Modern Software Product Support Processes and the Usage of Multimedia Formats

Parmit K. Chilana[1,2] **, Tovi Grossman**[1]**, George Fitzmaurice**[1]

[1]Autodesk Research
210 King St. E
Toronto, ON Canada M5A 1J7
{firstname.lastname}@autodesk.com

[2]The Information School | DUB Group
University of Washington
Seattle, WA 98195 USA
pchilana@uw.edu

ABSTRACT

Despite being an important channel for end-user assistance, few studies have directly investigated the interactions that occur in modern-day practice of software product support. We present results from a multi-dimensional analysis of product support activities at a leading design software company. We carried out a quantitative analysis of existing support requests, a survey with product support specialists, and follow-up interviews to understand the current practices in product support. In particular, we investigated the utility of different multimedia formats that modern web-based support systems enable. Our results showed that despite the value that these formats bring to support tasks, support specialists still face bottlenecks in remotely resolving software problems. We conclude by highlighting several opportunities in HCI for improving diagnosis and resolution of software issues over the web.

Author Keywords

Product support, support tools, issue diagnosis.

ACM Classification Keywords

H5.3. Information interfaces and presentation (e.g., HCI): Group and Organization Interfaces - Web-based interaction.

General Terms

Human Factors.

INTRODUCTION

Computer support or product support channels are inundated with support requests from end-users every day and cost commercial software companies millions of dollars [1]. Recent surveys show that calls for support, in fact, have been increasing every year[1]. There are currently over half a million support specialists in the United States alone, with a projected 14% increase in employment by 2018[2].

In its earlier days, software support was typically offered over the phone, but the trend in support today is to use the web [20]. Users and support specialists can now leverage new channels for communicating and troubleshooting, such as using screen sharing tools or attaching images and videos to requests and responses.

What is surprising is that even though support specialists are at the frontlines of interacting with software users, and are an important resource for providing end-user assistance, few HCI studies have directly investigated support workflows and the challenges that support specialists face in resolving user issues remotely. Furthermore, although media attachments in modern support systems have the potential to facilitate a type of conversational grounding [10] not possible in phone-based support, little is known about how these formats fit into support workflows and what utility they offer in helping specialists resolve issues.

In this paper, we address these gaps in existing research by investigating the product support practices at ABC, Inc, a large, globally distributed software company that serves more than 80 products to over 10 million customers worldwide, and has nearly 200 *product support specialists*. We used a mixed-method approach for our study: an analysis of ABC's internal archive of support requests, a company-wide survey of product support specialists, and follow-up one-on-one interviews with support specialists.

The main contribution of this paper is in illustrating how modern web-based support is practiced and the relevance of multimedia formats in resolving support issues. In addition, we identify four main bottlenecks that product support specialists face despite the potential availability of information through visual channels. Based on our findings, we discuss several opportunities for better facilitating the exchange of support-related information over the web. Our analysis will be useful for developers seeking to improve the design of support tools, support personnel and managers interested in learning from support practices in the field, and HCI researchers studying user-reported software issues.

RELATED WORK

Although technical help-giving [6, 27] and machine troubleshooting strategies [16, 21, 30] have been studied for

[1] http://www.supportindustry.com/2009supportmetrics.html

[2] http://www.bls.gov/oco/ocos268.htm

CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.
Copyright 2011 ACM 978-1-4503-0267-8/11/05....$10.00.

decades, few studies have investigated software support practices, particularly from the perspective of product support specialists. Here, we highlight works that have studied aspects relevant to software support and remote troubleshooting in a variety of domains.

Studies of Software Support

Existing studies of product support have broadly focused on organizational, process, and management problems in the provision of support [15, 20], but not the actual interactions that occur in the diagnosis and resolution of support issues. Other studies of software support activities have focused on organizational knowledge and management of issue repositories [7, 8, 12] with the goal of improving support productivity. In contrast, we focus on diagnosis techniques and tools used by support specialists to better understand and improve the resolution of user-reported issues.

Studies of web-based support have emerged recently, but largely within the context peer-to-peer help in open source software (OSS) communities where users can directly communicate with other users and developers about support issues [17, 24] and bugs [2, 3]. Commercial software support is largely understudied, with the exception of quantitative analyses of application-specific support request logs [14]. Our study contributes to this body by providing a multi-dimensional perspective on commercial web-based one-on-one software support practices.

Studies of Remote Troubleshooting

The challenges of remote troubleshooting have been discussed in various domains. For example, Poole et al. [22] describe the challenges of diagnosing remote home network set up problems by analyzing phone calls that customers make to support. Their findings illustrate that telephone-based support requests prevent the customer and support specialist from having a shared understanding of a problem. In response to this limitation, Fussell et al. [10] argue that shared visual space is essential for collaborative repair because it facilitates *conversational grounding*. Some works have already started to explore novel technologies that could facilitate this type of grounding for remote repair tasks [23], but not in the software domain. Other works propose multimedia tools for software learning and explaining complex steps [13, 26], but do not investigate how such tools can help resolve support issues. In our analysis, we pay particular attention to the forms of media utilized within web-based support and to what extent they facilitate conversational grounding for support specialists.

In summary, studies related to software support and troubleshooting do exist, but none have directly explored the different dimensions of modern web-based product support workflows and relevance of media formats. Our work complements existing works by providing an HCI perspective on how web-based support issues are understood, diagnosed, and resolved by support specialists.

RESEARCH SITE AND METHOD

We now describe our research site and the strategies used to collect data for understanding product support activities.

