Beyond the Hyperactive Molecule: Search, Salvage and Visualization of Chemical Information from the Internet

Wolf-Dietrich Ihlenfeldt, Johann Gasteiger

Computer Chemistry Center
University of Erlangen-Nürnberg, Nägelsbachstraße 25, D-91052 Erlangen, Germany

Abstract

The established exchange mechanisms for chemical information are under attack from new information distribution channels on the Internet. Increasingly chemical information is distributed by means of WWW pages and similar media. However, most of this information is still primarily intended for human browsing. The search for chemical information and the reuse of encoded structures and their attached data is still complicated and often impossible because of the unorganized structure of the information and the lack of chemistry-aware tools for search, display and salvage of chemical information which help the chemist on the extraction of reusable information from Webspace. The situation is complicated by the lack of standards and formats powerful enough to encode in computer-readable form sophisticated chemical information and informational relationships. The rapid evolution of information exchange mechanisms on the Internet is another problem. The unsettled situation demands a new generation of chemistry-aware tools for information retrieval from the Net. These tools must be capable of adapting to new trends and information models as well as new information types without constant redesign and should be themselves extendable and updatable by components distributed via the same network connections as the chemical data they are supposed to deal with. We introduce a set of tools which encapsulate the established Internet (especially WWW) information transfer and visualization methods and extend them to a new level of information handling for chemistry.

1 Chemical Information on the Internet

The Internet Revolution finally has also hit chemistry. A lot of chemical information can be found nowadays on the Net: descriptions of universities[1], research groups, projects, on-line publications and journals[2], some of them already published no longer on paper, and also hard chemical information, i.e. structures, data and results of experiments and computations. The latter type of information is the most interesting part in the ever increasing flood of information, because it bears, at least in theory, the promise of reusability. For example, it is desirable and useful to extract a molecular data set from a Web page which has been subjected to a certain set of computations in order to include it in another study, which relies on different methods, or to use the original results to generate another visualization focussing on an aspect possibly neglected in the original study, thus gaining new insights from closely looking at old results. However, these ideas must be considered largely utopic at the current time because of the structure of the chemical information predominant on the Web. A MPEG video film visualizing a molecular dynamics run or a VRML (Virtual Reality Modelling Language) scene showing a receptor channel[3] is essentially dead information, intended only for human consumption, and impossible to reinsert automatically or manually into the flow of a new computation, because nowhere the underlying chemical structures are coded in a way which allows an extraction of the structural information of the input data or the parameters and programs used to generate the beautiful pictures. From this standpoint, multimedia objects such as static rendered images, video, audio soundbites or VRML scenes are no progress at all in the sector of information reuse in chemistry. If structural information is provided on the Web, two of the most simplistic semi-standardized file formats are predominant (pdb and xyz), which can only encode a bare minimum of structural information and are incapable of storing any information in computer-readable form beyond a very limited statically defined set - xyz does not even allow multiple comment lines.

The concept of *hyperactive molecules*[4] is currently one of the centers of interest. Basically it simply means that one (only one - typical Web browsers do not yet allow easy run-time switching of MIME display tools) standard structure visualization program such as *Rasmol* or *Xmol* is assigned to files with certain extensions, following the

MIME (Multipurpose Internet Mail Extensions) standards. Attempts to define MIME standards for chemical data are underway[5]. For each structure file retrieved from the server, another such display tool is started. The capabilities of the displays may allow the saving of the imported files, but not the facile combination of multiple structures into a dataset or grouping within more structured hierarchical organizations. A hyperactive molecule is better than nothing - because it is the first established mechanism to couple true structural information with texts describing what has been done with the molecule - but we consider it a short-sighted ad-hoc mechanism, because it completely isolates single molecules (or molecular ensembles) from the global dataset and context they are part of, and once such a molecule has been retrieved, its connection to the text pages and auxiliary information is completely lost.

From these considerations, the informational situation on the Net with regard to the reusability of data could be classified as abysmal as far as chemistry is concerned. One cannot reasonably hope to change the way information is provided, but it should be possible to provide a set of extensible tools which help the chemist to extract as much reusable information as possible from the Net and which aid in the coherent and flexible transfer of structural information to new computational and visualization setups.

In our opinion, the future of chemical information should look like this: a continuous and uninterrupted flow of structural and other information of unlimited type and variety, originating from experiments and computations, pouring into documents, datasets, and visual or other multimedia objects, accessible world-wide by means of the Internet. From these objects the information can be salvaged, with minimal loss of information, and sent back for reevaluation into comparison, computation and visualization tools and programs. The tools and mechanisms we are introducing in this paper are intended to be a first step in this direction.

