5-1-2001

Why I Still Teach Dialog

Carol Tenopir
University of Tennessee - Knoxville

ONLINE DATABASES

BY CAROL TENOPIR

Why I Still Teach Dialog

A GROUP OF SPECIAL librarians who run information services for multinational corporations recently told me what they looked for in new MLS graduates. They want people who feel comfortable learning and searching multiple online systems, teaching end users, and choosing the best resources.

Several in the group complained that, although their new hires were excellent web searchers and web page designers, they did not have enough experience with fee-based online services. One manager said she looks to MLS graduates for less common attributes; she wants people who understand how information systems are structured, can search fee-based systems with confidence, and can formulate good search strategies.

More than one librarian was disappointed that new hires, as opposed to those five years ago, "didn't even know how to search Dialog." I was told, "If someone has good Dialog searching skills, I can teach them any other system." These librarians wanted to know why some library schools no longer require or even include Dialog searching in their curricula.

Feeling guilty

Even though all accredited LIS programs in the United States (and many others elsewhere) provide free student accounts to Dialog, many contemporary students see such fee-based, text-only online services as old-fashioned. They are much more interested in honing web search and web design skills. Because they have searched the web and library online catalogs for years, many consider themselves experts, with no need to select electives that focus on online searching. Some faculty members feel

Carol Tenopir (ctenopir@utk.edu) is Professor at the School of Library and Information Science, University of Tennessee at Knoxville

guilty about teaching how to search commercial services, because it smacks of training rather than education and because they worry that the system will become obsolete.

I even feel slightly defensive—am I old-fashioned, or not academically rigorous enough, because I still teach Dialog? Adding to my sins, I introduce it in a required, first-semester course, so that every student who earns a master's degree at my institution has a minimum of 15 hours online with Dialog (and most end up with many more hours).

My modus operandi

My conversations with these corporate librarians reassured me that my instincts are sound. Still, I should explain why I still teach Dialog, even to students who will become school library media specialists, archivists, academic reference librarians, or catalogers.

The first thing I tell my students is that most online services they will encounter have layers of interfaces, meant to make the systems easier to use, but also to hide how the search process really works. The systems may look like a Cadillac on the outside, but you can't tell how the engine works.

DialogClassic's command-driven interface is like a hot rod with the hood and body stripped away, so a searcher can see exactly how and why it goes. Even if students never use Dialog again, learning DialogClassic will teach them how all information retrieval systems (including web search engines) work.

Underlying structure

Whether librarians are searching or teaching patrons to search SilverPlatter, FirstSearch, ProQuest, Dialog, the web, or any other fee or free online system, they must know how the information they are searching is structured. Certain de facto standards have evolved over the last three decades to form the basic structure of the search engines that power all online systems. That structure allows a full range of search features that make the "car" go where we want. DialogClassic reveals this structure.

Sources are separated into records, each of which is broken down into fields that also may be broken down into individual words or phrases. (Web sources have a similar structure, based on HTML coding.) DialogClassic allows students to view records in a format that includes field names or tags, or to experiment with displaying each separate field alone.

Dialog Bluesheets not only display a typical record for each database with field tags indicated, they clearly identify every field in a record; describe whether it is searchable, sortable, or displayable; and show how to search it.

All students must understand how machine indexing creates the dictionary files/inverted indexes at the time the database content is loaded. For every field in every database, the Dialog Bluesheets show exactly how the system puts that field into the index or indexes. They tell which fields are in the subject-related "basic index" (which is the index searched by default) and which fields are put into separate "additional indexes." We discuss the advantages and disadvantages of keeping authors separate from subjects; I then have them look at other systems, such as nexis.com, that put all indexes together.

Machine indexing is a new concept to most students; they wonder how they can remember the differences among systems. With Dialog there is no guessing involved—if students can read a Bluesheet, they can see that some fields are word indexed, others are phrase indexed, and some are done both ways. I show them Dialog, then challenge them to look at all the systems we use and try to determine, without the luxury of BlueSheets, which fields in those systems are word indexed and which are phrase indexed. This helps these students move from system to system as better searchers and troubleshooters.

Most students don't seem to understand that when they search they are searching an index rather than the whole record. Comparing online searching to back-of-the-book indexes helps some-

what, but the Dialog Bluesheets really make the point. Dialog's Expand command lets students open a window on the index and see what words and phrases are available for searching. Expand is essential to understanding the index structure. Although not unique to Dialog, many major systems our students search (such as Dow Jones Interactive) don't have an equivalent to Expand. In these systems, searchers never actually see the index they are searching.

