IGWN Computing for O4, O5, and beyond

Duncan Macleod

CARDIFF UNIVERSITY
PRIFYSGOL CAERDYDD

Gravity Exploration Institute
Sefydliad Archwilio Disgyrchiant

LSC

IGWN Computing Overview

Collaboration operations

- Identity and Access management
- Collaboration services

Data handling

- Data generation
- Data distribution (low-latency)
- Data distribution (higher-latency)
- Data archival and release

Software and computing environments

- Low-latency computing platform
- Higher-latency computing platform

Resource provision, allocation, and accounting

Collaboration operations

Identity and Access Management (IAM)

Today

Identity management:

- LIGO/Virgo members given ‘LIGO.ORG’ identities
- Virgo members have a Virgo identity
- KAGRA members have a KAGRA identity
- Non-LIGO.ORG members given access to services via gw-astronomy.org COmanage registry (including KAGRA)

Browser auth (LIGO.ORG):

- SAML/Shibboleth

Terminal authentication (LIGO.ORG):

- Kerberos
- SAML/ECP (stored as X.509)

O4+

ID management -> Federated Identity:

- Each collaboration manages its own members
- IGWN directory defines internal roles
- Services defer authentication to home institution, authorization is done by IGWN

Terminal authorization -> OAuth2 Bearer tokens

- User authenticates via browser
- SciTokens implementation (JWT)
- Specifies access ‘claim’ per services
- Single token can include multiple claims

Collaboration services

Large proportion of IGWN computing personpower used to develop/maintain services required to run the collaboration

- IAM
  - see previous slide, plus
  - on-boarding/off-boarding
  - profile management
  - group management
- Knowledge base (wiki)
- Document control and review
- Version control system (GitLab)
- Communications
  - mailing lists
  - chat
  - remote conferencing
- Elections/voting system

Many groups also provide direct-access computing resources with:

- JupyterHub
- Web access to files/results

Another large proportion is used to provide operation support for users of each of these services, and the wider computing infrastructure.

Data handling

Data generation and low-latency distribution

Data generation handled per observatory

- ‘Raw’ data generated as GWF frame structures, written to disk and held in memory
- Low-latency strain calibration pipeline generates *h(t)* and related outputs (as GWF) with ~few second latency

Low-latency data distribution

- strain data from each observatory are sent directly to various end-points for analysis (LIGO-Caltech, Cascina, UW-Milwaukee, …)

O3/O4 operations:

- using a hybrid of custom networking software at each observatory and industry-standard streaming solution
- Users read data from posix locations (files)

O4/O5+:

- All streaming
- Users subscribe to the streams they want

Higher-latency data distribution and archival

Higher-latency data distribution

- Small strain files are aggregated into longer files
- Proprietary strain data are distributed from each observatory archive via CVMFS
- Available from any remote machine that ‘mount’ the CVMFS repositories
  - /cvmfs/kagra.osgstorage.org/
  - /cvmfs/ligo.osgstorage.org/
  - /cvmfs/virgo.osgstorage.org/
- Tier-0/1 data centres receive their own redundant copy of the Strain data and redirect CVMFS to the local cache

Data ‘discovery’ method provided via (authorised) query API

- (tstart, tstop, obs, dataset) -> file URLs

Permanent data archive provided by each observatory

- Perpetual storage of all instrumental data, strain data, and metadata (list of observing segments, etc…)

Open access data archive

- GW Open Science Center is now joint IGWN project
- ~1 hour around each event published alongside the relevant announcement
- Bulk data released in 6-month cadence with 18-month latency

More info: LIGO Data Management Plan

Software and computing environments

Computing environments

Low-latency analysis predominantly performed at the main observatory computing centre

- Low-latency alert infrastructure runs on dedicated resources with high priority to burst out into pool
- Event database HA via AWS
- Event orchestrator HA via Kubernetes
- Search pipelines run on dedicated or highly-prioritised resources in an HTCondor-managed resource pool
- Fast, direct access to small data files / shared memory

Higher-latency analysis is performed on any connected resource

- mainly HTCondor-based workflows
- mainly pledged resources run by IGWN member groups

In O4 all workflows should endeavour to run on the IGWN DHTC platform

- Jobs execute on any remote resource that matches the requirements (e.g. has data)
- Resources providers only need to provide execute resources and a job scheduler (not necessarily HTCondor)
- Software pre-compiled and distributed via CVMFS or container

Software development/management

Software deployment is controlled via change control board (SCCB)

- Software must be approved by SCCB and the relevant scientific review committee before being used for production analysis

IGWN Conda Distribution provides pre-built, automatically-distributed environments of approved software

- available via CVMFS on any machine (no authentication required)
- can be replicated on any workstation
- OS-independent
- compatible with most cloud notebook services (mybinder, etc)

Scientific software *development* is out-of-scope for Computing WG, but we provide training, assistance, and maintenance/ops support.

