

# ChemDig: new approaches to chemically significant indexing and searching of distributed web collections†

Georgios V. Gkoutos, Christopher Leach and Henry S. Rzepa\*

Department of Chemistry, Imperial College of Science, Technology and Medicine,  
London SW7 2AY, UK

Received (in Montpellier, France) 20th November 2001, Accepted 13th December 2001

First published as an Advance Article on the web

We describe an extension of the [ht://Dig](http://Dig) robot-based internet indexing and search engine to include the retrieval of information included in a variety of molecular data formats as defined by chemical MIME types. This is achieved by invoking chemical meta-parsers, software agents designed to provide key meta-data information about the content of the external chemical files. This meta-data can include, for example, derived molecular formula, molecular mass and atom connection table (SMILES) where the content of the file allows this, and other types of content such as author information and supplied keywords. These terms can be automatically added to the searchable terms, and the search outputs can be automatically linked *via* database requests to other external databases containing chemical information. We report our experience in applying this robot to indexing five different remote sites. We discuss different mechanisms for storing and searching for the chemical content, ranging from simple keyword-based searches qualified by chemically significant boolean terms, chemical similarity searches and our experiments in creating more highly structured content that expresses the chemical data using XML-based markup and where XSLT transforms for filtering, searching and rendering the information are used.

## Introduction and background

The widespread adoption of the World-Wide Web system has resulted in the creation of a substantive data, information and resource collection in molecular sciences,<sup>1</sup> albeit one that has been described as a library in which the books are strewn on the floor rather than classified by shelf. Such document collections are frequently expressed in hypertext markup language (HTML), the salient features of which can include links to other such documents and to chemical data files, together with links to server-based resources such as programs, databases and scripts. During the evolution of HTML through various versions to reach the current XHTML specification, the focus was on developing hypertext markup syntax as predominantly a carrier of bibliographic content for browser-based display. Because of this, early inclusion of explicit chemical content was invariably associated with markup-free legacy file formats rather than formats designed to take advantage of the (then) rather limited features of HTML. A more extensible and structured formalism for carrying data known as XML (extensible markup language) has been developed in response to this need. The molecular implementation of XML has focused on CML (chemical markup language).<sup>2</sup> We have demonstrated how molecular data expressed using XML and CML, together with associated formalisms for linking and transforming the data, can be chemically applied.<sup>3</sup> We recognised, however, the need to develop procedures for migrating the substantial inherited legacy of HTML files and associated chemical content into this more structured, data-centric and inter-operable XML environment. We describe here the background to the procedures we have developed to harvest

internet-based molecular resources, together with a discussion of various approaches that can be used to retrieve this content.

## Association of chemical content with HTML documents

Prior to the development of the extensible XML formalisms, we had proposed an infra-structure for using HTML as a linking mechanism for data file types directly associated with chemical information. This was known as the chemical MIME definitions, and these have been rapidly and widely adopted.<sup>4</sup> This mechanism attempts only to identify data associated with a limited set of accepted molecular file types, but it has always been recognised that as usage of these types has evolved, their internal data structures have often developed undocumented or incompletely defined variations. Moreover, the internal data structures were often designed for syntactical compactness rather than clarity and human or machine readability. Typical examples of such files include the Protein databank coordinate format (PDB), the JCAMP data exchange formats for chemical analytical information, and a variety of files for expressing molecular connectivity, coordinates and reactions. There are also many data files associated with widely used computational chemistry modelling program systems such as, for example, MOPAC or Gaussian. We believe that links to around 40 such file types are now to be found globally within HTML pages on chemical servers. However, we also recognised that hitherto, there has been no easy mechanism or tools for identifying, indexing, searching and comparing this content in an automatic and low-cost manner.<sup>5</sup> It is also true that the early focus on associating such content with HTML pages emphasised the presentational aspects rather than retrieval mechanisms. As a result, a variety of syntactical formalisms were used to create this association within the HTML syntax. Our first task, therefore, was to identify those formalisms for which automatic procedures could (or could not) be developed, and we list these below.

† Electronic supplementary information (ESI) available: list of less common chemical MIME media types, output template definitions and stylesheets associated with Table 4. See <http://www.rsc.org/suppdata/nj/b1/b110693g/>

1. The anchor (also known as the hyperlink), invoked as:

```
<a href="molecule_URI" title="Benzene, C6H6">
Prose description of the chemical object</a>
```

where "molecule\_URI" (uniform resource identifier) can be a relative file declaration or an absolute path to a file on a remote server.

2. An in-lined version of the anchor, invoked as:

```
<embed src="molecule_URI" width="50"
height="50">
```

and requiring the user to install a browser plugin.

3. An alternative in-lined version of the anchor, invoked as:

```
<applet codebase="location of code for
displaying the data" width="50" height="50">
<param name="data_source"
value="molecule_URI">
</applet>
```

but which does not require any local software to be installed, the components instead being downloaded from the remote server.

4. Modes 2 and 3, unfortunately, are mutually exclusive, with differing actions required from both the author of the content and its reader for each type. With the release of HTML 4.0 and its successor XHTML 1.0, rationalisation of modes 2 and 3 into one, properly cascading syntax takes the form:

```
<object data="molecule_URI"
type="chemical MIME_type"
classid="specification of how to implement an
object" title="Simple meta-information about
the chemical object">
<param name="run-time" value="initialisation
data">
  <object data="molecule_URI"
  classid="alternative object specification"
  type="chemical MIME type" title="Meta-
  information about the chemical object">
  </object>
</object>
```

Mode 4 has the advantage of conforming to a well-formed and valid XML document, and makes no exclusive assumptions about the state of the user's browser. However, it is often inadequately supported by the current generation of browsers<sup>6</sup> and is relatively rarely used. For this reason, we include it here as a "legacy" format, particularly because it does not expose any chemical data structures other than the MIME type, and will therefore always require special processing to extract this information.

5. Several HTML constructs are often used by authors to control the style or presentation of a page to the user in a compact and attractive manner. One such is the FORM:

```
<form>
  <select>
    <option value="molecule_URI"> Prose
    description of the chemical object
  </option>
  </select>
</form>
```

6. Image maps are widely used to link bitmap representations of molecules to explicit coordinate files, such as:

```
<map name="chemical_image">
<area shape="rect" coords="51,187,165,267"
href="molecule_URI">
</map>

```

7. Also common is the use of a scripting procedure to create user-selectable browser presentation. Many variations are

possible; a common one is associating the anchor in mode 1 above with an event or action such as:

```
<a href="#" onclick="ShowMolecule();">
Prose description of the chemical object</a>
<script>
function ShowMolecule()
{window.open('molecule_URI','1',
width="308",height="300");}
</script>
```

This category is a particularly difficult one to handle, since around ten different types of event could be trapped, and hence a general solution would involve parsing the logic in the script to re-assemble the intended content invocation. A partial solution to this problem would be for the author of such code to also declare a link element for each molecule\_URI as shown below.

