Sensemaking Sysadmins: Lessons from the Field

Eben M Haber
IBM Almaden Research Center
650 Harry Road, San Jose, CA 95120, USA
ehaber@us.ibm.com
+1 408 927-1224

INTRODUCTION

Computer system administrators play a crucial though unheralded role in maintaining the computer infrastructure that underlies much of modern life. As they work to configure, monitor, and troubleshoot complex computer systems, sensemaking is a necessary and important aspect of their work. The systems they manage are usually complex, e.g., something as conceptually simple as a web site may be comprised of dozens of components (including HTTP servers, web application servers, authentication servers, content servers, database management systems, network load balancers, etc.) distributed across multiple networks and multiple operating system platforms, and each of the components may have hundreds or thousands of configuration parameters. Troubleshooting or making changes to such systems requires understanding the interaction of all these components and integrating status information into a single picture. System administrators also deal with large amounts of data. For example, some monitoring tools generate gigabytes of log data per day, far too much for a human to read and understand directly, so evaluating the system's performance requires analysis tools. Finally, system administration is highly collaborative, as responsibility and expertise for various system components is usually spread across people and organizations. Understanding inter-component problems often involves many people bringing their separate expertises together to develop a common understanding of the source of the problem and its solution.

Sensemaking has been narrowly defined as “the process of searching for a representation and encoding data in that representation to answer task-specific questions” [10], and also more generally described as the process of creating “a sense of understanding a large, complex, problem, one with many interlocking pieces, sometimes ill-fitting data and the occasional bit of contradictory information” [9]. The work of system administration fits both these definitions, since in their tasks of configuring, maintaining, and troubleshooting large computer systems, system administrators have large amounts of data from disparate sources that must be collected, integrated, and analyzed to produce an understanding of system operation. This understanding is necessary for knowing whether systems are running properly, and for finding the cause and solution when systems fail.

Over the past two years I have been part of a group conducting ethnographic field studies in large data centers, observing the organization, work practices, tools, and problem-solving strategies of web, database, security, storage, and operating system administrators. Field studies offer insights into work that cannot be found in focus groups, lab studies, or surveys alone [5,6,8]. By examining administration work in context, we have observed how system administrators rely on sensemaking to cope with a complex environment. We have also observed how existing administration tools provide very limited support for sensemaking.

This paper is organized as follows. After a brief description of my previous and current work related to sensemaking, I will describe how the sensemaking activities of system administrators are best understood through ethnographic studies, which show real activities in a real-world context. I will then provide several case studies showing how system administrators engage in sensemaking as part of their work. Finally, I will discuss where existing administration tools fail to support sensemaking, and provide suggestions for how they might be improved.

BACKGROUND AND CURRENT WORK

My background in studying Sensemaking began in the early 1990's with my dissertation work on OPOSSUM [4], the user interface to a scientific database management system. The scientists using this system had immensely complicated scientific databases which had grown to the point where even their creators no longer understood the database structure. I developed approaches to metadata visualization that permitted the scientists to better make sense of their data. In the latter half of the 1990s, I spent four years working on SGI's MineSet™ [2], a commercial product aimed at helping users find trends and relationships in their data through data mining algorithms integrated with three-dimensional visualization and animation. From 2002 through the present I have been working on a project to study the work practices of computer system administrators through ethnographic field studies [1,7] with the goal of developing prototype tools to aid administration work .

ETHNOGRAPHIC STUDIES AND SENSEMAKING

Ethnography, defined literally as writing about people, is a technique commonly used in cultural anthropology where one immerses one's self in a culture for an extended period of time to better understand the culture's people and practices. Ethnography has been traditionally practiced for understanding foreign cultures, but in recent years it has been effectively used to understand the work practices of computer users in context, and to inform the design of computer systems that better meet user needs [refs]. Ethnography is almost always used to generate qualitative instead of quantitative data. Ethnographic techniques include observation, often as a full participant in daily activities, direct interviews, and collection of artifacts. Ethnographic accounts aim to provide a rich description of events with as much detail as possible, expressing not only what happened but also interpreting the meaning and significance of events [3].

