FIS 2005 http://www.mdpi.org/fis2005/

Using XML for structuring the chemical information: Towards a chemical knowledge representation

Lecture dedicated to the memory of Jacques-Emile Dubois

Ana G. Maldonado, Bo Tao Fan* and Michel Petitjean

Laboratory ITODYS, University Paris 7 - Denis Diderot, CNRS UMR-7086, 1 rue Guy de la Brosse, 75005, Paris, France. e-mail: maldonad@itodys.jussieu.fr

* Author to whom correspondence should be addressed. Tel: 33-1-44 27 44 12, e-mail: fan@paris7.jussieu.fr.

Abstract: The management of chemical information has become a particularly complex problem with the apparition of combinatorial chemistry and the growing of chemical databases. Currently, different approaches are used to structure this information without following a standard language or framework. However, a normalized framework would allow better data exchanges between the different fields in chemistry research. It should be flexible enough to describe common properties used in the different chemistry areas, but detailed enough to be used as search keys in such systems. In this paper, a quite unknown approach (for the computational chemistry community) is introduced during the development of new HTS software for analyzing the molecular similarity and diversity. We use markup languages like XML to represent and structure the information. The principles of the knowledge management theory are described in order to show that our general problem comes to a chemical knowledge representation challenge. The benefits of such approach are shown through several examples of applications in chemistry.

Keywords: Information management, markup languages, chemical information, high throughput screening, knowledge representation, XML.

1. Introduction

The information in chemistry is a staggering deal. For a long time, chemists have developed their own languages (chemical nomenclature and structural representations) that add a unique dimension to chemical informatics. Chemical informatics techniques are challenged to create an integrated information environment in which all aspects of chemical research and development can be dealt with in a unified system. In most cases, chemical entities (like atoms) cannot be used as unique search keys in such systems. Physicochemical properties should be predicted or calculated with a high degree of accuracy in the design of the chemical informatics tools that will draw on the existing knowledge base of chemistry. Since a few decades, new techniques have appeared to enrich the "chemical panorama" and to address new sources of chemical diversity. One of these techniques is the well-known combinatorial chemistry.

The combinatorial chemistry (real and virtual) is nowadays a common key very useful to predict, synthesize and test large quantity of molecules in pharmaceutical and agrochemical discovery. Therefore, as a molecular diversity motor, the combinatorial chemistry is becoming very popular. A number of papers and reviews [1-5] cumulate the tremendous influence and progress in chemical and pharmaceutical industry, which implied the growth of parallel and automatic synthesis the last 20 years.

The problem is that each year, millions of compounds are added to the chemical databases (e.g. CAS [6]). Their structural, physicochemical and biological properties are coded and stored, generating more information. The organization, analysis, retrieval and management of this huge amount of data give new insights for further research areas involved in chemical informatics, as well as, high throughput screening (HTS) and data-mining [7].

The combinatorial chemistry, as well as other techniques for the generation of chemical diversity, leads to a growing need for managing and structuring chemical information. Universities, chemistry communities and global organizations such as IUPAC, CAS, etc. review and evaluate the current approaches, looking for easy, fast, efficient and cheap ways to solve this problem.

One of the propositions is to use the so-called *eXtensible Markup Language* (XML) [8], a language for easing the information exchange that proves to be a powerful alternative to conventional binary file storage or database management systems. Even if this markup language was, in principle, considered to be the universal format for structured documents on the web, it is now widely used for representing any structured data, and particularly scientific and hence chemistry data.

The first implementation of XML in chemistry is the Chemical Markup Language (CML) [9]. Its origin comes from the 1994 WWW conference, where Peter Murray-Rust and Henry S. Rzepa had the idea to integrate chemistry and mathematics in the framework of markup languages. Since this date, several prototypes and applications have been reported. CML handles molecular information, in form of an extensible and customable scope. It covers several chemical, medical and pharmaceutical disciplines (from macromolecular sequences to inorganic molecules and quantum chemistry). The growth of use of XML/CML is shown by the creation of new tools to address the chemical data: JMol, JSpect, JChemTidy [10], ChemDig [11], Chimeral [12], etc.

In this paper, we show the results obtained using markup languages as a basis for representing structured information contained in a molecule database. The optimal management of this database is decisive for the success of the high throughput screening software we are developing. We introduce our managing problem and we show the benefits of structuring the information using markup languages.

2. Principles of XML

The history of markup languages [13] began in the middle of the 60's, when IBM introduced for the first time GML (*Generalized Markup Language*). This language allows the user to format text documents and to define their type. A few years later, this work was standardized into SGML (*Standardized GML*) which gave the rules to *structure* text documents. SGML is a powerful language, initially made for text edition, but it turns out to be quite complicated to implement and to use, which has prevented its wide adoption in other communities. Consequently, the applications and tools are rare and not very popular (except in the business edition).

In the 90's, the initial idea of GML was resumed with the creation by Tim Berners-Lee of HTML (*HyperText Markup Language*). HTML is a very simple language which allows the *presentation* of web pages. As its precursor, it uses tags that are embedded inside the information, but unlike SGML, the set of these tags is fixed, closed and standardized. Its simplicity allowed quickly a widespread adoption and was a revolution in the exchange and the presentation of documents in the WWW. But, its principal drawback is that no structure of the information can be locked up in the tags since the language is only concerned by the presentation of the information. Later, on October 1994, Tim Berners-Lee founded the World Wide Web Consortium (W3C) at the Massachusetts Institute of Technology, with the mission of developing protocols and guidelines that ensure long-term growth for the WWW.