8. Document links are part of the header and are intended to normalise all the other forms of document linking into a consistent meta-representation. We have shown<sup>6</sup> that it is possible to capture many of the common link invocations and to declare them as link objects:

```
<link type="chemical/*" rel="alt"
href="molecule_URI" title="description" />
```

This assembling of all document links also has the advantage of having the formal primary chemical MIME type declared, which makes it potentially very easy to identify presumed chemical content in the document.

9. The least exposed way of invoking molecular content would be to reference a remote database using a so-called CGI (common gateway interface) request, which might take the form:

```
<form action="http://remote-site/cgi-
bin/database-interface" title="Description of
action">
```

The molecular resource accessed *via* the database interface is specified by one or more variables collected from the <form> and passed to the database interface program or script. These variables only have significance in the context of the particular database interface referenced and their meaning is not available within the document containing the <form>. In general no conclusions can be drawn about what type of content is referred to. The title attribute, which could loosely carry this information, is very rarely declared. The existing mechanism, therefore, carries little if any meta-information about the remote CGI resources and hence this type of molecular resource linking is unlikely to be captured by any automatic robot mechanism. The proposed successor to the <form> content model is XFORMS, which is a powerful XML-based method that clearly separates the purpose of a request (the data collection semantics) from the manner of its presentation within the browser, and from the data (as name/value pairs) defining the request. Use of this more powerful model will in the future allow more significant identification of the purpose and likely content of remote database resources referenced *via* documents.

10. Finally, and arguably the most difficult category to identify, is chemical content expressed as:

```

```

Possible methods for identifying chemical content in such raster images is discussed in more detail below.

It becomes apparent from the above diversity of syntactical forms for including chemical information in HTML documents that methods for aggregating and transforming this content into a more systematically structured format are desirable. The next section outlines our approach for achieving this.

In this section we describe the characteristics of a typical indexing robot known as ht://Dig, and then describe its enhancement with a set of external chemical meta-parsers, the collective name for which we refer to as ChemDig.

The first implementations of so-called robot-based indexing (traversing) of an interlinked collection of HTML based documents were achieved in 1994 with software such as Lycos or WebCrawler. Such software was soon commercialised, and was followed by a large number of similar robot-based systems for generalised indexing of the internet. One widely used modern system is the UltraSeek server from Infoseek, which in addition to indexing HTML content, employs more advanced features such as meta-tag and image searching, and the ability to crawl client-side image maps. However, such general purpose robots do not in general attempt to identify and index chemical data. We report here our experiments in addressing this particular issue, using an OpenSource software wherever possible, and in particular a document indexing and traversing system known as ht://Dig<sup>7</sup> (a reference to "digging" the content of documents using hypertext transport protocols).

The essential operation of ht://Dig commences *via* the manual specification of one or more root documents in a configuration file, which serve as the starting point for traversing a document collection *via* the hyperlinks found. Using the HTTP protocols, the HTTP header is retrieved from the remote server and if the MIME type declaration contained in this header is found to correspond to HTML, then the content of the document is transferred to the internal syntax parser, and the marked-up HTML content analysed appropriately. Initially, those components of the document marked as the <head> </head> are separated from the <body> </body> of the document. For example, if an HTML markup declaration in the <head> component of the type <Title> The Three dimensional Structure of Mauveine </title> is identified, this title text string can be passed to the indexer, and if considered appropriate, the significant words assigned a high weighting factor. Other forms of declaration in the <head> such as <meta name=DC. Description content="Mauveine coordinates"> can be identified and associated with pre-defined weighting factors. Such meta-data declarations are particularly important in identifying key properties of the documents that might not otherwise be easily identified, such as the author of the document, any date or ownership associated with it, or even more specifically whether the document has any chemical information.<sup>8</sup> Thereafter, the text content of the <body> of the document is indexed according to well-defined algorithms, and appropriate weighting factors for specific elements of the body, such as <h1> </h1> headings, assigned.

If during the course of parsing the HTML document, any link to another HTML document is identified, then an attempt will be made to also retrieve this document and perform the same indexing. The configuration of the robot search can specify whether the traversing of the hyperlinks is restricted purely to any document at the same hierarchical directory level as the root level or below it, or whether links above or out of this directory structure are also allowed. The robot will also obey the so-called robot-exclusion rules, a mechanism whereby the administrator of a remote collection can specify whether any directories of the collection are out of bounds to the robot. Typically, traversing a remote document collection of tens of thousands of HTML documents can be completed in just a few hours, depending of course on the bandwidth of the network connecting the remote server and the indexing computer.

Most robot-based indexing software supports not only the parsing of simple HTML documents, but also of other types of documents specified by their MIME types, such as text/plain and more complex documents such as application/word or

application/pdf, corresponding to the Microsoft Word and the Adobe Acrobat formats. In general, however, subject-specific MIME types are not by default included in the indexing. Our interest in the ht://Dig indexing software arose because this code does allow a specification of external parsers for such file types, and we also noted that availability of the source code would allow us the option of specific modifications to be made if necessary. Other features of ht://Dig that attracted our attention included an extensible method for parsing meta-data declarations in document headers, which we felt might be useful in a chemical context. Finally, we noted that the system has been shown to be highly efficient for traversing collections of up to 1 million documents in a reasonable time, and so could be considered suitable for handling small and medium sized Web sites (e.g., intranets) containing less than this maximum number of documents.

## The ChemDig implementation

In specifying the functionality of ChemDig, we had four objectives in mind:

1. to identify the existence of chemical content from distributed document collections;
2. to convert and store structurally homogeneous chemical content in appropriate database;
3. to add content (meta-data) to the databases where applicable to aid in resource discovery;
4. to explore several novel mechanisms for retrieval of the chemical data from the databases.

Fig. 1(a) shows an overview of the core operation of ChemDig. Some of these components have been described in preliminary detail elsewhere.<sup>6,9</sup> The overall operation involved three discrete phases.

1. The ht://Dig software<sup>7</sup> contains a parser that corresponds to version 2.0 of the HTML standard, but which is capable of handling many exceptions to these standards. Some exceptions, however, are not handled explicitly, such as documents that make use of in-lined RasMol scripts to control the function and appearance of linked molecule coordinate files intended for a Chime display. The use of <and> script operators conflicts with the same operators used for containing HTML elements, and since the resulting HTML document is not well-formed, the HTML parser in ht://Dig will fail. We considered it essential to solve this problem by pre-processing any document handed to ht://Dig to ensure that it was well-formed. This was achieved using JchemTidy,<sup>9</sup> which can be used to search for occurrences of RasMol Scripts and declare the <and> operators as entities. JChemTidy also converts older HTML markup to the XHTML standard and an accompanying module called JChemMeta also normalises hyperlinks of the type described in the link modes above by inserting the corresponding link declarations, which the ht://Dig software does honour. The resulting file collection can be re-created on a local hard disk with the directory structures retained in readiness for indexing using ChemDig, and also can be used to replace the original collection.

