Is Your Team Any Good?

4 key metrics for measuring your team’s performance

cortex

Cristina Buenahora

BRIDGEWATER

cortex

⏳ Lead Time For Changes

🖥️ Deployment Frequency

DORA Metrics

🔔 Mean Time to Recovery

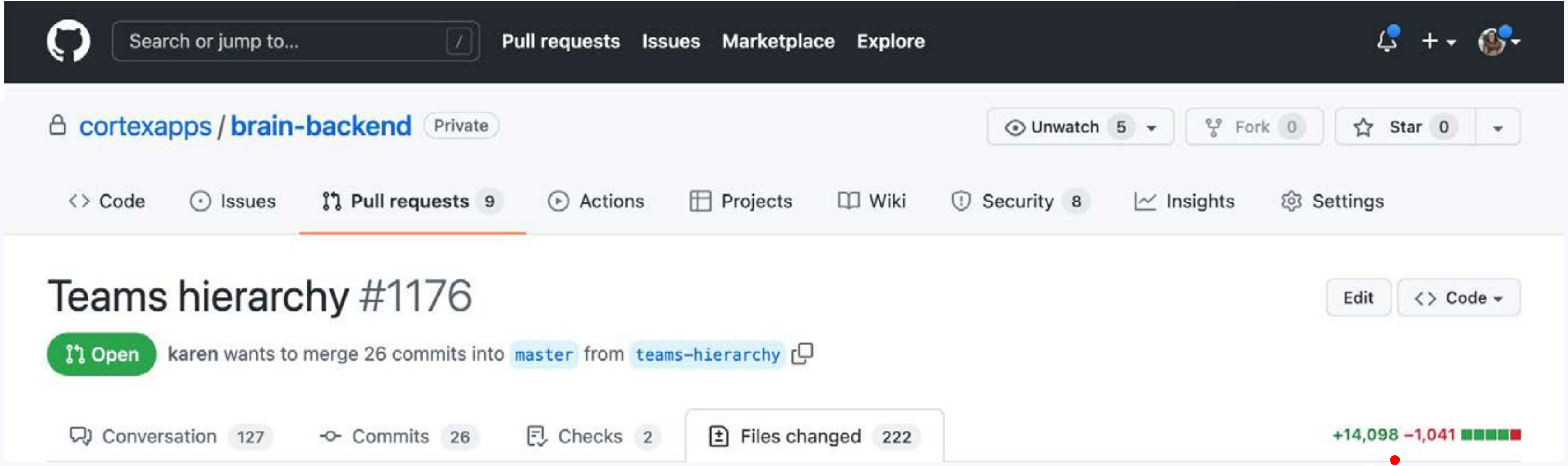
🐛 Change Failure Rate

⏳ Lead Time for Changes

Lead Time for Changes

Elite Performers	<1 hour
High Performers	1 day to 1 week
Medium Performers	1 month to 6 months
Low Performers	6+ months

* Accelerate State of DevOps 2021 Report



cortexapps / brain-backend Private

Teams hierarchy #1176

Open **karen** wants to merge 26 commits into `master` from `teams-hierarchy`

+14,098 −1,041

BAD

Long Lead Time for Changes

- Attempts to make huge changes in one go
- Code reviews sit around for too long
- Changing requirements during development
- Insufficient CI/CD pipelines

Short Lead Time for Changes

- Make sure everyone on the team can review PRs
- Reject tickets that aren’t fully fleshed out & clarify requirements upfront
- Escalate changing requirements

