

Using Chaos Engineering to Build a Reliable TiDB



Tang Liu (@siddontang) at PingCAP

Chief Engineer of PingCAP

Leader of TiKV

Open sources:

go-mysql, raft-rs, grpc-rs, etc...

About Me

Why Chaos ?

Chaos Engineering

Practice Chaos in TiDB

Chaos in Cloud

Agenda

Why Chaos ?

Why Chaos ?

Chaos Engineering

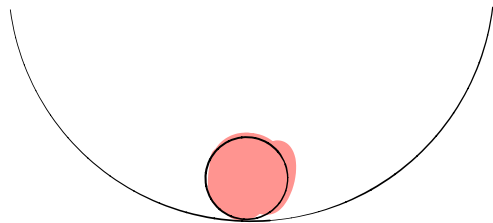
Practice Chaos in TiDB

Chaos in Cloud

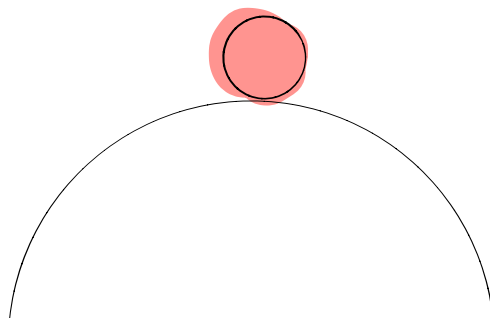
System

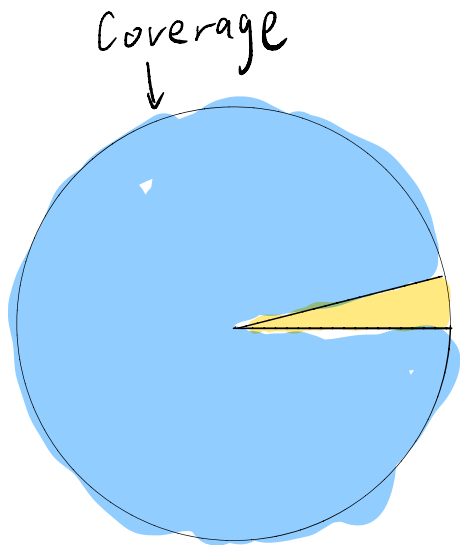


stable

