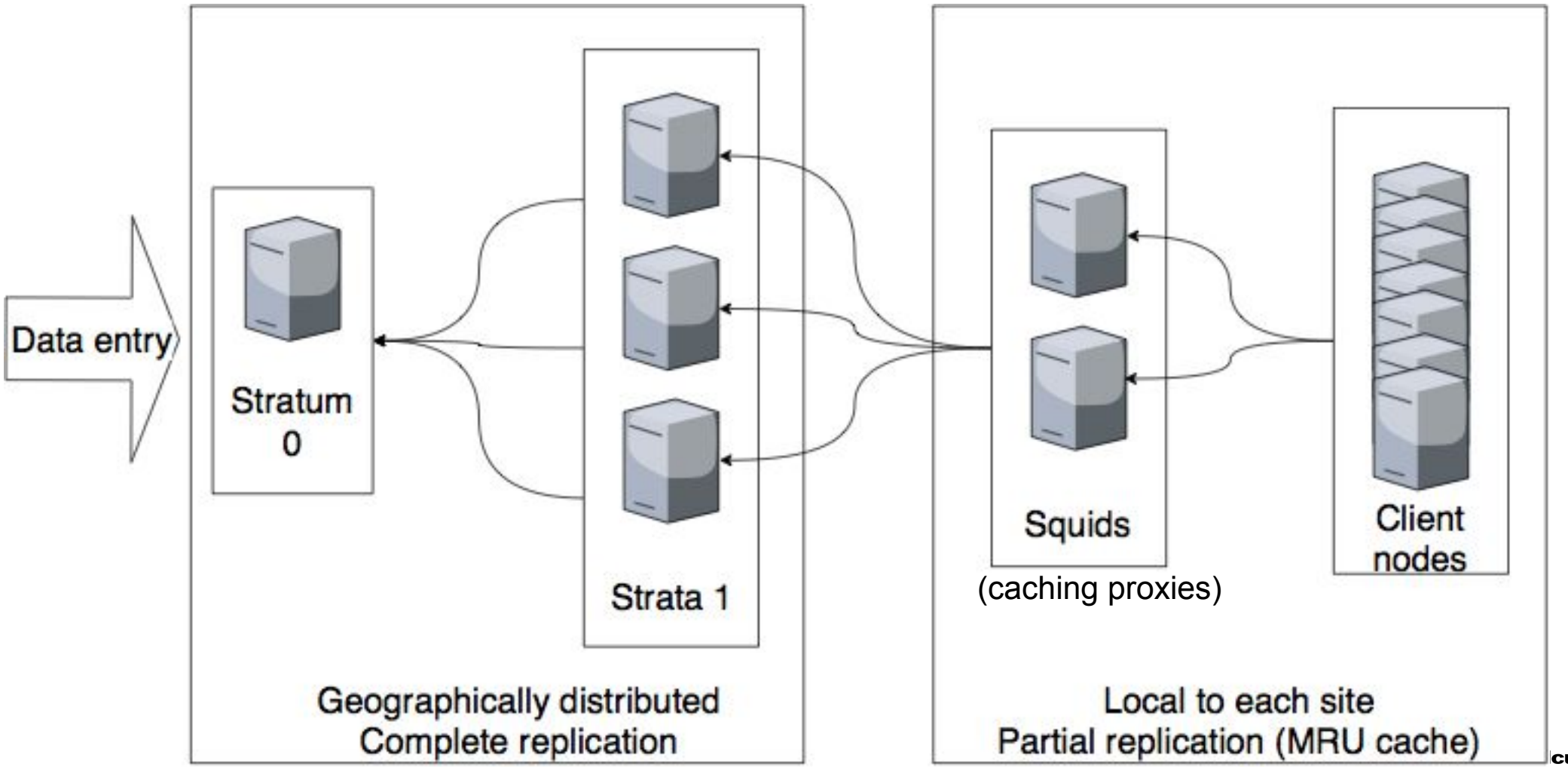
compute canada | calcul canada

Goal

Users should be presented with an interface that is as **consistent** and **easy to use** as possible across **all sites**. It should also offer **optimal performance**.

1. All software should be accessible on every site, reliably and performantly.
2. Software should be independent from the underlying OS stack.
3. Software installation should be tracked and reproducible via automation.
4. The user interface should make it easy to use a large and evolving software stack.

