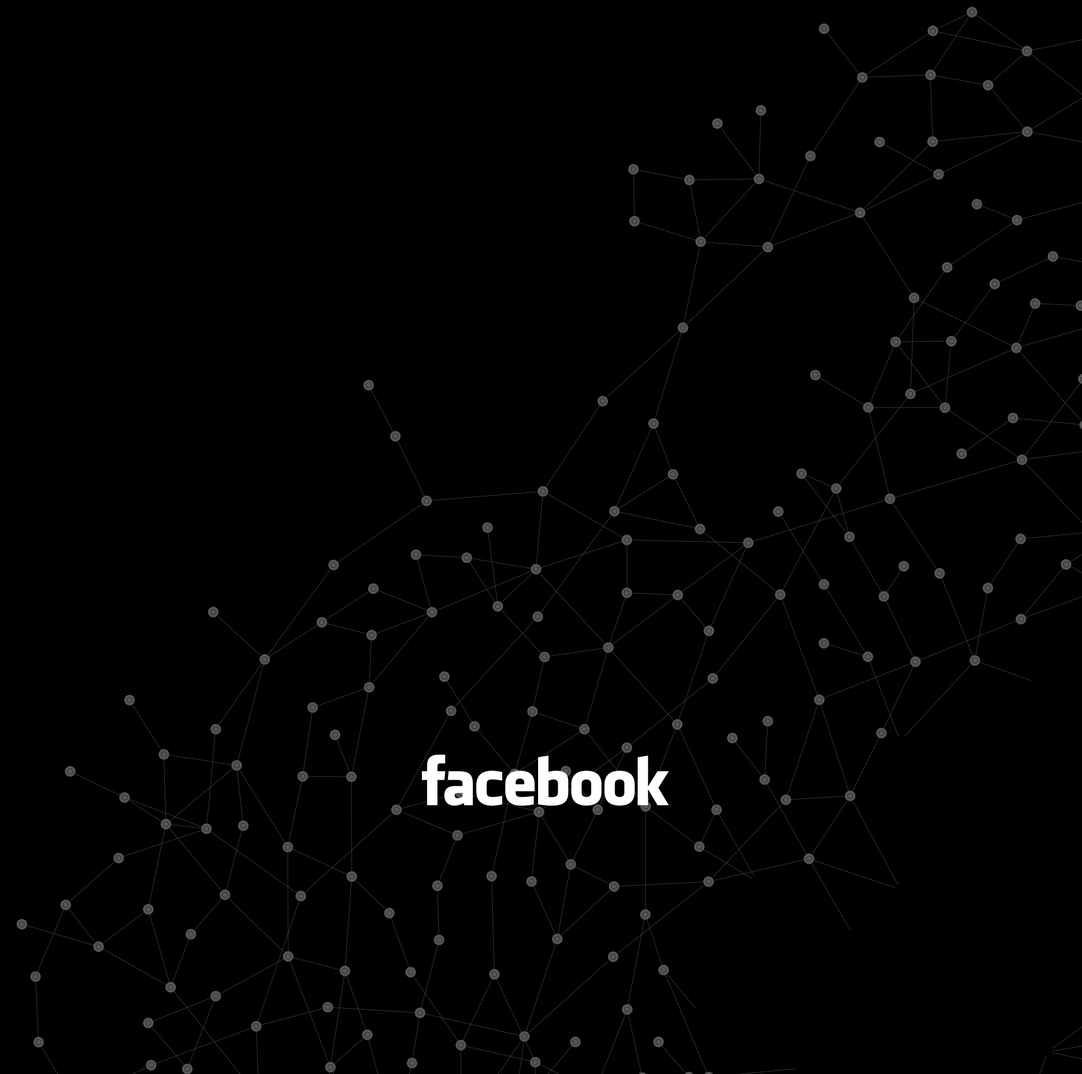
ONE

One Network Engineer

James Paussa
network infrastructure engineer

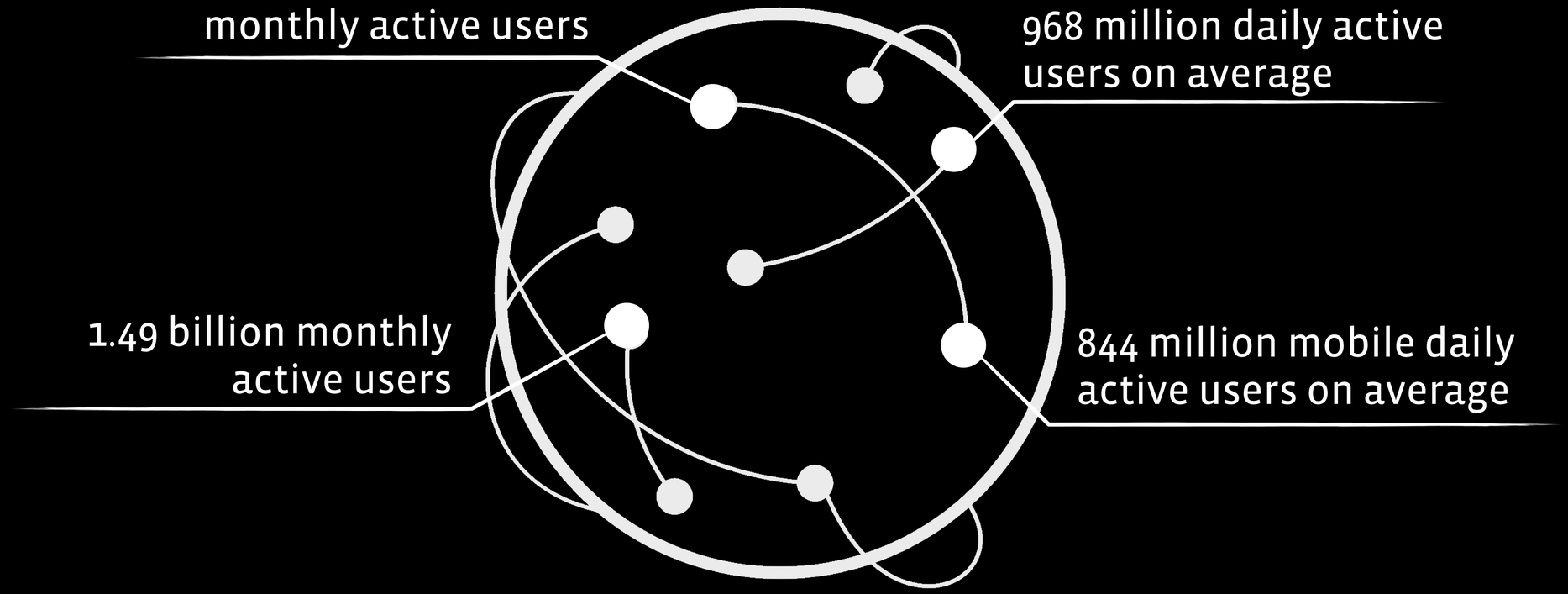
facebook



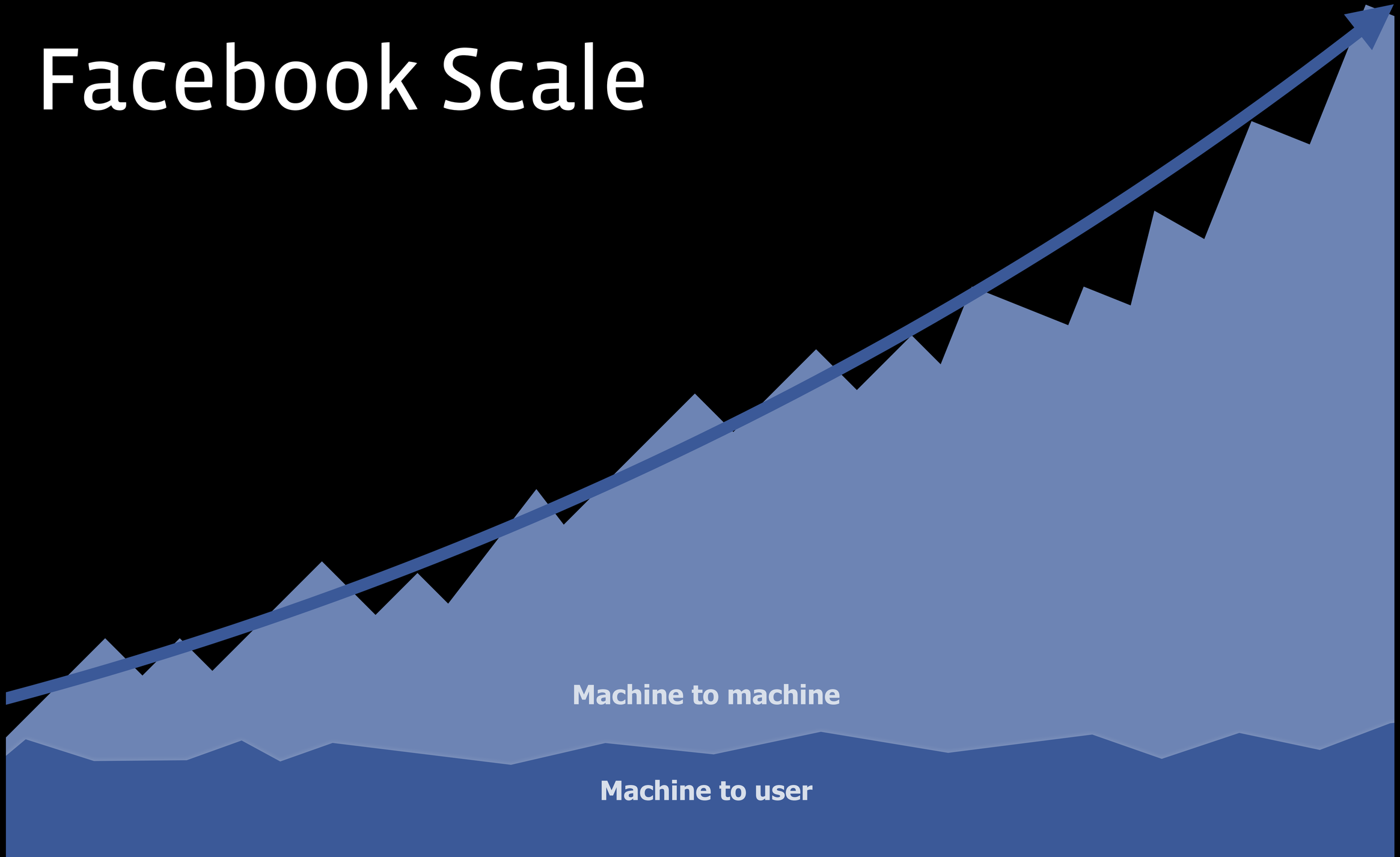
What Do We Do?

