Using an SGML Workbench for Digital Libraries

Jacques DUCLOY
CRIN - CNRS & INRIA Lorraine

1.0 Introduction

Traditionally, the entire information community was organized around a kernel of industrial companies where publishers played a very strategic role. This position was strengthened by two physical steps which had to be solved: getting a printed paper from a script and sending as many books as possible to all the potential readers throughout the World. Now, everybody agrees that anyone could write a «ready to print» novel, using a personal computer, and put it on a local Web server. When this paper contains what is needed to become a best seller it could be read by a very large public in a matter of a few days. This way of proceeding is growing in the academic world, and, as a result, the libraries of the laboratories are moving from importing the knowledge into a local team of searchers to exporting their know-how. Therefore, academic libraries need to master the whole bibliographic process.

At the same time, the costs of hardware and software are decreasing, with the consequence that any research team is now able to get the configuration which, fifteen years ago, was needed to put a large data-base such as MedLine on line. Thus, in financial terms, the handling of some large data bases or Digital Libraries has now become available to all academic organizations.

The same kind of evolution can be observed regarding many software aspects. Twenty years ago, one to three man-months were required to write any elementary program for a librarian in an assembly language. Now, this time is reduced to a few minutes if we use some up-to-date tools, especially those which come from the SGML world. When this working time becomes so much shorter, we enter into a quite interactive process for which very close co-operation between computer scientists and librarians is needed. Here we must recall that SGML holds a great advantage: its ability to be read by anybody, thereby becoming a common working language for several specialists coming from different areas.

All these reasons have driven CRIN, INRIA and INIST to investigate the use of SGML as a software engineering tool for Digital Libraries and to produce DILIB, an SGML workbench for Digital Libraries.

2.0 SGML, a common language for all the bibliographic data

The strong position of SGML, an international standard for the mark-up of electronic texts, to unify the description of main textual materials is now well known. In a DL context, three main classes of information have to be coded: the content itself, its bibliographical description and all the internal data, such as inverted files.

Several DTD are now very well known for describing the classical types of electronic documents we could meet in a Digital Library. For instance, TEI (Text Encoding Initiative) [TEI 97] aims at encoding electronic texts for interchange. TEI was designed and is growing in a linguistic applications context, for coding all kinds of existing documents. In a similar way, the ISO 12083 is issued by the American Association of Publishers for coding scientific papers, and is more oriented towards a producing process. HTML, at least, is becoming a «best seller» for this purpose, even if it cannot be used to describe complex documents.

Regarding cataloguing, some experiments on the use of SGML in documentary fields were carried out from the very origin of this standard. For example, the EEC's FORMEX [EC85] project proposes a DTD that uses the information contained in a CCF [PGI 88] record. TEI and ISO 12083 can also be used to carry bibliographical records, but all these approaches need a complex conversion from the original format to the target SGML one. So, a complementary approach consists in defining a DTD which is very close to the original bibliographic format. Thus, a record coming from Medline such as:

```
1/6 - (C) MEDLINE
NR  : 76145276
TI  : Induced fusion of fungal protoplasts [...]
LA  : Eng
AU  : Anne J; Peberdy JF
LP  : ENGLAND; GREAT BRITAIN; EUROPE
SO  : J Gen Microbiol; 1976 Feb: 92(2): 413-7
DEA : Acremonium/Cytology; Drug effects; Aspergillus
PTA : JOURNAL ARTICLE
```

could be marked-up as:

```
<doc>
  <NR>76145276</NR>
  <TI>Induced fusion of fungal protoplasts [...]</TI>
  <LA>Eng</LA>
  <AU><e>Anne J</e>
    <e>Peberdy JF</e></AU>
  <LP>ENGLAND; GREAT BRITAIN; EUROPE</LP>
  <SO><e>J Gen Microbiol</e>
    <e>1976 Feb: 92(2): 413-7</e></SO>
  <DEA><e>Acremonium/Cytology</e>
    <e>Drug effects</e>
    <e>Aspergillus</e></DEA>
  <PTA>JOURNAL ARTICLE</PTA>
</doc>
```

