OSG and GPUs: A tale of two use cases

CHEP 2018

July 12 2018

Edgar Fajardo on behalf of IceCube and Open Science Grid

Mats Rynge, Derek Weitzel, Gonzalo Merino, David Schultz,
Vladimir Brik and Heath Skarlupka

Two different GPU use cases

- IceCube has been using GPU's for many years. It has a well curated software stack to use GPU's using OpenCL.

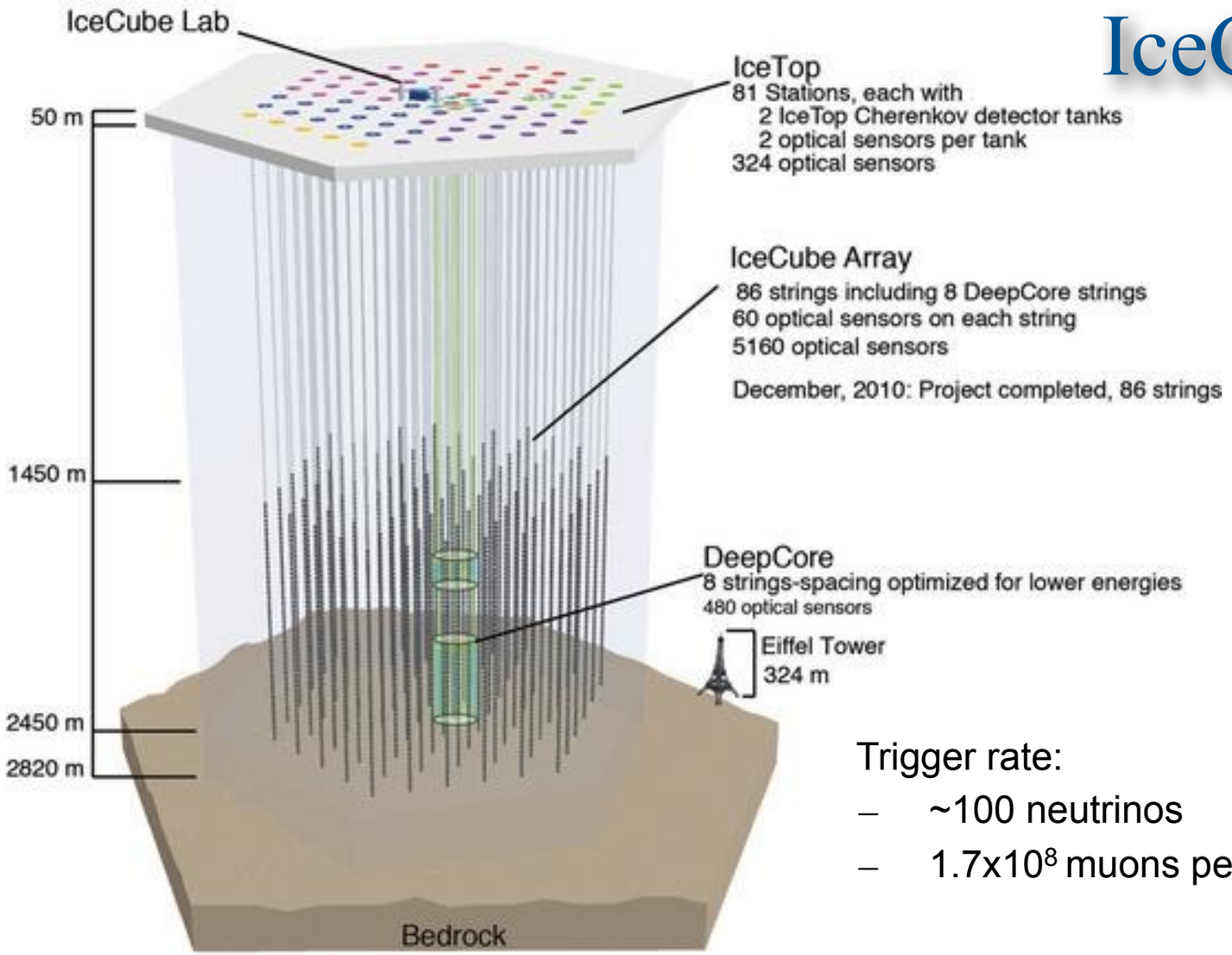
- OSG GPU users are fairly new mostly with the uprising of Deep Learning. Many different users coming from many different sciences. Each users expects the GPU stack (mainly CUDA) to be already setup for them.