- development practice
- continuous integration / testing
- packaging
- working with the DHTC platform

Computing WG does develop and maintain a large portfolio of infrastructure software (authentication/authorisation, I/O libraries, services).

Resource provision

Resource provision, allocation, and accounting

Internal estimates suggest a total CPU requirement of 1.3 GSU (billion CPU hours) across all scientific analyses.

Majority of computing provided by the LIGO Laboratory as part of their ongoing NSF funding.

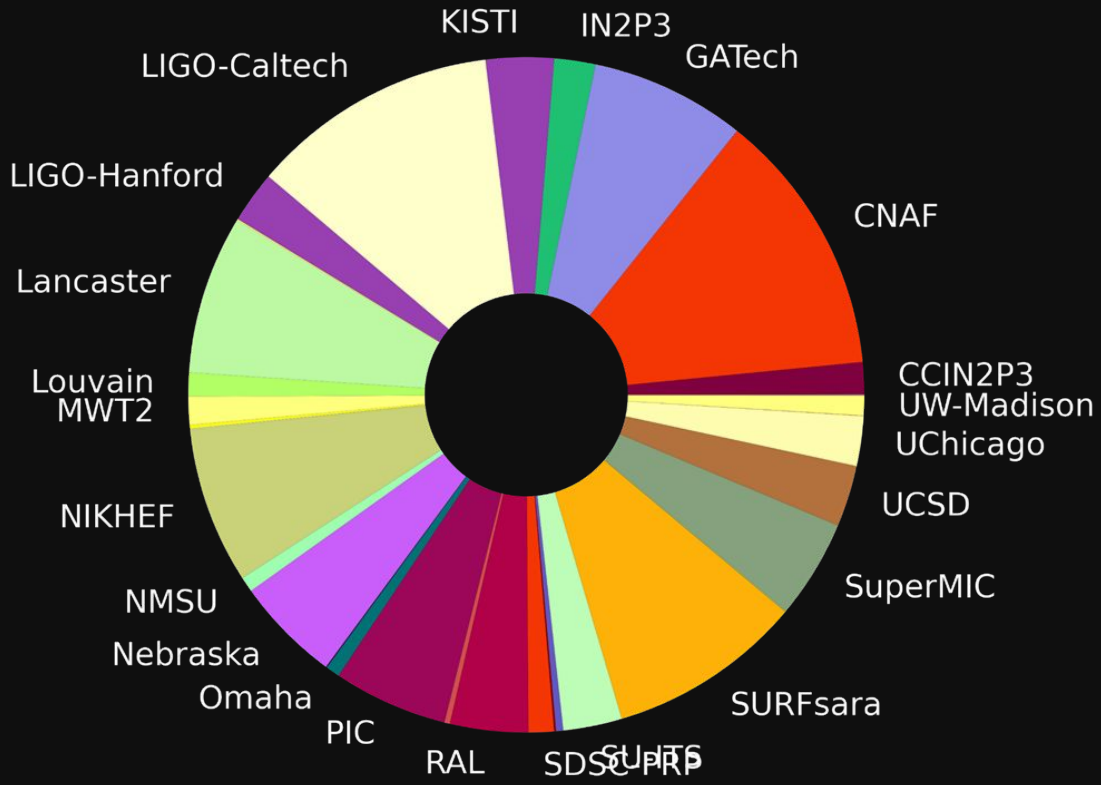
- LIGO Lab will no longer purchase hardware dedicated to high-latency data analysis
- More resources need to come from DHTC-connected providers
- Barrier-to-contribution for providers is lower than ever

