Meta

INFRASTRUCTURE

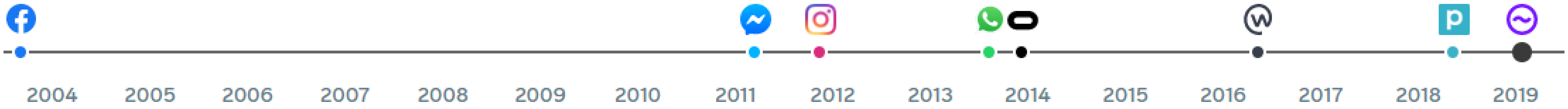
Silent Data Corruptions at Scale

EDPS 2023

Harish Dixit
Principal Engineer
Meta Infrastructure
Menlo Park. CA. USA

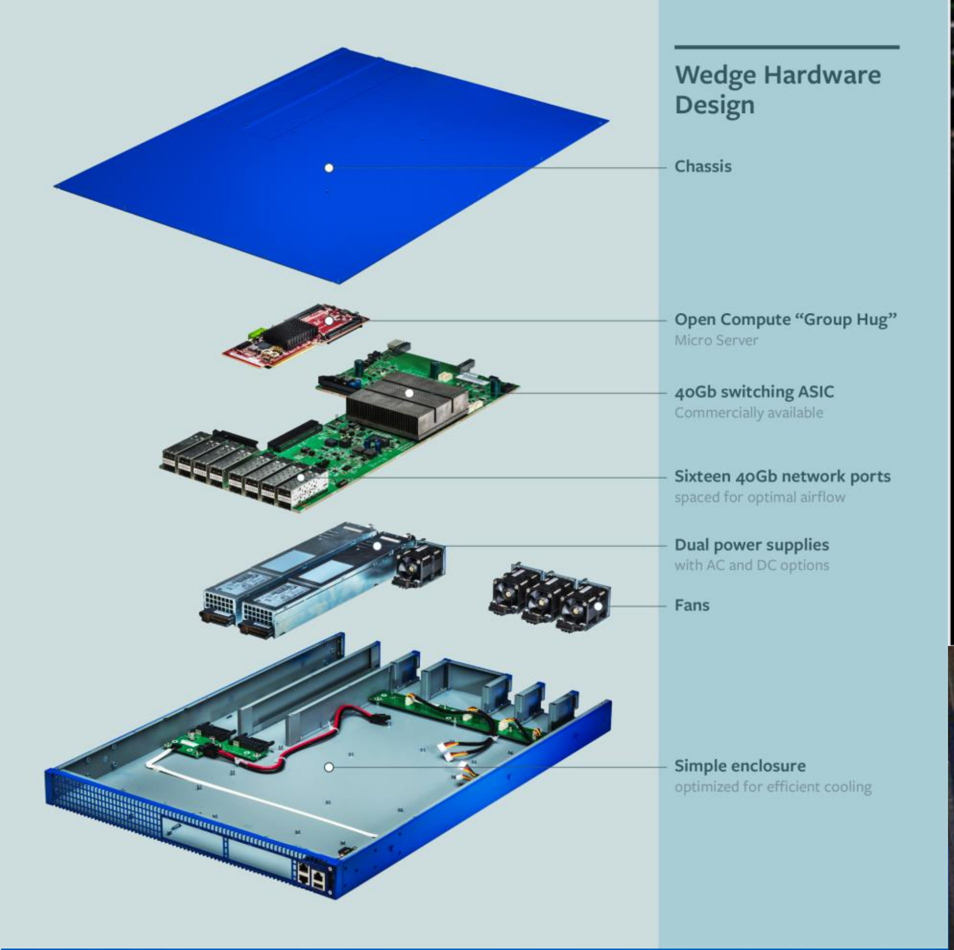
Family MAP : 3.81B

Globally, there are more than 3.81B people using Facebook, WhatsApp, Instagram or Messenger each month.



*MAP - Monthly Active People

Source: Meta Platforms Inc. Q1 2023

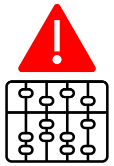

Wedge Hardware Design
Chassis
Open Compute "Group Hug"
Micro Server
40Gb switching ASIC
Commercially available
Sixteen 40Gb network ports
spaced for optimal airflow
Dual power supplies
with AC and DC options
Fans
Simple enclosure
optimized for efficient cooling

Silent Data Corruptions

$$(1.1)^{53} = 0$$

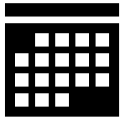
$$(1.1)^{53} = 0$$

Silent Errors in Compute Units

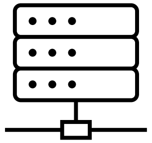


Defects
in silicon

Hard to
detect

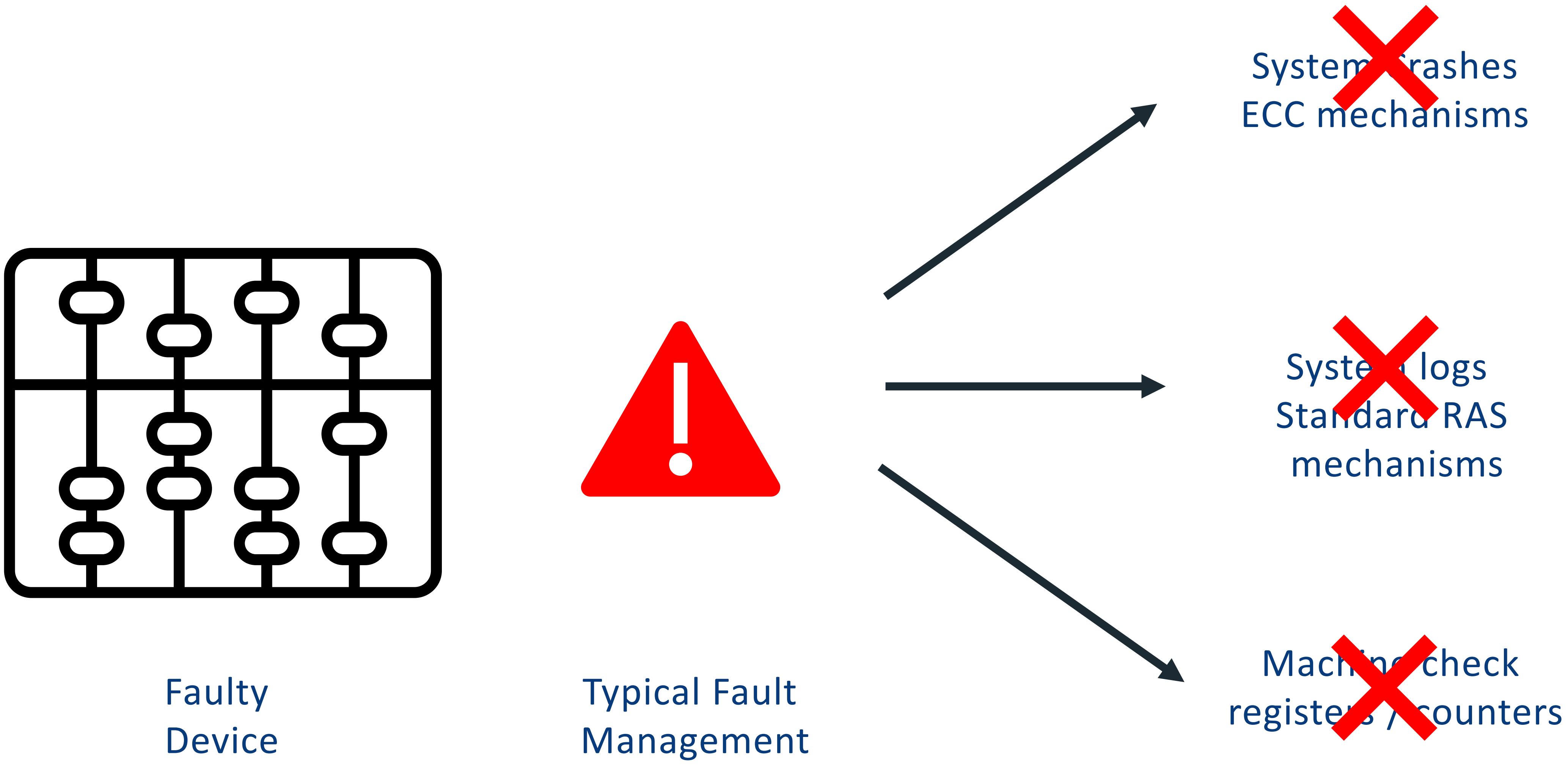


Undetected for
months/years



Significant impact
to services

What makes SDCs different ?



