MONITORING 101: POSTGRESQL

JASON YEE, DATADOG

@gitbisect

DATADOG

@gitbisect
Technical Writer/Evangelist
“Docs & Talks”
Travel Hacker & Whiskey Hunter

DATADOG

@datadoghq
SaaS-based monitoring
Trillions of data points per day
http://jobs.datadoghq.com

Honest Status Page @honest_update · Jul 22

Our new monitoring product just watches Twitter and IRC for our name + "down".

COLLECTING DATA IS CHEAP: NOT HAVING IT WHEN YOU NEED IT CAN BE EXPENSIVE

SO INSTRUMENT ALL THE THINGS!

British Airways Union Blames Massive IT Failure On Outsourcing IT Jobs To India

The carrier cancelled hundreds of flights from London yesterday.

28/05/2017 12:57 PM IST | Updated 28/05/2017 12:59 PM IST

ANI

WAT?!

NEIL HALL / REUTERS

LONDON -- British Airways GMB union has blamed the airline's 2016 decision of outsourcing IT jobs to India as the reason behind cancelling all Saturday flights from

4 QUALITIES OF GOOD METRICS

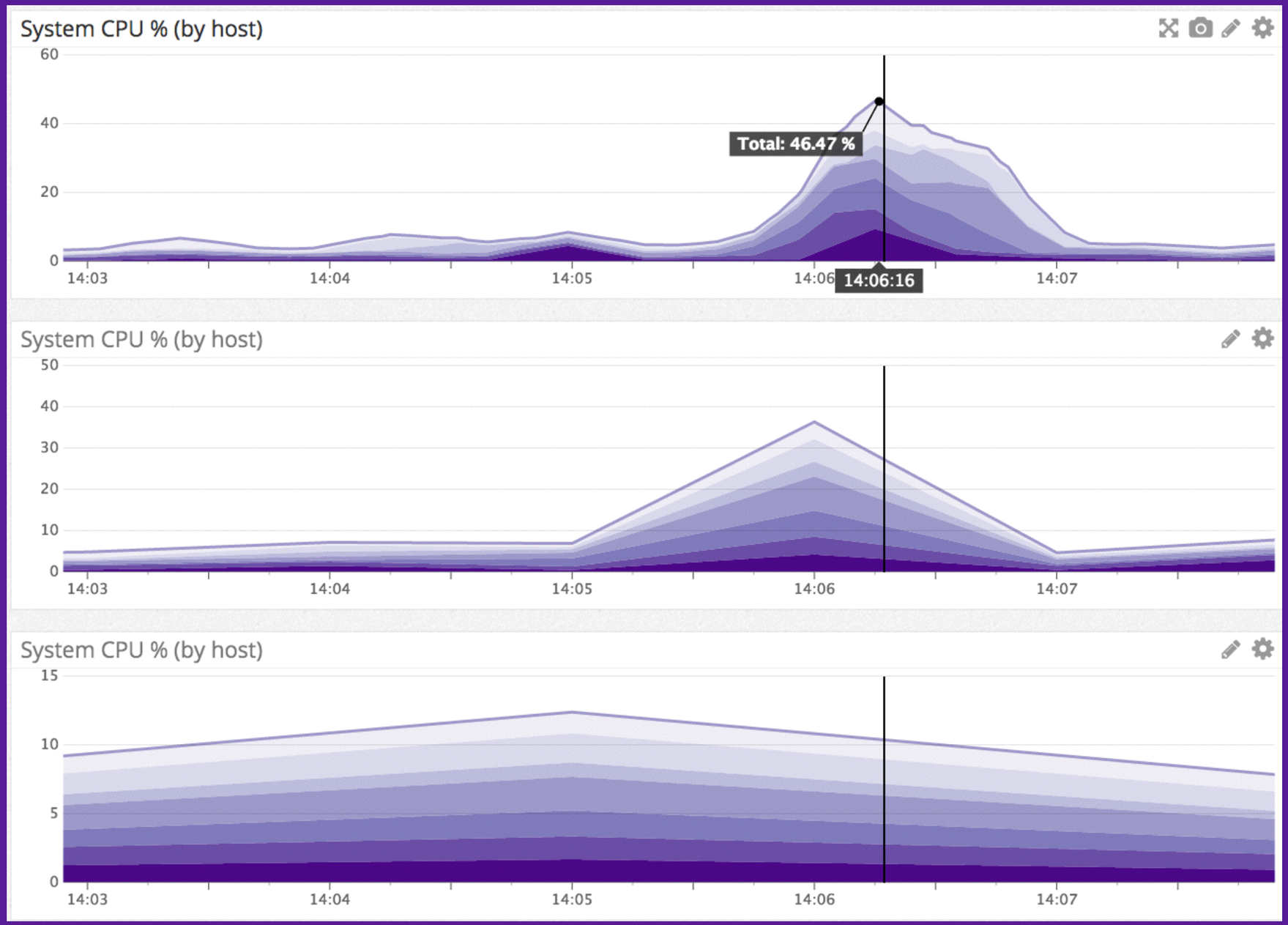
NOT ALL METRICS ARE EQUAL

1. MUST BE WELL UNDERSTOOD

2. SUFFICIENT GRANULARITY

TW: @gitbisect @datadoghq

RANK	PARTICIPANT	RESULT
G	Anthony ERVIN USA	21.40
S	Florent MANAUDOU FRA	21.41
B	Nathan ADRIAN USA	21.49
4.	Ben PROUD GBR	21.68
5.	Andrii GOVOROV UKR	21.74
6.	Bruno FRATUS BRA	21.79
6.	Bradley Edward TANDY RSA	21.79
8.	Simonas BILIS LTU	22.08