- Both use the same underlying software for resources provisioning: GlideInWMS.

- The lessons learned from the OSG/IceCube Grid GPU infrastructure were easily mapped to other VO's (mainly CMS).

IceCube

Thanks to David Schultz for this set of slides

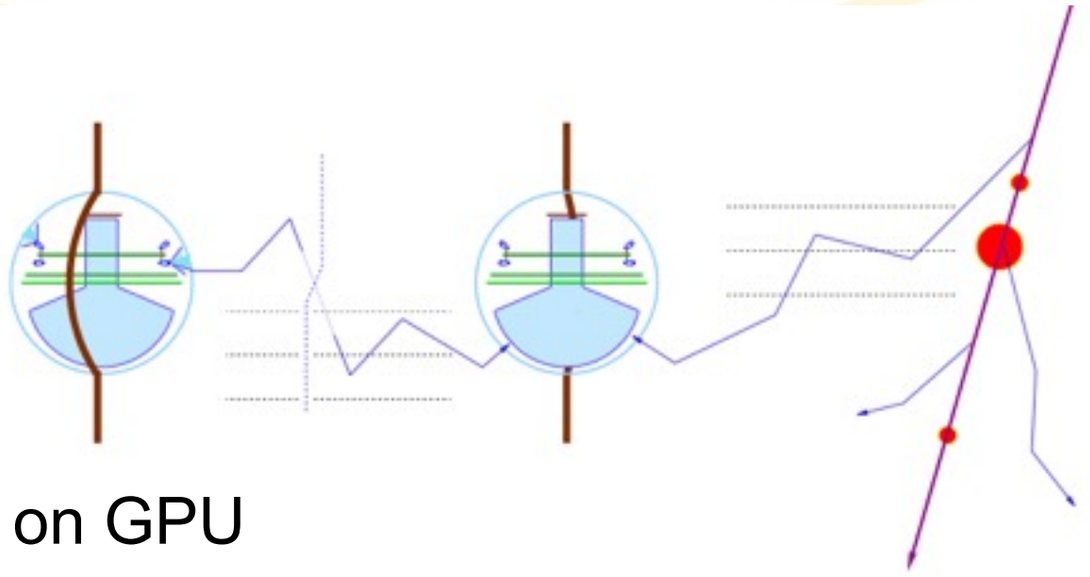


IceCube

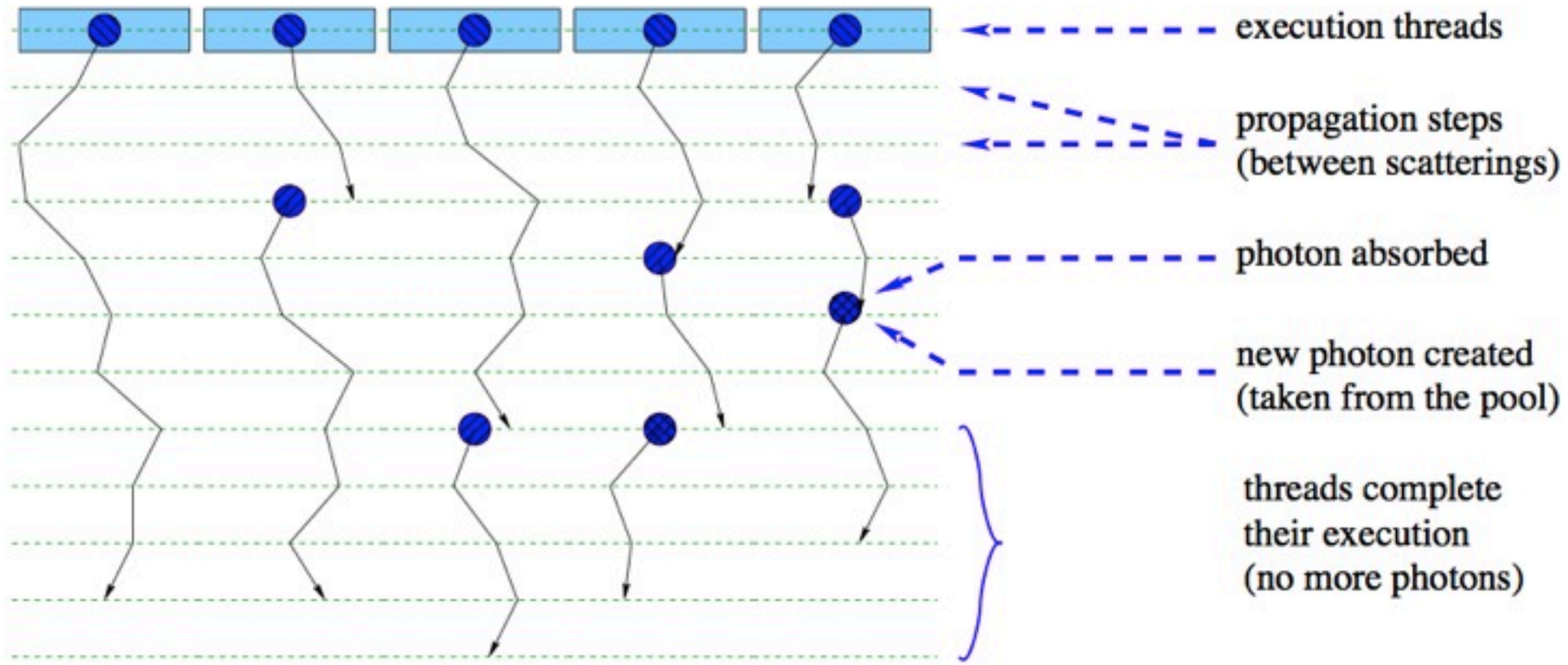
Trigger rate:

- ~100 neutrinos
- 1.7×10^8 muons per day

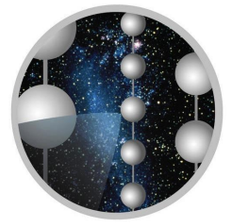
Photon Propagation

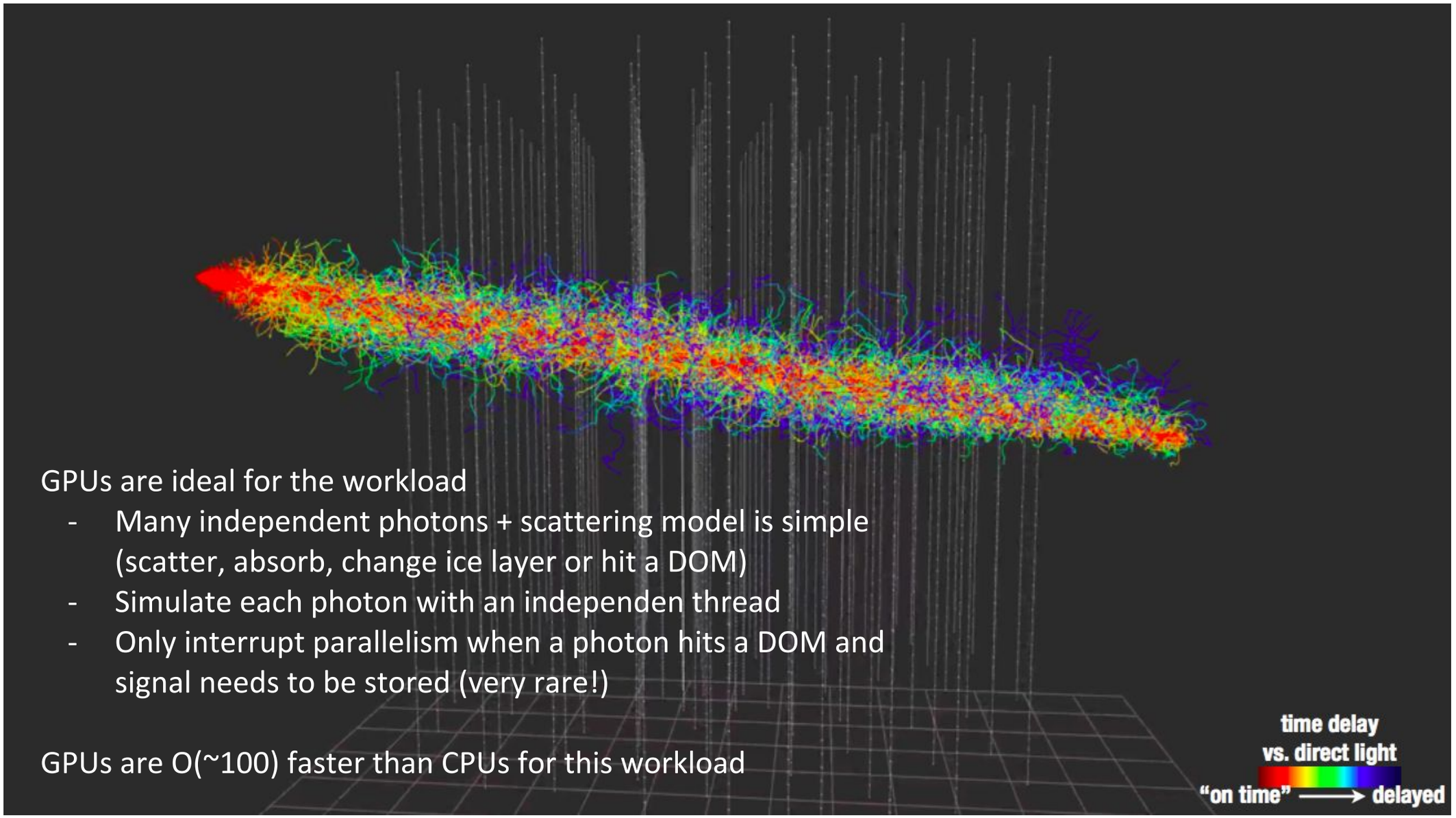


Direct photon propagation on GPU



GPUs: direct photon propagation





GPUs are ideal for the workload

- Many independent photons + scattering model is simple (scatter, absorb, change ice layer or hit a DOM)
- Simulate each photon with an independen thread
- Only interrupt parallelism when a photon hits a DOM and signal needs to be stored (very rare!)

GPUs are O(~100) faster than CPUs for this workload

OSG Users

Machine Learning with Clinical Data

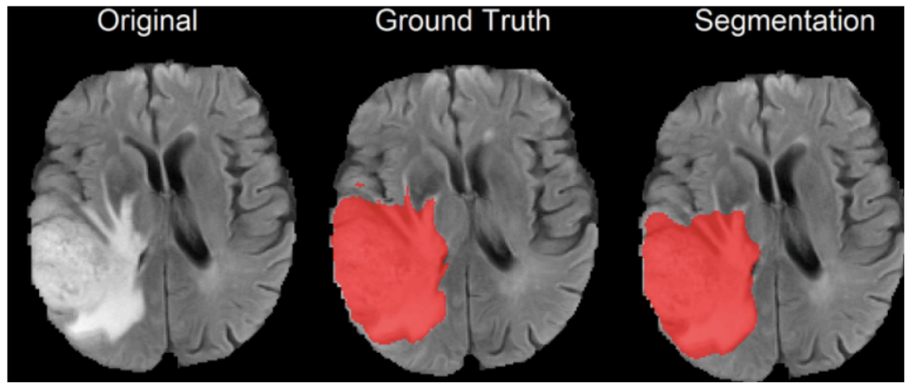
Rami Vanguri

Biomedical Informatics / Data Science Institute
Columbia University

OSG All Hands Meeting
March 20, 2018
University of Utah, Salt Lake City, UT

Clinical

- Segmentation is essential task during radiotherapy planning
- Automatic Brain Tumor Detection and Segmentation Using U-Net Based Fully Convolutional Networks
- Hao Dong, Guang Yang, Fangde Liu, Yuanhan Mo, Yike Guo



COLUMBIA UNIVERSITY IN THE CITY OF NEW YORK

Data Science Institute

IceCube South Pole Neutrino Observatory

UC San Diego

Why the sudden increase and interest in Deep Learning?

Academia and industry develop and maintained libraries for Deep Learning.

- Tensorflow (Google)
- Caffe && Caffee2 (UC Berkeley & Facebook)
- Keras
- Theano

The other buzzword: Big Data

The other pice of the puzzle: GPU's

- It turns out that deep learning algorithms do really well in GPU's.
- They also get (x1, x3) performance increased by running in more than one GPU.
- Some of the analysis problems in the field, fit well for ML with GPU's.