2. The second phase involves invoking the ht://Dig robot. Some changes to the ht://Dig source code were required, at version 3.1.1 when the project was started and currently at version 3.2. One significant limitation of ht://Dig is that parsing of the common attributes of elements such as title="..." or alt="..." is not yet implemented, and this was overcome by using JChemMeta<sup>9</sup> to insert a title attributed derived from chemical files into the meta-data declarations of the XHTML file, where they are accessible to the ht://Dig robot. The interface for calls to an external parser was written in the Java programming language to facilitate future implementation on a variety of different operating systems, and also to allow modular deployment in other indexing software.

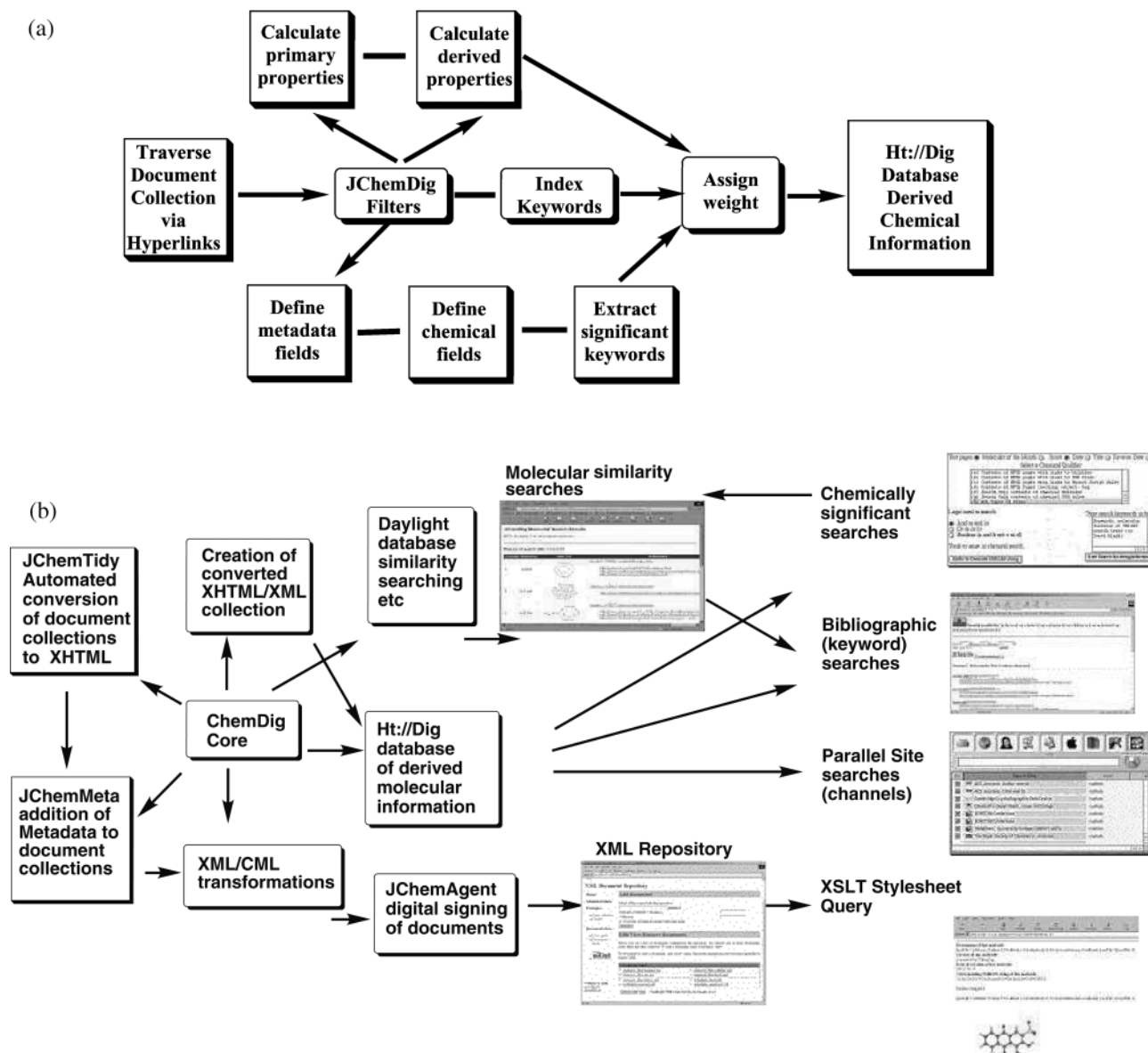


Fig. 1 (a) The core ChemDig functionality. (b) The extended ChemDig functionality.

3. The final stage involves meta-parsing of chemical files linked to the documents trawled by *ht://Dig*, and adding any derived fields to the document header of the original HTML file [Fig. 1(a)] By meta-parser, we mean a declared procedure for identifying meta-information about the internal content of specific files, such as any title, author, date, or chemically relevant information such as molecular formula. Our objectives were not only to identify the performance characteristics of such an operation, but also to gather statistics on the extent that external sites have adopted the various hyperlinking mechanisms described above to include specifically defined chemical data. We also wished to establish how links or pointers to other databases of chemical information sources might be automatically added, and perhaps most importantly to create an automatic mechanism for converting the information so-collected into an XML-based document collection. We note that this robot-based method for traversing a distributed collection of chemical information has some significant differences from the more traditional methods for registering compound information in specifically designed chemical databases. Although the current version of *ht://Dig* does not support fielded searches on the content of added meta-tags, other search engines such as InfoSeek do. Details of the chemical meta-parsing are given in the next section.

### Chemical meta-parsers

The most frequently used types for which meta-parsers were written are shown in Table 1. A list of less common types supported is available in the ESI.

The meta-parsers were not intended to act as precise validators or complete parsers of the specific content of each file type. The majority of chemical file types are defined as 7-bit ASCII file types, where the identification of information components is defined by either strings of text terminated with an end-of-line character, or strings of text with a characteristic prefix, and no other structured identifiers. Typically, these types of delimiters are used within the file structure to identify titles, comments, authors, dates or keywords used to specify details of the input for a computational chemistry calculation or instrumental output, and for many of the file types, definitions of these fields are documented to a greater or lesser extent. Such files are often found in the supplemental information sections of electronic journals. A more challenging type of file to parse is the much more finely grained output file from, for example, an instrument or modeling calculation, and these we have deferred for processing using appropriately mark-up formats based on XML and CML.

**Table 1** A list of frequently occurring links to chemical MIME media types<sup>a</sup>

Type	File name extension	Description
chemical/x-csml	csml, csm	RasMol script language
chemical/x-gaussian-input	gau, gif	Gaussian input format
chemical/x-jcamp-dx	dx, jdx	JCAMP spectral format
chemical/x-mdl-molfile	mol	MDL molfile
chemical/x-mopac-input	mop	MOPAC input format
chemical/x-pdb	pdb	Protein DataBank
model/vrml	vrml, wrl	Virtual reality modeling language (VRML)
chemical/x-xyz	xyz	Co-ordinate animation format

<sup>a</sup> A list of less common chemical file types is available in the ESI.