About ABC

ABC, Inc. is a leading software company specializing in 2D and 3D design, engineering and entertainment software for the manufacturing, building and construction, and media and entertainment domains. There are over 80 different ABC products currently in the market, serving over 10 million end-users in 185 countries.

Method Overview

Since ABC offers a range of products and product support activities at ABC are spread across different channels, we used a mixed-method approach for collecting data. We began with a pilot investigation where we conducted unstructured interviews with two product support specialists. Our goal in these initial interviews was to get an overview of the product support process, who the relevant staff members were and where they were located, and learn about the tools used by customers and the support specialists. To learn more about internal support-specific tools, we also interviewed one of the lead developers of ABC's internal customer support database. Through this process, we obtained access to an archival database containing the complete set of customer requests from the last 8 years.

Following this pilot investigation, we designed 3 strategies to systematically gain an in-depth understanding about the product support process: (1) an analysis of existing support requests, (2) a company-wide survey distributed to 200 product support specialists, and (3) follow-up interviews with a subset of product support specialists.

Quantitative Analysis of Archived Support Requests

The internal customer support database contained over 4 million records from various customer-related service activities, including technical support requests. We obtained access to the internal archival SQL server database, and wrote our own C# programs to run queries to explore these requests in more detail.

Through our analysis we found that there were 1, 665, 970 service requests in the database submitted between January 1, 2002 – May 15, 2010. Out of these, 381,060 were business requests, while 1,284,910 were product-related technical support requests. We first randomly read 20 technical support reports to gain an understanding of the report structure and report-specific activities. Next, to investigate some trends that we observed in this sample, we carried out quantitative analyses over the last one year of data which contained around 75,000 support requests.

Company-Wide Survey

The second method that we used was a company-wide survey of product support specialists. Our goal was to better understand the demographics of product support specialists and the prevalence of some trends that arose in our initial analysis of support requests. The questionnaire consisted of 30 questions that elicited a mix of multiple-choice and short open-ended responses. We focused our questions on the demographics of support specialists and their use of tools in communicating with customers and diagnosing software issues. Finally, we asked respondents if they wanted to opt-in for a follow-up interview. The survey took about 10-15 minutes to complete.

We distributed the survey online using the company's intranet and by advertising on internal mailing lists. All responses were collected anonymously. We received 72 responses from a pool of 198 product support specialists at ABC, giving us a response rate of 36.4%.

Semi-Structured Interviews

Lastly, to confirm trends in our quantitative findings and to learn more about experiences of support specialists, we conducted 16 semi-structured interviews. These interviews were carried out with product support specialists working across different ABC sites. Each interview lasted approximately 45 minutes. We carried out 11 of the interviews in person, 4 over the phone, and 1 using the company's internal telepresence conferencing facility. Our 16 interviewees included 7 product support specialists, 3 senior product support specialists, 5 product support team leads, and 1 product support technical lead. They had different specializations across 10 of the major ABC products and 2 of the interviewees provided support for all products. The support-related experience of our interviewees ranged between 2 to 15 years, with an average of 5.9 years.

In the first part of the interviews, we focused on confirming some of the trends that emerged in our analysis of support requests and the company-wide survey. Next, we used the critical incident technique [9] to probe into scenarios that were particularly challenging in providing support. Since we conducted the majority of interviewees within the product support's specialist work environment and they had access to the repository of support requests, we found that it noticeably facilitated recall among our interviewees.

We audiotaped and transcribed all of the interviews. All transcripts were organized, coded, and analyzed using the *NVivo* data analysis software. In the first pass, we coded for data related to multimedia formats mentioned by our interviewees. In the next pass, we examined critical incidents described by interviewees using an inductive analysis approach [25]. This approach of was useful in exploring different facets of the incidents described by support specialists and in identifying recurring themes.

Presentation of Results

Since our three data collection strategies produced a large amount of quantitative and qualitative data, we have combined our presentation of the results in terms of the major themes that emerged. We begin in the next section with an overview of product support at ABC.

OVERVIEW OF PRODUCT SUPPORT

We first provide a global overview of the process of product support provision and demographics about product support specialists.

Reporting and Tracking Tools

All support requests were logged in a web-based tracking system (even those reported via the phone initially). The web-based system allowed customers to update information and attach files during the course of the request. Customers could also track the progress of all their requests through this system. The required fields when submitting a request were the product name, operating system (OS), a one-sentence summary, and the full problem description.

Product support specialists accessed support requests via another web-based system where all the history and attachments related to individual requests were stored. The specialists could send information to the customers, add updates, and run queries through this interface as well.

Finally, every report submitted since 2002 was archived in a consolidated internal database. This database could be use by support specialists to search for existing solutions or to find similar issues. Developers and QA specialists also archived change requests or bugs in this database.

Reporting Process

When a customer submitted a request, it was handled by *tier 1 specialists* who had basic knowledge of ABC products, but were not specialized in any specific support area (Figure 1). They could search for known issues in the consolidated database and provide help, or they could escalate the issue to a product support specialist. From that point forward, the product support specialist and the customer communicated with each other directly. The support specialists who we studied were at the second tier and had extensive system-level and application-level expertise. When necessary, the support specialists would collaborate with other team members, developers, or QA specialists to diagnose and resolve complex issues.

