Maps, Context, and Tribal Knowledge:
On the Structure and Use of Post-Incident Analysis Artifacts in Software Development and Operations

J. Paul Reed | LUND UNIVERSITY
Maps, Context, and Tribal Knowledge:
On the Structure and Use of Post-Incident
Analysis Artifacts in Software Development
and Operations

J. Paul Reed

Lund 2018

J. Paul Reed

Number of pages: 143
Illustrations: 11
Version: 1.1 (d3719a4)

Keywords
Incident analysis, incident artifacts, incident retrospectives, Internet operations, organizational learning, post-incident reviews, postmortems, tribal knowledge, software development, software operations.

©Copyright: Division of Risk Management and Societal Safety, Faculty of Engineering
Lund University, Lund 2018
Avdelningen för Riskhantering och samhällssäkerhet, Lunds tekniska högskola, Lunds universitet, Lund 2018.

Thesis work submitted in partial fulfillment of the requirements for the MSc in Human Factors and System Safety.

Advisor: Dr. Anthony J. Smoker
Assessor: Dr. James M. Nyce
Submission date: December 19, 2018
Abstract

The role, functions, and mechanisms of post-incident and post-accident learning within organizations has been widely studied in various industries. With the increasing role software plays in society, large-scale software development and operations practices and organizations are having a larger impact on day-to-day life; despite this, organizational learning has not been widely examined within this context.

This research involves a two-pronged methodology to investigate organizational learning within, specifically, the software development and operations industry: first, an industry survey collected broad industry data regarding post-incident analysis language, usage, and template artifacts; second, this survey served as a basis for a deep organizational case-study involving three distinct teams at a well-known Internet software development and operations company.

The survey identifies some post-incident analysis behavioral and artifact-usage trends, including three specific archetypes of post-incident analysis templates: the ‘Record-Keeper,’ the ‘Facilitator,’ and the ‘Sign-Post.’

The organizational case study examines the specific mechanisms of post-incident organizational learning, including use of template artifacts to map the company’s complex, emergent socio-technical system, share intra- and inter-team context in lieu of static lists of incident remediation items, and use post-incident artifacts to curate organizational culture and transmit tribal knowledge between organizational actors. Other uses are also identified.

Together, these uses provide a set of existence proofs for various methods and mechanisms for how a high-performing software development and operations organization uses, at least in part, its post-incident artifacts to aid in and sustain organizational learning.
It is always important to acknowledge those who inspire us to learn, and continue learning; were it not for some influential professors in my undergraduate career, I would not have ever considered pursuing a graduate degree. Thank you to Drs. Tim Kearns, Phillip Nico, Clark Savage Turner, and Clint Staley, who all put the bug in my ear over a decade ago at Cal Poly that learning more in service of mastering a subject would be not only intellectually interesting, but personally worthwhile.

Years later at Lund University, the experience could not have been more fulfilling or true to that advice: Drs. Johan Bergström and Anthony Smoker where instrumental in creating a safe environment to nurture learning about organizational learning and safety.

There are voices, influences, and mentors who join us at various points, for varying lengths of time, on our professional journeys: my thanks to John Allspaw, Kevin Behr, Jabe Bloom, Arup Chakrabarti, Pete Cheslock, Tasha Drew, Kevina Finn-Braun, Dr. Nicole Forsgren, Jeff Gallimore, Harold ‘Waldo’ Grunenwald, Jason Hand, Tara Hernandez, Mark Imbriaco, Nora Jones, Courtney Kissler, J. Mike McGarr, Scott Nasello, Courtney Nash, Melanie Parish, Scott Prugh, and Matt Stratton for their help, course corrections, support, and insights. There are also influences over the years which may prove harder to trace, but left no less an indelible mark: to those who provided help, big and small along the way: I could not have gotten here from there without you. Thank you.

There are friends who are friends. And then there are friends who are family: my appreciation, gratitude, and love to James Castañeda (and his basset hound, Clipper), Tony Marchesano, Gabe Ortíz, Greg Poirier, and Paul Scott. Much love to Raymond Cavalcante and K. Arthur Endsley, who both also cheered me on; however, they have the additional distinction of helping me retrieve all that latent statistics knowledge lurking in my brain from my undergraduate years; the statistical reporting and analysis presented herein are light years better for their patient assistance, for which I am hugely grateful. (And readers should be as well!)
Finally, I would like to acknowledge the often difficult journey every person and organization takes to get to a place of self-introspection where we have learned enough to begin to understand how we best learn from our missteps. That road is a rocky one, to be sure; my hope is this work contributes to filling in some of the potholes along the way.

To fellow individuals and organizations engaged in that pursuit: my encouragement, empathy, and support. As the old saying goes: it really is about the journey, not the destination...
Contents

1 Introduction
   1.1 A New Lens for an Established Science ........................................ 1
   1.2 A Notable Paradox for “Knowledge Work” ....................................... 3

2 Literature Review
   2.1 Positioning the Research .............................................................. 6
      2.1.1 Learning What We Mean by “Learning” ..................................... 6
   2.2 Organizational Learning ................................................................. 9
   2.3 Operational Retrospectives ............................................................. 12
      2.3.1 Contexts for Post-Incident Analysis Processes .......................... 12
      2.3.2 Retrospectives in Practice ....................................................... 14
   2.4 Information Technology / Agile Retrospectives ................................ 15
   2.5 Intersections ................................................................................... 17
      2.5.1 IT Security ................................................................................ 17
      2.5.2 Difficulties With and “Resistance to” Learning ......................... 17
      2.5.3 IT-Focused Knowledge Management ....................................... 19
   2.6 Literature Review Themes .............................................................. 19
   2.7 Research Question .......................................................................... 21

3 Methodology
   3.1 Industry Survey ................................................................................ 22
   3.2 Organizational Study ....................................................................... 23
      3.2.1 Organizational Selection ......................................................... 23
      3.2.2 Research Engagement ............................................................. 24
   3.3 Analysis Methods ........................................................................... 25
      3.3.1 Survey ..................................................................................... 25
      3.3.2 Organization ........................................................................... 25
   3.4 Methodological Summary ............................................................... 26
4 Results

4.1 Phase I - Pilot Survey Results ........................................ 29
4.1.1 Profile Survey Data ................................................. 30
4.1.2 Post-Incident Analysis Event Survey Data ..................... 30
4.1.3 Analysis Event Artifact Survey Data .......................... 31
4.1.4 Post-Incident Analysis Event Artifact Usage Survey Data 32
4.2 Post-Incident Analysis Event Template Survey Data ........ 33
4.2.1 Template Results - Structural ................................. 34
4.2.2 Template Results - Thematic .................................. 35
4.3 Organizational Introduction ......................................... 36
4.3.1 Industry Reputation ............................................... 37
4.3.2 Market Reputation ............................................... 37
4.3.3 Informant Panel .................................................. 38
4.4 Phase II - Organizational Study Results ......................... 39
4.4.1 Most Common Artifact Uses .................................. 39
4.4.2 Different Groups Use Artifacts Differently ................. 42
4.4.3 Artifact Usage During an Incident? .......................... 45

5 Analysis

5.1 Survey Statistical Analysis ......................................... 48
5.2 The Role of Artifact Templates .................................... 54
5.2.1 The ‘Record-Keeper’ .............................................. 54
5.2.2 The ‘Facilitator’ .................................................. 56
5.2.3 The ‘Sign-Post’ .................................................. 57
5.3 Artifact Usage in High-Performing Organizations .......... 58
5.3.1 Mapping the Emergent Complex Socio-technical Terrain 59
5.3.2 Context-Sharing Over Remediation Action Items .......... 63
5.3.3 Curating Organizational Culture and Transmitting Tribal Knowledge ............................................... 68
5.4 Other Incident Artifact Uses ....................................... 76
5.4.1 Liaising With External Partners ............................... 76
5.4.2 Secondary and Tertiary Artifacts .............................. 77
5.5 Other Findings .......................................................... 79
5.5.1 ‘Blameless’ is Table Stakes .................................... 79
5.5.2 Living Processes and Artifacts ................................. 82

6 Discussion

6.1 Too Good to Be True? ................................................. 85
6.2 On Peering Inside a Nuanced, ‘Messy’ System .............. 87
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.3 Bringing Software Into the ‘Safety’ Fold</td>
<td>90</td>
</tr>
<tr>
<td>6.4 A Final Word on the Survey Statistical Analysis</td>
<td>92</td>
</tr>
<tr>
<td>6.5 Future Research Avenues</td>
<td>93</td>
</tr>
<tr>
<td>7 Conclusions</td>
<td>95</td>
</tr>
<tr>
<td>Appendices</td>
<td>101</td>
</tr>
<tr>
<td>A Survey Questions and Results</td>
<td>102</td>
</tr>
<tr>
<td>B Survey Statistical Analysis</td>
<td>106</td>
</tr>
<tr>
<td>B.1 Hypothesis Correlation Tests</td>
<td>106</td>
</tr>
<tr>
<td>B.2 Statistically Significant Contingency Tables</td>
<td>113</td>
</tr>
<tr>
<td>B.2.1 Job Role Hypothesis Contingency Tables</td>
<td>114</td>
</tr>
<tr>
<td>B.2.2 Organization Size Hypothesis Contingency Tables</td>
<td>115</td>
</tr>
<tr>
<td>C Survey Statistical Analysis Source Code</td>
<td>118</td>
</tr>
<tr>
<td>C.1 Correlation Tests</td>
<td>118</td>
</tr>
<tr>
<td>C.2 Contingency Tables</td>
<td>122</td>
</tr>
<tr>
<td>D Survey Artifacts Open Coding</td>
<td>125</td>
</tr>
<tr>
<td>E Structured Interview Guide</td>
<td>127</td>
</tr>
<tr>
<td>F Structured Interview Open Coding</td>
<td>130</td>
</tr>
<tr>
<td>G Inter-rater Reliability</td>
<td>139</td>
</tr>
<tr>
<td>G.1 Survey Artifact Coding Inter-rater Reliability</td>
<td>139</td>
</tr>
<tr>
<td>G.2 Interview Open Coding Inter-rater Reliability</td>
<td>140</td>
</tr>
<tr>
<td>G.3 Organizational Case Study Inter-rater Reliability</td>
<td>141</td>
</tr>
</tbody>
</table>
List of Figures

2.1 Thesis Research Positioning in the Broader Literature .......................... 8
3.1 Visual Representation of Research Methodology .................................. 27
5.1 Post-Incident Artifact Template Usage Among Different Organization Sizes .................................................. 50
5.2 Usage of Specific Types of Post-Incident Artifact Template Usage Among Different Organization Sizes .................. 51
5.3 Post-Incident Artifact Availability Among Different Organization Sizes .................................................. 53
5.4 Statistically Significant Post-Incident Artifact Distribution Types Among Different Organization Sizes .......... 54
5.5 ‘Record-Keeper’ Postmortem Template Mockup ..................................... 55
5.6 ‘Facilitator’ Postmortem Template Mockup ......................................... 56
5.7 ‘Sign-post’ Postmortem Template Mockup ........................................... 57
E.1 Structured Interview Guide, Page 1 ..................................................... 128
E.2 Structured Interview Guide, Page 2 ..................................................... 129
# List of Tables

2.1 The ‘Five W’s’ and ‘One H’ of Organizational Learning .......................... 12

4.1 Panel of Organization Case Study Informants ................................. 39
4.2 Differences in reference to usage and thematic codes between func-
tional teams ......................................................... 43

B.1 Survey Statistical Hypotheses Test Results ................................. 107
B.2 Job Role Survey Subpopulation Counts ................................. 114
B.3 Hypothesis H.a.10 Contingency Table ................................. 114
B.4 Hypothesis H.a.32 Contingency Table ................................. 114
B.5 Organization Size Survey Subpopulation Counts ......................... 115
B.6 Hypothesis H.b.13 Contingency Table ................................. 115
B.7 Hypothesis H.b.19 Contingency Table ................................. 115
B.8 Hypothesis H.b.23 Contingency Table ................................. 115
B.9 Hypothesis H.b.26 Contingency Table ................................. 116
B.10 Hypothesis H.b.29 Contingency Table ................................. 116
B.11 Hypothesis H.b.30 Contingency Table ................................. 116
B.12 Hypothesis H.b.31 Contingency Table ................................. 116
B.13 Hypothesis H.b.32 Contingency Table ................................. 117
B.14 Hypothesis H.b.37 Contingency Table ................................. 117

D.1 Survey Template Artifacts Open Coding ...................................... 125

F.1 Organizational Case Study Structured Interviews Open Coding .... 130

G.1 Survey Artifact Coding Inter-rater Reliability Panel Results ........ 140
G.2 Organizational Case Study Inter-rater Reliability Panel Results .... 141
Chapter 1

Introduction

1.1 A New Lens for an Established Science

Organizational learning has always been a topic at the very heart of the safety sciences. From Heinrich to Hollnagel, researchers have sought to understand how learning weaves its way through the genesis, response, analysis, and remediation of organizational incidents and accidents, as well as studying learning’s larger implications for the societal, organizational, and personal understandings and deconstructions of the concept of “safety.”

Learning at both team and individual levels has also long been of academic interest to safety scientists: much work has been undertaken to understand the mechanism and role learning plays for those ‘deep in the trenches,’ in an attempt to explain how learning at the organizational ‘sharp end’ contributes to broader “organizational learning” at the blunt end, sometimes in lessons and insight which are more than the sum of the learnings of their constituent individual actors.

One of the most visible forms of organizational learning is the post-incident analysis. In industries historically studied by safety scientists, this process takes different forms and goes by many names: medicine’s “morbidity and mortality gatherings” (punctuated by the “[wearing] the hair shirt” ritual [Bosk, 2003, p. 146]), “after action reviews” in the fire service (Crowe, Allen, Scott, Harms, & Yoerger, 2017), the all-encompassing accident investigations performed by public transportation safety agencies the world over, and the more generic “retrospective” and “postmortem.”

In addition to the post-incident analysis processes themselves, the output of these processes—the artifacts produced as a byproduct of these processes—has been the topic of much research, with an obvious goal: understanding the potential these post-incident analysis artifacts have to facilitate organizational learning,
more effectively aid that learning, and hopefully ultimately improve organiza-
tional outcomes as they relate to the genesis of incidents and accidents and the
organization’s response to them.

Interest in this longstanding wheelhouse of safety science research is not lim-
ited to these aforementioned industries: in the past six years, the idea of the
“blameless” “postmortem” has become part of the software development and large-
scale Internet operations zeitgeist. Technology companies seeking the holy grails of
continuous improvement and increased organizational learning have started look-
ing long and hard at these incident retrospective and remediation practices from
other industries. Those who have developed internal programs to incorporate this
feedback into their daily work have relayed alluring narratives on the software
development and operations conference circuit of the benefits. (Allspaw, 2011)

Organizational learning in a software development and operations context is
of industrial interest for the myriad reasons it is of interest to other sectors:

• Reduced occurrence of incidents and accidents (or the promise thereof),
in the context of the practice of ‘normal work,’ but also in the context of
repeat-incidents (which organizations universally lament as ‘avoidable’).

• Minimization of ‘blast radius’ when such events do occur.

• A quicker, crisper, more coherent response from software developers and
operators in the midst of such events.

• The cultivation and sustaining of so-called organizational “feedback loops”
and increased culture of “(safe!) experimentation” which many technology
companies considered ‘high performing’ by their industry peers tout as a
cornerstone of their organizational culture.¹ In fact, these aspects are two of
the “Three Ways” of DevOps (Kim, Behr, & Spafford, 2013, p. 87), a profes-
sional movement within the technology industry that has in the last five or
so years enjoyed widespread attention and much attempted industry adop-
tion, precisely because of the perception of the benefits feedback loops and a
culture that ‘embraces experimentation’ provides to a business’ bottom line.

That said, organizational learning is of particular industrial interest to the
software development and infrastructure operations sector for a couple of reasons
that are unique to the sector:

¹Facebook’s founding mantra of “move fast and break things” (Kushner, 2011), for example.
Technology—especially software but also, increasingly, hardware—mirrors the complexity and emergent behavior and interactions facing many other industries with operational or production focuses. But, unlike many of those industries, software is one of the few of humankind’s artifacts that is not only incredibly malleable, but is so on a timescale shared by few industries with which humans have built up a corpus of operational experience. Additionally, the pace of innovation of the hardware and software building blocks and the tools available to manipulate those business’ software products is unrivaled by most of the industries where these processes have been studied previously.

In spite of these unique characteristics (or, perhaps, because of them), we, as a society, increasingly find technology inching its way into every aspect of daily life. Some of these areas provide us with until-recently unimagined conveniences with very little trade-off in terms of traditional notions of ‘safety’ or risk. But other areas where society finds technology approaching—self-driving vehicles, smart-devices managing fire detection/suppression systems or physical access control, and Internet-connected medical devices delivering lifesaving treatments—the ability to harm due to inaction or unintended- or unexpected-interactions are all too real... and the prevalence of such examples continues to rise.

And thus, it is these areas of operator sense-making of undesirable events and their consumption and digestion of findings, revelations, and remediations mined for in the rubble of an accident’s aftermath that we find ourselves poised to observe and reflect upon once again.

1.2 A Notable Paradox for “Knowledge Work”

It is worth pausing momentarily to note a paradox staring back at us as we start our examination of how, exactly, practitioners in software development and operations use artifacts to learn from the incidents and outages they experience: because such work represents so-called “knowledge work,” it is often assumed (or even taken for granted!) that the organization learns from these incidents. After

---

2While “definitions of knowledge work abound,” for our purposes, the term describes work in which “the knowledge worker processes and manipulates information as an end in itself which means it is the informational content of the job that defines the task, the product, and ultimately the worker....” and “for most parts of the production of knowledge no possible measure of output can be conceived that would be logically separate from a measure of inputs. For most knowledge workers, then, the real substance of work is not the product, but the process.” (Pyöriä, 2005, p. 118) For a further nuanced deconstruction of the concept, including, specifically, an application to IT, see Pyöriä, 2005, pp. 122-124.
all, isn’t ‘knowledge’ part and parcel of the actual work itself? So isn’t it reasonable to think that as software developers and operations engineers perform their work, they are engaged not only in the use, but also the creation of knowledge, which they then will leverage in the future, especially during an outage or incident? After all, it’s all “knowledge work,” so ‘work’ must be functionally equivalent to use of ‘knowledge.’

While a reasonable (and potentially satisfying) assumption, there exists no evidence to suggest that knowledge work (or the performance thereof), unto itself, implies or involves learning, especially in the broader context of “organizational learning” which would inform or modify organizational behavior. Quite the contrary, if it is one thing the technology industry prides itself on, it is the idea of how unique and different it, its work, its context, and its challenges are from those of any other industry.3

On the contrary, there does exist evidence that software development and operations companies are subject to the same stresses present in other operational industries:

- **Software developers and operations engineers make tradeoffs**, just like ‘traditional’ industries; and more importantly, those “sacrifice[s] [are] always difficult.” And those “sacrifice decisions are readily criticized afterwards, ... especially when they are difficult.” (Woods, 2017, p. 11) This is similar to other industries, where operational tradeoffs are standard (and well-known) aspects of ‘work.’

- **Software development and operations are not uniquely or inherently coherent or ‘aligned’**: In looking at systems in other industries, we may focus on the fact that multiple (sub-)organizations and actors make their problems different from our own in software. For example, when looking at safety in aviation, the technology community may see the complexities around the operation of the aircraft itself (pilots) and the additional interactions of air traffic control (and controllers), the aircraft manufacturer (and related maintenance), and other actors and be thankful its environment does not involve so many different parties.

  But software development and operations may even experience this situation more acutely: not only are numerous tradeoffs made during the software’s implementation (Turner, 1999, pp. 43-48), those tradeoffs can be compounded (or even unmasked) by the ways in which that software is later

---

3Especially in relation to those industries we might associate with as being traditionally studied by the safety sciences...
operated, an endeavor often undertaken by other actors with other goals and incentives, and increasingly on (cloud-based) infrastructure run in far flung lands by entirely different companies (each with their own distinct values, goals, and incentives)! This combination can create the same type of emergent behavior and unexpected interactions we see in other industrial incidents and accidents, a phenomena David Woods, in the context of software, has dubbed “dark debt.”

(Woods, 2017, p. 25)

- **As an industry, we are not getting better at this;** that is, we do not possess some inherent quality or skill that makes us ‘automagically’ improve as we experience our own organizational incidents; and there is no evidence to suggest that we pay any attention to other software organization's incidents and outages in a complete enough way so as to be of use in reducing or eliminating our own organizational incidents and accidents, as we observe in, say, aviation accidents.

The STELLA report itself lists ten different incidents in introducing this “dark debt” concept, which were not only notable in their suggested causes, but in their impacts and costs. (Woods, 2017, p. 25) (It is worth noting that three of those incidents were related to software outages in commercial airline systems, and another three of those were in the financial or insurance industry.) And, of course, this is just from a single consortium’s report; a public collection of over 100 (and growing) incidents, along with their post-incident analysis reports, illustrates just how common operational incidents and outages not only are in software operations, but continue to be. (Luu et al., 2018)

Taken together, all of these aspects paint a picture that suggests software development and operations organizations suffer the same challenges (lamentably) as our more ‘traditional’ brethren. While this may represent a bit of a bruise to our professional egos, it also suggests that we, as software and operations technologists, would do well to slow down, look around, and pay more attention to the struggles other industries have wrestled with, what transformation other companies have embarked upon, and what stories they have to tell us... for our industry may not be so different, so incredibly-more-equipped, or so immune to the realities surrounding us as we may assume, or perhaps have convinced ourselves to believe.

---

4Building on the software industry's existing—and infamous—concept of “technical debt”; see Tom, Aurum, and Vidgen, 2013.
Chapter 2

Literature Review

2.1 Positioning the Research

Though the focus of this academic inquiry—organizations developing software, deploying it on infrastructure, and actively operating both—may represent a new contender in the arena of industries under study by the safety sciences, the foundational lens through which we will examine this cohort is familiar: organizational learning and, specifically, how tangible post-incident artifacts assist the process. (Or don’t.)

Thus, when reviewing the existing literature on the topic to understand both applicability in the academic field and previous research, it was useful to begin by grounding this investigation in (the answer to) an important question: What, exactly, do we mean by “learning?”

2.1.1 Learning What We Mean by “Learning”

Drupsteen and Guldenmund pointedly and directly posed precisely this question in 2014 in “What is Learning? A review of the Safety Literature to Define Learning from Incidents, Accidents, and Disasters.” (Drupsteen & Guldenmund, 2014) This paper proved useful not only in helping to frame a practical conception of “(organizational) learning,” but also reviews and categorizes a number of other papers that proved useful to map out the current terrain of research on learning, organizational and otherwise.

Drupsteen and Guldenmund reviewed and ultimately included 47 papers¹ in their study. They categorized the papers into three distinct categories of organizational learning:

---
¹More papers were initially reviewed; a number were excluded for various reasons; see Drupsteen and Guldenmund, 2014, p. 83.
• **Learning lessons**: “A first step in ‘Learning from Incidents’ is to detect an incident and to reflect on its meaning.” (Drupsteen & Guldenmund, 2014, p. 83) As such, Drupsteen and Guldenmund grouped papers that dissect ‘incidents’ and accidents as an ‘input’ for organizations to obtain learnings from.

• **Learning from incidents processes**: Guldenmund and Drupsteen clarify that analysis of incidents is not enough to obtain any supposed benefits of Learning From Incidents: “To successfully learn, it is important to go from identification of lessons learned, to the implementation of those lessons.” (Drupsteen & Guldenmund, 2014, p. 88) Two sub-categories of process are codified:
  
  – *Stepwise Learning from Incidents Processes*, which discusses different methods researchers have inductively described regarding how organizations manage and disseminate information regarding incidents and accidents to support organizational Learning From Incidents; and
  
  – *Sharing Lessons Learned*, which covers how actors within systems share knowledge to support Learning From Incidents.

• **Conditions for learning**: Drupsteen and Guldenmund categorize papers discussing necessary conditions for organizational Learning From Incidents, including organizational trust, incident impact, and people involved (or not, but who should be) into this bucket. (Drupsteen & Guldenmund, 2014, pp. 90-92)

Using their three-bucketed deconstruction of “organizational learning,” and combining it with my research interest, I created my own ‘map’ of the Learning From Incidents landscape, to describe the positioning of this research:
In this model, a distinction is made between organizational learning (encompassing discussions of the processes, limitations, and hurdles of organizational learning) and the practice of “operational retrospectives” carried out in various industries. Certainly these areas are related, but the distinction really concerns at what point in the post-incident aftermath the organization is in:

1. Analyzing the event, searching for and discussing remediation items, and the forms these post-incident analysis activities take (i.e. “operational retrospectives”),

2. The dissemination of any artifacts those post-incident processes produce,

---

2The term ‘remediation item’ is used throughout the thesis; it generally refers to work identified by teams to ‘remediate,’ fix, or otherwise address (what a group has determined to be) ‘causes’ of a particular incident or outage; specifically, it can also refer to scoped and defined work items that are documented within an organizational system-of-record, to be performed later.
how the organization consumes, digests, and reacts to them, and what happens after the ‘investigation’ is considered ‘concluded.’

2.2 Organizational Learning

The three papers reviewed and categorized solely in the topic area of Organizational Learning—as one might expect, several papers spanned the three codified categories; more on them shortly—generally discussed the mechanisms by which organizations learn (or don’t) from safety incidents (Lukic, Littlejohn, & Margaryan, 2010), crises as learning triggers (Deverell, 2009), and the current state of research in organizational learning from accidents. (Le Coze, 2013)

All three papers establish and describe frameworks related to “[Surfacing] factors that are important for effective [Learning From Incidents]” (Lukic, Littlejohn, & Margaryan, 2012, p. 950) and “[increasing] the effectiveness and impact of [Learning From Incidents] initiatives,”3 (Lukic et al., 2012, p. 950) and interestingly, two of these proposed frameworks are based on analyses and reviews of the pertinent Learning From Incidents literature, indicating a strong research interest in making sense of the current “fragmented literature” (Le Coze, 2013, p. 444) examining Learning From Incidents in various industrial contexts.

As a result of review of the literature in the context of his framework, Le Coze identifies “four selected lessons” related to Learning From Incidents categorized by “types of insights”:

1. From a political insight into reporting: reporting must be “understood as a practice embedded in a socio-legal-political context, requiring strategies to cope with conflict of interest between different actors...” (Le Coze, 2013, p. 447)

2. From a psychological/cognitive view of accident selection: “Beyond reporting, selecting signals about accidents relies on cognitive processes involving different strategies, including ‘making patterns,’ ‘drawing connections,’ ‘recognizing novelty’, and ‘sensing discrepancy.’” (Le Coze, 2013, p. 448) Le Coze goes on to say “One should not underestimate the level of expertise required for this task...,” (Le Coze, 2013, p. 448) (harking back to Klein and Hoffman’s work on “seeing the invisible” [Klein and Hoffman, 1992]).

3. Sociological insight into the investigation step: “Investigating accidents relies on the use of models which need the appropriate level of expertise in order

---

3These quotations were taken from Lukic, et al.’s 2012 paper, written two years later to reflect on the use of their framework defined in the above-described paper.
to be applied adequately,” that is “one should not underestimate the impact of unduly extending or translating high reliability organization studies, or any other social sciences models, into investigations [emphasis added].” (Le Coze, 2013, p. 449)

4. Finally, a political view on prevention: “Learning does not take place systematically after an event is investigated,” immaterial of the severity of the accident; “It is only when recommendations are implemented that results are obtained [emphasis added]....” (Le Coze, 2013, p. 451)

In the second framework, Lukic tries to address a critique that “very few papers addressed all the envisaged aspects when developing their Learning From Incidents approaches,” (Lukic et al., 2010, p. 428) and proceeds to dissect this aspect of “Learning From Incidents in the workplace” in four ways, via literature review:

- **Who is learning?** Lukic focuses on two concepts: inclusion and participation, i.e. whose input is sought, but also how engaged those included are in the post-incident analysis process (which can be constrained by other factors, such as a ‘blame and shame’ culture, etc.). (Lukic et al., 2010, p. 430)

- **What kind of learning process is adopted?** Lukic categorizes the mechanism of learning using Argyris and Schön’s work (Argyris & Schön, 1996) on single- and double-loop learning. (Lukic et al., 2010, p. 431)

- **What is the nature of the problems causing the incident?** Lukic uses Snowden’s Cynefin model (Snowden, 2002) to categorize the problems related to the incident as existing within an “obvious,” “complicated,” “complex,” “chaotic,” or “disordered” domain, as described by Cynefin. (Lukic et al., 2010, p. 432)

- **What type of knowledge is involved?** i.e. Conceptual (“knowing why” and “knowing what”), procedural (“knowing how”), dispositional knowledge (“attitudes, values, emotions, interests, and personal motivations”), or “locative knowledge,” that is meta-knowledge about the location and sources of one of the above types of relevant knowledge. (Lukic et al., 2010, p. 433)

Lukic proposes the above framework “could be useful in systematically analysing and identifying effective approaches to learning from incidents.” (Lukic et al., 2010, p. 441)

The final framework attempts to describe the learning processes from, specifically, crises. These are defined as “a situation that subjects a community of people, such as an organization, a state, or a municipality, to a serious threat to its basic
structures or fundamental values and norms, which, under time pressure and un-
certainties, necessitates making crucial decisions.” (Deverell, 2009, p. 180) As only incident-related postmortems activities and the resulting artifacts from analyses are to be considered, in some sense, the research is always looking at what could be described as an ‘organizational crisis’ of some degree.\(^4\)

In studying a case study of a repeated crisis involving a Swedish power utility, Deverell defines the following axes for his framework (Deverell, 2009, pp. 181-183):

- **Similar to Lukic’s framework, What is learned**: single- or double-loop learning?
- **What is the focus of learning**: incident response or accident prevention?
- **When are lessons learned**: inter- or intra-crisis; that is does learning occur “from one crisis and [prompt] making changes to prepare for another” (intercrisis’) or does it occur “to improve response during a single crisis episode” (intra-crisis’)? (Deverell, 2009, p. 182)
- **Lesson implementation**: are lessons distilled or implemented? That is, lessons that were “noticed, but not carried out to the extent that [they] alter organizational behavior” are considered to be “distilled,” but not implemented. (Deverell, 2009, p. 183)

Deverell further describes four propositions, two of which are relevant to this the research (Deverell, 2009, pp. 185-196):

- Organizations that are subjected to crisis experiences are likely to learn how to act at the event of a crisis, while they will still find it difficult to learn how to prevent future ones.
- If there is external critique toward the organization and credibility loss, then implementation of a crisis-induced lesson will be carried out at a greater rate.