2 Concepts

In order to demonstrate the feasibility of information salvage for chemical problems from Internet sources, we decided to implement a set of tools which help to recover and organize chemical information on WWW. A number of principal design issues were discussed and decided upon:

- Realism. No control on the established and evolving information provision mechanisms [6] is possible or intended. The goal of the project is to reclaim existing information, not to educate the scientific community to provide better structure in their contributed data.
- Continuity. No isolation from existing tools and techniques. Established tools such as Web browsers, molecular graphics programs and VRML scene browsers remain usable, both as stand-alone tools and in the context of our tool set. In the latter case these standard tools are often controlled by our tools and act as auxiliary helper applications. It really does not make sense to reimplement integrated HTML browser modules if normal Web browsers can be used as external helper applications.
- Encapsulation. The established path of browsing Web information is still supported, but we add another layer on top of it. The basically independent information tidbits attached to some master page are reorganized to become a hierarchical, classified entity which mirrors the logical structure of the described situation.
- Openness. No fixed, exhaustive set of possible information or information types can be listed. New types of data of interest to chemical systems are constantly introduced and made accessible by some network channel. The structure underlying the tool set must be continuously extensible to cope with new types of data and new formats to encode it, and, for example regarding the wild variety of chemical structure file formats, extensibility to read the same information in multiple formats is highly desirable. It is not necessary for the system to handle all information by itself, but mechanisms to identify, manage and encapsulate opaque data which is of meaning only to specific external programs must be provided.
- Properties. Information of all kind is coded as properties attached to a hierarchy of chemical information objects, such as data sets, reactions, molecules, molecular ensembles, atom groups, atoms, bonds and so on. A HTML text page for example might become a property of the structure data set which it links to, a molecular dynamics video a property of a molecular ensemble, and a VRML scene a property of a molecule. This architecture is open enough to accept nearly arbitrary data but sufficiently coherent to provide an organized, generalized, abstract view on the whole. The described chemical information objects are built by accessing, loading and merging the diverse resources available on the Internet. Due to the unorganized content structure, this process cannot be completely automatical. It is certainly possible to collect all chemical structure files referred from a HTML page into a flat data set, but linking molecules from separate files into a sensible sequence of reactions as described in the text page, augmented by an image displaying a crucial transition state, is beyond the scope of automatization with the currently predominant information encoding style for such data. This functionality must rely on human intervention until better data organization mechanisms than isolated structure files become accepted.
- Applets. The general trend in program design is toward small and modular programs which do not claim to be able to handle every conceivable task but are good at exchanging data with other applications bidirectionally, providing

the possibility for synergism. Therefore, tools which collect data, analyze data relationships, search for specific information etc. should not be implemented as a monolithic systems but rather as a tool suite with a powerful data exchange protocol.

• Scriptability. It is very advantageous to have one or a small number of core applications which can be easily adapted by scripts to perform different task within their domain. Additionally, scripts are more portable and can be easily distributed by the same mechanisms as the chemical data we are dealing with, so quick updates and custom tools are feasible. When scripts are treated as MIME information, standard Web browsers can invoke the same standard tools with different scripts on demand, starting with the URL (the Uniform Resource Locator, which identifies the access path to a Webspace object) of the script. Scripts can either be complete applications or configuration information for standard scripts. This makes it for example possible to generate new types of displays configured to highlight a specific feature of interest in a data set. The script either already contains the chemical information in encapsulated form or it downloads it autonomously after it is started.

The distribution of applets as scripts running on a standard interpreter via network channels is an idea currently pursued by a number of institutions and companies ([7], best known Sun and its Java project [8], other projects such as Candleweb [9] exist), but none of these projects provides the chemical awareness and feature set needed for our purposes.

3 Implementation Framework

Our tool suite was implemented around the open chemical data management library of the CACTVS system [10] and Tcl/Tk as scripting language. It consists of two separate parts: a small number of upward-compatible generic data manager programs (the first part) which are driven by a larger collection of scripts (the second part). Every script, which implements a specific tool, can run on manager programs with a script-specific minimal set of features and also unchanged on all more powerful, but also larger and more unwieldy manager programs of the series.

The CACTVS library provides the framework for the storage, import, export and management of chemical information of arbitrary type. It also includes the necessary RPC data exchange modules which facilitate inter-tool communication. Another important element are open xdr-encoded binary and ASCII file formats which can represent all information possibly kept in memory by the library, so data archival is always possible. The library is dynamically extensible in a number of key areas: basic data types, I/O modules for chemical structure files in different formats, property definitions for data classifications, filter modules to describe the validity range of properties, and property computation modules. All these extensions can be loaded at runtime from files or from databases on the network. Therefore it is possible to properly decode files which contain data of hitherto unencountered type and description if the necessary property definitions and data management modules can be located and downloaded in the decoder. The same is true for the I/O of structures and other data from files with formats not covered by the built-in modules. All custom data handling modules as well as structures with all attached data can be stored in databases accessible from the Internet.

