

AliEn development

ALICE T1/T2 Workshop - Bergen

Miguel Martinez Pedreira



Infrastructure

- New Catalogue and TaskQueue machines
 - SSDs → I/O performance significantly boosted
 - RAM → More in-memory records

- Updated MySQL version based on system
 - Very few changes to [j]AliEn
 - Changes on some behaviours
 - Timeouts, deadlocks, config items, backup method ignored, even readline usage...
 - Alerts added

- Updated LDAP version based on system (OpenLDAP)
 - Replication mechanism also updated
 - From slurpd to syncrepl
 - Quite some changes in config

- Using TLS only communications

- Packages updates: OpenSSL, httpd, CAs, ApMon...
 - Xrootd 3.3.6+xalienfs 1.0.14r1 built, [under test](#)
 - Proxy certificate support in the future ?

- Unified AliEn code (CS – CVMFS – APIs)

+ Broker

- Related to DB update point: fix to **job priorities**
 - As said, new MySQL doesn't behave as before regarding the order of result sets
 - JobAgents where ordered by priority (one per user), and then the oldest queueId was taken (implicitly)
 - What about pushing the execution of a part of my own jobs ? (e.g. merging)
- Introduced the use of 'Price' and 'Oldest Queue Id'





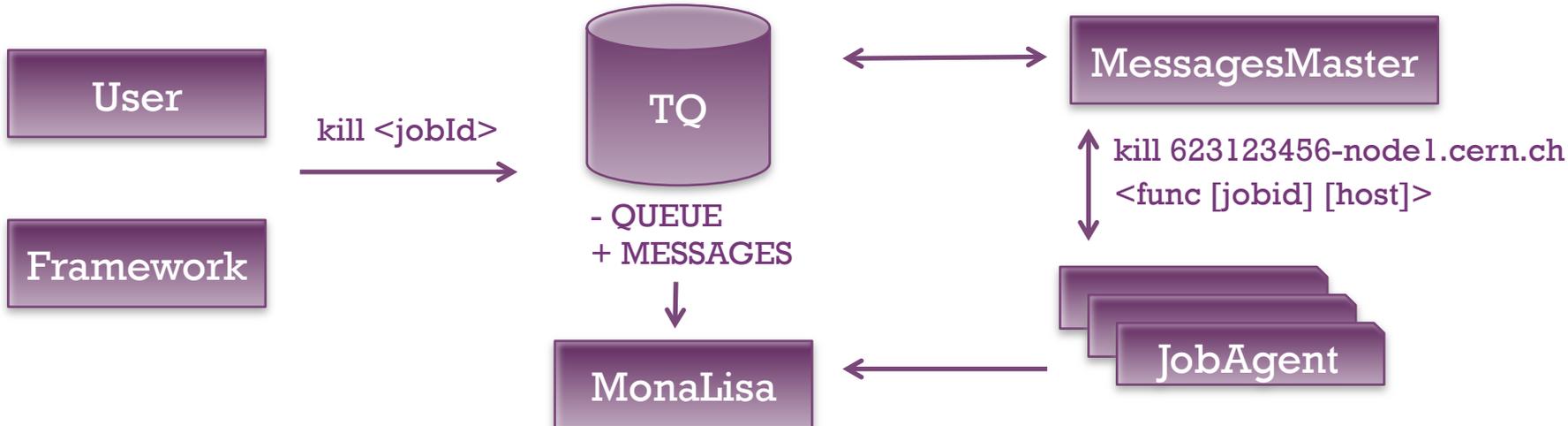
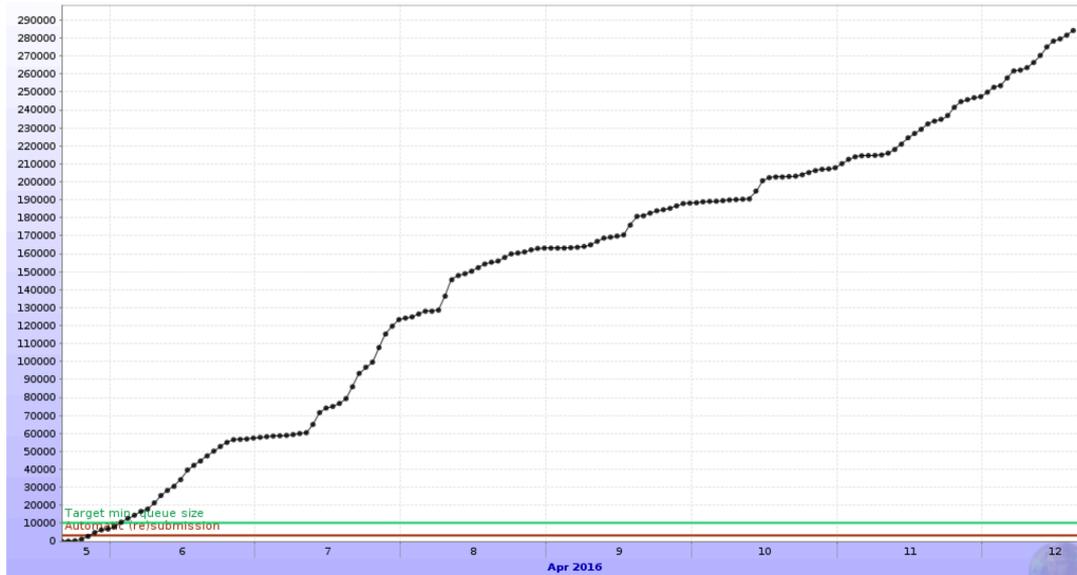
Site / JobAgent

