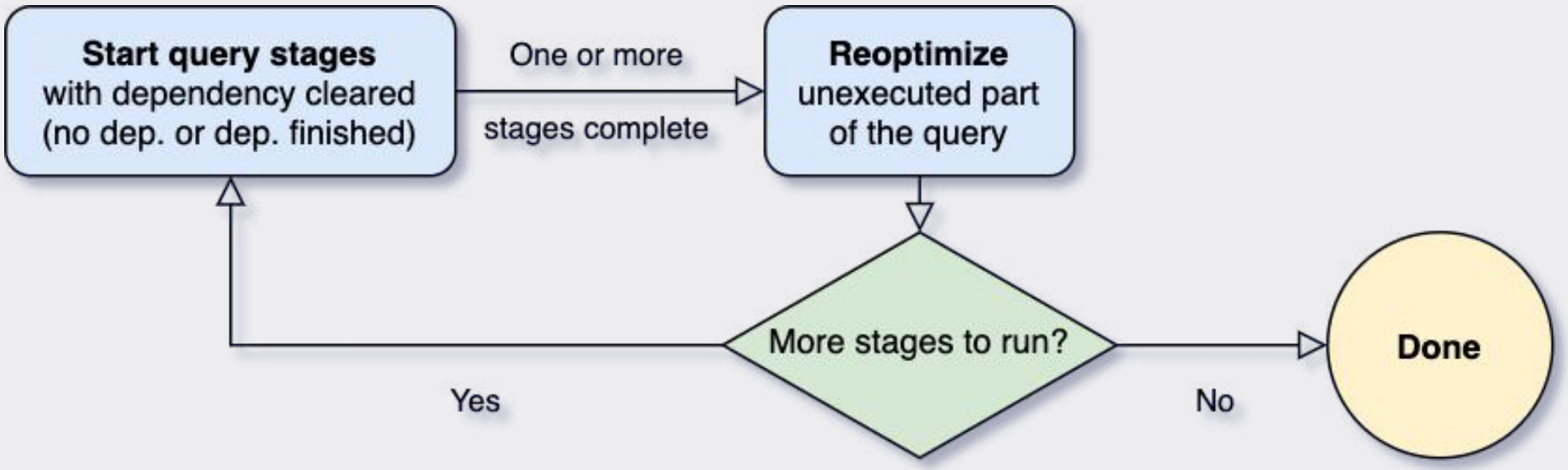
“Don’t make the assumption that things in Spark just work, there is a good chance that Spark underneath the hood is going to do something unexpected”

Holden Karau
Apache Spark PMC

2017, BeeScala



@adipolak

What can go wrong with Data flows?

EVERYTHING

- new component logic
- new data source
- introducing incompatible schema change
- Spark job runs twice, in parallel
- changing tables' relationship keys
- accidentally delete yesterday's `events/` partition
- data duplication

Testing is hard!

Distributed data systems with many moving parts

Mat Ryer
@matryer

I practice 'disagree and commit'.

I disagree with the failing tests and commit the code anyway.

3:03 AM · Jun 13, 2022 · Twitter Web App

89 Retweets **14** Quote Tweets **1,081** Likes

Unit testing

Integration testing

System testing

E2E ...

DATA+AI SUMMIT 2022

Chaos Engineering in the World of Large-Scale Complex Data Flow

Adi Polak
Treeverse

ORGANIZED BY databricks

@adipolak

Chaos Engineering

4 Principles:

- Defining a steady-state
- Acknowledging a variety of real-world events
- Running manual experiments in production
- Automating production experiments

@adipolak

#1 – Steady state

DevOps / SRE

- system's throughput
- error rates
- latency percentiles
- etc

Data Engineer

- Data products requirements

Data Product Requirements

- Data Quality
- Accuracy
- No duplications
- SLA

@adipolak

Data Duplication

In 3 lines of code..

```
1 # Read data from file
2 df = spark.read.parquet('s3a://bank_transactions/ts=123123123/')
3 # Some transformation over the data related to finance
4
5 updated_df =
  updated_df.write.mode('append').parquet('s3a://bank_transactions/ts=123123123/')
6
7
8
9
10
```

