Splunk .conf18
Post Incident Reviews

Why You Should Care & How to Get Started

Davis Godbout, Product Manager

October 2018 | Version 1.0

Forward-Looking Statements

During the course of this presentation, we may make forward-looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC.

The forward-looking statements made in this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward-looking statements we may make. In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

4:03pm

Something’s Amiss

Sam notices first @ mention
on Twitter; customer can’t access
Acme.com

Unfortunately:

2 cases in queue
Only checks once queue is empty

4:28pm

There’s a Problem

Sam logs into Twitter and sees @mentions of customers complaining

Oh No!

Who can help?

4:42pm

Investigation Begins

Cathy verifies service disruption. Sam files a ticket in the customer support team

First step

Cathy begins looking for investigation and triage documentation

4:53pm

Locating the Problem

Cathy locates documentation on methods to connect to the service hosting the site

Action Recorded

Cathy logs in to the affected server

5:02pm

A First Attempt

Cathy views all running processes on the host using ‘top’, spots an unknown service utilizing 92% of the CPU

Reaction

Cathy attempts to contact Greg via slack (he is the most familiar with processes that run on the host)

5:12pm

The Search for Greg

After 5 minutes of slack messaging, Cathy gives up on Slack and begins looking for Greg’s phone number

Unfortunately she can’t find it easily and searches through old emails

Success

Cathy finds Greg’s number in an email from a year ago and calls him

5:15pm

Keep Everyone Updated

Greg joins the investigation process and asks Cathy to update the StatusPage

Actions Recorded

StatusPage is updated and Sam mentions he has received 10 support requests and 3 additional mentions on Twitter

5:33pm

Resolution

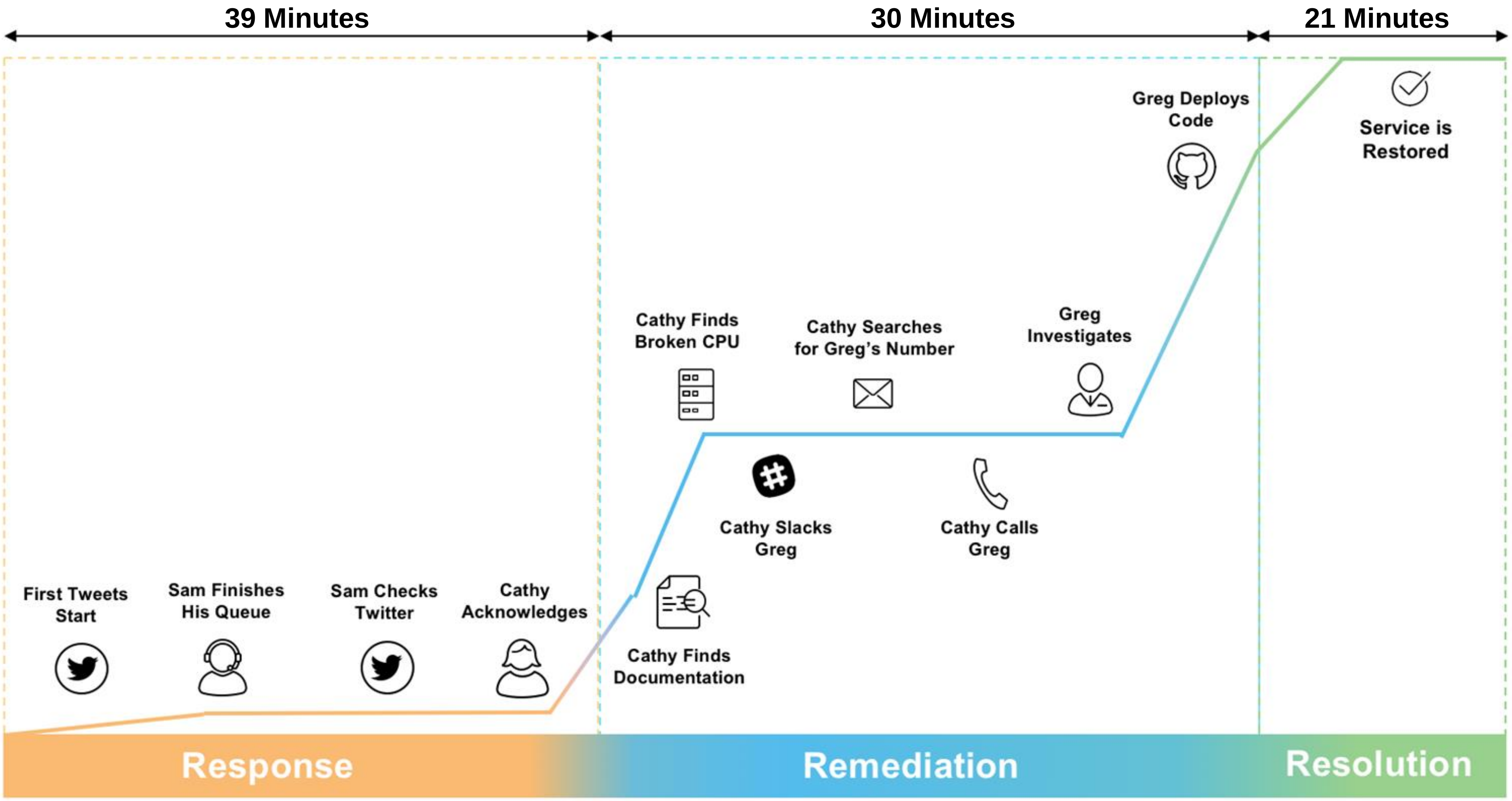
Greg fixed the issue and informs the team that he has solved the problem