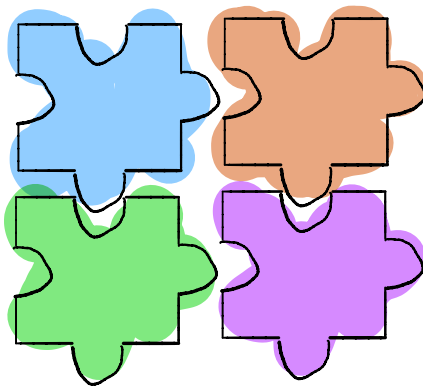


Unstable

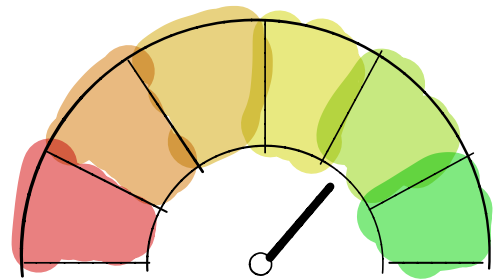




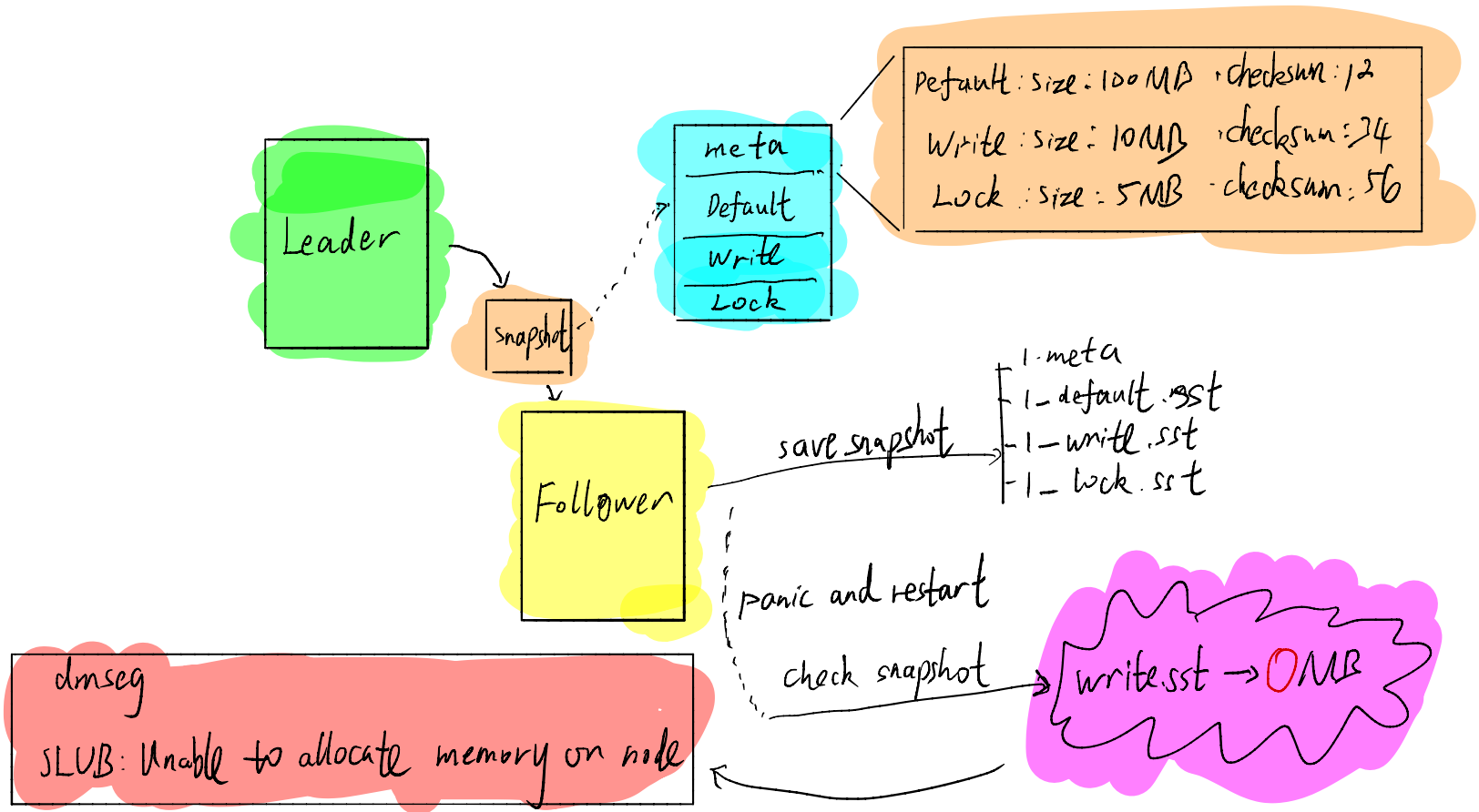
Unit Testing



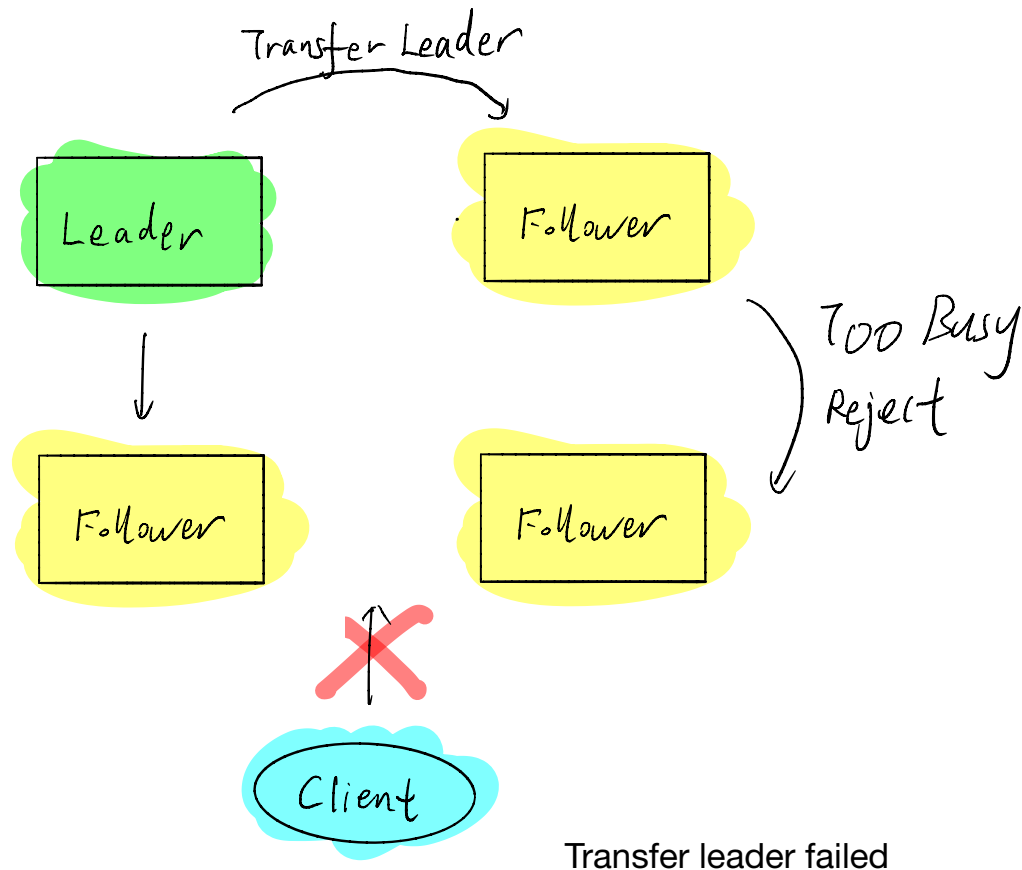
Integration Testing

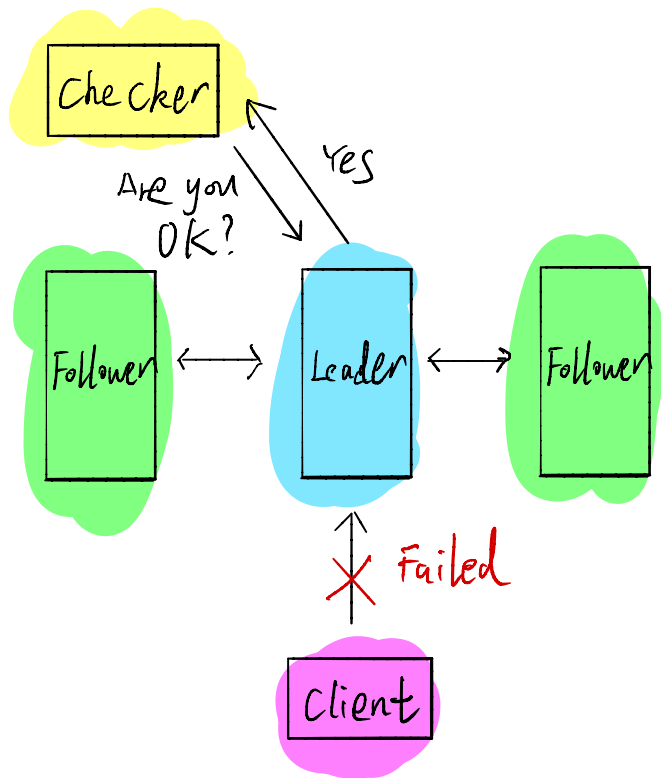


Performance Testing

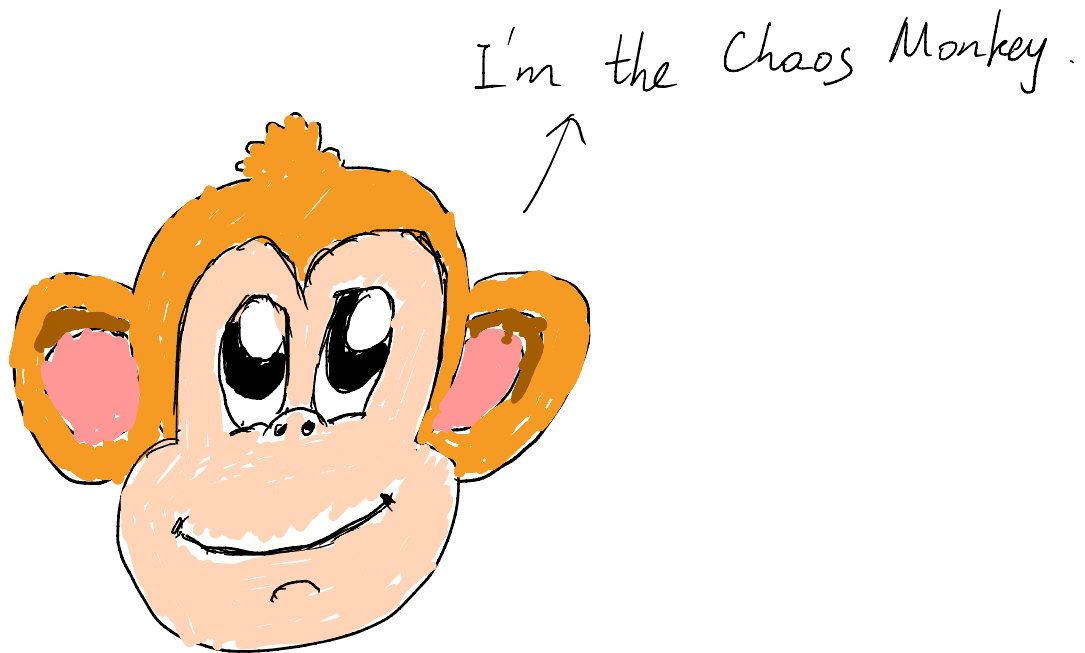


Snapshot file is empty!!!





Gray Failure



Chaos Monkey

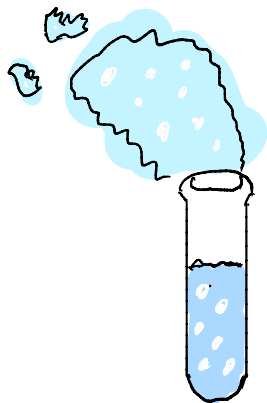
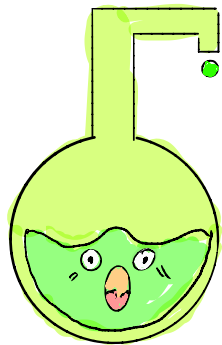
Chaos Engineering

Why Chaos ?

