Chapter 3

A Perspective on the Quest for Global Knowledge Interchange

Steven R. Newcomb
(includes some material cowritten with Michel Biezunski)

In 1989, Yuri Rubinsky[1] made a video that he hoped would compel any viewer to grasp the importance of SGML, the ISO standard metalanguage from which has come much of the “Internet revolution,” including HTML and XML. The intent of the video was to dramatize the enormous significance of a simple but revolutionary idea: any information—*any* information—can be marked up in such a way as to be parsable (understandable, in a certain basic sense) by a single, standard piece of software, by any computer application, and even by human readers using their eyes and brains.

In the video, aliens from outer space understand a message sent from Earth, because the message is encoded in SGML. This little drama occurs after the aliens first misunderstand a non-SGML message from Earth. (They have already eaten the first message, believing it to be a piece of toast.)

At the time, I was having great difficulty helping my colleagues understand the nature of my work, and I thought maybe Yuri’s video would help. One of my colleagues, who had funding authority over my work, was surprised that I had never explained to him that the purpose of my work was to foster better communications between humans and aliens. He was quite serious.[2]

[1] Yuri Rubinsky (1952–1996) was not only a great wit and a Renaissance man; he was also a leader in thought whose words, deeds, dreams, and dedication continue to inspire people who work together to realize the promise of global knowledge interchange.

[2] Still attempting to make his point, Yuri made several more videos, one of which, with no alien subplot, was ultimately published as *SGML, The Movie*.

This experience and many others over the years have convinced me that, while the technical means whereby true global information interchange can be achieved are well within our grasp, there are significant anthropological obstacles. For one thing, it's very challenging to interchange information about information interchange. As human beings, we pride ourselves on our ability to communicate symbolically with each other, but comparatively few of us want to understand the details of the process. Communication about communication requires great precision on the part of the speaker and an unusually high level of effort on the part of the listener. I suspect that this is related to the fact that many people become uncomfortable or lost when the subject of conversation is at the top of a heap of abstractions that is many layers thick. It's an effort to climb to the top, and successful climbs usually follow one or more unsuccessful attempts.

When you have mastered the heap of abstractions that must be mastered in order to understand how global information interchange can be realized, the reward is very great. The view from the top is magnificent. From a technical point of view, the whole problem becomes simple. Very soon thereafter, however, successful climbers realize that they can't communicate with nonclimbers about their discoveries. This peculiar inability and its association with working atop a tall heap of abstractions are evocative of the biblical myth of the Tower of Babel. Successful abstraction-heap climbers soon find themselves wondering why their otherwise perfectly reasonable and intelligent conversational partners can't understand simple, carefully phrased sentences that say exactly what they're meant to say.

You have now been warned. This book is about the topic maps paradigm, which itself is a reflection of a specific set of attitudes about the nature of information, communication, and reality. Reading this book may be quite rewarding, but there may also be disturbing consequences. Your thinking, your communications with others, and even your grasp of reality may be affected.[3]

Information Is Interesting Stuff

Information is both more and less real than the material universe. It's more real because it will survive any physical change; it will outlast any physical manifestation of itself. It's less real because it's ineffable. For example, you can touch a shoe, but you can't touch the notion of "shoe-ness" (that is, what it means to be a shoe). The notion of shoe-ness is probably eternal, but every shoe is ephemeral.

[3] The writings of Plato, the ancient Greek philosopher who pioneered many of the basic philosophical ideas, have been having similar effects on their readers for thousands of years.

The relationship between information and reality is fascinating. (By *reality* here I mean "the reality of the material universe"—or what we think of as its reality.) We all behave as if we believe that there is a very strong, utterly reliable connection between information and reality. We ascribe moral significance to the idea that information can be *true* or *false:* we say that it's true when it reflects reality and false when it doesn't. However, there is no way to prove or disprove that there is any solid, objective connection between symbols and reality. Symbols are in one universe, reality is in another; human intuition, understanding, and belief form the only bridge across the gap between the two universes. The universe of symbols is a human invention, and our arts and sciences—the information resources that human civilization has accumulated—are the most compelling reflection of who and what we are.

Money, the "alienated essence of work" as some philosophers have put it, is also information. I once saw Jon Bosak[4] hold up a dollar bill in front of an XML-aware technical audience, saying, "This is an interesting document." The huge emphasis that our culture places on the acquisition of money is a powerful demonstration of our confidence in the power of information to reflect reality or, more accurately, in the power of information to *affect* reality. In the United States, we have a priesthood called the Federal Reserve Board, answerable to no one, whose responsibility is to protect and maximize the power of U.S. dollars to affect reality. The Fed seeks to control monetary inflation, for example, because inflation represents a diminishment of that power.

Thinking of money as a class of information suggests an illustration of the importance of context to the significance of information for individuals and communities: given the choice, most of us prefer money to be in the context of our own bank accounts. Thinking of money as information leads one to wonder whether information and money in some sense are the same thing. Some information commands a very large amount of money, and the visions of venture capitalists and futurists are often based on such intellectual property. In some circles, the term *information economy* has become a pious expression among those who are called upon to increase shareholder value. (On the other hand, the economic importance of information can be overstressed. Information when eaten is not nourishing, and when it is put into fuel tanks, it does not make engines run.)