- Sending batch job parameters to MonaLisa
 - Helpful for site admins to debug jobs, linking queueId with batchId
 - SLURM_JOBID, SLURM_JOB_ID, LSB_BATCH_JID, LSB_JOBID, PBS_JOBID, JOB_ID, CREAM_JOBID → More to add ?
- ClusterMonitor startup
 - Recreation of httpd/conf automatically
 - Renaming old to httpd/conf.<timestamp>
 - Can be avoided touching '\$ALIEN_HOME/httpd/keep'
 - Checking if previously running, proper kill
 - Storing reliable PID to stop/start
 - Was failing in some cases...
- Batch interfaces update: ARC, SGE. SLURM soon



Site / JobAgent

- Kill payload when jobs are killed





Site / JobAgent

- Perl zombie processes in WNs
 - Problem found and fixed soon after last workshop
 - Validation fork
 - No more vulnerabilities reported so far

```

alicesgm 15737 0.0 0.0 3928 252 ?      SN   Dec03  0:00      |          \_ /usr/bin/time bash arc-submit.32315.sh
alicesgm 15738 0.0 0.0 108372 1084 ?     SN   Dec03  0:00      |          \_ bash arc-submit.32315.sh
alicesgm 15740 0.0 0.0 267520 38804 ?    SN   Dec03  0:02      |          \_ /cvmfs/alice.cern.ch/x86_64-2.6-gnu-4.1.2/Packages/AliEn/v2-19-223/bin/perl -w -I/cvmfs/alice.cern.ch/x86_64-2.6-gnu-4.1.2/Packages/AliEn/v2-19-223/scripts/Service.pl JobAgent
alicesgm 15842 0.1 0.1 290284 59532 ?    SN   Dec03  2:05      |          \_ /cvmfs/alice.cern.ch/x86_64-2.6-gnu-4.1.2/Packages/AliEn/v2-19-223/bin/perl -w -I/cvmfs/alice.cern.ch/x86_64-2.6-gnu-4.1.2/Packages/AliEn/v2-19-223/scripts/Service.pl JobAgent
alicesgm 969 0.0 0.0 0 0 ?         ZN   00:43  0:00      |          | \_ [perl] <defunct>
alicesgm 1525 0.0 0.0 0 0 ?         ZN   00:44  0:00      |          | \_ [perl] <defunct>
alicesgm 17991 0.0 0.0 0 0 ?         ZN   03:20  0:00      |          | \_ [perl] <defunct>
alicesgm 18259 0.0 0.0 0 0 ?         ZN   03:21  0:01      |          | \_ [perl] <defunct>
alicesgm 32653 0.0 0.0 0 0 ?         ZN   08:48  0:00      |          | \_ [perl] <defunct>
alicesgm 532 0.0 0.1 290284 59304 ?    SN   08:48  0:01      |          | \_ /cvmfs/alice.cern.ch/x86_64-2.6-gnu-4.1.2/Packages/AliEn/v2-19-223/bin/perl -w -I/cvmfs/alice.cern.ch/x86_64-2.6-gnu-4.1.2/Packages/AliEn/v2-19-223/scripts/Service.pl JobAgent

