Developing a Catalog of Software & Hardware

Karin Young
Manager
Publications Services
Massachusetts Institute of Technology
77 Massachusetts Avenue 11-305
Cambridge, Massachusetts 02139
(617) 253-5973
YOUNG@EAGLE.MIT.EDU

Massachusetts Institute of Technology
4,400 undergraduates
5,300 graduate students
1,700 faculty
3,700 administrative and support staff

We support computer users in the MIT community.
Staff: 40 FTEs;
a number of part-time students

Abstract
MIT has diverse and decentralized computer assets, ranging from small microcomputers to supercomputers. Not surprisingly, centralized support for each computer user requires more resources than are available. The *Catalog of Computer Hardware and Software at MIT* was developed as one way to tap into local expertise within this environment.

Why Produce A Catalog?

Information Services, MIT's central computer-users support group, conducted surveys in the fall of 1987 and 1988. Respondents were selected randomly from among the entire MIT community. They were asked to check, on a list of 20 services, those they would like to see added or expanded. In both surveys, a catalog of software and hardware at MIT ranked first.

What Is The Catalog?

The catalog lists the computer hardware and software at 31 sites at MIT. It is not an exhaustive list of all computers at MIT, but represents a rich (though not proportional) mix of personal computers, workstations, minis, mainframes, and supercomputers on campus. Major peripherals, such as printers, special boards, and plotters, are included. Software is restricted to generic applications, such as word processing or programming languages, and special software thought to be of general interest.

Site information includes the name of a contact person who will answer questions about equipment and may also let people try out equipment on site. Nested indices for hardware, software, and computer systems refer readers to the site information pages. (See Exhibits 1 and 2.)

We polled 31 site managers about the computer systems, peripherals, and software they were using. We then entered the data into an electronic database linked to a powerful word processing system. The link let us automatically compile the data into a fully formatted and indexed catalog.

Goal Of The Project

The primary goal of the project was to publish a catalog of hardware and software that would help people to make informed acquisition decisions.

Secondary benefits of the catalog are to 1) to increase Information Systems' contact with the community through the cataloging effort, 2) provide a database of information that will help IS to be more market-driven in the services it provides, and 3) decrease pressure on Information Systems consultants and the Microcomputer Center to provide hands-on access to a diverse selection of hardware and software.

Chronological Summary

This is a short overview of how the catalog developed. More details are given in the section on Lessons Learned.

The planning phase started in February 1988. It involved three staff people: a librarian from the application software group, the Assistant to the Director of Information Services, and the Manager of Publications Services in Information Services.

We visited Harvard and discussed the catalog they had produced. We called people around MIT who were in charge of catalogs. None were automated, but in the Communications Office they were working on automating the MIT course catalog on a VAX workstation. They had hired a consultant who designed a database using Progress, a large database and application generator. He also wrote a program that tied the data to Interleaf Publishing Software. The index tags were incorporated into the database. After the data had been entered, someone just had to "push a button" and the whole catalog would come out of the printer, as formatted and indexed camera-ready copy. This approach sounded good to me.

In June of 1988 we presented our Service Plan describing the project to the IServ managers, and we encountered a lot of resistance. People thought that site managers wouldn't cooperate, the information would be out of date too quickly to be useful, that some system managers might give "bad" advice or some people might inadvertently draw wrong conclusions from the data in the catalog (e.g., that some old systems were best just because there were more of them), and that the time and expense estimated for the project were too high. These problems are discussed further in the next section, "Lessons Learned."

In January, 1989 we signed a contract with a consultant and ordered a SUN 3/60 with Progress and Interleaf. We drew up our interview form and began the effort to collect data through visits to the sites.

We passed the completed interview forms to a part-time student who determined how to classify each item and checked the product and vendor names. Inconsistencies had to be resolved and holes filled in. For example, site managers gave catalog part numbers rather than model numbers, or they didn't know exactly what they had, or they didn't know the vendor or source of an item. Questions were sent back to the site managers via e-mail and/or telephone.

Once the data was checked, it was entered into the database. Of course there were bugs in the software. Our first printed pages were a disaster. Data was entered wrong, classified wrong, spelled wrong, and lost altogether. We had to go back to the original interview forms to check the output, item by item. For many months it seemed like every time we made headway in verification and data entry, something else set us back a few steps.

The catalog finally went to the printer on May 2, 1990, two years and three months after our first planning meeting.

Lessons Learned

1. Need For A Data Classification Standard

There are many opinions about how to classify computer hardware and software. For example, is TeX a word processor, a mathematical package, or desktop publishing? Is the Mac II a microcomputer or a workstation? Rather than try to make up our own scheme and take the inevitable arguments from people who would disagree with us, we selected Data *Sources*, a published guide to computer products and companies, as our standard for our classification and spelling. The student who was checking data looked up each product in *Data Sources* to determine how to classify it and to check the product and vendor name.

