

MOVING MOUNTAINS OF PLAYER DATA

SCALABLE INTERNET SERVICES
UCLA/UCSB - NOV 2017

SEAN MALONEY
RIOT GAMES



THAT IS BALONEY



@SEAN_SEANNERY



SMALONEY@riotgames.com

WHO IS THIS GUY?

Team Lead on Riot's big data and real-time services

FUN FACT:

Was a student in this class 6 years ago
Intern at Appfolio



SEAN MALONEY
BIG DATA ENGINEER

WHO IS THIS GUY?

Team Lead on Riot's big data and
real-time services

FUN FACT:

Was a student in this class 6 years ago
Intern at Appfolio

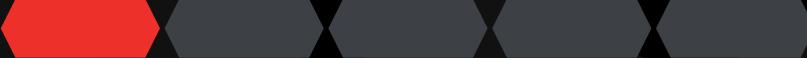


SEAN MALONEY
BIG DATA ENGINEER

MOVING MOUNTAINS OF DATA

1. INTRODUCTION
2. THE GAME PLATFORM: OUR MAIN DATA SOURCE
3. HOW WE INGEST AND QUERY DATA
4. HOW WE SCALE IN AWS
5. CONCLUSION - SEAN'S PRO TIPS

INTRODUCTION



WHAT IS LEAGUE OF LEGENDS?

The background image is a dynamic scene from the game League of Legends. It features several champions in various stages of combat or movement. On the left, a character with red hair and a sword is shown. In the center, a large, armored character with a sword is prominent. To the right, a woman with long red hair and a male character with a staff are visible. The setting is a dark, rocky terrain with a bright sun or moon in the background, casting light on the characters.

2009
LAUNCH

ONLINE
MULTIPLAYER

WINDOWS
/ OSX

40-50 MIN
GAMES

YOUR CHAMP



THE
TEAM

THE
BATTLE
GROUND



12 BILLION
GAME RELATED EVENTS

0.5 TRILLION
DATA POINTS

50 TB
STORAGE

DAILY

26 PETABYTES
PLAYER DATA

SINCE BETA

OUR MISSION

WE ASPIRE
TO BE THE MOST
PLAYER

FOCUSED
GAME COMPANY IN THE
WORLD

OFFENSIVE CHAT DETECTION

Data science team queries all chat messages in game

Sentiment analysis and classification

Jônas has ended ezes2396's killing spree! (Bounty: 500G)

Jônas (Master Yi): sry

ShadowMaster3000 (Vi): **** your mother **** yi you **** you noob

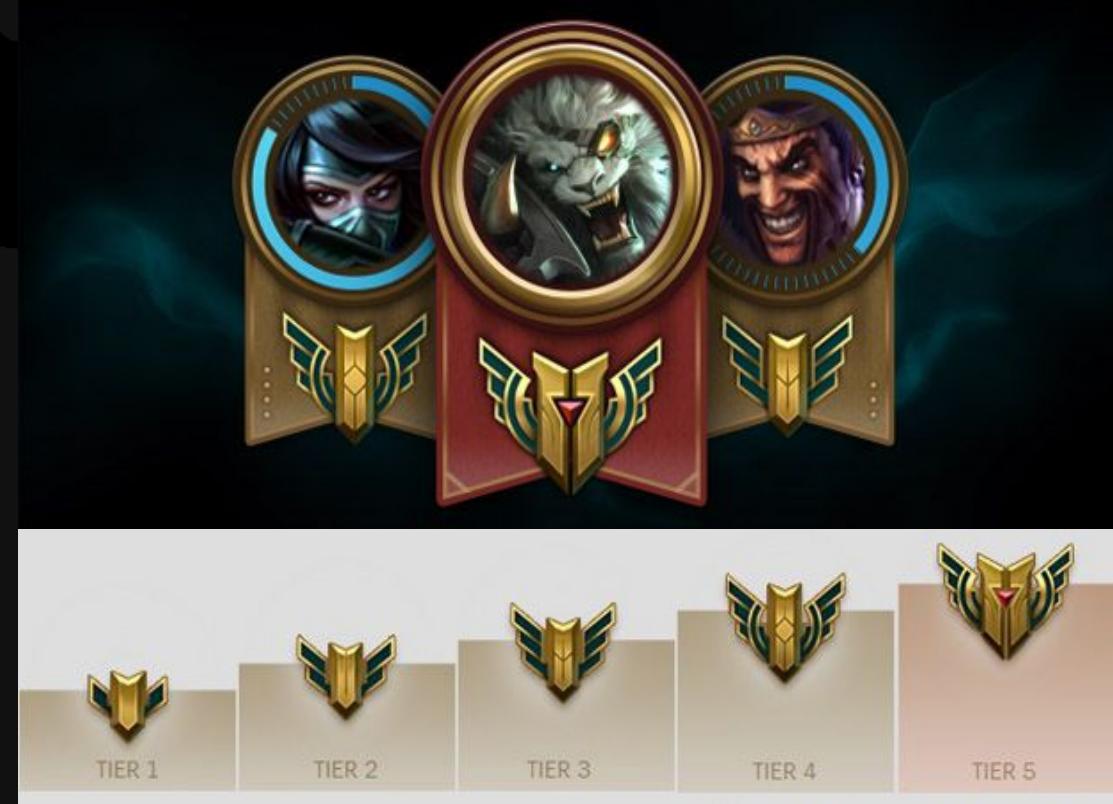
ShadowMaster3000 (Vi): i die and i make ulti and fier and YOPU KILL

Jônas (Master Yi): :DDDDDD

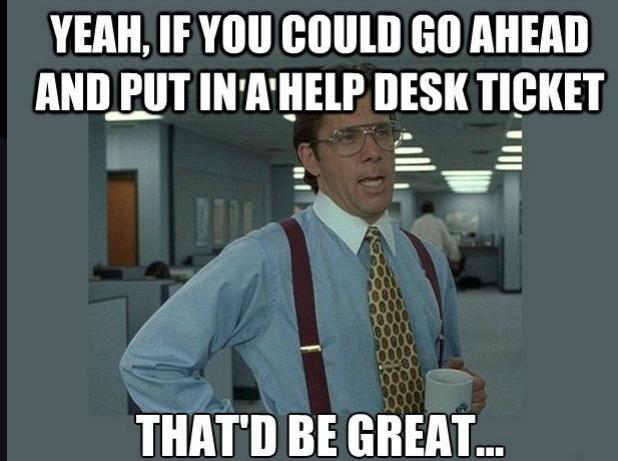
ShadowMaster3000 (Vi): i report you ****



CHAMPION MASTERY



PLAYER SUPPORT



GAME BALANCE

FIRST BLOOD RATE BY CHAMPION

RANK	CHAMPION	FIRST BLOOD RATE
1	TALON	24.9%
2	PANTHEON	20.2%
3	KATARINA	19.8%
4	EVELYNN	19.5%
5	LEBLANC	18.6%
6	LEE SIN	17.9%
7	TRYNDAMERE	17.6%

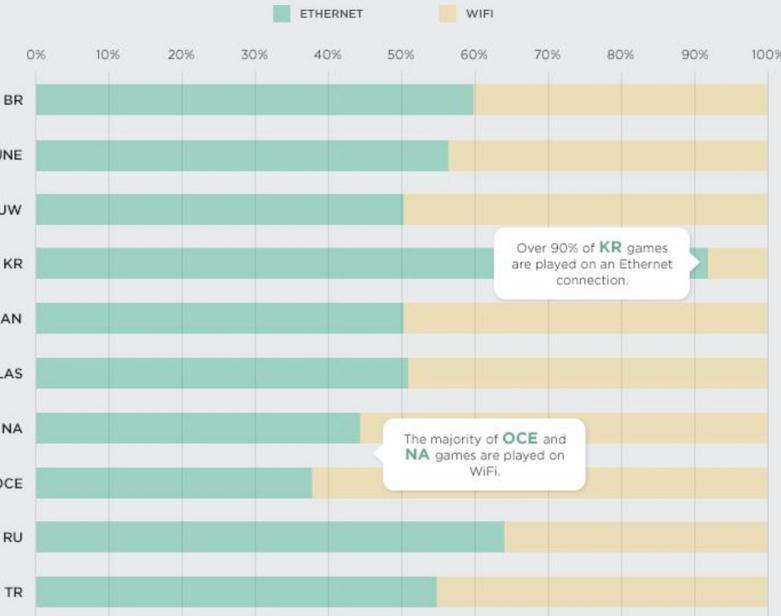
POPULARITY OF KEYBINDINGS FOR FLASH, WORLDWIDE

BY PERCENTAGE OF RECORDED GAMES

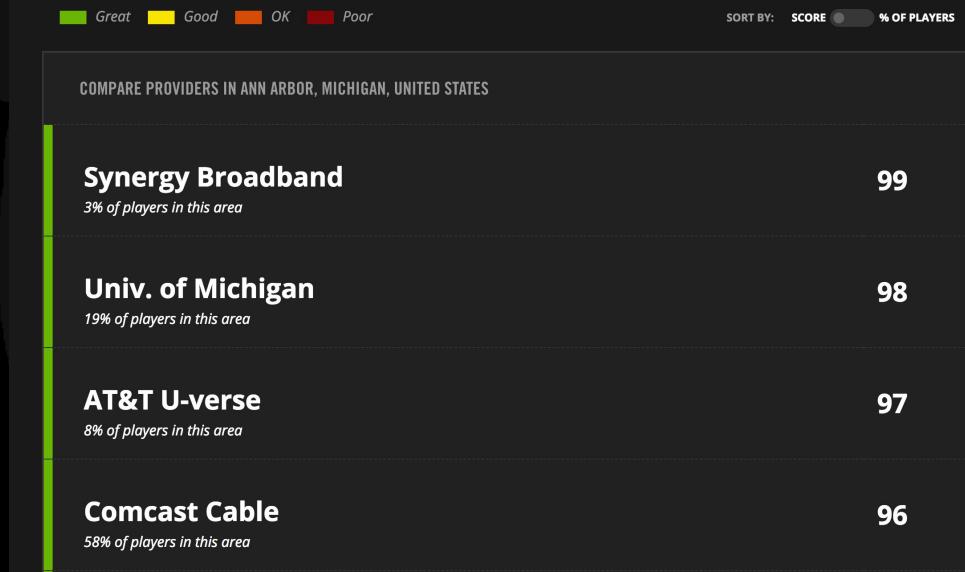


LATENCY AND NETWORK

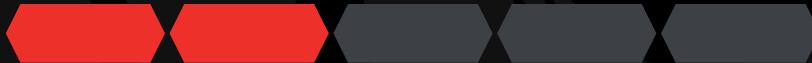
PERCENT OF GAMES PLAYED BY CONNECTION TYPE
BY REGION



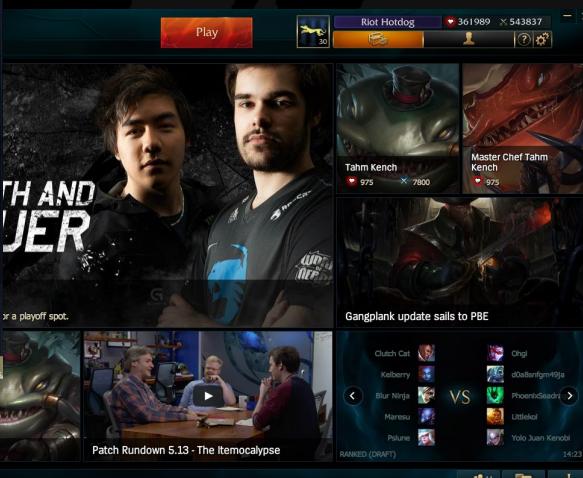
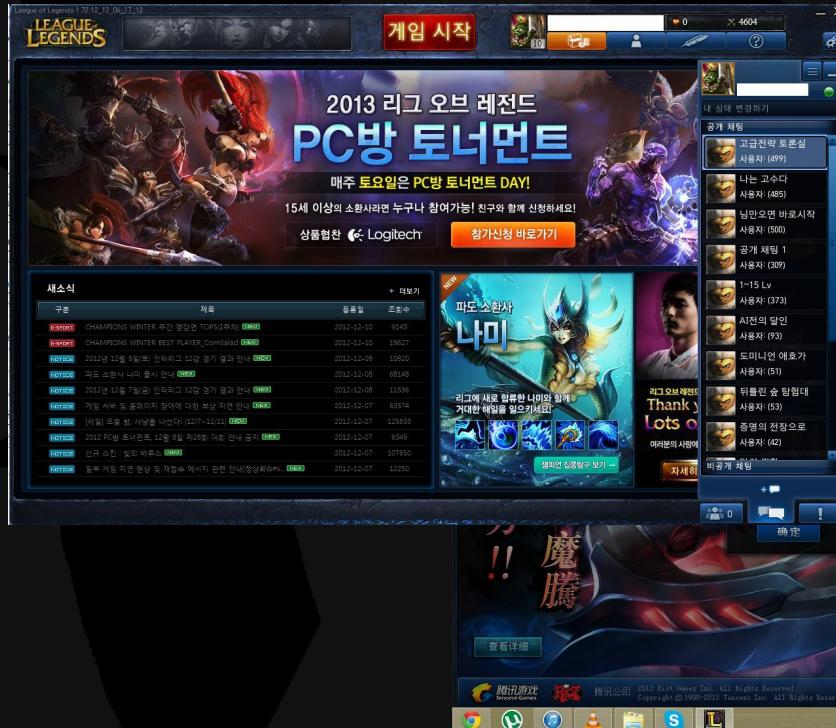
Internet Service Provider Leaderboard [?](#)