Measuring Lead Time for Changes

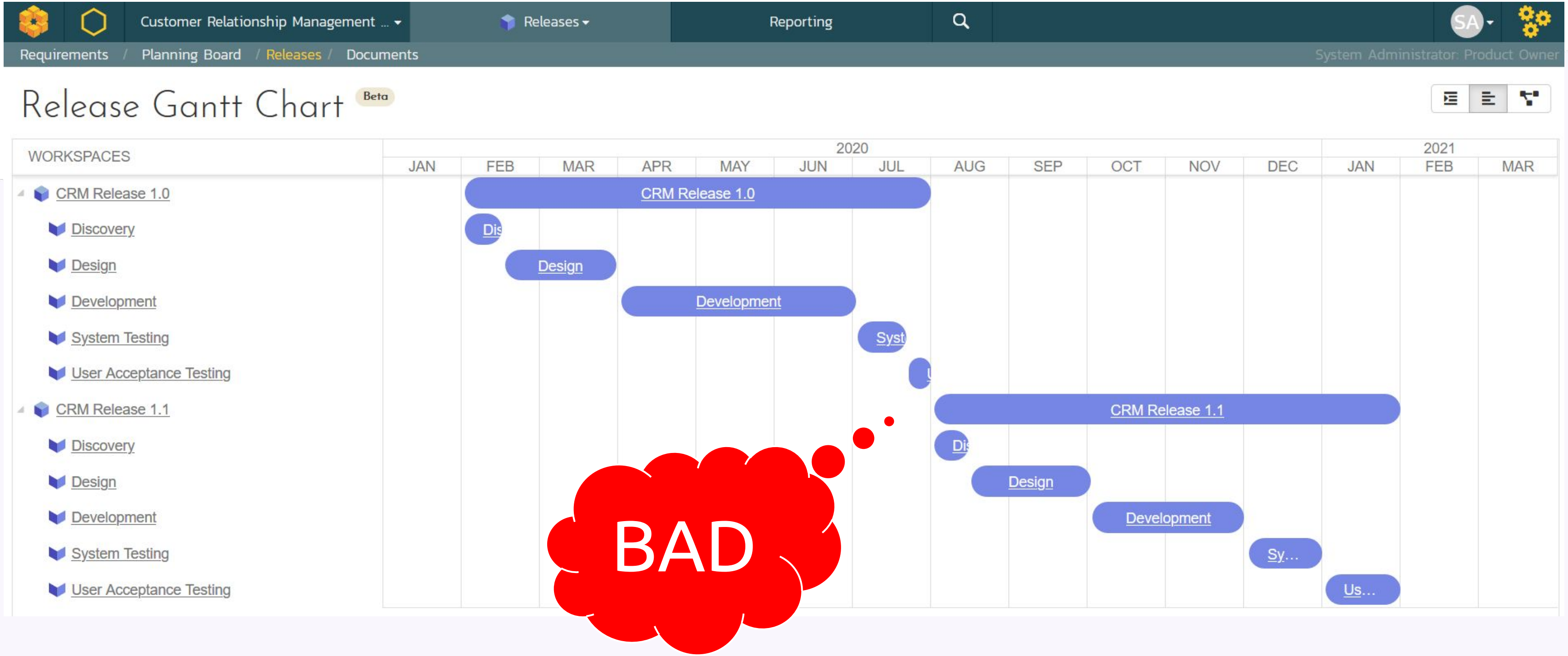
- How long did it take for a ticket to hit production from when:
    - It was created?
    - The engineer started working on it?
    - The pull request was open?
- How much time between a PR being merged and being released?

🖥️ Deployment Frequency

Deployment Frequency

Elite Performers	Multiple times a day
High Performers	Once a week to a month
Medium Performers	Once a month to every 6 months
Low Performers	Less than once every 6 months

* Accelerate State of DevOps 2021 Report


Customer Relationship Management ...
Releases
Reporting
SA
Requirements / Planning Board / Releases / Documents
System Administrator: Product Owner
Release Gantt Chart
Beta
WORKSPACES
2020
2021
JAN
FEB
MAR
APR
MAY
JUN
JUL
AUG
SEP
OCT
NOV
DEC
JAN
FEB
MAR
CRM Release 1.0
Discovery
Design
Development
System Testing
User Acceptance Testing
CRM Release 1.1
Discovery
Design
Development
System Testing
User Acceptance Testing
CRM Release 1.0
Dis
Design
Development
Syst
CRM Release 1.1
Di
Design
Development
Sy...
Us...

