

# Decomposing Twitter

QConNY 2013



**Jeremy Cloud**

@jeremycloud

Follow

[warn] one warning found

[error] 185 errors found

Yay, only 1 warning!

5:15 PM - 12 Apr 2013

3 FAVORITES



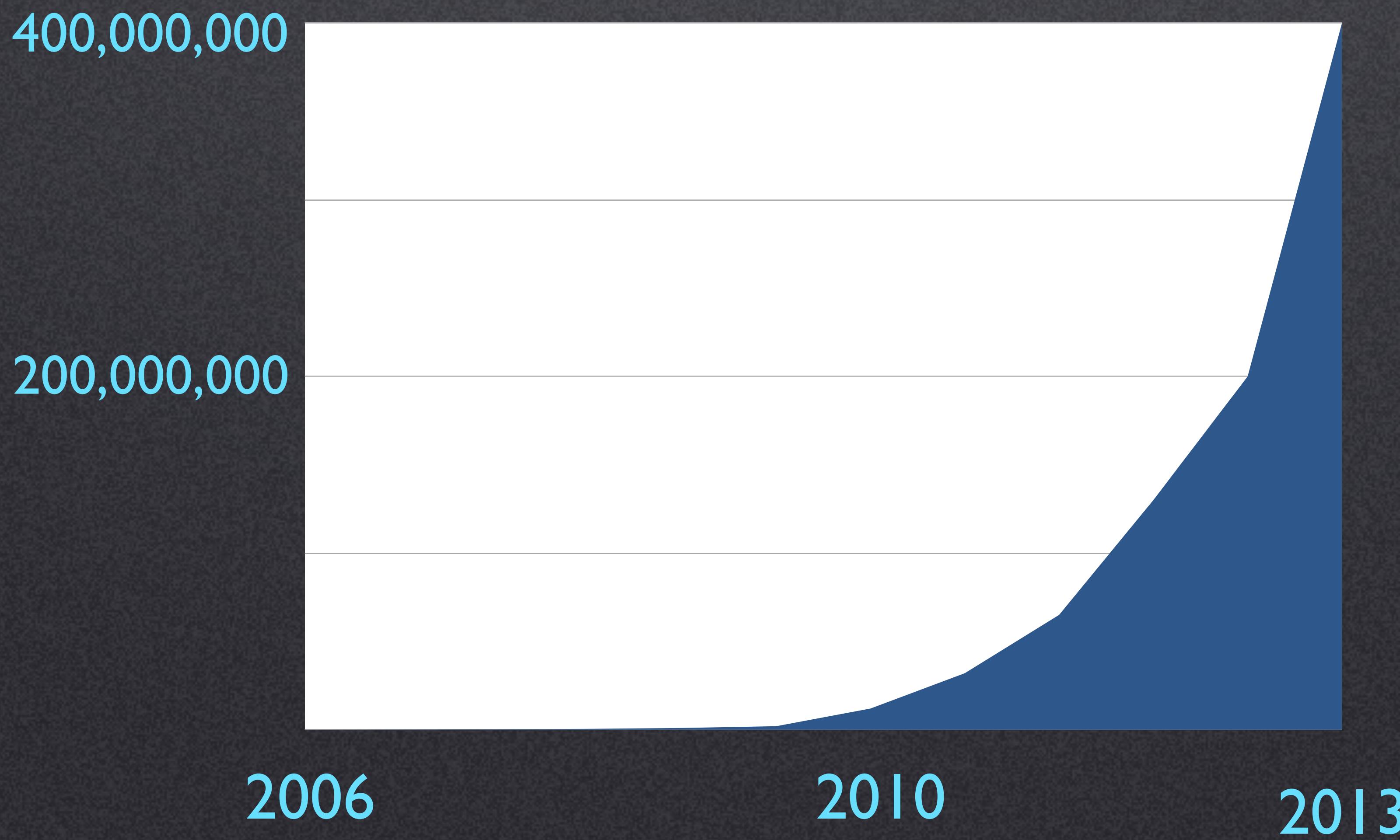
Tech Lead of Tweet Service Team



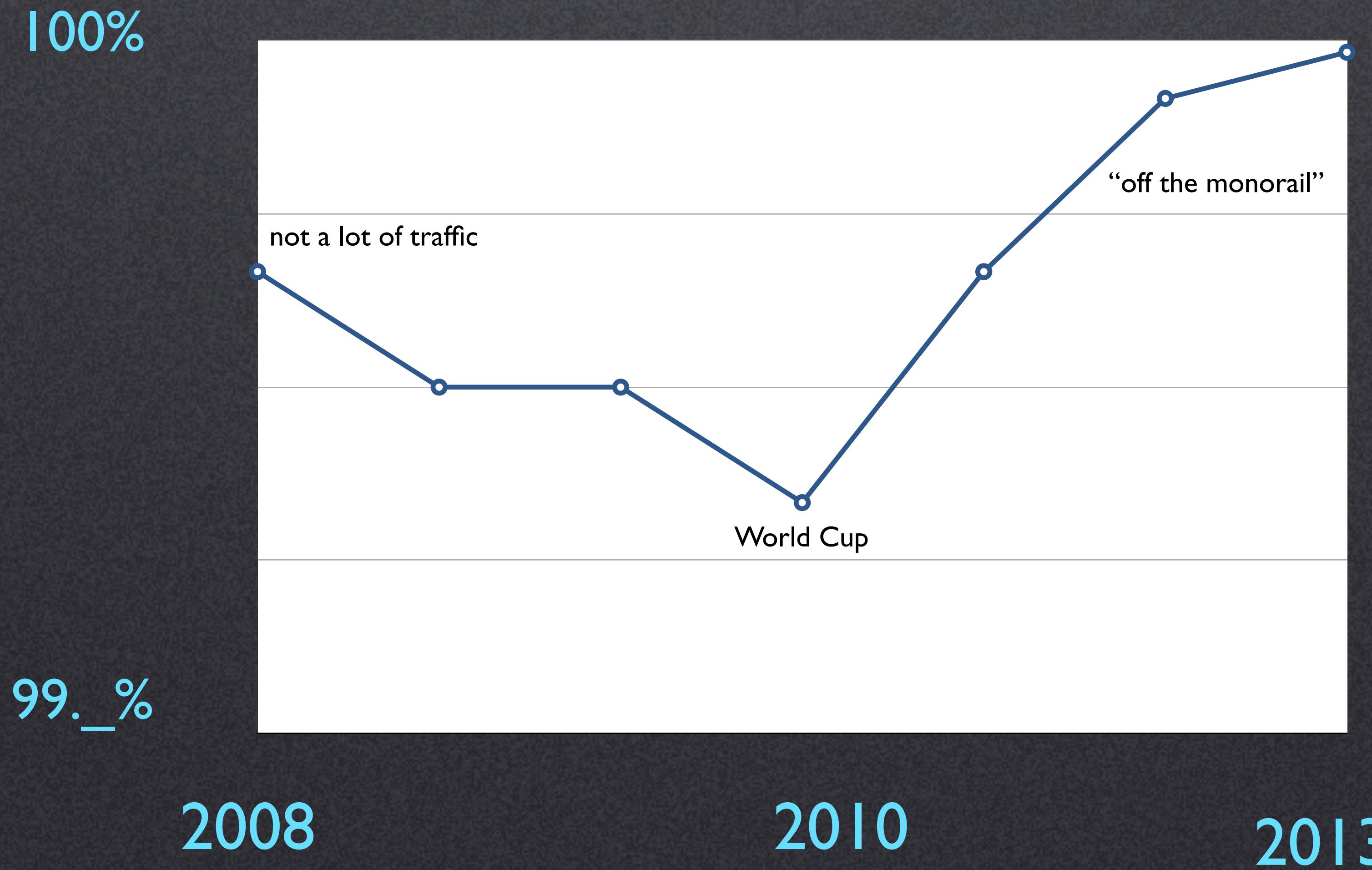
# Backstory

- ....> Exponential growth in traffic
- ....> Cost of failure has gone up
- ....> Success rate constantly improving
- ....> Company has grown 10x over the last 3yrs

# Tweets Per Day



# Site Success Rate



Routing

Presentation

Logic

Storage

Monorail



MySQL

# Challenges

- …› storage I/O bottlenecks
- …› poor concurrency, runtime performance
- …› brittle
- …› too many cooks in the same kitchen
- …› lack of clear ownership
- …› leaky abstractions / tight-coupling

# Goals of SOA

- …› not just a web-stack
- …› isolate responsibilities and concerns
- …› site speed
- …› reliability, isolate failures
- …› developer productivity

# API Boundaries

 **Dalai Lama quotes**  
@HisHoliness

 Follow

We must all live harmoniously with our neighbors.

9:56 AM - 24 Apr 2013

---

87 RETWEETS 35 FAVORITES





Routing

Presentation

Logic

Storage

Monorail



We have the technology.