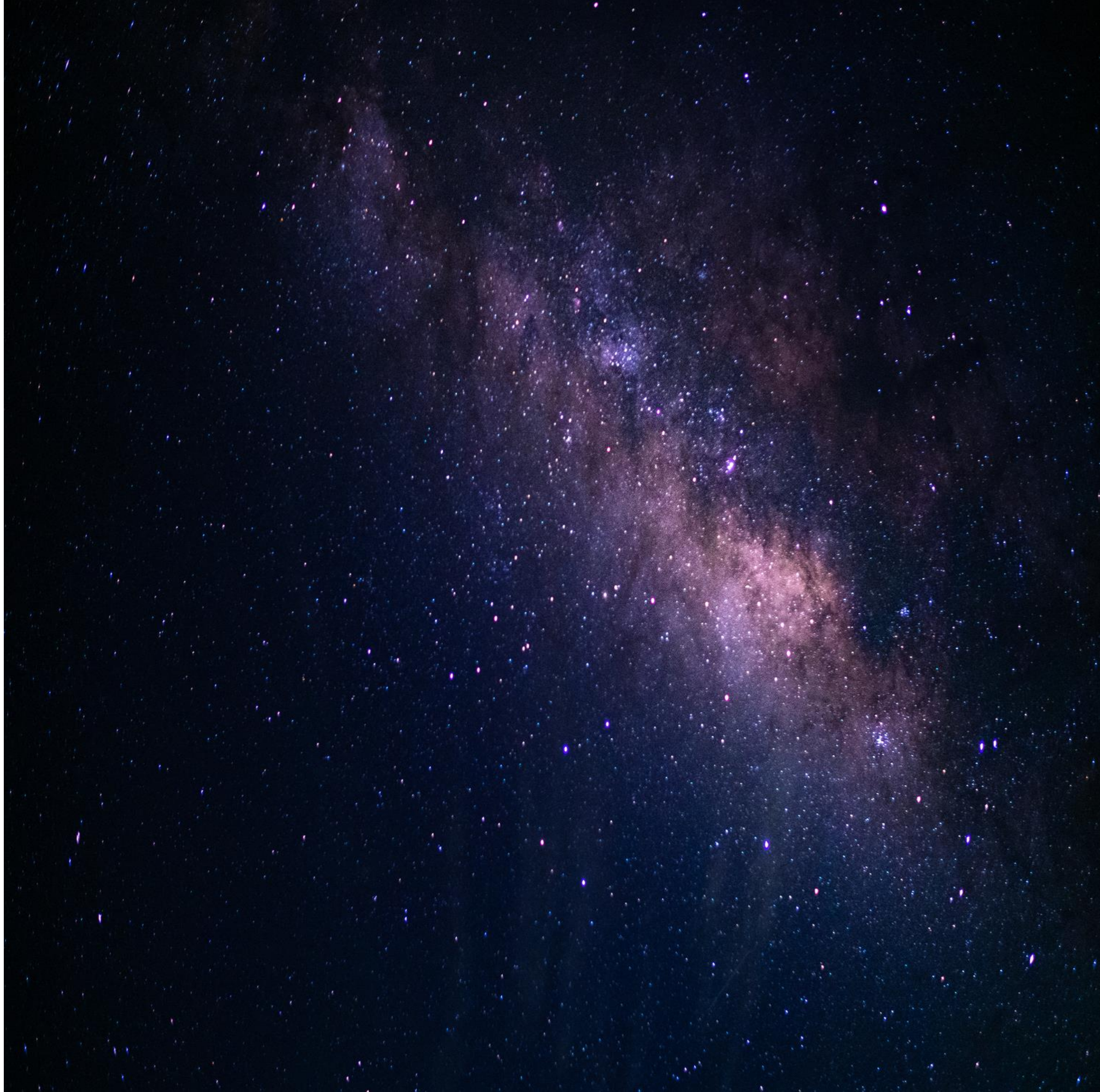
In the case of silent errors, none of these are available

But wait ….

Cosmic Rays ?

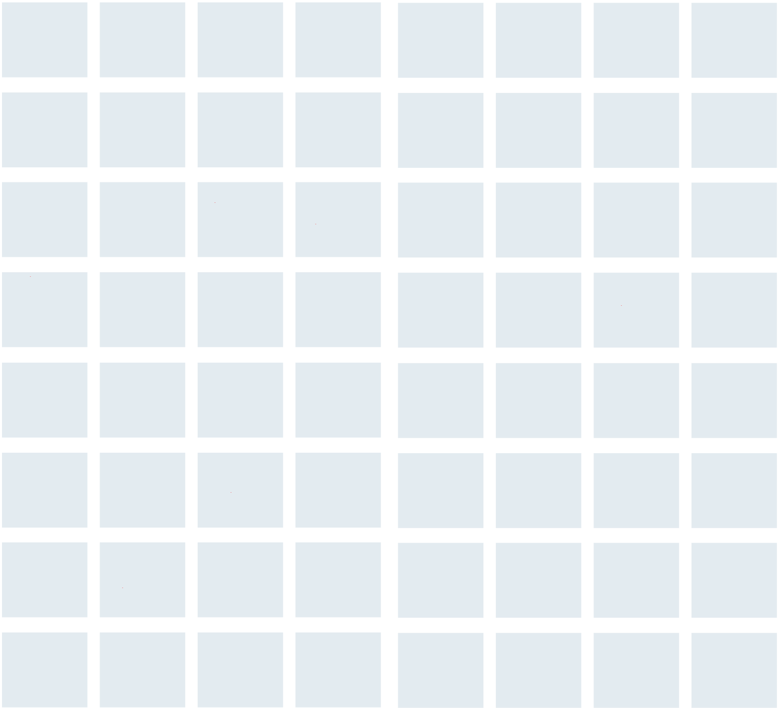
Pentium FDIV ?

Isn’t this a solved problem ?



