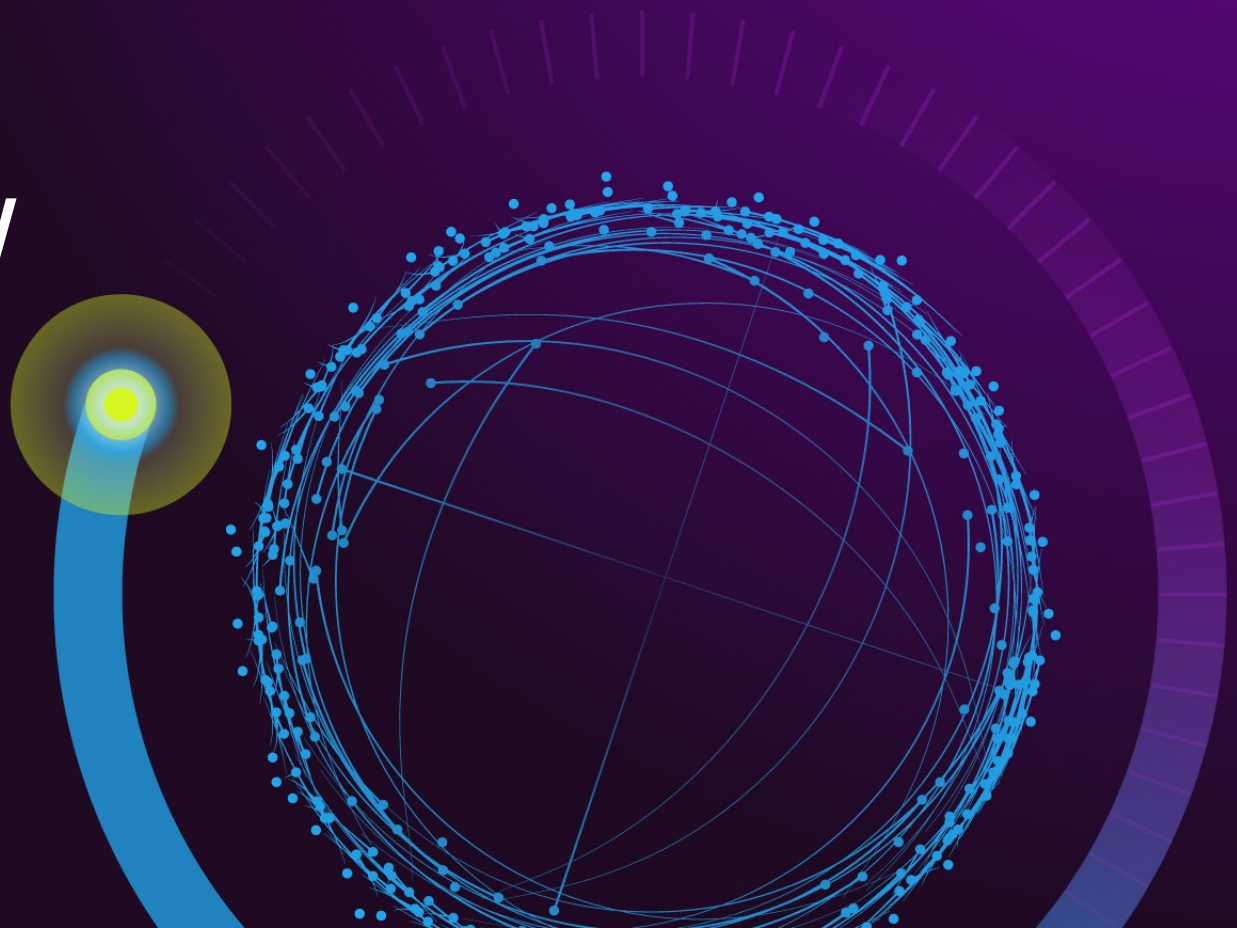
influxdata®
Act in Time

Production Gitops In Practice

Rick Spencer, InfluxData

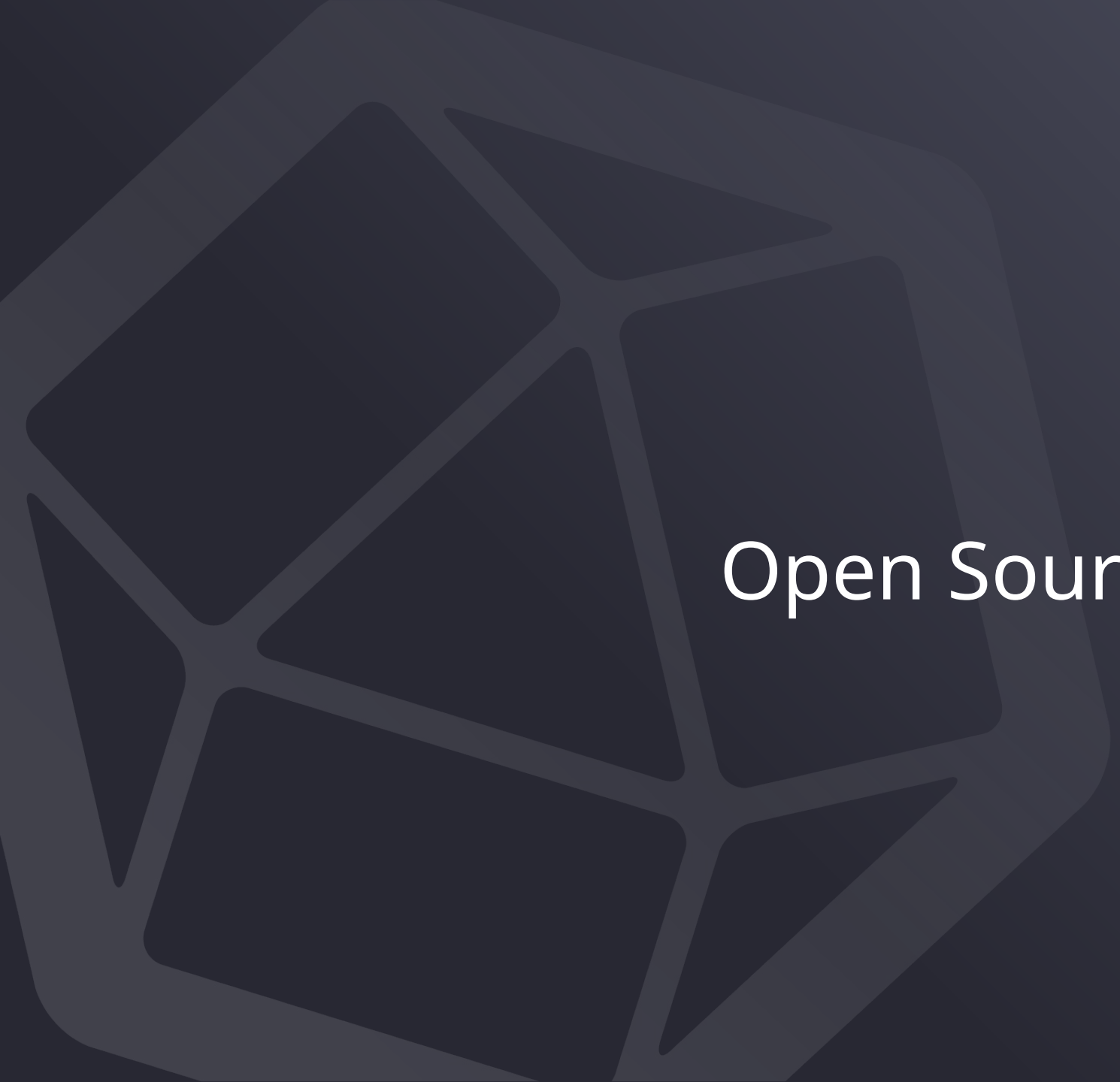
influxdata®
Act in Time

InfluxData Overview



INFLUXDB: THE OPEN SOURCE TIME SERIES DATABASE

Providing real-time visibility into stacks, sensors and systems

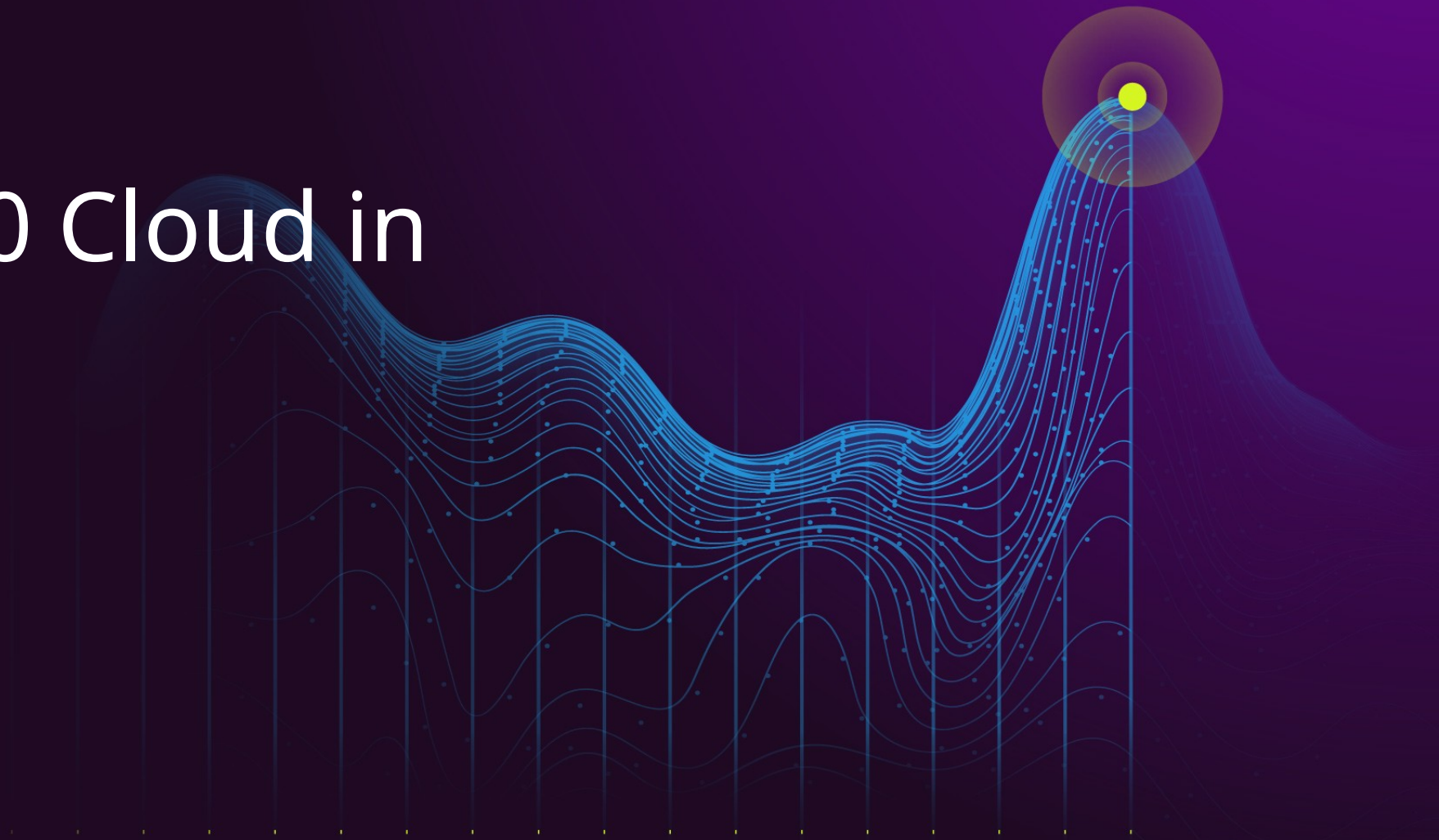


Open Source

InfluxDB Cloud (Cloud 2.0)

influxdata®
Act in Time

InfluxDB 2.0 Cloud in Production



2.0 Cloud Platform