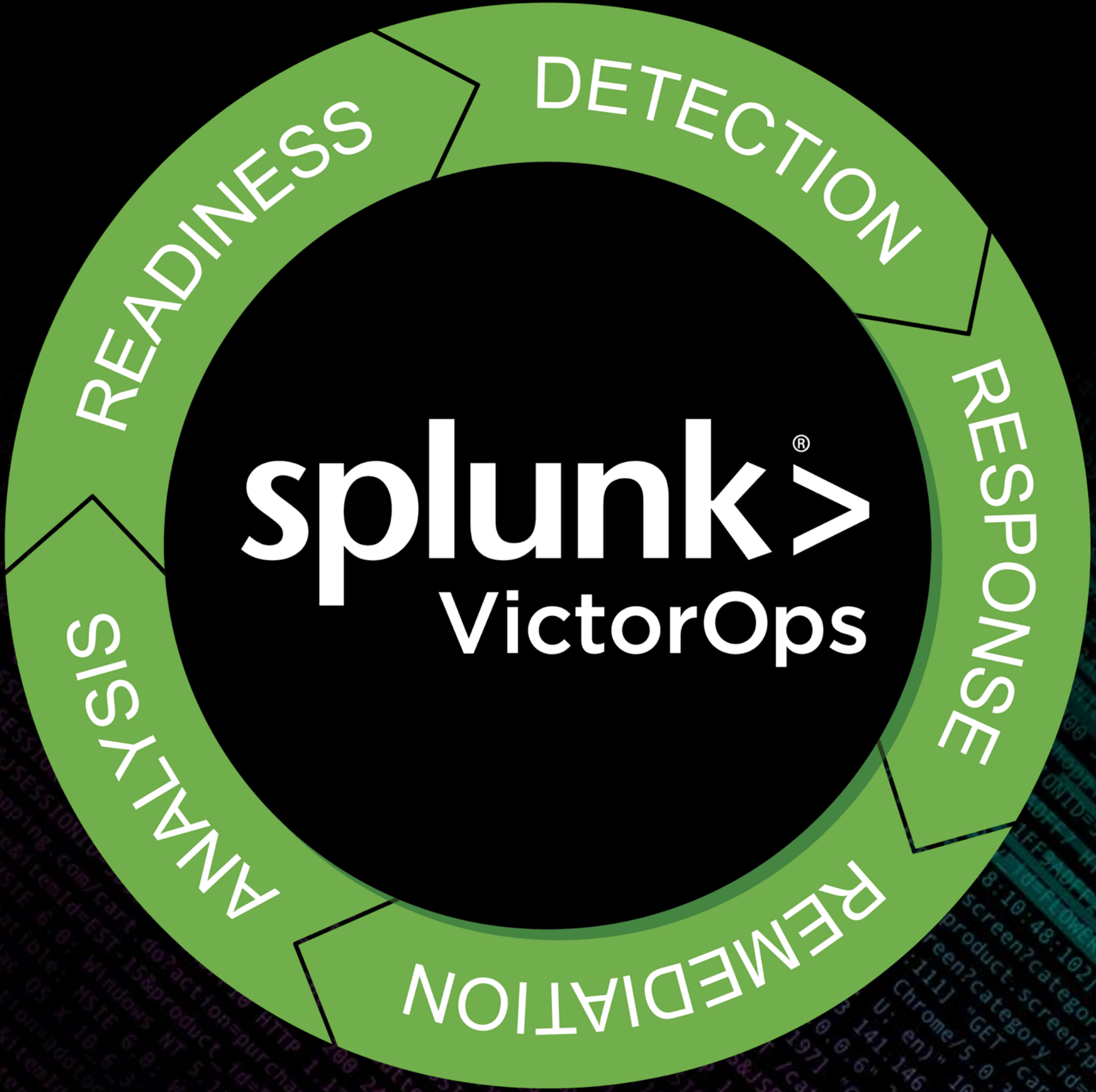
Final Steps

Sam closes the support ticket and updates the StatusPage

Incidents don’t end after resolution

Total Time: 90 Minutes





The Virtuous Cycle of DevOps

Detection

We didn’t detect this on our own.

We don’t have a clear path to responding to incidents (Support contacted Cathy as a result of chance not process)

Time To Acknowledge: 39 Minutes

Detection Checklist:

- Are we capturing an incident timeline of events?
- How and when did we know about this problem?
- Do we have a process to address system failures?

Response

It’s not common knowledge how to connect to critical systems regarding the services we provide.

