[1]

Modern Libraries

Björn Voigt[2]

Introduction

In the department of Computer Science has been an interesting project since sommer term 1995. It is called »CSLIB 2000« (Computer Science Library 2000). I have made a lot of experience in this project and so I decided to present a paper in the lesson »English for Computer Scientists«. I decided to choose some interesting themes and some own ideas which are not subject of the CSLIB 2000 project.

The number of publications has been increasing all the time.

If you want to be successful in managing the collection of publications you will need a well organized library and a good computer system.

But are libraries only a collection of publications? Ken Dowlin, city librarian of San Francisco and author of the book »The Electronic Library: The Promise and the Process« says in an article [1], libraries are also »socialization facilities, communication facilities and protected public spaces – even icons of the community«. Up to now computer systems of libraries can only handle documents and they do this very superficial, because they can not really read. The communication skills of computers are still not used in libraries.

Library	Location	number of documents
Russian State L.	Moscow	30.000.000
Saltykov-Scedrin L.	Sant Peters-burg	28.500.000
L. of Congress	Washington	26.000.000
State Library, house 1	Berlin/ Mitte	6.900.000
State Library, house 2	Berlin/ Tierg.	3.700.000
Humboldt University	Berlin	2.400.000

figure 1: The biggest libraries (World/ Berlin)

[1] Photo: Russian State Library (Lenin's Library)

[2] Homepage: http://www.cs.tu-berlin.de/~bjoern

The biggest libraries

Before I present some new electronic concepts for libraries let us take a look at some existing libraries.

The Russian State Library (photo) is the biggest library all over the world. This statement is right if you compare the number of documents (books and journals, figure 1). The Russian State Library still works with classical index card catalogues. Some of the catalogues are complete and nearly perfect. So we can come to the conclusion that it is possible to handle large libraries without computers.

The State Library Berlin

In Berlin we also have some big libraries. The two houses of the State Library have a good standard. So they have a model function when we think about modern libraries.

The house two of the State Library is situated in the Potsdamer Straße 33 in Tiergarten. In this library you can use a variety of index card catalogues (alphabetical, systematical, geographical etc.). The link catalogue Berlin Brandenburg which contains all books and journals of the big public libraries in Berlin and Brandenburg is available in micro fiche form. The librarians access to this catalogue with computers.

Instead of one large reading room there are many divided reading rooms. Especially for laptop users there are three cabins. But now around 50 percent of them use laptops in the reading rooms. Because of the noise of the laptops and of the copy machines it is not very silent in the reading rooms. Near the reading rooms there is a cafeteria, too.

The library is divided into different sections. For special interests there are, for instance sections of music, East Asian and Orient and CD-ROMs.

In the entrance hall there is a T-Online terminal. But this ter-

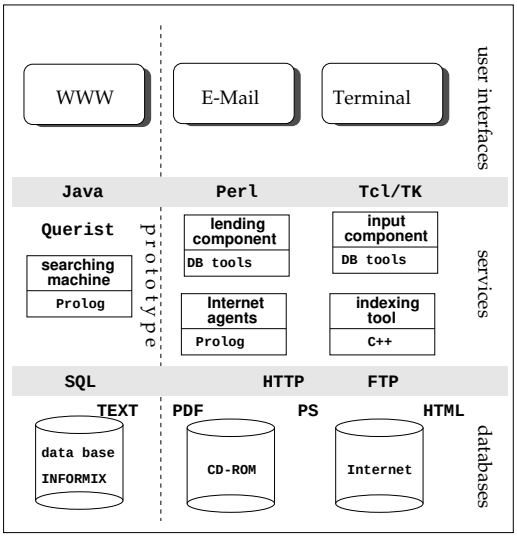

figure 2: CSLIB 2000 system overview

minal is not really integrated in the library.

The users appreciate their library because of its good equipment and the pleasant atmosphere

The Library of Computer Science

I will not write much words about our department library. Every student of our department should know it.