```

- Fix for procInfo in the JA
 - Some resource usage values and error handling
- Xrootd proxy (GSI): polished code and running smoothly (nearby users relocated)
- Fix for CVMFS environment parsing
- CVMFS in custom location (CVMFS_PATH)
- APISCONFIG automatically loaded if missing



Optimizers

- Job optimizers probing+forking now works
 - DB handlers were in bad state after forking and reusing
 - First attempt: just create a new one per child
 - Not working smoothly
 - Second attempt: clone parent handler + flags
 - InactiveDestroy
 - No problem on forking and getting proper results since then!
- We need more monitoring for the Optimizers
 - Right now, checked through the parent → not enough!
 - A possibility is to use PIDs of the processes and subprocesses and establish status and thresholds checks
- CVMFS -> OCDB optimizer
 - +100K files published
 - Catching errors, but none in a long time, auto-retry if failed (with a period in between publications)
 - Fixed slow performance after long runtime: module load to *ISA* on each `get`
- SPLITTING
 - Quotas calculations optimized
 - Introduced SortInputDataCollection & SortSubjobInputData



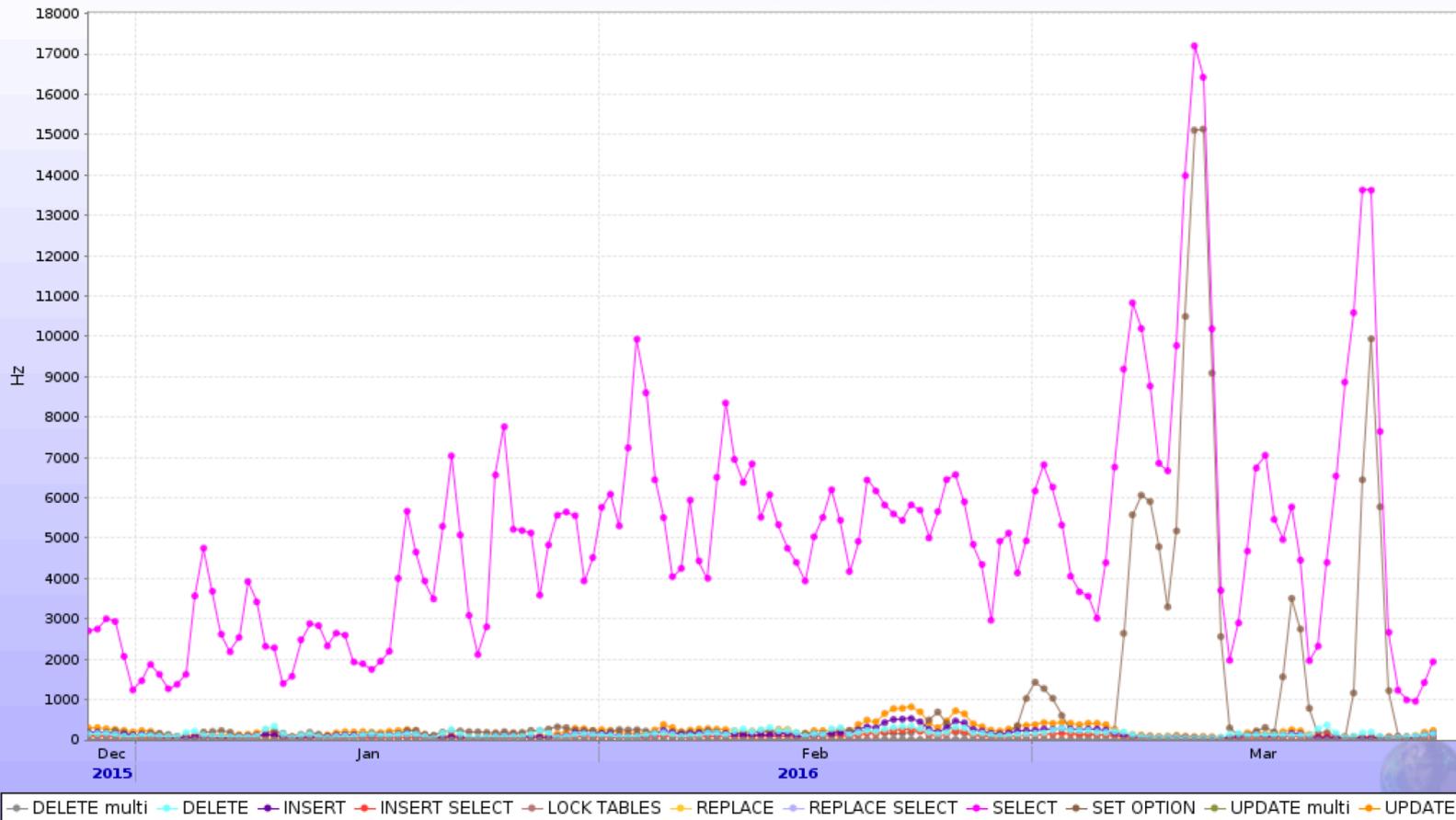
Catalogue: Status

- Powerful server
 - ProLiant DL380 Gen9
 - Xeon E5-2687W v3 3.10GHz
 - 40 cores
 - 755 GB RAM
 - SDD-based
 - MySQL 5.6 on latest Ubuntu
 - ~2 TB on disk – 2.7B PFNs, 2.1B LFNs – AVG load 4.5