CVMFS content delivery



Software: design overview

Easybuild layer: modules for Intel, PGI, OpenMPI, CUDA, MKL, high-level applications.
Multiple architectures (sse3, avx, avx2, avx512)
`/cvmfs/soft.computecanada.ca/easybuild/{modules,software}/2017`

Nix layer: GNU libc, autotools, make, bash, cat, ls, awk, grep, etc.
`module nixpkgs/16.09 => $EBROOTNIXPKGS=`
`/cvmfs/soft.computecanada.ca/nix/var/nix/profiles/16.09`

Gray area: Slurm, Lustre client libraries, IB/OmniPath/InfiniPath client libraries (all dependencies of OpenMPI). In Nix layer, but can be overridden using PATH & LD_LIBRARY_PATH.

OS kernel, daemons, drivers, libcuda, anything privileged (e.g. the sudo command): always local.
Some legally restricted software too (VASP)

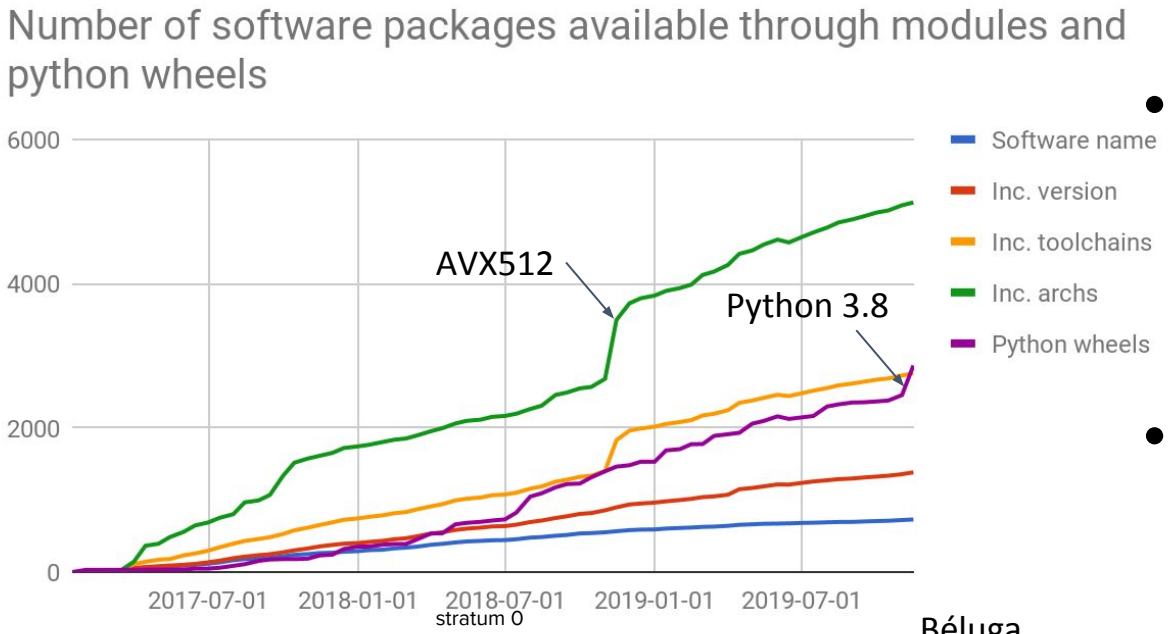
Compute Canada Software Stack

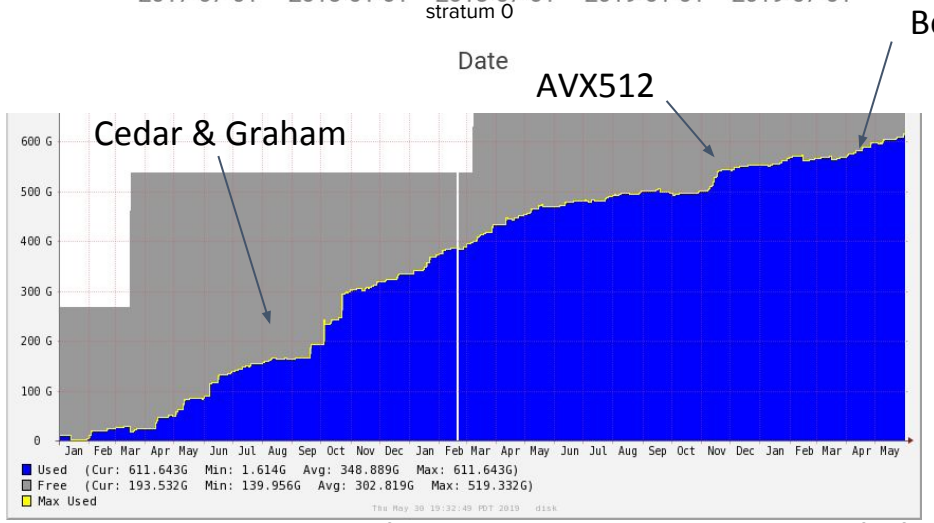
Available software

730+ scientific applications

5,000+ permutations of version/arch/toolchain