Access to systems for the first responder was clumsy and confusing.

Pulling in other team members was difficult without instant access to their contact information.

We aren’t sure who is responsible for updating stakeholders and/ or the status page.

Time to Response: 30 Minutes

Response Checklist:

- Did we have accessible documentation to understand how critical systems work?
- Did responders have access to the appropriate systems?
- Is contact information readily available?
- How did we communicate outward?

Remediation

A yet-to-be identified process was found running on a critical server.

We don’t have a dedicated area for the conversations that are related to the remediation efforts. Some conversations were held over the phone and some took place in Slack.

Someone other than Greg should have been next on the escalation path so he could enjoy time with his family.

Time to Recover: 21 Minutes

Remediation Checklist:

- What was the point when we identified the problem?
- Where did we coordinate a problem response?
- Do we have on-call coverage for critical people out of office?
- What tasks were performed?
  - What made a positive impact?

> “Learning from a postmortem is only as useful as what you put into practice afterwards and we realized that without any action items after the meeting, it was more or less just a Greek Senate debate.”

Ben VanEvery (Simon Data)

Readiness: Detection

- Add **monitoring of effected host** to detect potential or imminent problems
- Set up an **on-call rotation** so everyone know who to contact if something like this happens again
- Define **escalation policies** and alerting methods for engineers

Readiness: Response

- Build and make widely **documentation on how to get access to system** to begin investigating

- Build and make widely available **contact information** for engineers who may be called in to assist during remediation efforts

- Establish responsibility and process surrounding who is to maintain the **status page**

Readiness: Remediation

- Ensure that all responders have the **necessary access and privileges** to make an impact during remediation

- Establish **a specific communication client and channel for all conversation** related to remediation efforts and try to be explicit and verbose about what you are seeing and doing. Attempt to “think out loud”

The 3 Stages of Operational Maturity

Evolution of Incident Management

Analysis: 3 stages of maturity

“Reactive”

💩 happens

(no real process)

“Tactical”

“CYA”

post-mortem

“Integrated”

Post Incident Review

process feeds into

readiness

Where To Start

Step 1: Access to Data

Capture More Than Just the Monitoring Data

Observability of Systems

- Set up monitoring as part of feature release and deployment
- Remove “visibility silos” between infrastructure and applications

Observability of People

- What steps are teams taking to resolve incidents, can we solve this via automation?

Step 2: Change In Process

Problems With ‘Too-Narrow” Focus

Avoid the 5 Whys and RCA (Blame)

- Incidents are rarely due to one specific failure, instead they are a collection of systemic failures (lack of visibility/ knowledge, etc)

Build a timeline of events (both data and human interaction) to objectively analyze an incident

- Capture context within 72 hours of any major incident

Step 3: Empower People

People Built It, Help Them Fix It

Arm your team with training and knowledge

- Share information via runbooks and wikis

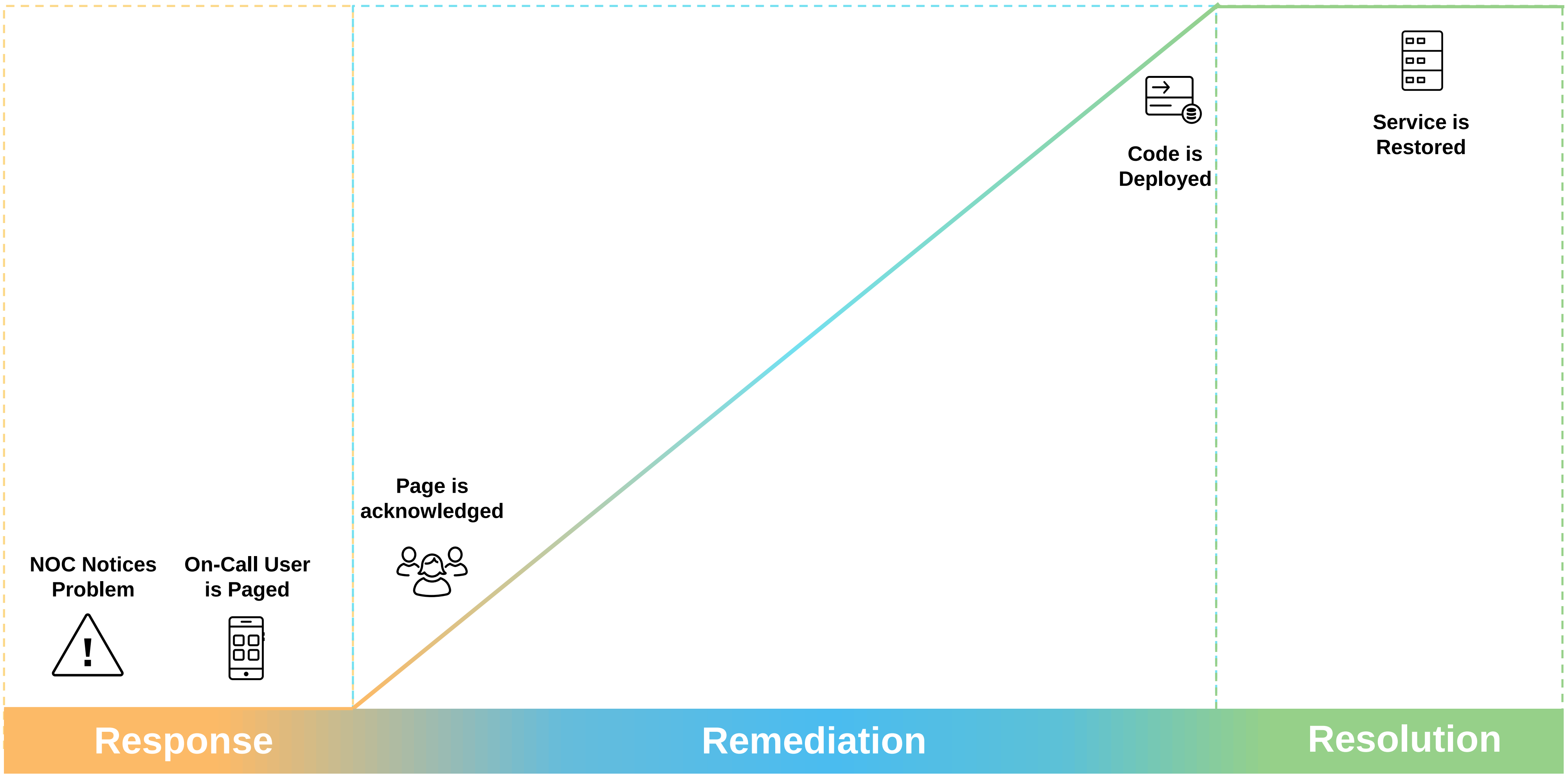
Incident Management is a “Team Sport”