Basic search features

Once they understand the underlying structure of records and indexes, students are able to test a range of search features on DialogClassic. It's the most powerful search engine of the Boolean logic–based options available. While I challenge some students to become power searchers, others just need to be exposed to basic features they will find in any online, CD-ROM, or web system.

Fundamental search features such as Boolean operators, proximity operators, and combining sets are all easily taught in Dialog. DialogClassic shows the students every set they create and every step of the process. They can see mistakes—and the system doesn't correct them! It offers the most flexibility in both set building and recombining sets to alter results.

Thus, DialogClassic is an excellent tool for teaching search strategies. I encourage students to recombine sets in different ways to see how it affects their results. We discuss how friendlier systems for end users might describe or present the concepts of connectors and sets. I challenge students to try stripping away the interface layers on their favorite library system to identify where Boolean connectors are being used and where proximity operators are automatically imposed.

Dialog assumes nothing

One reason DialogClassic is inappropriate for end users or infrequent searchers is that it adds nothing—what a searcher inputs is taken literally. The system assumes that users know what they are doing. This makes novice information professionals think and become not only better searchers but also better systems designers. After students make mistakes, we discuss how a system could be designed for end users to anticipate or correct common mistakes.

DialogClassic puts the burden for correctness and sound strategies on the *searcher*, which makes us think about how a redesign could put the burden on the *system*.

Dialog, for example, has no features for automatic plurals, matches between acronyms and spelled out versions, corrections of misspellings, or matches between British and American spelling. Thus, students learn to think of word form variations, the need for authority files, and the complexity of language. We compare results in DialogClassic with a system like Lexis-Nexis that offers some of these features. We discuss what other kind of automatic features could be part of systems for end users.

Controlled vocabulary

Many files on Dialog offer controlled vocabulary searching. This allows me to introduce the topic of human indexing. We try the same search on the

---

> ## End users don't need to know what is under the hood, but information professionals do

---

same databases using free-text searching and then using descriptors. We look at the structure of thesauri and discuss how this structure might be better-integrated into system search features. We talk briefly about jobs as indexers and the future of indexing but mostly focus on improving precision with descriptors and the advantages and disadvantages of descriptor searching. Since not all of Dialog's databases have descriptors, we compare the process of searching with and without descriptors.

Dialog's multifile search feature OneSearch appeals to students who are accustomed to searching megafiles, both on the web and with systems like ProQuest, Nexis, and Dow Jones Interactive. I have them try the same searches in single files and in various combinations of multifiles. These remind them that more information is not always better and that the vocabularies used in different sources may differ greatly.

Dialog offers variety

Dialog also offers a good range of subjects. We can compare the thesaurus for ERIC with Medical Subject Headings and PsycInfo. Students interested in science, medicine, or social sciences can search a wide array of choices. Everyone likes being able to search newspapers, general interest magazines, and information science literature. The humanities choices are not as strong but still of interest to nearly everyone. This makes Dialog much more appealing to the wide range of students in a required class; we can focus on more specialized sources in electives. Full-text, bibliographic, and directory databases help teach variations in structure, as well as the basics of reference source types.

DialogClassic also offers some variety in search engines and interfaces. Dialog's statistical/relevance ranking search engine, Target, is not the best, but it does allow students to try the same search in the Boolean logic system and a statistical system. They then can compare relevance ranking schemes by trying other web search engines.

For interfaces, I start with the plain, ugly, and no-layers-added command mode of DialogClassic. Once students know what happens underneath interface layers, I have them try DialogWeb. Again, they can compare the same searches in the same files. This helps students understand what is added at the interface layer, what is part of the file content, and what is underlying structure.

Betty Jo Hibberd, Dialog's academic program director, also recommends that students use the Dialog Intranet Toolkit. It allows students to set up individual search forms and, "with a little bit of HTML skills," they can create a customized interface to Dialog. "This is the kind of value-added service being expected from the librarian today," she said.

Not the final answer

For power searchers, the old-fashioned DialogClassic remains the best. For new students in an LIS program, DialogClassic helps them understand the workings of the systems they will be searching, teaching, or designing.

End users don't need to know what is under the hood, but information professionals do. By revealing all that, DialogClassic promotes understanding and confidence. It doesn't matter if they never search Dialog again.