 - Growing several M entries/day
 - Reached rates of 10-15M / day (many files per subjob)
 - Restoring takes more than a day, *optimize table* could save up to half a TB

Machines status

db06c																								
Machine status				Machine type						Disk	CPU utilisation (%)							Memory utilisation					Swap	
Machine	Online	Uptime	Load	Kernel	Machine model	CPU	CPUs	MHz	Space	usr	sys	iow	int	sint	steal	nice	idle	Total	Used	Buffers	Cached	Free	Used	Free
1. db6c	Online	284d 1:31	3.85	3.19.0-21...	ProLiant DL380 Gen9	Xeon E5-2687W v3 3.10GHz	40	1200	7.538	0.936	0.093	0	0.629	0	0	0	90.8	755.8 GB	298.4 GB	197.4 MB	454.4 GB	2.867 GB	0	0
Total							40			7.538	0.936	0.093	0	0.629	0	0	90.8	755.8 GB	298.4 GB	197.4 MB	454.4 GB	2.867 GB	0	0
Average		284d 1:31	3.85							7.538	0.936	0.093	0	0.629	0	0	90.8	755.8 GB	298.4 GB	197.4 MB	454.4 GB	2.867 GB	0	0

Command rates on aliendb06c.cern.ch



Command rates on aliendb06c.cern.ch

	Series	Last value	Min	Avg	Max	Total
1.	DELETE multi	0	0	0	0	0
2.	DELETE	155.2	0	147.6	4344	1148498564
3.	INSERT	151.5	0	139.6	2051	1086667046
4.	INSERT SELECT	87.47	0	76.66	1189	596562145
5.	LOCK TABLES	0.082	0	0.058	2.492	454111
6.	REPLACE	79.42	0	78.47	956.7	610595643
7.	REPLACE SELECT	0	0	0	0	0
8.	SELECT	1937	0	5010	23333	38987115405
9.	SET OPTION	158.6	0	961.7	21260	7483238637
10.	UPDATE multi	0	0	0	0	0
11.	UPDATE	231.8	0	223	2339	1735538108
	Total	2801		6637		51648669663