- A single engineer runs the network at a time
- Network monitoring and alarming
- Responsible for the entire production network

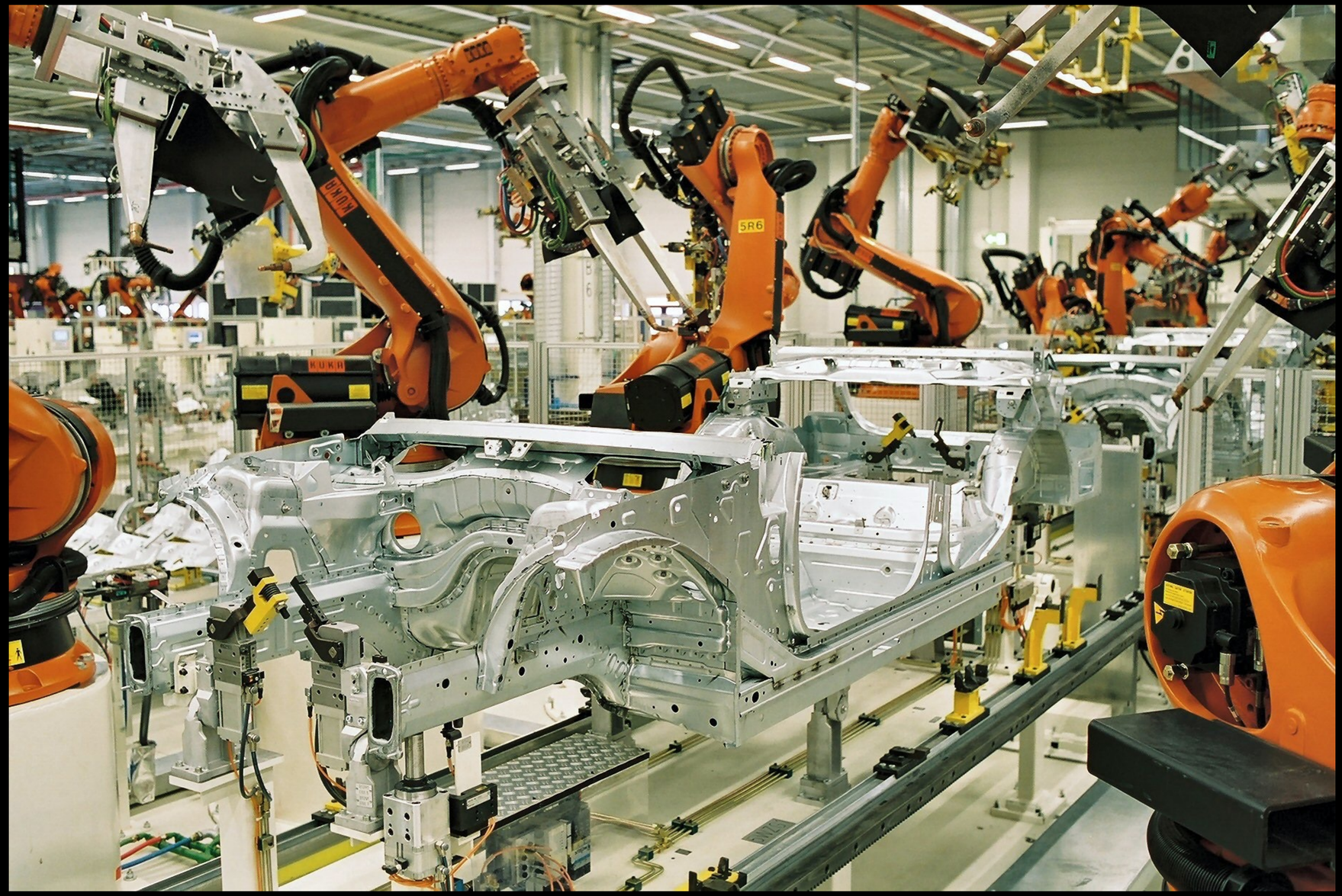
Facebook Scale



Facebook Scale



Automation



What We Don’t Do

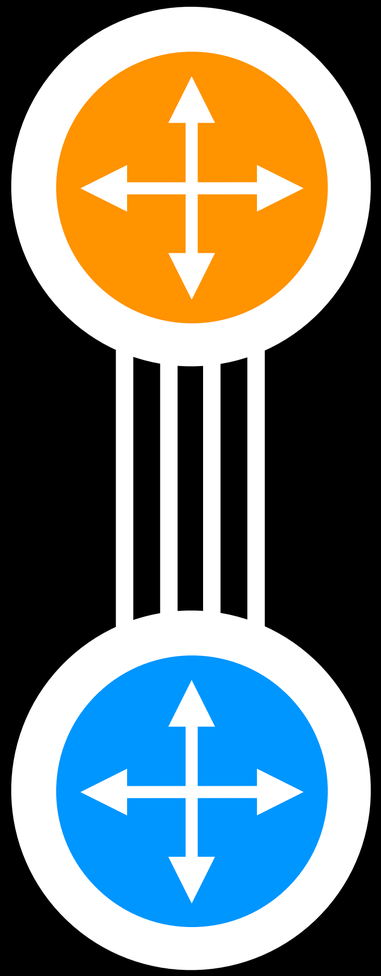
- On site work (cleaning fibres, LCs, etc)
- Working with remote hands
- Device deployment

Myths

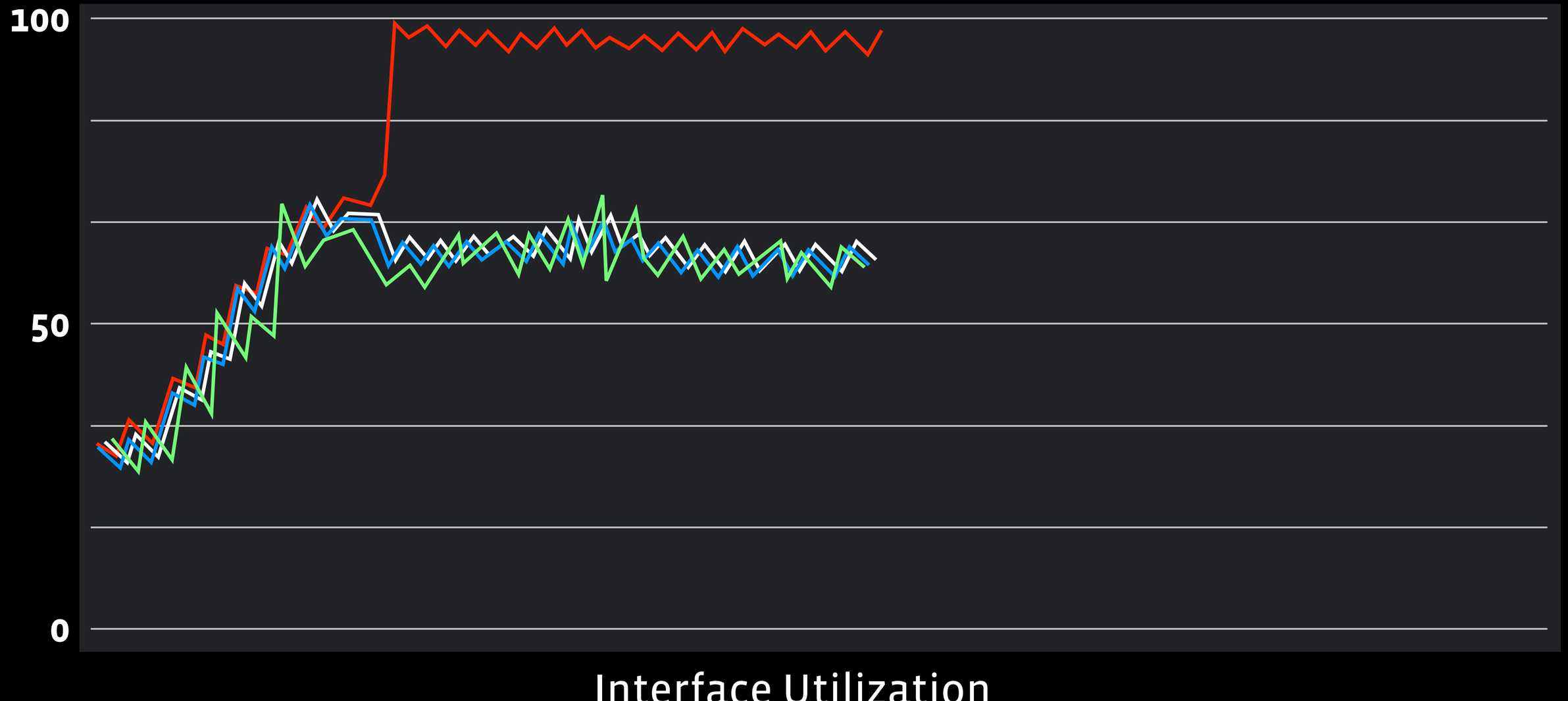
- Automation fixes everything
- We've fixed everything
- Doesn't apply

