Acta Cryst. (1998). A**54**, 820–832

Data Languages and Dictionaries for Crystallography

Sydney R. Hall

Crystallography Centre, University of Western Australia, Nedlands 6907, Australia. E-mail: srh@crystal.uwa.edu.au

(*Received 15 May 1998; accepted 26 August 1998*)

Abstract

Advances in computing over the last 50 years have profoundly influenced the evolution of crystallography's data-rich and calculation-intensive methodologies. The increasing emphasis on data integrity and accessibility over this period has necessitated, and spawned, new and innovative computer applications. The recent use of universal data languages to facilitate the interchange, publication and archiving of data, and the rigorous definition of data in computer-readable dictionaries, has improved the efficiency of these applications and provided new insights into our science. These developments have been aided by the International Union of Crystallography, which, through its Commissions and journals, plays a leading role in coordinating and promoting information dissemination, data standards, databases and ensuring their convenient access. This paper describes the uses of data languages and dictionaries, with particular reference to the development of the Crystallographic Information File (CIF) which is now widely used in the structural sciences.

1. Introduction

Simple, reliable and robust ways of disseminating information are fundamental objectives in science. Publications such as *Acta Crystallographica* have expertly fulfilled this role for over the past half century. More recently, new electronic approaches to dissemination have evolved which both complement and compete with the traditional publication modes, and these are the focus of this paper. Improved data-handling methodologies in structural studies, for example, have made possible the efficient transfer of results in computer-readable form to colleagues, publications or archives. New insights into these results can be achieved by searching databases for previously undiscovered relationships between the data. Learning more about the distributions of molecular geometries, ligand interactions and protein folds, for instance, will enhance future structural studies. In all these activities, an appropriate data-handling framework empowers the computer to function as a valued scientific assistant, bringing unexpected new information to our attention.

It is claimed that the 'computer' is one of the three most important scientific discoveries of the 20th century, the other two being the 'atom' and the 'gene' (Kaku, 1997). Certainly computers have played a central role in the development of crystallographic techniques and, not surprisingly, crystallographers have pioneered many of the computer solutions to large-scale data-handling problems. This symbiotic growth exists because the primary activity of our discipline, the transformation of diffraction intensities into molecular images, is data-rich and calculation-intensive. In contrast, the methodology of data acquisition can be less demanding than for many other sciences so that calculations are usually the rate-determining steps in a structure study. Indeed, throughout the 1950s, 60s and 70s, these calculations were considered to be among the 'major challenges' for electronic computers. So much so that many computing centres during this period were dominated by a crystallographic workload and, not infrequently, by crystallographers as staff!

The scope and accessibility of crystallographic applications has always been closely linked to the almost 'exponential' relationship between the size of a structure and the resources required for the calculations, and this meant that new theoretical approaches were invariably dependent on their 'scalability' to faster and larger computers. For example, the application of structure-invariant theory (Karle & Hauptman, 1950; Sayre, 1952) is still being improved computationally, and the success of the latest methodologies (*e.g.* Miller *et al.*, 1994) are

Sydney Reading Hall is the Director of the Crystallography Centre of the University of Western Australia. He is Editor of Section C of Acta Crystallographica and was a member of the IUCr Commission of Crystallographic Computing from 1975 to 1984 (Chair 1981–1984). He was an Executive member of the Society of Crystallographers in Australia from 1983 to 1988 (President 1985–1987), President of the Asian Crystallographic Association at its foundation (1987–1990) and a Councillor of the Asian Crystallographic Association from 1987 to 1993. His research interests are in matrix-based symmetry notation for computer and database applications, and data-handling approaches involving the STAR File and CIF format. He is the coordinator of the Xtal program package.

ISSN 0108-7673

strongly dependent on computer performance. Rapid advances in solid-state electronics have also led to faster and better diffraction measuring instruments and the leap-frogging of data measurement and calculation capacity has spawned a continuum of innovative data-handling approaches.

All these developments have contributed significantly to the volume of data that needs to be exchanged, stored and published. In science, this recent rapid growth in data is often referred to as 'the information explosion', whereas the technical issues of expanding computer capacities is known in the computing industry as 'Big Data'. In crystallography, the growth of structural databases, such as the Cambridge Structural Database and the Protein Data Bank, are indicative of the vastly increased capacity of scientists to produce results, and the pressures on crystallographic journals to publish them. In anticipation of the need for more efficient data communications and electronic publishing trends, the IUCr decided in 1987 to set up a Working Party on Crystallographic Information (WPCI) to recommend a standard procedure for data handling. It was the WPCI that initiated the development of the Crystallographic Information File (CIF) (Hall *et al.*, 1991) and promoted its adoption for efficient exchange and publication of data.

This paper will concentrate mostly on the development of the CIF and its applications within crystallography. It will also introduce the basic concepts of data languages and dictionaries, which will continue to play a significant role in the computerization of our techniques. The CIF developments are, as with most other crystallographic advances, linked closely to the rapid growth in computer performance, solid-state electronics and global networks. The sheer enormity of these latter developments will continue to exert pressure on science, and this discipline in particular, to find new ways to manipulate and disseminate information.

2. The adoption of the CIF format

An essential requirement for any information exchange process is an agreed protocol. At the simplest level, this protocol must specify the way in which data (*i.e.* numbers, characters or text) are arranged or constructed in the exchange medium. The construction rules define the *syntax* or the *format*. At a higher level, the protocol may also define the 'meaning' of the data, or the semantic rules. For example, in a dictionary of the English language, the grammatical arrangement of the words is the syntax and the descriptions of the words provide the semantic information.

The overall process of transporting and storing electronic data is referred to as *data handling*. Crystallographers have over the years used countless data-handling approaches and formats. In the earliest days of computing, data exchange between laboratories was infrequent and the formats of commonly used programs, such as *ORFLS* (Busing & Levy, 1962) and *XRAY* (Stewart, 1963), served this purpose well. With the introduction of magnetic tapes and floppy disks, the storage and transportation of data became easier and cheaper, and simplified the deposition of structural information in databases such as the Cambridge Structural Database (CSD) and the Protein Data Bank (PDB). The CSD and PDB facilitated depositions by specifying standard formats such as ASER, BCCAB and PDB. For example, the PDB format uses 80 column lines with identifier codes and static formats within each line, a style not dissimilar to most computer languages and crystallographic packages at the time it was introduced. Simple data representations with a rigid pre-ordained syntax worked well for more than two decades, but, as we shall see, have now reached their limits.

