The Chemical Sample: A Fundamental Object for Molecular Modeling[†]

George D. Purvis III

CAChe Scientific, Inc., Beaverton, Oregon 97077

Received July 2, 1993[®]

Motivated by a need to provide chemists with intuitive computer-based chemical modeling software that integrates the practice of theory and experiment, we have applied object-oriented analysis to discover fundamental objects and events in chemical research and to define their relationships. In our analysis, the central chemical object is a chemical sample. The canonical chemical sample is a collection of atoms with a state that may change. It is a dynamic object that behaves in chemical modeling software just as a real chemical sample would in the real world. Depending upon its state, a model chemical sample may contain atoms, molecules, crystals, surfaces, clusters, reacting species, and other chemical substances. In this framework, a molecule, or other substance, is simply the name of a collection of atoms and bonds that depends upon their types and positions. We describe a molecular modeling software architecture designed to handle the dynamic structure, properties, and changes in a chemical sample.

INTRODUCTION

For most experimental chemists, computed-based chemistry systems are in competition with existing tried-and-tested laboratory methods. Computer-based systems must prove themselves by increasing productivity. Unfortunately, the invaluable productivity that comes from the occasional breakthrough innovation inspired by computed-based modeling and information systems is often offset by the difficulty of using it. There are many reasons a system is hard to use, but one difficulty is learning a new way of doing research while the old way continues to be used. Computer-based chemistry systems that parallel the way chemists conduct their research leverage centuries of chemistry experience and have an inherent ease-of-use advantage.

In the past decade, computer scientists have developed a powerful paradigm for designing and building software systems which parallel the real world. It is known as the object-oriented method.[1] Applied to the development of computer-based chemistry systems, object-oriented methods naturally lead to systems which are founded on the traditional terminology and methods of chemistry research. The following sections introduce object-oriented methodology, show how to apply it to chemistry research, and present its implications for information storage.

Computer modeling and database systems provide chemists with access to chemical information. Simple access to chemical information expressed in standard, unambiguous, and easily interpreted form[2] promotes its use[3] and accelerates chemical research. Although the perception of simplicity is ultimately in the eye of the user, it is fairly certain that an individual would find access to chemical information simpler if he used a single access method than if he used several access methods and switched among them. Ultimately, user interfaces to modeling and database systems can be expected to merge into a single chemical information interface.

Although various file formats have been proposed as standards for certain types of chemical information,[4–8] today standardization of computer-readable chemistry information in general, and chemical structure information specifically, is almost non-existent. Most research groups adopt their own file formats for their results, while chemistry software vendors invent proprietary data formats and interfaces to the data their programs manipulate.[5,7]

The lack of standardization requires chemists looking for chemical data to learn several different computer interfaces. Since chemists rarely have time to learn more than one interface, information retrieval is often incomplete. Furthermore, the retrived data are frequently in a nonstandard format and must be translated, often with the loss of information, before it can be analyzed by other programs. The lack of standardization complicates access and seriously limits the synergy that could be achieved through the sharing of scientific data.

LEVELS OF INCOMPATIBILITY

The situation is slowly beginning to change as ASTM[9] and IUPAC, two standards setting organizations, begin to develop standards for chemical information. Development of effect standards begins with understanding where current incompatibilities exist. Incompatibilities can appear at the physical, logical, and conceptual level. Obvious incompatibilities are encountered at the physical format level. Differences in physical format can be as basic as the use of different delimiters between data fields or a different ordering of the data.

The logical organization of the data, or so-called information model, can also introduce incompatibilities. For example, in one information model, atoms and bonds might be represented as rows in different tables connected by relations from a third table. In a second information model, atoms might be represented as rows in a table with bonds implied by a list of attached atoms. This second model is built upon the assumption that bonds are always between atoms, a serious problem for the chemist who needs to store the structure of a compound containing π-coordination bonds (a bond between a metal and a π-bond) or topological bonds.[10]

Information model incompatibility can also arise when different data types are used for the same information. For example, elements are sometimes identified by atomic symbol, other times by atomic number. Because the atomic symbol and atomic number are not completely equivalent, information can be lost in a conversion. A similar incompatibility occurs with atom coordinates which can be Cartesian, internal (z-matrix), or fractional.