In summary, if we pose the question very simply as ‘What are the five W’s and one H\(^5\) of Organizational Learning?’ these three papers competently cover the answer:

\(^4\)And, in some cases, the degree, severity, and seriousness of that crisis only becomes fully coherent in retrospect; see the Knight Capital case for an example. (Murphy, 2016)

\(^5\)Who, What, When, Why, Where, and How?
Table 2.1: The ‘Five W’s’ and ‘One H’ of Organizational Learning

<table>
<thead>
<tr>
<th>Question</th>
<th>Paper Addressed</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Who?</td>
<td>A Framework for Learning from Incidents</td>
</tr>
<tr>
<td>• How?</td>
<td>(Lukic, Littlejohn, &amp; Margaryan, 2012)</td>
</tr>
<tr>
<td>• What?</td>
<td></td>
</tr>
<tr>
<td>• When?</td>
<td>Crises as Learning Triggers</td>
</tr>
<tr>
<td>• Where?</td>
<td>(Deverell, 2009)</td>
</tr>
<tr>
<td>• Why?</td>
<td>What Have We Learned About Learning?</td>
</tr>
<tr>
<td>• (Why not?)</td>
<td>(Le Coze, 2013)</td>
</tr>
</tbody>
</table>

2.3 Operational Retrospectives

A fair amount of research has been published in what might be referred to as “operational retrospectives,” that is: the practice of incident retrospectives in a specific industrial context.

2.3.1 Contexts for Post-Incident Analysis Processes

Perhaps one of the most central issues with respect to the conduct of “operational retrospectives” is the lens through which organizations fundamentally view the cause(s) of accidents. Catino shines a spotlight on this very question, and in his review of the literature, describes two main categories of opinion on accident cause (Catino, 2008, p. 53):

- Individual blame logic (IBL), “an accusatory type of approach which tries to identify the guilty individuals”; and

- Organizational function logic (OFL), “which intends to identify the factors within the system which favored the occurrence of the event.”

Critically, Catino observes: “The two approaches are characterized by two distinct inquiry logics, which” (as we might expect), “generate different consequences.” (Catino, 2008, p. 53)

Catino describes individual blame logic as believing in:

- The “voluntariness of actions,” that is “people are... free agents capable of choosing between safe and unsafe behavior,”
• Individual responsibility,
• A “strengthened” sense of justice, precisely because “identification of the blame tends to satisfy the people involved and the public in general,”
• And convenience. (Catino, 2008, p. 55)

The last two are particularly notable, as Catino notes “[Individual Blame Logic] is in agreement with the Western legal system,” (Catino, 2008, p. 55) which may explain why it appeals to Western senses of ‘justice’ (generally retributive in nature, instead of restorative) and why it is considered “convenient.”

In contrast, Organizational Function Logic recognizes the impact of the “functioning of surrounding organizations and institutions” (Catino, 2008, p. 55) on incidents and accidents. As a result of this framing, human error is viewed “more as a consequence than as a cause” (Catino, 2008, p. 55), identification of latent factors in accident events is considered relevant, and ‘mistakes’ are “socially organized and systematically produced.” (Catino, 2008, p. 55) Organizational Function Logic, notably, highlights the limitation of ‘sequence-of-events’ and other linear accident models. (Catino, 2008, p. 56)

Catino argues Individual Blame Logic is limiting in a number of ways Organizational Functional Logic is not:

• “Once the ‘guilty actors’ found according to [Individual Blame Logic] are removed, it is very probable the organizational system will continue to function with the same organizational conditions and mechanism [emphasis added] which lead to the error and to the accident.” (Catino, 2008, p. 56)

• Because of this, in “adopting the [Individual Blame Logic], the organization is not able [emphasis added] to understand its own errors.” (Catino, 2008, p. 56)

• Individual Blame Logic is also beneficial to organizations and systems, Catino argues, precisely because “when a single person is identified as responsible..., the individual responsibility is split from the responsibility of the organization, with overall significant economic and financial advantages for the system.” (Catino, 2008, p. 57)

• Individual Blame Logic does not value investigation of so-called ‘near-misses,’ whereas Organizational Function Logic considers “both an accident and a near miss [to be] of equal interest, if they are morphologically similar.” (Catino, 2008, p. 57)
Catino succinctly summarizes the IBL/OFL debate thusly: “an [Organizational Functional Logic] looks to the future and improves the organization, whereas an [Individual Blame Logic] favors organizational change inertia and does not eliminate the condition of risk.” (Catino, 2008, p. 60)

2.3.2 Retrospectives in Practice

Sanne and Lundberg, et al., both provide insights into (industry-specific) mechanics of retrospective practices. Sanne describes a notable dichotomy between incident reporting systems and informal “storytelling” in Swedish rail: “the incident reporting scheme is not integrated into technicians’ practices and cultural frame and it does not seem to serve their interests. However, storytelling is of limited value from the perspective of organizational learning and occupational protection.” (Sanne, 2008, p. 1206) Also notably, Sanne found in his ethnographic research of supervisors and technicians that “technicians might have a different conceptualization of incidents.” (Sanne, 2008, p. 1206) Storytelling was found to be “an integral part of [technicians’] practices and their specific accident etiology and it provides a way for the technicians to address risks,” though like the discussion of Individual Blame Logic and Organizational Function Logic, this is from a “narrow perspective.” (Sanne, 2008, p. 1206)

Lundberg, et al. conducted structured interviews with 22 informants from six different industries; in terms of investigating accidents and suggesting remediation items, and then actually implementing them, they found evidence of various forms of bias, including author bias, confirmation bias, frequency bias, political bias, sponsor bias, and professional bias. (Lundberg, Rollenhagen, & Hollnagel, 2010)

Lundberg et al., summarizes the findings beautifully as “a warning against believing in the rationality—but not the sensibility—of investigation and remediation.” (Lundberg et al., 2010, p. 2138)

Finally, from an operational retrospectives framing, Lindberg, et al., pose the question: “Learning from accidents: What more do we need to know?” (Lindberg, Hansson, & Rollenhagen, 2010) They describe a model of experience feedback known as CHAIN (CHain of Accident INvestigation steps), which models Learning From Incidents as five steps: reporting, selection, investigation, dissemination, and prevention. (Lindberg et al., 2010, p. 715) The paper’s relevant finding here is its conclusion that “the discipline of experience feedback has not been sufficiently self-reflective.” (Lindberg et al., 2010, p. 716) It also argues nine topics of interest for future studies of experience feedback (Lindberg et al., 2010, p. 720); in
the context of this research, four are relevant:

- How accident databases actually used.
- How and to what extent conclusions from accident investigations are disseminated.
- The actual effects of accident investigation reports on preventative measures.
- The integration of experience feedback systems into overall systems of risk management.

Lindberg notes “We give the highest priority to the last [three] of these issues, since they are key issues for the development of evidence-based accident investigation processes.” (Lindberg et al., 2010, p. 720)

2.4 Information Technology / Agile Retrospectives

‘Retrospective’ and ‘postmortem’ generally have different contextual meanings in software development and information technology (IT) operations: rather than referring to review of software operational incidents and accidents, these terms refer to the review of large-scale, multi-year software projects (i.e. development of a major release of an operating system [Collier, DeMarco, and Fearey, 1996]) or massive information technology rollouts (i.e. replacing all computer systems with updated hardware and software in an organization such as a state-run department of motor vehicles or a hospital [Megerian, 2013]).

Discussions of these types of “postmortem reviews” goes back to the mid-1990s and that context has been reinforced since the Agile movement’s adoption of the term in the context of “sprint retrospectives,” i.e. reviews of shorter, 2-4 week software development cycles. (Starr, 2012)

Torgeir Dingsøyr has updated the findings of Collier, et al. in his 2005 paper on the subject. (Dingsøyr, 2005) His findings further codify what Collier, et al. describe, but with more generic and industry-applicable results. Still, this work focuses on “practical methods to harvest experience from projects that are either completed or have finished a major activity or phase,” not on “postmortem reviews” which look at operational incidents or accidents involving software or IT operations.

---

6The text reads “...to the last four of these issues”, however, only three of those four listed in the paper are relevant to this research.
Dingsøyr lays the foundation for learning by referring (again) to Argyris and Schön's single- versus double-loop learning model (Dingsøyr, 2005, p. 294), and also the impact of participation or communities of practice within learning. (Dingsøyr, 2005, p. 294) This is a useful discussion, as many of the industrial movements in the past five years (namely, “DevOps”) are linked by communities of practice and events emphasizing the fostering of such communities.

Dingsøyr also describes other industrial advice, but again, all in the context of postmortem software projects, not IT operations. The timescales for such post-mortems are also more elongated than industrial practice suggests is the case for operational retrospectives (where the timescale is marked in days-to-weeks, not the months-to-years seen in projects).

Finally, Dingsøyr reviews many suggestions technologists would recognize as common sense advice discussed within the industry today to incorporate into a software operations post-incident analysis process:

• An emphasis on an understanding and sincere belief “that everyone did the best job he or she could, given what was known at the time, his or her skills and abilities, the resources available, and the situation at hand” (Dingsøyr, 2005, p. 300),

• Inclusion of a “broad audience for a postmortem” (Dingsøyr, 2005, p. 301),

• The requirement for participant preparation before holding the postmortem,

• Identifying and having a facilitator for the postmortem, and

• Potential output from the postmortem; notably, Dingsøyr offers two examples; which a team should pick, he says, depends on the intent of the postmortem process itself.

Regarding intent, Collier and DeMarco, too, exhort the importance of presenting the results of the postmortem at “each of upper management’s regular scheduled organizational reviews.” (Collier et al., 1996, p. 71) They remark “The postmortem must have a well-understood link to the conduct of future projects. This link must be in the form of a commitment by management [emphasis added].” (Collier et al., 1996, p. 67)

It should be reiterated that all of these suggestions are in the context of software project postmortems. Though, it is notably interesting that many of the same suggestions we see in an industrial context for IT operations postmortems can be traced back to these papers published in the mid-1990s into the early 2000s.
2.5 Intersections

Some of the papers reviewed in the Drupsteen/Guldenmund paper did not fit entirely within one of the three defined buckets above. In cases of thematic overlap, some papers were analyzed from an intersectional point of view, with the following overlaps emerging.

2.5.1 IT Security

As discussed previously, the vast majority of ‘retrospective’ or ‘postmortem’ results with regards to software were in the context of software development projects, not operations. There existed one notable exception: IT security and incident response as related to security incidents and breaches.

A series of three articles by Peter Stephenson were found early in the literature review process describing how to model postmortems (Stephenson, 2003b), completing a postmortem investigation (Stephenson, 2003a), and looking ahead to formal modeling of events (Stephenson, 2004), but all of these were in the context of IT security incidents and breaches.

Thus, the only research overlap found between IT / Agile Retrospectives and Operational Retrospectives is what I would broadly categorize as “IT Security” practices and research on those practices.

2.5.2 Difficulties With and “Resistance to” Learning

Various papers tackle the topic of why, despite espousing that learning is not only important, but leads to better outcomes, there still exists a paradox where organizations do not (or cannot seem to) learn.

Drupsteen and Groeneweg appear in this corner of the literature as well, presenting a framework aimed at modeling Learning From Incidents. (Drupsteen, Groeneweg, & Zwetsloot, 2003) Despite this being yet another framework to contend with, their four-stage model,7 totaling eleven steps spread across the stages (Drupsteen et al., 2003, p. 65) exists in order to try to map out bottlenecks in the learning process, which they define is either “a step that is not performed, not performed well, or the relevant information [for the step] is not used.” (Drupsteen et al., 2003, pp. 64-65)

They then surveyed 303 practitioners across various industries to ask where they experienced bottlenecks in the eleven identified steps. Perhaps unsurpris-

---

7Investigating and analysing incidents, planning interventions, intervening, and evaluating.
ingly the top three bottlenecked steps were at the beginning of the process (“reporting”) and at the very end of the process (“intervention” and “evaluation.”) They also found, generally, that “Follow-up steps (from stage two on) were more often neglected than the earliest steps....” (Drupsteen et al., 2003, p. 70) They concluded that “learning potential was especially lost at the reporting and the evaluating steps, and the latter was a critical step for the learning to learn process,” a finding observed “in all industries.” (Drupsteen et al., 2003, p. 75)

Huber, et al., conducted a similar survey, though more focused through the lens of resilience engineering. They ask “whether a company is aware of positive ways in which people, at all levels... contribute to the management and containment of the risks it actually faces [and] whether the organization has the adaptive capacity necessary to respond to the changing nature of risk as operations shift and evolve.” (Huber, van Wijgerden, de Witt, & Dekker, 2009, p. 90) Their findings mostly confirmed existing work looking at the practices around resilience engineering (the necessity for adaptive capacity, the reaffirmation of the delta between work as imagined and work as performed, etc.), but of note here, they concluded “All incidents must be reported and analyzed, but it can be very difficult for managers and operators alike to agree on what counts as an ‘incident.'” Also: “analyzing an incident is not the same as learning from it; for this, a whole suite of follow-up activities is necessary [emphasis added].” (Huber et al., 2009, p. 94)

Birkland looks at the tail-end of this learning process and, specifically, what he terms “Fantasy Documents” that get produced as a result of investigation and remediation activities which are themselves a “fantasy.” (Birkland, in fact, calls this “fantasy learning.” [Birkland, 2009, p. 150]) The scope and effects of incidents Birkland examines are larger than what we might refer to as ‘day-to-day’ operations: Birkland cites John Kingdon's term of “focusing event” to describe the types of accidents referred to in this paper, i.e. Challenger, Chernobyl, etc. (Birkland, 2009, p. 150) Because of this, the paper examines the learning process from a larger, societal perspective, including policy discussion and generation, and the political machinations involved.

That said, many of the forces he describes also affect IT in an operational incident context (especially where millions of dollars are lost [Megerian, 2013] or the event broaches an ‘existential cliff’ for the organization, e.g. Knight Capital [Murphy, 2016]). Birkland identifies five “broad patterns” of lessons learned processes and learning documents in this context, though “The first four of [my] examples falls into a class I call ‘fantasy learning’ that generates ‘fantasy lessons learned documents....’ Only... the fifth [scenario] is an example of sound instrumental learning,” that scenario being: “An event happens, and a thorough and careful investigation
is initiated, which leads to policy change as a result of careful investigation, assessment, and policy design.” (Birkland, 2009, pp. 149-150)

Finally, Dien, et al., examine industrial Operating Feedback Systems (OFS) and posit that the issue may, in fact, be that we analyze incidents and accidents as opposed to organizations. (Dien, Dechy, & Guillaume, 2012) They focus on the analysts’ role in this process, and various ways analysis may be hindered and/or otherwise affected by the analyst’s role themselves in the process. They conclude a major problem can be that “analysts either [are] not... able to address the whole scope or do not have self interest to extend scope of investigation. They could also face difficulties to have access to relevant organizational data and to make sense of it [emphasis added],” (Dien et al., 2012, p. 1406) This is of particular interest in an IT context for organizations that have separate problem management teams (a common organizational structure, especially in larger enterprises), and may staff those teams with personnel with less technical knowledge and background (or even none).

2.5.3 IT-Focused Knowledge Management

Two papers reviewed focused on IT-related retrospective processes, but not in an operational context.

Dingsøyr, et al. argue “[their] experience with postmortem analysis proves that it is an excellent method for knowledge management, which captures experience and improvement suggestions from completed projects and works even in small- and medium-size companies that cannot afford extensive Knowledge Management investments.” (Birk, Dingsøyr, & Stålhane, 2002, p. 43) While this paper describes much of the same methodology and structure for postmortem analyses previously described (Dingsøyr, 2005), the noteworthy aspect is the focused reference to the ‘lightweight’ Knowledge Management benefits that can be experienced by implementing such a process, even in a software development project perspective (as opposed to a software operations one).

2.6 Literature Review Themes

During review of the above literature, the five following themes emerged as somewhat consistent across these papers’ discussion and analysis of organizational learning, Learning From Incidents, and IT retrospectives:

1. Many of the papers introduced entire frameworks or analyzed their results by using an established framework. These frameworks all try to decon-
struct ‘the (organizational) learning process’ or the ‘incident life cycle,’ but of course, they all do it slightly differently. To wit: Dien lists seven different root cause analysis frameworks.⁸ (Dien et al., 2012, pp. 1399-1400)

2. Many of these frameworks make repeated reference to, and emphasize the importance of, detecting and distinguishing between “single-” and “double-loop learning,” especially in the context of organizational learning and post-incident analysis.

3. Despite the focus on these (theoretical) frameworks, there doesn’t seem to be much research on what organizations actually did with the artifacts from a post-incident analysis process. Most of the discussion on this topic is focused on the organizational behavior(s) (or failure[s]) to the extent that they relate to and fit within the framework being presented.

4. None of the reviewed literature studied any of these post-incident analysis processes or (lack of) organizational learning (and the associated behaviors and/or hurdles) in the context of IT or IT operations (with the narrow exception of IT security).

5. Finally, a recurring theme in the research is a prevailing pessimism regarding organizations’ abilities to learn, at least in the short term, in any ways that would prove meaningful to them within the context of preventing and/or more effectively addressing large-scale incidents and accidents on a long time scale.

---

⁸None of which, I might add, were introduced in any papers in this literature review, which is to say the seven Dien, et al. describe are in addition to the frameworks cataloged in this literature review.
2.7 Research Question

Keeping in mind the aforementioned issues and context, and the somewhat fascinating revelation that research on Learning From Incidents lacks vivid, deep description of how organizations use what they produce during a postmortem process (to say nothing of looking at that question, specifically, in the IT operations industry), the research question we find ourselves preoccupied with answering is:

*In industrial practice, in what ways and to what purposes are the artifacts produced in a post-incident review process utilized by a software development and operations company?*
Chapter 3

Methodology

The thesis research was conducted in two distinct phases: an exploratory industry survey, followed by investigation of the artifacts and practices of a specific organization.

3.1 Industry Survey

Research was initiated with a pilot survey with an intended purpose of seeking details regarding post-incident analysis activities within the software development and operations industry, as well as collecting examples of industry artifacts produced by those activities. The survey was open to the public and did not require registration or other uniquely-identifying information. The survey was advertised via various industry-related social media outlets.

The survey’s purpose was to:

1. Gain some initial, high-level perspective into the language used, roles performed, practices, and outputs of software development and operations organizations’ post-incident review-events, in a real-world industrial context.

2. Collect examples of post-incident review artifacts created and used by software development and operations organizations; specifically, blank templates for teams’ “post-incident review-event artifacts” were solicited.

3. Facilitate the creation of a base-corpus of data to perform preliminary open coding on the structural elements of and themes found within industry post-incident review-event templates.

4. Better understand the various roles engaged in the creation and use of these post-incident review-event artifacts, to use in building a list of potential
informant roles (including ‘must-have’ participants) to further engage with
during the organizational study phase.

In addition to template artifacts, the survey collected generalized data on how,
when, and how often post-incident review-events are held, what language teams
use to refer to that event, which artifact(s) result from that event, what elements
are contained within the artifact(s), who is responsible for creating the artifact(s),
where the artifact(s) are stored, and whether the artifact(s) are disseminated through-
out the organization (and, if so, via what mechanisms).

Additionally, the survey results provided insight into where within the organi-
zational study phase to search for examples of artifact storage and use. The survey
results also further informed the questions for the structured interviews,¹ as well
as provided a primary analytical lens through which to view the artifacts provided
by that organization, especially in terms of being able to code those organization-
specific artifacts with a schema based on wider sample of industry artifacts, and
later compare and contrast the organization-specific artifacts with the examples
provided from the survey.

3.2 Organizational Study

The second phase of the research focused on an engagement with an industrial
organization to better understand real-world behavior with respect to the post-
incident review artifacts, as well as the storage, retrieval, and use of those artifacts
under real-world circumstances and situations.

3.2.1 Organizational Selection

Two types of organizational engagement were considered for Phase II:

- Informants and artifacts from two different organizations would be studied,
  with specific traits identified and differentiated between the two. Traits
  considered included company size, technology sub-industry (e.g. business-
to-business, consumer, etc.), regulated versus un-regulated, organization ma-
turity (company age or potentially some other metric) publicly traded versus
privately owned, employee location (e.g. centralized versus majority-
remote), degree of integration between software development and opera-
tions teams, and headquarter location (e.g. United States versus Europe).

¹See Appendix E.
• Focusing the phase on a single organization, but selecting informants from
different teams, in different roles, and at different levels within the orga-
nization (individual contributor versus manager). Presumably, the artifacts
used by these informants would be similar in nature and structure, given
the artifacts are produced within the same organization.²

Obviously, there exist pros and cons to both engagement types, largely culmi-
nating in the position of the research’s focal point: in the former, it is a broader in-
dustry focus, delving into differences between different types of organizations, while
in the latter, the focus is solely on a single organizational socio-technical system,
but delving into the differences between actors and roles within an organization. In
essence, the differences are a matter of (from an industrial perspective) “going
wide” versus “going deep.”³

Of course, both lenses offer great potential for findings. Ultimately, for this
research, the second type of engagement was selected: a deep focus on a singular
organization and its (hopefully!) many different uses of these artifacts.

3.2.2 Research Engagement

After selecting the organization, engagement activities were drawn from grounded
theory, case-study approach, and phenomenology to research and explore the or-
ganization’s use of its post-incident analysis artifacts. These activities included:

1. Initial investigation, via structured interview, into the processes used by the
organization to analyze incidents as well as what artifacts result from those
process. (Case-study.)

2. Semi-structured interviews with at least eight informants from at least two
distinct teams/roles within the organization, focusing on their storage, re-
treival, and use of post-incident analysis artifacts. (Grounded Theory and
phenomenology.)

3. Analysis of post-incident artifacts from at least two distinct incidents via a
document review and coding. (Case-study and phenomenology.)

4. Examination of additional artifacts not directly resulting from the post-
incident analysis process outlined in step 1 above, which informants may

²Though, this would remain an open question, to be answered during engagement with the organization.
³To quote Allspaw’s description of heuristics. (Allspaw, 2015, pp. 56-60)
refer to; the goal is to examine where information from these ‘primary’ artifacts spread to so-called ‘secondary’ artifacts (i.e. operational ‘runbooks,’ documentation, bug ticket tracking systems, Agile sprint backlog work items, tacit knowledge discussed by engineers, or other places) within the organization.

3.3 Analysis Methods

Different methodologies were used to analyze data provided from the survey and by the organization.

3.3.1 Survey

The survey data was analyzed in two ways:

1. The tabular question data was broken down across various axes and patterns were sought. The axes were demographic in nature. Two demography questions were asked of respondents (company size and respondent role). As patterns were observed, statistical analysis was undertaken to test the validity of the hypotheses generated from the pattern search.

2. Every post-incident analysis-event template provided was first-cycle coded, using open coding. The two coding profiles used were structural coding (Saldaña, 2010, pp. 66-70), to describe the elemental aspects found in the templates, and descriptive coding (Saldaña, 2010, pp. 70-73) to capture (non-structural) themes repeatedly observed in various templates.

Because the survey data was used to inform and ‘prime’ the Phase II organizational investigation, it was not further (i.e. second-cycle [Saldaña, 2010, pp. 149-185]) coded.

3.3.2 Organization

Data from the organizational research fell into two categories, and was thus analyzed in two distinct ways.

The provided post-incident analysis artifacts were first-cycle coded, using using the same structural coding schema used for the templates from the survey. Where appropriate, the descriptive codes were also applied to the organization’s artifacts.

The semi-structured interview data, was initially analyzed by:
• Listening to the interviews and documenting notable statements or patterns, as a first-cycle Initial Coding. (Saldaña, 2010, pp. 81-86)

• Transcribe the interviews, via an automated (computer transcription) service.

• Review and correct each transcript.\(^4\)

• First-cycle code each interview, using both structural coding (i.e. references to artifacts themselves, practices, and their direct uses) and descriptive coding, to identify common themes.

Dekker and Nyce argue “Researchers must engage in second-order analysis, transforming informant statements about present work into terms designers can use in building future work.” (Dekker & Nyce, 2004, p. 1630) This second-order analysis was performed largely via second-cycle coding methods:

• Each interview was run through a focused coding (Saldaña, 2010, pp. 155-159) process, reducing the original 59 codes to 11.

• These focused codes were then used in an axial coding round (Saldaña, 2010, pp. 159-163), to produce a higher order of analysis and allowing the interviews to be further deconstructed to find coherent complex themes within that data. (Deverell, 2009)

Finally, to validate the data analysis, an inter-data reliability process was undertaken with four industry members not related to the original organization and one human factors subject matter expert. Validation of the structured coding for the survey data and the focused and axial coding for the interview data was conducted with this group. The final results of the research were also preliminarily shared with two additional informants from the subject organization, in order to correct any glaring analytical errors or other issues with the data collection and analysis. (See Appendix G.3.)

3.4 Methodological Summary

Visually, the research methodology can be summarized as follows:

\(^4\)While the automated transcription service had an accuracy of about 60%, due to the technical nature of the language in the interviews, a fair amount of corrections were required. It is also noteworthy the number of times the transcription service produced a written statement that was exactly the opposite of what the informant said, i.e. turning a “no” answer into a “yes” answer, and vice-versa.
A “Black Box”

Requirements
(Functional, Non-functional)

Priorities
(Implicit, Explicit)

Circumstances
("day-to-day work")

Legend

Artifacts Under Study

Artifact Use Under Study

Phase 1

Industry Sample
“Primary” Artifacts

Incident / “Primary” Artifacts

Phase 2

“Secondary” Artifacts

Incident / Post-incident

Figure 3.1: Visual Representation of Research Methodology

Industry Sample
“Primary” Artifacts

Incident / “Primary” Artifacts

Phase 1

Phase 2

“Secondary” Artifacts

Incident / Post-incident
Worthy of note in Figure 3.1:

- The software and infrastructure that make up the ‘production environment,’ individual incidents and outages, and the post-incident analysis process themselves have been treated as a so-called ‘black box’ for the purposes of this research; only the (primary) artifacts resulting from that black box (and the secondary-artifacts, resulting from use of the primary artifacts) are under investigation.

- Phase I concerns the so-called ‘primary’ post-incident analysis artifacts, both from the selected organization and from the broader industry.

- Meanwhile, Phase II concerns the ‘secondary’ post-incident analysis artifacts, limited to those within the organization, as well as the various uses of all the (organization-specific) artifacts by the system’s different actors (in their roles), and whether (and how) such use impacts the future design and operation of the ‘production’ system and/or future incidents.
Chapter 4

Results

The Phase I industry survey not only heavily informed the design and questions for the structured interviews in Phase II, it also yielded some noteworthy insights into the uses of post-incident analysis event artifacts across the industry, at least from a macro-perspective. As such, they are worth reporting in their own right.

4.1 Phase I - Pilot Survey Results

The industry pilot survey, which initiated the thesis research, was finalized on June 12, 2017 and publicly announced on June 28 via Twitter.\(^1\) Over the next few months, respondents were solicited via social media (mostly Twitter), industry presentations, and direct email to DevOps community leaders, to pass along to their social media followers and professional networks.

The survey consisted of fifteen mandatory, multiple choice questions and two optional questions.\(^2\) The questions were split into four categories:

1. **Profile**: Questions about the respondent and the organization they work in.\(^3\)

2. **The Post-Incident Analysis Event**: Questions about the event(s) that occur after an incident or outage to analyze or otherwise make sense of:

3. **Analysis Event Artifacts**: Questions regarding the types, creation process, and content included in artifacts generated as part of a post-incident analysis event.\(^4\)

---

\(^1\)https://twitter.com/jpaulreed/status/880177867698487296

\(^2\)The optional questions solicited post-incident analysis meeting and/or report templates, and allowed respondents to provide their email address, should they desire more information about the results of the industry survey.

\(^3\)Questions 1 and 2.

\(^4\)Questions 7, 12, 13, and 14.
4. **Post-Incident Analysis Event Artifact Usage**: Questions regarding the storage, dissemination, and use of post-incident analysis artifacts after the analysis event has occurred.\(^5\)

Appendix A contains a copy of the survey and results.

A total of 307 responses were collected, the majority of these—71.2%—being collected during the last two weeks of August, 2017. The survey was closed on September 4, 2017. Outside of the survey, two additional responses submitted via email contained long-form description of the respondent’s company’s post-incident analysis activities.\(^6\) While their long-form responses were not converted into survey data, they also submitted post-incident analysis templates, which were included in the template survey analysis.

### 4.1.1 Profile Survey Data

The first two survey questions collected profile information for respondents, namely organization size (total employees) and respondent job title.

The two largest cohorts of respondent job titles (each roughly equal and totaling about 70%) were operations engineers and engineering managers. Respondents were mostly from larger organizations (over 2000 employees) at 29%, followed by organizations with 101-500 employees, at 27%.

Refer to Appendix A, specifically questions 1 and 2, for a complete breakdown of respondent profile data.

### 4.1.2 Post-Incident Analysis Event Survey Data

While the post-incident analysis event itself and its activities were treated as a black box in this investigation, it is still useful to establish a common sense of language and context for these events. As such, the survey asked respondents about some high-level characteristics of these events.

Far and away the most commonly used term by companies for the event where they performed post-incident analysis is “postmortem” (59.6%). “Retrospective” distantly followed at 17.3%.

A third of respondents said they hold one post-incident analysis event a month, with another third holding 2–4 events per month. Notably, 15% of respondents say

---

\(^{5}\)Questions 8, 9, 10, 11.

\(^{6}\)One respondent sent their company’s template via email in order to provide it with a long-form explanation of the context and their use of that template; the other respondent sent a template via email because the survey had since closed, but they still wanted to provide a sample template for the research.
they hold zero events in a month, while 7.5% say they hold two or more events per week. Just over half of respondents say they hold these events within three working days of the incident; 20% of respondents hold their analysis events within 24 hours of the incident.

While the survey did not dig into the identity of, or activities performed by, a ‘facilitator,’ these post-incident analysis events were overwhelmingly (68%) hosted by an organizational actor who played some facilitation role in the post-incident analysis event.

4.1.3 Analysis Event Artifact Survey Data

While examining items recorded or created during the post-incident analysis event, three element ‘bands’ (i.e. groups of first, second, and third most common elements) were identified by respondents as most commonly included in the resulting artifact:

1. Leading the bands, the two most common elements created during the post-incident event are a list of remediation items (90.5% of respondents) and an event timeline (85.3%).

2. The second band were descriptions of the customer impact (60.8%) and technical analyses (66.8%).

3. The third band included an analysis of the organization’s incident response (58.9%) and description of the business impact (56.7%).

