SCIENTIFIC PUBLICATIONS IN XML - TOWARDS A GLOBAL KNOWLEDGE BASE.

Peter Murray-Rust[1] and Henry S. Rzepa[2]

[1] Unilever Centre for Molecular Informatics, Department of Chemistry, University of Cambridge, Lensfield Road, Cambridge. CB2 1EW.
[2]Department of Chemistry, Imperial College of Science, Technology and Medicine, London, SW7 2AY, England.

ABSTRACT

Recent developments on the World-Wide Web provide an unparalleled opportunity to revolutionise scientific, technical and medical publication. The technology exists for the scientific world to use primary publication to create a knowledge base, or Semantic Web, with a potential greatly beyond the paper archives and electronic databases of today.

Keywords: Extensible-markup-language (XML), Chemical-markup-language (CML), Extensible-stylesheet-language-transformations (XSLT), XML Schema, Semantic-Web.

1. Introduction and history

We are delighted to be invited to contribute to the new CODATA journal which is being launched at a critical time for the scientific world's data, information and knowledge. This article reflects a presentation at CODATA2000 (Baveno, Italy, 2000) by P-MR. It illustrates some of the ways forward in STM (Scientific, Technical and Medical) publishing and discusses how it could go beyond the traditional publication ("paper") in several respects:

- The technologies discussed here should be precisely those used for its authoring of and publication in this journal.
- There are several aspects of the publication process that we argue should be directly managed by the International Scientific Unions (ISUs), and possibly through their organ CODATA.
- It urges the creation of new types of scientific publication such as the "datument" (a seamless integration of data and document) and gives some indication of the technologies and infrastructure available to innovative authors and publishers

A global approach to information has been anticipated for centuries, such as in Diderot's Encyclopedia, and through visions such as that of Samuel Butler (Butler, 1863), who wrote in 1863

> *"I venture to suggest that ... the general development of the human race to be well and effectually completed when all men, in all places, without any loss of time, at a low rate of charge, are cognizant through their senses, of all that they desire to be cognizant of in all other places. ... This is the grand annihilation of time and place which we are all striving for"*

In the 20th century, global visions included Vannevar Bush's Memex (Bush, 1945), Licklider's Galactic Network (Licklider, 1962) and Garfield's ideal library or Informatorium (Garfield, 1962). In 1965 J. D. Bernal (Goldsmith, 1980) a crystallographer, took up this theme that information could be universal by urging us to:

> *"...get the best information in the minimum quantity in the shortest time, from the people who are producing the information to the people who want it, whether they know they want it or not" (our italics).*

Although the technology to bring these visions to pass did not yet exist, Bernal evangelised the creation of data depositories, which led in the 1970's to the convention that all crystallographic publications should be accompanied by "supplemental data". Journals accepted this practise and it has for some time usually been a precondition of publication. The practice and the technology has been strongly supported by the International Union of Crystallography (IUCr) which has also taken the lead in developing the protocols required. The primary mechanism has been the development of CIF - the Crystallographic Information File (Hall, Allen, Brown, 1991) - which is a self describing electronic format for crystallographic data.

CIF has been aimed at capturing the whole of the scientific experiment, which includes the raw data, the experimental details, the derived data (results), human-readable text, and the metadata associated with publication. Many crystallographic papers are published in the Union's journals, and Acta Crystallographica C accepts all papers (about 1-2 printed pages) directly in CIF format. The format is rich enough to allow for the following:

- Deposition of sufficient information to allow independent repetition of the experiment
- Automatic checking of internal inconsistencies in the data. It happens that crystallographic experiments are often overdetermined and so errors can often be detected in this way.
- Detection of "unusual" values. Misinterpretation of crystallographic data often leads to statistical outliers (e.g. unusual spacegroups, intramolecular geometry, etc.) Programs have been written to analyse CIFs and detect possible anomalies and the editors can bring these to the attention of authors.
- Direct typesetting of the "paper" from the electronic submission.
- Automatic deposition of the "data" in repositories such as the Cambridge Structural Database (Cambridge Crystallographic Database Centre, n.d.), and the Protein DataBank (Protein Data Bank, n.d.).

The IUCr system works very well and many thousands of papers have been printed and published in this way (International Union of Crystallography, n.d.). There is an overseeing committee, COMCIFS, which conducts much of its business electronically and one of us (P-MR) has been involved with this for many years. We have highlighted the role of IUCr/CIF because it has inspired some of the aspects of the present article.

In this article we argue that the STM (Scientific, Technical and Medical) community should adopt a publication process where conventional "documents" and "data" are seamlessly integrated into "datuments" (Rzepa, Murray-Rust, 2001a). This portmanteau neologism emphasises that the electronic object is a complete integration of conventional "human-readable" documents and machine-processable data. It must be emphasised that word-processor formats such as Word (or even TeX) and e-paper (such as PDF) create conventional documents that can normally only be usefully processed by humans. They do not produce datuments where machines can read and process the information in a semantically rich manner (we use "process" and avoid "understand" but this word may help to appreciate our emphasis). In passing we urge this journal to adopt and evangelise the publication of semantically-rich datuments!