#2 - Vary Real-world Events

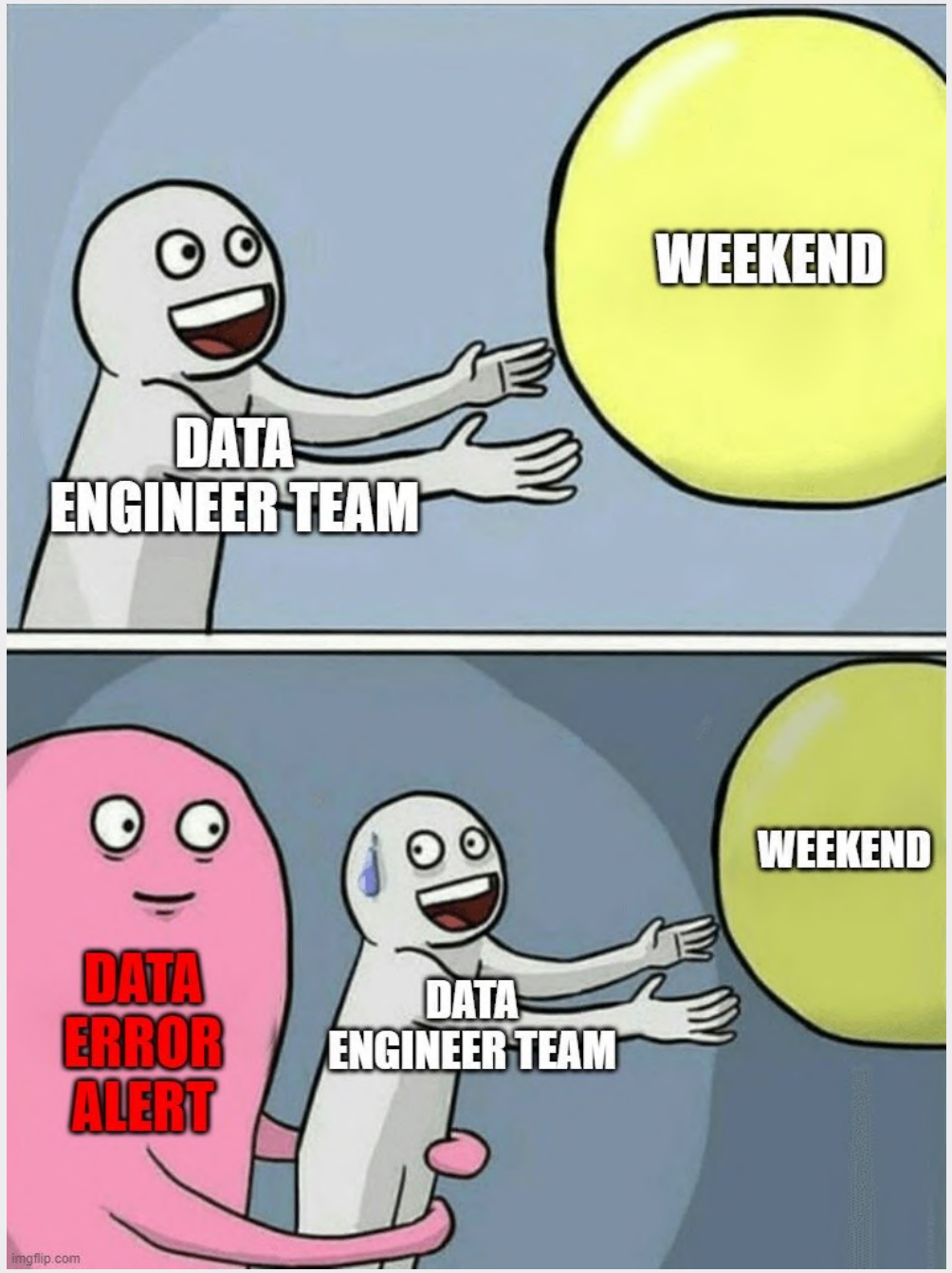
DevOps / SRE

hardware failures like :

- servers dying
- spike in traffic

Data Engineer

- schema change
- corrupted data record
- data variance
- accidentally delete yesterday's `events/` partition


