

LIGO containers in diverse computing environments

Thomas P Downes

Center for Gravitation, Cosmology & Astrophysics

University of Wisconsin-Milwaukee

LIGO Scientific Collaboration

'AUTOMATIC FOR THE PEOPLE'



LIGO–Virgo Advanced Detector Network

O1: September 2015 -- January 2016

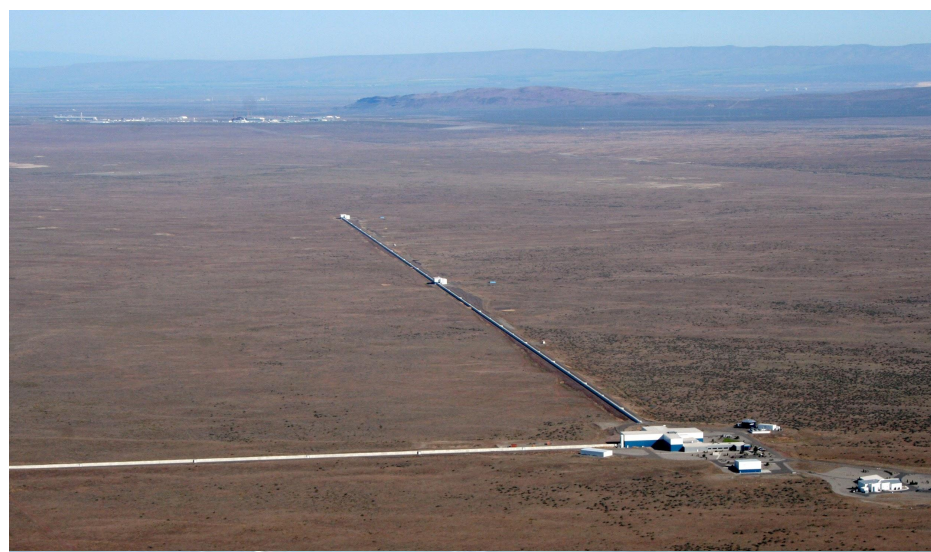
O2: December 2016 -- August 2017

O3: ~1 year of observing TBA

Upper-right: LIGO Hanford, Washington State, USA

Lower-right: Virgo ca. Pisa, Italy

Unshown: LIGO Livingston, Louisiana, USA

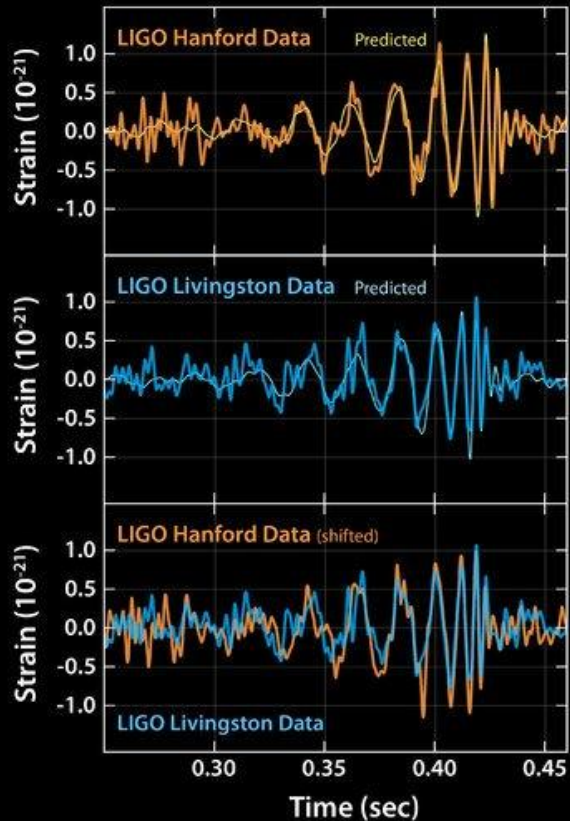


UW–Milwaukee and the CGCA



Kenwood Interdisciplinary Research Complex (2016)

- UWM recently identified as R1 by Carnegie
- CGCA: ~50 faculty/students/staff
- 6.5 FTEs dedicated to LIGO research support and identity management
- Highlights
 - ◆ LIGO.ORG Shibboleth Identity Provider
 - ◆ Primary Collaboration Wiki (w/Shibboleth ACLs)
 - ◆ Gitlab / Container Registry
 - ◆ Expanded [HTCondor cluster](#) coming online
 - ~5000 cores / 2PB
 - ◆ Gravitational Wave Candidate Event Database
 - LIGO-Virgo Alert System
- Also home to NANOGrav Physics Frontier Center



Images courtesy LIGO Laboratory & Fisher Price

Small amount of data: $\sim 1\text{MiB/sec}$!