2. The WWW and Markup Languages

The World-Wide Web arose from the need for high-energy physicists at CERN to communicate within a large dispersed community. Berners-Lee (Berners-Lee & Fischetti, 1999) pioneered this through the development in 1980 of an electronic notebook he called "Enquire-Within-Upon-Everything" that allowed links to be made between arbitrary nodes, and in 1989 he created a markup language (HTML) which could, *inter alia*, express such links (more precisely called URIs). Markup languages arise from document technology and have been in use since around 1969, one of the first having been the development of Generalized Markup Language (GML) by Charles Goldfarb at IBM (Cover, n.d.). The initial role was to provide typesetters with instructions on how to set the text (italics, paragraphs, etc) and was implemented by additional markup characters embedded in the running text. These characters could be recognised by

machines and used as *formatting* or *styling* instructions rather than being part of the actual content. Using HTML as an example:

```
<p>This starts a paragraph with some <i>embedded italics</i>.</p>
```

The "tags" in angle brackets are recognised by the processor as markup and used as instructions rather than content to produce the rendered sentence:

This starts a paragraph with some *embedded italics*.

Although this example will be familiar to many readers it is important, since it illustrates the critical importance of separating content from style (or form). The p tags precisely define a paragraph, a unit for structuring the document. A machine could now easily count the paragraphs in a document and the number of characters (but not words!) in each. HTML is enormously successfully because it is simple to create and extraordinarily useful. It does the following:

- Provides a (rather flexible) document structure (paragraphs, headers, tables, lists)
- Supports embedded images (and other multimedia)
- Supports human interactivity (through FORMS)
- Manages (flexible, non-robust) hypertext
- Supports programs through applets and plugins
- Support for metadata (META tag)
- Manages text, including some degree of formatting, styling and screen layout.

This is a very substantial list and we see HTML as a key component of the datument. However its success has generated many problems which HTML in its original form cannot solve:

- It can only support a fixed tagset (about 100) and even this number is regarded as unmanageable (no software yet implements this full set accurately and completely). Any other tags which might be present (*e.g.* <molecule>) are simply ignored. Specialised information in areas such as e.g. e-Commerce and STM cannot usually be encoded precisely or robustly.
- Much of the behaviour (semantics) is undefined. This has led to different manufacturers creating their proprietary methods of supporting functionality (*e.g.* through scripting languages or plugins).
- It is designed to be error-tolerant. Browsers may try to recover from non-conforming documents and may do so in different ways. Humans are good at recognising and often correcting errors in HTML (missing links, broken formatting, incomplete text). Machines cannot normally manage broken HTML and it can also be very difficult to interpret the proprietary dialects from some vendors
- Author provided metadata is often entirely absent. If present, it will likely adhere to a general form (schema) of limited utility in STM areas.
- The emphasis on style in many of the tags (fonts, colors, layout, etc.) has muddled the separation of content from style. The World-Wide Web Consortium (World Wide Web consortium, n.d.) has developed technologies (CSS or Cascading style sheets and XSL, or eXtensible stylesheet language) to overcome this, but as with HTML, CSS is incompletely implemented by most browser manufacturers at present. Moreover most commercial tools for authoring HTML emphasise presentation or interactivity (to capture the reader's attention) and produce HTML where the content is horribly subservient to the style (*i.e.* markup is often broken/badly formed).

HTML was constructed according to SGML rules (Standardised Generalised Markup Language, ISO-8879:1986). Confusingly named, SGML is not a markup language but a metalanguage for constructing markup languages. Such markup languages (MLs) are not new and SGML-derived MLs have been used in a number of vertical domains (especially publishing, aerospace, telecomms, petrochemical, and defence). Scientific publishing has developed MLs such as DOCBOOK (DocBook, n.d.) and ISO:12083 (Kennedy, 1999), designed to support scientific publications including abstracts, authorship, affiliations, textual content, images and citations. They are primarily used by technical editors who can manage the content at an appropriate level for in-house application. The information can be re-used; for example lists of authors and references can be compiled and used for checking or subsequent publications. The ML does not imply a particular style; thus references can be processed through stylesheets to provide the specific rendering for (say) volume number, author name, etc. If a different house style is required, a different stylesheet is used; the manuscript itself doesn't need altering. We have stressed this process, because it is usually opaque to most authors, who have to adopt a given style for a given journal. Indeed if they change the journal they publish in, it is usually their responsibility to change the style through the manuscripts. This is often resented and compliance can be poor!

These conventional markup approaches are inadequate for datuments as there is usually no domain-specific support. The W3C and Berners-Lee recognised the need for a next generation of markup to carry robust, precise technical data. One of their first efforts was MathML, a markup language for mathematics, since the alternative had been to generate non-scalable bitmapped images of mathematical symbols and equations, or to use fonts which not every reader had access to. We note in passing that the debate within the mathematical community as to whether MathML should primarily serve the needs of presentation or of content highlights the difficulty in achieving such separation. MathML was originally developed with SGML, but is now based on XML or eXtensible Markup Language (Murray-Rust & Rzepa, 1999; Murray-Rust & Rzepa, 2001a; Murray-Rust & Rzepa, 2001b) which we now describe.

SGML is very powerful but also highly complex; in fact most manufacturers could not implement all of it. XML has been designed to be simpler, easier to use, smaller and is a fully conforming subset of SGML (essentially "SGML-lite"). It allows new markup languages to be defined through Document Type Definitions (DTDs) or the more recent XML Schema formalism. A DTD specifies a set of rules (syntax, structure and vocabulary) to which a document *must* conform; those that do are said to be "valid". Schemas allow more precise constraints and allow the definition of data types (this is discussed in greater detail in a separate article on STMML, for which we provide a schema, Murray-Rust & Rzepa, 2002).