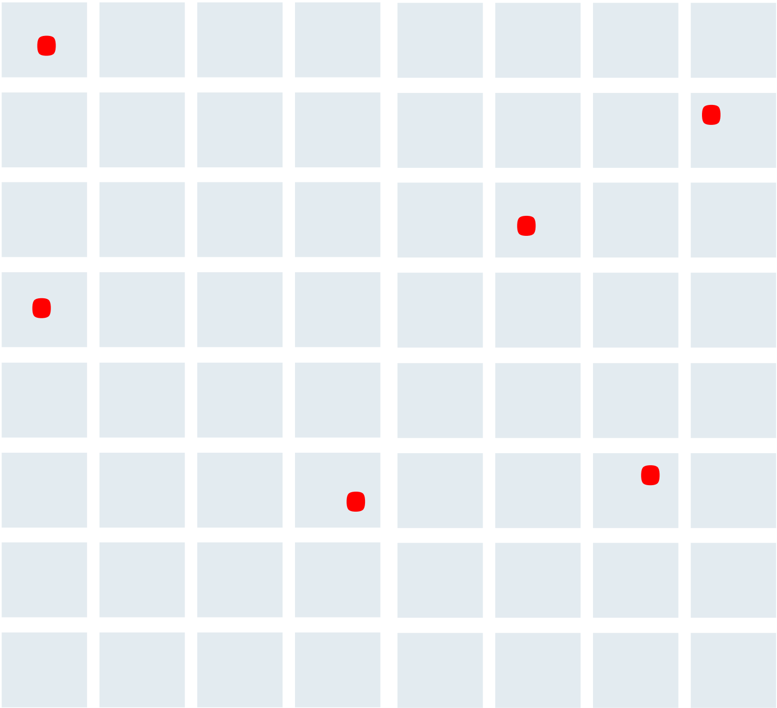
Cosmic Ray induced faults

1 fault in a million devices



Silent Data Corruptions

1 fault in a thousand devices



Blog - https://engineering.fb.com/2021/02/23/data-infrastructure/silent-data-corruption/
Paper - https://arxiv.org/abs/2102.11245

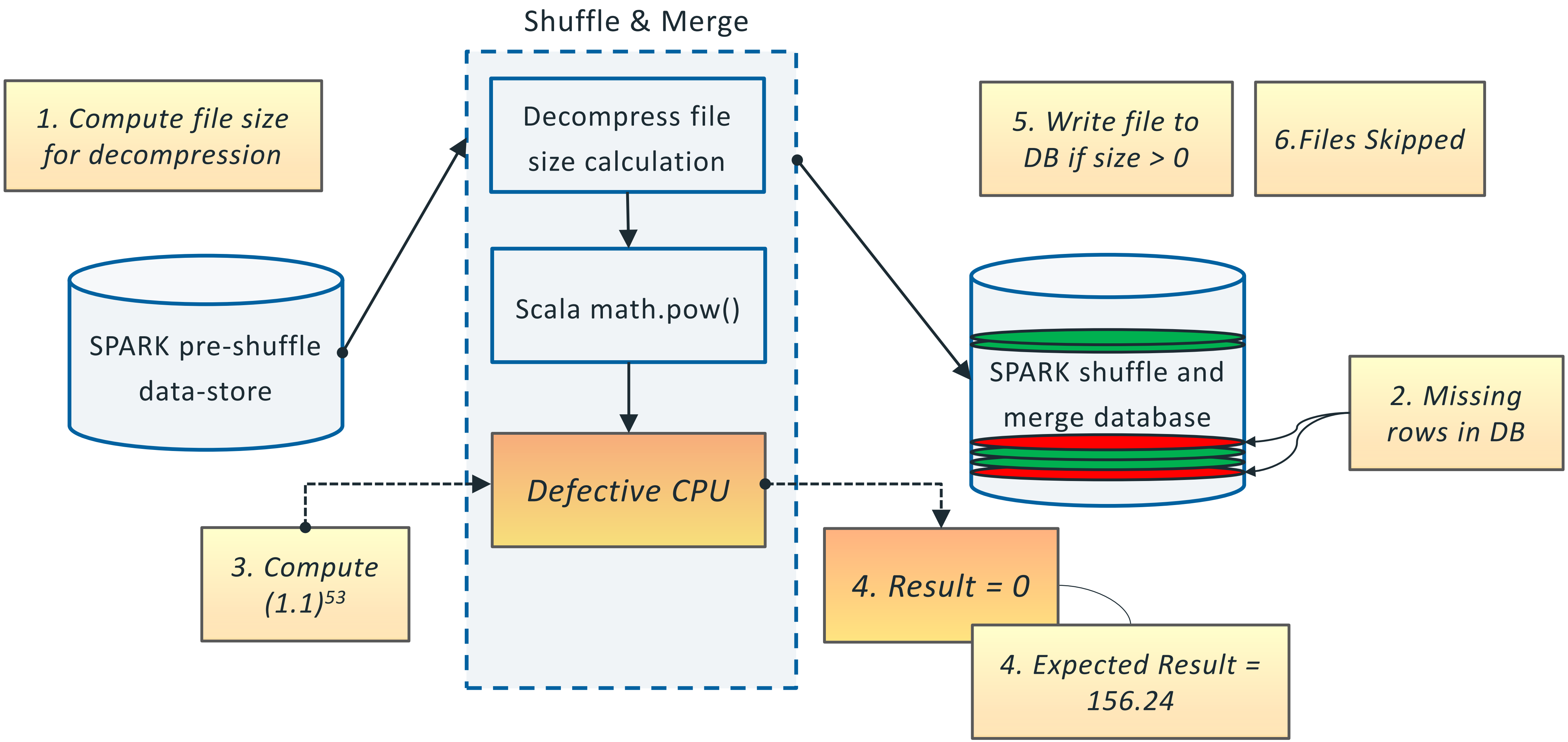
POSTED ON FEBRUARY 23, 2021 TO DATA INFRASTRUCTURE

Mitigating the effects of silent data corruption at scale


A needle in haystack situation
….. where the needle keeps moving, changing size and shape
…. and the haystack gets larger every day

How did we find these elusive SDCs?