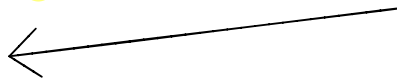
Chaos Engineering

Practice Chaos in TiDB

Chaos in Cloud



Do Experiments



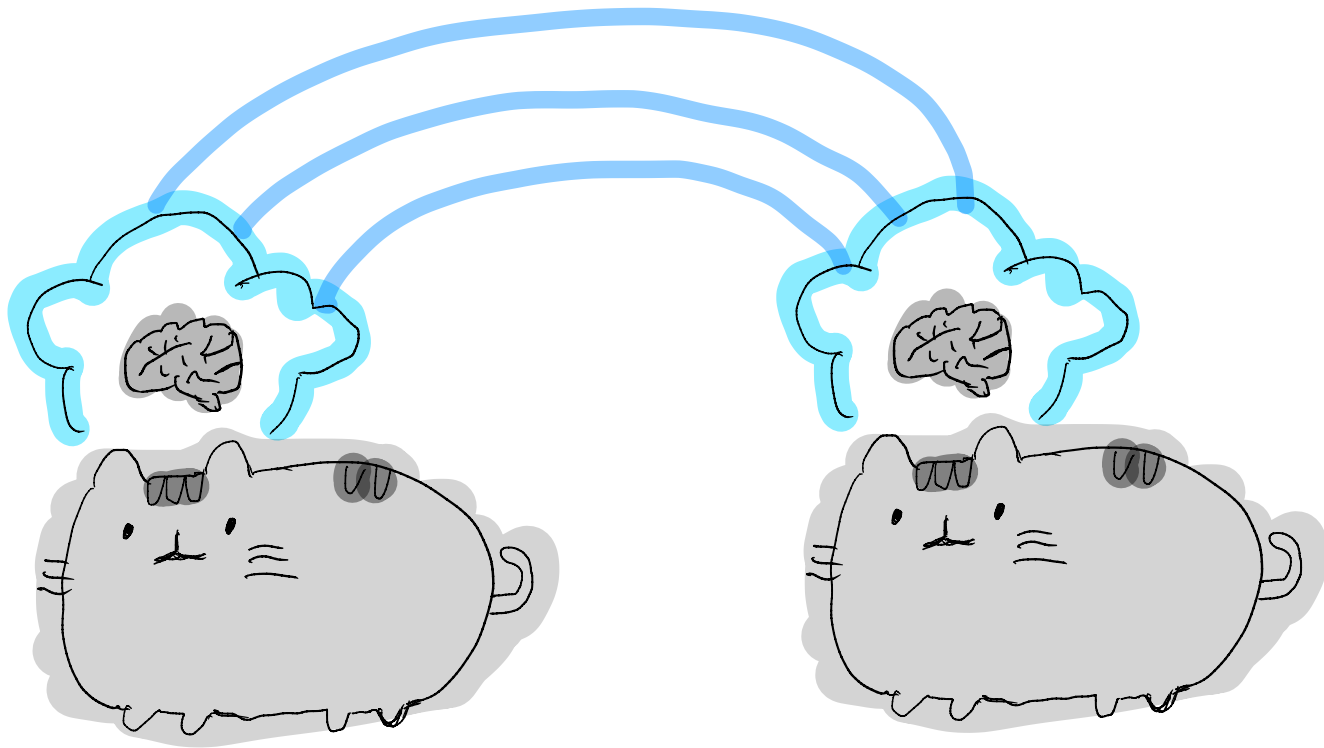
Observe the behavior



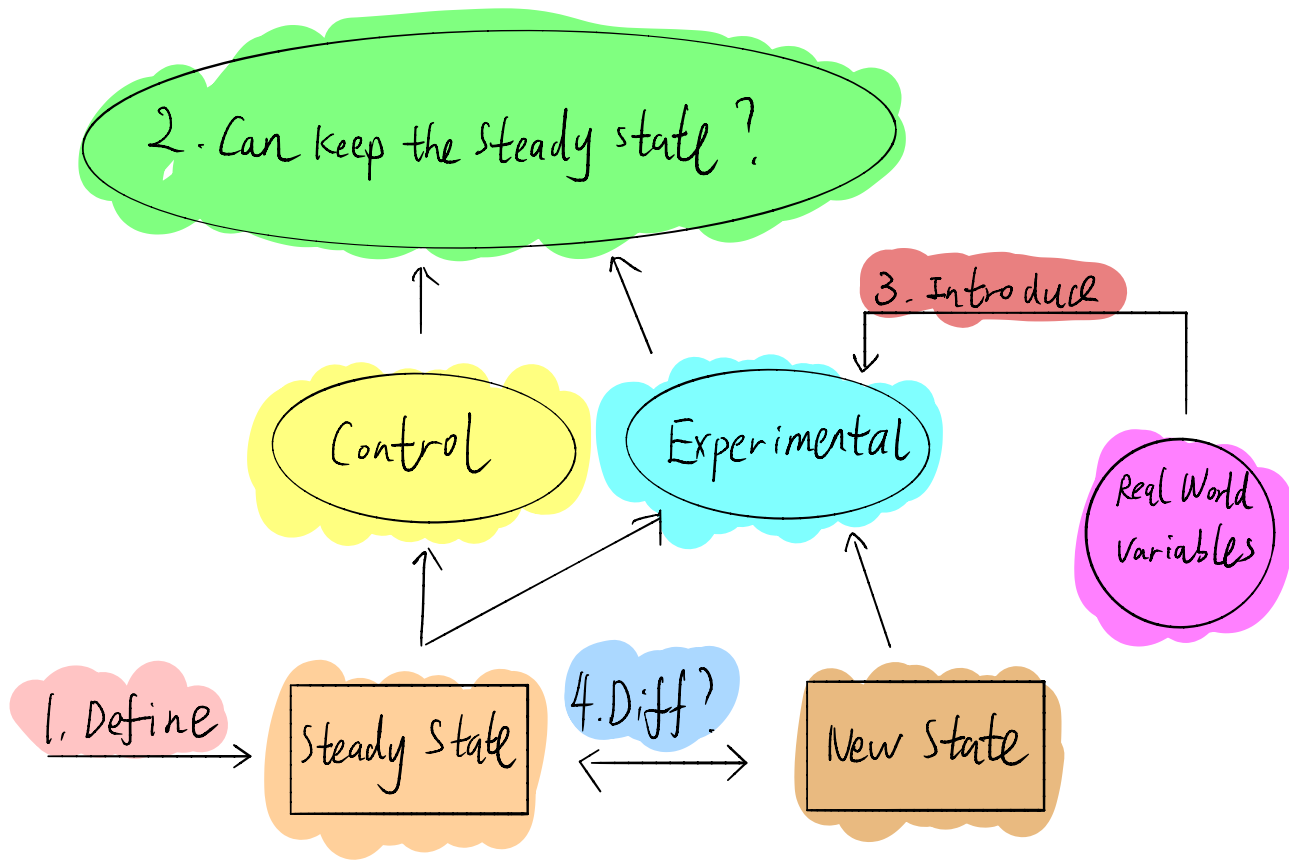
Experimental

Control

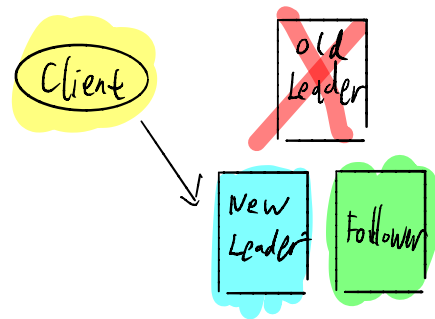
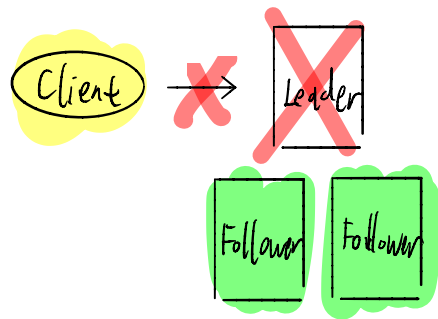
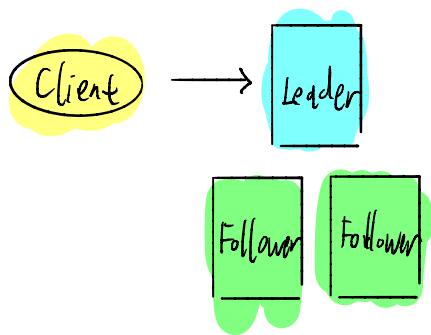
Chaos Engineering



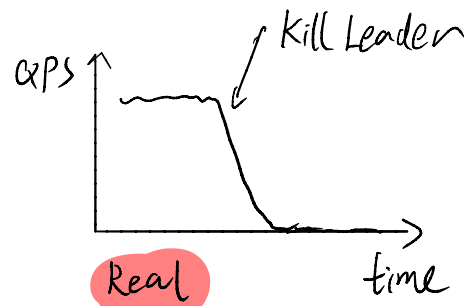
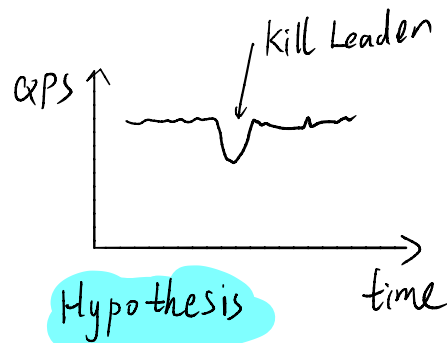
Fault Tolerance



Four steps of Chaos



1. steady state : QPS
2. Assumption : QPS keeps stable
3. kill Leader
4. check QPS



Build a hypothesis around Steady State Behaviour



Vary Real-world Events



Run Experiments in Production



Automate Experiments to Run Continuously



Minimize Blast Radius

Chaos Principles

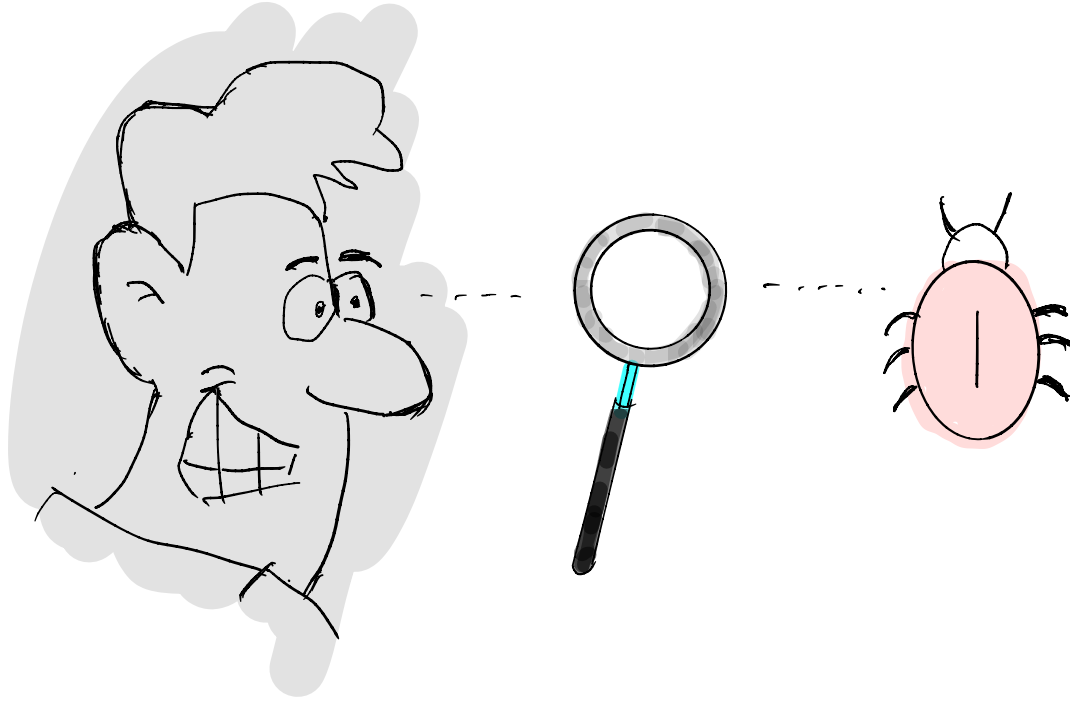
Practice Chaos in TiDB

Why Chaos ?