MathML nicely illustrates many of the points we need:

- Each domain of human information will require one or more markup languages; MathML fulfils most of the requirements for mathematics in STM.
- Many of these MLs will need specialist tools for authoring, processing and display. The W3C have incorporated MathML authoring and display into their Amaya browser/editor (Amaya supporting primarily the presentational rather than the content-based components).
- The resultant marked-up information will be machine-processable. Thus MathML can be fed to a symbolic algebra processor; it can be integrated, used to generate graphical display, etc.

The potential for machine-processing is enormous. We are therefore urging all domains to develop rich markup languages for primary publication of datuments. Because of the central importance of chemical and molecular information in STM and because we have already developed both the DTD and Schema approach, we shall use CML (Chemical Markup Langauge, Murray-Rust & Rzepa, 1999). in the examples. Syntactically, these examples could be replaced by any other modularized markup language (*e.g.* GML for geography, GAME for genomes, MAML for microarrays, HL7/XML for healthcare, CellML in biology, etc.)

For readers not familiar with XML syntax we illustrate its features with an example of chemistry in Chemical Markup Language (CML):

```
<cml:molecule id="m01" title="methanol" xmlns:cml="http://www.xml-
cml.org/schema/CMLcore">
  <cml:atomArray>
    <cml:atom id="o1" elementType="O" hydrogenCount="1"/>
    <cml:atom id="c1" elementType="C" hydrogenCount="3"/>
  </cml:atomArray>
  <cml:bondArray>
    <cml:bond atomRefs2="o1 c1" order="S"/>
  </cml:atomArray>
</cml:molecule>
```

This consists of a single *element*, `cml:molecule`, which contains two *child elements* `cml:atomArray` and `cml:bondArray`. `cml:atomArray` has two `cml:atom` children, `cml:bondArray` has one `cml:bond` child. The `cml:molecule` has three *attributes*, `id`, `title` and the *namespace* attribute `xmlns:cml`. The namespace attribute has predefined semantics; it asserts that all elements prefixed by `cml:` belong to the namespace identified by the *namepsace-URI* `http://www.xml-cml.org/schema/CML2/Core`. This namespace is owned by the creators of CML, who can therefore ensure that there are no name collisions with any other namespace both within the document and between document collections. *No other elements or attributes have XML-defined semantics* - all semantics are imposed by CML. Thus the CML Schema defines an enumeration (list) of allowed `elementType`s and defines their meaning and use.

There is no default way of "displaying" or "browsing" CML.. The information can be processed in many different ways. Among these could be:

- Calculation of 2- or 3-dimensional coordinates and interactive display on screen
- Calculation of molecular weight
- Calculation of molecular properties (*e.g.* through quantum mechanical methods)
- Use as a database query ("is CH_3-OH in this database?")

Note that the semantics cannot be deduced from inspecting examples. It must be formally defined (*e.g.* in an XML Schema or similar tool). Thus CML defines that the `atomRefs2` attribute contains two references to *id* attributes on `atom` elements.

3. Validation

Publishers provide human-readable "guidelines for authors" for document preparation, but non-compliance is common. There are usually no guidelines for data preparation! If an author deposits supplemental data, how does the publisher know it is "correct"? A key aspect of XML is that documents can be *validated* (Figure 1). For publishing purposes validation implies a contract between the author and the publisher, *which is machine-enforceable*. A Document Type Definition (DTD) or more recently a Schema (there are several approaches) formalises the syntax, vocabulary, document structure and (with Schemas) some of the semantics. The Schema is a set of machine-based rules to which a datument must conform. If it does not, it is the author's responsibility to edit it until it does. If it conforms, it is assumed that the author has complied with the publishers requirements.

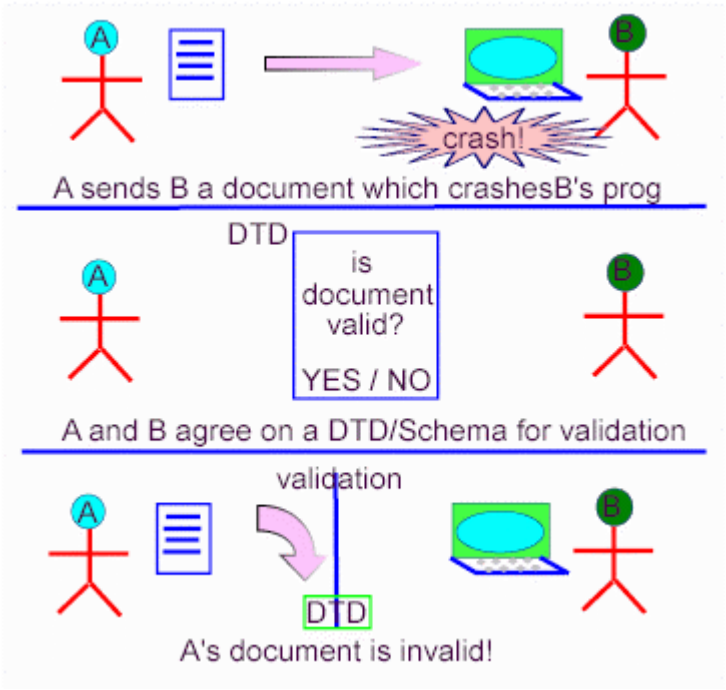

Figure 1. Document Validation.

Validation guarantees that the datument conforms to rules. The more powerful the rules, the more "invalid data" can be detected. Thus Schemas can allow the detection of *some* unallowed data, particularly with a controlled vocabulary. An atom in CML is not allowed an elementType of "CO" (presumably "Co"), or a hydrogenCount of -1. It is, however, allowed a formalCharge of "+20". This might be corrupted data, or a legitimate description of a highly ionized atom. Individual Schema-based rules (*e.g.* for different journals) could allow discrimination between these possibilities. We discuss Schemas in depth in a subsequent paper.

