;login:

inside:

USENIX

The Advanced Computing Systems Association

document it with a picture

A Case Study

Introduction

Shortly after joining the Storage Group at HP Labs as system administrator, I was asked to add some disks to the group's production XP512 disk array. At that time, I didn't have much experience with that array model, so one of the researchers helped me navigate the console menus. We documented the whole process using a digital camera, which was a method of documenting sysadmin tasks I had never seen before. This method proved to be quite useful: A few months later, I needed to add more disks to the array, but I had forgotten some details of the procedure. The pictures were there to help me recall the necessary steps, saving me hours of research.

Digital photography is a technique particularly well suited for documenting tasks that require interacting with hardware, as it shows the steps taken chronologically. Well-chosen pictures can be self-explanatory and are often clearer than documentation manuals, because they show precisely the task being performed. It is a time-efficient technique, too: It adds little overhead to the task being performed and can help eliminate future mistakes, thereby saving more time.

Our department has developed a set of simple Perl scripts [Laff03] that enables us to generate a Web page quickly, right after we download the pictures to a laptop. There is no need to order the pictures chronologically by hand either: The camera itself assigns reasonable (sequential) filenames to the images.

After the Web page is generated, we publish it on our intranet and may add some comments later by editing the HTML code (see Figure 1). We usually document interaction with the disk array console this way, too. The photos of the screens are easy to read if taken correctly, as Figure 2 shows.

With some discipline, this is an efficient way to document a variety of system administration tasks. Taking some notes during the process helps, but even without comments the images usually serve well enough. Experience has also shown us that when we don't take pictures, we find ourselves regretting it later.

This approach is often useful for infrequent but complex tasks, especially ones that involve physical operations. In the next section, we will compare this approach to some alternatives, and will then summarize our experiences in the last section. Pictures and references are collected at the end.

Why This Is a Good Idea

Our main focus is documentation of system administration tasks as a training tool. Some of the ideas discussed in this article will apply to documentation of forensic procedures or Capability Maturity Model-related documentation of tasks, but that is not our primary objective.

Eric Anderson's thorough survey [Ande99] of the LISA papers published between 1987 and 1999 shows only a few papers discussing sysadmin training. Two papers ([Shar92] and [Hunt93]) describe the use of text-based trouble-ticket software as a

by Hernan Laffitte

Hernan Laffitte has worked in UNIX administration and as a consultant since 1993. He currently works as sysadmin for the Storage Systems Department at Hewlett-Packard.

hernan@hpl.hp.com

and Eric Anderson

Eric has been a researcher at HP Labs working on storage and information management since 1999. He wrote his dissertation on system administration at U.C. Berkeley.

anderse@hpl.hp.com

training aid for novice system administrators. The Indiana University USAIL Web-based education project is presented in Tomp96. Finally, Wendy Nather's seminal *;login:* article [Nath93] discusses the benefits of the school of hard knocks.

Suppose you are performing a complex system administration task and you want to document what you are doing. You want the documentation to explain the task to a new sysadmin, or maybe to yourself six months from now. You can use commands such as "script" and "tee" to keep a record of what happens on your terminal. Maybe take a screenshot or two to show how you use a graphical tool.

In fact, there are several ways to go about documenting a sysadmin procedure. A basic taxonomy of the different documentation techniques would be:

1. Text document: just writing the documentation manually.
2. Snapshots: Any time-frozen representation of the state of a system taken at different points in time: e.g., photographs, screenshots, cut-and-paste of the text in a Telnet session.
3. Time series: techniques to generate a continuous picture of a system as it changes through time: e.g., video recording, commands like "tee" and "script," a log of a Telnet session, or some screen recording utility such as Lotus ScreenCam.

We can of course create multimedia documents that combine the three techniques.