In 1998, the W3C recommended the XML [8] (*eXtensible Markup Language*) with the same goal: ***representing and structuring*** the data for better exchange and reuse of the information. XML is deliberately intermediate between SGML and HTML. It is simpler and less restricting than SGML, but more complex and more constraining than HTML. Contrary to HTML and like SGML, XML makes use of a language which allows the description of formats.

As described above, XML works with `<tags>` that can be defined freely to structure the information. There are always an `<open>` and a `</close>` tag. The data is then "encapsulated" inside the tags. The tags are called elements. XML elements are *extensible* (to carry out more information) and they have *relationships* (they are related as parents and children). When some pieces of information are edited in this way, it is then easily processable by a program or a user that knows the sense of these tags.

```
<element>
      <element1> ... </element1>
      ...
</element>
```

One can define as many tags as needed, without restriction. As a result, the information is then structured inside tree hierarchies. A simple example of XML code in chemistry is given below:

```
<molecule>
      <name> Hydrochloric acid </name>
      <atomList>
            <atom> H </atom>
            <atom> Cl </atom>
      </atomList>
</molecule>
```

In this example, the "father" element `<molecule>` could contain two (or more) "child" elements: `<name>` and `<atomsList>`. The `<atomsList>` element contains itself two new elements `<atom>`. We could choose another structure in function of our needs.

Moreover, each tag can have attribute/value pairs. Attributes are used to provide additional information about elements:

```
<element attribute1="value1" ... attributen="valuen"/>
```

Attributes often provide information that is specifically related to an element structuring the data. For instance, the `molecule CAS_number` (CAS: Chemical Abstract Service) could be relevant for a certain kind of application. It will therefore be represented as an attribute of the `<molecule>` element, but other information such that the `nature` or the `toxicity` could as well be attributes of this element. Attribute values are always enclosed in quotes `("")`.

```
<molecule CAS_number="7647-01-0"/>
```

It is also possible to add rules that control the order and nesting of the tags, as well as how they can be combined. This set of rules constitutes a DTD (Document Type Definition). The DTD is one of the building blocks of XML: it allows validating the XML document for further export or reuse. The XML Schema (the XML version of the DTD) and the namespaces (that avoid name collisions) are other forms of control. Query and transformations of XML documents are common tasks for which specific XML languages have been developed and that we briefly explained below.

The W3C has developed a wide range of generic protocols based on the XML syntax. We may loosely refer to these as the "XML family" or even "XML". Numerous applications were developed in all the areas of research since the apparition of XML in 1998. The more popular are related with science, multimedia and the semantic web.

In the figure 1, we illustrate this XML family according to the W3C standards, as well as some current applications. There are excellent web-tutorials [14] and books [15] of XML, even if the recommendation [8] is the reference document.

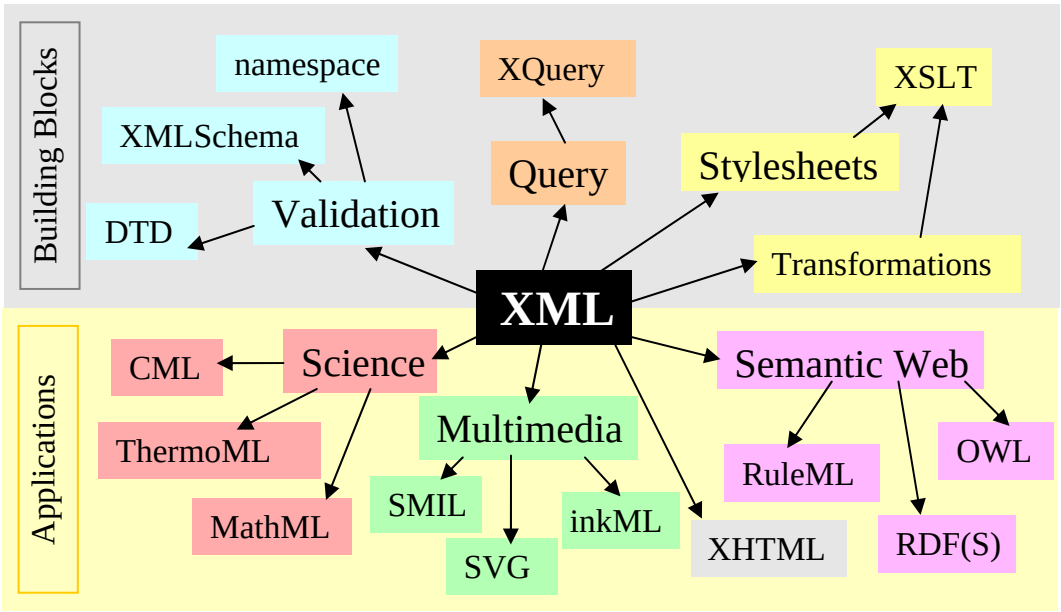

Figure 1. The XML family (adapted from [16])

We choose to detail the building blocks and the application areas of the figure 1 in the table 1. We summarize the objectives of these components with their extended name and we provide some references.

