

### New backend for the AliEn File Catalogue

Miguel Martinez Pedreira



A Large Ion Collider Experiment

European Organisation for Nuclear Research



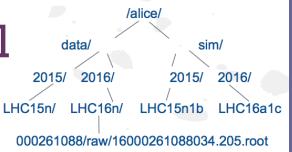


## + Current implementation

- MySQL-based AliEn File Catalogue
  - 3B logical entries
- One (powerful) DB master
  - 1.5TB RAM, 2.4TB on disk size
- DB slaves for hot standby / backups
  - 4h to dump, ~2 days to restore

	Machine status Machine					e type	Disk CPU utilisation (%						%)				Mem	ory utilisat	ion				
Machine	Online	Uptime	Load	Kernel	OS	Machine model	CPU	CPUs	MHz	Space	usr	sys	iow	int	sint	steal	nice	idle	Total	Used	Buffers	Cached	Free
1. db6c		20d 22:34	5.71	4.4.0-64	16.04	ProLiant DL380 Gen9	Xeon E5-2667 v4 3.20GHz	32	3500		9.981	1.15	0.276	0	0.365	0	0	88.23	1.476 TB	213.1 GB	226.5 MB	1.264 TB	3.989 GB
Total								32											1.476 TB	213.1 GB	226.5 MB	1.264 TB	3.989 GB

# Catalogue in a nutshell



#### LFN namespace

- /alice/data/2016/LHC16n/000261088/raw/16000261088034.205.root
- 1180 tables (max 50M), 3B entries, namespace split into tables
- Metadata

-rwxr-xr-x alidaq alidaq 264403565 Sep 09 22:10 0f24bce32446ea22840d188e035b11a9

#### GUID namespace

- 76CEBD12-76A0-11E6-9717-0D38A10ABEEF
- 173 tables (max 210M), 2.8B entries, split by time intervals (append)
- Version 1 UUIDs (MAC+timestamp)

### Physical File Pointers

root://alice-tape-se.gridka.de:1094//10/33903/76cebd12-76a0-11e6-9717-0d38a10abeef root://voalice10.cern.ch//castor/cern.ch/.../16000261088034.205.root

3.5B entries, 1B physical files, pointers to ZIP members, 70PB over 70 Storage Elements

# Catalogue in a nutshell

### LFN namespace

- /alice/data/2016/LHC16n/000261088/raw/16000261088034.205.root
- 1180 tables (max 50M), 3B entries, namespace split into
- Metadata
- -rwxr-xr-x alidaq alidaq 264403565 Sep 09 22:10 0f24bce32

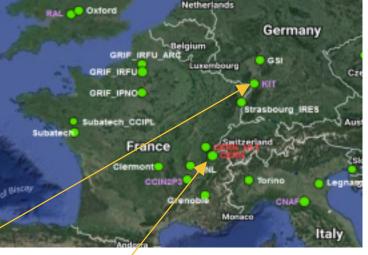
### GUID namespace

- 76CEBD12-76A0-11E6-9717-0D38A10ABEEF
- 173 tables (max 210M), 2.8B entries, split by time interva
- Version 1 UUIDs (MAC+timestamp)

### Physical File Pointers

root://alice-tape-se.gridka.de:1094//10/33903/76cebd12-76a0-11e6-97/7-0d38a10abeef root://voalice10.cern.ch//castor/cern.ch/.../16000261088034.205.root/

3.5B entries, 1B physical files, pointers to ZIP members, 70PB over 70 Storage Elements

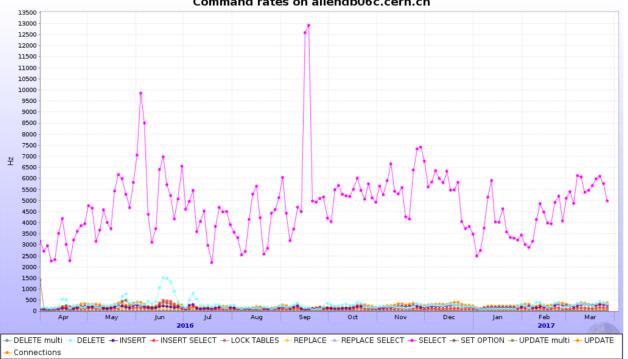




Averages (1y)

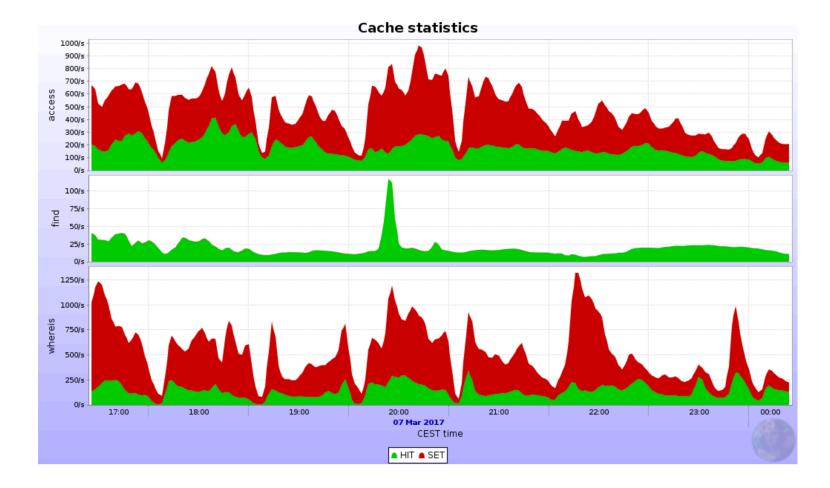
9223 Hz Reads 618 Hz Changes 282 Hz Deletes

- 77500 running jobs
- 15:1 select/change ratio
- **10:1** read/write data volume