So, there are lots of ways of documenting a sysadmin task, of which taking a photo of the computer screen does not immediately appear to be the most efficient option. However, in some cases it can the best solution. Here are some reasons why:

- Physical procedures can only be documented by photos and video. Until we can buy a wearable Memex [Bush45] that records all our actions, a digital camera can be the next best thing. With a camera, you document what you are seeing now – and will probably see again the next time you have to perform the same task. See Figure 3 for examples of a "this part goes here" kind of operation that would be difficult to document in a non-pictorial manner.
- Pictures are more selective than video. It's easier to use them to show the important parts only. Browsing the pictures is faster, and less boring. Taking photos is also less intrusive than filming. As a downside, there is always the possibility of forgetting to photograph some important step.
- For some GUIs, taking a screenshot or running a screen-recording program is not feasible. Some reasons for this:
    - Hardware limitations: The consoles of some devices (disk arrays, printers) are simply too dumb.
    - Security: Even if the console of a dedicated device is a PC-like machine running a full-featured operating system, for security reasons, it may not be connected to the network. This makes it difficult to download a screenshot, even if one could record the shots to the local disk.
    - Self-discipline: In our case, we decided not to connect the console of our main disk array to the network. This way, we don't have the convenience of being able to wipe out all its contents from the comfort of our desks.
- A regular screenshot will not usually show where the cursor is or which button it's clicking on at the moment. A ScreenCam movie will show this information, but it's platform-dependent. Taking a photo of the screen is a reasonable compromise. See Figure 2 for an example.

- In a text-only environment, "script" and "tee" may be an alternative, but their output can sometimes be illegible. For example, CURSES-based tools often generate near-infinite sequences of control characters. Finding useful information in the resulting ASCII animation can be a chore (see Figure 5).
- Digital photos are cheap and convenient. As of August 2003, a 2-megapixel camera with 128MB of flash RAM can be bought for around $150 new or $100 used. This represents less than a few hours of the fully loaded cost of an administrator. So if the pictures save you a day of work, the camera has already paid for itself. These small cameras fit into a pocket, and the time cost to taking the pictures and uploading them to a Web site can be less than 10 minutes. Adding a few comments to the resulting Web page adds some time, but always less than writing a text-only document. For example, accurately describing in writing the steps to replace a printer's transfer kit would be more difficult than just having the pictures.
- Documents of a known "good" state before performing a complex operation can also be useful, especially when trying to debug a problem in some operation. In one case we failed to document a procedure because it was performed by a company service engineer. When something went wrong, we brought in more experienced administrators, but they could only see the current, broken state, rather than the steps taken to get there or information from before the operation. As a result, solving the problem took longer than necessary. Subsequent to this experience, we have set a new policy of documenting all changes regardless of who performs them.

Experience and Summary

Several months elapsed before I had to install a second disk group in the XP512 disk array. Of course, by that time I had already forgotten most of the procedure for adding the disks. Did I need to click on the button labeled "Maintenance" or the one labeled "Install"? Fortunately, the pictures were there to help.

The second run was also documented with the camera. This time it wasn't necessary to add as many comments as before, only to comment on the differences from the previous run. Currently, every time I have to perform some task on the disk array, I keep referring to these photos. I also take pictures of new procedures to help my memory the next time they need to be done. We now have more than 10 sets of documentation photographs.

The first set of pictures was taken before I joined the company, when the Storage Department acquired the XP512 disk array. Because we are a research group, the idea was to assemble the array ourselves, and we took pictures of the whole process for future reference. While installing the array, we realized that it was a natural step to also take pictures of what we were doing on the console and interleave them with the pictures of what we were doing at the hardware level (connecting cables, installing disks, etc.). The resulting set of pictures is a "story" that clearly reflects all the steps we took to install the disk array.

We believe that this technique will be even more useful at larger sites. A senior administrator could perform a task the first time and record the pictures, and then a junior administrator could use the pictures to perform the same steps later. If they also take pictures as they go along, then if a problem occurs, the senior administrator can easily review the steps that were taken.

Accurately describing in writing the steps to replace a printer's transfer kit would be more difficult than just having the pictures.

REFERENCES

[Ande99] Eric Anderson and Dave Patterson, "A Retrospective on Twelve Years of LISA Proceedings," *Proceedings of LISA '99 13th Systems Administration Conference*, November 1999.

[Bush45] Vannebar Bush, "As We May Think," *Atlantic Monthly*, July 1945.

[Hunt93] Tim Hunter and Scott Watanabe, "Guerrilla System Administration: Scaling Small Group Systems Administration to a Larger Installed Base," *Proceedings of LISA '93 Seventh Systems Administration Conference*, November 1993.