1. SaaS data platform, but data has gravity
2. Multi Cloud (AWS, GCP, Azure)
3. Multi Region
4. Private Instances
5. Kubernetes as cloud abstraction layer

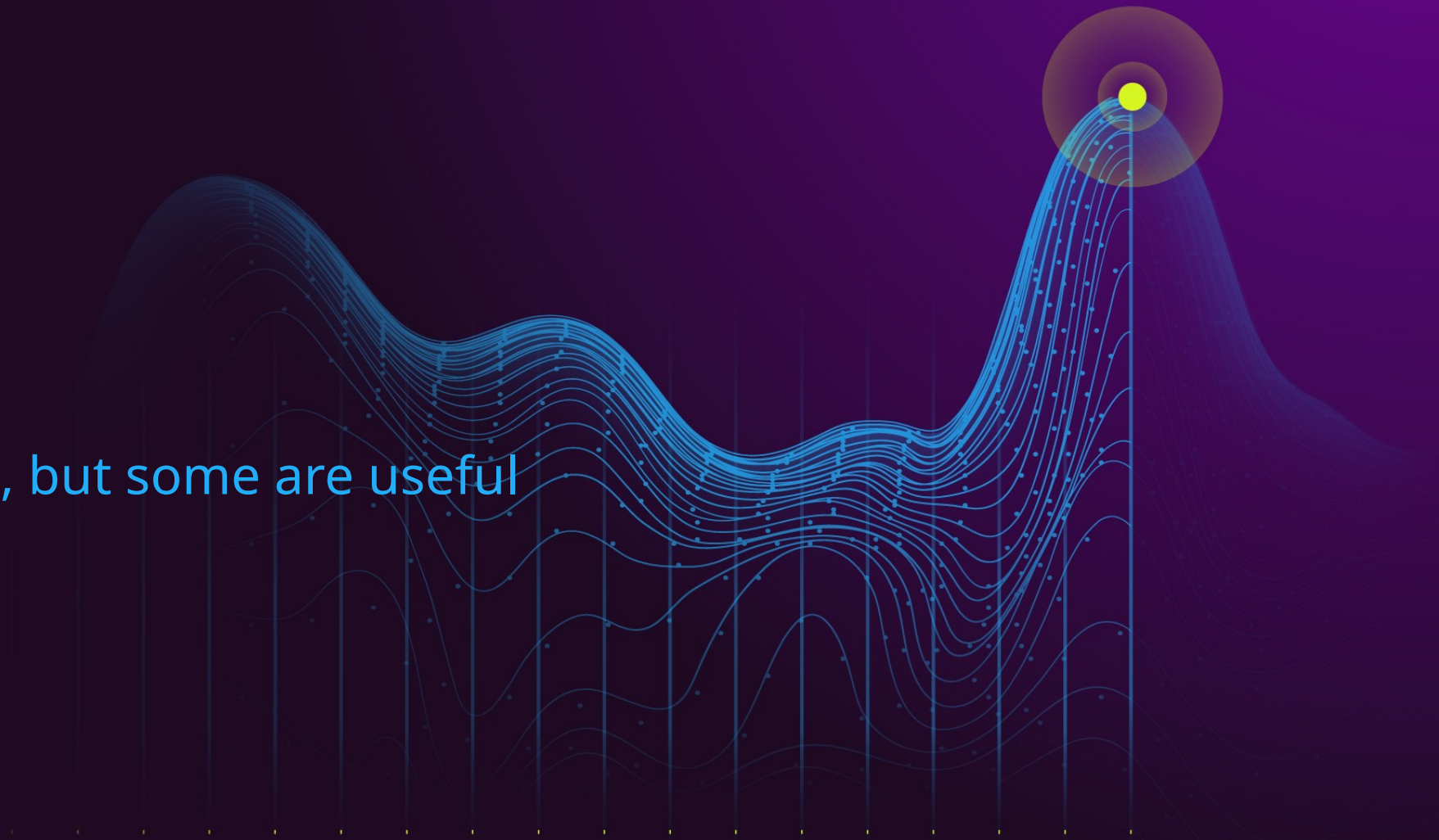
Signifiers of Gitops

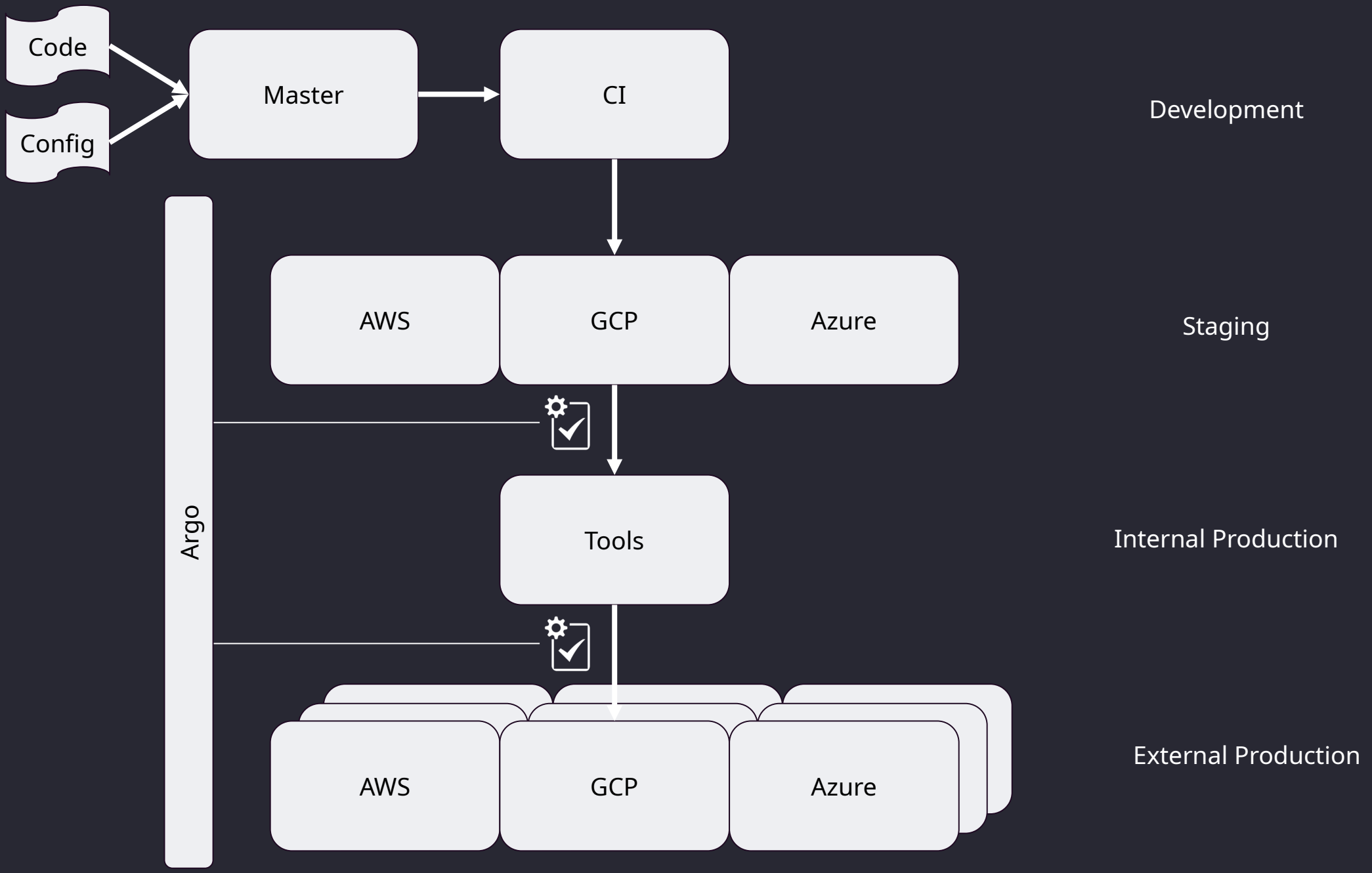
1. Developers get code into production by landing it in the main branch, automation takes care of the rest
2. Infrastructure as code is delivered in a similar manner to application code
3. Testing in Production

