Using Spark to Generate Analytics for International Cable TV Video Distribution

Christopher Burdorf
NBC Universal

Problem Definition

- Gigabytes of metadata associated with international video distribution stored in Oracle and flat files needs to be processed to analyze usage
  - Store resulting data in Hbase/Hadoop – media Ids with usage over weeks, months, years, in total and on a per channel basis
  - Query Hbase tables, generate graphs and display them in a web page.
  - Generate report listing media files ordered by least recent usage – for offlining purposes
  - Generate histogram of days since last aired for media on storage cluster

Spark usage

- Queries from Oracle database get paths to schedule files stored on Isilon
- Schedule files are read and media ID's, air times, etc are stored in Scala classes and then to RDDs
- Spark map/reduceByKey is then used to produce counts for each media ID usage per country location based on week/month/year
- Spark join used to compute list of online media files (data from Oracle) and their last usage date (from schedule files). The result is sorted in ascending order on the most recent broadcast timestamp field.

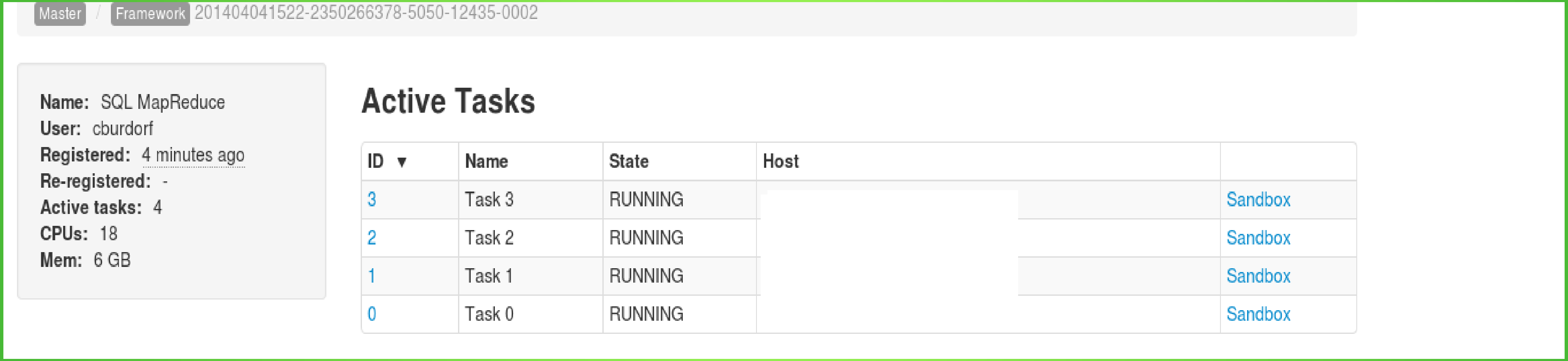
HBase/WebApp

- Broadcast frequency count data written from Spark App to CSV format then bulk-loaded into HBase
- Java/Spring web app queries HBase generates and displays graphs to web pages on demand.
- Spark App executed and HBase updated nightly

Production Environment

- Linux
- Oracle
- Clustered storage system
- Spark/Scala
- Java/Spring Web App

Mesos Cluster configuration



Master / Framework 201404041522-2350266378-5050-12435-0002

Name: SQL MapReduce
User: cburdorf
Registered: 4 minutes ago
Re-registered: -
Active tasks: 4
CPUs: 18
Mem: 6 GB