- Perform “chaos events” that allow people to run through failure scenario and learn without cost

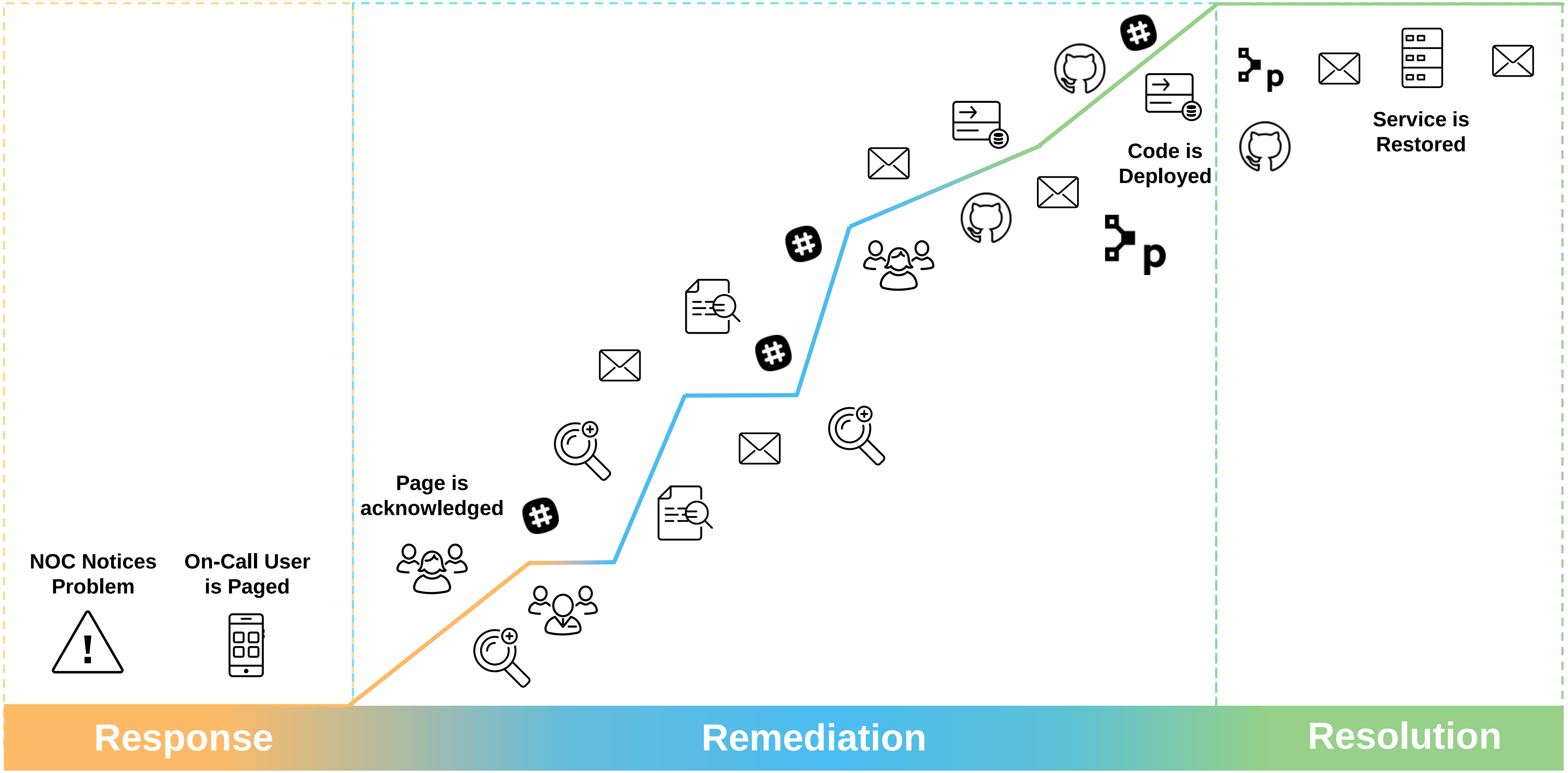
Why This Matters

The Time Invested Is Worth It

Analysis: Connecting Dots

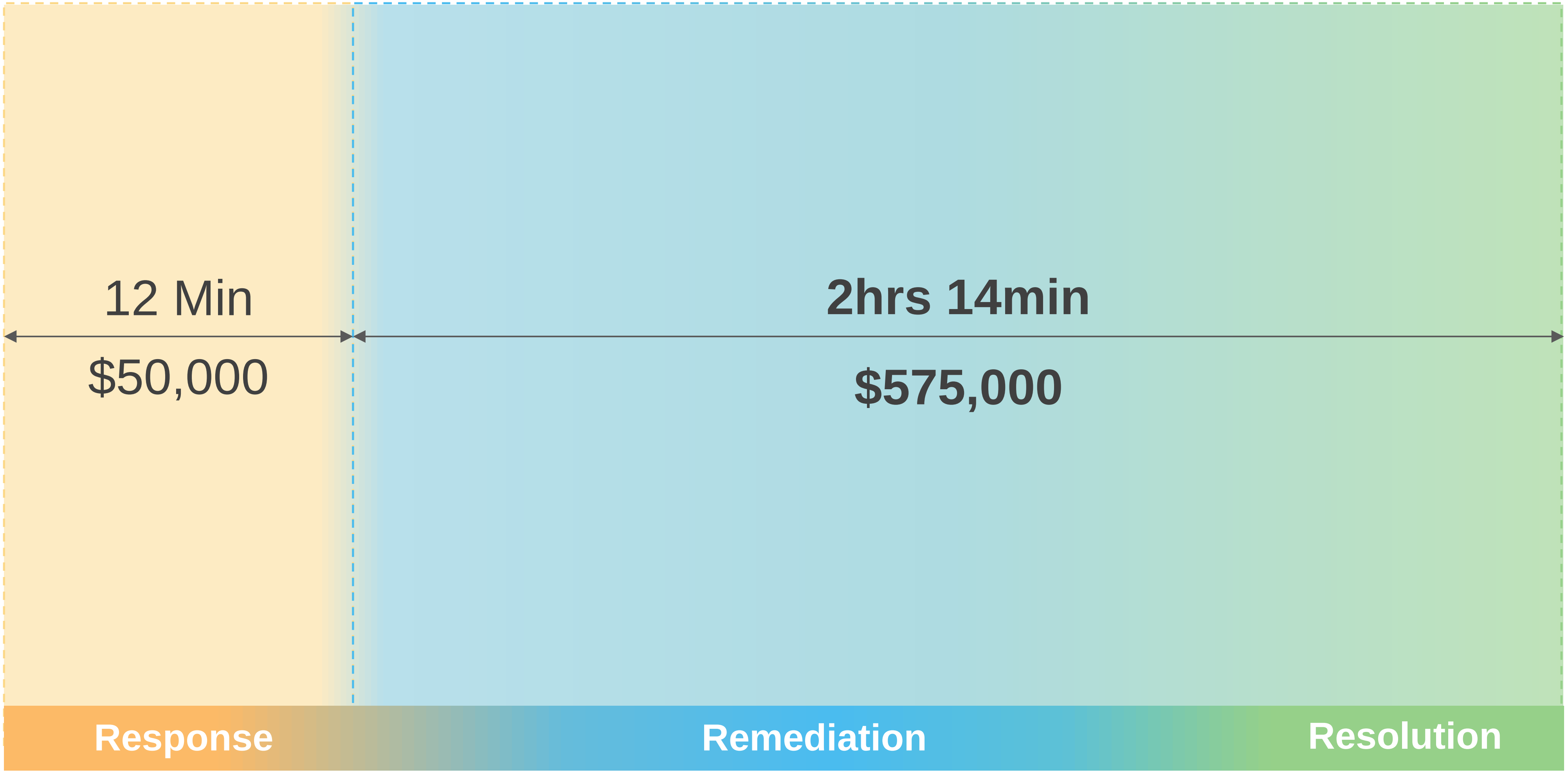


Analysis: Connecting Dots



Analysis: Connecting Dots

Cost of Downtime: $250,000/ hour



12 Min

$50,000

2hrs 14min

$575,000

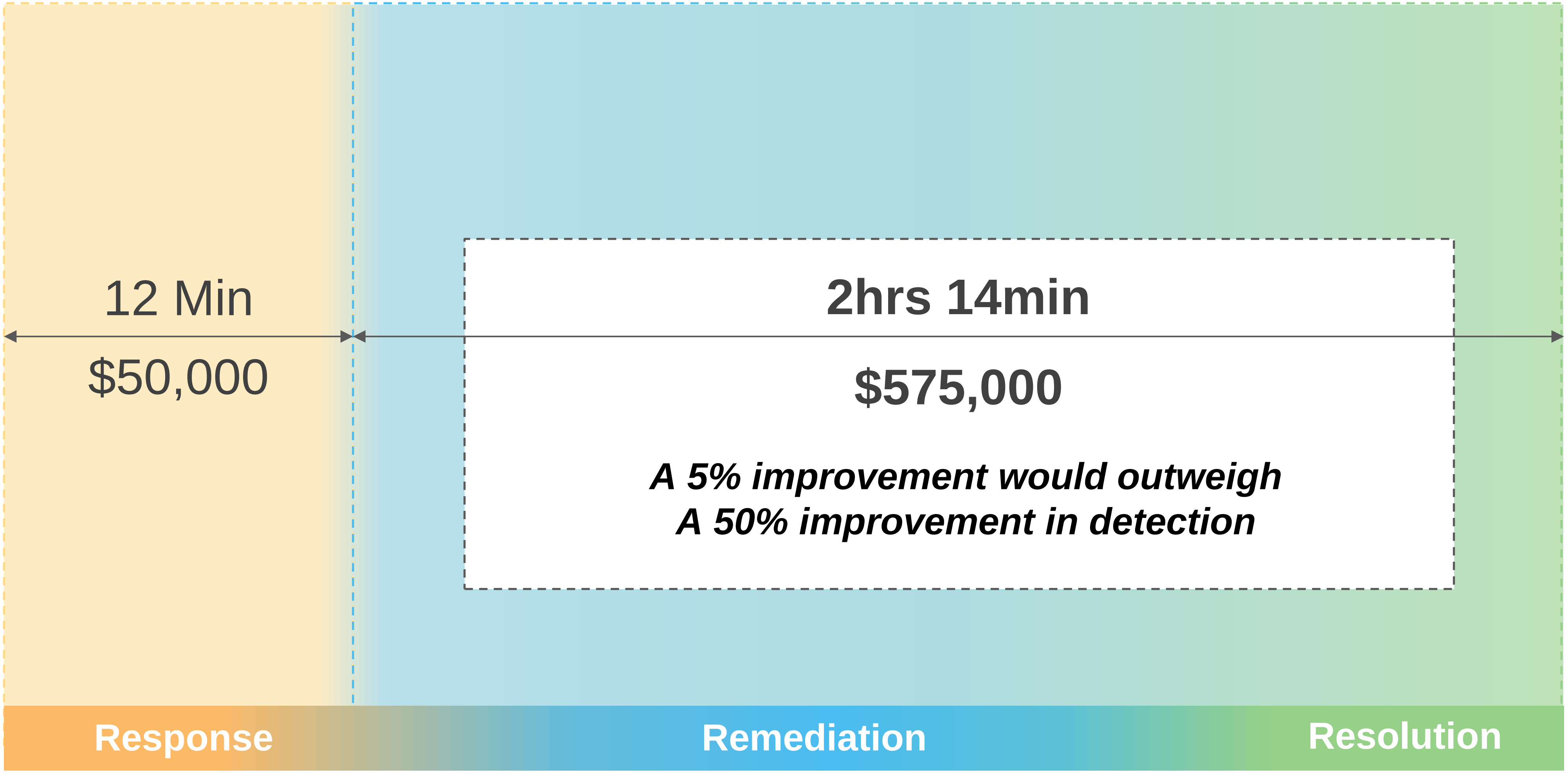
Response

Remediation

Resolution

Analysis: Connecting Dots

Cost of Downtime: $250,000/ hour



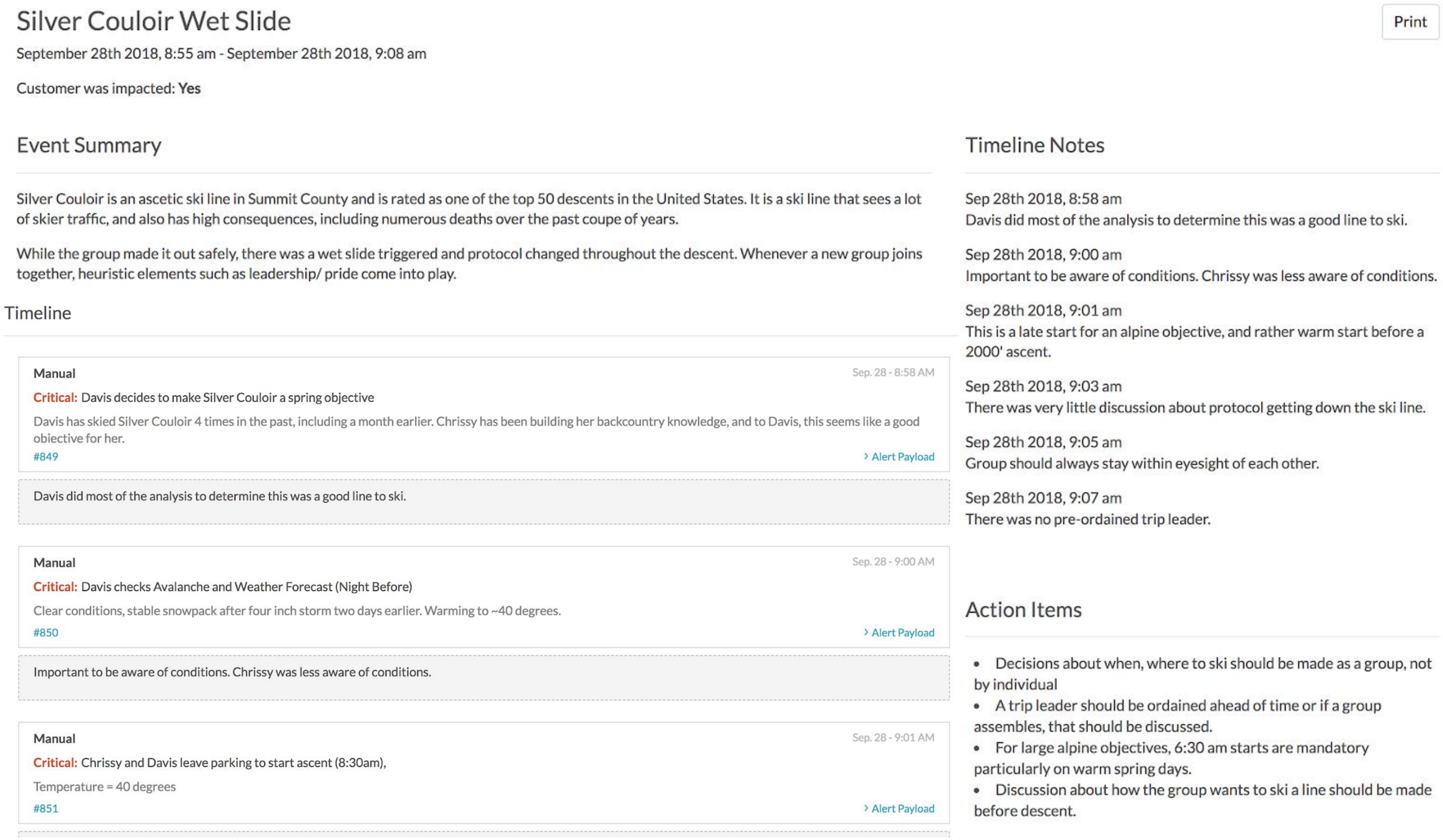
12 Min

$50,000

2hrs 14min

$575,000

A 5% improvement would outweigh