[Laff03] See *http://www.hpl.hp.com/ personal/Hernan_Laffitte* for source code samples and full-color example pictures.

[Nath93] Wendy Nather, "Think or Thwim: The Cold Creek Approach to Systems Administration Training," *;login:*, vol. 18, no. 4, July/August 1993, p. 22.

[Shar92] James M. Sharp, "Request: A Tool for Training New Sys Admins and Managing Old Ones," *Proceedings of LISA '92 Sixth Systems Administration Conference*, October 1992.

[Tomp96] Raven Tompkins, "A New Twist on Teaching System Administration," *Proceedings of LISA '96 10th Systems Administration Conference*, September 1996.

In conclusion, digital photography is an efficient and useful way of documenting sysadmin operations, and is excellent for reminders on infrequently performed tasks. It complements other approaches to documenting procedures, and should therefore be considered in the arsenal of techniques administrators can use to make their lives easier.

Future work being considered in this area includes:

- Find an efficient way to perform character recognition on the pictures of the devices' consoles.
- Newer digital cameras support audio clips. Having one of these perform speech recognition on the sound bits could further ease and improve the documentation process.

Acknowledgments

Figures

Figure 1a: A Web page generated by our Perl script from the contents of the digital camera. The three links under each thumbnail take the user to versions of the image in different resolutions.

Figure 1b: A detail of the same page, after adding comments. The HTML code could be more polished.
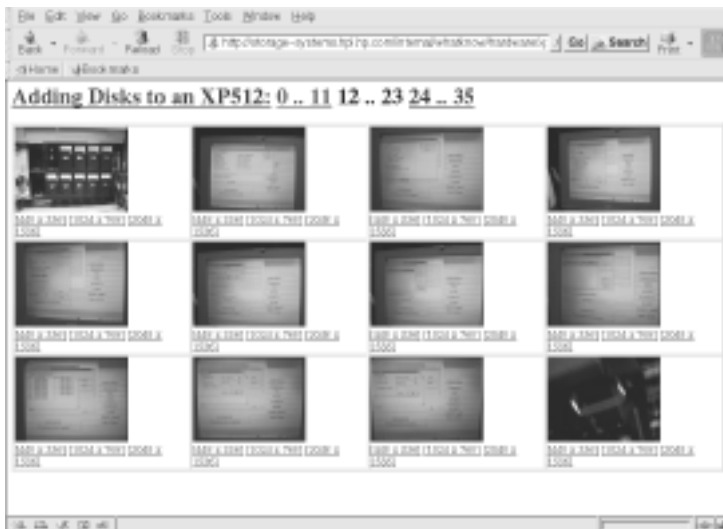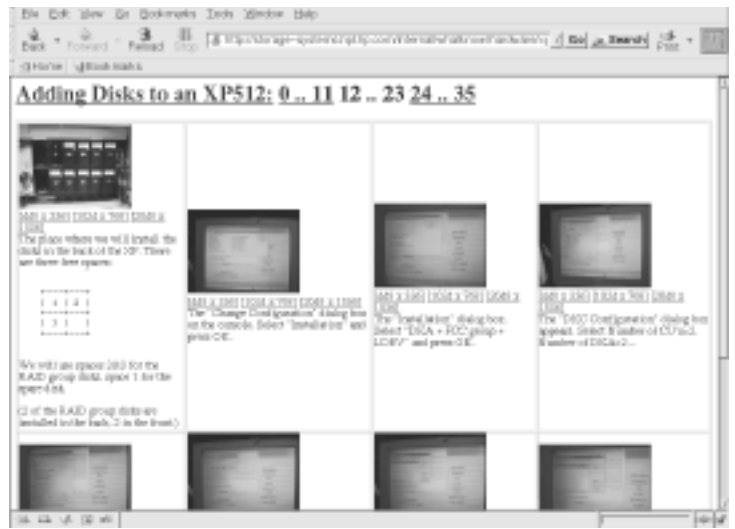

Figure 1a

Figure 1b

Figure 2: The text from a screenshot is usually legible. We generally use a Canon S30 digital camera, which provides 2048x1536 resolution (3-megapixel). The department also has two 2-megapixel HP digital cameras.