Fixed-format files were quite adequate for most crystallographic exchange activities until the 1980s. About then, local and international computer networks came into common use and these permitted much faster data communication between laboratories. Somewhat before then, however, it was apparent that many different fixed formats complicated data exchange and that some form of standardization was essential to make transfers more efficient. In the late 1970s, the IUCr Commissions on Crystallographic Data and Crystallographic Computing set up a joint working party to recommend a standard format for crystallographic data. This led to the development of a Standard Crystallographic File Structure (SCFS) (Brown, 1983, 1985) which was based on a partially fixed format, similar to those of databases, in which key words on each line identify blocks of data containing items in a specific order.

The SCFS format proved useful for a number of years, despite the fact that its introduction occurred at a time when quite powerful computers, such as the ubiquitous VAX780, became affordable to individual laboratories, and these computers opened up an era in which new data types and file formats flourished. The expansion of data requirements seriously challenged the extensibility of the fixed-format SCFS. Moreover, because of increased access to network facilities, there was a growing interest in submitting manuscripts to journals electronically, especially for small-molecular-structure studies. The latter led to a proposal at the XIV IUCr General Assembly and Congress in Perth in 1987 for *Acta Crystallographica* to accept electronic submissions so as to reduce manual preparation of data and resulting transcription errors and to minimize labour-intensive data-checking procedures. As a result of an Assembly resolution, the IUCr set up the Working Party on Crystallographic Information (WPCI) to investigate the feasibility of electronic publication submissions.

Because electronic manuscript submissions involved a much wider range of data types than previously

contemplated, a more flexible approach to data handling was needed than that possible using the SCFS format. The WPCI met during the 1988 ECM11 conference in Vienna and recommended the development of an exchange protocol, to be named the Crystallographic Information File (CIF) format, based on the Self-defining Text Archive and Retrieval (STAR) File (Hall, 1991). This approach provided a flexible and extensible file structure in which text and numerical data may be arranged in any order (see §4.3 below). Other universal formats, such as ASN.1 (ISO, 1987) and JCAMP-DX (McDonald & Wilks, 1988), were considered less efficient for the repetitive list data used in crystallography. The data structure of the STAR File is illustrated in Fig. 1 using a computational chemistry specification for a water molecule.

A small working group was commissioned by the WPCI to specify the CIF syntax and to recommend the data items required for manuscript submission to *Acta Crystallographica*. This group proposed that the CIF syntax (see §4.4) be more restrictive than that of the STAR File in order to facilitate its rapid implementation in existing software packages. A proposal to adopt the CIF format as a standard was tabled by the WPCI at the XV IUCr Congress in Bordeaux in 1990 as part of the Open Meetings of the IUCr Commissions on Crystallographic Data and Computing. It was subsequently adopted by the IUCr as the preferred format for data exchange (Hall *et al.*, 1991). The responsibility for administering the CIF standard and approving new data items is that of the IUCr Committee for the Maintenance of the CIF Standard (COMCIFS). This Committee continues to play a crucial role in the coordination of CIF activities and the updating and addition of new data definitions to the standard dictionaries. Information about their current activities and other CIF developments is available *via* the web site http://www.iucr.org/iucr-top/cif/.

3. The integrity and accessibility of data

The ability to easily and accurately disseminate information is essential to any scientific discipline. Crystallography has always placed special value on the integrity and accessibility of its data and a need to validate data before it is placed in the public domain. For well over a century, printed scientific journals and texts have been responsible for delivering this information, with editors, referees and publishers acting as arbiters of integrity, and libraries being the key holders to data access. IUCr journals have been part of this process for the past 50 years.

To properly understand the evolution of modern data-handling and dissemination processes, one must appreciate the way that scientific data were published earlier this century. In the 1900s, a scientific paper was often hand-written (*i.e.* manuscript), a practice that persisted on occasions up to the earliest days of *Acta Crystallographica*. Publishing a paper then was much more difficult than it is today. The choice of journals was limited, the effort needed to produce and copy a manuscript was very large, and the review process was lengthy and rigorous. This meant that authors treated papers as rare and important public statements, which had to be checked meticulously for transcription errors from hand-written laboratory notes. The technical problems for scientific publishing houses at that time were the same as those confronting the authors. There were daunting data and text entry and verification requirements, with no help from word-processors, copiers or faxes! The combination of these human-intensive tasks provided copious opportunities for errors, and it is a testimonial to the care and dedication of authors and journal staff that the frequency of mistakes in publications was so low. They did exist in significant numbers, however, as databases were to discover later when entering and checking the numerical data. Rather remarkably, many of the 'manuscript-based' publication practices continued until well into the 1980s, and for some journals they still exist!

3.1. *Access to computer databases*

The most dramatic changes to scientific data dissemination in this century came not from publishers but from computer databases. About 30 years ago, the improved cost/performance of computers made them suitable as permanent depositories for large-scale data. What started out as local archives quickly grew into national and international databases capable of providing data access on a scale that individual publishers or libraries could not match. Because most databases offer value-added information rather than the primary data available from publishers, these services were, and still are, complementary. This balance in data-access roles is maintained because databases usually depend on journals for validated data and, of course, the still predominant *publish-or-perish* ethos strongly supports the need for publishers! In the longer term, however, it is uncertain that the nexus between databases and journals needs to be maintained.

Another factor influencing dissemination modes is the rapidly increasing capacity of data-acquisition devices. In crystallography, recent improvements through the introduction of area detectors have led to tenfold increases in diffraction data measurements. Similar gains have been made with NMR spectroscopic data. The ability of scientists to elucidate structures faster has improved research productivity in many allied fields and this is expected to flow into publications. Increases in journal submission numbers and costs at a time when subscriptions are decreasing is already testing the capacity of scientific publishers using traditional editorial and publication processes (Hall, 1997).

3.2. *The advent of electronic publications*

The past decade has seen increasing pressure on scientific publishers to provide electronic services commensurate with the fast-access network expectations of readers and the burgeoning technical capabilities of authors. As has already been discussed, the inefficient and error-prone nature of manual publication processes inhibits the cost-effective and reliable dissemination of information. For these reasons, the IUCr formed, in 1990, an Electronic Publishing Committee (EPC) with Ted Maslen as the Chair, to advise on publication planning and the appropriate electronification of its journals, tables and monographs. This planning role was quite independent of the publication responsibilities of the IUCr Commission on Journals. An early recommendation of the EPC was that *Acta Crystallographica Section C* accept electronic manuscripts in CIF format and this was implemented in 1991. Five years later, with CIF submissions exceeding 90%, this format became mandatory. Largely as a result of this recommendation, the CIF format has become widely used in structural chemistry, as a deposition format for databases such as the CSD, ICSD, PDB and ICDD, and as a submission format to *Acta B* and *Acta C*, *Zeitschrift für Kristallographie*, and the structural journals of the American Chemical Society and The Royal Society of Chemistry.

