



Florida Cooperative Extension Service

Circular 1087
November 1992

SGML Implementation Procedures¹

Dennis G. Watson²

INTRODUCTION

Implementing SGML (Standard Generalized Markup Language) in document processing is a major task for most institutions. It often involves altering the procedures and methods people have previously used in producing documents. Before jumping into SGML implementation, it is important to preview the steps involved. A prior understanding of basic SGML terminology is required for adequate comprehension of this document. SGML implementation can be divided into nine steps:

1. Recognize application
2. Establish goals
3. Form working group
4. Analyze documents
5. Write document type definition (DTD) and SGML declaration
6. Validate DTD
7. Verify DTD
8. Develop tools for users
9. Document, train, evaluate, and maintain

RECOGNIZE APPLICATION

Before expending effort to implement SGML, one must first determine if the application is indeed appropriate for SGML. Typical SGML applications have multiple output formats, output devices, or input sources. Example output formats are printed publications, electronic on-line databases, and multimedia CD-ROMs. Other requirements which indicate probable SGML applications are:

- longevity of document format beyond the current generation of computer hardware and software,
- document interchange with other institutions, and
- multi-purpose or multi-product databases.

ESTABLISH GOALS

Setting goals is an important step in SGML implementation. SGML is seldom implemented only for the sake of conforming to a standard. SGML is usually implemented because it is a solution to a problem. Goals typically fall into one or more of three categories: interchanging data, publishing data, and managing data. Goals are used to direct the SGML implementation and should be referred to often. SGML implementation is ultimately a management decision and goals should be stated in terms that address management concerns. SGML cuts across the different objectives of authors, editors, and production staff. The goals should be developed from a holistic viewpoint. Goal setting should include short and long-term goals. An example goal of an SGML implementation might be to develop a single input format for documents that can be automatically processed for multiple output formats (e.g. printed and CD-ROM). The goal may be qualified by specifying a certain class of documents (e.g. fact sheets). Throughout an SGML implementation project focus must remain on the overall goals. Flexibility should be maintained without sacrificing the goals.

1. This document is Circular 1087, Florida Cooperative Extension Service, Institute of Food and Agricultural Sciences, University of Florida. For more information, contact your county Cooperative Extension Service office. Publication date: November 1992.
2. Dennis G. Watson, associate professor, Agricultural Engineering Department, Cooperative Extension Service, Institute of Food and Agricultural Sciences, University of Florida, Gainesville FL 32611.

The Institute of Food and Agricultural Sciences is an equal opportunity/affirmative action employer authorized to provide research, educational information and other services only to individuals and institutions that function without regard to race, color, sex, age, handicap, or national origin. For information on obtaining other extension publications, contact your county Cooperative Extension Service office.

Florida Cooperative Extension Service / Institute of Food and Agricultural Sciences / University of Florida / John T. Woeste, Dean

FORM WORKING GROUP

Forming a working group is a critical step in implementing SGML. This step normally requires some prior work in selling a solution and gaining administrative support. A working group's effort will include document analysis, writing a document type definition, and reviewing the SGML implementation. Thus, a working group should represent the interests of those who will be affected by the SGML solution. This may include authors, editors, production workers, database developers, and distribution personnel. A typical working group would consist of users from authoring editorial, production, data processing, training, and management departments. Last, but not least, a working group should include an SGML specialist. In some cases, the SGML specialist is a consultant.

A successful working group often must be successful politically. The group should include political members as well as the appropriate technical members, who will be the users. Political members are those who have the power to make a decision and a demonstrated willingness to make a decision. A political member should also be able to promote compromise.

The working group leader should have a good working knowledge of the SGML standard and ability to apply SGML to problems. The leader should be strong, yet flexible. A working group leader must be prepared to handle strong egos, opposing viewpoints, and possibly even hostility. The leader should have strong political support from administration.

Group selection is very important and should be stacked in favor of the desired outcome. Choose users with a positive attitude toward the goals and a willingness to change the status quo in favor of a better system. Users with power to make and implement decisions are extremely valuable. All user groups should be represented on the working group. A working group could consist of five to ten members.

ANALYZE DOCUMENTS

Document analysis is a critical step in implementing SGML. Document analysis consists of:

1. Identifying a set of documents of the type to be implemented with SGML.
2. Analyzing the documents.

3. Identifying the data elements within the documents.
4. Naming the data elements.
5. Identifying the hierarchical structure of elements.
6. Determining if elements are optional or repeatable.