“Modeled” LIGO searches compare data to many simulations

As our detectors become more sensitive we are seeing increased demand

- More data: observing runs are longer in duration
- Instrument sensitivity at low frequencies: longer numerical simulations
 - Higher event rate: candidate events are scrutinized in detail

Approximately a factor of 2-3 in growth each observing run!

We need to make greater use of resources not directly managed by LIGO

- LIGO researchers receiving computing resources from their institutions
- Open Science Grid resources (may also be a part of institutional resources)
 - Virgo computing resources in Europe

Researcher / administrator attention is our scarcest resource!

Increasing demand for LIGO Computing

LIGO Computing Environment and Practices

- ~5 clusters at various LIGO-affiliated institutions at any given time
- *Our own clusters are a diverse computing environment: lots of replicated work*
- Long e-mail chains across time zones
- Divine intervention required to replicate analyses in the future
- Staffing budgets flat on ~10yr timescale
- Still in many ways in early days of computing: just reaching 50k-core scale

Approach cannot be sustained from either user or administrator perspective!



- Typical jobs run out of home directory shared on submit and execute nodes (NFS)
- Typical jobs read instrument data from local shared file system (NFS, HDFS, GlusterFS)

The low-cost approach to development suddenly has costs when you have more and better data!

Must make it easier for development practices to more closely mimic what “we want the users to do” at similar up-front cost in time and technical understanding.

Technical debt: it seemed like a good deal at the time...

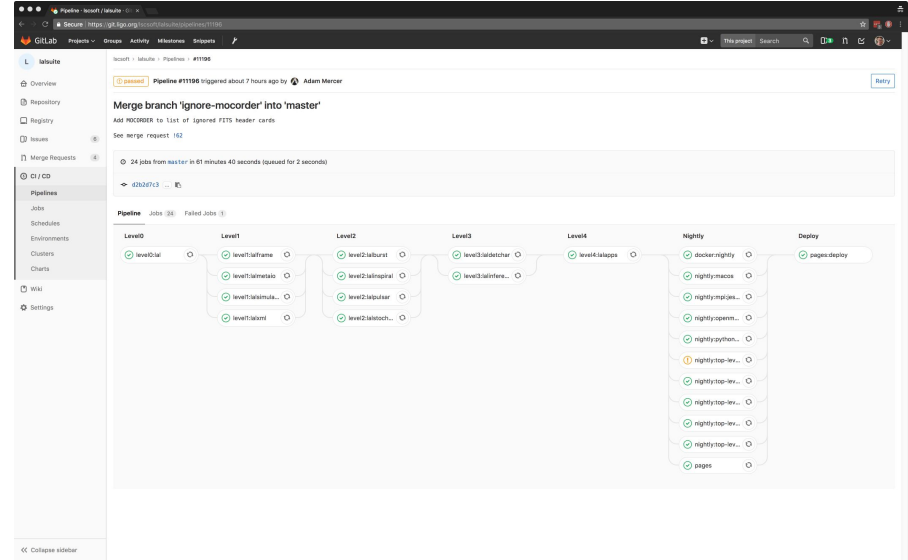
Reject thesis that scientific use cases are special: use standard tools!

Even really smart people have work that can and should be performed by a robot

Continuous integration w/fork + merge to reduce impact of broken changes to code

Continuous deployment w/agnostic outputs (Tarballs, Docker image, .deb/.rpm, pypi)

Users can self-deploy to their workstation, but can we continuously deploy to the grid?