ICANHASCHEEZBURGER.COM

MySQL

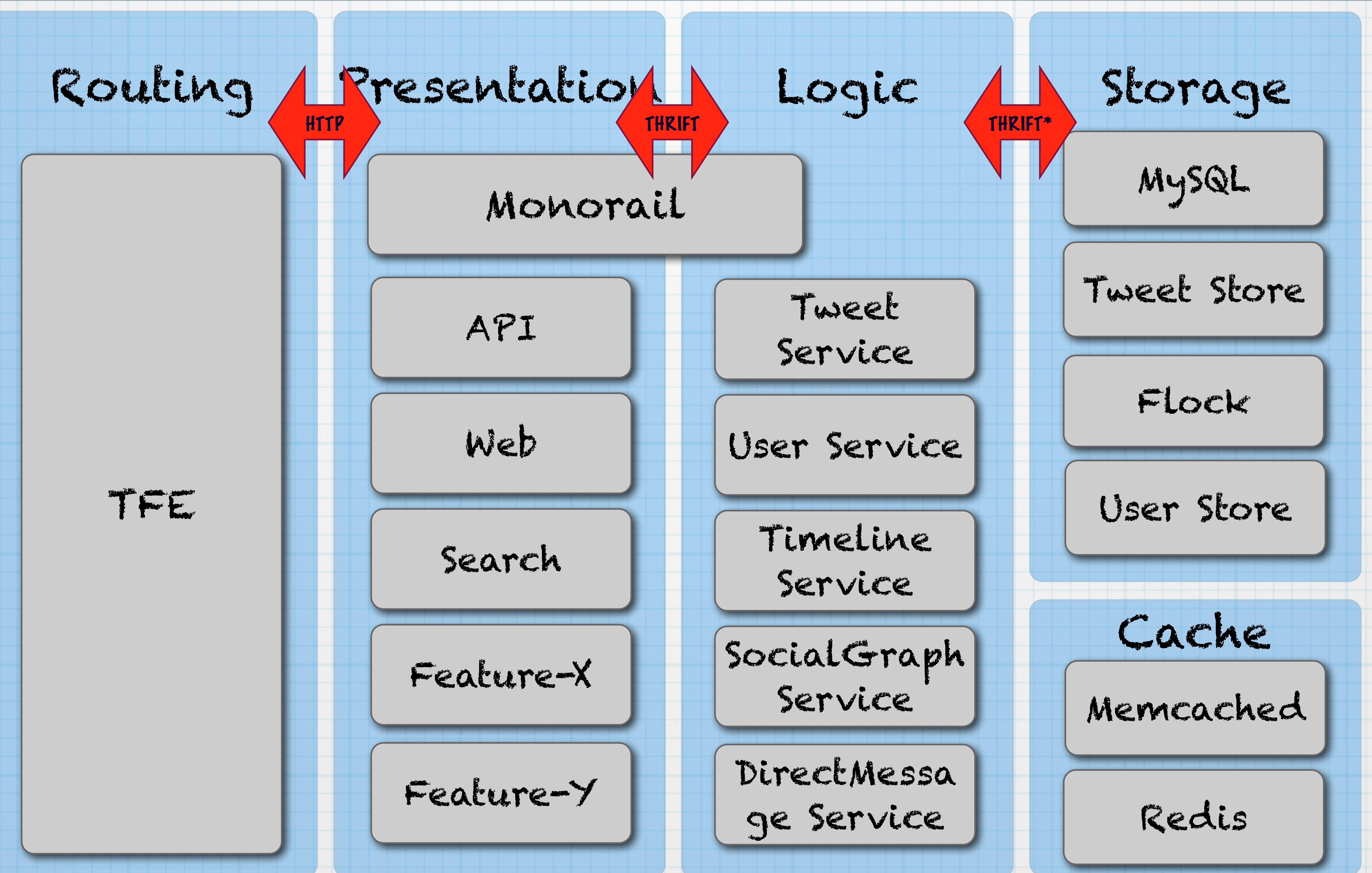
Tweet

Flock

Cache

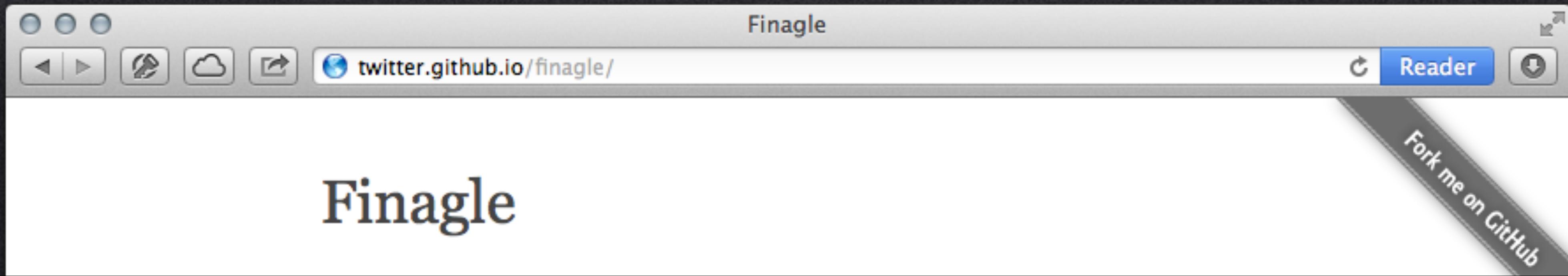
Memcache

Redis



# Finagle

- …› service discovery
- …› load balancing
- …› retrying
- …› thread/connection pooling
- …› stats collection
- …› distributed tracing



# Future[T]

- …⇒ async, non-blocking IO
- …⇒ fixed number of threads
- …⇒ very high levels of concurrency

# Tweet Service

## Writes

- ....> daily: > 400 MM tweets
- ....> avg: > 5000 TPS
- ....> daily peak: 9000 TPS
- ....> max peak: 33,388 TPS

# Tweet Service

## Reads

- …› daily: 350 Billion tweets
- …› avg: 4 MM TPS
- …› median latency: 7 ms
- …› p99 latency: 120 ms (bulk requests)
- …› server count: low hundreds
- …› cache: ~ 15 TB

user metadata



Aliina Burton Cloud  
@baby\_cloud

Follow

Hanging out at Red Rocks Amphitheater with  
@jeremycloud pic.twitter.com/YEfUgPjMBg

1:03 PM - 15 May 2013 from Colorado, US, United States

1 RETWEET 5 FAVORITES

mentioned-user metadata



counts



url metadata



geo metadata



media metadata  
card metadata



mentions

media

cards

reply/

rt/rp/fv

pctd

conversati

```
1 {  
2   "id" : 334730787213701120,  
3   "userId" : 524748670,  
4   "text" : "Hanging out at Red Rocks Amphitheater with @jeremycloud http://t.co/YEfUgPjMBg",  
5   "createdVia" : "oauth:1021743",  
6   "createdAt" : 1568641913,  
7   "urls" : [],  
8   "mentions" : [ {  
9     "from" : 43,  
10    "to" : 55,  
11    "screenName" : "jeremycloud",  
12    "userId" : 15062340,  
13    "name" : "Jeremy & Cloud"  
14  } ],  
15   "hashtags" : [],  
16   "cashtags" : [],  
17   "media" : [ {  
18     "from" : 56,  
19     "to" : 78,  
20     "url" : "http://t.co/YEfUgPjMBg",  
21     "displayUrl" : "pic.twitter.com/YEfUgPjMBg",  
22     "mediaUrl" : "http://pbs.twimg.com/media/BKUz4KOCCAA0Tz9.jpg",  
23     "mediaUrlHttps" : "https://pbs.twimg.com/media/BKUz4KOCCAA0Tz9.jpg",  
24     "expandedUrl" : "http://twitter.com/baby_cloud/status/334730787213701120/photo/1",  
25     "mediaId" : 334730787217895424,  
26     "nsfw" : 0,  
27     "sizes" : [ {  
28       "sizeType" : 0,  
29       "resizeMethod" : 0,  
30       "contentType" : 1,  
31       "width" : 1136,  
32       "height" : 852  
33     }, {  
34       "sizeType" : 4,  
35       "resizeMethod" : 1,  