Link Imbalance

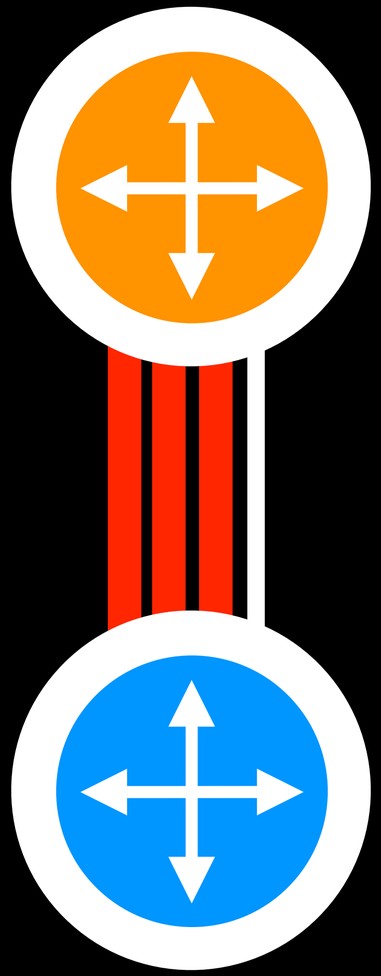
Link Imbalance



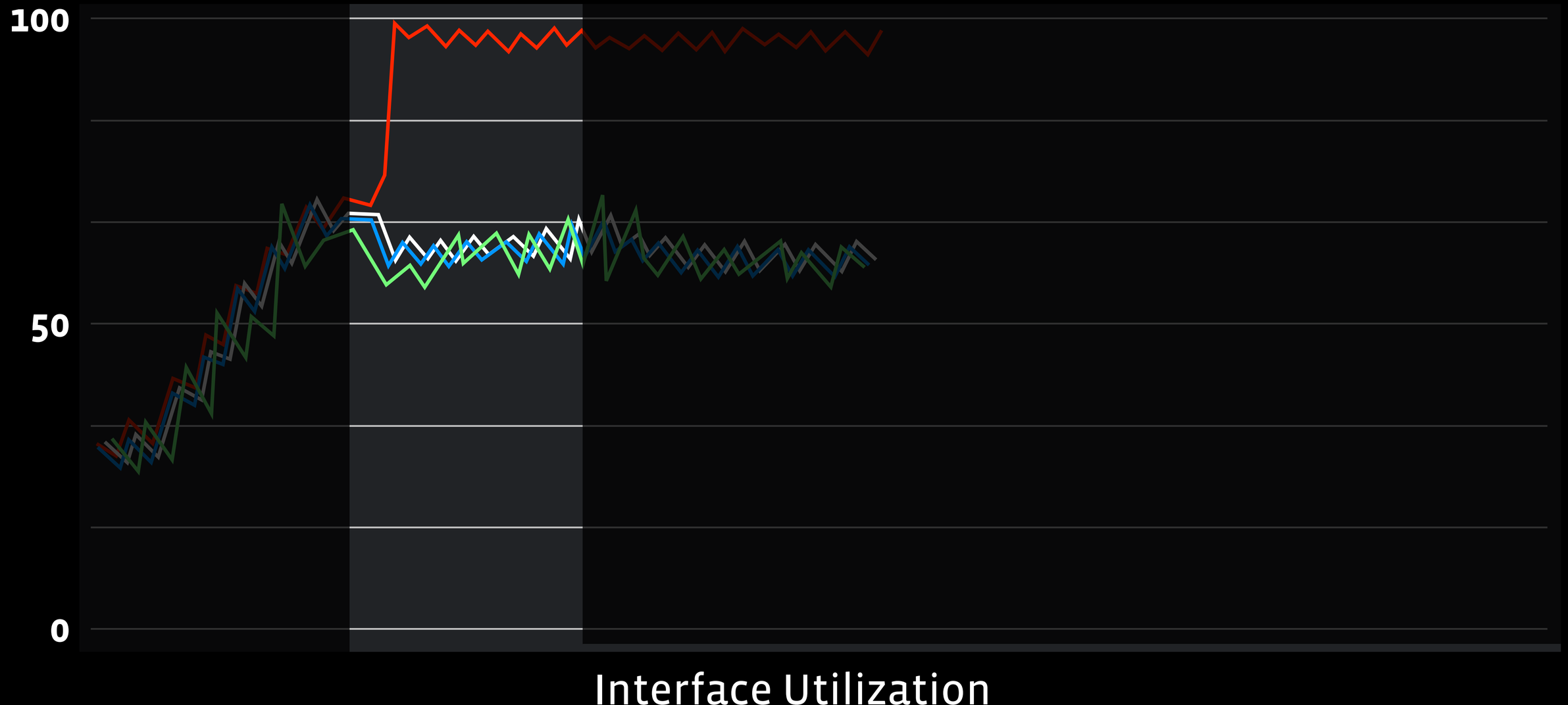
Link Imbalance



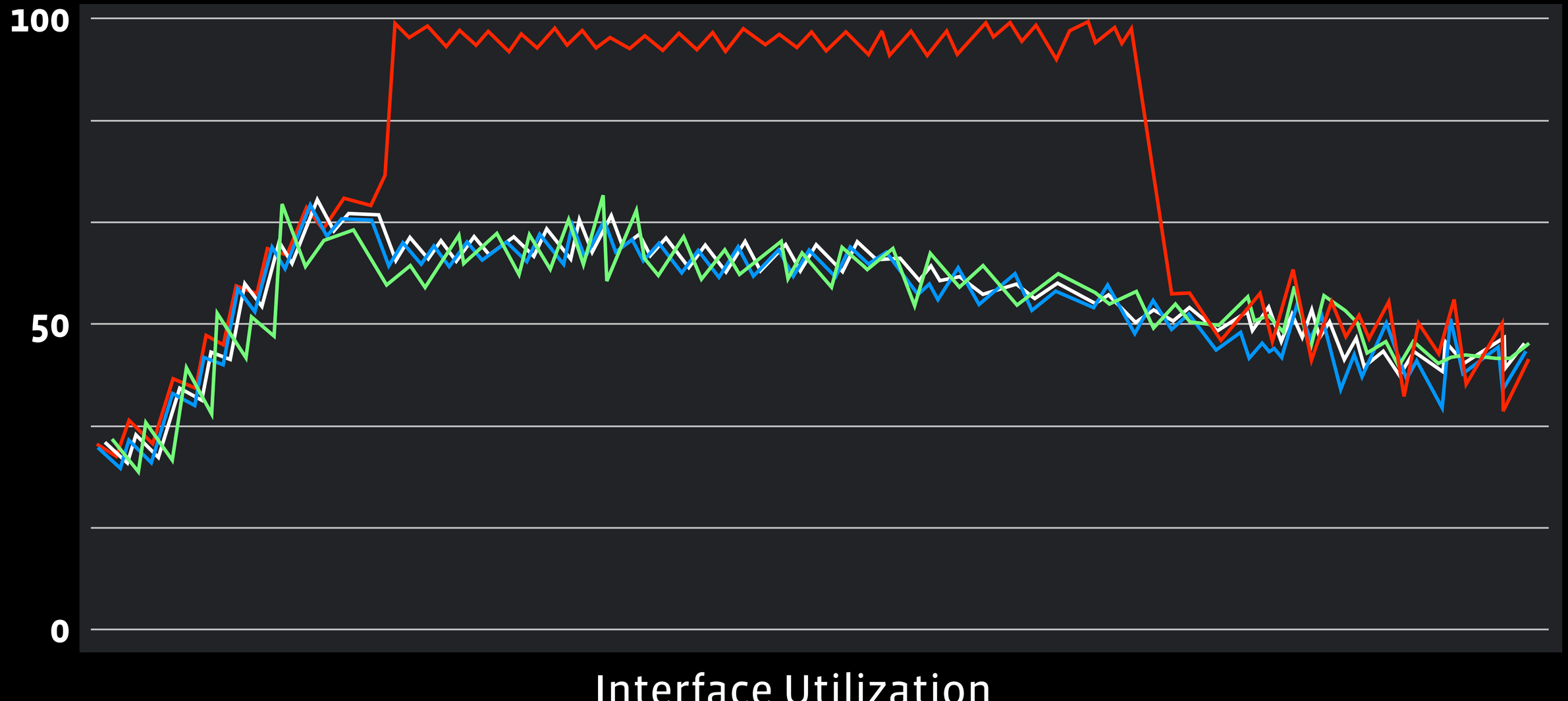
Link Imbalance



Link Imbalance



Link Imbalance



Link Imbalance

srsly wtf.