	Name	Extended Name	Allows to…	Ref.
	XML	Extended Markup Language	provide a common syntax and semantics for all computational processes and ease the exchange and reuse of information	[8]
Building Blocks	XMLSchema	Extensible Markup Language Schema	control the syntax of XML	[17]
	DTD	Document Type Definition	define the possible tags and attributes as well as how they can be combined	[18]
	Namespace	Namespace	put the vocabulary in a restricted space to avoid collision names.	[19]
	XQuery	Extended Query	make queries in XML	[20]
	XSLT	Extensible Stylesheet Language Transformation	transform the XML documents in other formats	[21]
Applications	MathML	Mathematical Markup Language	structure mathematical data	[22, 23]
	ThermoML	Thermodynamic Markup Language	structure thermodynamic data	[24]
	CML	Chemical Markup Language	structure chemical information	[9, 23, 25]
	SVG	Scalable Vector Graphics	draw vectorial graphics using XML	[26]
	InkML	Ink Extended Markup Language	format data input with an electronic pen or stylus as part of a multimodal system.	[27]
	SMIL	Synchronized Multimedia Integration Language	enable simple authoring of interactive multimedia presentations	[28]
	XHTML	Extended Hyper Text Markup Language	present and structure document contents in the web	[29]
	RDF	Resource Description Framework	graph model to describe web resources	[30]
	RuleML	Rule Markup Language	express first order logic rules to perform reasoning on data	[31]
	OWL	Web Ontology language	represent ontologies on the web	[32]

Table 1. Detailed applications and building blocks of XML

A lot of efforts have been done in all the areas of science to define standard schemas, vocabularies and ontologies. Some examples are shown in figure 1: MathML, ThermoML and CML. It is important to note that the construction of a markup language for chemistry was one of the priorities of the W3C working groups [13, 25]. This is an ongoing effort for the case of the Chemical Markup Language (CML) [9], which is an extensible base for chemically aware markup languages. CML represents a collaborative approach to tackling some of the problems of the interchange of chemical information over the Internet and other networks [33-36]. It allows the user to structure in a known framework the chemical information that could be extracted, analyzed, exchanged or visualized. Our actual research interest is on the high throughput screening (HTS), which involves the management of chemical information with the use of markup languages.

3. Structuring the chemical information

In most chemo-informatics projects, the management of chemical information is a big challenge. In our case, when we made the choice to use chemical informatics techniques, we also encountered the problem of designing the chemical databases. The format used, as well as the integration of this information in our in-house projects were priorities. But the translation in an easy and standard format

which allows exchange with other users or software is a crucial point that could determinate the life-time of any software.

In the next part, we will present the use of high throughput screening (HTS) as a solution to chemical database problems by using markup languages. We will then discuss the needs for representing the chemical knowledge as a long-term solution to the future chemical information challenges.

3.1 Context of the problem: High Throughput Screening

As it was pointed out in the introduction, the combinatorial chemistry is a powerful molecular diversity motor [37]. It allows the user to produce and to select compounds in a rational way, and to test their different biological activities. This approach is applied to generate big databases of hundreds of thousands of molecules. The predicted number of potential drug-like targets encourages the medicinal chemists to use molecular similarity and diversity techniques in drug design.

However, the management of these huge amounts of data needs the intervention of other technologies. The use of high throughput screening (virtual or real) in library design is the logical consequence of the uncontrolled use of combinatorial chemistry that generates an important volume of information.

In chemical library research, as a part of drug discovery, the cost-effectiveness considerations dictate that the libraries of molecules should be structurally as diverse as possible and should have a realistic size [38-40]. Several works have proposed solutions to these problems, and commercial chemical database management systems have appeared to address this issue: MDL ISIS Host [41], Daylight Database Package [42], CambridgeSoft ChemFinder [43], Oxford Molecular RS3 Discovery [44], Synopsys Accord [45], and Tripos UNITY [46]. But new approaches to manage chemical libraries are continuously proposed in the literature.

We have developed a new system based on the concept of molecular diversity [47]. This system integrates the structure of chemical information in the library management, with a more complex concept of similarity. In this context, we have developed a new approach to manage, reuse and describe our molecular database, as well as its associated information.

"Structurally similar molecules tend to have similar properties". This basic chemist belief principle called "similarity property principle" [48] states that structurally similar molecules are more likely to have resembling properties [49-50]. However, this principle is questioned by numerous experiences with contradictory results. A recent review [51] claims that the molecular similarity should be justified for every specific activity. The identification of the more informative representation of molecular structures is then of great importance in similarity and diversity studies. Another work [52] states that even if similar structures have generally similar activity, minor modifications can make molecules to lose their activities completely. The authors insist then on the importance of selecting comprehensive compound sets for testing, and on taking into account series of analogs to avoid this paradox.

It is known that molecular activity is the result of the interplay of a number of complex processes, which cannot be easily represented by a set of linear relationships (see figure 2). To better describe these processes, non-linear variable mapping can be used, where the properties are represented by a non-linear function of structural, topological and molecular descriptors [53-55].

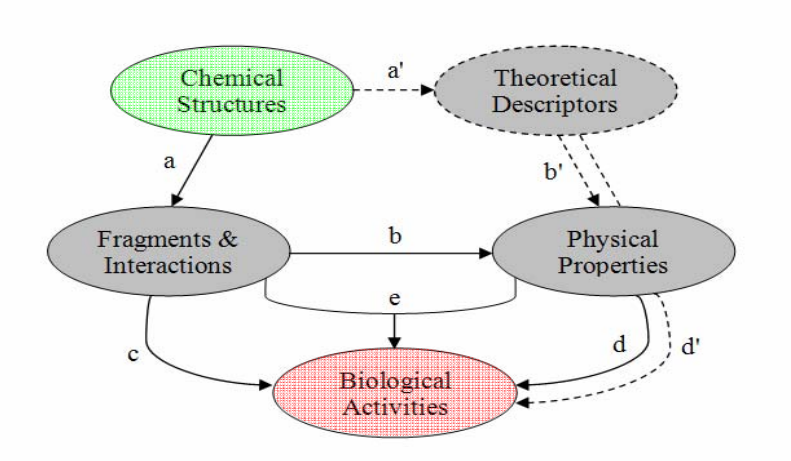

Figure. 2. Complex relationships between chemical structures, physical properties and biological activities. Grey circles denote descriptors. (a) structure fragmentation, (b-e) QSPR, QSAR and SAR methods which depend on data and descriptors. Dotted pale paths (a', b', d') indicates methods which frequently lead to ambiguous interpretations. Adapted from [54].