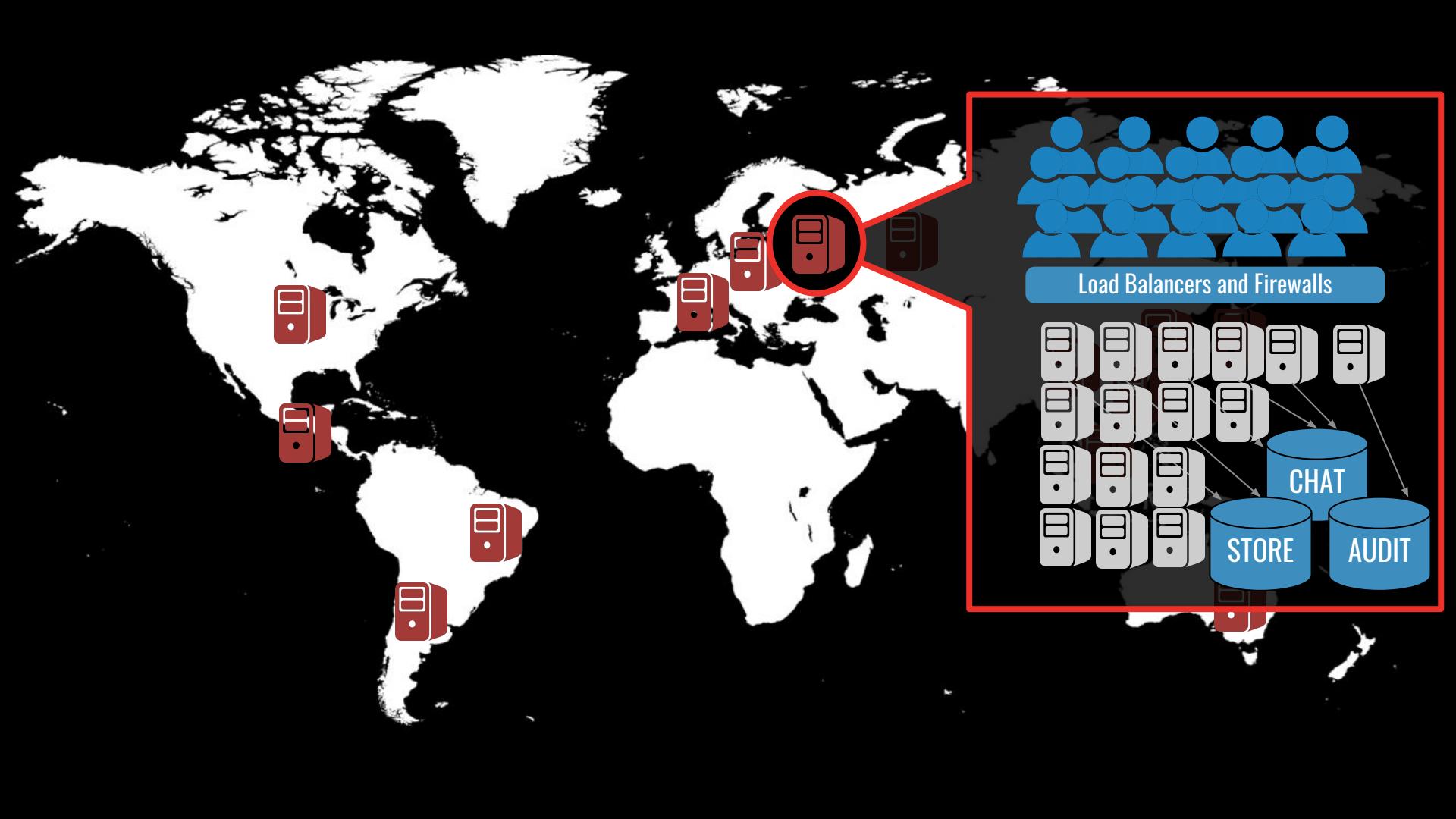
THE GAME PLATFORM

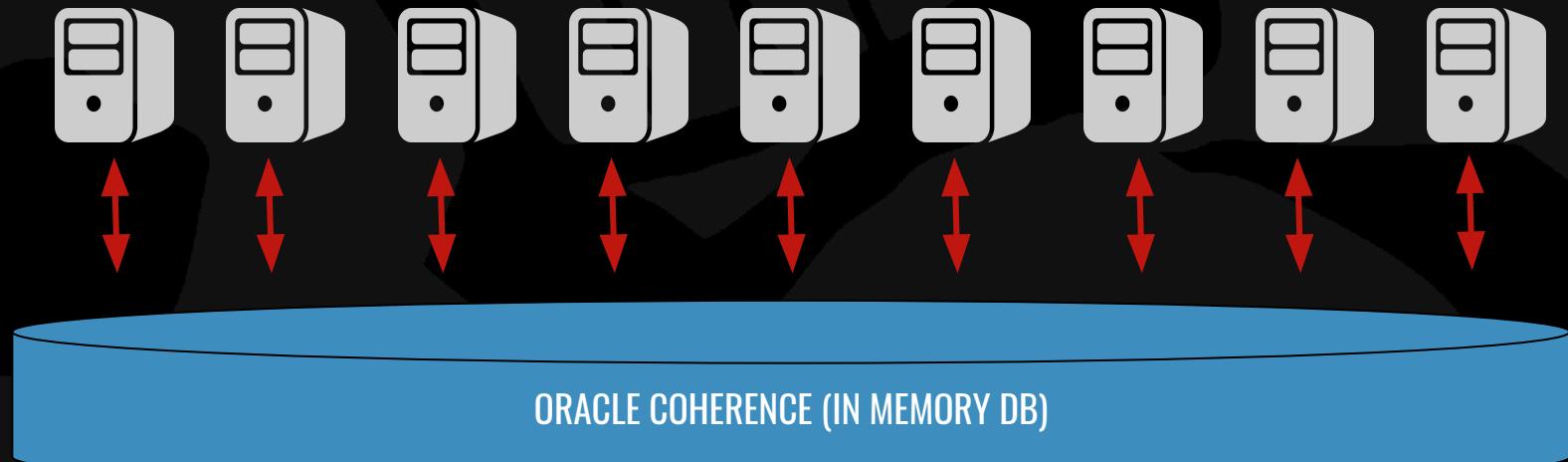


THE CLIENT.









PRIMARY DB

HOT BACKUP DB

2nd BACKUP DB /
ETL

CHAT

CHAT

CHAT

STORE

STORE

STORE

AUDIT

AUDIT

AUDIT

GAME

GAME

GAME

ETC.

ETC.

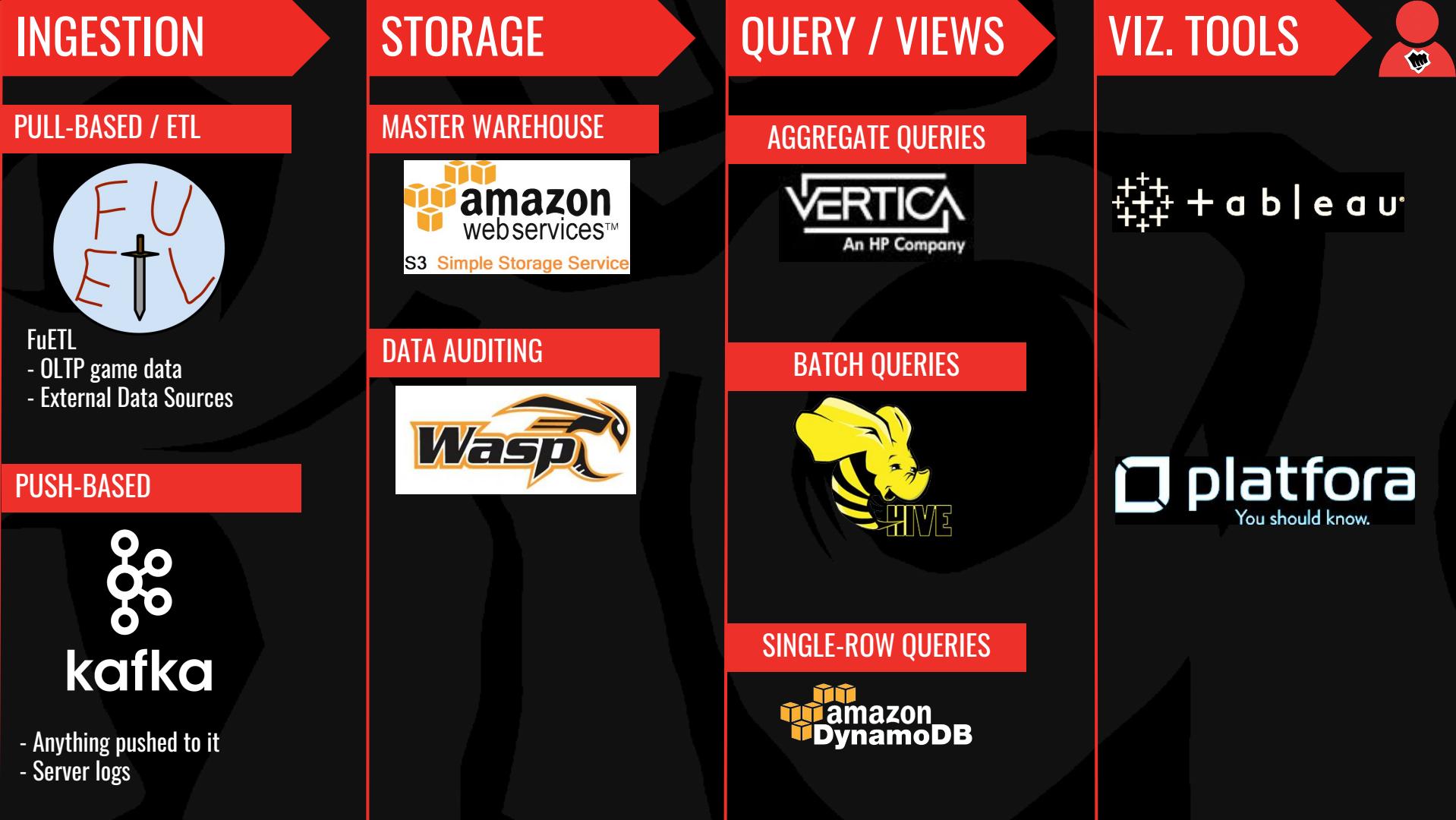
ETC.





DATA INGESTION





INGESTION

PULL-BASED / ETL



FuETL

- OLTP game data
- External Data Sources

PUSH-BASED



kafka

- Anything pushed to it
- Server logs

STORAGE

MASTER WAREHOUSE



DATA AUDITING



QUERY / VIEWS

AGGREGATE QUERIES



BATCH QUERIES



SINGLE-ROW QUERIES



VIZ. TOOLS





Distributed ETL Software written in Ruby.