This survey question allowed respondents to indicate other items recorded or created and included in their organization’s artifacts; 21 respondents did so. Noteworthy responses included:

• “Root cause analysis” (Multiple respondents.)
• “Where did we get lucky?” (Multiple respondents.)
• “Regulatory impact assessments”
• “Countermeasure Hypothesis”
• “Recurrence Countermeasure Analysis”

7Question 12 did shed some light on the activities of some facilitators.
In answer to the question about which elemental items were recorded during a post-incident analysis event, one respondent noted “This is extremely non-formal at my current place of employment.” In answer to who was responsible for creating or recording these items, another informant said “Usually me, because I’m the only one with enough sense to think that this is important information.” Both sentiments suggest that while some environments may hold post-incident analysis events, they are less formalized or repeatable and/or may not be facilitated in any way, resulting in varying types and completeness of artifacts produced from one incident analysis event to another.

However, the majority of respondents—60.6%—said they use a template for the artifact produced in a post-incident analysis meeting, with the event timeline and list of remediation items being the two most common (three-quarters for both) artifact elements to have a template available for participant use.

The various event-review items would seem to be created either by a single individual (56.4%), or by a various individuals each creating the different elemental items (34.9%) which, in total, make up a particular post-incident artifact. That said, the 8.8% of respondents who indicated ‘Other’ created these event-review items painted a more detailed picture of who actually authored what part of the artifacts, which implies the structure of this survey question was not as clear to respondents as it could have been and/or did not frame the question in a way appropriate to capture their own organizational experience.

### 4.1.4 Post-Incident Analysis Event Artifact Usage Survey Data

Most teams store the elements of these post-incident analysis artifact in a software-based ticketing system\(^8\) (59.0%), with free-form text datastores (i.e. team or company ‘Wiki’-systems\(^9\) [47.6%], Google’s cloud-based Docs product [31.6%]) following as the next most popular artifact datastores.

While not polled explicitly in this survey, the results around this question imply that teams put different artifacts from their analyses into the most appropriate

---

\(^8\)That being either the post-incident review event facilitator or an identified scribe/note-taker.

\(^9\)The survey question was not specific enough to determine whether those separate individuals involved in the creation of these items were spread across roles and teams, or whether they all worked in the same role/team, e.g. and Incident/Problem Management team common in larger enterprises.

\(^10\)Examples include Atlassian’s Jira, ServiceNow, or the open source Bugzilla.

datastores; that is, they do not put all of the artifacts in a single datastore, instead choosing a tool that is able to best represent the data. (For example, remediation items might be best tracked in the team's ticket tracking system, as opposed to a documentation-specific store, because the remediation items describe work to be performed; therefore these artifacts are best tracked in the context of the team’s ‘normal’ work, i.e. tickets. Elemental items such as, say, process changes, might, however, be better tracked in a document store or team ‘runbook,’ since the incident response process may involve referring to those datastores, as opposed to looking at tickets in a ticket tracking system.) At the point of conducting the survey, this was speculation, but questions were built into the Phase II organizational study structured interview script to try to gain more insight into, specifically, this question.

The majority of respondents (57.0%) said they store these artifacts in well-known locations, open to everyone to review. 15.6% of respondents said access to these artifacts was limited to certain people and/or teams. For organizations with open access to these artifacts, distribution and publication to the rest of the organization was reportedly “not consistent” (44.6%), with slightly over a third of respondents having some cadence and process around publishing artifacts and the associated data. Email (46.6%) and chat rooms (40.1%) were the two most popular mechanisms to distribute the post-incident analysis artifacts.

### 4.2 Post-Incident Analysis Event Template Survey Data

One of the industry survey’s final optional questions asked “If possible, please upload a sample blank template your team uses for one of its post-incident review-event artifacts.”

Twenty-four submissions were received, all of them templates related to the organization’s post-incident review-event itself, that is, generally recording information about that event and (as an artifact) created during the event. All of the templates were open coded in order to find common structural elements and common themes. The templates were then coded again at a later date to validate the initial coding; during this second coding, a small number of additional codes were added. A total of thirty-one structural codes and seven thematic codes were identified. These two coding sessions were combined to produce the final post-incident analysis event artifact coding; see Appendix D.

It should be noted: beyond templates that included instructions or definitions ($n = 13$) or described the post-incident analysis event’s objectives ($n = 6$), and a template that happened to be submitted with actual incident data ($n = 1$), it was
difficult to know exactly how the templates were used in situ. As such, the focus of the coding was largely on the structural elements of the templates, in order to gain an understanding of common industry elements in these templates, and to inform investigation during Phase II. This also implies that for templates which do not include instructions, definitions, or objectives, the practice of post-incident analysis may not be codified within the organization beyond either the activities lead by the facilitator or the practices which would fall under general ‘institutional knowledge.’ It may also imply that such templates (especially) are living documents, as much as the post-incident analysis-event itself is fluid in practice within a given organization.

4.2.1 Template Results - Structural

The top three structural elements found in the templates, appearing in at least two-thirds of the samples, were:

1. **Summary** – A short narrative summary of the event and potentially one to two notable analytical findings and/or follow-up items; 79% of templates contained this.

2. **Basic Timeline** – A ordered timeline of incident-related events, created as a byproduct of the incident analysis event (as opposed to being generated output of the incident response process itself, i.e. time-stamped chat logs or operational graphs/metrics, etc.). These basic timelines generally included two pieces of information: a timestamp and an event trace of unspecified type (i.e. in the instructions listed in some templates, suggested events included detection data/methodology, events noticed and/or experienced by operators, and actions taken by operators). In very few cases, the identity of specific event actor was included in each timeline entry, though this was not common.

   (The coding made a distinction between templates including a ‘basic timeline’ containing the above information, and a ‘detailed timeline’ which was defined as a timeline containing either more detailed information than the above or a combination of separate ‘basic’ timelines.)

   75% of the coded templates included a basic timeline; 21% included a detailed timeline.

3. **Action Items** – A list of actions to be taken to fix, remediate, or “prevent” the incident in the future. The open coding made a distinction between a
list of action items and a list of links to the ticketing system(s) tracking any such items. Two-thirds of templates contained a list of action items.

See Appendix A (question 7) for a complete list of the structural elements found in the templates.

4.2.2 Template Results - Thematic

While more nuanced, there was enough information either in templates with instructions and definitions or in structural elements that were notable in their presence (or absence) to identify some common themes. The top themes observed were:

1. **Prevention of Recurrence** – A full third of the coded templates included verbiage referencing the concept of “preventing recurrence” in some form, usually in prompts for sections to list incident “action items.”

2. **De-emphasis of Asking ‘Why?’** – 17% of templates included instructions or references asking those analyzing incidents to de-emphasize discussion of the “whys” of the incident and its occurrence and instead focus on “how” the incident occurred.

   This is notable in that it suggests an intent to challenge use of the “Five Whys” (Liker, 2004, pp. 252-254) as a valid post-incident analytical technique. (“Five Whys” is commonly within the software industry: 21% of templates directly reference its use and two survey respondents said their organization calls their post-incident analysis process, in its entirety, “Five Whys.”) The genesis of this de-emphasis is not clear, though some industry voices have long argued that ‘why?’ is not as useful an analytical question to ask as is ‘how?’ (Allspaw, 2014)

3. **Invocation of ‘The Space’** – 17% of templates included direct instructions or specific vocabulary for facilitators to use when ‘invoking’ the post-incident analysis meeting ‘space.’

   The purpose is to remind the participants of the organizational goals for holding the post-incident analysis meeting, and to set (or reinforce) group expectations about the structure, priorities, agenda, and behaviors during the meeting.

---

12“Lists of Action Items” was its own structural code.
13That is, the ethereal ‘feel’ of the space and, potentially, certain aspects of the physical space, i.e. table/chair layout, no use of laptops/cell phones, etc.
the post-incident analysis event. Specific examples include introductory statements reminding participants of the “blameless” intent for the event or encouraging the documenting of divergent or dissenting observations and opinions about the incident.

Some templates continually facilitated the “invocation of the space” via prompts throughout the template with reminders about the goal for organizational learning, blam[lessness], the complexity of the organization’s systems, and the requirement to also examine “what went well” or how the team was “lucky.”

4. **The ‘No Root Cause’ but ‘Use Five Whys’ Dichotomy** – A number of templates contained an interesting dichotomy: these templates included verbiage noting or reminding incident analysis participants that “There is no such thing as Root Cause.” But then they proceeded to describe steps to perform the “Five Whys” technique, which is a linear cognitive model whose expressed purpose is to identify a problem’s “root cause.”

See Appendix D for the complete open coding of both the structure and noted themes of the 24 template artifacts.

### 4.3 Organizational Introduction

Before presenting the Phase II research results, it is important to contextually position the company with whom the case study and phenomenological research was conducted. For ease of continued reference, the company will be hereafter referred to pseudonymously as “DevOpsCo.”

DevOpsCo is a well-known business-to-consumer (“B2C”) company, offering products in the media space. Founded over twenty years ago, the company has evolved its products, offerings, and brand awareness to keep pace with the changing consumer demands and desires in that space. DevOpsCo’s products are broadly available across the world. As of the beginning on 2018, DevOpsCo reported over 100 million customers, about half of those in the United States. Approximately 40% of DevOpsCo’s staff is employed within the software development and operations departments.

---

14Refer to section 5.2.2 for more discussion of these types of templates.
4.3.1 Industry Reputation

DevOpsCo enjoys a prominent and largely positive reputation within the software development and operations industry. It was an early adopter of cloud-based computing infrastructure, and completed a years-long strategic effort to become fully cloud-native.

Because of this early foray into cloud computing and because, as an organization, it has been open about that transition process as well as the methodologies, lessons, and tooling it employed to complete that effort,\[15\] it is widely considered to be a leader in the space of developing software for and operating infrastructure in a cloud-computing environment. (Interestingly, while most of DevOpsCo’s infrastructure is hosted within a public cloud, not all of it is run within a cloud provider. That said, DevOpsCo no longer operates “traditional” datacenters of its own.)

While being known as an industry technology innovator and leader, DevOpsCo is simultaneously known for its unique approach to how it runs its (non-technology) operations and how it treats employees. Among Silicon Valley companies, where it is headquartered, it has a unique and highly identifiable corporate culture. Again, this is largely due to the public nature by which it demonstrates (even advertises) its cultural values. There have been many public discussions around whether certain actions or behaviors represent a “DevOpsCo-like culture” or the antithesis of such, which is notable in that the company’s self-identity is clearly established internally, but also externally beyond the walls of the company. Because of this, many software engineers in Silicon Valley associate DevOpsCo’s name with specific small, memorable, ‘pithy’ slogans representing DevOpsCo’s culture (which DevOpsCo fosters and encourages).

Finally, DevOpsCo maintains a heavy presence on the technology conference circuit, with many of its engineers speaking on myriad different technical, operational, and organizational topics.\[16\]

4.3.2 Market Reputation

DevOpsCo also enjoys a relatively positive reputation in the economic markets; DevOpsCo currently has a market capitalization of over $100 billion US, and has enjoyed increasing revenue per share for the last decade.

---

\[15\] Many of these tools have been released publicly, licensed as open source software.

\[16\] DevOpsCo has on occasion had numerous speakers talking about the company’s policies, practices, and innovations at a single conference, so much so, the event has been jokingly referred to as a ‘DevOpsCo-conference.’
Aside from the performance of DevOpsCo’s stock, the financial press has created specific terms to refer to a class of company which has a notable impact on the technology sector as a whole; DevOpsCo has been described in these terms, indicating the financial press’ general views about the importance of its stock and impact to larger market indicators. In fact, some analysis has been done in the financial press showing the S&P 500 index can be impacted by fluctuations in DevOpsCo’s (and so-called ‘sibling stocks’) stock price, which in turn can lead to fluctuations in the larger market.

As with many stocks with this performance history, there is currently much discussion about whether DevOpsCo’s stock is overpriced, part of a larger ‘bubble’ in the technology sector. What is notable here is not so much the substance of that discussion, as the fact that DevOpsCo’s stock is so often referenced as an indicator during larger, market-focused discussions.

4.3.3 Informant Panel

In order to gain insight into not just one, but various departments’ and actors’ uses of post-incident analysis artifacts, the panel of informants consisted of employees from three major operational areas within DevOpsCo: software development, (cloud) operations, and security.

A total of twelve informants were interviewed and contributed to the collection of DevOpsCo’s post-incident artifacts created during the research. These informants represented a mixture of individual contributors and managers, with a wide-spanning range of both tenure at DevOpsCo and career experience in the broader technology industry.

The summary table below lists the various roles, operational areas, and other pertinent statistics about the panel of informants, as of the date of the organization case study.
<table>
<thead>
<tr>
<th>Team</th>
<th>Anonymized Title</th>
<th>Company Tenure</th>
<th>Industry Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operations</td>
<td>OpsEng1</td>
<td>5 years, 11 months</td>
<td>24 years</td>
</tr>
<tr>
<td>Operations</td>
<td>OpsEng2</td>
<td>9 months</td>
<td>9 years</td>
</tr>
<tr>
<td>Operations</td>
<td>OpsEng3</td>
<td>1 year, 1 month</td>
<td>18 years</td>
</tr>
<tr>
<td>Security</td>
<td>SecEng1</td>
<td>4 year, 8 months</td>
<td>21 years</td>
</tr>
<tr>
<td>Security</td>
<td>SecEng2</td>
<td>3 year, 2 months</td>
<td>9 years</td>
</tr>
<tr>
<td>Security</td>
<td>SecProgramMgr1</td>
<td>8 months</td>
<td>7 years</td>
</tr>
<tr>
<td>Security</td>
<td>SecMgr1</td>
<td>2 year, 3 months</td>
<td>13 years</td>
</tr>
<tr>
<td>Development</td>
<td>Dev1</td>
<td>6 years, 5 months</td>
<td>17 years</td>
</tr>
<tr>
<td>Development</td>
<td>Dev2</td>
<td>4 years, 11 months</td>
<td>16 years</td>
</tr>
<tr>
<td>Development</td>
<td>Dev3</td>
<td>1 year, 9 months</td>
<td>14 years</td>
</tr>
<tr>
<td>Development</td>
<td>Dev4</td>
<td>6 years, 1 months</td>
<td>15 years</td>
</tr>
<tr>
<td>Development</td>
<td>DevMgr1</td>
<td>2 years, 3 months</td>
<td>28 years</td>
</tr>
</tbody>
</table>

Table 4.1: Panel of Organization Case Study Informants

4.4 Phase II - Organizational Study Results

4.4.1 Most Common Artifact Uses

Each of the twelve interviews were open coded to capture not only references to specific processes (incident response, incident analysis, or other post-incident events and activities), tools (ticketing systems, team collaboration tools, etc.), and post-incident artifacts, but also direct references to the use of any the artifacts generated during these post-incident activities. Repeated organizational themes were also included in the open coding.

Setting aside codes related to the mechanics of specific organizational processes, tools, and the incident artifacts themselves, the top five codes observed in the interviews, by frequency, were:

1. Artifact Usage for Trend Identification & Analysis – References to using a specific artifact to identify and analyze, specifically, system trends, both technical and social, inter- and intra-team.

2. Creation of Domain- and Group-Specific ‘Tribal Knowledge’ – Direct references to the creation, curation, and communication of internal organizational “tribal knowledge” within and among teams, especially around technical design patterns that do (or, more commonly, do not) work within the
company’s larger system or operational practices that have been successful for the organization in the past.

3. **Artifact Usage for Work Planning** – References to the use of artifacts in the process of planning future work; while many of these references were to what could be considered ‘remediation items’ as a result of the incident, there were also a surprising number of references to direct and indirect use of the artifacts to manage the intake and prioritization of work items (which may not be related to the incident itself), and also into larger technical system design projects, which may only be tangentially related to the specific incident that was the genesis of those artifacts.

For example, multiple informants recalled referring themselves and their colleagues back to artifacts to either create requirements specifications for large refactoring projects, or when justifying specific technical decisions in those specifications to others. In this way, the analysis performed and recorded in the post-incident artifacts directly influences the future of the system, both from an operational and architectural standpoint.

4. **Creation and Promulgation of “Context”** – Similar to the use of justifying requirements described in item 3, informants made numerous direct references to creating “context”—their word—to discuss topics within their teams, to pass along to other teams, to set expectations to colleagues and management, and to position data reported to management. In fact, the use of artifacts to create and share “context” to others was so prevalent in the interviews and pervasive in the way informants describe their communication, this “context” could almost be considered a ‘secondary artifact’ of incidents, albeit a more cognitive and therefore ephemeral one.

5. **Storytelling** – Informants made multiple references to the use of post-incident analysis artifacts to tell stories to each other and others (including externally, at conferences and in the broader industry) about how they operate their systems, the sharp(er) end edges of those systems, what strategies and patterns were successful and unsuccessful in the course of performing work and so on.

This was coded as distinct from “tribal knowledge” because informants spoke of “tribal knowledge” as a more ethereal concept. (As an analogy, we might think of it as the “tribal knowledge” a fish possesses about surviving in water, with a specific school of fish: once a fish is inducted into that ‘school,’

---

17 Usually in the form of analysis documents and write-ups, not remediation items/tickets in the ticketing system.
the knowledge within the school is furthered as a group, but often not explicitly. In terms of ‘storytelling,’ we are referring separately to a process that, to continue our analogy, is like a fish telling another school of fish about its own school’s experiences; or like giving a presentation to a group of mammals about piscine life.)

Another way to parse the distinction: “tribal knowledge” was referred to less as the size of the ‘tribe’ being referred to grew: that is, informants commonly referred to “tribal knowledge” within their teams, or within their operational areas (Development, Security, Operations), but referred to “stories” and “storytelling” at the broad organizational or industry level. In a sense, “tribal knowledge” became diluted as the ‘tribe’ became larger, and “storytelling” (along with context-sharing) served as a way to pass tribal knowledge along to other parts of the company’s socio-technical system. In that regard, storytelling is unique because it involves an explicit construction of a cognitively recognizable narrative, including structured elements such as specific character archetypes and a lesson or ‘moral of the story.’

(This ‘morphing’ of the conception of the part and parcel of work and messy realities of organizational life is similar to the morphing of narratives describing these aspects of work in Swedish rail as observed by Sanne, 2008. In their case, the equivalent of DevOpsCo’s “tribal knowledge” is “integrated into the participants’ cultural frame and suited to their daily needs.” (Sanne, 2008, p. 1218) The challenges with storytelling versus incident reporting that Sanne describes, however, are dissimilar to DevOpsCo’s experience. What is similar here is the structure, expression, and utility of propagating the details of an incident or outage morph as we move from the sharp end of the system to the blunt end: in DevOpsCo’s case, the metamorphosis is from locally rational, tacit team-level knowledge to knowledge which is distilled to a more widely applicable frame, a defined narrative structure created to contain and give form to the knowledge, and then explicitly to propagate that knowledge.)

While the number of additional open codes was quite high (61 in total), when the process (4), tooling (7), and artifact (16) codes are removed from consideration, the remaining codes—direct references to uses of artifacts, or repeated themes—do (perhaps unsurprisingly) support or relate to the five most frequent codes above.

Other notable, frequently encountered codes included:

• ‘Long term’ usage of an artifact
• Use of an artifact to create a ‘knowledge base’
• Examples of a specific usage of an artifact for analysis purposes
• Use of an artifact to create documentation
• Examples of a specific use of an artifact for software requirements specification purposes
• Use of an artifact in relation to (enterprise) risk management

The full list of open codes related to the types of post-incident artifact use can be found in Appendix F.

4.4.2 Different Groups Use Artifacts Differently

One finding observable even in the first-order open coding frequency data was the different ways in which operational teams—developers, operations, and security—make use of post-incident artifacts.

Generally, operations and security team members talked more than developers did about using the artifacts to perform operational and technical trend analysis and to refine their operational and incident management processes (codes 3.b, 3.j), including the review (and potential re-review) of artifacts involved in past incidents (code 3.k). Related to these uses, operations and security teams reported that their use of the artifacts, specifically prompted them to conduct further analysis and/or research on a particular incident or topic more than developers used artifacts for this purpose (code 3.h.2). This seemed to compliment another difference in usage: in the aftermath of an event, operations and security engineers would more often use artifacts for the creation and curation of a knowledge base and for writing additional documentation (codes 4.c, 3.d). Additionally, both the operations and security teams used the artifacts to construct narratives and tell stories about incidents more often than developers used the artifacts for that purpose (code 4.d). And finally, operations and security teams used the artifacts for risk analysis more often than developers did, and for coordinating with external parties more often (codes 3.f, 15). Table 4.2 summarizes these observations in the data.
The above might imply development teams do not use of post-incident analysis artifacts as often as their their operational- and security-focused colleagues. This is not so: DevOpsCo’s development teams just use them differently. Their use of the artifacts tends to play a more prevalent role in the process of planning their future work (code 3.c), specifically and directly informing the requirements of development and refactoring projects they are discussing, designing, and writing specifications for, prior to implementation (code 3.c.1).

While the security team’s descriptions of their usage of post-incident artifacts mimicked in many ways the operation team’s usage of the artifacts, the security
team did reference their use of the artifacts in the planning of future work (code 3.c) more often than the other teams. This was interesting, given they combine this work-planning use with trend analysis (code 3.b).\textsuperscript{18}

While the creation of tribal knowledge (code 4.b) was referenced by all teams,\textsuperscript{19} the operations team referred notably often to the use of artifacts for this particular purpose. Relatedly, references to the role the artifacts play in on-boarding and training purposes for newly-hired employees (code 3.a), i.e. as a way to induct new members into the ‘tribe,’ was strong in the operations cohort. These two uses may be effectuated through the use of documentation and storytelling, which may explain the notably high references by the operations team described above. The operations team also bucketed the artifacts, a derivative, meta-use of the artifacts, so as to start to analyze trends and cohort (at the company level) differences in incidents. This is an interesting (though not surprising) finding, given that a the operations team is tasked with finding ‘commonality’ and ‘trends’ among organizational incidents, and bucketing is a natural strategy to do that.\textsuperscript{20}

Of course, there were some ways in which all the teams made use of the post-incident artifacts: all groups made a made prevalent reference to the use of artifacts to help them create and communicate context (code 7). This may be less indicative that the teams’ usage in this context is similar as much as an indication of a structural property of the way the organization aligns its actors and conducts its work (in which case, it would be beneficial to employ the same mechanisms to achieve this, i.e. reliance on “context”).

Finally, while not an example of different usage of post-incident artifacts, one remarkable difference in operational teams: their references to the concept of “root cause” in the structured interviews: operations informants referred to the concept the least, followed by security informants, and developers. Coincidentally: the number of references per informant to “root cause” roughly doubled in each operational cohort.

\textsuperscript{18}The security team informants reported the trends they pay attention to involve emerging and on-going security threats and changes in the security landscape.

\textsuperscript{19}Supporting the notion that “tribal knowledge” is not only especially important at the team and organizational group level, but also strongest, most pertinent, and therefore most useful in that scope.

\textsuperscript{20}Though notably, there was much discussion during the informant interviews on the actual utility of incident-bucketing. OpsEng1 said “There was an earlier attempt at bucketing [incidents] with labels to say this is a [Type A problem]. This was a [Type B] problem. The issue we had with that was way too many labels and not enough tickets for there to develop a useful pattern or anything interesting other than ‘Hey, a [Type A] change popped up once a quarter.’” OpsEng1 continued “Honestly, we’re still kind of figuring it out. We found a lot of ways not to do it well.”
4.4.3 Artifact Usage During an Incident?

One of the pressing questions this research hoped to definitively address was whether or not post-incident artifacts are used during future incident response. The answer is a definite... probably.

When directly asked whether or not past incident artifacts were used during an incident response response, some informants asserted this seldom, if ever happens:

> During an incident, we probably would not, in the initial steps, pull up, like try to find related things that have... well, seemingly related things that have happened in the past regarding whatever is being affected. We would not do that.

— OpsEng2

> So, I'd say it’s so hard to make a connection what I’m seeing right now and what’s, how a user is affecting it and what the solution was or how they experienced it six months ago. So I wouldn’t want to waste hours looking for something... And if I went back to some old, if I leaned heavily, and this is where I can say I’ve seen engineers who leaned so heavily on their assumptions, or they’ll do a search “Who else saw this problem?” The world’s constantly changing. You can’t put a lot of weight onto what you find in an [incident ticket] from two weeks ago.

— Dev1

However, other informants, from across the various operational cohort groups, did give detailed accounts of situations, seemingly both hypothetical and drawn from personal experience, in which they would or had used the artifacts during an incident:

> For example, we had a system called ['SysAlpha'] and we had a series of incidents while we were rolling it out over this past summer, so when we had a new incident, I’m like “boy, this sounds familiar.” And I went into that [incident reports] folder and I searched for the word ['SysAlpha'] and I was able to find the exact answer that I was looking for. So it’s kind of a human thing, you know, you remember things. It’s not written down, we
don't have them indexed in any way, but a lot of us have good institutional and organizational memory and we'll go back and try to find it that way.\textsuperscript{21}

— DevMgr1

One other big [use] would be when you're in a new incident and you're like “Wow, this just feels familiar. This feels like about three months ago, wasn't there that thing?” So then you're like “Oh yes, there it is.” And [you're then] reviewing your notes from there. ... So I look at [my own notes] from that time period, I find extra details that I kept just for myself. Sometimes I'm able to give novel or really specific insights on a new incident that’s happening. Like “Oh, we’ve seen that before, you need to do X, Y, and Z” and I think that would be another big reason for reviewing other [incidents]. You prefer to not ever firefight the same incident twice, but sometimes there are resonances, right?

— Dev2

So we have like a similar incident in the future, it’s a reference for us to go back to; and that’s, at least in my case as a responder, when I’ve gone back to those artifacts, it’s usually just, again, it’s been just to get some context as far as what may have happened in the past, which to either give guidance as far as how should we respond now? Or if there’s a particular threat actor, that we suspect might be coming at us again. ... Just the way to go back and figure out what's going on.

— SecEng1

In one example, the informant said they would not personally make use of the artifacts during the incident if they were the in Incident Commander role, but based on some heuristic, they would ask the team to make use of the artifacts:

So I think I won't do that analysis as an Incident Commander as much that I'll say “Hey team, can you take a look at the solution that was implemented for this security ticket that was opened? I, from looking at it, looks like they

\textsuperscript{21}DevMgr1 went on to contextualize their behavior and task goal thusly: “So often times during an incident, the engineers were really focused at looking on the graphs, looking at what's happening at that moment in time. And if I'm involved, being the pointy-haired manager, I kind of look more at historically what's happening. ... If I happen to be around, I help out by doing more of the ancillary searches for historical kinds of things to say 'Hey, did you guys consider looking here?' It really just provides more rocks to look under.”
might be related, they might not be. So can you please take a look an see did we miss something or this is a totally different thing and a different part of the architecture or the code?” So I would definitely ask that question.

— SecProgramMgr1

It is unclear why this dichotomy existed in the informant responses, but the difference between the informants’ answers a more complex mechanism at play and/or interaction with the artifacts during an incident response situation (which may be associated with the roles a particular organizational actor tends to play in that situation) that is beyond the scope of the research question.

That said, there is certainly interesting overlap with Allspaw’s findings regarding the heuristics teams use during incident response. (Allspaw, 2015) For instance, developers especially reported a “vague recollection” or “smell” that prompted them to go search for previous post-incident artifacts.22

This supports the third heuristic Allspaw defined, which he called “Convergent Searching” (Allspaw, 2015, pp. 57-60), including its two-subtypes, i.e. “Look to confirm/disqualify a specific and past diagnosis...” or “a general and recent diagnosis that comes to mind by matching signals or symptoms that appear similar.” (Allspaw, 2015, p. 57) This would suggest that as a functional act of implementing this heuristic, engineers may, as a possible technique, search for (and, if successful, use) post-incident artifacts in the context of understanding and reacting to an active incident. But, as previously stated, this process is clearly more intricate than that, since the common initial informant reaction to this question was that they did not use the artifacts in that way

---

22DevI described this experience thusly: “I’m like ‘Crumb, this is really, really familiar. There must be so... how do I find it again?’ I’d say this is a real weakness there in that a lot of times people will have this hint in the back... there’s like a smell in the air. ‘Oh, this smells like a problem from six months ago. But what was it? What was the problem?’”

47
Chapter 5

Analysis

The process of analyzing the Phase I survey data and collected artifact templates and the Phase II organizational interviews and internal artifacts represented a marked shift from the open coding, bucketing, and tabulation presented in the results above to a secondary, higher order analysis focused on refining the open coding. Those refinements were the driving force behind the second-order analysis and sensemaking of the refined codes, facilitating the identification of the complex, inter-connected, three-dimensional themes both in the structured interview data and in the provided artifacts, from both the broader industry survey and the organizational case study and phenomenology data.

Ultimately, the survey data played an incredibly helpful role in informing the structured interviews and highlighting themes to explore further with informants; also, the broader industry artifact templates obtained from the survey helped to provide a dataset with which themes in the organizational artifacts could be compared and contrasted against.

5.1 Survey Statistical Analysis

The survey results were, for the most part, stable when statistically analyzed across the survey profile data, i.e. job title, ‘combined job titles’ (e.g. ‘developers,’ ‘operations engineers,’ and ‘management’), and organizational size. In effect, the reported survey results showed no strong relationships between these profile factors and the answers in the survey... with some exceptions.

Two statistical results in particular demonstrated absolutely no correlation:

1. Organizational Size versus Use of Artifacts to Update Documentation: there is no evidence of a relationship between the size of the organization and
whether or not it creates documentation updates ($p = 0.939$). In looking at the data, it’s roughly a coin toss chance whether an organization uses incident artifacts to update their documentation, and this is true whether the organization is large or small; put another way, large organizations are no better (or worse) than smaller organizations in updating their system or operational documentation in response to incidents.

2. **Organizational Size versus Use of Artifacts to Create Remediation Items:** there is also no relationship between organization size and whether the organization creates remediation items ($p = 0.961$). The data is clear on this point: 9 in 10 organizations, immaterial of number of employees, use post-incident artifacts to create remediation items. This is consistent with other survey data which suggests the reasoning behind creating these items is to “prevent recurrence.”

While there were few statistically significant correlations found between profile data and survey results—meaning, in effect, that the numbers presented can be taken at “face value” in terms of industry practices, but are not unique or specific to a particular job title or size of company—twelve artifact usage behaviors and characteristics were found that were dependent on these profile factors, and one that was almost statistically significant. The categories showing a relationship were:

1. **Combined Role versus Use of Artifacts to Update Documentation:** While organizational size had no relationship with use of incident artifacts to update documentation, interestingly, job titles and the use of artifacts to update documentation did show a relationship ($p = 0.004$): operations engineers used incident artifacts more often (55% of the time) to update documentation, runbooks, and the like than did developers (30%) or managers (40%).