[†] Presented at the 3rd International Conference on Chemical Structures, June 6–10, 1993, Noordwijkerhout, The Netherlands.

[®] Abstract published in *Advance ACS Abstracts,* January 15, 1994.

0095-2338/94/1634-0017$04.50/0

A more subtle incompatibility arises at a conceptual level when there is basic disagreement about the meanings associated with the terms and ideas of chemistry. For example, is a molecule a set of bonded atoms or is it the smallest component of a substance which can be separated by ordinary physical means? Does the term "molecule" indicate a single conformation, or a set of conformations the atoms and bonds could assume? Because experimentally measured molecular properties are averages over molecular geometries, the separation of conformations into different molecule files within an information system may confuse the experimental chemist who may not understand where to find the property value equivalent to the one he measures.

Effective standardization should begin with agreement at the conceptual level. It is pointless to discuss the physical or logical compatibility of data if there is fundamental disagreement on the meaning or interpretation of that data. Few formal attempts to review the high-level information needs of chemists have been made, and published references are negligible.[2] Therefore, we have used an object-oriented analysis to abstract the information needs and fundamental terms of chemistry. Our purpose is to share this analysis so that it can be discussed and a consensus established.

As discussed in the next section, object-oriented analysis begins by identifying and defining the fundamental "physical" objects that are part of chemistry research. Then basic relationships between objects are listed and diagrammed, resulting in a conceptual model of chemistry research outlined in the fourth section. One of these entities, the chemical sample, seems to be particularly appropriate for the archiving and accessing of chemical structure information. The final section details the conceptual organization of the chemical sample.

OBJECT-ORIENTED ANALYSIS

The following succinct description of object-oriented development is provided so that its application in the next section can be followed. Comprehensive descriptions can be found in several good references.[1,11,12]

The object-oriented paradigm is based upon the observation that explaining the way something works is simplest when we first identify its important parts and their behavior before we attempt to describe the processes by which the component parts work together. In the object-oriented paradigm, each part is an object. Objects are entities that can exist on their own and be components of other objects. They are nouns: things or concepts. They have attributes (color, texture, parts, state, etc.) as well as behavior and responsibilities. For example, a metal rod may be an important object in a metallurgical experiment. It has the attributes of length, radius, and mass. Also, it may have the behavior of glowing white when heated or vibrating when struck. It may have the responsibility of providing a rigid support for other parts.

Objects also have important relationships with other objects. Three classes of relationships have been identified. An object that has another object as a component part is said to have a "has-a" relationship with the component. Some objects are specializations of more general objects. For example, an alkane is a special kind of organic molecule. Such specialized objects have an "is-a" relationship with the more general object. Finally, some objects exchange messages with other objects to acquire data or to have a task performed. Such objects "talk-to" each other.

When used to develop computer software, object-oriented methodology is applied to each of the three stages in the software development cycle: analysis, design, and programming. Each development cycle begins with analysis and ends, after several iterations, with the final programming.

During the object-oriented analysis stage, a list of the results required by the software user and the objects which that user will employ when working with the software is produced. The attributes, behaviors, and responsibilities of each object are identified as well as its relationships to other objects. The objects found in a good object-oriented analysis are entirely independent of ultimate implementation in a computer program and the same analysis could be used to obtain the required results manually or by another method. As a result, a good analysis is easily understood by the intended user.

The next stage, object-oriented software design, augments the list of objects with those necessary for obtaining the required results on a computer. Specification of the user interface and implementation-specific objects such as network messages appear at this stage.

Finally, object-oriented programming implements the designed objects. Often it uses one of the object-oriented programming languages (Smalltalk or C^{++}), and sometimes it uses an object-oriented database for persistent storage of the objects. Inside a program, object attributes are simply data while behavior and responsibilities are functions. Thus, an object in a program is simply a combination of data and code. Object-oriented languages bind data and code together to make programming with objects easier. The has-a relationship is implemented when one object contains a second object as data. The talks-to relationship is implemented when one object calls a function in another object. One of the great advantages of object-oriented languages lies in the implementation of the is-a relationship. The is-a relationship is supported directly using a mechanism called inheritance. Inheritance allows objects of the same general type to share common data and code. Inheritance supports the reuse of existing objects without modification of the original objects.