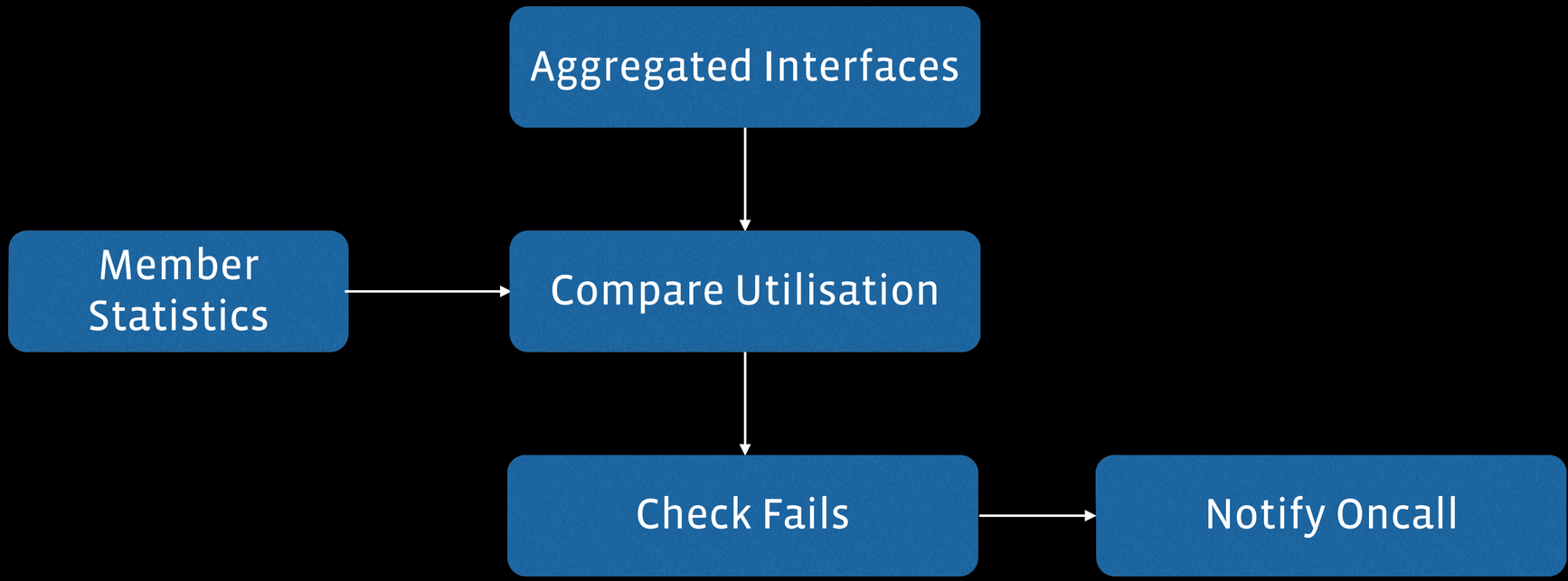
Link Imbalance

- Intercluster links only
- >80%+ Util
- Migratory

What Now?