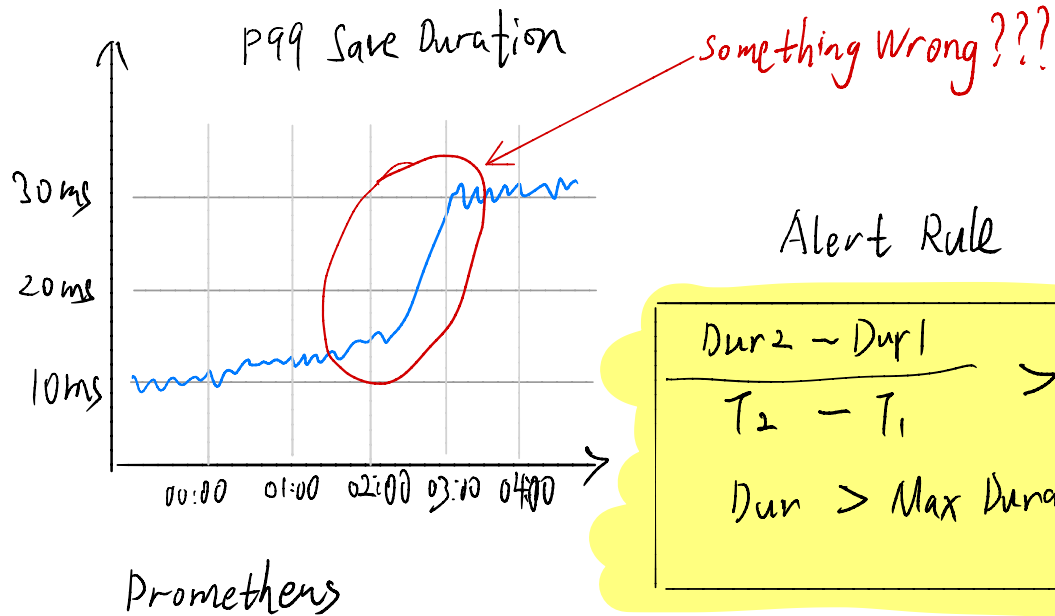
Chaos Engineering

Practice Chaos in TiDB

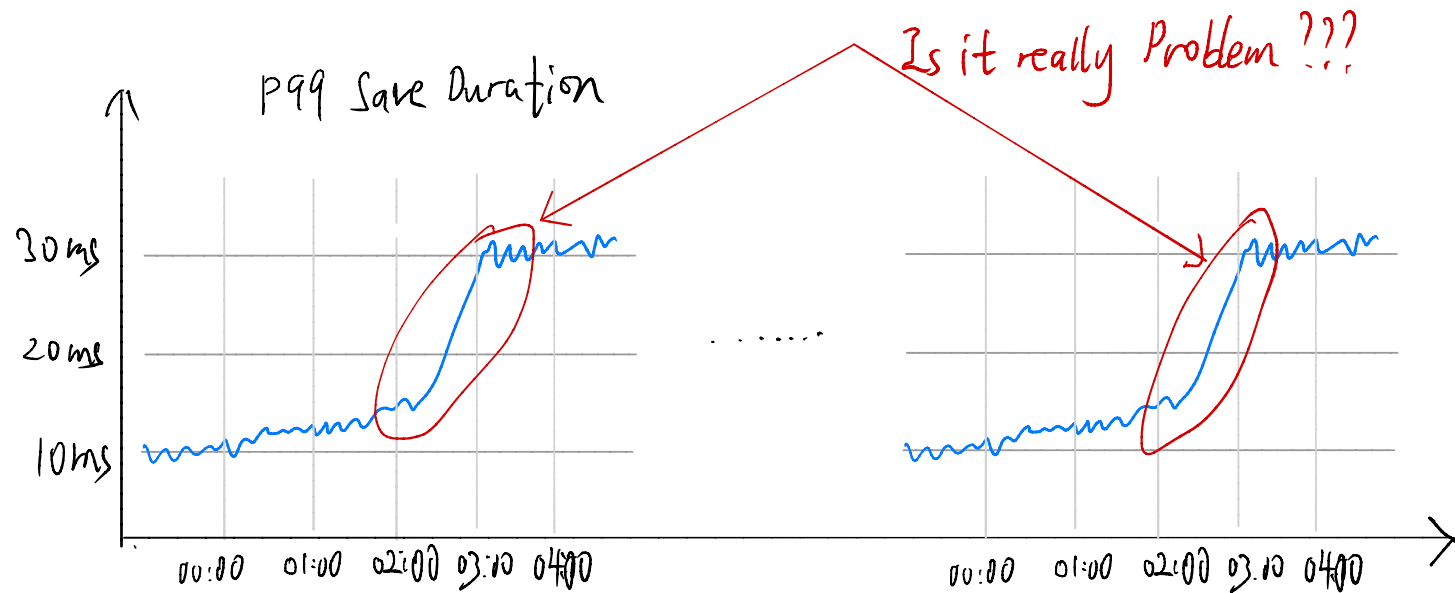
Chaos in Cloud



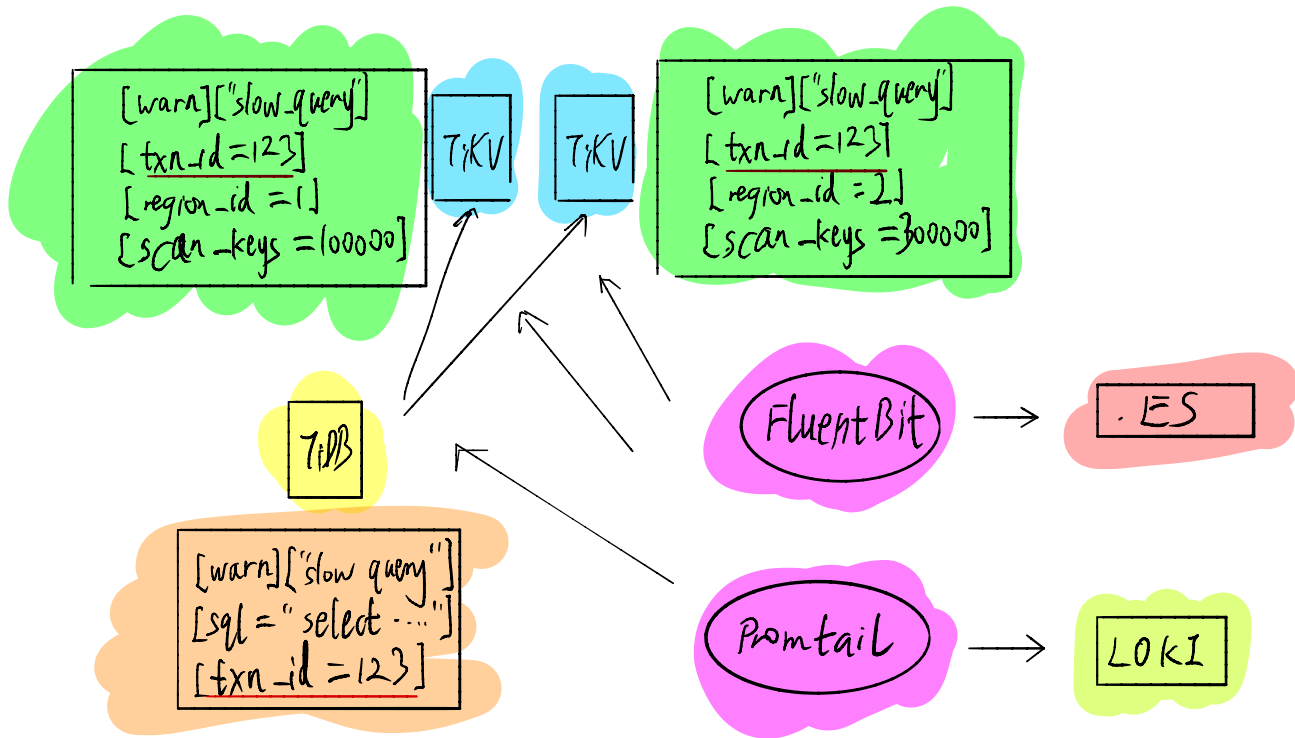
Failure Detection



Find problems from metrics



Learn from history



Learn from log

Load

Get

RPC

RPC

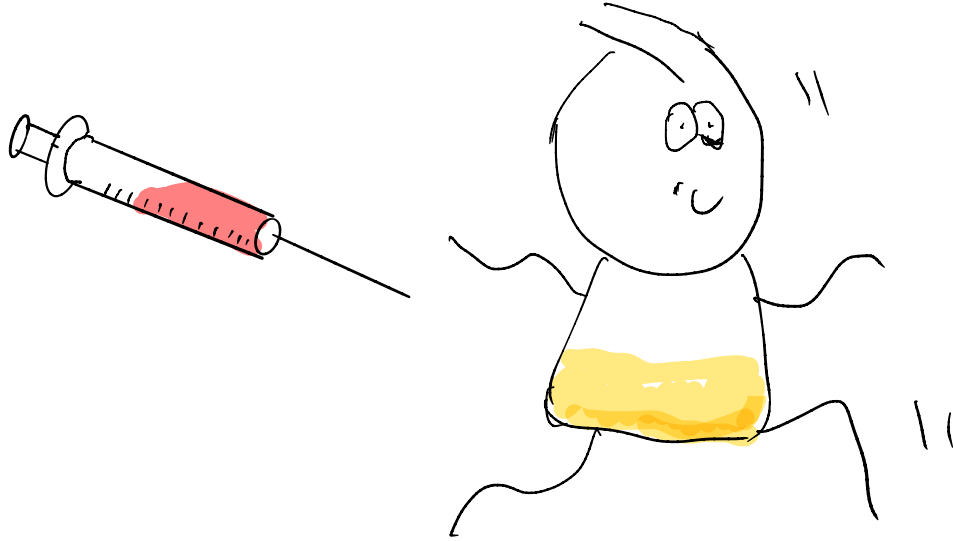
RPC