A smaller number of chemical file types are expressed as 8-bit binary files, where the structure is not defined by end-of-line markers, and for which byte-code structure definitions are not readily available, being considered proprietary. When these types are encountered, we chose merely to pass back to the index engine the existence of the file and its formal file-name. There was one exception to this, when the 8-bit binary file was detected as a “gzipped” compressed format, typically employed in chemical/x-pdb and model/vrml formats to reduce the file size. Here it is trivial to internally decompress the format and parse it as a 7-bit ASCII file. In general, however, the binary compression mode is propriety and no attempt was made to read the contents of such files.

#### Handling chemical content expressed in bitmap images

Largely for historical reasons, much Web-based document content is carried in the form of in-lined raster images, encoded in formats such as GIF, JPEG or PNG (and animated versions, including video animations). These represent a notoriously difficult problem in what has been described as “machine vision”, or software image recognition. Whilst a major activity outside of the chemical arena, surprisingly little work has been done on chemical recognition in such images.<sup>10</sup> This work has centred on deriving atom and bond connection tables from line diagrams, which of necessity has to be a process with effectively zero error rate. Human intervention is essential to achieve this. Less attention has been given to identifying simple chemical meta-information from such images, or indeed answering questions such as “does this image contain any representations of chemistry?”. Recent developments in “machine vision” hold the prospect of automatically generating such meta-information during the aggregation process,<sup>11</sup> but currently, only some simple methods can be applied to identify potential chemical content in images. These include:

1. Image files can carry associated meta-fields in the form of HTML “alt” and “longdesc” descriptors, which can be added to the indexing process. Unfortunately, few image files carry any sensible description, and when they do, there is no guarantee that it is appropriately associated with the image (*i.e.*, such descriptions are often inherited from other images during the authoring process).

2. GIF and PNG files can contain invisible text-based fields with useful chemical information such as atom coordinates and Molfile connection tables.<sup>12</sup> We wrote a parser for these files that can detect the presence of such information, and flag it as indicating the likely presence of a chemical structure file (chemical/x-mdl-molfile).

3. It is also possible in a general sense to automatically convert an arbitrary raster image into a vector description such as a SVG.<sup>13</sup> With chemical structure diagrams of course, such a description is far more concise. It may be possible from the resulting number and connectivity of the vectors to recognize patterns typical of chemical structures, and hence add

corresponding meta-information. This area too is under-developed and one where we anticipate rapid progress in the future.

#### Defining external meta-parsers

The external meta-parsers are specified *via* the ht://Dig configuration file, an abbreviated example of which is illustrated in Scheme 1. The configuration includes specifications of particular meta-data declarations to be indexed. We have included examples of the Dublin Core set together with some chemical extensions we have previously proposed.<sup>8</sup>

The relevancy ranking is evaluated on the basis of the weighting factors associated with the original index entries and the frequency of occurrence of the search string in the document itself. In indexing the content, we have chosen a default weight of 1 for any string located in the body of a HTML document, and factors of 10 for strings originating in the document header, including meta-data declarations, and strings occurring within the body of an external chemical document.

#### Chemical validation and derived information

With the most common chemical legacy file formats, we chose to implement some chemical parsing, validation and derived molecular information. An overview of these and other post-processing operations is given in Fig. 1(b).

For the file types chemical/x-mdl-molfile, chemical/x-mdl-rdfile, chemical/x-mdl-rxnfile, chemical/x-mdl-sdfile and certain flavours of chemical/x-pdb, it is possible to straightforwardly derive a molecular formula and molecular weight for the substance. For several of the formats containing molecule atom and bond information, such as chemical/mdl-molfile and chemical/x-pdb, the parser was also modified to pass a request to an external program to validate the molecular content and to derive a unique SMILES string corresponding to the molecule. If the chemical validation was successful these strings were returned to the ht://Dig index engine as keywords to enable the users to search for the unique SMILES string of a molecule.<sup>14</sup> We employed two external programs to validate the file and to derive the SMILES identifier. A CGI-type request can be issued to the Daylight toolkit running on a remote machine, and a similar operation is also possible using JME (Java Molecular Editor).<sup>15</sup> If the validation process fails, an error message is returned instead. We do note that the Daylight and JME canonicalisation routines do not always produce identical unique SMILES strings for the same molecule. Normalisation of this string must be done at the generation stage, since it cannot be achieved at the searching stage (the Daylight system, for example, will always re-normalise a SMILES string as part of its own search sequence).

These variables are all passed back in standard form for inclusion in the ht://Dig database, with the option of assigning discrete weighting factors to these entries. The

```

database_dir: /disk1/www/htdig/new/tests
allow_in_form: search_algorithm
start_url: http://www.ch.ic.ac.uk/chemime/tests/
search_algorithm: exact:1 synonyms:0.5 endings:0.1
external_parsers: chemical/x-pdb "/usr/java/bin/java
chemical.Htdigfront" \
chemical/x-jcamp-dx "/usr/java/bin/java chemical.Htdigfront" \
chemical/x-mopac-input "/usr/java/bin/java chemical.Htdigfront" \
chemical/x-mdl-molfile "/usr/java/bin/java chemical.Htdigfront" \
chemical/x-pdb "/usr/java/bin/java chemical.Htdigfront" \
chemical/x-xyz "/usr/java/bin/java chemical.Htdigfront" \
model/vrml "/usr/java/bin/java chemical.Htdigfront" \
use_meta_description: true
#the defined meta tags
keywords_meta_tag_names : DC.chem.coordinates
DC.chem.substance.smiles \
DC.chem.substance.formula DC.chem.substance.mw DC.Title
DC.Publisher \
DC.Date DC.subject DC.Format DC.Coverage DC.Type DC.description
DC.Creator \
DC.description
# weighting given to metadata elements
meta_description_factor: 10
title_factor: 10
keyword_factor: 10
heading_factor_1: 10
heading_factor_2: 9
text_factor: 4
template_map: Long ${common_dir}/chemical.html

```

**Scheme 1** Example configuration file for ht://Dig.

derived meta-data is also added to the original HTML document in the form of specific meta-declarations, such as DC.CHEM.coordinates, DC.CHEM.substance.smiles, *etc.* (Scheme 1). This would allow subsequent searches to be modified by such qualifiers; this is currently not an option with ht://Dig but is available with some other search tools.

## Results and discussion

The initial phase involved selecting a set of distributed sites as the test basis (Table 2). We selected a number of sites from a standard collection of chemical resources,<sup>16</sup> using criteria such as geographical distribution and the indicated presence of possible structured chemical content. Initially, a single configuration file containing the start\_URL for the root document of the project at each site was used. Retrieval of the chemical content of these sites served to highlight several problems that needed specific solution.