Active Tasks

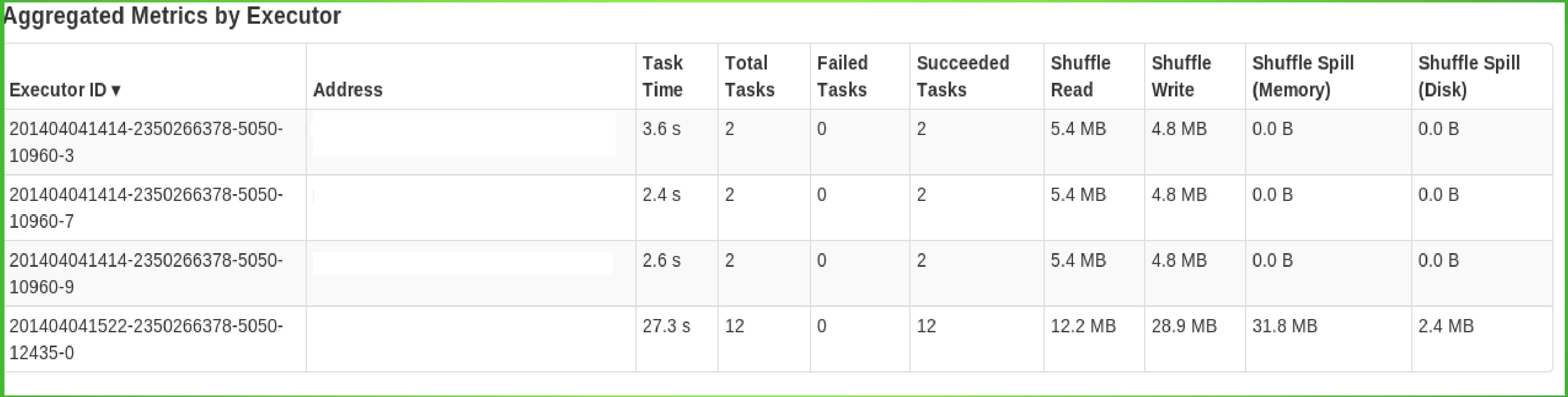
ID ▼	Name	State	Host	
3	Task 3	RUNNING	[redacted]	Sandbox
2	Task 2	RUNNING	[redacted]	Sandbox
1	Task 1	RUNNING	[redacted]	Sandbox
0	Task 0	RUNNING		Sandbox

- 18 cores, 23GB ram

Partitions from Spark app

```
Added rdd_0_3 in memory on server1:33229 (size: 42.9 KB, free: 883.2 MB)
 Added rdd_0_7 in memory on server1:33229 (size: 42.8 KB, free: 883.1 MB)
Added rdd_0_16 in memory on server2:45795 (size: 43.0 KB, free: 883.2 MB)
 Added rdd_0_15 in memory on server2:45795 (size: 42.9 KB, free: 883.1 MB)
 Added rdd_0_14 in memory on server2:45795 (size: 42.8 KB, free: 883.1 MB)
 Added rdd_0_2 in memory on server2:45795 (size: 43.0 KB, free: 883.0 MB)
Added rdd_0_13 in memory on server2:45795 (size: 43.2 KB, free: 883.0 MB)
Added rdd_0_10 in memory on server2:45795 (size: 42.9 KB, free: 882.9 MB)
 Added rdd_0_17 in memory on server2:45795 (size: 43.1 KB, free: 882.9 MB)
Added rdd_0_12 in memory on server2:45795 (size: 42.8 KB, free: 882.9 MB)
Added rdd_0_8 in memory on server2:45795 (size: 43.1 KB, free: 882.8 MB)
Added rdd_0_6 in memory on server2:45795 (size: 43.0 KB, free: 882.8 MB)
Added rdd_0_11 in memory on server2:45795 (size: 42.9 KB, free: 882.7 MB)
Added rdd_0_9 in memory on server2:45795 (size: 42.8 KB, free: 882.7 MB)
Added rdd_0_1 in memory on server4:43454 (size: 42.8 KB, free: 883.2 MB)
 Added rdd_0_5 in memory on server4:43454 (size: 42.9 KB, free: 883.1 MB)
Added rdd_0_0 in memory on server3:41154 (size: 43.0 KB, free: 883.2 MB)
Added rdd_0_4 in memory on server3:41154 (size: 42.9 KB, free: 883.1 MB)
```