(Anyone with experience in an operations role is unlikely to consider this finding surprising or remarkable: keeping operational documentation, i.e. runbooks, wiki pages, etc. is a familiar activity to that role; or they are familiar with lament over their team not having time for this activity.\(^1\))

Relatedly, this correlation extended to template usage: both operations engineers (17%) and managers (14%) reported using templates to perform these documentation updates more than did developers (2%; $p = 0.032$). It is not

\(^1\)Especially in hindsight, while working an incident or outage!
clear which specific practice(s) would explain these differences, but it is possible that the type of documentation being updated plays a role here: if operational runbooks or other structured documentation is being updated or created in response to an incident, then it is reasonable to assume that in some organizations, these documents could be created from templates that allow engineers to standardize the structure and format of the document for easy (operational) searching and reference later.

2. Organizational Size versus Use of a Template Artifact: The size of the organization does factor into whether it will use a template artifact to drive an incident analysis process ($p = 0.001$), specifically smaller organizations (0-50 employees) do not tend to use templates—only 35% of these smaller companies say they use templates—while every other larger size of organization is 60% (or more) likely to use one.

![Percentage of Organizations Using Artifact Templates](image)

Figure 5.1: Post-Incident Artifact Template Usage Among Different Organization Sizes

This finding is also intuitive, especially given some of the write-in answers regarding the non-standard formats of retrospectives and the value that smaller organizations may place on incident analysis and management activities.

As noted above with the correlation of template usage and job role, there exist specific types of template usage that were correlated with organizational size:

(a) **Timeline Templates**: As above, smaller organizations use a template for the timeline less often (30%) than larger organizations ($p = 0.004$);
however, interestingly, the peak usage of a template for the timeline is in mid-sized (101-500) organizations (65% of the time), but dips by 7 to 13% for other-sized organizations (51-100: 56%; 501-2,000: 58%; 2,000+: 53%).

(b) **Remediation Item Templates**: Like timeline templates, remediation item templates are most used among mid-sized organizations (51-100 and 101-500: 59%, \( p = 0.001 \)), with less use by smaller organizations (24%), and a curious drop-off as organizations get larger (55% for companies of 501-2,000 and a surprising 43% for companies over 2,000!)

(c) **Business and Customer Impact Templates**: Here again, the most use of templates for describing these two impact-related items is by mid-sized companies: both the 101-500 and 501-2,000 cohorts showed higher template usage for these two impact items than did both their smaller and larger counterparts. Business impact templates were used 30% and 38% of the time, respectively, dropping off to 25% for smaller and larger cohort sizes (\( p = 0.035 \)); and customer impact templates were used 44% of the time for both the mid-sized companies, dropping to 25% for smaller companies and 30% for larger companies (\( p = 0.009 \)).

![Figure 5.2: Usage of Specific Types of Post-Incident Artifact Template Usage Among Different Organization Sizes](image)

Again, the reasons behind these interesting mid-sized company ‘peaks’ are not immediately known, but there does seem to be a ‘sweet-spot’ in terms of template usage by companies. One possible reason for this may be that smaller companies, in general, don’t tend to use or have templates, and as companies grow, they put in place structured templates for various items.
that the organization decides it is important to report on. Per the findings reported above, as the organization crests over 2,000 people, template usage for any of these items and/or artifacts is less, and where they are used in larger organizations, they may be inconsistently used, or multiple templates are used by different teams. This bifurcation could emphasize different items (and thus these specific items do not have a strong showing, consistently, across the company’s usage of templates, which would be observed in a dropoff of certain items for larger companies).

In any event, further study would be necessary to definitively describe the reason(s) for these differences in template usage of different-sized companies. Two study design issues further research should consider are the arbitrary company size categories used in this survey, as well as the somewhat subjective deconstruction and bucketing of ‘types’ of templates offered as options in the survey.

3. **Organizational Size versus Post-incident Artifact Accessibility**: At the edge of statistical significance, organizational size is weakly associated with accessibility of post-incident artifacts \((p = 0.051)\): the survey data suggests that as organizational size increases, especially over 2,000 people, post-incident artifacts become generally less accessible (41% in such larger organizations, versus over 63% for organizations up to 500 people). Additionally, 13% of organizations under 2,000 employees report restricting access to their post-incident artifacts; for companies over 2,000 employees, this number doubles to 27%.

The reason for this is not clear; possibilities include it not occurring to teams in larger organizations to share any incident analysis findings widely and broadly, teams may not be empowered to share their findings widely and broadly, or it may not be culturally acceptable or ‘safe’ for teams to share findings widely and broadly in larger organizations. In any event, this finding suggests more research on this specific question could yield interesting results.

The last two statistically significant hypotheses are related to this artifact availability and distribution:

(a) **Distribution via Chat Rooms**: Companies of 51-100 and 101-500 use chat rooms to distribute post-incident analysis artifacts more often than all other groups (56% and 51% respectively, \(p = 0.001\)); this number drops to 26% for organizations over 2,000. Reasons behind this
were not specifically probed, but a potential explanation is companies circulate these artifacts through their ‘normal’ communication channels, and chat rooms are used more by mid-sized companies (and/or large[r] companies have more bifurcated communications mediums in use by their employees).

(b) **Distribution via Company Blog**: Distribution of artifacts via a company blog is uncommon, with around 12% of companies of size 101-500 doing this. What is interesting is how much this drops for the other cohorts: 0% (!!) to 4%. Reasons for this were not investigated as part of this research.

Two additional relationships between organizational size and incident analysis practices were established\(^2\), but as they are beyond the scope of the research question, they are presented in Appendix B.

See section 6.4 for some final thoughts on the statistical analysis of the industry survey, Appendix A for the survey questions and aggregate results Appendix B for the statistical analysis of the survey, including the hypotheses tests and a breakdown of hypotheses showing correlations, and Appendix C for the software source code used to perform the analysis.

---

\(^2\)Namely: organizational size is correlated both to the number of post-incident analysis events that occur and how long those events occur after the incident.
The Role of Artifact Templates

To analyze the post-incident analysis templates, the elemental structures found in each template were reviewed in context with various repeated themes also found in the templates (see section 4.2.2). Taking these two aspects of the templates together began to paint a picture of different template ‘archetypes,’ that is, there exist templates that serve different roles, and organizations use these templates to facilitate certain operational practices and, in some cases, facilitate certain behaviors within the context of the organization's process of creation, curation, and consumption of the (completed) artifact after an incident.

The differences in these template archetypes is so fundamental that they can be observed by mere visual inspection: a sample of the structure of each type of template is provided, to highlight these differences.

The ‘Record-Keeper’

This archetype of template is likely what most practitioners think of when they conjure up an image of a post-incident report. Structurally, it contains the information collected and discussed during a post-incident analysis event. The document is organized for easy lookup (i.e. generally outline format, with specifically defined section headers, useful for skipping to specific sections), usually contains lists of references to other record-keeping systems (ticket tracking system for remediation items, though sometimes data storage systems, such as operational graphs, metrics, and data or source code control system references) and includes
some summarization of information the organization considers important to highlight in their post-incident reports.

The role the ‘Record-Keeper’ template is primarily to record, at an organizational level, that a post-incident analysis occurred for a particular incident. For some organizations, the extent of the record they wish to keep is quite small (as an example, one particularly memorable template sample included just three sections: “Summary,” “Actions,” and “Timeline”); other organizations structure their templates to contain not only the data from the incident (timeline, TTD, TTR, business/customer impact, etc.), but all of the analysis (graphs, etc.), discussion, and remediation items, learnings, or outcomes from the incident analysis.

Fundamentally, this archetype of template serves the purpose of ‘memorializing’ the incident for the organization. (Incident ‘memorialization’ was a term used by informants in the Phase II organizational study. The process might be different in different organizations, but generally refers to the process of remembering, processing, potentially ‘grieving,’ and internalizing an incident, as a way to ‘cognitively digest’ it.) Despite this, it doesn’t imply anything about what the organization does as a part of that memorialization process, or even whether the organization acts on or ‘learns from’ the incident. They may; but they may not. In this way, the template can serve to ‘check the box’ in the organization’s consciousness that it responded to, analyzed—to the extent the organization is capable of that—and remediated the incident (again, to the extent the organization considers that function a priority), but it does not provide any indication as to whether the organization learned from the incident or is capable of deploying any learning that did occur.

---

3In the organizational study, various informants discussed this “memorialization” process in a way reminiscent of the five-step Kübler-Ross grief model. (Kübler-Ross, 1969)
5.2.2 The ‘Facilitator’

The ‘Facilitator’ template shares many of the same elemental structures as the ‘Record-Keeper,’ with one notable difference: instructions, prompts, and questions permeate the entire template document, serving to facilitate the process of the post-incident analysis event itself. For these templates, recording the results of the analysis event is secondary to helping to structure the post-incident analysis event itself. Though the template may still contain fields to be completed during the analysis process (so as to still serve a recordkeeping function), providing a level of consistency in the analytical process, the issues participants are asked to consider, and the framing participants are asked to hold is the primary concern of these types of templates.

Different templates focused on this to different extents: in some templates, it was clear that the purpose of the template was not primarily concerned with recording the analysis for the organization: rather, it was to provide guidelines and add guardrails to the process of holding the event and the interactions and discussions occurring during and after that event. In other templates, the prompts and questions were designed to be filled in, almost like a worksheet. But no matter the extent of the facilitation, a commonality in all of these types of templates is that the facilitation ‘voice’ was present throughout the entire template, as if to take the group present at the post-incident analysis event ‘on a journey’ every time it is held.

These types of templates were the most illustrative of themes that were important to organizations: some templates placed much emphasis on the ‘invocation of the space’ and were concerned with the interactions and the frames participants held during the analysis (generally in the form of ‘blameful’ versus ‘blameless’ and the use of counterfactual reasoning), while other templates focused more on prompting questions and attempted to highlight specific aspects of incident analysis the organization values.
As mentioned above, while the ‘Record-Keeper’ is the template type practitioners are likely to think of first, it is notable there were more ‘Facilitator’ templates, by a small number, collected in the survey. This would indicate that as the software development and operations industry begins to mature in its incident analysis and management practices, more organizations are trying to provide in situ, sharp-end guidance to their teams on what the organization considers important and how it expects engineers to conduct those post-incident analysis events.

5.2.3 The ‘Sign-Post’

A couple of templates provided stood out as so different from others that even though there were just a few observations in the data, they were so unique and distinct, they warranted their own archetype: the ‘Sign-Post.’ These templates serve one function: to direct people reading it to other systems of record, and to do so as quickly and clearly as possible.5

Common structural features of this template type include a short-form narrative summary of the incident, incident statistical information (date, business and/or customers impact statistics, etc.), and in some cases, an ‘eye-catcher,’ usually in the form of a graph related to the incident or a diagram of the system architecture. The other information contained in the template are links and references to other systems-of-record (usually ticket tracking or internal wiki/documentation systems) where readers can obtain more information about that incident should they desire.

---

4One from the survey and one from the organizational case study.

5For this reason, the name ‘Index’ template was also considered, like an index card in an old-style library card catalog.
In one template sample, the information provided was quite minimal and the document itself quite sparse: this sign-post template consisted of just two tables, fitting on a single page. The first table had rows for pertinent, short-form information about the incident review, i.e. date, incident ticketing system reference, incident severity, time-to-resolve, “root causes” (both business and technical), incident status, and document version. The second table contained a list of remediation items agreed to in the incident review, again with the short-form, referential details, i.e. the remediation action, a ticketing system reference to track that action, the owner, due date, status, and progress updates.

In the organizational case study example, this template was used to communicate information about large, “important” incidents to the rest of the company. The goal was also to be short, easy-to-read, and highly consumable, but without being ‘secretive,’ withholding information, or failing to offer citations for claims or further resources. One informant succinctly described this template archetype’s purpose:

> We have an “above-the-fold”-goal, that you should be able to look at the little bits here in the top and reasonably understand what’s going on without having to read the entire stream of comments and try to load all of this stuff up into your brain.

— OpsEng1

In this manner, these ‘Sign Post’ templates contribute less to the full organizational “memorialization” of the incident, and they do not facilitate the incident analysis process, but they do serve as a way to propagate information about incidents the organization deems important throughout the organizational consciousness, in a quick, efficient, but also credible way.6

5.3 Artifact Usage in High-Performing Organizations

High-performing software development and operations organizations go about their work differently.

This has been widely documented over the last decade in talks such as the seminal “10+ Deploys a Day” (Allspaw & Hammond, 2009) up through research presented in the 2017 State of DevOps Report: high-performing teams deploy their

---

6This is, in many ways, the antithesis of many safety programs in other industries, where signs with “Days Since Last Accident” are plastered all over a job site and posters expressing management’s exhortations for system actors to “Be Safer” and “Pay More Attention” are thought to somehow positively influence operational outcomes.
software 46 times more frequently than their low-performing counterparts, their mean-time-to-recover (MTTR) is 96 times faster, and they are one-fifth as likely to have any single random deployment cause an incident or outage. (Forsgren, Humble, Kim, Kirsten, & Brown, 2017, pp. 21-25)

Based on these (and other) metrics, DevOpsCo can be defined as a high-performing organization. An element of its high performing characteristics is DevOpsCo’s notable departure in the way they use their post-incident artifacts. These often appear similar to common and ‘commonsense’ industry practice, but DevOpsCo frames the purpose and end-goals differently, thus resulting in different (and improved) outcomes for DevOpsCo, both operationally and with regard to organizational learning.

5.3.1 Mapping the Emergent Complex Socio-technical Terrain

A well-known characteristic of complex adaptive socio-technical systems is their emergent nature and their state of constant evolution, hence the name. (Snowden, 2002, pp. 105-106) When humans attempt to make sense of these systems, via analysis, the difficulty in grappling with these properties has often been described with an analogy to Heisenberg’s uncertainty principle: the position of (various bounded subsystem) particles can be known or their direction can be known, but not both. So it should not be surprising that our attempts to understand our complex systems and learn at an organizational level are often frustrated by these characteristics.

DevOpsCo uses their post-incident artifacts as a sort of ‘map’ to help them evolve their own mental, “above the line” models (Woods, 2017, pp. 7-12) in two specific ways:

1. The artifacts provide evidence of linkages between disparate, ‘seemingly unconnected’ elements of the larger system, which actually turn out to be connected (often ‘implausibly’) in some way that had yet to be observed. These can be longstanding latent connections:

   [The issue] wasn’t happening all the time. So it was a very difficult problem to debug. And the way our ecosystem was at that point in time, there were dozens of systems that could have potentially been the culprit and it took us several weeks to determine what happened. And

---

7The analogy is often extended with an additional reference to the Observer Effect: by merely observing our complex socio-technical systems in a deliberate, analytical way, we contribute to and affect their continued emergent behavior and evolution.
it turned out it was probably something that had been in the ecosystem for three or four years, but it didn’t manifest itself until we moved to bigger computer servers with more capacity. ... So from that incident, we were able to analyze the situation and to find a path forward and declare a new architecture.

— DevMgr1

Or they can be new connections that are the result of system evolution:

Because the product is constantly evolving and improving, we may have indicators that were good at what we had in the past, but as we add new functionality or change our functionality or we refactor it, the expression of the symptom of the problem can change. ... And the reason I’m giving this example is that we’re constantly striving to make our indicators better. Even if it’s not obvious. Like the alert we had was probably good enough the first time around, but as we evolved and we encountered a slightly different manifestation of the problem, the alert, we found, wasn’t good enough.

— Dev3

2. They also serve to ‘shine a flashlight’ on ‘hot-spots’ of the system, which may be nudging up against or even starting to pierce the system’s Boundary of Acceptable Risk. (Rasmussen, 1997, pp. 189-191)

The old adage “where there’s smoke, there’s fire” is apt here: DevOpsCo’s artifacts serve as a mechanism to identify this ‘smoke’ and smaller ‘fires,’ and help put those signals into a proper context for the larger system (and its actors) to be able to notice them and react appropriately. This use was observed, for example, in the active updating and resetting of priorities regarding security threats:

We have a capability gaps document that we maintain and so we have quarterly planning for our team. ... So basically quarterly, I’ll go through that capability gaps document and see “does this still make sense?” essentially. ... And the timing of the [quarterly] retrospective and all that is such that we hope [it] will influence at least the other security teams when they’re doing their quarterly planning, right? ... I think this is a very powerful part of the incident response / postmortem
processes is that “here are real incidents that we actually saw.” So having this data of like “here’s what actually happening” versus your sort of theoretical attacks, I think, is super-useful.

— SecMgr1

It was also observed in Operations around subsystem usage patterns which caused ‘smoke’:

We found one [pattern]; it didn’t bite us very often. ... It happened just rarely enough that you kind of forgot, because it manifested in this interesting way. ... We found this underlying pattern that had been communicated at some point in the past. It probably felt like a good idea, but it didn’t turn out very well. ... The people that had developed the pattern and trained on it were from [System Layer A] and either the communication either via code or examples, they weren’t treating [System Layer B] very well. ... Under a failure scenario where [Layer B] itself is having problems, you make them worse. Because it’s already under duress. It’s already told you “I don’t have it. Please go away.” And, some number of milliseconds you come back later on and you’re shoving values down its throat. But it wasn’t happening enough that we said “we really have a [Layer B] problem.” And it took some time time and starting to look for certain kinds of patterns where that pattern of behavior bubbled up and we were able to address it.

— OpsEng1

Notably, these two processes happen both at a technology level—software-software and software-infrastructure linkages are identified, as in the example above—but also at a social level, i.e. teams that are not interfacing with each other, but who need to be, or where context may need to be (re-)set between two teams to achieve a certain operational outcome or re-prioritize work (and prompt a discussion around these issues).

For example, in a security context, an incident revealed situations where teams in two different locales, given the right type of security incident, would require each other’s assistance in order to achieve a successful outcome:

So say we worked on an incident and saw “Hey, so we don’t have the really strong relationships maybe with these overseas folks,” like they don’t know which [e-mail] distribution lists to go [to]. And even within a specific team maybe, who are here, closer physical proximity, maybe we [need to] work
with them closely, so maybe other members of the team don’t know as much. ... That definitely came out of that [incident].

— SecProgramMgr1

In another example, a security engineer made a change which improved DevOpsCo’s security posture, but also resulted in an incident. This revealed both a class of technical ‘hot-spot’ (i.e. the lack of metrics around software library usage throughout the organization), but also prompted that team to start discussions with teams still using the deprecated software library (i.e. an unknown social linkage):

I think the biggest takeaway from this incident was we just don’t have enough knowledge. I literally did not think that would cause a problem. It was that one bizarre use case that no one knew about that. Super, super weird use case. I don’t even think the team responsible for it... even realized that. “Oh yeah, that’s right. We’re using that.” So now we know... But that drove requirements: we need better metrics. ... Or maybe that triggers maybe we need to approach those teams and find out “why is this being used,” right? So today, actually. I was looking at this and saying “I wonder if there are people who are using this old version of the library” and I have metrics now to collect that info and I was able to say “Oh, they are.” So we need to reach out to those teams [to understand why].

— SecEng2

One of the operations engineers, responsible for shepherding incident reviews and helping drive the incident analysis process summed this social emergent mapping process up as:

This sort of analysis starts conversations between teams. And given the impact of the ticket and the services that were affected, talking about the upstream and downstream relationship between these services because some services depend on others, starting conversation between those teams is sort of a, like the social component that comes out this instead of the technological one.

— OpsEng2
5.3.2 Context-Sharing Over Remediation Action Items

With respect to how the software development and operations industry uses its post-incident analysis artifacts, it is clear from the survey that a main goal is remediation: 91% of survey respondents said collecting remediation items was a core purpose of the artifact and “preventing recurrence” was a top theme of the collected incident analysis templates. But DevOpsCo does not focus or even place a particularly high value on, specifically, the generation and recording of remediation items.

That is not to say DevOpsCo does not create remediation items in response to incidents, nor that remediation activities do not take place. Quite the contrary: numerous examples of such remediation tickets were provided. But the primary goal of post-incident analysis activities is not to generate those remediation items: it is to further understand the incident and understand the organization’s responses and reactions to how the incident unfolded:

So specific to security incidents, again, we have templates, we have [meeting] agendas. When we run [an incident analysis meeting], we focus on the response procedures initially. So we don’t want to get into... we don’t want to rat-hole on engineering solutions or security controls. We’re primarily focused on process improvements for how we could’ve responded better.

— SecMgr1

The program manager responsible for evolving DevOpsCo’s security response echoed this sentiment:

So we don’t necessarily discuss solutions in postmortem. We try not to discuss solutions at all. The focus is on the specific problem and how the handling of the problem went.

— SecProgramMgr1

The Operations team responsible for leading non-security incident analysis describes a similar focus, and remediation items are not a conscious, front-of-mind goal of DevOpsCo’s incident review meetings:

If we feel we don’t have a sufficient understanding or people’s understanding of the system was like totally blown out of proportion or something then we would definitely want to have like an in-person incident review to go over.
Thinking through the timeline of events and getting the context of what each person was thinking, how they were feeling during the incident, and how they responded to the various things that happened.

— OpsEng2

A second component of DevOpsCo’s distinct utilization of artifacts and de-emphasis of remediation items is hinted at in the above quotation: the use of context—the richer, the better—as a vehicle to exchange information between sharp-end actors, and pass information up to the ‘blunt end’ about the state of the system and trends impacting it. This practice was not only evident at different levels (manager versus individual contributor) within the organization, but also observed in each operational department. DevOpsCo’s reliance on context-sharing as a way to align people and teams was so pervasive, references to the practice, specifically, were observed in every single interview.

The reliance on context-seeking and context-sharing was observed in the Operations team, when the team was trying to understand why a series of incidents seemed to be occurring more often in a service owned by another team, and trying to ascertain whether a trend was forming:

[The development team we were talking to] did launch like a brand new service. And so with that comes a learning curve of bumps of innovation. So the next step was not to do much conclusion-making and [rather] have a conversation [with them] and give feedback and just wait for them to help us, give us context to their thinkings. [Incident data is] the only way you can start that conversation because otherwise you’re just like “Mmmm, yeah. I think the wind’s blowing that way now.” Yeah, [context] was the basis of that conversation.

— OpsEng3

Context-sharing influences how DevOpsCo determines what work is important and also serves as the mechanism by which it goes about planning that work:

But one of the things that’s helpful for incident reports is often times, I have to, I hate to say justify, but kind of share context with my partners to say “It’s important for us to work on this infrastructure versus working on your feature and here’s why.” … I use that data to help and analyze the impact of a project, which informs the urgency and priority of the project. So it’s
useful for me in terms of not only internally identifying areas where we need
to improve, but helping us to negotiate with our [internal] partners that, you
know, “do you agree we should spend time on this versus building this new
feature that you want us to do?”

— DevMgr1

The Security team uses context-sharing to help developers make better, more
up-to-date security-related decisions and trade-offs, without the security team it-
self becoming a bottleneck:

[DevOpsCo’s] sort of distributed vision of how [individual contributors] code
is they just go and do their thing. So it’s all about providing them context. So
yeah, we provide them context through that quarterly [security] retrospec-
tive, through the writeup they can read and then it’s through direct feedback.
... So they’ll obviously want to share and talk about their projects and I’ll
provide feedback on whether I think that’s aligned with where we should be.
You know, at the end, it’s still up to them to choose what they want to work
on. But I think I have a role of sort of the conscience that sort of tries to bring
it back to what we actually saw [in security incidents] last quarter versus
what we might see in the world.

— SecMgr1

In fact, context-sharing supersedes the use of (and management-by) remedia-
tion items to such an extent, that it is within the developers’ purview to decide not
to complete incident remediation items that have been suggested, discussed, or even
‘agreed to.’ This is an important departure from the industry practice commonly
observed in incident management and remediation:

Every team has the freedom to make that choice. I might choose to have
more outages if I feel if I can remediate quickly enough. I’m willing to have
an outage because I’m prioritizing all the other work I have to do. So that
actually happens: “This quarter, I’m not going to fix these problems. I’m just
gonna let it happen again and we’re going to have a little outage and we’ll
patch it up like we always do and that’s fine for my team.” So trend analysis
or not, I still will always have the freedom to not do anything about it, at least not this quarter. Maybe next quarter, hopefully I'll get to it later. ... We're just going to have a few outages once in awhile, and that's properly prioritized, given all the other priorities of the company.

— Dev1

Of course, this attitude that ‘outages are acceptable, given other priorities’ may imply it is culturally acceptable to cause customer-impacting outages. But there exists an interesting tension between the Operations team responsible for managing incidents and the Development teams doing the work that serves to counterbalance this. Perhaps unsurprisingly, the method DevOpsCo uses to ensure that tension does not become too taut? Providing additional context:

For instance, if we see a team that's struggling to either prioritize correctly or to get kind of like reliability worked on and tech debt and features or if they're being, you know, feature/reliability pulled, sometimes we'll invite their management chain into an [incident review] meeting or sync meeting just so make sure they're aware of the full context of “this is the situation of the team. They are all under water. This is the outcome of that, that you may have not seen. I know you see you have stressed people..., but they're not getting these things done and they're struggling in these ways and they're responsible for some really, really key stuff. All the indicators are this can't continue.” And the way DevOpsCo is structured, that is your responsibility, Team Manager or Area Director, to get that team the resources or the prioritization or load shedding they need so that they're not impacting our customers.

— OpsEng1

At first blush, this all may seem counterintuitive, especially given DevOpsCo’s industry reputation: how exactly do they accomplish that if developers can completely ignore remediation items, and all the Operations and Security teams can do in response is ‘share more context?’ One of the Operations engineers explained it thusly:

It’s kind of strategic accountability more than tactical accountability.

— OpsEng1

66
In other words, DevOpsCo is able to achieve its industry-envied operational success because its focus is on the goal of outcomes which constitute operational success, not the specific remediation items that may fall out of an incident analysis event. There exists a strong sense of shared alignment—“We’re highly aligned, loosely coupled,” as SecEng2 put it—across the organization (which itself is created and maintained by context-sharing), and so the different groups establish accountability by communicating, discussing, and potentially even debating their localized contexts (“local rationality” in human factors parlance [Dekker, 2006, pp. 11-13]) in lieu of—and this is the critical difference—the tracking and systematic ‘checking off’ of a static list of remediation items. This allows for a continuous cost/benefit analysis to occur, letting the organization revise its estimates of whether incident-remediation work is, in fact, worth scheduling and assigning resources to.

The Operations team’s framing of accountability described above may imply that those ideas around accountability are not shared across the organization. This mismatch in expectations and values often leads to a ‘siloining effect’ among software development and operations teams. That is not the case at DevOpsCo: the framing of accountability, who is accountable for what, and how to discuss those issues was remarkably strong and aligned across teams; a developer explained it as:

We use the [incident] artifacts to kind of influence our future work. It’s almost like a contract between what we plan on doing as a remediation and, when I say it’s a contract, if for instance the team doesn’t act on those next steps accordingly, the rest of the organization can look at that incident artifact and say “Hey, why aren’t you guys acting on this?” It’s a kind of way to reflect on what happened, identify what we think needs to be done, to fix it, and follow up on making those changes. And contracts like that are, I think are important, especially in organizations where you’re growing and you need to communicate and communicate clearly what’s going to happen. And if someone isn’t doing what they’re supposed to be doing, you can kind of gently nudge them with saying “Hey, this is the incident, how’s the progress coming on these remediation steps?”

— Dev3

---

8Which, at DevOpsCo, is very narrowly and specifically defined as part of its ‘culture’; in fact, it is clearly defined as a set of operational metrics which can be counted on less than one hand.
5.3.3 Curating Organizational Culture and Transmitting Tribal Knowledge

The third major use DevOpsCo makes of their post-incident artifacts is the cura-
tion of organizational culture and the transmission of operational “tribal knowl-
edge.”

Discussions of “culture” in an academic context are often bemoaned because
the word can refer to many different aspects of human experience, holds many po-
tential meanings, and “as a concept, has had a long and checkered history.” (Schein,
2010, p. 13) This research does not attempt to involve itself in the nuances of that
discussion; for our purposes, Schein’s formal definition of organizational culture
handily suffices: “The culture of a group can... be defined as a pattern of shared
basic assumptions learned by a group as it solved its problems of external adap-
tation and internal integration, which has worked well enough to be considered
valid and, therefore, to be taught to new members as the correct way to perceive,
think, and feel in relation to those problems.” (Schein, 2010, p. 18)

Put another way, culture in this discussion is narrowly bounded to describe
work performed within an analytical, planning, design, execution, and operational
context pertaining to one organization. One way to frame this bounded concep-
tion is a “This is how we do things here”-ethos or a shared understanding that
given certain inputs, there exist expected outputs which are ‘obvious’ within the
organization, due to a shared sense of connection, i.e. a ‘tribe,’ that tribe’s associ-
ated knowledge and a ‘cultural context.’ Furthermore, this conception of culture
includes where such outputs are not ‘obvious,’ the process for resolving them—so
the tribe (and thus the larger organization) can make progress in the performance
of work—is understood.

Schein further defines three ‘levels’ of culture, all of which serve to fill out the
relevant framing of “organizational culture” here, and all of which were observed
during the Phase II organizational case study:

1. **Artifacts** – These are “phenomena that you would see, hear, and feel” (Schein,
2010, p. 23); given this research is explicitly examining the use of post-
incident artifacts, numerous examples of these “phenomena” within the or-
ganization were observed. These, of course, were supplemented by the time
spent on the organization’s campus, in the organization’s ‘natural habitat.’
Artifacts other than the reports provided by the informants were especially
noticed in ‘informal moments,’ i.e. walking by engineers engaged in conver-
sation, lunch in the cafeteria, etc.
2. **Espoused Beliefs and Values** – Schien notes “If... you want to achieve [a clearer] level of understanding [of the meanings of the culture’s artifacts] more quickly, you must talk to insiders to analyze the espoused values, norms and rules that provide the day-to-day operating principles by which the members of the group guide their behavior,” (Schein, 2010, p. 25); the structured interviews and follow-up questions represented an exercise in obtaining precisely this higher order understanding of DevOpsCo’s organizational culture.

3. **Basic Underlying Assumptions** – “When a solution to a problem works repeatedly, it comes to be taken for granted. ... Basic assumptions, in the sense defined here, have become so taken for granted that you find little variation within a social unit.”9 (Schein, 2010, pp. 27-28) A number of DevOpsCo’s basic operating assumptions are cataloged below, elicited largely through the structured interviews. Interestingly, as Schein describes, these assertions were either discussed as if they were to be ‘taken-for-granted,’ or as an explicit piece of contextual knowledge which must be communicated to new members as part of their induction into the tribe, which is “in fact, a good way to discover some of the elements of an organizational culture.” (Schein, 2010, p. 19)

A working definition of tribal knowledge, interestingly, proves a bit more elusive: references to the concept are sparse outside of the ethnographic study of indigenous peoples.10 Despite this, the term has been defined in a couple of fields, which can be leveraged:

1. **Management sciences**, especially Lean Manufacturing - In these forums, “tribal knowledge” is generally defined as “special knowledge procured through experience by only a handful of employees. These employees are usually senior personnel in the quality, maintenance, or control department in the organization, who have acquired expertise on a equipment, system, or process over an extended period of time....” (Lin, Mehran, Badar, Foster, & Dean, 2016, p. 3) The focus in this field seems to be on researching methods to record and transmit that tribal knowledge, as it is generally considered “critical for the company’s sustainable growth.”11 (Lin et al., 2016, p. 3)

---

9Schein notes other researchers, notably Argyris and Schön, have referred to this concept as “theories-in-use” in their discussions of “double-loop learning.” See Schein, 2010, p. 28.