1. It is not uncommon when referencing molecule coordinate files from HTML pages to include so-called RasMol scripts, which serve to control the function and presentation of the molecule. These problems relate to the use of the characters <and> to define operations within the RasMol script, but which also have a conflicting meaning in HTML. The solution is to convert all occurrences of these characters to entities, a process that can be automated using JChemTidy.<sup>9</sup> This operation of JChemTidy requires a temporary but complete local copy of the remote site to be created using a mirroring tool such as w3mir (which needs only the syntax of the hyperlink to be correct and does not fail with other badly-formed HTML). JChemTidy corrects and converts this to a locally stored and well-formed XHTML collection, which can then be processed with ChemDig. This process breaks down in the specific instance when absolute URL references are used instead of the equivalent relative reference in any of the original documents. ChemDig in traversing this collection will still try to resolve the original (badly-formed) HTML documents and hence fail. This can be prevented by automatically correcting the local mirrored collection using an appropriate tool.<sup>17</sup>

2. The identification of linked files containing chemical content depends on correct use of the chemical MIME headers. In some cases, we encountered incorrectly configured

servers where the MIME type did not appear to correspond to the content of the files. Such reconciliation can only be done manually by inspection, and we made no attempt to index such sites.

3. Traversing a site using document links will not identify so-called orphaned files, that is those that are resident in the directories but are not accessible *via* any HTML document. We identified one site where the chemical file count obtained

**Table 2** Chemical file types and counts identified at chemical sites

Type of file	Number of occurrences
http://www.ch.ic.ac.uk/motm/	
text/html	215
chemical/x-csml	39
chemical/x-jcamp-dx	42
chemical/x-mdl-molfile	18
chemical/x-mopac-input	14
chemical/x-pdb	38
world/vrml	16
chemical/x-xyz	4
Total number	383
http://www.bris.ac.uk/Depts/Chemistry/MOTM/motm.htm	
text/html	679
chemical/x-mdl-molfile	79
chemical/x-pdb	217
chemical/x-csml	20
Total number	995
http://www.chem.ox.ac.uk/mom/	
text/html	284
chemical/x-mdl-molfile	83
chemical/x-pdb	121
chemical/x-csml	21
Total number	508
http://www.dcu.ie/~chemist/pratt/pdb/pdb.htm	
text/html	4
chemical/x-mdl-molfile	162
chemical/x-pdb	123
Total number	289
http://www.wsu.edu/~wherland/wwwlist01.htm	
text/html	2
chemical/x-mdl-molfile	464
chemical/x-pdb	15
Total number	481

by linked document traversing was significantly less than that obtained from a site directory listing. This again is not an issue that we attempted to resolve.

4. Some sites serve two functions, of serving original chemical documents and of acting as portals to other remote sites, in varying proportions. There are two ways of indexing a site containing many off-site references. Normally, *ht://Dig* operates only within the confines of the declared *start\_URL* (Scheme 1). It is possible to release this constraint and define instead a *hop\_limit*, which is simply the number of consecutive links allowed from the root document, irrespective of whether the link is at the site declared by *start\_URL* or elsewhere. A *hop\_limit* = 1 will only follow the links directly declared on the *start\_URL*. Increasing to a *hop\_limit* = 2 was found to already result in a high degree of irrelevant indexing resulting from non-chemical links incorporated into the documents, links that in general have no meta-data descriptors on which an automatic judgement can be made. When such uncontrolled indexing was encountered by the traversing robot, we flagged the site for manual inspection of the root document, and the off-site portal entries were inserted into the *ht://Dig* configuration file as separate *start\_URL* declarations.

5. On some sites, their evolution has been away from static linked documents towards closed databases accessed *via* CGI-style requests. As noted above, the absence of any current mechanism for clearly identifying the purpose of any such request means that little can be done to capture data or meta-information about such sites.

From these experiences, we conclude that the relatively loose standards and compliance of many Web sites means that a complete level of automation for the operation of a robot based on document traversals may not always be possible. However, the degree of human intervention required in the process was manageable, and would be particularly effective in an Intranet environment, for example.

### Creating chemically enhanced databases

At this stage, we were ready to produce searchable databases from the resulting molecular harvesting. We have evaluated three options for this procedure.

1. Creating *ht://Dig* based ChemDig databases taking two forms, one each for the sites, and a consolidated one integrating all the sites. These databases would contain an index of all the prose keywords found in the HTML documents together with the meta-data obtained from the meta-parsing of the linked chemical files.

2. Creating a chemical structure database containing all molecular connection tables (*e.g.*, Molfiles and PDB files) located by ChemDig. In principle, the database could also contain, for example, spectroscopic information gleaned from JCAMP files, but we did not investigate this option at this stage.

3. Creating a parallel XML-based repository, comprising a seamless integration of HTML as converted to XHTML with chemical files automatically converted to XML representation using CML. An XML file was created for each original HTML document containing transcluded molecule coordinate files for this process, and each document was inserted into an XML-based Object store. This store can be separately searched.

We have previously described how separate *ht://Dig* databases can be aggregated with other search engines into a Chemical Search Channel.<sup>18</sup>

**Searches using the ChemDig databases.** The *ht://Dig* database comprises all the bibliographic keywords (other than the usual default stop words, which are not indexed) contained on a central server. The presence of added chemical meta-data deriving from the operation of JChemMeta allowed the resulting database to be searched in two ways.

1. A simple keyword search invoked by using the following embedded XHTML entries invoked from a browser page:

```
<form method="post"
action="http://www.ch.ic.ac.uk/cgi-
bin/htsearch">
<input type="hidden" name="config"
value="database_name" />
<input type="text" size="9" name="words"
value="" /><br />
</form>
```

Here *htsearch* is the search program component of *ht://Dig*, whilst the values of the variables *config* and *words* contain the database to be searched and the user specified search string to be passed to it. This results in the following request to the server:

```
http://www.ch.ic.ac.uk/cgi-bin/htsearch?words=
value_entered_by_user&config=database_name
```

The search string can derive from occurrences either in the *<header>* components of the HTML document or the text components of the body. It can also include the ALT attribute of the *<IMG>* image element, although these are rarely defined with chemically significant values. Text strings gathered by or derived using the external chemical meta-parsers can also be searched for, and all searches can be performed with the usual boolean operators (AND, NOT and OR) and with regular expressions of the type *alizarin\**. Three derived text expressions can also be specified as a search string: the molecular formula (expressed as *CnHnElm*), the molecular mass specified as a numeric string, and the unique SMILES representation of a molecular connection table. These search terms have to be "quoted" to ensure they are treated as a single string.

2. The second mode makes use of the automatic insertion by the JChemTidy tool of meta-data headers of the type:

```
<meta name="DC.Subject"
content="Molecule-href, Molfile-format" />
```

This string is assumed to be unique to JChemTidy, and indicates that the document contains a formal link to a molecular connection table or coordinates, specifically expressed in the Molfile format. Similar entries can be inserted for other formats. This now allows a more restricted search to be conducted only for XHTML documents that contain links to such chemical files. The code to achieve this is:

```
<select name="words" multiple="multiple">
<option value="Molfile-format and">
Search only HTML pages with links to Molfiles
</option>
<option value="PDB-format and">
Search only HTML pages with links to PDB files
</option>
</select>
```

In effect, this forces a boolean operation specifying a search for the user provided string AND the JChemTidy unique string. A similar technique can be used to search only for keywords explicitly extracted from the contents of Molfiles and PDB files, for example (and again in principle any chemical file), by inserting a unique string into such files such as "molfile\_molecule" or "pdb\_molecule".