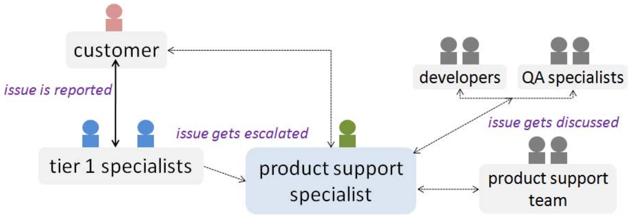

Figure 1: The issue reporting and handling process.

Demographics of Product Support Specialists

There are around 200 product support specialists at ABC who specialize in different products and areas of support. Among our 72 survey respondents, 44 had the title product support specialist, 10 were senior product support specialists, 13 were product support team leads, 2 were product support directors, and 4 had other titles such as customer support engineer. The median experience of the respondents in the field of product support was 8.8 years. The majority of our respondents worked on specific ABC product families, and only 11% across all products, mostly providing support for installation issues. In terms of training, 41% of respondents had Bachelor's degrees, 36% had Diplomas or Associates Degrees, 12% had Masters degrees, and 8% had Certificates in a range of fields such as computer science, civil engineering, animation and design, among others. Our respondents estimated that they addressed between 2 and 30 support requests every day, with an average of 5 each.

ANALYSIS OF SUPPORT REQUESTS

We carried out various analyses to understand the structure and contents of support requests and computed statistics about activities surrounding these requests.

Overview of Support Requests

We initially looked at a random sample of 20 existing support requests to see the structure of the information contained in these requests. The main information contained in these records of support requests included:

- the dates when the request was created and modified
- the request status (i.e., whether it was closed or open)
- the issue's summary and description
- the product and feature affected by the issue
- the issue's internal classification

Based on our informal screening of these support requests, we learned that some reports lasted a few hours whereas some lasted a few months. There was also variability in terms of the topics of the reports and the number of iterations in the conversations between the customers and support specialists. Some requests were about system configuration issues, such as installation. Others were related to application feature issues. Support specialists were frequently asking for more information from customers, and there were numerous files formats being sent back and forth.

To investigate these support request trends more systematically, we carried out a quantitative analysis over the last one year of support requests that had been closed.

Total number of support requests	74,837
Life-time of support requests (until marked as closed)	2.9 days (median)
Number of activities in support requests (messages and attachments exchanged)	6 (median)

Table 1: Summary of one year of support requests.

Trends in Support Requests

The results of our quantitative analysis are summarized in Table 1. We found that the median life-time of support requests, calculated as the time between when they were submitted to the time they were officially closed was about 2.9 days. The median number of activities (messages and/or attachment submissions) per request was 6, while the average was 12, so there appeared to be a high amount of activity in the support requests within their short life-times.

The distribution of the internal classification of support request areas is illustrated in Table 2. It is interesting to note the high proportion of *how-to's* in support requests, suggesting that customers consult support not only when something goes wrong, but also when they want to figure out how to accomplish a particular task. Some of our interviewees who worked on feature issues indicated that the number of how-to's was as high as 75%.

Area	*% of Support Requests*
How-to	26%
Incorrect Result	26%
Error message	15%
Crash/Hang	9%
Configuration	7%
Limitation/Wish	6%
Performance	3%
Documentation	3%

Table 2: Distribution of top 8 support areas.

Since the web was used as the main platform for support at ABC, there was opportunity for exchanging information in formats not possible through phone-based support. Next, we look at the file formats exchanged between customers and product support specialists in more detail.

FILE FORMATS IN SUPPORT REQUESTS

In our one year sample of requests, we found that 23,514 requests (31.4%) contained attachments, but only 12370 requests (16.5%) came in with an attachment at the time of submission. We further investigated these attachment types and found that they included a range of file formats, such as images (jpg, png), source files (ABC-specific drawing files), archive files (zip, rar), documents (doc, docx, pdf), text files, log files, and videos (avi, swf). Figure 2 shows the combined distribution of attachment types with initial requests.

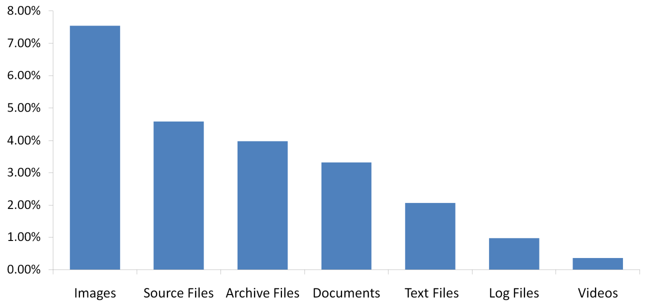

Figure 2: Percentage of support requests that initially came in with the various attachment types, in the one-year sample data.

To better understand the actual use and utility of these different multimedia formats, we asked format-specific questions in our survey and had follow-up questions in our interviews. We considered most of the formats that were

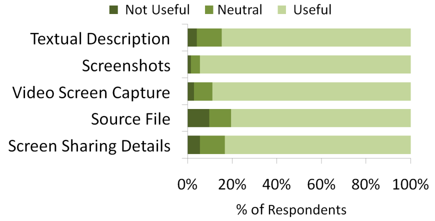

Figure 3 (a): Usefulness of formats for understanding issues from the perspective of support specialists. In general, any additional information was considered useful.

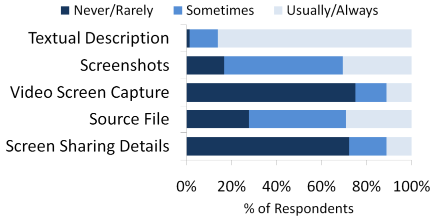

Figure 3 (b): The likelihood of customers submitting additional information with their initial requests from the perspective of support specialists.