10Wherein the “tribal knowledge” being referred to is, literally, the knowledge of a tribe of peoples.

11See Lin et al., 2016 and Peter, Shinn, and Fleener, 2011 for illustrative examples.
2. **Air traffic management** - In a field well-known to the safety sciences, researchers attempting to improve communications between dispatchers and air traffic controllers regarding flight routing presented a set of tools intended to represent and then surface tribal knowledge about specific airspace constraints in the national airspace system.

For their purposes, they defined tribal knowledge as “the traffic manager’s accumulated knowledge of routine airspace constraints and traffic patterns or flows.” (Spencer, Smith, Wilmouth, Klopfenstein, & Sud, 2005, p. 1) Interestingly, they note that it is not only the air traffic managers who hold tribal knowledge, but dispatchers as well: “Dispatchers have their own ‘tribal knowledge’ about the priorities and constraints associated with particular flights. Thus, we have outlined a procedure that provides dispatchers access to traffic managers’ ‘tribal knowledge’ in a usable form while also allowing them to communicate their preferences to the traffic manager.” (Spencer et al., 2005, p. 6)

In DevOpsCo’s case, an informant described their conception of tribal knowledge as:

> Just working in a company, in the culture, you understand the familiarity of the environment. And so, if there’s a bug in the environment or some error, it’s like “Oh, I know what that means. It means this problem happened.” Right? So it’s kind of like you can’t teach experience type of thing. It’s kind of like, it’s just experience, just in a particular environment, and you... if you see a particular error, you’ve, you’ve spent so much time working in that type of setting that you know how to identify that.

— OpsEng3

It is important to make clear a critical point: Schein notes “Observers can describe what they see and feel but cannot reconstruct from that alone what those things mean in the given group.” (Schein, 2010, p. 18) The examples below are not intended to capture and illustrate the full richness of DevOpsCo’s operational and organizational culture, nor its corpus of knowledge known among its internal tribes; to do so would be a thesis unto itself.12 Rather, the goal is to illustrate that DevOpsCo uses its post-incident analysis artifacts in the processes of curating13 its

---

12Or three.

organizational culture and communicating tribal knowledge between the tribes within it.

While the context-sharing discussed in section 5.3.2, as a vehicle, does play a role in the process described above, it is a distinct use of incident artifacts, because of the difference in both scope and form of expression of that use. The use of context-sharing represents a day-to-day activity, used to transmit information, priorities, and (when necessary) ‘litigate’ decisions. Alternatively, organizational culture and tribal knowledge, as described here, play an identity role: it is what DevOpsCo tells itself, both explicitly and tacitly, about who it is, why it is who it is, what it does, and why it does what it does. Unlike context-sharing, this process typically involves ‘storytelling’ as the mechanism of transmission, and therefore system actors become characters in a story that, to be effective, must be compelling, memorable, and structured as a narrative to make cognitive sense to other system-actors listening to those stories.

As an example of how embedded this practice of storytelling is, a developer described how they consider language, itself, used in these stories and their transmission as an artifact, and how storytelling and that language they’re built with are used to influence the technical architecture of the software developed at DevOpsCo:

Well, I mean it’s... the problem is [the language] becomes tribal knowledge. It’s in my head, I then share it with new employees. So I use that, y’know, when I talk to new employees, we re-enter problem, you know, if I’m working on the team and they’re like, I mean it just so could happen over and over again with like “I’m just going to call [your service] and it should never fail” and like “Da da da, let’s go back. No, no. I shouldn’t be able to affect you.” I might purposefully affect you just to force the issue at this point, you know, in some cases. Yeah, it’s... as an artifact, it’s loaded up into my brain and it’s used as a constant tool that I can use to help describe people on how they should architect their stuff. But it’s not written down. I don’t have an employee starts the first day I can say “please go read this doc.” I don’t even know where [that document] would possibly go.

— Dev1

Another distinct aspect: both organizational culture and tribal knowledge represent elements of the organization that are durable over time—that is, in the normal course of the organization’s work, they are resistant to drift, as tribe members may come and go and circumstances may change—even though the specifics of
why DevOpsCo ‘does it that way’ or values one aesthetic, behavior, value, object, or practice over another may fade away. Post-incident analysis artifacts contribute to this: teams dig up historical context to re-validate tribal knowledge or aspects of the culture.\textsuperscript{14}) Also distinct from the mechanism of ‘context-sharing’ described above: these ‘cultural’ stories are sometimes told outside the walls of DevOpsCo, in order to make clear what DevOpsCo values, who DevOpsCo ‘is,’ and what it would be like to work for DevOpsCo (which, in turn, functions as a recruiting tool).

One of the Operations engineers links these two processes together by describing how repeated sharing of “context” eventually turns into “tribal knowledge,” in this case around the (mis-)use of a platform feature:

\begin{quote}
It’s more that as we do future incident reviews, you should know that that’s a topic that you’re going to have to bring up or just keep talking about, so that we can kind of, I guess, informally start building a case for future teams to say “this is something we shouldn’t do.” And the belief is that if we just continue to verbalize and share like better practices, that it’ll start to percolate through the [organization], and case in point: this global configuration change. It did show up in this one incident. And we did bring it up again. And it happened to catch the ear of some director. And then some team that uses them a lot said “We will no longer use [these global configuration changes]. And I’m fairly confident that happened because it’s an accumulation of signals over like the last two, three months.
\end{quote}

— OpsEng3

The Security team, in particular, makes heavy use of the technique of storytelling to credibly\textsuperscript{15} create awareness around security aspects of DevOpsCo’s operational culture:

\begin{quote}
You want to make sure that [developers and operations engineers] have a realistic view of what’s really going on and that there really are bad people out there that want to do bad things to DevOpsCo, right? Just keep reminding
\end{quote}

\textsuperscript{14}See Dev4’s quotation later in this section for such an example. Schein describes this as a “transformation process,” noting that if it occurs, “group members will tend to forget that originally they were not sure and that the proposed course of action was at an earlier time just a proposal to be debated and confronted.” (Schein, 2010, p. 26)

\textsuperscript{15}SecMgr1 warned against a reliance on “FUD” to tell stories: “So you need to occasionally, without causing, like without... peddling fear, we call it FUD, right? Fear, uncertainty, and doubt. You don’t want to be FUD’ing your group...”
them of that. And to do that, these war stories and sorta pointers to the flow of incidents is helpful, because it grounds it in reality.

— SecMgr1

One developer uses the incident artifacts to find more accurate, engaging words and language to better promulgate DevOpsCo’s unique organizational culture:

So I sit in on [incident reviews] to learn about what the right wording, the right discussion, you know, how do I talk about this subject to somebody, especially when I have new people on the team and they come from another company where there was an [Operations] team and they didn’t have this DevOpsCo philosophy and they have get this philosophy. So I want to know what the right words are to describe these things... you should not be affected by downstream services. You should take these precautions. You should have fallbacks. Like these are repeated, these lessons come up over and over again. And so for someone who’s never heard of it, these are new things for them. For someone who has heard it, [the incident artifacts] help me understand more effective wording [to get them to buy into it].

— Dev1

On the Operations team, one of the engineers echoed this particular non-technical use of incident artifacts:

Given the impact of the ticket and the services that were affected, talking about the upstream and downstream relationship between these services, because some services depend on others... starting conversations between those teams is sort of a... like the social component that comes out of this instead of the technological one.

— OpsEng2

One concrete example of organizational culture informing operational practice is the story the organization tells itself about what times deployments are socially acceptable within the organization.16

16This example was particularly interesting, given DevOpsCo’s repeated public position that developers are empowered to deploy ‘whenever they want,’ which is technically true, but carries more nuance than such a blanket statement would imply.
There was a recent engineering all hands where one of our members showed a kind of a histogram of when incidents happen. And it’s remarkable: they usually happen during business hours. They usually don’t happen at night. Part of that is, like, we deal with incidents more immediately during the day, just kind of like scratch that itch. And at night we’re more, we basically have a stronger test for “Should I wake up and rally the teams to work it.”

— Dev3

This tacit knowledge around acceptable deployment windows has become such a part of the operational culture at DevOpsCo, it is encoded within the incident artifacts, and has made its way into the secondary artifacts and stories teams use to acclimate new hires to ‘the DevOpsCo Way’:

I think a lot of it was just kind of shadowing other people. Looking at the [Operations chat] channel, seeing what incidents are going on, how people are reacting to it, how they’re doing their [incident tickets], kind of just viewing how that works. And then, because I did that, I looked at certain [incident tickets] when I first started, seeing how that works and asking questions like “Hey, so I noticed this. Like when’s the right time to deploy or not deploy something,” right? And that actually drive “OK, should we create documentation for a new hire?”

— SecEng2

Incident artifacts also support the organization’s ability to curate its stories and tribal knowledge by providing a lasting, context-sensitive reference point. Below, a developer talks of trying to find “more context around” an engineering decision made long ago. But unlike the discussion above of the context-sharing mechanism, this example is one of curating and potentially updating (or, as in this case, reconfirming) tribal knowledge known to the team; here, it is an engineering requirement tacitly-known among developers for why a particular design decision had been made. Incident artifacts are referred back to, to reconfirm this decision and validate the story, thus reinforcing the team’s tribal knowledge (and, in this example, communicating that knowledge to a new tribe member):

There was some confusion about the rationale, or the rationale for doing this thing had been lost, even though [the project] was kind of actively under development, it was something that was particularly difficult. So a new
manager came in on that team and was asking “Why is it important that we are doing this feature?” And then I dug up that old [incident] document to get more context around that.

— Dev4

It should be noted: this curation process can also disconfirm current working theories, providing another way to update tribal knowledge, as was OpsEng3’s example above involving “global configuration changes.”

A final example of a use of incident artifacts to communicate tribal knowledge was in a presentation a team put together (emphasis added):

This [presentation] was about identifying, not necessarily the causes of incidents, but the hazards that have been present during all of our incidents.

— OpsEng2

There is an interesting, yet subtle distinction about the presentation’s focus: it was not about incident cause(s), but rather centered on common hazards present during memorable, impactful incidents.

In this way, this example represents a search of that “discretionary space” Dekker describes (Dekker, 2012, pp. 80-82) existing within the Rasmussan triangle (Rasmussen, 1997, pp. 189-191), looking specifically for common edges of the Boundary of Acceptable Risk, not limiting that exploration to a cognitively-constructed, linear model of [root] cause[s] that ‘caused the incident,’ and which could be ‘fixed’ by a static list of ‘remediation items.’ This knowledge was then packaged into a story the organization told itself in an internal engineering presentation, in order to communicate to system actors the sense of ‘if you design your systems in these ways, with these elements and patterns, we’ve observed that those elements tend to be present in incidents we experience.’ In other words, relaying to the rest of the ‘villagers’: there be dragons here! This, then, allows teams to (consciously) consider and potentially refactor their designs and operational patterns, thus ‘steering them away’ from the Boundary of Acceptable Risk, potentially in ways they, themselves, were not aware of, but other system actors are. This represents a transmission of one tribe’s knowledge to the larger, ‘company tribe.’ Repeat these stories enough, and they become part of the operational culture of the organization, as they did at DevOpsCo.17

17 One of the hazards identified was the use of “global configuration changes,” as described by OpsEng3 above.
One of the Operations engineer summarized this ‘tribal knowledge’ and storytelling use as:

*I think it starts with our team and ends up in DevOpsCo as a larger organization is trying to identify patterns that maybe in the minute we miss, but as a group or the kind of the, you know, the gestalt of the knowledge that comes out of these incidents, we can find certain patterns that DevOpsCo, as an organization, needs to address.*

— OpsEng1

### 5.4 Other Incident Artifact Uses

Over the course of the case study research, other interesting uses of post-incident artifacts were observed, including as a tool to communicate with external partners and manage expectations, and the creation of additional artifacts, derived from the direct artifacts related to incidents and incident management.

#### 5.4.1 Liaising With External Partners

As discussed above, one of the uses DevOpsCo relies on its artifacts for is the mapping of its complex socio-technical system, specifically of connections which are latent or newly formed.

This use actually extends beyond the organizational boundary, into partner interactions. Two examples were provided by informants. In both cases, they represent connections or system ‘hot-spots’ in other partners’ infrastructure and operations, and they used incident artifacts to liaise with partners regarding these issues to drive to an improved operational posture for the partner, and thus ultimately for DevOpsCo and its customers.

In the first example, DevOpsCo observed a repeated issue with a particular device. DevOpsCo collected data on the issue and the device’s behavior and used that data to share context with the partner and drive a solution, both from a technical and business perspective:

*We have some dependencies on outside partners that sometime causes a ripple through our systems.... Now there’s lots of things we’ve built over time so that impacts our customers as little as possible, but we still keep track of those. We keep an eye on them, because it’s good data for us to have later on. So what we did is we watched the impact from that outside service, and...*
just gathered all the information. ... The advantage being, we went to our [internal partner] engagement manager and said “Please show them this.” ... So we do have some incidents like that, that are important for us to record for other reasons, not necessarily to drive systemic change or engineering change, but because we need the data to drive some other process.

— OpsEng1

In another instance, a series of DevOpsCo incidents prompted them to build a set of monitoring dashboards, which in turn drove a conversation with a partner that resulted in the identification (and remediation) of a ‘hot-spot’ in the partner’s systems which had not yet surfaced to their level of general visibility, but which DevOpsCo was able to observe because of DevOpsCo’s specific usage patterns:

So, this dashboard came out of a series of incidents that weren’t sufficiently able to detect this high rate of... errors. The incident I’ve been referring to, we created graphs that were detecting what [the partner] called “bad hosts.” Once we created the dashboard, we realized there were other regions exhibiting a similar problem to a lesser degree, which we were able to show [our partner] before they became a larger problem.

— OpsEng3

Suffice it to say, the ways in which DevOpsCo uses incident artifacts—specifically for mapping the system terrain for unknown connections and dependencies and system ‘hot-spots’ and transmitting this information via context-sharing—can extend beyond the walls of the organization and directly drive improvements that benefit not only partners, but DevOpsCo itself.

5.4.2 Secondary and Tertiary Artifacts

Another notable use of the incident artifacts were to derive what could be called ‘secondary’ (or even ‘tertiary’) artifacts, that is artifacts that are based entirely on incident artifacts and related to incident remediation, analysis, or management, but which are not directly connected to a specific incident. These are slightly different from the uses described above, in that the end result is an additional artifact (i.e. a report, presentation, etc.) that can be observed unto itself.

DevOpsCo’s informants offered a quite a few examples; one was in communication to executives:
This data gets used outside of [the ticket tracking system], gets pulled into availability reports. It bubbles its way all the way up to some of our executive reports. So this is our first big step in clarity of communication.

— OpsEng1

Past incident data is also distilled and analyzed in real time and used to provide immediate operational context (as described previously) to developers making deployment decisions (in this way, it also reinforces ‘operational culture,’ specifically around acceptable deployment times, as previously discussed):

Our availability numbers show up in different places. So... our deployment tool, and again, I said earlier, we don’t build gates. We don’t say “You can’t deploy at certain times, you can’t do those kinds of things.” But we do want to provide context for decision-making, so we have things like DevOpsCo’s current availability is always up here in the corner [of the deployment tool], as a red, yellow, green kind of report. Because somebody may want to... “I have a risky change.” Here’s what we done to our customers as of late. Is now the time to do that risky change?

— OpsEng1

One team looked at incident artifacts for incidents it had been involved in to see if there were patterns around the health of the people on the team:

I saw a recent team’s initiative to create an application that would be able to use signals that come out of our incidents as a gauge of team health. So they would want to track things like how often are engineers getting paged? What’s the amount of time that they spend on tickets? What are the amount of alerts that resulted in the ticket that the engineers had to respond to, those sorts of things.

— OpsEng2

The creation and specification of heuristics to describe what a ‘healthy’ engineer ‘experience’ looks like at DevOpsCo represents a secondary artifact.

In a security context, incident artifacts can feed into the creation of additional artifacts in a legal context:
So let’s assume there was an investigation and now law enforcement was brought in. ... So eventually a request will then come in to Legal and then Legal will reach out to us. ... Usually Legal will say this is the information that is being requested, this is what Detective Smith is asking for, this is what it’s in the subpoena, and we’ll go and actually get that data and then hand it back to Legal, and let them do what they need to do. Now that’s what usually happens, like post-incident.

— SecEng1

These represent just a few of the myriad examples informants provided on how they analyze, ‘remix,’ and reshape data embedded in post-incident artifacts to improve their exploration of the discretionary space, decision-making, and ultimately operational outcomes. While many examples were provided, by all three operational areas, one of the Operations engineers responsible for coordinating and facilitating incident management and analysis summed up this phenomena of secondary and tertiary artifacts best:

I would guess that there are probably quite a few artifacts that come out of incidents that we’re never aware of.

— OpsEng1

5.5 Other Findings

Finally, two last findings noteworthy of inclusion have not so much to do with DevOpsCo’s use of its incident artifacts, but rather are observations around mechanisms and organizational ‘framings’ that facilitate their artifact use, make their use sustainable, and thus make their overall consumption and digestion of these various post-incident artifacts more successful.

5.5.1 ‘Blameless’ is Table Stakes

There is a fair amount of discussion in the technology industry (and, indeed, other ‘high tempo, high consequence’ industries ) regarding ‘blamelessness’ as it relates to incident management, analysis, and postmortems. Woods notes “The rubric surrounding these topics [of ‘blamelessness,’ ‘accountability,’ and ‘just culture’] can be difficult to parse.” (Woods, 2017, p. 27)

18 Also notably, Woods raises the issue of the conflation of “blameless” and “sanctionless,” namely that many organizations who consider themselves “blameless” are, in fact, “sanctionless,” to the extent that is even possible in their own contexts: “As a practical matter, it is difficult to forego sanctions entirely,” Woods notes. (Woods, 2017, p. 27)
A deep-dive discussion into these issues and their impact on organizational culture and outcomes is beyond the scope of this research. That said, what is worth noting is the pervasiveness of this “blameless” ethos around how DevOpsCo parses and acts upon post-incident artifacts and how it engages with itself. What is important in the context of this research is not so much that we understand what “blameless” means at DevOpsCo, but rather that they understand the use of the word, its meaning, its associated behaviors in practice, and as an operational lens and cultural framing, they value (their version of) the concept.

In this regard, the frequency with which it appeared in the structured interviews, across all of the informant teams and at each level of the organization was palpable:

One thing I very much appreciate about troubleshooting [at DevOpsCo] is that it’s not about blame and saying “Well, what was the system, what was the server, what was the team that messed something up?” But rather “here are the different things we saw.” … And that’s interesting, especially if there very obviously is a specific culprit system or something that I’m just glad that it doesn’t become a blame-fest, you know? It keeps it efficient.

— Dev2

It’s not about blame, you know? We had an incident this past weekend where it was a confluence of two errors. One engineer didn’t know what the other engineer was doing. He did something, it caused a problem. But when we came into the office on Monday morning, I mean, it wasn’t about blame. It wasn’t about, you know, “you did this, you did that,” it was like “this happened and this happened....”

— DevMgr1

The tone that I’ve always tried to take is one of sort of humility and learning; there’s this sort of dark side of security culture where it’s all about like “you guys are all idiots and I’m a genius” and I’ve seen that a lot in organizations. And it builds up over time, right? Because as a security organization, you

---

19 And there exist many great existing treatments of the [discourse on the] topic: see Woods, 2017 and Dekker, 2012, among others.
see a bunch of dumb, like what you think is dumb stuff. It’s obvious to you, but like as a non-security specialist, it’s not obvious. So then you build up these sort of antagonistic relationships, which fortunately we don’t have here at all.

— SecMgr1

Even though specific team members may blame themselves or assume responsibility for an incident, it is not generally a function of the incident review process (or encoded this way in the analysis artifacts):

[Going to that first incident review] was humiliating. Because no one wants to cause a service outage, right? Like, no one got shamed or anything, but that incident was mentioned and this was the impact.

— SecEng2

Again, stewardship of this process rests with the Operations team, who take this “blameless” mandate seriously too:

Incident reviews are not intended to be spectacle. We don’t want to invite a thousand engineers to come and tell you “You made a mistake!” So they are never intended to be spectacle.

— OpsEng1

In the midst of [reviewing incidents and creating artifacts], we strive to achieve a blameless culture. So these write-ups are intentionally avoiding saying “it came down to this guy screwing up.”

— OpsEng3

This “blameless” ethos extends up to the executive level:

We’re trusted to handle it. [Executives] would like to know the outcome. But never have we had a “well, these four [vice presidents] want to show up and make sure that they glare at the right people.” It doesn’t happen.

— OpsEng1
Ultimately, the importance of a constant undertone of “blamelessness,” contextualized to DevOpsCo’s practices and operational culture, underlie the success of both their incident analysis and management processes, and facilitate further use of their post-incident artifacts. Other organizations who do not value “blamelessness” or who do not confront the pressing questions of what a ‘just culture’ means in their own context will have difficulty reproducing DevOpsCo results or evolving the use of their own incident artifacts.

Put another way: to effectively use the information locked within post-incident artifacts to its fullest, “blamelessness” is a minimum bar. It is table stakes.

5.5.2 Living Processes and Artifacts

A final observation on DevOpsCo’s incident analysis processes and use of its artifacts: remediation activities resulting from an incident often involve process changes; this is common among companies who analyze their incidents and who consider action on that analysis to be important.

There are two notable facets about how DevOpsCo does this:

1. The first is the mechanism by which this can occur: in some cases, it is prompted by new team members joining the organization. In the initial explanations of “tribal knowledge” and organizational culture, questions from these new actors prompt the team to review and potentially modify its behavior; the newest Operations engineer on the team described this:

   No one really gave a sufficient answer as to why we do this other than “we think other teams are using this data.” So we track the different teams that are involved in an incident in this field and ultimately, we got rid of this field because we didn’t find that anyone was actually using it. And there was an argument that we wanted to associate customer impact to the different teams that were affected in the incident. And we have other, better ‘mechanisms’ of doing that.

   — OpsEng2

2. The second aspect is the rate-of-change by which DevOpsCo evolves both the process and the artifacts (and artifact templates) themselves. One of the operations engineers described a recent large evolution of their process early on in the organizational study:
We probably started playing with the [after incident report], I would say, a year ago, and goofed around a little bit and we had to start doing a few things differently inside the team so that we’d be successful. And we did a few tests. I’d say we’ve probably been more consistent within the last six months or so. That’s when they’ve really taken off.

— OpsEng1

But even in the three month span spent observing DevOpsCo, they undertook a number of experiments with formatting and other elements of their artifacts:

[If] all the sudden, now we’re generating five or ten of those [reports] a week because of the incident-review-self-service kind of stuff [we’re starting], and it’s taking off and becoming really popular within the organization, we thought we might need another way to kind of curate that a bit and make kind of bite-sized, impactful, larger organization learnings out of it, now that there’s more data coming in. We’re actually just doing an internal experiment today to try a couple of different formats, because we’re not exactly sure what’s going to work.

— OpsEng1

This evolution-mindset extends to the processes generating and further-utilizing those artifacts, too:

So we’re actually, today, we have a test run through of a new thing that we’re calling an IRL: Incidents, Retrospectives, and Learnings. And the idea is we would take these really kind of the meaty things and the interesting things that come out and have a presentation format that’s like three minutes on this one that would be more widely attended by the organization. And the goal there is not to do the architectural reviews…, but you know “we learned this out of the outage or this happening or this event and we think it’s important enough that the rest of the organization hears about it.”

— OpsEng1

It is interesting to note that the push to evolve these processes and the structure of the artifacts are all in service of continuing to improve the use of artifacts to highlight system connections and ‘hot-spots’ while creating create a map of their
complex socio-technical system, and curate and modify ‘tribal knowledge’ and operational culture, all via the mechanisms of high-value storytelling and the sharing of fluid, realtime context over static lists of remediation items created in the wake of an incident.
Chapter 6

Discussion

6.1 Too Good to Be True?

Today’s software development and operations companies are more interested than ever in “organizational learning,” its role and mechanisms, and potential methods to better promote and facilitate it.

Though they may not consciously conceive of it through this term and though the focus may be primarily on a ‘short game’ of improved operational outcomes, the amount of money larger technology companies throw at this problem through Incident Management, ‘Problem Management,’ and Business Continuity efforts (often staffed by entire teams) belies that this—the improved operational outcomes, at least—is front-of-mind for technology businesses. And the aforementioned increasing momentum behind “blameless” “postmortems” in the Internet operations zeitgeist illustrates engineers at companies of all sizes are starting to discuss it, if only notionally.1

This, in and of itself, is not particularly surprising, given the similar focus on ‘safety’ in the more traditional industries under examination by the safety sciences.2 Were the technology industry generally more cognizant of its own history and were it to place more value than it tends to on introspective reflection on that history,3 we might find ourselves less enamored with the potentials of “organizational learning” and further down the bumpy road of grappling with the tough issues it confronts us with.4

---

1See section 1.1.
2Road, aviation, and maritime transportation, chemical and nuclear plants, healthcare, etc.
3As celebrated computer scientist Alan Kay famously put it in an 2012 interview: “The lack of interest, the disdain for history is what makes computing not-quite-a-field.” (Binstock, 2012)
4A road we would fare far better traveling, were we to consult those more “traditional industries” for a map of the potholes they encountered while driving on it over the past forty years...
What has changed over the past twenty years in the technology industry is:

- Initially, the addition of a software operations function, that is not only the development of bits of software, but now the responsibility to operate it for customers, with some level of reliability, availability, and performance;

- Secondly, the rise of the idea of performing this task at “web-scale” or “cloud-scale”;

- And finally, performing this task, at that scale, *for services that matter*; services that interconnect with each other in ways we have trouble predicting; and services which are woven into the fabric of our lives—and increasingly society—often in ways we can’t always predict and sometimes do not expect.

Put another way: our industry is responsible for more of the operational foundations of society than just serving up ‘animated cat GIFs’ or selling pet food.⁵

Thus, the initial musings around the research question were stirred up not only because of increasing industrial interest around the topic, and not only because the questions we are asking and the answers we are searching for are having an increasingly palpable (if somewhat opaque) impact on our daily lives, but also because we do not have a very deep corpus of case study work on how organizations actually grapple, day-to-day, incident-to-incident, with operational issues.

This is likely for all the intuitive reasons around public, vulnerable sharing of incident and outage ‘dirty laundry’ being difficult. But there is also a challenge that the material we do have is often in the form of conference presentations, which present the challenge that, by virtue of the medium, they cannot be particularly comprehensive due to time constraints, are often presented by people within the same organization,⁶ and sometimes, a good story is better than a boring—or painful—reality.⁷

So, it was very much a desire to dig deeper into the whole question around what we, as an industry, tell ourselves about how we use these “post-incident event analysis artifacts,” all of these graphs and reports and work-tracking system tickets we laboriously and meticulously create; and then to go one step further and “walk

---

⁵To use summon venerable example of a ‘trivial’ service the Internet “revolutionized”; similarly, pets.com has served as a punchline for many-a-joke about the dubious “disruptions” many companies in the first dot-com bubble ‘pioneered’; see Abelson, 2000.

⁶This is not to imply that people telling their stories are lying, but rather they have a unique perspective, contextualized to their environment, which can sometimes make it difficult to see all of the ‘forest for the tree(s)...’

⁷This is to imply that sometimes, people exaggerate the efficacy of their incident analysis and organizational learning programs. Having personally spent time on the technology conference circuit where these topics are presented, I can personally attest that this is not a hypothetical...
the *Gemba* to see if, indeed, it is a story about a reality... or just good campfire fodder.

In that regard, running a pilot survey on industry incident artifact usage turned out to be a bit of fortuitous luck. While the original intent was to 'seed' questions for the case study structured interviews, gaining some insight into the 'macro-state' of the industry with regards to “postmortem” meetings, templates, and a hint of some of the practices, specifically through what information we choose to capture and report. Though unintentional, it provided a treasure chest of data to compare and contrast the case study and phenomenology-based Phase II research. And though I think the similarities and differences just between the high-level survey results and the deep-dive, rich, “thick descriptions” elicited through structured interviews with a living, breathing organization raise more questions than they answer, I am pleased to have been able to capture and present the harmony and dissonance between the two.

### 6.2 On Peering Inside a Nuanced, ‘Messy’ System

Any time an organization opens its doors and laundry hampers to external prying eyes, there is the concern that we are seeing a choreographed, rehearsed ‘version’ of the real system. Even though DevOpsCo touts transparency as a cultural value (at least internally), it is always hard to tell whether it is an espoused value or a lived one. This concern was heightened by the fact that DevOpsCo’s reputation, industrially, is so generally positive, there were a number of very real concerns about studying them:

- Readers (and myself, as an investigator) would just believe everything informants told us, at face value, unquestioningly and uncritically.

- Alternatively, readers (and myself, as an investigator) would react generally skeptically—“This is all too good to be true”—and the results of the research would amount to a literary critique of a book of fables, not an examination of real, sustained industrial practice, its required efforts, and its effects.

- Readers would read the accounts and quickly discount their value, under the reasoning that DevOpsCo is so unique, so special, that none of the findings

---

8 *Gemba* is a Japanese word for “the actual place”; the phrase comes from the creator of the Toyota Production System, Taiichi Ohno, and refers to literally walking through a plant to interact with (what we would call) sharp-end actors and see how work is actually performed.

9 As, per the survey, our industry overwhelmingly calls them.

10 Channeling my inner-Diane Vaughan...
or practices could ever be able to be implemented anywhere else, e.g. “Yeah, but that’s DevOpsCo; that would never work here!”

Having spent hours upon hours sifting through the mulch that is the organizational data, I can confidently state those concerns have been allayed, at least for this investigator: the data presented here may seem refined, so as to tell a story that justifies and accentuates the analytical findings; this is precisely true.

The raw data, however, like the socio-technical system it is derived from, is much messier, dirtier, and more nuanced than the themes presented herein. This might raise a disconcerting yellow flag, but on the contrary, I find it reassuring: the various informants I spoke with did not present an intentionally-uniform picture of the system, and in some cases, did not even present consistent answers. In cases where it was relevant, this presented opportunities to triangulate their answers with other informants, other organizational artifacts, or to ask them again after-the-fact in order to make deeper sense of their answers. Ultimately, this process yielded answers that were coherent (within their context) to an outside observer, and which demonstrated the challenges with honestly describing a system that, in fact, is not as clean, performant, stable, “unicorn-y,” or perfect as we would assume from the outside, both from a technology perspective and a people perspective.