Given a sound analysis, software design and programming can proceed systematically to completion using sound software engineering practices. Since the software implementation is a model of an existing physical system, it can be expected to work correctly.

The success of the resulting software is derived largely from the accuracy with which the analysis defines the required results and objects employed by the user to achieve them. As important as correct analysis is to the development of successful software, even good initial analyses are apt to be incomplete. When the original analysis is based upon the important physical objects, completion of the analysis usually adds new objects, attributes, and responsibilities. The resulting design and implementation changes rarely lead to significant rework that requires a complete revamping of the system architecture.

To ensure that the right objects are included in an analysis, the analysis should begin at a abstract, high level and be refined to a detailed lower level. For example, if we seek to integrate information about molecular structures into a system to help chemists conduct chemical research, we should begin by analyzing chemical research and discover where structural information naturally appears.

A CHEMISTRY MODEL

Using object-oriented analysis, we have build a high-level object model of chemistry research by following this recipe:

1. Write a short, accurate description of chemistry research
2. List the important nouns; these are the objects
3. List the verbs; these are object methods and relationships

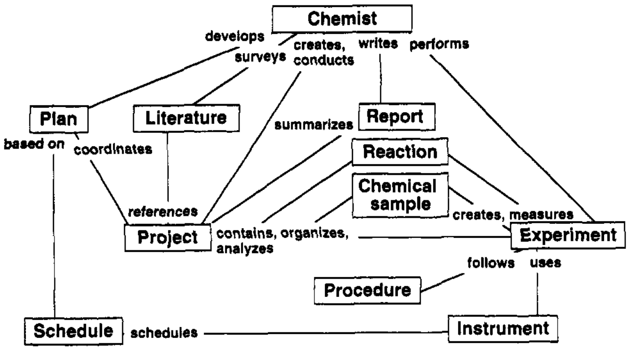


Figure 1. Initial conceptual model of chemistry. Boxes contain objects, and lines connect related objects.

4. Continue to
   a. write a short description (definition) for each object and
   b. add new important nouns, or verbs, to the list of objects or their methods and relationships
5. Diagram the objects and relationships, using boxes for the objects and lines for relationships
6. Test the model by running scenarios
7. Enhance the object descriptions by completing the lists of attributes, responsibilities, and relationships

Mindful that "*the words that are used in describing nature, which is itself complex, may not be capable of precise definition*",[13] the analysis began with the following statements about chemistry research:

> To solve a problem, achieve a goal or follow scientific interests, research chemists *create* and *conduct* research projects that *contain*, *organize*, and *analyze* selected results of experiments that measure or create chemical samples and reactions. Chemists *perform* experiments that *use scheduled* instruments and *follow* steps in the experimental procedure. *Based on* current schedules, chemists *develop* plans that *coordinate* their work on projects.
>
> Projects build on previous work and *reference* literature the chemist *surveys*. Chemists *write* scientific reports that *summarize* project status and results.

A total of 12 important nouns is underlined: chemist, project, experiment, chemical sample, reaction, result, literature, plan, procedure, instrument, schedule, and report. Each of these is an important noun that is shown inside a box in Figure 1. The italicized verbs indicate important relationships that are diagrammed by placing them on a line connecting the related object. The verb is placed closest to its subject. At this stage, the conceptual model is incomplete, but useful. As additional descriptions are developed, more objects and relationships are identified. A detailed high-level diagram of chemistry research developed by the CORA group is available in a technical report.[14,15]

The objects that are diagrammed in Figure 1 are fundamental entities in chemistry. These fundamental objects are independent of the existence of computers. Chemists in the 1940's easily could have recognized these objects and described each in detail. At this abstract level, chemical research has remained essentially unchanged for the past hundred years. In spite of the continuing advances in theoretical and computational chemistry, our experience of the past decade suggests that this framework is unlikely to change significantly in the next decade.