To take into account the complex relationships that explain the similarity between molecules, we think that the "similarity property principle" based exclusively in structural description of molecules is not enough. When addressing the problem of molecular diversity in chemical databases, we mixed structural descriptors to physicochemical properties expressed in a kind of molecular "pharmacophore". A new descriptor was formulated which allows the calculation of similarity/ diversity indexes between molecules or databases.

In any case, the descriptor computation uses complex structural databases. Their optimal computation will accelerate the screening of compounds and open a window to the management of chemical databases. When performing effective HTS, robust tools that can translate chemical information into suitable data are necessary. The treatment of chemical information of the test databases and of the existing data of our in-house programs is a determinant step of the HTS process (figure 3).

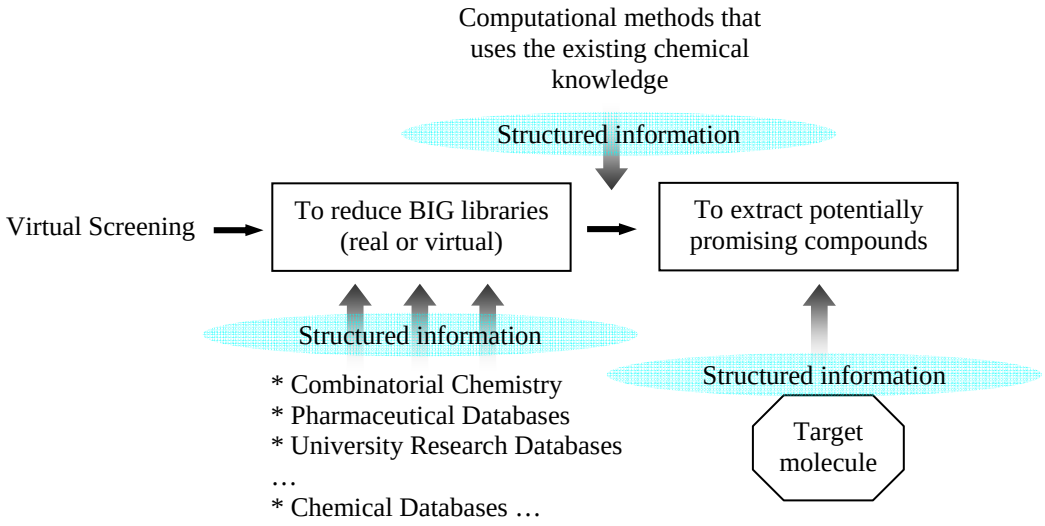

Figure 3. A Virtual High Throughput Screening process.

3.2 Treatment of chemical information

We have pointed out the goal of HTS in database management. We explain now, how in the context of a HTS similarity/diversity program, the search of a correct structure for our data allows us to find a solution for most screening process.

Our problem deals with the complexity of the data to be managed. When we define a diversity index as a function of structure vectors, we have to take into account the data attached to this vectors (physicochemical properties, position, distances, etc.) As a simplified model of a molecule, these descriptors have to be treated with the less loss of information, with the risk of introducing some mistakes in the analysis results. The direct comparison of vectors is interpreted as a measure of similarity or diversity between a target molecule (or group of molecules) and a database. The presence of big databases in our future analysis adds an extra variable: the speed of computation. In conclusion, we are looking for a powerful alternative to conventional binary file storage traditionally used in information exchange.

For the design of our chemical databases, we look for a language or a system which regroups the following characteristics:

- Complex molecule analysis power (capacity of "understand" chemical human-only concepts as: ions, aromaticity, tautomers, stereochemistry, etc.)
- Import / Export format available
- Compact information format
- Support of a wide variety of applications
- Easy to use and reuse (structure of the data, exchange, etc.)
- Fast automation in screening and analysis
- Open and extensible framework

Some of the available possibilities to structure the data, such as text format, conventional relational databases management systems (RDBMS), popular chemical molecular coding/representation (linear codes, binary codes, graphs, etc), in-house data structures or file formats, etc. were explored.

None of these approaches satisfies the primary goals. Certainly, combinations of them bring good results; but in general, they are limited to one molecular class or to small databases. Restrictions of biochemical/drugs applications include molecular models for toxicity, activity, permeability, etc. Other drawback is that the structural approaches are neither extensible, nor customizable.

For our particular case, we use common pre-treatment of the data like clustering (in function of the structure and other additional information) and format of filenames. In a first time, simple tabulated text files were used to describe the molecular information contained in a molecular database of .mol files. The second choice was given to the relational database systems (RDBMS): easier for the data edition, but not less constraining. The results of these two possible approaches are summarized in the Table 2.

