[*Language Industry Monitor* 14, March-April 1993]

Alan Melby on TEI-TIF

TIF, the terminology interchange format, is nearing completion within the wide-ranging Text Encoding Initiative. TIF proponent Alan Melby discusses some of the ramifications of encoding terminology this way within the larger document processing context.

“I am amazed at how quickly TIF has been adopted,”says Prof. Alan Melby of the Translation Research Group at Brigham Young University. “I presented it last year to the LISA members at two of their meetings. After a brief deliberation at their June meeting, they approved it. We had only been working with them for six months.” Melby has spent more than a decade in pursuit of terminological standards; it comes as a very welcome surprise to him that such a standard is now finally coalescing.

The Terminology Interchange Format (TIF - pronounce it tee-eye-eff) was developed by a working group within the Text Encoding Initiative (TEI), an international consortium of researchers from industry and academia. TEI is not, as its name might suggest, an effort to code a lot of text. Rather, the goal is to use SGML, essentially an abstract programming language, to build a general-purpose library for _text encoding,” the comprehensive labelling (or “tagging”) of the structure and content of texts. The resulting Document Type Definitions (DTDs) will be a vast SGML library of “data element types” (types of textual elements) which can be used to define specific data elements within a text. It will be a resource which anyone with SGML software can use. A TEI document can be anything from a poem to NLP lexicon, and within such a document you can specify everything from global attributes, such as document type, subject, and language, down to word and letter level phenomena. Think of TEI as the mother of all DTDs.

The TEI working group on terminological data, led by Melby, Gerard Budin, of the Austrian terminology organization Infoterm, and Kent State University‘s Sue Ellen Wright, have expended substantial effort over the past several years collating input from a variety of interested parties and they have molded a specification for terminology which has as broad an application as possible. Under TEI, a term tag can be as simple as <*term*> *dynamic time warping* </*term*> and embedded with a body of text. Or, it can be an autonomous DTD for interchange purposes and define records with a complex, nested structure, including equivalents in multiple languages. As a glimpse in the TIF Tutorial prepared by the group reveals, TIF identifies several dozen possible data element types for a term tag, with at least a hundred data elements (the actual tags) defined. Degree of synonymy, false cognate, reliability code - these and many more can all be specified within TIF. You can even have element types for defining the relationship between terms, i.e., hierarchical, sequential, or spatial. TIF is also becoming an internationally recognized standard on its own: it is now at the Committee Draft stage within the iso. When it is approved, it will be known as iso 12200.

SGML, and its embodiment in the TEI DTD, represents a crucial evolution in text-based information from a presentation-oriented approach, specifically its representation on a computer screen or on paper, to a content-oriented one. The TIF specification is a good example of the power of text encoding. Imagine, for example, having all the terms in a document tagged and linked to a central table with definitions, possibly even translations. Melby sees potentially a new paradigm for translation. “Much of the benefit derived from MT these days is consistency of terminology. But how much time do you think post-editors spend referring back to the source text?,” he asks. Probably a lot, seeing that translators are considered to be the best post-editors. “For certain kinds of materials, it may not be worth the effort of using MT to translate whole sentences at a time,” he suggests. Melby could envisage a scheme whereby the tagged terms of an SGML document are linked to an internal (or external) terminology glossary and automatically substituted upon demand, leaving the rest of the text untouched for the translator. This portends a new, multidimensional approach to documents; documents with multiple layers of embedded information. All we need now is the software.

TIF is therefore attractive not only on account of its intrinsic merits, but because it is also part of the vast TEI hierarchy. Melby realizes this was an important consideration for the LISA members, for example. Whether companies like those will ultimately go on to exploit the TEI DTDs in all of their diverse realms will depend on many factors. In any event, no one will be forced to have to keep reinventing the arduous DTD wheel. If a TEI DTD is not suitable as is, it can be modified for a specific application with much less effort than developing a DTD from scratch.

Melby also believes that companies are starting to think differently about terminology interchange. Traditionally reluctant to share their terminology databases, they are now beginning to realize that it is in their interest for their terminology to be accepted as a standard. "Take Microsoft for example. It has been actively circulating its GUI Guide with this goal in mind," points out Melby. The GUI Guide is Microsoft's multilingual termbase specifying the interface elements in Windows. As part of their evaluation process, LISA's terminology group converted the GUI Guide to TIF format. Melby also notes that the standards defined by such organizations as the American Society for Testing and Materials only become accepted if they are adequately circulated. _Issuing standards in electronic form is a great way to do this, but it will only be effective if they are in a broadly accepted format," he says.

