Riak to the Rescue
Migrating Big Data
Big Data.
Buzzwords.
Don’t believe the Hype.
Who am I?

Support
Development
SysAdmin
Managing Operations
Operations

8 Ops Engineers
4 Offices
Operations

650 physical
200 virtual
3 data centres
Contact

• Based in Berlin
• twitter: @geidies
• seb@meltwater.com
• http://underthehood.meltwater.com/
Migrating Big Data

- Meltwater
- Social Media Data Volumes
- Try and Fail
- Analyse and Succeed
- Things to Learn
Meltwater

News Monitoring
Meltwater News

Crawl the Web
Match new Articles
Morning Report
Analytics UI
Products

PR
m|news
m|press

Marketing
m|buzz / engage
icerocket
SaaS
Subscription model
24,000 clients
riak

- Open Source
- Dynamo Paper
- Erlang
2.0

OMG, OMG!!
thanks, basho.
Meltwater Buzz
Interesting Stuff
By Joan Doe - 2014/05/06

Something amazing happened yesterday. It was more interesting than what happened the day before, but maybe it won’t change the events that are about to come tomorrow.

What does Lorem ipsum dolor really mean? we know it is not real latin. But it looks pretty good, since the characters are evenly distributed. I once tried translating it, and it really doesn’t make any sense. Talking here is amazing. Wow, Denmark - it’s actually really cool being in Aarhus. You should have a chat with me after the talk if you have further questions. Please don’t hesitate to say hi. If you’re in Berlin, come stop by the meltwater office for a chat about big data, a cup of coffee, a game of table tennis of foosball. You can find us at Rotherstraße 22 in Friedrichshain.
Social Media

• 140 Characters
• Pages Long
Social Media

- Metadata
- Location
- Followers
- Threads
Social Media

- Extracted Metadata
  - sentiment
  - named entities
  - intent
- Editorial vs. Opinion vs. Both
m|buzz version 1

- Buzzgain
- php, MySQL, SolR
Attention!
Your Use Case

Research
Evaluate
Test
m|buzz version 2

Scalability,
Features,
Buzzwords!
“Some people, when confronted with a problem, think "I know, I'll use regular expressions." Now they have two problems.”

– Jamie Zawinski
Requirements

• Fail-Safety
• High Availability
• A Lot of Unstructured Data
• Near-Real-Time Indexing
• Time-Based Ordering instead of Relevancy
m|buzz version 2

- Hadoop Ecosystem
- Apache Projects
m|buzz version 2

Fetcher

HBase

API

M-R
 hourly
 daily

HDFS

Katta
It’s a trap!

- buzzwords
- commodity hardware
- scale
• Build upon lucene
• Master -> Worker -> Client
• communication through zookeeper
• multiple index copies
• copied from HDFS -> local disk
• OK in theory.
• Out Of Memory
• Garbage Collection Hell
• version 0.62 - odd bugs.
<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Commit</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.6.4</td>
<td>May 26, 2011</td>
<td>b930067</td>
</tr>
<tr>
<td>0.6.3</td>
<td>Nov 24, 2010</td>
<td>96dcd07</td>
</tr>
<tr>
<td>0.6.2</td>
<td>Aug 2, 2010</td>
<td>46c2b5e</td>
</tr>
<tr>
<td>0.6.1</td>
<td>Feb 28, 2010</td>
<td>d95891c</td>
</tr>
<tr>
<td>0.6.0</td>
<td>Feb 4, 2010</td>
<td>9953b47</td>
</tr>
<tr>
<td>0.6.rc1</td>
<td>Jan 12, 2010</td>
<td>606039a</td>
</tr>
</tbody>
</table>
keyspace

region

region

key a -> key c

key n -> key o

split
Fail-Safety
Fail-Safety

Does NOT mean High Availability

Data on a Single Node
Minutes.

55,000 posts / minute
Funny Regions
Overlapping
Gaps
Negative Length
Funny Regions

REGION => {NAME => 'buzz_data, 1333073443000_62gfsHBsE5vNSz168ByvP5tDPu0A, 1333173530871',
STARTKEY => '1333073443000_62gfsHBsE5vNSz168ByvP5tDPu0A',
ENDKEY => '1326306499000_evKK670FSV9MAas2CMZAr41wLm0A', ENCODED => 128988498, TABLE => {{NAME => 'buzz_data', FAMILIES => [{NAME => 'fm_contents',VERSIONS => '1', COMPRESSION => 'LZO', TTL => '2147483647', BLOCKSIZE => '65536', IN_MEMORY => 'false', BLOCKCACHE => 'true'}, {NAME => 'fm_input_info', VERSIONS => '1', COMPRESSION => 'LZO', TTL => '2147483647', BLOCKSIZE => '65536', IN_MEMORY => 'false', BLOCKCACHE => 'true'}, {NAME => 'fm_metadata', VERSIONS => '1', COMPRESSION => 'LZO', TTL => '2147483647', BLOCKSIZE => '65536', IN_MEMORY => 'false', BLOCKCACHE => 'true'}, {NAME => 'fm_output_info', VERSIONS => '1', COMPRESSION => 'LZO', TTL => '2147483647', BLOCKSIZE => '65536', IN_MEMORY => 'false', BLOCKCACHE => 'true'}]}}}
HBase

• .META. corruption
• Data Unavailability
• Slow Start of Regions
• Full Cluster Restarts Slow
• Hotspots
Good News!