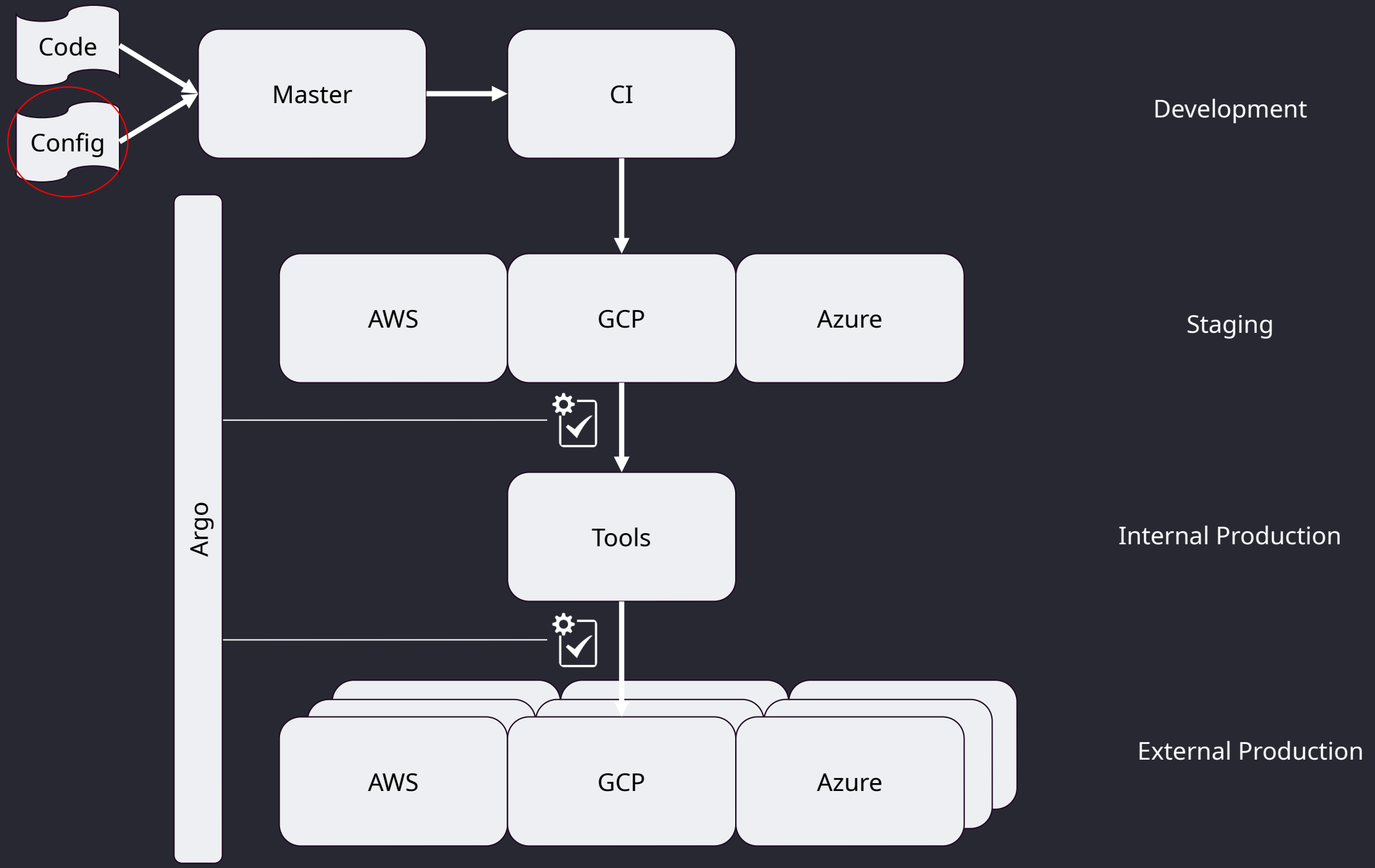
influxdata®
Act in Time

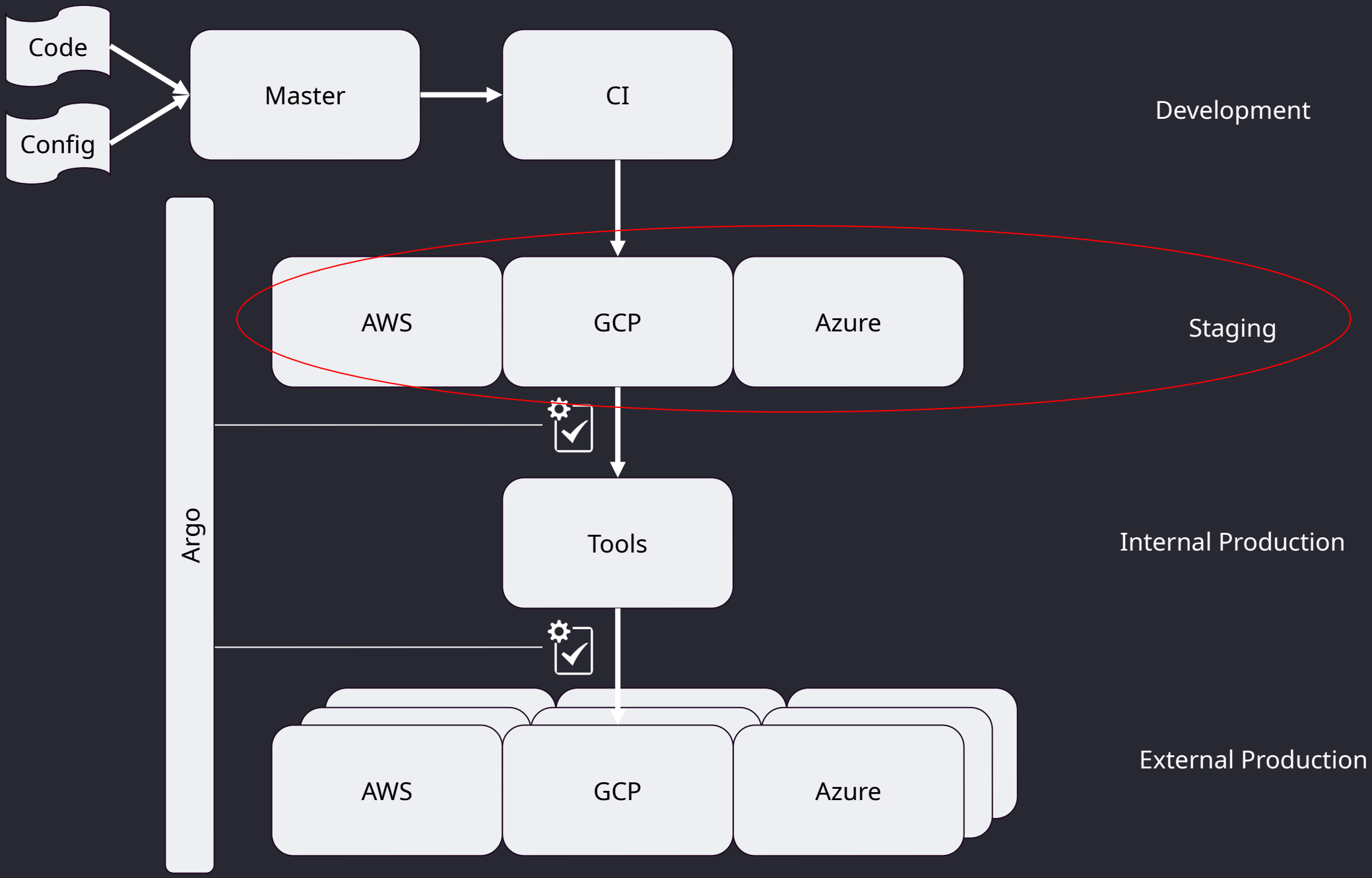
Diagrams

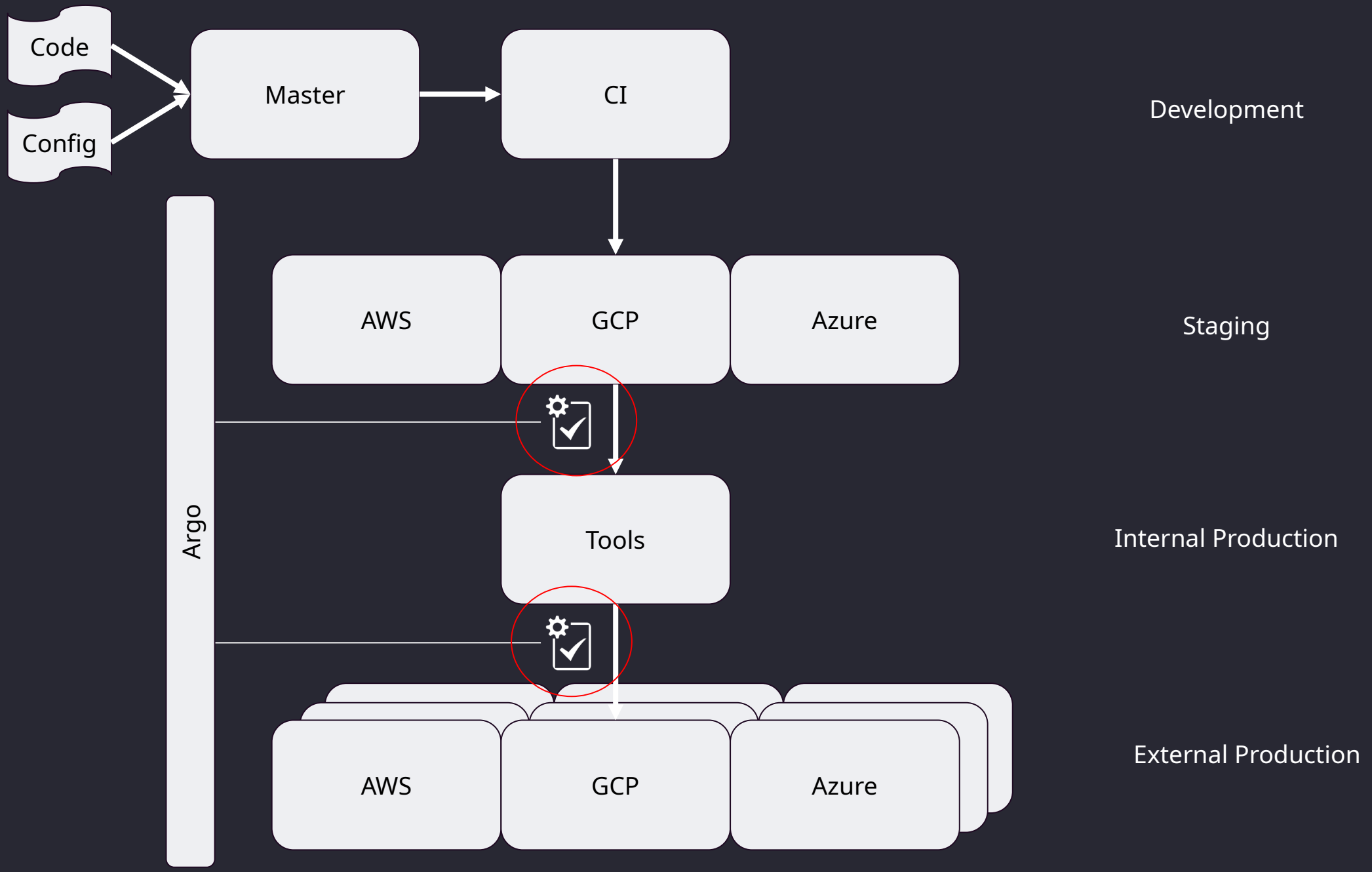
All models are wrong, but some are useful

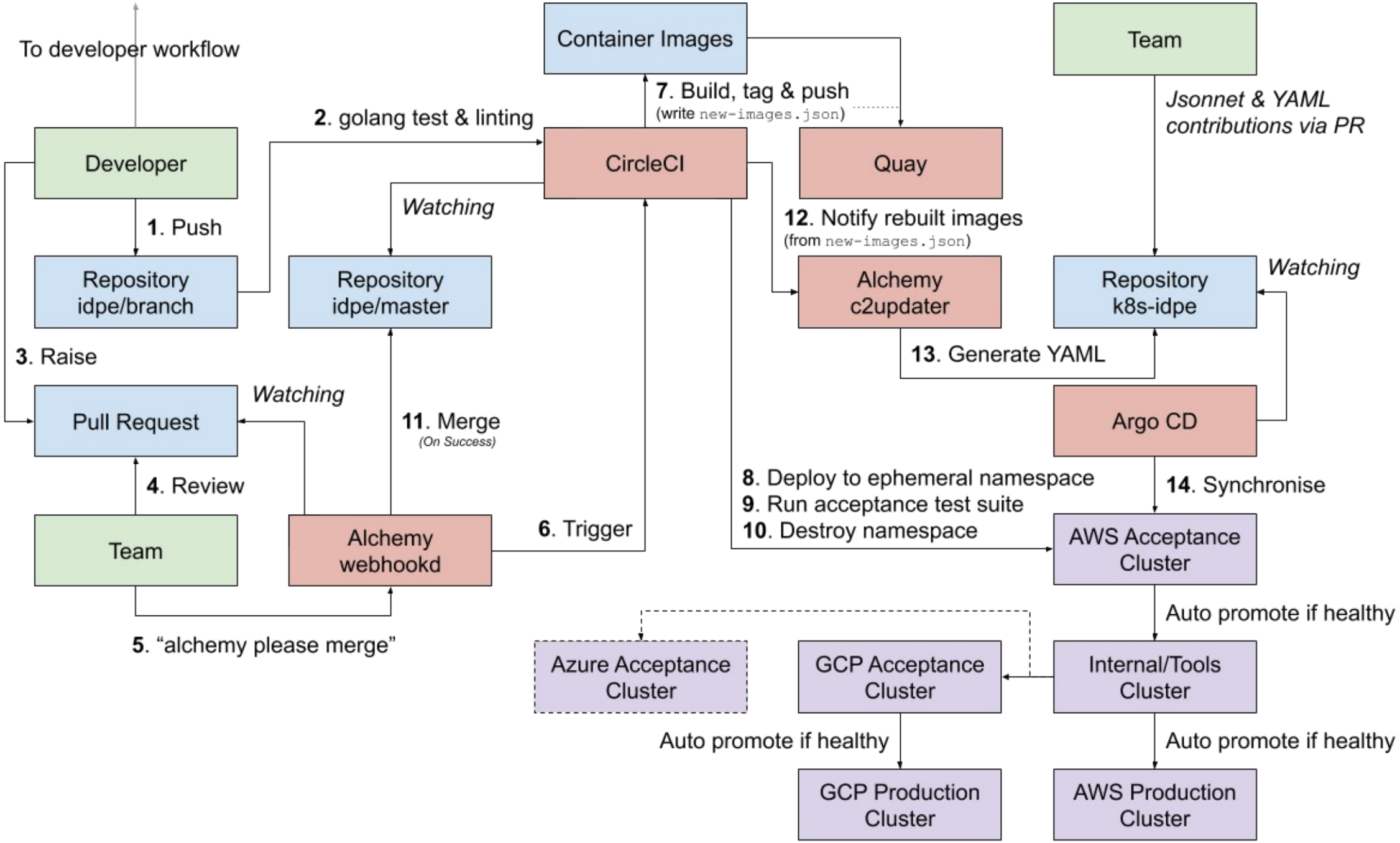



Code
Config
Master
CI
Development
Argo
AWS
GCP
Azure
Staging
Tools
Internal Production
AWS
GCP
Azure
External Production


Code
Config
Master
CI
Development
Argo
AWS
GCP
Azure
Staging
Tools
Internal Production
AWS
GCP
Azure
External Production


Code
Config
Master
CI
Development
Argo
AWS
GCP
Azure
Staging
Tools
Internal Production
AWS
GCP
Azure
External Production


Code
Config
Master
CI
Development
Argo
AWS
GCP
Azure
Staging
Tools
Internal Production
AWS
GCP
Azure
External Production