As a great example of this: OpsEng3 said “DevOpsCo strive[s] to achieve blameless culture”; but multiple (different) informants discussed the ‘rough-and-tumble’ nature of team-retrospectives, where the bar for what is considered ‘blame’ might be a bit higher than at an organization-wide postmortem. Does this mean DevOpsCo isn’t “blameless?” There are many ways to interpret the data to answer that question, but I take it as “blameless” is not an entirely defined concept within DevOpsCo, nor is it context- and nuance-free in practice at DevOpsCo. But enough informants discussed the concept, and the overlaps of how they discussed it lead me to believe that DevOpsCo generally is “blameless” (or, at least, internally considers themselves, as a group, to be), and the observed differences are the result of the messy realities of existing within a complex socio-technical system.

I am also given confidence about the accuracy of the picture painted by DevOpsCo informants because they were forthright about areas they could not discuss, and in every case—there were not many—they were transparent in their reasons for why they would not discuss it. I found their reasons satisfying. (The cited reasons were all either operational or legal in nature.)

The DevOps community often refers to ‘unicorn companies,’ those who seemingly are magical in how they go about developing software and operating their systems; conceptually, it is intended as contrast to ‘the horses,’ that is ‘boring,’ traditional companies who struggle to achieve the outcomes accomplished seemingly so effortlessly by the ‘unicorns.’

---

11The DevOps community often refers to ‘unicorn companies,’ those who seemingly are magical in how they go about developing software and operating their systems; conceptually, it is intended as contrast to ‘the horses,’ that is ‘boring,’ traditional companies who struggle to achieve the outcomes accomplished seemingly so effortlessly by the ‘unicorns.’
Additionally, the way in which data saturation in the structured interviews was reached provided confidence that the signal I was observing from the system was strong enough to parse and analyze. About two-thirds through the panel of informants, certain themes were so unmistakably present, the avenues to walk down for further analysis were clear. The artifacts themselves, combined with these strong themes, created such a strong, consistent thematic foundation, I felt confident building the analysis presented herein upon it.

Finally, the observation of how DevOpsCo evolved their incident management and analysis processes, practices, and artifact templates (even over the course of the short few months it took to conduct the organizational case study) gave me confidence that the observed practices were not merely an elaborate production for this researcher. In follow-up interviews with informants, questions were asked about inconsistencies in the ways certain artifacts were filled out. The answer invariably was due to an evolving use of the templates. At the beginning of the research, the team described the completion of a program launch to push initial postmortem activities further down to sharp-end teams and actors ‘in the trenches.’ In follow-up interviews, the team was grappling with the results of this change (namely that they were receiving much more incident data from teams conducting their own postmortems, and were having trouble processing it effectively with current methods); one informant detailed the new ways they were piloting (that day, coincidentally) to try to address this new influx of (what they considered to be) useful organizational information, ripe for learning. Certainly were this research to be repeated with DevOpsCo, I would imagine many of the core themes to remain stable, but the expressions of them to be, potentially, different. In fact, certain practices already were wildly different when I returned for wrap-up interviews with the case study informants. The window into this process of evolution, itself, but also the informants’ transparency and description of it, provides further evidence that the findings herein are not the plot of some elaborate TV show.

For readers, my hope is the presented analysis allays similar classes of concerns, should they, too, harbor them. DevOpsCo’s practices, including the way they are evolved, represent an existence proof of a set of practices that promote organizational learning and improved operational outcomes within a software development and operations company, culminating in what the industry would classify as a “high performer.” (Forsgren et al., 2017, pp. 20-28) It does not represent an exhaustively comprehensive set of practices, nor are they fully portable to other software development and operations environments, a fact DevOpsCo will readily admit.

---

12 It would be quite a tall order to spend effort, time, and money to continue to evolve internal processes just for the benefit of a single researcher and case-study; indeed, this evolution continued far beyond the case-study itself, through the following months of the writing of this thesis: see Appendix G.3.
This thesis is not an incident management or organizational learning recipe book. But DevOpsCo does represent a company in which those practices challenge some of the classic, most firmly held notions of what software development and operations companies ‘must do’ to ‘100% remediate’ incidents and ‘fully prevent future recurrence.’ The proof is the existence that these questions can be asked and these processes can be experimented with and evolved, and done so deliberately and thoughtfully, and doing so will not cause the company to go out of business.

Ultimately, the question we, as technology industry practitioners, must ask ourselves is: within our organizations’ and our own context, if we want to improve organizational learning and operational outcomes, should we be asking these questions DevOpsCo’s practices, experience, and results prompts us to ask? And if so, how do we grapple with and sustainably incorporate the answers into our own existing complex adaptive socio-technical systems?

### 6.3 Bringing Software Into the ‘Safety’ Fold

This thesis began with a conscious acknowledgment of the unique (at least, as of now) field under study, the so-called new lens through which ‘safety’ was once again explored. (See section 1.1.) Thus, as we conclude this specific investigation, it is only natural we pause to reflect upon this point again, so that we may truly appreciate the sights observed through this new framing, both new and fantastical, and comfortable and expected.

While discussed above, three particular findings are worth reiterating here, not only because they are worth further consideration by the safety sciences, but because they are notably unique even among software operations companies:

- As discussed in section 4.4.2, different groups use different post-incident analysis artifacts differently. Establishing this fact is, indeed, important and while it is not unique to the software industry, one of the aspects of this difference in usage of artifacts is just how differently those different roles use them in the software development and operations industry: the way Operations uses these artifacts is significantly different than how either developers or security engineers use them.

- Two particular aspects of the way DevOpsCo uses artifacts are unique, both in the safety sciences and software:

  1. The first is DevOpsCo’s prevalent, established pattern of pushing operational control and responsibility to the sharp-end of the organization, as opposed to distributing it across the sharp-end/blunt-end
spectrum, or concentrating it near executive leaders. The practice itself is not only uncommon, but that it results in better operational outcomes (as perceived by customers and DevOpsCo’s industry reputation) makes it an example of yet another existence proof.

2. The second is the admission that for certain operational incidents, ‘doing nothing’ may in fact be the best answer for the organization, as DevCo describes in section 5.3.2. The need to ‘do something’ in response to an operational incident or accident is well-known to the safety sciences, and generally expected in software development and operations (especially given the prevalent notion that incidents are often ‘caused’ by ‘bugs’ in software or infrastructure).

But here again, DevOpsCo represents an existence proof that even in costly operational incidents, the value may be determined to exist wholly within the organizational learning that can result from the organization’s response to the incident, and when engineers take into account the various trade-offs that would need to be made to remediate a particular issue, the analysis illustrates that no actual additional remediation work should be performed.

Notably, these two aspects fit hand-in-glove: these decisions about what responses to organizational incidents are the most relevant or make the most sense, in DevOpsCo’s case, must be made by the sharp-end actors; that is: the likelihood that they would be ‘uninformed’ or ‘incorrectly assessed’ would be much higher the further from that sharp-end work they are made.

• Finally, the use of post-incident analysis artifacts to map the (constantly changing) terrain of the socio-technical system, as discussed in section 5.3.1, is worthy of some additional discussion.

This particular use may not necessarily be novel in fields examined by the safety sciences: any noted ‘discoveries’ of dependent systems or actors revealed in incident and accident reports represents precisely this mechanism. But what is unique to software, and arguably of critical importance, is the rate of change of the terrain that represents this socio-technical system is incredibly high in software development and operations. This is due to numerous factors, including the malleability of software (discussed in section 1.1), the often high rate of personnel turnover in software organizations, and general pace of innovation in the industry, not only for the physical and logical infrastructure upon which we build these systems, but the tools
with which we build them and the rate at which (the market, and therefore business demands) we change them.

No other industry under observation of the safety sciences to date that comes immediately to mind\textsuperscript{13} grapples with this rate of change. And it is because of this system property of software operations that this use of post-incident artifacts—specifically to map its constantly- and quickly-changing socio-technical environment—is both unique, but also incredibly important. In fact, one could argue the scale of software operations systems can be limited by its ability to have a(n at least minimally actionable) map to use: without it, incidents and outages prevail.

Ultimately, perhaps the most stark realization is this “new lens” works—and provides value—bidirectionally: as software development and operations is brought into the safety sciences fold, adding it to the family of industries there is data with which to compare, contrast, and make conclusions, we also create a feedback loop for software developers and operators, giving some additional (and hopefully useful) form, structure, and explanation to their behaviors, and further making deeper sense of the people and practices within software’s unique, but also (still) nuanced and messy systems.

6.4 A Final Word on the Survey Statistical Analysis

As discussed in section 3.1, the original intent of the industry survey was to collect some ‘real world’ data on post-incident language usage, organizational behaviors, artifact usage patterns, and solicit actual artifacts used in the study. This was intended to help drive the form and structure of the organizational case study structured interviews.

As the survey data was analyzed for this purpose, a couple of interesting patterns were noted about what ‘large’ versus ‘small’ companies do, with respect to some of the questions asked in the survey. This prompted the researcher’s curiosity about testing the collected profile data—job title and organizational size—against the other language, usage, and behavioral data collected in the study. Somewhat unexpectedly, some notable correlations were found.

\textsuperscript{13}The changes at all levels of the system would be as if aviation had to contend with airport infrastructure built from different materials, while flying equipment which is worked on while in operation. One field with a reasonably high rate-of-change is healthcare, but even here, there are limits to the change of practices within the industry, due to the work involving humans which would prefer not-to-be harmed, and will accept a slower rate of change because of that desire.
Analyzing this data took this researcher down a bit of a rabbit hole with respect to statistical analysis techniques. (And wrangling the software tools required to perform analysis!) Because many aspects of artifact usage were surveyed, the generation of many hypotheses to test was quite easy. However, the ability to generate many testable hypotheses raises the spectre of the multiple testing effect, which was addressed using a False Discovery Rate adjustment to the p-values. (Benjamini & Hochberg, 1995) In some cases, this adjustment had an effect on the significance of the p-value; in other cases, the p-values remained statistically significant.

Ultimately, unadjusted p-values were reported and discussed for hypotheses with statistically significant p-values because even though in cases where the False Discovery Rate correction calls into question whether those notable results were obtained by lucky chance, taken together, the observed patterns have important implications for the direction of future studies. In this way, this research work unexpectedly provided not only a sketch for a map of further avenues for additional research in some very specific areas of post-incident artifact usage, but a mechanism as well, e.g. an industry survey deliberately designed for that purpose, and shown it can be a worthwhile tool for creating a richer, more detailed map of our understanding of those avenues.

Were an industry survey designed this specific purpose in mind—as opposed to the more generalized survey used as a primer for structuring the organizational study presented herein—it would surely be able to provide more insight, with stronger evidence of some of the correlations this study was only able to suggest exist. Despite the evidence not being as strong as any researcher would prefer, I still find the existence of behavioral differences based on job role and organizational size, as well as some suggestions of various aspects to further examine those differences, to be an (unplanned and therefore unexpected) successful product of the research activities.

### 6.5 Future Research Avenues

The bane of any investigator is the temptation to wander beyond the scope of the research question. This investigation was no exception, and there were numerous avenues left unexplored:

- Allspaw’s thesis (Allspaw, 2015) explores the question ‘What do engineers *actually do* while working an outage?’ Here, we explore ‘How do organizations *actually use* the artifacts generated by a post-incident analysis process?’ There is an obvious series of fascinating questions around what happens individually, organizationally, and technologically *between* these two
events, i.e. ‘What do organizations actually do during a post-incident analysis event/meeting?’

- The treatment of the differences between responses to operational incidents (server outage, etc.) and security incidents was shallow in this thesis; how these processes are different, why they’re different, what security engineers do in these situations, and how they overlap with current operational incident response practices would be interesting, especially given the only body of operational retrospective research found in the IT sphere was security (see section 2.5.1).

- Numerous informants made reference to the processes they use to find (data contained in) old incident artifacts; one informant said they maintained their own separate set of personal notes, including incident data. Investigation into the specific techniques and cognitive recall processes that cause engineers to search for data on previous incidents, as well as the strategies they use to do so, could yield insights into improved categorization, and thus more effective use of these artifacts.

- When asked whether previous incident’s artifacts are used during an incident, informants provided a cacophony of answers. The dichotomy is briefly discussed in section 4.4.3, but deeper research into this question and the reasons behind the differences in informants’ answers (even the same informant’s answers) could further help us to understand whether there is a cognitive mechanism at play here, or the difference is due to operational role, or something else entirely different is occurring in the world.

- As mentioned above, the creation and administration of an industry practices survey, with specific hypothetical tests designed into the survey regarding post-incident review behaviors and post-incident artifact usage patterns could yield a variety of more definitive conclusions about these areas, which we were only able to examine at a higher level here.
Chapter 7

Conclusions

Organizations who both develop and operate software can engage in a wide range of activities which can be broadly described as “post-incident analysis”; they produce a number of different artifacts as a result of these activities. Defining and categorizing these artifacts is useful because it leads to insight regarding their use within the organization and the extent of their contribution to organizational learning.

One of the most vivid artifacts to examine are the templates these organizations create to drive their post-incident analysis processes. Three template archetypes, each fulfilling a different role and purpose were observed from industrial survey samples:

- **The ‘Record-Keeper’** archetype serves to record and memorialize an incident. It is generally stored in some organizational system-of-record for posterity. It is the most ‘familiar’ type of retrospective artifact.

- **The ‘Facilitator’** archetype is similar to the Record Keeper, but woven throughout the template are instructions on running post-incident analysis events and guidelines and reminders on what the organization’s values, goals, and desires are for a post-incident analysis process. In this way, its focus is on the event itself and the practice of those analysis activities.

- **The ‘Sign-Post’** archetype, though uncommon, is notably different in that its role is to provide a vehicle for system actors to easily and quickly digest high-level summations of organizational incidents and their learnings. Because of this, it is commonly used either for short (cross-)indexing of incidents or for wide distribution throughout the organization to facilitate organizational learning.
Additionally, in the course of a deep phenomenological case study of a high-performing software development and operations company, extensive and deliberate use of their post-incident analysis artifacts was observed. These myriad uses were distilled into three distinct activities, each contributing to the company’s improved operational outcomes, which are widely publicized:

1. **Mapping the organization’s complex socio-technical system**, specifically by providing insight into heretofore unobserved connections within the system, both technical and social, and bringing actor awareness to so-called system ‘hot-spots,’ which left unaddressed, could lead to more (and more serious) incidents.

2. The management of incident remediation via context-sharing in lieu of static lists of remediation items: artifacts are used to share rich context between system actors and teams, *not* as a way to ‘check off’ lists of remediation items. This can, counterintuitively, lead to *sharp-end actors deciding to not-act on identified remediation items, and this is an organizationally-acceptable end-state for that remediation item within this particular organization.*

3. As a mechanism to curate organizational culture and transmit tribal knowledge: the information contained in the artifacts serve as the basis for stories the organization tells itself and others, which in turn is used to constantly modify and evolve operational behaviors away from practices which have caused issues in the past and inform software design, development, and operational practices such that previous lessons become encoded not only in a “this is how we do it here”-ethos, but in a way that can still be credibly referenced should questions such as “why, exactly, do we do it that way?” arise.

Though the structure and elements of a sample of post-incident artifacts are described and one organization’s varied uses for its post-incident artifacts are illustrated, ultimately, an answer to the question “do technology organizations leverage their post-incident artifacts in a way that would demonstrate ‘organizational learning?’” remains elusive.

Certainly, the organizational case study herein provides an existence proof of organizational learning through effective use of post-incident artifacts. It also demonstrates that such use not only contributes to one specific organization’s industry reputation as a “high-performer,” but it also details how the artifacts serve to influence both software development and operational outcomes that make said reputation credible.
While this investigation does not provide a clear answer to that eternal question, it does heavily suggest that should we want to promote organizational learning, we would do well to observe organizational behavior—that is, how are the maps of our socio-technical systems evolved, how do we communicate with each other about more effective, deeply-contextualized solutions to the hard problems presented to us by incidents, and how do we transmit this information throughout an organization and transform it from interesting-story into living-ethos—rather than continue to measure mean-times-to-anything as if it were some meaningful metric on the health of our systems, count how many remediation items we can conjure up to an incident that may never occur again, or spend significant organizational resources straining to ‘check off’ each and every item on a static checklist describing remediations to fix our dynamic, living, complex systems.
References:


Appendix A

Survey Questions and Results

**Question 1:** Approximately how many people total does your organization employ?

<table>
<thead>
<tr>
<th>Number of Employed</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>15.0%</td>
</tr>
<tr>
<td>51-100</td>
<td>10.7%</td>
</tr>
<tr>
<td>101-500</td>
<td>27.0%</td>
</tr>
<tr>
<td>501-2000</td>
<td>17.9%</td>
</tr>
<tr>
<td>2000+</td>
<td>29.3%</td>
</tr>
</tbody>
</table>

**Question 2:** Which title best describes your role?

<table>
<thead>
<tr>
<th>Role</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developer</td>
<td>16.3%</td>
</tr>
<tr>
<td>Operations Engineer</td>
<td>36.2%</td>
</tr>
<tr>
<td>Other Engineer (QA, release engineering, security, etc.)</td>
<td>9.1%</td>
</tr>
<tr>
<td>Engineering Management</td>
<td>32.9%</td>
</tr>
<tr>
<td>Non-engineering Management</td>
<td>3.3%</td>
</tr>
<tr>
<td>Other, non-engineering role</td>
<td>2.3%</td>
</tr>
</tbody>
</table>

**Question 3:** In cases where your team gets together to review an incident/outage, what name do they use to refer to that event/meeting?

<table>
<thead>
<tr>
<th>Name</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Postmortem</td>
<td>59.6%</td>
</tr>
<tr>
<td>Retrospective</td>
<td>17.3%</td>
</tr>
<tr>
<td>Learning review</td>
<td>1.3%</td>
</tr>
<tr>
<td>After action report</td>
<td>3.6%</td>
</tr>
<tr>
<td>Rapid improvement event</td>
<td>0.7%</td>
</tr>
<tr>
<td>Other – Root Cause Analysis (specifically)</td>
<td>5.6%</td>
</tr>
<tr>
<td>Other – (Post-) Incident Review/Retrospective (specifically)</td>
<td>5.9%</td>
</tr>
<tr>
<td>Other – Total</td>
<td>17.6%</td>
</tr>
</tbody>
</table>
Question 4: On average, how many of these review-events are held per month?

<table>
<thead>
<tr>
<th>Frequency</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>15.6%</td>
</tr>
<tr>
<td>1</td>
<td>34.2%</td>
</tr>
<tr>
<td>2-4</td>
<td>34.5%</td>
</tr>
<tr>
<td>5-8</td>
<td>8.1%</td>
</tr>
<tr>
<td>More than 8</td>
<td>7.5%</td>
</tr>
</tbody>
</table>

Question 5: On average, how long after the incident/outage do you hold the event where the team reviews it?

<table>
<thead>
<tr>
<th>Duration</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-24 hours after the incident</td>
<td>20.2%</td>
</tr>
<tr>
<td>1-3 working days after the incident</td>
<td>53.1%</td>
</tr>
<tr>
<td>4-5 working days after the incident</td>
<td>17.6%</td>
</tr>
<tr>
<td>Over 5 working days after the incident</td>
<td>9.1%</td>
</tr>
</tbody>
</table>

Question 6: Does the event have an identified facilitator?

<table>
<thead>
<tr>
<th>Facilitator Existence</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>68.4%</td>
</tr>
<tr>
<td>No</td>
<td>31.6%</td>
</tr>
</tbody>
</table>

Question 7: Which of the following items are usually created and/or recorded during this review-event? (Select all that apply.)

<table>
<thead>
<tr>
<th>Item</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>List of remediation items</td>
<td>90.6%</td>
</tr>
<tr>
<td>Event timeline</td>
<td>85.3%</td>
</tr>
<tr>
<td>Customer impact</td>
<td>68.1%</td>
</tr>
<tr>
<td>Technical analyses</td>
<td>66.8%</td>
</tr>
<tr>
<td>Incident response analyses</td>
<td>59.0%</td>
</tr>
<tr>
<td>Business impact</td>
<td>56.7%</td>
</tr>
<tr>
<td>Documentation updates</td>
<td>45.0%</td>
</tr>
<tr>
<td>Individual narratives</td>
<td>27.0%</td>
</tr>
<tr>
<td>Other</td>
<td>6.8%</td>
</tr>
</tbody>
</table>

Question 8: Where are these items from the review-event stored and/or recorded? (Select all which apply.)

<table>
<thead>
<tr>
<th>Storage Method</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team ticketing system (i.e. Jira, ServiceNow, etc.)</td>
<td>59.0%</td>
</tr>
<tr>
<td>A review-event specific tool (i.e. Etsy Morgue, etc.)</td>
<td>7.5%</td>
</tr>
<tr>
<td>Email / email distribution list</td>
<td>24.1%</td>
</tr>
<tr>
<td>Team / organization wiki</td>
<td>47.6%</td>
</tr>
<tr>
<td>Google Docs (or similar)</td>
<td>31.6%</td>
</tr>
<tr>
<td>Team runbooks</td>
<td>12.7%</td>
</tr>
<tr>
<td>Physical documentation (i.e. paper)</td>
<td>2.0%</td>
</tr>
<tr>
<td>Other (please specify)</td>
<td>8.8%</td>
</tr>
</tbody>
</table>
Question 9: Are these items from the review-event accessible to others within the organization?

<table>
<thead>
<tr>
<th>Access to items</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Publicly available to anyone within the organization</td>
<td>57.0%</td>
</tr>
<tr>
<td>Available, but in many different locations or in team-specific locations which are not known to everyone</td>
<td>27.4%</td>
</tr>
<tr>
<td>Access to these items is limited to certain people/teams</td>
<td>15.6%</td>
</tr>
</tbody>
</table>

Question 10: Are these items from the review-event distributed to others within the organization in any way?

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>34.9%</td>
</tr>
<tr>
<td>No</td>
<td>20.5%</td>
</tr>
<tr>
<td>Sometimes, but not consistently</td>
<td>44.6%</td>
</tr>
</tbody>
</table>

Question 11: If items from the post-incident review-event are distributed, how are they distributed?

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>They are not distributed</td>
<td>17.9%</td>
</tr>
<tr>
<td>Individual email</td>
<td>46.6%</td>
</tr>
<tr>
<td>Chat rooms (Slack, IRC, etc.)</td>
<td>40.1%</td>
</tr>
<tr>
<td>Team standup, scrum, or review meeting(s)</td>
<td>23.8%</td>
</tr>
<tr>
<td>Internal company/team newsletter (electronic or paper)</td>
<td>16.0%</td>
</tr>
<tr>
<td>Company blog / marketing communication</td>
<td>4.9%</td>
</tr>
<tr>
<td>Customer support communication</td>
<td>10.4%</td>
</tr>
<tr>
<td>Other</td>
<td>6.8%</td>
</tr>
</tbody>
</table>

Question 12: Who is responsible for creating these event-review items?

<table>
<thead>
<tr>
<th>Responsibility</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>The review facilitator</td>
<td>38.1%</td>
</tr>
<tr>
<td>An identified &quot;scribe&quot; / note taker</td>
<td>18.2%</td>
</tr>
<tr>
<td>Different attendees create different artifacts</td>
<td>34.9%</td>
</tr>
<tr>
<td>Other</td>
<td>8.8%</td>
</tr>
</tbody>
</table>

Question 13: For any of the review-event items produced, is there a template used?

<table>
<thead>
<tr>
<th>Response</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>60.6%</td>
</tr>
<tr>
<td>No</td>
<td>39.4%</td>
</tr>
</tbody>
</table>
**Question 14:** If a template exists for some review-event items, which items have a template? (Select all which apply.)

<table>
<thead>
<tr>
<th>Item</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event timeline</td>
<td>83.9%</td>
</tr>
<tr>
<td>List of remediation items</td>
<td>74.9%</td>
</tr>
<tr>
<td>Documentation updates</td>
<td>22.1%</td>
</tr>
<tr>
<td>Individual narratives</td>
<td>13.6%</td>
</tr>
<tr>
<td>Technical analyses</td>
<td>46.2%</td>
</tr>
<tr>
<td>Incident response analyses</td>
<td>49.2%</td>
</tr>
<tr>
<td>Business impact</td>
<td>42.7%</td>
</tr>
<tr>
<td>Customer impact</td>
<td>52.8%</td>
</tr>
<tr>
<td>Other</td>
<td>8.5%</td>
</tr>
</tbody>
</table>

**Question 15:** If possible, please upload a sample BLANK template your team uses for one of its post-incident review-event artifacts

Responses: 22 (See Appendix D.)

**Question 16:** If you would like to be contacted further about this study and its results, please provide your email address. It will not be used for any other purpose than communications related to this study.

Responses: 88
Appendix B

Survey Statistical Analysis

B.1 Hypothesis Correlation Tests

The Phase I industry survey results were statistically analyzed to search for correlations between the two pieces of collected profile data—respondent job title and organizational size—and differences in the use of post-incident analysis artifacts. The tested hypotheses were of the form “Does job role X correlate to artifact use Y?” or “Does (difference in) company size correlate to artifact use Y?”

For job title profile data, the responses (\(N = 307\)) were re-bucketed into broader buckets of “Developer,” (\(n = 50\)) “Operations” (\(n = 139\), representing “Operations Engineers” and “Other Engineer”), and “Management” (\(n = 111\), representing “Engineering” and “Non-engineering” management); “Other, non-engineering role” responses (\(n = 7\)) were removed from the dataset, leaving a final \(N = 300\). Organizational size data was kept in its original five categories (0-50, 51-100, 101-500, 501-2000, and 2000+) for analysis.

R version 3.4.1\(^1\) was used for the statistical analysis, specifically Pearson’s chi-squared tests.\(^2\) The data was organized into a contingency table, via R, and correlations between the combined role groups and various artifact usage survey questions and organizational size and various artifact usage survey questions were tested for. In cases where R emitted a warning regarding p-value simulation\(^3\), p-values were computed by Monte Carlo simulation.\(^4\) Tests where this was necessary are

\(^{1}\)See https://www.r-project.org/

\(^{2}\)See https://www.rdocumentation.org/packages/stats/versions/3.4.1/topics/chisq.test

\(^{3}\)This is necessary in cases where the data violates some assumptions about the statistical test, namely: no cells in the contingency table should be less than 1 (i.e., every possible pair of unique categories should occur at least once); and 80% of the expected cell counts in the contingency table should be greater than 5.

\(^{4}\)The simulate.p.value option was enabled in the chisq.test R function; see the above documentation for further information.
denoted below where the degrees of freedom is listed as “NA” and one of three samples computed with the simulated values is presented.

Because a large number of post-incident artifact usage data was collected in the survey, there exist a large number of hypotheses to test the profile data against. To address the multiple testing effect, the False Discovery Rate-corrected p-value\(^5\) was also calculated by using R’s \texttt{p.adjust} function\(^6\) and is reported in the following table. See section 6.4 for further analytical discussion of the impact of these FDR-corrected p-values on these correlation findings.

Based on this analysis, hypotheses showing a statistically significant correlation (\(p \leq 0.05\)) are bolded; hypotheses showing no statistically significant correlation (\(p \geq 0.95\)) are italicized. See section 5.1 for a long-form analysis of the statistical data. See Appendix C.1 for the source code to the R script used to analyze the survey data.