2. Responsibility For Data Quality

Although the site managers were very enthusiastic about having a catalog, it wasn't their first priority, and they did not feel responsible for the completeness or accuracy of their site data.

Many of the sites were so big that the system manager didn't know exactly what was being used there. Some managers were unfamiliar with some of the older equipment at their site. Even though we took a technical person with us on the interviews, much of the information that we collected was incomplete or inconsistent with D*ata Sources*.

When we returned their site data to the site managers for their review, it became clear that we could not rely on them for a sufficient quality check. We had to rely on consultants within the IS organization. And we had to depend on ourselves to check the printed data for its completeness and coherence.

3. Data Follow-Up Was Backbreaking

We sorely underestimated time requirements for controlling the quality of our data. Our first schedule, in fact, didn't contain a separate line item for this task. We naively assumed that what we'd get from the site managers would be correct and, if there were holes, they'd be easy to fill in.

We sent our interview forms to the managers ahead of time, but no one filled them out before we sat down together. After our interview, we sent questions about the data to the managers via e-mail and then followed up with a phone call, if necessary. But the managers are busy people, and it was often difficult to get responses from them.

If we had spent more time in the interview phase, perhaps some of the follow-up could have been eliminated. This would have meant managing the system managers more efficiently and spending from three to five hours for the initial interview.

4. Getting Buy-In From IS Consultants Was Difficult

The catalog represents a new breed of service, one that is truly distributed. It provides a way for the community to make decisions without help from a centralized support organization. This can feel unsettling to the people in the support organization who are used to setting standards and making computing decisions.

We verified that the site managers would cooperate with the cataloging effort by putting a survey on a postcard and sending it with a letter describing our goals. We sent out about 36 surveys and got responses from everyone. Over 90% of the managers agreed to participate. This positive response from the community got us over the hurdle of whether or not the catalog was worth pursuing.

To allay the fear that people would bypass IS consultants for computer needs analyses, we narrowed our primary audience to more sophisticated users. Information Services' strength is in the microcomputer area, in particular, administrative users. In the catalog we focussed on the researchers and academics who have a rich mix of hardware and software, much of which IS doesn't fully support (e.g., various workstations and specialty software). People felt comfortable that researchers and academics have a strong base of expertise and would be able to network intelligently via the catalog for acquisition purposes.

As for currency of the data, we felt that sites make large investments in equipment and although things change, it is not so frequent or radical that the data would be totally useless in less than a year. We were able to confirm our hunch because our printed pages went out for site managers' review about a year after the initial data had been collected, and although there were additions and deletions, the "site flavor" remained the same.

Our worst resistance was to the idea of having to hire a consultant and buy relatively expensive new software and a workstation to automate the catalog production and generate indices with page numbers. In fact, it took a year to overcome this hurdle.

5. Producing A Catalog Of This Type Is A Major Effort

Don't underestimate the size of this effort! It turned out to be comparable to a large customized database application. The Sun 3/60 had 140 MB of memory, but we ran out of space halfway through the project and had to add another 300 MB.

Besides pages describing the computers at a site, we wanted the catalog to provide useful ways to index into the site data. For example, we wanted someone who was wondering what statistical packages were being used on DOS-based computers to be able to find answers without having to browse through the whole catalog. We also wanted people who knew what they were looking for, e.g., who's using TeX, to be able to find out with minimum effort.

This meant that almost every word in the site data needed to be indexed. Our problem was to convince IServ that we really needed all these indexes. People wondered why we just couldn't hire students to type the data into Microsoft Word. Once they were convinced that a database and indices were needed for managing the data, they wondered why we needed a word processor. Why not put the data into FileMaker 4 and print it out?

We made a mock-up of the catalog with site information pages and indices. Then we wrote out our publishing requirements and sat down with the consultants to see what could be done with a simpler, cheaper platform.

We looked at various microcomputer packages: R:Base and WordPerfect on a DOS machine, FileMaker 4 and Word on a Mac II, Ventura, Ingres on a mini, SQL on a mainframe, to name a few. These either couldn't do everything we needed (nested indices with page numbers, many computer records for one site record) or they also required a programmer to tie the data to the word processing system.

Originally we thought we could collect and enter our data in about 160 hours over a two-month period. Collecting, verifying and entering the data ended up taking two part-time students more than 700 hours over an eleven-month period. Data verification and quality control, which hardly even showed up on our first plan, took one person over 200 hours. The consultant had estimated that he would spend about 150 hours over three or four months. It actually took him about 600 hours. He said it was the most complex indexing he's ever done. Luckily, we had a fixed-price contract!