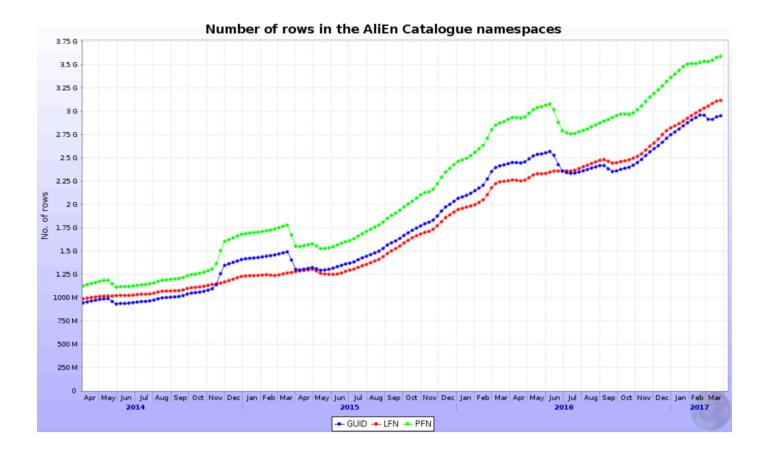


Command rates on aliendb06c.cern.ch





### + The AliEn catalogue in time



## + Future needs

- In Run3 we will have 5x more computing resources (300K CPUs + 5000GPUs)
- 10x more disk and tape storage =>
- ~10x more files to manage
- The goal is to sustain ~200kHz queries (stable) ~1Mhz queries (peaks)
  - Numbers are a bit *inflated* 
    - Query cache represents half or more of the select
    - Many of those queries won't be needed:
      - New backend schemas simplify (next slides)
      - Improvements on the framework
        - Preparing file envelopes for jobs at split
        - More aggressive caching in JAliEn
- Looking for a solution providing:
  - Horizontal scaling
  - No single point of failure
  - High query rate
  - HA

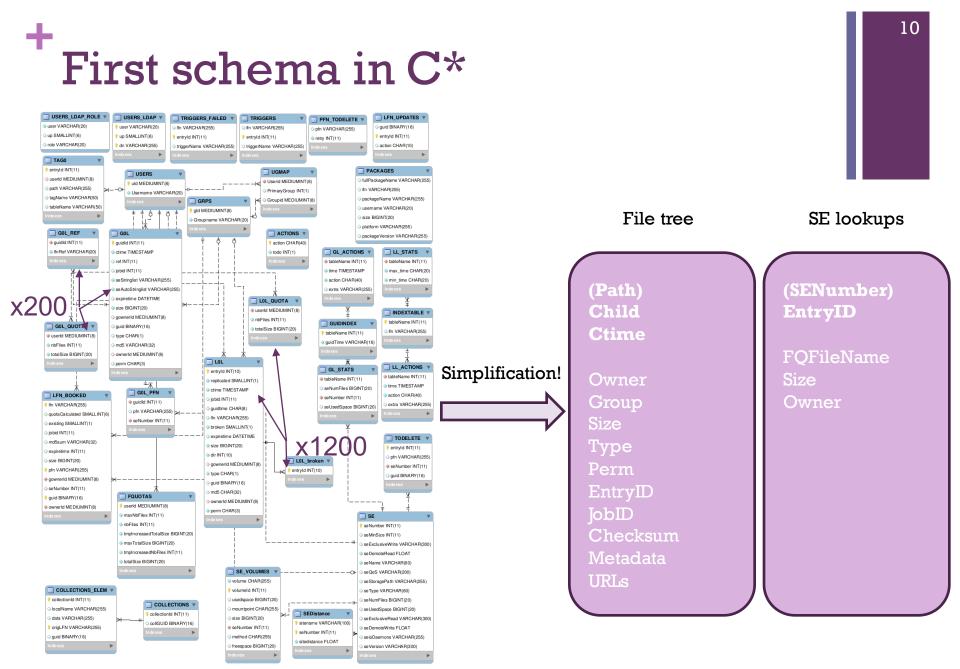
# Apache Cassandra



- Provides all the requirements mentioned before:
  - Horizontal scale
    - Add nodes to keep up ops/s
  - HA No point of failure
  - Performance (see later initial benchmarks)
- Consistency
  - Tunable levels, key factor for us
- We move from N to few tables for the namespace
  - Simplification
- Easy setup
- Mapping certain SQL operations not trivial
  - Groupings, quota calculations, 'where' possibilities...
  - NoSQL re-implementation, CQL helps too

#### Cassandra Write Data Flows





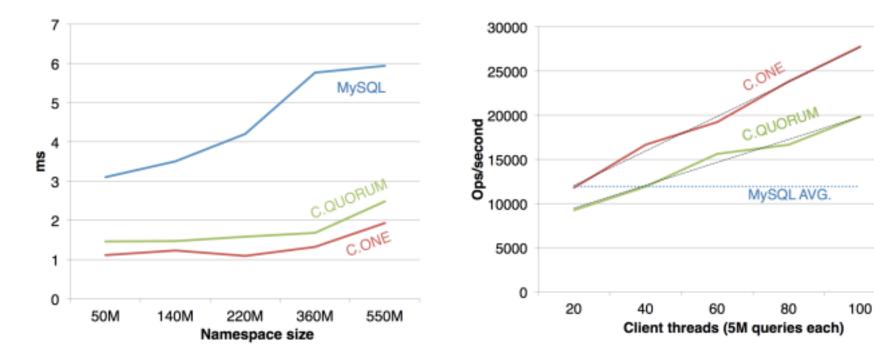
PS: some tables will stay in MySQL

# Cassandra benchmarks

- Setup a 5-node ring
  - Server power: 16-48 cores, 100-350GB RAM
  - Java 8 Oracle, no swap, nofile/memlock limits (no degraded mode)
  - Mapped namespace into a column family that is able to do `whereis` and `ls`: entry contains lfn+pfns metadata
    - Starting a new round of benchmarking on a implementation that allows `find` as well
  - RF 3, LeveledCompaction + LZ4 compression
- Data dump
  - MySQL to Cassandra -> slow
  - Artificial lfns and dirs -> very quick!
- Execution
  - Java sized thread pool, configure hierarchies, number of LFNS, etc...