Catalogue: Alternatives summary

- Mapping to FS
 - Btrfs or similar
 - CVMFS + Key-Value
- Simplification (guidless) + partitioning of current schema

+ Mapping to FS - Tool

- Created a dumper tool
 - Based on jAliEn
 - Few things missing added
 - Nicely manage threads with executors
- Creates the hierarchy
 - JSON files
- Archives
 - Content in archive file
 - Members are symbolic links
- Logs files/folders/collections
 - Discovered orphan entries to cleanup

```

root@pcalienstorage:/catalogue/jalien/alice/cern.ch/user/m/mmmartin/tutorial/output# ls -l
total 16
-rw-r--r-- 1 root root 705 Mar 10 17:47 myTestJobArchive.zip
lrwxrwxrwx 1 root root 20 Mar 10 17:47 resources -> myTestJobArchive.zip
lrwxrwxrwx 1 root root 20 Mar 10 17:47 stdout -> myTestJobArchive.zip
lrwxrwxrwx 1 root root 20 Mar 10 17:47 tut_jobs_output.file -> myTestJobArchive.zip
root@pcalienstorage:/catalogue/jalien/alice/cern.ch/user/m/mmmartin/tutorial/output# cat myTestJobArchive.zip
{"guid": "7aacb3fe-4d17-11e3-a9f4-1342442a9ec4",
 "pfns": [{"pfn": "root://dp0014.m45.ihep.su:1094//09/07945/7aacb3fe-4d17-11e3-a9f4-1342442a9ec4",
 "se": "ALICE::IHEP::SE"},
 {"pfn": "root://grid-se.chpc.ac.za:1094//09/07945/7aacb3fe-4d17-11e3-a9f4-1342442a9ec4",
 "se": "ALICE::ZA_CHPC::SE"}],
 "gowner": "mmmartin",
 "md5": "8a6ad2ee604a356a0d19961f1256653e",
 "owner": "mmmartin",
 "zip_members": [{"md5": "15886956c1aac80faab0fac026be03c7",
 "lfn": "resources",
 "size": "1311"},
 {"md5": "2b772766489349c31e70b8cce238e33d",
 "lfn": "tut_jobs_output.file",
 "size": "90"},
 {"md5": "0b72e6cec5ce1fe7dcd82478246880a",
 "lfn": "stdout",
 "size": "1344"}],
 "perm": "755",
 "ctime": "2013-11-14 11:28:54",
 "jobid": "334916689",
 "size": "1365"}root@pcalienstorage:/catalogue/jalien/alice/cern.ch/user/m/mmmartin/tutorial/output#

```

+ btrfs

- First try in standard ext4 in Ubuntu
 - Ran out of inodes after some million entries, as expected
- BTRFS (Binary trees)
 - Max number of files: 2^{64}
 - Max volume size: 16 EiB
 - Space-efficient packing of small files
 - Space-efficient indexed directories
 - Dynamic inode allocation
 - Writable snapshots, read-only snapshots (backups!)
 - Compression (zlib and LZO)
 - SSD (Flash storage) awareness
 - More: https://btrfs.wiki.kernel.org/index.php/Main_Page

```
[root@pcalienstorage:/catalogue/jalien/json_out# df -h .
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda5       7.7T  971G  6.7T  13% /catalogue
[root@pcalienstorage:/catalogue/jalien/json_out# df -i .
Filesystem      Inodes IUsed IFree IUse% Mounted on
/dev/sda5       0      0     0    -  /catalogue
```