BAD

Low Deployment Frequency

- Insufficient CI/CD pipelines
- Bottleneck of people with permission to deploy
- Lengthy manual testing process

High Deployment Frequency

- Make it easy to release
- Set up good integrated & end to end tests
- Drive a DevOps ethos

Measuring Deployment Frequency

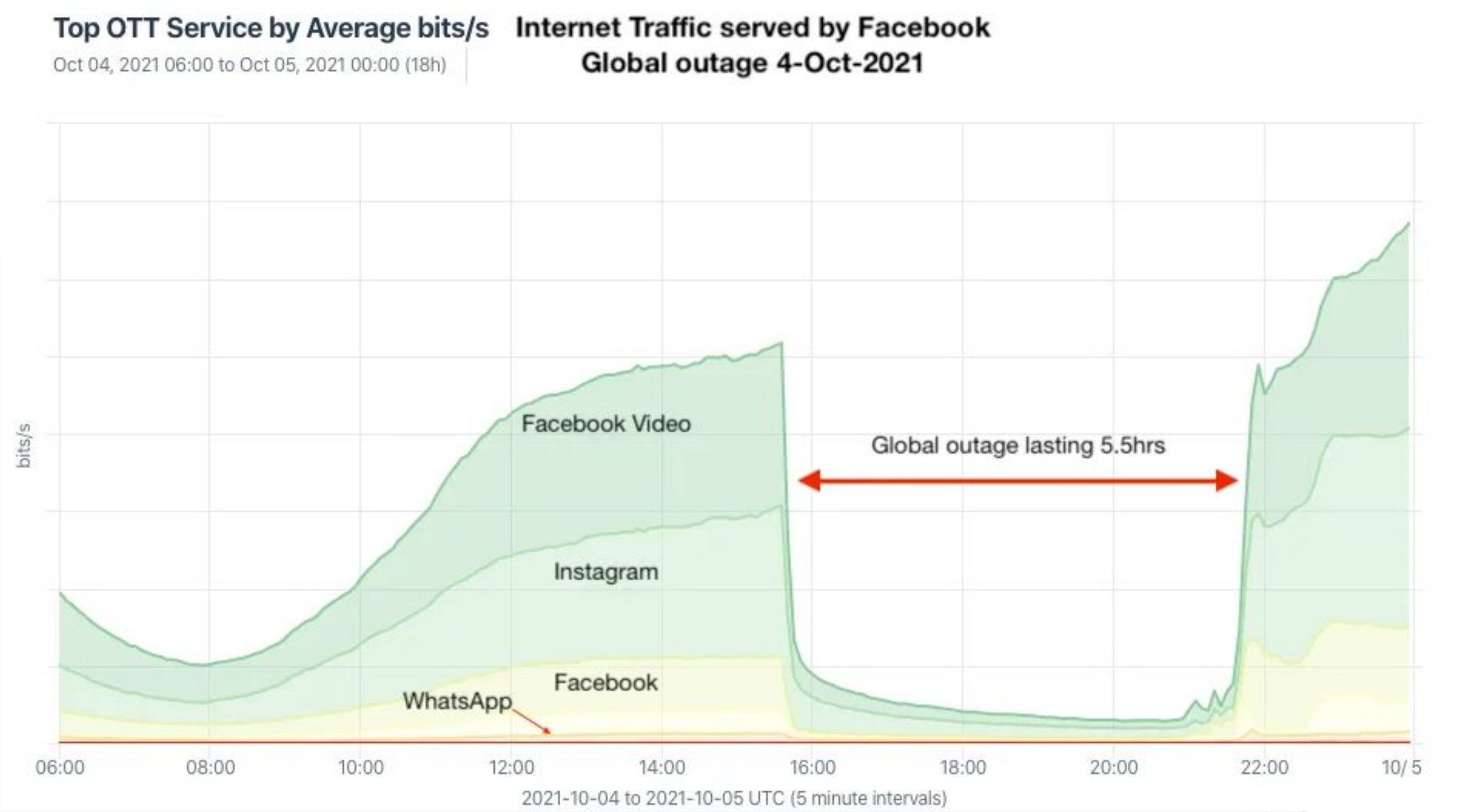
- How many releases in a sprint?
- What is the average amount of time between releases?

