

WARD SHAW
Director
Colorado Alliance of Research Libraries

Aristotle Meets Plato in the Library Catalog: Part I

This paper is part I of a presentation titled "Aristotle Meets Plato in the Library Catalog." In it, I hope to set forth some aspects of the theoretical context, or point of view, from which we at the Colorado Alliance of Research Libraries (CARL) approach the design and implementation of what the organizers of this clinic have called "user friendly" systems, to describe a bit the organizational and systems setting within which we work, outline some of the design principles that guide our development, and provide a brief overview of the system as it exists today. Part 2, by Ken Dowlin, will discuss the system in an application context at the Pikes Peak Library District in Colorado Springs. The system in question is one developed by the Colorado Alliance of Research Libraries, and available for installation elsewhere through Eyring Research Institute, to whom we have granted a marketing license. It forms the basis for MAGGIE III, the system in Colorado Springs.

The Theoretical Context

You will be relieved to learn that this is not Philosophy 101. However, as we try to address the question "What is user friendly?" it is important to uncover some basic assumptions that underlie our particular implementation of a public system. Let us examine our theoretical context, with the understanding that all of it is emphatically arguable. First, a public catalog is an information system. Information is the name of a process; specifically, the process by which people become informed. The process by which people become informed is closely related to, or maybe the same as, learning. The name for sparking learning is *teaching*. Hence, an important characteristic of public systems is that they teach, and one measure of their utility is the effectiveness with which they teach. Teaching, as any

who follow debates surrounding educational policy will appreciate, is not well understood. One is led inevitably to the conclusion that we do not know what we are designing or at least that we do not have any guaranteed rules to follow.

Aristotle was the champion of the *a posteriori* method. If he wished to learn about a triangle, for example, he would analyze its parts and the mechanisms of their assembly by observation. He invented classification and, for all but the name, the scientific method. Plato, on the other hand, concentrated on the *a priori* method of learning. If he wished to learn about a triangle, he would consider its “triangleness” and draw logical conclusions from that concept. For him, the whole was both greater than and different from the sum of its parts.

We have applied Aristotelean methodology with considerable skill and marvelous detail in the construction of our classification schemes, MARC records, analytics, authorities, etc. in the design of research libraries and their traditional access tools. The method has served us remarkably well in providing conceptual structures for managing and controlling enormous resources, and its use was dictated by the technologies available. The difficulty, of course, is that the tools we have constructed are complex in direct relation to the fineness of the analysis they represent and require of their users intimate knowledge of system structure as well as discipline structure. Divergence of the two structures is inevitable and extremely difficult to control. Part but not all of this difficulty is, to be sure, a function of the relatively inflexible (expensive) technology of their traditional implementation. A large portion of the problem is that one must force one’s thinking into the analytic patterns upon which the system is constructed, and it is thus exceedingly difficult to have new ideas. As McLuhan says, “the medium resists” and mightily.

Martin Heidegger, a twentieth-century phenomenologist, has written and spoken in detail about the concept of a tool, pointing out that a hammer, in the hands of a carpenter, is an extension of his arm. The carpenter uses the hammer to drive nails with wonderful efficiency and without thinking about it. While I do not have to know much about the hammer to pick it up, I must think about it in detail before I use it if only to avoid pounding my thumb. But Heidegger says that I am much more likely to conceive new uses for the hammer precisely because I see it as a tool for pounding rather than as a tool for driving nails. I look at its “hammer-ness,” as might Plato, and draw *a priori* conclusions from the concept. In this case the medium also resists but potentially productively. And the more general (or “platonic”) the tool, the more productive the resistance might be.

With electronic technology, the challenge is to enable users to manipulate our Aristotelian structures in Platonic forms, driving the systems to

explore what users conceive rather than what we have “analyzed in.” That we believe is the heart of “user friendliness” and is the sense in which we offer the title of this presentation. It is the basic context from which we attempt design.

The Organizational Context

CARL is a private nonprofit corporation in Colorado that has as members the libraries of the University of Colorado at Boulder, the University of Northern Colorado, the University of Denver, the Colorado School of Mines, as well as the Denver Public Library, and the Auraria Library which itself serves a consortium of three institutions of higher education in Denver. These are different kinds of organizations. They are state-supported, city-supported, and private. They are large general academic, large public, small special academic libraries. They differ in size from the University of Colorado—Boulder Library, a member of ARL; to the School of Mines Library serving a specialized academic clientele. They are also alike in certain important ways. They all have, as a part of their reason for being, the need to support graduate-level research, they all support large numbers of undergraduate students, and they all have a commitment of one kind or another to serve a wider user population than that of their immediate campus or city.

Governance of CARL is via its Council of Members consisting of the directors of each of the member libraries. In addition, CARL has a board of directors (not to be confused with the library directors), but in practice policy is set by the council.

CARL exists to create a single research resource for the various publics served by the member institutions. Said another way, CARL manages the collections of member institutions as if they were one collection. In order to accomplish this we have undertaken a whole series of network programs. The Colorado Organization for Library Acquisitions (COLA), for example, is a CARL program for cooperative acquisition of expensive material. It differs somewhat from other similar efforts in that the material purchased, although housed in the member libraries, is owned by CARL. We are developing a considerable collection enhancing those of the members. We also cooperatively purchase supplies and equipment for the members, when volume can generate savings. CARL's major program is the network online system. In order to create a single research resource, we needed one common mechanism to identify, locate, and control items throughout the network, and we also needed (and still need) a system for rapid, site-independent document delivery.