Most documentary organizations have chosen ISO 2709 [ISO 81] based formats, such as CCF, UNIMARC, USMARC, for the exchange of data and as a basis for their internal information structure. Basic ISO 2709 is a two-level (field and subfield) structure, tagged with numerical or alphabetic digits, and it is relatively easy to define an equivalent SGML structure, using original tags. For example, the beginning of a CCF record such as:

```
001     0 0   157028
201     0 0   00@ALegislative study - ...
210     0 0   00@AEtudes Legislatives ... Agriculture@Lfre
```

could be marked as follows:

```
<record>
<f001>157028</f001>
<f201 dir="00", ind="00"><sA>Legislative study - ...
<f210 dir="00", ind="00"><sA>Etudes Legislatives ...
  Agriculture</sA><sL>fre</sL>
```

Once a general schema of adapting ISO 2709 to SGML has been worked out, only two converters must be written to convert any kind of, for instance, MARC format to its equivalent SGML document and vice-versa:

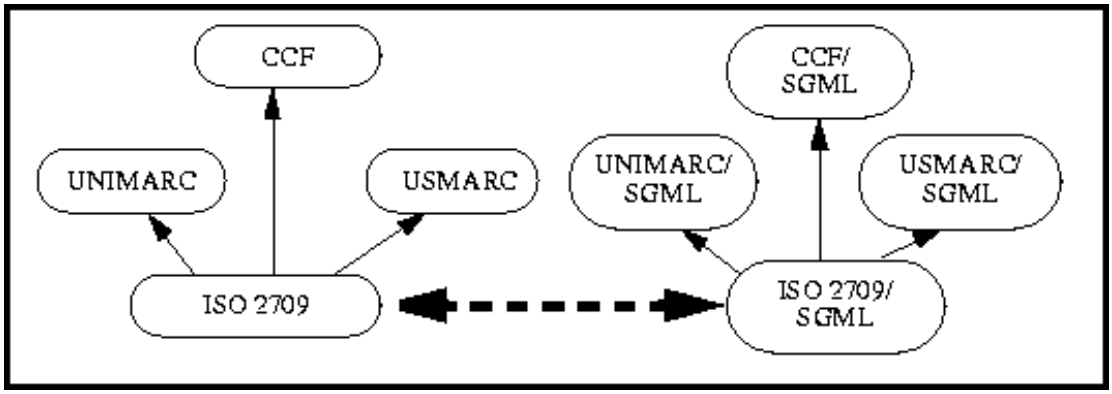

FIGURE 1.

We have used this approach for several years [DUS91], and several libraries' projects have begun to use it, such as the School of Library and Information Studies (UC Berkeley [LAR 96]).

We propose to extend this way of proceeding to all other data which are carried in a library platform, such as inverted files. Thus, all kinds of usual data could be implemented in such a way. A first and major result is the fact that any data may be easily interpreted by a human being. The second one is the ability of designing a set of homogeneous tools.

3.0 DILIB, a tool-box using SGML as a common interface

DILIB, (Document and Information LIBrary) is a workbench which is being developed by CRIN CNRS, INIST and INRIA Lorraine. It is composed of two

main kinds of tools: general purpose functions for handling SGML trees or records, and a set of coherent tools for building Information Retrieval Systems.

3.1 Generic tools for handling SGML data

A first set of tools is used to help in converting heterogeneous data into SGML. Two main levels (character set and data structure) are taken into account.

The character set recommended on the workbench is based on the SGML entity mechanism (and annex D of the ISO standard). Each character is represented by a symbolic chain delimited by an ampersand and a semicolon. For example, the French character «à» is represented by «`». DILIB provides commands to make this conversion. Concerning data structure level, in the previous section we gave some information on the conversion of ISO 2709 to SGML. We also provide some tools to convert other formats into SGML (generally this work can also be done by using public parsers like SGMLS).

Once data have been converted, they can be handled by various tools. A common problem consists in locating a particular element in a document. In order to solve this problem in a generic way, we use a path mechanism close to that of Unix. A path is composed by a string of tags separated by solidus characters. For example, let us consider the following record:

```
<doc><title>London by night</title>
    <author><f>John</f><l>Smith</l></author>
    <author><f>Paul</f><l>Templar</l></author>...</doc>
```

An example of a valid path is «doc/title» which gives access to the text element «London by night», another example is «doc/author/f» which points to the set of authors' first names: «John, Paul». With some extensions (introduction of conditions on contents), this mechanism is currently used to design parameters for DILIB Unix tools. For example, if we want to select records whose titles contain «London», we can use the following SgmlSelect command:

```
SgmlSelect -g doc/title#?London? < mySetOfRecords
```

All these commands are written in C, and we have also developed a set of C functions, whose kernel is an SGML tree handling library.