How to make GPU work in the Grid?

1. Submit a job that has “Request_gpus = 1”

2. The Frontend tells the factory to submit a pilot to some of the GPU entries. HTCondor-CE or CREAM

3. The job lands on a node. Starts HTCondor Startd and advertises “Gpus = 1”.

4. Job lands on the node and has correct “CUDA_DEVICE_NUMBER”.

Easy right? There is some more to it.

Once a pilot lands on a worker node

- Wrapper runs condor_gpu_discovery

Binary provided by HTCondor and shipped with every pilot

- Updates the pilot Ads: CUDACapability, CUDADeviceName depending on what is found in the worker node, and the CUDA_VISIBLE_DEVICES assigned to the pilot.

- Number of GPU's per pilot is determined by the site and configured at the factory level. Most sites prefer one GPU per pilot to increase turnaround

Another piece of the puzzle: Containers

I will mainly focus on Singularity (since it is the one mostly used in OSG) but most of it translates to docker.

Why to use containers at all?

- The train has left, most VO's are using containers and sites are already assuming that is the case for their EL6|EL7 upgrades.

- It allows for the usage of Deep Learning packages like Tensorflow, Keras which do not natively install on RedHat.

Once a pilot lands on a worker node

- All jobs are run inside a wrapper script (behind users back) that checks for the "+SingularityImage" jobAd and runs the job inside that image. This functionality of the wrappers is being merged into GlideInWMs on future version.

- For this to work the wrapper script for GPU always binds to the drivers: "singularity --bind /usr/lib64/nvidia:/host-libs"

- OpenCL support was also more tricky, as /etc/OpenCL/vendors/nvidia.icd needed to be bind-mounted or already installed in the image.

- This causes troubles when sites do not have the NVIDIA libraries on the standard location "/usr/lib64/nvidia". Even if they do if the CUDA version does not match the image it can fail.

- This will be mitigated once all sites move to Singularity 2.5.1 and the "nv" flag can be used.

Flow

There is no RPM for Tensorflow.
The whole installation is made for Ubuntu

1. Check tensorflow installation page

2. Create Image

3. Push to CVMFS

Example:

- You can run it in any singularity + CVMFS node

```
singularity shell --bind /cvmfs --bind /usr/lib64/nvidia:/host-libs /cvmfs/
singularity.opensciencegrid.org/opensciencegrid/tensorflow-gpu:latest/
Singularity: Invoking an interactive shell within container...

cuser1@cgpu-1:~$ python3
Python 3.5.2 (default, Sep 14 2017, 22:51:06)
[GCC 5.4.0 20160609] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow
```

Necessary to use GPU

Tensorflow Singularity image maintained by Mats Rynge

Result:

Happy User

Not so fast

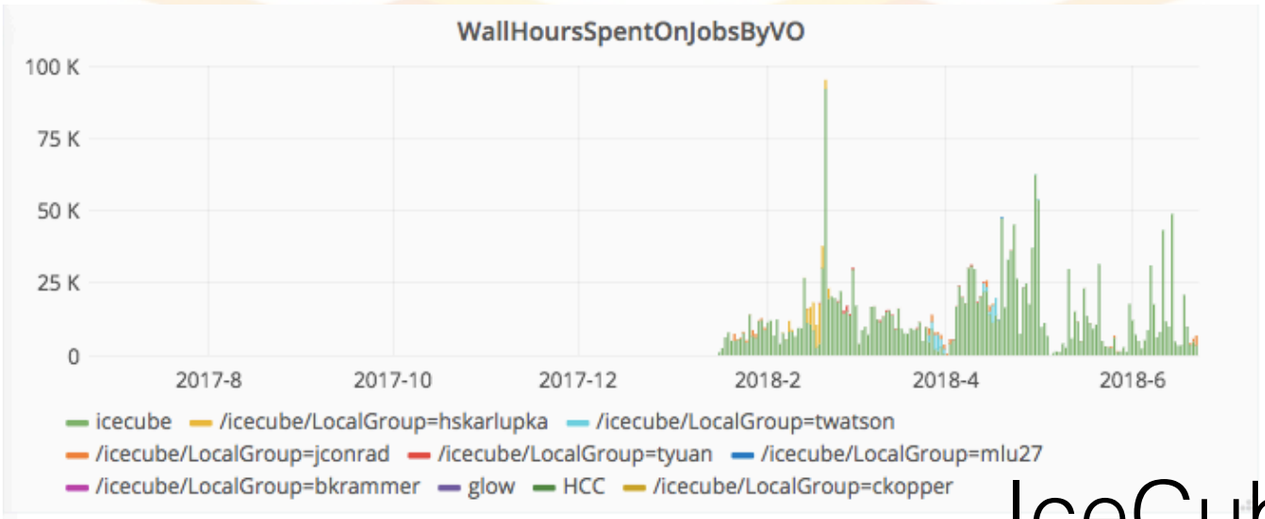
It works until the image drivers or the hardware were out of sync