+ Mapping to FS - Timing

- About creation
 - Rate depends on folder and usage at the moment...
 - Done with master and slave (mostly master because of backups)
 - 50 threads = +~5 load
 - Optimize
 - Use N slaves
 - Load G space in memory
 - Servers with decent RAM...
- Examples (disk)
 - /alice/data/2010/ - 57.2M entries – 7h – 15 threads
 - /alice/data/2012/ - 141.6M entries – 37h – 20 threads
 - /alice/cern.ch/user/c/ - 3.5M entries – 40 min – 15 threads
 - /alice/sim/2014/ - 215M entries – 27h – 12 threads
- Examples (ssd)
 - /alice/data/2012/ - 120M entries – 21h - 15 threads
 - /alice/data/2013/ - 69M entries - 13h – 10 threads
 - /alice/data/2014/ - 362K – 40m – 15 threads

+ Initial benchmark and issues

- What about (both for CVMFS or btrfs FC):
 - Quotas, SE lookups, booking tables+locking...
- Benchmarking tool similar to the dump
 - Goes over base directory
 - Times the command/lfn
 - Tried with `ls` over each directory: FS saturated with just 10 threads, >1 s/ls

03/24/16 17:17:15

```
avg-cpu: %user  %nice %system %iowait  %steal  %idle
          9.84  0.00   6.60  13.13   0.00  70.43
```

```
Device:   rrqm/s  wrqm/s   r/s    w/s  rkB/s  wkB/s avgrq-sz avgqu-sz  await r_await w_await svctm  %util
sda      0.00    0.40 1921.40  1.60 29697.60  12.80  30.90   4.92   2.56  2.56  0.00  0.52 100.08
```

- Preparing yet more dumps on SSD based btrfs
 - Provided SSD iops are greatly bigger
 - FS on disk discarded...



Type	Threads	Folders	Ms / ls
Local SSD	1	10K	3 hot 13 cold
Local SSD	10	1M	2-3
Local SSD	50	1M	13
Local SSD	100	1M	26
NFS SSD	1	10K	114 ?
NFS SSD	10	1M	39 ?
DB (test02)	1	10K	5-7
DB	10	1M	7
DB	50	1M	23

30.03.2016 17:00:20													
avg-cpu:	%user	%nice	%system	%iowait	%steal	%idle	100 threads local ssd						
	46.26	1.24	35.84	2.99	0.00	13.67							
Device:	rrqm/s	wrqm/s	r/s	w/s	rkB/s	wkB/s	avgrq-sz	avgqu-sz	await	r_await	w_await	svctm	%util
sda	665.60	7494.20	231.00	1244.60	4411.20	34972.80	53.38	0.47	0.32	0.25	0.34	0.11	16.16
sdb	138.20	0.00	3745.40	0.00	62200.00	0.00	33.21	0.51	0.14	0.14	0.00	0.08	31.28

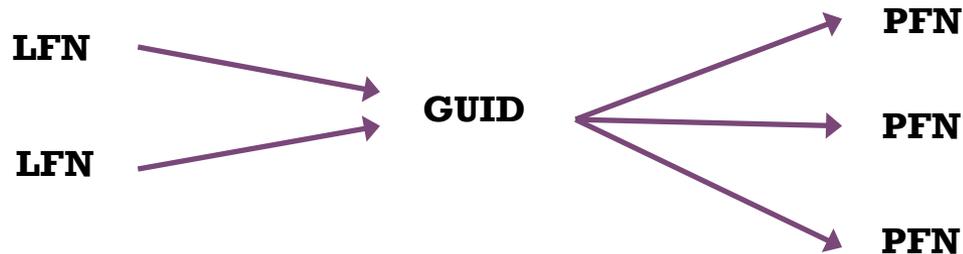
30.03.2016 16:11:06													
avg-cpu:	%user	%nice	%system	%iowait	%steal	%idle	10 threads local ssd						
	47.85	1.16	35.05	3.79	0.00	12.14							
Device:	rrqm/s	wrqm/s	r/s	w/s	rkB/s	wkB/s	avgrq-sz	avgqu-sz	await	r_await	w_await	svctm	%util
sda	618.40	12.00	278.80	53.40	4977.60	263.20	31.55	0.09	0.26	0.28	0.19	0.13	4.32
sdb	97.60	0.00	3040.40	0.00	50278.40	0.00	33.07	0.42	0.14	0.14	0.00	0.09	28.00