The benefits of a standard flexible data format such as CIF extend far beyond a convenient and speedy mechanism for transferring manuscripts to journals. Flexible data languages influence all existing and future data-handling activities. An obvious and immediate benefit is that the numerical structural data are now published and archived without manual handling. Transcription errors have been eliminated because the data generated by authors flows without intervention from the laboratory computer to the publisher, and then to archives and databases. This process alone has

```
data_water_molecule

_qchem_chemical_name_common          water
_qchem_chemical_name_IUPAC           'oxygen dihydride'
_qchem_chemical_formula              'H2 O'

loop_
      _qchem_molecular_site_number
      _qchem_molecular_site_label
      _qchem_molecular_site_symbol
      _qchem_molecular_site_x
      _qchem_molecular_site_y
      _qchem_molecular_site_z
      _qchem_molecular_site_mass
    1  O1  O   0.00000  0.00000  0.00000  15.994915
    2  H1  H   0.00000  0.75753  0.58707   1.007825
    3  H2  H   0.00000 -0.75753  0.58707   1.007825

loop_
      _qchem_basis_set_atom_name
      _qchem_basis_set_atom_symbol
      _qchem_basis_set_contraction_scheme
      _qchem_basis_set_funct_per_contraction
      loop_
            _qchem_basis_set_function_code
            _qchem_basis_set_function_count
            _qchem_basis_set_function_exponent
            _qchem_basis_set_function_coefficient

oxygen     O        (9,5,1)->[4,2,1]   {6:1:1:1,4:1,1}
           s 1     7816.540000      0.002031
           s 1     1175.820000      0.015436
           s 1      273.188000      0.073771
# data omitted for space
          d 7        0.900000      1.000000    stop_

hydrogen   H        (4,1)->[2,1]      {3:1,1}
           s 1        19.240600     0.032828
# data omitted for space
           s 2         0.177600     1.000000
           p 3         1.000000     1.000000   stop_
```

Fig. 1. Example STAR File.

dramatically improved the efficiency and reliability of data deposition.

Before considering other beneficial aspects of the CIF approach, it is important to highlight the role of software developers in this effort. Without their involvement, it is unlikely that any new format will be universally adopted. The rapid and comprehensive acceptance of CIF was possible because crystallographic packages, such as *SHELX* (Sheldrick, 1993), provided a CIF output option and this meant that most laboratories were able to automatically generate structural data in CIF format within two years of its adoption by the IUCr. Authors submitting a structural study for publication using a generated CIF then only need to add the text using a standard editor. Several CIF-editing programs are available for this purpose (Ferguson, 1996; Westbrook *et al.*, 1997; Bernstein, 1998).

3.3. *Automated validation of data*

One of the enormous benefits of submitting structural papers to a journal in a standard format such as CIF is the facility for automatic data checking. Authors currently e-mail the CIF submission directly to the IUCr Editorial Office, where it is stored, registered, and checked automatically. Specially developed utilities test the integrity of the file syntax and check that the submitted structural data are self-consistent and meet journal standards. These checks are the *Acta* validation process and the results are recorded on a validation report. Journal staff intervene in this process only if a serious error is detected and the authors need to be notified. In the future, communication with the author may also be handled automatically. Once the validation process is complete, the CIF is converted into a formatted preprint of the paper, and this, with the validation report, is forwarded to a Co-editor for scientific review.

The IUCr Editorial Office also provides automatic e-mail servers so that authors can validate a CIF paper prior to submission. The CHECKCIF server (checkcif@iucr.org) returns a validation report to the authors, and the PRINTCIF server (printcif@iucr.org) returns a PostScript image of the text. Since June 1998, authors have been required to use CHECKCIF prior to submitting to *Acta C*. This reduces editorial delays and involves authors directly in the administration of data standards. Such facilities, along with the widespread availability of CIF generating, reading and checking software, have induced non-IUCr journals and databases to adopt the CIF format for their own submissions.

Automatic handling facilities assist authors in other ways as well. Firstly, they reduce publication times. It will soon be possible to publish an error-free CIF as an *Acta C* CIF-access paper in a time as short as one month. Secondly, automatic handling contains the costs for publishers and this keeps journal subscription fees down – a critical consideration at a time when printing costs are rising and library budgets are falling. Thirdly, these facilities provide for automatic validation of the submitted data and the consistent maintenance of data standards. The detailed listing of the validation algorithms on the web (see http://www.iucr.org/iucr-top/journals/acta/dv.html) give authors a clear and predictable set of data standards to aim for. Their automatic application is fast, consistent and increasingly thorough, and this allows authors, editors and referees to focus on the scientific and interpretative aspects of the paper. On a more cautionary note, however, the automatic checks are not as sophisticated as the best referees yet (a useful parallel is the advance of grammar checkers in word processors), and this why the journal provides an 'explanation' proforma for use in the submitted CIF if the author wishes to query a validation error detected when running CHECKCIF. Nevertheless, in the future, the thoroughness and accessibility of automatic checking will provide authors with a much better opportunity for self-assessment prior to submission, and promise major improvements to the data quality and speed of publication.

Co-editors handling the review of *Acta C* papers also benefit significantly from electronic submission and automatic validation because it reduces data checking in the assessment of a paper. With FTP and e-mail access to the CIFs in Chester, reviewers are able to operate almost entirely from electronic files. Network software is also available (du Boulay, 1997) for the referee to view, manipulate and check the structural data in the CIF. After the manuscript is accepted, the updated CIF is archived electronically in the IUCr Editorial Offices. Because the archived CIFs accessible *via* the internet (http://www.iucr.org/iucr-top/journals) contain more information than is printed, they provide primary data not always available from structural databases. For these reasons, the CIF archive has become an important part of the overall IUCr electronic delivery strategy.