3.2 Clusterization and Information Retrieval tools

Another part of the workbench consists in an Information Retrieval System in kit form. We have chosen to define some basic access mechanisms very close to the Unix file system. On this basis, we have defined other items such as inverted files whose records are also coded into SGML form:

```
<index><kw>usmarc</kw><freq>223</freq>

  <list><e>000010</e><e>000035</e><e>00101</e>...</list></index
  >
```

Thus all the internal data such as inverted files can be handled by DILIB. This facility has been used to develop most of the infometric modules. For example, from an inverted file we can first build association files (always using a toolkit command) which look like this:

```
<assoc><ti><kw>usmarc</kw><f>223</f></ti>
       <tj><kw>SGML</kw><f>300</f></tj>
       <fij>15</fij><list><e>000035</e>...</list></assoc>
```

where «fij» is the frequency of co-occurrences of the key-words «usmarc» and «SGML». Then this association file can be used to build a cluster file which is also a set of SGML documents, one per cluster. Each cluster is mainly made up of a group of words with their inter-relations. The construction of clusters is carried through thanks to a classification based on a file of associations describing the relations existing between the keywords symbolising the document of the database.

The method of clusterization carried out thanks to DILIB is that of «single link». The principle of the method is the following: the algorithm of clusterization works through consulting the file of associations between words in a decreasing order of significance. The clusters are thus built in a successive way which works by putting into chains associations having common descriptive elements. The size of each cluster is limited to the group with the most significant associations which were used to build it (the cluster). The group of remaining associations is used in an additional way to create links between the various clusters.

4.0 Using an SGML workbench in a Digital Library context

4.1 A modular approach for the Digital Library architecture.

The extreme difficulty in the design of Digital Libraries derives from several factors:

- the high variety of the computer oriented problems we have to solve (information retrieval, linguistic processing, data base management, publishing process)
- the large heterogeneity of the document structures which come from many organizations,
- the high level of distribution of the data and associated processes.

The traditional way for solving these difficulties was based on DBMS technology. Now, a main scientific and technical result of the WWW is the feasibility of a Word Wide Application in which the coherency comes only from a common use of standards. HTTP, the «protocol component» of the WWW architecture, deals only with the transportation and not with their contents. HTML gives just a general framework to describe these contents. We propose to adopt the same kind of philosophy to design the Digital Library applications, and we will consider three levels of data communication: organizations, software packages and components.

Librarians are still using this approach for exchanging data between several organizations (figure 2). But the specifications of the ISO 2709 format were oriented mainly towards computer people. We have stated previously that using SGML at this level is useful. It is not strictly required and sometimes unusable, depending on a general agreement among the partners of a given network.

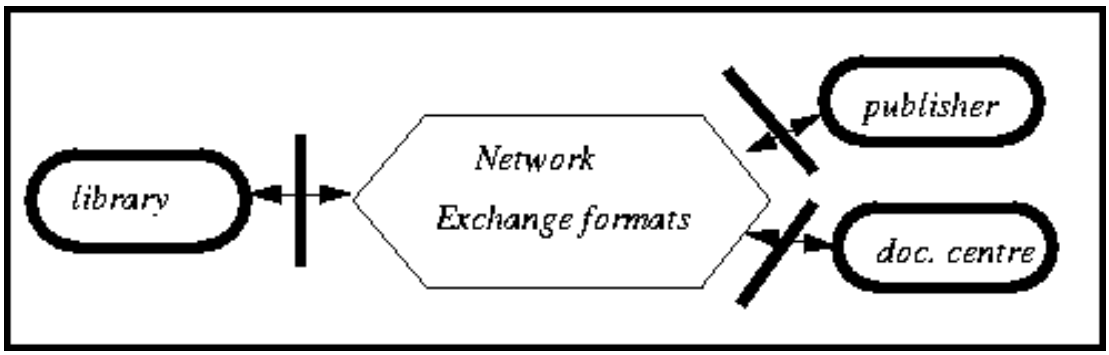

FIGURE 2. Using SGML for data exchange between organizations.

Now, if we look at the internal organization of a member of the previous network, for instance, a library or a documentary centre, we can apply the same philosophy of using a format to provide the communication between several packages of a complex application. But now, the choice of a good communication format only depends on a local decision, and SGML can be seen as a very generic basis. It solves the problem of interfacing between very different tools like a DBMS or a publishing system.