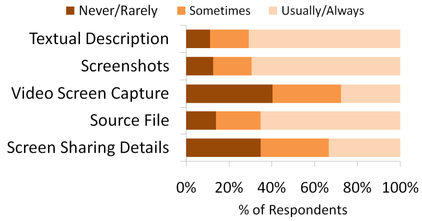

Figure 3 (c): The likelihood of support specialists requesting information in different formats from customers after the initial requests were submitted.

found in on our analysis: textual descriptions, screenshots, source files, and video screen captures. We also considered screen sharing which we noticed was also being used in our random sampling of requests.

We asked respondents to rank: the usefulness of formats for seeing information related to a customer's support request (Figure 3a); how likely it was for customers to include information in these different formats with their initial support request (Figure 3b); and, how likely it was for them to request information in these 5 different formats after seeing a customer's initial request (Figure 3c).

The data illustrated in Figure 3 points to a perplexing discrepancy in terms of the formats that the product support specialists said were useful for diagnosis, versus what they reported customers included in their request, and what support specialists *actually* asked customers to submit. For example, consider that 89% of respondents said that video screen captures were useful or very useful, but only 27% of specialists were likely to ask customers to submit videos. To understand this discrepancy and why product support specialists made such choices, we asked our interviewees to comment on the merits and challenges of using the different formats.

Screenshots

Our interviewees explained that screenshots were the most useful format usually because screenshots provided a more accurate visual sense of what was actually happening on the customer's screen, compared to the customer's paraphrased description. One interviewee commented:

We ask for screenshots for error messages. Our error messages are cryptic and numerical. If the error has something like 000bc and you type in 000dc, that's a significant difference for us. Instead of the additional havoc we have to deal with, we just ask users to submit us screenshots of the exact error message. (P16)

The interviewees generally agreed that often asking for a screenshot of the problem was easier since most OS today offer a built-in "print screen" function. Still, one surprising finding was that four of our interviewees (25%) described instances where customers did not understand the concept of a screenshot. One interviewee remarked:

We had a customer [who], when we asked for a screenshot, got out the company's digital camera and took a picture of the screen and then emailed that picture to us. (P04)

In addition, interviewees pointed out that although screenshots were a lot more useful than text alone, the screenshots alone did not always provide enough information for troubleshooting complex feature issues:

[The] problem with screen capture images is…[they don't] tell us the workflow...unless they've edited the image or drawn arrows or clouded something, often times, they're not even pointing at the issue itself. I may have a whole screen image of a bunch of stuff, but it may not be obvious where the issue is. (P12)

Video Screen Captures

Video screen captures were considered useful for understanding complex or multiple steps needed to reproduce a problem or for explaining things to customers when screenshots were not sufficient. For example, one of our interviewees explained

Videos are useful in specific circumstance. If you can't reproduce the problem or get the customer to describe their problem, ask for a video or get them to demonstrate it live in WebEx. Those circumstances are rare - more specific to workflow issues. (P01)

Video captures were also useful for capturing missing steps during the diagnosis process of a complex issue:

I told him [the customer]: I did this, here are 2 videos, this is the result I get, do you get the same thing? We ended up with 7 videos for each step, trying to exactly isolate where the problem was. He sent one video back just to make one thing clear. (P10)

One specific question we had was why despite being so useful were videos not requested from customers as much as screenshots. Interviewees explained that the main reason was they did not want their customers to struggle with installing 3rd party software for creating videos:

According to our interviewees, another issue with videos was that they consumed more time compared to screenshots:

It's also easier for customers to provide screenshots than videos. Videos are more complex. Not all customers have time. I've asked customers for videos before but they just tell us they're too busy to do it. Then what are we going to do. (P01)

Source files

Our interviewees explained that most of the tasks that users accomplished with ABC software products involved some form of drawings consisting of intricate processes. Thus, when customers were trying to explain what they did or what is wrong, the product support specialists benefited from seeing the actual file in question, so they could have the ability to reproduce the issue on their end:

We do end up asking for drawing files, so we can try to reproduce them on our system and test to see if the issue is with their[the csotmer's] network and to try to recreate the error with their system. (P01)

Another benefit of source files was that the specialists could see what the cause of a problem was, for example, for an error or crash situation:

At least 50% of the time I need the file or script. We get files because customers says, 'I want to do this' or 'I'm rendering and it's crashing'. At this point, a video won't be helpful because yes it'll show me the problem, but won't help me diagnose it. Then I need the file so I can reproduce the problem and break down the steps to reproduce and [see] the related steps. (P03)

One issue with sharing files was that they were often very large in size due to the complex drawings that they contained. Special FTP sites had to be set up in many instances for customers to upload large files, which users often had difficulty accessing. In addition, some interviewees felt that files could introduce information overload. Complex files contained "too much" information and the customers mostly failed in pinpointing the exact location of the problem:

I had a guy tell me the other day he couldn't select a line. The drawing he sent me had over 2000 objects in it, so I'm like ok that's great, but which line are you having a problem with? (P04)

Screen sharing

Our interviewees had mixed feelings about screen sharing: some said they live by screen sharing, while others said that they would only resort to screen sharing when other possibilities had been exhausted.

