Still Moving Data After All These Years

Scott Koranda

LIGO and University of Wisconsin-Milwaukee

March 4, 2009

LIGO-T0900090

What data?

Steady state streams from instruments...

LIGO interferometers

- Livingston, LA (here)
- Hanford, WA
- combined 1.2 TB per day

Virgo interferometer

- Cascina, Italy
- 650 GB per day

GEO600 interferometer

- Hannover, Germany
- 150 GB per day

Not all the steady state data is replicated.

- Raw LIGO data only replicated from sites to CIT
- Reduced LIGO data sets generated at sites and replicated
- Virgo data reduced for consumption by LIGO
- GEO data reduced for consumption by LIGO

Total steady state is $\approx 17 MB/s$.

Also burst and boutique streams to replicate...

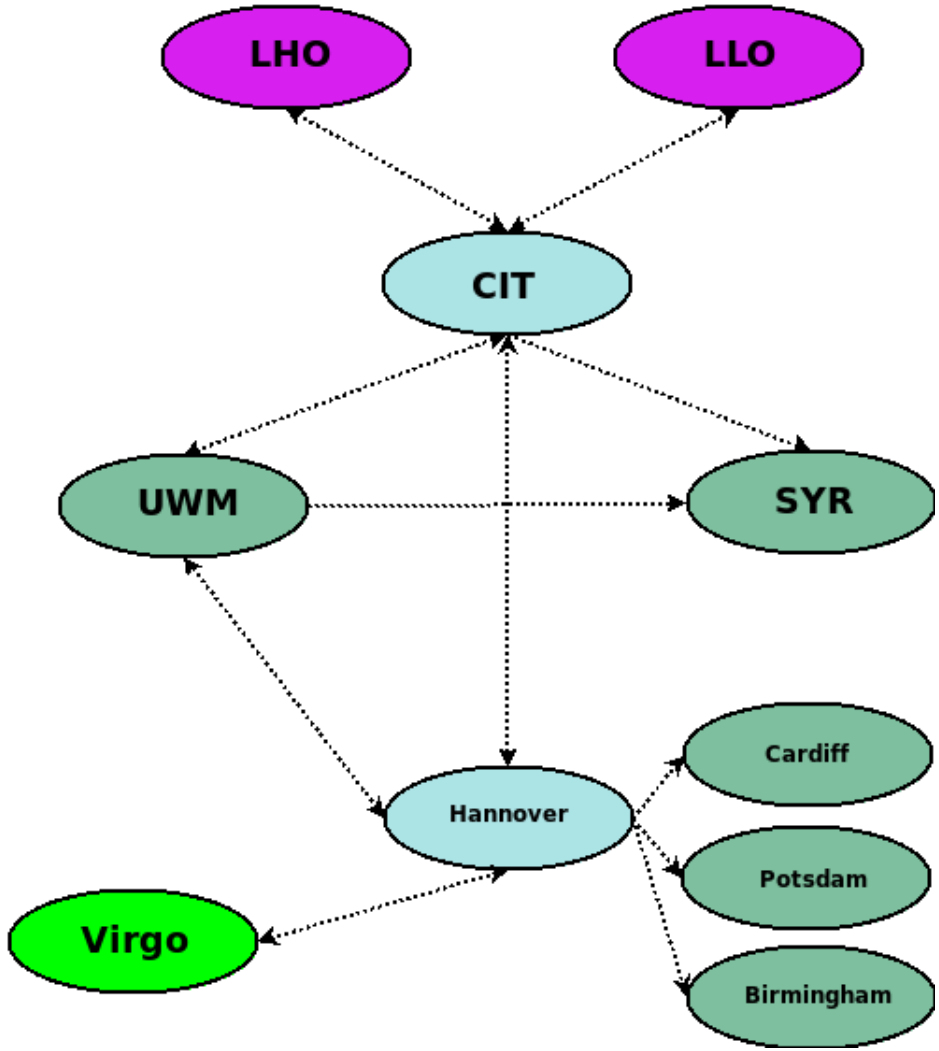
Typical scenarios:

- Version 4 calibrated S5 data is available and **we need it now**.
- ExtTrig group generated new injections and **we need it now**.