Over the past two years, I have been part of a group conducting ongoing ethnographic studies of system administration at various large computing centers across the United States. So far, we have conducted ten such studies, observing approximately 25 administrators over a total of 40 days [1,7]. Over the course of these studies we have found ethnography to be very valuable in understanding the sensemaking practices of system administrators. Sysadmin sensemaking goes on in a dynamic environment of constantly changing tasks, system state, and priorities, stress is frequently a factor given the serious consequences of having a machine unexpectedly unavailable, and sysadmins often work together to share information and coordinate work. Furthermore, administration tasks are lengthy, involving many steps, decision points, and often many different people working together. These factors would be very difficult to assess and replicate in a lab setting, yet they are easy to see when observing the work in context. In addition, focus groups and surveys are often not sufficient since, in our experience, administrators do not always correctly describe the details of their work. In post-observation interviews, administrators have not always accurately remembered and identified how they spent their time, what their biggest problems were, and which tools they used. Furthermore, people get acclimatized to their environments and thus may not realize and report when their tools and practices are working particularly well or badly. Ethnography does have limitations, of course. It is time and resource intensive, and it is not a good way to quantitatively evaluate different techniques or tools. On the whole, however, ethnography has provided us a detailed and accurate picture of system administration sensemaking activities.

SENSEMAKING SYSADMINS

This section contains four case studies we observed that illustrate different aspects of sensemaking among system administrators: collecting and integrating data from multiple sources, analyzing large data sets, and two examples of group sensemaking.

Case Study 1: Integrating Multiple Sources of Information

Given the complexity of modern computer systems, administrators spent much of their time collecting and integrating information from multiple sources. Two good examples of this come from our studies of security administrators, the people responsible for monitoring computers and networks to detect ongoing attacks and vulnerabilities to future attacks. Security administrators rely on a wide variety of tools to detect new attacks, including automated scanning of all network traffic for patterns suggesting an intrusion. Throughout the day, security administrators receive e-mail alerts from these automated systems; most of the time the alert is a false alarm, but it is up to the admin to make the determination.

In one case we were observing Aaron, a junior security administrator whose responsibilities included evaluating e-mail alerts from the automated intrusion detection systems. Aaron checked his e-mail five to ten times per hour to see if any new alerts have arrived. One afternoon received one such alert which specified that network traffic was detected involving the IP address of a once-compromised system. While the system had been repaired, extra attention is given to such systems in case the fix was insufficient and the machine still compromised. Specifically, the alert indicated that a file was transferred via HTTP from the formerly compromised machine to another internal machine. Aaron then searched the HTTP log using command-line tools, finding the name of the file that was transferred (perftool-tar.gz), which he recognized as a computer performance test suite. Aaron then gathered further information, looking up the owner of the machine in question using an online database. He then did a Google™ search for the owner's name, and found a number of references to parallel programming, commenting that, "If this is a person doing parallel programming, it should be alright." Just to be certain, Aaron pointed his web browser to the machine in question and found it was being used as a web server for a research group doing performance analysis of parallel computers. This clinched it, the file transfer was legitimate.

The second example happened at a meeting where several security administrators were discussing past attacks. They mentioned a hacker who had used a tool called "ettercap." Being unfamiliar with this tool, one of the observers began searching the web for information on "ettercap" using his laptop on the local wireless network. A few minutes later, Aaron informed us that Fred, a security administrator working remotely, had detected this traffic and asked about it in the security administrators chat room:

```
Fred:   any idea who was looking for ettercap?
        dhcp logs say <observer's machine name> is
        a netbios name.  nothing in email logs
        (like pop from that IP address).

Fred:   seemed more like research.

Fred:   smtp port is open on that host, but it
        doesn't respond as smtp.  That could be a
        hacker defender port.
```

```
Aaron: we were showing how <hacker> downloaded
       ettercap.  One of the visitors started
       searching for it.

Fred:  ah, ok.  thanks.
```

The automated network monitoring had detected traffic related to the dangerous "ettercap" package. In the space of only a few minutes, Fred had identified the name of the originating machine, checked the logs for other activity by that machine, and probed the network ports on the machine. Fred could see that it was probably someone doing research, but checked the chat room to verify with the other admins.