NameNode never crashed.
Great.
Changes...

...do you speak it?
m|buzz version 2.5

MapR Distribution
Your logic makes the puppy sad.
• Message Queue System

• Erlang

• Redundant Setup, fail-safe and high-available

• Write to Exchange -> Distribute to Multiple Queues
First Read Wins

Parallel Reads:
- couchbase
- vanilla HBase
- MapR HBase
couchbase scales!

...to four weeks of data.
2.2B entries
TTL
Are we there yet?
## Options

<table>
<thead>
<tr>
<th></th>
<th>Pro</th>
<th>Con</th>
</tr>
</thead>
<tbody>
<tr>
<td>custom WAL</td>
<td>works safely</td>
<td>doesn't scale (easily)</td>
</tr>
<tr>
<td>MySQL cluster</td>
<td>A lot of experience</td>
<td>hitting limit of scaling</td>
</tr>
<tr>
<td>commercial Object storage</td>
<td>commercial support</td>
<td>up-front investment</td>
</tr>
<tr>
<td>riak</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Requirements

✓ High Availability

✓ Data Safety

✓ Scalability

? Range Scans or TTL to limit data
riak

Key-Value model
Objects in Buckets
“While there are mechanisms such as Vector Clocks to help deal with these issues, if your application requires the kind of strong consistency found in ACID systems, Riak may not be a good fit.”

– riak documentation
m|buzz version 2.6
Commodity Hardware

- HP DL360 G1
- 4c CPU
- 32GB RAM
- 1x 2TB 7.2k spinner
- …37 of those.
Configuration

- levelDB
- erlang VM
- Map-Reduce
Future-Proof

Setting the ring-size to...

2048.
“2048 is definitely the upper bound of what we recommend, but with the right amount of machines, this can work.”

– riak mailing list
“Are you guys insane? We didn’t even know that was possible!!”

— riak mailing list re-niced
Numbers

- 37 nodes
- 55,000 writes per minute
- 350,000 reads per minute
- 1.8TB data per node
Hey, wait.

A good three weeks?
Let's do it.

parallel reads
gather numbers
stability
speed
riak is slow.

but consistent,
and massively parallel.
riak is slow.

riak is not as fast as a memory-only key-value store.
stability over speed.
stability

- availability during
  - node failures
  - upgrades
  - configuration updates
Search
m|buzz version 3
Naming Things
m|buzz version 3

API

ES/R

fetcher

fetcher

fetcher

fetcher

fetcher

fetcher

fetcher

enrichment

couchbase

elasticsearch

riak

ES/R
Putting it live

10:20 AM

Adarsh T. checked out few accounts looks good

Stian G. performing very well indeed

Adarsh T. true :-)

10:25 AM

Sebastian G. Ok. So here's the truth: for the past hour, we've been running of ES/R instead of the 2.2 API.

There is some side effects. Example given: I've got posts in my campaign that are only 2 minutes old.

:)

Matthias R. Tststs... sneaky bastard ;-)

Sebastian G. I wanted people to test unbiased. :)


Still live

- 58,000,000,000 key-value pairs written
- 365,000,000,000 reads
- 3.5ms mean (8ms 95th, 35ms 99th, 2s 100)
Monitoring

- Input “valves”
- Throughput of any intermediate processing step
- Output valves
- Distribution of data across cluster
- Handovers of data within the cluster
Dashboards

And APIs.
### localhost Hadoop Map/Reduce Administration

**State:** INITIALIZING  
**Started:** Fri Jun 25 14:34:57 PDT 2010  
**Version:** 0.20.2-dev. r  
**Compiled:** Fri Oct 23 21:54:18 PDT 2009 by jherr  
**Identifier:** 201006251434  

**Cluster Summary (Heap Size is 81.06 MB/995.88 MB)**

<table>
<thead>
<tr>
<th>Maps</th>
<th>Reduces</th>
<th>Total Submissions</th>
<th>Nodes</th>
<th>Map Task Capacity</th>
<th>Reduce Task Capacity</th>
<th>Avg. Tasks/Node</th>
<th>Blacklisted Nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Scheduling Information**

<table>
<thead>
<tr>
<th>Queue Name</th>
<th>Scheduling Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**Filter (JobId, Priority, User, Name)**

Example: `user:username` will filter by `username` only in the user field and `*username` in all fields

