What Brought Us Down?

Outage Trend Analysis at Google

SRECon 2015

Sue Lueder

About me

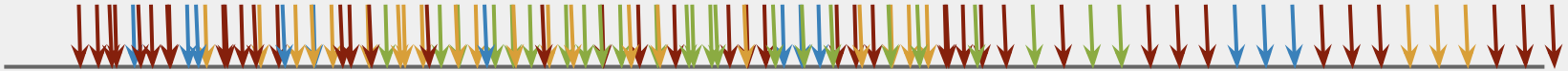
Sue Lueder

- Google SRE Program Manager
- Pre-Google: Wireless Software and Systems engineer
- Relevant Outside Interests: MS Organization Development (people and teams matter), Quantified Self, How to Measure Anything

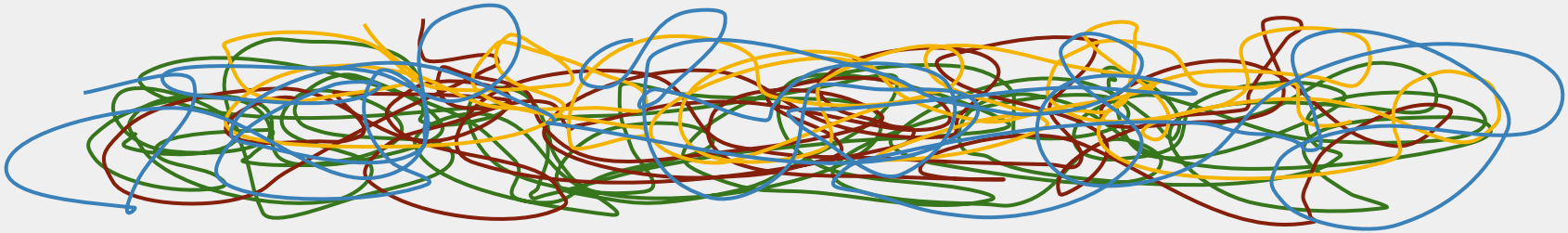
Changes in Production

Configuration Changes — Software Changes — Production Changes — User behavior

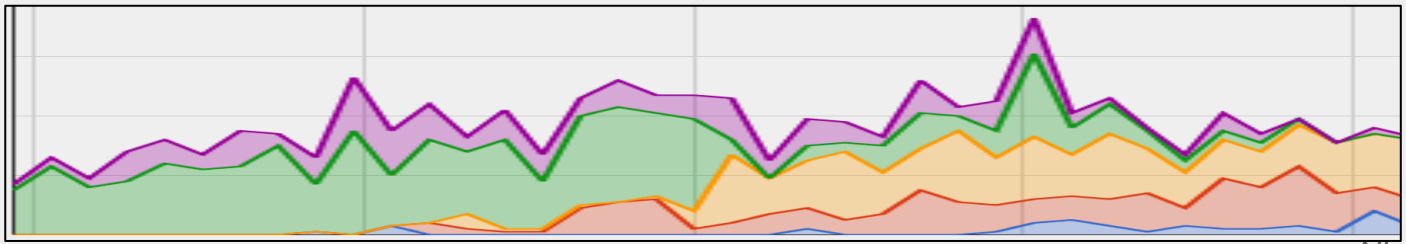
What non-SREs think it looks like:



What SREs know it sometimes feels like:



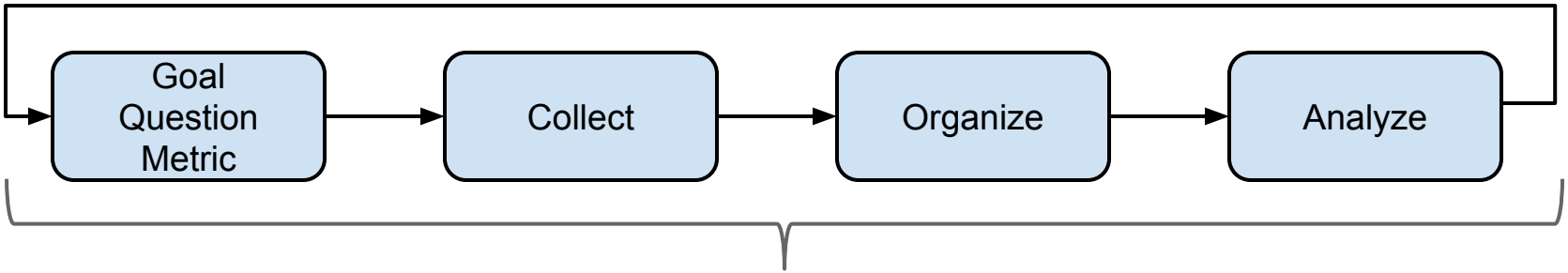
What the signals can look like (with some effort):



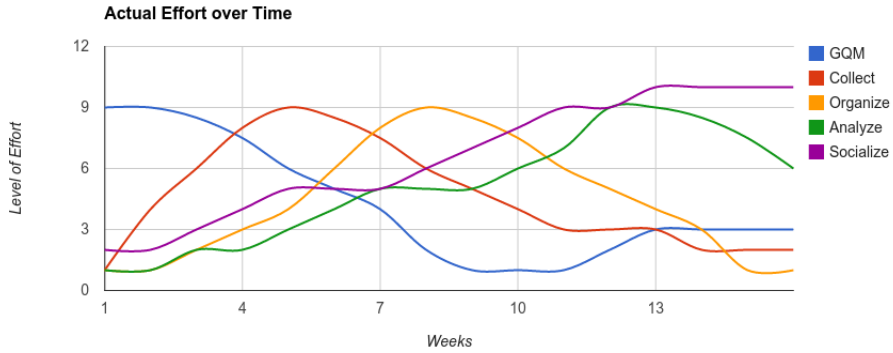
All data is fabricated

The Process in a nutshell

The Feedback Model



Socialize data



All data is fabricated

Goal Question Metric

Goal: Best in Class Incident Resolution Times

How long do we take to resolve incidents?