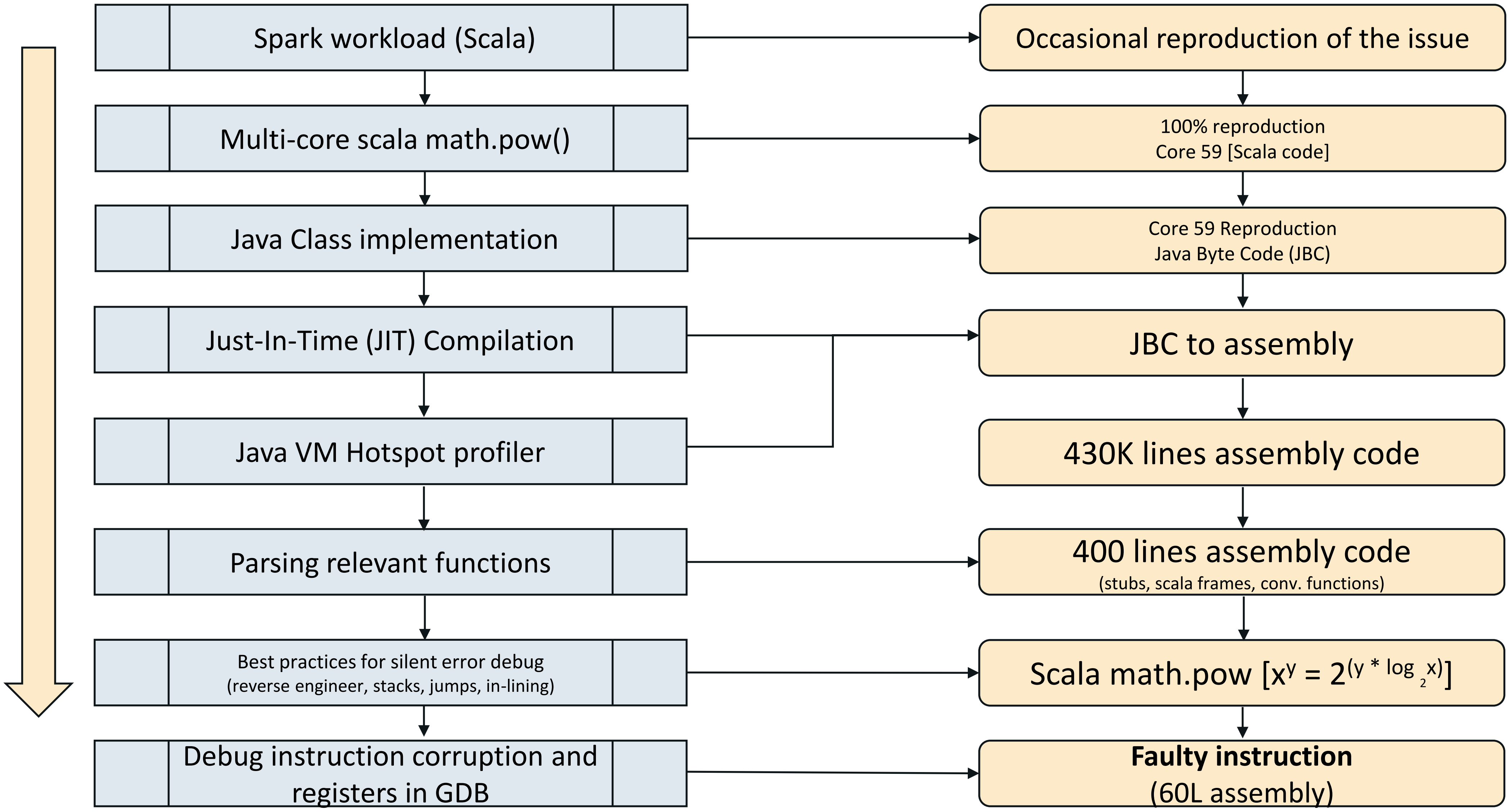
CPU Silent Data Corruption (Case Study in SPARK DB)



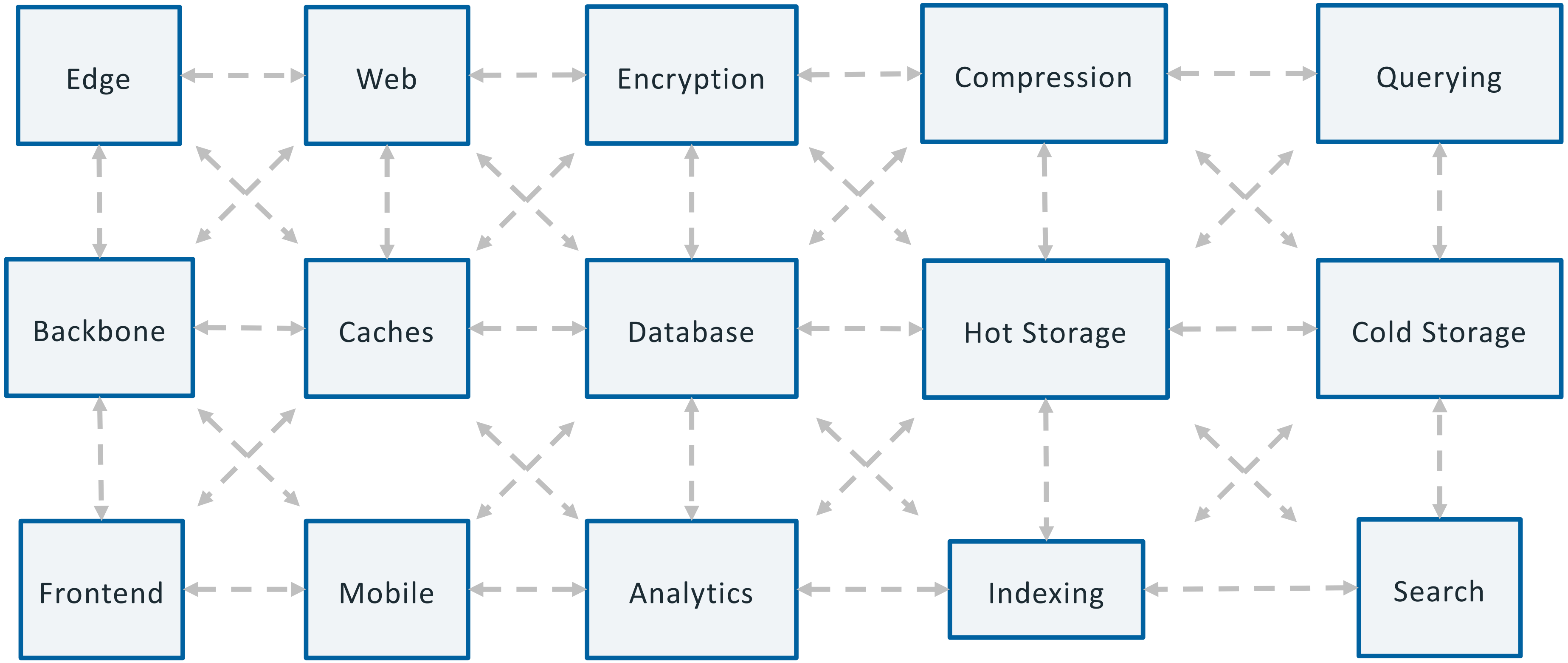
An example of a single faulty CPU encountering silent data corruptions

Result: Missing rows in a Spark Database Application (Highest Infra Severity Event)

Isolating the faulty instruction down to 60 lines assembly!



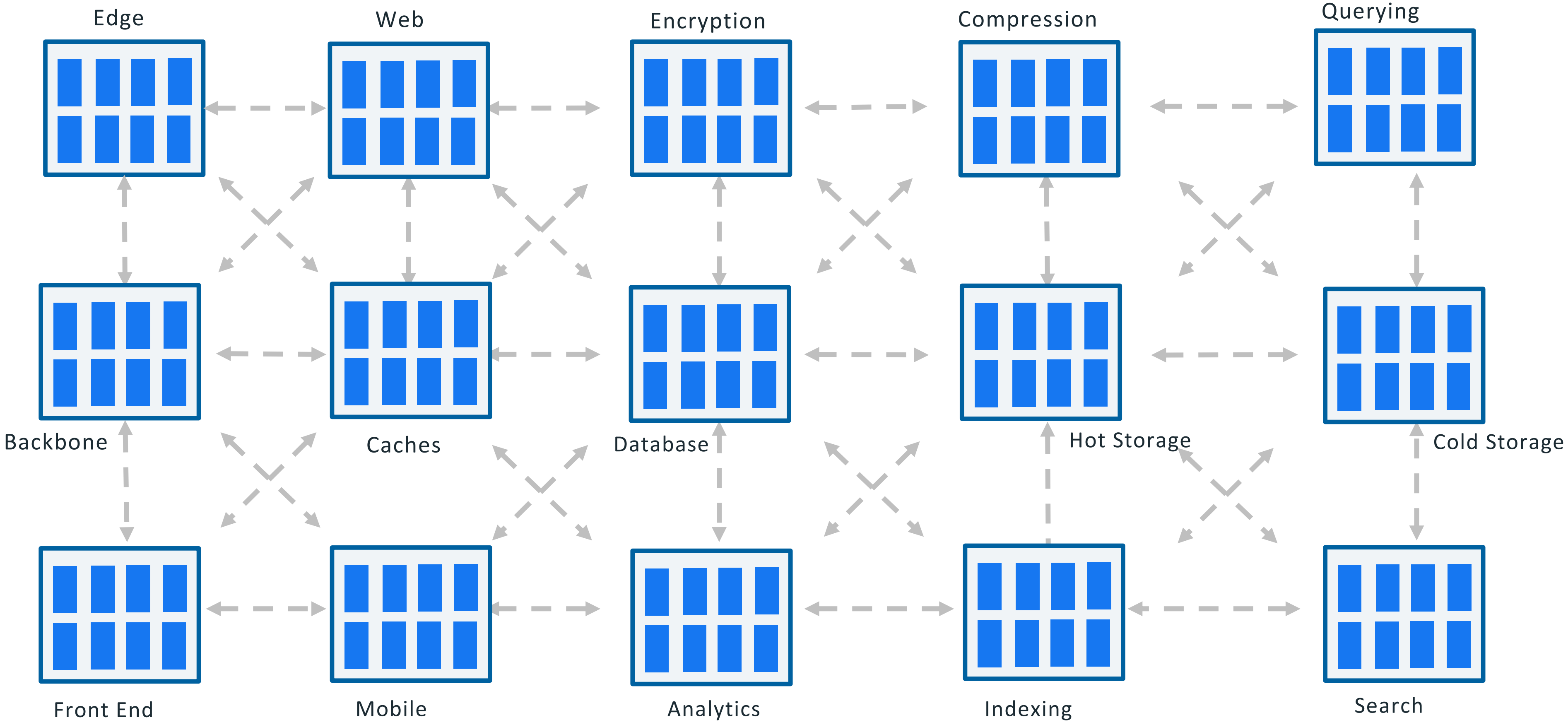
3.8B MAP translates to billions of computations and interactions every day