Same ETL applied to multiple regions / datacenters

Scales Horizontally

BEST LOGO EVER!

NA

First Name	Last Name	Address	City	Age
Mickey	Mouse	123 Fantasy Way	Anaheim	73
Bat	Man	321 Cavern Ave	Gotham	54
Wonder	Woman	987 Truth Way	Paradise	39
Donald	Duck	555 Quack Street	Mallard	65
Bugs	Bunny	567 Carrot Street	Rascal	58
Wiley	Coyote	999 Acme Way	Canyon	61
Cat	Woman	234 Purrfect Street	Hairball	32
Tweety	Bird	543	ItotltaW	28

Korea

First Name	Last Name	Address	City	Age
Mickey	Mouse	123 Fantasy Way	Anaheim	73
Bat	Man	321 Cavern Ave	Gotham	54
Wonder	Woman	987 Truth Way	Paradise	39
Donald	Duck	555 Quack Street	Mallard	65
Bugs	Bunny	567 Carrot Street	Rascal	58
Wiley	Coyote	999 Acme Way	Canyon	61
Cat	Woman	234 Purrfect Street	Hairball	32
Tweety	Bird	543	ItotltaW	28

Russia

First Name	Last Name	Address	City	Age
Mickey	Mouse	123 Fantasy Way	Anaheim	73
Bat	Man	321 Cavern Ave	Gotham	54
Wonder	Woman	987 Truth Way	Paradise	39
Donald	Duck	555 Quack Street	Mallard	65
Bugs	Bunny	567 Carrot Street	Rascal	58
Wiley	Coyote	999 Acme Way	Canyon	61
Cat	Woman	234 Purrfect Street	Hairball	32
Tweety	Bird	543	ItotltaW	28

First Name	Last Name	Address	City	Age
Mickey	Mouse	123 Fantasy Way	Anaheim	73
Bat	Man	321 Cavern Ave	Gotham	54
Wonder	Woman	987 Truth Way	Paradise	39
Donald	Duck	555 Quack Street	Mallard	65
Bugs	Bunny	567 Carrot Street	Rascal	58
Wiley	Coyote	999 Acme Way	Canyon	61
Cat	Woman	234 Purrfect Street	Hairball	32
Tweety	Bird	543	ItotltaW	28

OTHER DATA SOURCES



<REST>



FUETL CAN CONNECT TO



- Amazon S3
- SQS
- (S)FTP
- Hive
- Microsoft SQL Server
- MySQL
- DynamoDB
- Vertica
- Redshift
- REST websites

Create an ETL

Source Helper: mysql_fantasycls_server

Source Table(s):
Table: fantasy_users
Date_column: create_date
Hour_column:
Realm_column: region

Add

Target Helper: vertica_test_cluster

Target Table: warehouse.fantasy_users

Target Realm Column: dt

Target Date Column: region

Query:

```
select
<%= environment_id %>
, segmentation_date
, year(segmentation_date)
, month(segmentation_date)
, acct_id
, game_count
from fake_db.fantasy_users
where env = '<%= environment_name %>'
and segmentation_date = '<%= start_date %>'
and acct_id is not null
```

Create an ETL

Source Helper

Source Table(s)
Date_column: create_date
Hour_column:
Realm_column: region

Target Helper

Target Table

Target Realm Column

Target Date Column

Query

```
select
<%= environment_id %>
, segmentation_date
, year(segmentation_date)
, month(segmentation_date)
, acct_id
, game_count
from fake_db.fantasy_users
where env = '<%= environment_name %>'
and segmentation_date = '<%= start_date %>'  
and acct_id is not null
```

Create an ETL

Source Helper: mysql_fantasylcs_server

Source Table(s): Table: fantasy_users
Date_column: create_date
Hour_column:
Realm_column: region

Add

Target Helper: vertica_test_cluster

Target Table: warehouse.fantasy_users

Target Realm Column: dt

Target Date Column: region

Query:

```
select
<%= environment_id %>
, segmentation_date
, year(segmentation_date)
, month(segmentation_date)
, acct_id
, game_count
from fake_db.fantasy_users
where env = '<%= environment_name %>'
and segmentation_date = '<%= start_date %>'
and acct_id is not null
```

mysql_to_vertica/store_items (SQLToSQL)

Task Config

Schedule or Audit Runs



Environments:

16 selected ▾

Search

<input type="checkbox"/> No environment	<input checked="" type="checkbox"/> BR1	<input type="checkbox"/> CN1	<input type="checkbox"/> EDU1
<input checked="" type="checkbox"/> EUN1	<input checked="" type="checkbox"/> EUW1	<input type="checkbox"/> GLB	<input type="checkbox"/> HN1
<input type="checkbox"/> HN10	<input type="checkbox"/> HN11	<input type="checkbox"/> HN12	<input type="checkbox"/> HN13
<input type="checkbox"/> HN14	<input type="checkbox"/> HN15	<input type="checkbox"/> HN16	<input type="checkbox"/> HN17
<input type="checkbox"/> HN18	<input type="checkbox"/> HN19	<input type="checkbox"/> HN2	<input type="checkbox"/> HN20
<input type="checkbox"/> HN3	<input type="checkbox"/> HN4	<input type="checkbox"/> HN5	<input type="checkbox"/> HN6
<input type="checkbox"/> HN7	<input type="checkbox"/> HN8	<input type="checkbox"/> HN9	<input type="checkbox"/> ID1
<input checked="" type="checkbox"/> KR1	<input checked="" type="checkbox"/> LA1	<input checked="" type="checkbox"/> LA2	<input checked="" type="checkbox"/> NA1
<input checked="" type="checkbox"/> OC1	<input checked="" type="checkbox"/> PBE1	<input checked="" type="checkbox"/> PH1	<input checked="" type="checkbox"/> RU1
<input checked="" type="checkbox"/> SG1	<input checked="" type="checkbox"/> TH1	<input checked="" type="checkbox"/> TR1	<input type="checkbox"/> TREU
<input type="checkbox"/> TRKR	<input type="checkbox"/> TRNA	<input type="checkbox"/> TRSA	<input type="checkbox"/> TRTW
<input checked="" type="checkbox"/> TW1	<input checked="" type="checkbox"/> VN1	<input type="checkbox"/> WT1	<input type="checkbox"/> WT2
<input type="checkbox"/> WT3	<input type="checkbox"/> WT4	<input type="checkbox"/> WT5	<input type="checkbox"/> WT6
<input type="checkbox"/> WT7			

mysql_to_vertica/store_items (SQLToSQL)

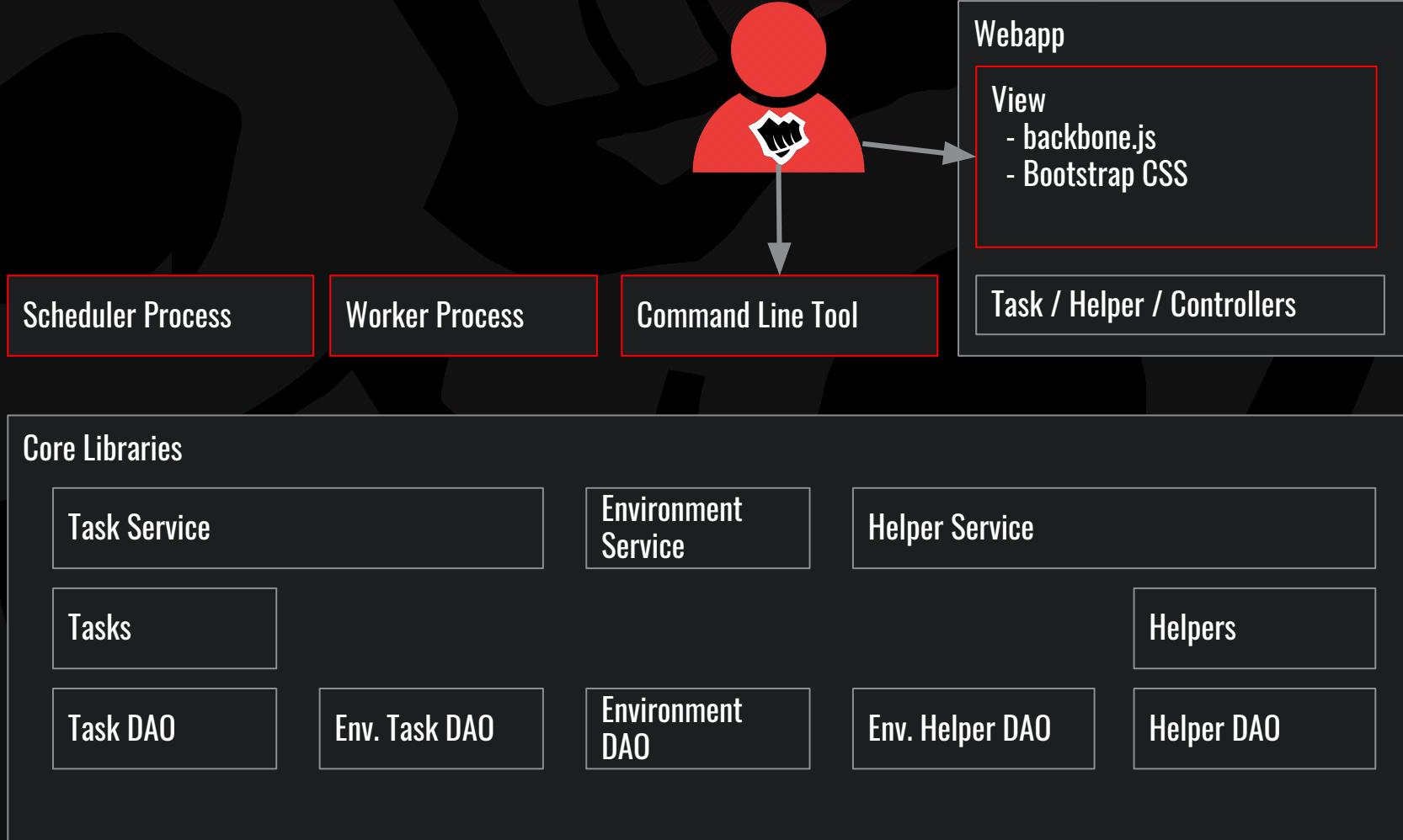
[Task Config](#)[Schedule or Audit Runs](#)