# + Cassandra benchmarks

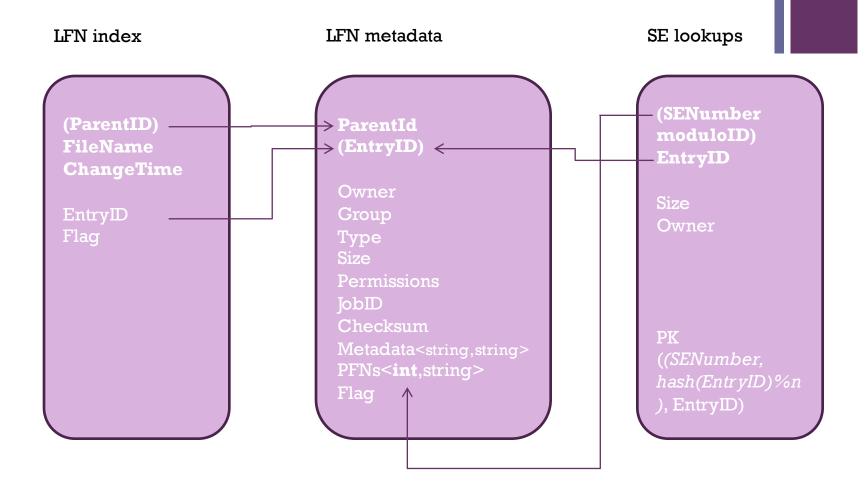
Initial benchmarking shows promising results



Time to retrieve logical and physical information of a file

Operations per second based on number of clients





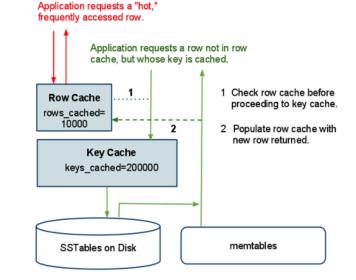
### + New vs old schema

#### Cons

- Need to loop over the lfn\_index (hierarchy) to do file operations
  - To avoid contention on the servers and thus latency:
    - Can be cached by client
    - (We hope) can be cache by Cassandra -> RowCache

#### Pros

- Some complex and heavy operations become much easier
  - mv dir: delete old parent and insert new
  - rm dir: mark/delete parent
- Easy to keep a trash bin

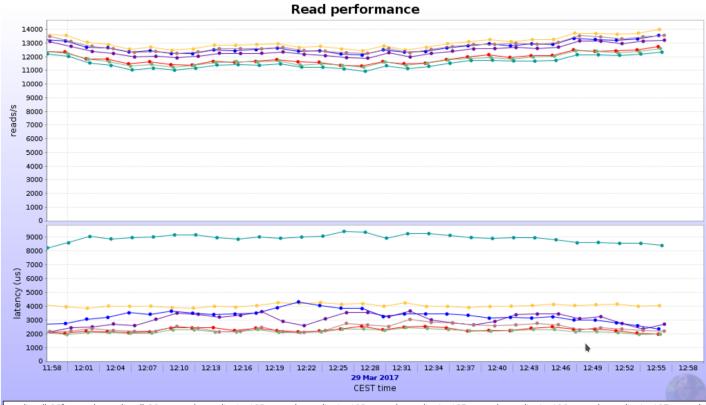


### -Monitoring

- Cassandra internally calculates and exposes an ample set of metrics -> MBean (JMX)
- Naming is misleading and documentation is scarce
- Pluggable to some extended tools
- ML will have a dashboard to have a global view of the cluster and detect problems -> <u>link to C\* monitor</u>
- Feeding most important metrics:
  - Read/Write latencies and throughput
  - KeyCache hits/requests
  - Compaction+GC stat
  - Timeouts, Unavailable, Exceptions
  - CF stats
  - Usual machine status: load, cpu/memory/network usage...

# + Stress test: read

1. 2. 3. 4. 5. 6. 7.



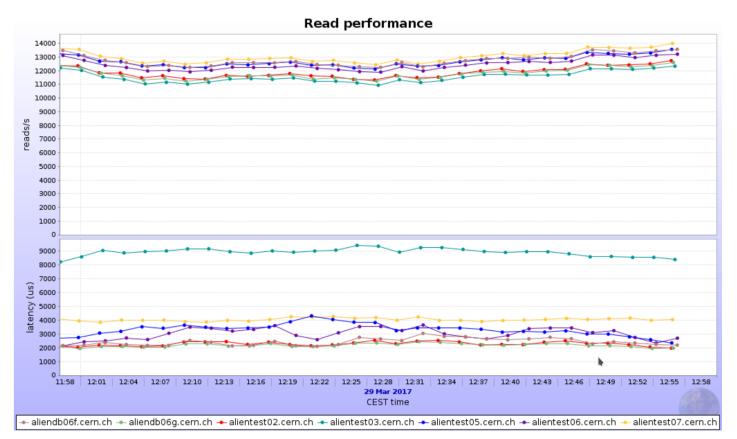
🔶 aliendb06f.cern.ch 🛥 aliendb06g.cern.ch 🛥 alientest02.cern.ch 🛥 alientest03.cern.ch 🛥 alientest05.cern.ch 🛥 alientest06.cern.ch 🔸 alientest06.cern.ch

Read o	perations /	second		
Series	Last value	Min	Avg	Max
aliendb06f.cern.ch	13569	12047	12778	13701
aliendb06g.cern.ch	12597	11061	11784	12727
alientest02.cern.ch	12737	11216	11876	12738
alientest03.cern.ch	12333	10850	11550	12415
alientest05.cern.ch	13563	12002	12705	13625
alientest06.cern.ch	13216	11679	12445	13299
alientest07.cern.ch	13995	12338	13039	14106
Total	92014		86179	