API

API

API

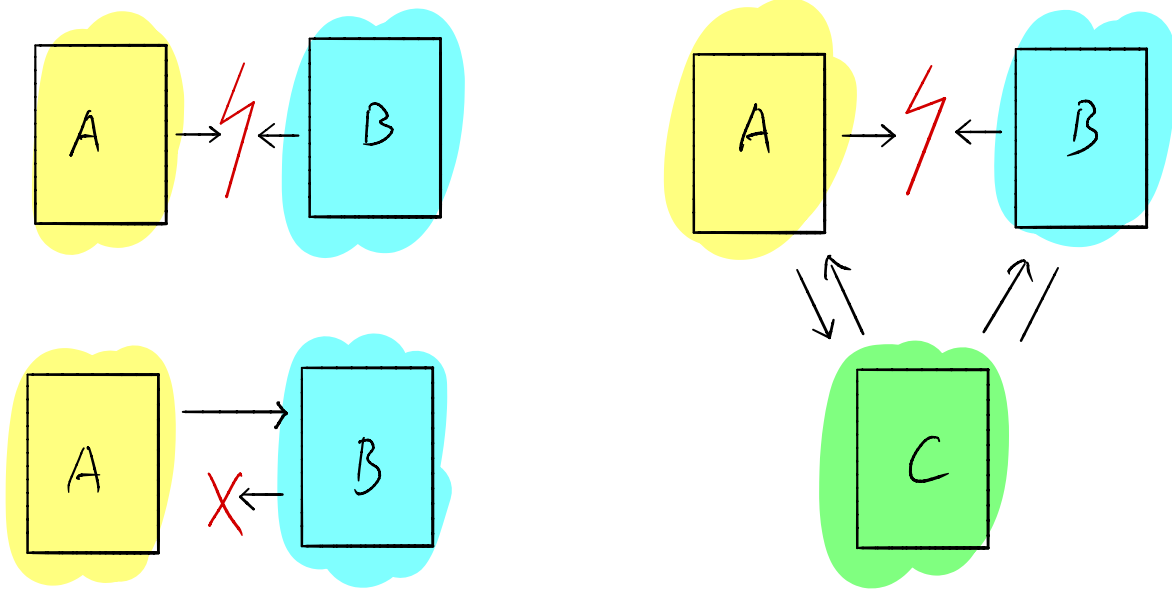
Tracing



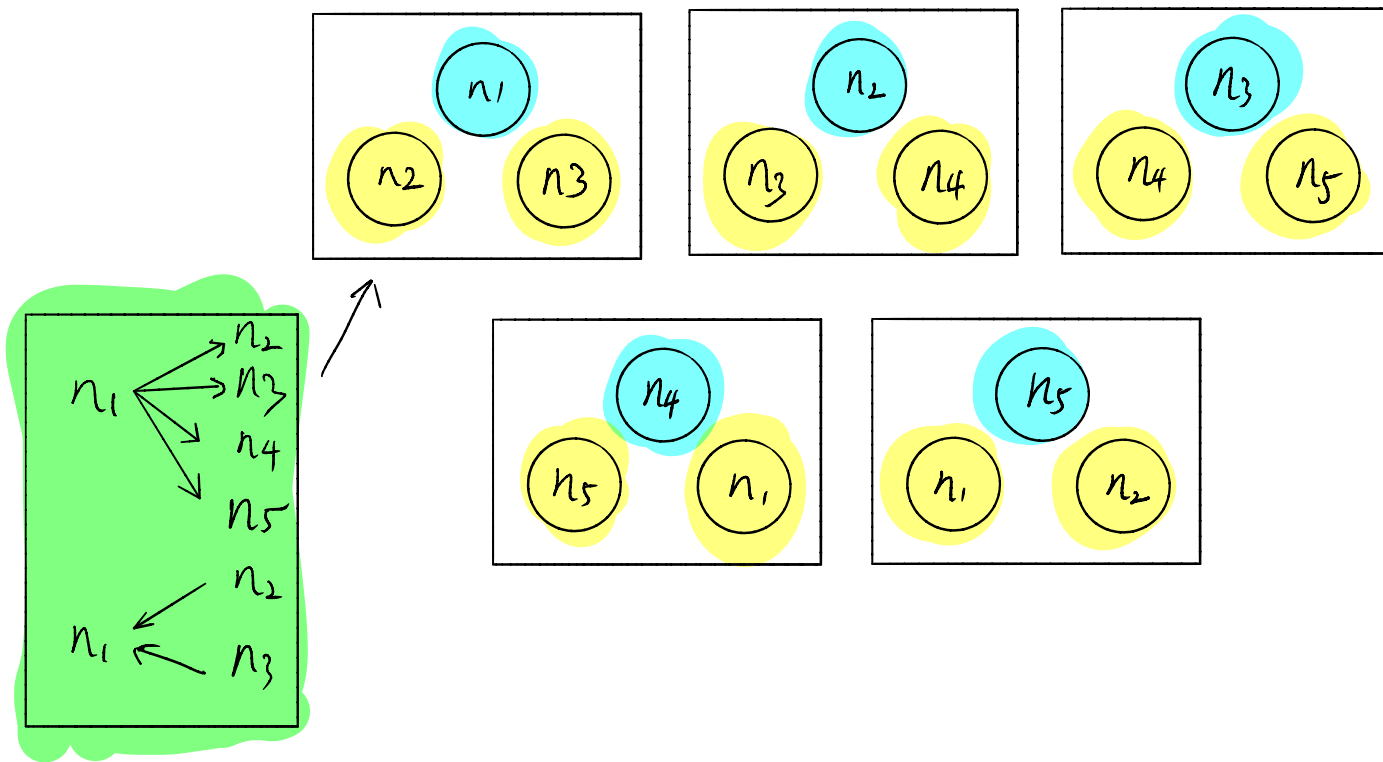
Fault Injection

Application	Stop	Scheduler	OOM	Kill	ClockSkew
MEM	AllocStall	Corruption	AllocFail	Limit	
CPU	ContextSwitch	Preemption	Full	Limit	
Storage	Delay	DiskFull	PowerDown	Corruption	
Network	Drop	Reject	Delay	Reorder	Full