We want to take a look at the computer support in our library. Since 1984 there has existed a system which is called IFB-BIBAS. It is subderived into the three systems BIBAS (in German **Bi**bliotheks- **A**usleih- **S**ystem, lending component), TFAKYR (historic name for **Fa**chbereich **Ky**bernetik **R**etrievalsystem, searching machine) and BIBEL (in German Bibliotheks- Literatur- Erfassungssystem, input component). You can access this systems with e-mail, terminals and terminal emulations (world wide web).

The system IFB-BIBAS has some important disadvantages:

- There is no good WWW interface.
- The data quality is quiet bad. Every document is described with only six attributes. Furthermore you have a lot of abbreviations in the data.
- The system runs on an old IBM mainframe. The library has not any support contracts with IBM. So a system crash can bring us back to the index card times.

The disadvantages are well known. The people who take part in the CSLIB library workshops have two opportunities in the next years. On the one hand they can decide to buy a new system, on the other hand they can decide to develop an own system. The assistant of Knowledge Based Systems Dr. Ulrike Reiner supports the second alternative and has founded the CSLIB 2000 project [3].

CSLIB 2000 system overview

One of the challenges of the project is the development of a module and interface concept. Because nobody has already made a paper about this I tried to do it. The members of the CSLIB project modelled three layers (database, service and user interface layer).

In the database layer you find the data sources and some database management functions. The data can be stored in relational databases like Informix, in CD-ROMs or in different Internet databases. The documents may be stored in various document formats, like Postscript, HTML and Adobe PDF.

The service layer consists of some modules which provide the specific functionality of a library. The most important modules are the input component, the searching machine and the lending component.

In the user interface layer we find the modules which provide the look&feel of the library system for the users and the librarians. Terminal and e-mail interfaces have already been part of the IFB-BIBAS system. The terminal interface will be replaced by a local graphical user interface (Windows 95, Motif etc.) and by a WWW interface. The WWW interface should support HTML forms or special Java applets for the input. The output will be formatted with HTML.

If we build a prototype we will have to manage the interfaces and interactions between the various programming languages (Prolog, C++ etc.), transport protocols (HTTP, FTP, etc.) and document formats (PS, HTML etc.). The discussion about interfaces and interactions have already taken a lot of time. Because this discussion is still not finished I avoided to draw the relations between the modules in figure 2. If we combine the modules in a vertical way we get a number of 3x5x3 = 45 possible subsystems of the CSLIB 2000 system. From these possible subsystems we have to choose some useful subsystems. As a first aim the CSLIB project members decided to build a prototype which consists of a WWW interface, a searching machine and a relational Informix database.

The data model

At the beginning of the project several groups dealed with the development of a data model for our system. In comparison with the old six attributes data model we developed a nearly complete data model for documents. It contains around 30 tables with around 100 attributes [3].

One of the best ideas was the partition in abstract and concrete documents. Abstract documents are publications (books, journals, articles, reviews etc.). We can store abstract documents independently of the fact weather they are present in the library or not. Concrete documents are the present subset of the abstract documents. They contain a reference to the abstract documents and dependently on the origin a signature and lending information, an ordering address with a price and so on.