THE CHEMICAL SAMPLE

Chemists experiment with chemical samples. Sometimes the samples are substances (A substance is homogeneous matter with reasonable definite chemical composition. It differs from a material in that it is "pure".[13]) or pure compounds. Other times, they contain more than one compound and have properties and composition that change as the sample progresses through an experiment. Chemists need to understand the structure and properties of chemical samples and how samples can change during an experiment. Experimentally, they obtain information about the sample by measuring its properties.

Chemistry database and modeling systems are based either on molecules (pure compounds or substances) or reactions. Molecular modelers and information specialists may not understand that this is a problem. Yet few chemists experiment solely with pure compounds. Thus, experimental chemists must learn to extract information about molecules, or reactions, and then interpret how that information applies to the chemical samples in their experiments.

To some extent, molecular modeling systems have attempted to accommodate this need by extending the definition of "molecule" to include interacting molecules. The confusion that results from having a molecule file or molecule record contain several molecules—or even a chemical reaction—is apparent.

The object-oriented method implies that computer-based systems should be implemented by software modules that mimic the attributes and behavior of physical objects. Applying that principle here means that the interface to information about matter should be through "measurements" or queries of the chemical sample.

Since sample-centered systems have not been built, it is worthwhile to analyze the chemical sample. To this end, we begin with a list of statements about the chemical sample:

A chemical sample is the part of the universe that is the subject of chemical study. Each sample *has a* name and a current state.

A sample *contains* matter (a set of interacting atoms and other components) and may undergo a chemical or physical transformation to a new state.

A chemical sample's state is sufficient for fixing the properties of the sample. A sample's macroscopic (thermodynamic) state is a distribution of its microscopic states. Depending upon the purpose in mind, various macroscopic state variables, such as its pressure, temperature, and volume, as well as microscopic state variables, such as mass, number of atoms, atom positions, energy, time, etc., may be used to specify its state.

A temporal sequence of sample states is called a trajectory and a collection of states ordered according to a range of geometrical coordinates or other internal descriptors is called a map. Under appropriate experimental conditions, a sample can *produce* trajectories or maps.

A closed chemical sample is one in which no matter can enter or escape. A sample from which matter can escape is an open chemical sample.

Thirteen important objects are underlined: sample, name, current state, macroscopic state, distribution, microscopic state, state variables, property, atom, component, range, trajectory, and map. There are two kinds of samples: open and closed. Figure 2 uses an exploded parts diagram to show the important entities and their relationships of a chemical sample.

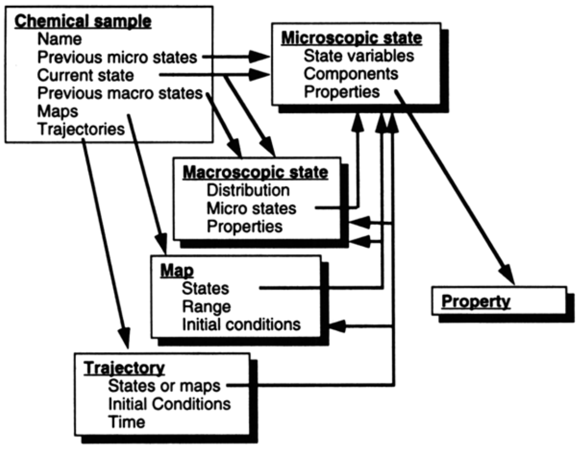


Figure 2. Diagram of a chemical sample. Arrows point to the parts of a composite object and branched arrows indicate that a subcomponent could be one of several different kinds of objects. Drop shadowed boxes denote sets of objects.

There are also methods associated with a chemical sample. A few of these methods are listed as follows:

a. Change the current state to a new state.
b. Get the description of the current state
c. Has the sample been in this state before?
d. Which state in this sample is the same as the current state in a comparison sample?
e. Pour another sample into this sample.
f. Extract a new sample from this sample.
g. View the sample in its current state.
h. View the specified trajectory.
i. View the specified map.
j. Measure a specific property of a specific state.