4. The concept of a data language

The way that data are *presented* has an important bearing on their comprehension. That is, the efficient transfer of any information, be it text, numbers or graphics, depends critically on providing *cues* to the meaning of the data. In this context, a cue may be prior knowledge of a fixed format or explicit descriptors which identify the data or even pointers to the relationship between data items. Cues represent the contextual thread of a *data language*, in the same way that human languages have grammatical rules that convert words into understandable sentences and computer languages link individual operations to form coherent and executable algorithms.

This section will consider the concepts of data and languages that are important to handling crystal-

lographic data, and this will provide the technical basis for the adoption of the CIF format.

The first requirement of a format is that it be machine interpretable, *i.e. parseable*. It must be possible for data to be read, interpreted, manipulated and validated entirely by computer software. Universal file formats also need to be flexible, extensible and general. These are important properties because scientific data types change and expand continually. In order that a file format has longevity, it must accommodate new data types without corrupting existing stored files. It is the inflexibility and restrictive scope of fixed formats that have curtailed their usefulness. For example, CSD formats do not facilitate atomic displacement parameters and the PDB format cannot handle the new data types needed to archive macromolecular studies, or the existing data types for studies where the number of atoms exceeds 99 999.

4.1. *Data models*

Which data languages are the most suitable for crystallographic purposes? In 1988, the WPCI (see §2) decided on the STAR File as the most appropriate model for the development of the CIF. To fully appreciate this choice, and to understand the advantages that a universal data language offers, it is useful to have some knowledge of relational and object-oriented approaches to handling data. In this section, we will look at this briefly, and see that the STAR File supports, *via* its associated data dictionaries (to be discussed in §5), a combination of both paradigms.

Relational database models present data as tables with a hierarchy of links that define the dependencies among tables. These explicit relationships enable certain properties of data to be shared and, for related data values, to be derived. The structure of the data links in a relational database is usually defined separately from the component data. This is an important strength of this approach. However, when data types and dependencies change continually, static relationships are inappropriate, and something must be performed to extend the relational approach. The object-oriented database model allows data items and tables to be defined without static data dependencies. A data item may be considered as a self-contained 'object' and its relationships to other objects handled by 'methods' or 'actions' defined within the objects. A database may have a base of statically defined explicit relationships with a dynamic layer of relationships provided by presenting some (or all) items as objects. Objects have well defined attributes, some of which may involve relationships with other data items but objects need not have pre-ordained links imposed by the static database structure. The STAR File and its dictionaries support aspects of both the basic relational model and the extended capabilities of an object-oriented model. This duality provides the flexibility and extensibility associated with object-oriented data, as well as the relational links that are so important for data validation and, ultimately, data derivation. Those requiring more details on data models should refer to any of the many texts on this topic (*e.g.* Kim, 1990; Gray *et al.*, 1992).

It will assist in understanding the STAR File data structure if we now introduce the basic concepts of data attributes. These will be considered in more detail in the description of data dictionaries (see §5).

4.2. *Data attributes*

Modern data languages describe and utilize the characteristics, or *attributes*, of data. Every data item, be it a number, text or graphical object, has a set of specific characteristics which distinguish it from other data. The number 20, for example, is a positive integer with the prime factor:quotients of 2:2 and 5:1. The same number expressed as 20°C or 20°Celsius has additional attributes, namely that this is a temperature value in degrees Celsius. This knowledge imposes enumeration constraints on data in the *temperature* class and the *degrees-Celsius* subclass, and these stipulate that the value cannot be less than −273. Such constraints are important in the validation sense, and are applicable only with semantic knowledge of the data. This is emphasized by comparing the values of 100°Celsius and 100 Kelvin. Both are measures of temperature but are restricted to different enumeration ranges, the knowledge of which is essential to their interpretation and application.

```
data_chemical_compound_source
    _name                '_chemical_compound_source'
    _category            chemical
    _type                char
    loop_ _example       'From Norilsk (Russia)'
                         'Extracted from the bark of Cinchona Naturalis'
    _definition
;             Description of the origins of the compound under study, or
              of the parent molecule if a simple derivative is studied.
              This includes the place of discovery for minerals or the
              actual source of a natural chemical product.
```

Fig. 2. A simple example of a DDL data definition.

Because data dependencies and links are critically important in the relational treatment of data, we shall consider one more example. Assume that the numbers 5, 3, 0 are the Miller indices h, k, l specifying a particular point in reciprocal space. As nonscalar data, the indices have the properties of being *non-associative* (*e.g.* 3, 5, 0 is not equivalent to 5, 3, 0) and *irreducible* (*e.g.* the index 3 alone has no meaning). They also represent a *reference pointer* (*i.e.* a unique access key) to specific reflection data (*e.g.* structure factors) in a list, such that the list is *invalid* unless these indices are present. Note that the reference pointer property, as with some other data dependencies, is unique to the particular list of data items. Other types of data dependencies will be described in §5.

4.3. *The syntax of the STAR File*

The most important property of a data language is that it be parseable, *i.e.* understood contextually, and this implies, indeed requires, that there be a precise *syntax* governing the permitted data constructions. The STAR File data structure consists of *tag–value pairs* or *tuples*, often referred to as *data items*, arranged as sequential text lines of standard visible ASCII characters. The file content is divided into any number of discrete units, or cells, each containing unique sets of data items. As a free-form language, blanks, horizontal tabs, new lines and form feeds (collectively known as *white space*) are ignored except as a separator of a string of 'non-white-space' characters, which is referred to as a *token*.

A formal specification of the STAR File syntax, using Backus–Naur form (BNF) (McLennon, 1983), is available.† Syntax definition scripts, such as BNF, provide a 'machine parseable template' for building the parsing software for a target data language. The syntax of the CIF, MIF and DDL formats, which will now be described, is based on that of the STAR File.

4.4. *The CIF format*

The Crystallographic Information File (Hall *et al.*, 1991) has the same syntax as a STAR File, except for the following restrictions.

(*a*) Lines may not exceed 80 characters in length.

(*b*) Data names and block codes may not exceed the length of a line.‡

(*c*) A data item is assumed to be of type *number* if it starts with the characters <0>–<9>, <+>, <−> or <.> *and* is not bounded by a <'> or <">. A number may be in integer, real or scientific format.

(*d*) A data item is assumed to be of type *character* if it is not a *number*.

(*e*) Only one level of `loop_` data is permitted.

