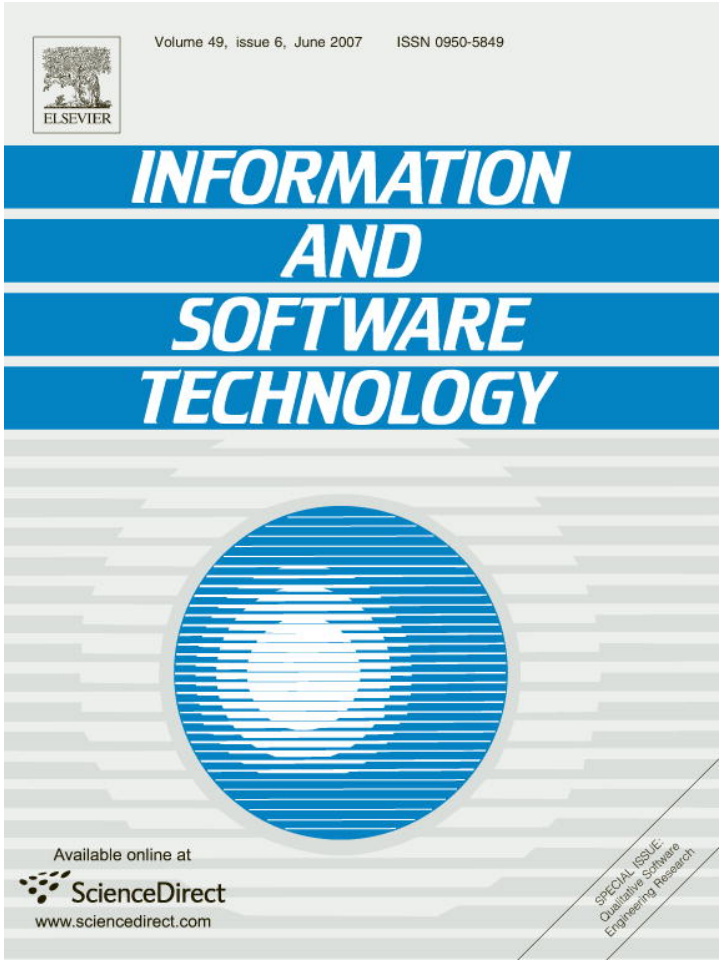

Available online at www.sciencedirect.com
ScienceDirect
Information and Software Technology 49 (2007) 576–587
INFORMATION AND SOFTWARE TECHNOLOGY
www.elsevier.com/locate/infsof

Revealing actual documentation usage in software maintenance through war stories

Wayne G. Lutters *, Carolyn B. Seaman

Department of Information Systems, UMBC, 1000 Hilltop Circle, Baltimore, MD 21250, USA

Available online 13 February 2007

Abstract

War stories are a form of qualitative data that capture informants' specific accounts of surmounting great challenges. The rich contextual detail afforded by this approach warrants its inclusion in the methodological arsenal of empirical software engineering research. We ground this assertion in an exemplar field study that examined the use of documentation in software maintenance environments. Specific examples are unpacked to reveal a depth of insight that would not have been possible using standard interviews. This afforded a better understanding of the complex relationship between project personnel and documentation, including individuals' roles as pointers, gatekeepers, or barriers to documentation.

Keywords: Software maintenance; Documentation; Storytelling; War stories; Qualitative research

1. Introduction

By nature, software engineering research is applied. Its agenda is driven by the real-world challenges of developing complex computational systems. Its core research questions emerge from this experience. Its greatest achievements are those that radically improve the actual work practices outside the walls of the laboratory.

In particular, empirical research in software engineering is dependent on practice for both its research problems and its data. From the field's inception, these researchers have maintained close ties to software engineering practitioners. This has kept their research grounded in and relevant to actual practice. One of the measures of the quality of this research is its practical enhancement of software design, development, testing, and maintenance processes.

Thus, empirical software engineering research is reliant upon data from actual projects for both the formative and summative evaluation of its understandings, models, and process improvements. These data have traditionally come from myriad sources, such as build statistics, instrumented software in beta-test, system interaction logs, user navigation patterns, software project management data, defect tracking, and configuration management systems.

There has been a demonstrated preference toward quantitative representation and analysis of these data from the field, routinely operationalizing key behavioral and environmental factors (e.g., as in metrics programs). While this has served the research community well, in the last decade more researchers have become aware of the limitations of this approach. There is a tradeoff between the computationally facile analysis and summarization of quantitative data and the contextual detail that is lost when complex behavioral phenomena are reduced to quantifiably tractable factors. Often it is these overlooked contextual factors, which fall outside the bounds of the resultant models, that prove essential to understanding the phenomena at hand.

Seeking to reintegrate this lost context, or pursue a holistic understanding from the start, a number of software engineering researchers have focused on qualitative methods of data collection and analysis, as discussed by Seaman [15]. Examples include Singer's studies of software main-

* Corresponding author. Tel.: +1 410 455 3941; fax: +1 410 455 1073.
E-mail addresses: lutters@umbc.edu (W.G. Lutters), cseaman@umbc.edu (C.B. Seaman).

0950-5849/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.infsof.2007.02.013

tainers [17], Rainer and Shepherd's longitudinal study on software project management [12], and Sharp and Robinson's study of extreme programming (XP) [16]. Some are even engaging interpretivist analytic approaches like Grounded Theory [18] (e.g., Carver's study of software inspections [2]).

One common form of qualitative data in these studies derives from interviews. Other forms come from document reviews, open-ended surveys, observations, and self-report mechanisms such as diaries. These data are analyzed via cycles of coding – tagging key words, phrases, and sections – and then examining the relationships and patterns between them. This heavily iterative process may involve deductive passes or inductive passes. Deductive passes validate concepts borrowed from the literature. Inductive passes generate new theory without external influence.

In this paper, we highlight one particular technique for qualitative data collection, "storytelling," and a specific story form, "the war story." We argue for its relevance in qualitative empirical software engineering research and the value of its inclusion in our research methods arsenal. We demonstrate its utility with a series of vignettes from our recent examination of documentation in software maintenance projects.