Are we faster at resolving certain incident types over others?

Are we getting better or worse in our resolution timing?

Time to Resolve
Number of action items in postmortems related to monitoring and debugging tools
Time to resolve for cascading issues
Number of user visible issues
Number of user visible issues found by users
Time to escalation
Resolution performance charted over time

Goal Question Metric (Example #2)

Teams are learning from systemic issues and improving processes

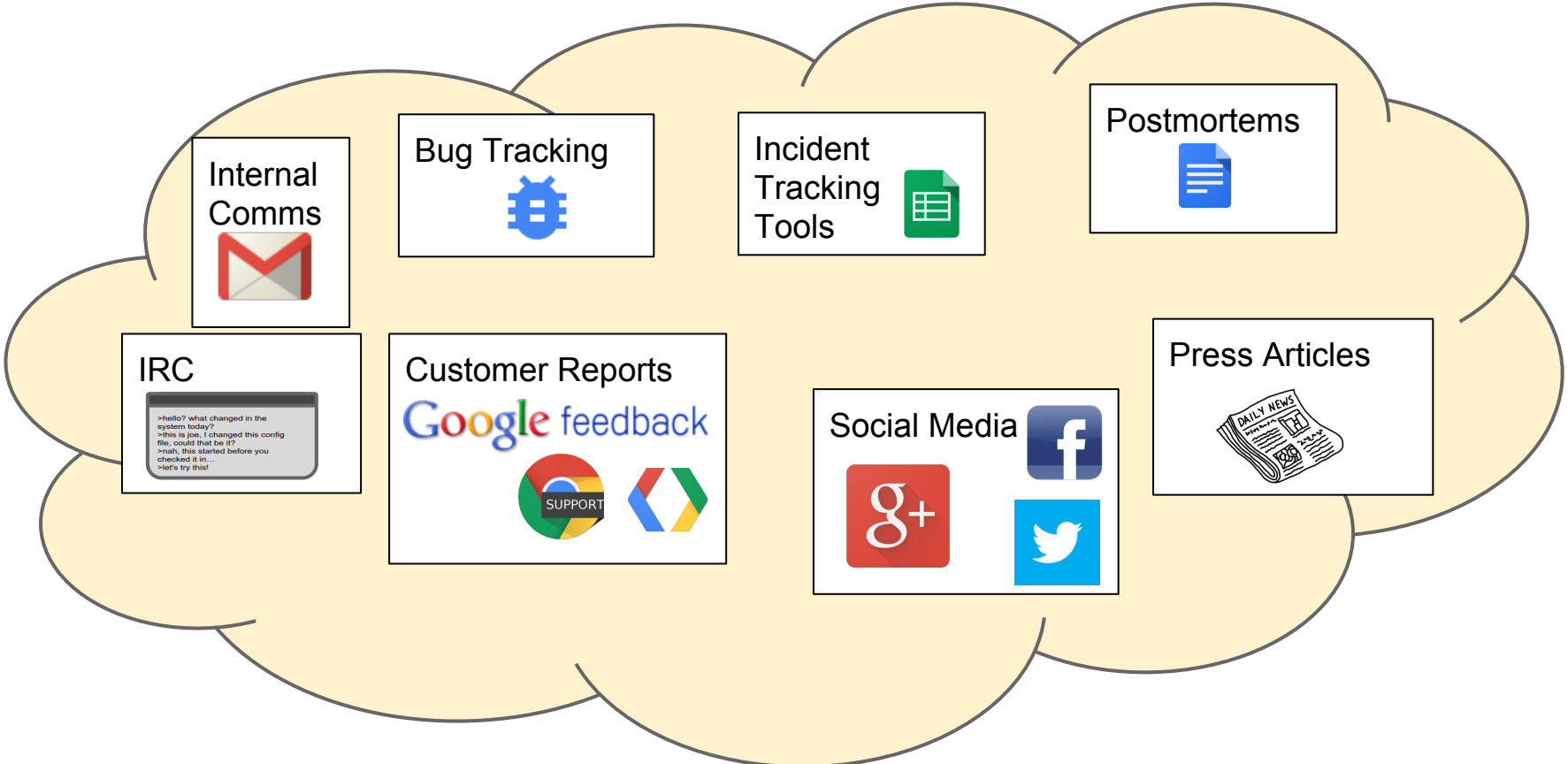
Are post mortems capturing the salient details?

Are teams able to learn from each other?

