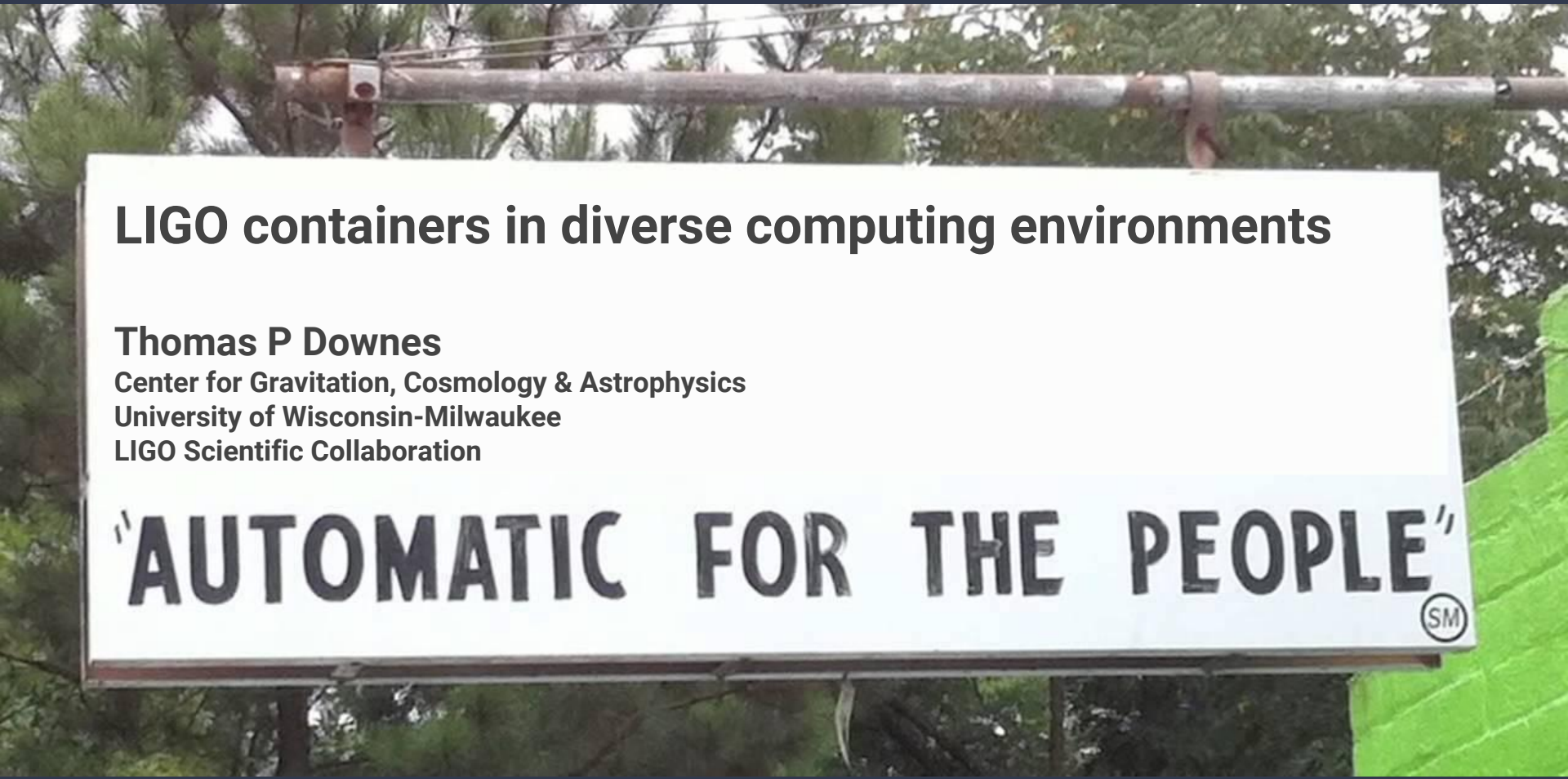
LIGO containers in diverse computing environments