1 second
Peak 46%

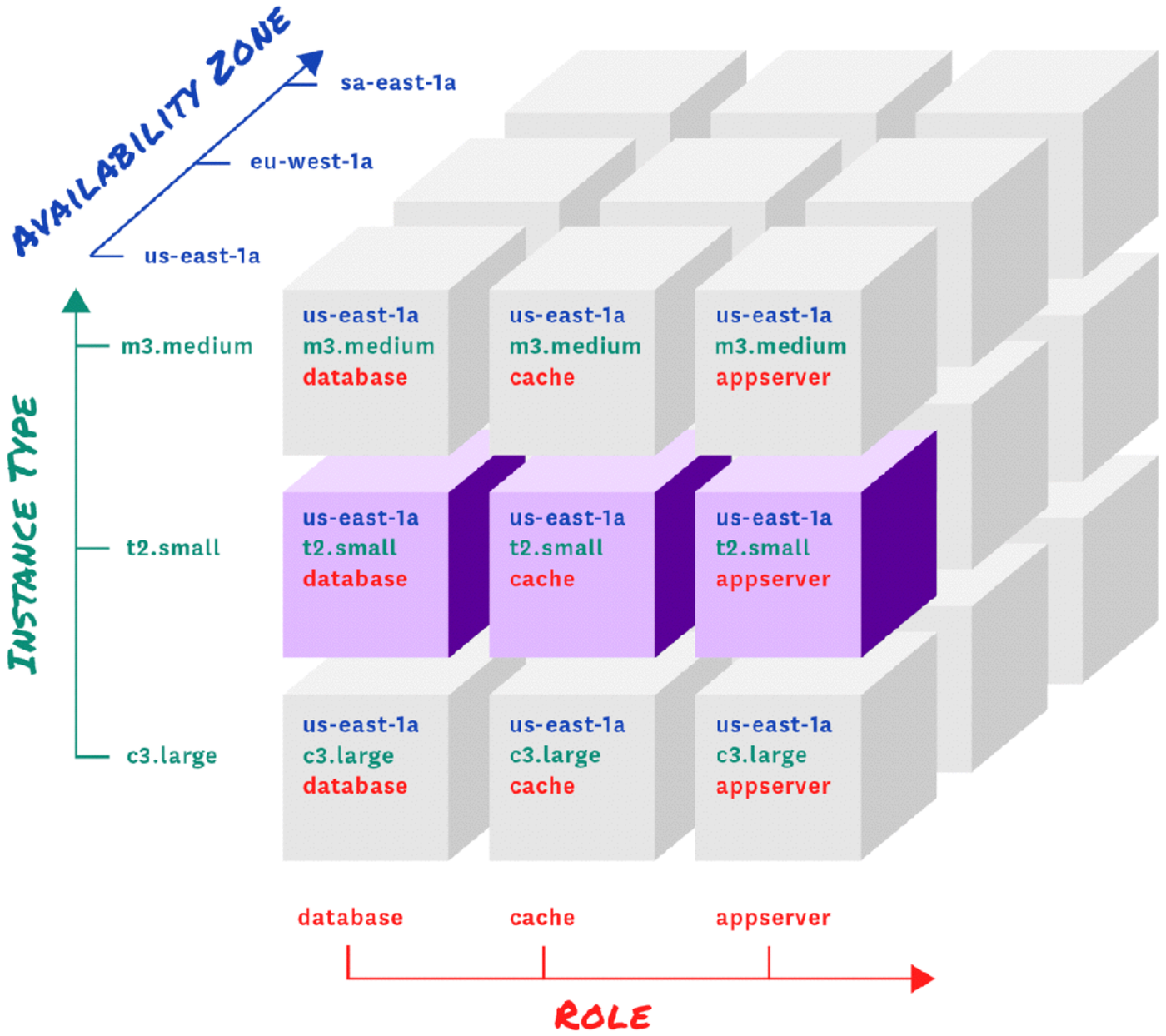
1 minute
Peak 36%

5 minutes
Peak 12%

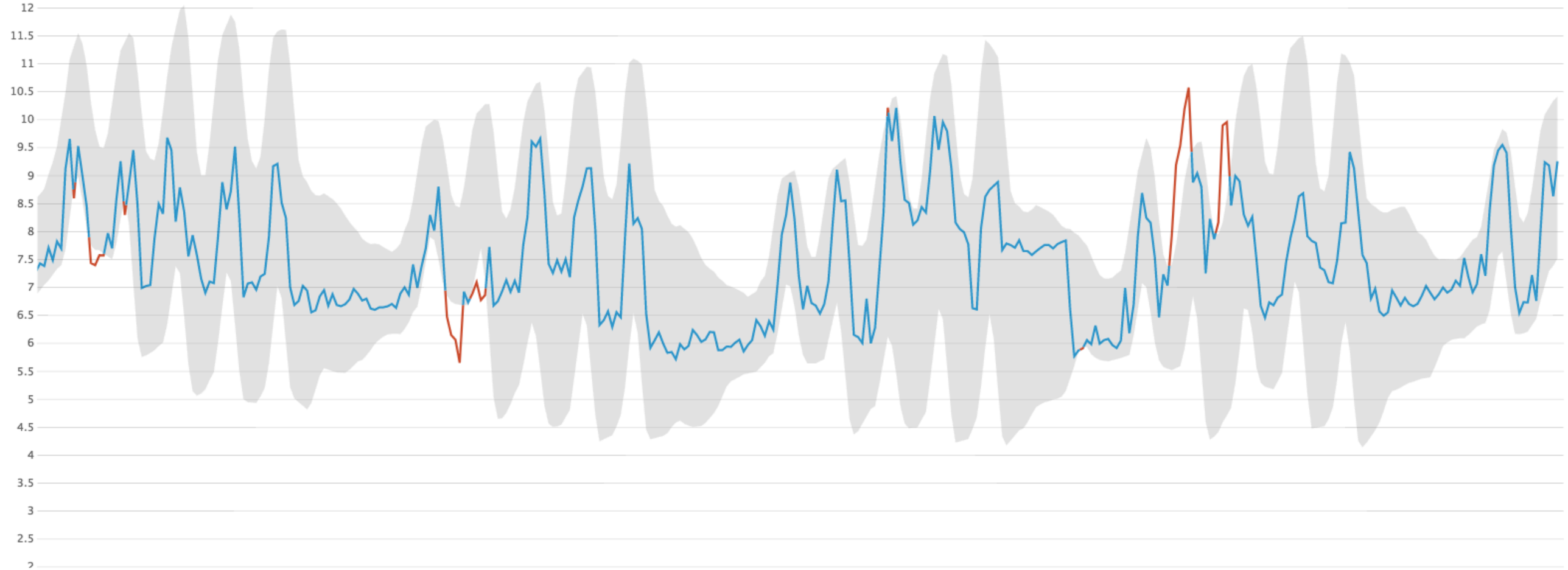
DATADOG

3. TAGGED & FILTERABLE

TW: @gitbisect @datadoghq


AVAILABILITY ZONE
sa-east-1a
eu-west-1a
us-east-1a
INSTANCE TYPE
m3.medium
t2.small
c3.large
us-east-1a m3.medium database
us-east-1a m3.medium cache
us-east-1a m3.medium appserver
us-east-1a t2.small database
us-east-1a t2.small cache
us-east-1a t2.small appserver
us-east-1a c3.large database
us-east-1a c3.large cache
us-east-1a c3.large appserver
database
cache
appserver
ROLE