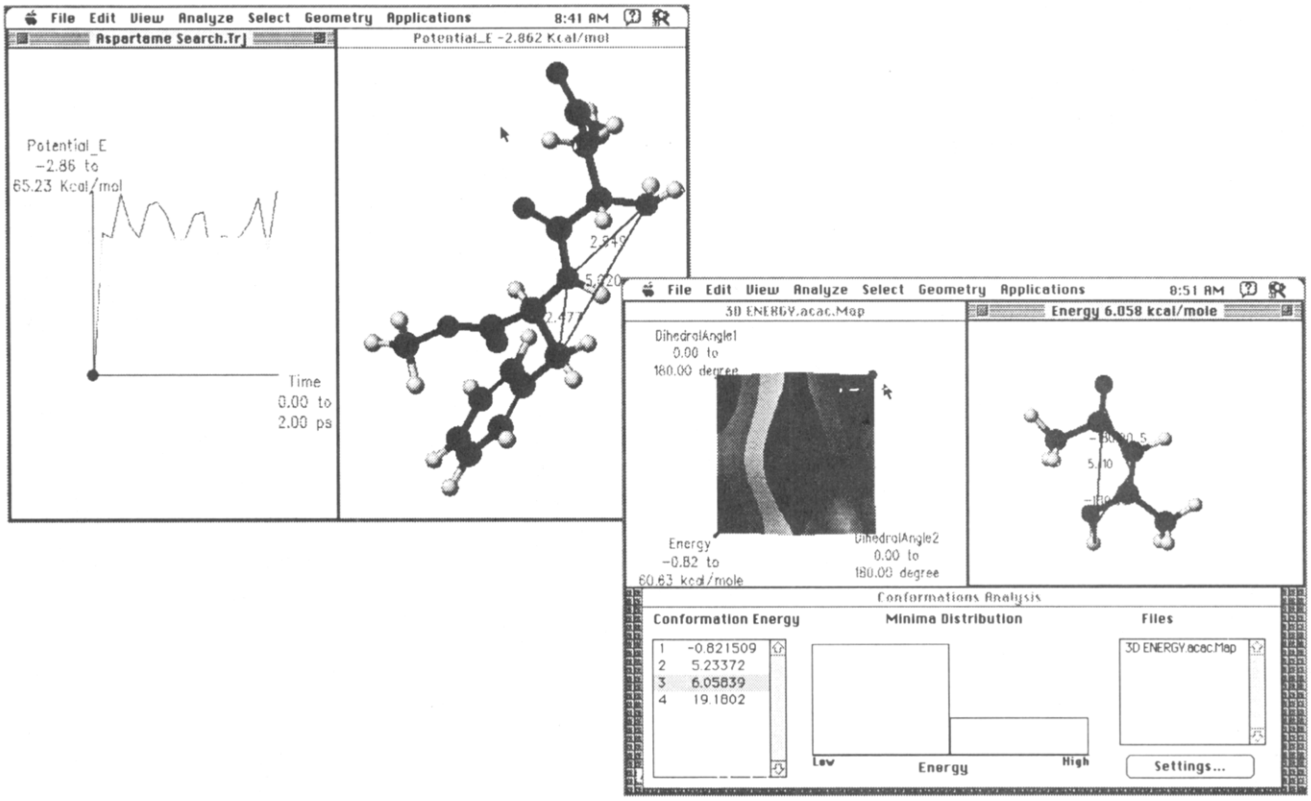
Figure 2 shows that a chemical sample is associated with many states. It has a current state and may have been in a number of previous states. Exactly how the sample changes from one state to another depends upon the implementation and is not prescribed by the object-oriented analysis. Implemented as a component of a database system, a chemical sample might store one list of previous macroscopic states and another list of previous microscopic states. The current state would simply be one of the states in the list of states. On the other hand, a chemical sample in a modeling system might calculate a new state when the state is changed.

Because the structure of an object is hidden from the user, the chemist may be unable to distinguish between using a database system and using a modeling system. We believe that this is in fact the way it should be. A user benefits when a single interface is needed for modeling or information retrieval.

Although it is not formally part of the analysis, it seems valuable to point out that the states associated with a chemical sample can be accessed simply. As shown in Figure 3, states can be selected from a list. Alternatively, they can be selected from a graph by rolling a ball along a trajectory or across a map. The selection mechanism applies to both macroscopic and microscopic states since both maps and trajectories contain either kind of state.