Failures can lead to contained and uncontained fan-outs

**illustrative only large-scale system diagram (not an architecturally directional flow representation)*

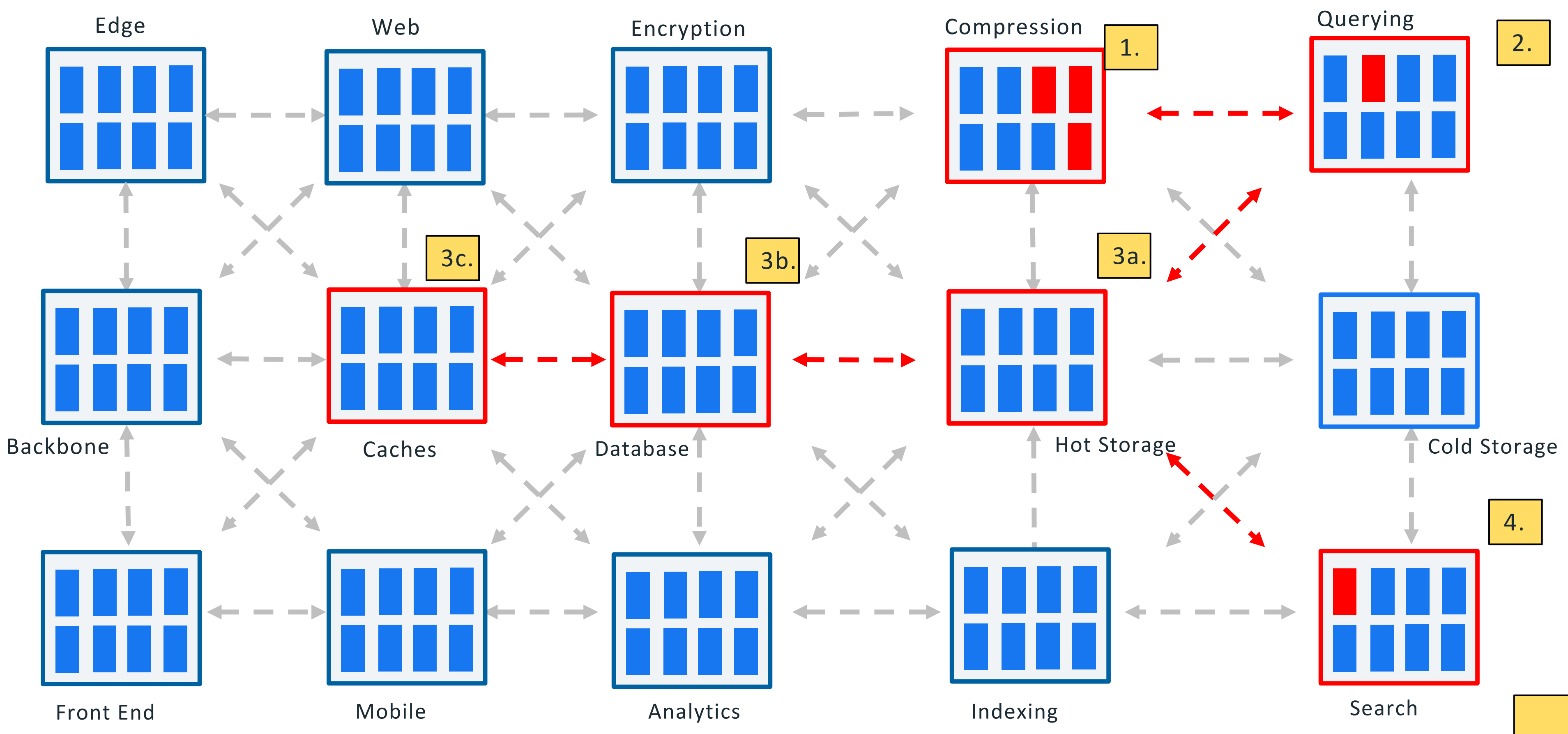
Systems at Scale



At a server level, services translate to numerous machines executing transactions with large fanout

All machines assumed to be of the same size for illustration

Systems at Scale



Result:
Numerous services impacted
to varying degree.

Corruption leads to cascaded impact across the fleet
Significant impact to services

*All machines assumed to be of the same size for illustration

Detecting silent data corruptions

Why is this a hard problem ?

Data Randomization

Eg: 3 x 5 = 15 but 3 x 4 = 10

Electrical Variations (V, I, f)