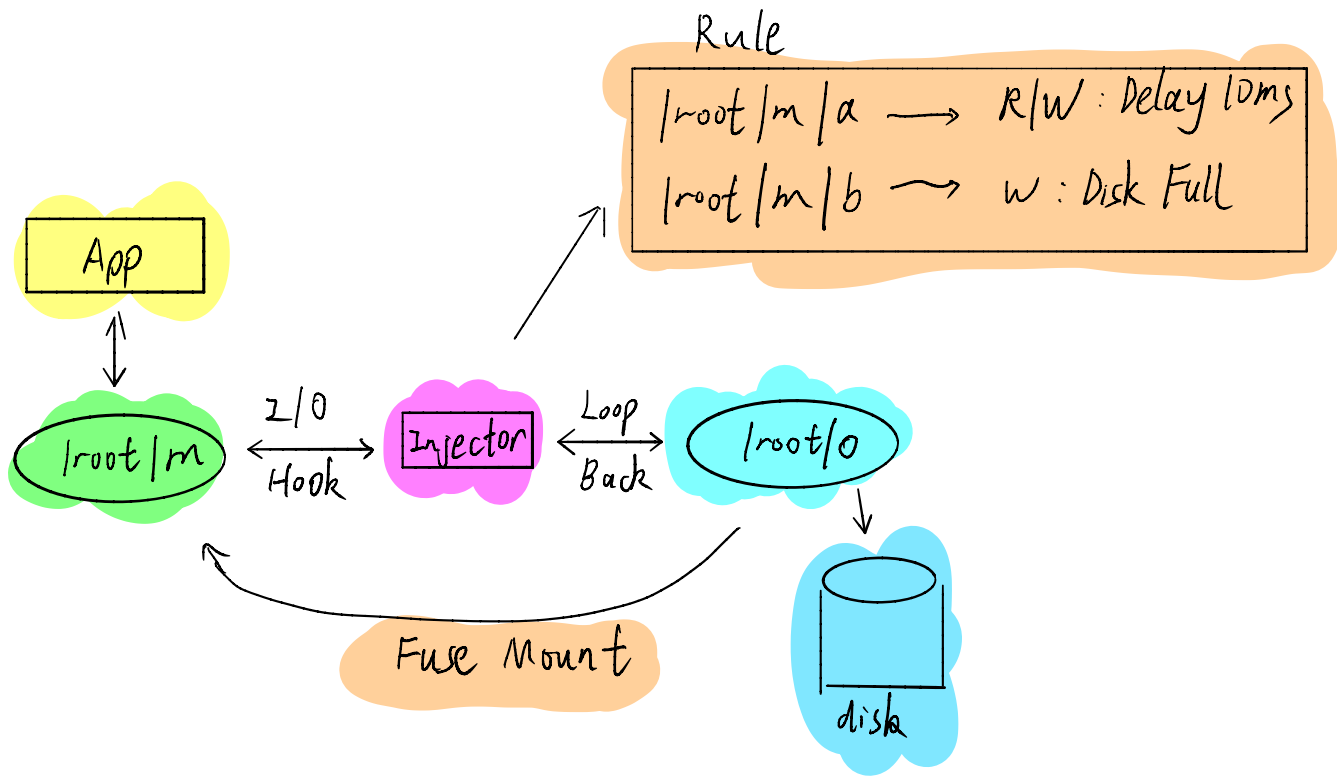
Fault Portrait



Network partition mode



Network partition ring



Fuse

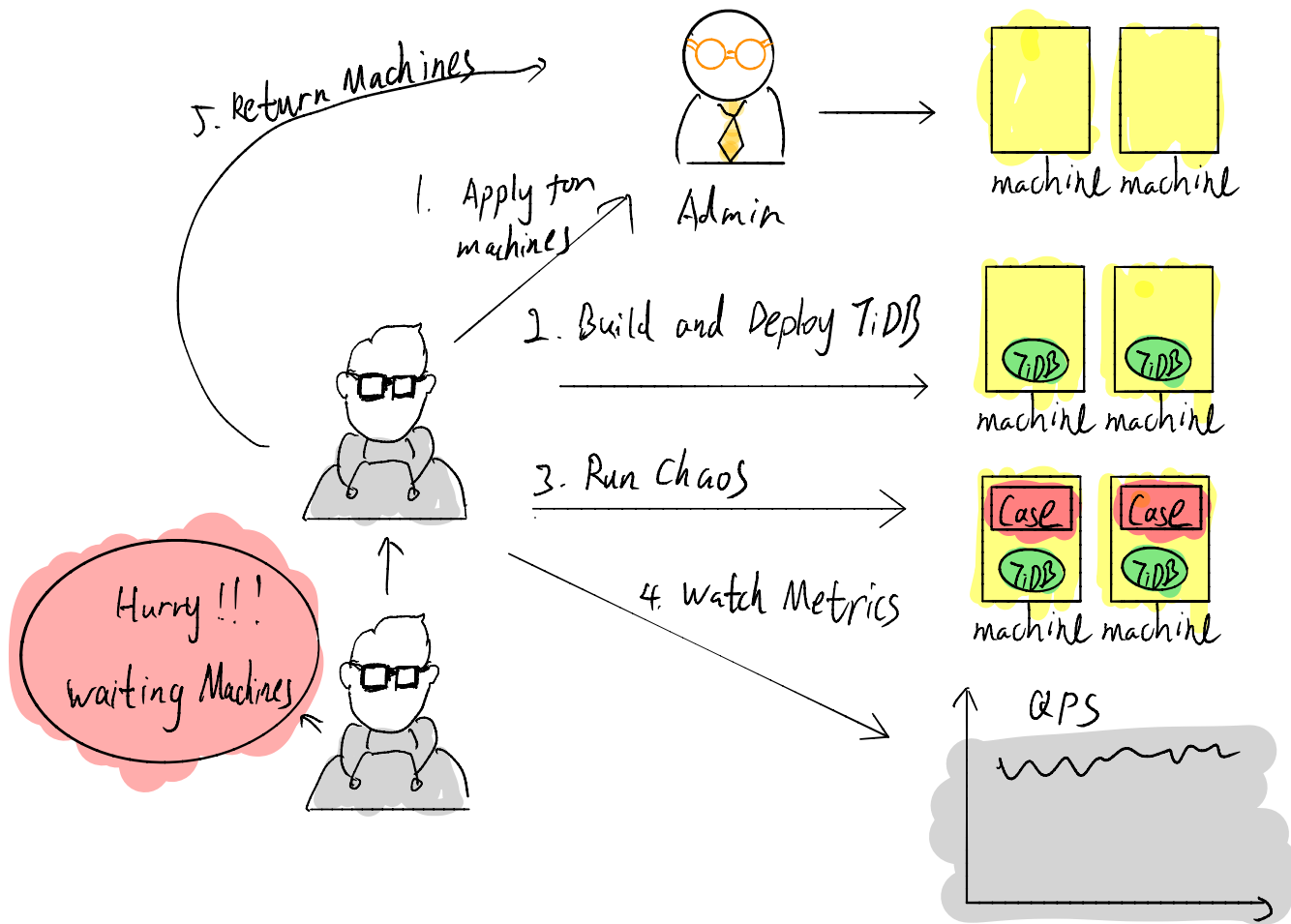
Chaos in Cloud

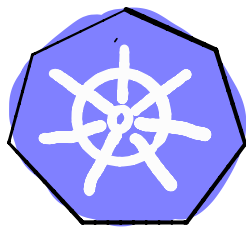
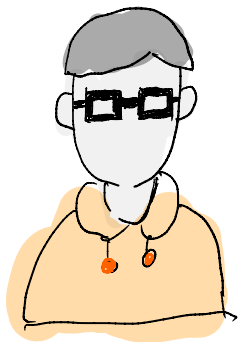
Why Chaos ?