Description / structure of data	Text files	RDBMS	Ideal data structure
Data Source	Mol file	Mol file + physicochemical properties	Any format
Data format size	Small	Quite big, if the data are structured and taking into account the internal structure of the database	Small
Data Pre-treatment	Structural clustering and filename coding	Structural clustering and filename coding	Not necessary
Exchange/ Reuse the data	No	Yes	Yes
Screening the data	-Inclusion of key words to allow the search. -Screening implemented by the user. -Creation of an algorithm for each screening -No possible automatic analysis if the format is modified	-Automatic if use commercial packages. -Need to be implemented by the user if not. -Reuse of the algorithms. -Possible automatic analysis if the format is changed in certain cases.	-Automatic and fast -Easy to implement and to reuse.
Query the data	-Need of key words -Difficult to implement	-Automatic if use commercial packages. -Possibility to reuse the results of the query.	-Automatic and fast -Possibility to reuse the results.
Application Implementation	Difficult. We use C as the implementation language	Easy.	Possibility to parse the data format in most languages: C#, java, etc
Conclusions	Incapacity in the data format to distinguish between the useful and non-useful information. Level of structure of data = 0	Possibility to structure the information at a high level but with the drawback of the growth of size and complexity of files.	The information should be structured in a high level to optimize the screening and query of the data, with as less as possible pre-treatment

Table 2. Some examples of structure/description implementation.

When using text files to describe the data, the main problem encountered was the incapacity of the algorithms to distinguish between the useful and non-useful information. Traditionally, the file formats in chemistry research are designed in an easy readable way. But with time, more information is added, and it is necessary to "clean" the files and choose the correct information for the correct computation. The presence of "research keys" is then desirable, to retrieve quickly and easily the desired data. In the best case, this kind of information structuration will bring the program to identify the desired data.

In a database oriented case, screening and queries are possible in certain formats and under certain conditions. Most RDBMS allow the user to save the queries and reuse it. But the structures are fixed and non extensible. If the data is modified (by addition of new properties or by reordering of its components) a re-edition is necessary, often manually. The export of the database is generally easy and they are numerous possibilities: text files (.txt, .csv, .tab), enriched text (rtf.), .xls, .mdb, .dbf, .db, .xml, etc. In certain cases (OLE objects included in the DB, imported DB from other formats, etc.), this step is tricky and data can be lost. Moreover, the generated files are quite big and their elements not modulable.

One alternative to conventional formats or approaches is the use of markup languages to structure the information contained in a database. In particular, the extended markup language (XML) presented in

section 2 is now considered to be the universal format for structured documents on the web. We think that XML offers a powerful and extensible mechanism for handling the chemical information. Several reasons support this affirmation: the XML protocols are all public and many of the tools are open source; XML have a modular approach easily adaptable to chemistry; the XML family has closed interoperability with other informatics standard, etc. (see figure 4). We have already shown in section 2 the mechanisms underlying XML and given some examples for the application to chemical information. We detail in the next section the application of markup languages to a chemical database in the framework of a HTS system. We demonstrate why XML is particularly relevant for our problem and for chemical database management in general.

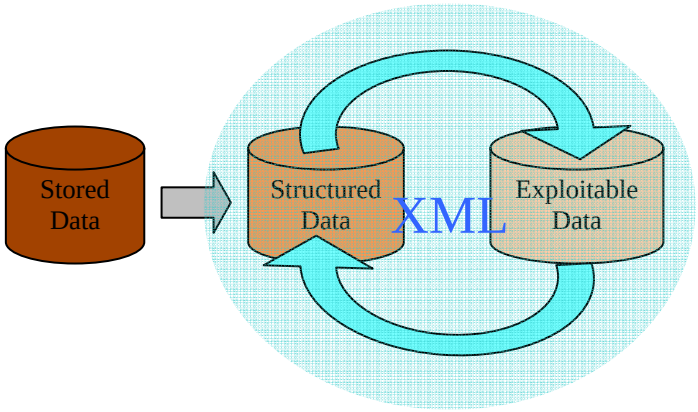

Figure 4. The XML cycle for structuring-exploiting the data

3.3 XML for structuring the information

The implementation of XML in a chemical information framework is a successful story made in part by the creation of CML and the adaptation of companies and universities to the markup languages as their information exchange format. When designing our chemical database in the framework of a HTS system, it was not necessary to think in advance to all the possibilities of future implementations of the data. The open and extensible framework provided by XML allows the extensions and modifications of the database structure quite easily and costless.

In our context, we were interested in three kinds of data: molecular structures, physicochemical properties and human-related concepts. We decided to modulate these data using the father-child elements and the attribute/value pairs, as shown in section 2. We deliberately used elements and attributes of CML as `<molecule>`, `<atom>`, `<formula>`, `name`, `id`, etc. A complete list of CML elements, attributes and types is available in [35, 55].

Information packaged into modules has considerable and numerous benefits. Some of them are enumerated below:

- Each module can be viewed as a set of reusable components.
- Complex data can be analyzed as a set of non-interacting parts which will simplify the searches.
- The need for understanding the context disappears.

For these reasons, we organize our data following an intuitive classification. Firstly, we assign to each kind of data file an element `<molecule>`. Each molecule has child elements, for example: `<formula>, <property>, <key>.` The rest of the data is structured along the descendants, for example: `connectivity table, atom sum`, etc. following a tree organization.