2. Examining war stories

This section defines and describes storytelling in general and the collection of "war stories" in particular. As storytelling is similar in many ways to other interview-based qualitative data collection techniques, we discuss these similarities, as well as some key differences. Finally, we step the reader through the mechanics of collecting war stories.

2.1. Storytelling: A data collection technique

Storytelling as a data collection technique refers to the process of eliciting data from study participants, in an interview setting, by asking for complete accounts of some event in the participant's experience. The most ancient of communication forms, storytelling is both familiar yet powerful. It forms the foundation for both the grand oral histories of populations and the Monday morning update around the office water cooler. This very fact that stories form the basis of human communication can be leveraged to better understand human behavior.

Storytelling is commonly employed throughout the social sciences as a scientific data collection technique, most notably in anthropology [11]. It is deeply rooted in the ethnographic tradition, which seeks to capture the emic perspective of study participants, that is their own perception of reality [3]. In doing so it preserves the participants' natural language and mental categories. It is holistic, paying attention to all contextual detail.

The advantages of storytelling are aptly summarized in this quote regarding the long interview (a type of storytelling employing an iterative approach that balances observation and interviewing):

> [This is] one of the most powerful of the methods in the qualitative armory… [it] can take us into the mental world of the individual, to glimpse the categories and logic by which he or she sees the world. It can also take us into the lifeworld of the individual, to see the content and pattern of daily experience. [7, p. 9]

That is, the storytelling process affords the researcher access to both the internal states and external environmental influences on the participant. The resultant rich contextual detail is the primary benefit.

Storytelling is also a common form of knowledge dissemination in the social sciences. Here the understandings gleaned from the collected stories during fieldwork are analyzed to inform a researcher-centered meta-story describing the phenomenon. VanMaanen [19] describes many of the literary forms these stories routinely take.

2.2. The War Story: A particularly informative story type

Just as there are many types of researcher tales, there are numerous types of stories to be collected, for example, life stories, process descriptions, and historical retrospectives. One particularly useful form of story is the "war story." This is typically a lengthy tale recounting a particularly momentous event that involves overcoming some type of difficulty through great effort. (Hence, the analogy to tales of harrowing battlefield experiences.) These stories are rich in detail and are told in a manner such that the participant is the protagonist or central character.

Within the information systems literature, war stories have proven a valuable resource for eliciting design requirements and understanding system adoption patterns. In his classic "Talking About Machines" Orr [10] collected war stories told amongst copier repair technicians as part of an ethnographic study of mobile knowledge workers. The insights gained from these contextually rich narratives informed a deep understanding of the relationship between technicians, customers, and the machines themselves. This understanding was then applicable to a host of information systems issues ranging from parts distribution systems to documentation creation to effective knowledge management structures for semi-autonomous mobile workers. The first author has examined the role war stories play in preserving contextual detail outside of formal knowledge management systems (e.g., [5,6]).

2.3. Comparison with other qualitative data collection approaches

The toolbox of qualitative data collection techniques is diverse. As each particular approach has well documented benefits and limitations, the common wisdom in study design is to balance the methods, to use more than one technique, and to triangulate among multiple independent

data sources. In selecting a particular technique, the researcher must clearly understand its strengths and weaknesses to reason thoughtfully about its fit within an overarching study design. To assist the reader, we will briefly position "war story elicitation" with regard to standard interviewing approaches.

The process of collecting war stories via storytelling shares much in common with interviewing. For starters, population sampling is similar in each. As each is a resource intensive process, the number of possible participants is limited. This requires that the researcher take a holistic approach where the study design shifts from attempting to control environmental complexity toward maximizing it instead. Diversity in sampling the study population is desired as it amplifies the contextual detail received.

Both approaches are rich self-report mechanisms that seek to preserve the participant's viewpoint. They are flexible in structure and position the researcher primarily in the role of the listener. Yet, the two methods are also quite different, most notably in terms of the robust capture of native categories and context.

Standard structured or semi-structured interviews typically rely upon the researcher's categorization scheme of the phenomena under study. Open, or long, interviews employ more of a conversational style of data collection, where categories can be co-constructed and participants have more freedom to travel outside the bounds dictated by the researcher's protocol. Storytelling takes this to an extreme, where the participant is in full control of the categorization schemes.

Standard interviewing approaches tend to elicit post hoc rationalizations about phenomena from participants. While these generalizations are useful, they have reduced contextual detail. Although storytelling also reduces contextual detail somewhat (at least as compared to observation), it does so to a lesser extent. Importantly, it is the participant who decides what details to include and what to omit in the service of telling the story. Good storytellers tailor their stories to their audience, so the framing of the questions by the researcher and the sensitizing concepts driving the probing are critical.

One common interviewing strategy to accomplish this level of focused response is the use of critical incidents – specific events in the interviewee's experience, often a major breakdown in their routine work practice. These are presented to the participant as a prompt or trigger intended to focus their thinking and prime their response. Storytelling is in many ways similar to this critical incident technique, and has many of the same advantages of grounding the participant's response. However, storytelling goes further, allowing the participant to determine for themselves what incidents are critical (i.e., what stories are important) and what is critical about them. Another important difference between storytelling and critical incident prompting is the emphasis on truthfulness present in the latter approach. The accuracy of a story told during a storytelling session is not as important as its ability to capture the participant's perception of how that story played out and why it was worth telling.

The methods of analysis for war story data are similar to those for any other qualitative data. Detailed field notes are taken during the interviews and the dialogue is audio recorded and transcribed. The resultant textual data are typically content coded, which involves labeling passages in the data to mark recurring themes or topics. Passages coded with the same theme are then combined to form a richer understanding of the theme [9]. We typically use a Grounded Theory approach [18], in which we build theory from emergent relationships among the data. In this way, we form new understandings rather than use collected data to validate pre-existing hypotheses. Theory generation is often the most appropriate study design given the novelty of our research objectives.

In analysis, the additional context of the war story gives the researcher more "handles" into the mental and environmental states in which the story unfolds. Simply, this provides more meaningful text for the researcher to code in the analysis process, increasing the likelihood of identifying unexpected categories, uncovering novel relationships between concepts, and discovering detail about the conditions under which certain propositions are valid. This is especially important for exploratory work, when explanations of phenomena are preliminary and still ill-formed.