figure 3: Sample O'REILLY

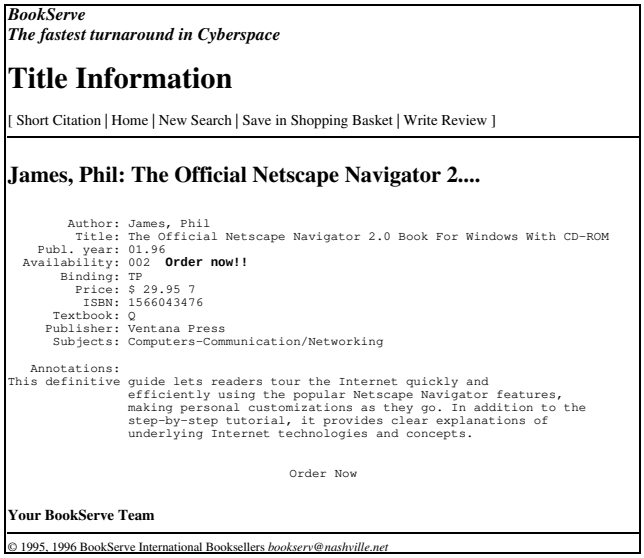

figure 4: Sample Bookserve

Data import

After we finished the data model we made a test. Everyone should input one of his favorite books. We had to fill a form with formal attributes like author, title, publishing house, abstract, literature references etc. The result was that we got a medium time of one hour input time for one book. Naturally this value is unacceptable.

After this test we had to think about solutions for this »input bottleneck«. The first solution was, that we expand the data model with a special subtype of documents which is called FAKYR document. The subtype FAKYR document includes the six attributes of the old TFAKYR system. So it was possible to import the data of present documents from the TFAKYR system.

The second solution is more interesting. Why the librarians should waste their time putting in new documents? The majority of new documents is saved in an electronic form. First of all we want to concentrate on getting the formal attributes of a document like author, title etc. The idea is that we have to look for data sources which contain the desired document descriptions.

In a first step we can use this as a help for putting in new documents. For instance I imported the formal data of the book »Being digital« by Nicholas Negroponte with the help of the cut&paste function from Negropontes web homepage.

Second we wish that everything goes automatically. We find the book descriptions in catalogues of the publishing houses and in the catalogues of the booksellers. Many of them offer their catalogues on their web servers. Beside the advertising effect many servers allows it to order books and journals on-line. Often you will find detailed book descriptions on the web pages.

Unfortunately each of the web servers has its own structure. Sometimes every description has its own structure. In the sense of advertising this may be desirable, but it hinders the automatic processing of this pages.

In figure 3 you see the advertising page of an O'REILLY book. In addition to the formal attributes you have a short abstract and a link to a full description. These are exactly the kinds of data we need to fill our database with. But it will be difficult to process this page by a program. The program has to search for the needed formal attributes. This will be difficult, because the attributes are not clearly separated.

In the second sample (figure 4) it is a bit easier to identify the attributes. A program can identify the attributes by reading the first column.

Internet Agents

I suggest to implement the automatic data import with Internet agents. Every agent has the aim to find as many as possible book descriptions. Every agent knows the document structure of a special server. The agents have basic methods to load and parse web pages and to bring back their results. Like human agents they should work silently and they have to observe permanently their objects. If the results of one agent are bad it will have to be coached.

In the true sense the work of Internet agents is a misuse of data. So webmasters can protect their servers with a simple method which is described in the »Standard for the Robot Exclusion«[4].

In the WWW you will also find complete works. A lot of journals publish a subset of their articles. There is also freeware literature in the net. For instance in figure 5 you can see the table of contents page of Shake-

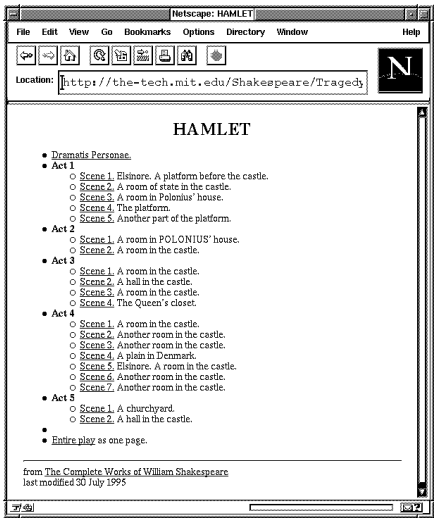

figure 5: Sample Hamlet

speare's Hamlet. The complete works of Shakespeare are available in the net. If you do not want to read the texts on a display you may print the text completely. The problem of distributing works with copyrights will be mentioned in section »Copy protection and payment«.