WEEKEND
DATA
ENGINEER TEAM
DATA
ERROR
ALERT
DATA
ENGINEER TEAM
WEEKEND
imgflip.com

@adipolak

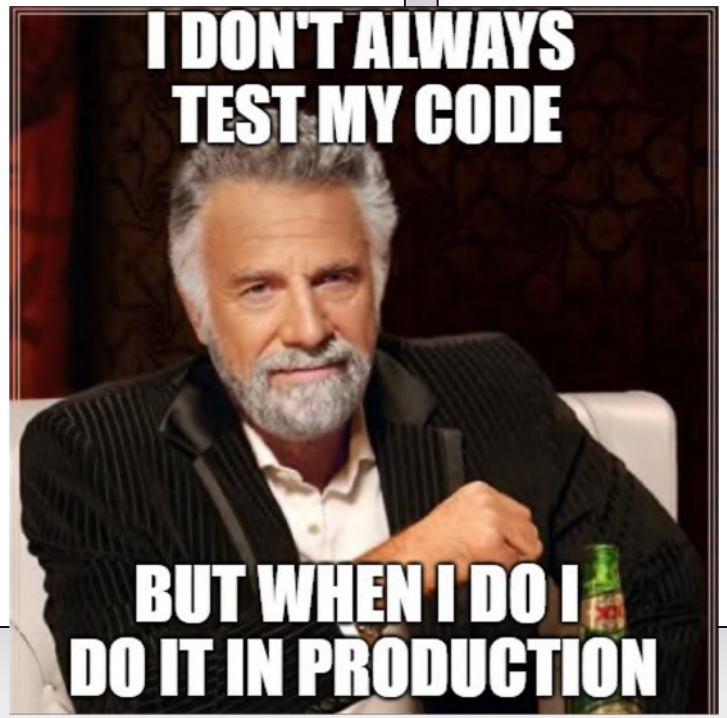
#3 – Run Experiments in Production

DevOps / SRE

- Production machines and network

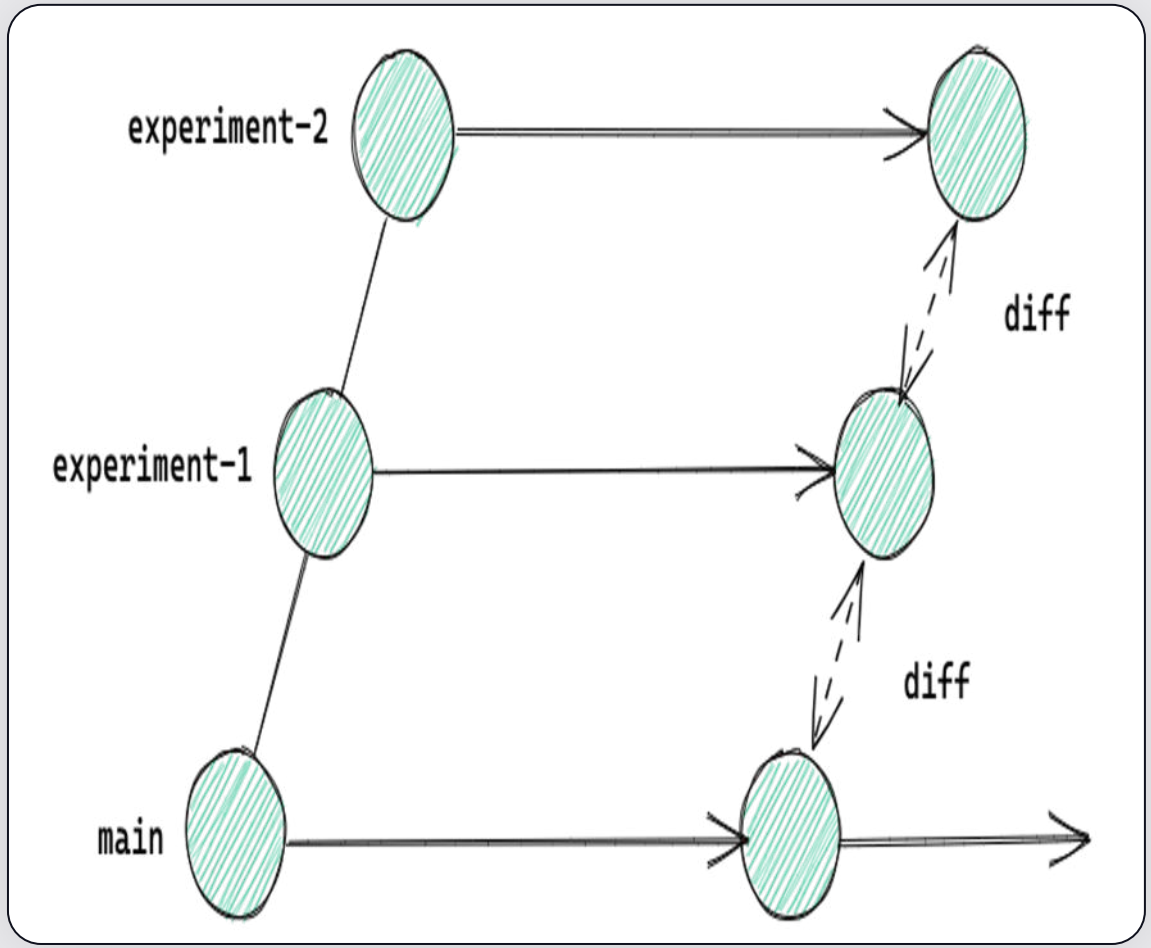
Data Engineer

- Production data



Experimental environment

On production data in production ..



s3://data-repo/collections/foo

s3://data-repo/main/collections/foo

```
lakectl branch create \
      lakefs://repo@testing-spark-3 \
      --source lakefs://repo@main


# output:
# created branch 'testing-spark-3.0',
# pointing to commit ID:
'd1e9adc71c10a'
```

@adipolak

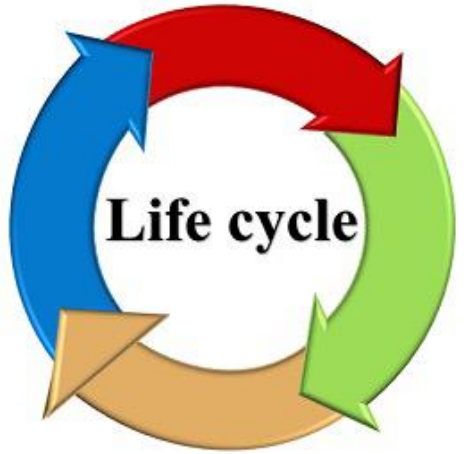
#4 – Automate Experiments to Run Continuously in Prod & Minimize Blast Radius