The obvious advantage of sharing screens was that it maximized the potential for "shared understanding" [10] for the situation at hand. Product support specialists who worked on system configuration issues particularly found this shared understanding to be useful because of the number of system variables involved. But, since modern screen sharing software allowed support specialists to even take control of the customer's systems, other benefits could also be gleaned as described in this account:

Without the [screen sharing], it would be a nightmare. Our stuff is pretty complicated and even some of our advanced users just don't get it sometimes. We depend on [operating system] components for setting up servers and some people just want us to do it for liability issues if they are not confident, they think it will get messed up. So with [screen sharing] we can share files, we can connect desktops, it's like live meetings, remotely access the system. (P02)

Other interviewees described the downsides to screen sharing that they had experienced with customers. Some interviewees felt that the screen sharing sessions ended up taking too much of their time in cases where they needed to investigate the underlying issue further. For example, one interviewee pointed out screen sharing put him "on the spot" when he was dealing with complex features:

So, when I get an email case, I can sit down with it, try couple of things. It may take me an hour. But, if I connect right away, I just wasted my hour and the customer's time. I didn't make myself look good because I appear like I'm failing in trying to diagnose. Whereas if I get their file, I can look at it and tell them this is what your problem was and this is how you fix it after an hour. (P11)

In summary, we found that the use of multimedia formats did help support specialists in establishing common ground [5] with customers, but customers were not always aware of *how* to make use of these formats and did not always know *which* format would best describe a problem.

BOTTLENECKS IN ISSUE RESOLUTION

Despite the use of multimedia attachments, our findings shed light on a number of challenges that product support specialists faced in resolving support issues. These challenges were synthesized from the *critical incidents* that we elicited in our interviews. In this section, we discuss the four major themes that appeared within these critical incidents and group them as *bottlenecks* in issue resolution.

1. Unclear problem descriptions and steps to reproduce

One bottleneck that came up repeatedly in our interviews was that customers usually provided descriptions that were either too vague, brief, or general and not specific enough to the context of the issue. Furthermore, the descriptions often lacked clear steps to reproduce.

For example, one issue was the mismatch between the actual problem and what the customer *believed* was the problem in the initial description:

[The customer] said I'm running this script but my filter is disappearing. [He asked] Is there something wrong with my scripting command? So he wrote it up and sends me the script which is 100 lines long. He said this should describe my problem. I spent a lot of days figuring out what he had done, I spoke to QA and we looked at it together. We weren't sure if the script that he had provided was a clear example of what he was describing...So, I got him on the phone and talked to him and tried to figure it out. And that's when he goes to say, oh actually that was just an example of 'this', but I really want to do 'that'. The 'that' was a lot more complex. (P03)

As discussed in the previous section on formats, even when customers did try to be helpful and sent in additional information such as screenshots or source files, this information was not always helpful in establishing an understanding of the problem if it lacked an accompanying explanation of the relevant context.

I had one case this morning, customer basically had a view where they basically have this kind of nice round hand rail in their program and when they go to put in another view of it, it just completely vanishes for no good reason. So, I asked him for his file. Initially, he just sent me a screenshot and we don't always get all the information we need from that. The problem is that it doesn't show me his problem. It could be happening because of many things, like he could've selected the wrong view. (P13)

This type of problem was also evident in our survey, which showed that 77% of the support specialists would find it somewhat useful or extremely useful to see additional tags or annotations on images or videos sent by customers to better understand the context of the issue.

The support specialists also felt that customers did not see the value of reproducibility information in the descriptions they provided. For the types of complex software at ABC, often leaving out even a little step had a great impact on the diagnosis and the resolution of the issue. One product support team lead explained a case that his team had worked on extended for a month just because the customer kept missing one step in the reproducibility information:

There's one that's been going on for a while that I wasn't able to reproduce ...the case has been open for a month..first it was in tier 1 hands, and it finally got escalated to me because they couldn't reproduce it. I tried to find out if it was reported before. I asked them to uninstall the hotfix and they replied that as soon as they put the hotfix on, nothing worked. I did that too and couldn't reproduce it. So I asked them [customer] for the exact file that they had used..I verified it that I had the same dll file. When I tested it again in a new environment with this file and followed their steps, I was able to reproduce it. I may have missed out a step earlier because yesterday morning they [customers] clarified the steps again, but it took a while to get us there. (P09)

One factor influencing unclear descriptions and incomplete reproducibility steps could be that the current interface which the customers use to submit requests does not explicitly ask them to provide steps to reproduce. But, as other studies have shown, even in interfaces where the requirement to submit the steps to reproduce information is explicitly stated (i.e., in OSS communities [2, 24]), reporters rarely follow these instructions. We discuss possible ways of mitigating this problem in our discussion section.

2. Variability in system environments

Another bottleneck that affected the resolution of support issues was variability within customers' system environments. Customers were reported to be using a wide range of OS, software, and hardware configurations, with any combination of these potentially being at the root cause of an issue. Thus, even in the rare case when a customer succeeded in providing a clear description of the problem and the steps to reproduce, sometimes support specialists were stalled in diagnosing the customer's issue because they could not replicate the underlying system environment.