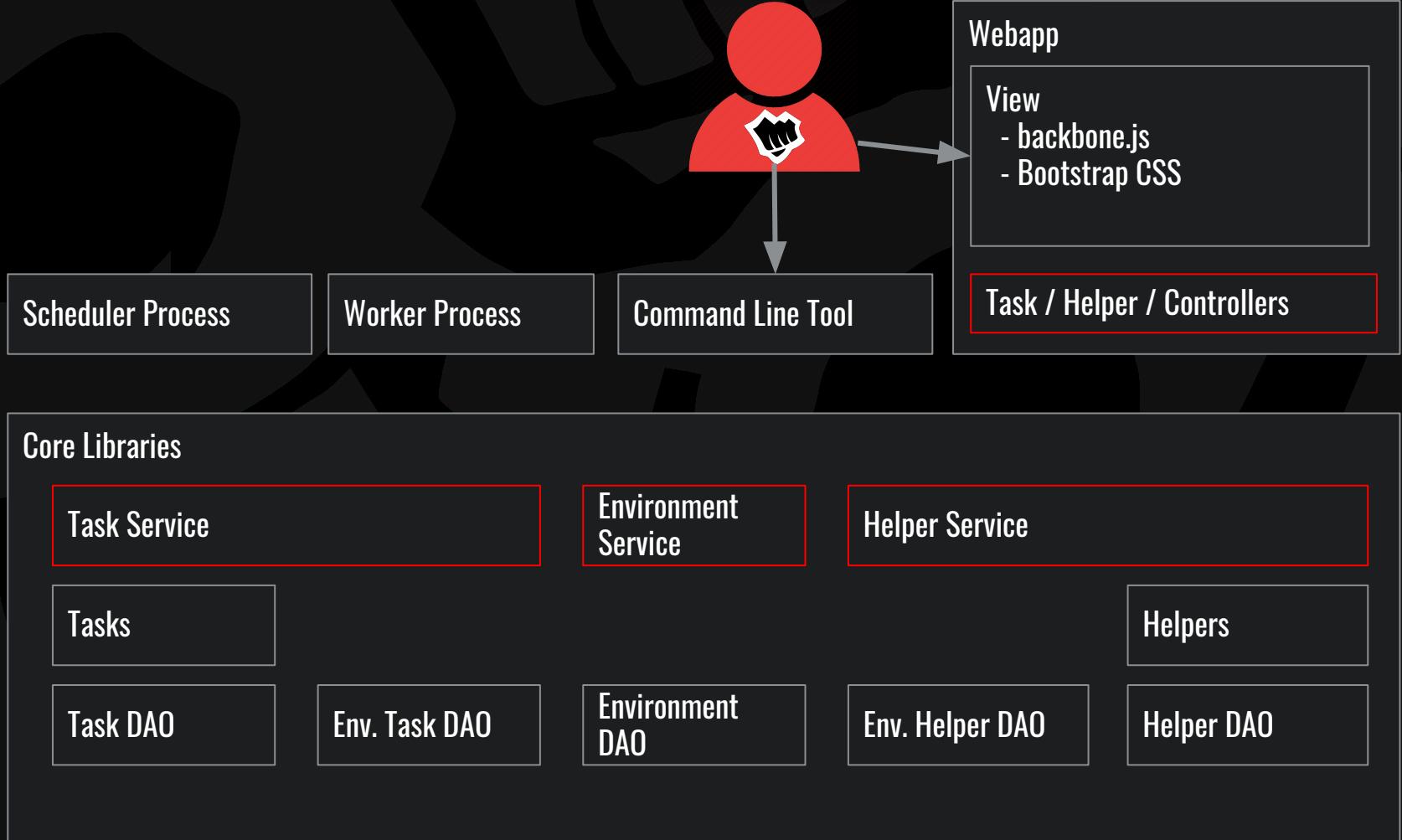
Environments: 16 selected ▾

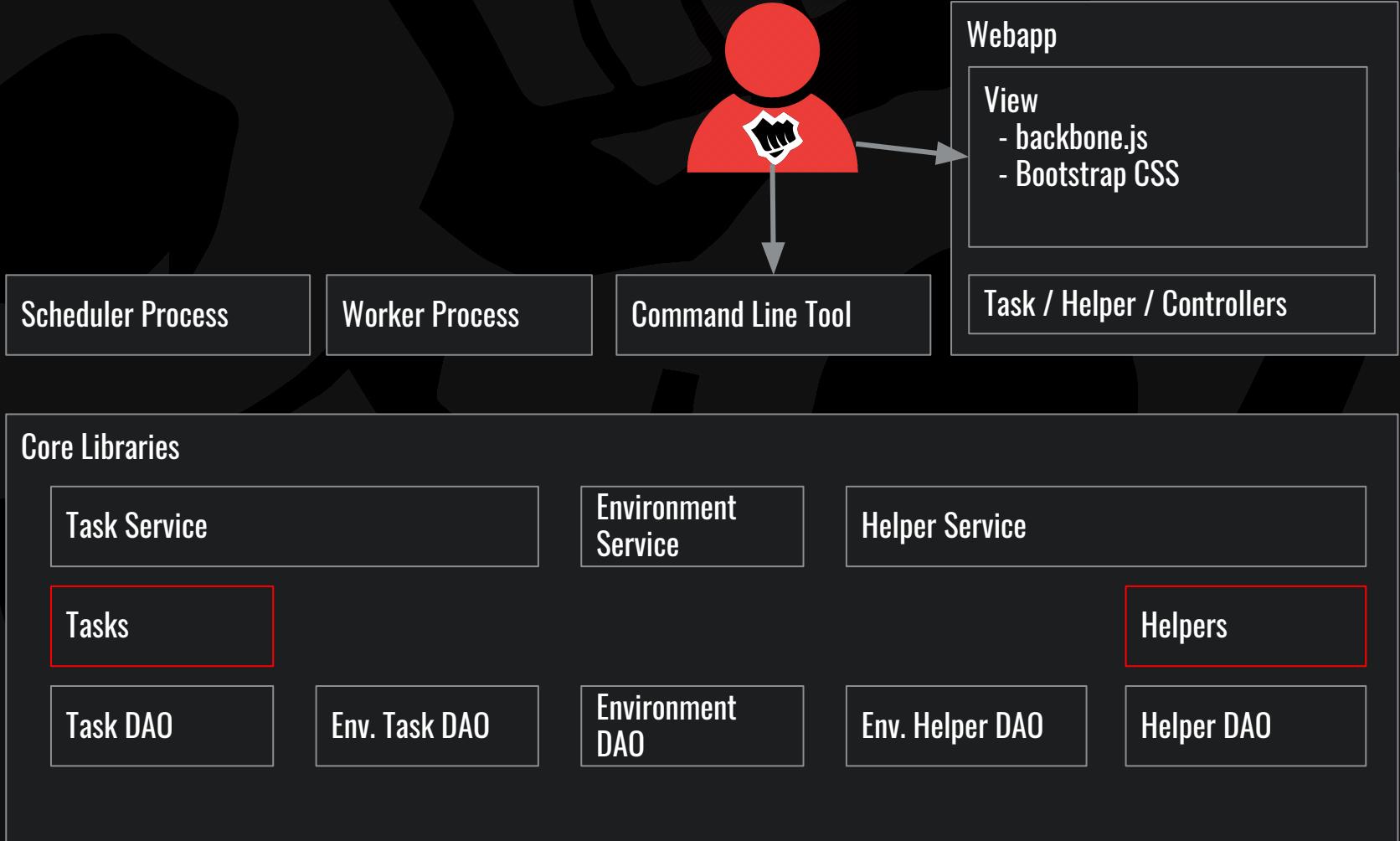
BR1 (9)	success	at 7/9/2015, 4:00:44 PM	Will run again at approximately: 7/9/2015, 10:00:44 PM		Elapsed: 0:00:10 Median: 0:00:08 Longest: 0:00:33
EUN1 (3)	success	at 7/9/2015, 4:15:40 PM	Will run again at approximately: 7/9/2015, 10:15:40 PM		Elapsed: 0:00:05 Median: 0:00:05 Longest: 0:00:39
EUW1 (2)	success	at 7/9/2015, 4:20:30 PM	Will run again at approximately: 7/9/2015, 10:20:30 PM		Elapsed: 0:00:07 Median: 0:00:07 Longest: 0:00:31
KR1 (4)	success	at 7/9/2015, 4:03:15 PM	Will run again at approximately: 7/9/2015, 10:03:15 PM		Elapsed: 0:00:11 Median: 0:00:12 Longest: 0:00:32
LA1 (37)	success	at 7/9/2015, 4:03:22 PM	Will run again at approximately: 7/9/2015, 10:03:22 PM		Elapsed: 0:00:06 Median: 0:00:06 Longest: 0:00:26
LA2 (38)	success	at 7/9/2015, 3:58:17 PM	Will run again at approximately: 7/9/2015, 9:58:17 PM		Elapsed: 0:00:06 Median: 0:00:06 Longest: 0:00:56
NA1 (1)	success	at 7/9/2015, 4:03:08 PM	Will run again at approximately: 7/9/2015, 10:03:08 PM		Elapsed: 0:00:07 Median: 0:00:08 Longest: 0:00:30

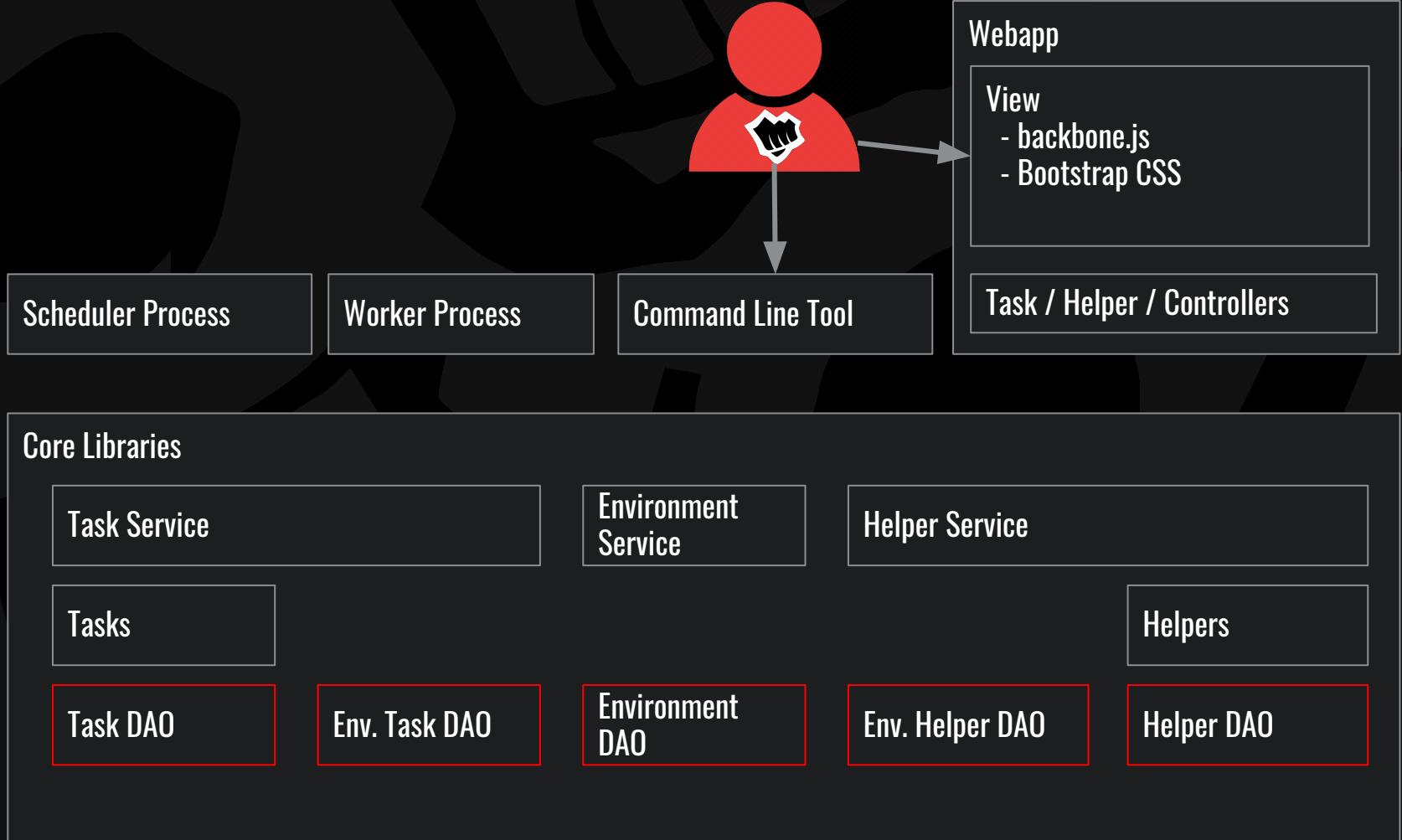
[Latest Run](#)[Run History](#)[Live Logs](#)[Overrides](#)

Timestamp	Status	Message	Payload	Interval
7/9/2015, 4:03:07 PM	success	Transferred 2203 rows of data	2203	No Interval
7/9/2015, 4:02:59 PM	running	None	None	No Interval









FuETL STATISTICS



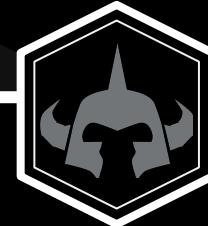
5213

ACTIVE REGIONAL
ETLS



23125

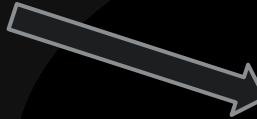
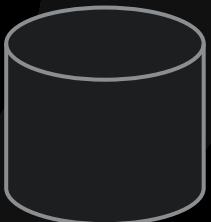
DAILY ETL RUNS