Having more contextual detail in the data also helps avoid researcher bias, because there are fewer "holes" for the researcher to fill in. That is, in more generalized accounts (e.g., from a semi-structured interview), the researcher is tempted to theorize about what contextual conditions might affect the validity or applicability of a finding. By using war stories (or any type of storytelling), the relevant context is provided concretely, so there is neither a need nor an opportunity for the researcher to make such assumptions. Bias through "leading" questions from the interviewer is also avoided when using storytelling as they are relegated to the more passive role of listener compared to standard interviewing.

2.4. War story elicitation procedure

The process of war story elicitation is similar to that of the open (or long) interview. The procedure begins with the researcher establishing a state conducive for the participant's memory recall. Typically this involves generic remembering prompts such as *What was the last project you worked on?* or *What did you do last week?* Then the researcher asks a very open-ended question and requests a story in response: *Can you tell me about a time when…?* Ideally, the participant then recalls and recounts a relevant story. The researcher may use neutral prompts to continue the elicitation if it falters (*Explain further*, *What happened next?*). When the story is completed, the researcher may engage in a collaborative unpacking of the story with the participant (*So if I understand correctly, this story started when…*).

One key variation in this method involves the location of the interview. Some argue that it is best to remove a participant from the distractions of their routine environment, for example, by relocating from a busy office to a quiet conference room. This allows them to focus solely on developing their story. We believe that it is critical to interview in the participant's natural setting, where we can encourage them to identify items in their environment and demonstrate practices while recounting their stories. In fact, "props", or objects in the participant's environment (e.g., a book on their bookshelf, a photo on their desk, or a poster on their wall), are often helpful for generating deep and engaging stories. This additional component of focused, interaction-driven observation becomes a critical balance to the stories themselves.

3. Example study – Documentation in software maintenance

In order to demonstrate the particular utility of war stories in empirical software engineering research, we have chosen one of our ongoing field studies. This study examines the known challenge of effective documentation use in software maintenance activities. We will briefly introduce the study here and then walk the reader though an illustrative collection of actual story distillations in the next section.

3.1. Motivation

Prior survey work revealed the diversity of information resources used in software maintenance activities [14]. It found that many of the artifacts created during the design phase specifically to support maintainers were of little use when actually needed. It also pointed to the need for better repositories of information and experience for maintainers, by which the search for relevant, task-specific information could be facilitated. Work by Rodriguez et al. [13] has also addressed the problem of knowledge management in software maintenance, but did not focus on documentation as a form of experience. Lethbridge et al. [4], in a series of empirical studies of various types, made several discoveries about documentation use in software maintenance, including a correlation between the use of some kinds of documents and how up-to-date they were kept.

This current study was designed to extend and deepen this prior work by gathering additional contextual detail about specific cases of documentation use in maintenance activities. Its intent was to inform the design of a maintenance experience repository (possibly analogous to the lessons learned repository described in [1]), its contents, and an approach to contextualizing work artifacts for inclusion in the repository.

3.2. Site selection

In selecting field sites for this data collection activity we attempted to sample for maximum diversity, as this would increase the transferability of our findings. However, we did have some key constraints. First, the software maintainers must have been part of at least a five member team. Issues of information flow across time become trivial with one or two individuals. Second, this maintenance group must have been distinct from software developers. Many organizations have a fluid assignment of human resources between development and maintenance projects, while others have distinct divisions. Organizations with the latter were the focus of our study as specific maintenance cultures (with their own "war story" lore) were likely have more mature artifact identification, selection, and contextualization strategies.

We currently have six participating organizations in the ongoing study, all at various stages of data collection and analysis. Data collection has been completed at three of these organizations, and the data analyzed from two. The results discussed here as examples of the storytelling approach are based on this analysis. These two organizations are Fortune 500 multi-national software development and consulting companies. The groups studied within these two companies were responsible for the maintenance of software supporting operational legacy satellites for a government customer. In each organization we interviewed the manager of the software maintenance team and all, except one, of the software maintainers yielding sixteen total participants ($N = 16$). The other organizations participating in this study include smaller private sector firms and some internal software maintenance groups, which will greatly diversify the final data set.

3.3. Study design

The centerpiece of this study was the collection and analysis of "war stories" (following methods as described in Section 2 above) that documented detailed instances of both successful and unsuccessful information gathering and use. (The full interview guide can be found in the Appendix A.) We used the following four main story prompts to approach the documentation issue from different angles and outcomes:

Prompt #1:Could you tell me about a time when you had trouble finding information you needed to perform a particular maintenance task?

Prompt #2:Could you tell me a time when your project was saved by a document you found?

Prompt #3:Could you tell me about a time when you found exactly the information you were looking for in a place you did not expect? Where and why did not you expect that?

Prompt #4:Has there ever been a time when you had plenty of documentation available, but none of it was useful? Why was none of it useful?

Following the storytelling, we collaboratively unpacked the stories via a more traditional semi-structured interview

with participants drilling down on general aspects of behaviors identified in their war stories.

3.4. Study implementation

War story elicitation is difficult – the tendency was for participants to generalize and it was challenging to identify the triggers that would foster their recall of more specific detail. Indeed, not all of our prompts were successful in drawing out the stories that we had anticipated. Sometime they would languish in the general, but other times they would kick-start stories that appeared at first to be tangential. Frequently at the end of these stories participants would wrap back around and ask us "so, did that answer your question?" Not surprisingly these tangential stories were often the most data rich and instructive stories of the session.

During the telling of the story, the participant had control of the floor and the interviewer was largely an active listener – occasionally offering neutral prompts such as "tell me more," and "could you expand on that?" This was a role reversal in conversational control from most traditional approaches to interviewing and was difficult to manage at first for our research team. The natural tendency was to cut in and direct their stories (e.g., keep them from wandering off too far on a topic) or interrupt their narrative with our own questions. With practice we became more adept at being active listeners, enveloped in their tales.