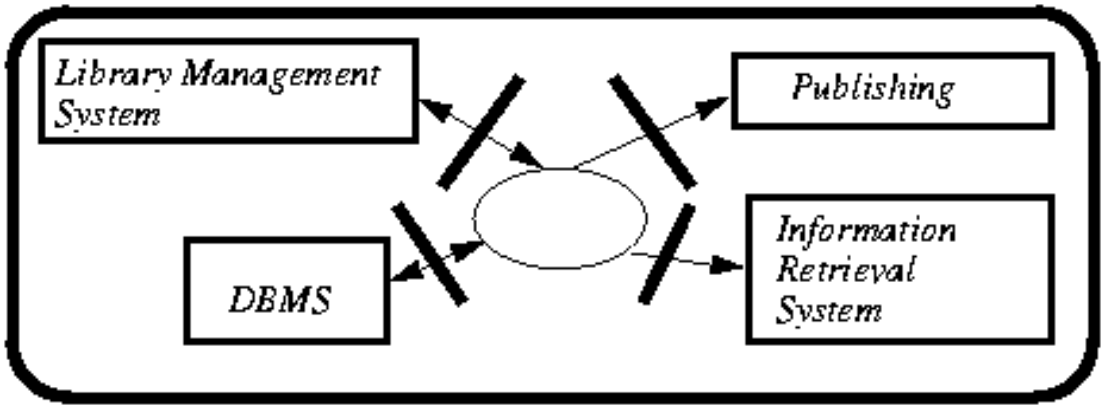

FIGURE 3. Using SGML for the communication between the main packages of a given organization.

At least, we could make the same choice for the design of the interfaces between the various components of a software package such as an Information Retrieval System (figure 4). We presented earlier some principles about that.

Until now, our experiments tend to confirm the advantages of this solution. From a purely hardware or performance point of view, we lost some disk resources and

some elapse time. But the consequences were minor in most of our applications and they were more than offset by the consistent benefits in productivity. In some very specific cases, we had to adopt a different strategy, proceeding in two steps: using a rigorous SGML architecture at the prototyping level, then a more preforming choice at the execution level.

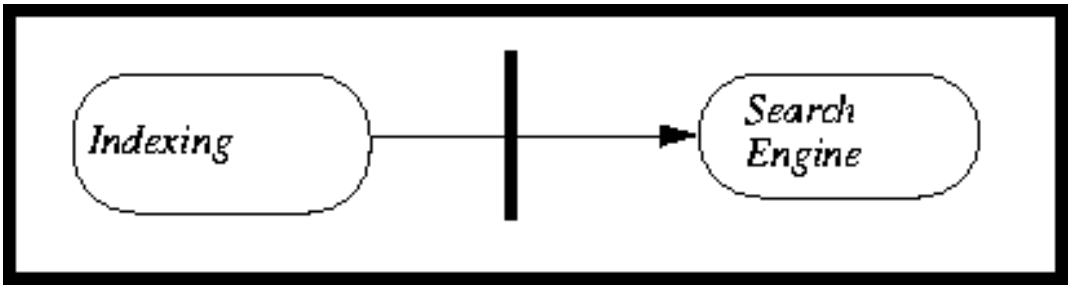

FIGURE 4. Using SGML for interfacing the modules of a software package

4.2 An experiment using DILIB with a DIENST architecture

At an ERCIM meeting on Digital Libraries we presented a prototype of a DIENST [DAV 94] server using a DILIB based interface.