Information has far too many strange and wonderful aspects to allow them all to be discussed here; I regret that I can only mention in passing the mind-boggling insights offered by recent research in quantum physics, for example.

For purposes of this writing, anyway, the most interesting aspect of information is the unfathomable relationship between information and the material universe, as well as

[4] Jon Bosak is widely regarded and admired as the father of XML.

the assumptions we all make about that relationship in order to maintain our global civilization and economy. That unfathomable relationship profoundly influenced the design of the topic maps paradigm. Those who would understand the topic maps paradigm must appreciate that there is some sort of chasm between the universe of information (that is, the world of human-interpretable expressions) and the universe of subjects that information is about—a chasm that is (today, anyway) bridgeable only by human intuition, not by computers. The topic maps paradigm recognizes, adapts itself to, and exploits this chasm. (We'll discuss this later.)

Information and Structure Are Inseparable

Excuse me for saying so, but there is no such thing as "unstructured information." Even the simplest kind of information has a sequence in which there is a beginning, a middle, and an end, some concept of unit, and, usually, several hierarchical levels of subunits. Information always has at least one intended mode of interpretation, and the interpretability of information is always utterly dependent on the interpreter's ability to detect structure.

Written and spoken natural languages have structures, although their structures are so subtle, variable, nuanced, and driven by human context that computers are still unable to understand natural languages reliably, despite many years of intense effort by many excellent minds. The fact that computers cannot reliably understand natural languages does not justify terming natural languages "unstructured." This strange term, *unstructured information*, was coined in order to distinguish information whose structure can be reliably detected and parsed by computers (*structured information*) from information, such as natural languages, that does not readily submit to computer processing given state-of-the-art technology (*unstructured information*).

Formal Languages Are Easier to Compute Than Natural Languages

Computers aren't reliable translators of human communication, but humans can translate simple aspects of their various affairs into the patois of computers. We call these expressively impoverished languages *formal languages*, which makes them sound a lot better than they are. Virtually everything that computers do for our civilization involves the use of formal languages.

If you think you are unfamiliar with formal languages, you are mistaken. Dialing a telephone number constitutes a kind of formal utterance; telephone numbers have a rigid syntax that constitutes a kind of formal language. Around the globe, different localities use different formal languages for controlling the behavior of telephone switches. In North America, for example, one of the syntactic rules of the local formal language for dialing telephone numbers is that, in order to reach a telephone whose number is outside the local area but still within North America, a *1* must be the first digit dialed when the dial tone is heard. This syntactic rule is not very expressive, but, like most of the features of most formal languages, it's simple, deterministic, and highly computable. It's so easily understood by machines, in fact, that this simple syntactic rule has been enforced by telephone switches in North America for decades.[5]

Generic Markup Makes Natural Languages More Formal

Starting in 1969, a research effort within IBM began to focus on generic markup in the context of integrated law office information systems.[6] By 1986, Charles Goldfarb had chaired an ANSI/ISO process that resulted in the adoption of Standard GML, also known as Standard Generalized Markup Language (SGML, ISO 8879:1986). Today, SGML is the gold standard for nonproprietary information representation and management; XML, the eXtensible Markup Language of the Web, corresponds closely to a Web-oriented ISO-standard profile of SGML called WebSGML. The Web's traditional language for Web pages, HTML, is basically a specific SGML tag set or markup vocabulary. XML, like SGML, allows users to define their own markup vocabularies.

SGML was based on the notion that natural language text could be marked up in a generalized fashion, so that different markup vocabularies (or tag sets) could be used to mark up different kinds of information in different ways, for different applications, and yet still be parsable using exactly the same software, regardless of the markup vocabulary. Since interchangeable information always takes the form of a sequence of characters, the ability to mark up sequences of characters in a way that is both standard (one piece of software works for everything) and user-specifiable (users can

[5] Less than ten years ago, the whole world was changed when the World Wide Web made it possible to give, in effect, telephone numbers to sources of information. These "telephone numbers" are known as Web addresses. For example, one such Web address, *http://www.w3.org*, is the most important source for information about the World Wide Web: it is the Web address of the World Wide Web Consortium. Needless to say, Web addresses are expressed by way of formal languages, one of which is known as the Hypertext Transport Protocol (HTTP).

[6] The team ultimately included Goldfarb, Mosher, and Lorie, whose initials became the name of the language: *GML*.

invent their own markup vocabularies) has turned out to be a key part of the answer to the question, “How can global knowledge interchange be supported?”

The SGML and XML languages that ultimately grew out of the early GML work now dominate most of the world’s thinking about the problem of global information interchange. These languages represent an elegant and powerful solution to the problem of making the structure of *any* interchangeable information easily and cheaply detectable, processable, and validatable by *any* application.