Eg: 3 x 5 = 15, but repeated
3 x 5!= 15 across device characteristics

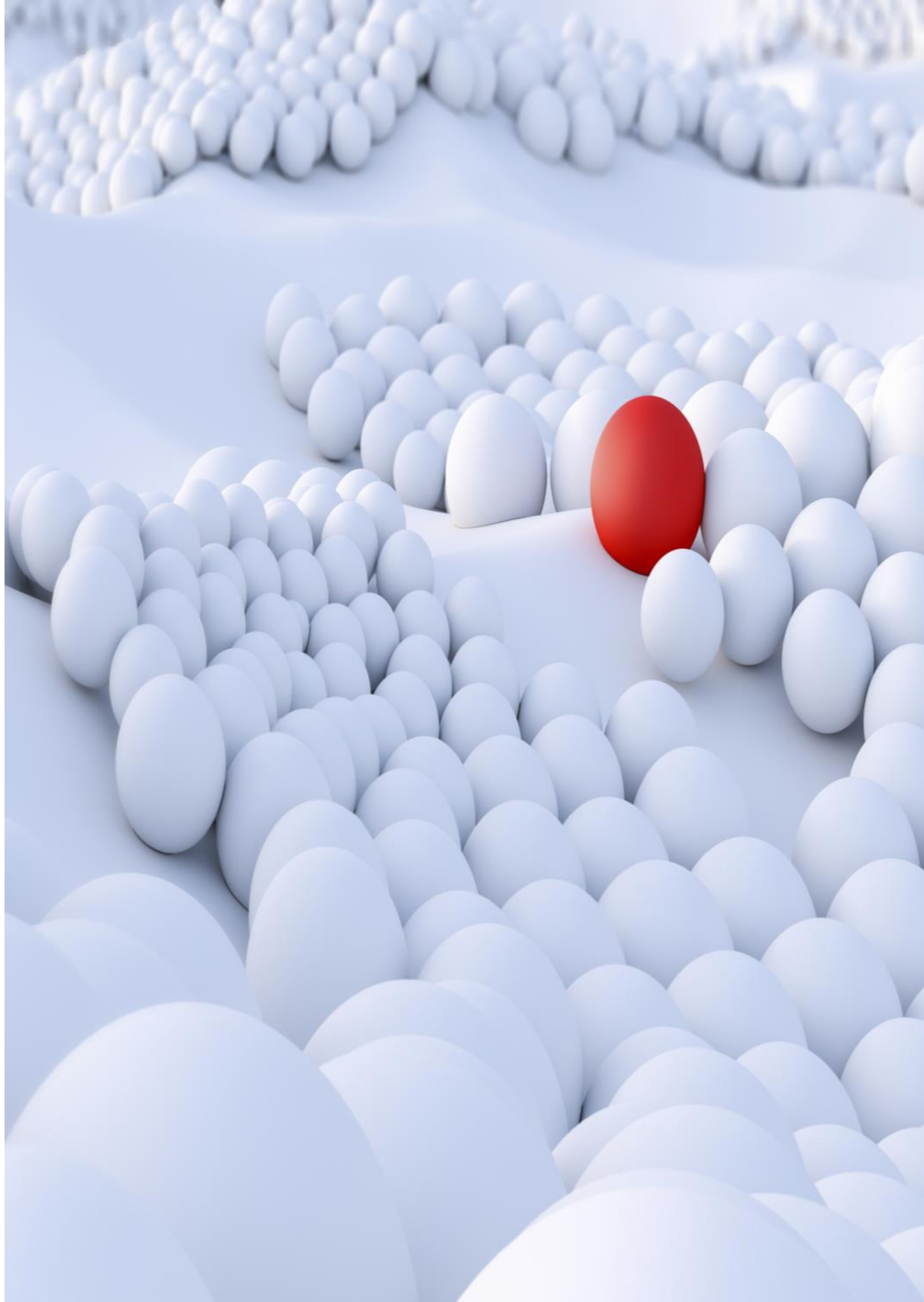
Environmental Variations (changing T, regional factors etc)

Eg: 3 x 5 = 15, but repeated
3 x 5 != 15 in all regions

Lifecycle Variations

Eg: 3 x 5 = 15 today
but tomorrow 3 x 5 = 13

From 1 SDC to 100s – How did we scale our approach?

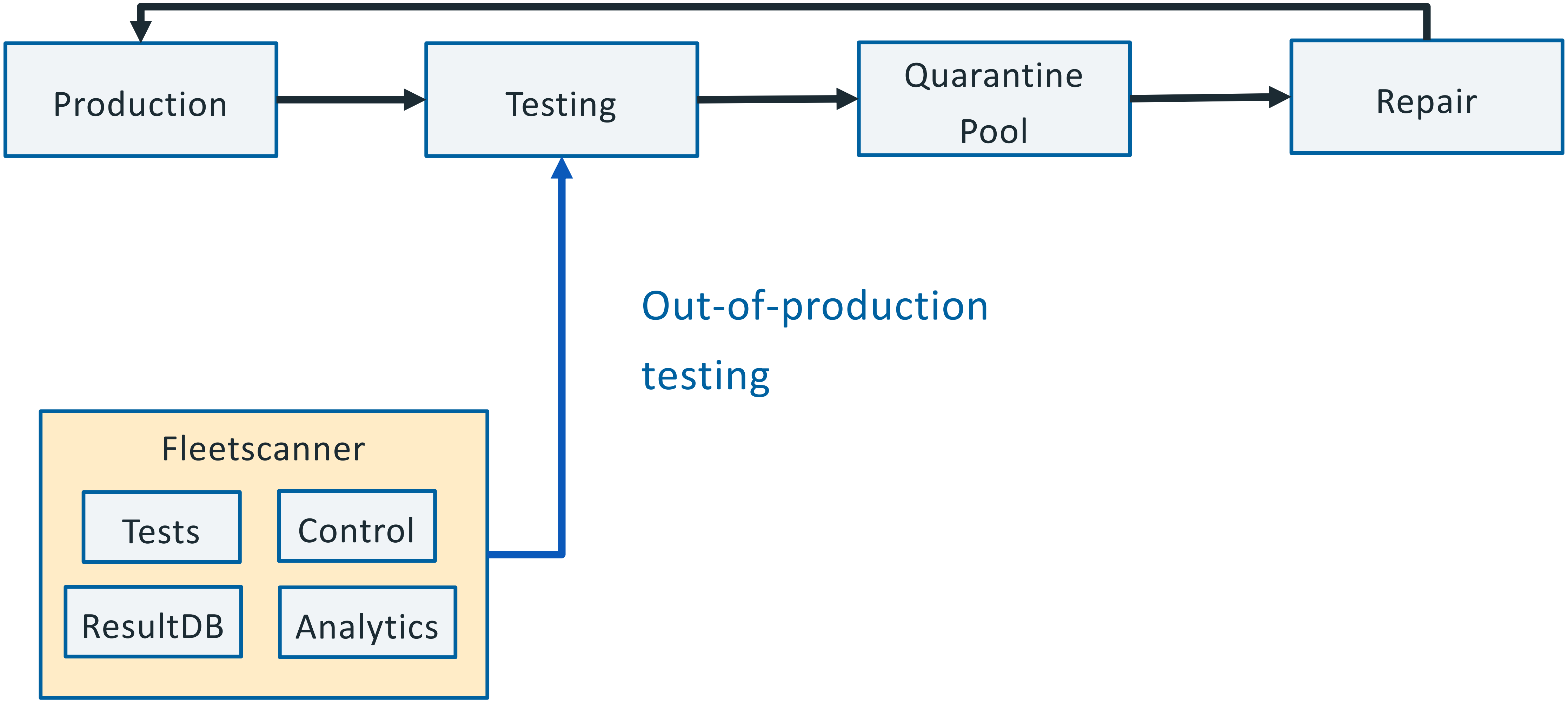


Test continuously in the fleet

- Fleetscanner (Out-of-production testing)
- Ripple (In-production testing)

In addition to – testing at the manufacturer and at datacenter intake.

Test continuously in the fleet (Fleetscanner)



Fleet Scanner

Taking pitstops to run tests!