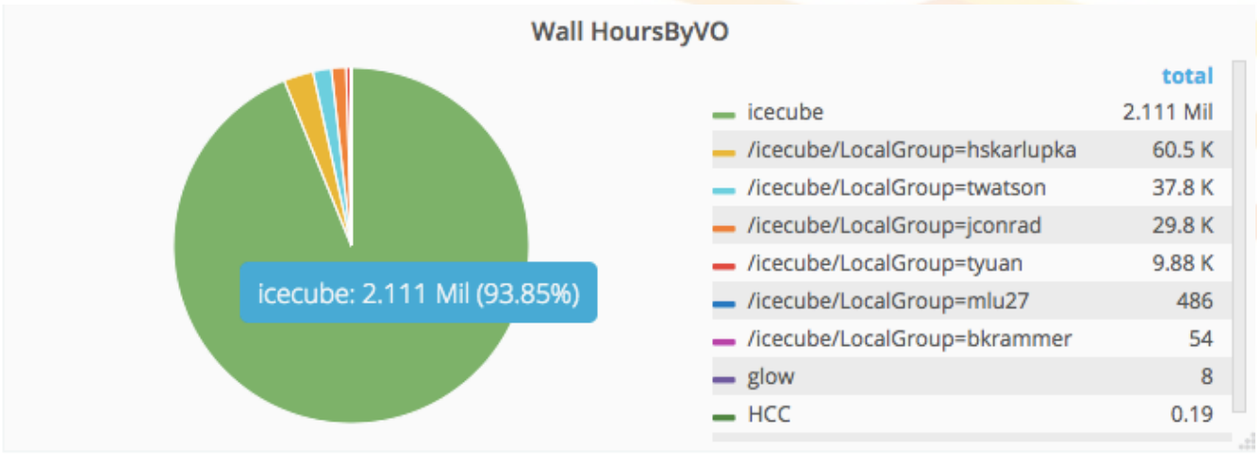
Current GPU's deployed on the Grid

- Four sites offering GPU's in US: Nebraska, UCSD, Syracuse, Vanderbilt (CMS only) with Factory Entries
- EU (IceCube Only): Manchester, London_QMUL
- All behind CE: HTCondor-CE or Cream
- OSG accounting (GRACC) has been updated and the probes to account GPU time. See Brian's earlier talk.
- HTCondor 8.8 includes reporting of GPU Wall Time as they do with CPU.