- Detection
- Mitigation

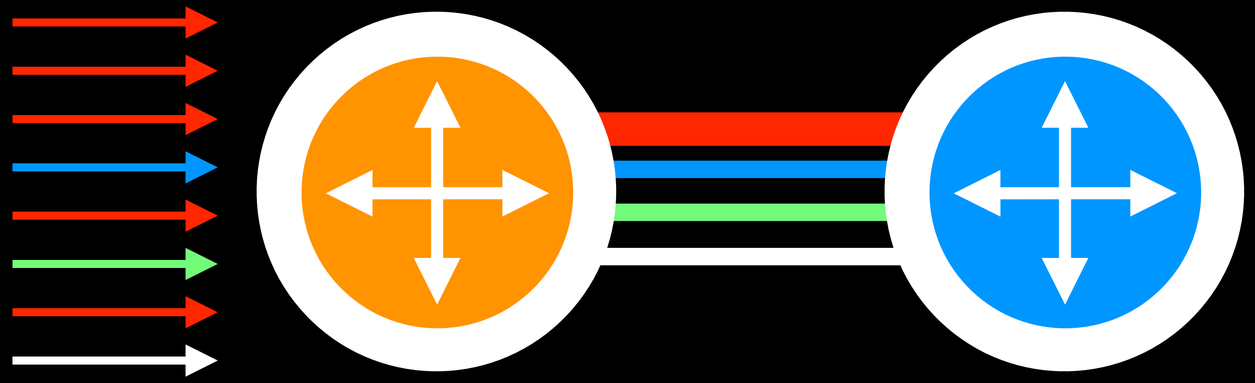
Imbalance Detection



Imbalance Mitigation

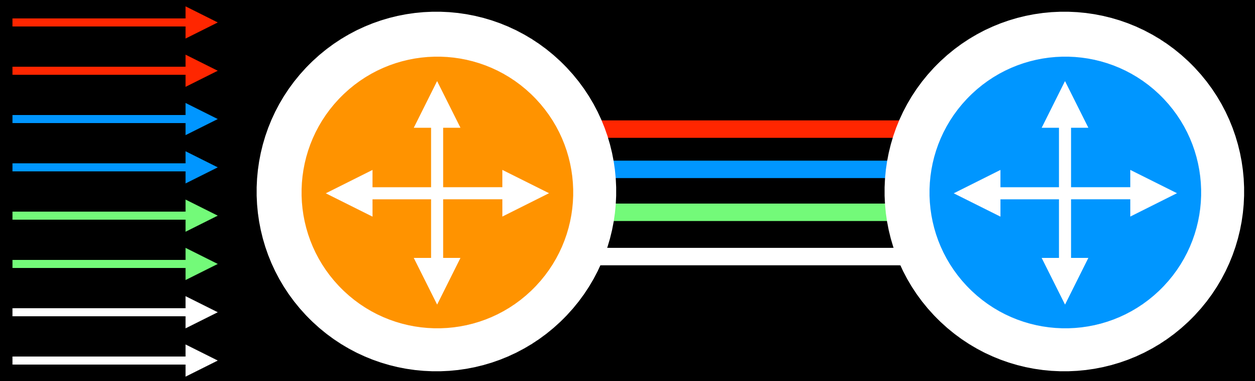
SIP + DIP + SPORT + DPORT + Hash Key

Imbalance Mitigation

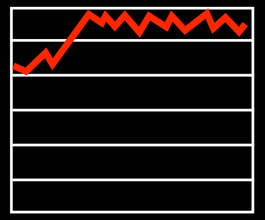


Roll Hash

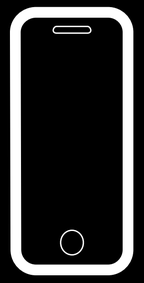
Imbalance Mitigation



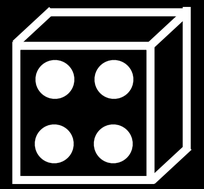
Link Imbalance



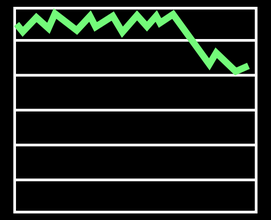
Detect



FBAR



Roll Hash



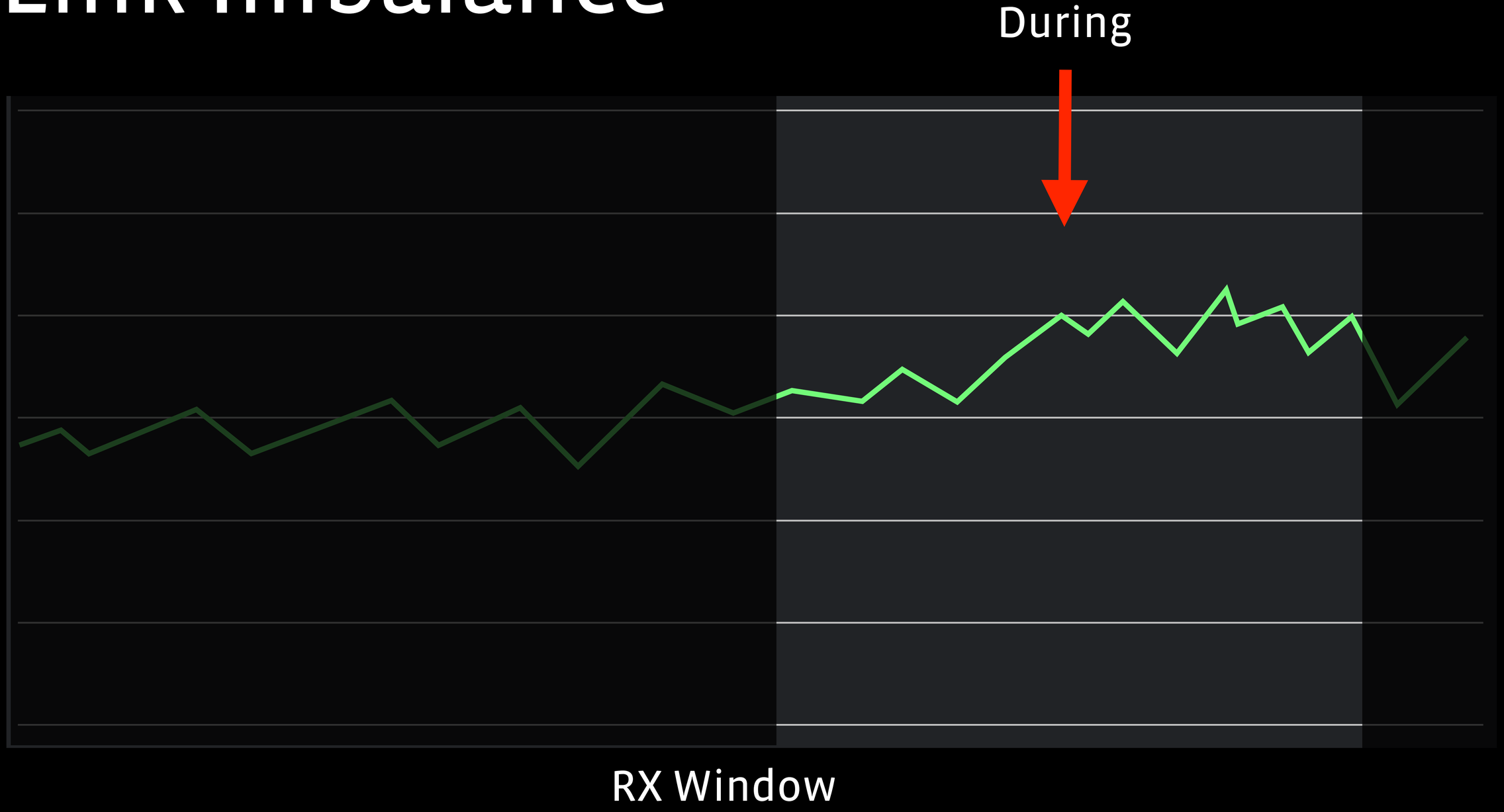
Resolved

SUIT GAS DIVERTER

Link Imbalance