The set of molecular files creates a structured molecular `<index>` ready to be query, transformed and exchanged:

```
<?xml version="1.0" encoding="iso-8859-1" standalone="no" ?>

<!-- Data Structure for mol files --!>
<index>
         <molecule name="water-1a.mol"/>
            <formula>
                  <ConnecTable>table-1a.skc</ConnecTable>
                  <AtomSum>3</AtomSum>
                  ...
            </formula>
            <property>
                  <HBondDonnor>0</HBondDonnor>
                  <PotPCharged>1</PotPCharged>
                  ...
            </property>
            <key>
                  <ID>1a</ID>
                  <Aromatic>0</Aromatic>
                  ...
            </key>
        </molecule>
        ...
</index>
```

The construction of a DTD is the second necessary step if we want to validate this XML format. The DTD for this sample XML file could be:

```
<!-- Document Type Definition for the data structure proposed --!>
<!ELEMENT index (molecule+)>

<!ELEMENT molecule (formula,property,key)>
<!ATTLIST molecule name CDATA #REQUIRED>

<!ELEMENT formula (ConnecTable,AtomSum)>
<!ELEMENT ConnecTable (#PCDATA)>
<!ELEMENT AtomSum (#PCDATA)>

<!ELEMENT Property (HBondDonnor,PotPCharged)>
<!ELEMENT HBondDonnor(#PCDATA)>
<!ELEMENT PotPCharged (#PCDATA)>

<!ELEMENT key (ID,Aromatic)>
<!ELEMENT ID (#PCDATA)>
<!ELEMENT Aromatic (#PCDATA)>
```

Once the data is modulated and structured, we can generate automatically the XML document which contains an index of all the molecules, from the information contained in the mol file and using a simple C program. Consequently, our HTS system uses XML as a convenient way to exchange the

input/output information and to retrieve the different kinds of data necessary to the application: molecular connectivity tables, physicochemical properties, intermediate data structures and storage of the final results.

However, not all the elements can be generated automatically, especially those concerned by the human-related chemical concepts. Another drawback is that the XML document generated is quite large, because of the verbose characteristic of the language (130 molecules takes about 19 pages of text only taking into account the key element). The generated document can be readable using any web browser.

Until now, our examples do not include the structuration of the atomic information (atoms, bonds). This information is currently contained in the .mol document. Our approach is then a mixture between conventional approaches (connectivity tables) and innovative approaches (markup languages). To implement this mixture, it was necessary to “paste” the data structures with scripts that adapt the old structures into the new ones.

The reason why XML is not directly used to process the data (we do not compute anything with XML) is simply that, in principle, XML is not suitable for such a task. The data is first parsed and then transformed into internal data structures before being processed by the application. When allowing the data to be easily marked-up and by creating namespaces, XML becomes a perfect way for data exchange, retrieve and import/export processes.

Certainly, 100% XML chemical data exists. In this case, the information normally contained in a connectivity table (atoms, coordinates and bonds) is expressed in form of CML elements. The element `<formula>` will show the entire connectivity of atoms and bonds, and most of the `<property>` elements could be automatically calculated from the atomic information. The transformation of connectivity tables (as well as other older formats: pdb, xyz, skc, etc.) in XML files, is now possible with a minimum loss of information. CML structured chemical data is used for molecular representation (2D and 3D), robotic capture of information, data exchange, etc. XML computing tests in chemistry have been equally done (for example a "black-box" approach to computational chemistry and physics have been proposed [9, 55]).

The export and import process of chemical information is made through the use of XSLT (Extensible Stylesheet Language Transformation). XSLT make an automatic translation of XML files, respecting the DTD and the namespaces. We can then query our data and exchange molecular information without changing our original data structuration. In this way, the database could be augmented automatically to take into account new molecular discoveries.

Usually, data files have similar contents, but they differ from data dictionaries and conventions, so that they are not compatible with each other. The exchange of chemical information is seriously handicapped by this fact. The reuse and extraction of knowledge from scientific documents, reports and web documents, in an automatic way, is almost inexistent. All these needs have been already claimed by W3C working groups and associations [56] that support today the standardization of such XML applications in sciences. A summary of the advantages-drawbacks of the XML implementation in science is shown in table 3:

Advantages	(+) Meta language adapted for each scientific domain (mathematics, chemistry, medicine). (+) The structure of the information is open and flexible. (+) The queries/screening of DB is simplified and can be done automatically. (+) "Intelligent" searches are possible. (+) Numerous tools can parse it (C library, JAVA, Perl, etc.).
Drawbacks	(–) Verbose language (the files are long because the syntax is heavy: it is necessary to write a lot, even to express simple things).

Table 3. XML implementation advantages and drawbacks.

Currently, several commercial and university chemical databases are adapting the XML/CML (or a compatible format): CAS [6], NIST [57], Cambridge University [58-59], U.S. Governmental global agencies and non-profit societies such as Drug Regulatory Authorities (DRA) [60], the National Cancer Institute (NCI) [61], the Protein Data Bank (PDB) [62], pharmaceutical companies, etc.

3.4 Towards a knowledge representation

We have seen that markup languages optimize the data structure and allow fast and easy automation in screening and analysis process. The chemical information is then "tagged" and the concepts of "atom" or "molecule" become processable by a machine.

The problem is that the machines still do not have access to the meaning of the manipulated information. Knowledge representation allows expressing machine understandable information. Usually, the formalism is based on logical languages that allow modeling some ontologies that conceptualize the knowledge of the domain. In this context, the term ontology refers to a machine readable set of definitions that creates taxonomy of classes, relationships between them, and logical axioms [32]. There is a strong need in chemistry to have ontologies that cover the most relevant chemical information.

Currently, a standard chemical ontology is not yet available, and common efforts have to be done between companies, publishers, scientifics and associations to construct a generic and extensive chemical ontology which allow us to transform the current *document/information system* in a *knowledge-representation system.*

It is important to remark that the uses of markup languages are not restricted to molecular information management. It is applicable to all aspects of chemical informatics, data-handling and publication: data files and publications structure, molecular format transformation, log files from computational chemistry, peer to peer systems, instrumental output, etc. Conversion of this data to a k*nowledge-oriented system* will have a dramatic effect on the processing, searching, maintenance and re-use of chemical information.