TW: @gitbisect @datadoghq


12
11.5
11
10.5
10
9.5
9
8.5
8
7.5
7
6.5
6
5.5
5
4.5
4
3.5
3
2.5
2

WORK METRICS

RESOURCE METRICS

EVENTS


TW: @gitbisect @datadoghq
WORK METRICS
THROUGHPUT
SUCCESS
ERROR
PERFORMANCE

RESOURCE METRICS

- UTILIZATION
- SATURATION
- ERROR
- AVAILABILITY

- EVENTS
- CODE CHANGES
- ALERTS
- SCALING EVENTS
- ETC

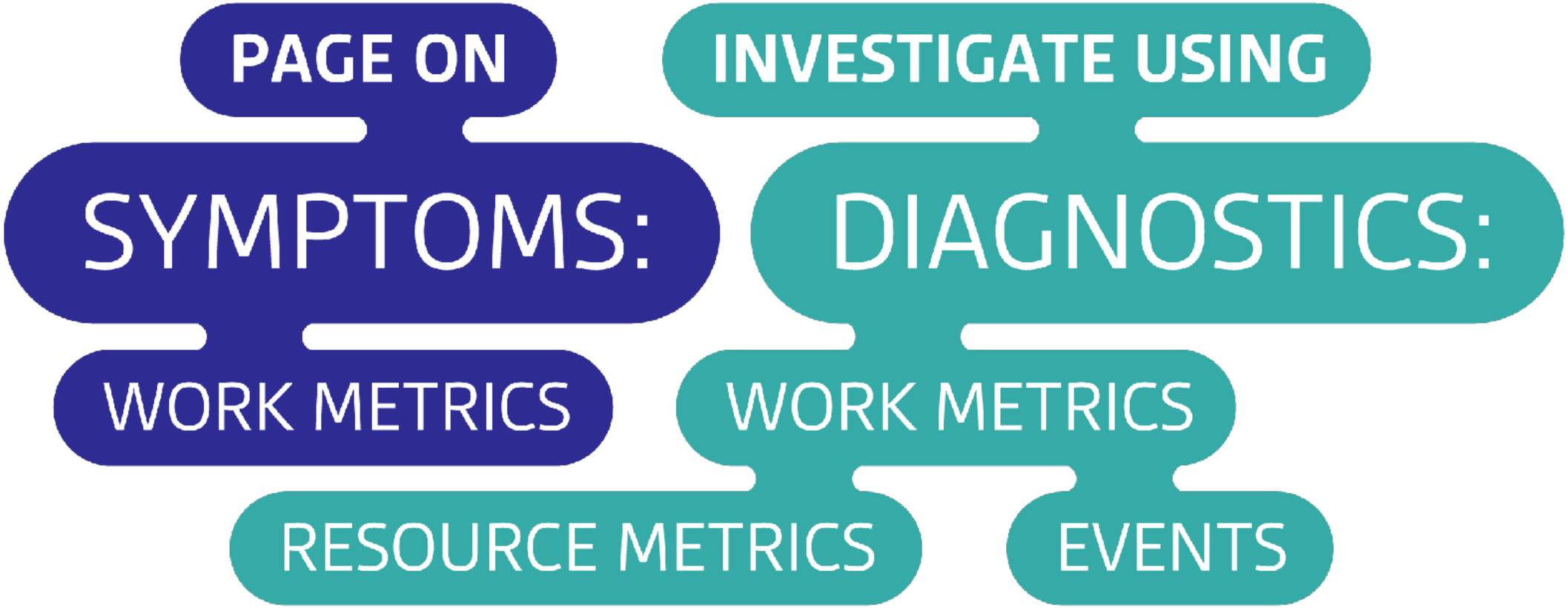
P.S. - June 1! Mark your calendar!

Adrian Cole
@adrianfcole

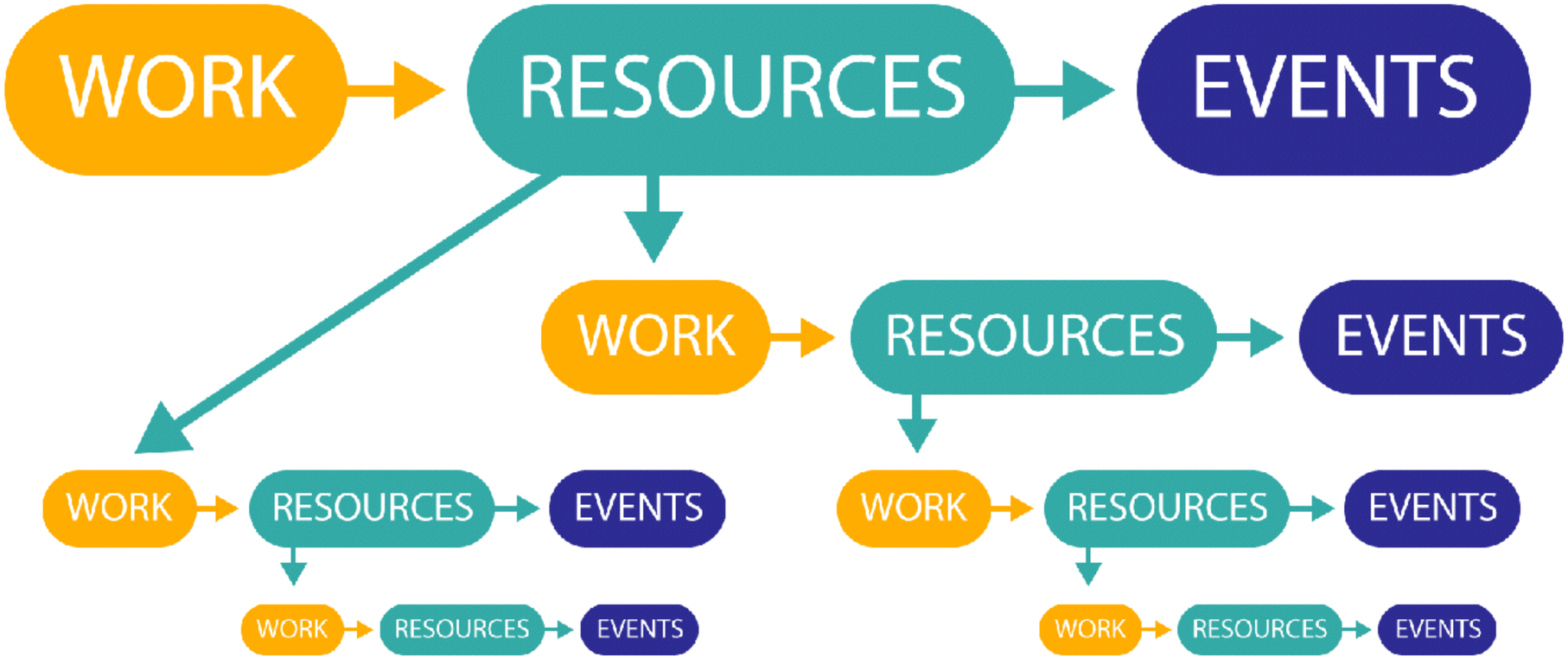
Following

Q: Are we losing money?
A: Can't answer that, but I can tell you what average CPU usage was 5ish mins ago..

WHAT TO PAGE ON?



RECURSE UNTIL YOU FIND THE TECHNICAL CAUSES



TW: @gitbisect @datadoghq

SCALING & MONITORING POSTGRESQL AT DATADOG

MOAR RESOURCES!

MOAR INSTANCES!