To developer workflow
Container Images
Team
7. Build, tag & push
(write new-images.json)
Jsonnet & YAML
contributions via PR
2. golang test & linting
Developer
CircleCI
Quay
Watching
12. Notify rebuilt images
(from new-images.json)
1. Push
Repository
idpe/branch
Repository
idpe/master
Alchemy
c2updater
Repository
k8s-idpe
Watching
3. Raise
13. Generate YAML
Watching
Pull Request
11. Merge
(On Success)
Argo CD
8. Deploy to ephemeral namespace
9. Run acceptance test suite
10. Destroy namespace
4. Review
14. Synchronise
Team
Alchemy
webhookd
6. Trigger
AWS Acceptance
Cluster
Auto promote if healthy
5. “alchemy please merge”
Azure Acceptance
Cluster
GCP Acceptance
Cluster
Internal/Tools
Cluster
Auto promote if healthy
Auto promote if healthy
GCP Production
Cluster
AWS Production
Cluster

Observed Benefits

1. Dramatic decrease in time working on release process
2. Fewer and less severe incidents due to more frequent and smaller releases as well as reduction in potential for human error
3. Easy to add more regions
4. Consistency between all production environments (and master) makes everything easier

influxdata®
Act in Time

How we Implement Gitops

Engineering Management View

Required Elements

1. CI to build artifacts (containers) and run tests (unit and integration)
2. A way to represent infrastructure as code (jsonnet)
3. A way to deploy K8s configuration (ArgoCD)
4. Fast deployments pipeline
5. Feature flagging (ConfigCat)
6. Production metrics and reliable alerts

Optional Components

1. Automated feedback loop (tests and metrics)
2. Pipeline for high level operations (Argo Workflow)
3. Canary deployments
4. External availability validation
5. Testing or acceptance environments *

Jsonnet - example

serviceA.jsonnet

```
{
  service: { /* ... */ },
  deployment: {
    local labels = { component:
'serviceA' },
    apiVersion: 'apps/v1',
    kind: 'Deployment',
    metadata: {
      labels: labels,
      name: 'serviceA',
    },
    spec: {
      replicas: 1,
      selector: {
        matchLabels: labels,
      },
      template: {
        spec: {
          containers: [ /* ... */ ],
        },
      }
    }
  }
}
```

serviceA-aws-prod.jsonnet

```
local serviceA =
  import './serviceA.jsonnet';

// extend serviceA with instance
// specific settings
serviceA {
  deployment+: {
    spec+: {
      replicas: 4,
      serviceAccountName: 'serviceA-
account',
      resources: {
        requests: { memory: '2G' },
        limits: { memory: '4G' },
      }
    }
  }
}
```