4. Conclusion

The management of the growing chemical information has become a particularly hard problem since the apparition of combinatorial chemistry. Chemical information lacks of a standard structure which allows exchanges in different fields of chemical research. We argue that the use of non-official standard for chemistry data, makes more difficult the exchange, retrieve, reuse and export/import of chemical information. XML offers the possibility to ease these essential steps in the discovery process.

In this work, we have proposed to represent the structured chemical information used by our in-house HTS software as a merge of conventional file storage (text file) and markup languages (XML file). XML proves to be a convenient way to retrieve the different kinds of data necessary to the application: molecular connectivity tables, physicochemical properties, intermediate data structures and storage of the final results. The current limitations of XML have been pointed out, as well as, the need for a formal chemical ontology.

A prominent position of markup languages in chemistry is given by the well-known Chemical Markup Language (CML) developed by Peter Muray Rust and Henry Rzepa. The transformation of our chemical data in a 100% XML/CML structured data could be then interesting for future displaying and managing of molecular information (using CML and Jumbo Software [9] for example). This transformation should include translation of .mol files (connectivity tables) in CML, normalization of the current molecular information and physicochemical properties to this format, and to finish, integration of the data into a single document.

Acknowledgment

The authors would like to thank Raphaël Troncy for the early review of this paper and the fruitful discussions that followed.

References

[1] Blaney, J.M. and Martin, E.J., Computational approaches for combinatorial library design and molecular diversity analysis, *Curr. Opin. Chem. Biol.*, **1997**, 1, 54-59.
[2] Walter, H.M., Combinatorial Chemistry: a "Molecular Diversity Space" Odyssey Approaches 2001, *Pharmaceutical News*, **1996**, 3, 23-26.
[3] Michael, R.P., Tomi, K.S. and Walter, H.M., The Generation of Molecular Diversity, *BioMed. Chem. Lett.*, **1993**, 3, 387-396.
[4] Stu Borman, The many faces of combinatorial chemistry, *Chem. Engin. News*, **2003**, 81, 45-56.
[5] Willett, P., Using Computational Tools to Analyze Molecular Diversity, In DeWitt, H., Czarnik, A.W. (Eds.) Combinatorial Chemistry; A Short Course, American Chemical Society Books, Washington DC, 1997.
[6] Chemical abstract service. http://www.cas.org/
[7] Bajorath, J., Virtual Screening in drug discovery: Methods, expectations and reality. Available at the URL: http://www.currentdrugdiscovery.com
[8] Extended Markup Language (XML) 1.0, W3C Recommendation, 04 February 2004. http://www.w3.org/TR/REC-xml
[9] Chemical Markup Language (CML). http://www.xml-cml.org
[10] Georgios V. Gkoutos, Philip R. Kenway, Henry S. Rzepa. JChemTidy : a tool for converting web document collections to an XHTML representation. *J. Chem. Inf. Comp. Sci*. **2001**, 41, 253-258.
[11] Georgios V. Gkoutos, Christopher Leach, Henry S. Rzepa. ChemDig : new approaches to chemically significant indexing and searching of distributed web collections. *New. J. Chem*. **2002**, 26, 656-666.
[12] Peter Murray-Rust, Henry S. Rzepa, Michael Wright, Stephen Zara. A universal approach to web based chemistry using XML and CML. *Chem. Commun*. **2000**, 1471-1472.
[13] Peter Murray-Rust, Henry S. Rzepa. Markup Languages – How to Structure Chemistry-Related Documents. *Chemistry International*, **2002** 4, 24-34.
[14] For an XML tutorial see : http://www.w3schools.com/xml/xml_whatis.asp
[15] Elliotte Rusty Harold, XML Bible (2nd Edition), Wiley Eds. 2 edition, 2001.

