

# Nebraska Site Report

HEPiX 2017  
Brian Bockelman



# Requisite Slide

- We bought more computers!
  - The facility is at about 35k cores (general purpose computing resources and dedicated CMS Tier-2 combined).
- We store bytes of data!
  - About 10PB across all filesystems; 3PB usable dedicated to CMS.
- We have network!
  - 100Gbps upgrade happened 2 years ago — but utilization rates are regularly around 30Gbps!

# Focus on Hardware

- I'm not a hardware person, but a few highlights:
  - Newest purchase has 56 hyperthreads / host and 256GB RAM: specialized CMS jobs can use nearly 5GB RAM / core.
  - Wide majority of worker nodes on 10Gbps. As 40Gbps uplinks are *cheap*, minimal blocking on top-of-rack switches.
    - This means 200Gbps of filesystem traffic *is a non-event*.

# New! Shiny!

- Nebraska might not be the biggest research computing site on the planet - but we are proud of the services we are able to offer:
  - **OpenStack**/Ceph - specialized projects can put together unique environments.
  - **CVMFS** for Data Federations.
  - End-to-end **container** setups.

# What Can You Do On OpenStack?

- Our local OpenStack offering is about 1 year old. What can be done:
  - JupyterHub instance for HTCondor training.
  - \*New\* OSG Accounting Service: GRÅCC.
    - Based primarily on micro-services, mostly run in Docker containers.
  - CVMFS repositories.
  - Spin up worker node VMs into the CMS Tier-2 cluster with the idle capacity.

# CVMFS For Data Federations

- We have spearheaded improvements to CVMFS that allow efficient publishing of data in federations.
- Nebraska runs [ligo.osgstorage.org](http://ligo.osgstorage.org); this is a copy of the LIGO detector data.
  - Only accessible if you have a LIGO-registered X509 proxy.
- Necessitated exporting of the LIGO data over HTTPS in addition to Xrootd. To sustain worldwide LIGO load, we need to be able to export @ 20Gbps.

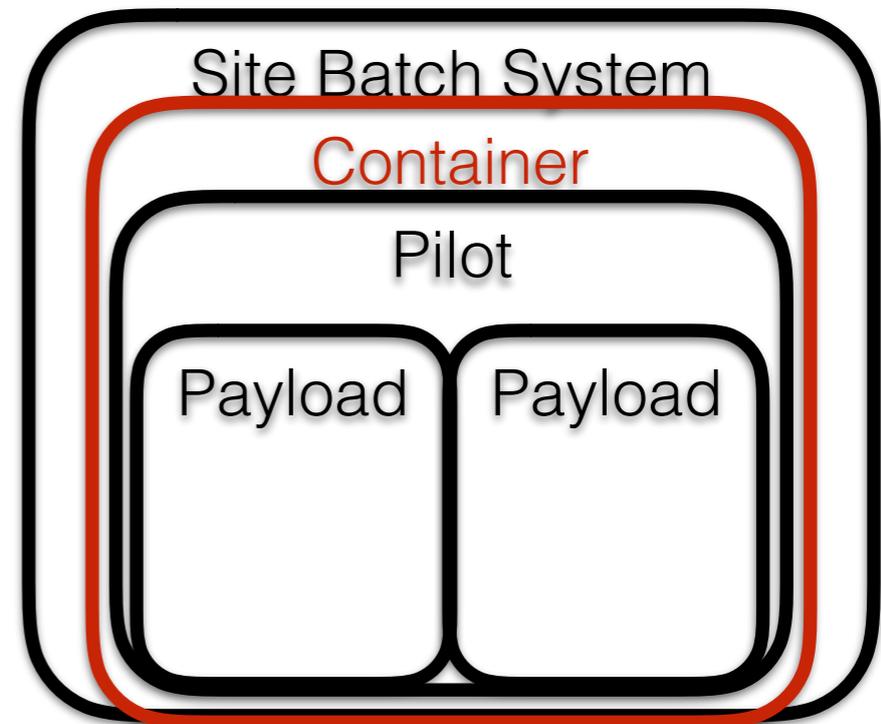
# A Computing Turducken

- Since 2012 (pre-Docker!), Nebraska has invested heavily in container technologies.
- This resulted in the native cgroup and namespace support in HTCondor.
  - In 2016, we migrated the CMS Tier-2 site to Docker containers. While the HTCondor cgroup code is **much beloved**, it's difficult to beat Docker's image authorship tools!
- This allowed us to **complete migration to RHEL7**.
  - Pilots will start inside RHEL6 or RHEL7 container, depending on pilot attributes.
  - We make the Singularity container system available inside Docker: pilots can further start concurrent payloads inside RHEL6 or RHEL7, regardless of the pilot's Docker container.