(*f*) The *save frame* command is not permitted.§

4.5. *The MIF syntax*

The *Molecular Information File* (MIF) (Allen *et al.*, 1995) has the syntax of a STAR File and is an exchange format of chemical data which is independent of proprietary software and database systems. The MIF philosophy, and its data definitions, are complementary to those of CIF and represent a coalescence of two format developments: the Standard Molecular Data (SMD) (Barnard, 1990) and STAR/CIF. Although the CIF construction is able to store a representation of the topology of molecules, its data types do not fulfil the needs of the chemical community. Molecular information embraces the broad spectrum of data related to chemical and molecular structure and includes both items for spectroscopic measurements, thermochemical data, electrochemical properties, crystal structure information, and so on. These represent the object-oriented data descriptors of molecular chemistry and it is intended that all of these will eventually be accommodated in the MIF approach. In most MIF applications, a data block will usually specify a complete chemical entity, *i.e.* a fully defined molecule or a query substructure. The MIF syntax, unlike that of a CIF, places no restrictions on line lengths or nested loop levels. CIF data items can readily be incorporated into a MIF. It should be noted, however, that the reverse may not be true (*i.e.* MIF data in a CIF) because of the more restrictive CIF syntax.

5. Crystallographic dictionaries

The syntax and semantics of STAR data languages are specified separately; the syntax by a simple set of construction rules, and the semantics as definitions of each data item within a data dictionary file. Data definitions provide semantic information which is essential for the unambiguous exchange of data items, enumeration constraints, relationships between different data items, and construction information on the valid use of the data. The syntax of data definitions and the dictionary file as a whole conform identically to those of a STAR File, in that the sequence order of both the definition blocks and the attributes is arbitrary. Such a construction enables the dictionary file to be expanded or updated independently of the data files and provides the versatility for incorporating new items, immediately and seamlessly, into a data-interchange mechanism.

† Hall & Spadaccini (1994) and a more recent version at the web site http://www.cs.uwa.edu.au/star.

‡ The original restriction was 32 characters but that has been recently expanded for the mmCIF data items.

§ Save frames are used in DDL2 dictionary files (see §5.1) but not in data files.

5.1. *Dictionary definition languages (DDL)*

The definition of each data item in a STAR dictionary file starts with a header line which matches that of the defined data name. The header is followed by a sequence of data declarations, each of which specifies a separate attribute of the defined data item. Each attribute is a component of the total definition information and only those attributes appropriate to the particular defined item appear in the definition sequence. The attributes constitute the vocabulary of the *dictionary definition language* (DDL) and provide the semantic tools of the dictionary. The organization of attributes within each definition and of the definitions within a dictionary file conforms to the requirements of a STAR File and may be accessed using standard STAR parsing tools (see §6).

The ways that the DDL is used to define data are most easily understood from a simple example. Fig. 2 contains the definition of the data item `_chemical_compound_source`. This data item can be defined with only a few attributes, `_name`, `_category`, `_example` and `_definition`, because these are sufficient to uniquely specify the source of a chemical compound and enable its validation.

Two different versions of DDL are currently in use. The first, DDL1 (Hall & Cook, 1995), is used for the definition of data items in the core and powder (Toby, 1997) dictionaries. The second, DDL2 (Westbrook & Hall, 1995), has stronger relational attributes and is used in the definition of macromolecular data (Bourne *et al.*, 1997). Although the DDL versions are closely related, and most CIF-application software is conversant with both, some important differences should be noted. The attributes employed by DDL1 are listed in Fig. 3 and an overview of the differences between the DDL1 and DDL2 versions is shown in Fig. 4. Additional information on dictionary languages is available from the web sites http://www.iucr.org/iucr-top/cif and http://ndbserver.rutgers.edu/mmcif/ddl/. Copies of the associated dictionaries of DDL attributes are available from the first address.

5.2. *Currently approved data dictionaries*

A variety of local and global dictionaries conform to the STAR/CIF syntax, and at least two, NMR (Ulrich *et al.*, 1996) and imgCIF/CBF (Hammersley, 1997) are used for other data activities. Only the dictionaries containing standard COMCIFS-approved data items will be described here but readers can refer to more extensive information at http://www.iucr.org/iucr-top/cif/. Copies of the latest approved versions of these dictionaries are available from that site as well.

DDL1 and DDL2 dictionaries. The attributes of the dictionary definition languages, DDL1 and DDL2, are defined in their own separate dictionaries. The DDL1 dictionary (Hall & Cook, 1995) attributes are used in the core CIF, core MIF and powder CIF dictionaries. The DDL2 dictionary (Westbrook & Hall, 1995) attributes are used for the mmCIF definitions.

CIF core dictionary. The core dictionary contains definitions of the data items that are common to most 'small-molecule' crystal structure studies.

CIF macromolecular (mmCIF) dictionary. The mmCIF dictionary (Fitzgerald *et al.*, 1996) contains definitions of data used in three-dimensional macromolecular structure studies. The Nucleic Acid Database (NDB) stores all its data in terms of mmCIF definitions and has an extensive data-handling software for this

`_name`	Identifying dataname of the item
`_definition`	Text description of the item
`_example`	Example value of the item
`_example_detail`	Description of the example value
`_enumeration`	Restricted *set* of values for the item
`_enumeration_detail`	Description of the restricted values
`_enumeration_default`	Default value for the item
`_enumeration_range`	Restricted *range* for the values
`_list`	Flag (yes/no/both) if item in a looped list
`_list_level`	Nested loop level for list item
`_type`	Type of data (numb/char/null)
`_type_conditions`	Special conditions of type (e.g. esd)
`_type_construct`	Construction rules for a value
`_units`	Measurement units symbol of item
`_units_detail`	Description of measurement units
`_category`	Group, or basis set, of the item
`_list_link_child`	Named items dependent on defined item
`_list_link_parent`	Named items on which this item depends
`_list_reference`	Named item which is reference pointer for this item
`_list_uniqueness`	Named items that must be unique in list
`_list_mandatory`	Flag (yes/no) if this item required in list
`_related_item`	Named items closely related to this item
`_related_function`	Relationship codes *alternate/convention/conversion/replace*.

Fig. 3. DDL1 attribute data names used in CIF dictionaries.

purpose (http://ndbserver.rutgers.edu/mmcif/software/). The PDB offers the option of receiving mmCIF data and distributes new structure-factor data in mmCIF format.† It is currently preparing to accept mmCIF data for deposition of structural data entries and structure factors.

Powder CIF (pdCIF) dictionary. The crystallographic powder dictionary (Toby, 1997) contains definitions of data used in powder diffraction studies. This format is supported by the International Centre for Diffraction Data (ICDD).