SGML was designed to be resolutely application-independent, and as such, it is almost limitlessly fiexible. The downside of this uncompromising abstraction is that it offers very little guidance to people who are trying to implement working systems. Obviously this has hampered its acceptance in the real world. Together with SGML-based software (such as the forthcoming IntelliTag package from WordPerfect which Melby hopes will soon be tested with the TEI DTDs), the TEI DTD may prove to be a decisive factor in finally launching SGML into the mainstream of computing. "The next step," says Melby, "is to start demanding SGML-conformancy from all of our software vendors."

Brigham Young University's Translation Research Group assists developers incorporate TEI TIF interchange facilities within terminology management systems. It is also preparing a general TIF validation utility which will be circulated freely.

Translation Research Group, Linguistics Department, Brigham Young University, 2129 Jesse Knight Humanities Building, Provo, UT 84602, usa; Tel +1 801 378 4414, Fax +1 801 378 4649, Email trg@jkhbhrc.byu.edu

Termbase Design and TIF

When building a terminology database, should you take the relational database route or keep to the familiar structured free-text path? There are convincing arguments for both approaches.

The experts differ on the best strategy for designing termbases. Two models prevail: the relational database and structured free-text. For keyterm, cap debis opted to build it around a relational database engine, partly because of its mainframe and Unix background. Other pc-based packages have followed the structured free-text file format that originated with mtx, the widely used terminology package from LinguaTech. Most notable of these is TIF-compliant MultiTerm, which extended the mtx approach in a number of ways. The design debate is not an academic one; the ease with which users can set up termbase, enter data, and search for terms significantly effects the usefulness of a termbase. Translators, in general, are not interested in becoming database experts; sql is not usually a language in their arsenal.

Traditional relational databases have not always been well suited for handling terminological or lexical data. Requirements such as fixed length fields and fixed length keys can frustrate efforts to work with text-oriented data. Some still do not support alternate character sets - devastating for multilingual applications. Newer packages do support variable length fields and memo fields, but do not necessarily provide editing and search functions within fields for handling a large amount of text. On the other hand, relational databases do offer extensive control over data entry, and the use of field types (i.e. dates) makes complex Boolean searches possible.

The structured free-text format pioneered by Alan Melby in the mtx software ten years ago is hierarchical rather than relational in nature. As a result, an mtx-type database is easier for an end user to set up. A well designed structured free-text file is not fiat like that of a simple “card file” program; rather, its fields can have complex internal relationships, thereby making it fiexible enough to handle such things as synonyms and alternate entries.

Gregory Shreve, of Kent State University‘s Department of Applied Linguistics, has argued for many years in favor of relational databases for terminology. Says Shreve, “Alan Melby and I have disagreed for a long time about the virtues of relational versus hierarchical databases. Fact is, hierarchical systems are an old software technology. The problem that most people have with relational databases is that they don‘t understand the design process for setting up the relational tables, in particular the process of normalization.”

“Given a good design, a relational system can be extremely flexible and, I believe, will outperform a database model such as that used in the mtx software. A properly set up relational database will have a number of primary relations that are linked by co-relations that consist primarily of keys. These provide extensive and flexible linkage between data items without duplicating any data elements. Further, with a good database design the ability to implement complex searches is enhanced. mtx-type databases are record-oriented, the logical organization within the record is up to the user and very flexible from that viewpoint. That trades off against the ability to manipulate the individual data categories that you gain in a relational system.”

“A well designed relational system should be able to take the total pool of relations and generate a large number of different virtual or logical records from a common physical data pool. There is, in a sense, no single physical record, but a large, relationally linked data pool navigated in different ways by different users, who each have their own ’views‘ of the data. In my opinion, people who think that relational systems are too inflexible don‘t understand their full power because they are still thinking about a single relation as the analogue of a physical record in a hierarchical system.”

Alan Melby believes that the ideal implementation of a termbase lies somewhere between traditional databases with fixed length fields at one end of the spectrum and completely unstructured text on the other. It consists of records of structured text with automatic validation and multiple indexed fields. He feels that hierarchical records are more intuitive than groups of relational tables and more closely correspond with the natural form of terminological and lexical data.

Melby is not opposed to relational databases. Rather, he has had to be pragmatic. “Ten years ago, we were writing software for pc xts and we couldn‘t fit a relational database into a 100 Kb tsr, so we adopted the structured free-text format. We have since extended this approach with data validation, automatic indexing, and Boolean search facilities, things which are normally only available with relational databases. The time may come, however, when relational databases are easier to set up and use.” A package like keyterm seems to be pointing in that direction. In the spirit of friendly competition, Melby would like to see both approaches pushed to their maximum capabilities and have the market decide.

Whatever your philosophy, if your terminological data is TIF conformant, you can have it both ways: you can convert your terminology to either structured text records or complex relational database tables. With TIF, your data is not imprisoned within a given system. Because it is a comprehensive - and extendable - common format, TIF frees software engineers to implement the approach that they feel is best.