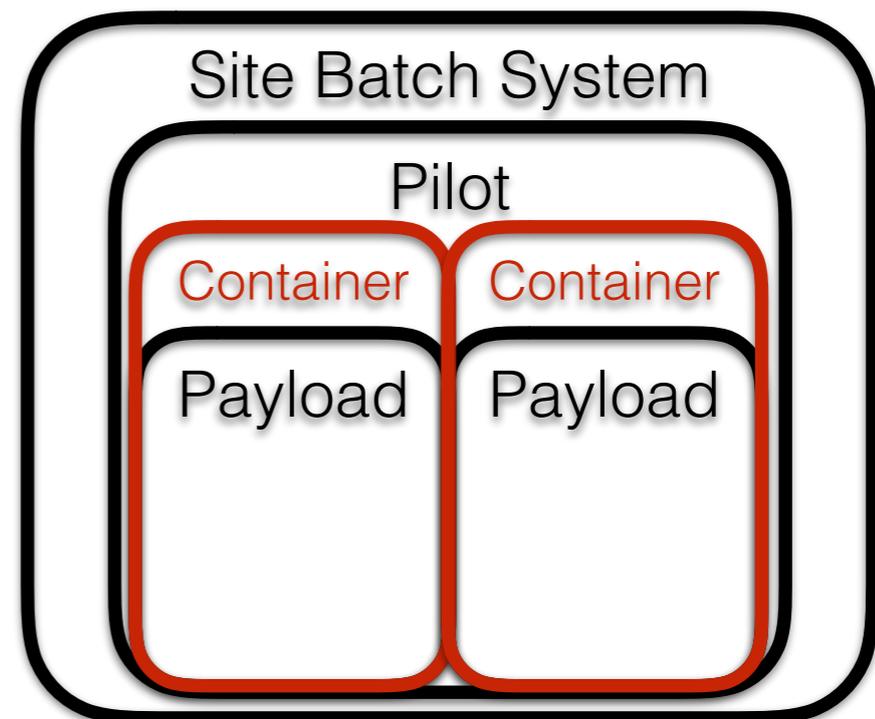
# Who is in the container?

- Three options when using containers:
  - A: Batch system starts pilot inside a container.
  - B: Pilot starts each payload inside its own container.
  - C: Combine A and B.
- Nebraska implements C: all pilots are launched inside a Docker container; CMS and OSG pilots launch a Singularity container per payload.

Option A:



Option B:



**See my talk on Wednesday!**

# Farewell to Friends

- Nebraska is a strong contributor to the OSG Software area. As OSG starts to wrap up its second funding cycle, the emphasis has shifted to *simplify, simplify, simplify*.
- Our next major release will remove large chunks of software:
  - **GRAM**: Already gone from most sites for about a year.
  - **GIP/BDII**: Replacement (OSG Collector) integrated into HTCondor-CE.
  - **Gratia (central-only)**: Move from a monolithic MySQL database at FNAL to a decentralized architecture. Database is ElasticSearch instance at Nebraska. Cutover from Gratia to GRACC on Tuesday!
  - **bestman2**: Replaced by LVS load-balanced GridFTP.

# Retirement Parties We Are Planning

- **glexec**: Replaced by Singularity, but not yet removed.
- Without glexec, we no longer need pool accounts. Without pool accounts, we no longer need **GUMS**.
  - Authorizations start to become flat files managed by Puppet. No more authz database or webservice to maintain.
- As VO authorization simplifies, we eliminate the need for a **VOMS-Admin** server.
- Working to replace “cookie cutter” **8-core pilots** with whole-node pilots. Take advantage of unique RAM/core resources.

# Looking Further

- Finish decommission of GUMS and glexec; finish commission whole-node pilots.
- **Going container crazy:** As we run more small services, I predict we will start work in the service orchestration layer (i.e., Kubernetes).
- Serious decrease in X509 client certificates - start to replace with OAuth2 / JWT.
- CVMFS: expand publication of data federation, use of HTTPS.
  - Integrate with GridFTP to allow remote writes.

**It's going to be a great year in Nebraska!**

**It's going to be a great  
year in Nebraska!**

Questions?