Type	Modules
AI	5
Bioinformatics	239
Chemistry	63
Data	19
Geo/Earth	23
Mathematics	82
MPI libraries	7
Physics	48
Various tools	176
Visualisation	28
Misc	38





- Two major new clusters with Skylake CPUs
- Built new modules with AVX512 for most packages
- High deduplication
- Further details

Documentation

- List of modules
  - https://docs.computecanada.ca/wiki/Available_software
- List of Python wheels
  - https://docs.computecanada.ca/wiki/Available_wheels
- Mounting our software stack
  - https://docs.computecanada.ca/wiki/Accessing_CVMFS

Cluster stack on Windows ?!

Tweet épinglé

Maxime Boissonneault
@mboisso

If I told you that I want to use my HPC cluster's software environment on my Windows laptop, how crazy would you say I am ? Discover the answer during my talk at @PEARC_19

Traduire le Tweet

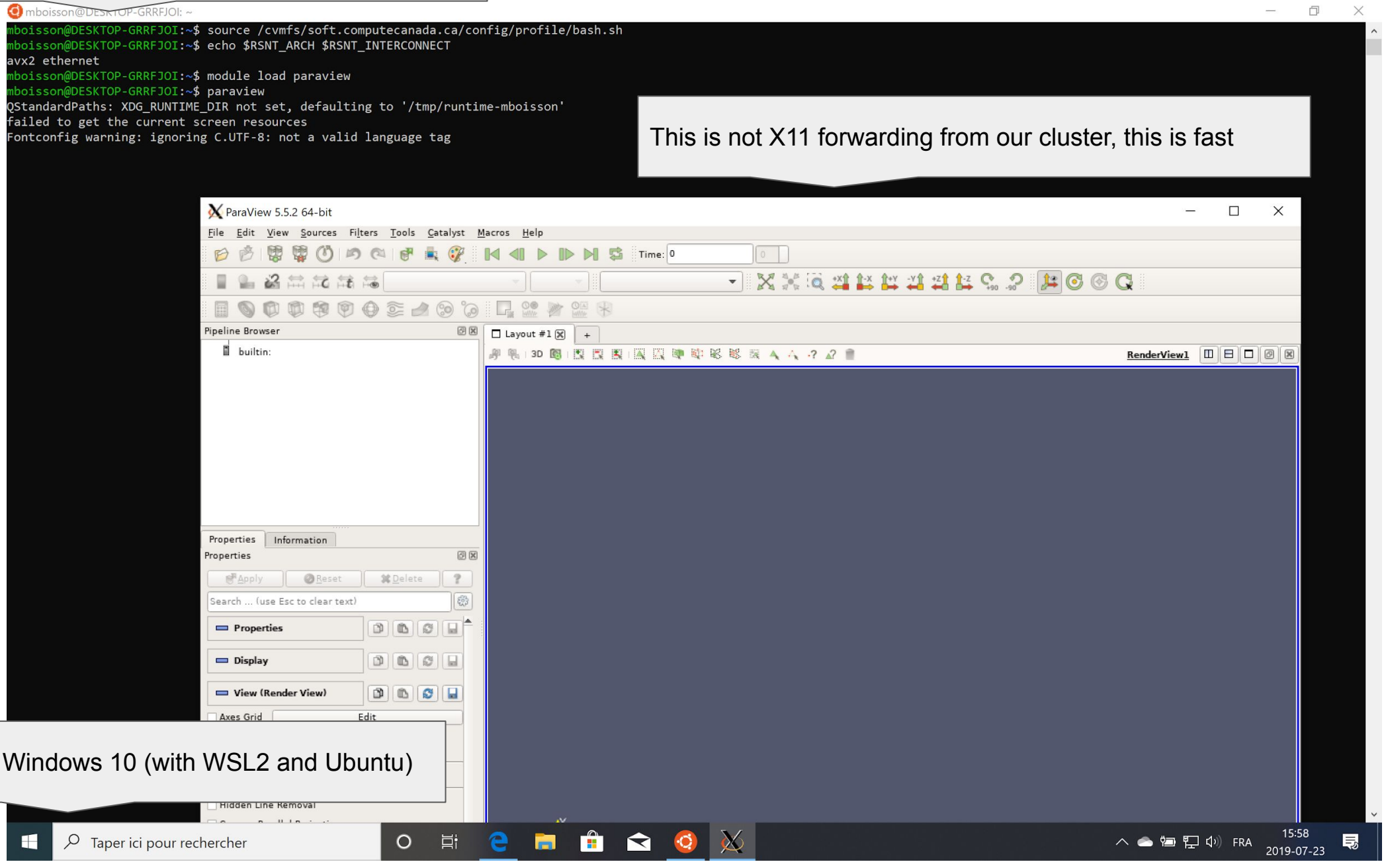
Are you out of your mind?	**60%**
That's a cake walk	40%