With our 3-megapixel camera, photographs of the screen are easy to read, provided they are taken with a steady hand.

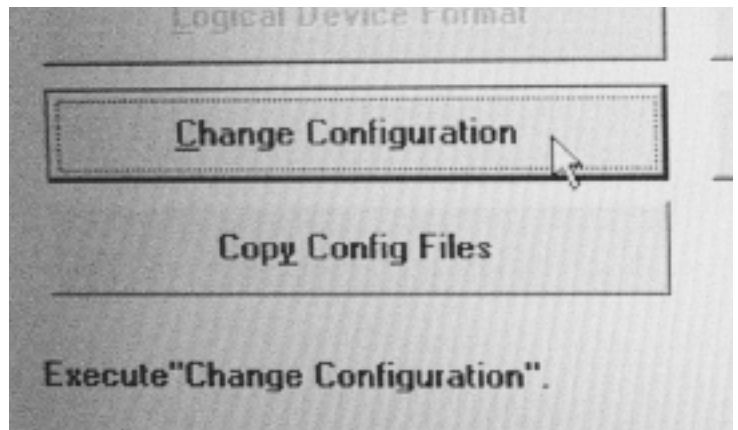This figure also shows how we can record the fact that the mouse pointer is here, clicking on this button, etc. This is a useful feature for recording the interaction with a GUI.

Figures 3a–3b: An example of "this piece goes here."
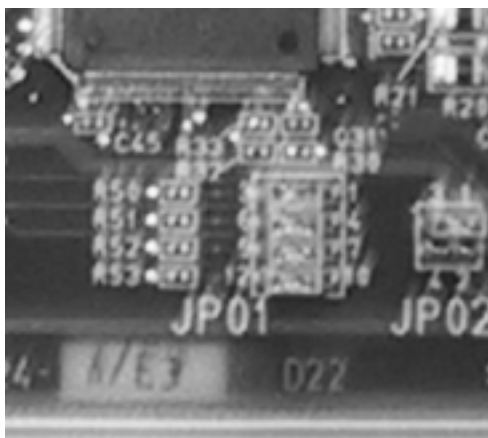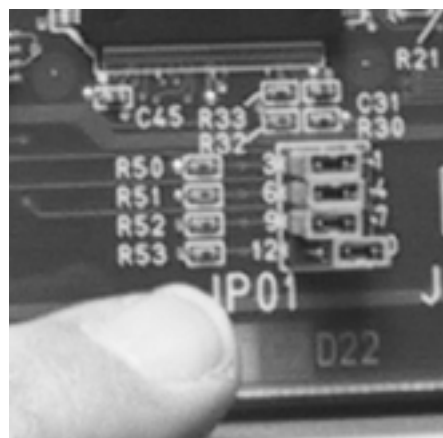

Figure 2

Figure 3a

Figure 3b

Figure 3c

Figure 3d

Figures 3c/3d: Moving dip switches around is also best described graphically, as this "before and after" sequence illustrates.

Figure 4: We also document the mistakes we make during the procedure. In this case, a disk we forgot to install. The second panel shows the empty slot; the third panel, the author's hand holding the missing disk.
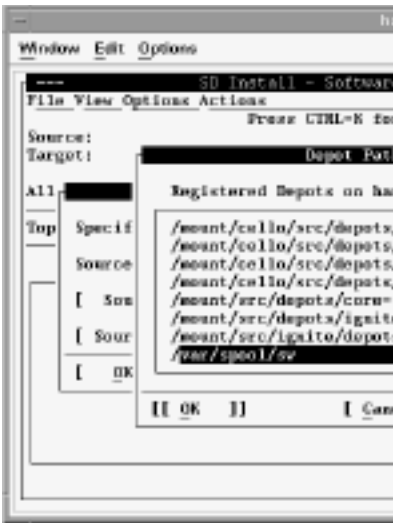
Figure 4

Figure 5a

Figure 5b

Figures 5a–5b: What the competition looks like. A screenshot of a CURSES-based GUI (left) and the output it generates when we record it using the "script" command (right). Tracking what the user did can be quite challenging.