Writes →  Repl →  Reads →

REQUIREMENTS

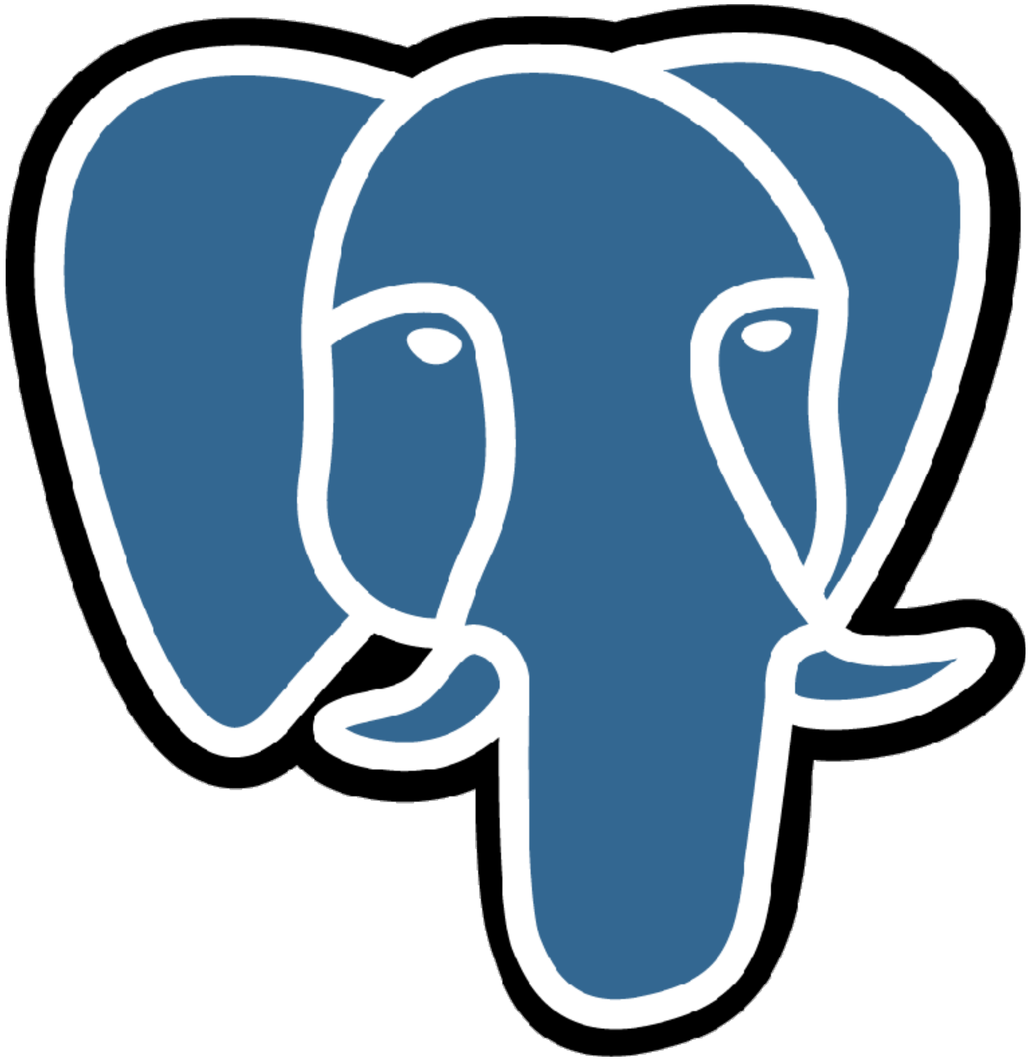
- Write master is writeable, read replicas are readable!

REQUIREMENTS

- Write master is writeable, read replicas are readable!
- Read replicas are up to date and don’t lag

REQUIREMENTS

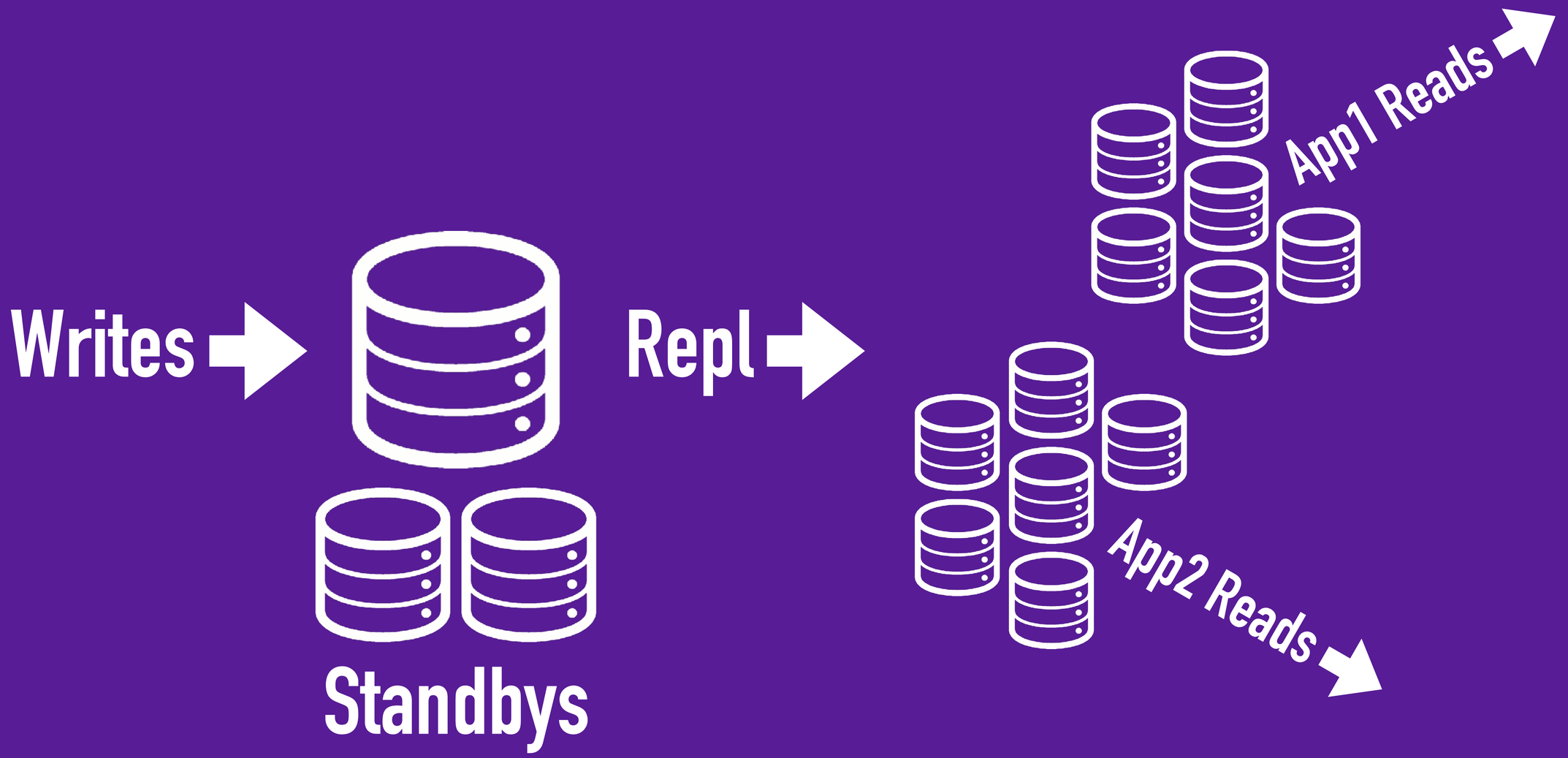
- Write master is writeable, read replicas are readable!
- Read replicas are up to date and don’t lag
- Additional read replicas can be provisioned quickly




WALL·E

SOLUTIONS

- PostgreSQL!
  - http://bit.ly/pg-repl-docs
- WAL-E
  - https://github.com/wal-e/wal-e