**Output templates for ChemDig.** Customizable output templates based on HTML for presenting to the user the results of the search allow inclusion of the title of any HTML or external chemical file that contains at least one occurrence of the search terms. In the absence of a title attribute, the name of the file is displayed. The computed relevance ranking of the document is displayed in the form of a star rating, together with a meta-data term specifying the document description if available. In a chemical file, if no title field is specified, this is by default taken

as the first comment line found in the file. Next, the fully specified URL of the document is given, with an anchor-based (<a></a>) hyperlink to that document. If the document comprises one of the chemical MIME types, the user will of course have to implement a viewer for that document type. Typically, if the document is of the type chemical/x-mdl-molfile, then an external program such as RasMol or a browser plug-in such as Chime or Chem3D would be appropriate.

The output template was also modified to automatically include several additional terms. Firstly, for each connection table list item located, an entry was inserted to pass this specific query to the Daylight database to search for similar molecules, and an entry inserted to start a conversion process to CML.<sup>19</sup> Secondly, we include CGI requests for some chemically relevant databases, which were considered as capable of containing information for further chemically related queries that can be invoked by the user. These were presented as active links in the results page of the query and were

independent of the number of the hit results. The output template (available in the ESI) gives search output as shown in Fig. 2.

**Molecular substructure searches.** The ht://Dig search interface is essentially bibliographic in nature. It is possible to use it to search for molecules *via* simple descriptors such as molecular formula or a more unique atom connectivity descriptor such as SMILES. Such fields have been added by our tools to the meta-information of the HTML file that invokes a molecular coordinate file. Such searches, however, can only return precise matches, and cannot be used for sub-structure searches. A clearly identified need is to be able to search for similar molecules to that identified, perhaps by an ht://Dig search.

Because ChemDig handles each Molfile or PDB file explicitly, it is straightforward to pass these files directly to a specially constructed molecular database. We chose the Daylight THOR/MERLIN system to demonstrate this in operation.

(a)

Database: (a) (b) (c) (d) (e) (f)

Select a Chemical Qualifier (CQ):

1. Contents of XHTML pages with links to MDL Molfile coordinates
2. Contents of XHTML pages with links to MOPAC Input
3. Contents of XHTML pages with links to PDB coordinates
4. Contents of XHTML pages with links to XYZ coordinates
5. Contents of XHTML pages with links to JCAMP-DX spectra
6. Contents of XHTML pages with links to VRML models
7. Contents of XHTML pages with links to Rasmol Scripts/CShL
8. Contents of XHTML Pages invoking a CGI Process
9. Search only contents of chemical/MDL Molfiles
10. Search only metadata in chemical/PDB files
11. Search only metadata in chemical/MOPAC Input
12. Search only metadata in chemical/Gaussian Input
13. Search only metadata in chemical/XYZ coordinates
14. Search only metadata in chemical/JCAMP Spectral files
15. Search only metadata in VRML Models
16. Search only metadata/connection tables in GIF images
17. Search All types of Chemical Files

Search Logic:

☒ CQ# AND KW<sup>h</sup>

☐ CQ# OR KW<sup>h</sup>

☐ Boolean (a and b not c or d in KW)

Display style:

☒ Score

☐ Date

☐ Title

☐ Reverse Date

Search keywords (KW)

alizarin

Start Search for string/selection

(a) <http://www.ch.ic.ac.uk/motm/>

(b) <http://www.bris.ac.uk/Depts/Chemistry/MOTM/motm.htm>

(c) <http://www.chem.ox.ac.uk/mom/>

(d) <http://www.dcu.ie/~chem/pratt/pdb/pdb.htm>

(e) <http://www.wsu.edu/~wherland/wrwlst01.htm>

(f) All

(g) CQ = Chemical qualifier

(h) KW = Keyword term

(b)

(b) ChemDig Search results for 'molfile\_molecule and (alizarin or alizarine)'

Documents 1 - 1 of 1 matches. More ☆'s indicate a better match.

[alizarin.mol]☆☆☆☆☆☆☆☆☆☆☆☆☆☆ 100

[Search Daylight database](#)

[Convert to XHTML/CML and Sign http://www.ch.ic.ac.uk/motm/alizarin.mol/](#) 04/20/01, 2243 bytes

Repeat Search with other Databases

[NIST Chemistry WebBook](#) [Brookhaven Protein Databank](#) [ChemFinder](#)

Fig. 2 (a) Input window for ChemDig search. (b) Resulting output for the default search.

The captured fields include not only the molecular connection table, but also the absolute URI indicating the location of the original file. A search page can be defined in HTML and the output results can include not only the list of molecules, but also a link to the original site (Table 3).

**Searches based on XML documents.** JChemTidy normalises HTML documents located using `ht://Dig` into XML-conforming XHTML representations. Similarly, some chemical coordinate files such as the MDL molfile (versions V2000 and V3000) can be passed to a script that converts them to an XML-conforming CML representation.<sup>19</sup> The resulting two components (XHTML and CML) can be coalesced into a single XML document, the identification of each component being achieved using the appropriate namespaces. In effect, the first generation chemical/MIME mechanism has been replaced by a more structured and extensible approach enabling more finely grained information to be expressed within the document. This now allows the documents to be searched for patterns using XSLT stylesheet transformations, on the premise that all the significant content is marked up in a well-defined and syntactically correct manner with a resolvable structure to the data. The desired search pattern, which can be a mixture of bibliographic content carried in XHTML and molecular, atomic and bond information carried in CML, is then specified in an XSLT stylesheet declaration. Application of such a stylesheet to an XML document or document collection can be performed by generic software such as the browser itself (for example, Internet Explorer 6), or more specific software designed to run under batch conditions (such as Saxon<sup>20</sup>). Unlike the Daylight searches described above, these functions can be accomplished using OpenSource software.

To illustrate the concept, we include two simple pattern searches (Table 4) that involve operations such as counting the number of molecules contained in an XML document and identifying any molecule containing sulfur atoms. The stylesheet also includes processing instructions for transforming

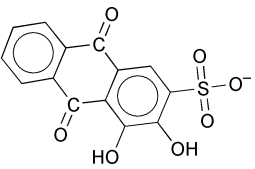
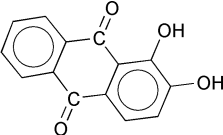
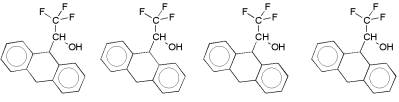
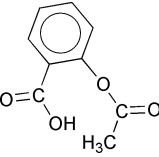
any located molecules to Molfile format, for example, for display in a browser window.<sup>19</sup> This concept is readily extensible, since new XSLT search queries can be either written directly using XSLT grammars, or assembled from stylesheet library components, or potentially generated dynamically using an appropriate chemical query tool. The stylesheet can also include filters to display any specified property and meta-data fields detected in the XML document.