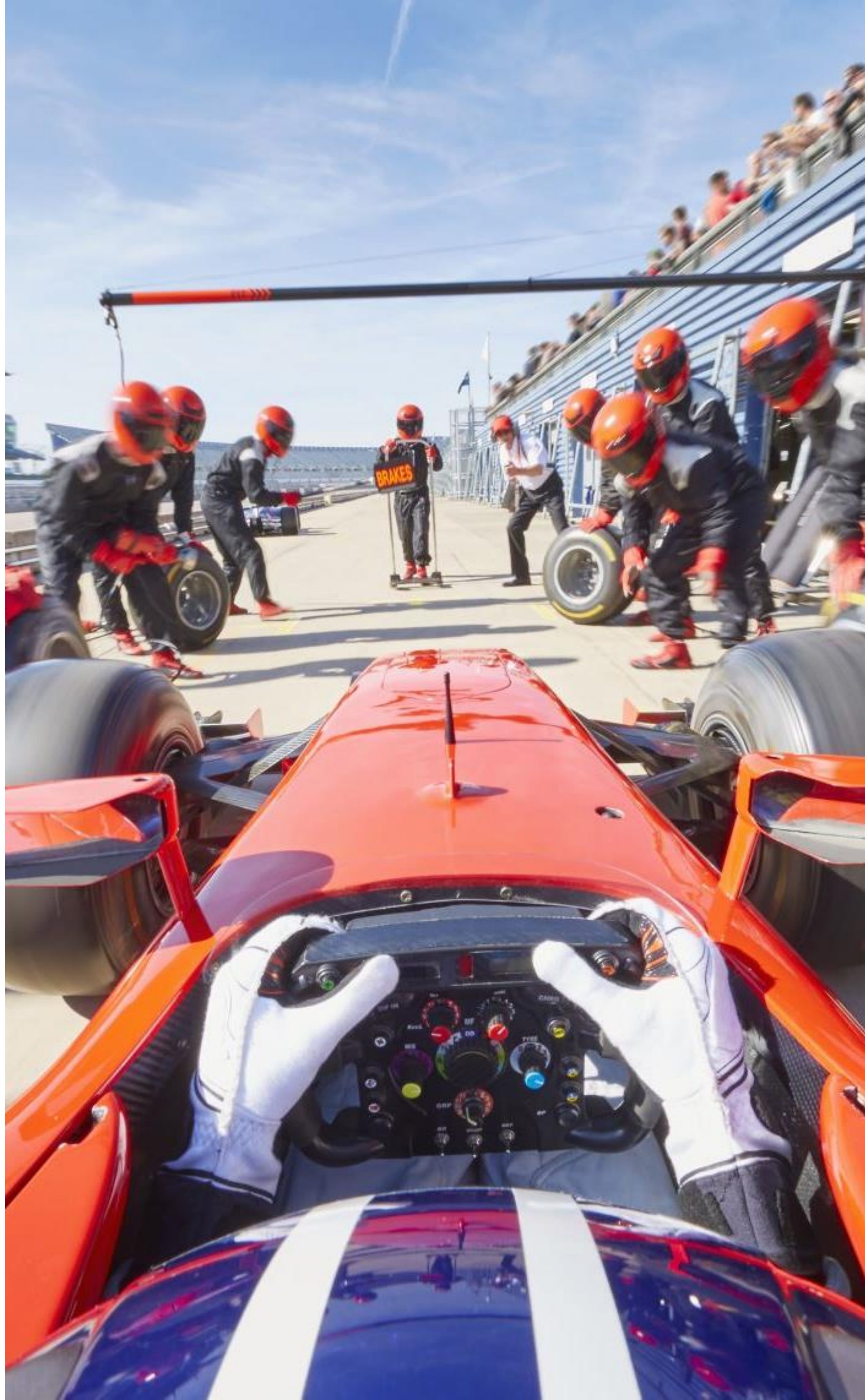
- Non-Production States
- Run directed tests
- Test time: Order of minutes
- Time to fleet coverage: 6 months
- 100s of devices detected with silent errors

4B

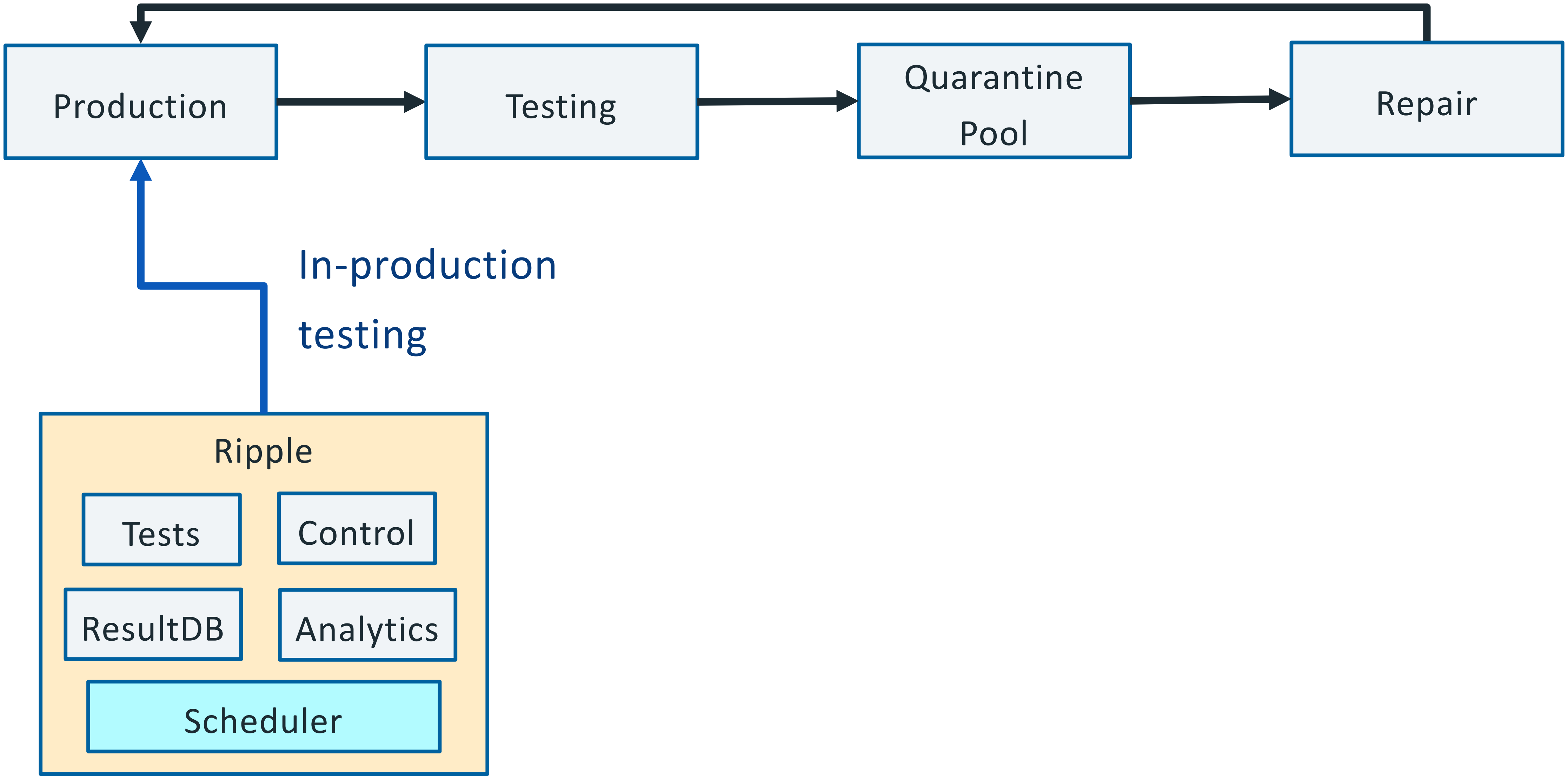
fleet seconds (lifetime)

FLEET TESTING TIME

BUT THIS IS TOO SLOW ACROSS A LIVE FLEET…..