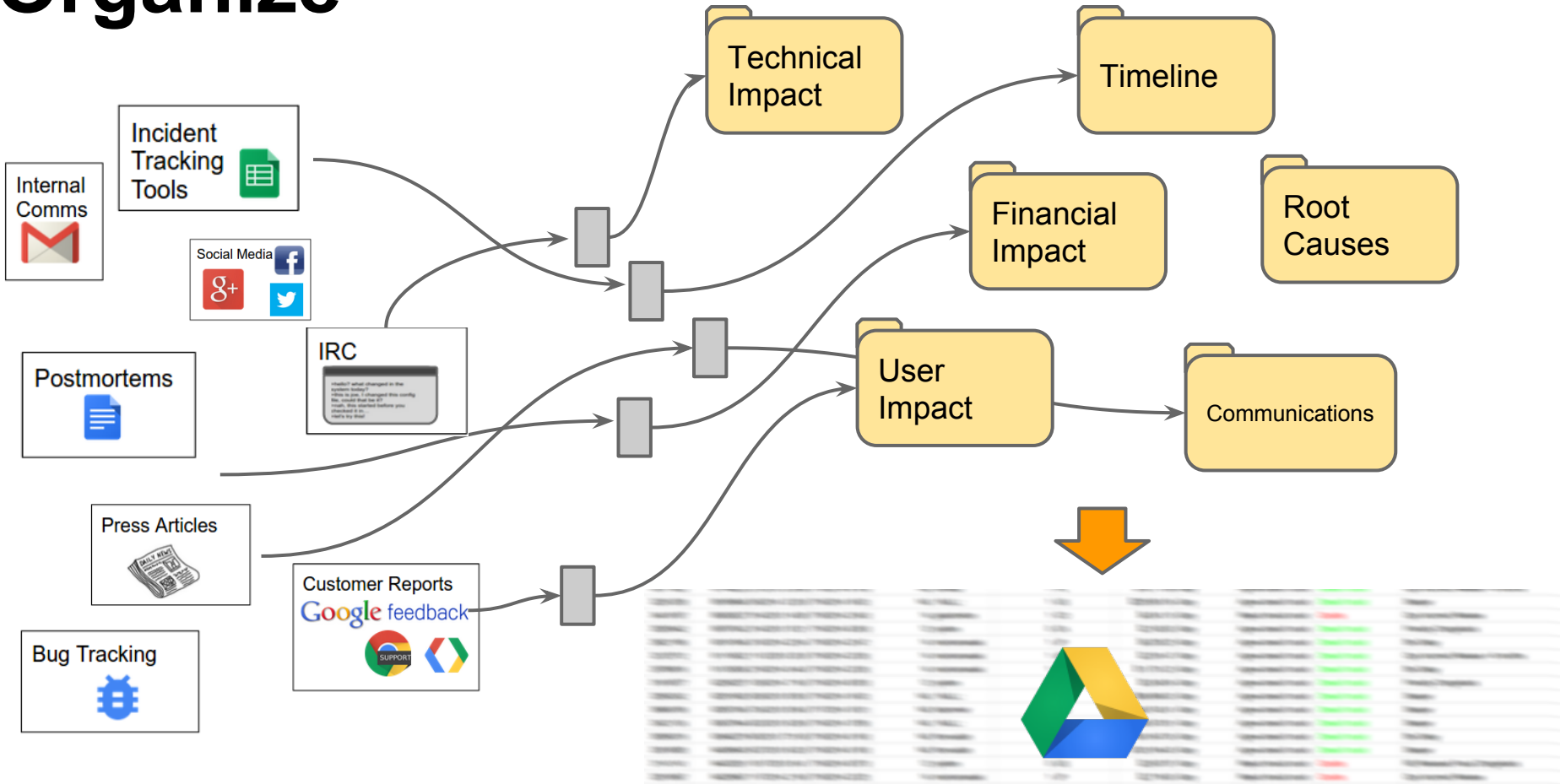
Are cut corners in behaviors causing outages?

Time to complete postmortems
Postmortem coverage of outages
Postmortems with thorough root cause analysis
Action item closure rate
Incidents over time
Incidents due to workflow breakdowns
Incidents due to planning failures
Postmortem review status

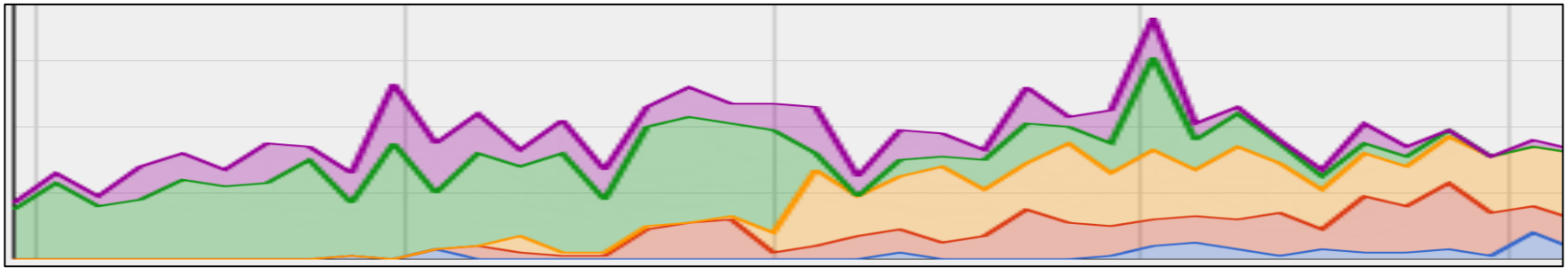
Collect your data



Organize


Internal Comms
Incident Tracking Tools
Social Media
IRC
Postmortems
Press Articles
Bug Tracking
Customer Reports
Google feedback
Technical Impact
Timeline
Financial Impact
Root Causes
User Impact
Communications

Analyze (the fun part)