🔔 Mean Time to Recovery

Mean Time to Recovery

Elite Performers	<1 hour
High Performers	<1 day
Medium Performers	1 day to 1 week
Low Performers	Over 6 months

* Accelerate State of DevOps 2021 Report

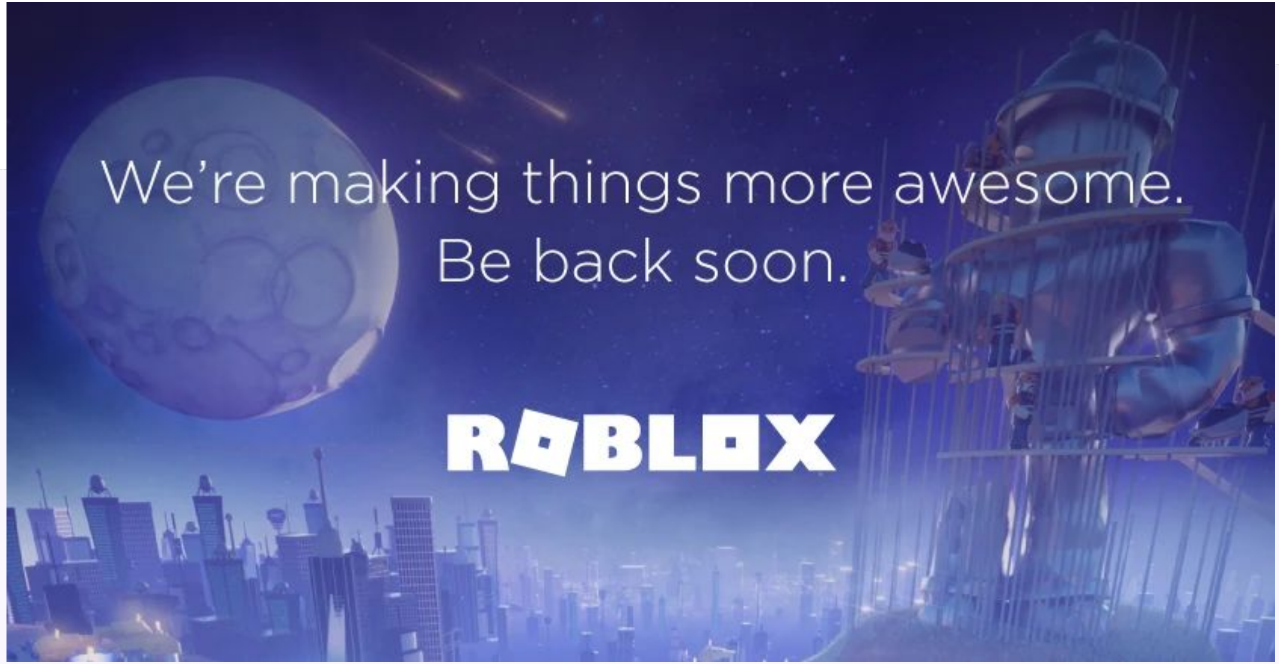


Top OTT Service by Average bits/s

Oct 04, 2021 06:00 to Oct 05, 2021 00:00 (18h)

Internet Traffic served by Facebook
Global outage 4-Oct-2021

** https://en.wikipedia.org/wiki/2021_Facebook_outage


We're making things more awesome.
Be back soon.
ROBLOX

Long Mean Time to Recovery

- Risky infrastructure
- Poor ability to rollback changes
- Bad incident management process
- Tribal knowledge / insufficient documentation

Short Mean Time to Recovery

- Tight incident management processes
- Ability to rollback quickly (and lots of people know how)
- Tools to quickly diagnose what is wrong
- Clear runbooks that are easily accessible

Measuring Mean Time to Recovery

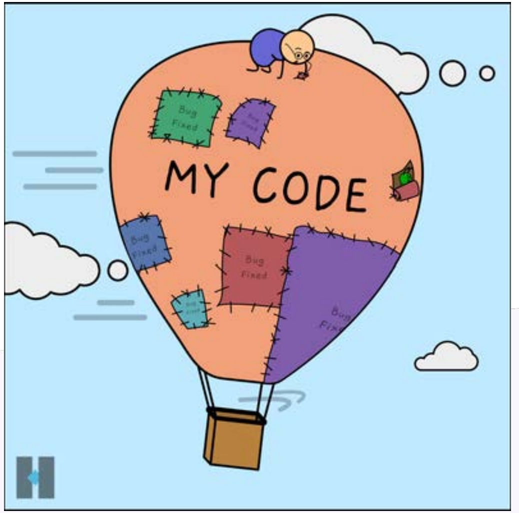
- How long was the outage?
- How much time between when the fix was discovered and when it was released?
- How long did it take to discover the outage?