Output YAML file

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    component: serviceA
  name: serviceA
spec:
  replicas: 4
  resources:
    limits:
      memory: 4G
    requests:
      memory: 2G
  selector:
    matchLabels:
      component: serviceA
  serviceAccountName: serviceA-account
  template:
    spec:
      containers:
        ...
---
apiVersion: v1
kind: Service
...
```

Availability: Super Smooth Deploys

The Challenge

- Customers expect very high availability for their API calls (99.999% succeed)
- Deployments cause pods to restart, frequent deployments cause frequent restarts

Our Solution

- Custom controller code to ensure proper ordering of restarts
- “Be one with the retries”

Metrics

How we gather metrics

1. Telegraf
   - Agent for collecting metrics & writing them to InfluxDB or other outputs
   - Open source project that can be run everywhere
   - Written entirely in Go - single binary, no external dependencies
   - We run it as a sidecar

2. Different types of metrics
   - Application specific metrics - Redis, Etcd, Nginx, Zookeeper
   - Custom metrics - sending data to Telegraf from services directly
   - Istio and other services - with Telegraf Prometheus Input Plugin

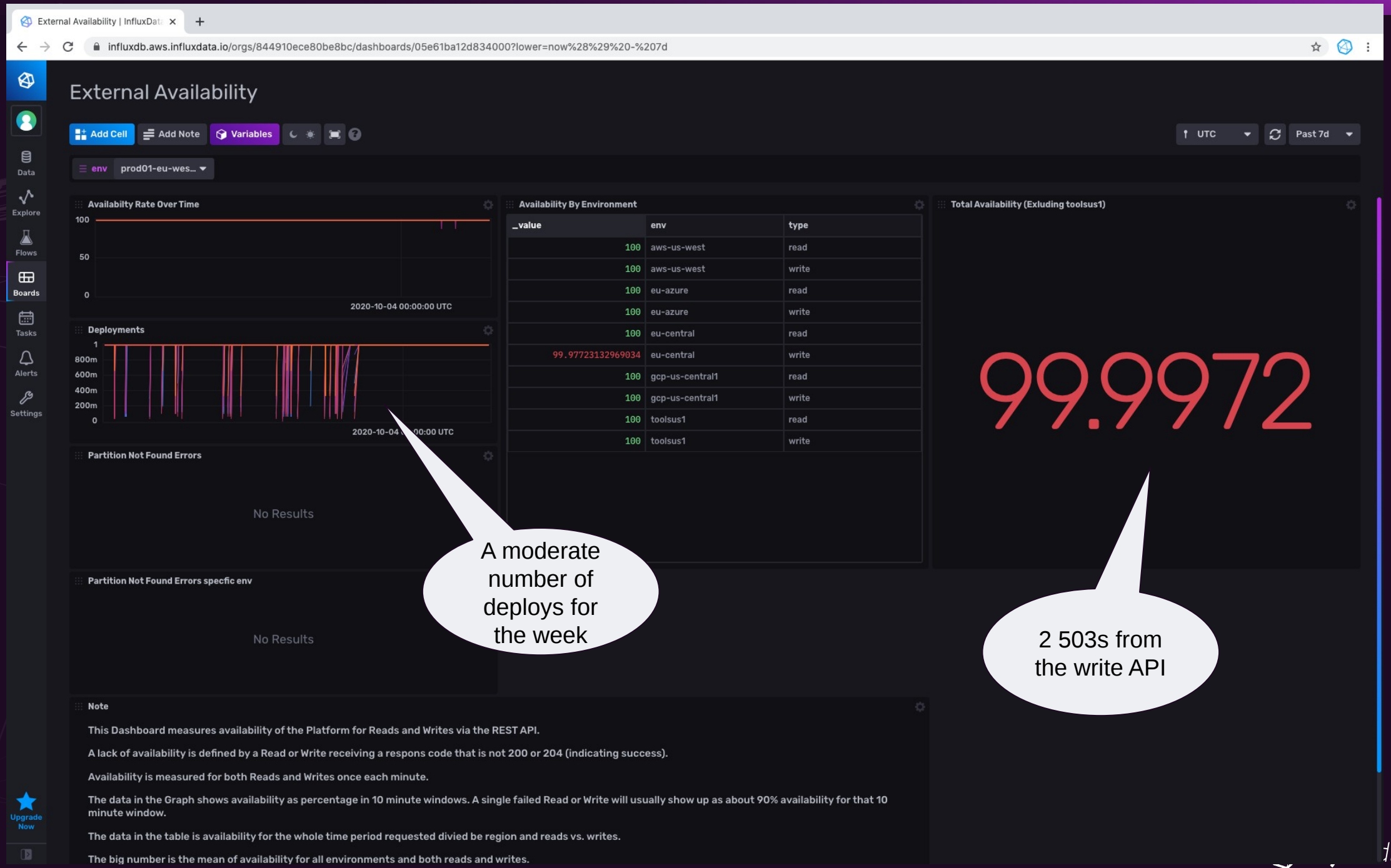
Putting Metrics to Work

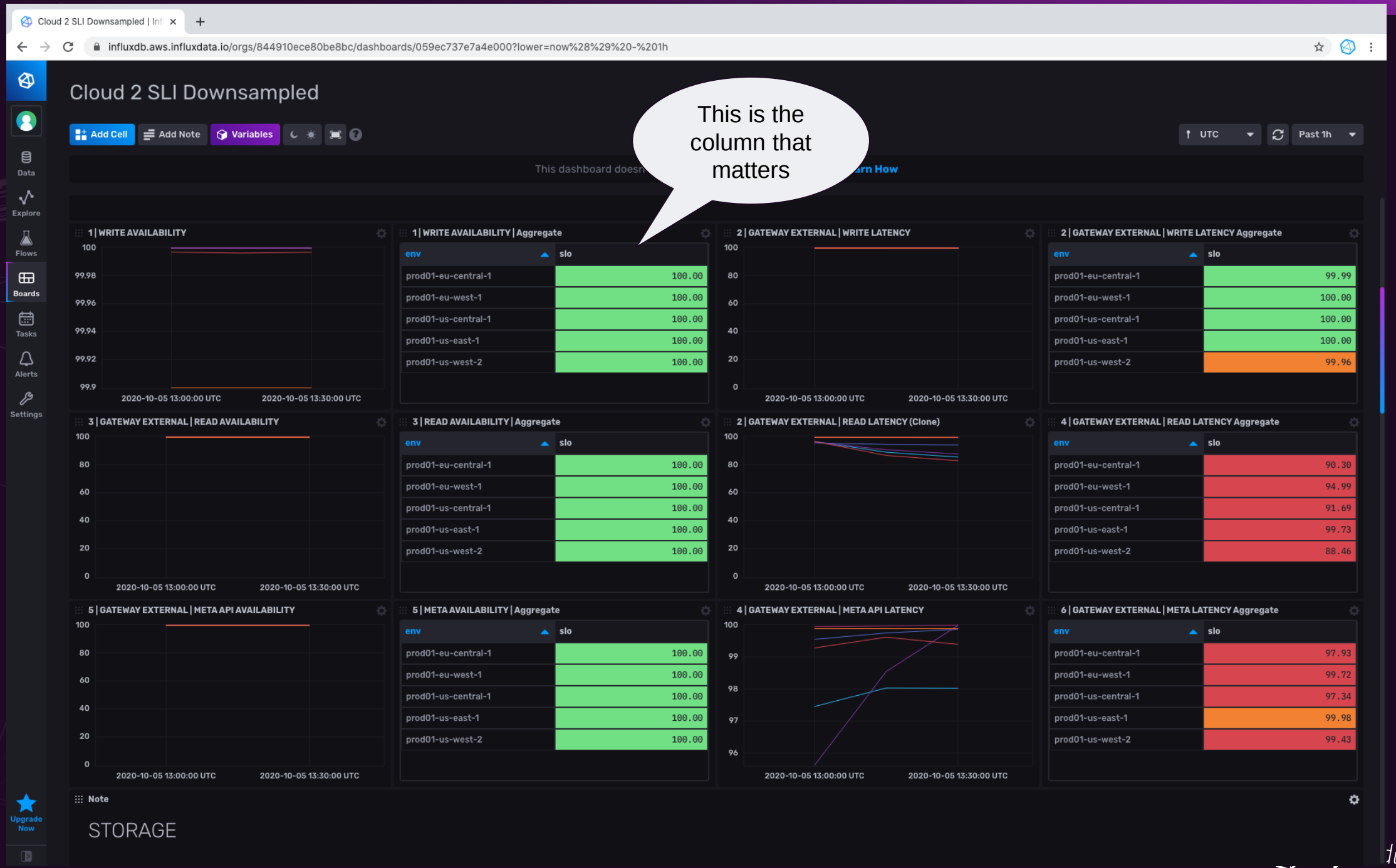
- Alert on urgent issues
- Alert on leading indicators
- Dashboards for overall system health
- Dashboards for troubleshooting
- Dashboards for feature flags

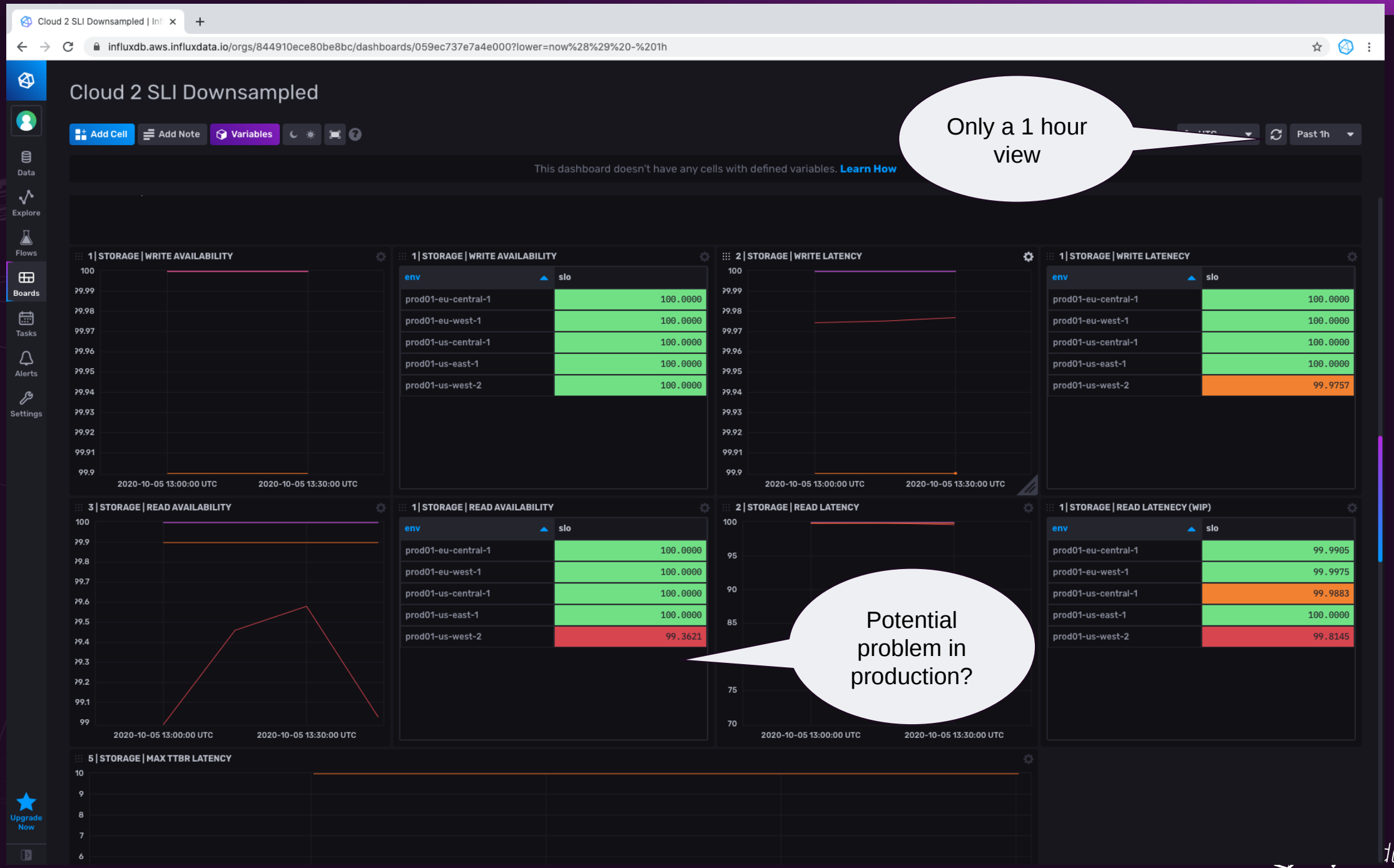
Alerts (and Dashboards)

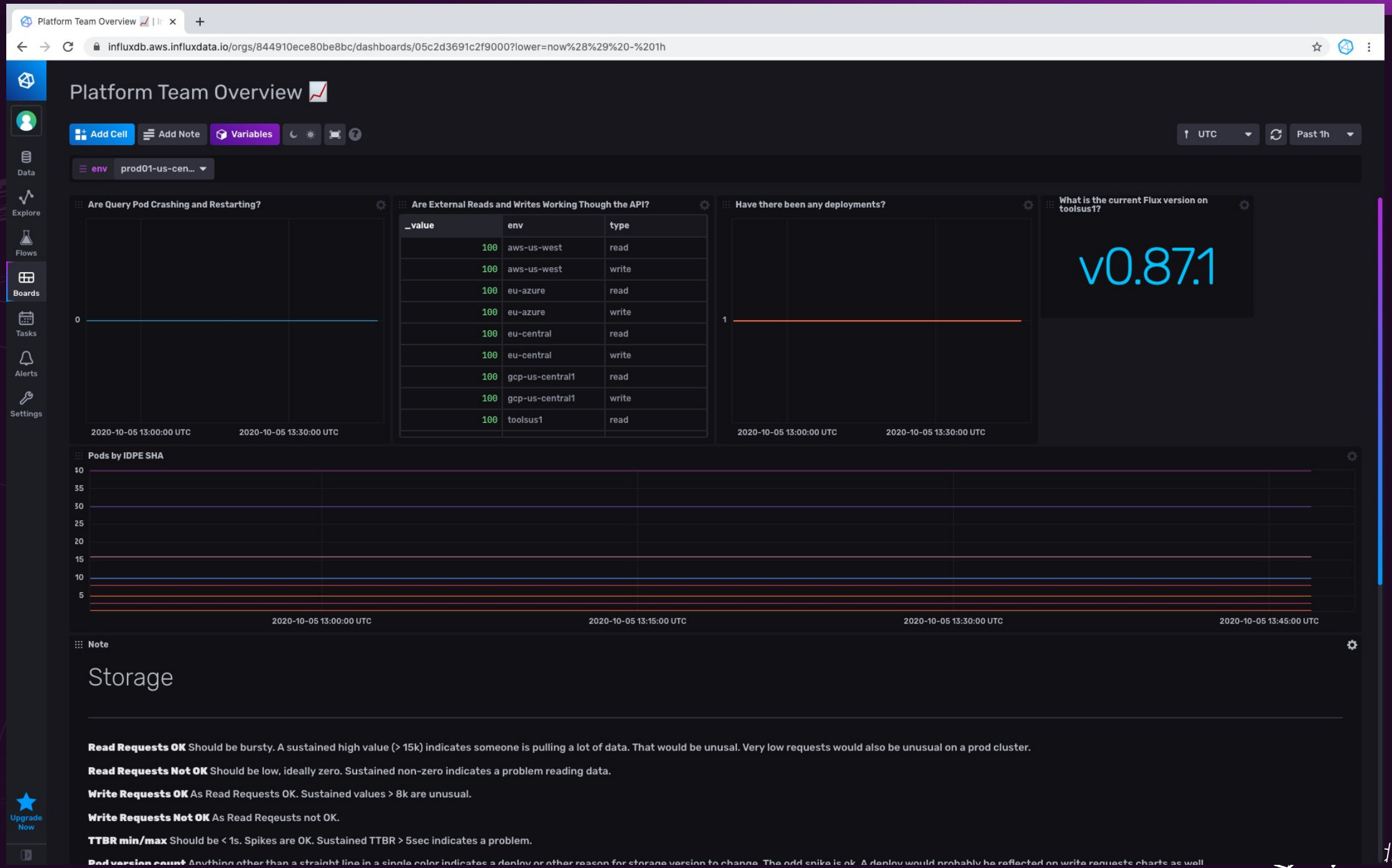
Alerts

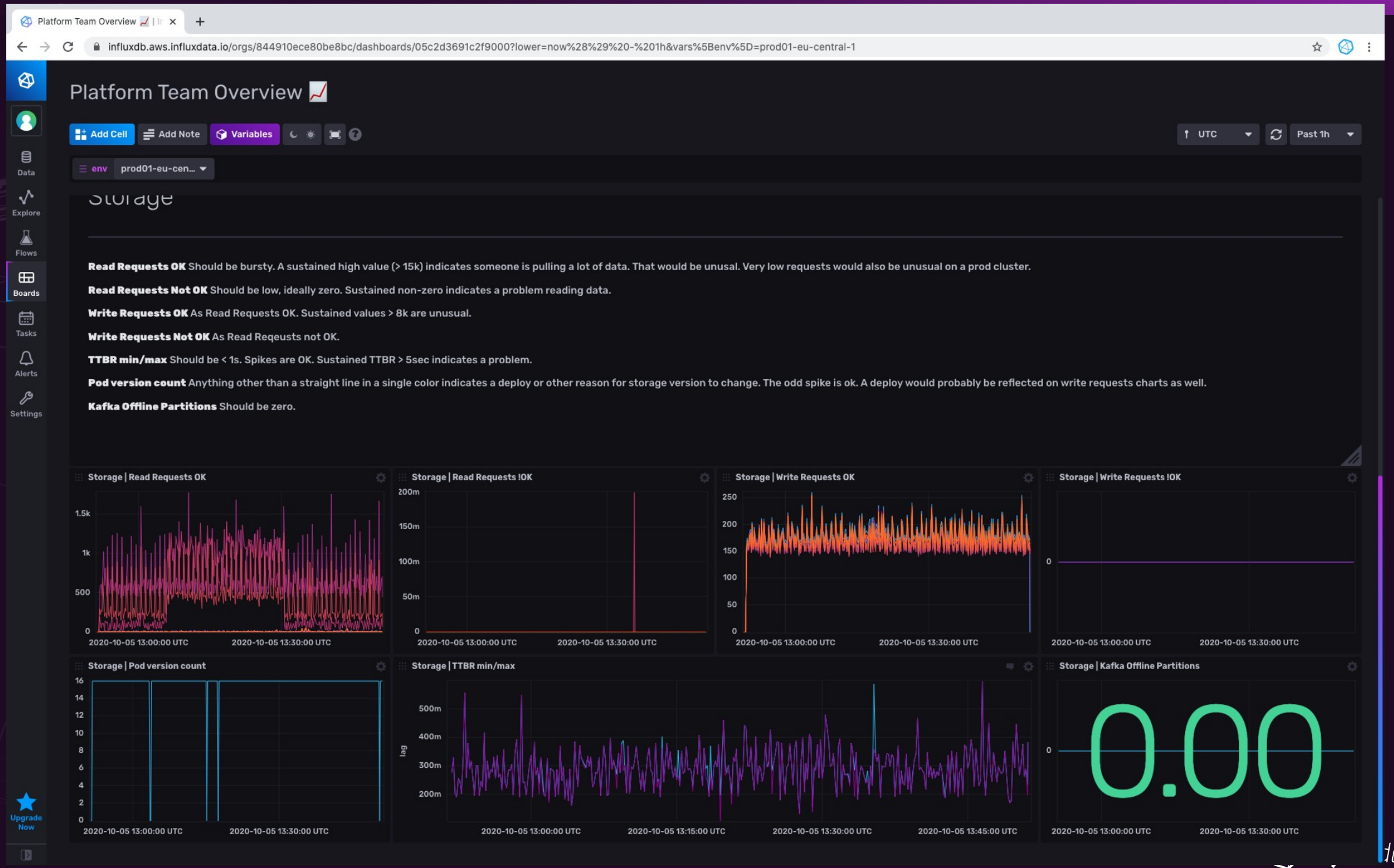
1. Kubernetes Health
2. Application Health
3. Deadman Alerts
4. Externally Running Availability Tests

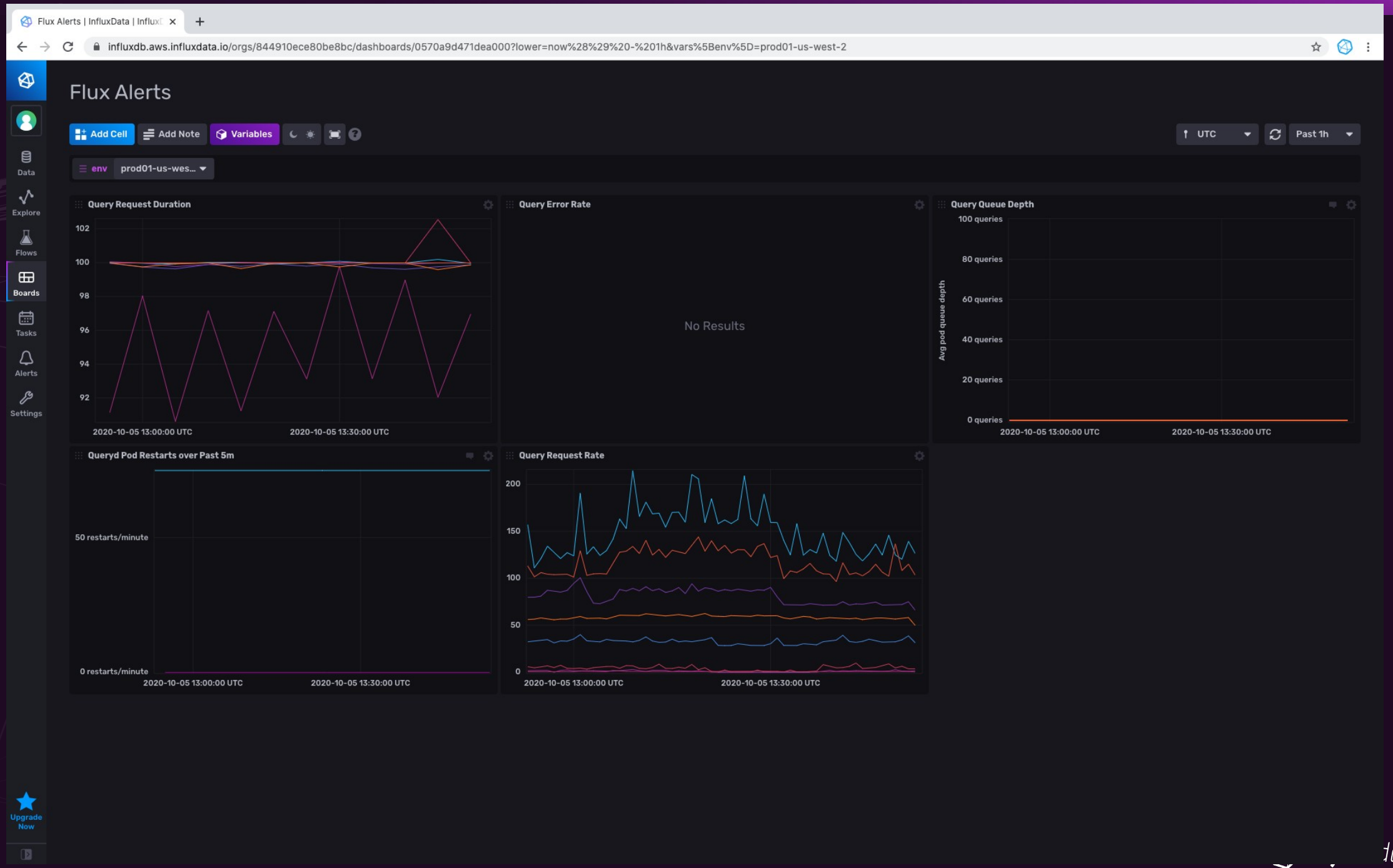

External Availability | InfluxDat
influxdb.aws.influxdata.io/orgs/844910ece80be8bc/dashboards/05e61ba12d834000?lower=now%28%29%20-%207d
External Availability
Add Cell
Add Note
Variables
UTC
Past 7d
env
prod01-eu-wes...
Data
Explore
Flows
Boards
Tasks
Alerts
Settings
Upgrade Now
Availabilty Rate Over Time
100
50
0
2020-10-04 00:00:00 UTC
Deployments
1
800m
600m
400m
200m
0
2020-10-04 00:00:00 UTC
Partition Not Found Errors
No Results
Partition Not Found Errors specfic env
No Results
Availability By Environment
_value
env
type
100 aws-us-west read
100 aws-us-west write
100 eu-azure read
100 eu-azure write
100 eu-central read
99.97723132969034 eu-central write
100 gcp-us-central1 read
100 gcp-us-central1 write
100 toolsus1 read
100 toolsus1 write
Total Availability (Exluding toolsus1)
99.9972
A moderate number of deploys for the week
2 503s from the write API
Note
This Dashboard measures availability of the Platform for Reads and Writes via the REST API.
A lack of availability is defined by a Read or Write receiving a respons code that is not 200 or 204 (indicating success).
Availability is measured for both Reads and Writes once each minute.
The data in the Graph shows availability as percentage in 10 minute windows. A single failed Read or Write will usually show up as about 90% availability for that 10 minute window.
The data in the table is availability for the whole time period requested divied be region and reads vs. writes.