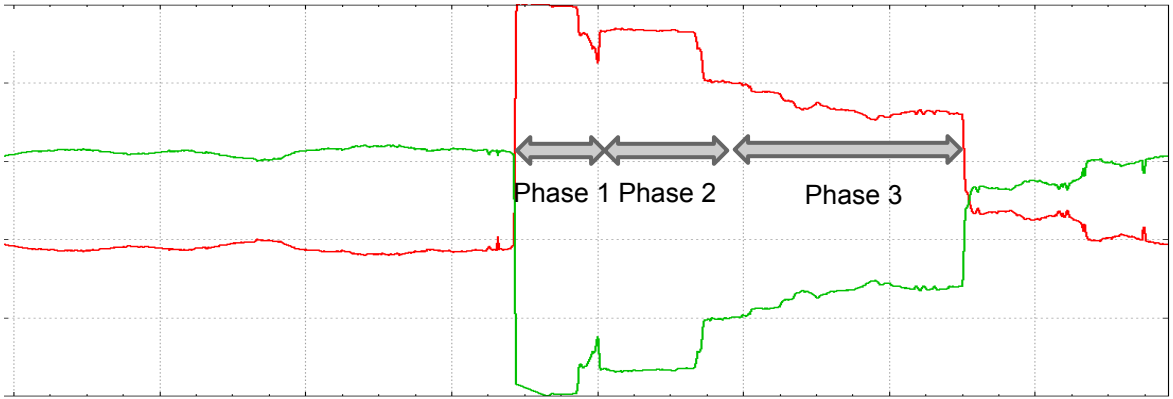


All data is fabricated

Incident Timeline

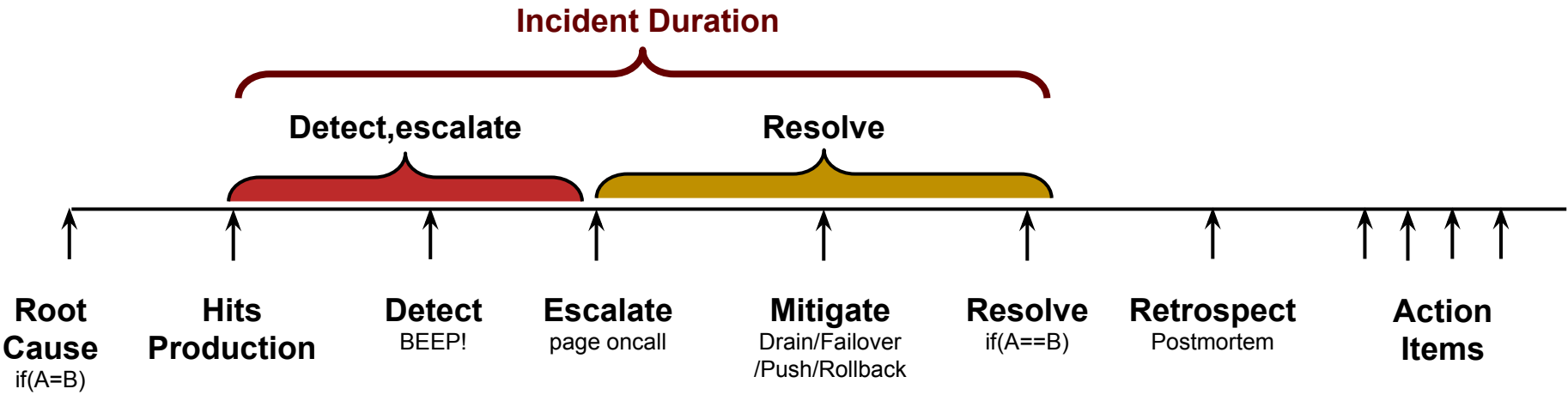
Challenges of Measuring Incident Timing

- SREs in Multiple Timezones
- Various tools to pull timestamps from
- Multiple Incident Phases