Perhaps the most fundamental insight that led to the predominance of SGML and XML is the notion of *generic markup*, as opposed to *procedural markup*. Procedural markup is exemplified by tag sets that tell applications what to do with the characters that appear between any specific pair of tags (an element start tag and an element end tag). For example, imagine a start tag that says, in effect, “Render the following characters in italics,” followed by the name of a ship, such as *Queen Mary*, followed by an end tag that says, in effect, “This is the end of the character string to be rendered in italics; stop using the italic font now.” This set of instructions is indicated by the following syntax:

```
<italics>Queen Mary</italics>
```

These font-changing instructions are very helpful for a rendering application, but they are virtually useless for supporting applications that are looking for occurrences of the names of ships because many things are italicized for many reasons, not just the names of oceangoing ships. It turns out that generic markup offers significant economic benefits to the owners of information assets. For example, a start tag (for example, `"ship-name"`) that, in effect, says, “The next few characters are the name of a ship,” that is, what *kind* of thing that character string is, is just as useful for rendering purposes as one that says, “Italics start here,” but the generic tag can support many more kinds of applications, including applications that weren’t even imagined when the information asset was originally created. Generic markup is not application-oriented; it is information-oriented. It provides information (*metadata*) about the information that is being marked up.

A start tag is a piece of formal, computer-understandable data that can appear in the midst of natural language data that the computer does not understand. Because of generic markup, we can now use computers to help us manage and interchange information in a hybrid fashion: the computer understands the computer-oriented formal information, and the rest is often explicitly rendered for human consumption.[7]

[7] The use of XML as a kind of communications protocol for business transactions between Web-connected business applications is probably less challenging. In such applications, XML is not necessarily chosen for its ability to represent hybrid resources. Instead, XML is chosen simply because “well-formed” XML is easily parsed by free software, and perhaps also because it is not difficult to debug problems in information that is represented in XML because XML is directly readable by human beings.

But problems remain.

- How, for example, are computers supposed to understand what the tags mean? The `"ship-name"` tag, by itself, could easily be misunderstood as indicating the beginning of the name of the recipient of some sort of shipment of merchandise, for example. Let's forget about computers for a moment and consider human beings instead. No matter which natural language you choose, most of the people on this planet can't read it. Even those who can read English may use a local dialect that may cause them to be misled as to the significance of a tag name. In general, how are human beings supposed to understand that this particular tag's intended purpose is limited to marking up the names of oceangoing ships? It is difficult to see how the dream of global knowledge interchange can be realized in the absence of a rigorous way to provide metadata about any kind of metadata, including markup.
- What about information that isn't marked up very well (or at all) to begin with?
- What about information whose structure is arguable or ambiguous? It can only be marked up one way at a time, unless you're willing to maintain two versions of the same source information—a strategy that can often be more than twice as expensive as maintaining a single source.
- What if you need to regard information as having a structure that is different from the structure its markup thrusts upon you, and you don't have the right or ability to change it, copy it, or reformat it?

As you can see, generic markup is only part of the answer to the problem of supporting global knowledge interchange. Much of the rest of the answer has to do with other kinds of metadata—kinds of metadata that are not internal to the information assets but are information assets in their own right. Although they are strikingly and subtly different from other kinds of metadata, topic maps are, among other things, just one of many kinds of such external metadata information assets.

A Brief History of the Topic Maps Paradigm

The work on topic maps began in 1991 when the Davenport Group was founded by UNIX system vendors (and others, including the publisher O'Reilly & Associates). The vendors were under customer pressure to improve consistency in their printed documentation. There was concern about the inconsistent use of terms in the documentation of systems and in published books on the same subjects. System vendors wished to include O'Reilly's independently created documentation on X-Windows, under license, seamlessly in their system manuals. One major problem was how to provide master indexes for independently maintained, constantly changing technical documentation aggregated into system manual sets by the vendors of such systems.

The first attempt at a solution to the problem was humorously called SOFABED (Standard Open Formal Architecture for Browsable Electronic Documents).

The problem of providing living master indexes was so fascinating that, in 1993, a new group was created, the Conventions for the Application of HyTime (CApH) group, which would apply the sophisticated hypertext facilities of the ISO 10744 HyTime standard. HyTime had been published in 1992 to provide SGML with multimedia and hyperlinking features. The CApH activity was hosted by the Graphic Communications Association Research Institute (GCARI, now called IDEAlliance). After an extensive review of the possibilities offered by extended hyperlink navigation, the CApH group elaborated the SOFABED model as topic maps. By 1995, the model was mature enough to be accepted by the ISO/JTC1/SC18/WG8 working group as a "new work item"—a basis for a new international standard. The topic maps specification was ultimately published as ISO/IEC 13250:2000.[8]

During the initial phase, the ISO/IEC 13250 model consisted of two constructs: (1) topics and (2) relationships between topics (later to be called *associations*). As the project developed, the need for a supplementary construct, one able to handle filtering based on domain, language, security, and version, emerged; as a result, a mechanism for filtering was added, called *facet.* This approach was soon replaced by a more powerful and elegant vision based on the notion of *scoping*. The notion of scope in topic maps is one of the key distinguishing features of the topic maps paradigm; scope makes it possible for topic maps to incorporate diverse world views, diverse languages, and diversity in general, without loss of usefulness to specific users in specific contexts and with no danger of irreducible "infoglut."