Another caveat applies to project management. Because this project is so complex and crosses department boundaries, many people are involved. In our case, all but one were working on the project part time. Project management was a real problem. It is, therefore, important to keep good records of every decision, issue, and specification so that things don't fall out of the loop or get mistranslated as they are passed on.

Systems Software

Disk/Tape/File Utilities

- Macintosh Series
  - **Disk Express (ALSoft, Inc.)**
    - *Planning Office, 170*
  - **Network DiskFit (SuperMac Technology)**
    - *Planning Office, 170*
  - **Symantic Utilities for Macintosh (Symantec Corp.)**
    - *Planning Office, 170*
- PC DOS Series
  - **Corefast (CORE International)**
    - *Planning Office, 173*
  - **Norton Utilities (Peter Norton Computing)**
    - *Bursar's Office, 81*
    - *Planning Office, 173*
  - **PC Tools Deluxe (Central Point Software, Inc.)**
    - *Planning Office, 173*
  - **XTree (Executive Systems, Inc.)**
    - *Comptroller's Accounting Office, 97, 98*

Miscellaneous Systems Software

- Apollo DN3010
  - **Domain PAK (Apollo Computer, Inc.)**
    - *Electrical Engineering and Computer Science, 108*
  - **DSEE (Apollo Computer, Inc.)**
    - *Electrical Engineering and Computer Science, 108*
- DEC LSI-11/23
  - **RSXLib (Data Translation, Inc.)**
    - *Nuclear Reactor Laboratory, 158*
- Macintosh Series
  - **MACSbug 6800 Debugger (Corvus Systems, Inc.)**
    - *Office of Financial Planning and Management, 161*
  - **Soft PC (Insignia Solutions, Inc.)**
    - *Planning Office, 170*
- UNIX Family
  - **Domain PAK (Apollo Computer, Inc.)**
    - *Electrical Engineering and Computer Science, 108*
  - **DSEE (Apollo Computer, Inc.)**
    - *Electrical Engineering and Computer Science, 108*
  - **GDB (Free Software Foundation)**
    - *Electrical Engineering and Computer Science, 110*
- VAX Series
  - **Code Management System (DEC)**
    - *Purchasing and Stores, 176*
  - **GDB (Free Software Foundation)**
    - *Electrical Engineering and Computer Science, 110*
  - **Module Management System (DEC)**
    - *Purchasing and Stores, 176*
  - **VAX Performance Advisor (DEC)**
    - *Purchasing and Stores, 176*

Print Utilities

- Macintosh Series
  - **Silicon Press (Silicon Beach Software, Inc.)**
    - *Microcomputer Center, 148*
    - *Planning Office, 170*
- PC DOS Series
  - **PrintMerge (Polaris Software)**
    - *Center for Real Estate Development, 86*
- Symbolics 3600
  - **DVI Previewer (MIT-developed)**
    - *Artificial Intelligence Lab, 70*
- VAX Series
  - **Imprint (Northlake Software)**
    - *Whitaker College of Health Sciences and Technology, 205*