[16] Harold Bolev, Stefan Decker, Michael Sintek. *Tutorial on Knowledge Markup and Semantic Resources. IJCAI-01 (International Joint Conference on Artificial Intelligence)* Seattle, 6 August 2001.
[17] The World Wide Web Consortium (W3C). http://www.w3c.org
[18] Document Type Definitions. http://www.xmlfiles.com/dtd/
[19] For a namespace description see: http://www.w3.org/TR/REC-xml-names/
[20] XQuery 1.0: An XML Query Language. W3C Working Draft 11 February 2005. http://www.w3.org/TR/xquery/
[21] XSL Transformations (XSLT) 1.0, W3C Recommendation 16 November 1999. http://www.w3.org/TR/xslt
[22] Mathematical Markup Language (MathML). http://www.w3.org/Math/
[23] Peter Murray-Rust, Henry S. Rzepa. Scientific Publications in XML – Towards a global knowledge base. *Data Science*, **2002**, 1, 84-98.
[24] ThermoML an XML-based approach for storage and exchange of experimental thermophysical and thermochemical property data, *J. Chem. Eng. Data.* **2003**, 48, 1.
[25] Georgios V. Gkoutos, Peter Murray-Rust, Henry S. Rzepa, et al. The application of XML Languages for Integrating Molecular Resources. *Internet J. Chem*. **2001**, article 6.
[26] Scalable Vector Graphics. http://www.w3.org/graphics/SVG
[27] Ink Extended Markup Language. http://www.w3.org/2002/mmi/ink
[28] Synchronized Multimedia Integration Language. http://www.w3.org/AudioVideo/
[29] XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition). A Reformulation of HTML 4 in XML 1.0. W3C Recommendation 26 January 2000, revised 1 August 2002. http://www.w3.org/TR/xhtml1/
[30] Resource Description Framework. http://www.w3.org/RDF/
[31] Rule Markup Language. http://www.ruleml.org/
[32] Web Ontology language. http://www.w3.org/2004/OWL/
[33] Peter Murray-Rust, Henry S. Rzepa. Chemical Markup, XML and the Wold Wide Web. 1. Basic Principles. *J. Chem. Inf. Comp. Sci.*, **1999**, 39, 928-942.
[34] Peter Murray-Rust, Henry S. Rzepa. Chemical Markup, XML and the Wold Wide Web. 2. Information Objects and the CML-DOM. *J. Chem. Inf. Comp. Sci*. **2001**, 41, 1113-1123.
[35] Peter Murray-Rust, Henry S. Rzepa.Chemical Markup, XML and the Wold Wide Web. 3. Toward a signed Semantic Chemical Web of Trust. *J. Chem. Inf. Comp. Sci.* **2001**, 41, 1124-1130.
[36] Peter Murray-Rust, Henry S. Rzepa. Chemical Markup, XML and the Wold Wide Web. 4. CML Schema. *J. Chem. Inf. Comp. Sci.* **2003**, 43, 757-772.
[37] Bayada, D.M., Hamersma, H. and Van Geerestein, V.J., Molecular Diversity and Representativity in Chemical Databases, *J. Chem. Inf. Comput. Sci.* **1999**, 39 1-10.
[38] Robert, S.P. and Smith, K.M., Novel Software tools for Chemical Diversity, *Perspectiv. Drug Disc. Design*, **1998**, 9/10/11, 339-353.
[39] Pearlman, R.S., Novel Software Tools for addressing Chemical Diversity, *Network Science.* **1999**. Available at the following URL: http://www.netsci.org/Science/Combichem/feature08.html
[40] Martin, E. and Wong, A., Sensitivity analysis and other improvements to tailored combinatorial library design, *J. Chem. Inf. Comput. Sci.*, **2000,** 40, 215-220.
[41] MDL Information Systems, Inc., 14600 Catalina Street, San Leandro, CA 94577, USA. For more information see the URL: http://www.mdli.com
[42] Daylight Chemical Information Systems, Inc., 441 Greg Avenue, Santa Fe, NM 87501, USA. For more information see the URL: http://www.daylight.com
[43] CambridgeSoft Corporation, 100 Cambridge Park Drive, Cambridge, MA 02140, USA. For more information see the URL: http://www.camsoft.com
[44] Oxford Molecular Ltd. Medawar Centre, Oxford Science Park, Sandford-on-Thames, Oxford, OX4 4GA, UK. For more information see the URL: http://www.oxmol-.co.uk/

[45] Synopsys Scientific Systems Ltd. 175 Woodhouse Lane, Leeds, LS2 3AR, UK. For more information see the URL: http://www.synopsys.co.uk/
[46] Tripos, Inc., 1699 South Hanley Rd. St. Louis, Missouri, 63144, USA. Information available at the following URL: http://www.tripos.com/
[47] Ana G. Maldonado, J.P. Doucet, Michel Petitjean, Bo-Tao Fan, Molecular Similarity and Diversity in Chemoinformatics: from theory to applications. *Molecular Diversity. In Revision.*
[48] Martin, Y.C., Kofron, J.L. and Traphagen, L.M. Do structurally similar molecules have similar biological activity?, *J. Med. Chem*., **2002**, 45, 4350-4358.
[49] Johnson, M.A. and Maggiora, G.M. (Eds.) Concepts and applications of Molecular Similarity, Wiley & Sons, New York, 1990.
[50] Walters, W.P., Virtual Screening - An Overview, *Drug Discovery Today*, **1998**, 3, 160-178.
[51] Patterson, D., Cramer, R.D, Allan, M.F., Robert, D.C. and Laurence, E.W., Neighborhood Behavior: A Useful Concept for Validation of "Molecular Diversity" Descriptors, *J. Med. Chem*., **1996**, 39, 3049-3059.
[52] Nikolova, N. and Jaworska, J., Approaches to Measure Chemical Similarity - a Review, *QSAR Comb. Sci.*, **2003**, 22, 1006-1026.
[53] Gorse, D., Rees, A., Kaczorek, M. and Lahana, R., Molecular Diversity and its analysis, *Drug Disc. Today*, **1999**, 4, 257-264.
[54] Japertas, P., Didziapetris, R. and Petrauskas, A., Fragmental Methods in the design of new compounds. Applications of the Advanced Algorithm Builder, *Quant. Struc.-Act. Relat*., **2002**, 21, 23-37.
[55] CML components available at: http://wwmm.ch.cam.ac.uk/moin/CmlSchemaComponents
[56] Davies, XML in chemistry, *Spectroscopy Europe*, **2002**, 14/1 and A. N. Davies, XML in chemistry, *Chemistry International*, **2002**, 24(4), 1-9.
[57] National Institute of Standards and Technology. http://www.nist.gov/
[58] Chemistry in Cambridge University. http://www.ch.cam.ac.uk/c2k/
[59] Colloquia of Chemical Laboratory. Cambridge University. http://www.ch.cam.ac.uk/today/
[60] Drug Regulatory Authorities. http://www.fip.org/resources/regulatory_authorities.htm
[61] National Cancer Institute. http://www.cancer.gov/
[62] Protein Data Bank. http://www.rcsb.org/pdb/
[63] P. Murray-Rust, H. S. Rzepa, M. J. Williamson and E. L. Willighagen. Chemical Markup, XML and the Worldwide Web. Part 5. Applications of Chemical Metadata in RSS Aggregators, *J. Chem. Inf. Comp. Sci.* **2004**, 44, 462-469.