As an aside,[9] note that the scope and subject identity point aspects of the topic maps paradigm were first developed and articulated by Peter J. Newcomb and Victoria T. Newcomb during a 1997 breakfast conversation at the Whataburger restaurant in Plano, Texas. In our family, we still sometimes call those aspects the Whataburger model, although the Whataburger interchange syntax has not survived. The XTM conceptual model accurately reflects the Whataburger model, however; it has stood the test of time. It's interesting to note how the syntax of topic maps has evolved since Whataburger. The syntax that minimally and accurately reflected the Whataburger model turned out to be inexplicable to most people; it was a marketing fiasco. Michel Biezunski, who for many reasons is the primary hero of the story of topic maps, is not coincidentally also the origin of what I call Biezunski's Principle. Simply put, Biezunski's Principle is: *There is no point in creating a standard that nobody can understand.*

[8] For more information, see *http://www.y12.doe.gov/sgml/sc34/document/0129.pdf.*

[9] One far too verbose for a simple footnote!

(Another way he sometimes puts it is, "I'm not interested in convincing anyone that we are smarter than they are.") The whole idea of having a syntactic element type that corresponds to the notion of a topic is, in strictly technical terms, totally unnecessary baggage that actually obscures the deeper and beautifully simple structures that topic maps embody. Even so, the `<topic>` element type is the foundation of the syntax of topic maps, both in the ISO standard and in the XTM specification. This is because people intuitively and quickly grasp the notion of `<topic>` elements, and the whole idea that a topic can be represented syntactically as a kind of hyperlink is an inherently exciting one. For me, the popularity of the `<topic>` element type and the marketing success that the topic maps paradigm now represents are convincing demonstrations of the power of Biezunski's Principle. (I think Biezunski's Principle owes much to the work of Tim Berners-Lee and others, whose design for the World Wide Web succeeded in opening a whole frontier of human interaction and endeavor, where other designs, including more intellectually elegant and powerful ones, had failed to get serious global traction. But that's another story.)

The ISO 13250 standard was finalized in 1999 and published in January 2000. The syntax of ISO topic maps is at the same time very open and rigorously constrained, by virtue of the fact that the syntax is expressed as a set of *architectural forms.*[10] (Architectural forms are structured element templates; this templating facility is the subject of ISO/IEC 10744:1997 Annex A.3.[11]) Applications of ISO 13250 can freely subclass the element types provided by the element type definitions in the standard syntax, and they can freely rename the element type names, attribute names, and so on. Thus, ISO 13250 meets the requirements of publishers and other high-power users for the management of their source codes for finding information assets.

However, the advent of XML and XML's acceptance as the Web's *lingua franca* for communication between document-driven and database-driven information systems created a need for a less flexible, less daunting syntax for Web-centric applications and users. This goal, which was achieved without losing any of the expressive or federating power that the topic maps paradigm provides to topic map authors and users, is the purpose of the XTM (XML topic maps) specification.

The XTM initiative began as soon as the ISO 13250 topic maps specification was published. An independent organization called TopicMaps.Org,[12] hosted by IDEAlliance, was founded for the purpose of creating and publishing an XTM 1.0 specification as quickly as possible. In less than one year, TopicMaps.Org was chartered and

[10] Enabling technology for XML and SGML architectural forms is freely available at *http://www.hytime.org/SPt.*

[11] You can access the text of this annex at *http://www.ornl.gov/sgml/wg8/document/n1920/html/clause-A.3.html.*

[12] See the organization's Web site at *http://www.topicmaps.org.*

the core of the XTM 1.0 specification was delivered at the XML 2000 conference in Washington, DC, on December 4, 2000, with the final version of XTM 1.0 delivered on March 2, 2001.

Michel Biezunski (of InfoLoom) and I (of Coolheads Consulting) were the founding cochairs of TopicMaps.Org and coeditors of the Core Deliverables portion of the XTM specification as well as of the remaining portions of the Authoring Group Review version of the specification. In January 2001, Graham Moore (of Empolis) and Steve Pepper (of Ontopia) became the new coeditors, and Eric Freese (of ISOGEN/DataChannel) became the chair of TopicMaps.Org. More recent events in the history of XTM and TopicMaps.Org are discussed in Chapter 4.

Data and Metadata: The Resource-Centric View

Metadata is not only “about data”—it is also always data, itself. One person’s data is another person’s metadata. There is, in general, no difference between data and metadata; it’s all a matter of perspective.

It is normal to think of metadata as being somehow “in orbit” around the data about which the metadata provides information. The existence of a *metadata* Web site that provides information about *data* Web sites affects global knowledge interchange in two ways.

1. When users are at the metadata Web site, their attention can be directed at one or more data Web sites, and users can know the reasons why.
2. When users are at the data Web site, they may derive more useful information if they also know about the availability of the metadata Web site and its reasons for expressing metadata about that data.

The idea that metadata can be externally and arbitrarily associated with data is a powerful one, but, by itself, this attractive and simple idea leads nowhere. When a single data Web site is associated with (that is, pointed at by) millions of metadata Web sites, the result can easily be “infoglut”—such a tidal wave of information that, as a practical matter, its overall utility is zero. There needs to be a way to use computers to determine the relevance of all this information to the user’s specific situation and to show the relevant information while hiding the rest.