36       "contentType" : 1,  
37       "width" : 150,  
38       "height" : 150  
39     }, {  
40       "sizeType" : 2,  
41       "resizeMethod" : 0,  
42       "contentType" : 1,  
43       "width" : 600,  
44       "height" : 450  
45     }, {  
46       "sizeType" : 1,  
47       "resizeMethod" : 0,  
48       "contentType" : 1,  
49       "width" : 1024,  
50       "height" : 768  
51     }, {  
52       "sizeType" : 3,  
53       "resizeMethod" : 0,  
54       "contentType" : 1,  
55       "width" : 340,  
56       "height" : 255  
57     },  
58     "mediaPath" : "BKUz4KOCCAA0Tz9.jpg",  
59     "isProtected" : 0  
60   },  
61   "hasTakedown" : 0,  
62   "counts" : {  
63     "retweetCount" : 0,  
64     "replyCount" : 0,  
65     "favoriteCount" : 0  
66   },  
67   "deviceSource" : {  
68     "id" : 1031313,  
69     "parameter" : "oauth:1021743",  
70     "internalName" : "oauth:1021743",  
71     "name" : "Camera on iOS",  
72     "url" : "http://www.apple.com",  
73     "display" : "<a href=\"http://www.apple.com\">Camera on iOS</a>",  
74     "clientAppId" : 1021743  
75   },  
76   "perspective" : {  
77     "userId" : 15062340,  
78     "favorited" : 0,  
79     "retweeted" : 0  
80   },  
81   "cards" : [ {  
82     "url" : "http://t.co/YEfUgPjMBg",  
83     "cardType" : 1,  
84     "title" : "",  
85     "description" : "",  
86     "images" : [ [ {  
87       "mediaType" : "mobile",  
88       "width" : 340,  
89       "height" : 255,  
90       "url" : "https://pbs.twimg.com/media/BKUz4KOCCAA0Tz9.jpg:small",  
91       "url2x" : "https://pbs.twimg.com/media/BKUz4KOCCAA0Tz9.jpg:large"  
92     }, {  
93       "mediaType" : "web",  
94       "width" : 340,  
95       "height" : 255,  
96       "url" : "https://pbs.twimg.com/media/BKUz4KOCCAA0Tz9.jpg:small",  
97       "url2x" : "https://pbs.twimg.com/media/BKUz4KOCCAA0Tz9.jpg:large"  
98     }, {  
99       "mediaType" : "tablet",  
100      "width" : 600,  
101      "height" : 450,  
102      "url" : "https://pbs.twimg.com/media/BKUz4KOCCAA0Tz9.jpg:medium",  
103      "url2x" : "https://pbs.twimg.com/media/BKUz4KOCCAA0Tz9.jpg:large"  
104    } ]  
105  },  
106  "nullcast" : 0,  
107  "conversationId" : 334730787213701120,  
108  "language" : {  
109    "language" : "en",  
110    "rightToLeft" : 0  
111  }
```