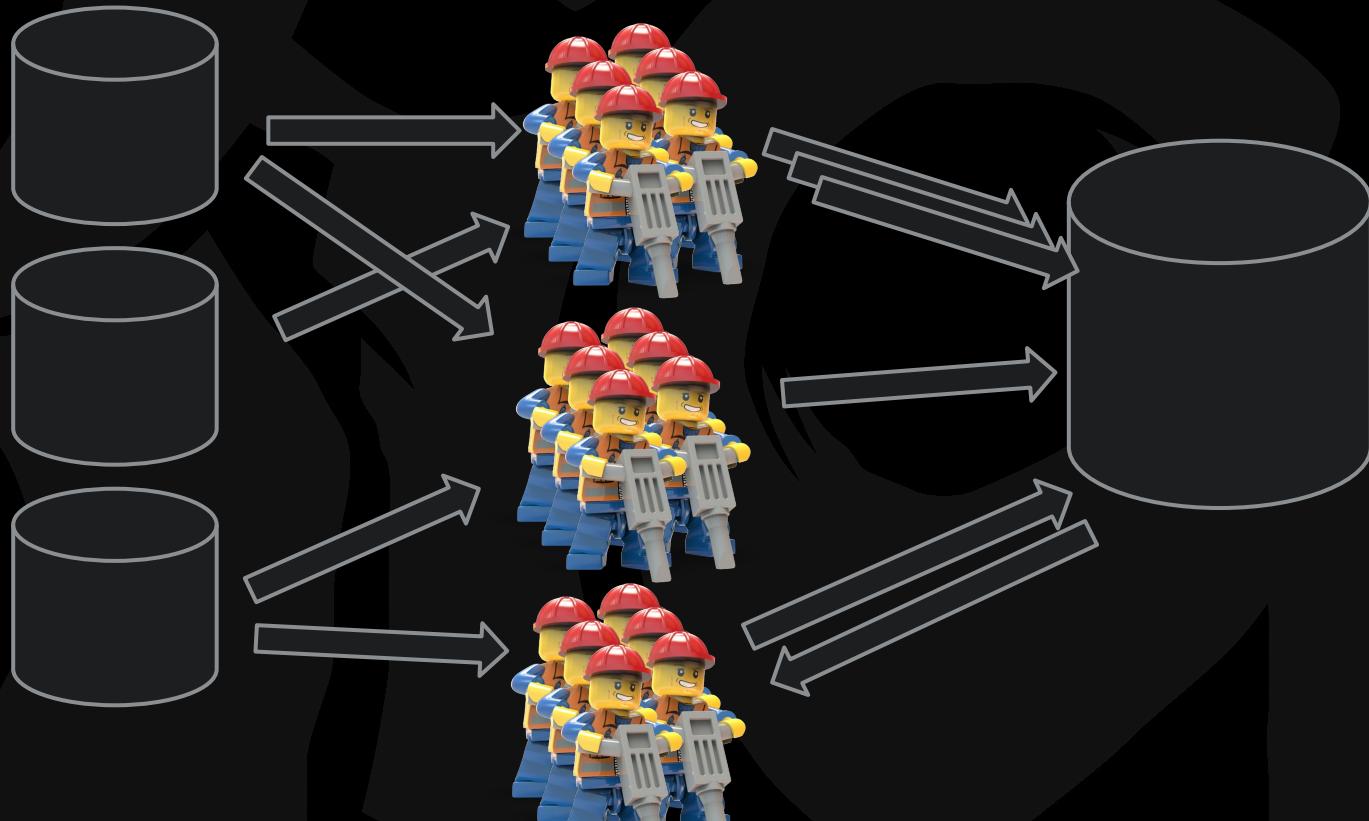
14 TB

DATA MOVED DAILY

FuETL SCALING



FuETL SCALING



Message Queues



SCHEDULER
aka
PRODUCER



ETL1

ETL2

ETL4

ETL3



WORKER
aka
CONSUMER

Message Queues

- Redundancy
- Delivery Guarantees
- Easy to Scale
- Asynchronous Communication
- Abstraction / Decoupling

Message Queues

- Amazon Simple Queue Service
- Apache ActiveMQ
- RabbitMQ
- HornetQ
- Microsoft MQ (MSMQ)

Message Queues



SCHEDULER
aka
PRODUCER



ETL1

X

X



WORKER
aka
CONSUMER

Message Queues



SCHEDULER
aka
PRODUCER



ETL1

ETL1

X



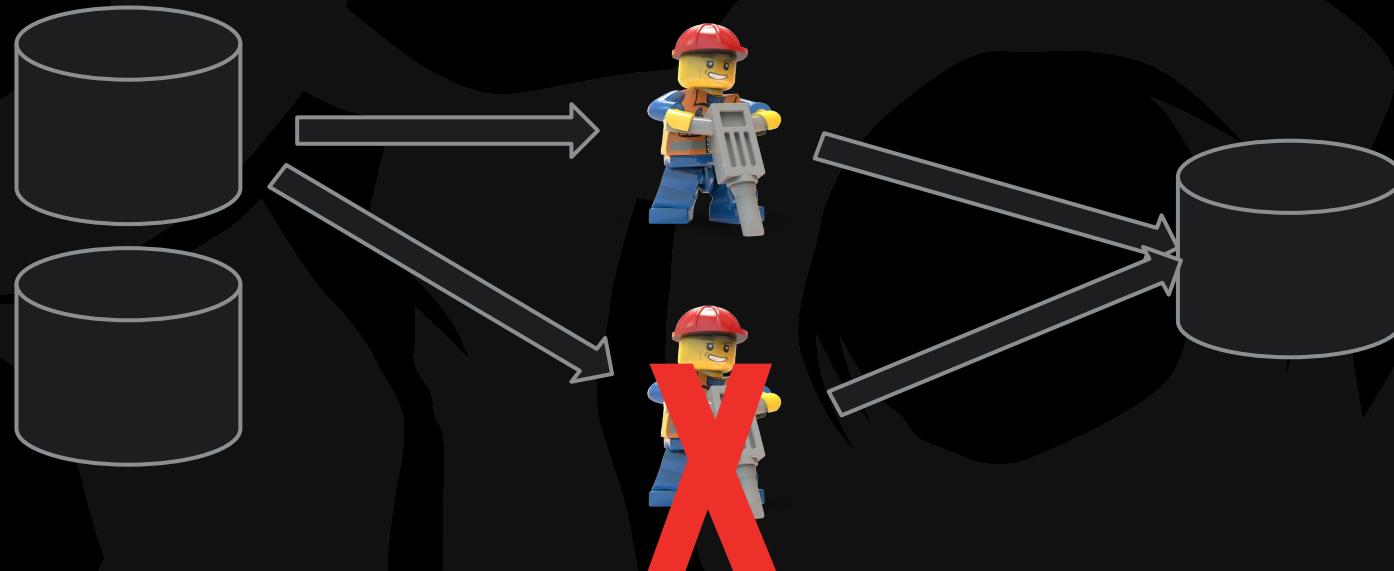
WORKER
aka
CONSUMER

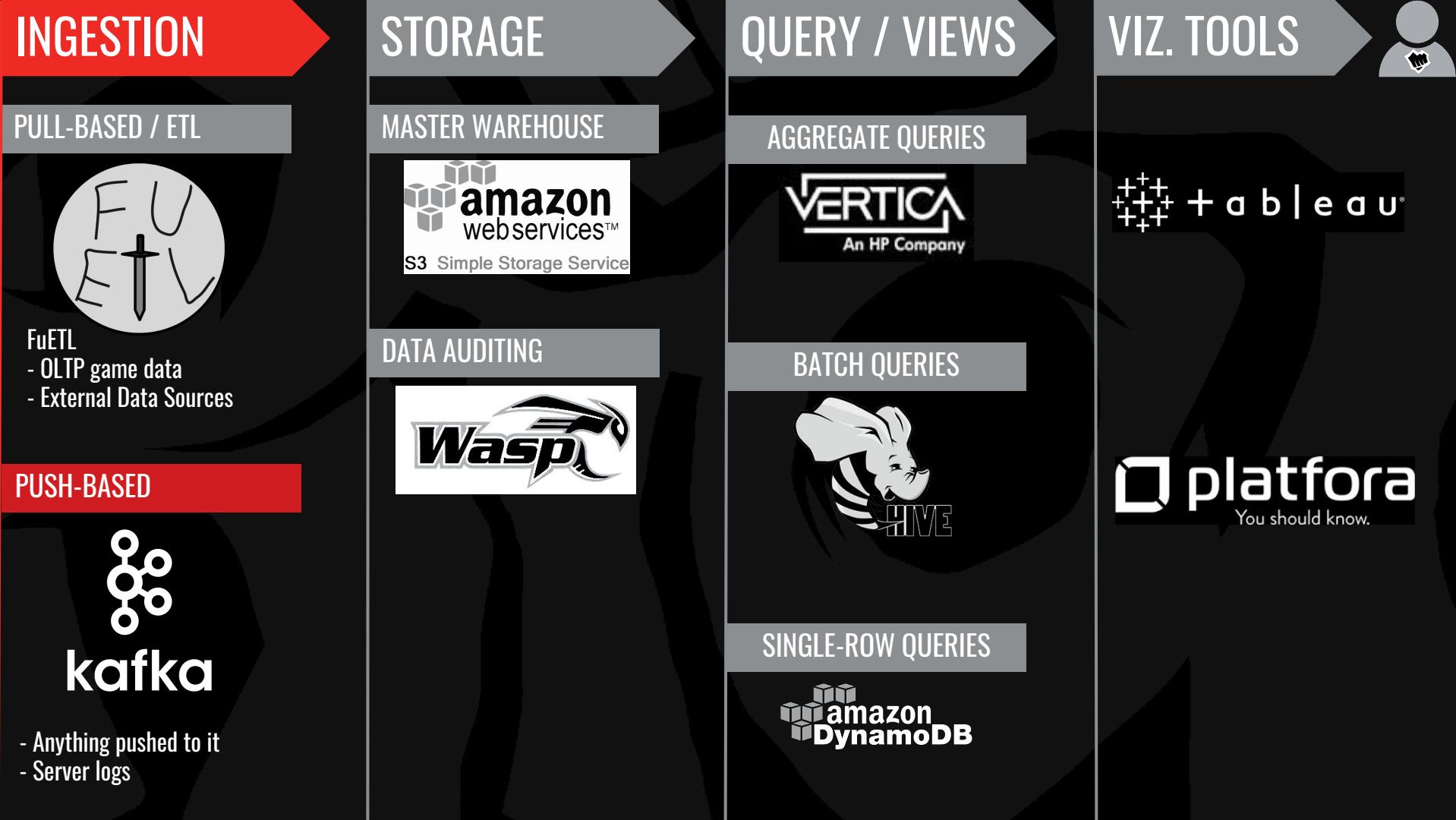
What will happen

In the big data / OLAP world....
(hint: no primary key validation)

```
INSERT INTO games_played
(SELECT * FROM games_played_na
WHERE date >= '2015-10-25')
```

KEEPING INTEGRITY





Kafka

The new hotness in big data

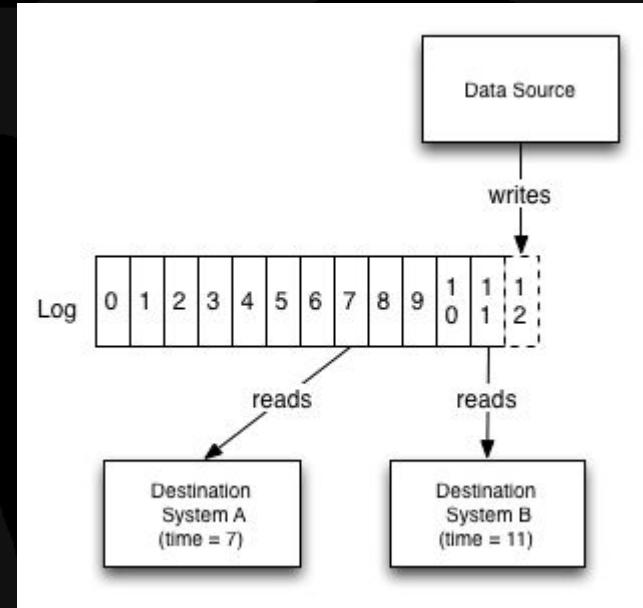
Open-source project maintained by
Confluent