Burst replication scenarios involve anywhere from few 100 GB to 10's of TB

That's what data we replicate...

Replicate where?


LHO
LLO
CIT
UWM
SYR
Cardiff
Hannover
Potsdam
Virgo
Birmingham

Early on the only questions asked were...

How fast?
How fast?
How fast?
How fast?
How fast?
How fast?

and

Is it here yet?
Is it here yet?
Is it here yet?
Is it here yet?
Is it here yet?
Is it here yet?

People thought only important metric was raw transfer rate.

Fast is easy...

We leverage GridFTP from Globus Toolkit

- Third-party transfers
- 32 concurrent control channels (typically)
- Each with 2 parallel data streams
- Tuned TCP windows
- Pipelining

With supporting disk I/O we can *easily* utilize 70% (or more) of available bandwidth.

Most importantly we leverage the GridFTP Client API

- Wrap C API for use with Python
- Enables tight integration with transfer logic
- Especially simple for third-party transfers

More people should be coding against this API!

So fast is easy...

A little harder is "*Where is the data*?"

Today need to track over 30 million unique files replicated across more than 10 sites.

I expect the need to grow to 60 million files within two years.

We leverage Replica Location Service (RLS) from Globus Toolkit

- Catalog file $\leftrightarrow$ URL pairs
- At least one `file://` and `gsiftp://` per file
- Lightweight bloom filters push catalog indices to other sites
- Query any site to find locations, query at location for URLs
- MySQL backend with InnoDB engine, no scaling issues yet

We run perhaps the largest RLS network?

```
$ globus-rls-admin -S rls://localhost
Version:    4.7
Uptime:     1439:51:48
LRC stats
  update method: lfnlist
  update method: bloomfilter
  updates bloomfilter: rls://ldas.mit.edu:39281 last 03/02/09 14:12:35
  updates bloomfilter: rls://ldas.ligo-la.caltech.edu:39281 last 03/02/09 14:12:50
  updates bloomfilter: rls://ldas.ligo-wa.caltech.edu:39281 last 03/02/09 14:12:25
  updates bloomfilter: rls://ldas-cit:39281 last 03/02/09 14:12:21
  updates bloomfilter: rls://nemo-dataserver.phys.uwm.edu:39281 last 03/02/09 14:12:16
  updates bloomfilter: rls://ygraine.aei.mpg.de:39281 last 03/02/09 14:11:27
  updates bloomfilter: rls://dataserver.phy.syr.edu:39281 last 03/02/09 14:10:00
  updates bloomfilter: rls://ldr.aei.uni-hannover.de:39281 last 03/02/09 14:10:32
  lfnlist update interval: 86400
  bloomfilter update interval: 600
  numlfn: 24221203
  numpfn: 48851223
  nummap: 48851223
RLI stats
  updated by: rls://ldas.ligo-wa.caltech.edu:39281 last 03/02/09 14:10:56
  updated by: rls://nemo-dataserver.phys.uwm.edu:39281 last 03/02/09 11:47:00
  updated by: rls://ldas.ligo-la.caltech.edu:39281 last 03/02/09 14:05:42
  updated by: rls://charlie.amp.uni-hannover.de:39281 last 03/02/09 13:32:57
  updated by: rls://ldas-cit.ligo.caltech.edu:39281 last 03/02/09 14:12:25
  updated by: rls://ldr.aei.uni-hannover.de:39281 last 03/02/09 13:03:01
  updated via bloomfilters
```

LSC

Replicating data fast is easy with GridFTP...

Cataloging and tracking data location is easy with RLS...

So what's hard?

Metadata is hard.

Collecting and managing all the data *about the data* is the hard part.

More effort put into managing metadata than any other aspect of data replication.

Why metadata?