urls

perspectivals

client-app

geo

spam

language

Tweet  
Reads/  
Writes

Tweet  
Service

## Upstream Services

Url Service

Media

Cards

User Service

Spam

Language  
Service

Geo

Timeline  
Service

Cache

Memcached

Storage

Tweets

TFlock

MySQL

Write-Only

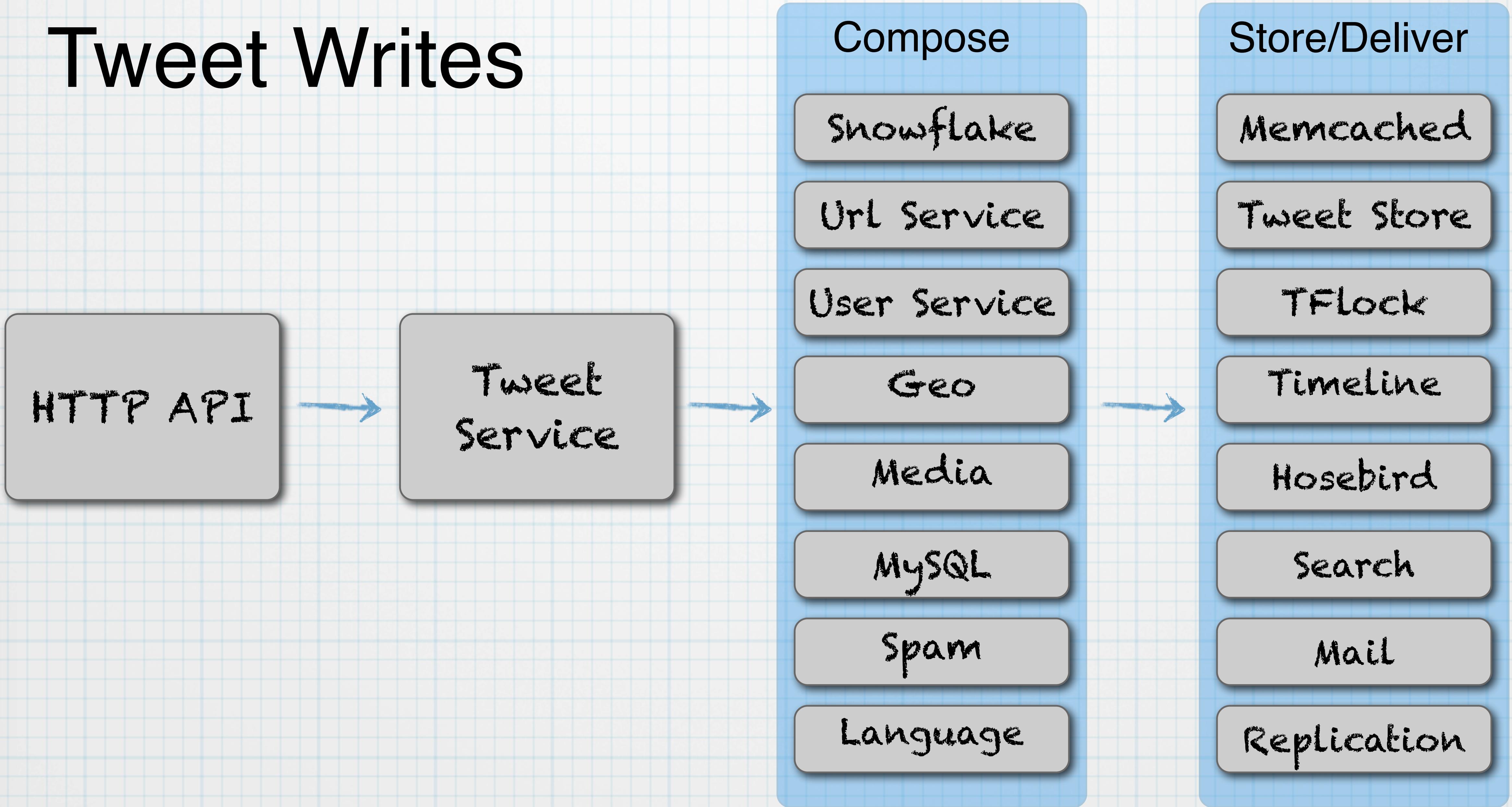
Snowflake

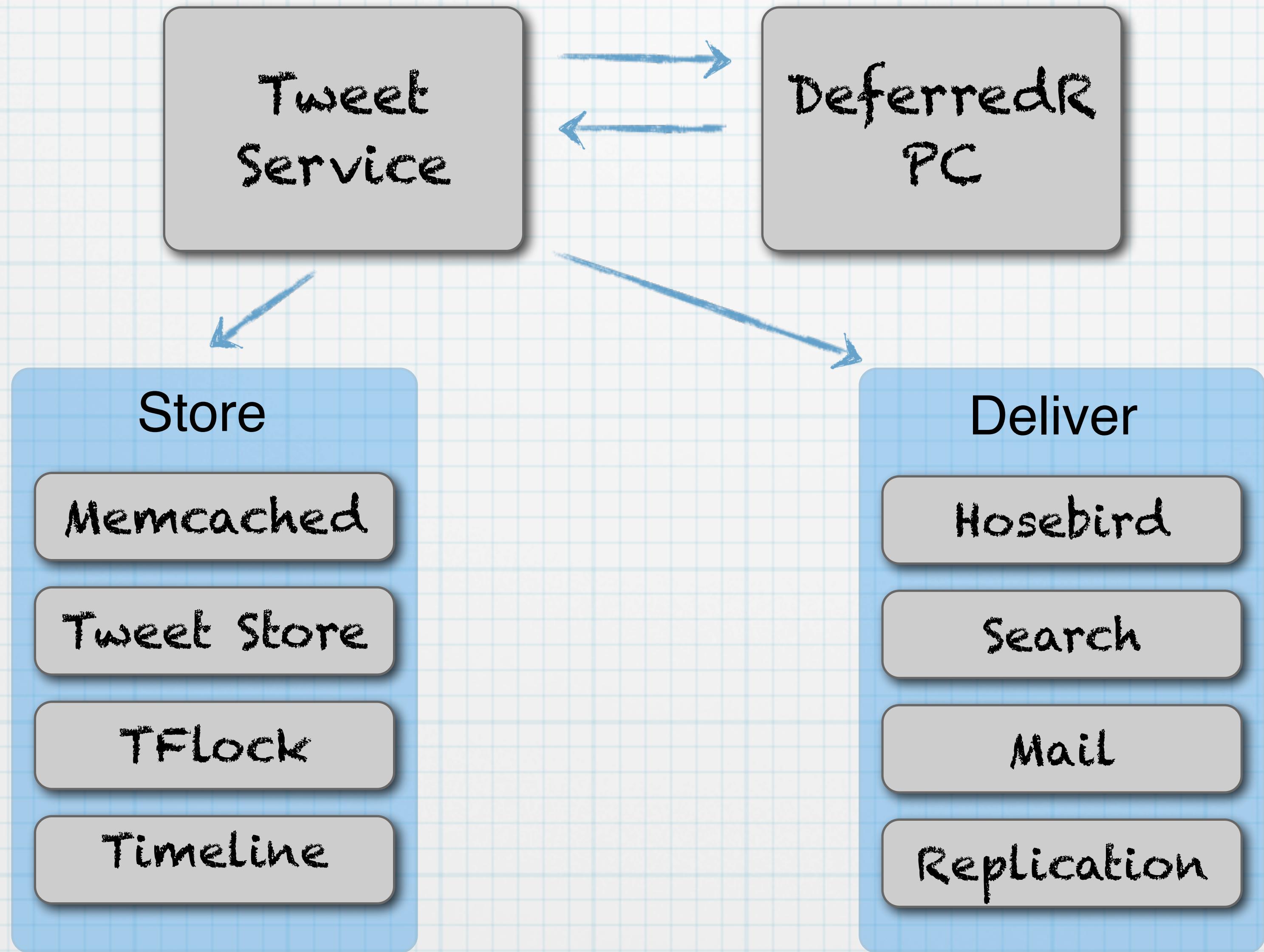
Search

Mail

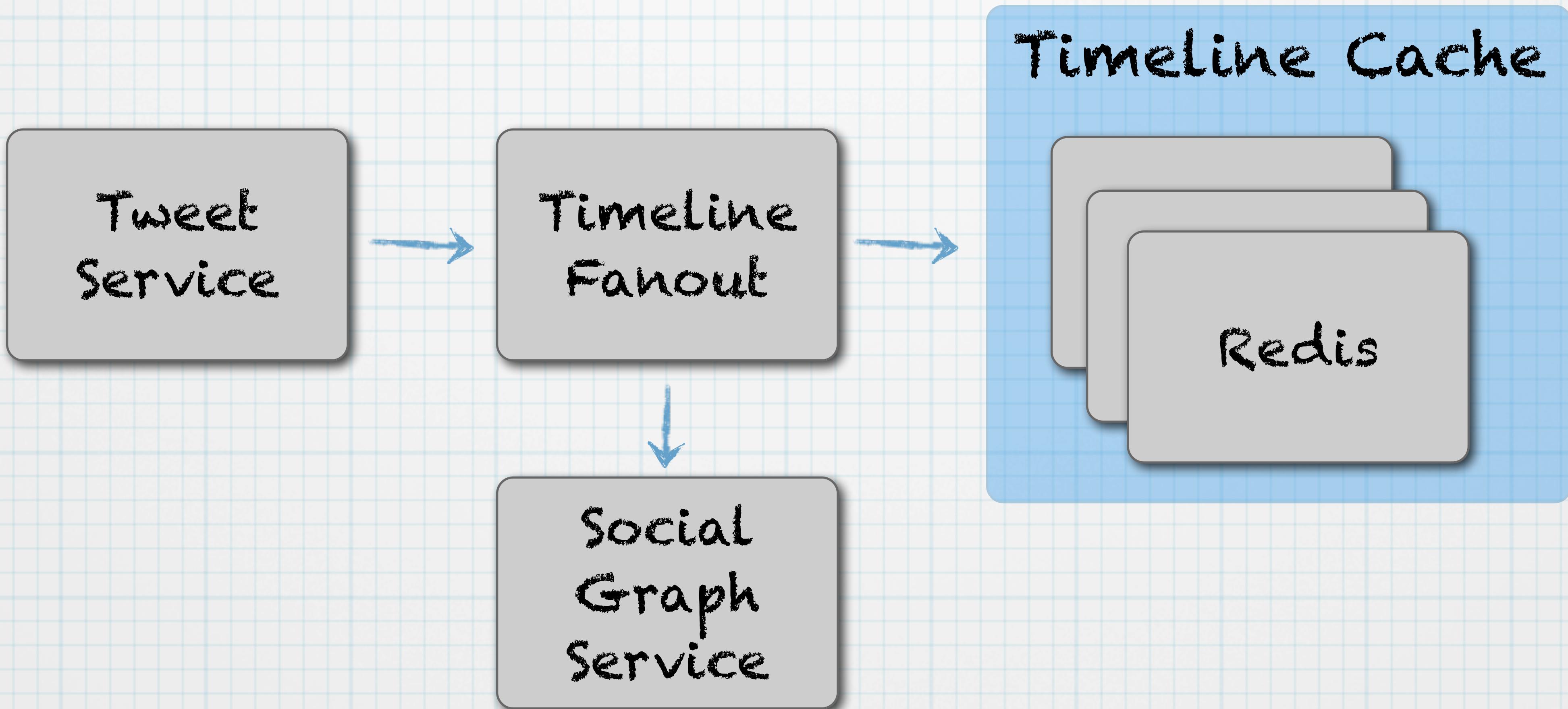
Firehose

# Tweet Writes

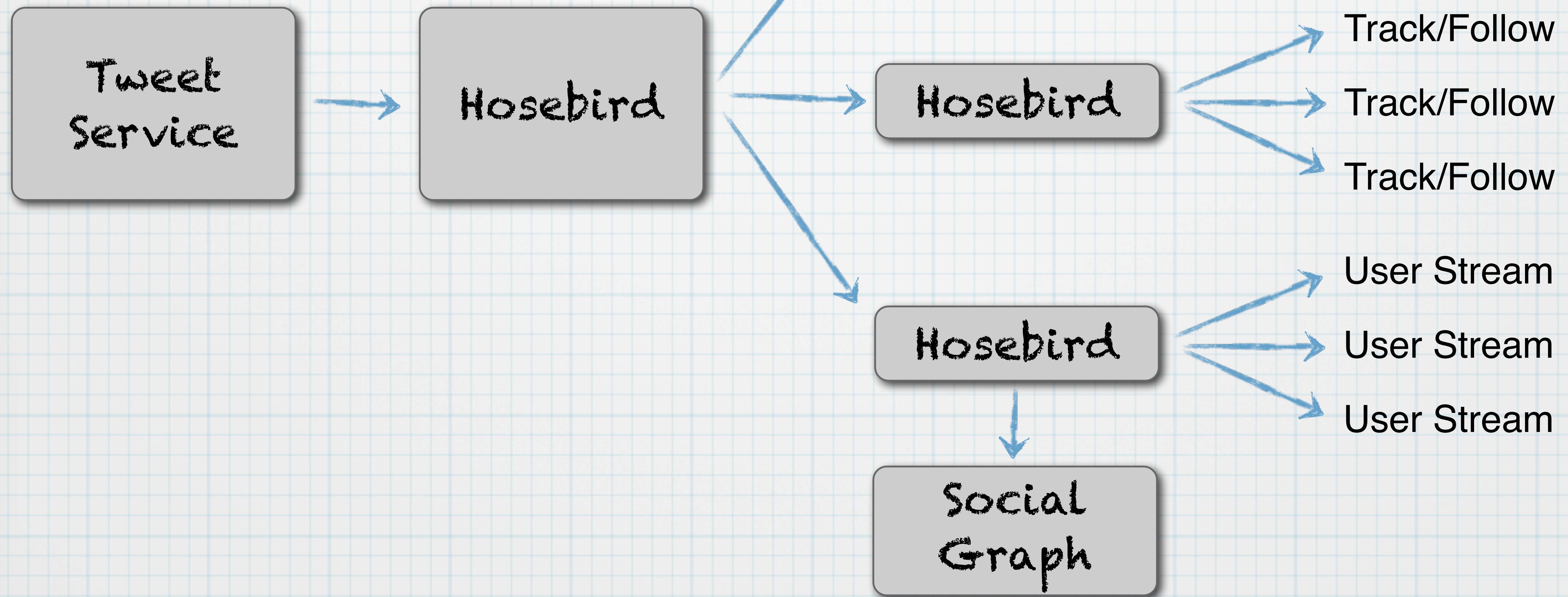




# Fanout



# Streaming



# Top 3 Lessons Learned

# Lessons Learned #1

Incremental change eventually wins

Step 1: Make the smallest possible change

Step 2: Verify/tweak

Step 3: GOTO Step 1

# Deploy Often and Incrementally

- …► Keep the changes in each deploy small
- …► Validate on a canary server

# Enabling a New Feature

- …› Dynamic configuration
- …› Deploy with feature turned off
- …› Turn on to 1%, verify, increment

# Bringing up a New Service

- ....> Find something small to break off
- ....> Validate correctness and capacity
- ....> Slowly turn up
- ....> Repeat

# Tweet Service Read-Path

- …→ Reads from secondary services
- …→ Reads from new API services
- …→ Reads from Monorail

# Tweet Service Write-Path

- …→ Moved store/deliver effects one-by-one
- …→ Moved compose phase
- …→ Switched from Monorail to new API service