	Avera	ge latency	(us)		
	Series	Last value	Min	Avg	Мах
1.	aliendb06f.cern.ch	2194	1960	2426	3108
2.	aliendb06g.cern.ch	1974	1864	2176	2534
3.	alientest02.cern.ch	1969	1836	2276	2698
4.	alientest03.cern.ch	8394	7785	8923	9571
5.	alientest05.cern.ch	2349	2342	3308	4982
5.	alientest06.cern.ch	2692	1969	3023	3845
7.	alientest07.cern.ch	4040	3767	4041	4670
	Total	23614		26176	

	ദ
	-

# + Stress test: read

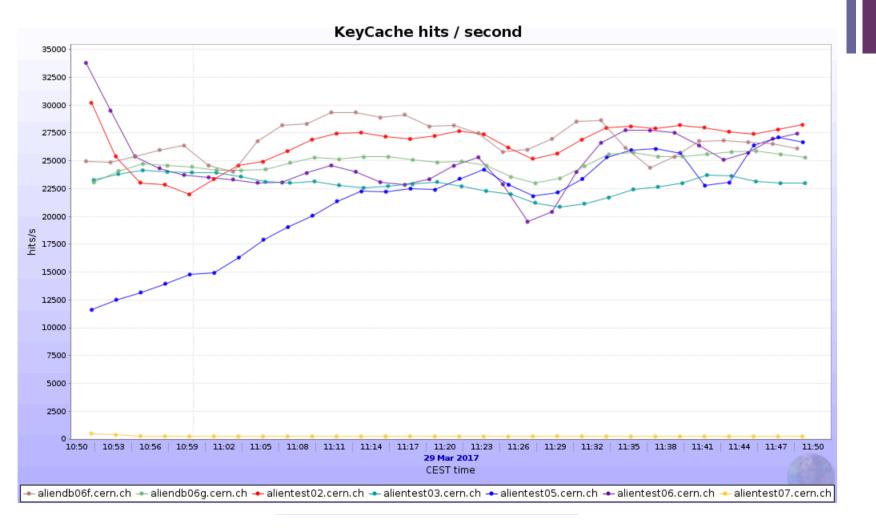


What is this about?

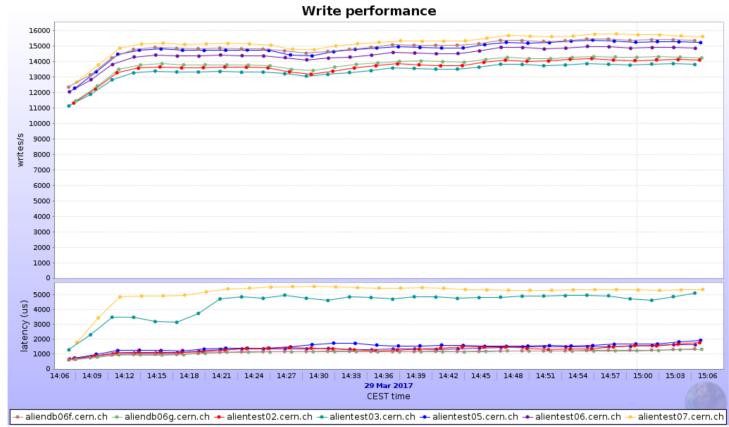
#### Machines status

alientest,db06f,d	d																							
Machine status			IS			Machine	e type			Disk	CPU utilisation (%)								Memory utilisation					
Machine	Online	Uptime	Load	Kernel	OS	Machine model	CPU	CPUs	MHz	Space	usr	sys	iow	int	sint	steal	nice	idle	Total	Used	Buffers	Cached	Free	
1. db6f		22d 20:04	18.13	4.4.0-65	16.04	ProLiant DL380 Gen9	Xeon E5-2687W v4 3.00GHz	48	3199		22.33	6.738	0.034	0	0.564	0	1.921	68.41	755.8 GB	178 GB	364.6 MB	571.7 GB	5.762 GB	
2. db6g		19d 19:41	14.86	4.4.0-66	16.04	ProLiant DL380 Gen9	Xeon E5-2687W v3 3.10GHz	40	1768		25.13		0.01	0	0.902	0	2.253	63.85	755.8 GB	19.51 GB	279.8 MB	97.95 GB	638.1 GB	
3. alientest02		83d 21:23	28.8	4.4.0-57	16.04	ProLiant DL380p Gen8	Xeon E5-2690 v2 3.00GHz	40	3000		30.79	11.07	0.002	0	1.532	0	2.477	54.13	377.9 GB	172 GB	10.56 GB	163.6 GB	31.74 GB	
4. alientest03		76d 17:58	62.59	4.4.0-59	16.04	ProLiant DL380 G6	Xeon X5560 2.80GHz	16	2794		65.14	12.19	0.001	0	4.269	0	4.013	14.39	141.7 GB	17.98 GB	332.3 MB	94.68 GB	28.67 GB	
5. alientest05		8d 1:53	37.64	4.4.0-67	16.04	ProLiant DL380p Gen8	Xeon E5-2697 v2 2.70GHz	48	2699		34.74	12.37	0.035	0	2.039	0	3.843	46.97	188.9 GB	57.67 GB	3.423 GB	127.1 GB	690.1 MB	
6. alientest06		83d 21:24	58.49	4.4.0-57	16.04	ProLiant DL380p Gen8	Xeon E5-2697 v2 2.70GHz	48	2999		54.8	12.55	0.007	0	2.689	0	3.006	26.94	188.9 GB	62.19 GB	2.689 GB	99.19 GB	24.81 GB	
7. alientest07		83d 21:26	61.68	4.4.0-57	16.04	ProLiant DL380 G6	Xeon X5560 2.80GHz	16	2794		48.34	18.14	11.69	0	6.412	0	6.288	9.125	55.03 GB	16.45 GB	209.8 MB	38.14 GB	244.1 MB	
Total								256											2.406 TB	523.7 GB	17.83 GB	1.164 TB	730 GB	
Average		54d 3:59	40.31								40.18	11.56	1.683	0	2.629	0	3.4	40.55	352 GB	74.82 GB	2.548 GB	170.3 GB		