The interpreted, string-oriented scripting language Tcl and its GUI/graphics complement Tk [11] are quickly becoming a widely adopted standard for the scripting and GUI parts of complex applications. Our programs do not only contain the standard Tcl/Tk distribution but also a number of popular public-domain extensions and, most important, interface commands to the functions of the CACTVS library. All information stored by the CACTVS data manager can be accessed and manipulated as Tcl-compatible strings and lists and is therefore usable on the scripting level for any data analysis or display task. The graphical capabilities of the system include a full-featured set of GUI widgets, 2D molecule plots and statistics-type graphs plus image display facilities. An important element currently lacking are built-in 3D molecular graphics, but see section 6 for future plans. However, the use of standard molecular graphics programs as slaves is always possible. Internet protocols such as HTTP are easily implemented as script procedures once the language provides basic socket and server commands. Until now, we have not found it necessary to use a protocol library such as CCL[12], although we are contemplating this step and some important Internet protocols (most notable gopher) are currently unsupported in our tools.

We have used these libraries and toolkits to implement a set of general-purpose script-driven data manager programs with increasing functionality. These programs are extended along orthogonal feature addition axes, starting from a common core. One of the capability axis is increased scripting and graphics support: from Tcl without graphics via standard Tk to Tk with extensions such as GL widgets (but not yet full molecular graphics support). Another axis is the inclusion or omission of networked database support. Finally, for distribution purposes, versions with more or less support libraries linked as dynamic objects are available. All these programs share the same scripting language subsets according to the included functionality. Adding database or GL support adds new commands or subcommands, but the rest of the language used to control the program remains unchanged.

4 Application Examples

In this paper we will discuss three examples of tools which facilitate the access to chemical information on the Internet in novel ways, and we also demonstrate how these tools can themselves be distributed on the same channels. All tools are relatively small scripts (none of them exceeding 1000 lines of Tcl/Tk code) implemented in a few days. They follow the interoperation models of the CACTVS tools[13] and therefore can collaborate with tools from this collection (such as the structure editor or the 2D structure browsers).

According to our experiences, the productivity when implementing new services using one of the standard data managers and implementing the rest of the features as script outperforms classical programming by at least one order of magnitude. Of course this ratio declines if, for reasons of necessity or due to performance considerations, the generic data manager programs need to be enhanced. Due to their complex internal structure, changes can become significantly more complicated than with programs with a clean single-purpose layout. However, this step is necessary only very infrequently.

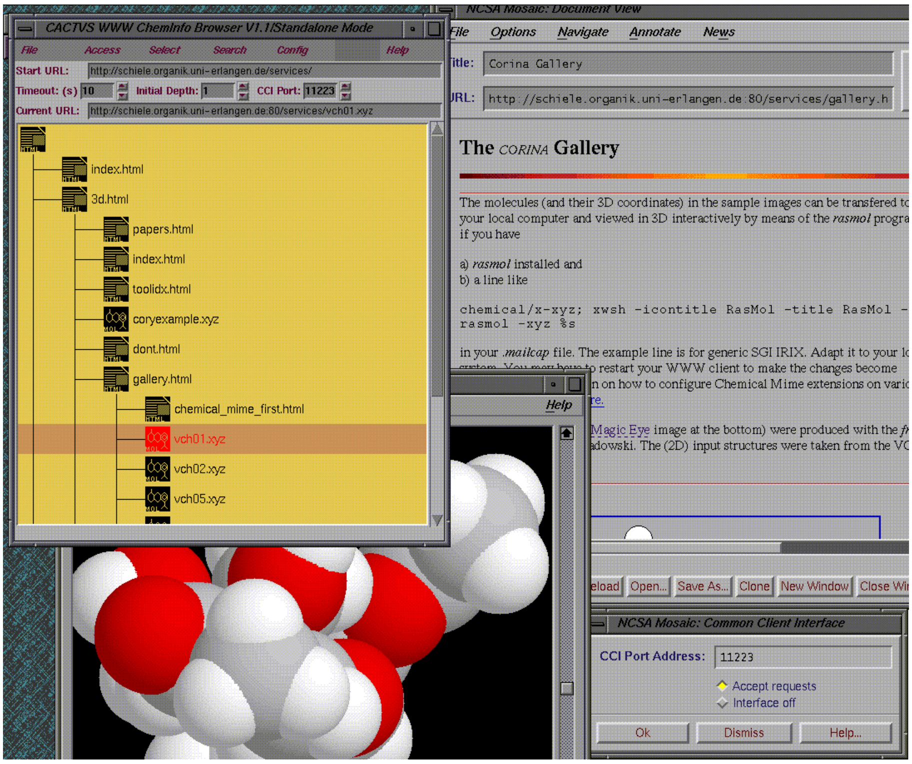

Figure 1. Web Information Access Tool, using *Mosaic* and *Rasmol* as helper applications

4.1 Chemical Web Information Access Tool

The first example presented in this paper is also the simplest, but it demonstrates some of the concepts and mechanisms which make the tool suite flexible and strong in the interaction aspect.