A search based purely on XSLT patterns is not very efficient when dealing with a large collection of such documents, where pre-indexing is required for speed. To evaluate this mode, we used another OpenSource program called eXist<sup>21</sup> to create a testbed CML repository. CML documents can be stored either internally within eXist using a native XML-database written in Java that stores data to disk and indexes them, or externally using a relational backend such as MySQL, PostgreSQL or Oracle. Documents can be retrieved, stored, viewed and edited dynamically (Fig. 3), since the eXist server can be accessed *via* HTTP. The built-in search engine provides fast XPath queries, using indexes for all the element, text and attribute nodes present in the original document collection, and is designed to cope with large collections. The eXist database can also be configured to associate any retrieved documents with declared stylesheet libraries. This allows the document to be appropriately transformed prior to presentation to the user. We also note that since these stylesheets themselves are XML documents, they too can be deposited and stored in eXist, and if necessary searched for. For this to be effective, the stylesheet document should be annotated with meta-data to describe its operation and allow search retrieval.

## Conclusions

When the Web started being populated with molecular content around 1994, it received often justified criticism for having none of the formal rigour of a database system, and for being

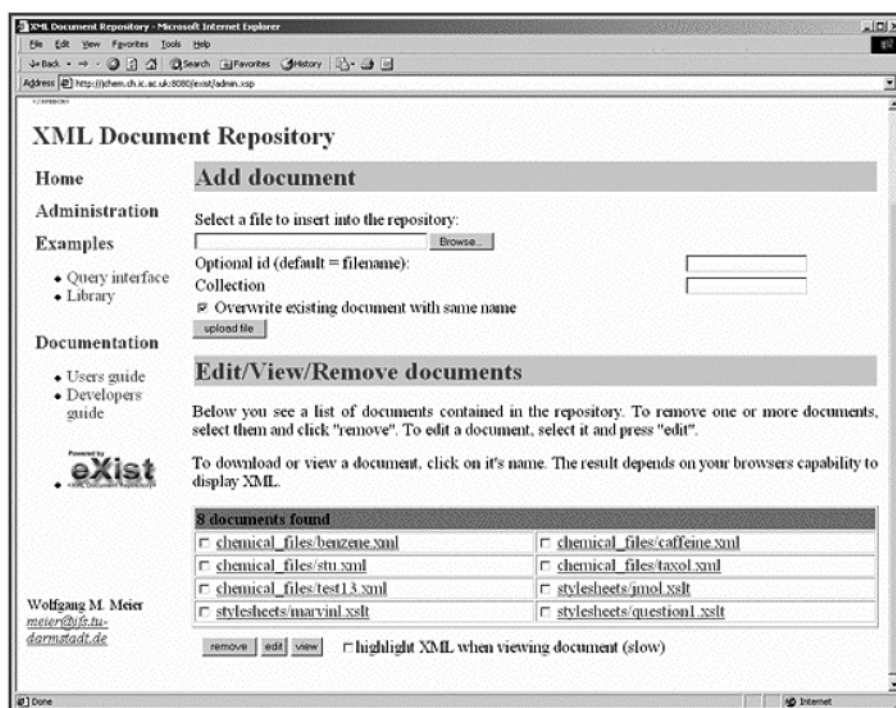
**Table 3** Substructures search of Daylight database resulting from ChemDig

Molecular search results				
MERLIN ready: 8.13 searchable structures				
Result of search with: <chem>Oc1c(O)c2C(=O)c3ccccc3C(=O)c2cc1S(=O)(=O)[O-]</chem>				
Number	Similarity	SMILES	References	
1.	1.0000		Search using HTDIG	
2.	1.6418		Search using HTDIG <a href="http://www.faidherbe.org/site/cours/dupuis/images6/alizar.pdb">http://www.faidherbe.org/site/cours/dupuis/images6/alizar.pdb</a>	
3.	0.3152		Search using HTDIG <a href="http://www.ch.ic.ac.uk/motm/nanobiotic/pirkle.rs.pdb">http://www.ch.ic.ac.uk/motm/nanobiotic/pirkle.rs.pdb</a> <a href="http://www.ch.ic.ac.uk/GPCR/TEST/imperial/motm/nanobiotic/pirkle-rs.pdb">http://www.ch.ic.ac.uk/GPCR/TEST/imperial/motm/nanobiotic/pirkle-rs.pdb</a> <a href="http://www.ch.ic.ac.uk/GPCR/CAROL/motm/nanobiotic/pirkle-rs.pdb">http://www.ch.ic.ac.uk/GPCR/CAROL/motm/nanobiotic/pirkle-rs.pdb</a>	
4.	0.2490		Search using HTDIG <a href="http://www.chem.ox.ac.uk/mom/aspirin/ASPIRIN.mol">http://www.chem.ox.ac.uk/mom/aspirin/ASPIRIN.mol</a> <a href="http://www.bris.ac.uk/Depts/Chemistry/MOTM/aspirin/aspirin.pdb">http://www.bris.ac.uk/Depts/Chemistry/MOTM/aspirin/aspirin.pdb</a> <a href="http://www.faidherbe.org/site/cours/dupuis/images6/aspirine.pdb">http://www.faidherbe.org/site/cours/dupuis/images6/aspirine.pdb</a>	

**Table 4** XSLT querying of XML/CML generated using ChemDig<sup>a</sup>

Query	XSLT query and template strings
Find total number of molecules	<code>&lt;xsl:value-of select = "count(/cml:molecule)"/&gt;</code>
Convert any molecules containing sulfur to Molfiles	<code>&lt;xsl:for-each select = "cml:molecule[cml:atomArray/cml:atom/cml:string = 'S']"&gt;</code> <code>&lt;textarea rows = "10" cols = "80"&gt;</code> <code>&lt;xsl:call-template name = "convertV2000"&gt; &lt;xsl:with-param name = "enLine"&gt;</code> <code>&lt;/xsl:with-param&gt;&lt;/xsl:call-template&gt;</code> <code>&lt;/textarea&gt;</code>
Display any molecules containing sulfur using Jmol	<code>&lt;xsl:for-each select = "cml:molecule[cml:atomArray/cml:atomsol/cml:string = 'S']"&gt;</code> <code>&lt;xsl:with-param name = "display"&gt;jmol&lt;/xsl:with-param&gt;</code> <code>&lt;xsl:with-param name = "width"&gt;400&lt;/xsl:with-param&gt;</code> <code>&lt;xsl:with-param name = "height"&gt;300&lt;/xsl:with-param&gt;</code> <code>&lt;xsl:with-param name = "id"&gt;</code> <code>&lt;xsl:value-of select = "generate-id()"/&gt;</code> <code>&lt;/xsl:with-param&gt;</code>

<sup>a</sup> To invoke these stylesheets, use Internet Explorer 6.