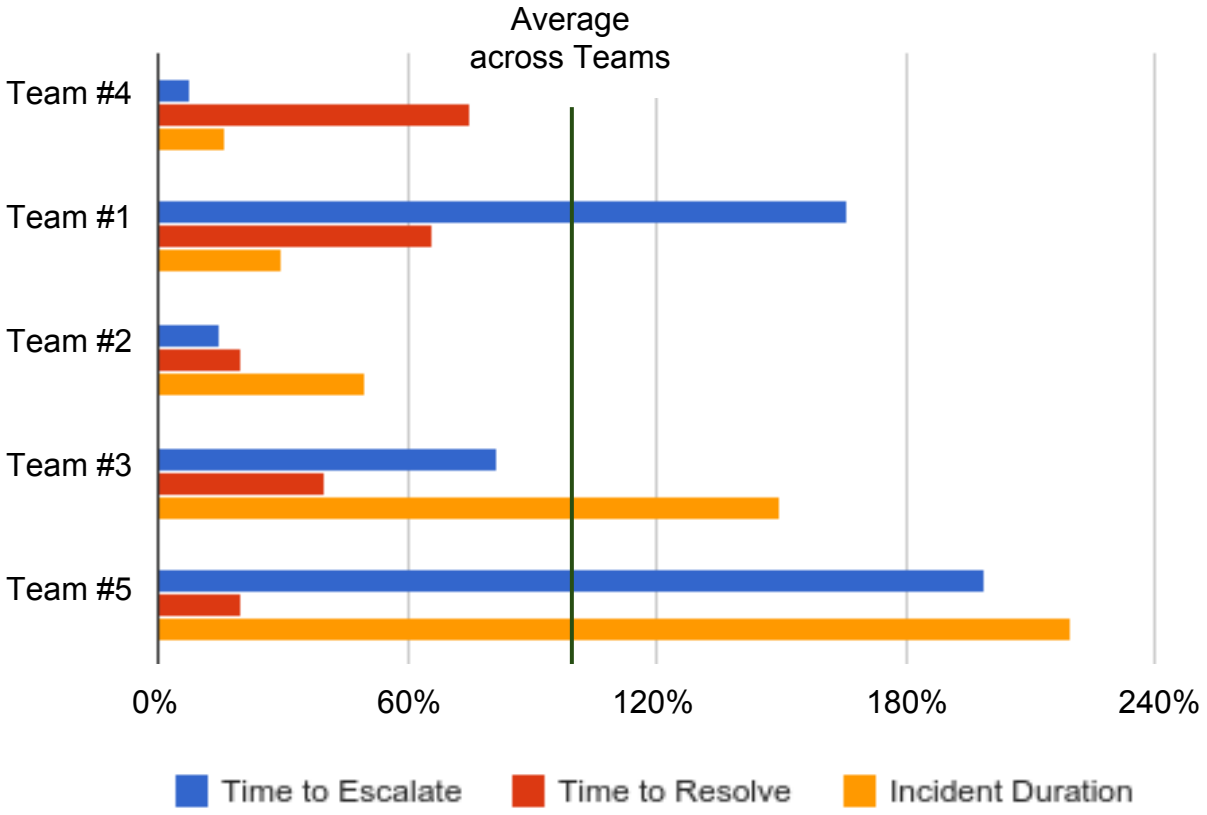


All data is fabricated

Incident Timeline



Looking at Team Incident Timing

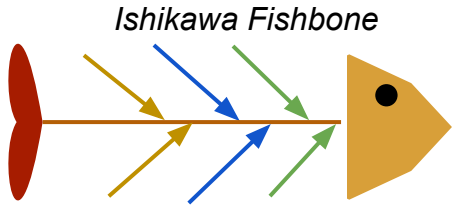


All data is fabricated

Root Cause Analysis

Challenges of Root Cause Analysis

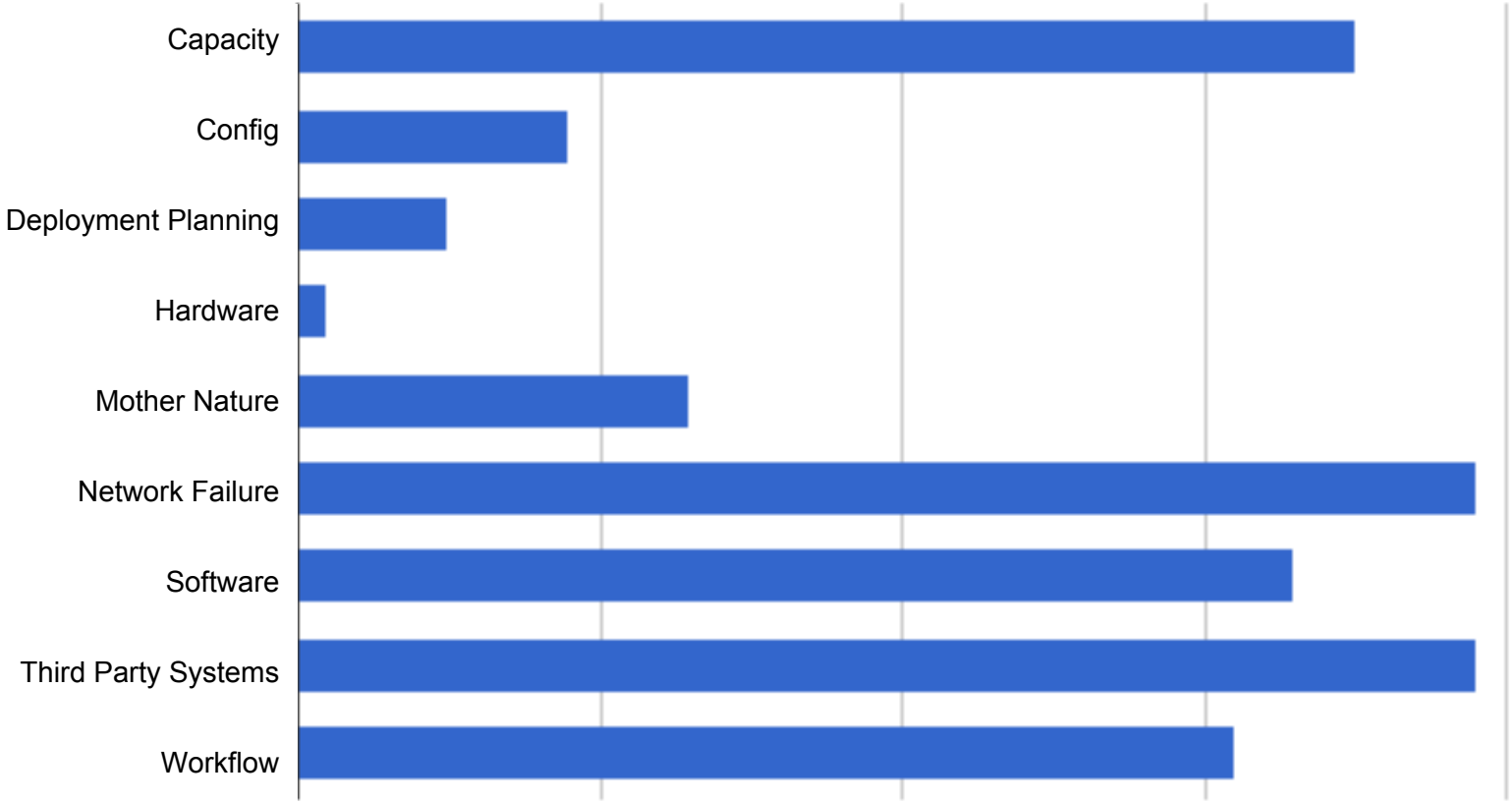
- Root cause depth
- Root cause category alignment
- What about "human errors"?



Root Cause Categories

Category	Sub Categories
Capacity	Memory, CPU, Network Bandwidth, Concurrency, Hardware
Deployment Planning	Production Change, Dependencies
Software	Interface, logic, multithread, concurrency, performance, resource, syntax
Workflow	Communications, Requirements Misunderstanding, Uncaught Mistakes in Code, Missing Failsafe
Network Failure	3rd Party Networks, DesignProblem, DesignViolation, Environmental, Hardware, Power, ProcessFailure, Provider, ServiceConf
Third Party Systems	
Config	
Mother Nature	
Hardware	