Carefully select a representative sample of the documents to be implemented with SGML. Analyzing the documents and identifying the data elements often requires a reiterative process of "tearing" a document into two or three pieces. For example the first iteration may "tear" a short document into three pieces: front matter, body, and back matter. The next iteration includes "tearing" each of these pieces into two or more pieces. This process continues until the document is divided into as many pieces as appropriate to meet the requirements of the SGML implementation goals. Each piece is called a data element (or just element) and must be named. The process is described further, with examples, in a related publication.

WRITE DOCUMENT TYPE DEFINITION (DTD) AND SGML DECLARATION

Once the document analysis is complete, the document type definition (DTD) can be written. The DTD should be written by someone very familiar with SGML. Depending on the detail included in the document analysis step, there is usually more than one way to write a DTD. A DTD is analogous to a computer program. Two computer programmers given the same specifications could write different programs to do the same task. In DTD writing, there is often more than one "right" solution, although one solution may be better than another to meet the goals of the SGML implementation. The SGML standard specifies a language and syntax for writing a DTD.

Writing the SGML declaration is a relatively simple step when compared to the other steps. The SGML standard defines a default declaration. In many cases, the default is used. Sometimes changes are made to accommodate longer names for elements. A person very familiar with the SGML standard should write the SGML declaration.

VALIDATE DTD

DTD validation is the process which determines whether the DTD conforms to the SGML standard. A software program, called a validating SGML parser, is used for this step. The program reads the DTD in

ASCII format and displays any errors it finds in testing the DTD for conformance with the SGML standard. This step does not determine whether the DTD is accurate for the documents for which it is intended. The process is similar to compiling a computer program. A compiler requires that the source code be syntactically correct, but not that it will solve the problem. Thus a validated DTD does not insure a solution.

VERIFY DTD

DTD verification is the process which determines whether a DTD is indeed the desired solution to the problem. Verification is a type of evaluation. The DTD is used to add tags or markup documents (of the type for which it was intended) which are different from the ones used in the document analysis phase. During the verification process, users note elements which were not included in the DTD and other information pertinent to the DTD evaluation. DTD verification can easily become an ongoing reiterative process. It is a rare DTD that never needs changing from the day it was first written. As changes are made the verification needs to be repeated. A formal verification process should be established to elicit feedback from users.

DEVELOP TOOLS FOR USERS

If implementing SGML were as simple as having a DTD, many more institutions would be using SGML. Viewing SGML from a holistic perspective includes analyzing how the DTD will be used by authors, editors, production staff, and other users. Based on the analyses, appropriate software tools and procedures are developed. For example, an SGML application typically requires authors to insert SGML tags into a document or learn to use a new authoring program which inserts the appropriate tags. For many authors, switching from their "tried and true" word processor can be traumatic. For these reasons, many SGML applications require creative solutions to ease the trauma of implementation at the level of the authors. If possible, simultaneous implementation of a feature desired by authors can encourage author acceptance. The production staff requires a different

set of tools for their purposes. Software is needed to convert from SGML to the desired output format. In many cases, commercial software is available to assist in this process.

Without careful forethought, an SGML application can result in failure due to frustration or unacceptance on the part of the users. From the very beginning, software tools and procedures must be considered that will make the transition easier for each group of users.

DOCUMENT, TRAIN, EVALUATE, AND MAINTAIN

Once a DTD has been verified, it is ready for widespread use. Documentation on the SGML implementation process should be completed. This documentation includes the goals of the implementation, results of the document analysis, and results of the DTD verification. Another set of documentations is required for users. A user's guide describes step-by-step how to perform the necessary steps to use the SGML implementation. Authors need to know how to insert tags in their documents. The production staff needs to know how to generate printed output or document databases from the SGML documents.

Training sessions should be conducted to familiarize users with the processes and software tools. The sessions should include hands-on experience.

Evaluation should include feedback from all user groups. The author feedback will be valuable in modifying the software tools for authors. If the authors aren't using SGML, then the entire implementation will fail. Likewise the production staff must be able to make adequate use of the SGML documents.

Maintenance is an important stage for long-term success of the SGML implementation. The DTD may need revision from time-to-time and any software purchased or developed for users will certainly be updated. Plans should be made for the maintenance phase.