# Moving an Effect

- …→ Decide in calling service
- …→ Pass flag with RPC call
- …→ Slowly turn up

# If Reliability Matters...

Make big changes in small steps

# Top 3 Lessons Learned

…→ 1) Incremental change eventually wins

…→

…→

# Lessons Learned #2

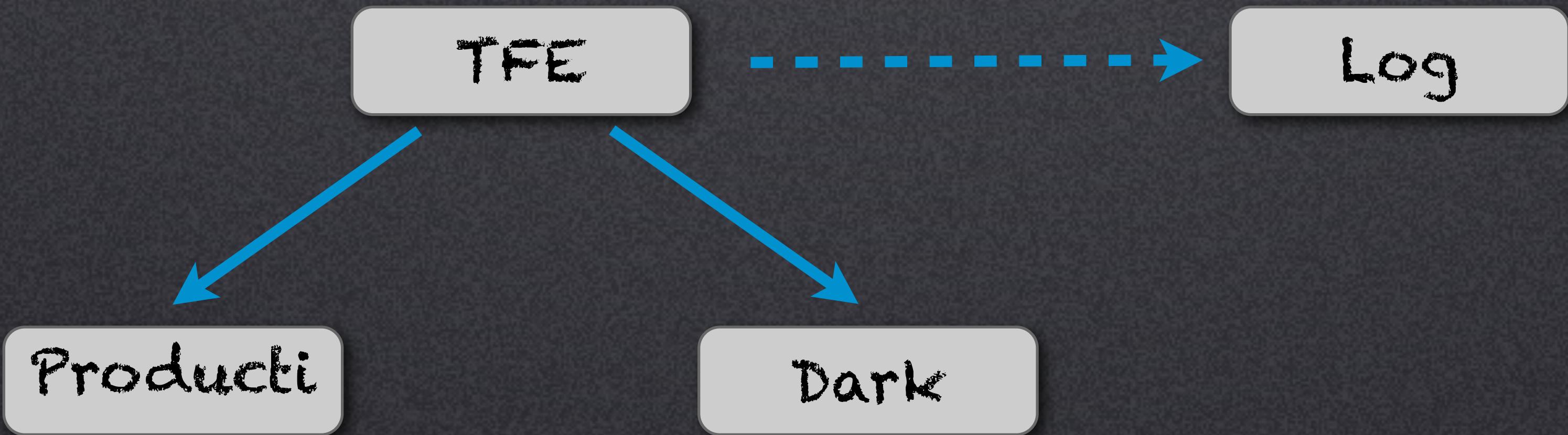
Integration testing is hard

# Scenario #1

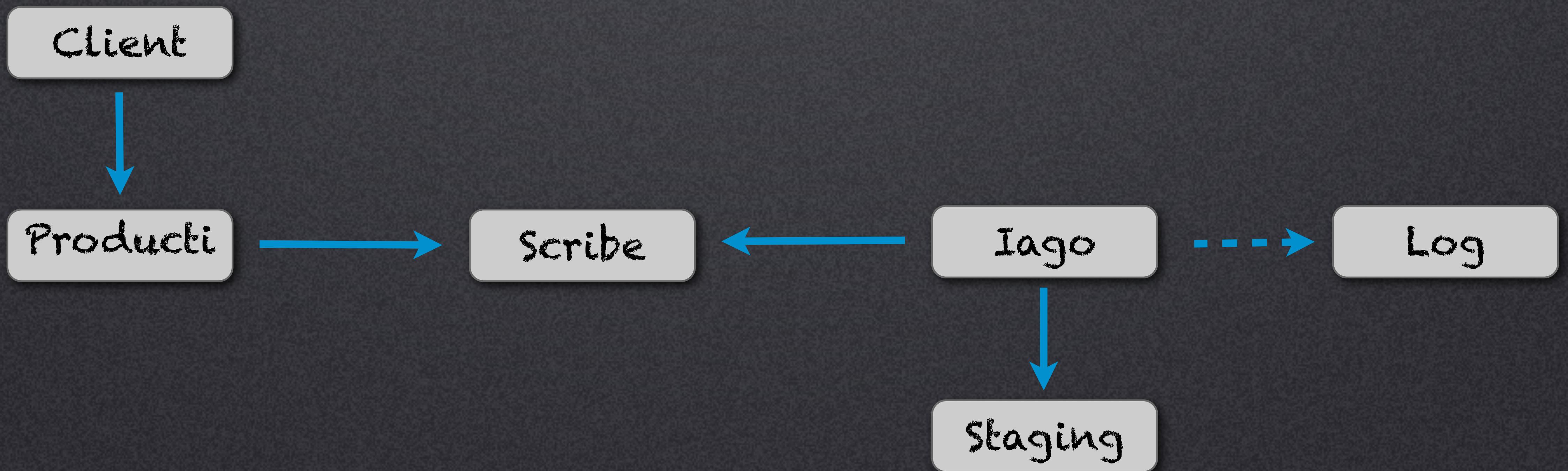
Testing a change within a single service