DevOps / SRE

- Discover weaknesses

Data Engineer

- Manage Data lifecycle in stages



Production data stages

Development

- Experimentation
- Debug
- Collaborate

Deployment

- Version control
- Best Practices & Data Quality

Production

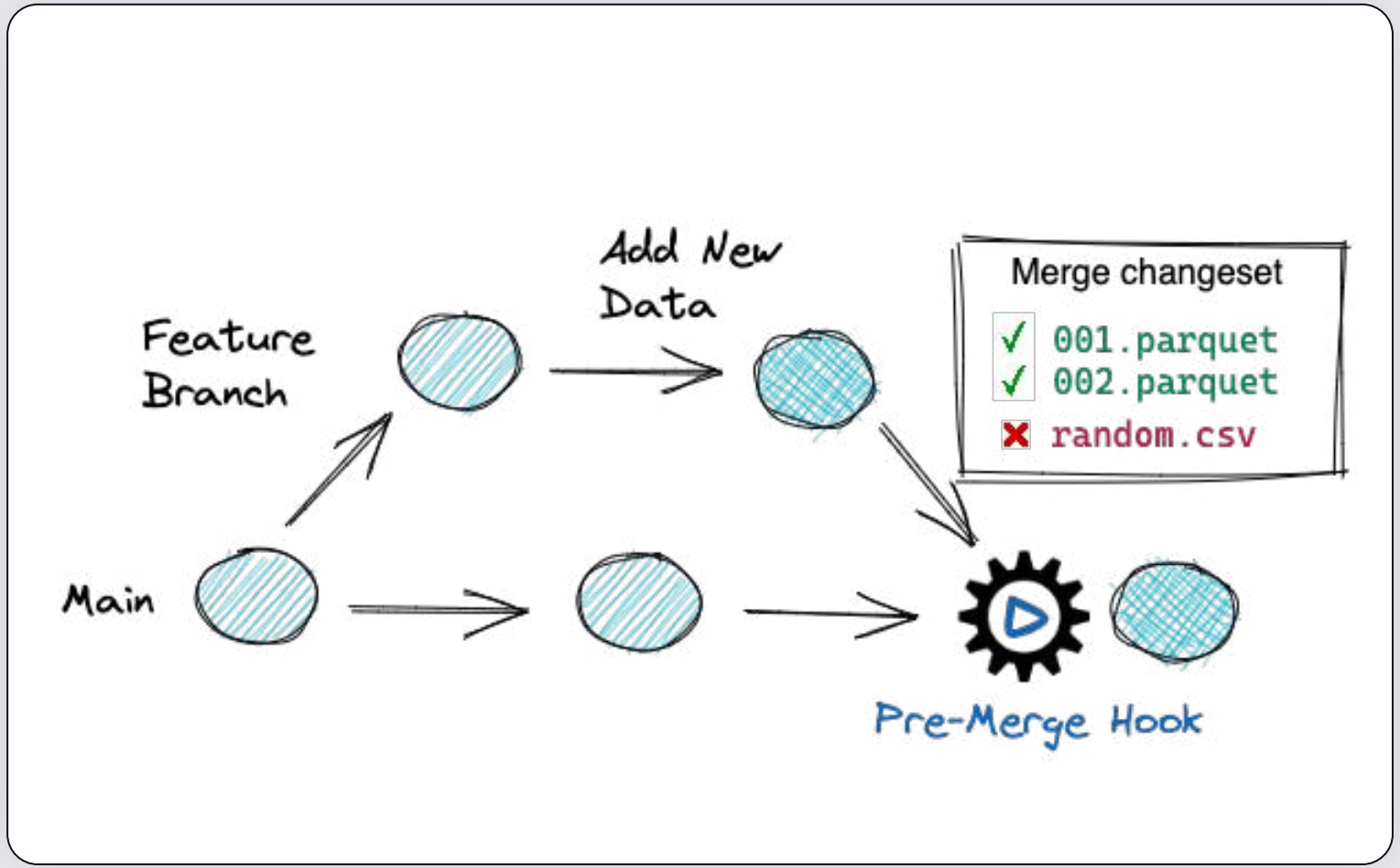
- Roll Back
- Troubleshoot

What's the best way to automate data stages propagation in production?