Very fast distributed message queue

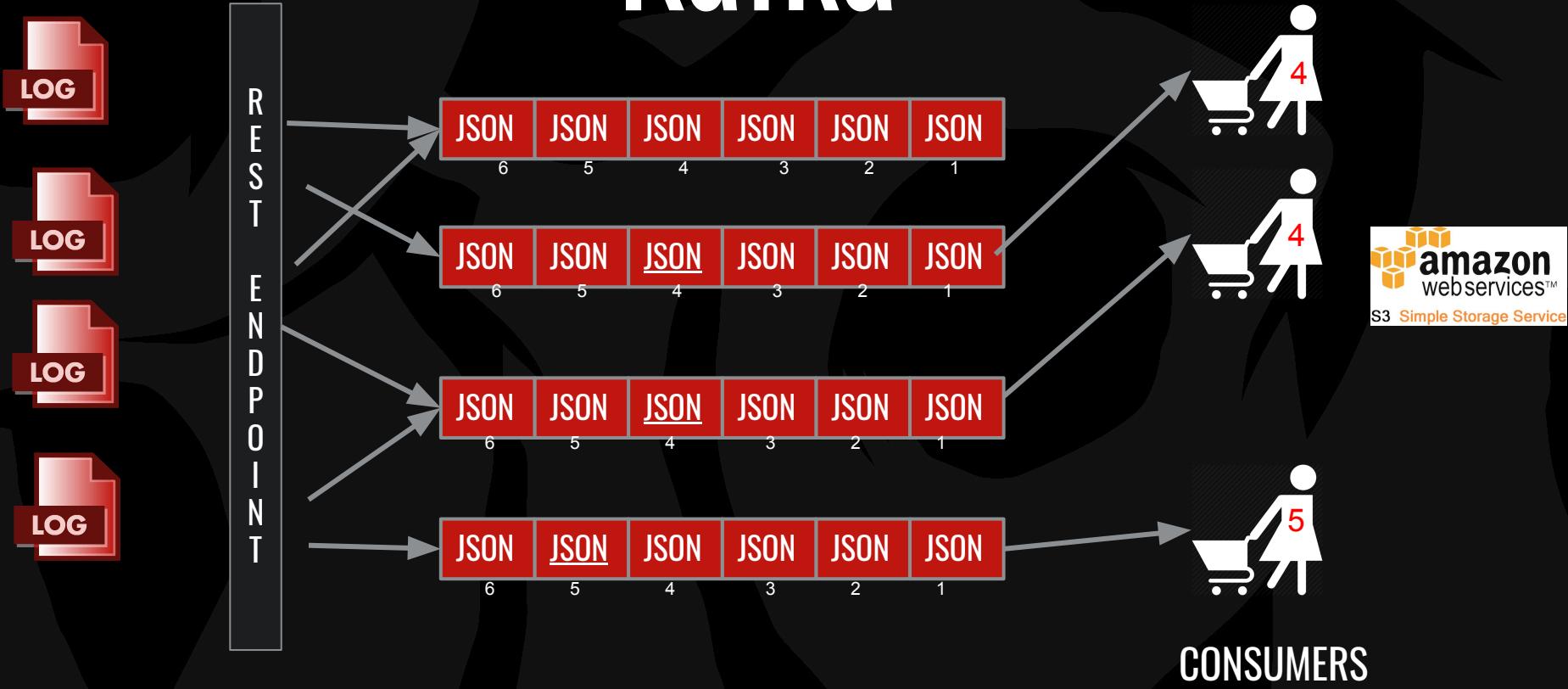
Data is replicated across “partitions” to
ensure no loss

Kafka

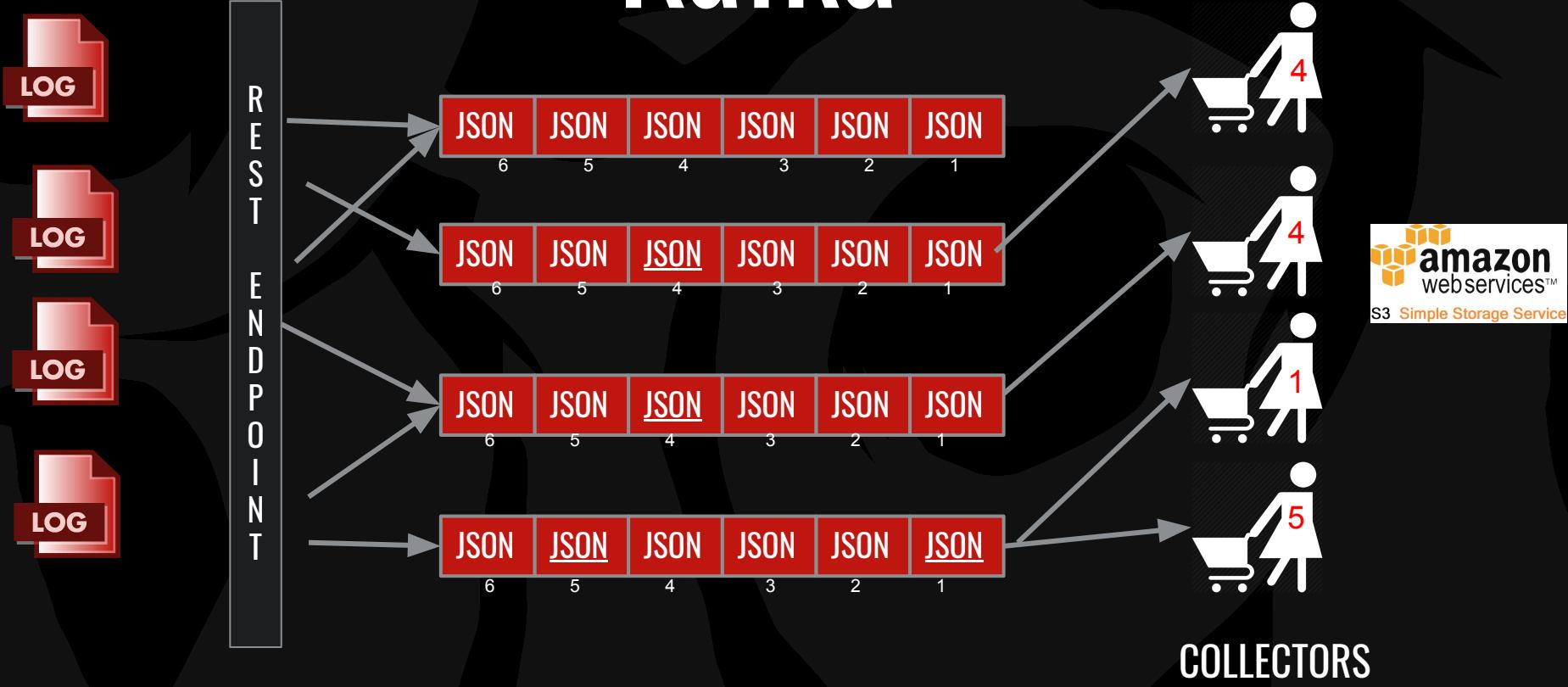
Has a DB Commit Log
(ooh revolutionary)



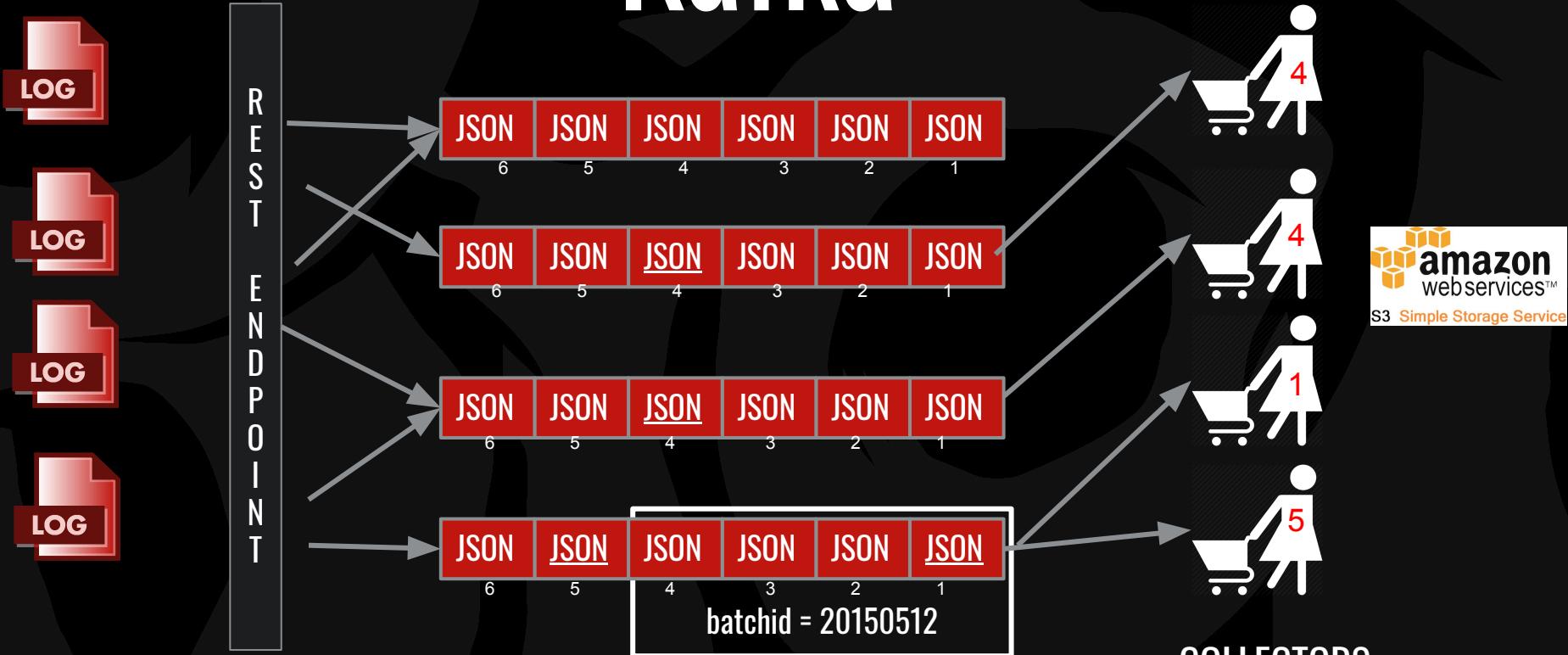
Kafka



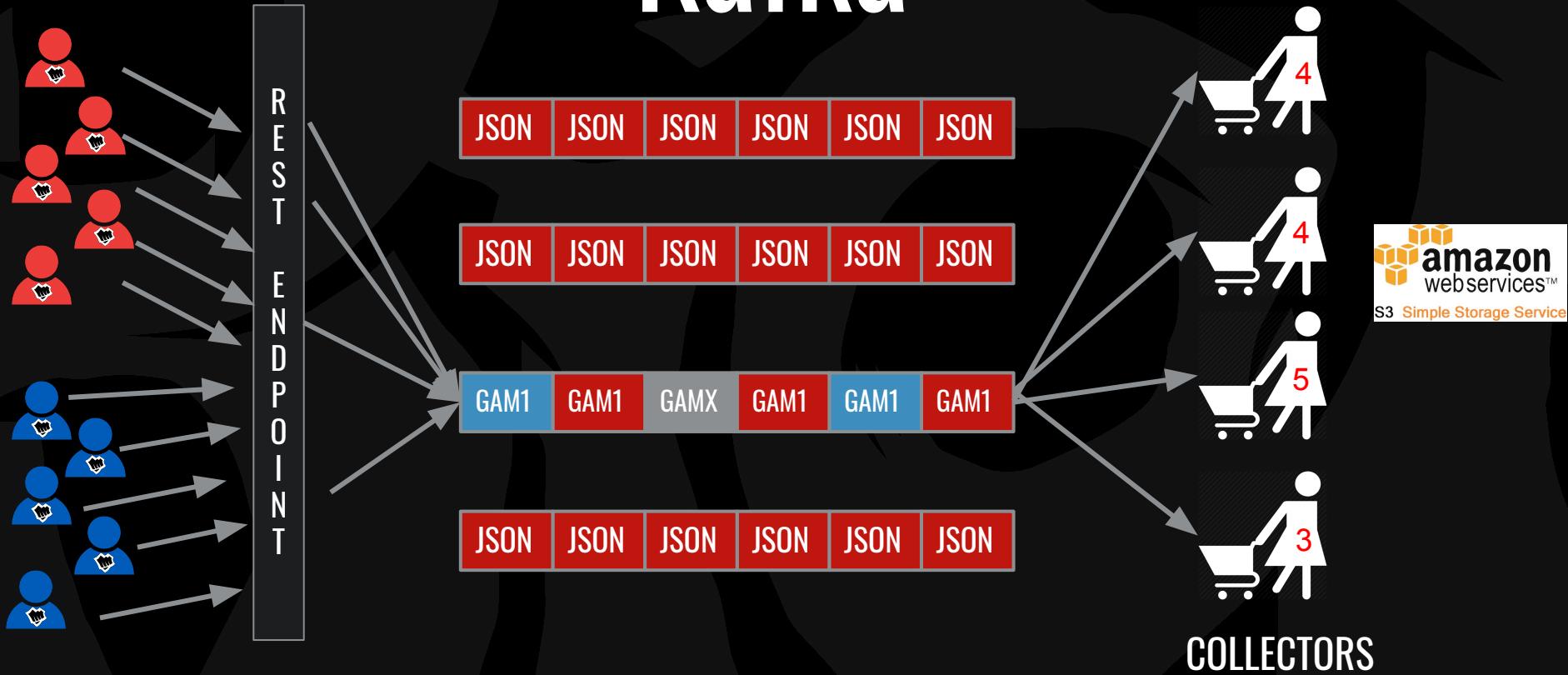
Kafka



Kafka



Kafka



Idempotency

Idempotent - an operation that will produce the same results if executed once or multiple times