We relied on a Grounded Theory approach (described in Section 2.3 above) for the analysis of these data. Although there is prior work that suggests the usefulness of experience repositories for maintenance [4,12,13], there is a paucity of literature specifically addressing what maintainers need from such a repository, in terms of content, organization, and contextual information. Thus, it was important that the stories we collected be fully analyzed without bias from previous, ungrounded perceptions of what works and what does not. The Grounded Theory approach allowed us to do that.

As mentioned in Section 2.3, the data analysis of the war story data followed typical procedures for coding and examining passages for trends and themes. The codes used in the analysis were post-formed and were derived from the early sets of data. The two authors worked together in collaboration with a student assistant, coding interview transcripts independently and then meeting to resolve differences until we reached a stable set of categories and codes.

Fig. 1 shows a summary of the coding scheme that was derived. The first codes to be created had to do with topics related specifically to documents (e.g., characteristics, quality, properties, and missing). With some reflection, however, we discovered the importance of the parts of stories that referred the human role in documentation use, which we coded with "Human sources of information." Also, we determined that the location of documentation, and how it was found, was a crucial element in many of the stories, so we created a code to capture this. The last two codes, "Story" and "Great quotes" were added as an aid in writing, to help locate specific pieces of data that we wanted to reference.

4. Example results

Our results thus far include such findings as:

- Human sources of information are an important substitute when documentation is not available, and word of

Respondent background – pre-maintenance, educational experience
Respondent maintenance experience
Transition to maintenance –the process of transitioning a product from development to maintenance
Conditions of maintenance environment
Configuration management
Types of documentation
Characteristics of documentation
Quality of documentation –how good or bad a particular document or set of documents is, not what makes it good or bad; general for projects, high level
Properties of documentation – characteristics or structural elements of documentation, e.g. TOCs, indices, tables, appendices, that affect how effective the documentation is
Missing documentation – documentation that does not exist
Creating documentation – comments about the circumstances of how some document is created
Location of documentation – where, physically, documentation is kept
Importance of documentation – comments about the importance of different types of documents and what affects the relative importance of documents
Use of documentation – comments about how a document is used
Volume of documentation – comments about how much documentation there is
Human sources of information – situations or circumstances (specific or general) when the needed information came from a human, or how human information is gained
Quality of process – comments about how good or bad a development or maintenance process is
Configuration management
Management influence – what role managers play in the creation or use of documentation
Tools – any mention of use of tools
Story – story to be referred to later
Great quotes – great quote to be referred to later

Fig. 1. Coding scheme (abridged).

mouth is relied on as a method for finding knowledgeable people.
- Structural properties of documentation (e.g., tables of contents, indices) have a substantial effect on the ability of maintainers to make use of it.
- Documentation that is written from the perspective of a maintainer (sometimes even written by a maintainer) is especially useful.
- The location of documentation is a relevant issue, as it is frequently the case that the needed information is contained in (or is believed to be contained in) available documentation, but cannot be found.
- There are times when unofficial or informal documentation (a developer's personal notes, emails, etc.) becomes the saving factor when facing a difficult maintenance task.

In this article, our objective is to present a substantial subset of the study's findings in order to illustrate how the war stories approach yielded additional insights and added richness to the findings, beyond what would have been gleaned through a traditional interview-based study.

As noted above, our findings have shown that the location of documentation is an important issue. We found that, in some instances, the needed information was readily available, but was not identifiable because it was buried in voluminous documentation. In other instances, the physical document itself could not be located. And in still other cases, the maintainer had ready access to the document he/she needed, but he/she did not know it existed. We had not specifically included questions about document location in the interview design because our previous work had not indicated this as a significant factor. Consequently, our discovery of document location as a relevant issue is an example of emergent, unexpected theory. This is one of the advantages and deep satisfactions of conducting any type of qualitative research.

Our basic findings on document location could have been discovered through a traditional semi-structured interview study. But because of the war stories methodology that we employed, we were able to add richness and complexity to these findings that would not have been possible otherwise. To illustrate this point, the following subsections describe three of these additional insights and how we discovered them.

4.1. People as providers of and pointers to documents

In the following passage from an interview transcript, participant P10 was responding to our storytelling prompt #3 (as listed in Section 3.3), about finding information in unexpected places. The maintenance task being described involved shifting a satellite's position by 15°, so that solar radiation did not heat up its sensors too quickly. Yet, they had to trick the onboard system into thinking that the shift had not occurred, so that other location dependent calculations would not be affected.

P10: *…And so then to do that is simple, but to test out all the repercussions of doing that is pretty complicated and so we had gotten pretty far through the testing and somebody said "Well did you check for stray light, any kind of stray light that might cause bad readings on the sensors?" And nobody had. We didn't know how we would do that. Somebody searched through his whole documentation files. This is one of the [satellite] engineers who sent over some drawings and then I got out rules and compasses and checked all the limits that they had for the sensors to make sure it wasn't admitting stray light.*

Interviewer: *This clearly is not sitting in the user's manuals or Appendix A, but how were you able to find this particular engineer to work with?*

P10: *Well, the first time we met him was in the meeting where they were going to check over our test results. …And then [as] we are going over these results people start thinking, well, maybe we didn't consider all the possibilities and so the testing got expanded more and more. And there were things like that where he remembered emails or various things he would send over, old emails that had hints about how to proceed with the testing and what the impact might be.*

Had we asked a more straightforward, interview-type of question, such as *What are some of the unexpected places where you find needed information?* we might very well have learned that sometimes engineers pass along emails and other types of informal documentation in certain situations. But the added insight we gained here, because it was imparted as part of the context of a story, was that this particular engineer was very familiar with the problem the maintainers were addressing because he or she was present in the meeting where the issue was first raised. Thus, the insight that we gained from this exchange was that sometimes non-maintenance personnel (in this case an engineer) can provide documentation that is otherwise unavailable, and whose location is unknown, when they understand the maintenance problem well.

This led us to investigate the role of colleagues and other people in the maintenance environment, not as human sources of information that are an alternative to documentation, but as sources of, and pointers to, written information. (In studies of collaborative problem solving these individuals who know "who knows" are commonly referred to as "expertise concierges" [8].)