The big number is the mean of availability for all environments and both reads and writes.


Cloud 2 SLI Downsampled | Inf
influxdb.aws.influxdata.io/orgs/844910ece80be8bc/dashboards/059ec737e7a4e000?lower=now%28%29%20-%201h
Cloud 2 SLI Downsampled
Add Cell
Add Note
Variables
UTC
Past 1h
This is the column that matters
Data
Explore
Flows
Boards
Tasks
Alerts
Settings
Upgrade Now
1 | WRITE AVAILABILITY
1 | WRITE AVAILABILITY | Aggregate
env
slo
prod01-eu-central-1 100.00
prod01-eu-west-1 100.00
prod01-us-central-1 100.00
prod01-us-east-1 100.00
prod01-us-west-2 100.00
2 | GATEWAY EXTERNAL | WRITE LATENCY
2 | GATEWAY EXTERNAL | WRITE LATENCY Aggregate
prod01-eu-central-1 99.99
prod01-eu-west-1 100.00
prod01-us-central-1 100.00
prod01-us-east-1 100.00
prod01-us-west-2 99.96
3 | GATEWAY EXTERNAL | READ AVAILABILITY
3 | READ AVAILABILITY | Aggregate
2 | GATEWAY EXTERNAL | READ LATENCY (Clone)
4 | GATEWAY EXTERNAL | READ LATENCY Aggregate
prod01-eu-central-1 90.30
prod01-eu-west-1 94.99
prod01-us-central-1 91.69
prod01-us-east-1 99.73
prod01-us-west-2 88.46
5 | GATEWAY EXTERNAL | META API AVAILABILITY
5 | META AVAILABILITY | Aggregate
4 | GATEWAY EXTERNAL | META API LATENCY
6 | GATEWAY EXTERNAL | META LATENCY Aggregate
prod01-eu-central-1 97.93
prod01-eu-west-1 99.72
prod01-us-central-1 97.34
prod01-us-east-1 99.98
prod01-us-west-2 99.43
2020-10-05 13:00:00 UTC
2020-10-05 13:30:00 UTC
Note
STORAGE


Cloud 2 SLI Downsampled | Inf
influxdb.aws.influxdata.io/orgs/844910ece80be8bc/dashboards/059ec737e7a4e000?lower=now%28%29%20-%201h
Cloud 2 SLI Downsampled
Add Cell
Add Note
Variables
Past 1h
This dashboard doesn't have any cells with defined variables. Learn How
Only a 1 hour view
Data
Explore
Flows
Boards
Tasks
Alerts
Settings
Upgrade Now
1 | STORAGE | WRITE AVAILABILITY
1 | STORAGE | WRITE AVAILABILITY
env
slo
prod01-eu-central-1 100.0000
prod01-eu-west-1 100.0000
prod01-us-central-1 100.0000
prod01-us-east-1 100.0000
prod01-us-west-2 100.0000
2 | STORAGE | WRITE LATENCY
1 | STORAGE | WRITE LATENECY
prod01-us-west-2 99.9757
3 | STORAGE | READ AVAILABILITY
1 | STORAGE | READ AVAILABILITY
prod01-us-west-2 99.3621
2 | STORAGE | READ LATENCY
1 | STORAGE | READ LATENECY (WIP)
prod01-eu-central-1 99.9905
prod01-eu-west-1 99.9975
prod01-us-central-1 99.9883
prod01-us-east-1 100.0000
prod01-us-west-2 99.8145
Potential problem in production?
2020-10-05 13:00:00 UTC
2020-10-05 13:30:00 UTC
5 | STORAGE | MAX TTBR LATENCY


Platform Team Overview
influxdb.aws.influxdata.io/orgs/844910ece80be8bc/dashboards/05c2d3691c2f9000?lower=now%28%29%20-%201h
Platform Team Overview
Add Cell
Add Note
Variables
UTC
Past 1h
env
prod01-us-cen...
Data
Explore
Flows
Boards
Tasks
Alerts
Settings
Upgrade Now
Are Query Pod Crashing and Restarting?
0
2020-10-05 13:00:00 UTC
2020-10-05 13:30:00 UTC
Are External Reads and Writes Working Though the API?
_value
env
type
100 aws-us-west read
100 aws-us-west write
100 eu-azure read
100 eu-azure write
100 eu-central read
100 eu-central write
100 gcp-us-central1 read
100 gcp-us-central1 write
100 toolsus1 read
Have there been any deployments?
1
2020-10-05 13:00:00 UTC
2020-10-05 13:30:00 UTC
