



Hope for the SGML Savant Is Near & Far Away

a well thought out commitment to the Standard Generalized Markup Language (SGML) has its benefits, and there's a new genre of SGML publishing tools around. In fact, there are new categories of tools that can do quite nicely in an integrated SGML publishing system that takes into account the publishing life cycle from authoring through distribution and reuse. SGML finds its application right at the start of any publishing life cycle, even before the writing starts, where new tools help analyze document requirements and create a working document content model or SGML document type definition (DTD).

Indeed, a commitment to SGML requires planning, analysis, and commitment, and part of that commitment is the analysis of your current documents and the building of one or more models that describe the documents. All the while, building such a model—called a document type definition—requires not only a clear understanding of the document set it expresses but also some programming skill to create the actual DTD. If you think this sounds like a ripe situation for having a good toolbox, you're right.

But until recently your choices of DTD construction or document modeling tools were few. The modeling process itself was like writing a program. After document analysis, you learned the language (SGML), its syntax, and semantics. Then you used a text editor to create a DTD

expressing your document model, ran the DTD through a parser to validate it, then repeated the cycle to fix problems or improve the model. Even a couple years ago, SGML professionals were without modern CASE-like development tools.

BETTER TOOLS NEARER TO THEE

Microstar Software Ltd., developer of Near & Far Designer, positions itself as a supplier of integrated SGML tools. Indeed, with its life cycle view of documents, Microstar offers tools similar to a traditional CASE (Computer-Aided Software Engineering) suite of software.

Microstar asserts that documents have a life cycle similar to other corporate assets, which can be developed with what

they call
Computer-Aided Document Engineering (CADE).

Microstar's suite of products work with Designer and include Near & Far Author (an SGML authoring tool integrated with Microsoft Corporation's Word) and Near & Far Library (a groupware document design and management system).

Corporate publishing and MIS organizations used to have nothing in common;

now relationships between the two are growing: MIS is becoming more and more aware of the value of documents as data, and has long been concerned with integration issues. One likely result of this changing environment is that MIS organizations, long familiar with CASE systems, will feel comfortable with Microstar's tools philosophy. And MIS organizations will certainly like the rigor in defining data elements through data dictionaries, and standardizing use of these elements throughout an enterprise.

Although Microstar considers Designer to be a document CASE-like tool, it does not conform to the CASE Data Interchange (CDI) format, an emerging standard for MIS CASE tools.

Near & Far does offer import/export capabilities with delimited ASCII and Lotus Notes databases, suggesting a capability to integrate document model information with corporate data dictionaries.

The data dictionary capabilities with Lotus Notes are provided through Near & Far's CADE Groupware product, which is a Notes application template. Import/export via ASCII will definitely require some customization work to get to or from corporate data dictionaries. Nonetheless, the fact that Microstar has even considered integration shows some ability to integrate with MIS data dictionary definitions. If your publishing group needs to garner support from your corporate MIS organization for SGML efforts, these capabilities may well do the trick.



AND THEN, THE OFF-THE-SHELF DTD PROBLEM

The federal government, many foreign governments, and many in the defense, aerospace, automobile, and financial industries have already standardized on SGML. Or perhaps you have developed an Internet Web presence and learned about HTML, a version of SGML that is growing in complexity and power. There are many reasons why you might find the benefits of SGML so alluring—rich re-purposing of document content, vendor independence, and interchangeability with other publishing systems or media—but the attraction has consequences. The SGML-enamored will find themselves wrestling with large data management and design issues.

What design issues? Here are some, to start:

- If you need more than one DTD, what is the relationship between them?
- Is it possible to modularize DTDs, sharing common sections and managing them independently?
- If you need to move document information between proprietary and standard DTDs, how do you guarantee you can do this?
- How do you develop and communicate DTDs among various interested parties?
- Are there any tools, similar to CASE tools, for DTDs?

And if standard DTDs exist, why not use them “out of the box”? Indeed, many standard or quasi-standard DTDs are available for use: the Department of Defense CALS initiative (Mil-M-28001), Air Transport Association ATA-100, American Association of Publishers (ISO 8879), Software Publishing Industry (DocBook), and Web Publishing (Hypertext Markup Language, now in version 3.0), to name only a few.