In another passage from an interview transcript, respondent P11 was reacting to prompt #2 about a piece of documentation that "saved" the project or maintenance task.

P11: *Well, we did pretty well with the [satellite] mission in getting everything that the development project had. I think that part of that was the fact that I was involved prior to launch so I was interacting with the development team directly. So we had quite complete documentation and I had all the documentation for how they burn software into EEPROM in the lab and so forth, which is again*

useful to have. Although, once we are in maintenance we don't usually do that, but it's good to have in place. And I did actually use it once testing a contingency change (not intended to be used in the normal course of business) that was a patch that we developed for [the satellite] to determine the efficiency of a particular hardware failure that the spacecraft was not designed to handle. It involved essentially a rewrite of the attitude control system. So it was a large software change and it was useful early in testing to be able to load it like a software build on the ground so you could see if it could run without worrying about the correctness of the additional steps that are involved in loading on orbit, which involves ground system, telemetry, and so on…so there is a lot of intervening stuff. So it's much more complicated to do the change on orbit than it is to [try it out on the ground]. So it was useful to be able to do that. We had to verify that we could do it the other way and we did, but as a first whack at the problem it was useful to be in with us, just kind of put it in there and verify that just as a piece of software running on the board it was okay and then we could worry about all the rest.

In summary, the story is about a maintenance task where it was useful to test the software change on the ground by burning it into EEPROM in the lab, without actually loading it onboard the orbiting satellite itself. The team was able to do this only because P11 happened to have the documentation on burning code into EEPROM. From this passage, we gained an example of an unexpectedly important piece of documentation. As in the previous example, we might very well have gleaned this from a standard interview question such as, *What types of documentation are real project-savers?* In this case, the additional insight that came with the story context was that this crucial piece of documentation (the EEPROM-burning instructions) was available because the maintainer happened to be involved in the original development process.

These two examples helped to shape our understanding of how people in the maintenance environment play a role in providing and locating needed documentation. We found that maintainers often turn to people (especially colleagues) not just to provide information that they cannot find in documentation, but also to provide documentation, or to provide the location of a document. We also found instances where the location, or even existence, of the needed documentation was dependent on personal circumstances (e.g., a maintainer who had been involved in development or an engineer who had discovered a problem that was then turned over to some software maintainers).

4.2. Control over documentation

The next examples show the downside of the fact that documentation is sometimes available or locatable only through people. The issue of control over access to documentation came up in several instances. In the following excerpt, respondent P05 tells a story where control over information by an outside entity created an insurmountable barrier to access.

P05: *Well, I'm doing it right now, which is why I'm sitting here with this pseudocode [points to large pile of documentation next to him]. This is the [name] satellite and I'm creating a PC-based simulator. …How does the data get out of the RAM? Is it pulled or pushed? There's nothing in the documents that I've been able to find that tells me what signals the code to send the data. Because the encoder has to know what the beginning and the end of the frame are before it can do its error checking and so on. You have to have some kind of sync pattern. I don't know any of this and it's not in any of the documentation I have. So I worked my way through the [personnel monitoring the satellite] and finally got to Bob [a pseudonym] and said 'Bob, you have to call the people at [the satellite manufacturer] and ask them this information.' So we called the people at [the satellite manufacturer] – maybe they have some documentation we don't have – and they said they could give us this information for a cost. There isn't any money. So I hit my last chance at finding somebody who actually knew how it worked and they're not available. So I'm going through this pseudocode…Long, tedious, serious pain the buns. It's going to cost me probably weeks maybe up to a couple months to figure this problem out, whereas if I had someone who knew how it worked, it would cost me maybe a day. But that's the situation.*

This tale was in response to prompt #1, asking for a story about lack of access. It gives an example of an outside entity who tries to exact a price (in this case monetary) for access to needed documentation. The price was too high so there was no access and a different, more time-consuming, approach had to be employed.

It is possible that this entire story would have been missed in a standard interview, in response to a question like, *What do you do when you have trouble finding the documentation that you needed to get the job done?* If it were even mentioned, the emphasis would probably have been on the fact that the maintainer was forced to spend a lot of time combing through pseudocode and the reasons for the lack of access to the documentation might very well have been left out. But in our study, we gained an important example of the types of barriers to documentation and information that can exist when others have control over those resources.

Another type of barrier, related to control over documentation, is illustrated in the following passage, also from participant P05:

P05: *…but what happens is that a problem comes up and they say, [P05], the solid state recorder, you modeled with it fidelity B and we need it with fidelity A, and so I have to rewrite the model. …And so I had to go back and learn the solid state recorder in much more depth. And do it quickly. And I repeatedly had to go to the [personnel*

monitoring the satellite] first, and they would refer me to the spacecraft engineers at [a non-US space agency]. [There was] actually a secret locked room with documents in it that I was blindfolded and taken into [laughs]. …It's locked because these are the only copies, and they're very, the [non-US space agency] people have tighter security restrictions and they just didn't want to take any chances that these would walk away. Because some documents get lost. People borrow them and they never come back. So you have to be careful about that.

This passage constituted a tangent in the interview conversation, so it was not directly in response to any particular storytelling prompt. One might argue, intuitively, that the storytelling approach to interviewing would encourage such tangents. At the very least, our approach created an atmosphere that was friendly to telling stories, once it was clear that was what we were seeking.

This second example of a barrier to information established by an outside party is important because it is different in nature from the first example. Here, the motive for such barriers is not self-gain, but a more altruistic concern for the safety of the documents themselves. A less positive experience is recounted below. This story was recounted to us by P04 initially in response to prompt #1.