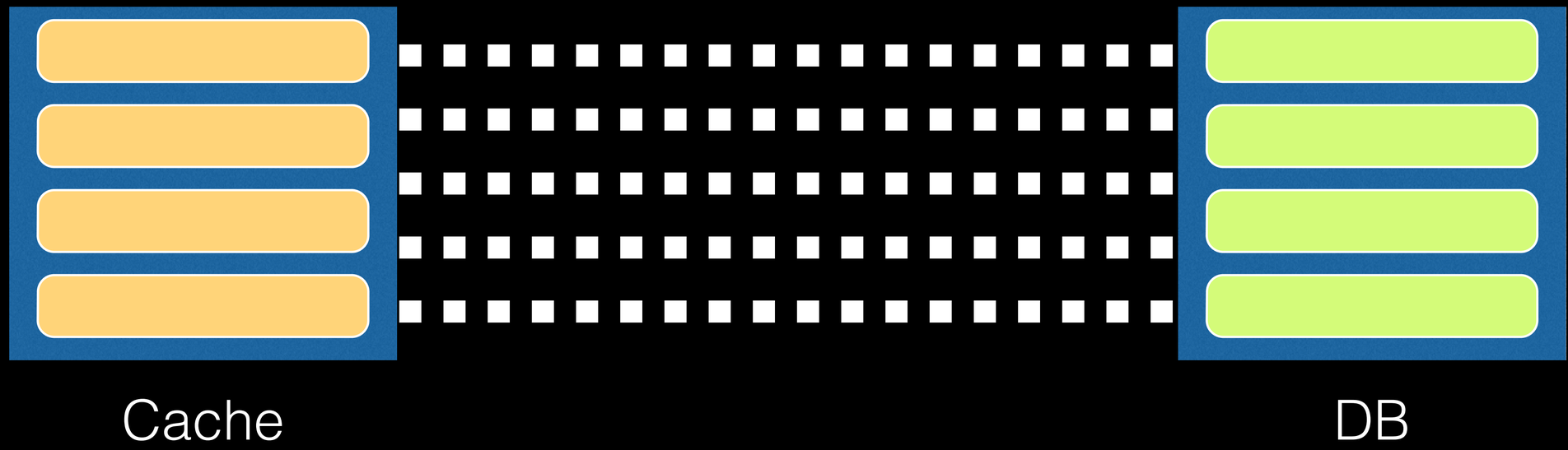
ZOMG!! WTF!! NO, NO, NO! &(#$&*(

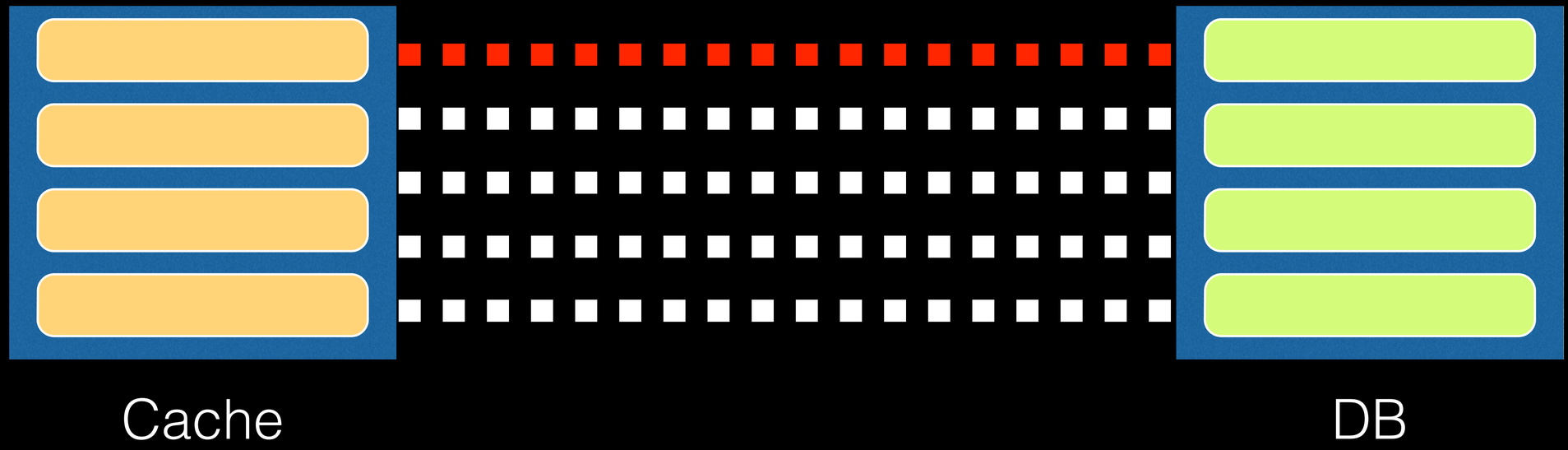
Link Imbalance

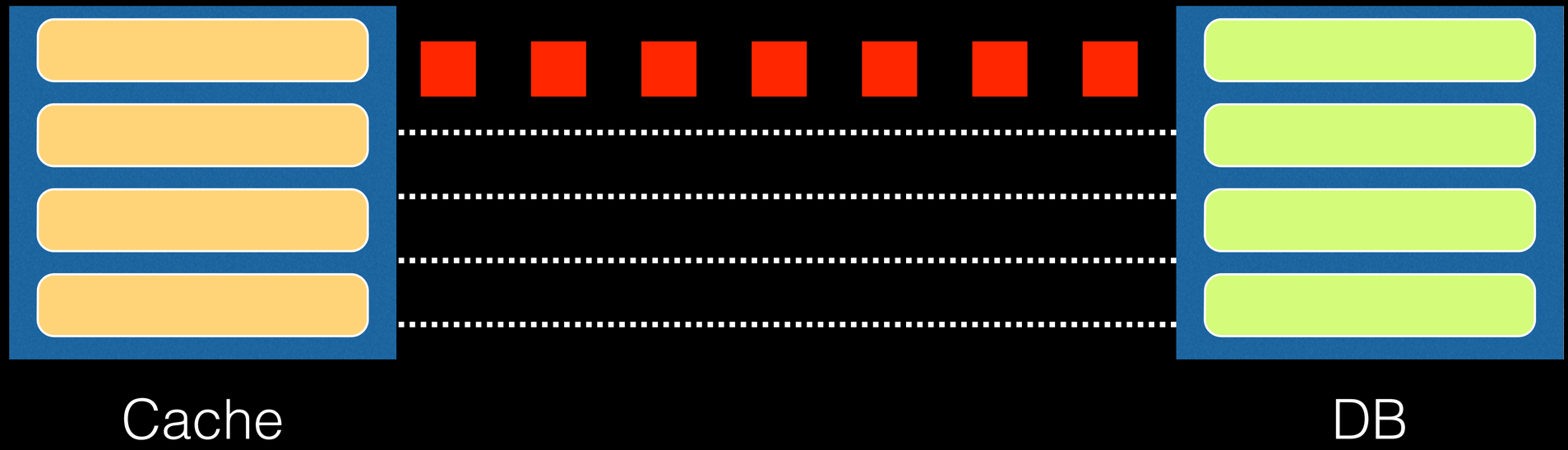


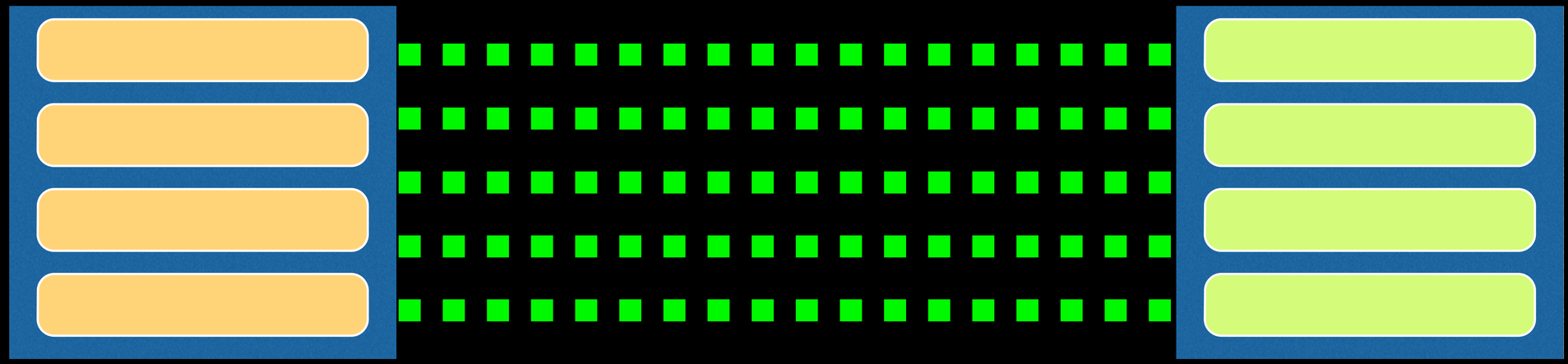
Link Imbalance

It Isn’t (always) The Network




Cache
DB

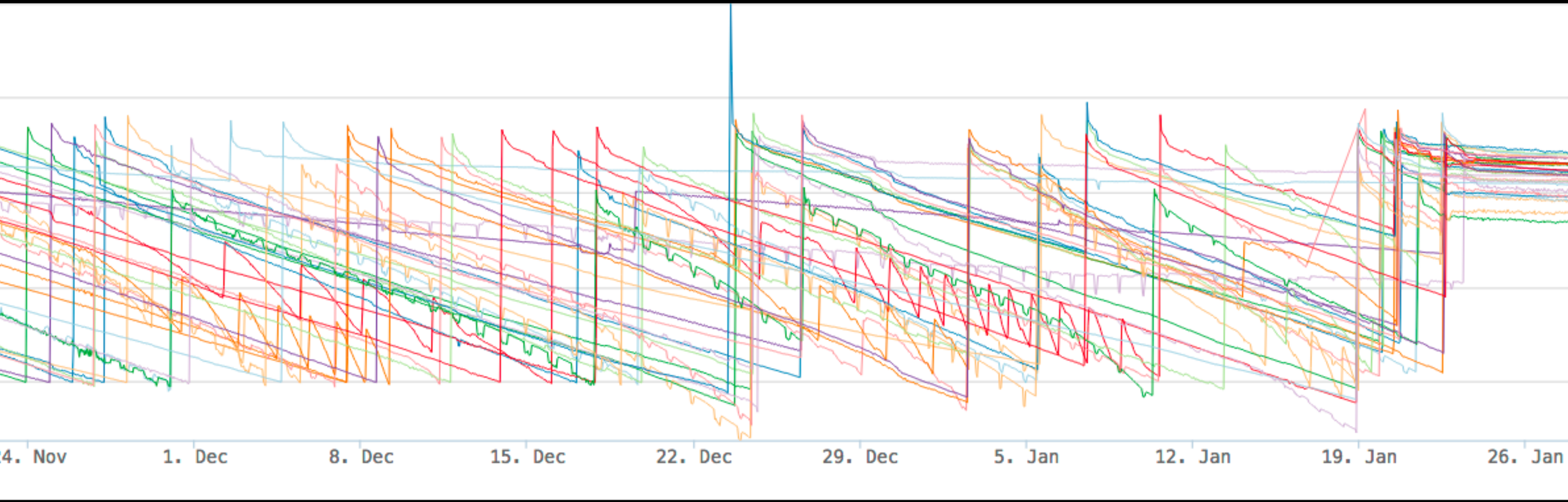

Cache
DB


Cache
DB

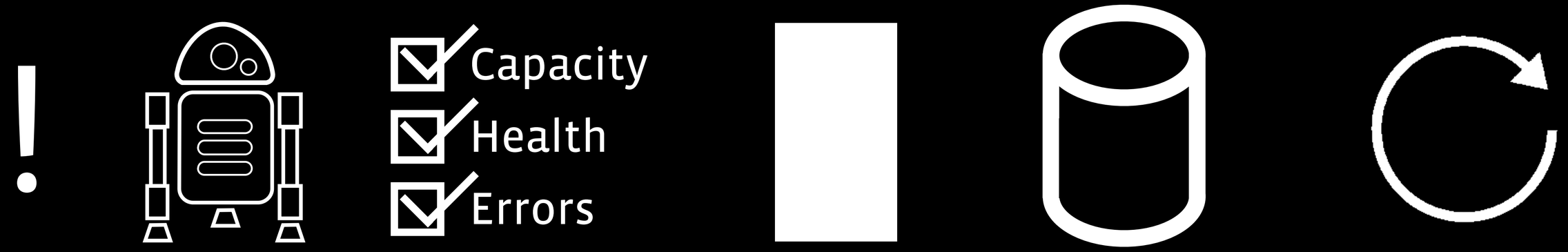
Link Imbalance - Lessons Learned

- Resolved issues
- Root Cause
- Software helps
- Service owner identified
- Resolution time
- Small loss, significant impact