Programming Languages

- Alliant FX
  - **FX/C (Alliant Computer Systems Corp.)**
    - *Aeronautics and Astronautics, 61*
    - *Earth, Atmospheric, and Planetary Sciences, 102*
  - **FX/Fortran (Alliant Computer Systems Corp.)**
    - *Aeronautics and Astronautics, 61*
    - *Earth, Atmospheric, and Planetary Sciences, 102*
- Apollo DN3010
  - **Domain C (Apollo Computer, Inc.)**
    - *Electrical Engineering and Computer Science, 108*
  - **Domain Fortran (Apollo Computer, Inc.)**
    - *Electrical Engineering and Computer Science, 108*
  - **Domain Pascal (Apollo Computer, Inc.)**
    - *Electrical Engineering and Computer Science, 108*
- Apollo DN320
  - **Domain C (Apollo Computer, Inc.)**
    - *Earth, Atmospheric, and Planetary Sciences, 103*
  - **Domain Fortran (Apollo Computer, Inc.)**
    - *Earth, Atmospheric, and Planetary Sciences, 103*
  - **Domain Pascal (Apollo Computer, Inc.)**
    - *Earth, Atmospheric, and Planetary Sciences, 103*
- AT&T 6386 WGS
  - **REXX (IBM)**
    - *Sloan School of Management, 187*
- DEC LSI-11/23
  - **FORTRAN for LSI (DEC)**
    - *Nuclear Reactor Laboratory, 158*
  - **VAX Pascal (DEC)**
    - *Nuclear Reactor Laboratory, 158*
- Encore Multimax 320
  - **Harris C (Harris Corp.)**
    - *Laboratory for Computer Science, 120*
  - **Verdix Ada Compiler (Encore Computer Corp.)**
    - *Laboratory for Computer Science, 120*
- Harris HCX-9
  - **Harris C (Harris Corp.)**
    - *Laboratory for Computer Science, 124*
  - **Verdix Ada Compiler (Encore Computer Corp.)**
    - *Laboratory for Computer Science, 124*
- HP9000 Series 300
  - **C/XL (Hewlett-Packard Company)**
    - *Artificial Intelligence Lab 67*
- IBM Model 4381
  - **APL (IBM)**
    - *Sloan School of Management, 186*
  - **Assembler (IBM)**
    - *Sloan School of Management, 186*
  - **FORTRAN (IBM)**
    - *Sloan School of Management, 186*
  - **Pascal (IBM)**
    - *Sloan School of Management, 186*
  - **REXX (IBM)**
    - *Sloan School of Management, 186*
- IBM PC RT (running UNIX)
  - **CLU (MIT-developed)**
    - *Electrical Engineering and Computer Science, 112*
- Macintosh Series
  - **Lightspeed C (Symantec Corp.)**
    - *Microcomputer Center, 148*
    - *Office of Financial Planning and Management, 161*
    - *Whitaker College of Health Sciences and Technology, 204*
  - **Lightspeed Pascal (Symantec Corp.)**
    - *Microcomputer Center, 148*
  - **QuickBASIC (Microsoft Corp.)**
    - *Microcomputer Center, 148*
    - *Planning Office, 170*
- PC DOS Series
  - **C Compiler (Microsoft Corp.)**
    - *Nuclear Engineering, 153*
    - *Nuclear Reactor Laboratory, 157*
    - *Office of Financial Planning and Management, 163*
    - *Planning Office, 173*
    - *Urban Studies and Planning, 198, 199, 201*
  - **FORTRAN (Microsoft Corp.)**
    - *Nuclear Engineering, 153*
    - *Nuclear Reactor Laboratory, 157*
  - **Macro Assembler (Microsoft Corp.)**
    - *Office of Financial Planning and Management, 163*
  - **QuickC (Microsoft Corp.)**
    - *Office of Financial Planning and Management, 163*
  - **Turbo Pascal (Borland International, Inc.)**
    - *Planning Office, 173*
- Stellar GS 1000
  - **FORTRAN (Stellar Computer, Inc.)**
    - *Aeronautics and Astronautics, 62*
- Sun 3 Series
  - **G++ (C++ compiler) (Free Software Foundation)**
    - *Mathematics, 144*
  - **GCC (C compiler) (Free Software Foundation)**
    - *Mathematics, 144*

Exhibit 1. Pages from Indexed Listing for Systems Software.

Health Sciences and Technology

Department: Academic Computing Facility, Health Sciences and Technology
20A-111
Contact: L C Manager
20A-111 3-
lc @hist .mit.edu
Availability: Information only

Shared Facilities

Location: 20A-111

Workstations
DEC VAXstation II (1)
DEC VAXstation II/GPX (1)
IBM PC AT (5)

Summary

Mid-Range Computers
Apple Macintosh II (2)
DEC VAXstation II (1)
DEC VAXstation II/GPX (1)

Microcomputers
Apple Macintosh SE (10)
IBM PC AT Model 339 (6)

Apple Macintosh II

Operating System: MacOS
Use: Academic, Administration, Research

Input Devices

Scanners
HP ScanJet

Memory Devices/Boards

National Semiconductor N523D

Output Devices

Laser Printers
Apple LaserWriter IINT

Applications Software

Graphics
Silicon Beach SuperPaint

Word Processing/Text Editing
Design Science MathType
Microsoft Word
WordPerfect

Health Sciences and Technology

Data/Telecommunications Software

File Transfer/Terminal Emulator
Freesoft Red Ryder

DEC VAXstation II

Operating System: Ultrix
Use: Academic, Administration, Research

Magnetic/Optical Storage Devices

Hard Disks
Emulex EMS/180 70 MB

Output Devices

Laser Printers
Apple LaserWriter Plus

Terminals/Monitors

Terminals
DEC VT100

Applications Software

Office Automation
Uniplex Advance Office System

Word Processing/Text Editing
AT&T Device Independent TROFF
Free Software Foundation GNU EMACS
Scribe Document Production System

DEC VAXstation II/GPX

Operating System: Ultrix
Use: Academic, Administration, Research

Magnetic/Optical Storage Devices

Hard Disks
Emulex SM802 220 MB

Terminals/Monitors

Terminals
DEC VT100

See DEC VAXstation II for software listing

Apple Macintosh SE

Operating System: MacOS
Use: Administration

See Apple Macintosh II for hardware listing

See Apple Macintosh II for software listing

Exhibit 2. Pages from Site Information.