<table>
<thead>
<tr>
<th>ID</th>
<th>Tested Hypothesis</th>
<th>X-squared</th>
<th>Degrees of Freedom</th>
<th>p-value</th>
<th>FDR-corrected p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>H.a.1</td>
<td>... the number of review-events held per month.</td>
<td>5.529</td>
<td>NA</td>
<td>0.686</td>
<td>0.963</td>
</tr>
<tr>
<td>H.a.2</td>
<td>... the length of time after the incident/outage the review-event is held.</td>
<td>4.766</td>
<td>NA</td>
<td>0.575</td>
<td>0.906</td>
</tr>
<tr>
<td>H.a.3</td>
<td>... the review-event having an identified facilitator.</td>
<td>2.788</td>
<td>2</td>
<td>0.248</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.4</td>
<td>... remediation items are created during the review-event.</td>
<td>4.007</td>
<td>NA</td>
<td>0.139</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.5</td>
<td>... a timeline is created during the review-event.</td>
<td>1.611</td>
<td>2</td>
<td>0.447</td>
<td>0.827</td>
</tr>
<tr>
<td>H.a.6</td>
<td>... a customer impact analysis is created during review-event.</td>
<td>1.890</td>
<td>2</td>
<td>0.389</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.7</td>
<td>... technical analyses are created during review-event.</td>
<td>2.224</td>
<td>2</td>
<td>0.329</td>
<td>0.820</td>
</tr>
</tbody>
</table>

---

\(^5\)See Benjamini and Hochberg, 1995.  
\(^6\)See \url{https://www.rdocumentation.org/packages/stats/versions/3.4.1/topics/p.adjust} for more information.
<table>
<thead>
<tr>
<th>ID</th>
<th>Tested Hypothesis</th>
<th>X-squared</th>
<th>Degrees of Freedom</th>
<th>p-value</th>
<th>FDR-corrected p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>H.a.8</td>
<td>... incident response analyses are created during review-event.</td>
<td>1.151</td>
<td>2</td>
<td>0.562</td>
<td>0.906</td>
</tr>
<tr>
<td>H.a.9</td>
<td>... a business impact analysis is created during review-event.</td>
<td>2.875</td>
<td>2</td>
<td>0.237</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.10</td>
<td>... documentation updates are created during the review-event.</td>
<td>11.093</td>
<td>2</td>
<td>0.004</td>
<td>0.148</td>
</tr>
<tr>
<td>H.a.11</td>
<td>... individual narratives are collected during the review-event.</td>
<td>2.637</td>
<td>2</td>
<td>0.268</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.12</td>
<td>... review-event items being stored in a team ticketing system.</td>
<td>3.251</td>
<td>2</td>
<td>0.197</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.13</td>
<td>... review-event items being stored in a review-event specific tool.</td>
<td>2.140</td>
<td>NA</td>
<td>0.344</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.14</td>
<td>... review-event items being stored in email/distribution lists.</td>
<td>0.495</td>
<td>2</td>
<td>0.781</td>
<td>0.963</td>
</tr>
<tr>
<td>H.a.15</td>
<td>... review-event items being stored in a wiki system.</td>
<td>1.685</td>
<td>2</td>
<td>0.431</td>
<td>0.827</td>
</tr>
<tr>
<td>H.a.16</td>
<td>... review-event items being stored in Google Docs (or similar).</td>
<td>0.579</td>
<td>2</td>
<td>0.749</td>
<td>0.963</td>
</tr>
<tr>
<td>H.a.17</td>
<td>... review-event items being stored in team runbooks.</td>
<td>1.061</td>
<td>2</td>
<td>0.588</td>
<td>0.906</td>
</tr>
<tr>
<td>H.a.18</td>
<td>... review-event items being stored on physical documentation.</td>
<td>0.503</td>
<td>NA</td>
<td>0.853</td>
<td>0.969</td>
</tr>
<tr>
<td>H.a.19</td>
<td>... review-event items being accessible to others within the organization.</td>
<td>1.793</td>
<td>4</td>
<td>0.774</td>
<td>0.963</td>
</tr>
<tr>
<td>ID</td>
<td>Tested Hypothesis</td>
<td>X-squared</td>
<td>Degrees of Freedom</td>
<td>p-value</td>
<td>FDR-corrected p-value</td>
</tr>
<tr>
<td>-------</td>
<td>----------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>--------------------</td>
<td>---------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>H.a.20</td>
<td>... review-event items being distributed to others in the organization.</td>
<td>1.943</td>
<td>4</td>
<td>0.746</td>
<td>0.963</td>
</tr>
<tr>
<td>H.a.21</td>
<td>... review-event items not being distributed.</td>
<td>0.733</td>
<td>2</td>
<td>0.693</td>
<td>0.963</td>
</tr>
<tr>
<td>H.a.22</td>
<td>... review-event items being distributed via email.</td>
<td>0.262</td>
<td>2</td>
<td>0.877</td>
<td>0.969</td>
</tr>
<tr>
<td>H.a.23</td>
<td>... review-event items being distributed via chat rooms.</td>
<td>1.157</td>
<td>2</td>
<td>0.561</td>
<td>0.906</td>
</tr>
<tr>
<td>H.a.24</td>
<td>... review-event items being distributed via team standup.</td>
<td>3.722</td>
<td>2</td>
<td>0.155</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.25</td>
<td>... review-event items being distributed via company/team newsletter.</td>
<td>3.416</td>
<td>2</td>
<td>0.181</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.26</td>
<td>... review-event items being distributed via company blog and/or marketing commu-</td>
<td>1.946</td>
<td>NA</td>
<td>0.372</td>
<td>0.820</td>
</tr>
<tr>
<td></td>
<td>nication.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H.a.27</td>
<td>... review-event items being distributed via customer support communication.</td>
<td>0.002</td>
<td>2</td>
<td>0.999</td>
<td>0.999</td>
</tr>
<tr>
<td>H.a.28</td>
<td>... the role responsible for creating review-event items.</td>
<td>0.847</td>
<td>NA</td>
<td>0.992</td>
<td>0.999</td>
</tr>
<tr>
<td>H.a.29</td>
<td>... the produced review-event items being created with templates.</td>
<td>2.381</td>
<td>2</td>
<td>0.304</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.30</td>
<td>... the event timeline being created with a template.</td>
<td>0.173</td>
<td>2</td>
<td>0.917</td>
<td>0.969</td>
</tr>
<tr>
<td>H.a.31</td>
<td>... the list of remediation items being created with a template.</td>
<td>0.176</td>
<td>2</td>
<td>0.916</td>
<td>0.969</td>
</tr>
<tr>
<td>ID</td>
<td>Tested Hypothesis</td>
<td>X-squared</td>
<td>Degrees of Freedom</td>
<td>p-value</td>
<td>FDR-corrected p-value</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>--------------------</td>
<td>---------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>H.a.32</td>
<td>... documentation updates being created with a template.</td>
<td>6.912</td>
<td>2</td>
<td>0.032</td>
<td>0.592</td>
</tr>
<tr>
<td>H.a.33</td>
<td>... individual narratives being created with a template.</td>
<td>5.247</td>
<td>NA</td>
<td>0.074</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.34</td>
<td>... technical analyses being created with a template.</td>
<td>0.236</td>
<td>2</td>
<td>0.889</td>
<td>0.969</td>
</tr>
<tr>
<td>H.a.35</td>
<td>... incident response analyses being created with a template.</td>
<td>1.890</td>
<td>2</td>
<td>0.389</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.36</td>
<td>... business impact analysis being created with a template.</td>
<td>1.840</td>
<td>2</td>
<td>0.399</td>
<td>0.820</td>
</tr>
<tr>
<td>H.a.37</td>
<td>... customer impact analysis being created with a template.</td>
<td>3.352</td>
<td>2</td>
<td>0.187</td>
<td>0.820</td>
</tr>
</tbody>
</table>

Organizational size correlates to...

<table>
<thead>
<tr>
<th>ID</th>
<th>Tested Hypothesis</th>
<th>Value</th>
<th>Degrees of Freedom</th>
<th>p-value</th>
<th>FDR-corrected p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>H.b.1</td>
<td>... the number of review-events held per month.</td>
<td>62.666</td>
<td>NA</td>
<td>0.001</td>
<td>0.007</td>
</tr>
<tr>
<td>H.b.2</td>
<td>... the length of time after the incident/outage the review-event is held.</td>
<td>30.192</td>
<td>NA</td>
<td>0.002</td>
<td>0.012</td>
</tr>
<tr>
<td>H.b.3</td>
<td>... the review-event having an identified facilitator.</td>
<td>4.111</td>
<td>4</td>
<td>0.391</td>
<td>0.579</td>
</tr>
<tr>
<td>H.b.4</td>
<td>... remediation items are created during the review-event.</td>
<td>0.620</td>
<td>NA</td>
<td>0.959</td>
<td>0.959</td>
</tr>
<tr>
<td>H.b.5</td>
<td>... a timeline is created during the review-event.</td>
<td>2.272</td>
<td>NA</td>
<td>0.679</td>
<td>0.910</td>
</tr>
<tr>
<td>H.b.6</td>
<td>... a customer impact analysis is created during review-event.</td>
<td>4.744</td>
<td>4</td>
<td>0.315</td>
<td>0.530</td>
</tr>
<tr>
<td>H.b.7</td>
<td>... technical analyses are created during review-event.</td>
<td>5.863</td>
<td>4</td>
<td>0.210</td>
<td>0.457</td>
</tr>
<tr>
<td>ID</td>
<td>Tested Hypothesis</td>
<td>X-squared</td>
<td>Degrees of Freedom</td>
<td>p-value</td>
<td>FDR-corrected p-value</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>--------------------</td>
<td>---------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>H.b.8</td>
<td>... incident response analyses are created during review-event.</td>
<td>8.938</td>
<td>4</td>
<td>0.063</td>
<td>0.179</td>
</tr>
<tr>
<td>H.b.9</td>
<td>... a business impact analysis is created during review-event.</td>
<td>5.292</td>
<td>4</td>
<td>0.259</td>
<td>0.465</td>
</tr>
<tr>
<td>H.b.10</td>
<td>... documentation updates are created during the review-event.</td>
<td>0.793</td>
<td>4</td>
<td>0.939</td>
<td>0.959</td>
</tr>
<tr>
<td>H.b.11</td>
<td>... individual narratives are collected during the review-event.</td>
<td>2.097</td>
<td>4</td>
<td>0.718</td>
<td>0.816</td>
</tr>
<tr>
<td>H.b.12</td>
<td>... review-event items being stored in a team ticketing system.</td>
<td>3.072</td>
<td>4</td>
<td>0.546</td>
<td>0.697</td>
</tr>
<tr>
<td>H.b.13</td>
<td>... review-event items being stored in a review-event specific tool.</td>
<td>11.627</td>
<td>NA</td>
<td>0.020</td>
<td>0.082</td>
</tr>
<tr>
<td>H.b.14</td>
<td>... review-event items being stored in email/distribution lists.</td>
<td>0.876</td>
<td>4</td>
<td>0.928</td>
<td>0.959</td>
</tr>
<tr>
<td>H.b.15</td>
<td>... review-event items being stored in a wiki system.</td>
<td>2.386</td>
<td>4</td>
<td>0.665</td>
<td>0.810</td>
</tr>
<tr>
<td>H.b.16</td>
<td>... review-event items being stored in Google Docs (or similar).</td>
<td>5.628</td>
<td>4</td>
<td>0.229</td>
<td>0.460</td>
</tr>
<tr>
<td>H.b.17</td>
<td>... review-event items being stored in team runbooks.</td>
<td>3.484</td>
<td>NA</td>
<td>0.484</td>
<td>0.663</td>
</tr>
<tr>
<td>H.b.18</td>
<td>... review-event items being stored on physical documentation.</td>
<td>3.248</td>
<td>NA</td>
<td>0.502</td>
<td>0.663</td>
</tr>
<tr>
<td>H.b.19</td>
<td>... review-event items being accessible to others within the organization.</td>
<td>15.477</td>
<td>8</td>
<td>0.051*</td>
<td>0.172</td>
</tr>
<tr>
<td>ID</td>
<td>Tested Hypothesis</td>
<td>X-squared</td>
<td>Degrees of Freedom</td>
<td>p-value</td>
<td>FDR-corrected p-value</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>--------------------</td>
<td>---------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>H.b.20</td>
<td>... review-event items being distributed to others in the organization.</td>
<td>4.793</td>
<td>8</td>
<td>0.779</td>
<td>0.848</td>
</tr>
<tr>
<td>H.b.21</td>
<td>... review-event items not being distributed.</td>
<td>3.818</td>
<td>4</td>
<td>0.431</td>
<td>0.613</td>
</tr>
<tr>
<td>H.b.22</td>
<td>... review-event items being distributed via email.</td>
<td>5.232</td>
<td>4</td>
<td>0.264</td>
<td>0.465</td>
</tr>
<tr>
<td>H.b.23</td>
<td>... review-event items being distributed via chat rooms.</td>
<td>18.217</td>
<td>4</td>
<td>0.001</td>
<td>0.007</td>
</tr>
<tr>
<td>H.b.24</td>
<td>... review-event items being distributed via team standup.</td>
<td>6.711</td>
<td>4</td>
<td>0.152</td>
<td>0.352</td>
</tr>
<tr>
<td>H.b.25</td>
<td>... review-event items being distributed via company/team newsletter.</td>
<td>2.041</td>
<td>4</td>
<td>0.728</td>
<td>0.816</td>
</tr>
<tr>
<td>H.b.26</td>
<td>... review-event items being distributed via company blog and/or marketing commun-</td>
<td>18.570</td>
<td>NA</td>
<td>0.001</td>
<td>0.007</td>
</tr>
<tr>
<td></td>
<td>ication.</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H.b.27</td>
<td>... review-event items being distributed via customer support communication.</td>
<td>8.685</td>
<td>NA</td>
<td>0.068</td>
<td>0.180</td>
</tr>
<tr>
<td>H.b.28</td>
<td>... the role responsible for creating review-event items.</td>
<td>13.487</td>
<td>NA</td>
<td>0.334</td>
<td>0.537</td>
</tr>
<tr>
<td>H.b.29</td>
<td>... the produced review-event items being created with templates.</td>
<td>17.604</td>
<td>4</td>
<td>0.001</td>
<td>0.007</td>
</tr>
<tr>
<td>H.b.30</td>
<td>... the event timeline being created with a template.</td>
<td>15.010</td>
<td>4</td>
<td>0.005</td>
<td>0.026</td>
</tr>
<tr>
<td>H.b.31</td>
<td>... the list of remediation items being created with a template.</td>
<td>18.240</td>
<td>4</td>
<td>0.001</td>
<td>0.007</td>
</tr>
<tr>
<td>ID</td>
<td>Tested Hypothesis</td>
<td>X-squared</td>
<td>Degrees of Freedom</td>
<td>p-value</td>
<td>FDR-corrected p-value</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------------------------------------</td>
<td>-----------</td>
<td>--------------------</td>
<td>---------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>H.b.32</td>
<td>... documentation updates being created with a template.</td>
<td>4.212</td>
<td>NA</td>
<td>0.379</td>
<td>0.579</td>
</tr>
<tr>
<td>H.b.33</td>
<td>... individual narratives being created with a template.</td>
<td>8.889</td>
<td>NA</td>
<td>0.060</td>
<td>0.179</td>
</tr>
<tr>
<td>H.b.34</td>
<td>... technical analyses being created with a template.</td>
<td>6.822</td>
<td>4</td>
<td>0.146</td>
<td>0.352</td>
</tr>
<tr>
<td>H.b.35</td>
<td>... incident response analyses being created with a template.</td>
<td>5.544</td>
<td>4</td>
<td>0.236</td>
<td>0.460</td>
</tr>
<tr>
<td>H.b.36</td>
<td>... business impact analysis being created with a template.</td>
<td>10.328</td>
<td>4</td>
<td>0.035</td>
<td>0.130</td>
</tr>
<tr>
<td>H.b.37</td>
<td>... customer impact analysis being created with a template.</td>
<td>13.567</td>
<td>4</td>
<td>0.009</td>
<td>0.040</td>
</tr>
</tbody>
</table>

Table B.1: Survey Statistical Hypotheses Test Results

**B.2 Statistically Significant Contingency Tables**

For tested hypotheses which were found to be statistically significant, the contingency tables of the counts at each combination of factor levels (i.e. combined role or company size) are presented below, along with the tested hypothesis and the survey question from which the hypothesis was generated. These tables show the specific answers in each surveyed subpopulation, and were used to generate potential explanations for the observed correlations.

The contingency tables were generated using R’s `table` function. For easy reference, the population counts of each subpopulation are presented before the relevant contingency tables. See Appendix C.2 for the R source code that generated these tables.

---

7 The un-rounded p-value for this test was 0.05051; the investigator considered that to be ‘on-the-line’ of statistical significance.
8 See Appendix A for the survey questions and results.
9 See https://www.rdocumentation.org/packages/base/versions/3.4.1/topics/table
B.2.1  Job Role Hypothesis Contingency Tables

<table>
<thead>
<tr>
<th>Job Role</th>
<th>Subpopulation Count</th>
<th>Subpopulation Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developers</td>
<td>n = 50</td>
<td>16.7%</td>
</tr>
<tr>
<td>Operations</td>
<td>n = 139</td>
<td>46.3%</td>
</tr>
<tr>
<td>Managers</td>
<td>n = 111</td>
<td>37.0%</td>
</tr>
</tbody>
</table>

**Hypothesis H.a.10**: Job role shows a correlation to whether documentation updates are created during the review-event. (Question 7)

<table>
<thead>
<tr>
<th></th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developers</td>
<td>70.00%</td>
<td>30.00%</td>
</tr>
<tr>
<td>Operations</td>
<td>45.32%</td>
<td>54.68%</td>
</tr>
<tr>
<td>Managers</td>
<td>60.36%</td>
<td>39.64%</td>
</tr>
</tbody>
</table>

**Hypothesis H.a.32**: Job role shows a correlation to whether documentation updates are being created with a template. (Question 14)

<table>
<thead>
<tr>
<th></th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Developers</td>
<td>98.00%</td>
<td>2.00%</td>
</tr>
<tr>
<td>Operations</td>
<td>83.45%</td>
<td>16.55%</td>
</tr>
<tr>
<td>Managers</td>
<td>85.59%</td>
<td>14.41%</td>
</tr>
</tbody>
</table>
B.2.2 Organization Size Hypothesis Contingency Tables

<table>
<thead>
<tr>
<th>Organization Size</th>
<th>Subpopulation Count</th>
<th>Subpopulation Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>n = 46</td>
<td>15.3%</td>
</tr>
<tr>
<td>51-100</td>
<td>n = 32</td>
<td>10.7%</td>
</tr>
<tr>
<td>101-500</td>
<td>n = 81</td>
<td>27.0%</td>
</tr>
<tr>
<td>501-2000</td>
<td>n = 55</td>
<td>18.3%</td>
</tr>
<tr>
<td>2000+</td>
<td>n = 86</td>
<td>28.7%</td>
</tr>
</tbody>
</table>

**Hypothesis H.b.13**: Organizational size shows a correlation to whether review-event items are stored in a review-event specific tool. (Question 8)

<table>
<thead>
<tr>
<th></th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>97.83%</td>
<td>2.17%</td>
</tr>
<tr>
<td>51-100</td>
<td>100.00%</td>
<td>0.00%</td>
</tr>
<tr>
<td>101-500</td>
<td>93.83%</td>
<td>6.17%</td>
</tr>
<tr>
<td>501-2000</td>
<td>92.73%</td>
<td>7.27%</td>
</tr>
<tr>
<td>2000+</td>
<td>84.88%</td>
<td>15.12%</td>
</tr>
</tbody>
</table>

**Hypothesis H.b.19**: Organizational size shows a correlation to whether review-event items are accessible to others within the organization. (Question 9)

<table>
<thead>
<tr>
<th></th>
<th>Available - Known Location</th>
<th>Available - Unknown Location</th>
<th>Not Available</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>63.04%</td>
<td>23.91%</td>
<td>13.04%</td>
</tr>
<tr>
<td>51-100</td>
<td>62.50%</td>
<td>25.00%</td>
<td>12.50%</td>
</tr>
<tr>
<td>101-500</td>
<td>64.20%</td>
<td>27.16%</td>
<td>8.64%</td>
</tr>
<tr>
<td>501-2000</td>
<td>58.18%</td>
<td>29.09%</td>
<td>12.73%</td>
</tr>
<tr>
<td>2000+</td>
<td>41.86%</td>
<td>31.40%</td>
<td>26.74%</td>
</tr>
</tbody>
</table>

**Hypothesis H.b.23**: Organizational size shows a correlation to whether review-event items are distributed via chat rooms. (Question 11)

<table>
<thead>
<tr>
<th></th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>54.35%</td>
<td>45.65%</td>
</tr>
<tr>
<td>51-100</td>
<td>43.75%</td>
<td>56.25%</td>
</tr>
<tr>
<td>101-500</td>
<td>48.15%</td>
<td>51.85%</td>
</tr>
<tr>
<td>501-2000</td>
<td>69.09%</td>
<td>30.91%</td>
</tr>
<tr>
<td>2000+</td>
<td>74.42%</td>
<td>25.58%</td>
</tr>
</tbody>
</table>
Hypothesis H.b.26: Organizational size shows a correlation to whether review-event items are distributed via company blog and/or marketing communications. (Question 11)

<table>
<thead>
<tr>
<th></th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>95.65%</td>
<td>4.35%</td>
</tr>
<tr>
<td>51-100</td>
<td>100.00%</td>
<td>0.00%</td>
</tr>
<tr>
<td>101-500</td>
<td>87.65%</td>
<td>12.35%</td>
</tr>
<tr>
<td>501-2000</td>
<td>100.00%</td>
<td>0.00%</td>
</tr>
<tr>
<td>2000+</td>
<td>98.84%</td>
<td>1.16%</td>
</tr>
</tbody>
</table>

Hypothesis H.b.29: Organizational size shows a correlation to whether review-event items are created using templates. (Question 14)

<table>
<thead>
<tr>
<th></th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>65.22%</td>
<td>34.78%</td>
</tr>
<tr>
<td>51-100</td>
<td>28.13%</td>
<td>71.88%</td>
</tr>
<tr>
<td>101-500</td>
<td>30.86%</td>
<td>69.14%</td>
</tr>
<tr>
<td>501-2000</td>
<td>34.55%</td>
<td>65.45%</td>
</tr>
<tr>
<td>2000+</td>
<td>39.53%</td>
<td>60.47%</td>
</tr>
</tbody>
</table>

Hypothesis H.b.30: Organizational size shows a correlation to whether, specifically, incident event timelines are created using templates. (Question 14)

<table>
<thead>
<tr>
<th></th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>69.57%</td>
<td>30.43%</td>
</tr>
<tr>
<td>51-100</td>
<td>43.75%</td>
<td>56.25%</td>
</tr>
<tr>
<td>101-500</td>
<td>34.57%</td>
<td>65.43%</td>
</tr>
<tr>
<td>501-2000</td>
<td>41.82%</td>
<td>58.18%</td>
</tr>
<tr>
<td>2000+</td>
<td>46.51%</td>
<td>53.49%</td>
</tr>
</tbody>
</table>

Hypothesis H.b.31: Organizational size shows a correlation to whether, specifically, incident remediation items are created using templates. (Question 14)

<table>
<thead>
<tr>
<th></th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>76.09%</td>
<td>23.91%</td>
</tr>
<tr>
<td>51-100</td>
<td>40.63%</td>
<td>59.38%</td>
</tr>
<tr>
<td>101-500</td>
<td>40.74%</td>
<td>59.26%</td>
</tr>
<tr>
<td>501-2000</td>
<td>45.45%</td>
<td>54.55%</td>
</tr>
<tr>
<td>2000+</td>
<td>56.98%</td>
<td>43.02%</td>
</tr>
</tbody>
</table>
Hypothesis H.b.36: Organizational size shows a correlation to whether, specifically, a business impact analysis is created using templates. (Question 14)

<table>
<thead>
<tr>
<th></th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>89.13%</td>
<td>10.87%</td>
</tr>
<tr>
<td>51-100</td>
<td>75.00%</td>
<td>25.00%</td>
</tr>
<tr>
<td>101-500</td>
<td>69.14%</td>
<td>30.86%</td>
</tr>
<tr>
<td>501-2000</td>
<td>61.82%</td>
<td>38.18%</td>
</tr>
<tr>
<td>2000+</td>
<td>74.42%</td>
<td>25.58%</td>
</tr>
</tbody>
</table>

Hypothesis H.b.37: Organizational size shows a correlation to whether, specifically, a customer impact analysis is created using templates. (Question 14)

<table>
<thead>
<tr>
<th></th>
<th>No</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-50</td>
<td>82.61%</td>
<td>17.39%</td>
</tr>
<tr>
<td>51-100</td>
<td>75.00%</td>
<td>25.00%</td>
</tr>
<tr>
<td>101-500</td>
<td>55.56%</td>
<td>44.44%</td>
</tr>
<tr>
<td>501-2000</td>
<td>56.36%</td>
<td>43.64%</td>
</tr>
<tr>
<td>2000+</td>
<td>69.77%</td>
<td>30.23%</td>
</tr>
</tbody>
</table>
Appendix C

Survey Statistical Analysis Source Code

C.1 Correlation Tests

This R script was written to test all of the hypotheses for correlations against the two types of profile data.

This R script reads in the survey data, provides human-readable labels to the variables of the data, via the levels attribute\(^1\). It then uses a nested loop to test all of the various survey questions against the two profile types, i.e. job role and organization size.

An exception handler is used to detect when the Chi-Squared test results in a warning regarding the P value, and a Monte Carlo simulation must be used. In these cases, the simulation was run three times; the results of those three simulations are presented in Appendix B.1

```
#!/usr/bin/Rscript

surveydata <- read.csv("survey-data.csv", colClasses= rep("factor", 2))

levels(surveydata$role) <- c('Dev', 'OpsEng', 'EngOther', 'EngMgr', 'MgrOth', 'Other')
levels(surveydata$rolecombined) <- c('Dev', 'Ops', 'Mgr')
levels(surveydata$orghowmany) <- c('0-50', '51-100', '101-500', '501-2000', '2000+')
```

\(^1\)See https://www.rdocumentation.org/packages/base/versions/3.4.1/topics/levels
levels(surveydata$eventname) <- c('Postmortem', 'Retrospective', 'Learning review', 'After action report', 'Rapid improvement event', 'Other')

levels(surveydata$howmanypermonth) <- c('0', '1', '2-4', '5-8', '8+')
levels(surveydata$howlongafter) <- c('0-24h', '1-3d', '4-5d', '5+d')
levels(surveydata$hasfacilitator) <- c('Yes', 'No')

trueFalseQuestions <- c(
    "created_timeline_tf",
    "created_remediation_items_tf",
    "created_doc_updates_tf",
    "created_individual_narratives_tf",
    "created_analysis_technical_tf",
    "created_analysis_incidentresp_tf",
    "created_impact_biz_tf",
    "created_impact_cust_tf",
    "stored_ticketingsystem_tf",
    "stored_specialized_tool_tf",
    "stored_email_tf",
    "stored_wiki_tf",
    "stored_gdocs_tf",
    "stored_runbook_tf",
    "stored_paper_tf",
    "dist_not_distributed_tf",
    "dist_email_tf",
    "dist_chat_tf",
    "dist_standup_tf",
    "dist_newsletter_tf",
    "dist_blog_tf",
    "dist_custsupport_tf",
    "template_timeline_tf",
)
"template_remediation_items_tf",
"template_docupdates_tf",
"template_narratives_tf",
"template_analyses_tech_tf",
"template_analyses_incidentresp_tf",
"template_impact_biz_tf",
"template_impact_cust_tf"
)

for (ques in trueFalseQuestions) {
  levels(surveydata[[ques]]) <- c('No', 'Yes')
}

profileType <- c("rolecombined", "orghowmany")

testTypes <- c(
  "howmanypermonth",
  "howlongafter",
  "hasfacilitator",
  "created_timeline_tf",
  "created_remediation_items_tf",
  "created_doc_updates_tf",
  "created_individual_narratives_tf",
  "created_analysis_technical_tf",
  "created_analysis_incidentresp_tf",
  "created_impact_biz_tf",
  "created_impact_cust_tf",
  "stored_ticketingsystem_tf",
  "stored_specialized_tool_tf",
  "stored_email_tf",
  "stored_wiki_tf",
  "stored_gdocs_tf",
  "stored_runbook_tf",
  "stored_paper_tf",
  "itemsaccessible",
"one.lf/two.lf/zero.lf"
for (profile in profileType) {
  for (test in testTypes) {
    cat("----\n"
    cat(profile, " vs.", test, "\n")
    surveydataTestTable <- table(surveydata[[profile]],
      surveydata[[test]])

    result = tryCatch({
      testResult <- chisq.test(surveydataTestTable)
      print(testResult)
    }, warning = function(w) {
      #cat(message(w))
      cat("chisq test requirements not met; using simulated
        p-value with Monte Carlo simulation, 3 trials:\n"
      for (n in c(1,2,3)) {
        cat("Simulation ", n, ":\n", sep="")
      }
    })
  }
}
C.2 Contingency Tables

This R script was written to provide more insight into the hypotheses which showed a statistical correlation.

As above, the survey data is read in and human-readable labels are applied to the data variables. Then, for the significant job role hypotheses, the Chi-Squared Test is performed again, the subpopulation data counts are provided, the total answer distribution is provided, and then a contingency table showing the specific answer provided by each subpopulation is shown. These tables are re-created in Appendix B.2. A similar loop, executing the same operations is then performed for notable hypotheses that were correlated to organizational size.

#!/usr/bin/Rscript

surveydata <- read.csv("incident-reviews-no-other-role.csv",
colClasses= rep(’factor’, 2))

levels(surveydata$rolecombined) <- c(
  ’Developers’,
  ’Operations’,
  ’Manager’
)
levels(surveydata$orghowmany) <- c(
  ’0-50’,
  ’51-100’,
  ’101-500’,
  ’501-2000’,
  ’2001-5000’,

levels(surveydata$template) <- c('Yes', 'No')

levels(surveydata$itemsaccessible) <- c('Available - Known loc', 'Available - Unknown loc', 'Not Available')

trueFalseQuestions <- c("created_doc_updates_tf", "stored_specialized_tool_tf", "dist_chat_tf", "dist_blog_tf", "template_timeline_tf", "template_remediation_items_tf", "template_docupdates_tf", "template_impact_biz_tf", "template_impact_cust_tf")

for (ques in trueFalseQuestions) {
  levels(surveydata[[ques]]) <- c('No', 'Yes')
}

jobRoleTests <- c("created_doc_updates_tf", "template_docupdates_tf")

for (test in jobRoleTests) {
  cat("\n----\n")
  cat("rolecombined vs.", test, "\n")
  surveydataTestTable <- table(surveydata$rolecombined, surveydata[[test]])
orgSizeTests <- c( "stored_specialized_tool_tf", 
  "itemsaccessible", 
  "dist_chat_tf", 
  "dist_blog_tf", 
  "template", 
  "template_timeline_tf", 
  "template_remediation_items_tf", 
  "template_impact_biz_tf", 
  "template_impact_cust_tf"
)

for (test in orgSizeTests) {
  cat("\n----\n")
  cat("orghowmany vs.", test, "\n")
  surveydataTestTable <- table(surveydata$orghowmany, 
    surveydata[[test]])
  print(chisq.test(surveydataTestTable))

  cat("Subpopulation count\n")
  print(apply(surveydataTestTable, 1, sum))
  cat("Answer distribution\n")
  print(apply(surveydataTestTable, 2, sum))
  cat("Contingency table\n")
  print(surveydataTestTable / apply(surveydataTestTable, 1, sum))
}
Appendix D

Survey Artifacts Open Coding

The following table lists first cycle open coding of the artifacts received as part of the Phase I industry survey, specifically the optional question soliciting post-incident analysis event templates. The table is ordered by prevalence of specific template-element in the received templates, from greatest to least.

The artifacts from the Phase II organizational phenomenological and case study research were included for comparison. The A codes were from the initial open coding; the B codes were added during the second round of open coding.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Phase I Artifacts</th>
<th>Phase II Artifacts</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.25</td>
<td>Summary</td>
<td>19</td>
<td>6</td>
</tr>
<tr>
<td>A.26</td>
<td>Timeline - Basic</td>
<td>18</td>
<td>0</td>
</tr>
<tr>
<td>A.5</td>
<td>Action Items</td>
<td>16</td>
<td>1</td>
</tr>
<tr>
<td>A.17</td>
<td>Incident Metrics (TTD, TTR, etc.)</td>
<td>15</td>
<td>4</td>
</tr>
<tr>
<td>A.15</td>
<td>Impacts - Business / Technical</td>
<td>14</td>
<td>3</td>
</tr>
<tr>
<td>A.16</td>
<td>Impacts - Customer</td>
<td>14</td>
<td>5</td>
</tr>
<tr>
<td>A.21</td>
<td>Meeting guidelines/Definitions</td>
<td>13</td>
<td>1</td>
</tr>
<tr>
<td>B.2</td>
<td>Ticketing system links</td>
<td>13</td>
<td>0</td>
</tr>
<tr>
<td>A.18</td>
<td>Incident Response Roles/List of Responders</td>
<td>11</td>
<td>3</td>
</tr>
<tr>
<td>A.2</td>
<td>“Root Cause”</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>A.11</td>
<td>Description of “Actions Taken” (outside of timeline)</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>A.20</td>
<td>List of Corrective Actions/Remediations</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>A.1</td>
<td>“Contributing” Causes/Factors</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td>A.7</td>
<td>Appendix - Other</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
<td>Phase I Artifacts</td>
<td>Phase II Artifacts</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>A.28</td>
<td>Documents which specifically prompt a discussion of “preventing recurrence”</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>A.10</td>
<td>Current Status of Incident</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>A.19</td>
<td>Lessons Learned / “Takeaways”</td>
<td>6</td>
<td>3</td>
</tr>
<tr>
<td>A.22</td>
<td>Possible Improvements</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>A.23</td>
<td>Retrospective Objectives / “Invoking the space”</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>A.24</td>
<td>Severity</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>A.4</td>
<td>5 Whys</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>A.9</td>
<td>Author</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>A.27</td>
<td>Timeline - Detailed</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>B.1</td>
<td>Meeting Attendees</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>A.3</td>
<td>“Stabilization”</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>A.14</td>
<td>Findings - Technical</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>B.3</td>
<td>How Detected / “symptoms”</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>A.31</td>
<td>Templates de-emphasizing “why” in lieu of “how”</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>A.32</td>
<td>Templates discussing, some form an invocation of the space</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>A.6</td>
<td>Analysis (Round 1 only)</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>A.8</td>
<td>Appendix - Technical</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>A.29</td>
<td>Templates stating “there is no root cause,” but also referring to “Five Whys”</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>A.2.1</td>
<td>“No Root Cause”</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>A.13</td>
<td>Findings - Organizational</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>A.30</td>
<td>Templates referring to “luck”</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>A.33</td>
<td>Templates prompting a judgment of what is “important”</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>B.35</td>
<td>“Minimize impact”</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>A.12</td>
<td>Document version / Last modified</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Table D.1: Survey Template Artifacts Open Coding
Appendix E

Structured Interview Guide

The following guide was used by the investigator to conduct the structured interviews with the DevOpsCo informants. The guide is included to help contextualize the structure of the interviews. The first page is a visual representation of the intended path of the interview, while the second page is a list of questions the investigator intended the informants to answer during the course of the interview.