- Scientists need to find data knowing only metadata, not file names.
  - GSP start time, end time, interferometer, frame type, ...
- Sites choose which data to replicate based on metadata.
- Integrity of replicated data ensured by metadata (checksums).

Some metadata issues confronted

- Strategies for partitioning and management of the partitions
- Strategies for replication of the metadata
- Scaling the services up and up again and again
- Integrity of the metadata (yet to be implemented)

Metadata service

- Simple listener on GSI protected socket serving metadata.
- Sites connect and pull only metadata needed for data they want to replicate.

- GridFTP for moving data fast
- RLS for tracking where data is
- Metadata services for knowing what data to replicate

Putting it all together with some glue...

LIGO Data Replicator (LDR)

- GridFTP servers for exposing file sytems
- RLS catalog for tracking data locations
- Metadata service for replicating metadata
- Customized scheduler (fairly simple)
- Customized GridFTP client
- Data finding service for apps

LSC

LIGO Data Replicator (LDR)

- globus-gridftp-server
- RLS
- LDRMetadataServer
- LDRMetadataUpdate
- LDRTransfer
- LDRSchedule
- LDRDataFindServer

Where is LIGO going with data replication and LDR?

Explore and leverage new GridFTP functionality

- Especially interested in GridFTP multicasting
- Requires change to a push and/or subscribe architecture

New backend(s) for RLS

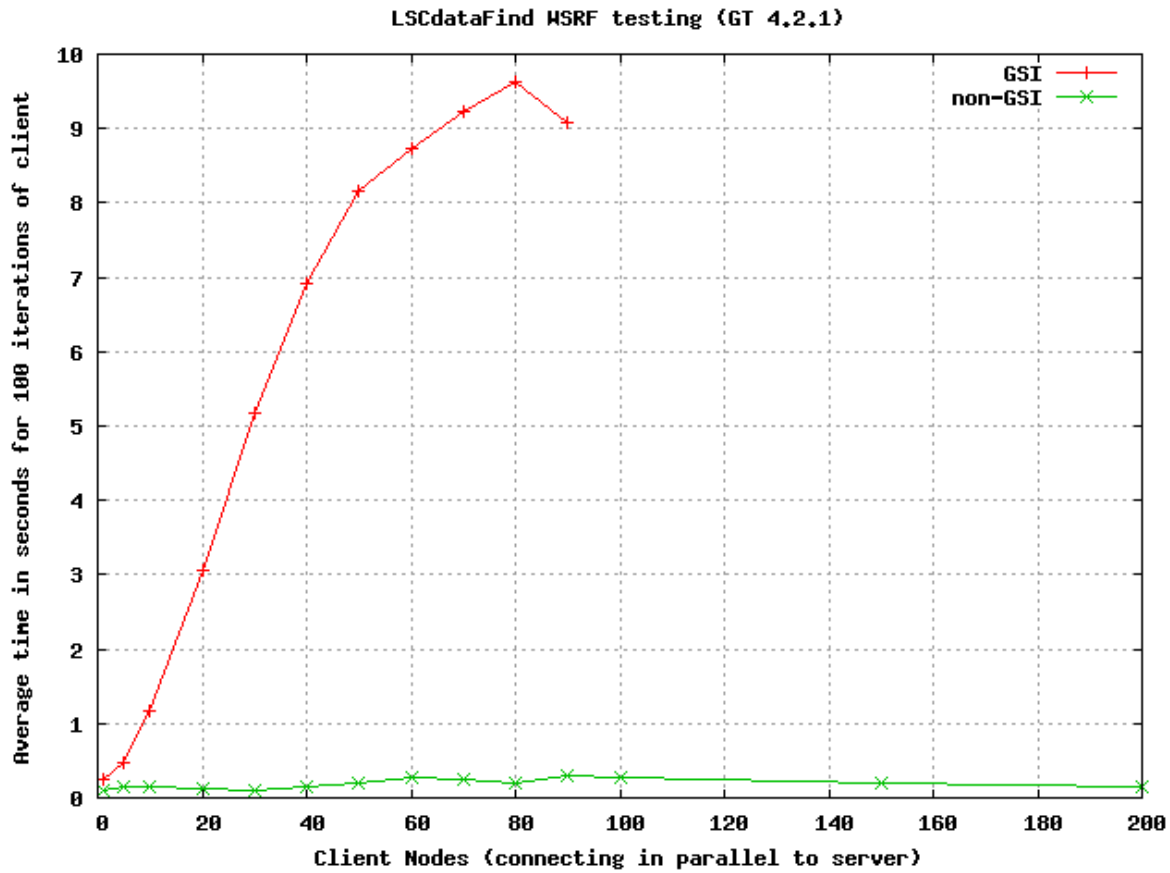
- In collaboration with Ann and Rob from RLS team
- Want closer integration with storage
- Moving or deleting a file should be automatically "detected"
- LIGO diskCache tool is fast in-memory hash of file locations
- Work underway with RLS team to abstract RDBMS interface