Frequency of Root Causes

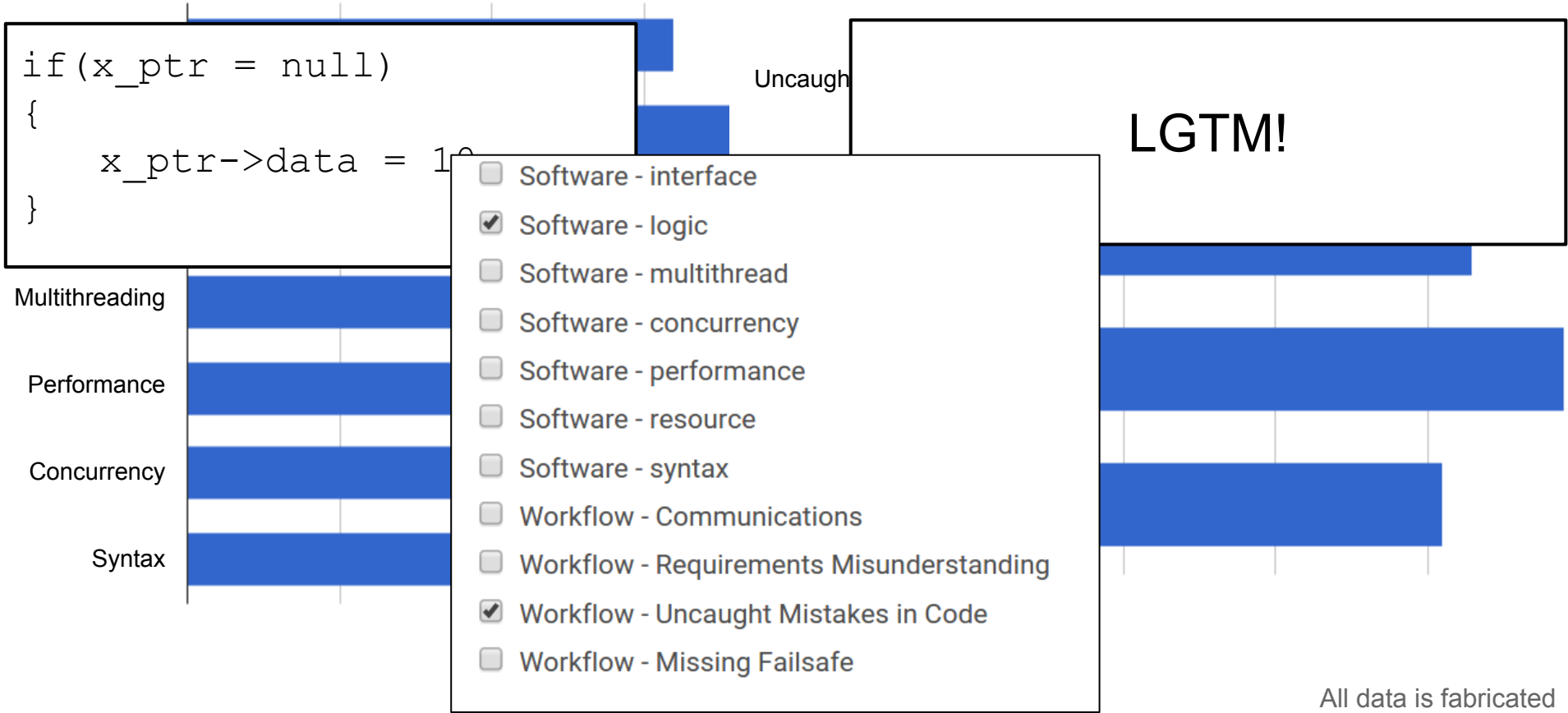


All data is fabricated

Software

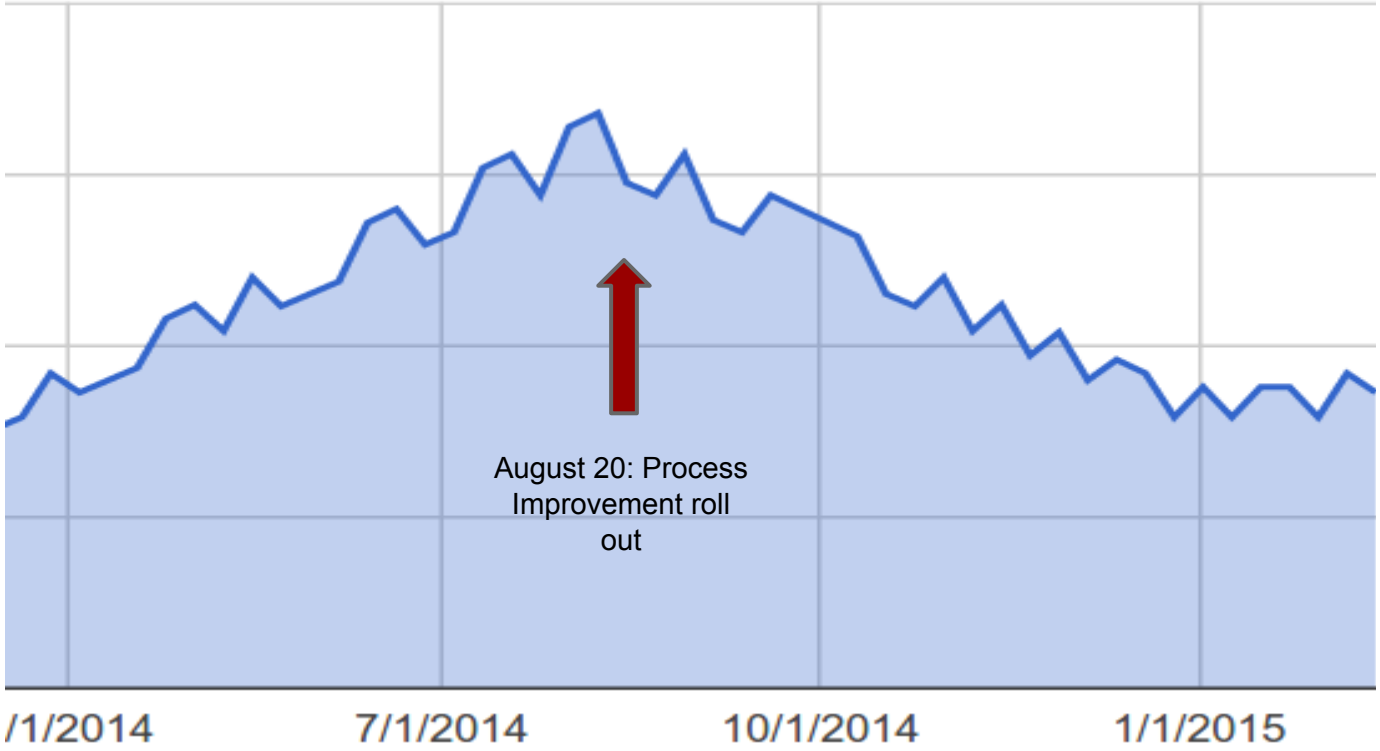
Workflow

```
if(x_ptr = null)
{
    x_ptr->data = 1
}
```