The searching machine

The searching machine is a component which allows the user to search for document descriptions or documents. The future searching machine of the CSLIB 2000 system will have much in common with the Internet searching machines like Alta Vista, Webcrawler, Yahoo, Lycos etc. There are two common ways to search for documents.

Libraries traditionally work with a classification scheme. The scheme is hierarchical. The books in the bookshelves and the index cards in the systematic catalogue are sorted with this classification scheme. Some popular Internet searching machines like Yahoo are also primary working with a classification schema (figure 6). One disadvantage of this machines is that the provider had to classify new documents by hand.

One member of our project supposed that we should design a searching machine with virtual three dimensional library rooms. The virtual library has bookshelves, assistants and a reading room. In the reading room you will also find other people who use this system. With your space mouse you can walk through the rooms. If you find an interesting book you may open this by clicking on its cover.

The second way of searching for documents is catchword based. The idea is that the library user often knows some catchwords which describe the searched document. The catchwords may be the author's names, specialist terms, abbreviations etc. The majority of the searching machines work catchword based. One advantage of the catchword based machines is that they can include new documents automatically with their indexing component.

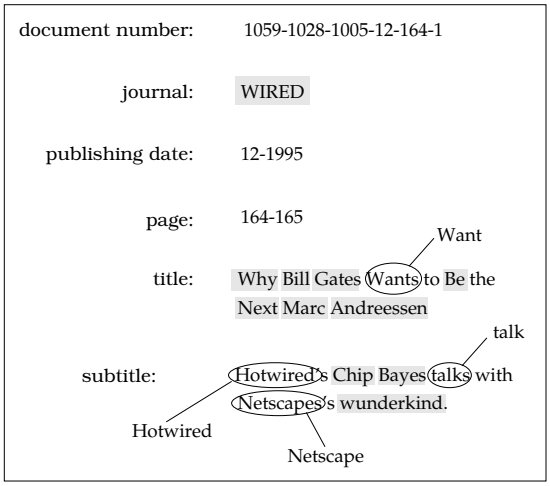

figure 9: Indexing of an article

In figure 7 you see a catchword based searching machine of the Library of Congress. I want to search for a Modula-2 book by the author Niklaus Wirth. I put in the two catchwords »Modula« and »Wirth«. In addition I have to specify the type of this catchwords (title, name). Some seconds after submitting the filled form I get the result in figure 8.

The indexing component

The indexing component is an important tool for the searching machine. It allows you to search with catchwords.

The indexing program has to read all document descriptions.