Under WWW, chemical structures are necessarily leaf nodes. They are referred from HTML pages, but cannot contain themselves references in forward or backward direction if stored in any of the popular file formats[14]. Displaying the referencing pattern, i.e. the links leading to the structure files, and providing functionality to extract and collect larger data sets which share a common context is a valuable and often requested feature, but very awkward to do with standard Web browsers.

We have implemented a tool (Figure 1) which takes URLs as input and analyses the type, content and embedded links of the indicated file. HTML pages can be further expanded, and different icons represent the various linked information, including files with molecular structures. The icons representing the Webspace objects can be subjected to the standard operations available within Web browsers. HTML pages are viewable using *Mosaic* as a slave, and structure files can be exported to the standard MIME hyperactive molecule viewers. *Mosaic* is preferred because of its CCI interface [15] - multiple pages can be displayed without spawning multiple viewers, and *Mosaic* talks back to the access tool, allowing navigation from within the viewer. Note that here the normal control hierarchy is inverted: *Mosaic* is used as a slave, performing selected services on behalf of the access tool.

However, besides the advantage of showing the link structure around the entry point, the tools offers a number of functions not found in traditional Web browsers. First, multiple information sources can be linked. In the simplest case it means that structures referred from a page are collected to form data sets. With human intervention, more complex data interpretation and preparation is possible, for example by attaching related HTML pages, images, videos, VRML scenes etc. found in the link vicinity of a page to a pair of molecules which were identified as starting material and reaction product respectively of a reaction mentioned in a text page. These molecules are merged into a reaction object, building a structure of higher information level than the plain molecules. This enhanced information can be stored in suitable files (for example, without the auxiliary data, as MDL reaction file) for later input in databases etc. Furthermore, molecules (the most elementary transferable unit) as well as enhanced objects such as reactions, including all their auxiliary data can be directly exported to a CACTVS system workbench. To the workbench the tool appears as a structure generator, depicted with its icon on the workbench plane, and is, in most aspects, indistinguishable from other structure generation tools in the system. Structures obtained from this source can be passed on to other computation, data analysis, visualization, filtering and I/O stations defined in the visual programming environment of CACTVS, including other external tools which accept structures flowing in. This tool is no exception: if a structure is imported into it by dropping it on the iconic workbench representation of the tool, and the structure, as part of its data load, has a valid link environment property, the tool expands the reference tree and offers its processing and viewing options for the remembered link collection. With this mechanism, chemical objects imported in formats which have some means of encoding a link environment (see section 4.2 for an example of such a source) can unfold their context, enabling for example the human researcher to read the web pages this structure was attached to.

4.2 Substructure Search on WWW

Searching information on the Web is notoriously difficult. Text-oriented search engines such as Lycos[16] have recently become publicly accessible, but they do not provide the functionality chemists use to retrieve structure information. No text-oriented search engine can perform substructure, full structure or formula searches. Furthermore, due to the link-oriented nature of the Web, the most interesting information is probably not the structure itself, but the document one or two levels upward where they are referenced from and which describe the context. However, WWW links are unidirectional (in contrast for example to Hyper-G [17] links) and cannot be traversed in reverse direction, starting from the structure file, which definitely does not contain any backward links. Providing the classical structure-oriented search functionality in Webspace for the chemical community is desirable, but not trivial, and includes significant preprocessing and information hunting.

The actual implementation of substructure, full structure and formula searches was easier than expected by means of a combination of a CACTVS system structure database, two maintenance scripts and a graphical query tool.

Two continuously running processes are gathering and verifying structure information on the Web. The gatherer,