**Running Jobs**

none

**Completed Jobs**
Master: ip-10-110-247-97.ec2.internal:60000

Local logs, Thread Dump, Log Level, Debug dump

Attributes

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HBase Version</td>
<td>0.92.0, r1234893</td>
<td>HBase version and revision</td>
</tr>
<tr>
<td>HBase Compiled</td>
<td>Sun Feb 19 02:45:09 PST 2012, gkesavan</td>
<td>When HBase was compiled</td>
</tr>
<tr>
<td>Hadoop Version</td>
<td>1.0.1, r1243785</td>
<td>Hadoop version and revision</td>
</tr>
<tr>
<td>Hadoop Compiled</td>
<td>Tue Feb 14 08:12:14 UTC 2012, hortonfo</td>
<td>When Hadoop was compiled</td>
</tr>
<tr>
<td>HBase Root Directory</td>
<td>hdfs://ip-10-140-6-87.ec2.internal:8020/apps/hbase/data</td>
<td>Location of HBase home directory</td>
</tr>
<tr>
<td>HBase Cluster ID</td>
<td>b3d30236-4f60-4a39-ac40-76d5ff16b455</td>
<td>Unique identifier for each node.</td>
</tr>
<tr>
<td>Load average</td>
<td>⬤</td>
<td>Average number of regions per reg.</td>
</tr>
<tr>
<td>Zookeeper Quorum</td>
<td>domU-12-31-39-0A-06-14.compute-1.internal:2181</td>
<td>Addresses of all registered ZK servers</td>
</tr>
<tr>
<td>Coprocessors</td>
<td>[]</td>
<td>Coprocessors currently loaded (none)</td>
</tr>
<tr>
<td>HMaster Start Time</td>
<td>Tue Feb 21 19:18:24 EST 2012</td>
<td>Date stamp when 1st HMaster started</td>
</tr>
<tr>
<td>HMaster Active Time</td>
<td>Tue Feb 21 19:18:24 EST 2012</td>
<td>Date stamp when the 1st HMaster active.</td>
</tr>
</tbody>
</table>

Tasks

Show All Monitored Tasks Show non-RPC Tasks Show All RPC Handler Tasks Show Active RPC Calls Show Client Operations View as JSON

<table>
<thead>
<tr>
<th>Start Time</th>
<th>Description</th>
</tr>
</thead>
</table>

Tables

<table>
<thead>
<tr>
<th>Catalog Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.ROOT:</td>
<td>The .ROOT table holds references to all .META regions.</td>
</tr>
<tr>
<td>.META:</td>
<td>The .META table holds references to all User Table regions.</td>
</tr>
</tbody>
</table>

1 table(s) in set. [Details]

<table>
<thead>
<tr>
<th>User Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>usertable</td>
<td>(NAME =&gt; 'usertable', FAMILIES =&gt; [{NAME =&gt; 'family', MIN_VERSIONS =&gt; '0'}])</td>
</tr>
<tr>
<td>#</td>
<td>Owner Node</td>
</tr>
<tr>
<td>----</td>
<td>--------------------</td>
</tr>
<tr>
<td>0</td>
<td><a href="mailto:riak@riak3.ack">riak@riak3.ack</a></td>
</tr>
<tr>
<td>1</td>
<td><a href="mailto:riak@riak1.ack">riak@riak1.ack</a></td>
</tr>
<tr>
<td>2</td>
<td><a href="mailto:riak@riak2.ack">riak@riak2.ack</a></td>
</tr>
<tr>
<td>3</td>
<td><a href="mailto:riak@riak4.ack">riak@riak4.ack</a></td>
</tr>
<tr>
<td>4</td>
<td><a href="mailto:riak@riak3.ack">riak@riak3.ack</a></td>
</tr>
<tr>
<td>5</td>
<td><a href="mailto:riak@riak1.ack">riak@riak1.ack</a></td>
</tr>
<tr>
<td>6</td>
<td><a href="mailto:riak@riak2.ack">riak@riak2.ack</a></td>
</tr>
<tr>
<td>7</td>
<td><a href="mailto:riak@riak4.ack">riak@riak4.ack</a></td>
</tr>
<tr>
<td>8</td>
<td><a href="mailto:riak@riak3.ack">riak@riak3.ack</a></td>
</tr>
<tr>
<td>9</td>
<td><a href="mailto:riak@riak1.ack">riak@riak1.ack</a></td>
</tr>
<tr>
<td>10</td>
<td><a href="mailto:riak@riak2.ack">riak@riak2.ack</a></td>
</tr>
</tbody>
</table>
necessary but not sufficient

dashboard
API
fool-safe performance configuration
good documentation
Buzzwords

Be amazed.
Doubt.
Evaluate.
Hardware

There is no such thing as “too much RAM”
Scale
You’ll need it.
Configuration Management

who’s the master of puppet?
Monitoring looks exciting even when things work.
Operational Stability beats Features when it comes to Big A Lot of Data.
Thank you.

@geidies - seb@meltwater.com

http://underthehood.meltwater.com/

slides w/ notes on github.com/geidies/slides