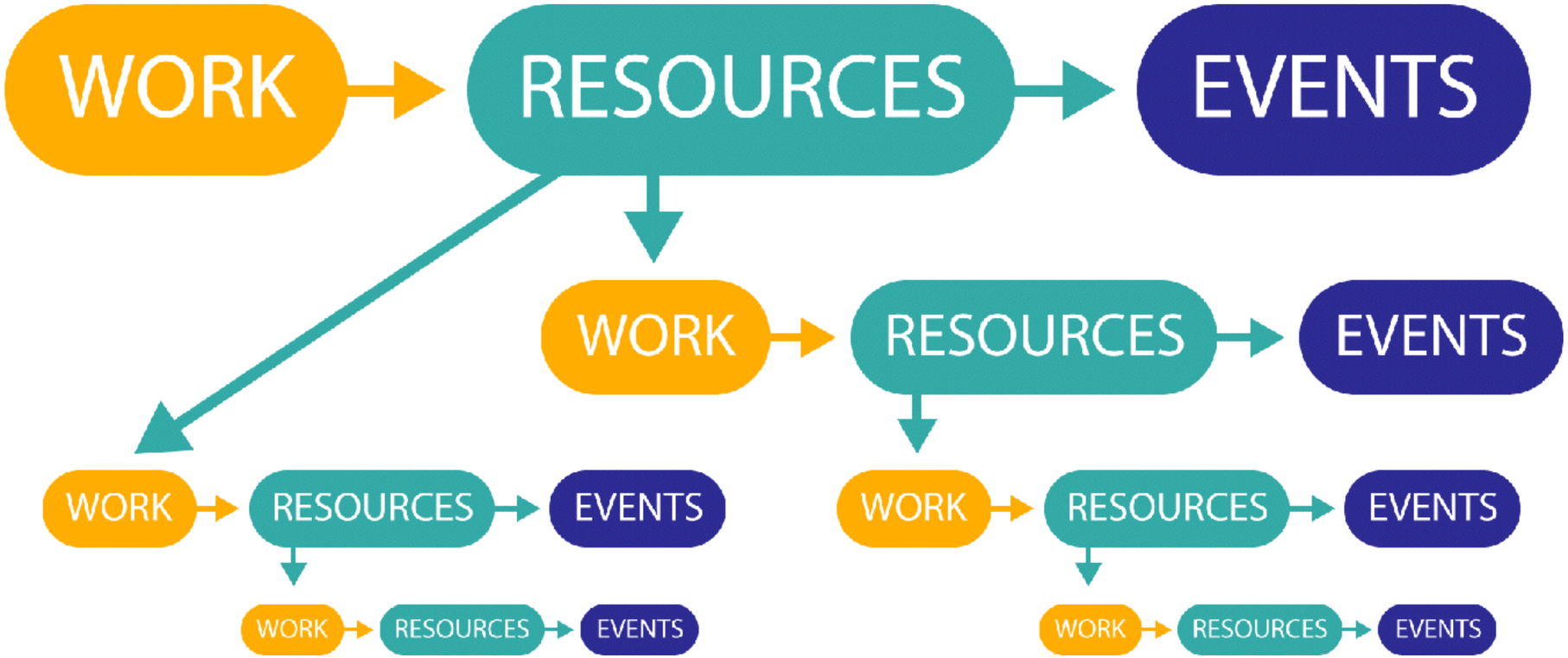

Writes
Repl
App1 Reads
App2 Reads
Standbys

WHAT DO WE MONITOR AT DATADOG?

WHAT METRICS DO WE GATHER?

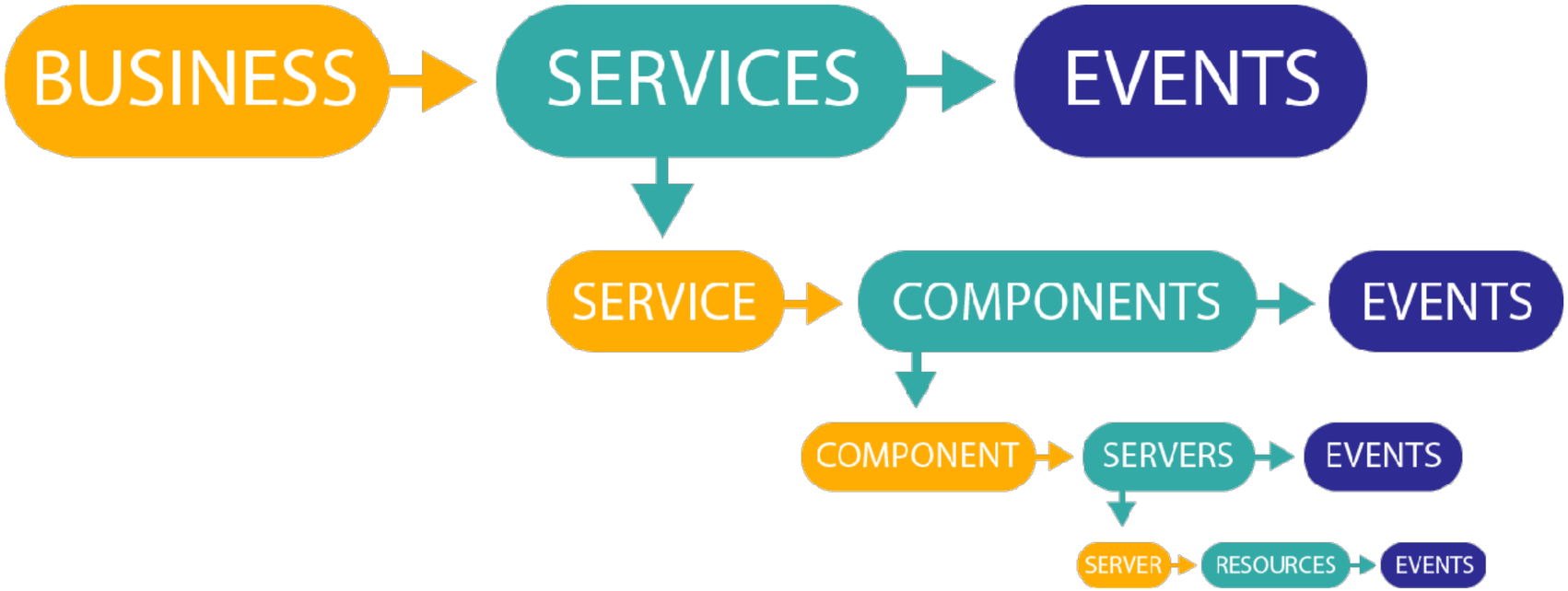
connections
commits
rollbacks
disk_read
buffer_hit
rows_returned
rows_fetched
rows_inserted
rows_updated
rows_deleted
database_size
deadlocks
temp_bytes
temp_files
bgwriter.checkpoints_timed
bgwriter.checkpoints_requested
bgwriter.buffers_checkpoint
bgwriter.buffers_clean
bgwriter.maxwritten_clean
bgwriter.buffers_backend
bgwriter.buffers_alloc
bgwriter.buffers_backend_fsync
bgwriter.write_time
bgwriter.sync_time
locks
seq_scans
seq_rows_read
index_scans
index_rows_fetched
rows_hot_updated
live_rows
dead_rows
index_rows_read
table_size
index_size
total_size
table.count
max_connections
percent_usage_connections
replication_delay
replication_delay_bytes
heap_blocks_read
heap_blocks_hit
index_blocks_read
index_blocks_hit
toast_blocks_read
toast_blocks_hit
toast_index_blocks_read
toast_index_blocks_hit

ALERT ON WORK METRICS, BUT RESOURCE METRICS BECOME WORK METRICS? ALERT ON EVERYTHING?