### + Stress test: write

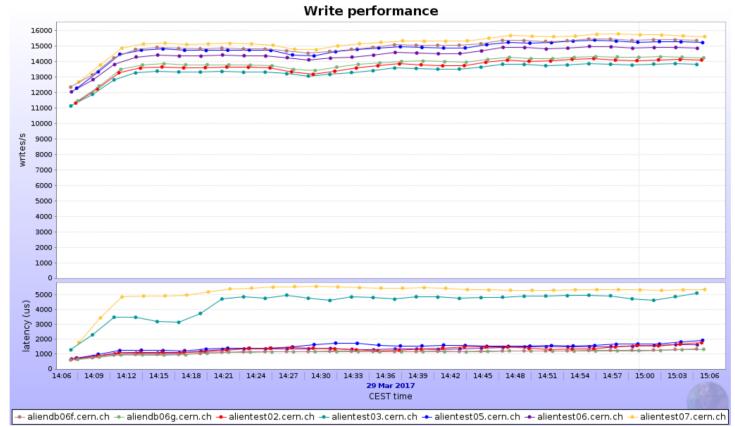


	Avera	ge latency	(us)		
	Series	Last value	Min	Avg	Max
1.	aliendb06f.cern.ch	1348	570.4	1140	1380
2.	aliendb06g.cern.ch	1321	617.2	1117	1338
3.	alientest02.cern.ch	1763	642.8	1325	1824
4.	alientest03.cern.ch	5093	1135	4378	5625
5.	alientest05.cern.ch	1927	678.7	1489	2030
6.	alientest06.cern.ch	1619	640.8	1321	1703
7.	alientest07.cern.ch	5340	1391	5126	5748
	Total	18414		15900	

write operations / second	Write	operations l	second
---------------------------	-------	--------------	--------

Series	Last value	Min	Avg	Мах
1. 📕 aliendb06f.cern.ch	15361	12150	14885	15571
2. aliendb06g.cern.ch	14243	11282	13834	14425
3. alientest02.cern.ch	14106	11193	13651	14336
4. alientest03.cern.ch	13814	10900	13377	14029
5. alientest05.cern.ch	15230	12097	14801	15471
6. alientest06.cern.ch	14858	11751	14417	15103
7. alientest07.cern.ch	15625	12342	15203	15923
Total	103241		100171	

# + Stress test: write



	about?	

	Machines status																								
alientest,db06																									
	Ma	chine statu	IS			Machin	e type			Disk			CPU	utili	sation	(%)				Memory utilisation					
Machine	Online	Uptime	Load	Kernel	os	Machine model	CPU	CPUs	MHz	Space	usr	sys	iow	int	sint	steal	nice	idle	Total	Used	Buffers	Cached	Free		
1. db6f		22d 23:20	16.25	4.4.0-65	16.04	ProLiant DL380 Gen9	Xeon E5-2687W v4 3.00GHz	48	3199		23.4	7.453	0.069	0	0.74	0	1.803	66.53	755.8 GB	177.7 GB	362.8 MB	544.3 GB	33.51 GB		
2. db6g		19d 22:57	16.71	4.4.0-66	16.04	ProLiant DL380 Gen9	Xeon E5-2687W v3 3.10GHz	40	2786		26.68		0.032	0	1.134	0	2.377	60.85	755.8 GB	19.44 GB	282.8 MB	106.8 GB	629.3 GB		
3. alientest02		84d 0:40	34.58	4.4.0-57	16.04	ProLiant DL380p Gen8	Xeon E5-2690 v2 3.00GHz	40	3000		37.16		0.051	0	3.543	0	3.853	35.93	377.9 GB	173.3 GB	10.6 GB	173.4 GB	20.55 GB		
4. alientest03		76d 21:14	56.66	4.4.0-59	16.04	ProLiant DL380 G6	Xeon X5560 2.80GHz	16	2794		49.29		0.243	0	3.634	0	10.91	26.78	141.7 GB	18.02 GB	335.8 MB	103.8 GB	19.55 GB		
5. alientest05		8d 5:10	35.96	4.4.0-67	16.04	ProLiant DL380p Gen8	Xeon E5-2697 v2 2.70GHz	48	2700		35.59	17.42	0.079	0	3.687	0	3.879	39.34	188.9 GB	48.75 GB	3.184 GB	118.1 GB	18.79 GB		
6. alientest06		84d 0:41	32.1	4.4.0-57	16.04	ProLiant DL380p Gen8	Xeon E5-2697 v2 2.70GHz	48	2999		27.68		0.045	0	1.969	0	3.362	56.53	188.9 GB	51.35 GB	2.812 GB	116.5 GB	18.18 GB		
7. alientest07		84d 0:43	53.67	4.4.0-57	16.04	ProLiant DL380 G6	Xeon X5560 2.80GHz	16	2794		53.48	12.13	0.873	0	5.487	0	6.401	21.63	55.03 GB	16.77 GB	244.1 MB	34.61 GB	3.411 GB		
Total								256											2.406 TB	505.3 GB	17.79 GB	1.169 TB	743.3 GB		
Average		54d 7:15	35.13								36.18	12.14	0.199	0	2.885	0	4.655	43.94	352 GB	72.19 GB	2.541 GB	171.1 GB			