🐛 Change Failure Rate

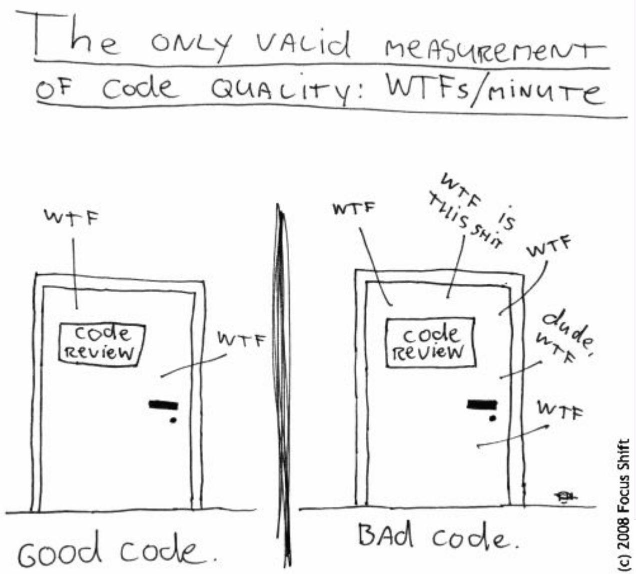
Change Failure Rate

Elite Performers	**0 - 15%**
High Performers	**15 - 30%**
Medium Performers	**15 - 30%**
Low Performers	**15 - 30%**

* Accelerate State of DevOps 2021 Report


MY CODE
Bug Fixed
Bug Fixed
Bug Fixed
Bug Fixed
Bug Fixed
Bug Fixed

Defect In Production?
Works
On My Machine
quickmeme.com


The only valid measurement of code quality: WTFs/minute
WTF
code review
WTF
Good code.
WTF
WTF is this shit
WTF
code review
dude, WTF
WTF
BAd code.
(c) 2008 Focus Shift

High Change Failure Rate

- Sloppy code reviews
- Insufficient testing
- Staging environments with inaccurate / insufficient data