EXAMPLE:

Non-Idempotent:

- $x = x * 5;$

- Submitting a purchase

Idempotent:

- $\text{abs}(\text{abs}(x)) = \text{abs}(x)$

- Cancelling a purchase

Idempotent?

In the transactional OLTP world....

```
INSERT INTO games_played
(SELECT * FROM games_played_na
WHERE date >= '2015-10-25')
```

Idempotent?

In the big data / OLAP world....

```
INSERT INTO games_played
(SELECT * FROM games_played_na
WHERE date >= '2015-10-25')
```

Idempotency

Use application logic to make **idempotent**

```
msg = queue.pop;  
if (processed_games.contains( msg.game_id )  
{  
    return; //do nothing  
else {  
    process_game(msg);  
}
```

INGESTION

PULL-BASED / ETL



FuETL

- OLTP game data
- External Data Sources

PUSH-BASED



kafka

- Anything pushed to it
- Server logs

STORAGE

MASTER WAREHOUSE



DATA AUDITING



QUERY / VIEWS

AGGREGATE QUERIES



BATCH QUERIES



SINGLE-ROW QUERIES



VIZ. TOOLS



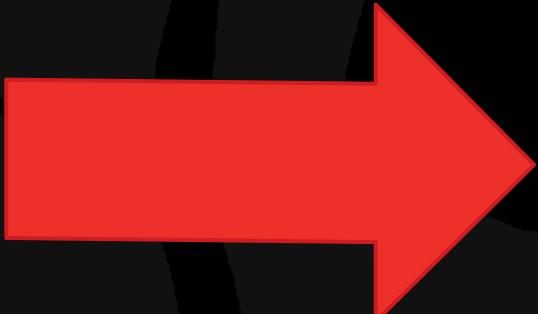
AMAZON S3 STRUCTURE

HIVE

- schema1
 - table1
 - env
 - dt
 - time
 - table2
 - table3
- schema2
 - table1
 - ...
- schema3
- schema4

AMAZON S3

- s3n://datawarehouse/
 - schema1/
 - table1/
 - env/
 - dt/
 - time/
 - table2/
 - table3/
 - schema2/
- s3n://telemetrydata/
 - application1/
 - table1/
 - env/
 - dt/
 - table2/
 - application2/



[Upload](#)[Create Folder](#)[Actions ▾](#)[Versions:](#)[Hide](#)[Show](#)[All Buckets /](#)[merged / audit_event_queue_dodge](#)

	Name
<input type="checkbox"/>	env=BR1
<input type="checkbox"/>	env=BR1_\$folder\$
<input type="checkbox"/>	env=EUN1
<input type="checkbox"/>	env=EUN1_\$folder\$
<input type="checkbox"/>	env=EUW1
<input type="checkbox"/>	env=EUW1_\$folder\$
<input type="checkbox"/>	env=ID1
<input type="checkbox"/>	env=ID1_\$folder\$
<input type="checkbox"/>	env=KR1
<input type="checkbox"/>	env=KR1_\$folder\$
<input type="checkbox"/>	env=LA1
<input type="checkbox"/>	env=LA1_\$folder\$
<input type="checkbox"/>	env=LA2
<input type="checkbox"/>	env=LA2_\$folder\$
<input type="checkbox"/>	env=NA1
<input type="checkbox"/>	env=NA1_\$folder\$
<input type="checkbox"/>	env=OC1
<input type="checkbox"/>	env=OC1_\$folder\$

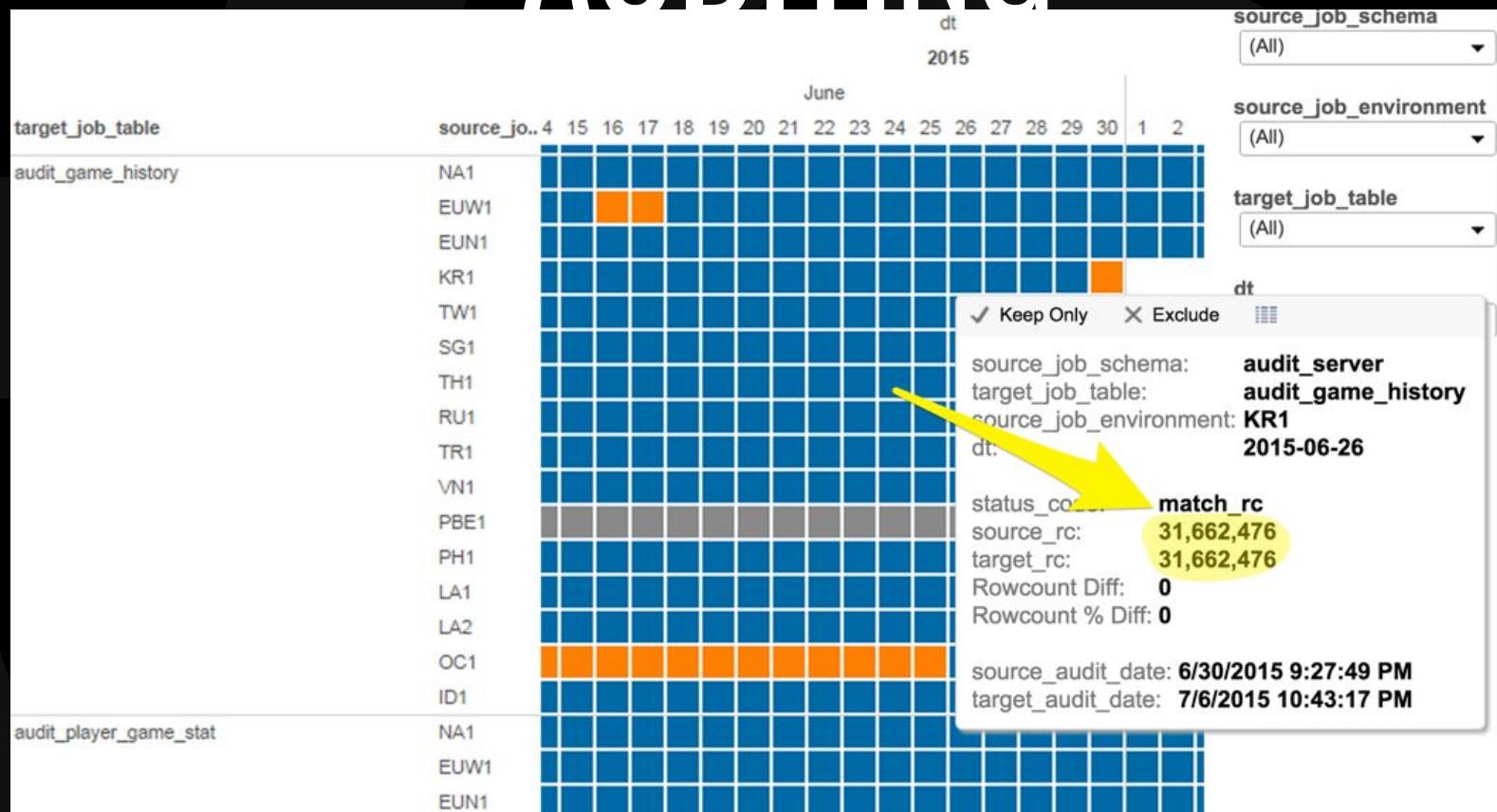


Upload Create Folder Actions ▾ Versions: Hide Show None Properties Transfers

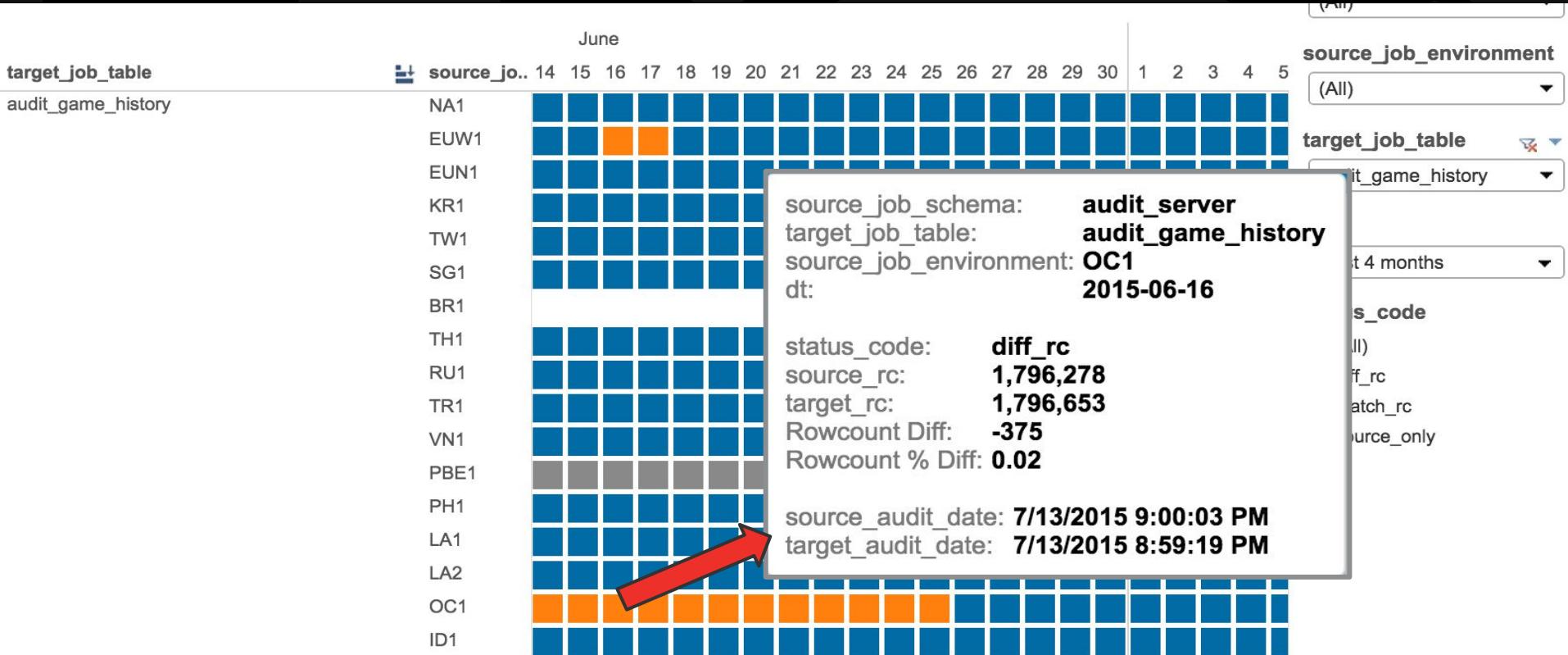
All Buckets / / merged / audit_event_queue_dodge / env=EUW1 / dt=2015-10-27

	Name	Storage Class	Size	Last Modified
<input type="checkbox"/> 	abf621d7-8f0d-43b7-93d4-f0f7afaaea7f-000000	Standard	26.7 MB	Wed Oct 28 15:04:13 GMT-700 2015

AUDITING



AUDITING



INGESTION

PULL-BASED / ETL



FuETL

- OLTP game data
- External Data Sources

PUSH-BASED



kafka

- Anything pushed to it
- Server logs

STORAGE

MASTER WAREHOUSE



S3 Simple Storage Service

DATA AUDITING



QUERY / VIEWS

AGGREGATE QUERIES



BATCH QUERIES



SINGLE-ROW QUERIES



VIZ. TOOLS





Warehouse
Auditing
Service
Platform

REST micro-service built with Java and docker.