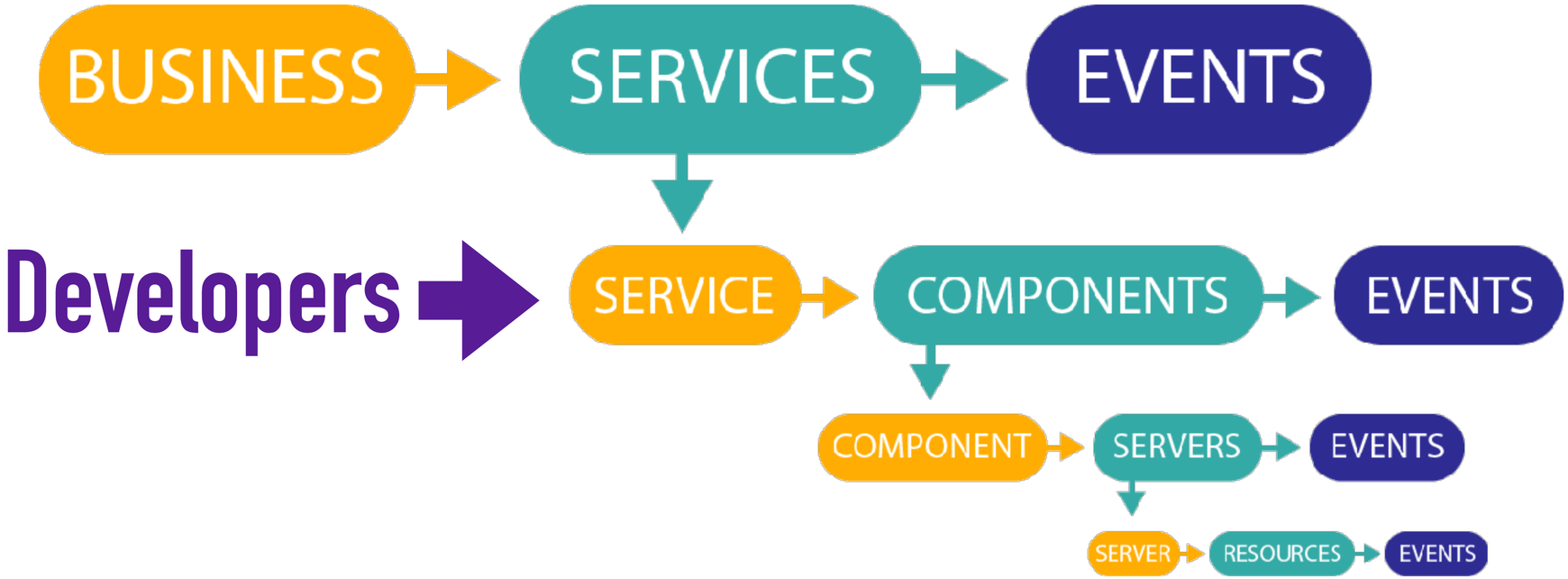


WHO TO ALERT?