+ CVMFS + Key-value approach

- CVMFS uses a similar data-representation as the AliEn FC
 - Main difference: the structure (SQLite based) is sent to the clients and DBs splitted per directories
 - The hard work relies now on the client!
 - We could use CVMFS as logical namespace
 - Can we create a namespace tool that everybody can use in the same way ?
 - And is known and trustable
 - We hold the metadata in attributes
 - Complemented by a 'service' that takes care of authorization
 - Namespace would be public but access to files
 - PFNs on key-value store using a key, like the LFN e.g.
 - What do we do about files that are written and just after read ?
 - The delay is on the best case in the order of some minutes (Jakob)
 - Merging of jobs, user space...
 - Which other features need to be addressed?
 - SE and User quotas: column-based could be tried instead ?
 - Booking table
 - Collections
 - Metadata (tags)
 - Security

+ GUIDless catalogue

- GUIDs provide us with extra flexibility in the catalogue
 - But in practice is not used (perms for links, mirrors, lfn name changes)
 - Extra lookups + space

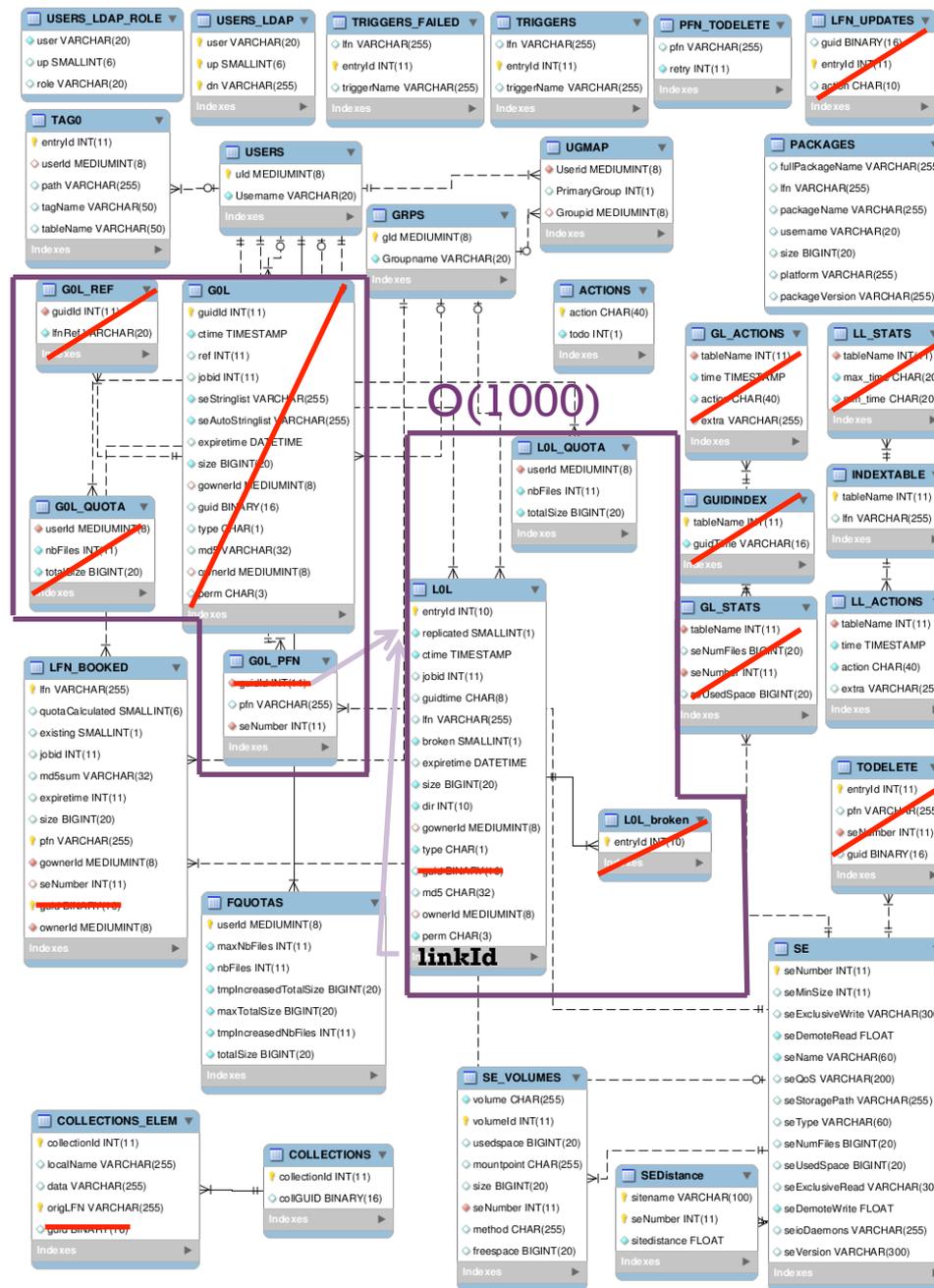


- We still lack having more integrity and optimizing the catalogue
 - FK + surrogated keys
 - Query minimization
 - Using InnoDB everywhere
 - Row-level locks, and more
 - Indexes
 - Table cleanups and split, collections, tags...



O(200)

O(1000)



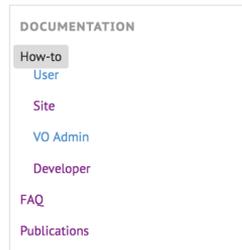
+ GUIDless catalogue

- Use LFN+timestamp for uniqueness
 - We could keep pfn naming as in old version
 - Though LFN is more human readable
- What do we get:
 - ~40% of the catalogue goes away
 - From GUID tables and PFN links
 - All the LFN-PFN lookups also gone
 - We can still keep links in the way are used in ALICE (previous slide)
 - Simplicity!
- We can also partition
 - Code and DB are ready for this