Test continuously in the fleet (Ripple)



Ripple testing

Testing along with workloads

- Workload colocation
- In-production tiny tests
- Test time: Order of milliseconds
- Time to fleet coverage: 15 days

100M

fleet seconds per month

FLEET TESTING TIME

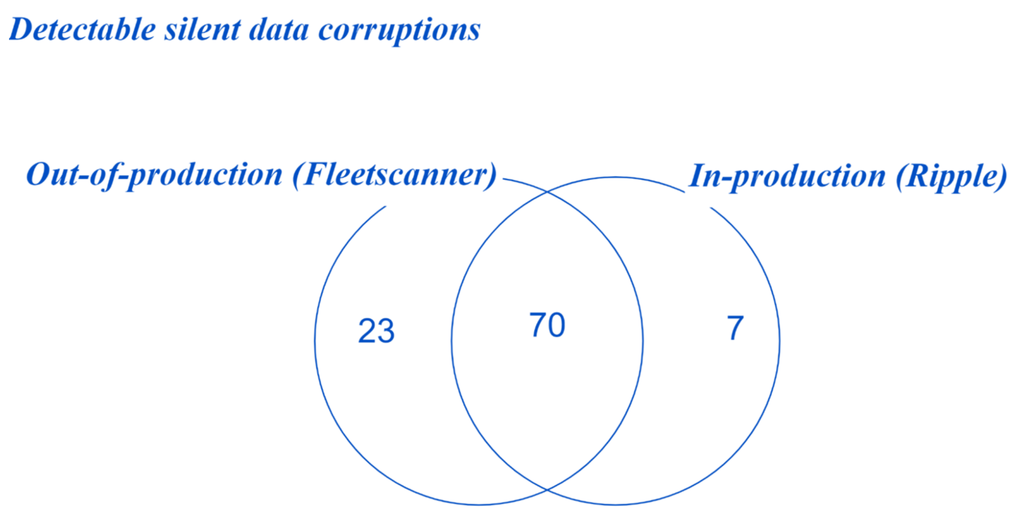
BOTH METHODS OF TESTING PROVIDE UNIQUE COVERAGE!

Blog - https://engineering.fb.com/2022/03/17/production-engineering/silent-errors/
Paper - https://arxiv.org/abs/2203.08989

Key Results

3 years of infrastructure testing using both mechanisms (for a large defect family)

Metric	Fleetscanner	Ripple
Test Iterations	~68M (lifetime)	~2.5M (per month)
Performance aware	No	Yes
Time to equivalent SDC coverage	~ 6 months (70%)	~ 15 days (70%)

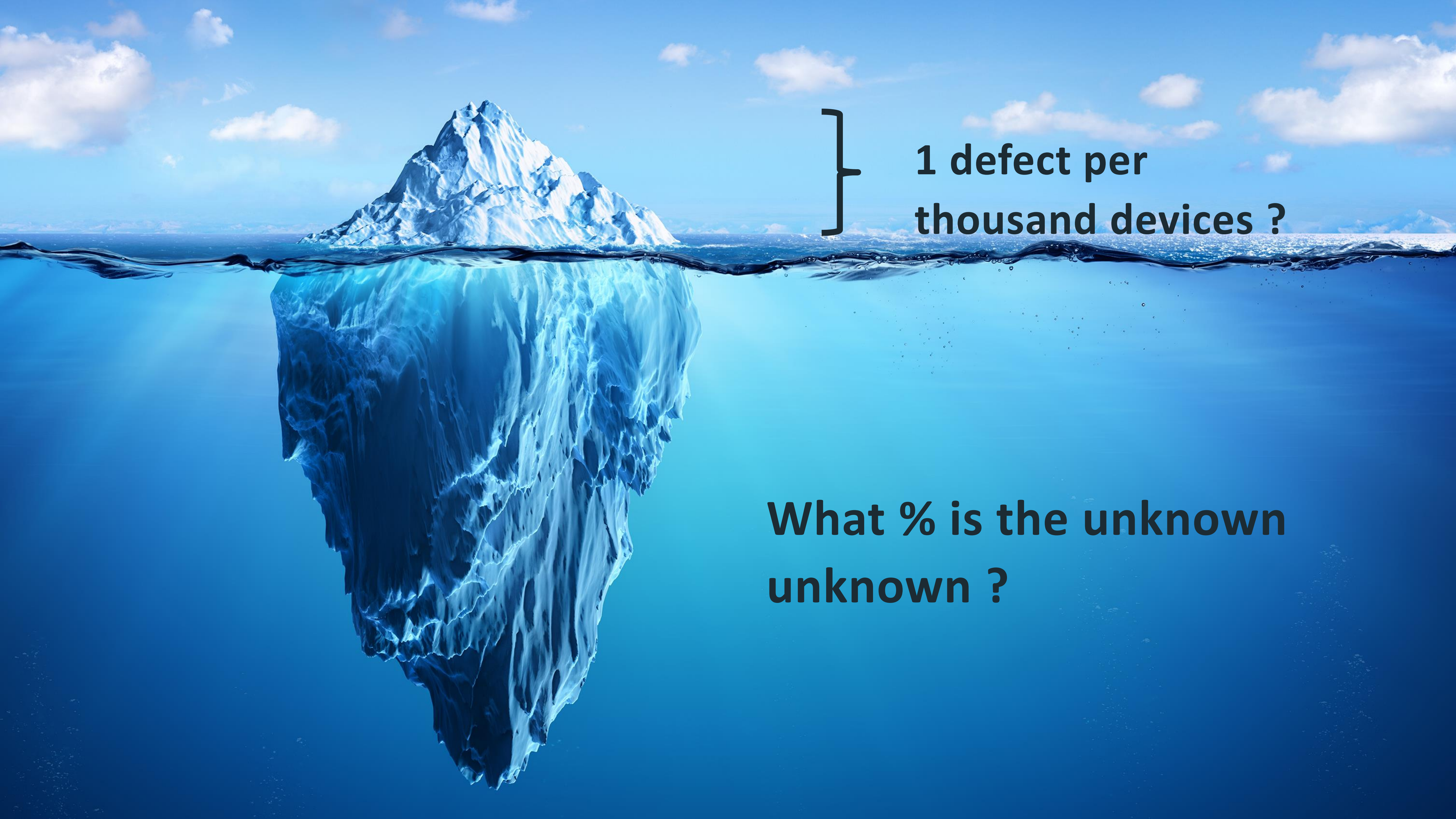


Data Snapshot: March 2022

Interesting observations:

$$Int[(1.1)^3] = 0, expected = 1.$$
$$Int[(1.1)^{107}] = 32809, expected = 26854.$$
$$Int[(1.1)^{-3}] = 1, expected = 0.$$

- Compiler and optimization dependent.
- Impacting computations involving non-zero operands, results, with varying degrees of precision.
- Impacts wide variety of applications.
- Impacts multiple instruction types and functional subblocks
- Vectors, Floating computations, large data moves, gather-scatter ops, encryption, shared memory coherency, string corruptions etc.
- Instances of impact in coldstorage (backups), presto (queries) and kernel code etc.


1 defect per
thousand devices ?
What % is the unknown
unknown ?

What's next ? – Dealing with SDCs

HARDWARE DESIGN STRATEGIES	PRODUCTION FLEET STRATEGIES	SOFTWARE & APPLICATION STRATEGIES
Fault tolerant designs	Manufacturing validation	Algorithmic Fault Tolerance
Data Protection Priorities	Out of production testing	Probabilistic Redundancy
Specialized Screening	In production testing	Software fault correlation
@scale collaboration		

With increased silicon density and technology scaling,
we are more likely to see silent data corruptions in future CPUs and ASICs.

Silent errors are a foundational computing problem!

Thank You!!

Meta

INFRASTRUCTURE