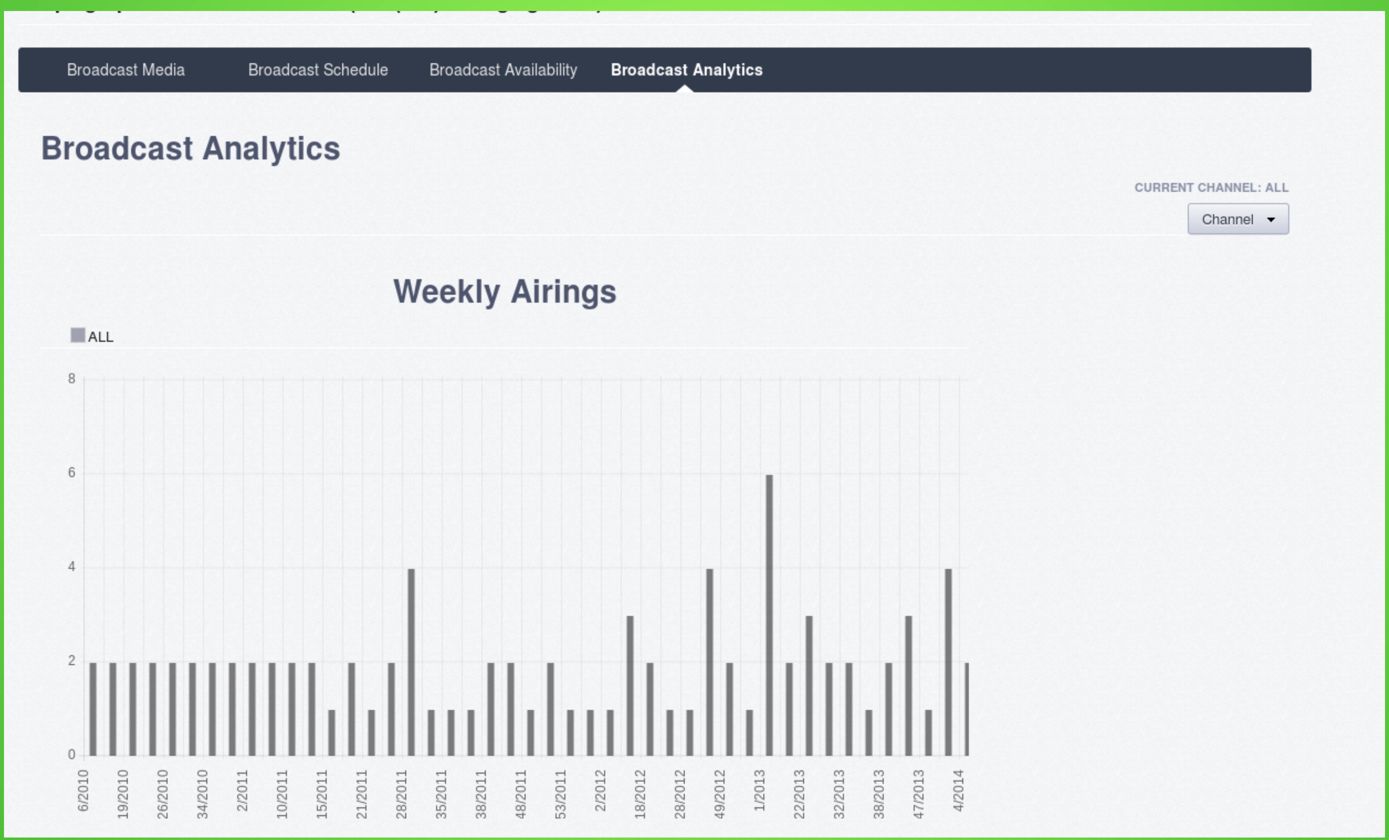
Shuffle Spill



Aggregated Metrics by Executor

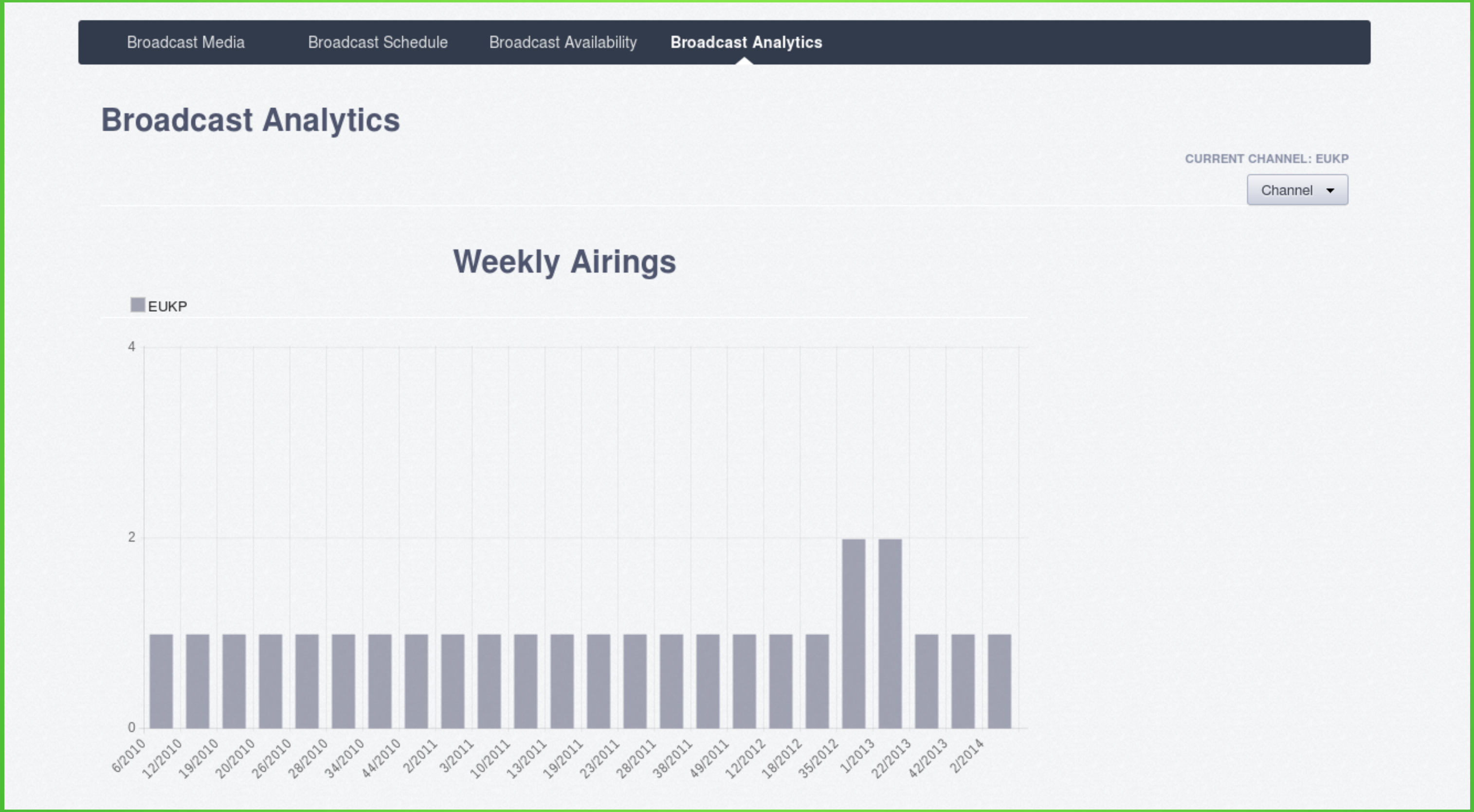
Executor ID ▾	Address	Task Time	Total Tasks	Failed Tasks	Succeeded Tasks	Shuffle Read	Shuffle Write	Shuffle Spill (Memory)	Shuffle Spill (Disk)
201404041414-2350266378-5050-10960-3		3.6 s	2	0	2	5.4 MB	4.8 MB	0.0 B	0.0 B
201404041414-2350266378-5050-10960-7		2.4 s	2	0	2	5.4 MB	4.8 MB	0.0 B	0.0 B
201404041414-2350266378-5050-10960-9		2.6 s	2	0	2	5.4 MB	4.8 MB	0.0 B	0.0 B
201404041522-2350266378-5050-12435-0		27.3 s	12	0	12	12.2 MB	28.9 MB	31.8 MB	2.4 MB

Results



Broadcast Media
Broadcast Schedule
Broadcast Availability
Broadcast Analytics
Broadcast Analytics
CURRENT CHANNEL: ALL
Channel
Weekly Airings
ALL
8
6
4
2
0
6/2010
19/2010
26/2010
34/2010
2/2011
10/2011
15/2011
21/2011
28/2011
35/2011
38/2011
48/2011
53/2011
2/2012
18/2012
28/2012
49/2012
1/2013
22/2013
32/2013
38/2013
47/2013
4/2014