What is the current Flux version on toolsus1?
v0.87.1
Pods by IDPE SHA
40
35
30
25
20
15
10
5
2020-10-05 13:00:00 UTC
2020-10-05 13:15:00 UTC
2020-10-05 13:30:00 UTC
2020-10-05 13:45:00 UTC
Note
Storage
Read Requests OK Should be bursty. A sustained high value (> 15k) indicates someone is pulling a lot of data. That would be unusal. Very low requests would also be unusual on a prod cluster.
Read Requests Not OK Should be low, ideally zero. Sustained non-zero indicates a problem reading data.
Write Requests OK As Read Requests OK. Sustained values > 8k are unusual.
Write Requests Not OK As Read Reqeusts not OK.
TTBR min/max Should be < 1s. Spikes are OK. Sustained TTBR > 5sec indicates a problem.


Platform Team Overview
influxdb.aws.influxdata.io/orgs/844910ece80be8bc/dashboards/05c2d3691c2f9000?lower=now%28%29%20-%201h&vars%5Benv%5D=prod01-eu-central-1
Platform Team Overview
Add Cell
Add Note
Variables
UTC
Past 1h
env
prod01-eu-cen...
Data
Explore
Flows
Boards
Tasks
Alerts
Settings
Upgrade Now
Storage
Read Requests OK Should be bursty. A sustained high value (> 15k) indicates someone is pulling a lot of data. That would be unusal. Very low requests would also be unusual on a prod cluster.
Read Requests Not OK Should be low, ideally zero. Sustained non-zero indicates a problem reading data.
Write Requests OK As Read Requests OK. Sustained values > 8k are unusual.
Write Requests Not OK As Read Reqeusts not OK.
TTBR min/max Should be < 1s. Spikes are OK. Sustained TTBR > 5sec indicates a problem.
Pod version count Anything other than a straight line in a single color indicates a deploy or other reason for storage version to change. The odd spike is ok. A deploy would probably be reflected on write requests charts as well.
Kafka Offline Partitions Should be zero.
Storage | Read Requests OK
Storage | Read Requests !OK
Storage | Write Requests OK
Storage | Write Requests !OK
Storage | Pod version count
Storage | TTBR min/max
Storage | Kafka Offline Partitions
0.00
2020-10-05 13:00:00 UTC
2020-10-05 13:30:00 UTC
2020-10-05 13:15:00 UTC
2020-10-05 13:45:00 UTC


Flux Alerts | InfluxData | Influx
influxdb.aws.influxdata.io/orgs/844910ece80be8bc/dashboards/0570a9d471dea000?lower=now%28%29%20-%201h&vars%5Benv%5D=prod01-us-west-2
Flux Alerts
Add Cell
Add Note
Variables
UTC
Past 1h
env
prod01-us-wes...
Data
Explore
Flows
Boards
Tasks
Alerts
Settings
Upgrade Now
Query Request Duration
102
100
98
96
94
92
2020-10-05 13:00:00 UTC
2020-10-05 13:30:00 UTC
Query Error Rate
No Results
Query Queue Depth
100 queries
80 queries
60 queries
40 queries
20 queries
0 queries
Avg pod queue depth
2020-10-05 13:00:00 UTC
2020-10-05 13:30:00 UTC
Queryd Pod Restarts over Past 5m
50 restarts/minute
0 restarts/minute
2020-10-05 13:00:00 UTC
2020-10-05 13:30:00 UTC
Query Request Rate
200
150
100
50
0
2020-10-05 13:00:00 UTC
2020-10-05 13:30:00 UTC

Organizational Challenges

1. Change of mindset from releases to Continuous Delivery
2. Rolling out breaking changes incrementally
3. Overcoming fear of breaking production

Our Next Steps

1. Feed metrics into automated feedback loop to gate promotion
2. Testing health *during a deploy*
3. Gather more metrics around GitOps processes
4. Game days
5. Canary