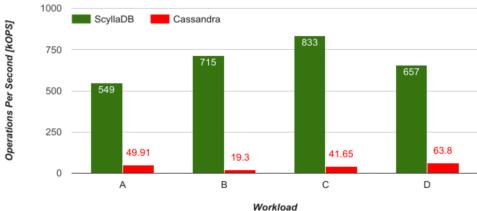


- Tune Cassandra
  - Investigate and optimize CPU usage
  - Tune JVM+GC, RowCache/KeyCache sizes...
- Run benchmarks with `cassandra-stress`
  - Comparable to similar clusters of the community?
- Get closer to a production workload
  - Mixed set of select/update/insert/delete as in the current catalogue
- Exercise critical operations
  - Backups
  - Addition/replacement of nodes



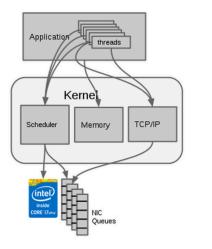


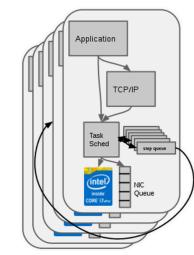
• "Next generation Cassandra"



#### Throughput of ScyllaDB Vs. Cassandra Using 2TB Data for Different YCSB Workloads

Difference relies on core implementation

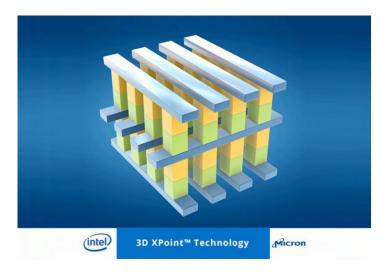




Full report here



- Biggest memory breakthrough in 25 years
- Not very clear yet how it will work, but provides:
  - Higher data volume than RAM
  - Low latency
  - <sup>1</sup>/<sub>2</sub> price RAM?
- Persistent RAM
  - Remove slow I/O layers -> In-memory DB?
  - Booking area...





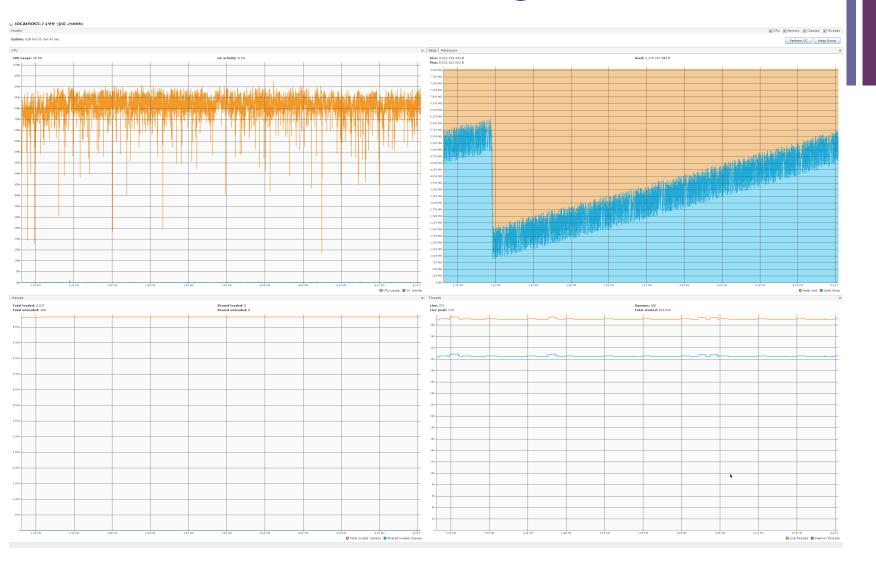
### Questions?