Chaos Engineering

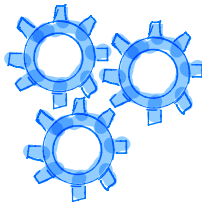
Practice Chaos in TiDB

Chaos in Cloud

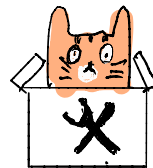




Kubernetes

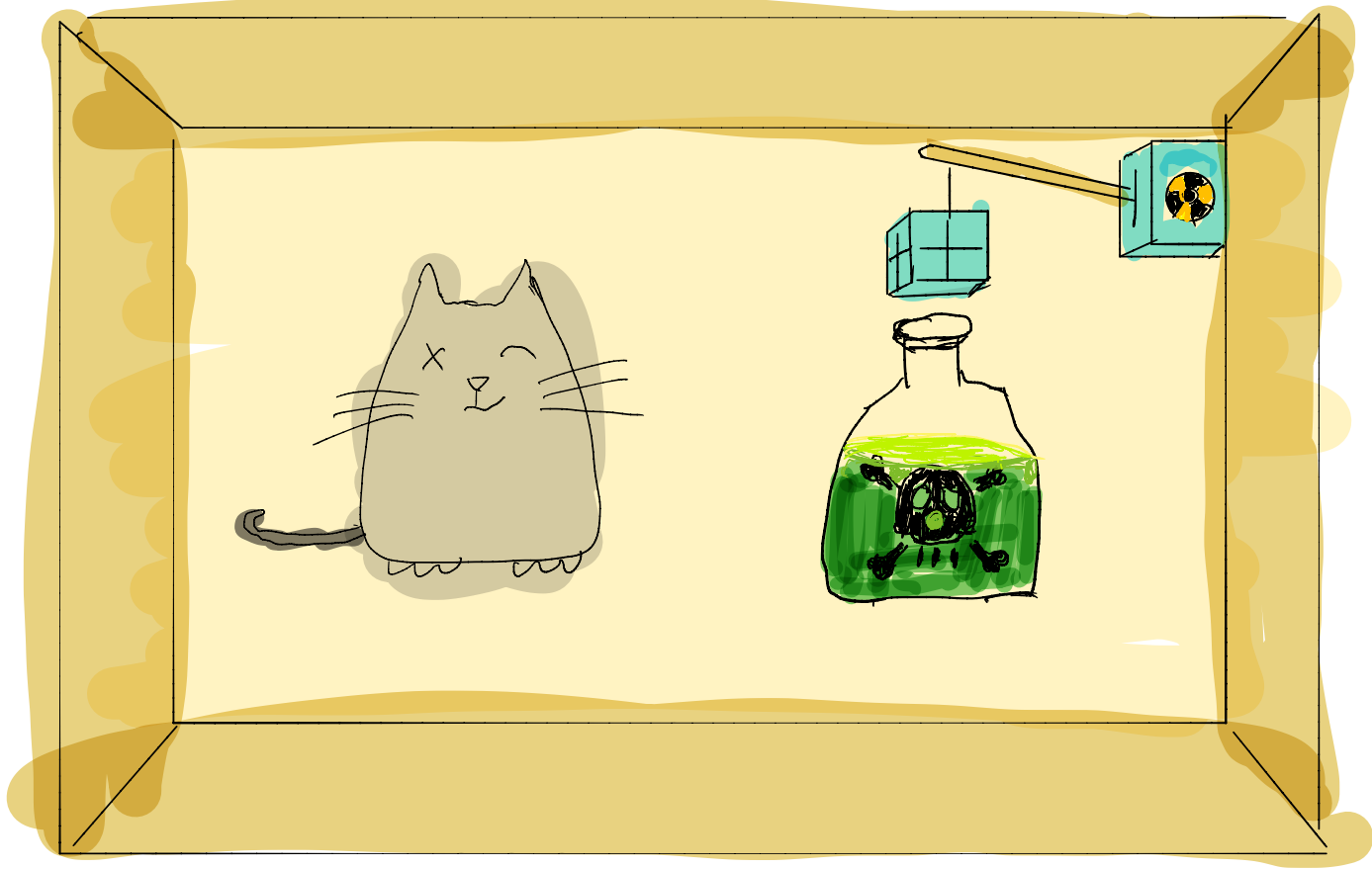


Automation

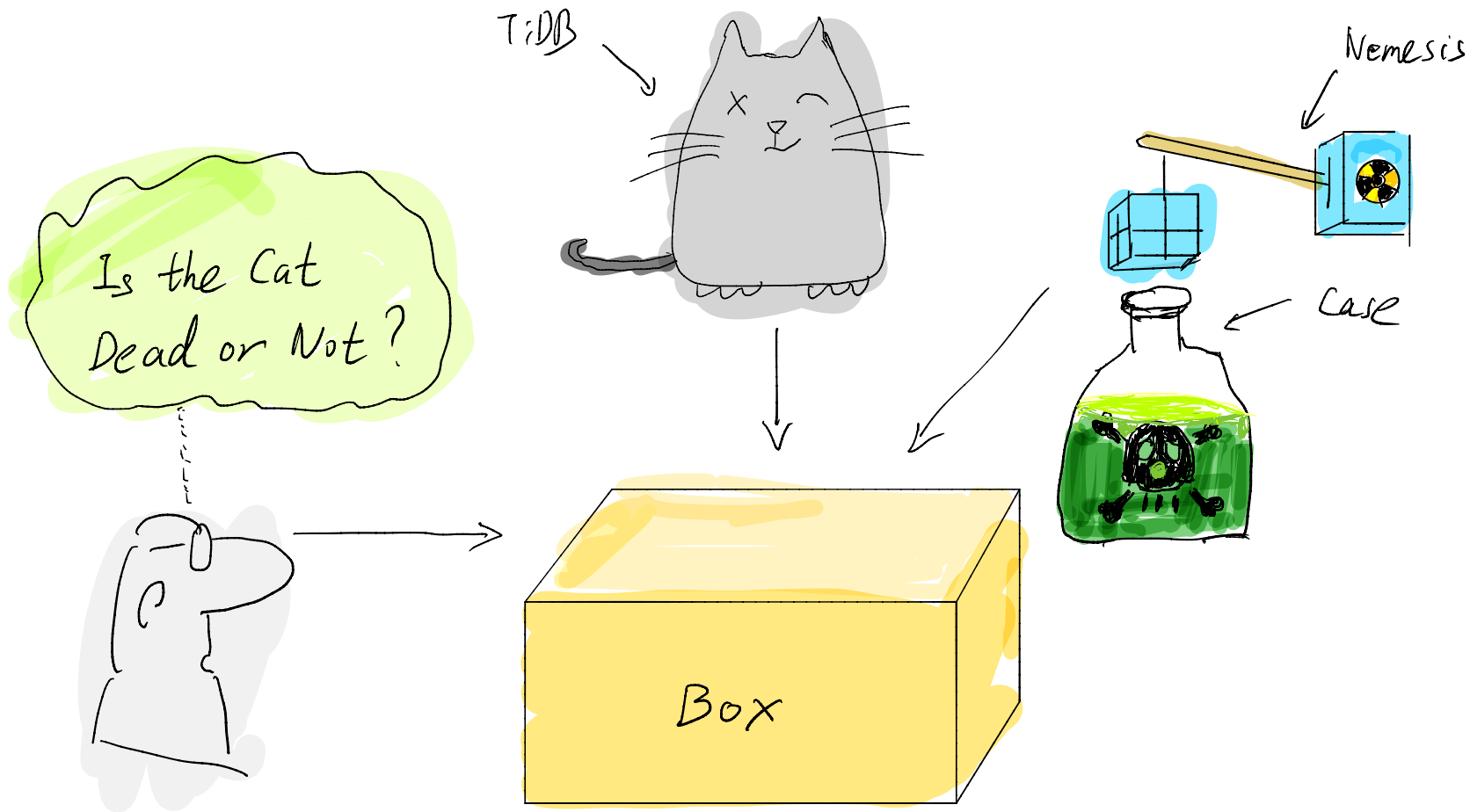


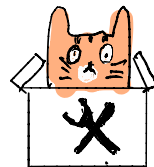
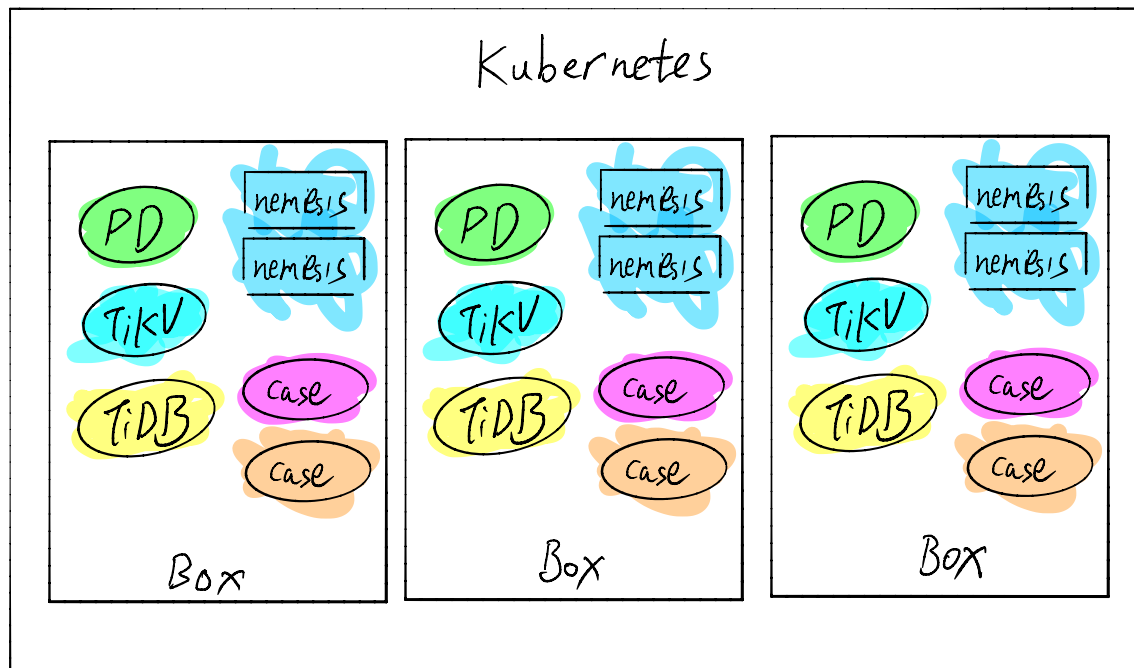
Schrödinger