4. Importance of domain vocabularies

The construction of a DTD immediately emphasizes the need for a communal vocabulary. An element such as <molecule> or <organism; must be *processed* in exactly the same way regardless of the author, the reader or the processing software. We emphasize "processing"; the implementator must adhere to the same software specifications and the software must behave in a predictable manner. For many scientists this will require a change in their thinking, and we emphasize the consequences here:

In its strictest form this attitude is a *controlled vocabulary*. Only certain terms may be used and their meaning is specified by a trusted authority. An example is the use of "codes" developed by the World Health Organisation to describe morbidity and mortality via the International Classification of Disease or ICD-10 (World Health Organisation, 1992-1994). This dictionary, whose concept is over 100 years old, now lists about 10000 diseases and related concepts. Each is associated with a code (*e.g.* "cholera" in the 9th edition (World Health Organisation, 1978) has the unique code "001"; "Bitten or struck by crocodile or alligator, while in a sports or recreational area" maps to "W58.3" in the 10th edition).

Controlled vocabularies are widely used in certain areas of STM, especially where there is an emphasis on some or all of: safety, intellectual property including patents, regulatory processes, classification and indexing (*e.g.* in libraries), legal requirements including government, and commerce. They force the discipline to be mapped onto a generally agreed or mandated vocabulary, and often require substantial formal guidelines or training sessions to ensure consistency of interpretation. Thus many clinical trials use ICD codes as their basis for identifying indications or adverse drug events (safety).

Controlled vocabularies often create tensions in STM disciplines. Major reasons are:

- The vocabularies take time to develop and are often out-of-date (at least in parts) by time of publication.

- Many disciplines have a fluid use of terms and synonymity is a major problem. This often reflects different educational systems, and professional practice.

- A vocabulary is usually developed for a particular process. Thus the WHO codes were designed as statistical tools to collect disease data at a national level. The ninth edition (ICD-9) was independent modified to create ICD-9CM or "Clinical Modifications (National Center for Health Statistics, n/d) and is used for insurance coding in some countries.

- The science and technology moves faster than the vocabulary can support. Moreover differences of opinions in the content of the vocabulary, its structure, and its use occur. Thus the International Pharmaceutical Manufacturers Association has developed a vocabulary (MedDRA) for reporting disease codes in clinical trials and drug safety documents. Authors, publishers and readers are often confronted with a variety of controlled vocabularies, and find that all of them have drawbacks.

There are many vocabularies which are much less controlled, and which have a more fluid nature. Until recently most of these were periodically issued in printed book form by authorities such as the ISUs (*e.g.* IUPAC has nomenclature commissions which regularly produce the definitive names for molecules). Publishing houses produce dictionaries of science and technology, often in specific domains. Authors and publishers are often free to choose whichever vocabulary fits their concepts. Some dictionaries will discuss synonymity and provide for differences in interpretation but in general the vocabulary support is fluid and poorly defined.

Markup languages require us to have absolute precision in syntax, and structure. It is highly desirable to have additional precision in semantics (the meaning and behaviour of documents). The attachment of semantics to documents is not generally appreciated but is a critical process. Without semantics we have Humpty-Dumpty: <glory/> means 'a nice knock-down argument' (Carroll, 1872). Therefore we must have a formal means of attaching semantics to every XML element and attribute and their content. At present these are:

- **A human-readable prose description**. This can be as simple as a definition in a dictionary, which may or may not give an indication as to how it might be used. An example from CIF: *_chemical_compound_source* is defined as:

  > *'Description of the source of the compound under study, or of the parent molecule if a simple derivative is studied. This includes the place of discovery for minerals or the actual source of a natural product.'*

  This formalizes the concept, but (deliberately) gives wide latititude in its implementation and content.

- **A human-readable set of instructions for machine implementation**. Another CIF entry (abbreviated), for *_atom_site_U_iso_or_equiv*

```
Data type: numb (with optional s.u. in parentheses)
Enumeration range: 0.0 -> infinity
Units: A^2^ ( angstroms squared)
Definition
   Isotropic atomic displacement parameter, or equivalent isotropic
   atomic  displacement parameter, U(equiv), in angstroms squared,
   calculated from anisotropic atomic displacement  parameters.
   U(equiv) = (1/3) sum~i~[sum~j~(U^ij^ a*~i~ a*~j~ a~i~ a~j~)]
   a     = the real-space cell lengths
   a*    = the reciprocal-space cell lengths
   Ref: Fischer, R. X. & Tillmanns, E. (1988). Acta Cryst. C44, 775-776.
```

- This specifies carefully how the concept must be implemented. The constraints **data type**, **Enumeration range** and **Units** are all machine-processable. The definition includes an implementable algorithmic constraint. Since CIF predates XML this is only able and not machine-processable (*i.e.* it acts as a specification for a human programmer but it cannot be used to generate software automatically). XML Schemas provide mechanisms to overcome this.

- **defined by software**. Many elements of a controlled vocabulary are effectively defined by a software implementation. Thus the description of the HTML language requires certain elements to have specified behaviour. `<img>` supports the display of raster images but the precise look-and-feel may vary between implementations and file types. Implementation through software is useful and powerful where authors, publishers and readers/processors all use the same system. Because STM is increasingly multidisciplinary, this becomes problematic. Often a reader may have to download specialist software which is idiosyncratic and which may not have enough functionality, especially the export of semantically rich data. Moreover the semantic rules are often buried deep in the software and difficult to understand precisely - with binary executables this is usually impossible.