A 50% improvement in detection



Silver Couloir Wet Slide

September 28th 2018, 8:55 am - September 28th 2018, 9:08 am

Customer was impacted: **Yes**

Event Summary

Silver Couloir is an ascetic ski line in Summit County and is rated as one of the top 50 descents in the United States. It is a ski line that sees a lot of skier traffic, and also has high consequences, including numerous deaths over the past coupe of years.

While the group made it out safely, there was a wet slide triggered and protocol changed throughout the descent. Whenever a new group joins together, heuristic elements such as leadership/ pride come into play.

Timeline

Manual Sep. 28 - 8:58 AM

Critical: Davis decides to make Silver Couloir a spring objective

Davis has skied Silver Couloir 4 times in the past, including a month earlier. Chrissy has been building her backcountry knowledge, and to Davis, this seems like a good objective for her.

#849 › Alert Payload

Davis did most of the analysis to determine this was a good line to ski.

Manual Sep. 28 - 9:00 AM

Critical: Davis checks Avalanche and Weather Forecast (Night Before)

Clear conditions, stable snowpack after four inch storm two days earlier. Warming to ~40 degrees.

#850 › Alert Payload

Important to be aware of conditions. Chrissy was less aware of conditions.

Manual Sep. 28 - 9:01 AM

Critical: Chrissy and Davis leave parking to start ascent (8:30am),

Temperature = 40 degrees

#851 › Alert Payload

Timeline Notes

Sep 28th 2018, 8:58 am
Davis did most of the analysis to determine this was a good line to ski.

Sep 28th 2018, 9:00 am
Important to be aware of conditions. Chrissy was less aware of conditions.

Sep 28th 2018, 9:01 am
This is a late start for an alpine objective, and rather warm start before a 2000' ascent.

Sep 28th 2018, 9:03 am
There was very little discussion about protocol getting down the ski line.

Sep 28th 2018, 9:05 am
Group should always stay within eyesight of each other.