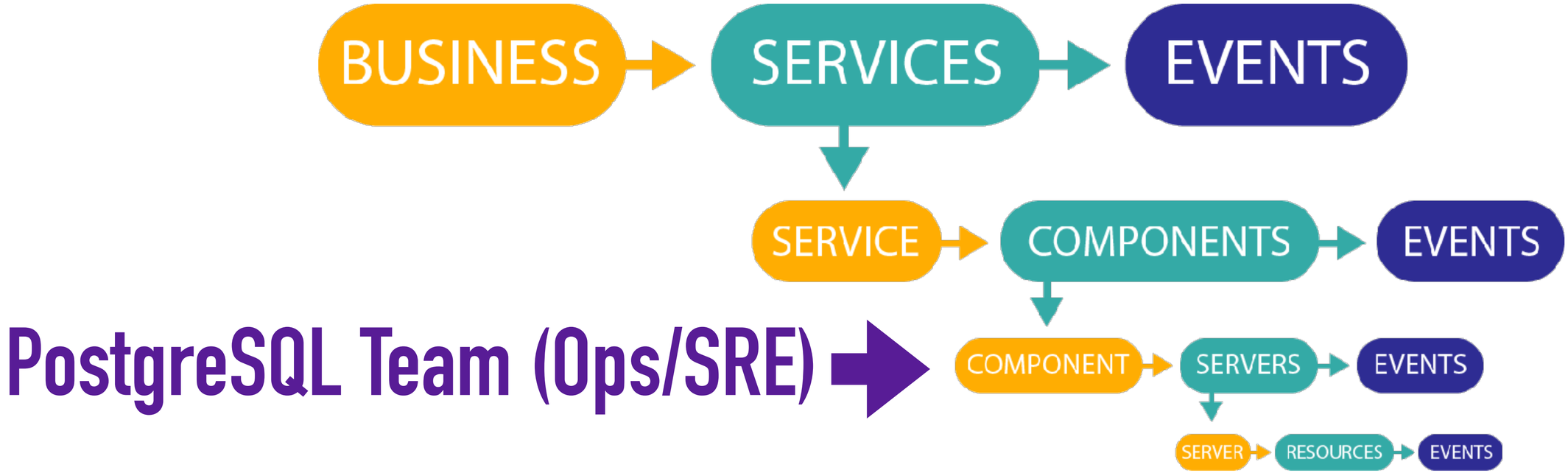
Leadership



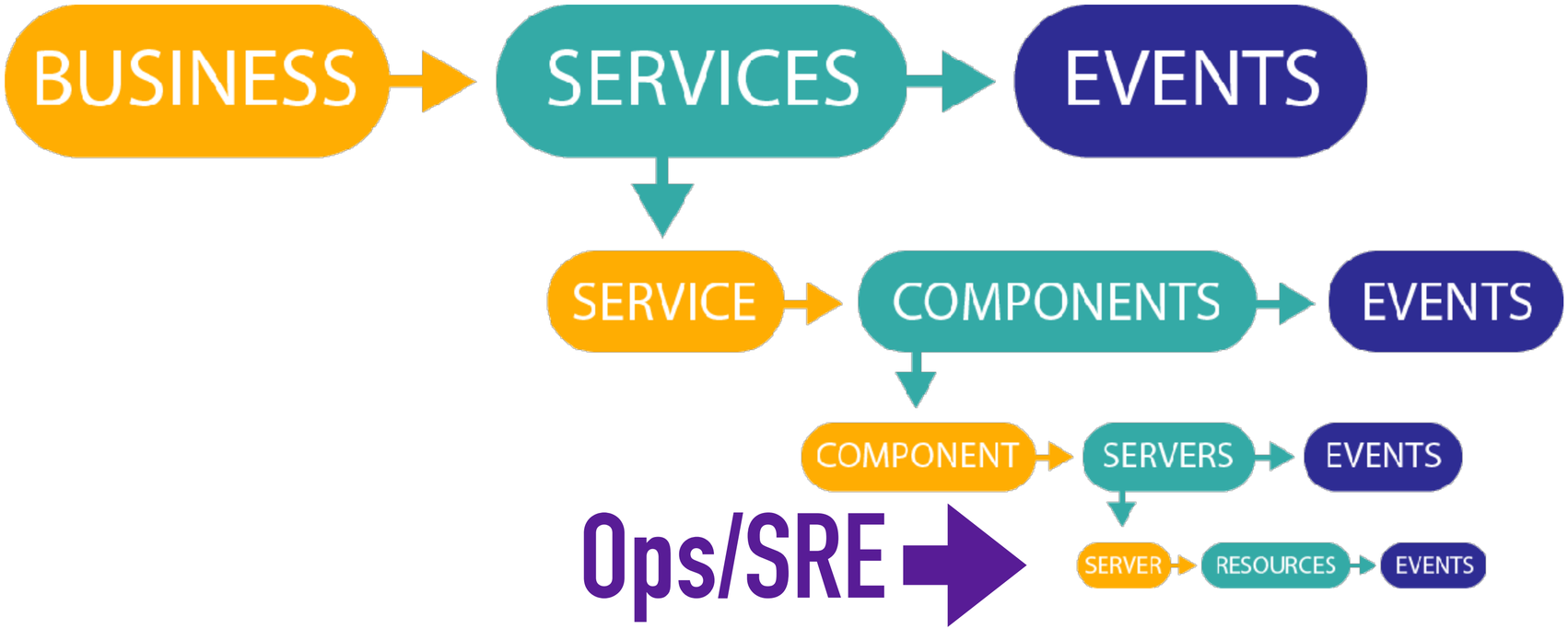
WHO TO ALERT?


BUSINESS
SERVICES
EVENTS
Developers
SERVICE
COMPONENTS
EVENTS
COMPONENT
SERVERS
EVENTS
SERVER
RESOURCES
EVENTS

WHO TO ALERT?


BUSINESS
SERVICES
EVENTS
SERVICE
COMPONENTS
EVENTS
PostgreSQL Team (Ops/SRE)
COMPONENT
SERVERS
EVENTS
SERVER
RESOURCES
EVENTS

WHO TO ALERT?


BUSINESS
SERVICES
EVENTS
SERVICE
COMPONENTS
EVENTS
COMPONENT
SERVERS
EVENTS
Ops/SRE
SERVER
RESOURCES
EVENTS

POSTGRESQL WORK METRICS (AVAILABILITY)

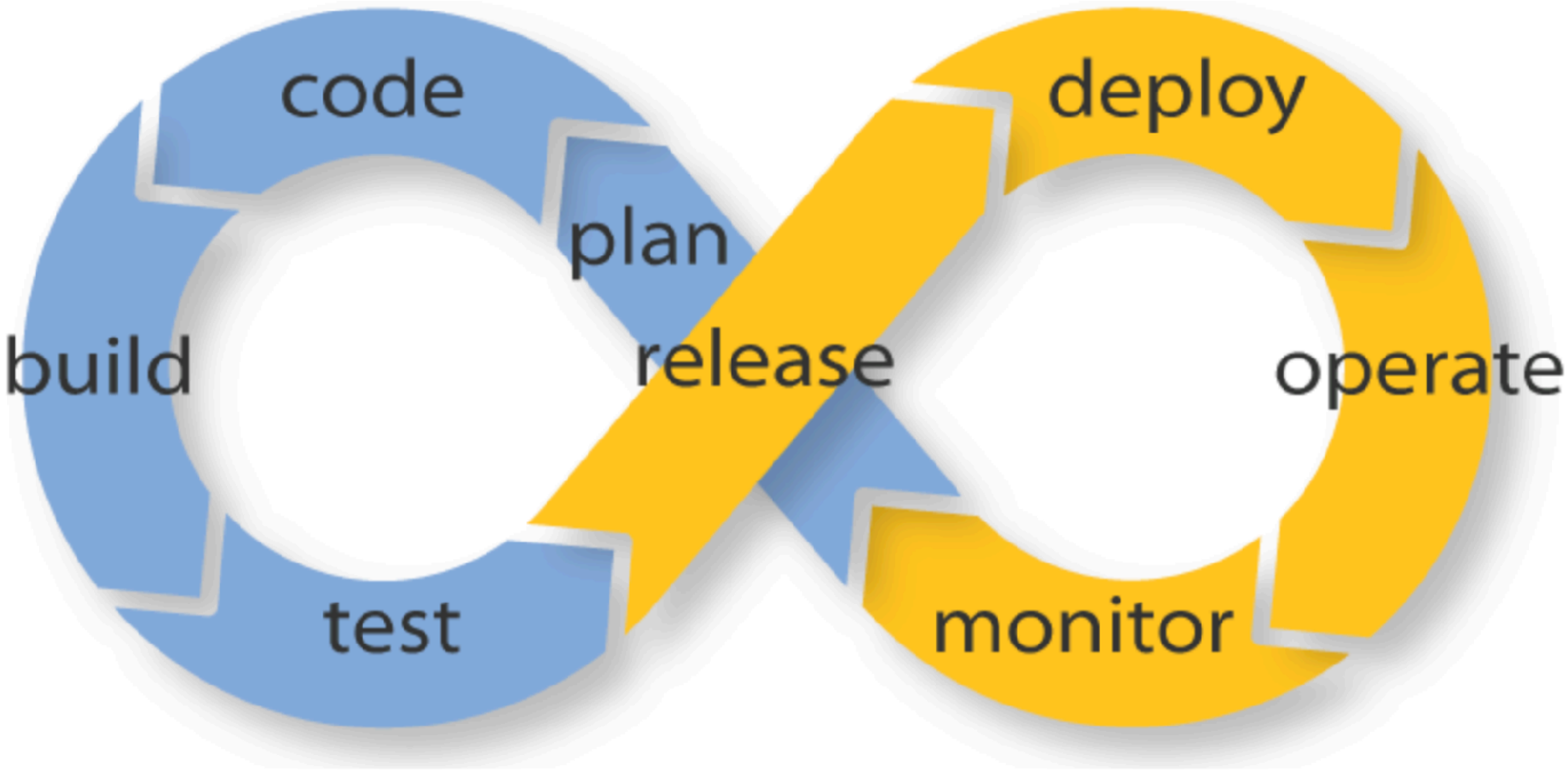
WHAT ARE WE ALERTING ON?

- Base backup is too old
- Standby is missing
- Replication lag is too high

POSTGRESQL RESOURCE METRICS = OPS WORK METRICS (CAPACITY)

WHAT ARE WE ALERTING ON?