- **Formal semantics**. We believe that the STM community should move towards the adoption of formal rules for expressing semantics and ontology (semantics is the branch of semiotics, the philosophy or study of signs, that deals with meaning. Ontology is defined as a description, such as a formal specification of a program, of the concepts and relationships that can exist for an agent or a community of agents). The preferred W3C approach is the use of RDF syntax (World Wide Web Consortium, 1999), coupled with DAML+OIL for the ontology and inference layers (DARPA Agent Markup Language, n.d.). This is at an early stage and will probably take a few years to develop to a stage where it is largely accepted in the STM community. For example, it has taken about 5 years for XML to become recognised as the universal syntax for information. Ontologies are a more challenging concept. As an interim stage, therefore, we shall use XML Schema and XSLT (World Wide Web Consortium, n.d.) for implementing machine-processable vocabularies as far as possible. This will define the validation process, and also support transformations. In a following article we show this concept applied to general STM information. Behaviour will still need bespoke tools and we expect that maps, molecules and maths will all need purpose-built tools to create and display them.

Our central message is that we need carefully constructed and curated machine-processable ontologies. We believe that Scientific Unions and Learned Societies have a major role to play, and that openness and free access to ontologies is critical.

5. Dictionaries and Ontologies

Except for rigidly controlled vocabularies we believe it is best to use an abstract specification for the markup itself, and to add domain ontologies through separate (XML-based) dictionaries. Thus we would avoid:

```
<p>The compound had a <meltingPoint>23</meltingPoint></p>
```

and prefer something like:

```
<p>The compound had a <item dataType="float" title="melting point"
   dictRef="chem:mpt" units="units:c">23</item></p>
```

We use the abstract concept "item" to allow any data item to be marked up, and links to a specified dictionary to add the human-readable and machine-processable semantics. Indeed it is possible (and often desirable) to let the dictionary carry the dataType and units information.

This design is shown schematically in Figure 2;

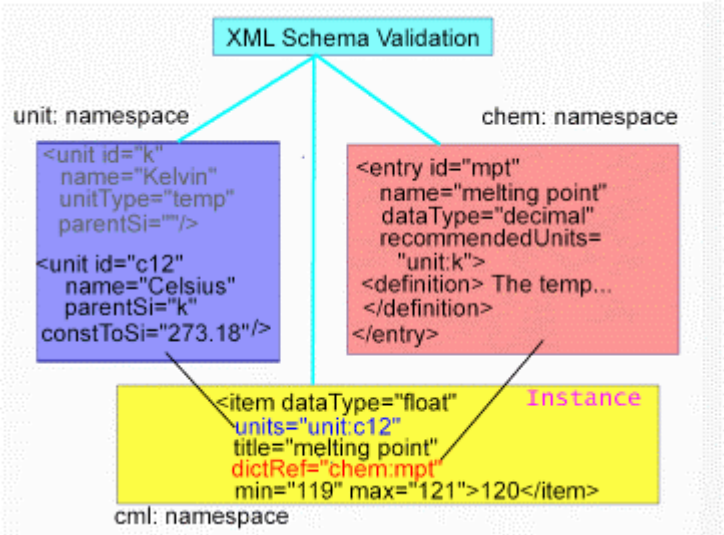

Figure 2. The use of controlled dictionaries in Schema-based validation.

The data are marked up using a simple generic ML described in a separate article on STMML (Murray-Rust & Rzepa, 2002). The ontology is provided as a set of dictionaries, in this case for the concepts themselves and the scientific units. There is no technical limit to the number of dictionaries or their content.

We have earlier published this multi-dictionary concept as a "hyperGlossary" (Murray-Rust, Leach & Rzepa, 1995; Murray-Rust & West, 1995). We collaborated with the W3C to develop an infrastructure in the W3C-LA program (W3C-LA program, n.d.). The technology has now advanced to a stage where such a concept can be easily implemented and, probably most importantly, where is value is recognised. The primary concepts are:

- **Each dictionary can be developed in a modular fashion**. The only common requirement is that the dictionaries should use the same markup elements.

- **Each dictionary should be uniquely identifiable**. Until recently this could require central curation but the acceptance of the namespace concept means that every organisation can maintain dictionaries without namespace collisions with other providers. They must, of course, make sure that their own dictionaries are individually distinct.

- **Every dictionary must have metadata**. This metadata should be for (a) discovery and (b) description of the contents and the means to assure trust. If dictionaries are modified, version numbers must be carefully described and curated. Past versions of dictionaries must be preserved so that the context of historical documents can be precisely conserved.

The STMML language has been developed to act as an infrastructure for a dictionary-based system. We believe that for a very large section of STM data (*i.e.* that does not require bespoke software) a dictionary-based approach can provide complete markup.

6. Authoring and authoring tools

At present most STM publications are created by authors in a publisher-specific manner (Figure 3);

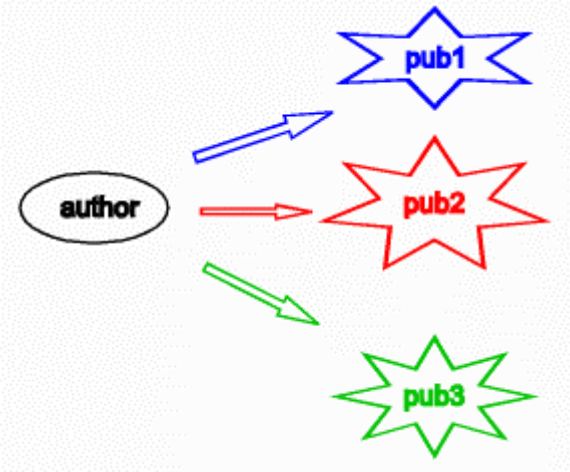

Figure 3. The traditional publishing process.