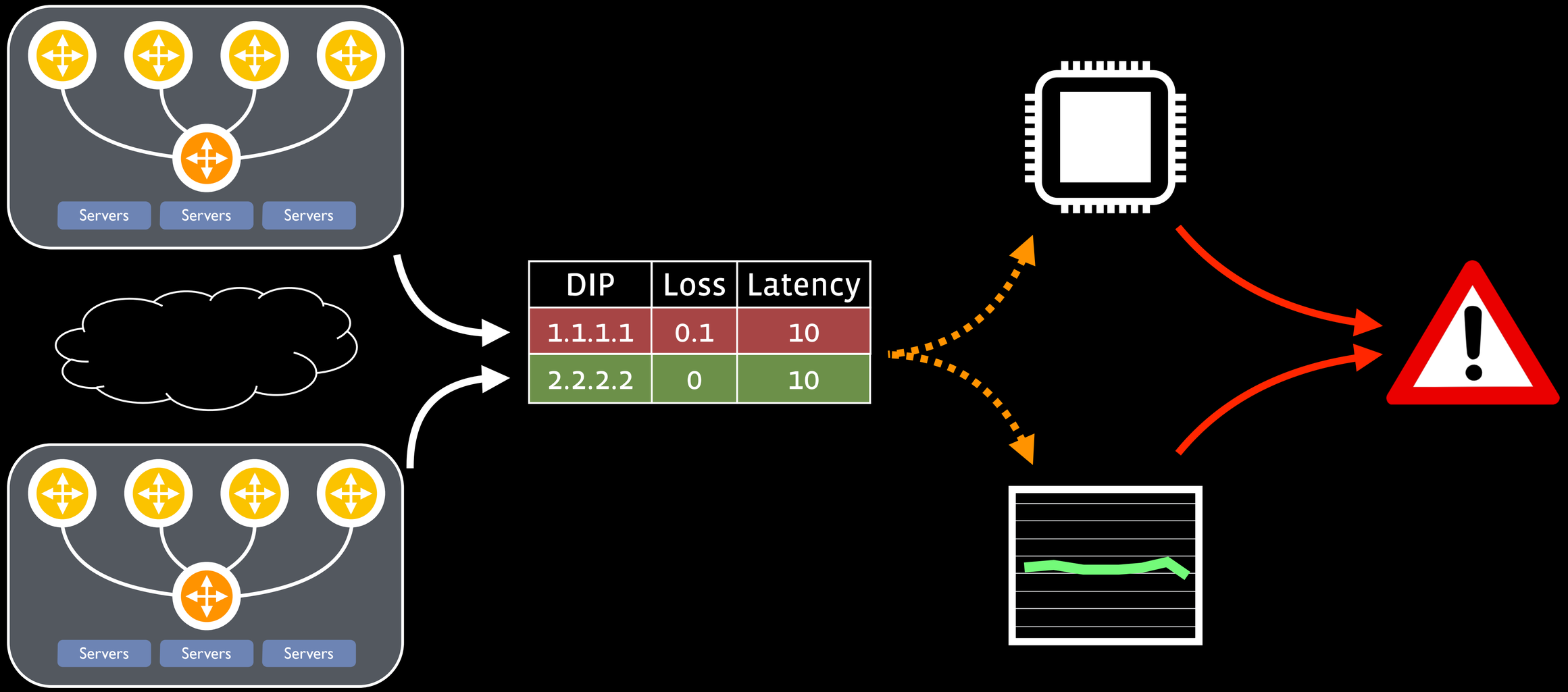
3 Months In A Leaky Boat



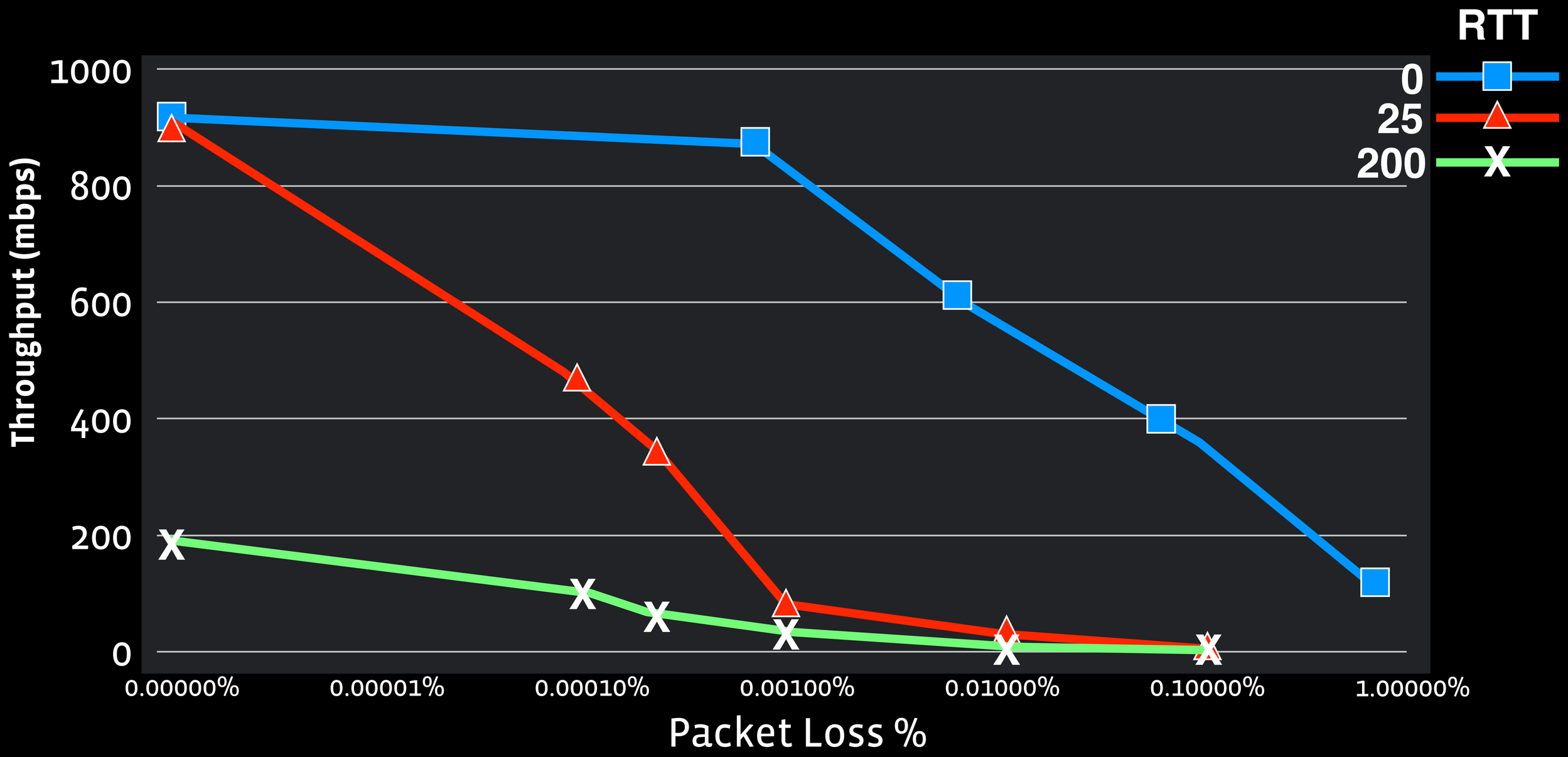
Memory Issues



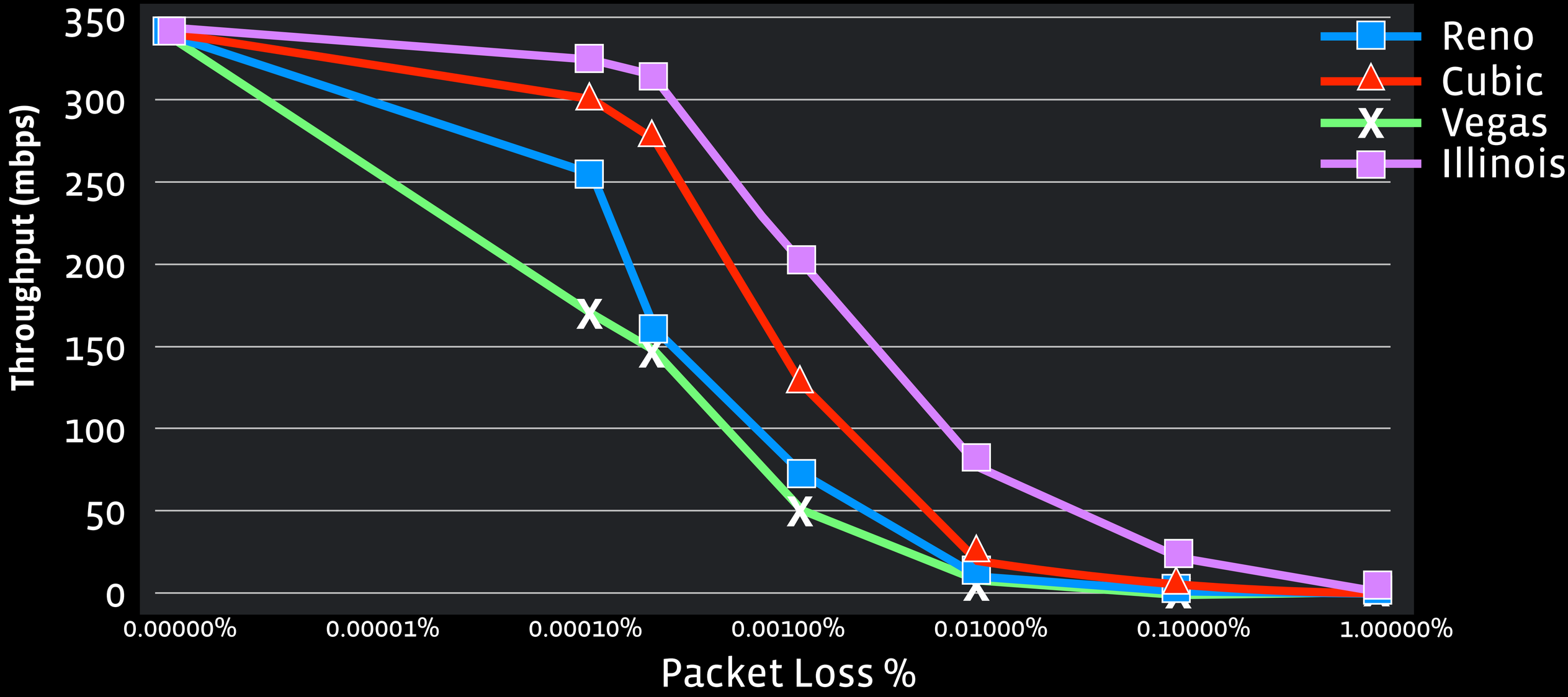
Detection


Servers
Servers
Servers
DIP
Loss
Latency
1.1.1.1
0.1
10
2.2.2.2
0
10
Servers
Servers
Servers

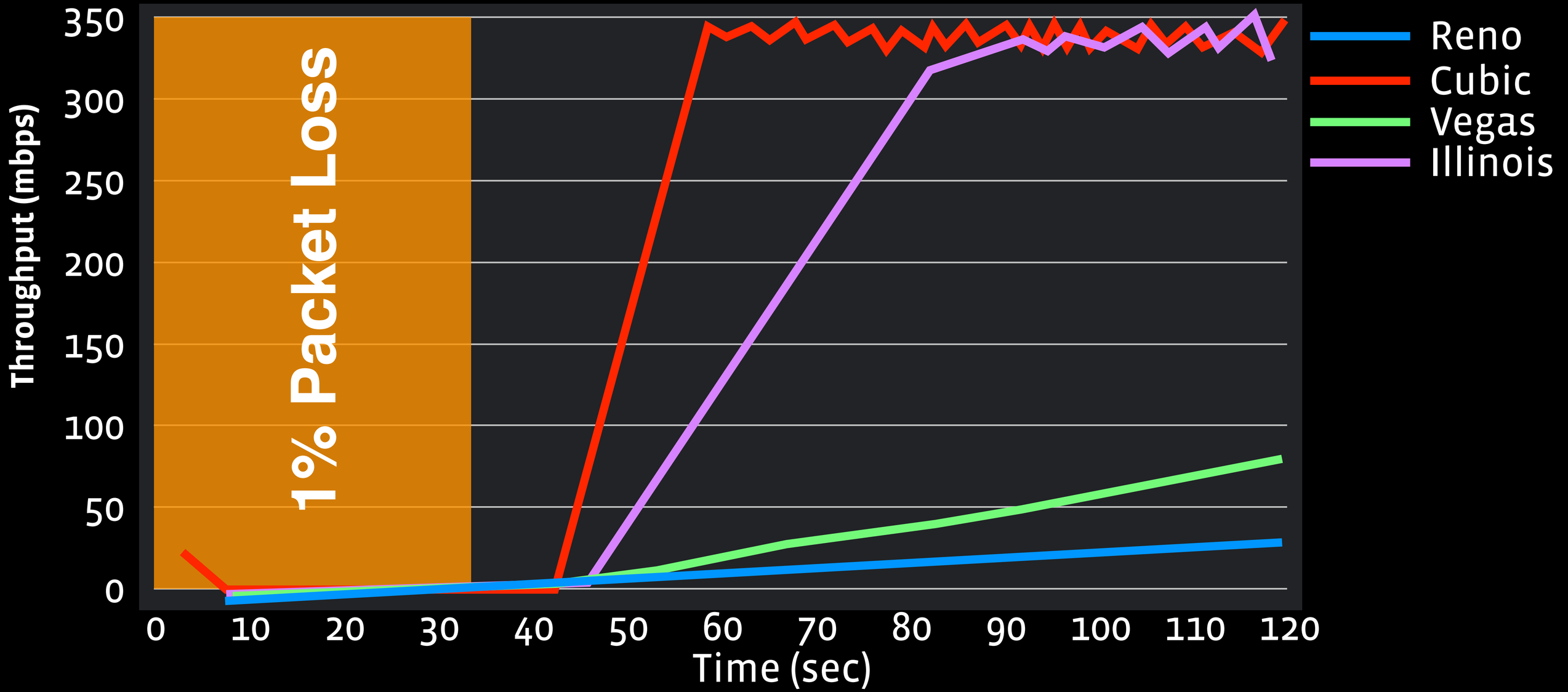
Loss Effects on Throughput



Different algos?