For example, one interviewee pointed out that it could be something as simple as an automated OS update that could cause incompatibly with an application's features:

There's another one [case] that I just closed today. This one had a ton of iterations. It's been running for over 3 months. It's not even solved, but the customer's last response was just close it and I'll look at it in the fall. He sent me his files and they worked fine for me. We can see the files ok, using the same OS, and same version of the application. I sent him screenshots, videos, and screenshare, done it all…there are things like an [OS update] that break things. There are 2 different versions of [XY software] and there's nothing in common. Same software. But, routinely just an [OS update] may break it. But you never really know what did it. (P14)

One product support specialist who worked on an application that was dependent on the system environment described an extreme case in diagnosing an error that the customer was experiencing. He had to physically obtain the customer's system in order to do further troubleshooting:

…a lot of times the error comes from customization. Our program allows them [customers] to make things really customized so we end up getting their whole system, restore it, and then try to reproduce it. (P02)

A related problem identified by support specialists was that customers lacked knowledge about their system environment, concurring with other findings [14]. For example, customers often did not know the version of the software they were using, details about their OS, graphics card, memory, driver information, among other system configuration settings. Furthermore, there were also application-specific settings which could cause incompatibilities. In our discussion section we discuss possible opportunities that could assist users in reporting their context at the system and application levels.

3. Privacy or security concerns

Another class of bottlenecks that emerged in the critical incidents described by interviewees had to do with privacy or security concerns in sharing data. Strict lockdowns in the customer's work environment prevented them from providing relevant information needed for diagnosis:

A couple of private firms wouldn't send me the data so the SR [support request] kind of died. What can I do. I've had a few calls, I send them a WebEx link and their firewall is blocking it, so it's kind of frustrating. I can try to supplement it with a video...but we ran into [issues with] that too. A lot of workplaces block Flash, so I send them a link to a swf file (video) and they can't play it. Sometimes customers take that video home with them and then play it from there. (P07)

Some customers required specialized *non-disclosure agreements* (NDA) in order to share files. But, such cases often ended up entangled in legal issues and stalled the diagnosis of the issue:

They [customers] indicated that they were unable to send the model because they needed a non-disclosure agreement (NDA) first. I retrieved an NDA at first but they refused it because they didn't want to sign a two way NDA and said that it had to be a one way [NDA]. So, we had to go through our legal department and they indicated that they didn't do one-way NDA agreements. About a month later, they [legal team] figured out we could do one way. I sent that to the customer—they then sent us their NDA with modifications, then we sent it back. That's basically where it's been. (P05)

Since privacy issues sometimes blocked the progress in a support request completely, product support specialists had come up with workarounds in helping customers self-diagnose their issues:

I've been making suggestions along the way for the customers. I've even pointed out that new release has come out [which could help]. None of them has provided a resolution, so it's a situation *that I haven't heard anything from the legal dept or the customer in terms of where things stand...essentially to isolate a problem in our modeling software, it's really hard, even if there was some way to make the model private...we need to know the parameters, whole geometry, in order to isolate the problem. If the problem is happening in an area considered private or secret, then we can't really deal with that. (P05)*

Some support specialists used the strategy of asking the customer to extract a limited portion of the file that showed their issue or to send in as much as information possible. Other support specialists disagreed because the extent of the privacy or security concerns was so great that an excerpt of a file was not helpful:

Getting the customer to edit out or do something to hide a part just wouldn't help. I mean their issues are either black and white - they can send the data or not. Hiding the name of the building that they are creating doesn't make it ok to send it out. (P04)

Although privacy issues related to files are entangled in larger organizational and social issues, we point out some possibilities in our discussion section about how to enable the exchange of sensitive information more securely.

4. Variability in user workflows

The final bottleneck we discuss, that emerged in our data, was related to support specialists struggling to understand the specialized domain or application-specific workflows being used by customers. For instance, many support specialists felt that the root cause of a customer's issue was sometimes in the workflow of what the customer was trying to accomplish, rather than a defect with the application:

It's half workflow issue, like they [customers] are confused...it's a relatively complex thing. It's working as it should for the most part, but the way they've done it is just more complicated. We have to get them started from scratch and tell them to go through these steps and tell them why it's doing it. (P 10)

Other interviewees said that customers often assumed the support specialists would be familiar with their idiosyncratic workflows:

When I ask a customer to tell me how to reproduce the issue, he'll be like just draw a wall. And they don't say click here, here and there. I had a case where we kept going back and forth and I couldn't reproduce the issue. It turned out that the customer kept leaving out a step. They [customers] are architects and engineers and we're the people writing the steps so there's kind of a disconnect. (P13)

Half of our interviewees pointed out that regardless of their knowledge of an application, sometimes the lack of familiarity with a customer's domain also impeded the diagnosis process:

When [YZ software] was first released, it was supposed to be for palette design. Now, we've seen everything from a prison facility with a fence that went all around it, to a fire-escape for sky scrapers, a floating barge that loaded and unloaded oil rigs in sea...and I'm like you know what...this is a little beyond me. (P15)

This interviewee further commented that he could just walk across the hall and talk to someone in such situations. Other support specialists worked around by calling friends in industry who could sometimes help them understand latest industry-related issues relevant to a support request. Collaboration appeared to play a role in dealing with a lack of domain knowledge, as has been illustrated in other studies [4]. Most support specialists noted that online discussion forums were becoming popular for discussing workflow-related issues and the specialists even encouraged customers to consult these forums:

Largely because there's more than one way of doing a lot of this stuff. The community can share best practices. Because honestly we're not users of the product, so I don't know the best way to design a cul-de-sac, but someone out there probably does. (P11)

Our interviewees felt that they could already see community-based discussions changing the current state and future of one-on-one product support.