Each publisher requires:

- a particular document structure
- a particular technology (*e.g.* formats of text, images, references, and domain-specific data)
- a particular ontology (usually implicit)

The author has to change each of these according to the publisher's requirements, and independently of the content. The publisher (or author) then has to make significant technical edits, often as a result of author non-compliance. Author's data are transformed into text-oriented formatting languages for rendering to human-readable output, either paper or e-paper, and during this process the machine-processability is lost. Supplemental data is transmitted in a large variety of formats, often proprietary and binary. The archival value of these is very limited.

XML has the potential to revolutionize this if publishers and authors cooperate. With agreed XML-based markup languages authors can have a single environment independent of the publishers's requirements. Publishers can transform the XML into their in-house system. The original datument, *which contains all the "supplemental data"* can be archived *in toto* along with the semantics and ontology (all in XML). This is shown in Figure 4;

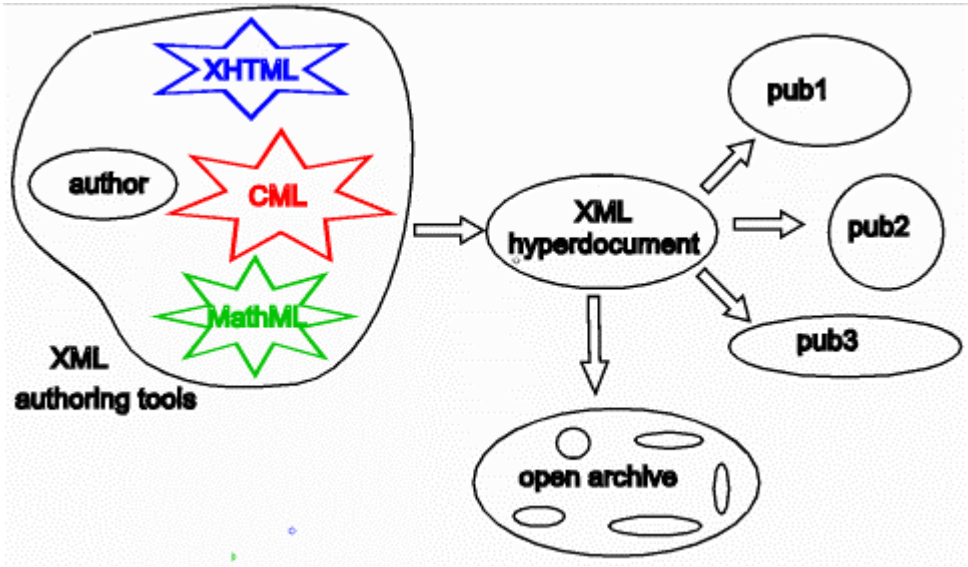

Figure 4. The publishing process based on XML "datument" processing.

This requires commitment from and cooperation in the community. There must be investment in a common toolkit and agreement on open ontologies. The publishing community has already invested in SGML and discovered its value, so the transition to XML should be relatively straightforward to implement. However a major change is required in authoring tools. Instead of proprietary text-based tools, with little useful support for semantics of either text of data, we require XML-based tools with domain-specific XML components. We have shown that this is technically possible; we discuss below the social factors required to make it happen.

A more detailed view of a potential architecture is shown in Figure 5;

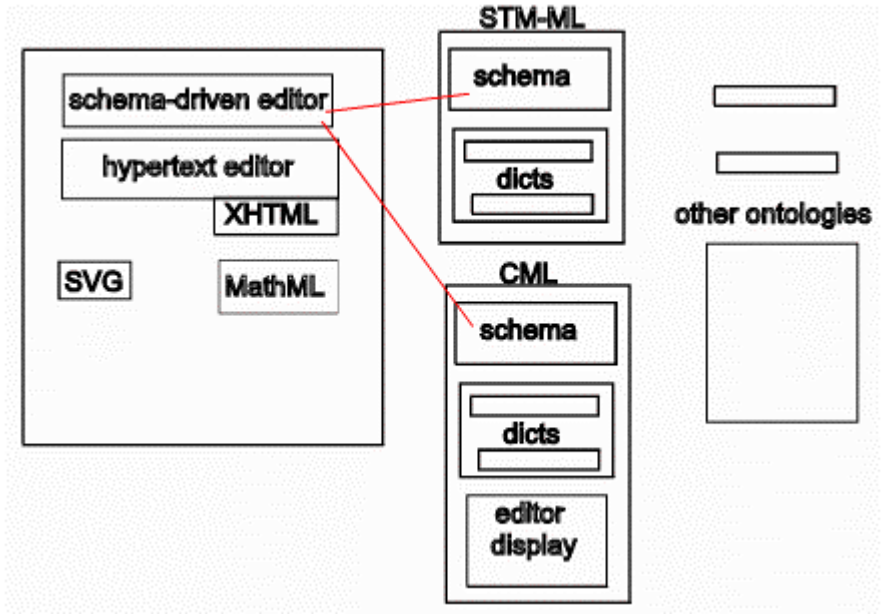

Figure 5. Schema-driven XML editing and display

This shows a generic XML-editor/display tool. It contains *generic* mechanisms to manage any domain-specific schema and therefore ensures that a resulting datument is valid. It will also contain generic mechanisms for supporting domain-specific software such as editors and browsers (*e.g.* for molecules,

maps, etc.). Hopefully it will also contain inbuilt support for W3C tools (World Wide Web Consortium, n.d.) such as MathML and SVG - the Amaya browser is a proof-of-concept implementation of such a tool.