Schrodinger



Schrodinger's Cat





Schrodinger Architecture

PD

- Size: 3
- Repo: master
- Config: XXX

TiKV

- Size: 10
- Repo: master
- Config: XXX

TiDB

- Size: 2
- Repo: master

PD

- Size: 3
- Repo: Release 2.1
- Config: XXX

TiKV

- Size: 6
- Repo: Release 2.1
- Config: XXX

TiDB

- Size: 2
- Repo: Release 2.1

Kill PD, start 10s later
Hang TiKV 30s
Isolate TiKV $\leftrightarrow$ TiDB
Delay TiKV Read 30ms
Max out PD CPU

Nemesis

Bank:

Init: 1000 users
1000 \$ each

Action:

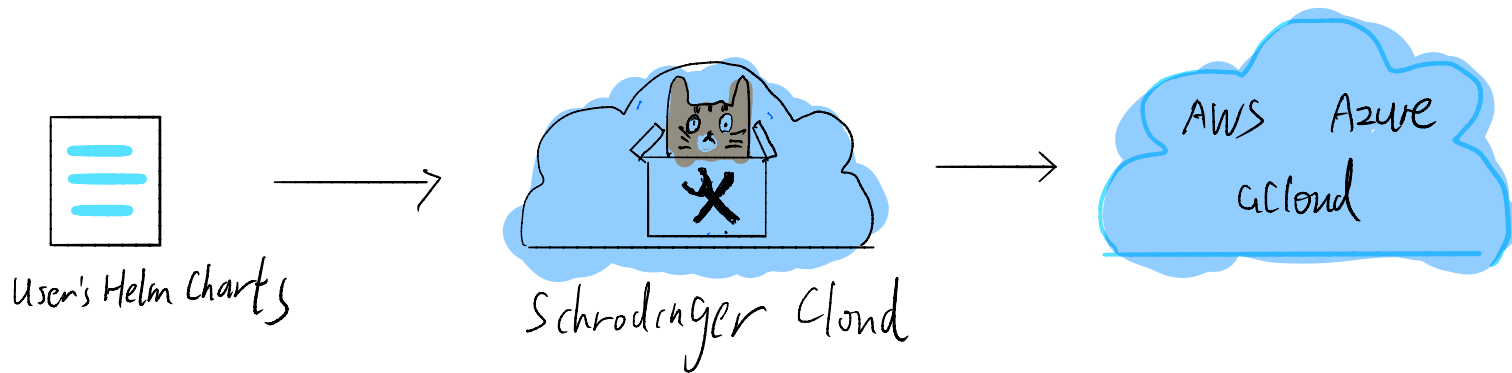
Transfer money

check:

sum(all) = 1,000,000

Case

Ray & Prison



Schrodinger Cloud

kubernetes

Helm

Helm Chart

Operator

Pod

Container

Daemonsets

Linux

Package Manager - RPM.

RPM File.

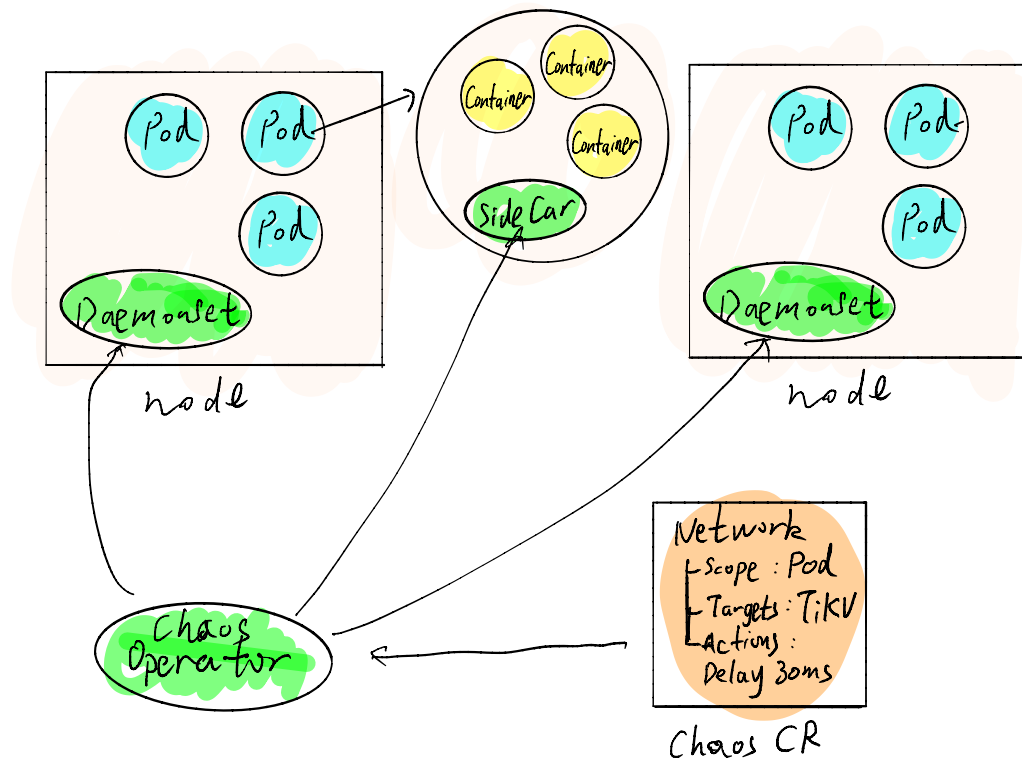
systemd

Process

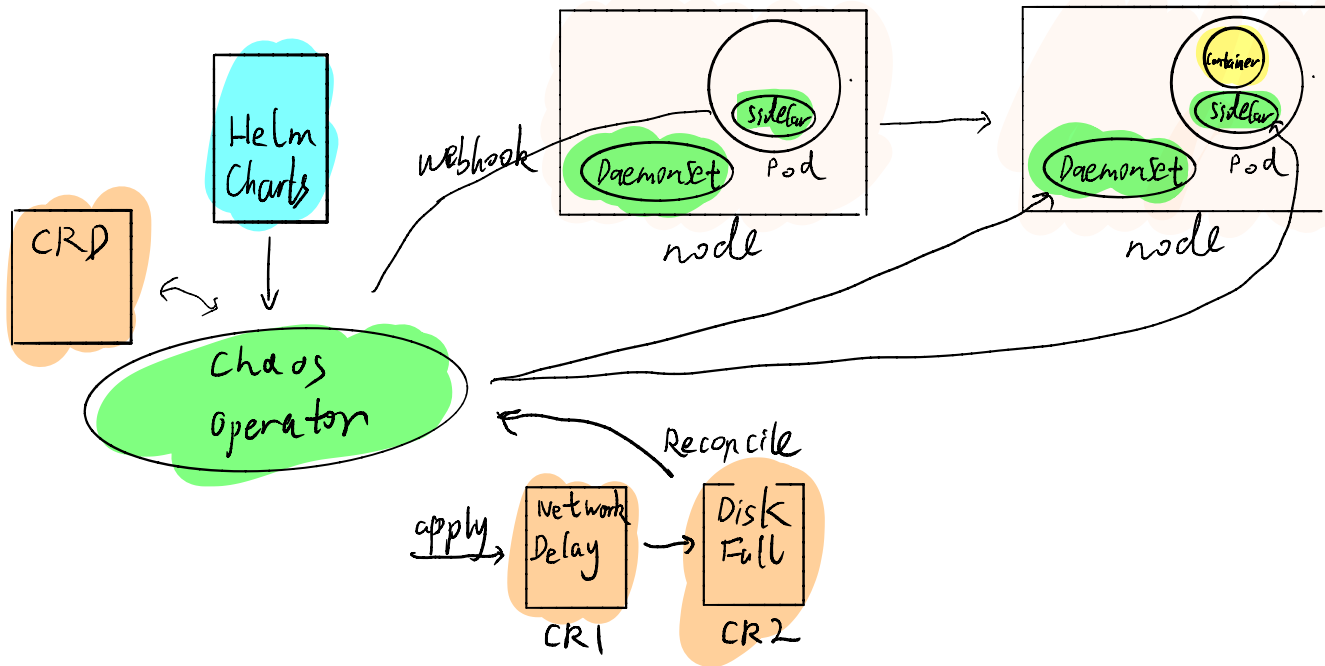
Thread

Daemon

Concepts Mapping



Chaos Operator



Run with your own Helm charts



Email: tl@pingcap.com

Twitter: [siddontang](#)

GitHub: [siddontang](#)

Thank You!!!