Difficulty: Easy to Medium

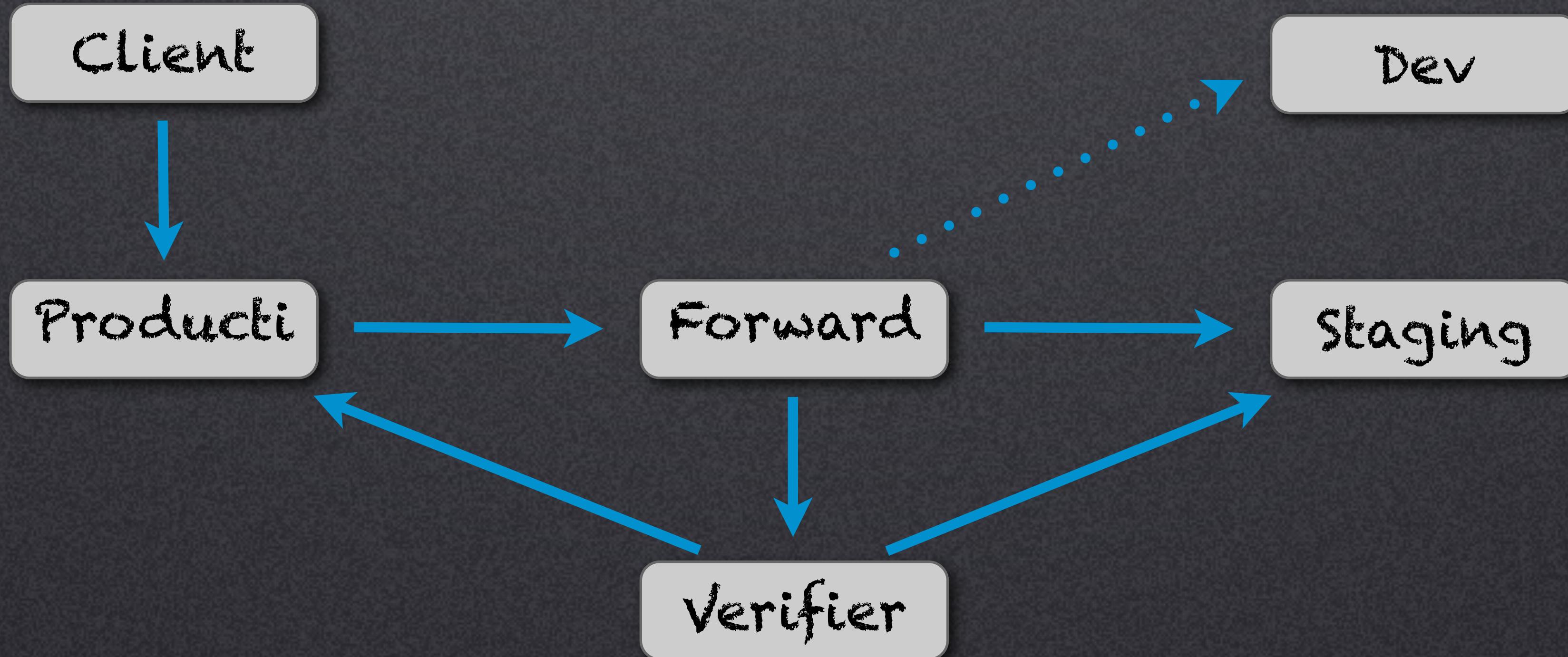
# Tap Compare



# Capture and Replay



# Forked Traffic



# Scenario #2

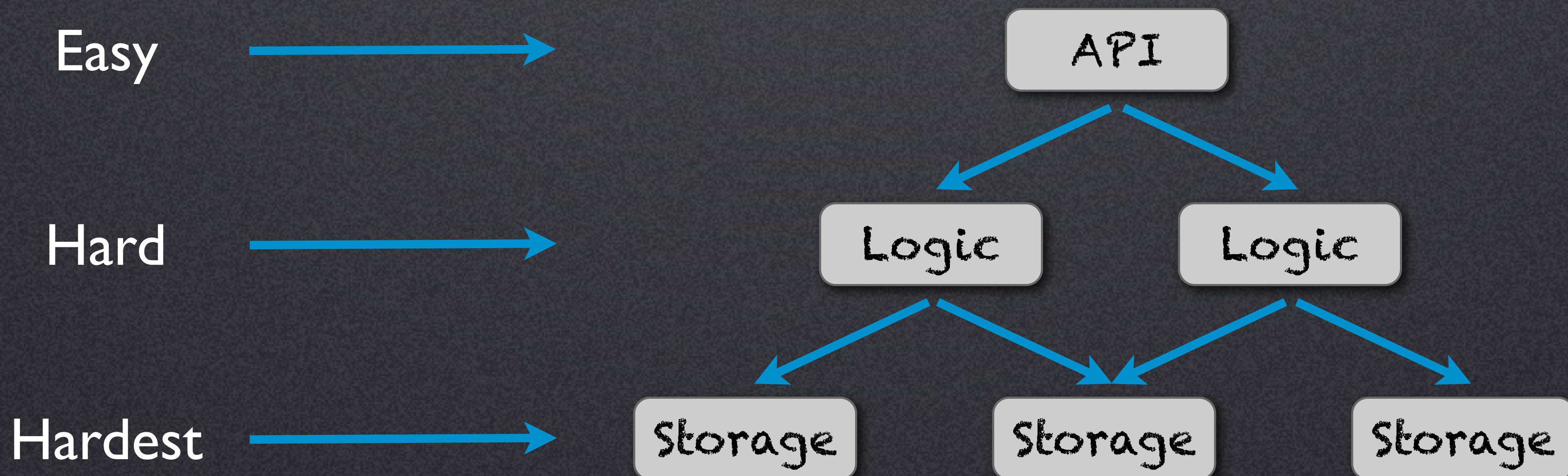
Testing a change that spans multiple services

Difficulty: Medium to Hard

# Challenges

- …› Need to launch multiple services with change
- …› Need to point those services at each other
- …› A single service may have many dependencies (memcache, kestrel, MySQL)
- …› Can't run everything on one box

# Testing Changes



# Unsolved Problem

Manual and tedious process

# Moral of the story

Plan and build your integration testing framework  
early

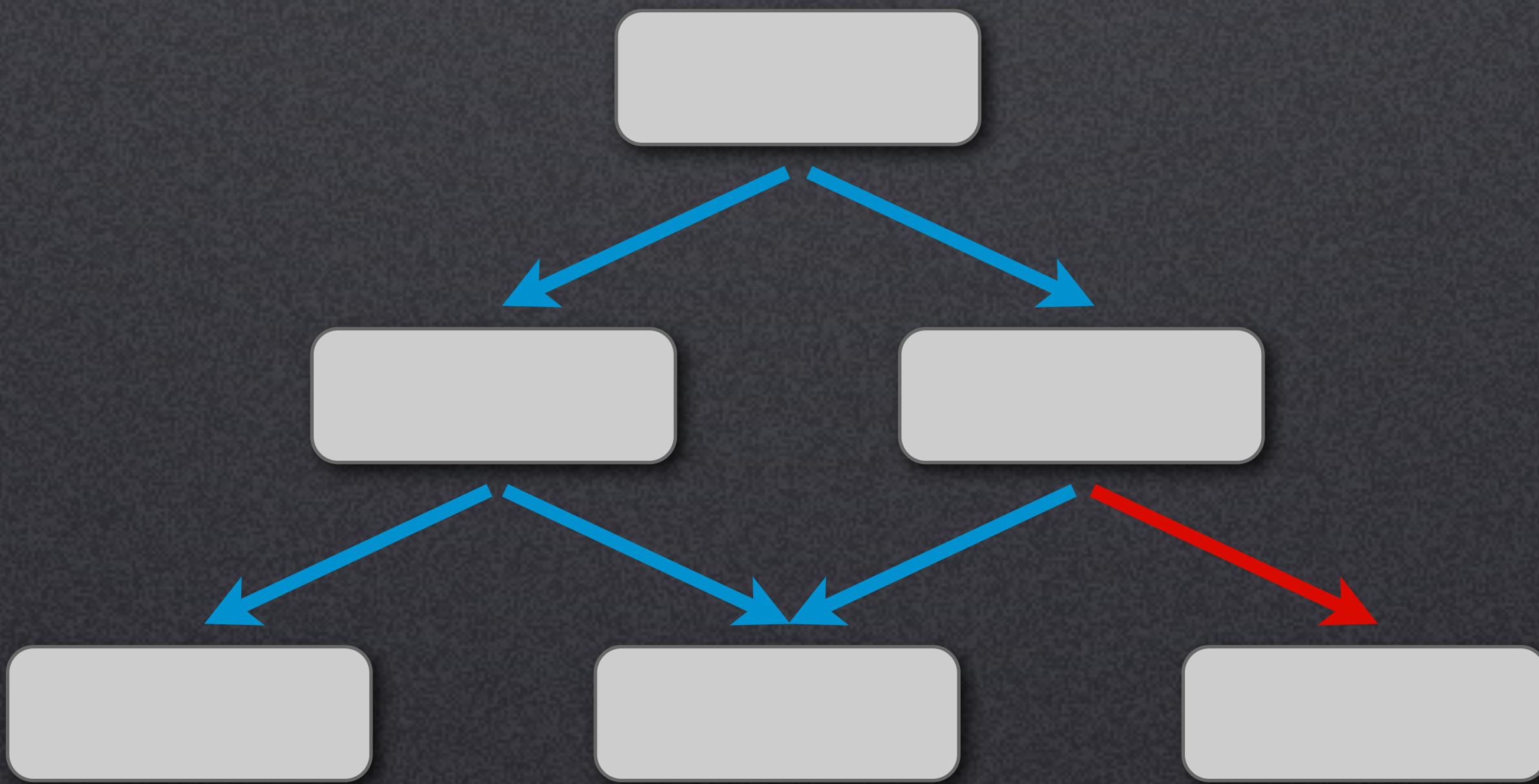
# Top 3 Lessons Learned

- …→ 1) Incremental change eventually wins
- …→ 2) Plan integration testing strategy early
- …→

# Lessons Learned #3

Failure is always an option

# More Hops = More Failures



# Retries can be tricky

- ....> Dogpile!
- ....> Use a library like Finagle if possible
- ....> Route around problems
- ....> Idempotency is critical

# Plan for entire rack failure

- …› Over provision
- …› Mix services within rack
- …› Test this scenario in production

# Degradate gracefully

- …› Avoid SPOFs
- …› Overprovision, add redundancy
- …› Have a fallback strategy
- …› Dark-mode a feature instead of total failure

# Partial Failures

- …→ More common
- …→ Harder to handle
- …→ Harder to imagine

# If Reliability Matters...

Spend most of your time preparing for failures

# Top 3 Lessons Learned

- …→ 1) Incremental change eventually wins
- …→ 2) Integration testing is hard
- …→ 3) Failure is always an option

@JoinTheFlock