7. The publication as a component of a database

Although the common use of XML will create significant savings (time for authors, staff costs for technical editors) this is not the major benefit. The new set of benefits are exactly those that Bernal foresaw, but which have waited until now for the technology to develop. The collected XML hyperpublications together with the ontologies effectively create a *machine-processable knowledge base for e.g. the STM domain*.

At present primary publications do not create knowledge without a lot of expensive additional human action, such as secondary publishing - abstracting, collating, validating, etc. While much knowledge will always have to be created by humans, XML publishing allows a significant proportion to be created by machine. If the metadata, structure, datatypes, ontology, semantics and processing behaviour of a piece of information are determined, it essentially becomes a self-describing *information component*. These information components - which might be implemented by a mixture of XML protocols and Object-Oriented code - can be regarded as standalone, self-describing, parts of a knowledge base. Protocols such as XML Query are able to search a heterogeneous aggregate of such components, and RDF will be able to make deductions from their metadata.

There are qualitative differences from the existing approaches such as relational databases (RDBs). It is extremely difficult to represent all the information within a publication as fine-grained components. Usually, therefore, publications are held as BLOBs (binary large objects), often in proprietary format and a subset of the information (*e.g.* authorship, citations, etc.) is extracted to serve as metadata. Moreover RDBs are expensive to install and maintain so that they are conceptually centralised, with a priesthood of designers and data managers. The author and the user have to work within a rigidly designed structure which is usually supported by bespoke tools and technology. It is not surprising that primary publications are not normally authored to an RDB schema!

XML, however, springs from a document-centric technology which allows considerable flexibility; SGML, and now XML, are the technologies of choice for publishers. We contend that most STM publication is now technically supportable by XML, and that by combination of different markup languages all information, even at a fine grained level, can be captured *without loss*. Any part of it can be retrieved, and hence a *collection of marked up XML publications constitutes a knowledge base*

If each datument has sufficient high-quality metadata there is no essential need for a knowledge base to be centralised. By collecting those publications of interest, any reader can create their own personal base, in effect what has become known as a peer-to-peer model. XML query and RDF, together with the markup-related software and ontologies allow querying of this collection. At present, of course, a brute-force query may be excessively expensive, but we can expect developments in intelligent indexing and query caching. In the near future we shall probably see RDBs used in conjunction with XML, perhaps to optimise the initial query and retrieve only those datuments worth searching by XML technologies. Since, however, these are general requirements from all domains we can expect rapid progress.

8. Shifts in publishing paradigms

The arrival XML technology coincides with changes in the purpose and means of STM publishing. Among the reasons for publication are:

- Communicating ideas to other scientists
- Formally recording and authenticating a scientific piece of work, including priority and Intellectual property rights/legal requirements
- Inviting peer-review and other comment
- Creating a "publication" for the author's career and justifying past/future funding

- Making data and procedures (recipes, software, etc.) publicly available for re-use and archival
- Developing a community of common interest
- Generating direct income

The major approach is still the "peer-reviewed paper" created through the offices of a scientific union, learned society or commercial publisher. Historically this arose because of the need to create and distribute printed pages. The publisher has gradually acquired other roles such as ownership of the intellectual property and management of this market. While much of this is beneficial the scientific community is showing increasing dissatisfaction with this model. A number of new initiatives have emerged which challenge the private ownership of datuments (Pubmedcentral, n.d., SPARC, n.d., ePrints Initiative, n.d., Open Archives Initiative, n.d., Public Library of Science, n.d.).

We argue that *technology is no longer the limiting factor which centralises the role of the publisher*. Given appropriate tools, an individual STM author can create a finished datument, requiring little or no technical editing. The same datument would be created whether it was destined for peer-review or for personal publication. XML stylesheets could allow different processing of this datument by different types of reader/user. This will allow the community to explore the social aspects of publishing without being constrained by technology.

How is this likely to come about? It will require targeted investment and the community has to recognise its value. In many disciplines (crystallography, genomes, synthetic chemistry, etc.) the data are seen of great *communal* value; i.e. the author wishes them to be re-used. However data are expensive to collect and (even with XML) expensive to maintain. Genomic data are (mostly) Open and freely available; crystallographic data are partly open and partly on a pay-per-use basis. Synthetic recipes in e-form are all on a subscription basis. The same variation will be found throughout the STM world. The more open the data, the more widely they are re-used and the greater involvement of the community in developing methods for creating tools.

Fortunately the design and implementation costs of tools are greatly reduced by XML. Since the infrastructure is commerce-driven, tools are generic, high-quality and low-cost. Domains therefore only have to implement a subset of the functionality. This is still a major commitment, but it is manageable.

CODATA, ISUs and learned societies have an opportunity and a responsibility in this field. They already possess much of the metadata and ontologies, but not in e-form. *The conversion of ontologies to e-form must be a critical activity*. They also have the role of coordinating infrastructure and ontologies within their domains, which does not apply *de facto* to commercial publishers. Indeed, if publishers within a domain indulge in *ontological competition*, the information infrastructure of the domain could be seriously undermined. If however collaboration, exemplified by the pioneering examples in *e.g.* crystallography, can be achieved, the future is bright indeed.