 - Data and Users
 - This can be done in N servers...



+ AliEn web

- New Drupal web in → <http://alien.web.cern.ch/>
 - Old URL redirects here
 - Intuitive URIs to Docs
 - http://alien2.cern.ch/index.php?option=com_content&view=article&id=140&Itemid=144
 - <https://alien.web.cern.ch/content/documentation/howto/site/eosinstallation>



Documentation

Please choose one of the categories below:

- Howto - A multitude of detailed howto pages.
 - User Howto - AliEn at User Level
 - Site Howto - AliEn at Computing Site Level
 - VO Admin Howto - AliEn at Virtual Organization Level
 - Developer Howto - Developing AliEn Startup Guide
- FAQ - Frequently Asked Questions
- Publications - A list of scientific publications related to AliEn.

+ jAliEn

- JobAgent
 - Full chain now working!
 - With CVMFS PackMan: environment setup
 - Without connection through ROOT
 - All features from Perl implemented
 - Site services pending
 - To be used in Titan: see Pavlo's slides
- Brokering fully working as well
- Hopefully soon: TJAliEn ROOT plugin
- CS to be gradually adopted
 - Starting with SPLIT optimizer



+ Some To-Dos

- Improve splitting/inserting forking algorithm
 - Now: the parent splits 1 process and gives 15 jobs to it
 - Good when there are many masterJobs to split, not in the opposite case
- JDL for basket compaction
- Block CE matching if failed N times
- Hard stop for sites
 - Kill active (running, assigned, started) jobs and resubmit them blocking that site in requirements
- Federations
- From alienv, incompatibility detection

+ Questions / Comments

- Record of 102K+ running jobs 😊
- We ~~still miss~~ have a *real* test environment
 - Replica of TQ, Catalogue, ADMIN dbs...
 - Using production LDAP (could be cloned too)
 - Running usual central services!
 - Of course, without all the concurrency of the GRID...
- Questions?