25 votes · Résultats finaux

Without reinstalling packages
Without X11 forwarding
Without sshfs

https://rebrand.ly/computecanada-pearc19

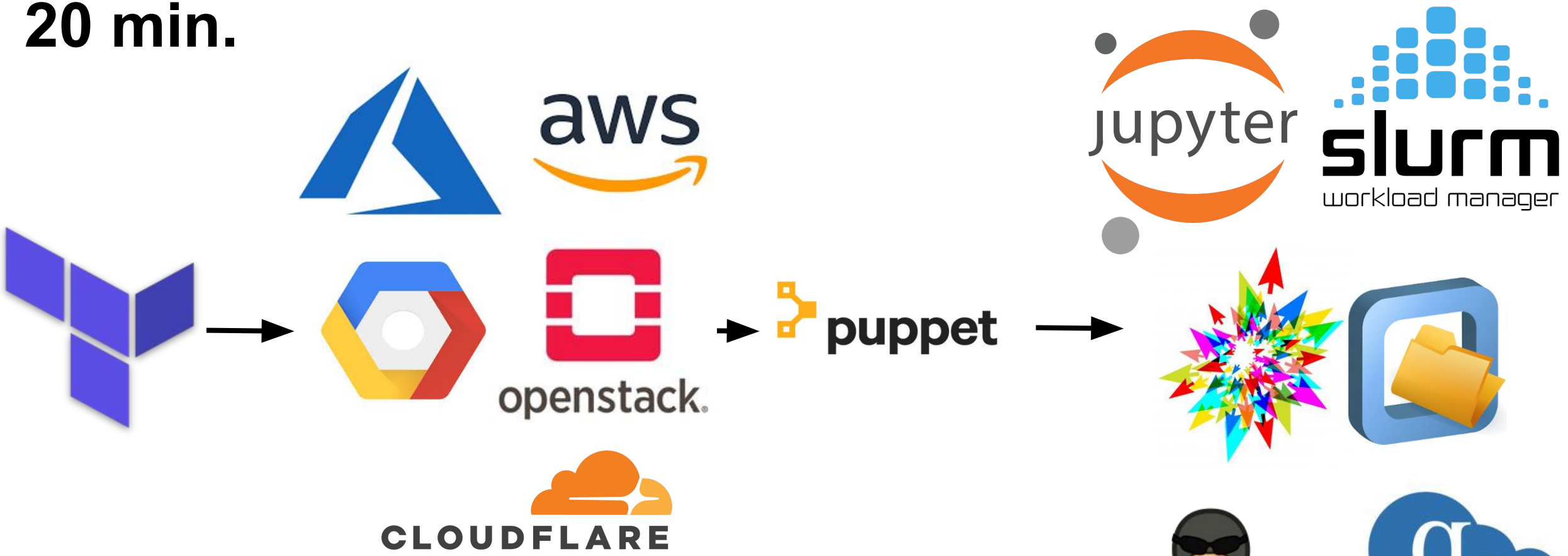
This is not a remote server, this is my laptop