9. References

arXiv.org e-Print archive (n.d.) Homepage of the arXiv.org e-Prints. Available from: http://arXiv.org/

Berners-Lee, T. & Fischetti, M., (1999) *Weaving the Web: The Original Design and the Ultimate Destiny of the World-Wide Web*, London: Orion Business Books.

Butler, S., (1863). Quoted in Dyson, G., (1999) *Darwin Among the Machines*, London: Penguin Books.

Bush, V., (1945) As we may think, *Atlantic Monthly*, July. Retrieved April 22, 2002 from the World Wide Web: http://www.stanford.edu/class/history34q/readings/Bush/Bush_AsWeMayThink.html

Cambridge Crystallographic Database Centre (n.d.) Homepage of the CCDC. Available from: http://www.ccdc.cam.ac.uk/

Carroll, L., (1872) Chapter 6, *Through the Looking-Glass, and what Alice found there*, London: Macmillan & Co.

International SGML Users' Group (n.d) SGML Users' Group History. Retrieved April 28, 2000 from the *The XML Cover Page* http://www.oasis-open.org/cover/sgmlhist0.html

DARPA Agent Markup Language (n.d.) Homepage of DARPA Agent Markup Language. Available from: http://www.daml.org/

DocBook (n.d.) Home page for DocBook: The Definitive Guide. Retrieved April 15, 2002 from the World Wide Web: http://www.docbook.org/

ePrints Initiative (n.d.). Homepage of the eprints initiative. Available from: http://www.eprints.org/

Garfield, E. (1962) The Ideal Library - The Informatorium, *Current Contents*, June, 1. Retrieved April 15, 2002 from the World Wide Web: http://www.garfield.library.upenn.edu/essays/V1p001y1962-73.pdf

Gkoutos, G., Murray-Rust, P., Rzepa, H. S. & Wright, M., (2001) *J. Chem. Inf. Comp. Sci.*, *41*, 1124.

Goldsmith, M., (1980) *Sage, A Life of J. D. Bernal*, pp 219, London: Hutchinson.

Hall, S. R., Allen, F. H. & Brown, I. D., (1991) *Acta Cryst.*, *A47*, 655-685.

International Union of Crystallography (n.d.) Homepage of the IUCR. Available from: http://www.iucr.org/

Kennedy, D., (1999) ISO-12083. Retrieved April 15, 2002 from *XMLXperts* Web site: http://www.xmlxperts.com/12083.htm

Licklider, J. C. R. & Clark, W., (1962) as quoted by Leiner, B. M., Cerf, V. G., Clark, D. D., Kahn, R. E., Kleinrock, L., Lynch, D. C., Postel, J., Roberts, L. G. & Wolff, S. (2000), A Brief History of the Internet, Retrieved April 15, 2002 from the *Internet Society*: http://www.isoc.org/internet-history/brief.html#JCRL62

Murray-Rust, P., Leach, C. & Rzepa, H. S., (1995) Chemical Markup Language, *Abstr. Pap. Am. Chem. S.*, *210*, 40-COMP Part 1.

Murray-Rust, P. & Rzepa, H. S., (1999) Chemical markup Language and XML Part I. Basic principles. *J. Chem. Inf. Comp. Sci.*, *39*, 928.

Murray-Rust, P. & Rzepa, H. S., (2001a) Chemical Markup, XML and the World-Wide Web. Part II: Information Objects and the CMLDOM. *J. Chem. Inf. Comp. Sci.*, *41*, 1113.

Murray-Rust, P. & Rzepa, H. S., (2001b) Chemical Markup, XML and the World-Wide Web. Part III: Towards a signed semantic Chemical Web of Trust, *J. Chem. Inf. Comp. Sci.*, *41*, 1124.

Murray-Rust, P. & Rzepa, H. S. (2002) STMML. A Markup Language for Scientific, Technical and Medical Publishing, *Data Science*, submitted for publication.

Murray-Rust, P. & West, L. (1995), Terminology in a Global Context - The Virtual Hyperglossary, *J.International Cooperation in Terminology*, *2*, 34-38.

National Center for Health Statistics (n.d.) Homepage of the National Center for Health Statistics. Available from: http://www.cdc.gov/nchs/icd9.htm.

Open Archives Initiative (n.d.) Open Archives Initiative. Available from: http://www.openarchives.org/

Public Library of Science (n.d.) Homepage of the Public Library of Science. Available from: http://www.publiclibraryofscience.org/

Pubmedcentral (n.d.) Homepage of PubMedCentral. Available from: http://www.pubmedcentral.nih.gov/

Protein Data Bank (n.d.) Homepage of the PDB. Available from: http://www.rcsb.org/pdb/

Rzepa, H. S & Murray-Rust, P., (2001) A New Publishing Paradigm: STM Articles as part of the Semantic Web, *Learned Publishing*, 2001, *14*, 177.

World Wide Web Consortium (n.d.) *W3C-LA program*. Retrieved from *W3C Website*: http://www.w3.org/W3C-LA/

SPARC (n.d.) Homepage of the SPARC organisation. Available from: http://www.arl.org/sparc/

World Health Organisation (1978) *International Statistical Classification of Diseases and Related Health Problems*, 9th Revision, Geneva: World Health Organisation.

World Health Organisation (1992-1994) *International Statistical Classification of Diseases and Related Health Problems*, 10th Revision, Geneva: World Health Organisation.

World Wide Web Consortium (n.d.) Homepage of the World Wide Web Consortium. Available from: http://www.w3c.org/

World Wide Web Consortium (1999) *The RDF (Resource Description Framework) specifications*. Retrieved April 15, 2002 from the World Wide Web: http://www.w3c.org/RDF/.