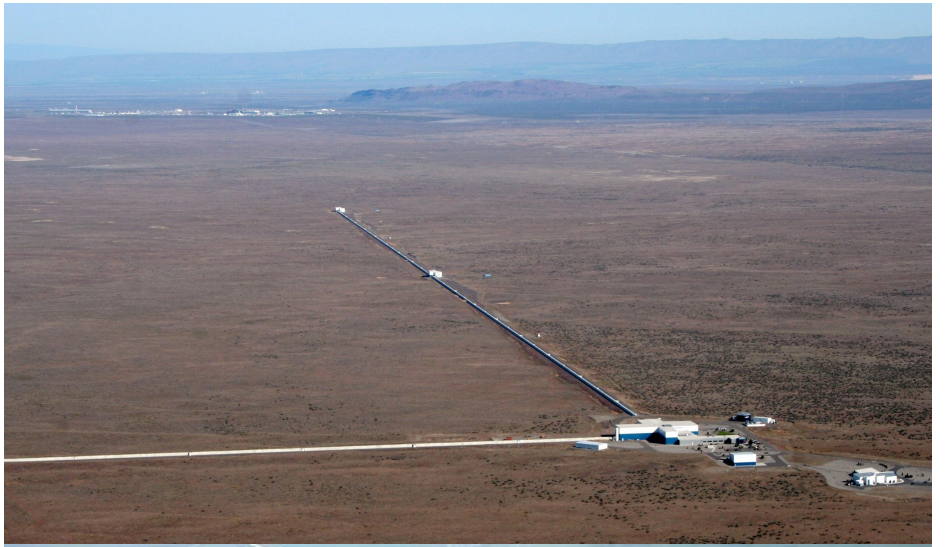
Thomas P Downes
Center for Gravitation, Cosmology & Astrophysics
University of Wisconsin-Milwaukee
LIGO Scientific Collaboration



LIGO–Virgo Advanced Detector Network

O1: September 2015 -- January 2016
O2: December 2016 -- August 2017
O3: ~1 year of observing TBA

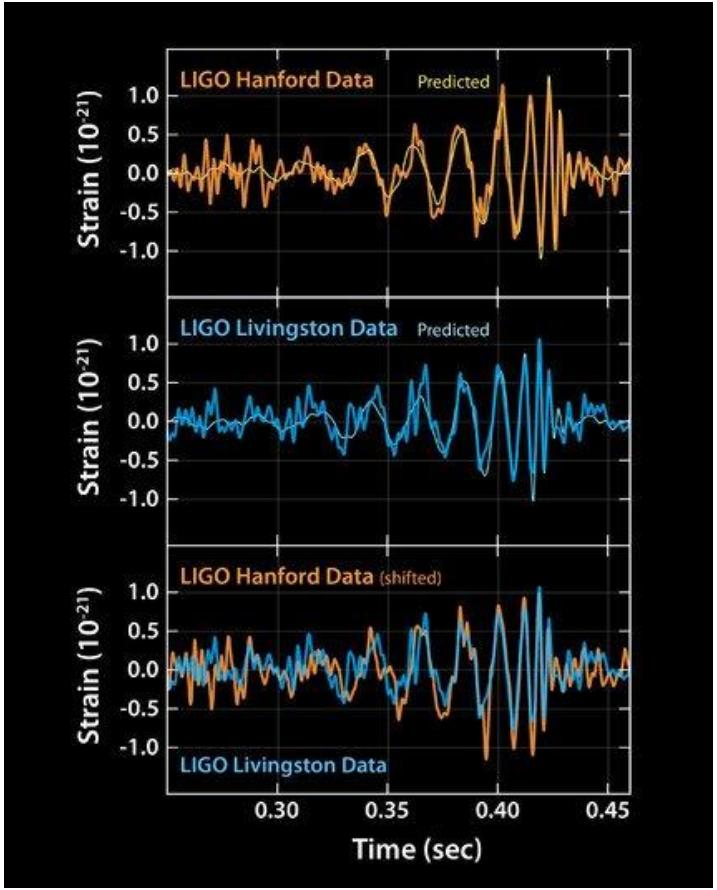
Upper-right: LIGO Hanford, Washington State, USA
Lower-right: Virgo ca. Pisa, Italy
Unshown: LIGO Livingston, Louisiana, USA



UW-Milwaukee and the CGCA

Kenwood Interdisciplinary Research Complex (2016)

- UWM recently identified as R1 by Carnegie
- CGCA: ~50 faculty/students/staff
- 8 computing staff 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: ~1MiB/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

Enter CVMFS...

LIGO Computing

A greatly reduced list of our other problems

- Actual jobs typically run via shared home directory + dependency chain into system
- Lots of replicated work across clusters
- Long e-mail chains across time zones
- Divine intervention required to replicate analyses in the future

Enter containers...