One of the most important tasks for security administrators is to make sense of a huge amount of computer activity and determine which patterns are the result of legitimate activity and which indicate an attack. While they make use of automated software to detect suspicious patterns, admins must gather and integrate additional information from sources as disparate as log files, online phone books and personal web pages to make a final determination. There was no specific support for integration, however, it was either performed in the heads of the administrators or via simple ad hoc command-line tools.

Case Study 2: Analyzing Large Datasets

Sometimes system administrators must find meaning in very large data sets. One example of this is network monitoring for security administration, where the logs can exceed several gigabytes per day, obviously far too large for a human to read and understand directly. In one episode during our observation, Aaron (the security admin) learned of a new MyDoom virus that communicated on network port 1034. To find out if there was any suspicious activity on this port, he created an ad hoc command-line filter to examine records from the network monitoring log:

```
./bin/ra –xcz args.out –port 1034 | awk ‘{print
$7}’ - | awk –F.  ‘{print $1, $2, $3, $5}’ | sort
–u
```

He then copied the list into a file called `mydoom.o` to process it further by associating each IP address with its host name using the following commands:

```
for a in ‘cat mydoom.o’; do echo $a; host $a | awk
‘{print $5}’ -; done
```

Given the names and addresses of possibly compromised machines, he had the machines taken off the network, and began looking up the machines’ owners using an online database. He also looked up more detailed information about the operation of the virus through a Google™ search. One of the machines was clearly infected, so he called the owner to give instructions on how to remove the virus.

This pattern of activity was typical when working with the network logs: the administrators would use ad hoc filtering, temporary files, and online resources to verify their findings.

Case Study 3: Group Sensemaking I – All Together

As described above, sometimes computer systems are made up of dozens of components, with expertise about the different parts distributed across many different people and organizations. Most of the time these different people work independently, each making sure that their own part of the system is working correctly. Occasionally they must all work together to solve a problem, however, as we saw while observing web administrators involved in a "crit sit." "Crit sits" are *critical situations* that are initiated when a customer is extremely unhappy with the performance of a system. The idea of a crit sit is to bring all responsible people into a single room until the problem is solved, even if weeks or months are required. Conference calls and chat rooms are set for those who can't be physically present, and those peripherally related to the problem.

The crit sit we observed involved seven to ten people at various times, with each person having a backup for when they couldn't be present. The problem they were addressing was the intermittent and unpredictable failure of a web application, so the various administrators monitored their own components as they waited. As failure became evident, they collected data and shared possible clues. Between failures there were numerous discussions to develop theories as to the cause and strategies for a solution. The whole group would converse via voice and chat room, sometimes using the whiteboard to map out possibilities. Often two or three people would engage in sub-discussions by instant-messaging or face-to-face to work out specific issues. When data could not be gathered using the available tools, administrators worked together to create simple, ad hoc tools to do the job. At a number of points, finding the right person to do tasks became an issue as backups and primary sysadmins traded places and joined or left the room, conference call, or chat room. This crit took more than two months to resolve; the cause was a very subtle interaction between two of the components.

This case study shows system administrators performing group sensemaking. A problem existed due to interactions between the components of a very complicated system, and the experts on the different components needed to work together to understand the cause and find a solution. The overall strategy was a cycle of shared observation of the system in question, developing hypotheses as individuals, small groups, or the group as a whole, and implementing changes to attempt a fix. The administrators collected data using mostly standard management tools, occasionally building their own ad hoc tools when necessary. They used off-the-shelf communication tools such as chat rooms, telephones, and whiteboards for sharing information. There were no tools supporting data integration aside from the whiteboard and the online chat room.

Case Study 4: Group Sensemaking II – Focus on One

The final case study provides an example of a different style of group sensemaking, and also of the need for system administrators to understand the operation of their systems.

Web administrator George was assigned to deploy a new web server and connect it to an authentication server for one of the customer accounts he supported. His manager sent detailed instructions for the process, which included sample commands for the twenty or so steps to be performed under a very tight deadline. The first few steps for creating the new web server appeared to go well, but configuring the authentication server to work with the new web server produced a vague error message: “Error: Could not connect to server.”