6. Software for application to CIF data

A wide range of software is available for processing STAR Files and CIFs and development continues on new utilities that will aid the interchange, archival and publication of CIF data. Details of most CIF utilities may be obtained from http://www.iucr.org/iucr-top/cif/software. Here is a representative but by no means complete summary of the most recent utilities and tools.

6.1. *Selected search and validation utilities*

CYCLOPS (Bernstein & Hall, 1998) is a Fortran routine that scans text files for data names and checks these against dictionaries. *HICCuP* (Edgington, 1997) is a Python (Lutz, 1996) and Tcl/Tk utility that checks and corrects CIF data. *cif2cif* (Bernstein, 1996) is a Fortran routine that checks and reformats a CIF. *STAR-BASE* (Spadaccini & Hall, 1994) is an ANSI C program for searching and extracting data from a STAR File. It handles the *full* STAR File syntax; source available from http://www.crystal.uwa.edu.au/~yaya/software/star_soft.html.

6.2. *Application libraries*

CIFtbx (Hall & Bernstein, 1996) is a Fortran function library for programmers developing software to read or write CIF data (a primer is available from http://www.iucr.org/iucr-top/cif/software/ciftbx). *CIFLIB* (Westbrook *et al.*, 1997) is a C function library for developing software to read or write mmCIF data (source available from http://ndbserver.rutgers.edu/mmcif/software/). *CIFSIEVE* (Hester & Okamura, 1998) generates Fortran and C routines for reading CIF data in other software, based on simple changes to the dictionary (source from ftp://ftp.nirim.go.jp/pub/sci/cif). *starlib* (Mading *et al.*, 1998) is a C++ library for reading and writing STAR files (source available from http://www.bmrb.wisc.edu/sb_lib/starlib/).

6.3. *Conversion and manipulation utilities*

cif2pdb (Bernstein & Bernstein, 1998) is a Fortran utility that converts CIF into PDB data. *pdb2cif* (Bernstein *et al.*, 1998) is an *Awk* (or *Perl*) utility that converts PDB into CIF data. *cif2sx* (Farrugia, 1997) is a Fortran utility that converts a CIF into a *SHELX*93/97 file. *DIFRAC* (Flack, 1996) is a Fortran utility that converts a variety of diffractometer output files to CIF format. *ciftex* (McMahon, 1993) is a C utility that converts a CIF text item into TEX®. *cif_filter* (du Boulay, 1997) is a Tcl/Tk script that enables Netscape® to pass a CIF to other software. *Xtal_GX* (Hall & du Boulay, 19987) and *PLATON* (Spek, 1998) are Fortran crystallographic packages for reading and validating CIFs, and displaying and plotting crystal structures. *CIF_Input_Tool* (Westbrook *et al.*, 1997) is a GUI tool to create and edit mmCIF data. *CIFED* (Bernstein, 1998) is a program, under development, for screen-based editing of CIF data.

7. Future directions

In reviewing a fast changing topic such as this there is a need to speculate on what's likely to happen in the future! The past decade has seen major changes in the use of data and the sophistication of handling approaches, and these have influenced many branches of crystallography. In the future, better data-handling models and supporting software will certainly lead to new approaches to exploiting any impending data explosion. Fortunately, the task of containing and harnessing the data maelstrom of 'a gigahertz and terabyte world' will rest mostly with computer manufacturers, but many innovative applications will be needed to make intelligent use of data in our discipline and these remain future challenges for us to meet and to solve.

In predicting future data-handling directions, one must consider the ramifications of the current statistic‡ that 'half the world's population has never made a phone call', and that high-bandwidth digital networks are expected to eliminate such communication obstacles in the next decade. It is logical to assume that many more scientists will use networks actively in the future. A more conservative viewpoint is, however, that scientific productivity has peaked because computer and human capacities have reached their limits! While those who are struggling with current data-handling problems share this sentiment, computer manufacturers (Mashey, 1998) say that future reality is dramatic increases in capacity and performance over the next five years. It follows that new handling tasks await us.

† Latest information is available from http://www.pdb.bnl.gov/mmcif.html.

‡ *A Survey of Telecommunications: the Death of Distance – a Giant Effort* (30 September 1995). *The Economist.*

So what's next? At the moment, crystallographers are among the leaders in large-scale data-handling solutions. Nevertheless, we know that only the simplest data-handling problems have been addressed so far. There is a need for a much more comprehensive effort to automate many of the tasks that currently require manual intervention. After all, computers, networks and measurement devices will continue to make quantum leaps in efficiency, whereas unaided humans will not and therefore represent the 'bottleneck' in some data-handling processes. If real improvements in data handling are essential for our advancement and well being, many of these will need to be performed with less human involvement, just as agents or daemons already operate on current computers.

7.1. *Knowledge bases*

One of the most promising avenues for advancing automation is the development of intelligent data dictionaries. The present data dictionaries are already significant knowledge bases. The mmCIF dictionary (Fitzgerald *et al.*, 1996), with its 1500+ data definitions, represents a large and important reference library for the biomedical macromolecular community. But it is still early days for these developments, and there is a real need for new DDL attributes capable of extending the relationships between data objects. The implications of such extensions are significant. A definition paradigm which encompassed all known data properties could, at least theoretically, allow dictionaries to define or derive all crystallographic techniques and crystal structure relationships!

Is this totally over the top? Well, next year it certainly is, but in a decades' time, perhaps not. At the moment, most scientific knowledge resides in printed scientific publications, printed texts, and computer software and hardware. Very little of this is easily machine parseable and is therefore inaccessible or incomprehensible to the automated systems capable of the significant efficiency gains needed. The accessibility problems are compounded because existing knowledge sources are enormously redundant and contradictory. So where do intelligent dictionaries come in?

7.2. *Intelligent paradigms*

A primary function of a 'smart' dictionary is to facilitate the automatic derivation of data from defined relationships. A new DDL attribute is currently being tested which specifies the methods by which data items can be derived from other data items. *Methods*, in this context, refers to the process by which algorithmic code is used to relate and derive data. For example, a *calendar date* can be derived by knowing the *day*, the *month* and the *year*. Such a derivation is trivial for a human but the ability to do this automatically from a data definition has important implications. For instance, if a DDL dictionary contains this relationship, a search utility can take remedial action when the date value is missing but the related day, month and year values are not. A derived value could then be automatically returned even if the date was not present in the file or database.