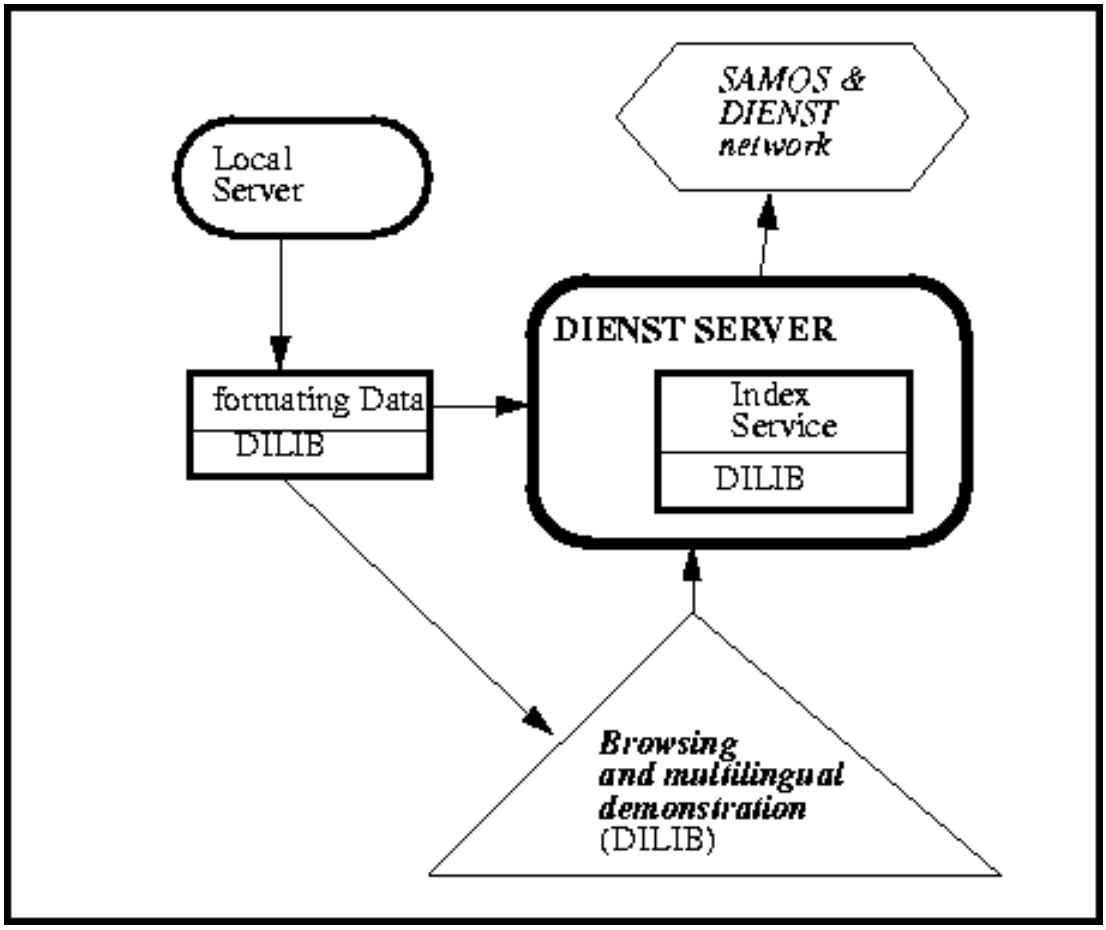

FIGURE 5. The Budapest prototype

In this experiment Dilib was used two times: first to develop the reformatting program from BibTeX to RFC 1801, the bibliographic format of DIENST; twice to design and implement a new version of a DIENST «user interface» server, in which a multilingual browsing graph was automatically generated from the collection of documents using a clustering mechanism.

This experiment was developed in only a few weeks, demonstrating also the good design of the DIENST architecture in which each main function is implemented by one specific server.

5.0 Conclusion

SGML is mainly known as a universal exchange format. Its young, but successful, story shows that it has some very advantageous qualities to have become so popular. We think that one of these deals with its fitness to be read or written by man and directly used by a machine. Hence, it becomes a very interesting tool for designing tool interfaces.

So, its use for building some Information Retrieval Systems is to be considered as «very natural». Perhaps, its readability by librarians or information science specialists is a more consistent advantage. With the Dilib project, we have tried to use it in designing a new way of investigating among sets of data and documents.

6.0 References

[DAV 94] James R. Davis and Carl Lagoze, «A protocol and server for a distributed digital technical report library», Technical report - TR94-1418 Cornell University. http://cs-tr.cs.cornell.edu/Dienst/UI/2.0/Describe/ncstrl.cornell%2fTR94-1418

[DUS91] Dusoulier, N., Ducloy, J. (1991): «Processing of data and exchange of records in a scientific and technical information center. Formats : what for?» UNIMARC/CCF Workshop - Florence (IT) (IFLA/UNESCO), 05-07 June 1991.

[EC85] EC - *FORMEX - Formalized Exchange of Electronic Publications. Office for Official Publication in the European Communities* - Luxembourg, 1985 (ISBN 95-825-5399-X).

[ISO 86] ISO 8879 - Standard Generalized Markup Language (SGML) - 1986.

[LAR 96]Ray Larson, and Jerome McDonough USMARC DTD and utilities to convert USMARC records to SGML formatted records http://www.ua.ac.be/WGLIB/MARCSGML/sgml.html

[PGI 88] UNESCO - PGI & UNISIST «CCF: The Common Communication Format - Second Edition», Paris 1988 (PGI-88/WS/2).

[TEI 97] *the TEI Home Page*. http://www.uic.edu/orgs/tei/