DISCUSSION

Our results illustrate that in remote resolution of software issues, common ground [6] between a user and a support specialist can be enhanced by the use of multimedia formats. However, users do not appear to always use appropriate mediums for describing the features of a problem or fail to adequately describe the sequence of steps desired by support specialists. Although these findings of unclear or incomplete user descriptions are consistent with previous studies of troubleshooting [2, 6, 24, 27], our study shows that commercial web-based support practices are plagued with additional concerns, such as privacy control in exchanging sensitive files. Also, with increased push for end-user tailoring and customizability of applications [13, 18], the disadvantage for support specialists is that they have to deal with unique scenarios of feature usage, more variability in steps used to accomplish tasks, and occurrence of more problems due to different combinations of application features tried by users.

We now discuss the implications of our findings for improving the resolution of software issues through web-based support.

Opportunities for Improving Issue Resolution

We consider three levels of opportunities for improving remote issue resolution: (1) the integration of reporting formats and applications; (2) the design of automatic tools for capturing the user's system and application level context; and (3) the integration of community-based support channels.

Integration of reporting formats and applications

Although a vast number of freeware tools online today could be useful for diagnosis purposes (i.e., *Jing* for quick video captures), our results showed that they were rarely used. The support specialists felt that the main problem which hindered users in making use of external multimedia tools was the lack of integration with the application state. Users were either not aware or were hesitant to download freeware that they did not know how to use only for the purpose of creating a support request. Thus, one way of mitigating this concern would be to allow users to generate screenshots or videos within the reporting interface related to an application. Recent tools such as the *Windows Problem Steps Recorder* could be exploited for this purpose, although at the time of this study, we did not find widespread usage of this tool in the support requests submitted to ABC.

In addition, as discussed in our findings, support specialists often found it difficult to make sense of a user's problem from raw multimedia attachments. For complex issues, they needed to see a pointer to the issue that the user was experiencing whether in a screenshot, a video, or a source file. A potential solution is to provide seamless ways for users to add annotations or highlights to images, videos, and documents that are attached to a request.

Automatic capture tools

As illustrated in our findings, when support issues are intricate and involve several variables, users struggle in looking up and providing the necessary system or application-state level information. Our results point to two opportunities for relieving the burden on the end-user to describe these levels of context: (1) automatic capture of information about the user's system environment, and (2) automatic capture of application-level sequence of interactions.

System-level: Automatic capture and analysis of system logs has been discussed in prior work [2, 14] but our findings suggest that even basic system level information about the OS, version numbers, installed packages, and permissions can reduce iterations between end-users and support specialists. Simple ways of capturing the relevant environment information are needed along with facilities for attaching this information automatically at the time a request is created so that the burden on the user decreases and the support specialist does not have to repeat the same request to all users.

Application-level: The other bottleneck that support specialists faced was dealing with descriptions where users struggled in describing their own actions and history of application-level interactions. We believe there are opportunities in this space to explore how document histories can be captured automatically so that when users submit support requests about feature issues, their steps to reproduce get automatically attached. For example, the Chronicle system [11] which captures and visualizes a document's workflow history, could be adapted to be used in the context of issue reporting.

Although efforts in automatic capture of system-level details and application-level details could alleviate some of the burden that users have in describing in their context, there are some inherent challenges in doing automatic capture that are implied in our findings. For example, a major concern that emerged in our findings was preserving the privacy of the user's information. To address this, users could be provided with regulated tools to control what portion of their system logs get included in an automatic capture, or what portion of a user's file gets recorded or submitted. Users could adjust the levels of sensitivity from project-to-project and modify the type and extent of the automatic capture of context. Another potential tension surrounding automatic captures is information overload from the captured data. We believe it is imperative for support specialists to be included in the design of such tools upfront so that they can provide a perspective on support workflows and pinpoint to data that would be most useful for diagnosing different types of issues.

Community-Based Support Channels

Finally, our results suggest that given the variability in user workflows and environments and the high cost of one-on-one support, community-based forums will emerge as the next frontier in modern support. Online discussion boards that facilitate the exchange of troubleshooting experiences have existed within OSS communities for over a decade [17, 24], but their uptake in commercial contexts is also on the rise [19, 28]. Such online user communities could be particularly useful for users having domain-specific workflow problems where they could benefit by learning about best practices in the community from other users. The opportunities that we have discussed in the previous sections, such as integration of reporting tools and privacy control in exchanging sensitive information, should be considered for community-based help initiatives as well.

Limitations

Finally, we acknowledge our study has a number of limitations. First, although the support specialists and the support requests that we investigated came from 80 different products at ABC, our results could potentially be biased because of organization or corporate culture. In addition, some of the findings could be a direct result of the specifics of the support tool being used, rather than the overall process itself (although our investigations showed that the particular support management system used at ABC is used by over 6000 companies globally). The theme of workflow variability could be unique to our study, since we were dealing with software for design work, which is known to be creative, with ill-defined problems [29]. There are also limitations associated with the methods that we used. For example, an inherent limitation of semi-structured interviews is that no two interviews end up being the same. Thus, we generalize the findings with some caution. Still, we believe that since we took a multi-dimensional approach in studying this topic and our research site was a large company, it is possible that our results reliably illustrate the different facets of modern product support.

FUTURE WORK AND CONCLUSIONS

Our study results have provided a lens into the work of product support specialists at a large, distributed software company where the web is used as the main platform for interacting with end-users. Our results illustrate the advantages of different multimedia formats that get exchanged in support requests, but also point to larger bottlenecks faced by support specialists in diagnosing and resolving issues remotely. We believe that the opportunities we have discussed for improving issue resolution are ripe for future work in HCI. For example, in our own work, we will be tackling the design of automatic capture tools that take privacy concerns into account.