To further illustrate the importance of derived information, here is a typical example of how the methods attribute could be applied to crystallographic data. Fig. 5 shows a definition for `_cell_volume` in which the attribute `_evaluation_method` specifies the unit-cell volume as a function of the cell lengths and angles, in terms of the interpretive script language Python (Lutz, 1996). The evaluation process would work as follows. Assume that a CIF is being read with

DDL1

- **Data names** identify the category of data as _<*category*>_<*detail*>.
- Definition are declared as data blocks with `data_`<*data name*>.
- An **irreducible set** of items is declared within one definition. eg. indices *h,k,l*.
- Items which appear in **lists** are identified with the attribute `list_`.
- List dependencies are declared within each definition eg. `_list_reference.`

DDL2

- Data names identify the category as _<*category*>.<*detail*>
- Definitions are declared starting with `save_`<*data name*> and end with `save_`.
- All items are defined as separate frames related by `_item_dependent.name`.
- List and no-list data items are not distinguished.
- Dependencies declared in a category definition. eg. `_category_key.name`.
- Identifies **sub-categories** of data within category groupings. eg. matrix.
- Provides **aliases** to equivalent names, including those in DDL1 dictionaries.

Fig. 4. Comparison of DDL1 and DDL2 versions.

a search utility that uses CIF dictionaries for validation and checking support. The item _cell_volume is requested but its value is not present in the file. The utility automatically extracts the dictionary definition shown in Fig. 5 and transfers the script from _evaluation_method to a Python handler, which parses the script, identifies the length and angle items needed to evaluate the cell volume, requests these values from the CIF, and calculates the volume. The calculated cell volume is passed back to the utility which responds identically to the request as if the value had been present in the CIF.

By embedding machine-readable methods into the dictionaries, a clarity and precision in communicating results will be achieved which has not been possible in the past. For example, in structural studies the ambiguities in understanding and comparing *R* factors, atomic displacement parameters and other terms from different laboratories would be greatly reduced. The use of methods, with the coalescence of internationally approved dictionaries and local dictionaries containing scientific advances in individual laboratories, will promote scientific collaboration, and the exploitation of data, well beyond that achievable at the moment.

The use of methods will not be confined to simple linear data relationships. It is anticipated that complex numerical, logical and iterative expressions can be specified in this way, and that future methods scripts will initiate Fourier transforms, geometry calculations and even generate graphics images. Standard mathematical packages, such as *Mathematica*® (Wolfram, 1996), would be deployed to assist in the execution of the method script. The implications of this approach for future data handling are profound at several levels.

It would mean that only primitive data need be archived in data files and the related data can be derived when needed using methods algorithms in the dictionary. This would help to reduce the amount of data that needs to be exchanged and archived. Some derived quantities (*e.g.* atomic coordinates) may continue to be archived but, having the methods information in associated dictionaries, specifying precisely how they were derived, will enable new derivations to be evaluated as better approaches are developed.

The extensive use of methods may also mean that crystallographic programs, as we know them, might cease to exist in the future. Instead, calculations could be initiated by data requests and executed by driver utilities that use the methods attributes in dictionaries as the algorithm source. The high degree of parallelism afforded by future networked computers and the technological advances in processor speeds and memory capacity should make the computational burdens of these 'lazy evaluation' schemes quite acceptable.

In closing this paper, one cannot help but contemplate one's own earliest experiences in the field and ponder on the enormous strides that computers have brought to this discipline. Today, one can in a few seconds import a CIF from a web site on the other side of the world and display the contained structure instantly on the screen. This would have been impossible even a couple of years ago. And 35 years ago, using the fastest (and perhaps only) computer in the city, it took 36 h to calculate a single projection of a Fourier map – which then had to be hand-contoured!

```
data_cell_volume
    _definition
;               Crystal unit cell volume V in angstroms cubed.
;
    _name                          '_cell_volume'
    _category                        cell
    _type                            numb
    _type_conditions                 esd
    _units                           A^3^
    _units_detail                   'cubic angstroms'
    _enumeration_range               0.0:

    _evaluation_method
;
    a  = _cell_length_a
    b  = _cell_length_b
    c  = _cell_length_c
    al = _cell_angle_alpha *6.28319 /360
    be = _cell_angle_beta  *6.28319 /360
    ga = _cell_angle_gamma *6.28319 /360
     s = (al + be + ga) /2

 _cell_volume = 2*a*b*c* sqrt( sin(s)
                              *sin(s-al)
                              *sin(s-be)
                              *sin(s-ga)    )
;
```

Fig. 5. An example _evaluation_method attribute.

One must stand in awe of these advances, and eagerly await the excitement and the challenges that this field is certain to bring in the future.

The development of crystallographic data languages, data dictionaries and associated applications is due to the efforts of many people. Without these contributions, the CIF approach would have been a useful concept that was never widely used. It is not possible here to acknowledge all of the contributors individually. Those most involved in these developments are listed in the references, and the importance and significance of their efforts are reflected in their publications, archived dictionaries and software. It is important on this occasion to acknowledge contributors who have given support to CIF developments by way of encouragement, guidance, promotion and funding. Particular thanks are due to the following. To Howard Flack, whose challenge ten years ago led to the formulation of the STAR File concepts; to Richard Goddard for his constructive feedback during the design stages of the STAR File; to the late Ted Maslen who, as Chair of the WPCI and EPC, promoted the CIF approach for electronic journals; to the IUCr Past-Presidents André Authier and Philip Coppens, and their Executive Secretaries, the late Jim King and Mike Dacombe, for supporting innovation and change at a time when there were significant risks involved; to the IUCr staff members Brian McMahon, Peter Strickland and Mike Hoyland for their outstanding help in the development and administration of CIF dictionaries, submission procedures and validation software; to the package software developers, and to George Sheldrick in particular, for implementing the CIF format expertly and promptly, to Helen Berman and Paula Fitzgerald for their commitment and determination that the mmCIF approach was the way forward for representing macromolecular data, and to Herbert and Frances Bernstein for their enthusiasm for the CIF approach and software development efforts. Acknowledgment is also due to all those agencies that funded these efforts, and a personal recognition to the Australian Research Committee for several grants supporting CIF developments at this University.

References

Allen, F. H., Barnard, J. M., Cook, A. F. P. & Hall, S. R. (1995). *J. Chem. Inform. Comput. Sci.* **35**, 412–427.

Barnard, J. M. (1990). *J. Chem. Inform. Comput. Sci.* **30**, 81–96.

