

Status and Plans for LDR Development

Scott Koranda

University of Wisconsin-Milwaukee

September 24, 2008



LIGO-G080534-00-Z

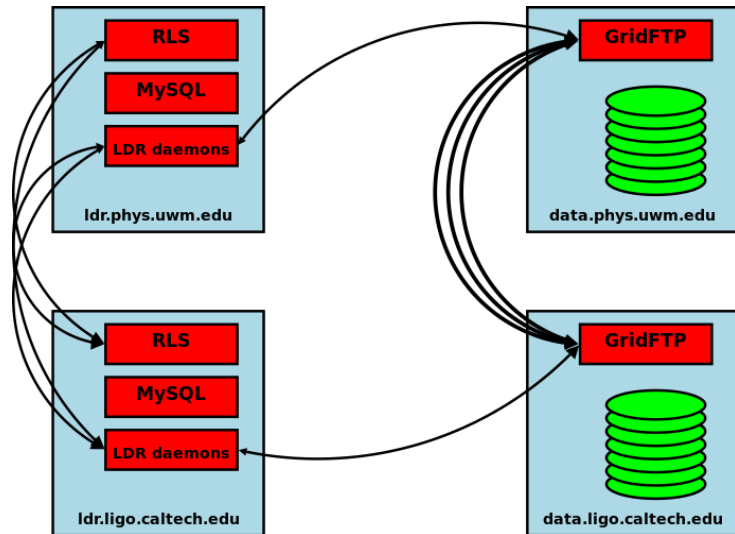
LDR effort for S6 in six areas

1. Native packaging for CentOS, Debian, Solaris x86_64, SL
2. Full support of third-party transfers
3. Improved publishing scripts
4. Lower latency in support of “real time” pipelines
5. Better manageability for metadata and psets
6. Scalable LDRdataFindServer

Native packaging for CentOS, Debian, Solaris x86_64, SL

- ▶ Necessary to push out updates faster and easier
- ▶ RPMs for CentOS 5.2 x86_64 and i386 testing at Syracuse
- ▶ RPMs for SL 4.2 x86_64 ready for testing
- ▶ Debianization under way
- ▶ Solaris packages after Debian
- ▶ Manual being updated
- ▶ Status: need few more weeks for Debian and Solaris

Full support of third-party transfers



Full support of third-party transfers

- ▶ Enabled multiple concurrent third-party transfers
 - ▶ Server to server transfers
 - ▶ Better stability and dramatic throughput increase
 - ▶ In production at Cardiff, UWM, Hannover, Cascina
 - ▶ Hannover pulls from UWM at sustained 80 MB/sec
- ▶ Need to parallelize md5 checksum after transfer
- ▶ Full support enables simple architecture
- ▶ Status: Ready to be coded, estimate one week effort

Improved publishing scripts

Files published into LDR when metadata and URLs recorded

- ▶ New design allows “plug in” modules
 - ▶ File discovery (directory scanning, socket listening, ...)
 - ▶ Checksum computing (parallel md5, SFT, ...)
 - ▶ File size discovery (simple stat, ?)
- ▶ Options for logging, heartbeat monitoring, ...
- ▶ S6 change: no science, coincidence attributes
- ▶ Status: currently in development, ready for some testing

Lower latency in support of “real time” analysis

- ▶ Testing now on LIGO lab next generation gateway boxes
 - ▶ 16 core, 2 GB per core, Solaris 10 AMD Opteron
 - ▶ Substantial increase in performance over current hardware
 - ▶ No statement yet on latency support for S6 for $h(t)$
- ▶ Better scheduling in LDRSchedule by Kevin Flasch
- ▶ Combine LDRSchedule and LDRTransfer when time permits
 - ▶ Remove latency due to asynchronous daemons
 - ▶ Cannot promise this by start of S6
- ▶ Status: testing on new hardware, no coding

Better manageability for metadata and psets

*Recall “pset” is a Publication Set...the metadata for an “interesting” set of data
“S6 LHO raw frames” is an example pset*

- ▶ Make it easy to “find” psets and the necessary metadata
- ▶ No more “priming the pump” for new pset
- ▶ Easily “retire” old psets to improve MySQL performance
- ▶ Develop new LDRMetadata services using Globus 4.2 Java WS
- ▶ Status: still coding, no surprises, slow progress

Scalable LDRdataFindServer

Used by some clients for finding data at LIGO compute site

- ▶ Current server in Python, no true threading
- ▶ Original design only for serial queries from cluster head node
- ▶ Need to support parallel queries from cluster cores
- ▶ Kevin Flasch testing skeleton written in C
 - ▶ Globus 4.2 C WS for threading, protocol handling
 - ▶ SOAP XML serialization and de-serialization in C *fast*
 - ▶ Preliminary scaling tests promising
- ▶ Will support both GSI authenticated and no-auth modes
- ▶ Status: scaling tests in progress now on service skeleton