Low Change Failure Rate

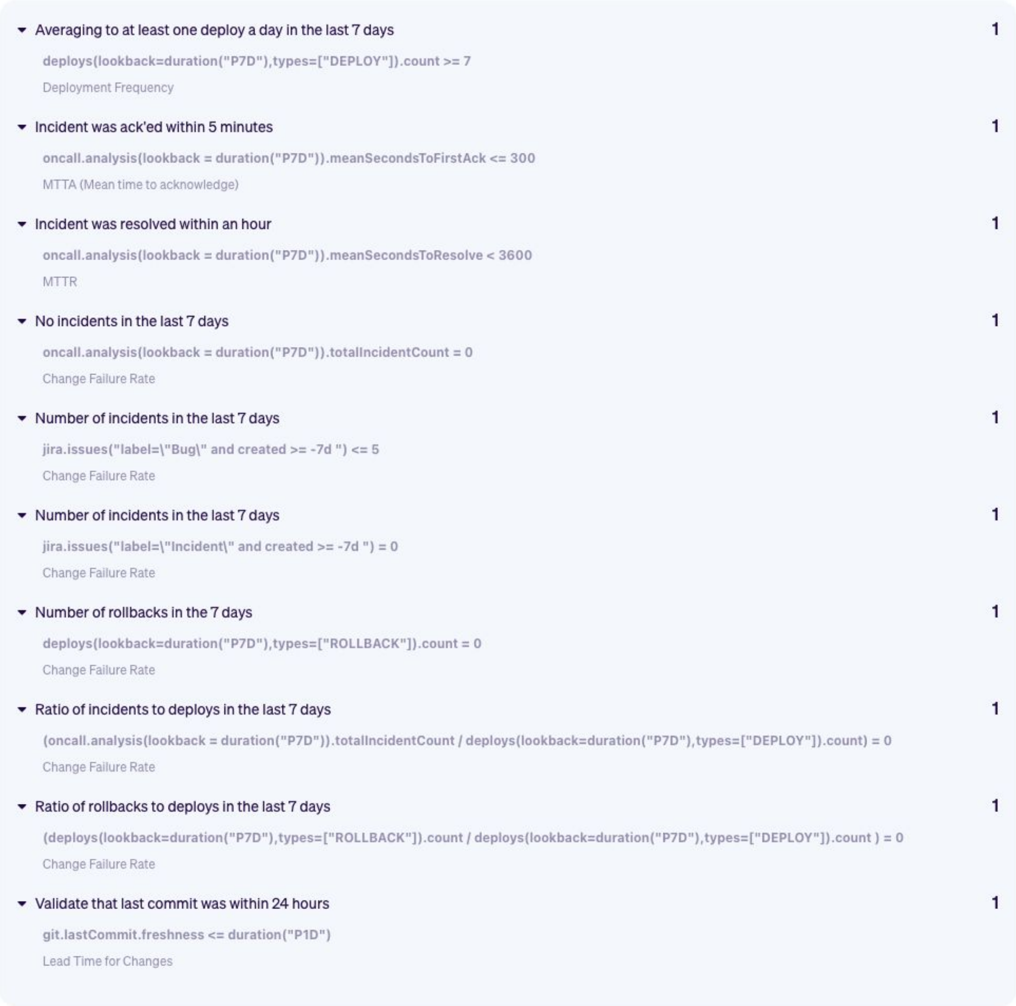
- DevOps ethos → a culture of quality
- Representative development and staging environments
- Strong partnership between product and engineering

Measuring Change Failure Rate

- How many releases have caused downtime?
- How many tickets have resulted in incidents?
- How many tickets have had follow up bug tickets?
- Digging deeper - How many of these issues are the result of
    - Insufficient unit tests?
    - Insufficient end to end tests?
    - Bad data?

Putting it all together: Speed v. Stability

- Speed: Lead Time for Changes & Deployment Frequency
- Stability: Mean Time to Recovery & Change Failure Rate



Averaging to at least one deploy a day in the last 7 days

1

```
deploys(lookback=duration("P7D"),types=["DEPLOY"]).count >= 7
```

Deployment Frequency

Incident was ack'ed within 5 minutes

1

```
oncall.analysis(lookback = duration("P7D")).meanSecondsToFirstAck <= 300
```

MTTA (Mean time to acknowledge)

Incident was resolved within an hour

1

```
oncall.analysis(lookback = duration("P7D")).meanSecondsToResolve < 3600
```

MTTR

No incidents in the last 7 days

1

```
oncall.analysis(lookback = duration("P7D")).totalIncidentCount = 0
```

Change Failure Rate

Number of incidents in the last 7 days

1

```
jira.issues("label=\"Bug\" and created >= -7d ") <= 5
```

Change Failure Rate

Number of incidents in the last 7 days

1

```
jira.issues("label=\"Incident\" and created >= -7d ") = 0
```

Change Failure Rate

Number of rollbacks in the 7 days

1

```
deploys(lookback=duration("P7D"),types=["ROLLBACK"]).count = 0
```

Change Failure Rate

Ratio of incidents to deploys in the last 7 days

1

```
(oncall.analysis(lookback = duration("P7D")).totalIncidentCount / deploys(lookback=duration("P7D"),types=["DEPLOY"]).count) = 0
```

Change Failure Rate

Ratio of rollbacks to deploys in the last 7 days

1

```
(deploys(lookback=duration("P7D"),types=["ROLLBACK"]).count / deploys(lookback=duration("P7D"),types=["DEPLOY"]).count ) = 0
```

Change Failure Rate

Validate that last commit was within 24 hours

1

```
git.lastCommit.freshness <= duration("P1D")
```

Lead Time for Changes

cortex

Thank you

Cristina Buenahora

cristina@cortex.io