Bernstein, H. J. (1996). Personal communication.

Bernstein, H. J. (1998). Personal communication (yaya@bernstein-plus-sons.com).

Bernstein, H. J. & Bernstein, F. C. (1998). Personal communication (yaya@bernstein-plus-sons.com).

Bernstein, H. J., Bernstein, F. C. & Bourne, P. E. (1998). *J. Appl. Cryst.* **31**, 282–295.

Bernstein, H. J. & Hall, S. R. (1998). *J. Appl. Cryst.* **31**, 278–281.

Boulay, D. du (1997). *Viewing of Cif Files under Netscape*, http://www.iucr.org/iucr-top/cif/software/Xtal/ciffilter/.

Bourne, P. E., Berman, H. M., McMahon, B., Watenpaugh, K. D., Westbrook, J. & Fitzgerald, P. M. D. (1997). *Methods Enzymol.* **277**, 571–590.

Brown, I. D. (1983). *Acta Cryst.* A**39**, 216–224.

Brown, I. D. (1985). *Acta Cryst.* A**41**, 399.

Busing, W. R. & Levy, H. A. (1962). *A Fortran Crystallographic Least Squares Program.* Report TM-305. Oak Ridge National Laboratory, Oak Ridge, Tennessee, USA.

Edgington, P. (1997). *HICCuP: for checking and correcting CIF.* Cambridge Crystallographic Data Centre, Cambridge, England.

Farrugia, L. (1997) *(IUCr) CIF2SX*, http://www.iucr.org/iucr-top/cif/software/cif2sx/.

Ferguson, G. (1996). *Acta Cryst.* A**52**, C574.

Fitzgerald, P. M. D., Berman, H. M., Bourne, P. E., McMahon, B., Watenpaugh, K. D. & Westbrook, J. (1996). *Acta Cryst.* A**52**, C575.

Flack, H. D. (1996). *(IUCr) DIFRAC Single-Crystal Diffractometer Output-Conversion*, http://www.iucr.org/cif/software/difrac/.

Gray, P. M. D., Kulkarni, K. G. & Paton, N. W. (1992) *Object Oriented Databases.* Prentice Hall: New York.

Hall, S. R. (1991). *J. Chem. Inform. Comput. Sci.* **31**, 326–333.

Hall, S. R. (1997). *Electronic Publishing in Science*, pp. 89–94. Paris: ICSU Press.

Hall, S. R., Allen, F. H. & Brown, I. D. (1991). *Acta Cryst.* A**47**, 655–685.

Hall, S. R. & Bernstein, H. J. (1996). *J. Appl. Cryst.* **29**, 598–603.

Hall, S. R. & du Boulay, D. (1998) *Free Xtal-GX Distribution Page.* University of Western Australia, Australia, http://www.iucr.org/iucr-top/cif/software/Xtal/GX/.

Hall, S. R. & Cook, A. P. F. (1995). *J. Chem. Inf. Comput. Sci.* **35,** 819–825.

Hall, S. R. & Spadaccini, N. (1994). *J. Chem. Inf. Comput. Sci.* **34,** 505–508.

Hammersley, A. (1997) *Draft CBF/imgCIF Definition*, http://www.bernstein-plus-sons.com/software/CBF/.

Hester, J. & Okamura, F. P. (1998). *J. Appl. Cryst.* In the press.

ISO (1987). ISO 8224: 1987(E). *Specification of Basic Encoding Rules for Abstract Syntax Notation One* (ASN.1). Geneva, Switzerland: International Organization for Standardization.

Kaku, M. (1997). *Visions – How Science will Revolutionize the 21st Century.* New York: Anchor Books.

Karle, J. & Hauptman, H. (1950). *Acta Cryst.* **3**, 181–187.

Kim, W. (1990) *Introduction to Object Oriented Databases.* Boston: MIT Press.

Lutz, M. (1996) *Programming Python.* Sebastopol, USA: O'Reilly and Associates.

McDonald, R. S. & Wilks, P. A. (1988). *Appl. Spectrosc.* **42**, 151–162.

McLennon, B. J. (1983) *Principles of Programming Languages, Design, Evaluation and Implementation.* New York: Holt, Rinehart and Winston.

McMahon, B. (1993) *ciftex: IUCr typesetting tool*, ftp://ftp.iucr.org/pub/ciftex.tar.Z.

Mading, S., Bhatamadi, S., D'Silva, T., Argentar, D., Ioannidis, Y., Ulrich, E. L., Markley, J. L. & Livny, M. (1998). *starlib: C++ Library for STAR Files in the BioMagResBank NMR Database.* University of Wisconsin, USA (see http://www.bmrb.wisc.edu/).

Mashey, J. (1998) *Big Data and the Next Wave of InfraStress.* Seminar Notes. Silicon Graphics/Cray Research, Mountain View, USA.

Miller, R., Gallo, S. M., Khalak, H. G. & Weeks, C. M. (1994). *J. Appl. Cryst.* **27**, 613–621.

Sayre, D. (1952). *Acta Cryst.* **5**, 60–65.

Sheldrick, G. (1993). *SHELX93. Program for the Refinement of Crystal Structures.* University of Göttingen, Germany.

Spadaccini, N. & Hall, S. R. (1994). *J. Chem. Inf. Comput. Sci.* **34**, 509–516.

Spek, A. L. (1998). *PLATON. Program for Analysis of Molecular Geometry*. University of Utrecht, The Netherlands.

Stewart, J. M. (1963). *XRAY Crystallographic Program System.* Computer Science Report, University of Maryland, USA.

Toby, B. H. (1997). *(IUCr) Powder CIF Dictionary*, http://www.iucr.org/iucr-top/cif/pd/.

Ulrich, E. L., Agentar, D., Klimowicz, A., Westler, W. M. & Markley, J. L. (1996). *Acta Cryst.* A**52**, C577 (see also http://www.bmrb.wisc.edu/).

Westbrook, J. D & Hall, S. R. (1995) *The Macromolecular Structure DDL Home Page*, draft DDL V 2.1.1, http://ndbserver.rutgers.edu/mmcif/ddl/.

Westbrook, J. D., Hsieh, S.-H. & Fitzgerald, P. M. D. (1997). *J. Appl. Cryst.* **30**, 79–83 (see also http://ndbserver.rutgers.edu/mmcif/).

Wolfram, S. (1996) *The Mathematica Book*. Cambridge: Wolfram Media/Cambridge University Press.