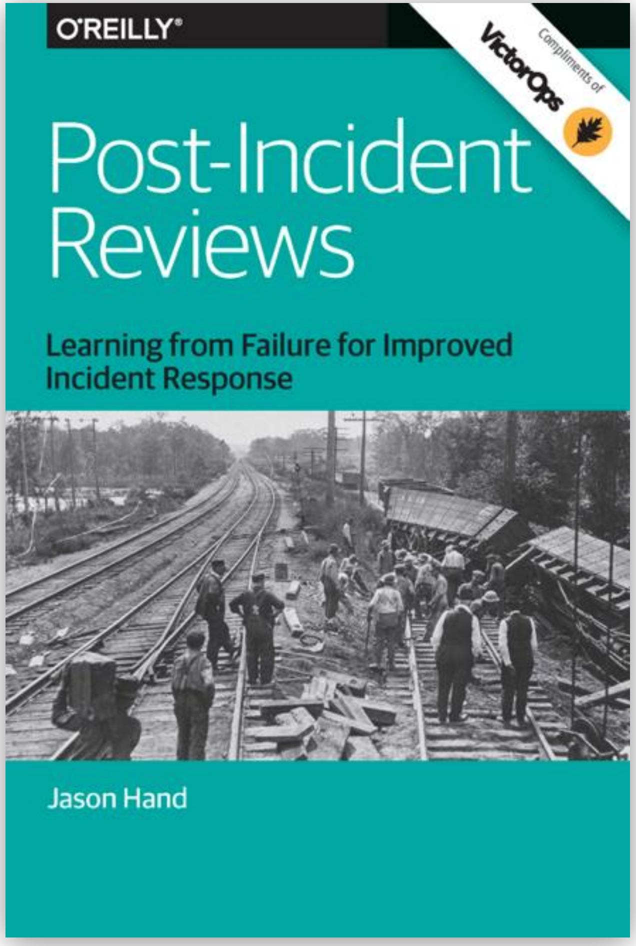
Sep 28th 2018, 9:07 am
There was no pre-ordained trip leader.

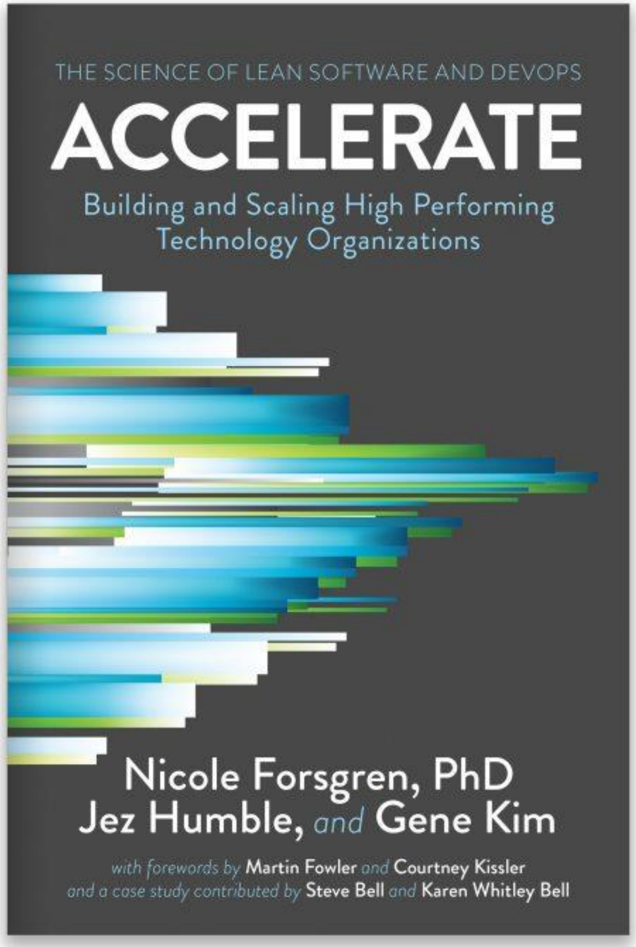
Action Items

- Decisions about when, where to ski should be made as a group, not by individual
- A trip leader should be ordained ahead of time or if a group assembles, that should be discussed.
- For large alpine objectives, 6:30 am starts are mandatory particularly on warm spring days.
- Discussion about how the group wants to ski a line should be made before descent.

It's in Your Hands

Recommended Reading:




THE SCIENCE OF LEAN SOFTWARE AND DEVOPS
ACCELERATE
Building and Scaling High Performing Technology Organizations
Nicole Forsgren, PhD
Jez Humble, and Gene Kim
with forewords by Martin Fowler and Courtney Kissler
and a case study contributed by Steve Bell and Karen Whitley Bell

Abstract

Okay, you’ve decided to go the “DevOps” route – you've created a culture of observability and aggregated your teams and monitoring tools into a single platform like Splunk + VictorOps... what next? An important part of this journey is understanding how to leverage this powerful platform to empower teams to create measurable processes and conduct blameless post incident reviews. In this session, we’ll discuss the barriers preventing effective incident reviews. From there, we’ll delve into how you can build incident timelines that pinpoint warning indications across all of your monitoring tools and document every individual response. Finally, we’ll share some thoughts on the skills, ethos and processes you’ll want to cultivate in your teams as you go beyond blameful processes like root cause analysis, and move towards a culture of continuous improvement.