For the next few hours, George was involved in increasingly intense troubleshooting. Through telephone, e-mail, instant messaging, and in-person conversations, he worked with seven different people, including his manager, the network team, his office mate, the architect of the system, a technical support person, a colleague, and a software developer. Each asked him questions about system behavior, entries in log and configuration files, error codes, and so on, and each suggested commands to run. Each sought his attention and trust, competing for the right to tell him what to do (see 7).

We refer to this collaboration pattern as “Seven People, One Command Line,” as various people participated in troubleshooting, but only George had access to the system. His manager wanted to know when the problem would be fixed and whether others should be redirected to help him complete the task on time. The support person wanted to resolve the problem ticket and end the call as quickly as possible. His colleague wanted to help within the limitations imposed by his own responsibilities. The system architect wanted to know if there was any problem in the overall design without being mired in the details. Other specialists waited for instructions to manipulate the subsystems they were responsible for.

The problem was eventually found to be a network port misconfiguration. George misunderstood the meaning of a certain configuration parameter for the new web server. George’s misunderstanding affected the remote collaborators significantly throughout the troubleshooting session. We witnessed several instances in which he ignored or misinterpreted evidence of the real problem, filtering what he communicated by his incorrect understanding of the system configuration, which in turn greatly limited his collaborators’ ability to understand the problem. George’s error propagated to his collaborators. The solution was only found by Thad, the one collaborator who had independent access to the systems, which meant his view of the systems was not contaminated by George’s incorrect understanding (see 7). Improved tools for sharing system state could have helped resolve the problem sooner, since other collaborators could have seen the un-interpreted information and possibly corrected George’s misunderstanding.

This episode also demonstrates the need that admins feel to truly understand the functioning of their systems. A revealing interchange happened near the end of the troubleshooting when Thad had found a solution and was trying to get George to apply it. For George, knowing the solution was not enough, he needed to understand why it how it would work. After a heated discussion via instant messaging about which network port numbers to use, George phoned Thad and they conversed as follows:

```
G:  What are you talking about? 7236?
T:  Yah
G:  We thought that it came in on 7137 and went
    back on 7236, but we were wrong, that 7236 is
    like an ACTPS listener port or something?
T:  It will still come in on 7135 to talk to the
    pdserver apparently...
G:  right
T: ...what's happening is it's actually trying to
    make a request back, um, though the 72... well
    actually trying to make it back through the
    7137 to the client...
G:  right
T:  ...the webseal client...
G:  right
T:  ...and it's not happening.
G:  I know.  I know that.  But I can't tell it
    to...
T: The instance, just create it with the 7236.
    Trust me.
G:  why? that port's not, that's going the wrong,
    that's only one way too.
T:  Trust me
G:  It's only one-way
T:  Do it!
G:  Do you understand what I'm saying?
T:  Just do it!  It only has to be one way, man.
G:  Why?
T: Cause it's the pdserver talking back to the
    webseal server.
G:  Yah, but how does the webseal talk to the pd
    to make some kind of request?
T:  7135 is the standard port it uses in all
    cases.  So we had it wrong.  Our assumption on
    how it works was incorrect.
G:  (skeptical) All right, all right. (sighs)
T:  Just try it, and if it doesn't work you can
    beat me up after that.
G:  I want to right now.
```

Thad kept asking George to “Do it!” but George insisted on understanding why Thad’s solution should work. George only agrees to go along once Thad starts to point out how their earlier assumptions had been incorrect.

This case showed an example of a different kind of group sensemaking where one person was the focus. George reached out to other people to help understand his problem, yet until Thad joined in, all contact with the problematic system was mediated my him. In addition, as the primary responsible person, George felt that he couldn’t just find a solution, he needed to understand it.

BETTER TOOLS TO FOR SYSADMIN SENSEMAKING

Throughout our studies of system administrators, we have found that the available administration tools do not do a great job of supporting admins’ sensemaking needs. Specifically:

- Sysadmins need to collect, integrate, and analyze data from many sources. In case study 1, the security administrators use a wide variety of tools and sources to determine whether a specific pattern of activity was legitimate. In case study 3, a group of administrators collected data from different components in a system to determine why the system was crashing. Yet in all these cases we saw no explicit support for this process, admins either kept data in their heads or used temporary files and ad hoc analysis tools.