Modernize the metadata service

- Better tools for management of collections of metadata
- Work underway to move to a GT 4.2.1 WSRF service
- Coding against the GT 4.2.1 Java WS

Modernize the data finding service

- Faster, faster, faster! Scale, scale, scale!
- Support 10 Hz across most LIGO clusters
- Prototype built on GT 4.2.1 C WS



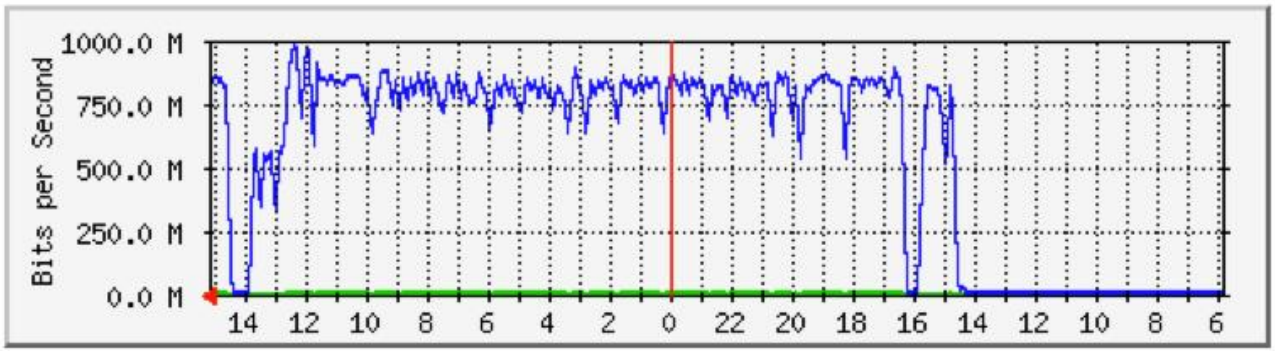
LDR as a service

- Smaller sites just want the data without any hassles
  - Expose what storage is available with GridFTP server
  - Leverage third-party transfers
  - All other LDR components run from Milwaukee
- Birmingham University (UK) will be first customer

Dynamic Circuit Networks (DCN)

DCN is a switching service that creates short-term dedicated bandwidth between end-users that require dedicated bandwidth, including reliable connections lasting from minutes to days.

- NYSERNet worked with CIC OmniPoP and WiscNet to connect Syracuse and UWM LIGO clusters.
- 22 TB of LIGO data replicated



	Max	Average	Current
In	16.3 Mb/s (0.2%)	7506.6 kb/s (0.1%)	10.7 Mb/s (0.1%)
Out	979.9 Mb/s (9.8%)	545.8 Mb/s (5.5%)	853.6 Mb/s (8.5%)

The team

Xavier Amador * Stuart Anderson * Carsten Aulbert * Antonella Bozzi * Duncan Brown * Gerald Davies * Henning Fehrmann * Kevin Flasch * Steffen Grunewald * Scott Koranda * Dan Kozak * Greg Mendell * Brian Moe * Livio Salconi * Igor Yakushin * Apologies to anyone left off...