P04: *All right, a year and a half ago when I transferred over to [a project], that was when [there was a transition in responsibility for maintaining a related set of satellites from one maintenance group to another]. There were different satellites that apparently the code base was the same when they developed them, so they kind of maintained them collectively. And we were told to start looking at some of the bugs that were there and work on fixes for it. In order to find the right source code, we had to try and find where the last release package of software was. The machines upstairs that we had [that had been housed with the previous maintenance group] were transferred over here and they were just kind of turned on to make sure that they worked, but no one ever cataloged what was on them. So we had no clue where the software was in the machines that we had. We had no clue whether we had all the correct machines. The library that they keep downstairs of the last release software wasn't up-to-date and there were multiple versions in there and in the reversions that were in there that were never officially released. …We had documents that were turned over from [the previous maintenance group] that were build instructions from two versions ago before they had transitioned to a different build tool. All of the User Manuals would conveniently disappear. …Not uncommon apparently when a transition goes on for half of the engineers to go on vacation and they take the documentation with them. So, yeah we had fun trying to find the key information for how to build – I mean even to just build the software. There were no 'read me' files, there were no documents on there that were current, on how to compile it. …Similar stories for the whole of the rest of the project.*

Interviewer: *Prompt #2*

P04: *Yeah, same project. In one of the other developer's accounts we found a text file that was her notes on how to build the system and that gave us 90% of what we needed to actually get it to build and if we hadn't found that document – we could have spent another month trying to figure it out.*

This story had two parts, each of which was in response to one of our storytelling prompts. Had we asked, as we would have in a traditional interview, about what strategies are employed when it is hard to get documentation, and what types of documentation tend to be really crucial in saving a project or a task, we might have heard elements of both parts of this story, but would probably not have heard how they related to each other. What would have been missing is the reason that the developers' notes were so crucial in this situation, which is why the rest of the documentation was unavailable. The other interesting aspect of the story is that this small piece of documentation that made it through the barrier created by the other maintenance group, quite possibly by accident, gave an indication of the value of the documentation that remained inaccessible.

These stories about documentation being inaccessible through barriers erected by outside entities (i.e., the demand for payment, or the locked room, or the "disappearance" of documents), although each is an isolated example, together give us a rich picture of how people in the maintenance environment can play a negative role in identifying and gaining access to documentation.

4.3. The value of a gatekeeper

Another insight into the role of people in locating and identifying documentation came through a series of stories told to us by different informants, but that turned out to be related. Because these data were in the form of stories, we had enough contextual information to identify them as related.

Early on, we identified an important human information source through a recommendation from the inaugural meeting with one of the group managers. Participant P09 was described as someone who had experience across multiple projects and who, although semi-retired and remotely located, was still playing a significant role in the maintenance team as a consultant. When we interviewed P09, two illuminating examples were revealed: her personal notes and her list of contacts. They are described in the passages below. The first describes P09's personal notebook:

P09: *Yes, there's something firing off way in the back of my brain and why don't you go check on this and then I send them off looking for something and then I'll go looking through my notebook and – of things that I wrote down and somehow I'll know kind of where to look in the notebook, because I kind of know what time frame it was five years ago, so I'll look through my notebook and find the*

commands that we sent. And find a note beside them that says, write these down in the work request so that the next time this happens they'll be – and it wasn't crossed off, so this is something that got lost on the wayside, and I happened to be still around and could remember – and of course by that time the [personnel operating the satellite] has also turned over and you can imagine how much documentation they have lying around their control center.

The second document mentioned by P09 was her list of contacts, described below:

P09: *Testing – yeah, we actually go down to the simulator and we'll bring up the software and we'll put the spacecraft in this specific configuration and then we'll just sit and watch the telemetry and understand how it works that way. So we'll look at the software and we'll look at – we'll just watch simulations or tests run. Or go over to the actual spacecraft in the control center and see what's happening there or talk to an engineer.*

Interviewer: *And how do you wind up finding those folks?*

P09: *The engineers?*

Interviewer: *Yeah.*

P09: *When we receive software – when we receive a mission to maintain, we make sure we know who was the lead for every piece of software and every piece of hardware that we're responsible for or that we'll have interaction with. And we get their name and their phone number.*

Interviewer: *Good, so it's a local hit list that you maintain?*

P09: *Yeah. And if we're missing somebody, [the personnel monitoring the satellite] usually has a good list too. And nowadays since we start working on the mission early as developers or testers we get to know the folks who actually built the stuff.*

Clearly this maintainer's notebook and her list of contacts were important pieces of documentation for her. However, this might have seemed to be an isolated case. Later, during two other interviews, we found references to these documents again. This first passage was in response to prompt #3 about finding information in unexpected places:

P10: *Well, we had a problem, and this was a hardware problem in our simulator where we came in to do something and nothing would happen. Well, not nothing but [it would] not properly initialize and I tried everything I could think of and then somebody else came in and tried everything they could think of and then [P09], who you talked to before, she was [working on a different satellite] in those days under development. She remembered an electrical engineer's name and I tracked him down and called him up and he came over and looked at things and within fifteen minutes he found a box that had a fuse blown. Well, it looked as though it was actually more complicated than that, but at any rate he tracked down the box and another guy came over and did some soldering and fixed it up.*

The second passage, below, constitutes the end of a story in response to prompt #1, asking for a story about having trouble finding needed documentation. The full story was about the investigation of a hardware failure in the maintenance testbed. The particular passage below was in response to the interviewer following up by asking how they managed to find the right engineer to help them with the hardware failure.

P11: *Well again, it was sort of oral tradition and word of mouth. I called a colleague who isn't at this site anymore, but was the test lead on the project and I said, 'Who built this box?' And she said, 'Well it was this guy.' So I contacted him and said 'Help!' and he came and bailed us out. So we were sort of reliant on oral tradition here, which is not quite the best way to do things, but in this case it worked. But you don't want to be reliant on calling up somebody and saying, 'Do you remember who it was that did this because I really need to talk to him…'*

Interviewer: *And just out of curiosity, that didn't happen to be [P09], did it?*

P11: *Yes, it was [P09]. And she gave me the name of the guy who actually built the box. And he brought in some other people who had worked on the integration and test of the hardware and who have been extremely helpful diagnosing the problem. So that's an example of something we didn't have. We didn't have the instructions, which would have been helpful. It all turned out okay, but could have saved us a little heartburn if we had the information up front.*

These two passages not only validated the fact that P09's list of contacts (and presumably her memory and her notebook as aids in navigating that list) was useful, but also gave us more insight into how it was useful. It had become a group asset, but only through the gatekeeper of P09. Organizational resources were allocated to keep that source, and its gatekeeper, available to the team. As before, these details constituted part of the context of larger stories and so would probably not have been mentioned in a regular interview setting.

