MICROCOMPUTER
LABORATORY MAINTENANCE

Norman A. Garrett
Eastern Illinois University

Terry D. Lundgren
Eastern Illinois University

The continuing improvement of available technologies has brought unanticipated problems to microcomputer laboratory maintenance. A major problem area in state-of-the-art microcomputer labs is maintenance of the hard disks. Hard disks become corrupted, filled with extraneous files, and are subject to sabotage. The resource costs of maintaining the integrity of the hard disks can be enormous with concomitant loss of user availability. We present a tested, reliable software solution to maintain hard disks which is virtually maintenance free.

As microcomputer technology has improved over the past decade, a corresponding improvement in microcomputer laboratories has also taken place. The continuing decrease in price, coupled with an increase in power, has made it possible for many schools and organizations to install state-of-the-art equipment in their computer labs.

While this rich technology is a boon to the users, it has multiplied the problems of lab maintenance (Toperczer, 1991). The methodology for establishing and maintaining labs in the past is well documented (Dennis and Joseph, 1991). But the continuing improvement of available technologies has brought new problems which were not anticipated. For example, dealing with software on stand-alone, two disk drive machines typically required securing and distributing applications on floppy disk. Adding hard disks to the lab radically changes the nature of maintenance.

We will examine the problems involved in lab maintenance, particularly those related to hard disk maintenance, and present a feasible plan to maintain lab hard disks while minimizing maintenance resources. We suggest that any laboratory have a management strategy in place for optimizing resources (Johnson and Dileepan, 1990). However, we do not offer a management approach, rather we demonstrate a technical hardware/software solution to the problem of hard disks maintenance.

Hard Disk Problems

The standard microcomputer configuration today generally includes a hard disk. Regardless of
use as stand-alone units or through a network, the hard disk was not specifically designed for a laboratory situation and is typically the focus of intensive labor by laboratory personnel. Some of the kinds of problems faced on a daily basis are these:

1. Hard disks tend to accumulate “junk” files. These are files accidentally created by lab users. For example, leaving off the last colon in the COPY C:\DATA\*.* B: command results in a big junk file named B on the hard disk.

2. Users tend to use the same machine and reason, correctly, that if their files were on the hard disk rather than their floppy, access time would be quicker. A few users with big files can quickly clog a hard disk. Of course, the users feel no need to delete these files when their task is completed.

3. Software configuration files are altered by users. Many of today’s applications allow extensive customization of configurations. Users do need to know how to alter configurations. On the other hand, allowing them to do so causes instant deterioration of any lab uniformity.

4. Important files stored on the hard disk can be corrupted, often rendering the system useless until lab personnel can intervene. This corruption can occur from such simple operations as an improperly issued COPY command.

5. Critical files can be accidentally deleted, rendering applications useless until the files are restored, again by human intervention. For example, deleting COMMAND.COM or CONFIG.SYS with a careless DEL command.

6. Knowledgeable users (hackers) can cause intentional or malicious damage to the software on lab systems. For example, configuring a word processor to black/black on fore/background, or altering the AUTOEXEC.BAT file.

**Strategic Considerations**

The key to feasible hard disk maintenance in a lab setting is to minimize human intervention while solving the system problems stated above. In addition, a solution should not introduce artificial elements into the system that would mislead a user or create an unrealistic user environment.

For example, extensive menu shells offer one solution to disk management, but do not offer users a realistic picture of the user/MS-DOS interface. What is needed is good disk management which is reasonably user transparent. If menu shells are used, then the user should be given the opportunity to exit to the DOS command line. In a learning environment, it is essential that the user gain an understanding of the system rather than learn to press 1, 2, or 3 to access an application.

A solution must also be feasible. Consider that a common solution in the literature to dealing with public-access computing with a hard disk is a full, daily backup (Rubenking, 1991). Though clearly a correct solution, it is rarely feasible. Few labs have the time and personnel resources to fully back up or restore every machine, every day even with the fastest backup software.

Another often discussed problem of public-access computing is how to keep users from using the “dangerous” DOS commands such as FORMAT, DEL, COPY, etc... The length of the list depends on the administrator. A familiar solution is to remove or rename these commands if they are external and/or to patch COMMAND.COM to rename them if they are internal. However, this is not consistent with a policy which expects to equip users with a broad understanding of microcomputers and their potentials as they would be in a business environment (Lanasa, 1989). Artificial constraints instituted solely to control users do not provide a reasonable instructional environment. We counsel prudence in the development of such controls.

Generally, our solution has been developed for the typical business applications case. The solution can easily be modified to fit other situations. We assume an MS-DOS system and applications such as Lotus 1-2-3, dBASE, and WordPerfect. The methodology can easily accommodate a different configuration while allowing for elements that do not want or need regular maintenance.
Related Content

An Analysis of Process Characteristics for Developing Scientific Software
[www.igi-global.com/chapter/analysis-process-characteristics-developing-scientific/69627?camid=4v1a](www.igi-global.com/chapter/analysis-process-characteristics-developing-scientific/69627?camid=4v1a)

Information Systems Success and Failure—Two Sides of One Coin, or Different in Nature? An Exploratory Study
[www.igi-global.com/chapter/information-systems-success-failure-two/18641?camid=4v1a](www.igi-global.com/chapter/information-systems-success-failure-two/18641?camid=4v1a)

Users' Perspectives on Using Wikis for Managing Knowledge: Benefits and Challenges
[www.igi-global.com/article/users-perspectives-on-using-wikis-for-managing-knowledge/181674?camid=4v1a](www.igi-global.com/article/users-perspectives-on-using-wikis-for-managing-knowledge/181674?camid=4v1a)

The Importance of Ease of Use, Usefulness, and Trust to Online Consumers: An Examination of the Technology Acceptance Model with Older Customers
[www.igi-global.com/chapter/importance-ease-usefulness-trust/18275?camid=4v1a](www.igi-global.com/chapter/importance-ease-usefulness-trust/18275?camid=4v1a)