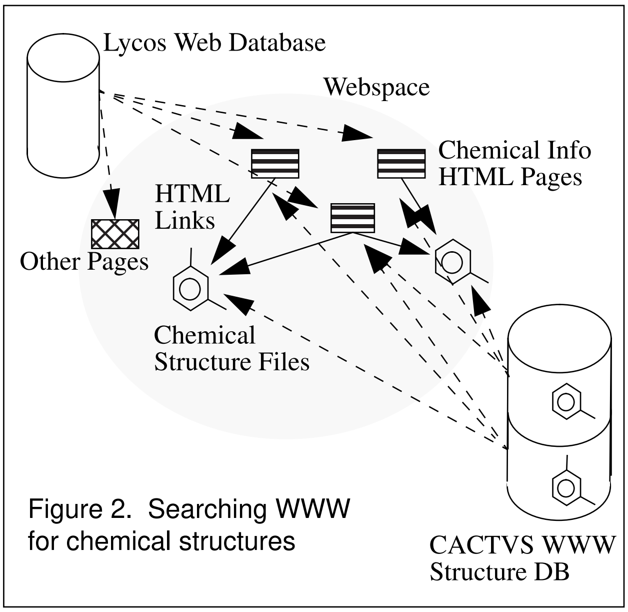

Figure 2. Searching WWW for chemical structures

a kind of Web spider, is implemented as short (a few hundred lines) script executed by a non-graphical data manager. It systematically expands a list of start URLs up to a certain depth (adjustable, currently 2 levels), trying to find chemical structure files. These are primarily identified by their filename endings and verified to be syntactically correct by tentatively reading them and keeping the structure in a local file for further processing if the read succeeded. During the scan, all links leading to each structure file are registered and become part of the data attached to the structure. The start URLs are obtained from a Lycos search on the full catalog (currently about 6 million pages) with general chemistry-oriented keywords such as ‚chemistry‘ or ‚molecule‘. The currently used URL start list obtained from these queries has about 10.000 entries. After finishing a scan, the found and successfully converted structures and their link history are stored in a database. Typically, structures harvested from the Web require extensive postprocessing to render them searchable. For example in the case of xyz and most pdb files, hydrogens and charges need to be guessed, connectivity must be reconstructed and the final detection of aromatic systems and multiple bonds asks for sophisticated algorithms. The second maintenance process continuously checks the database entries, verifying that the location pointed to as well as the links in the link origin fields are still accessible and the structure in the remote file is still the same (Figure 2).

The access to this database mandates a specialized access tool. While it might be possible to use a forms-based interface, probably using SMILES strings for elementary substructures and structure search operations, this is certainly inconvenient. The transfer of hit lists as dynamically generated HTML pages with links to the referring pages and the database entry, probably. aug-

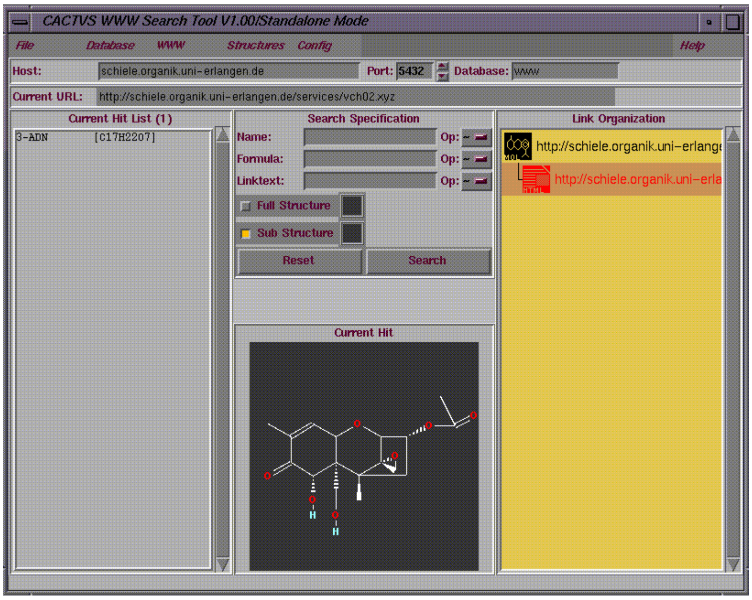

Figure 3. WWW structure search tool

mented with 2D images generated on the fly, is likewise complicated and stress for our server. Structures retrieved this way would end up at the site of the query poser in a standard display tool or as cumbersome single-structure ASCII text files. Instead, we choose to provide a free access tool consisting again of a short Tcl script, running on a standard data manager program. The tool as a stand-alone program provides basic functionality such as interactive search composition, display of hitlists, display of selected structures from hitlists, visualization of the link tree leading to the structures etc. Furthermore, it interacts with other tools from our collection by drag&drop operations: A structure editor can be used to drop substructure search fragments and full structures; and the structure search tool can export link trees to the Web information access tool described in the previous section. Additionally, this tool can, just as the previous and next one, become a slave of a CACTVS workbench and pump its information load into processing pipes assembled on a workbench.

This combination of three tools, two of them not graphical, implements a fully operational structure search system for chemical information in WWW and provides a previously not available type of information access to the chemical community. The flexibility of the scripting/manager concept proved to be very valuable for a time-effective implementation of the concept.

4.3 Computed NMR Shifts Archives

The computation and subsequent analysis of geometry and shieldings of NMR-active nuclei by ab-initio methods is a rapidly expanding field. One of the problems is the duplication of efforts because no exchange mechanism for the computational results exists. The situation is complicated by the fact that some potential participants in an exchange scheme insist on multi-step information disclosure levels, i.e. it was demanded that the fact that computations have been performed on specific molecules with certain methods, programs and basis sets is made public, but without prematurely revealing the actual numerical results except to selected authorized collaborators.