Contemporary tools are *really* good for moving fast

Webhook Automation

*GitLab allows me to automate webhooks on behalf of **all** LIGO researchers who “docker push” to our container registry*

GitLab Container Registry produces nightly build/public release of LIGO Algorithm Library

`docker pull containers.ligo.org/lscsoft/lalsuite:nightly`

Below: API-triggered DockerHub rebuilds of our cluster login and job environment



GitLab

The screenshot shows a web browser window with the URL `https://hub.docker.com/r/ligo/software/builds/`. The page title is "PUBLIC | AUTOMATED BUILD" and the repository is "ligo/software". It indicates the last push was "2 days ago". The "Builds" tab is selected, showing a table of automated builds.

| Status | Actions | Tag | Created | Last Updated |
|-----------|---------|------------------|------------|--------------|
| ✓ Success | | stretch-proposed | 2 days ago | 2 days ago |
| ✓ Success | | jessie-proposed | 2 days ago | 2 days ago |
| ✓ Success | | stretch | 2 days ago | 2 days ago |
| ✓ Success | | jessie | 2 days ago | 2 days ago |

On the right side of the build table, there is a "Source Repository" section showing the repository name `lscsoft/docker-ligo-software`.

DockerHub or GitLab Container Registry builds
container and generates webhook

[DockerHub: +1 hour @ 5GB worker node image]

[GitLab Container Registry: θ (minutes)]

LIGO Webhook Relay validates and
forwards event to CVMFS Publisher

CVMFS Publisher receives event and
places it in job queue

Job queue pulls container images and
publishes them 1-by-1
[+13 minutes @ 5GB]

Available to clients at
[/cvmfs/ligo-containers.opensciencegrid.org](https://cvmfs/ligo-containers.opensciencegrid.org)

Within hour, a developer can test changes via Docker or on Open Science Grid using Singularity and CVMFS!

Publishing of Docker images to CVMFS for use with Singularity

Thanks, CERN + Open Science Grid!

- CERN + OSG improved support for our Debian clusters and users
 - Very responsive to bug reports and discussion list
- OSG infrastructure serves as LIGO's Stratum 1 CVMFS Replicas
- Code to convert Docker images to CVMFS is a fork of [OSG's nightly script](#) developed by Brian Bockelman and Derek Weitzel
- **Issues:** Data w/Auth not First Class Citizen in CVMFS ecosystem
- **Issues:** CVMFS + MacOS (or Docker on MacOS) not easy
 - LIGO data-on-demand on MacOS that would be big selling point that would lower “cultural” barriers to adoption at grid scale

Success so far...

- Service active for 4 months
- Two pipelines ported to use Singularity + CVMFS + HTCondor file transfers
- Removing typical LIGO dependency on local shared filesystems
- Work performed by user experienced w/OSG but not with containers

Problems so far...

- LIGO sysadmins and users don't have much experience managing file transfers
 - Must have working examples of “more resources easier” to have any hope of getting researchers to pay **any up-front cost at all** in “non-science” modifications to workflow
- LIGO data available over CVMFS + X509 authz helper
 - But.. many sites replace this with local symbolic link outside of /cvmfs at arbitrary mount point (e.g. /hdfs, /gpfs, etc.). Problematic for bind mounts w/o OverlayFS
 - Workflow at UWM can interact with X509 authz helper to hang process table
- I have to figure out what HTCondor does with “+SingularityImage” by D_FULLDEBUG logging
 - “Sophisticated” user work-around: invoke singularity w/arguments directly
 - Edge-cases solved at grid level with wrappers/GlideIns; slower adoption within HTCondor
- How to organize and present containers for reproducibility in the long (long) term
 - Tags come and go, but manifest digests are forever. Real people use tags.

These applications are distributed as fairly simple Docker Compose applications

- Webhook Relay: <https://github.com/lscsoft/webhook-relay>
 - Validates webhooks (to best of ability) and relays events it is configured to expect
- Webhook Queue: <https://github.com/lscsoft/webhook-queue>
 - Receives webhooks (from Relay or direct from service) and places event on a job queue
- Relay + Queue can easily be re-implemented (e.g. AWS API Gateway + Lambda + SQS)
 - *Wanna help?*
- CVMFS-to-Docker worker: <https://github.com/lscsoft/cvmfs-docker-worker>
 - Processes job queue, gracefully moving to next job upon failure
 - Uses singularity to convert Docker image to directory structure in CVMFS
 - Adds several typical OSG bind points for sites without OverlayFS

The infrastructure is freely available