@adipolak

Branching Strategy

Like source control -



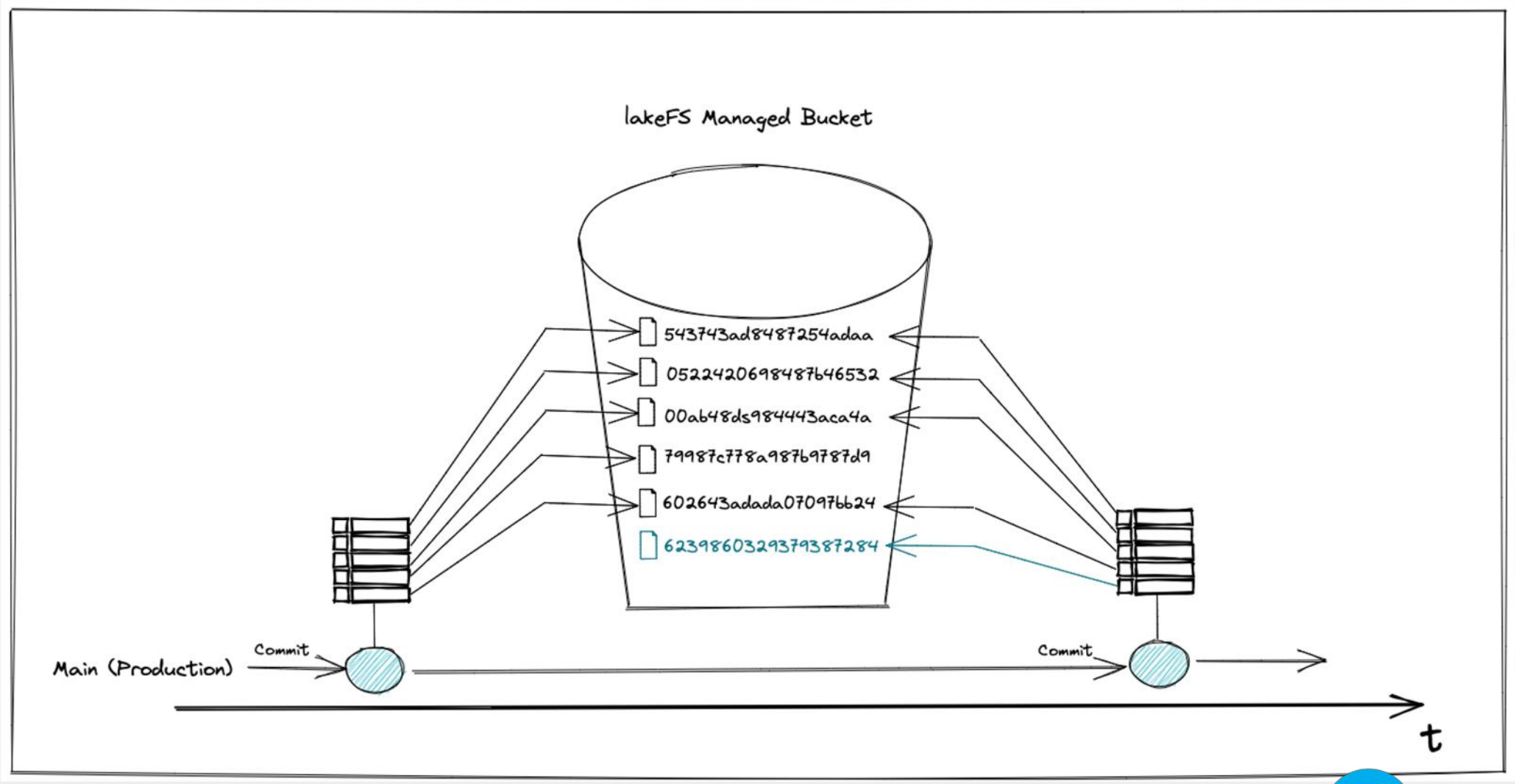
@adipolak

Git for Data

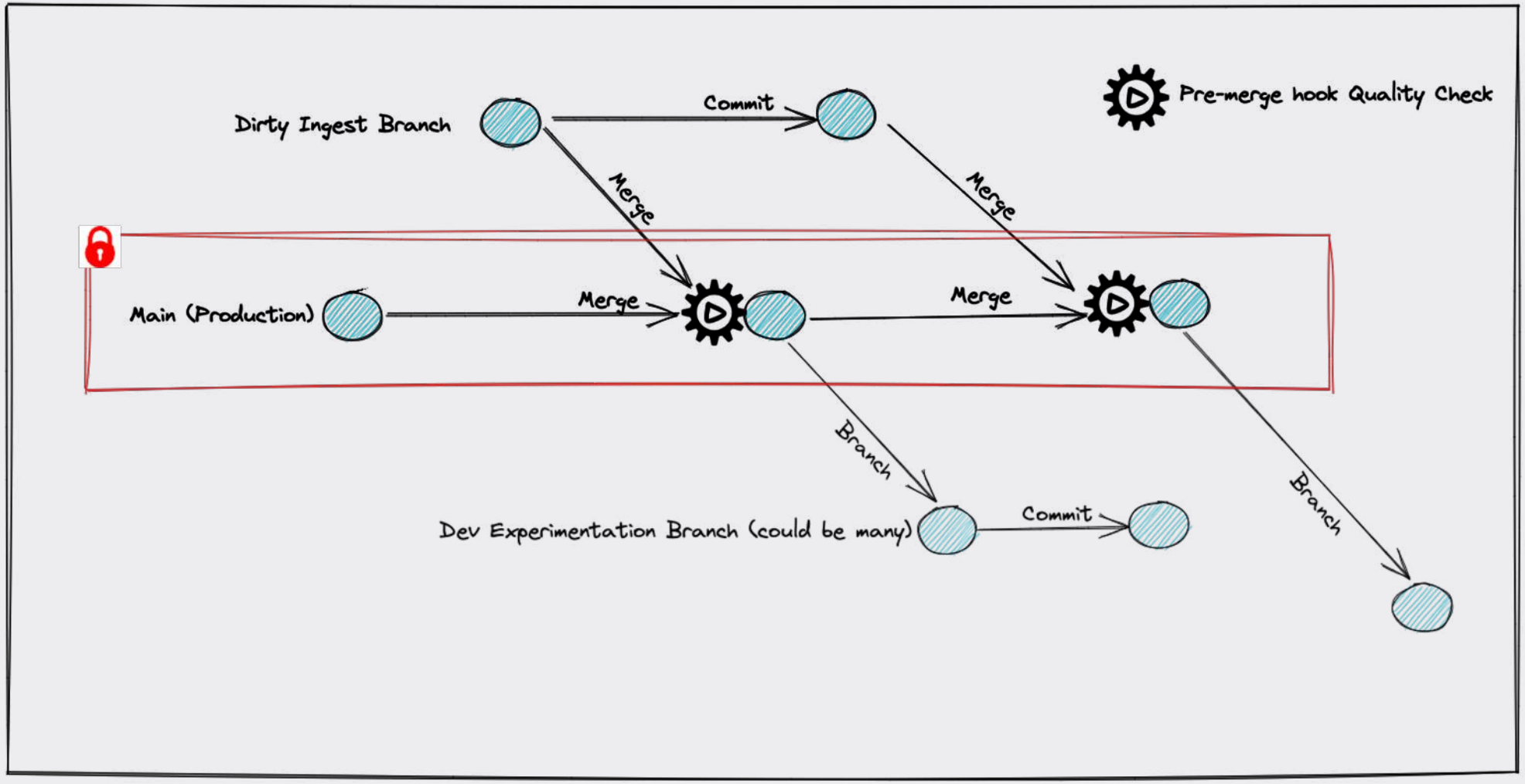
lakeFS – Git for Data

Challenge	Solution
Quickly recover from an error	$ lakectl **revert** main^1
Develop in Isolation	$ lakectl **branch create** my-branch
Troubleshoot	$ spark.read.parquet ('s3://my-repo/<**commit_id**>')
Atomic Update	$ lakectl **merge** my-branch main
Reprocess	$ lakectl **branch create** new-logic $ lakectl **merge** new-logic main