Open Science Grid

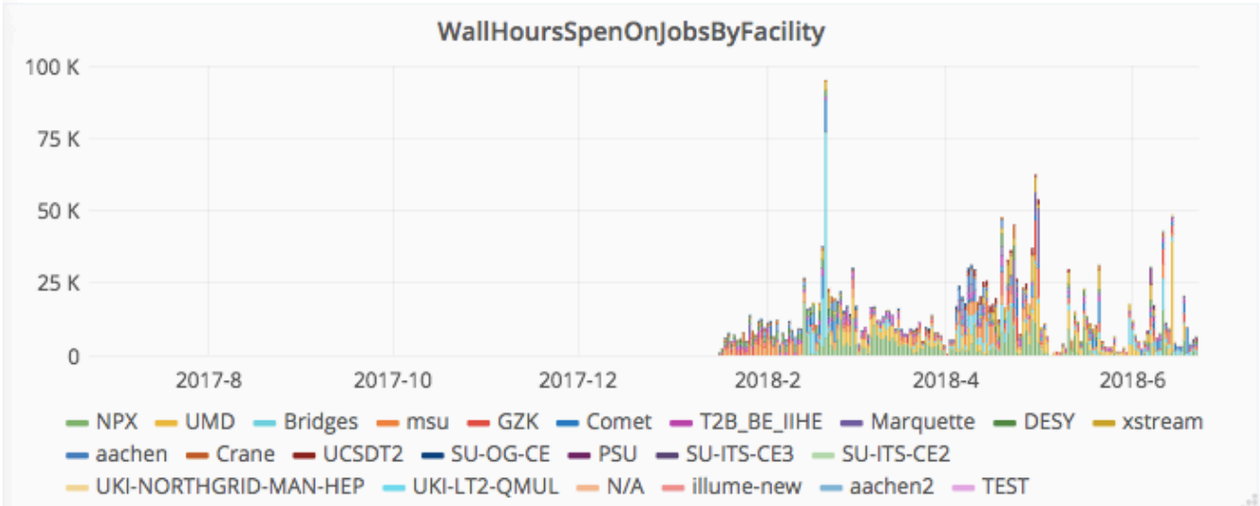
GPU Usage

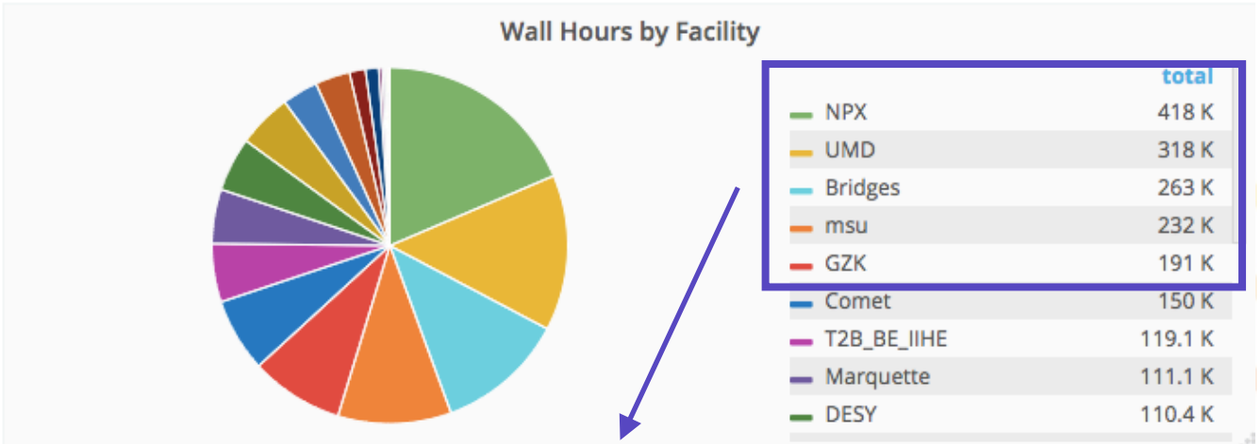




IceCube is 93% of the GPU Usage

By Facility





Top 5 are Pyglidein IceCube only and no containers

GPU GRAC Usage Dashboard. Brought to you by Derek Weitzel

IceCube South Pole Neutrino Observatory

UC San Diego

In Summary

Industry/Academia

Nvidia

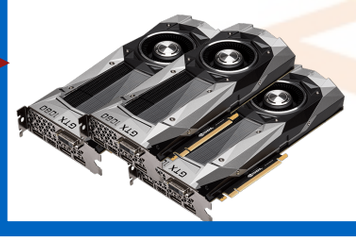
Open Science Grid

IceCube
South Pole Neutrino Observatory

CMS

GlideinWMS

Grid Sites



A monster with many heads

Problems:

- In general drivers are maintained by site admins and singularity image by VO's
- A whole new level of dependency trees are introduced:
- Cuda version
- CudaNN version
- Gcc version
- Library version
- Driver version

Open Science Grid

Questions?

Contact us at:

1-900-GPU-masters

IceCube South Pole Neutrino Observatory

UC San Diego

Open Science Grid

Questions?

Contact us:

osg-software@opensciencegrid.org

IceCube South Pole Neutrino Observatory

UC San Diego