Figure 2's description of a chemical sample omits mention of molecules, bonds, mixtures, and other chemical components customarily associated with chemical samples. Clearly elaboration is required. We continue by elaborating on microscopic states.

> Chemists analyze chemical samples to identify their current state, properties, and components. Sample components include single atoms, bonds, molecules, radicals, polymers, formulations, crystals, van der waals complexes, mixtures, substances, functional groups, proteins, etc. Components are identified by names which may be systematic or common. The composition of a sample



```
1   -0.821509
2    5.23372
3    6.05839
4    19.1802
```

Figure 3. States of a chemical sample can be easily navigated by picking from a list (lower right) or rolling a ball on an energy map (center) or along a trajectory profile (upper left).

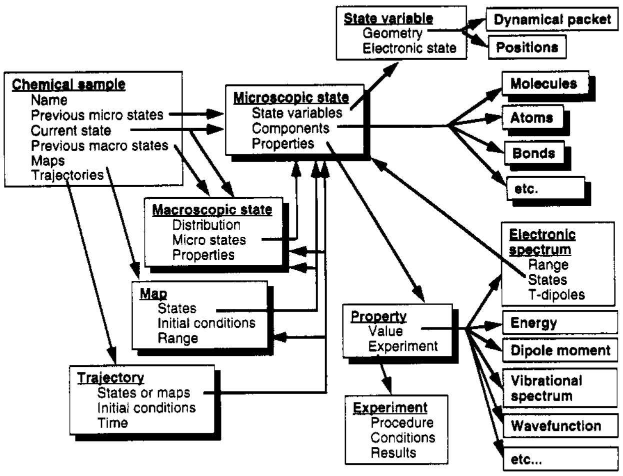


Figure 4. Exploded parts diagram of a chemical sample. Arrows point to the parts of a composite object and branched arrows indicate that a subcomponent could be one of several different kinds of objects. Drop shadowed boxes denote sets of objects.

depends upon the structure and state of the chemical sample and can vary with time. For example, molecules are named collections of atoms that depend upon the number, types, and positions of atoms (stereo chemistry) as well as the patterns and types of bonds. In this context, "molecule" is simply terminology for describing a particular arrangement of atoms and bonds in space. Bonds are named interactions between atoms.

The electronic state and the geometry and momentum of atoms specify a microscopic state. The geometry could be either a set of nuclear positions or a dynamical packet (e.g. vibrational wave function). The electronic state includes net charge.

Properties of a microscopic state include energy, electronic spectrum, vibrational spec- energy, electronic spectrum, vibrational spectrum, wave function, dipole moment, etc. A property value is measured in an experiment that produces results by following a procedure under specified conditions.

Over two dozen new objects are underlined. Generally, properties of macroscopic states parallel those of the microscopic state and are omitted to simplify the discussion. Figure 4 shows how they fit into the exploded parts diagram of a chemical sample. As shown in Figure 4, each state of a chemical sample can have different atoms, molecules, bonds, and so forth, reflecting changes of state caused by transformations like evaporation or reaction. Each state can have many different kinds of properties with each property having a value measured by experiment.

The attributes and relationships of properties have not been detailed here. However, the electronic spectrum is expanded to show that some properties of a state can refer to other states. The excitation energies and transition dipoles depend upon the nature of the final state in the transition.

The chemical sample object diagrammed in Figure 4 can behave in a computer model as a real chemical sample might behave in the physical world. Matter can be added to the sample and evaporate from it. Molecules and bonds can form or break as the sample reacts. Properties of pure states, as well as thermodynamic properties, can be measured.

CONCLUSION

Computer-based chemistry systems that parallel the way chemists conduct their research have an inherent ease-of-use advantage. Similarly, standardization in chemical information enhances ease of use and sharing of information, but effective standardization requires agreement first at a conceptual level.

Object-oriented software development can be applied to chemistry research and can lead to software systems which are based on the traditional terminology and methods of chemistry research. We have applied object-oriented analysis to propose a conceptual model of chemistry research which we suggest is a first step toward building a consensus as to the conceptual model. Our analysis leads us to the conclusion that a central chemical entity is the chemical sample: a collection of atoms that has a state which may change. The analysis suggests that chemistry modeling and database systems that parallel the way chemists conduct their research should use the chemical sample as a central organizing concept.