A central network-accessible database called Sharc/Shigo was set up at our site[18] as a repository for ^{31}P NMR data. The setup and maintenance of the database currently requires a full CACTVS system setup, and all supported query operations can also be issued from within CACTVS. However, this is too complicated and beyond the scope of the casual user who just wants to be informed whether some structure has already been computed. Consequently, a custom access and query tool which is simple enough to be usable without training, was once more implemented as a script running on the generic data manager program with graphics and database support. The manager program is distributed by anonymous ftp, and the most up-to-date scripts which implement the access tool and a 2D structure editor (useful in order to define fragments for substructure searches) are available from the Sharc/Shigo home page [19]. Provided the local MIME configuration is correct and the data manager program installed, clicking on a link on the Sharc homepage will download the newest script and start it up in the correct manager program[20]. As long as the manager program needs not to be updated, new features in the search engine, enhanced information content in the records or improvements in the user interface can be continuously and painlessly made available to the public. Current search methods supported by the query and access tool (Figure 4) include full structure and substructure searches, lookups on certain chemical properties such as total charge, point-group or structure minimization character, including computational history information (author, method etc.). If the record is not restricted in its accessibility, the full data content of any record can be downloaded into the access tool for closer analysis. The Sharc/Shigo browser has built-in 2D structure visualization with annotation, for example for the anisotropic shielding factor assigned to each atom, and a text display which provides a detailed numerical result summary and information about run parameters, applied programs, computers run on, and program performance. The tool transparently computes necessary information not contained in the database records but used in the visualization scripts, such as 2D display coordinates. The tool

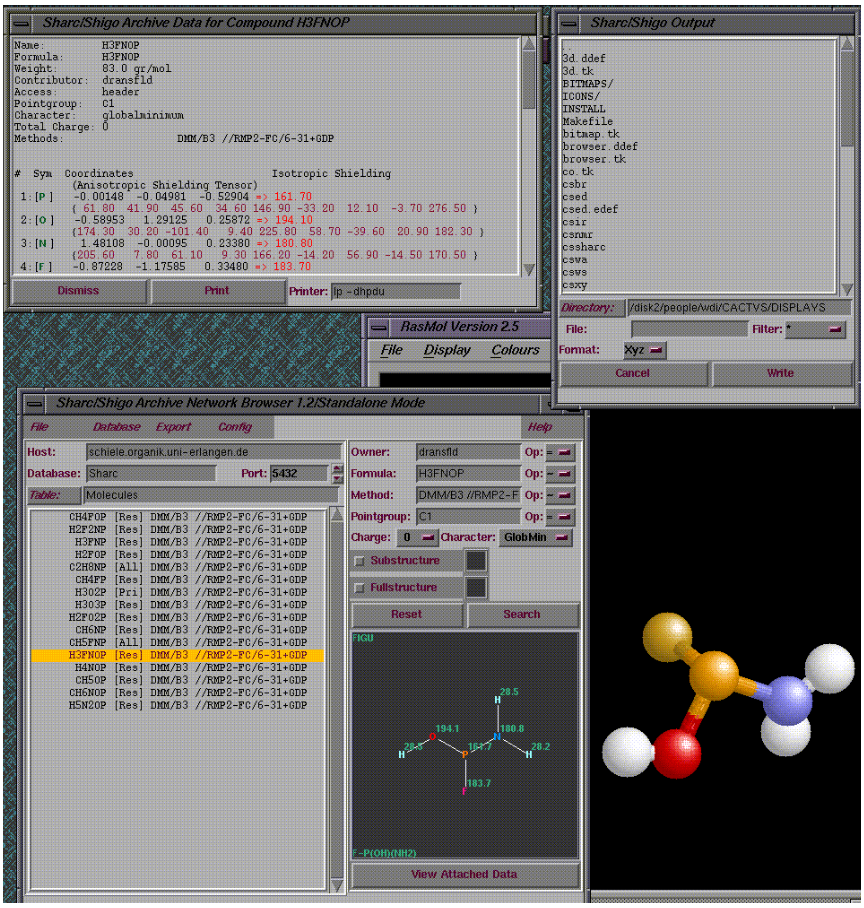

Figure 4. Sharc/Shigo query and retrieval tool in stand-alone mode

interfaces seamlessly to popular 3D molecular graphics programs (*Xmol*, *Rasmol*, *Fm3d*) for three-dimensional structure inspection, providing them with properly formatted input files.

Finally, this tool once more has the capability to operate as a structure source in connection with CACTVS workbenches, so the information retrieved from the archives can be directly analyzed and manipulated within the CACTVS environment. This situation is depicted in Figure 5: a hit list (structures with P-N bonds) is exported to a workbench via the yellow icon representing the access tool. It is the top square in the central column of icons on the workbench in the background. The data set is temporarily stored in an archives object (immediately below the Shigo tool icon). On the workbench, transparently additional information required during the following operations is computed for the imported data set, using the property and computational method database of CACTVS to invoke the necessary procedures to obtain the requested additional information from the data already attached to the imported molecules. Here, first the structures are sent to a 2D display tool, and then, streaming downward, a statistical plot tool is invoked and loaded with the structures in order to examine a possible systematic relationship between the computed shielding value and other structural properties. In this case the ideal of uninterrupted information flow from archives with prior work to an integrated environment for data analysis and submission to new computations has been achieved by the combination of Web pages, MIME-compatible script transfer, generic data manager programs, CACTVS structure databases and CACTVS workbenches. The structures from the archives are well annotated, so the precise nature of the work already performed on a molecular system is sufficiently disclosed.