All data is fabricated

Tracking improvements

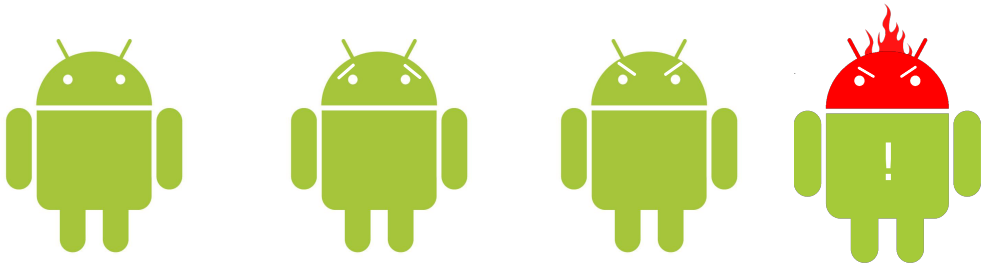


All data is fabricated

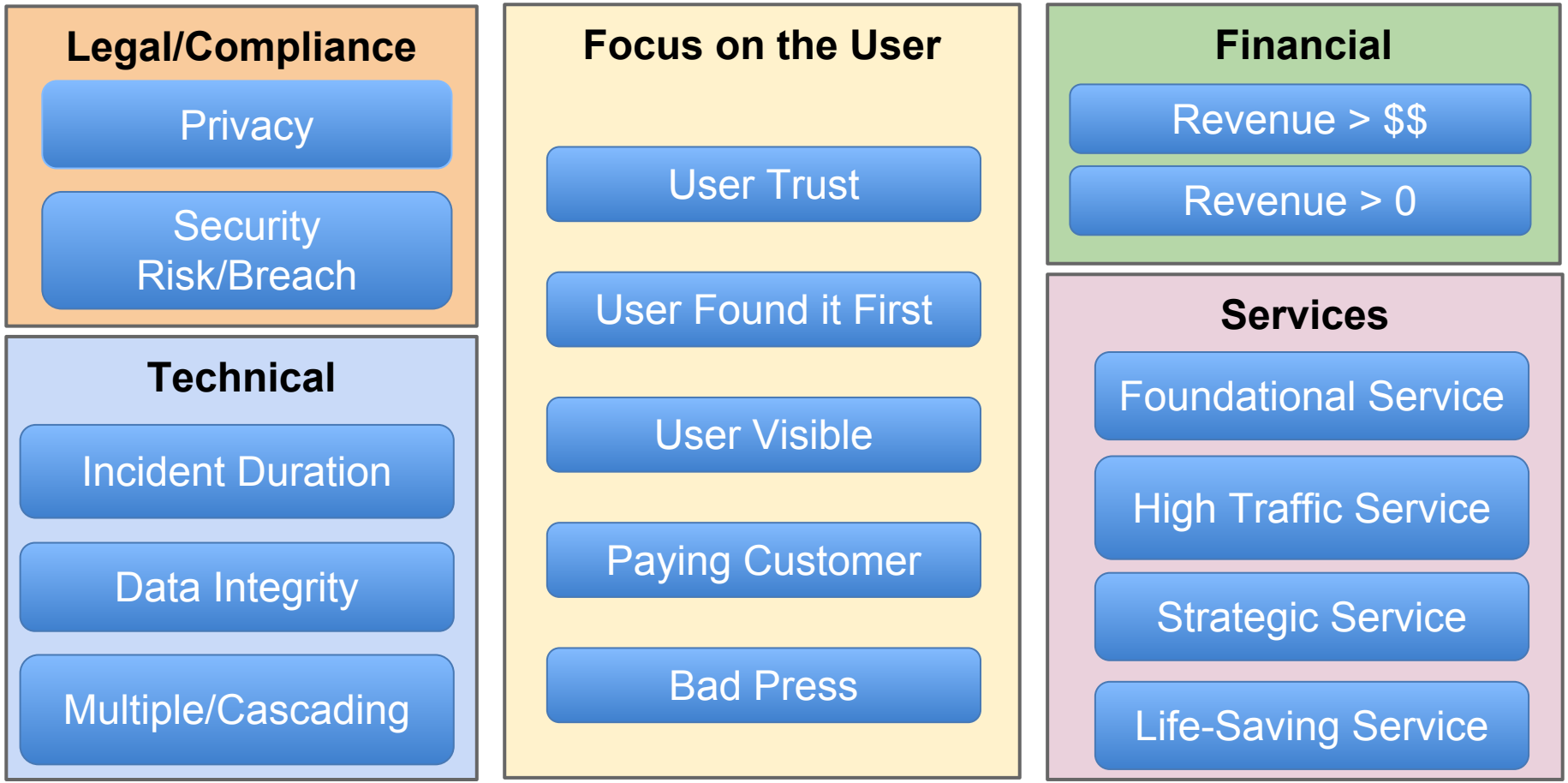
Incident Severity

Challenges of Measuring Severity

- Normalizing severity across teams
- Signal to Noise Ratio
- Limited Time and Resources
- Boiling Frog Problem
- Squeaky Wheels