It is ironic that the recent huge improvement that information technology has brought to the accessibility of information—such as providing instant hyperlink traversal to any Web site, anywhere in the world—has itself made more and more information *inaccessible* due to the sheer quantity of it. The dream of global knowledge interchange recedes, even as it becomes real. Our power to filter out unwanted information must keep pace with the quantity of unwanted information. It's a race that we currently appear to be losing.

Although it may sound strange, it is imperative that we develop technical, economic, and business models that will allow businesses to make money by *hiding* information—by providing information that can be used to hide other information. It's also imperative that these models absolutely support and cherish diversity. This is because particular information filtration problems may, as a purely practical matter, require hiding information that emanates from a variety of sources and that reflects a variety of worldviews. These diverse sources may not even know about each other, much less deliberately design their products in such a way as to make them "federable" (that is, usable in concert) with one another. This is what the topic maps paradigm is all about: making diverse metadata sources more or less automatically federable.

One of the things that a metadata Web site may usefully provide is information as to which other Web sites have information on specific topics. Such metadata Web sites are often (and misleadingly) called search engines. But search engines do not usually provide topically organized information. Yahoo! is one notable exception, but it works only for a small number of topics and only in ways that are consistent with Yahoo!'s singular and necessarily self-serving view of the wide world of information. Instead, unlike Yahoo!'s topically oriented features, most search engines merely provide information about which other Web sites provide information that contains certain strings of characters. A user interested in information on a particular topic must be clever enough and lucky enough to be able to sneak up on relevant information on the basis of strings that he or she hopes will be found in such information—and not found in too much other information. The user must guess the language of the desired Web sites' information well enough to imagine which strings are relevant.

When a user attempts to find information, the user usually has a particular topic in mind about which he or she wishes to know more. The user is not interested in Web sites or specific information resources, except insofar as they offer information that is specifically relevant to that topic. The first order of business, then, really should be to allow the user and the computer to agree about exactly what topic the user wants to research. Once the computer has established the exact topic, the computer's task should be to hide all the information about the topic that, for one reason or another, the user should not be bothered with and to render only the remaining information. This kind of user interaction with the Web is supportable if topic maps are widely used because the topic maps paradigm explicitly permits and supports business models based on the development and exploitation of lists of topics that have names and occurrences in multiple languages for use in multiple contexts and that can themselves be found on the basis of their relationships with many other findable topics.

Still, there is an unbounded number of topics, there is an awful lot of information out there, and the sheer quantity is growing at a phenomenal rate. Many individual pieces of information can often be regarded as being relevant to many different topics simultaneously. Nobody will ever categorize everything, but many people will categorize some of it many times over, often in different and even conflicting ways.[13] The topic maps paradigm explicitly permits and supports business models that are based on the development and exploitation of categorizations of information resources. Every category can be represented as a topic. Similarly, every system of categorization can also be represented as a topic. In fact, there is nothing that can't be represented as a topic. The exploitation of preexisting categorizations is not only the key to hiding unwanted information; it's also the key to finding it in the first place, unless it happens to contain some string that you are lucky enough to guess and that doesn't also appear in more than a few other resources.

Metametadata, Metametametadata . . .

One way to federate metadata is to create metadata about the metadata. Then, of course, we may need to federate that metametadata with other metametadata, using metametametadata. The absurdity of this approach is obvious: there is little opportunity for benefit to be realized from standardization in a model that requires infinitely recursive metalevels. There must be a better way. And there is: the topic maps paradigm moves in the other direction by recognizing the existence of a single, implicit, *underlying* layer. It's the same underlying universe that is known in philosophical circles as *Platonic forms*[14] (so named for Plato, the ancient Greek philosopher mentioned earlier).

Subjects and Data: The Subject-Centric View

The notion of "shoe-ness" has already been mentioned as a notion that is eternal but ineffable, while any given shoe is ephemeral but concrete. As Plato might have pointed out, only our minds can sense shoe-ness, and only directly; we cannot sense shoe-ness with any of our five physical senses, even though we can certainly sense a given shoe in a variety of ways. We can be aware of shoe-ness—even the shoe-ness of

[13] Aristotle, who extended and applied Plato's ideas, proposed a very famous and influential system of categorization. Aristotle did not have to face the current situation in which many diverse, evolving, and useful worldviews—systems of categorization—must be allowed and encouraged to participate fully in a global civilization.

[14] The term Platonic form escapes simple description. A good Web page on the topic is *http://www.soci.niu.edu/~phildept/Dye/forms.html.*

a particular shoe— only with our minds. For Plato, shoe-ness exists in a plane of existence that is somehow more exalted, perhaps because it is more permanent than anything our five senses can sense. Plato's idea that there is a plane of existence that is accessible only by our minds is exploited by the topic maps paradigm in order to make data resources federable without endless layers of metadata upon metadata.