- Sysadmins need to share information about system state with each other. This problem was especially acute in case study 4, where a problem persisted because one person had access to a system, and those helping him relied on his descriptions of system behavior.

- Sysadmins need situational awareness of their systems to ensure that they are functioning correctly (as seen with security administrators in case studies 1 and 2). The tools for monitoring system behavior were quite primitive, however, with security administrators relying on e-mail messages from monitoring tools and command-line filters of log files.

Administration sensemaking could be aided by application of improved tools. For example:

- Data collection, integration, and analysis could be helped by tools/workspaces that make the process more explicit. Since many admins are used to creating ad hoc tools, a workspace supporting end-user programming would be especially useful in creating more formal tools that gather and process information from different sources. This would also permit analysis for one problem to be applied to future problems as well.

- Sharing of system state during group sensemaking could be aided by tools explicitly designed for this task. Screen and command-line sharing tools exist, but they can be difficult to set up (especially when security is an issue), presenting an obstacle to their use. Sysadmins need something as simple and easy as a "work together" button that would quickly allow them to share system state with another admin.

- Finally, situational awareness could be enhanced through advanced visualization and data mining techniques.

CONCLUSIONS

This paper introduced computer system administrators as an excellent example of workers involved in sensemaking, and presented ethnographic field studies as an ideal way to study their sensemaking activities. Sysadmins work in a dynamic, stressful environment, interacting with many other people, aspects that are hard to replicate and evaluate anywhere but the field. Several case studies were presented to provide examples administration sensemaking tasks, such as data collection, integration, and analysis, both for individuals and groups. Finally, the limitations of existing administration tools with respect to sensemaking were discussed along with future directions for improved tools to support sysadmin sensemaking. As advances are made in the field of sensemaking, computer system administrators will be an ideal population to study and evaluate new approaches.

REFERENCES

1. Barrett, R., Kandogan, E., Maglio, P. P., Haber, E., et. al. (2004). Field Studies of Computer System Administrators: Analysis of System Management Tools and Practices. In *Proceedings of ACM Conference on Computer-Supported Cooperative Works*.

2. Brunk, C., Kelly, J., Kohavi, R. (1997). MineSet: An Integrated System for Data Mining. Proc. of the third international conference on Knowledge Discovery and Data Mining.

3. Grau, M.E. Definition of Ethnography. In C. A. Grant & G. Ladson Billings (Eds.), *Dictionary of Multicultural Education,* Oryx Press, 1997.

4. Haber, E. M., Ionnidis, Y., and Livny, M. OPOSSUM: Desktop Schema Management Through Customizable Visualization. (1995) Proc. of the XXI VLDB Conference, pp. 527-538.

5. Halverson, C. A., The Value of Persistence: A Study of the Creation, Ordering and Use of Conversation Archives by a Knowledge Worker, (2004) Proc. *37th Annual HICSS'04 – Hawaii International Conference on System Sciences, pp. 40108.1.*

6. Luff, P., Hindmarsh, J., Heath, C. (1999). *Workplace Studies: Recovering Work Practice and Information System Design.* Cambridge, MA: Cambridge University Press.

7. Maglio, P. P., Kandogan, E., and Haber, E. (2003). Distributed cognition and joint activity in collaborative problem solving. In Proc. of the Twenty-fifth Annual Conference of the Cognitive Science Society. Boston, MA. LEA.

8. Orr, J. E. (1996). *Talking About Machines: An Ethnography of a Modern Job*. Ithaca, NY: Cornell University Press.

9. Russell, D. M., (2003) Learning to see, seeing to learn: visual aspects of sensemaking. *Human Vision and Electronic Imaging Conference*.

10. Russell, D. M., Stefix, M., Pirolli, P., Card, S. K., (1993) The Cost Structure of Sensemaking, *Proceedings of ACM INTERCHI'93 Conference on Human Factors in Computing Systems.* pp. 269-276.