In summary, clearly the location of documentation is an important issue in maintenance and in maintainers' ability to gain the information they need to perform maintenance tasks. Further, the story-based data in our study helped us recognize the role of people in the location of documentation and in determining whether or not maintainers could access that documentation. The stories that our participants shared with us gave us a number of examples of how this role plays out, including:

- individuals being able to provide previously unlocatable documentation because of particular personal circumstances (e.g., being part of the development of the system under maintenance),
- individuals or groups exercising control over documentation by erecting intentional barriers to it,

- the commitment of organizational resources to preserving a valuable written source of information by supporting the individual who serves as the gatekeeper to that document.

5. Discussion

In the previous section, we demonstrated the storytelling method we used in this study with several examples of how this approach enabled new discoveries. Instead of asking questions that allow the respondent to generalize on their experience, we asked them to recount specific stories that illustrated the experiences that we were trying to capture. The resulting data were different from traditional interview data in several ways. Most importantly, they contained a considerable amount of contextual information that was included as part of each story. Respondents were encouraged to provide such context because we asked specifically for stories, not for generalized answers. The contextual information that was shared was chosen by the interviewee as the context that was necessary to ensure that the interviewer understood the story.

5.1. Specific strengths of the storytelling approach

This context allowed us to do several things. It enabled us to make connections between different stories and recognize when they were related. For example, we were able to recognize participant P09 as a resource valued by several interviewees. It also allowed us to reason about the conditions under which certain conclusions were valid. For example, we concluded that non-maintainers can be providers of crucial documentation, particularly when they are familiar with the maintenance problem being addressed. The added context also helped us to fully understand the import of a story. For example, we understood better the importance of the developers' notes when we understood that most other documentation was missing in that case.

Another way that our story-based data were different from traditional interview data was the level of detail. Asking respondents to recount whole stories allowed them to include detail that would have had to be glossed over if they had to generalize over several instances.

The added insights we gained are not only valuable in educating researchers about the complexities of documentation in software maintenance, but also have implications for practice. An example is the story about the semi-retired developer who continued to be involved with the maintenance team as a consultant. This serves as an illustration of how maintenance organizations can benefit from such arrangements and the ways in which they can be cost-effective. The implication is that the time saved by utilizing this experienced maintainer's list of contacts and captured organizational memory was worth the cost of the ongoing consulting relationship. Maintenance organizations should be encouraged to investigate the viability of such arrangements. It could be hypothesized, further, that such concrete example stories would be more compelling and convincing to a practitioner audience than generalized descriptions aggregated from less detailed data.

Another implication of the study is that the barriers erected by those who control needed documentation often result in significant delays and added effort. For example, the documentation that was available only for a price from the satellite manufacturer resulted in the maintainer spending large amounts of time combing through unfamiliar pseudocode for a small but crucial detail. Another example is the barrier erected by the maintenance group who made their documentation largely unavailable to the new group taking over the maintenance task. This would have resulted in huge delays for the new group, had they not come upon some developer's personal notes coincidentally. It is thus worthwhile, for the purposes of maintenance planning, to identify such barriers ahead of time and attempt to eliminate them, or at least plan for them. An overarching conclusion from this set of findings is that there is a need in industry for more organized ways of making documentation available to maintainers in a painless and flexible way.

5.2. Notable limitations of the storytelling approach

We have described the benefits of the war stories approach in this example study. However, it is useful to examine the costs as well. While we expected that the storytelling sessions with our participants would last longer than with a traditional interview, that did not appear to be so, although we cannot be certain without some cases for comparison. Few of our storytelling sessions lasted more than an hour, and many were shorter. Certainly some participants were more talkative than others, but there was no evidence that they were more talkative than they would have been in response to more traditional semi-structured interview questions. Data analysis, as described above, was similar to, and as time-consuming as, with any field study. However, as the reader can see from Section 4 and its subsections, a study based on war stories does appear to take up more publication real estate in its write-up, especially given that only a tiny subset of the findings were presented here. Another cost of this type of study is in additional transcription labor. All of the interviews were transcribed verbatim from audio recordings, which is very time-consuming. While it is often the case in more traditional interview-based studies that detailed, paraphrasing field notes are sufficient to capture the data of interest, a storytelling approach really requires verbatim transcripts in order to record the very contextual details that make this approach worthwhile.

5.3. Appropriateness of the storytelling approach

Recall that all good study design involves making informed tradeoffs among various techniques to find the best fit to one's research questions and environment. Storytelling is certainly not an appropriate approach for all

types of studies. Studies with a tight focus that is not related to processes or activities unfolding over time would not benefit from storytelling because the factors of interest would not likely to be related in the details of a story. For example, a study meant to identify traits of software modules that would lead to quality problems during maintenance might not be a good candidate for storytelling. Characteristics of software are not often the subjects of stories.

It is also not appropriate for all participants. We found a fairly wide variation in the abilities of our participants to formulate their thoughts in the form of stories. While many of them appeared to be born storytellers (one even sat in a rocking chair while expounding), others seemed to need more of an externally imposed structure to scaffold their narrative. Our interview guide (see Appendix A) included, after the storytelling prompts, several more traditional interview questions. Looking at the proportion of each interview transcript devoted to the storytelling part of the interview, as compared to that devoted to the more traditional questions, we see a wide variation that mirrors the varying comfort levels with storytelling that we observed among the participants.

6. Summary and implications for further research

We have collected a set of "war stories" about the many ways that maintainers gain the information they need to perform maintenance tasks. These included specially crafted arrangements with retired developers, for example, or coincidental discovery of crucial documentation. We are confident that there are myriad similar "war stories" in other organizations that reveal similar arrangements, both successful and not.

The lessons learned from these efforts are valuable in and of themselves for the discipline. In addition, they will serve as valuable guidance in resolving some of the issues in successfully building an experience repository. First, they will help identify the routine development work artifacts that would be the most promising candidates for inclusion in the repository. Second, they will reveal the selection criteria used by maintainers in choosing amongst a host of meaningful artifacts. Third, the stories will also shed light on appropriate contextualization strategies for the artifacts.