The topic maps paradigm recognizes that everything and anything can be a subject of conversation, and that every subject of conversation can be a hub around which data resources can orbit. Unlike the resource-centric view in which metadata orbits data resources, in the subject-centric view, data orbits subjects. If the subject itself happens to be a data resource, the orbiting data can, of course, be called metadata. But one of the essential lessons of the topic maps paradigm is that *all* data is *data about subjects*, but only some subjects are themselves data; most subjects are not information resources. When the problem of global knowledge interchange is approached with this subject-centric attitude, the solution becomes much simpler and easier. Indeed, for many people, and particularly for the people who have used it the most, the topic maps paradigm passes the most convincing test of all: the solution, once finally found, is obvious.

There is one problem: computers cannot access subjects unless those subjects happen to be information resources themselves. A computer cannot access the Statue of Liberty, for example, or love, or hot chocolate, or shoe-ness. There is no computer-processable pointer to any of these things. As a practical matter, there is no human-processable pointer to these things either—people can't wave their hands and produce these things out of thin air. However, people have another gift that makes it unnecessary to produce concrete things in order to discuss them: the ability to communicate symbolically, to understand each other on the basis of symbols. It's an everyday miracle that I can say to you the words, "Statue of Liberty," and you will immediately know I'm talking about a certain large greenish statue of a woman, created by Gustav Eiffel, that is situated on Liberty Island in New York Harbor, with a somewhat smaller prototype located in Paris, France. There is very little chance that you will misunderstand me (although it's possible that I could be referring to a certain unconventional pattern of play in American football).

If you've followed this discussion so far, you're ready to understand some imagery that was pivotal in the development of the topic maps paradigm. Imagine a chasm with two high cliffs, one on the left side of the chasm and one on the right. There is no physical bridge across the chasm. On the left-hand cliff is the universe of symbols and expressions. All written, pictorial, and other symbolic expressions exist on the left-hand cliff. On the right-hand cliff is the world of subjects of conversation. (The conversations themselves, since they are in the universe of symbolic expressions, are found only on the left-hand cliff.) On the right-hand cliff we find love, the Statue of Liberty, shoe-ness, the smell of hot chocolate, Minnie Mouse's high-heeled shoes, and every other thing that is or can ever be symbolized by the expressions found on the left-hand cliff: every actual and possible topic of conversation, without exception.

The first thing to realize about this imagery is that, while there is no bridge across the chasm, crossing it is the everyday miracle that our brains accomplish whenever we successfully understand any symbolic expression. We sense certain symbols, and somehow we intuit the corresponding thing on the right-hand cliff. Human intuition (the human brain, if you like) is the only transportation facility that can cross the chasm. This means that it must be true that it's possible for symbols to represent reality or, at least, that we constantly *assume* that symbols represent reality. (As engineers, we are compelled to admit that the fact that everybody assumes that it's true is good enough to get the job done.) As in the case of monetary information, for example, the validity of that assumption is what the high priests at the Federal Reserve Bank are supposed to ensure. Actually, civilization itself rests entirely on the unprovable assumption that information has some bearing on reality, so maybe we can afford to take a chance on it.

The second thing to realize about this imagery is that all data and all metadata are entirely on the left-hand cliff. The left-hand cliff has some reality, too, because information (expressions) do indeed exist. Wondrous to say, there is no "missing bridge to reality" problem on the left-hand cliff. When a subject happens to be an information resource, even an inanimate computing device can take us where we want to go by understanding and executing the symbols (Web addresses, for example) that uniquely identify that information resource. Indeed, history seems to show that the ease of accessing such addressable subjects—information resources—has in fact seduced us into thinking that only resources—symbolic expressions that can be addressed by computers—can be the hubs around which data can be organized.

And here is where the topic maps paradigm performs a bit of chicanery. Computers can't directly address the Statue of Liberty, for example, but they can address information about the Statue of Liberty. More to the point, they can address an information resource that serves as a surrogate for the Statue of Liberty. Since we're stuck with the limitations of computers (and the underlying limitations of symbolic expressions), the key is to allow anyone and everyone to establish conventions for such surrogates, according to their own needs and convenience, whereby arbitrary subjects can be uniquely represented by specific addressable information resources. The topic maps paradigm accomplishes this trick by taking the position that a certain specific kind of reference to an information resource must be interpreted not as a reference to that resource but rather as a reference to whatever subject of conversation is indicated by that information resource, when that information resource is perceived and understood by a properly qualified human being. In some sense, then, the topic maps paradigm lets the computer take a virtual journey across the chasm by riding on human perception and intuition.[15] The referenced resource becomes more than a resource: it becomes a symbolic surrogate, on the left-hand cliff, for something on the right-hand cliff, on the other side of the chasm, where only human intuition can reach.

Understanding Sophisticated Markup Vocabularies

If you want to understand the topic maps paradigm, you must understand something about markup vocabularies in general that is not yet widely understood: the structure of an interchangeable resource is not necessarily the same as the structure of the information that is being conveyed.

Back in 1986, SGML had just been adopted by the community of nations as the one-and-only markup language for everything and everybody. But Charles Goldfarb, its inventor and guardian, knew that much work remained to be done. He saw that many kinds of multimedia information and many business niches for such information would continue to be invented indefinitely. One of the things he wanted to do was to show that SGML could be used to encode multidimensional synchronizing information: to impose simultaneous, arbitrary temporal structures on arbitrary collections of information objects and their components.