In the process of updating acknowledgements statements for resource providers

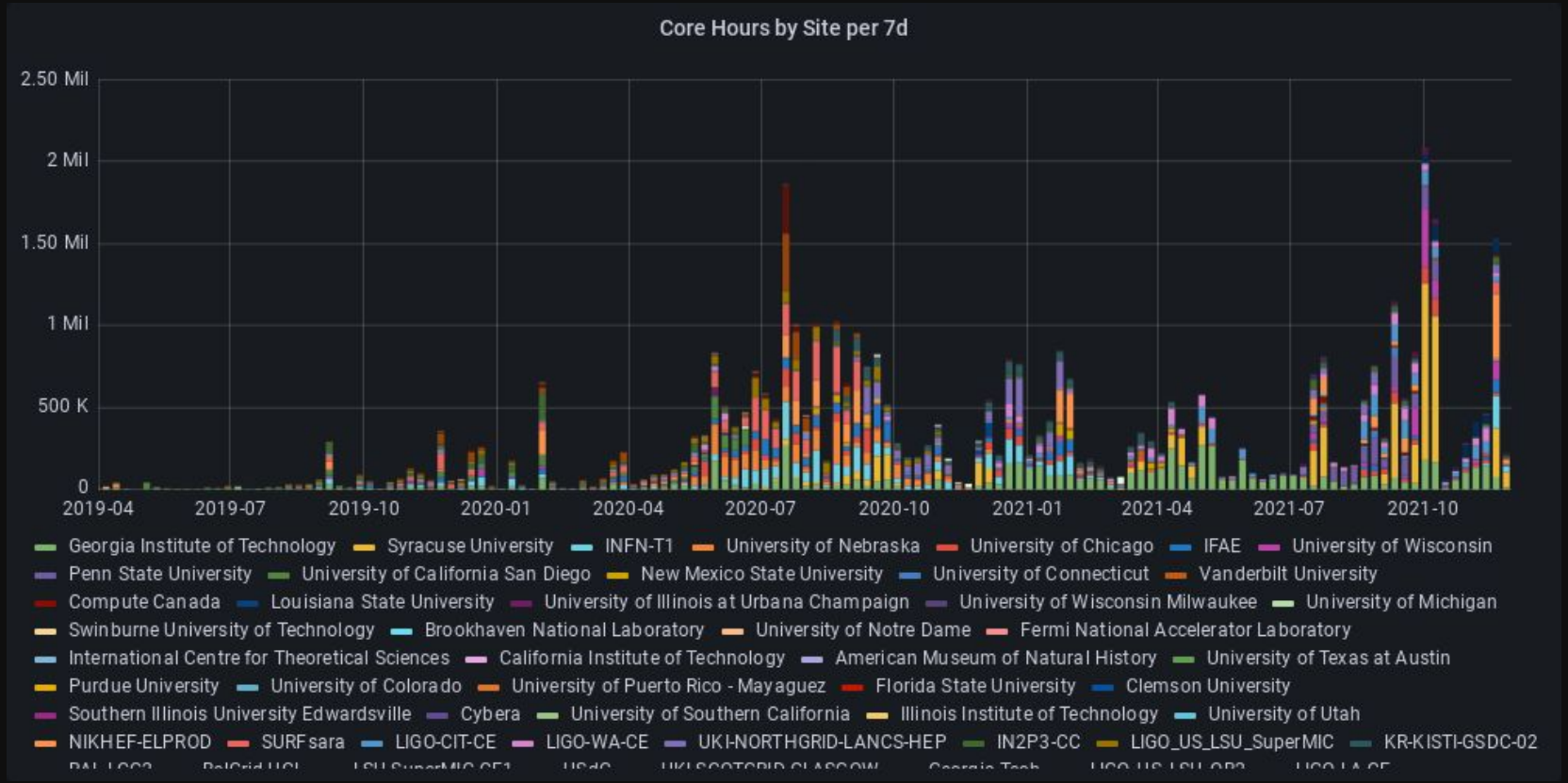
O4 usage model is basically a free-for-all:

- no actual gate to computing usage for any collaboration member
- all usage must be ‘tagged’ with a key that includes the observing run, the working group name, the search, and the pipeline, e.g. `ligo.dev.o4.cbc.pe.mcmc`
- no analysis is ever prevented from running, but priority can be given to allow queued analyses faster/greater access to resources
- usage is estimated in advance, and audited after-the-fact

In O4 usage will be tagged closer to the scientific result target (e.g. which paper this is for, or which GW event ID is being processed)



IGWN DHTC usage (<20% of total)



IGWN DHTC usage https://gracc.opensciencegrid.org/goto/d4EJiXpnz?orgId=1

IGWN Computing Overview

Collaboration operations

- Federated identity
- API authorization moving to bearer tokens
- Large service burden for day-to-day ops

Data handling

- data written as GWF
- low-latency distribution with Kafka
- higher-latency distribution via CVMFS

Software and computing environments:

- low-latency workflows -> dedicated resources
- higher-latency workflows -> DHTC platform
- OS-indep. reference software distribution

Resource provision:

- DHTC platform allows easy contribution of resources