How Does lakeFS Work?



What Does A Typical Environment Look Like?



Cross-Data Collections Consistency with lakeFS

DATA+AI SUMMIT 2022

Demo

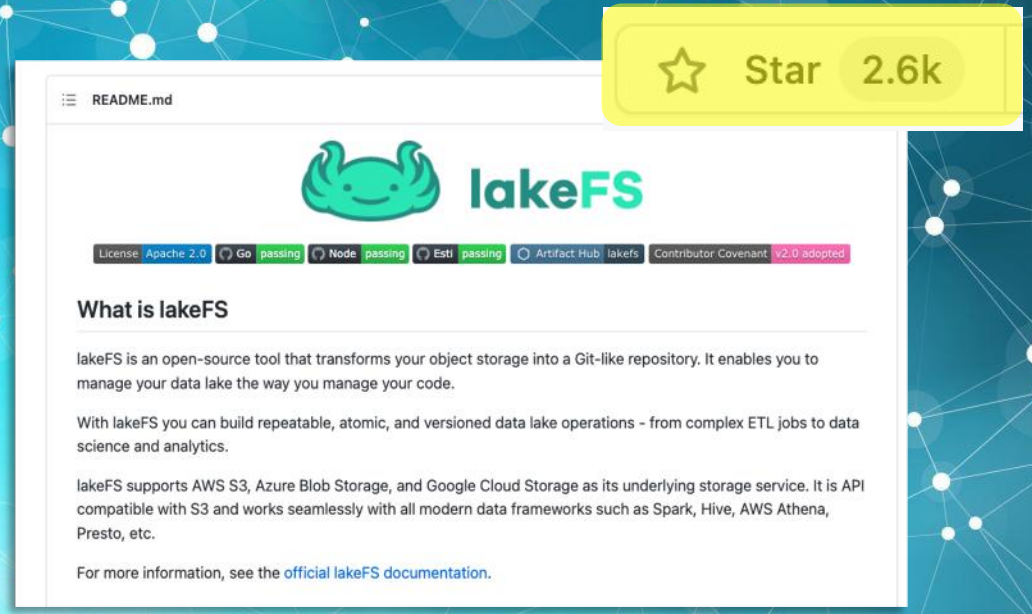
Recap

- Adopting Principles of Chaos Engineering to data systems
- Data Lifecycle Management stages
- Git for Data
- lakeFS

lakeFS Community

Trusted by more than 1K Companies

By more than ***4K*** members



redhat
Walmart
yotpo.
General Motors
AppsFlyer
Zillow
intel
NASA
Adobe
VOLVO
Airbnb
Shopify
WiX
Shell
ORACLE