Accordingly (and not coincidentally in order to have some fun), Dr. Goldfarb turned his attention to the problem of representing music abstractly.[16] Musical works are inherently multidimensional; to begin with, musical harmony is the result of multiple simultaneous melodies. Since an interchangeable document is necessarily a one-dimensional

[15] In a way, it's not very different from the insertion of formal, computer-processable tags into natural language data that the computer cannot understand. In the end, the utility of marked-up natural language information (and the utility of subject-indicating referenced information) is available only to human minds, but, because of the formality of the markup and the formality of the expression of reference to subject-indicating information, computers can be used to vastly enhance the productivity of the human minds to which the information is being made available.

[16] Dr. Goldfarb and I first met in July 1986 at the first meeting of the ANSI X3V1.8M committee, which he chaired. The mission of ANSI X3V1.8M was to create a Standard Music Description Language standard. We have been colleagues in the development of ANSI and ISO standards ever since, and we have both invested much of ourselves in our brainchildren. Ultimately, the music standard metamorphosed into the ultra-generalized ISO HyTime standard (ISO/IEC 10744:1997; see *http://www.ornl.gov/sgml/wg8/document/n1920*), and the music standard became an application of HyTime. HyTime is a holistic solution to the question of how to create metadata assets that impose all kinds and combinations of arbitrary alternative structures on arbitrary sets of arbitrary information resources.

sequence of characters, the question immediately arises, in the case of a musical document, as to whether the concurrent melodies (or instrumental and/or vocal parts) should be expressed separately or whether all the notes that are supposed to sound synchronously in all of the concurrent melodies should appear adjacent to one another in the interchange file. Either way, the structure of the interchange syntax will be inconvenient for at least some applications. Either way, at least some of the basic structure of the information will be obscured by the interchange syntax. Therefore, for the sake of reliable information interchange, there must be a separate and distinct model of the information that is being conveyed by the music language, in addition to the syntactic model that governs the structure of that information while it is represented as an interchangeable document.

There are many kinds of information whose structure, like the structure of music information, must respond to one set of requirements when the information is being interchanged and to another, often contradictory set of requirements when the information is in ready-to-use form. Many decision makers are not yet ready to hear this message, for a variety of reasons.

Historically, the overwhelming majority of markup applications have been basically batch-typesetting jobs, which start at the beginning of the document and process each data segment in more or less the same sequence in which it appears in the document. The rendering of HTML documents by Web browsers is one example. The use of the word *document* to denote a class of information objects appears to have the connotation that all such information objects are intended to be rendered and used in the same order in which they are interchanged.

Currently, significant investments in the marketing of XML technology are directed at business-oriented information technology professionals. Such professionals are urged to regard XML as an opportunity to represent relational databases as interchangeable documents. All such documents, regardless of their schemas, are parsable by a single standard parsing technology, without reconfiguration. It's obvious that a relational table is exportable and importable as a sequence of named or numbered rows, each of which is itself a sequence of named or numbered fields.

The Document Object Model (DOM)[17] recommended by the World Wide Web Consortium (W3C) provides a convenient application programming interface (API) to the syntactic structure of information being interchanged in the form of XML documents. The DOM is extremely useful, but it has been oversold as the *ne plus ultra*

[17] The W3C DOM is not an object model; it's an API to a "DOM tree" whose exact nature is still being specified by a W3C working group. The task of this working group is to produce an object model (or at least a set of constraints on the structure of a DOM tree) called the XML InfoSet.

API to interchangeable information. The DOM does provide applications with random access to every part of an interchangeable document, so it makes many applications much easier to develop than they otherwise would be. However, the DOM cannot provide direct access to the semantic components of what a document *means;* it can only provide direct access to the syntactic components of how a document is represented for interchange.

Fortunately for the widespread acceptance of XML technology, which is basically a tremendous step toward global knowledge interchange, there are many popular kinds of information whose interchange is required for many kinds of economic reasons, including virtually all of the billboards on the information highway, for which the interchange structure can quite usefully be the same as the structure of the API. The DOM is a great all-purpose API for all of these kinds of information.

Topic maps are another matter, however. As in the case of music information, the structure of topic map information is not the same as the structure of interchangeable documents.

- Topic map documents can point to other topic map documents, saying, in effect, "The referenced topic map must be merged with the current one before the current one can be understood as its author intends." If any single subject is represented by `<topic>` elements in both topic maps, the topic maps paradigm requires that the result of processing the two documents must be, among other things, exactly one resulting topic (represented in some application-internal form) that has the union of the characteristics (the names, occurrences, and participations in associations with other topics) of the two `<topic>` elements. Therefore, the *only* way to understand an interchangeable topic map document is to process it fully, performing such merging and redundancy-elimination tasks as the paradigm requires.
- The element-containment structure of a topic map document, even in the absence of any requirement to merge it with another topic map document, bears no resemblance to the structure of the relationships between topics that are expressed by that document.