- Connection limit
- Disk
- Memory
- CPU


code
plan
build
release
deploy
operate
test
monitor

MONITORING TO IMPROVE PERFORMANCE

WHERE TO GET THE MOST PERFORMANCE GAINS?

http://bit.ly/pg-perf-15m

1. Cut Activity
2. Slow Queries
3. Scale Stack
4. Fix Hardware
5. Postgresql.conf

CUT ACTIVITY

	Name	Hits ↓	Avg Latency	Total time
☆	SELECT key, org_id, source_typ…	6.70M	2.32 ms	4h 18min
☆	WITH sub_contexts SELECT key,…	6.30M	127 ms	9.3d
☆	WITH sub_contexts SELECT key,…	746k	38.9 ms	8h 4min
☆	WITH sub_contexts SELECT key,…	700k	15.8 ms	3h 3min
☆	WITH sub_contexts SELECT key,…	296k	130 ms	10h 38min
☆	WITH sub_contexts SELECT key,…	226k	42.6 ms	2h 40min
☆	SELECT t.oid, typarray FROM pg…	159k	23.0 ms	1h 1min

SLOW QUERIES

Name	Hits	Avg Latency ↓	Total time
WITH sub_contexts SELECT key,...	90	2.17 s	196 s
WITH sub_contexts SELECT key,...	58	1.49 s	86.7 s
WITH sub_contexts SELECT key,...	49	1.09 s	53.2 s
WITH sub_contexts SELECT key,...	194	759 ms	147 s
WITH sub_contexts SELECT key,...	45	750 ms	33.7 s
WITH sub_contexts SELECT key,...	22	740 ms	16.3 s
WITH sub_contexts SELECT key,...	70	559 ms	39.1 s

PERFORMANCE: LATENCY VS POTENTIAL

HOW DO YOU DEFINE PERFORMANCE?

```
SELECT * FROM table_x
WHERE nonindexed_col=1

SELECT * FROM table_x
JOIN table_y ON
table_x.foo=table_y.bar
WHERE table_y.indexed_col=1
```

PERFORMANCE: RAM VS DISK

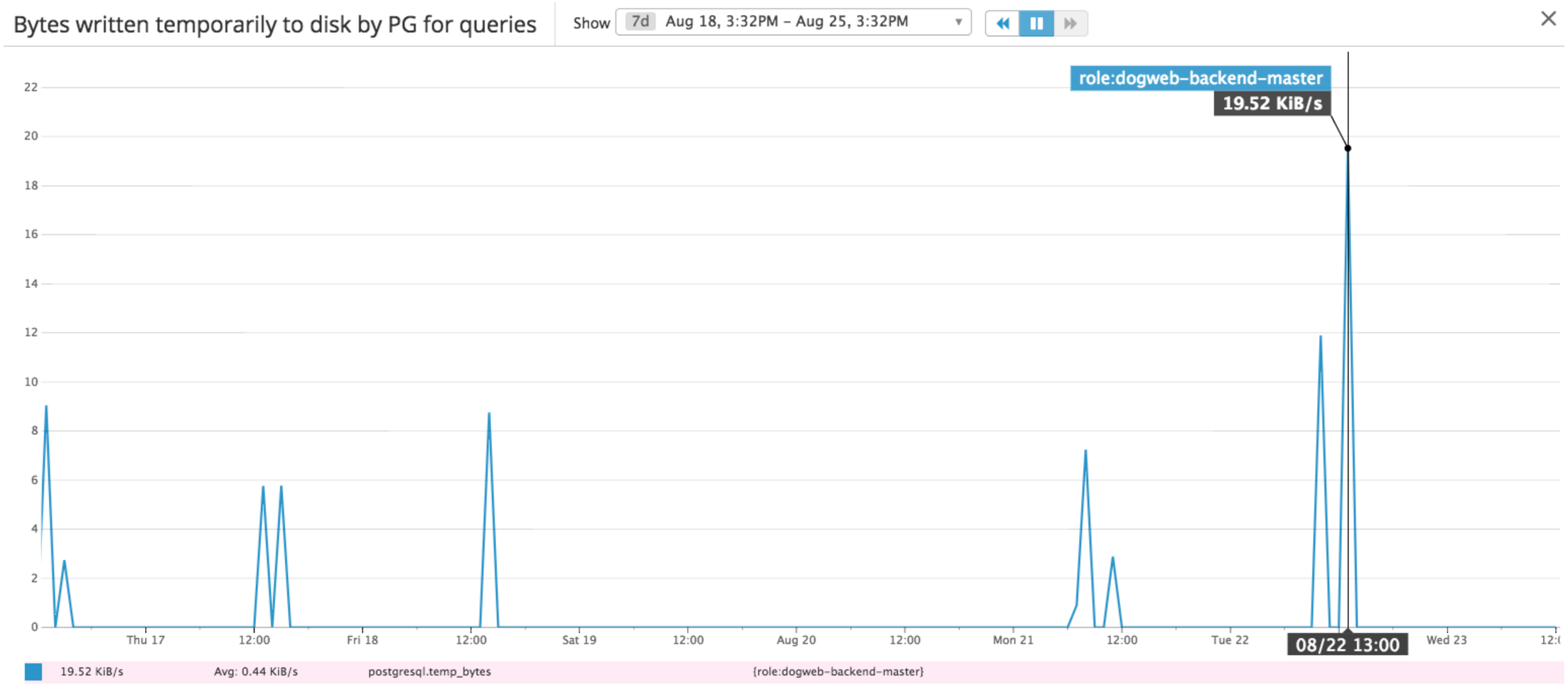
"Aside from shared_buffers, the most important memory-allocation parameter is `work_mem`... Raising this value can dramatically improve the performance of certain queries..."

ROBERT HAAS

"Aside from shared_buffers, the most important memory-allocation parameter is `work_mem`... Raising this value can dramatically improve the performance of certain queries, **but it's important not to overdo it.**"

ROBERT HAAS

FINDING **INEFFICIENT** QUERIES



EXPLAIN ANALYZE

http://bit.ly/pg-explain

- Explain displays the execution plan

EXPLAIN ANALYZE

http://bit.ly/pg-explain

- Explain displays the execution plan
- Analyze runs it and gathers stats

EXPLAIN ANALYZE

```
Merge Right Join  (cost=25870.55..31017.51 rows=229367 width=92) (actual time=2884.501..5147.047 rows=354834 loops=1)
  Merge Cond: (a.uid = b.uid)
  ->  Index Scan using foo on bar a  (cost=0.00..537.29 rows=9246 width=27) (actual time=0.049..41.782 rows=9246 loops=1)
  ->  Materialize  (cost=25870.49..27204.80 rows=106745 width=81) (actual time=2884.413..3804.537 rows=354834 loops=1)
        ->  Sort  (cost=25870.49..26137.35 rows=106745 width=81) (actual time=2884.406..3099.732 rows=111878 loops=1)
              Sort Key: b.uid
              Sort Method:  external merge  Disk: 8928kB
...
 Total runtime: 5588.105 ms
(14 rows)
```

http://bit.ly/pg-auto-explain

SUMMARY

1. Remember the 4 qualities of good metrics
    1. Well understood
    2. Sufficiently granular
    3. Tagged & filterable
    4. Long-lived

SUMMARY

2. Understand the difference between work metrics, resource metrics & events
3. Alert on the appropriate work metrics

QUESTIONS?

JASON YEE

@gitbisect

jason.yee@datadoghq.com