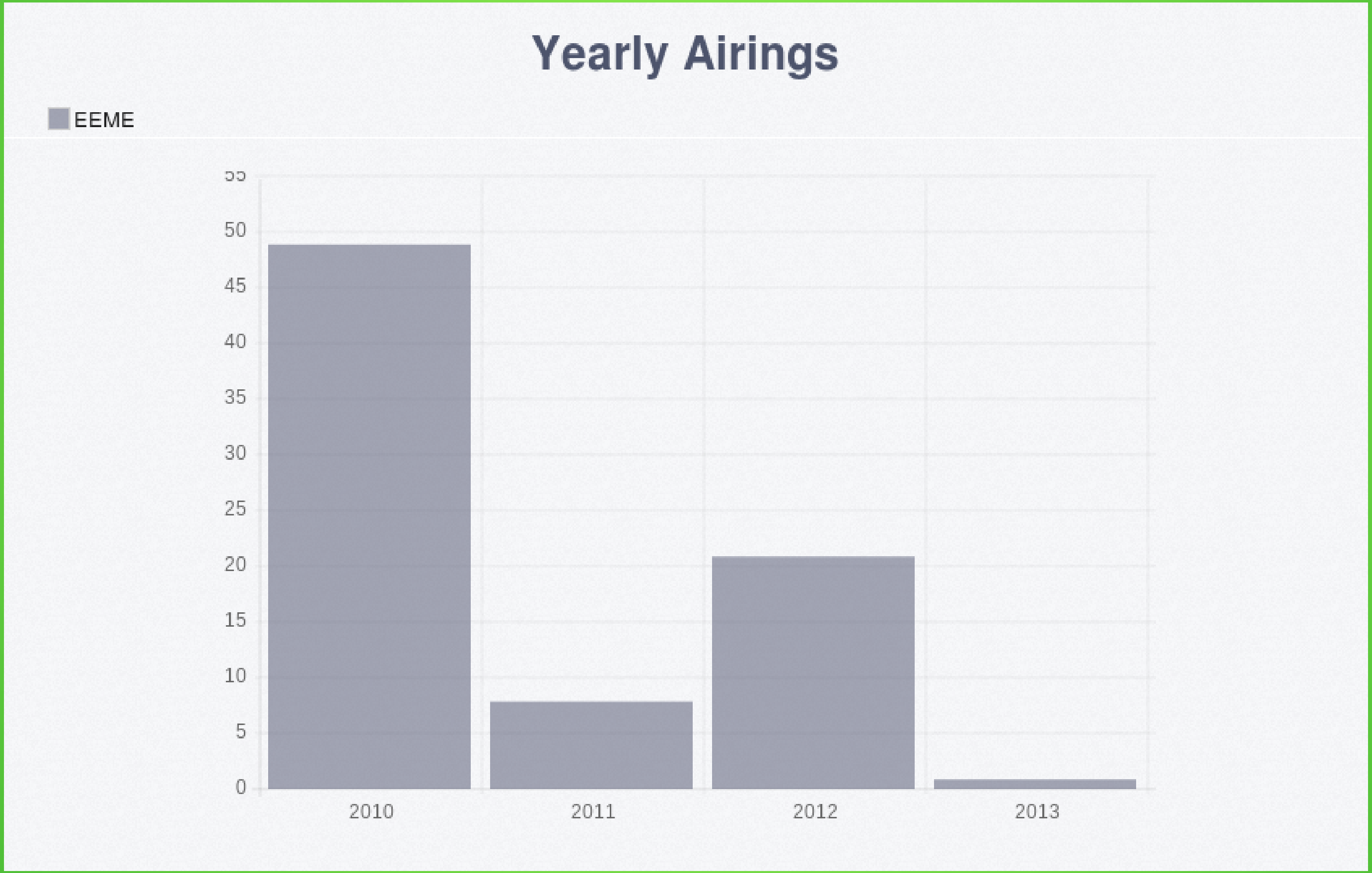
Per channel



Broadcast Analytics

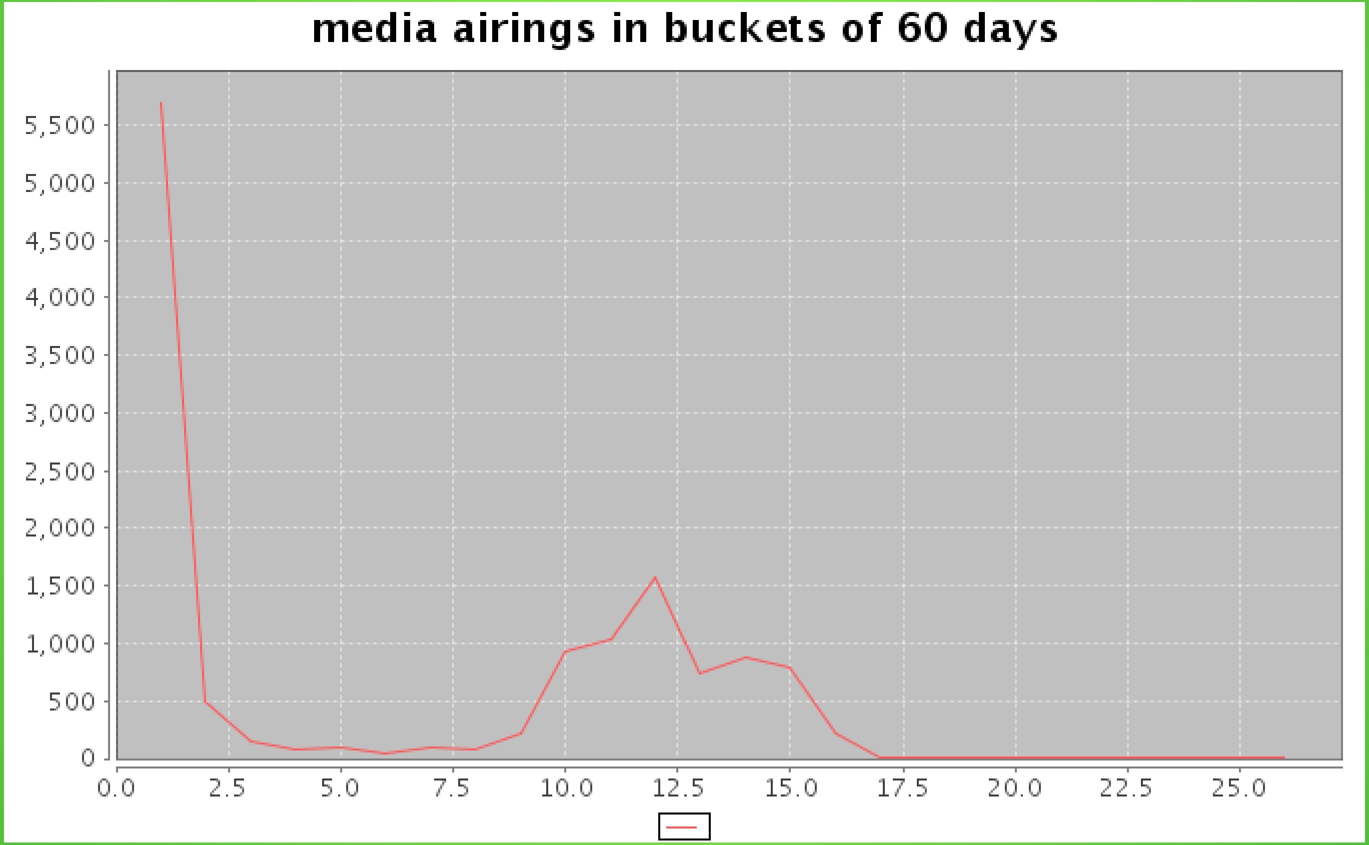
CURRENT CHANNEL: EUKP

Per channel



Yearly Airings
EEME
55
50
45
40
35
30
25
20
15
10
5
0
2010
2011
2012
2013

Storage cluster utilization
days since last aired per media file



Conclusions

- Performance improvements
  - Small data: 10x faster on 18 core vs 1 core even though a good portion of the app is sequential file and DB I/O.
  - Large data: not even possible on 1 box in cluster – runs out of memory.
- Spark/Scala super fun programming env – thanks to all developers
- Mesos cluster management wonderful ease of use
- Minor issue: Mesos Mac OSX build
- Future work