**Fig. 3** eXist database interface to XML document repository.

particularly difficult to search in the specific context of molecular information. Amongst the significant problems identified were the issue of badly formed HTML files that were not syntactically valid, of no semantic markup for explicit molecular information (even the molecular formula was not really parsable chemically), of a variety of often inconsistent ways in which a variety of imprecisely defined chemical format files were declared, linked and displayed, of the almost complete lack of meta-data describing any chemical data that might have been described, and of opaque mechanisms for linking documents to server-based databases where the internal data structures were not exposed. The creation of information and data grid based projects such as Globus<sup>22</sup> emphasizes the importance of establishing scalable procedures to address such issues.

We have presented here a number of approaches based on a traversing robot, which can be used to rectify some of these deficiencies. Although these still require some manual intervention, it has proved possible to construct various forms of a structured chemical database by traversing a typical site containing molecular information. These solutions range from the

traditional customised server-based chemical database to distributed "self-defining" document collections searchable at the client using stylesheets. Some issues still remain. The ubiquitous use of bitmap images to describe chemical structures means that not only is molecular constitution and connection irretrievably lost from such documents, but that even meta-information signifying the presence of a molecule is not captured. We believe the development of appropriate "machine vision" mechanisms for recognising meta-content in images is desirable. A second issue relates to the current non-availability of on-line resources for identifying chemical uniqueness from identified molecular connectivity; there is probably much undetected duplication. The resolution of such problems must lie in persuading the community to focus on capturing molecular content into a self-describing document structure, comprising finely grained molecular mark-up with globally unique associated identifiers.<sup>23</sup> Such a highly distributed global information and knowledge base would represent a novel and extensible mechanism for scientific dissemination, which is seen by many as the ultimate vision of a semantic World-Wide web.<sup>24</sup>

## Acknowledgements

One of us (G. V. G.) thanks Merck Sharp and Dohme and the EPSRC for the award of a scholarship. We also gratefully acknowledge the help given us by Paul May at Bristol and Karl Harrison at Oxford, and Daylight Information Systems.

## References

- 1 See, for example: H. S. Rzepa, P. Murray-Rust and B. J. Whitaker, *Chem. Soc. Rev.*, 1997, 1–10.
- 2 P. Murray-Rust, *World Wide Web Journal*, 1997, pp. 135–147; P. Murray-Rust and H. S. Rzepa, *J. Chem. Inf. Comput. Sci.*, 1999, **39**, 928.
- 3 P. Murray-Rust, H. S. Rzepa and M. Wright, *New J. Chem.*, 2001, **25**, 618–634; P. Murray-Rust, H. S. Rzepa, M. Wright and S. Zara, *Chem. Commun.*, 2000, 1471–1472.
- 4 H. S. Rzepa, P. Murray-Rust and B. J. Whitaker, *J. Chem. Inf. Comput. Sci.*, 1998, **38**, 976–982.
- 5 J. S. Brecher, *Chimia*, 1999, **52**, 650; C. Leach and H. S. Rzepa, in *Electronic Conference on Heterocyclic Chemistry '96*, ed. H. S. Rzepa, J. Snyder and C. Leach, Royal Society of Chemistry, Cambridge, 1997, article 121, CD-ROM ISBN 0-85404-894-4; L. Patiny, *Internet J. Chem.*, 2000, **3**, article 2; S. K. Lin and L. Patiny, *Internet J. Chem.*, 2000, **3**, article 1.
- 6 G. V. Gkoutos, H. S. Rzepa and M. Wright, *Internet J. Chem.*, 2000, **3**, article 7.
- 7 A. Scherpbier, <http://www.htdig.org/>
- 8 S. Weibel, *Bull. Am. Soc. Inf. Sci.*, 1997, **24**, 9–11; G. V. Gkoutos and H. S. Rzepa, in *Electronic Conference on Synthesis in Organic Chemistry (ECSOC-2)*, ed. S.-K. Lin and E. Pombo-Villar, MDPI, Basel, 1999, CD-ROM, ISBN 3-906980-01-4.
- 9 G. V. Gkoutos, P. Kenway and H. S. Rzepa, *J. Chem. Inf. Comput. Sci.*, 2001, **41**, 253–258; G. V. Gkoutos, P. R. Kenway and H. S. Rzepa, *New J. Chem.*, 2001, **25**, 635–638.
- 10 P. Ibison, M. Jacquot and F. Kam, *J. Chem. Inf. Comput. Sci.*, 1992, **32**, 373–378; R. Simon and A. P. Johnson, *J. Chem. Inf. Comput. Sci.*, 1993, **33**, 338–344; R. Simon and A. P. Johnson, *J. Chem. Inf. Comput. Sci.*, 1997, **37**, 109–116.
- 11 R. M. Clark, O. A. Adjei and H. Johal, “Machine classification of textures using incremental learning based on the mean and variance of the multi-dimensional feature space”, International Conference on Mechatronics and Robotics 2000, Saint Petersburg, Russia, May 2000
- 12 See: W. D. Ihlefeldt, <http://www2.chemie.uni-erlangen.de/services/gifcreator/>
- 13 G. V. Gkoutos, P. Murray-Rust, H. S. Rzepa and M. Wright, *Internet J. Chem.*, 2001, submitted.
- 14 D. Weininger, A. Weininger and J. L. Weininger, *J. Chem. Inf. Comput. Sci.*, 1989, **29**, 97–101.
- 15 P. Ertl and O. Jacob, *J. Mol. Struct. (THEOCHEM)*, 1997, **419**, 113–120.
- 16 For a review of such sites, see: W. A. Warr, *J. Chem. Inf. Comput. Sci.*, 1998, **38**, 966–975. We used <http://www.chemsoc.org/> as the source for identifying sites.
- 17 For mirroring tools, see: J. Langfeldt, <http://langfeldt.net/w3mir/>. For renaming and link renormalising tools, see: <http://www.xlanguage.com/>
- 18 G. V. Gkoutos and H. S. Rzepa, *Internet J. Chem.*, 2000, **3**, article 8.
- 19 G. V. Gkoutos, P. Kenway, P. Murray-Rust, H. S. Rzepa and M. Wright, *Internet J. Chem.*, 2001, **4**, article 5.
- 20 See: M. Kay, <http://saxon.sourceforge.net/>
- 21 See: W. M. Meier, <http://exist.sourceforge.net/>
- 22 See: N. Antonopoulou and A. Shafarenko, *J. Supercomput.*, 2001, **20**, 20, 5–35; G. Aloisio, M. Cafaro and P. Falabella, *Lect. Notes Comput. Sci.*, 2000, **1823**, 32–40 and <http://www.globus.org/>
- 23 IUPAC Chemical Identifier (IChI) Project: chair A. McNaught, *Chem. Int.*, 2001, **23**, issue 3. For details, see: <http://www.iupac.org/projects/2000/2000-025-1-050.html>
- 24 H. S. Rzepa and P. Murray-Rust, *Learned Publishing*, 2001, **14**, 177.