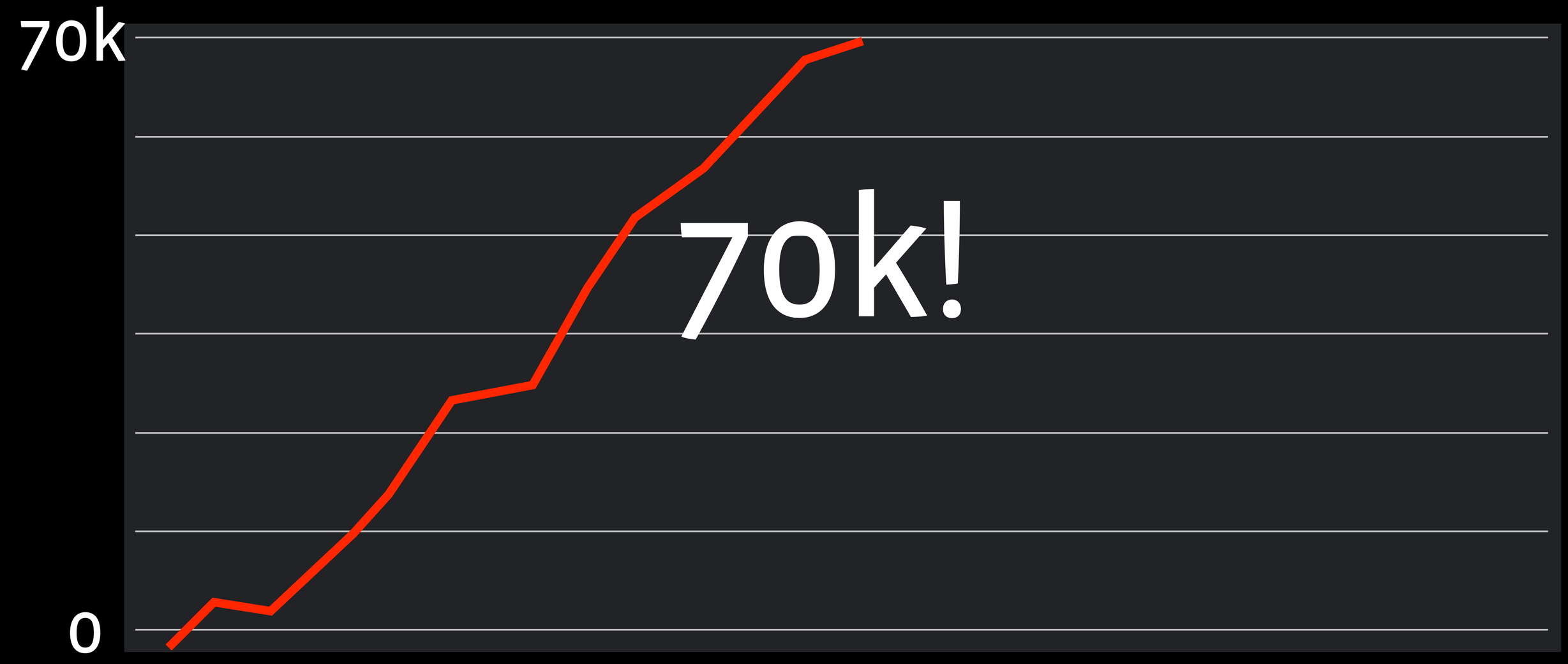


Recovery time



So, wait, how does this apply to me?

Alarms



Interface Issues

Alarms Now

Automation - One Month

3.37b	750k
0.99%	99.6%

Why?

Sleep

Why?

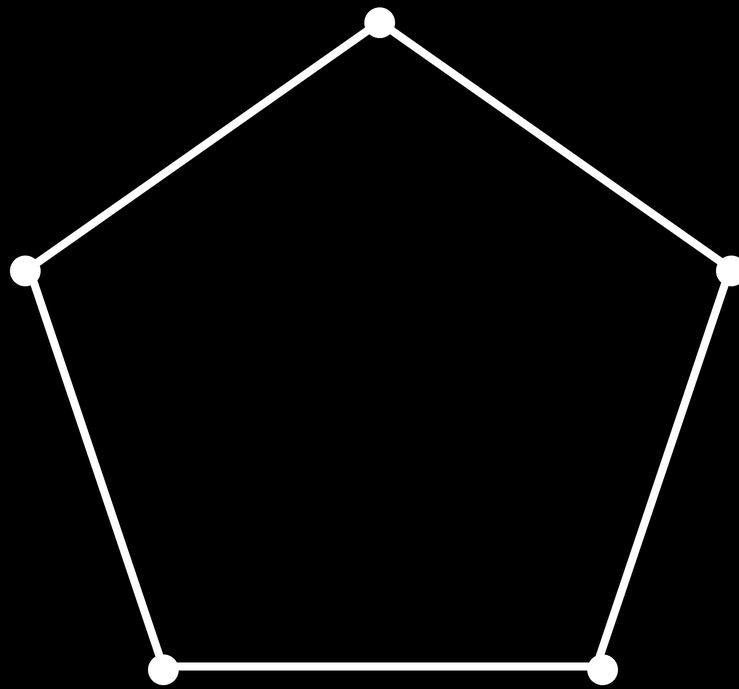
750,000 * 2 = 1,500,000

1,500,000 / 60 = 25,000

25,000 / 160

150+

Lessons Learned & Take Aways



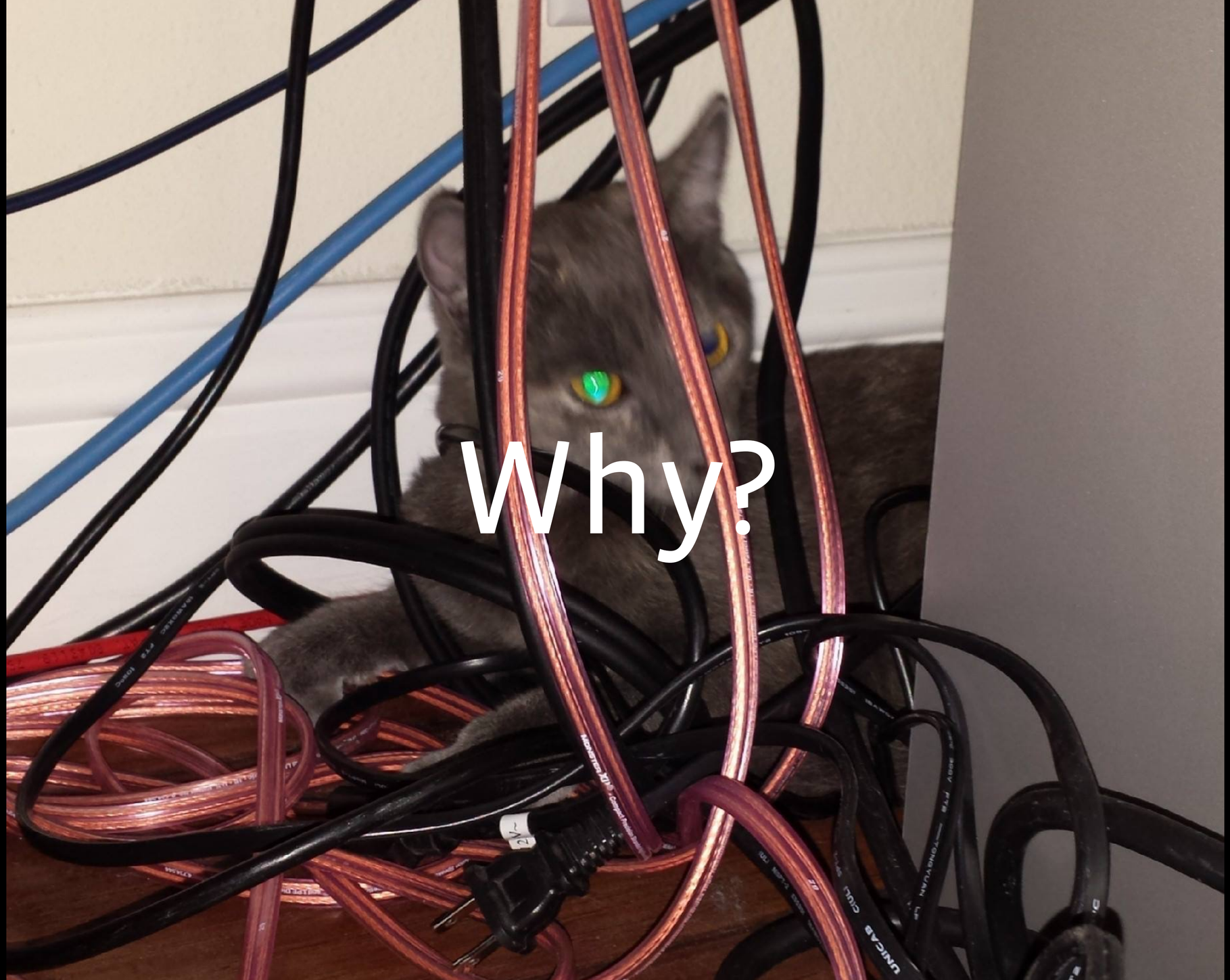
Use what you've got

Prototype early, iterate often

Duct tape keeps things running

Spend the time to root cause issues

The sooner the robots take over the better


Why?

DONE IS BETTER THAN PERFECT

?