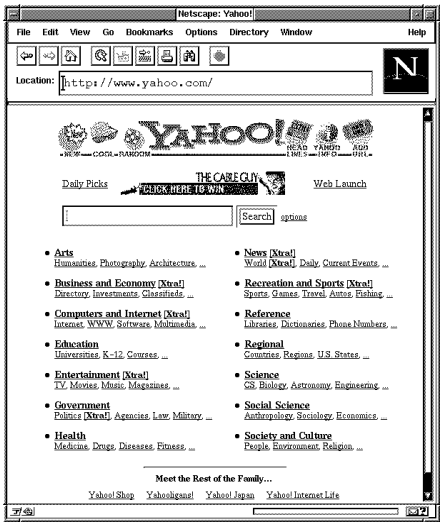

figure 6: Yahoo

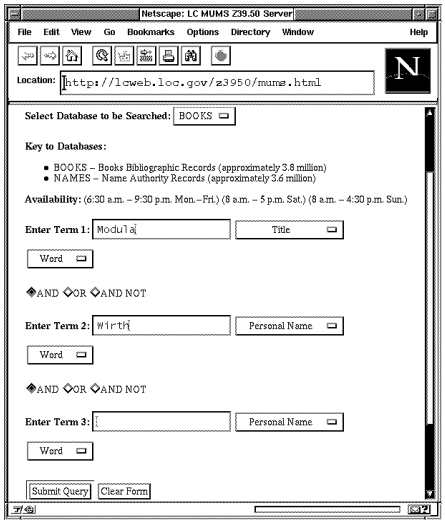

figure 7: Library searching machine

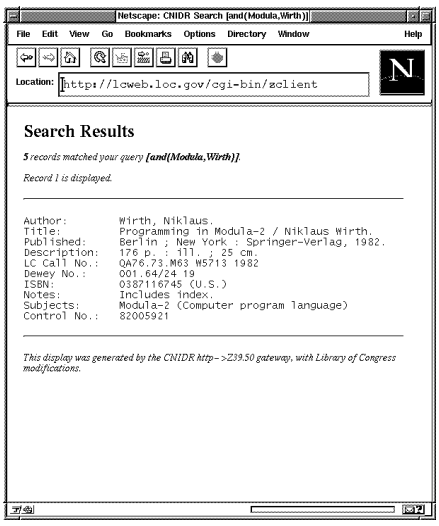

figure 8: Search result

While reading it stores the most significant catchwords with their source in a alphabetical sorted catchword memory. You can imagine this catchword memory as an index of a book. But the catchword memory contains the catchwords of a set of books.

With a short example (figure 9) I want to explain the indexing procedure.

First of all the program recognizes the document number. Then it reads the formal attributes of the article. Within the attribute values it has to decide word by word weather it is relevant or not.

Words which are probably irrelevant are stored in a stop word list. This list contains word groups like pronouns (my, your, …), articles (the, this, …), prepositions (in, by, …) and fill words (Oh, yeah, …).

The words with a gray background are supposed to be relevant. The words which are marked with an ellipsis are not in their normal form. For instance the words »Wants« and »talks« are in third person singular. Good indexing programs check this. They look up this words in a table which contains normal forms for a set of words.

A special problem of the word by word indexing is that some words have dependent of their context several meanings. The word »Netscape« can stand for a software product line (»Netscape Navigator«, »Netscape Communications Server«, …) or for the software house (»Netscape Communications Corp.« or its former name »MCOM Mosaic Communications Corp.«).

Summarizing I want to point out that a lot of detail problems of indexing are still not sufficiently solved. On the homepage of my group DELTA [2] you find some texts about indexing and information retrieval.

Copy protection and payment

Up to this point I ignored the fact that a library user does not only want to search documents but he also wants to read the founded books or he wants to make copies of parts of them. The library user comes in trouble if he notices that the book which he has found in a catalogue or in a searching machine is not available. Sometimes a librarian could help but in most cases he has only two opportunities. Weather he goes to another library or he buys the book in a book store. Sometimes the book is also out of print. I hope that computer scientists may help to improve this unsatisfactory situation.

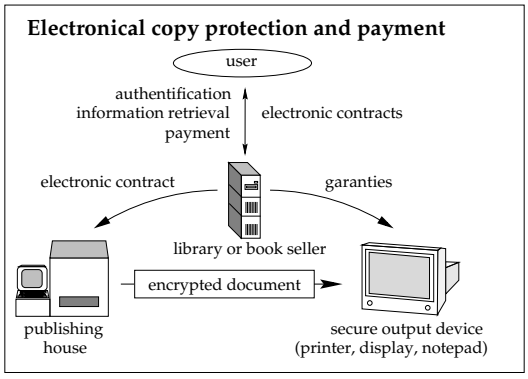

figure 10: Copy protection and payment

Remember that the majority of publications are stored in the digital archives of the publishing houses. A lot of publishing houses and libraries have an Internet access. So I come to the conclusion that we have the infrastructure and the software to transfer documents between the houses. But the publishing houses can not make money with this transfers and we can not protect the data against misuse. One type of misuse is that the library user makes some copies of a document in the library and sells them to others.

Nobody goes in public libraries to make copies of complete books which are available in bookstores, because the copies are often more expensive then the book and the quality is not comparable. But digital copies or prints from digital copies are much cheaper then the books or free of charge. Also the quality remains equal.