Unfortunately, no structured database data model is likely to be usable “as is” in a new data environment. There may well be useful parts of other DTDs you can use or learn from. Use a standard document model and you still may want some additional capabilities for internal use.

Readers wanting more information on Near & Far Designer may contact Microstar Software Ltd., 3775 Richmond Road, Nepean, Ontario, Canada K2H 5B7; 800/267-9975; 613/596-2233; Fax 613/596-5934; Internet—cade@microstar.com.

And that may be a strong advantage of tools like Near & Far. Microstar’s Ray Luoma claims to have customers using Near & Far in CADE groupware as an integrated toolset for developing document models. A team of designers uses CADE groupware to define elements and attributes of documents, argue about them, and record the results of their decisions. They then access the elements, build them into models, and save the models back to the CADE groupware database. Within Near & Far, they can cut and paste items to build new models, all the time ensuring the consistency of their document objects.

HOW CLOSE IS NEAR & FAR?

Near & Far presents a simple, well-designed user interface. The program runs remarkably quickly even on a low-end 386 PC with 4MB RAM, running Windows 3.1. If you are comfortable with standard Windows functions such as Save, Undo, and Redo, you’ll move quickly to the document model functions. The documentation is well designed, clearly written, and provides useful examples.

Near & Far considers document models to be hierarchically (tree) structured. Take a test drive with the company memo DTD they supply and you’ll understand and be able to modify the example DTD. You’ll see elements (such as the memo header, to and from block, and paragraphs), connections between elements, and occurrence indicators. Connections allowed include ordered lists, unordered lists, and selection (“one of...”). You can move or copy portions of the model (branches of the tree) within the model by simply dragging and dropping them.

Occurrence indicators are properties of objects specifying how many times an object or structure can or must occur at any particular point in a model. For example, you probably would specify that a memo DTD must have at least one “to” recipient, but may have several.

When you’re done with your model, you can save it in Near & Far’s native binary format, in a Lotus Notes database, or you can export it to an SGML DTD. You can also produce reports describing your model.

After you’ve developed one or more DTDs, you’ll want others to review but not change them. For review, use either the CADE Groupware product or Near & Far Lite, a DTD viewer available from Microstar Software, to explore the graphical, tree-like structure of your document models.

Near & Far Lite makes document models accessible to everyone in an organization, such as a document designer who needs to look at alternative models; an author who needs to understand what he or she is expected to write; a manager who needs to approve document structures; and an electronic publishing specialist who needs to design CD-ROM applications. The built-in screen capture utility lets you add pictures of document models to presentations, documents, and email. This capability lets you easily discuss and exchange ideas on your document model.

PROS AND CONS, HERE AND THERE

Near & Far is an innovative SGML development tool, but it does have some limitations. The biggest faults have to do with integration and proprietary formats.

Proprietary behavior in the tool is somewhat troublesome. You would expect an SGML tool to import and export an existing DTD and not rearrange anything—including white space and comments. Near & Far failed this test and also occasionally reported errors which other mature SGML tools did not. Lastly, newer SGML standards, such as HyTime for multimedia, are not simple self-contained tree structures. Thus the tree-structured model may not work in upcoming SGML multimedia applications.

Those criticisms notwithstanding, you should try out Near & Far in limited situations. If you want to learn SGML, Near & Far costs no more than some SGML training courses. You can use the Designer to develop real DTDs quickly and validate their design. And if you develop families of DTDs, with modular shared sets of tags, you may also find Near & Far helpful. If your organization has standardized on Lotus Notes, then you have additional incentive to consider Near & Far.

Is Near & Far Designer sufficient as a robust SGML development environment? No, but then any SGML toolbox should contain more than one tool. If your organization is getting started with SGML, Near & Far bears a closer look.

Robert J. Boerl is Advanced Systems Specialist in the Information Services Division of Factory Mutual Engineering in Norwood, Massachusetts. Martin Hensel is founder of Martin Hensel Corporation, a consulting firm that builds SGML-based editorial and production systems. Comments about this column may be addressed to CD-ROM Professional, 649 Massachusetts Avenue, Suite 4, Cambridge, MA 02139 or via email at davidg@onlineinc.com. ■