Design Principles

In creating our system, we attend to several design principles which are derivable from the theoretical and organizational contexts just described. I haven't time, obviously, to discuss them in detail but will briefly outline a few of them to provide a flavor of our approach.

First, the approach we use is heuristic, rejecting the algorithmic and simulation approaches as variously cumbersome, slow, and requiring impossible degrees of prior specification. As a result, our design principles are essentially statements of supposed value, and in some cases they are in direct conflict with each other. Each should be preceded by some substantial qualifier such as "generally, in most cases, it is probably the case that...." Negotiating between the principles requires constant trade-offs and modifications. Some principles regarding the overall system follow:

- The system must make it easy for users to view the network as a whole.
- The system must support local differences in both policy and practice.
- The system must promote experimentation.
- The system must provide very fast response time.
- The system ought not to require the user to understand the structure of a bibliographic record or of its associated files but rather ought to promote and support the construction of his own concept of organization. (We are indebted to Christine Borgman for alerting us to the idea of the user's "conceptual map.")
- The user must feel in control of the system, and not the other way around.
- The system must adapt to the user's skill level.
- The user should be able to get real results very quickly and then be able to experiment with variations very easily so that he may use the system to "explore."

Some principles from a hardware/software point of view are:

- Both hardware and software must be modular in design, allowing relatively easy changes to part of the system without dire consequences for the rest.
- Pass constant values to software as data.
- Separate message content and message form.
- Keep data structures flexible.
- Minimize disc accesses.

And some principles relating to screen design:

- Avoid library jargon and especially avoid computer jargon.
- Keep screens uncluttered.
- Avoid cuteness.

- Provide a cursor at the spot user typing will appear, and make that spot consistent from screen to screen.
- Don't tell users they have done something wrong. Rather, let results speak for themselves and provide positive suggestions. Assume that users are in control.
- Don't use blinking fields or reverse video.
- Systems have style. Keep it consistent.
- Pay attention to layout as well as content.

In summary, users know best what they do albeit sometimes with considerable professional help. System designers know best what the system can do. The goal of user friendliness is to provide a powerful, flexible, informative way for users to drive and control the system to their various ends. It is emphatically not to presume their ends or to channel their thinking according to predefined routes.

The System Overview

I'd like now to give you an overview of the CARL installation and the software (Pikes Peak Library District has a different configuration). The CARL hardware base is an eight-processor Tandem Nonstop II system. Each processor has 4 million bytes of main memory—32 million for the current system. There are 6 billion bytes of disc memory for the files. In the six library sites, 390 terminals communicate with the system via various network communications equipment.

Bibliographic records in the database come from all six institutions. From the system point of view, these records are organized in a common way and each field in each record contains an ownership bit map to indicate which institution "owns" which field. From the user's point of view, however, the records are organized by institution—that is, the user searches and examines records one institution at a time. Early versions of the system required a cumbersome reentry of each search when switching from one institution's files to another's, and more recently we have made that switching extremely easy. Ultimately we will support global searches. This progression was designed for political reasons. Individual institutions are wary of potential work loads on less heavily worked library subsystems such as interlibrary loans created by users from other institutions looking directly at their records. This fear has eased considerably with experience, partly because users who identify items they want at other institutions tend to go there directly rather than use traditional interlibrary methods to get the material. As these perceptions have changed, the system has changed to reflect (lead?) new concepts.

The software is organized into four distinct modules. First, the background software builds the database and creates the necessary indexing. Records are taken from OCLC, Autographics, and one or two other sources that members create as a result of their own cataloging activities. The software converts these records into our internal format and maintains the appropriate indexes. The various local fields are processed to create item records for circulation. The second software module is the public access catalog or PAC which provides searching of and switching between whatever data are resident on the system.

The third program module is the circulation system. This is a full-service system supporting charge, return, inquiry, holds, recall, tracers, overdues, fines, lists, letters, reserves, conversion, statistical reports, and secured full edit control over all files and records. We interface directly with various academic computing centers for the transmission of accounting data generated by system activity. Of primary importance is that circulation status of items shows up instantly in PAC so that users have up-to-the-minute information about availability of items they discover.

The fourth software module is bibliographic maintenance. Maint, as we call it, is used primarily for editorial changes to the MARC records. All fields are fully editable, and the program performs format checking and correction where appropriate to ensure MARC compatibility. Additionally, users can add and delete records. All changes are immediately processed and reflected in PAC and CIRC.

The fifth module is acquisitions. Currently ready for beta test in one of the member libraries, it is scheduled for systemwide installation in the summer. The sixth module, serials control, is now in design. User access to these modules, as well as to Tandem or locally developed services, is available and secured through NEWPEX.

The CARL database at the moment contains 1.85 million institution-unique bibliographic records and perhaps 3 million holdings. In addition to the 300 dedicated terminals, we provide free dial-up access to PAC, currently handling about 150 calls per day. We average about 1,200,000 message transactions per day with an average response time of .4 seconds. By the end of 1986, we anticipate a database of 2.5 million records and 450 dedicated terminals, generating 1.8 million daily transactions. Over 20,000 people use the system on a typical day.