5 Program Availability

Most of the described tools and support programs can be downloaded free of charge by anonymous ftp or WWW browsers as portable ASCII scripts and (binary) executables for major OS and processor architectures. Program source code is currently not available. Please use

`http://schiele.organik.uni-erlangen.de/cactvs/`

as entry point.

6 Future Developments

Until now the graphical capabilities of the tools in the set are limited to GUI building, statistical plots, 2D-structures with annotations, image display and other visualizations essentially 2D in nature. We plan to incorporate the new Molecular Inventor 3D chemical structure graphics package from Silicon Graphics into the system. Once this project has been completed, dynamically configurable 3D visualizations without loss of the full set of underlying chemical information will become possible. These will neither be precomputed, rigid scenes of arbitrary information content (as nowadays VRML scenes are), nor simple transfers to a standard molecular graphics package (as nowadays hyperactive molecules handling mechanisms proceed), but the transferred script will either define a complete small visualization application script designed for a specific purpose, or provide some standard script running on the standard data manager program with auxiliary scene and annotation hints highlighting the discussion

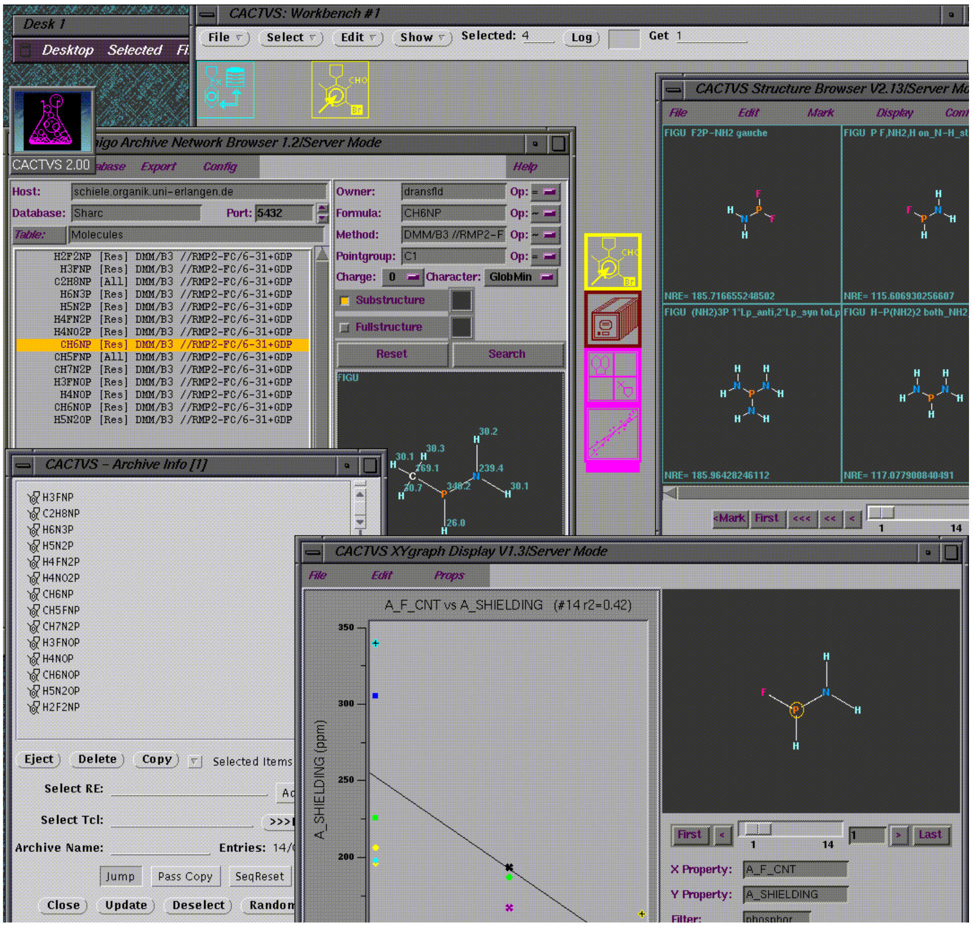

Figure 5. The Sharc/Shigo database access tool working as slave of a CACTVS workbench

points of the author, something which cannot be achieved with simple passing of structures to standard programs as in the case of MIME. In either case, the chemical information will be reusable - annotation and display hints are treated internally just as yet another property in the data load of the molecule, and custom visualization notwithstanding the visualizer will probably continue to support standard molecular graphics operations - which is not possible in VRML scenes because the original structure has been lost and only triangles and other geometric primitives remain.