The investigator used the presented structured interview guide to both guide the interview and take notes during each individual informant's interview(s). These notes were later reviewed during the analysis phase.
Production
Retrieval
Storage

Who uses the artifacts?
What instigates the artifacts' retrieval?
Where are the artifacts stored?
Where do you retrieve the artifacts from?
When are the artifacts retrieved?
Why are the artifacts retrieved/used?
How are the artifacts used?

Organizational Learning (??)

How? Why?

Use

When?

What? Where?
1. Tell me how incidents are analyzed here.
2. What artifacts or output one out of the [post-incident analysis] process?
3. What are those artifacts stored?
4. What do you do with them?
5. Do you ever revisit them again?
   a. If so, when?
   b. What makes you do that?
   c. Where do you go to find them?
   d. Once you have them, how do you use them?
6. If you think of things as a lifecycle, do these artifacts feed back in any way into that lifecycle?
7. In your opinion, could your organization or team be using these artifacts in any other ways that would be useful for your work?
Appendix F

Structured Interview Open Coding

The raw thematic and topic codes for the open coding of the informant structured interviews follow, along with the observed counts for each operational cohort (Operations, Security, and Developers). Observed counts per operational cohort, to correct for the different sample sizes, is also provided.

Codes ending in X indicate markers in the interviews where informants describe or explain the ‘why’ or reasoning behind the associated code exists / is used / etc.

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Total Refs</th>
<th>Refs Per Ops</th>
<th>Refs Per Sec</th>
<th>Refs Per Dev</th>
<th>Total Refs Ops</th>
<th>Total Refs Sec</th>
<th>Total Refs Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Description of specific practices or mechanics</td>
<td>15</td>
<td>3.67</td>
<td>1.00</td>
<td>0.00</td>
<td>11</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>0.a</td>
<td>Retrospectives</td>
<td>38</td>
<td>7.00</td>
<td>1.25</td>
<td>2.40</td>
<td>21</td>
<td>5</td>
<td>12</td>
</tr>
<tr>
<td>0.a.X</td>
<td></td>
<td>26</td>
<td>4.33</td>
<td>0.25</td>
<td>2.40</td>
<td>13</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>0.b</td>
<td>Incident response</td>
<td>40</td>
<td>6.33</td>
<td>3.75</td>
<td>1.20</td>
<td>19</td>
<td>15</td>
<td>6</td>
</tr>
<tr>
<td>0.b.X</td>
<td></td>
<td>12</td>
<td>0.33</td>
<td>1.50</td>
<td>1.00</td>
<td>1</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>0.c</td>
<td>Incident analysis</td>
<td>31</td>
<td>5.67</td>
<td>3.00</td>
<td>0.40</td>
<td>17</td>
<td>12</td>
<td>2</td>
</tr>
<tr>
<td>0.c.X</td>
<td></td>
<td>17</td>
<td>2.00</td>
<td>0.75</td>
<td>1.60</td>
<td>6</td>
<td>3</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>Direct reference to an artifact</td>
<td>1</td>
<td>0.33</td>
<td>0.00</td>
<td>0.00</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
<td>Total Refs</td>
<td>Refs Per Ops</td>
<td>Refs Per Sec</td>
<td>Refs Per Dev</td>
<td>Total Refs Ops</td>
<td>Total Refs Sec</td>
<td>Total Refs Dev</td>
</tr>
<tr>
<td>-------</td>
<td>-------------------------------------------</td>
<td>------------</td>
<td>--------------</td>
<td>--------------</td>
<td>--------------</td>
<td>----------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>1.a</td>
<td>Jira Ticket</td>
<td>4</td>
<td>0.33</td>
<td>0.75</td>
<td>0.00</td>
<td>1</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>1.a.X</td>
<td></td>
<td>3</td>
<td>0.33</td>
<td>0.50</td>
<td>0.00</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>1.a.1</td>
<td>Jira Incident ticket</td>
<td>24</td>
<td>2.33</td>
<td>0.50</td>
<td>3.00</td>
<td>7</td>
<td>2</td>
<td>15</td>
</tr>
<tr>
<td>1.a.1.X</td>
<td></td>
<td>2</td>
<td>0.00</td>
<td>0.00</td>
<td>0.40</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>1.a.2</td>
<td>Non-incident Jira ticket</td>
<td>28</td>
<td>2.33</td>
<td>2.75</td>
<td>2.00</td>
<td>7</td>
<td>11</td>
<td>10</td>
</tr>
<tr>
<td>1.a.2.X</td>
<td></td>
<td>3</td>
<td>0.00</td>
<td>0.50</td>
<td>0.20</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>1.a.3</td>
<td>Jira Security ticket</td>
<td>13</td>
<td>0.00</td>
<td>3.25</td>
<td>0.00</td>
<td>0</td>
<td>13</td>
<td>0</td>
</tr>
<tr>
<td>1.a.3.X</td>
<td></td>
<td>2</td>
<td>0.00</td>
<td>0.50</td>
<td>0.00</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>1.b</td>
<td>Google doc</td>
<td>16</td>
<td>1.67</td>
<td>1.25</td>
<td>1.20</td>
<td>5</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>1.b.X</td>
<td></td>
<td>3</td>
<td>0.00</td>
<td>0.25</td>
<td>0.40</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1.b.1</td>
<td>Google doc - technical design doc</td>
<td>6</td>
<td>1.00</td>
<td>0.00</td>
<td>0.60</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>1.b.1.X</td>
<td></td>
<td>2</td>
<td>0.00</td>
<td>0.00</td>
<td>0.40</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>1.b.2</td>
<td>Google doc - Incident Analysis</td>
<td>13</td>
<td>1.67</td>
<td>0.50</td>
<td>1.20</td>
<td>5</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>1.b.2.X</td>
<td></td>
<td>5</td>
<td>0.00</td>
<td>0.25</td>
<td>0.80</td>
<td>0</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>1.b.3</td>
<td>Google doc - Timeline</td>
<td>14</td>
<td>2.67</td>
<td>1.25</td>
<td>0.20</td>
<td>8</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>1.b.3.X</td>
<td></td>
<td>3</td>
<td>1.00</td>
<td>0.00</td>
<td>0.00</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1.b.4</td>
<td>Google doc - Q &amp; A doc</td>
<td>11</td>
<td>2.67</td>
<td>0.75</td>
<td>0.00</td>
<td>8</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>1.b.4.X</td>
<td></td>
<td>3</td>
<td>0.67</td>
<td>0.25</td>
<td>0.00</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1.b.5</td>
<td>Google doc - Security postmortem doc</td>
<td>4</td>
<td>0.00</td>
<td>1.00</td>
<td>0.00</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>1.b.5.X</td>
<td></td>
<td>4</td>
<td>0.00</td>
<td>1.00</td>
<td>0.00</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>1.c</td>
<td>Metrics</td>
<td>10</td>
<td>1.00</td>
<td>0.00</td>
<td>1.40</td>
<td>3</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>1.c.X</td>
<td></td>
<td>1</td>
<td>0.00</td>
<td>0.00</td>
<td>0.20</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1.d</td>
<td>Template (any type)</td>
<td>11</td>
<td>1.00</td>
<td>1.75</td>
<td>0.20</td>
<td>3</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>1.e</td>
<td>Slack Channel (as a specific &amp; distinct artifact)</td>
<td>13</td>
<td>0.33</td>
<td>1.50</td>
<td>1.20</td>
<td>1</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>1.e.X</td>
<td></td>
<td>2</td>
<td>0.00</td>
<td>0.00</td>
<td>0.40</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
<td>Total Refs</td>
<td>Refs Per Ops</td>
<td>Refs Per Sec</td>
<td>Refs Per Dev</td>
<td>Total Refs Ops</td>
<td>Total Refs Sec</td>
<td>Total Refs Dev</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------</td>
<td>--------------</td>
<td>--------------</td>
<td>--------------</td>
<td>----------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>1.f</td>
<td>Google Group (as a specific &amp; distinct artifact)</td>
<td>2</td>
<td>0.00</td>
<td>0.50</td>
<td>0.00</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>1.z</td>
<td>Artifacts mentioned, not listed above</td>
<td>52</td>
<td>3.00</td>
<td>5.75</td>
<td>4.00</td>
<td>9</td>
<td>23</td>
<td>20</td>
</tr>
<tr>
<td>1.z.X</td>
<td></td>
<td>4</td>
<td>0.33</td>
<td>0.25</td>
<td>0.40</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

### 2 codes – Tools Used

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Total Refs</th>
<th>Refs Per Ops</th>
<th>Refs Per Sec</th>
<th>Refs Per Dev</th>
<th>Total Refs Ops</th>
<th>Total Refs Sec</th>
<th>Total Refs Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.a</td>
<td>Direct reference to tools used</td>
<td>1</td>
<td>0.33</td>
<td>0.00</td>
<td>0.00</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2.a.X</td>
<td></td>
<td>2</td>
<td>0.00</td>
<td>0.50</td>
<td>0.00</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2.b</td>
<td>Slack</td>
<td>22</td>
<td>1.67</td>
<td>2.50</td>
<td>1.40</td>
<td>5</td>
<td>10</td>
<td>7</td>
</tr>
<tr>
<td>2.b.X</td>
<td></td>
<td>4</td>
<td>0.00</td>
<td>1.00</td>
<td>0.00</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2.c</td>
<td>Google Docs</td>
<td>20</td>
<td>1.33</td>
<td>2.25</td>
<td>1.40</td>
<td>4</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>2.c.X</td>
<td></td>
<td>3</td>
<td>0.67</td>
<td>0.00</td>
<td>0.20</td>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2.d</td>
<td>Trello</td>
<td>8</td>
<td>0.33</td>
<td>0.00</td>
<td>1.40</td>
<td>1</td>
<td>0</td>
<td>7</td>
</tr>
<tr>
<td>2.d.X</td>
<td></td>
<td>1</td>
<td>0.00</td>
<td>0.00</td>
<td>0.20</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2.f</td>
<td>Google Groups</td>
<td>3</td>
<td>0.33</td>
<td>0.50</td>
<td>0.00</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2.f.X</td>
<td></td>
<td>3</td>
<td>0.00</td>
<td>0.75</td>
<td>0.00</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>2.z</td>
<td>Other tools</td>
<td>26</td>
<td>1.00</td>
<td>0.25</td>
<td>4.40</td>
<td>3</td>
<td>1</td>
<td>22</td>
</tr>
<tr>
<td>2.z.X</td>
<td></td>
<td>1</td>
<td>0.00</td>
<td>0.00</td>
<td>0.20</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

### 3 codes – Artifact Usage

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Total Refs</th>
<th>Refs Per Ops</th>
<th>Refs Per Sec</th>
<th>Refs Per Dev</th>
<th>Total Refs Ops</th>
<th>Total Refs Sec</th>
<th>Total Refs Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.a</td>
<td>References to ‘early on’/on-boarding uses of artifacts (especially ones that fall away after the actor has ‘spun up,’ organizationally)</td>
<td>16</td>
<td>2.00</td>
<td>1.25</td>
<td>1.00</td>
<td>6</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>3.a.X</td>
<td></td>
<td>2</td>
<td>0.67</td>
<td>0.00</td>
<td>0.00</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

132
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Total Refs</th>
<th>Refs Per Ops</th>
<th>Refs Per Sec</th>
<th>Refs Per Dev</th>
<th>Total Refs Ops</th>
<th>Total Refs Sec</th>
<th>Total Refs Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.b</td>
<td>References to trying to track/identify “trends” of security/organizational; References to use to manage organizational issues, i.e. growth issues and/or identifying “patterns”</td>
<td>58</td>
<td>9.00</td>
<td>5.50</td>
<td>1.80</td>
<td>27</td>
<td>22</td>
<td>9</td>
</tr>
<tr>
<td>3.b.X</td>
<td></td>
<td>6</td>
<td>1.00</td>
<td>0.50</td>
<td>0.20</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3.c</td>
<td>References to use of the artifacts for future (work) planning purposes</td>
<td>47</td>
<td>3.00</td>
<td>5.25</td>
<td>3.40</td>
<td>9</td>
<td>21</td>
<td>17</td>
</tr>
<tr>
<td>3.c.X</td>
<td></td>
<td>4</td>
<td>0.00</td>
<td>0.25</td>
<td>0.60</td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>3.c.1</td>
<td>References to use of the artifacts for, specifically, software design/requirements planning purposes</td>
<td>20</td>
<td>1.00</td>
<td>1.00</td>
<td>2.60</td>
<td>3</td>
<td>4</td>
<td>13</td>
</tr>
<tr>
<td>3.d</td>
<td>Specific reference to use and management of sustaining documentation, as related to incidents and incident reviews</td>
<td>23</td>
<td>2.00</td>
<td>3.50</td>
<td>0.60</td>
<td>6</td>
<td>14</td>
<td>3</td>
</tr>
<tr>
<td>3.e</td>
<td>Specific uses for “the business” (as opposed to developers/operations, etc.)</td>
<td>23</td>
<td>3.33</td>
<td>2.50</td>
<td>0.60</td>
<td>10</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
<td>Total Refs</td>
<td>Refs Per Ops</td>
<td>Refs Per Sec</td>
<td>Refs Per Dev</td>
<td>Total Refs Ops</td>
<td>Total Refs Sec</td>
<td>Total Refs Dev</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------</td>
<td>--------------</td>
<td>--------------</td>
<td>--------------</td>
<td>----------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>3.f</td>
<td>References to (enterprise) risk management/risk management decisions</td>
<td>20</td>
<td>3.33</td>
<td>2.00</td>
<td>0.40</td>
<td>10</td>
<td>8</td>
<td>2</td>
</tr>
<tr>
<td>3.f.X</td>
<td></td>
<td>2</td>
<td>0.00</td>
<td>0.50</td>
<td>0.00</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3.g</td>
<td>References to use to manage organizational issues, i.e. growth issues and/or identifying “patterns”</td>
<td>1</td>
<td>0.33</td>
<td>0.00</td>
<td>0.00</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3.h</td>
<td>Repeated references to specific practices/uses</td>
<td>58</td>
<td>4.67</td>
<td>5.50</td>
<td>4.40</td>
<td>14</td>
<td>22</td>
<td>22</td>
</tr>
<tr>
<td>3.h.X</td>
<td></td>
<td>16</td>
<td>1.00</td>
<td>0.75</td>
<td>2.00</td>
<td>3</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>3.h.1</td>
<td>Putting graphs/data into Google docs</td>
<td>9</td>
<td>1.33</td>
<td>0.25</td>
<td>0.80</td>
<td>4</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>3.h.1.X</td>
<td></td>
<td>1</td>
<td>0.00</td>
<td>0.00</td>
<td>0.20</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3.h.2</td>
<td>Use to prompt further analysis/research</td>
<td>26</td>
<td>3.67</td>
<td>3.50</td>
<td>0.20</td>
<td>11</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td>3.j</td>
<td>Reference to long term uses/uses beyond the incident itself</td>
<td>37</td>
<td>8.00</td>
<td>2.50</td>
<td>0.60</td>
<td>24</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>3.j.X</td>
<td></td>
<td>2</td>
<td>0.67</td>
<td>0.00</td>
<td>0.00</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3.k</td>
<td>Examples of reviewing docs further in the past; specific factors that drive people back to review/use artifacts</td>
<td>16</td>
<td>2.00</td>
<td>1.50</td>
<td>0.80</td>
<td>6</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>3.k.X</td>
<td></td>
<td>4</td>
<td>0.00</td>
<td>0.75</td>
<td>0.20</td>
<td>0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
<td>Total Refs</td>
<td>Refs Per Ops</td>
<td>Refs Per Sec</td>
<td>Refs Per Dev</td>
<td>Total Refs Ops</td>
<td>Total Refs Sec</td>
<td>Total Refs Dev</td>
</tr>
<tr>
<td>------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>------------</td>
<td>--------------</td>
<td>--------------</td>
<td>--------------</td>
<td>----------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>3.L</td>
<td>References to “vagueness,” “feelings” or a “a smell” as reason to use artifacts</td>
<td>8</td>
<td>0.67</td>
<td>0.25</td>
<td>1.00</td>
<td>2</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>3.L.X</td>
<td></td>
<td>3</td>
<td>0.33</td>
<td>0.25</td>
<td>0.20</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>“Social” use of artifacts</td>
<td>65</td>
<td>4.67</td>
<td>6.00</td>
<td>5.40</td>
<td>14</td>
<td>24</td>
<td>27</td>
</tr>
<tr>
<td>4.X</td>
<td>Reference to the concept of incident “memorialization”</td>
<td>2</td>
<td>0.33</td>
<td>0.00</td>
<td>0.20</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4.a</td>
<td>Reference to use of artifacts to gain/spread institutional/tribal knowledge; Direct references to/examples of “tribal knowledge” in practice</td>
<td>13</td>
<td>3.00</td>
<td>0.75</td>
<td>0.20</td>
<td>9</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>4.b</td>
<td>Reference to use of artifacts to gain/spread institutional/tribal knowledge; Direct references to/examples of “tribal knowledge” in practice</td>
<td>54</td>
<td>6.67</td>
<td>4.50</td>
<td>3.20</td>
<td>20</td>
<td>18</td>
<td>16</td>
</tr>
<tr>
<td>4.b.X</td>
<td></td>
<td>3</td>
<td>0.00</td>
<td>0.75</td>
<td>0.00</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>4.c</td>
<td>References to building/using/improving the “knowledge base” of the org or teams</td>
<td>30</td>
<td>5.33</td>
<td>2.75</td>
<td>0.60</td>
<td>16</td>
<td>11</td>
<td>3</td>
</tr>
</tbody>
</table>

4 codes – Social Usage of Artifacts
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
<th>Total Refs</th>
<th>Refs Per Ops</th>
<th>Refs Per Sec</th>
<th>Refs Per Dev</th>
<th>Total Refs Ops</th>
<th>Total Refs Sec</th>
<th>Total Refs Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.d</td>
<td>References to “storytelling” or related mechanisms by which stories are told, i.e. conference talks</td>
<td>41</td>
<td>6.67</td>
<td>3.25</td>
<td>1.60</td>
<td>20</td>
<td>13</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td><strong>Additional Thematic / Topic Codes</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Other potential uses of ideas to the artifacts that currently aren't implemented; direct reference to future/potential usage of an artifact</td>
<td>27</td>
<td>1.33</td>
<td>2.25</td>
<td>2.80</td>
<td>4</td>
<td>9</td>
<td>14</td>
</tr>
<tr>
<td>5.X</td>
<td></td>
<td>1</td>
<td>0.00</td>
<td>0.00</td>
<td>0.20</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5.a</td>
<td>Specific references to improved artifact usage and/or what's required/what makes improved usage difficult (seen this thematically a few times in answers to question 7).</td>
<td>12</td>
<td>0.33</td>
<td>1.25</td>
<td>1.20</td>
<td>1</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>The discussion of “resilience,” especially differences between how ops/core engineers refer to it and developers refer to it</td>
<td>14</td>
<td>1.00</td>
<td>0.00</td>
<td>2.20</td>
<td>3</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
<td>Total Refs</td>
<td>Refs Per Ops</td>
<td>Refs Per Sec</td>
<td>Refs Per Dev</td>
<td>Total Refs Ops</td>
<td>Total Refs Sec</td>
<td>Total Refs Dev</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------</td>
<td>--------------</td>
<td>--------------</td>
<td>--------------</td>
<td>----------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>6.a</td>
<td>References to “strong” vs. “weak” signals</td>
<td>9</td>
<td>1.33</td>
<td>0.00</td>
<td>1.00</td>
<td>4</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>References to providing “context”</td>
<td>47</td>
<td>4.67</td>
<td>3.00</td>
<td>4.20</td>
<td>14</td>
<td>12</td>
<td>21</td>
</tr>
<tr>
<td>8</td>
<td>Discussion of bucketing of incidents (how/differences across the cohort groups)</td>
<td>17</td>
<td>4.00</td>
<td>1.00</td>
<td>0.20</td>
<td>12</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>8.X</td>
<td>Situational examples of “exploring the discretionary space”</td>
<td>1</td>
<td>0.00</td>
<td>0.25</td>
<td>0.00</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>Situational examples of “exploring the discretionary space”</td>
<td>3</td>
<td>0.33</td>
<td>0.25</td>
<td>0.20</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>Discussion of/references to automation (either in the process itself or in general)</td>
<td>4</td>
<td>0.00</td>
<td>0.50</td>
<td>0.40</td>
<td>0</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>Metrics the org cares about (as related to incidents)</td>
<td>9</td>
<td>1.33</td>
<td>0.25</td>
<td>0.80</td>
<td>4</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>12</td>
<td>References to what constitutes “an incident”</td>
<td>5</td>
<td>0.00</td>
<td>1.00</td>
<td>0.20</td>
<td>0</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>12.X</td>
<td>A direct statement to the sentiment “no one person knows how the system works.”</td>
<td>2</td>
<td>0.00</td>
<td>0.50</td>
<td>0.00</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>A direct statement to the sentiment “no one person knows how the system works.”</td>
<td>10</td>
<td>1.33</td>
<td>0.00</td>
<td>1.20</td>
<td>4</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
<td>Total Refs</td>
<td>Refs Per Ops</td>
<td>Refs Per Sec</td>
<td>Refs Per Dev</td>
<td>Total Refs Ops</td>
<td>Total Refs Sec</td>
<td>Total Refs Dev</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------------------------------</td>
<td>------------</td>
<td>--------------</td>
<td>--------------</td>
<td>--------------</td>
<td>----------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>14</td>
<td>Specific reference to “blameless” approach</td>
<td>9</td>
<td>1.00</td>
<td>0.50</td>
<td>0.80</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>15</td>
<td>Specific references to external parties outside of the organization, but related to retrospectives/artifacts in some way</td>
<td>19</td>
<td>2.67</td>
<td>2.50</td>
<td>0.20</td>
<td>8</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>A reference to, specifically, “root cause”</td>
<td>18</td>
<td>0.67</td>
<td>1.25</td>
<td>2.20</td>
<td>2</td>
<td>5</td>
<td>11</td>
</tr>
<tr>
<td>99</td>
<td>“Interesting” / notable statements (marker to revisit)</td>
<td>129</td>
<td>14.33</td>
<td>9.50</td>
<td>9.60</td>
<td>43</td>
<td>38</td>
<td>48</td>
</tr>
</tbody>
</table>

Table F.1: Organizational Case Study Structured Interviews Open Coding
Appendix G

Inter-rater Reliability

Inter-rater reliability is an important component of any research in the social sciences. However, due to the nature and structure of the research presented above, conducting inter-rater reliability review proved challenging—though not impossible—for a couple of reasons:

• Fundamentally, the research represents two different modes: an industry survey and analysis of the results and artifacts provided and an in-depth organizational case study. Due to the differing nature of these investigations, there is little, if any, overlap in inter-rater reliability surveys, calculations, or results.

• The research protocol for the organizational case study required that the organization and informant identities remain completely anonymous; this necessitated the raw research (largely informant interviews, but also internal organizational artifacts) remain confidential as well, which means they could not be shared with inter-rater reliability participants directly. Because of this, only derived data was shared, which of course complicates both calculations and any conclusions.

G.1 Survey Artifact Coding Inter-rater Reliability

As discussed in section 4.2 and Appendix D, post-incident review event templates were coded for both structural and thematic elements.

Five inter-rater panelists—four with industry experience related to the research area and one with human factors/systems safety expertise—were asked to list structural elements of post-incident analysis templates from their industry ex-
perience. These structural elements were then compared for similarity to the open codes generated from review of the survey artifacts.

Because panelists could not be provided direct access to the survey template artifacts, agreement tests were affirmative, that is no penalty was assessed for identified elements which were not accounted for in the coding. The results were as follows, representing a range of “moderate” to “almost perfect” inter-rater agreement (Landis & Koch, 1977, p. 165):

<table>
<thead>
<tr>
<th>Panelist</th>
<th>Template Structural Elements Identified</th>
<th>Percent Agreement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13</td>
<td>53.9%</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>100.0%</td>
</tr>
<tr>
<td>3</td>
<td>13</td>
<td>61.5%</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>83.3%</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>80.0%</td>
</tr>
</tbody>
</table>

Table G.1: Survey Artifact Coding Inter-rater Reliability Panel Results

G.2 Interview Open Coding Inter-rater Reliability

In an attempt to account for the inability to provide direct access to the organizational case study research data, a ‘tiered-coding model’ was derived from the raw data to be presented to inter-rater reliability panelists for review. The idea behind the three tiers was to represent the progression and derivation of raw open codes found in the organizational data through to the analytical themes presented in the research.

In the first tier, a set of twenty-two open codes were selected related to the use of incident-related postmortem, retrospective, and analysis artifacts (taken from both the survey templates and organizational case study artifacts); panelists were asked to categorize these open codes into a set of eleven higher-order codes, representing a more thematic grouping of uses. In the second tier, panelists were asked to group the eleven thematic use codes into three analytical themes, derived from those presented in chapter 5. In the final chain of the model, panelists were presented with twelve informant quotations, and asked to place each quotation in one of the three thematic analytical categories provided in the second tier.
The inter-rater panel’s agreement results were as follows:

<table>
<thead>
<tr>
<th>Panelist</th>
<th>Tier 1 Agreement</th>
<th>Tier 2 Agreement</th>
<th>Tier 3 Agreement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>40.8%</td>
<td>50.0%</td>
<td>33.3%</td>
</tr>
<tr>
<td>2</td>
<td>52.1%</td>
<td>46.1%</td>
<td>41.7%</td>
</tr>
<tr>
<td>3</td>
<td>34.6%</td>
<td>47.8%</td>
<td>41.7%</td>
</tr>
<tr>
<td>4</td>
<td>35.1%</td>
<td>56.0%</td>
<td>58.3%</td>
</tr>
<tr>
<td>5</td>
<td>40.9%</td>
<td>57.7%</td>
<td>16.7%</td>
</tr>
</tbody>
</table>

Table G.2: Organizational Case Study Inter-rater Reliability Panel Results

There is obviously high variability in agreement results.\(^1\) This is likely attributable to the unfortunate fact that panelists were not able to review the case study artifacts or the informant interview transcripts directly, and thus were reviewing informant quotations not only out of context of the material, but out of context of the structure and format of the interview (which was not provided to them; see Appendix E for a sample of the structured interview guide). Therefore, they were basing their ratings on their own experiential contexts and providing meanings from their own backgrounds. This is not necessarily bad, but it would help to explain the inter-rater variability.

### G.3 Organizational Case Study Inter-rater Reliability

For the final inter-rater reliability assessment, two additional informants from DevOpsCo were provided with a draft copy of the thesis. These informants were not members of the research’s cohort of twelve informants described in section 4.3.3. After reading the draft, these two informants were asked a set of five questions, designed to elicit both convergence with and divergence from the thesis’ descriptions of DevOpsCo’s practices and organizational culture and their own organizational and operational experiences at DevOpsCo.

The goal of this inter-rater exercise was not to validate or compare/contrast the other informants’ perceptions, descriptions, and experiences, but rather to offer insight into (and hopefully validation of) the fidelity of the representation of the informants’ perceptions, descriptions, and experiences in this research.

\(^1\)Agreement ranges from ‘slight’ to ‘moderate.’ (Landis & Koch, 1977, p. 165)
These structured interviews with the two inter-rater reliability informants yielded three reportable inter-rater reliability findings:

1. Both informants reported that the descriptions and resulting analysis of DevOpsCo’s use of post-incident analysis artifacts were familiar to them, and were consistent with their experiences at DevOpsCo.

2. One informant noted that while they found the descriptions accurate, they work on another part of the product, slightly removed from the operations, security, and development teams from which the informants for this research were selected. As such, their team uses post-incident artifacts slightly differently, due to the nature of incidents they tend to experience. The informant described it thusly: “There’s emergent other kind of operational-like functions. The ops function is very localized to, like, they can create a culture within the DevOpsCo [System A] path, but there’s divergent cultures that can emerge, because they’re not being helped shaped by the Ops team.” While this informant describes a different experience with the artifacts on their team, they went on to remark “Your description applies to the section [of the company] which I would expect it to....”

3. The other informant noted that while the practices described herein were accurate, they reflect an earlier version of what DevOpsCo currently does with respect to the creation and subsequent use of at least some of the post-incident artifacts. Interestingly, they went on to describe a process which is a descendant of the “IRL” process OpsEng1 describes in section 5.5.2.
This is notable, because it reconfirms the case study observation of the evolution of how DevOpsCo creates and uses post-incident analysis artifacts, as described in section 5.5.2. It also demonstrates this evolution is sustained within DevOpsCo, at least in between the interval of the conclusion of the organizational case study and the inter-rater reliability interviews (about a year). Finally, it reinforces the confidence in the research, discussed in section 6.2, that the behaviors observed were not-for-show, and do represent how DevOpsCo goes about using post-incident analysis artifacts, even if those observations are only valid for a period in time. (It is heartening to find that DevOpsCo’s current practices are recognizable, even if distantly, and historically traceable to the practices and uses described within this research.)

Ultimately, the investigator considers these results promising with respect to providing a validation that the thesis’ description of DevOpsCo’s socio-technical system was as true as possible to the informants’ expressions and descriptions of it. The elements where each informant describes divergence from the details documented in the thesis offer testimony to the realities of the “nuanced, ‘messy’” organization any researcher would observe in practice, and the tall order of trying to fully, faithfully, and completely describe an evolving complex socio-technical system.