AliEn Catalogue - Miguel Martinez Pedreira

## + B slides - CPU debug: cpu sample

Sample: O CPU Stop				
Status: sampling inactive				
CPU samples Thread CPU Time				
O D Sapphet				Thread Dump
Hot Spots - Method	Self Time [%]  Self Time	Self Time (CPU)	Total Time	Total Time (CPU)
org, apache cossandra.utils.concurrent.WaltQueuefAbstractSignal.awaltUntil	57,953,8(39.9%)	369,809 ms	57,953,807 ms	369,809 ms 🔺
arg.apache.cossandra.concurrent.SEPViorker.rum ()	32,654,2(22.5%)	6,844 ms	105,252,394 ms	6,469,277 ms
org. apache. cossandra. concurrent. Named Thread Factory. I amb da Sthread Local De allocator 50 ()	12,032,7 (8.9%)	6,307 ms	13,584,565 ms	1,126,903 ms
org.apache.cassandra.utbls.concurrent.WaltQueuefAbstractSignal.awaitUninterruptibly ()	11,489,6 (7.9%)		11,489,692 ms	128,527 ms
org.apache.cassandra.concurrent.SEPWorker.dowaitSpin ()	8,684,97 (6%)	192,041 ms	8,686,032 ms	193,103 ms
io.netty.channel.epoll.Native.epolWait0[native] D	6,578,88 (4.5%)	6,578,886 ms	6,578,886 ms	6,578,886 ms
org. apache. cassandra. utila. CoalescingStrategies\$TimeHorizonMovingAverageCoalescingStrategi.coalescingStrategies	3,869,77 (2.7%)	6,741 ms	4,017,093 ms	8,496 ms
net.jpountz.is4.LZ48lockinputStream.readFully ()	3,847,62 (2.6%)	3,847,627 ms	3,847,627 ms	3,847,627 ms
net/pountz.1x4.LT4/HU.LZ4_decompress_fast[native] 0	B,244,10 (2.2%)	3,244,101 ms	3,244,100 ms	3,244,100 ms
org.apache.cassandra.utils.ChecksumType\$2.update ()	866,530 (0.6%)		866,530 ms	866,530 ms
org.apache.cassandra.db.committog.AbstractCommittogService\$1.run ()	430,436 (0.8%)		430,610 ms	173 ms
org.apache.cassandra.utils.ByteBufferUtil.read ()	413,215 (0.8%)		513,468 ms	513,468 ms
org.apache.cassandra.io.compress.edSequentialWriter.getOnDIskFilePoInter ()	385,808 (0.3%)		385,808 ms	385,808 ms
org.apache.cassandra.utils.concurrent.WaltQueuesRegisteredSignal.signal.()	314,885 (0.2%)		314,884 ms	314,884 ms
org.apache.cassandra.net.OutboundTcpConnection.enqueue.()	296,954 (0.2%)		296,954 ms	239,426 ms
lo.netty.channel.epoll.Native_eventFdWrite[native] ()	228,116 (0.2%)		228,116 ms	228,116 ms
org.apache.cassandra.utils.CoalescingStrategies.parkt.oop()	147,317 (0.1%)		147,317 ms	1,754 ms
lo.netty.channel.unix.FleDescriptor.writevAddresses[native] ()	113,583 (0.1%)	113,583 ms	113,583 ms	113,583 ms
io.netty.channel.unkr.FileDescriptor.readAddress[native] []	97,645 (0.1%)	97,645 ms	97,645 ms	97,645 ms
org.apache.cassandra.lo.util RebufferinginputStream.read ()	62,033 (0%)	62,033 ms	99,559 ms	99,559 ms
net/pounta:xxhash.XXHash(NIXXH32_free[native] 0	61,014 (O%)		61,014 ms	61,014 ms
org.apache.cassandra.io.util RebufferinginputStream.readUnsignedVint ()	57,502 (0%)		70,439 ms	70,439 ms
erg.apache.cassandra.io.util Buffered DataOutputStreamPlus.write ()	51,790 (ON)		131,283 ms	131,283 ms
net/pountz.1x4.LT4/HUL1Z4_compress_llmitedOutput(native) ()	51,262 (0%)	51,262 ms	51,262 ms	51,262 ms
io.netty.util.ResourceLeal/DetectorgDefaultResourceLeak. <init> 0</init>	48,862 (0%)	48,862 ms	48,862 ms	48,862 ms
org apache. cassandra. ublis bbree. BTree. slice ()	44,249(0%)	44,249 ms	44,942 ms	44,942 ms
org apache.cassandra.db.marshal.AbstractType.compare ()	43,356 (0%)	43,356 ms	44,050 ms	44,050 ms
org apache cassandra lo sotable format big BigTableReader.getPosition ()	40,695 (0%)	40,695 ms	530,425 ms	530,425 ms