At the pace of innovation in computing systems, we can expect product support to continue to play a key role in facilitating post-deployment user experiences. Although there may not be a single way to alleviate the bottlenecks in issue resolution that emerged in our study, our work lays the foundation upon which designers of support tools, support professionals, and HCI researchers can begin to address some ways of improving interactions in remote software support.

REFERENCES

1. Association of Support Professionals. Technical Support Cost Ratios, 2000. http://www.asponline.com/tscr.pdf.
2. Bettenburg, N., Just, S., Schröter, A., Weiss, C., Premraj, R., and Zimmermann, T. What makes a good bug report? *Foundations of Software Engineering* (2008), 308-318.
3. Breu, S., Premraj, R., Sillito, J., and Zimmermann, T. Information needs in bug reports: improving cooperation between developers and users. *ACM CSCW* (2010), 301-310.
4. Chilana, P., Wobbrock, J., and Ko, A. Understanding Usability Practices in Complex Domains. *ACM CHI* (2010), 2337-2346.
5. Clark, H.H. and Brennan, S.E. Grounding in communication. *Perspectives on Socially Shared Cognition 13*, (1991), 127–149.
6. Crabtree, A., O'Neill, J., Tolmie, P., Castellani, S., Colombino, T., and Grasso, A. The practical indispensability of articulation work to immediate and remote help-giving. *ACM CSCW* (2006), 219-228.
7. Cunningham, S.J., Knowles, C., and Reeves, N. An ethnographic study of technical support workers: why we didn't build a tech support digital library. *ACM JCDL* (2001), 189-198.
8. Das, A. Knowledge and productivity in technical support work. *Management Science 49*, 4 (2003), 416-431.
9. Flanagan, J. The critical incident technique. *Psychological Bulletin 51*, 4 (1954), 327-358.
10. Fussell, S.R., Kraut, R.E., and Siegel, J. Coordination of communication: effects of shared visual context on collaborative work. *ACM CSCW* (2000), 21-30.
11. Grossman, T, Matejka, J, and Fitzmaurice, G. Chronicle: Capture, Exploration, and Playback of Document Workflow Histories. *ACM UIST* (2010), 143-152.
12. Halverson, C.A., Erickson, T., and Ackerman, M.S. Behind the help desk: evolution of a knowledge management system in a large organization. *ACM CSCW* (2004), 304-313.
13. Harrison, S.M. A comparison of still, animated, or nonillustrated on-line help with written or spoken instructions in a graphical user interface. *ACM CHI*. (1995), 82-89.
14. Jiang, W., Hu, C., Pasupathy, S., Kanevsky, A., Li, Z., and Zhou, Y. Understanding customer problem troubleshooting from storage system logs. *USENIX File and Storage Technologies* (2009), 43-56.
15. Kajko-Mattsson, M. Problems within front-end support. *J Software Maintenance and Evolution: Research and Practice 16*, 4-5 (2004), 309-329.
16. Kim, C., Choe, P., Lehto, M.R., and Allebach, J.P. Effect of providing a web-based collaboration medium for remote customer troubleshooting tasks. *HCI International* (2007), 47-53.
17. Lakhani, K.R. and Von Hippel, E. How open source software works: “free” user-to-user assistance. *Research policy 32*, 6 (2003), 923-943.
18. Mørch, A. Three levels of end-user tailoring: Customization, integration, and extension. *Computers and Design In context*, (1997), 51-76.
19. Nambisan, S. and Baron, R.A. Interactions in virtual customer environments: Implications for product support and customer relationship management. *Journal of Interactive Marketing 21*, 2 (2007), 42-62.
20. Negash, S., Ryan, T., and Igbaria, M. Quality and effectiveness in web-based customer support systems. *Information & Management 40*, 8 (2003), 757-768.
21. Orr, J.E. *Talking about machines: An ethnography of a modern job*. Cornell University Press, 1996.
22. Poole, E.S., Edwards, W.K., and Jarvis, L. The Home Network as a Socio-Technical System: Understanding the Challenges of Remote Home Network Problem Diagnosis. *ACM CSCW 18*, 2-3 (2009), 277-299.
23. Ranjan, A., Birnholtz, J.P., and Balakrishnan, R. Dynamic shared visual spaces: experimenting with automatic camera control in a remote repair task. *ACM CHI* (2007), 1177-1186.
24. Singh, V., Twidale, M.B., and Rathi, D. Open source technical support: A look at peer help-giving. *HICSS* (2006), 118.3.
25. Strauss, A. and Corbin, J. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. Sage Publications Inc, 1998.
26. Sukaviriya, P., Isaacs, E., and Bharat, K. Multimedia help: a prototype and an experiment. *ACM CHI* (1992), 433-434.
27. Twidale, M. and Ruhleder, K. Where am I and who am I?: Issues in collaborative technical help. *ACM CSCW* (2004), 378-387.
28. Wiertz, C. and de Ruyter, K. Beyond the call of duty: Why customers contribute to firm-hosted commercial online communities. *Organization Studies 28*, 3 (2007), 347-376.
29. Wolf, T.V., Rode, J.A., Sussman, J., and Kellogg, W.A. Dispelling design as the black art of CHI. *ACM CHI*, (2006), 521-530.
30. Yamauchi, Y., Whalen, J., and Bobrow, D.G. Information use of service technicians in difficult cases. *ACM CHI* (2003), 81-88.