We have also described a qualitative method, the collection of war stories, which is under-utilized in empirical software engineering research. In three illustrative examples we unpacked the insights gleaned from war stories about specific maintenance events. We demonstrated the value of the additional contextual detail by contrasting our findings with the likely results of a more common semi-structured interview approach. This is just one additional technique and data type to add to the qualitative methodological toolkit employed in empirical software research. We can envision it being applied to a number of topics, including the implementation of software metrics programs, the use of agile methods, and software process improvement, among others. This research community should continue to explore and evaluate the utility of field research methods used in the social sciences. At the least this will provide balance to current quantitative modeling approaches; at the most it will generate new theory to explain complex behaviors and provide insight into the many challenging problems in this domain.

Acknowledgements

The authors owe a debt of gratitude to Sumita Das, who performed some of the data collection and analysis tasks for this study. We would also like to thank Raimund Feldman, A. Gunes Koru, Medha Umarji, and David Gurzick, as well the anonymous ISTJ reviewers for their editorial insights. We acknowledge the Department of Information Systems at UMBC for their generous seed grant for this project. Lastly, we deeply appreciate the openness, interest, and exceptional storytelling of our software maintainers.

Appendix A. Interview guide

Interviewee: Interviewer:
Date/time: Recorder:

Objective. Gather "stories" about how information gathering was or was not successful in the context of maintenance (which includes bug fixing, enhancement, and help desk support).

Define "maintenance". Depending on the actual position of the interviewee, replace the verb "maintain" below with something more specific (e.g., enhancing, fixing).

Define"document". Explain that we are going to be asking them about "documents" and our definition, in this context, of "document" is very broad, including anything written down to document a software system, including source code, communications, reports, notes, as well as formal documentation.

Warm-up. Ask about their background and experience, current position, recent maintenance projects, and the like.

Storytelling prompts

(1) Could you tell me about a time when you had trouble finding information you needed to perform a particular maintenance task?
(2) Could you tell me a time when your project was saved by a document you found?
(3) Could you tell me about a time when you found exactly the information you were looking for in a place you did not expect? Where and why did not you expect that?
(4) Has there ever been a time when you had plenty of documentation available, but none of it was useful? Why was none of it useful?

Reaction questions

(1) What documentation do you usually have available when maintaining software?
(2) What documentation would you like to have available when maintaining software?
(3) Which of these documents do you find the most useful? Why?
(4) Which of these documents do you find the least useful? Why?
(5) What makes a document useful?
(6) What documents require the least effort to use? Why?
(7) What documents do you use the most frequently? Why?
(8) What do you look for in documents?
(9) Nobody likes to write documentation, and it's expensive to do. Which types of documentation do you think justify this effort? Which do not? Why?

References

[1] V.R. Basili, M. Lindvall, I. Rus, C. Seaman, B. Boehm, Lessons-learned repository for COTS-based software development, Software Technology Newsletter 5 (3) (2002) 4–7.
[2] J. Carver, The impact of background and experience on software inspections, Empirical Software Engineering: An International Journal 9 (3) (2003) 259–262.
[3] D.M. Fetterman, Ethnography, second ed., Sage Publications, Thousand Oaks, CA, 1998.
[4] T. Lethbridge, J. Singer, A. Forward, How software engineers use documentation: the state of the practice, IEEE Software 20 (6) (2003) 35–39.
[5] W.G. Lutters, Storytelling in Collaborative Work: The Challenge of Preserving Sensitive Interactions, ACM Press, SIG-GROUP Bulletin, 23:2, 2003, 22–23.
[6] W.G. Lutters, M.S. Ackerman, Achieving safety: A field study of boundary objects in aircraft technical support, in: Proceedings of the ACM Conference on Computer-Supported Cooperative Work, ACM Press, CHI Letters, 4:3, 2002, 266–275.
[7] G. McCracken, The Long Interview, Sage Publications, Newbury Park, CA, 1988.
[8] D.W. McDonald, M.S. Ackerman, Just talk to me: A field study of expertise location, in: Proceedings of the ACM Conference on Computer-Supported Cooperative Work, 1998, pp. 315–324.
[9] M.B. Miles, A.M. Huberman, Qualitative Data Analysis: An Expanded Sourcebook, second ed., Sage Publications, Newbury Park, CA, 1994.
[10] J.E. Orr, Talking About Machines: An Ethnography of a Modern Job, Cornell University Press, Ithaca, NY, 1996.
[11] M.Q. Patton, Qualitative Research and Evaluation Methods, third ed., Sage Publications, Thousand Oaks, CA, 2002.
[12] A. Rainer, M.J. Shepperd, Re-planning for a successful project schedule, in: Proceedings of the International Metrics Symposium, IEEE Press, 1999.
[13] O.M. Rodríguez, A.I. Martínez, A. Vizcaíno, J. Favela, M. Piattini, Identifying knowledge management needs in software maintenance groups: A qualitative approach, in: Fifth Mexican International Conference in Computer Science (ENC'04), 2004, pp. 72–79.
[14] C.B. Seaman, Qualitative methods in empirical studies of software engineering, IEEE Transactions on Software Engineering 25 (4) (1999) 557–572.
[15] C.B. Seaman, The information gathering strategies of software maintainers, in: Proceedings of the International Conference on Software Maintenance, IEEE Press, 2002.
[16] H. Sharp, H. Robinson, An ethnographic study of XP practice, Empirical Software Engineering 9 (2004) 353–375.
[17] J. Singer, Work practices of software maintenance engineers, in: Proceedings of the International Conference on Software Maintenance, IEEE Press, 1998, pp. 139–145.
[18] A.L. Strauss, J. Corbin, Basics of Qualitative Research: Grounded Theory Procedures and Techniques, second ed., Sage Publications, Newbury Park, CA, 1998.
[19] J. VanMaanen, Tales of the Field: On Writing Ethnography, The University of Chicago Press, Chicago, 1988.