In other words, the API to topic map information is not, and can never be, the same as an interchangeable topic map that conveys that same information. From this interesting fact the question arises, "What is meant by an element type name, such as `<topic>`, in an interchange syntax like the interchange syntax of topic maps, in which there is no direct correspondence of the element structure to the structure of the information being interchanged?"

The answer is that the meaning of such a tag name is, like all other tag names, exactly what the designers of the interchange syntax intended it to mean. For example, for every `<topic>` element, a conforming topic map application must have an application-internal representation of that topic (that is, a topic whose subject is the same as that of the `<topic>` element). If there is no such internally represented topic, the application must create one; if there is already such an internally represented topic, the application must add to it (union it with) all the information about that topic that is represented by the `<topic>` element. The meaning of the `<topic>` tag name is still quite clear and rigorous; the only difference is that the meaning has to do with the creation of an application-internal form of the interchanged information—a form with its own API that must be used by conforming applications.

The Topic Maps Attitude

The topic maps paradigm is a step along the road to global knowledge interchange. It may well turn out to have been quite a significant step. Nonetheless, it is very obviously not the last step. If it successfully moves our species forward toward global knowledge interchange, the topic maps paradigm will owe much of its success to the fact that it is resolutely responsive to current technological, economic, and anthropological conditions, and just as resolutely responsive to certain philosophical values and attitudes. Some of these values came from the comparatively young traditions of the markup languages community.[18] Other values are derived from much older traditions. What follows is a summary of the values and perspectives that I find most remarkable.

- We must recognize that civilization is what makes it possible for us to have breakfast every morning, and civilization's increasing ability to develop and exploit information resources is generally correlated with the richness and quality of life available to each human individual living on our planet. Global knowledge interchange is important to every single living human being.
- We must cherish diversity by giving diverse worldviews the ability to be expressed and exploited alongside and in federated combination with all other worldviews. This includes respecting communities of interest, encouraging their formation, and not coincidentally causing them to provide themselves with usable interfaces for use by other communities of interest.
- We must understand that worldviews provide essential contexts for communication and that communication rests on our intuitive ability to cross the chasm between symbolic expressions and reality. We must work to provide

[18] The vanguard of the markup languages community still meets annually at a very lively conference called Extreme Markup Languages, where a significant portion of the history of topic maps has occurred in plain public view. See *http://www.idealliance.org* for details of the next conference.

computers with increasing sensitivity to (that is, apparent awareness of and ability to act upon) diverse human contexts.

- We must accept partial solutions and partial expressions, demanding neither comprehensiveness nor perfection. There never will be any such thing as a "complete" topic map, or one true ontology suitable for all contexts, or a holy grail of "knowledge." A single human being or organization can accomplish something only within some limited scope. Providing a way for incomplete, imperfect utterances to contribute, in some useful way, to the ongoing intellectual life of the human species is essential.
- We must understand and adapt to the fact that different subjects of conversation have different kinds of reality, for example, an information asset is real in one sense, the Statue of Liberty in another, shoe-ness in a third, and Minnie Mouse's high-heeled shoes in a fourth. At the same time, we must understand and exploit the fact that all subjects are, in some sense, the same, in that we humans seem to find them worthwhile to discuss.
- We must provide a way for ordinary people to quickly and easily gain a superficial understanding of global knowledge interchange—a way that does not compromise a deeper level of abstract simplicity and power.
- We must abandon "simplifying assumptions" that actually interfere with our ability to manage and maintain our increasingly complex civilization (for example, the resource-centric view of metadata and the idea that the interchange structure of information should always be the same as the structure of the information itself).
- We must provide technology that is suitable as a foundation for business models that, in the aggregate, make many significant contributions to global knowledge interchange and the general availability of knowledge.
- We must recognize infoglut as the single most formidable remaining enemy of global knowledge interchange in a world where the connectivity problem is already well on the way to being permanently solved.
- We must recognize that subjects of conversation are the true axis points of information, even though they are not addressable by computers. Creating addressable information resources to represent nonaddressable subjects allows the addressable resources to be used as public "hooks," called *published subject indicators* (see Chapter 5), on which topic relationships, names, and relevant information can be "hung."
- We must acknowledge that generic markup is the most natural and most economically conservative way to interchange and archive valuable information assets whose future exploitability cannot be completely predicted (that is, practically all information assets).
- We must accept that markup (whether generic or procedural) will always be too rigid or otherwise inadequate for all applications. Thus we must support the ability to impose arbitrary structure on arbitrary information by means of external, independently maintained metadata.

- We must understand the need for markup and other metadata to be described, even as they themselves describe other data.
- We must recognize that the federation of knowledge assets is an ongoing activity that must account for the evolution of the knowledge assets to be federated, without losing the value of investments in previous federating activities.

Summary

This chapter shows that topic maps provide us with two different and important views into an information space: (1) a resource-centric view, one in which we use metadata to describe the resources we reference with topics, and (2) a subject-centric view, in which topic maps provide the tools necessary to represent, to “talk about” subjects. These views, when coupled with the “topic map attitude” that topic maps, where possible, should be unified through merging, provide us with the opportunity for global knowledge interchange.