Another aspect which needs some more work is security. At the moment, our programs do not yet contain security and script integrity check mechanisms. In order to make the use of network-distributed scripts trustworthy, features along these lines must be included.

7 Related Work

The only other attempt known to us with the aim of facilitating the exchange of chemical information on the Net is the Explorer EyeChem project[21] by Rzepa and coworkers. The system works by linking two IRIS explorer programs and two *Mosaic* clients via CCI connections. The structure information is stored as an Explorer pyramid type, optionally converted from a number of quantum chemistry program output formats. The scope of the system, as far as the network aspect is concerned, is limited to collaborative visualization.

A second project trying to make chemical structures searchable on the web is the Molecular Hyperglossary database[22] by Leach et al. from Imperial College, London. This system requires manual registration of entries and does not attempt to localize existing but unregistered information, or to verify its contents. Its advantage is that keywords, molecule names etc. which were input together with the structure and cannot be extracted or generated automatically with reasonable effort and are therefore not handled in our implementation, are searchable as text fields in the Hyperglossary. Substructure or full structure search is currently not supported, although in principal the necessary data is assembled in the database. Only a single relevant URL is returned, no link environment or pointer to the structure file, if the file location is not the relevant URL itself, which it typically is not.

8 Conclusion

We have developed a set of tools which support the reuse of chemical information found scattered in Webspace in a novel manner. With the aid of these tools, valuable structural and other chemistry-related information can be reclaimed, reorganized and resubmitted to further studies including new computations, visualization and comparison with results from other sources. We consider these tools to be a first step toward a continuous flow of computer-readable chemical information on the Internet. Scientists which tap into this stream could utilize chemical information generated by colleagues in an unprecedented manner, reducing ignorance of previous work, duplication of efforts, misunderstanding, data input errors and other effects adversary to the progress of science.

W. D. Ihlenfeldt expresses his thanks to the Fachinformationszentrum Berlin (FIZ) for financial support.

References

[1] Example: http://www.riken.go.jp/index.html
[2] Journal of Computer-Aided Molecular Design: http://wucmd.wustl.edu/jcamd/jcamd.html, Journal of Molecular Modelling: http://www.organik.uni-erlangen.de/info/about-JMOLMOD.html
[3] Chemical videos: http://www.pc.chemie.th-darmstadt.de/molcad/movie.html, Chemical VRML scenes: http://www.pc.chemie.th-darmstadt.de/vrml
[4] P. Camilleri, D. Eggleston, H. S. Rzepa, M. L. Webb, *J. Chem. Soc. Chem. Commun.* 1994, 1135
[5] See http://www.ch.ic.ac.uk/chemime2.html
[6] K. Hughes, From Webspace to Cyberspace, via URL http://www.eit.com/~kevinh/cspace/
[7] P. Wayner, *Byte*, 3/1995, 105-114
[8] Hot Java homepage: http://java.sun.com/ Chemical application examples: http://java.sun.com/applets/applets/ChemicalModels/index.html
[9] Candleweb homepage: http://www.oslonett.no/~candle/
[10] W. D. Ihlenfeldt, Y. Takahashi, H. Abe, Proc. 28th Annual Hawaii Int. Conf. on System Sciences, Vol. V, (1995), 227-236
W. D. Ihlenfeldt, Y. Takahashi, H. Abe, S. Sasaki, *J. Chem. Inf. Comput. Sci.* 1994, **34**, 109
[11] J. K. Ousterhout, Tcl and the Tk Toolkit, Addison-Wesley 1994. The official homepage of the Tcl/Tk team at Sun is http://www.smli.com/research/tcl
[12] CCL is the Common Code Library of Mosaic etc. See http://
[13] See http://schiele.organik.uni-erlangen.de/cactvs/
[14] Work on CML, a Chemical Markup Language, which could include this feature, has begun.
[15] CCI: Common Client Interface, see http://www.ncsa.uiuc.edu/SDG/Software/XMosaic/CCI/cci-spec.html
[16] Information about Lycos can be found at http://lycos.cs.cmu.edu/
[17] Hyper-G homepage: http://hyperg.iicm.tu-graz.ac.at/
[18] A customized postgres95 database engine on port 5432 on host schiele.organik.uni-erlangen.de
[19] http://schiele.organik.uni-erlangen.de/sharc/
[20] Carelessly executing downloaded scripts of **any** type, shell, tcl, or CACTVS tool scripts, is not recommended unless the site of origin is completely trustworthy or the script interpreter has security mechanisms built-in (Secure Tcl, Java or Telescript).
[21] See http://www.ch.ic.ac.uk/rzepa/CG/CG.html
[22] See http://www.ch.ic.ac.uk/ectoc/papers/leach/