Severity Flags



Severity Flags enable…

Weighted Severity Calculations to show top issues and trends by "perspective"

Calculated Severity= SecurityFlag*securityweight
+ UserFound*userfoundweight
+ UserVisible*uservisibleweight
+ PayingCustomer*payingcustomerweight
+ LostRevenue*lostrevenueweight
+ GreatLostRevenue*greatrevenuelossweight
+ Cascade*cascadeweight
+ BadPress*badpressweight
+ Privacy*privacyweight
+DataIntegrity*dataintegrityweight
+ UserTrust*usertrustweight

Finance

Public Relations

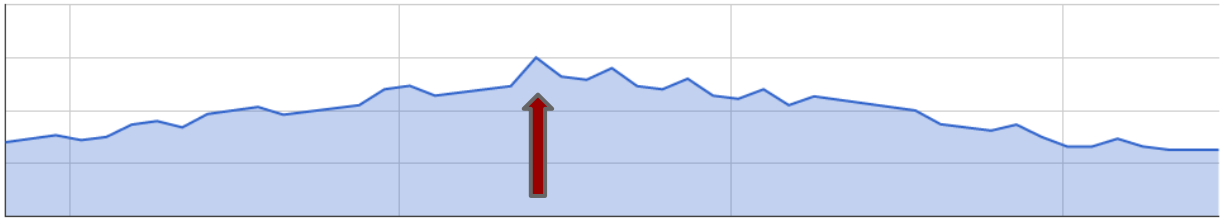
Quality

All data is fabricated

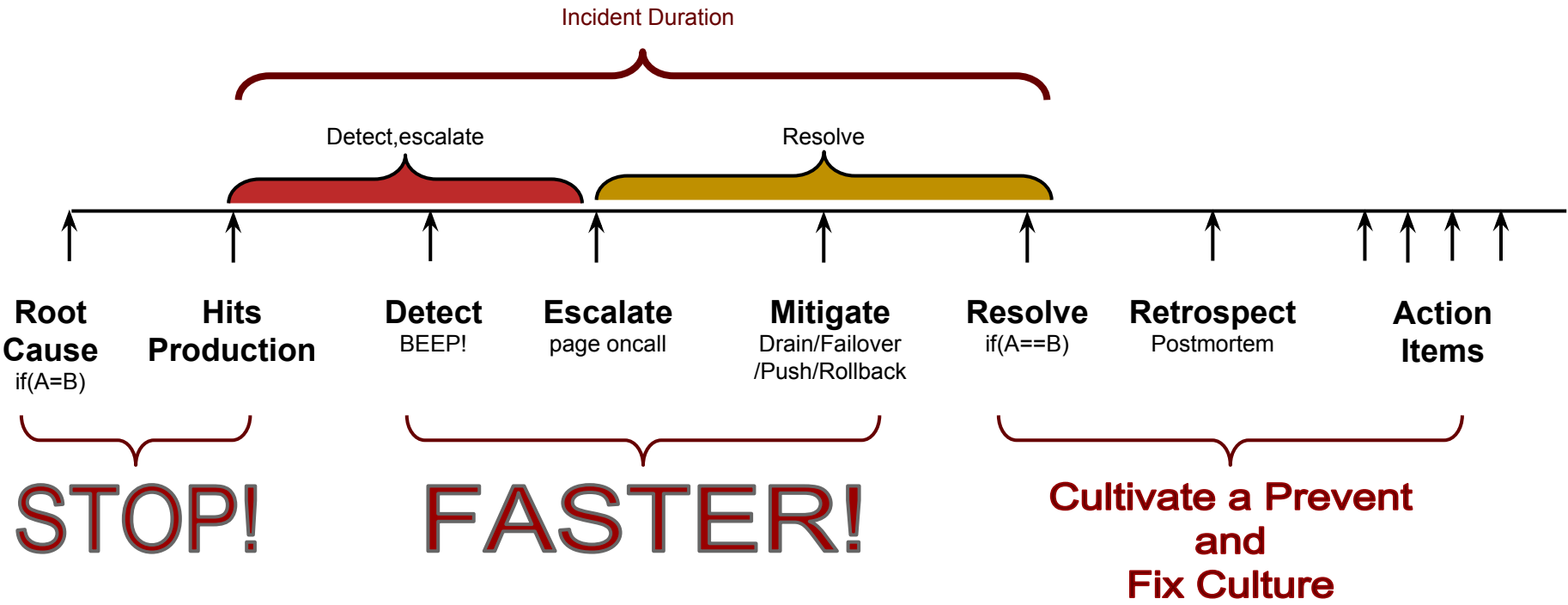
Now What?

Challenges of Designing Fixes

- Change Fatigue
- Sufficient time
- Limited SRE Resources
- Cause & Effect
- Power of measurements



Finding Fix Opportunities



Lessons Learned

What doesn't work

- Excessive manual effort
  - Automate early and often
- SPOF
  - Find a way to shard and crowdsource data collection and analysis
- Making assumptions about what you'll find
  - Listen to the data and leave room for new discoveries
- Limiting to just SRE perspectives
  - Think about the whole life of an incident and share/include widely

What Works

- **Engage** stakeholders formally and informally
- **Socialize** Data Early and Often
- **Intentionally** design, execute, and measure fix initiatives
- Use **readily available** tools
- Start with and maintain a **flexible** data schema

Thank You!

"**In God we trust**, **all others bring data**."

- William Edwards Deming