The publishing houses get the same problems like the software houses with illegal software copies. My idea is that we have to protect the digital documents against misuse and we have to make prints from digital copies as expensive as paper copies (around 10 Pfennigs a page). In figure 10 you see the structure of a system which realizes this demands.

The payment for the prints will be managed between the library or the book seller and the user (for instance with the possible future standard for electronic payment SET [5]). In this case it is nearly the same weather the object in the center is a library or a book seller. But it should be an authority, because it is responsible for its users and secure output devices.

So we get a liability chain from the publishing house over the library or book seller and its devices to the user. In technical view we can realize the liability chain with electronic contracts and garanties which may be implemented with cryptographic methods.

The encrypted document will be decrypted on the last possible point within the output device. Therefore we need browsers without save-functions, printers with decryption algorithms etc.

Before we have such secure devices on the market the librarians have to look for their users.

Some of the improvements and problems of a future library system will be shown in the following short story.

Short story

Please imagine that we are living in the year 2002. The Library of Computer Science is still in the 5th floor of our Franklin House. Beside the

bookshelves there are some terminals. The system which runs on this terminal is called CES. That stands for **C**omputer Science Library 2000 **E**xpert **S**ystem. Students from the CSLIB 2000 project developed this system. After patching some bugs the system runs nearly autonomous. Frank is a German student of computer science in the second term. Because he didn't found a book about Windows 2000 he asks CES. The dialogue between Frank and CES is in ordinary English.

CES> Hey boss, please insert your CSLIB card.

Frank does this and the system answers.

CES> Hello Frank. Since some month you haven't lent any books from us. But how about your girl friend Nathalie. Since three weeks I have written her reminders but she hasn't answered yet. She has to bring the book »Computer Poems« back.

Frank> Hello CES. How do you know that Nathalie is my new girl friend?

CES> Earlier she consulted me very often. She knows that I'm a good listener and that I have the right books for almost every situation. Lastly Nathalie told me about you. Congratulations! Unfortunately my developers from the CSLIB 2000 project still didn't implement my vision component. So I can't recognize the photo on her web homepage.

Frank> Don't worry! There is only a comic picture on her homepage. But let us talk about books. I need a good introduction to MS Windows 2000.

CES> Should I explain to you twice that we can't buy such books since the economy measures in 1996?

Frank> I know. But are you really incapable of helping me?

CES> Of course I can help you, Franky boy. Last night I talked with my friend which works as an document server for Rainbow Press Inc. Rainbow Press offers a new book from Ted Douglas with the title »Windows 2000 for Beginners«. Do you want to take advantage of this offer?

Frank> Yes, please.

CES> Rainbow Press is connected. Please take a look at the advertising and at the table of contents. There is a special offer from Rainbow Press. As an CSLIB user you can browse the full text of Ted Douglas' book for free on my display. The browsing time is limited to 20 minutes. There are no advertising interruptions. After that you can print out marked pages for 10 Pfennigs a page. Do you want to take this?

Frank> OK.

He reads some pages of the book and than he comes to the conclusion:

Frank> I believe that isn't the right book for me. I have to go, because my lesson begins. Please write me an e-mail if you find something better. Good bye.

CES> Of course. Don't forget your CSLIB card and please send my kind regards to Nathalie. Good bye.

References

[1] David Pescovitz, »The future of Libraries«, in WIRED, december 1995, page 68.

[2] André Bernard, Björn Voigt, Thomas Buhrmeister, Torsten Schäfer, Jurij Kostasenko: »Gruppe DELTA« (texts about information retrieval and indexing in German), http://www.cs.tu-berlin.de/~ul/csliblv/www-neu/gruppe4/gruppe4.html, 1995/96.

[3] »CSLIB 2000« (homepage of the CSLIB 2000 project group), http://www.cs.tu-berlin.de/~ul/csliblv/www/.

[4] Standard for the Robot Exclusion, http://info.webcrawler.com/mak/projects/robots/robots.html

[5] »Secure Electronic Transactions« (specification), http://www.mastercard.com/set/set.htm