Source and target comparison.

Reports and visualizations we can use to find problems.

INGESTION

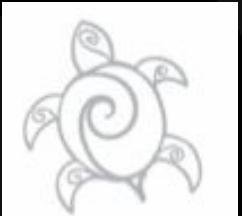
PULL-BASED / ETL



FuETL

- OLTP game data
- External Data Sources

PUSH-BASED



HONU

- Anything pushed to it
- Server logs

STORAGE

MASTER WAREHOUSE



DATA AUDITING



QUERY / VIEWS

AGGREGATE QUERIES



BATCH QUERIES



SINGLE-ROW QUERIES



VIZ. TOOLS





row_id	timestamp	col_a	col_b	col_c
1	2023-01-01T00:00:00Z	Value A1	Value B1	Value C1
2	2023-01-02T00:00:00Z	Value A2	Value B2	Value C2
3	2023-01-03T00:00:00Z	Value A3	Value B3	Value C3
4	2023-01-04T00:00:00Z	Value A4	Value B4	Value C4
5	2023-01-05T00:00:00Z	Value A5	Value B5	Value C5
6	2023-01-06T00:00:00Z	Value A6	Value B6	Value C6
7	2023-01-07T00:00:00Z	Value A7	Value B7	Value C7
8	2023-01-08T00:00:00Z	Value A8	Value B8	Value C8
9	2023-01-09T00:00:00Z	Value A9	Value B9	Value C9
10	2023-01-10T00:00:00Z	Value A10	Value B10	Value C10
11	2023-01-11T00:00:00Z	Value A11	Value B11	Value C11
12	2023-01-12T00:00:00Z	Value A12	Value B12	Value C12
13	2023-01-13T00:00:00Z	Value A13	Value B13	Value C13
14	2023-01-14T00:00:00Z	Value A14	Value B14	Value C14
15	2023-01-15T00:00:00Z	Value A15	Value B15	Value C15
16	2023-01-16T00:00:00Z	Value A16	Value B16	Value C16
17	2023-01-17T00:00:00Z	Value A17	Value B17	Value C17
18	2023-01-18T00:00:00Z	Value A18	Value B18	Value C18
19	2023-01-19T00:00:00Z	Value A19	Value B19	Value C19
20	2023-01-20T00:00:00Z	Value A20	Value B20	Value C20
21	2023-01-21T00:00:00Z	Value A21	Value B21	Value C21
22	2023-01-22T00:00:00Z	Value A22	Value B22	Value C22
23	2023-01-23T00:00:00Z	Value A23	Value B23	Value C23
24	2023-01-24T00:00:00Z	Value A24	Value B24	Value C24
25	2023-01-25T00:00:00Z	Value A25	Value B25	Value C25
26	2023-01-26T00:00:00Z	Value A26	Value B26	Value C26
27	2023-01-27T00:00:00Z	Value A27	Value B27	Value C27
28	2023-01-28T00:00:00Z	Value A28	Value B28	Value C28
29	2023-01-29T00:00:00Z	Value A29	Value B29	Value C29
30	2023-01-30T00:00:00Z	Value A30	Value B30	Value C30
31	2023-01-31T00:00:00Z	Value A31	Value B31	Value C31

row_id	timestamp	col_a	col_b	col_c
1	2023-01-01 08:00:00	100	200	300
2	2023-01-01 08:05:00	105	205	305
3	2023-01-01 08:10:00	110	210	310
4	2023-01-01 08:15:00	115	215	315
5	2023-01-01 08:20:00	120	220	320
6	2023-01-01 08:25:00	125	225	325
7	2023-01-01 08:30:00	130	230	330
8	2023-01-01 08:35:00	135	235	335
9	2023-01-01 08:40:00	140	240	340
10	2023-01-01 08:45:00	145	245	345
11	2023-01-01 08:50:00	150	250	350
12	2023-01-01 08:55:00	155	255	355
13	2023-01-01 09:00:00	160	260	360
14	2023-01-01 09:05:00	165	265	365
15	2023-01-01 09:10:00	170	270	370
16	2023-01-01 09:15:00	175	275	375
17	2023-01-01 09:20:00	180	280	380
18	2023-01-01 09:25:00	185	285	385
19	2023-01-01 09:30:00	190	290	390
20	2023-01-01 09:35:00	195	295	395
21	2023-01-01 09:40:00	200	300	400
22	2023-01-01 09:45:00	205	305	405
23	2023-01-01 09:50:00	210	310	410
24	2023-01-01 09:55:00	215	315	415
25	2023-01-01 10:00:00	220	320	420
26	2023-01-01 10:05:00	225	325	425
27	2023-01-01 10:10:00	230	330	430
28	2023-01-01 10:15:00	235	335	435
29	2023-01-01 10:20:00	240	340	440
30	2023-01-01 10:25:00	245	345	445
31	2023-01-01 10:30:00	250	350	450
32	2023-01-01 10:35:00	255	355	455
33	2023-01-01 10:40:00	260	360	460
34	2023-01-01 10:45:00	265	365	465
35	2023-01-01 10:50:00	270	370	470
36	2023-01-01 10:55:00	275	375	475
37	2023-01-01 11:00:00	280	380	480
38	2023-01-01 11:05:00	285	385	485
39	2023-01-01 11:10:00	290	390	490
40	2023-01-01 11:15:00	295	395	495
41	2023-01-01 11:20:00	300	400	500
42	2023-01-01 11:25:00	305	405	505
43	2023-01-01 11:30:00	310	410	510
44	2023-01-01 11:35:00	315	415	515
45	2023-01-01 11:40:00	320	420	520
46	2023-01-01 11:45:00	325	425	525
47	2023-01-01 11:50:00	330	430	530
48	2023-01-01 11:55:00	335	435	535
49	2023-01-01 12:00:00	340	440	540
50	2023-01-01 12:05:00	345	445	545
51	2023-01-01 12:10:00	350	450	550
52	2023-01-01 12:15:00	355	455	555
53	2023-01-01 12:20:00	360	460	560
54	2023-01-01 12:25:00	365	465	565
55	2023-01-01 12:30:00	370	470	570
56	2023-01-01 12:35:00	375	475	575
57	2023-01-01 12:40:00	380	480	580
58	2023-01-01 12:45:00	385	485	585
59	2023-01-01 12:50:00	390	490	590
60	2023-01-01 12:55:00	395	495	595
61	2023-01-01 13:00:00	400	500	600
62	2023-01-01 13:05:00	405	505	605
63	2023-01-01 13:10:00	410	510	610
64	2023-01-01 13:15:00	415	515	615
65	2023-01-01 13:20:00	420	520	620
66	2023-01-01 13:25:00	425	525	625
67	2023-01-01 13:30:00	430	530	630
68	2023-01-01 13:35:00	435	535	635
69	2023-01-01 13:40:00	440	540	640
70	2023-01-01 13:45:00	445	545	645
71	2023-01-01 13:50:00	450	550	650
72	2023-01-01 13:55:00	455	555	655
73	2023-01-01 14:00:00	460	560	660
74	2023-01-01 14:05:00	465	565	665
75	2023-01-01 14:10:00	470	570	670
76	2023-01-01 14:15:00	475	575	675
77	2023-01-01 14:20:00	480	580	680
78	2023-01-01 14:25:00	485	585	685
79	2023-01-01 14:30:00	490	590	690
80	2023-01-01 14:35:00	495	595	695
81	2023-01-01 14:40:00	500	600	700
82	2023-01-01 14:45:00	505	605	705
83	2023-01-01 14:50:00	510	610	710
84	2023-01-01 14:55:00	515	615	715
85	2023-01-01 15:00:00	520	620	720
86	2023-01-01 15:05:00	525	625	725
87	2023-01-01 15:10:00	530	630	730
88	2023-01-01 15:15:00	535	635	735
89	2023-01-01 15:20:00	540	640	740
90	2023-01-01 15:25:00	545	645	745
91	2023-01-01 15:30:00	550	650	750
92	2023-01-01 15:35:00	555	655	755
93	2023-01-01 15:40:00	560	660	760
94	2023-01-01 15:45:00	565	665	765
95	2023-01-01 15:50:00	570	670	770
96	2023-01-01 15:55:00	575	675	775
97	2023-01-01 16:00:00	580	680	780
98	2023-01-01 16:05:00	585	685	785
99	2023-01-01 16:10:00	590	690	790
100	2023-01-01 16:15:00	595	695	795

BATCH

OLAP

POINT

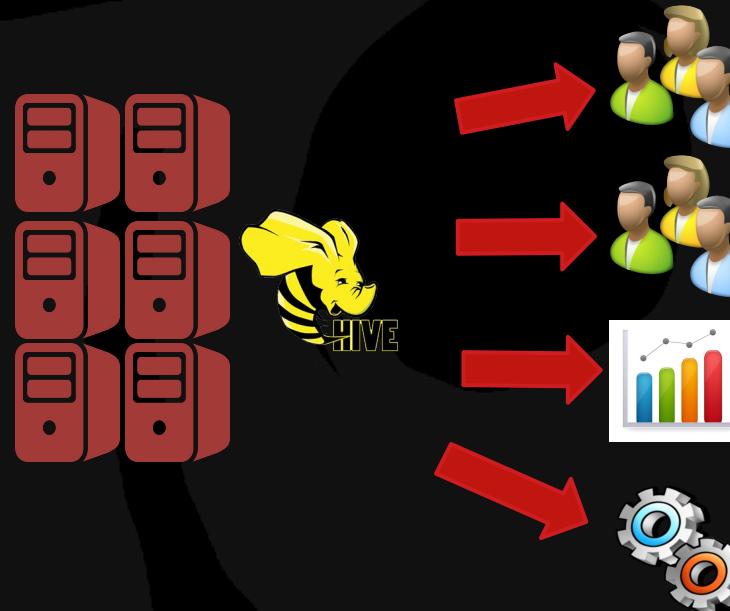
SCALING IN AWS



SCALING

RESOURCE CONTENTION

Hive .08 pre YARN, immature resource scheduling



AWS Infrastructure Today

EMR

EC2

Storage

Networking



Data Science



Analytics /
Hue



ETL



Telemetry



DynamoDB
Loading



Platfora



Solr (real
time)



Auditing



ETL



Telemetry
collectors



Rocana
(real time
dashboard)



Data
dictionary



Point Data
Service

RDS



Metastore



ETL App DB

Data Science

Fraud

VPC

AWS Direct
Connect

AWS Direct
Connect

AWS Direct
Connect

AWS Direct
Connect

DYNAMODB



Point Data
Store

S3



Source of "Truth"

CONCLUSION



SEAN'S PRO TIPS OF THE DAY

DO

- Keep idempotency in mind and use MQ architecture
- Get an auditing solution for DW accuracy
- Prepare for multiple data access patterns
- Allocate time for tuning AWS infrastructure

DON'T

- Don't underestimate simple problems in big data.
- Don't forget to track cost. AWS bills can surprise you
- Don't wait. Create S3 permissions and naming standards early
- Don't stop. Believing

QUESTIONS?

ENGINEERING BLOG
<http://engineering.riotgames.com>

CAREERS
<http://www.riotgames.com/careers>

SEAN MALONEY

 SMALONEY @
riotgames.com

 @SEAN_SEANNERY