- Although LIGO's first run was ~15 years ago, it's still in many ways in early days of computing
  - Only now reaching 50,000 core-year scale
  - Bottom-up approach to algorithm development with reviews led by senior faculty focused on scientific accuracy of implementation
- Staffing budgets flat on ~10yr timescale
  - Can barely maintain legacy infrastructure
  - Rely on "lightweight" grid solutions and adopt "heavier" tools only as ***truly*** necessary
- My role: introduce a more professional approach to our software and related infrastructure
  - Fork/branch/review/merge
  - Don't send a grad student to do a robot's job

The Culture of LIGO

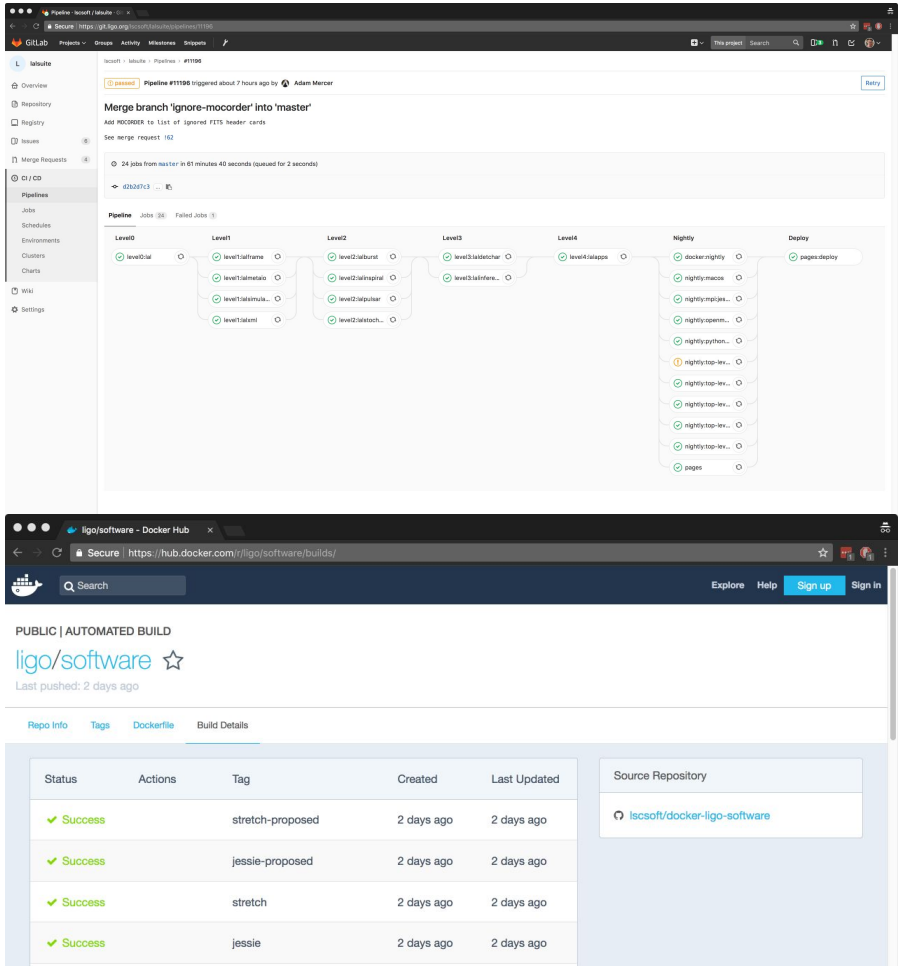
Automation helps

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

Above: Nightly build/public release of LIGO Algorithm Library container. Available via

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

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



DockerHub or GitLab Container Registry builds container and generates webhook
[DockerHub: +1 hour @ 5GB worker node image]
[GitLab Container Registry: 0(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

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

Automated publishing of Docker images to CVMFS

Thanks, CERN + Open Science Grid!

- CERN + OSG improved support for our Debian clusters and users
- 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
  - https://github.com/opensciencegrid/cvmfs-singularity-sync
- Issues: Data/Auth not exactly First Class Citizens in CVMFS
- 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 data available over CVMFS+X509
  - But.. many sites replace this with local symbolic link outside of /cvmfs at arbritrary mount point (e.g. /hdfs, /gpfs, etc.)
- Sites w/o OverlayFS
  - Hacky: pre-create all data mount points within container
  - chroot inside container?!?
  - Long-term solution: unprivileged overlayfs
- I have to figure out what HTCondor does with “+SingularityImage” by D_FULLDEBUG logging
  - What Condor does is not complex, but not obvious from pp. 790-791 of manual
  - “Sophisticated user” work-around: invoke singularity w/arguments directly
  - Edge-cases solved at grid level with wrappers/GlideIns; slower adoption within HTCondor

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