### + B slides - CPU debug: GC



# B slides - CPU debug: methods/cpu usage

	Call Tree - Method	Total Time [%]	Total Time T	otal Time ( 🔻
	C[FF]alexies/team/alexies	595.	707 (100%)	595,707 ms 595,707 ms
	😉 sun nio ch. SarverSochetAdaptor accept ()	595,	707 (100%)	595,707 ms
	© 3 of me samplestrete/sciencing/0.01.194.2.1	0.0	00 ms (0%)	0.000 ms 595,707 ms
4.8	org.apache.cassandra.net.in.comingTcpConnection.run ()	595,	707 (100%)	595,707 ms
9	See any approximate and the constraints and th	595.	707 (100%)	595,707 ms
	o 🦌 net jountz bł. 1248kdoliputStream read ()	595,	707 (100%)	595,707 ms
	• a why Space 21 to 1240/columnot and m of the second seco		707 (100%)	595.707 ms
	Sun rioch. ChannelligutStream read 0	595,	707 (100%)	595,707 ms
	L © saffine - ⊙ saffine	0.01	00 ms (0%) 00 ms (0%)	0.000 ms
	Set time	0.01	00 ms (0%)	0.000 ms
	- (0 set true - 0 set true - 0 set true	0.0	00 ms (0%)	0.000 ms
	() Self time	0.01	00 ms (0%)	0.000 ms
• • M	saping/article/concentry/18.114.2.11		707 (100%)	595,707 ms
	y org apache cassandra.net.incomingTopConnection.receiveMessages ()		707 (100%)	
	• y is not Outdotoctorean reading 0     • y is not Outdotoctorean reading 0     • y is not Outdotoctorean reading 0     • y is not outcompositioned on readvates sage 0     • y is not outcompositioned on readvates sage 0	24	.759(99.8%) 47 ms (0.2%)	594,759 ms 947 ms
	O Self time	0.01	00 ms (cm)	0.000 ms
• • • M	2) ∂ s0 time analysis/strete incoming-0.08.1.84.2.11	0.01	00 ms (0%) 707 (100%)	0.000 ms 595,707 ma
4.8	org, apache, cassandra net incomingTppConnection, run ()	595.	707 (100%)	595,707 ms
9	% gr g sock assessfor net locompt (g convertismented) to 1 = 1 = 1 = 1 = 1 = 1 = 1 = 1 = 1 = 1		707 (100%)	595,707 ms 594,767 ms
	🗢 📓 orp. apache. cassandra net. IncomingTopConnection. received Message 0	93	99 ms (0.2%)	939 ms
	L 0 saftma S aftma S aftma	0.01	00 ms (0%) 00 ms (0%)	0.000 ms
9 🚥 M	ssagingService-Incoming-/188.184.2.11	595.	707 (100%)	595,707 ms
1.8	or gapte assessed net literational (Contention III) and (Contention III)	595.	707 (100%) 707 (100%)	595,707 ms 595,707 ms
	🖕 😼 (so. 6. b.	595.	707 (100%)	595.707 ms
	* 2 m e point bi L2400/objetham med 0 + 2 m e point bi L2400/objetham med 10	595,	707 (100%)	595,707 ms 595,707 ms
	• ½ intiposita bit 248000000000000000000000000000000000000		707 (100%)	595,707 ms
	O Saff time	0.01	00 ms (0%)	0.000 ms
	□ © set one © set time	0.0	00 ms (0%)	0.000 ms
	(B) Self time	0.01	00 ms (0%) 00 ms (0%)	0.000 ms
	L 0 Saftme O Saftme	0.01	00 ms (0%)	0.000 ms
- 🚥 e	olfEventLoopGroup-2-12			595,707 ms
	saping/sector.execution_07.08.19.8.2.24 supporterior.execution_07.08.19.8.2.13		707 (100%)	595,707 ms
- 🚥 M	ssagingService-Incoming-/188.184.2.24	595.	707 (100%)	595,707 ms
~ == M	sapingstreet-concentry 0.03.18.2.10 sapingstreet-concentry 0.03.18.2.2.0	595.	707 (100%)	595,707 ms
- 📟 M	ssagingService-Incoming-/188184.2.26	595,	707 (100%)	595,707 ms
~ == M	sanjajosterie-tencening/10.11.19.12.26 sanjajosterie-tencing/17.13.19.14.14	595.	707 (100%)	595,707 ms 595,707 ms
🔶 📟 M	ssagingService-Incoming-J137.138.99.137	595,	707 (100%)	595,707 ms
~ == M	ssapingsforce/cncming-717.3109.144		707 (100%)	595.707 ms
- 📟 M	ssagingService-incoming-/108.104.2.25	595.	707 (100%)	595,707 ms
~ - M	saping/article.ton:enia/p18.184.25 saping/article.ton:enia/p18.184.25	595.	707 (100%)	595,707 ms 595,707 ms
~ 🚥 M	ssaainaService-Incomina-0.137.138.59.137	595.	707 (100%)	595,707 ms
M	sapingservice.execution;2030.124.225 mg.exitations:execution;2030.124.225 mg.exitations:execution;2030.124.235	595.	707 (100%)	595,707 ms 556,851 ms
- 🚥 C	mpactionExecutor18616	595.	707 (100%)	531,795 ms
	Niltrent software-2-16		707 (100%)	247,867 ms 247,857 ma
⊶ 📟 ej	allEventLoopGroup-2-4		707 (100%)	246,577 ms
ej	Hireat argings-24		707 (100%)	246,513 ms 245,846 ma
o- == ej	AllEventtsepGroup-2-2	595.	707 (100%)	245,830 ms
e	Jikeatagiong-27	595.	707 (100%)	245,676 ms 244,357 ms
- == ej	ollEventLoopGroup-2-10	595.	707 (100%)	242,758 ms
0- III ej	likentasjoner>25 likentasjoner>25	595.	707 (100%)	241,188 ms 240,991 ms
⊶ 🚥 ej	lifeed.op/opu-2-21 Uireed.op/opu-2-21		707 (100%)	
	ill rent softwar-3-10 ill rent softwar-3-3		707 (100%)	240,029 ms 239,654 ma
~ == e	MEvent.sopGroup-2:20		707 (100%)	238,630 ms
o- 💷 ej	Jikentagionp-54 Jikentagionp-52	595,	707 (100%)	238,180 ms 237,520 ms
- == ej	ollEventLoopGroup-2-31	595.	707 (100%)	236.719 ms
o- 💷 ej	likentagionp-2-77 likentagionp-2-74	595,	707 (100%)	236,404 ms 235,118 ms
- 🚥 ej	Jikenta 5000-20-21 Jikenta 5000-20-21	595.	707 [100%]	210.810 ms 209,376 ms
- 📟 ei	ollEventLoopGroup-2-27	595,	707 (100%)	208,844 ms
o- == ej	Jifeed approp26	595	707 (100%)	206.354 ms 205,130 ms
- 📟 ej	olfEventLoopGroup-2-23	595,	707 (100%)	204,664 ms
- 🚥 ej	ollEventLoopGroup-2-30	595.	707 (100%)	203,726 ms
0- == er	likentagionp-2-22 likentagionp-2-24	595,	707 (100%)	202,716 ms 201,917 ms
e- 🚥 ej	olftventloop6roup-2-25	595.	707 [100%]	201.174 ms
	Illeventagionup-2-29 rendenum variante de la construcción de l		707 (100%)	198,532 ms 48,220 ms
- = SI	aredPool Warker-145	595.	707 (1.00%)	47,473 ms
- = SI		595,	707 (100%) 707 (100%)	45,606 ms 45,546 ms
0- == SI	Markowski kale star star star star star star star star	595	707 (1.00%)	45,263 ms

 0.1
 25, 27 ma

 0.1

AliEn Catalogue - Miguel Martinez Pedreira

# B slides - CPU debug: jstack info

"(ML ThP) [ util.process ] Worker 250448, started: Wed Mar 29 16:57:08 CEST 2017" #1096365 daemon prio=5 os prio=0 tid=0x00007f58680c6800 nid=0x1547 waiting on condition [0x00007f57632f1000]

java.lang.Thread.State: TIMED\_WAITING (parking)

at sun.misc.Unsafe.park(Native Method)

- parking to wait for <0x0000006c0002870> (a java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObject)

at java.util.concurrent.locks.LockSupport.parkNanos(LockSupport.java:215)

at java.util.concurrent.locks.AbstractQueuedSynchronizer\$ConditionObject.awaitNanos(AbstractQueuedSynchronizer.java:2078)

at java.util.concurrent.ScheduledThreadPoolExecutor\$DelayedWorkQueue.poll(ScheduledThreadPoolExecutor.java:1129)

at java.util.concurrent.ScheduledThreadPoolExecutor\$DelayedWorkQueue.poll(ScheduledThreadPoolExecutor.java:809)

at java.util.concurrent.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java:1066)

at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1127)

at java.util.concurrent.ThreadPoolExecutor\$Worker.run(ThreadPoolExecutor.java:617)

at java.lang.Thread.run(Thread.java:745)