ACKNOWLEDGMENT

The author wishes to acknowledge David Maier and Judy Cushing at the Oregon Graduate Institute for information on object-oriented methodologies and their applicability to chemistry. He also acknowledges many useful conversations with Mark Thompson at Battelle Pacific National Laboratories and Steve Tyler and Sam Cole at CAChe Scientific.

REFERENCES AND NOTES

(1) Rumbaugh, J.; Blaha, M.; Premeriani, W.; Eddy, F.; Lorensen, W. *Object-Oriented Modeling and Design*; Prentice Hall: Englewood Cliffs, NJ, 1991.

(2) Ash, J.; Chubb, P.; Ward, S.; Welford, S.; Willett, P. *Communication, Storage and Retrieval of Chemical Information*; Ellis Horwood Ltd.: Chichester, U.K., 1985.

(3) Heller, S. R. Chemical Information Activities: What the Future Holds *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 284–291.

(4) Gragg, C. E. Recent ASTM Standardization Developments for Chemical Information. *J. Chem. Inf. Comput. Sci.* **1993**, *33*, 18–21.

(5) Chemical Abstracts Service Registry Structure Standard Distribution File, Chemical Abstracts Service: Columbus, OH.

(6) Bebak, H.; Buse, C.; Donner, W. T.; Hoever, P.; Jacob, H.; Klaus, H.; Pesch, J.; Roemelt, J.; Schilling, P.; Woost, B.; Zirz, C. The Standard Molecular Data Format (SMD Format) as an Integration Tool in Computer Chemistry. *J. Chem. Inf. Comput. Sci.* **1989**, *29*, 1–5.

(7) Dalby, A.; Nourse, J. G.; Hounshell, W. D.; Gushurst, A. K. I.; Grier, D. L.; Leland, B. A.; Laufer, J. Description of Several Chemical Structure File Formats Used by Computer Programs Developed at Molecular Design Limited. *J. Chem. Inf. Comput. Sci.* **1992**, *32*, 244–254.

(8) Barnard, J. M. Draft Specification for Revised Version of the Standard Molecular Data (SMD) Format. *J. Chem. Inf. Comput. Sci.* **1990**, *30*, 81–96.

(9) Gragg, C. ASTM E49.51 On Chemical Structural Information; ASTM: 1916 Race St., Philadelphia, PA 19103–1187, 1993.

(10) Anelli, P. L.; Ashton, P. R.; Ballardini, R.; Balzani, V.; Delgado, M.; Gandolfi, M. T.; Goodnow, T. T.; Kaifer, A. E.; Philp, D.; Pietraszkiewicz, M.; Prodi, L.; Reddington, M. V.; Slawin, A. M. Z.; Spencer, N.; Stoddart, J. F.; Vicent, C.; Williams, D. J. Molecular Meccano. 1. [2]Rotaxanes and a [2]Catenane Made to Order. *J. Am. Chem. Soc.* **1992**, *114*, 193–218.

(11) Zdonik, S. B.; Maier, D. *Readings in Object-Oriented Database Systems*; Morgan Kaufmann Publishers, Inc.: San Mateo, CA, 1990.

(12) Pascoe, G. A. Elements of Object-Oriented Programming. *Byte* **1986**.

(13) Pauling, L. *General Chemistry*; Dover Publications, Inc.: New York, 1970.

(14) Purvis, G. D., III; Bair, R. A.; DeVaney, D. M.; Feller, D. F.; Thompson, M. A.; Maier, D.; Cushing, J. B. The Chemistry Conceptual Model. The Chemistry Object Research Architecture (CORA) Working Group, Report No. 1, 1992; available from the author.

(15) Maier, D.; Cushing, J. B.; Hansen, D. M.; Purvis, G. D., III; Bair, R. A.; DeVaney, D. M.; Feller, D. F.; Thompson, M. A. Object Data Models for Shared Molecular Structures. Symposium on Computerized Chemical Data Standards; Data Interchange and Information Systems (American Society for Testing and Materials): Philadelphia, 1993.