This is not X11 forwarding from our cluster, this is fast

This is Windows 10 (with WSL2 and Ubuntu)

Magic Castle Replicates a Compute Canada Cluster in **20 min.**

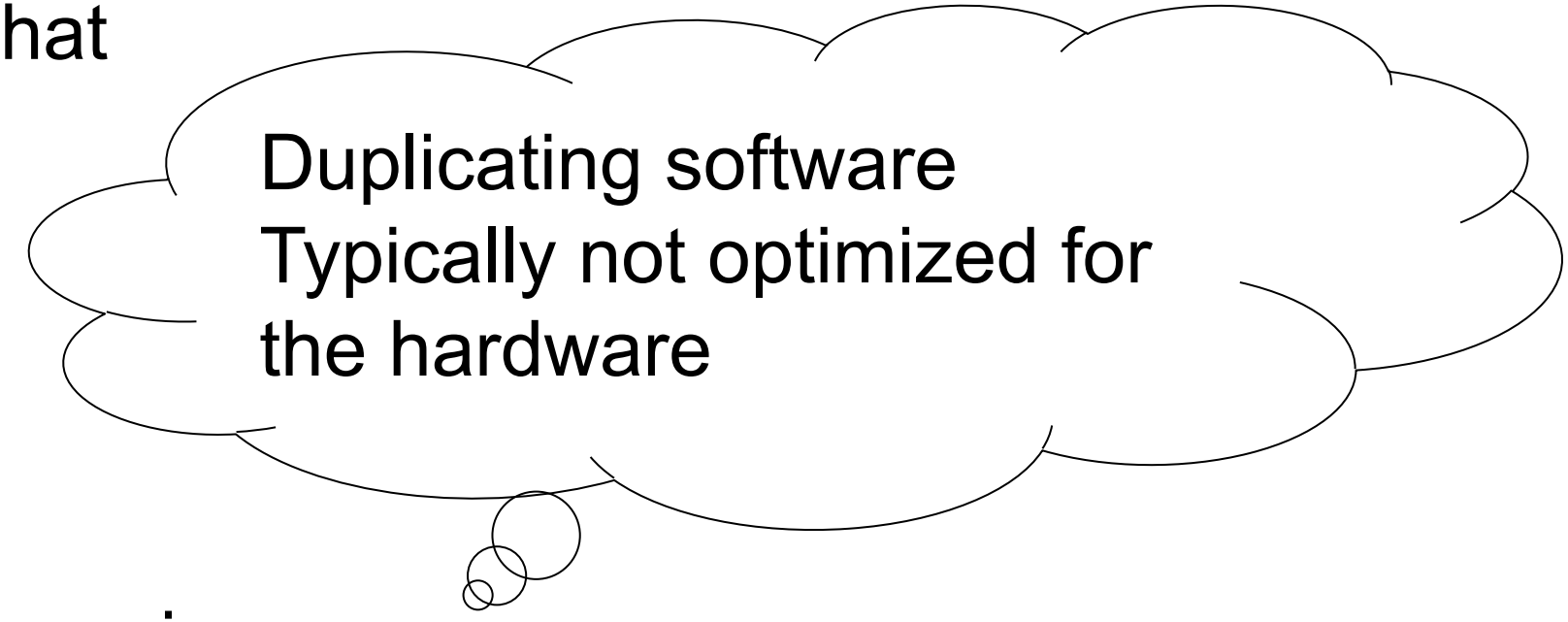


https://github.com/ComputeCanada/magic_castle

Extra slides

Python vs Anaconda

- Python is bad at packaging
  - Anaconda fixes that



- Really ??
  - conda install gcc
  - conda install openmpi
  - conda install cudatoolkit

Solution is Python Wheels

```
$ ls /cvmfs/soft.computecanada.ca/custom/python/wheelhouse/*/* | wc -w
2865
$ avail_wheels tensorflow_cpu
name            version    build           python    arch
--------------  ---------  --------------  --------  ------
tensorflow_cpu  2.0.0      computecanada   cp37      generic
$ avail_wheels tensorflow_gpu
name            version    build           python    arch
--------------  ---------  --------------  --------  ------
tensorflow_gpu  2.0.0      computecanada   cp37      generic
```

- https://docs.computecanada.ca/wiki/Available_wheels
- https://github.com/ComputeCanada/wheels_builder

Module usage dashboard