Acta Cryst. (1991). A**47**, 655–685

International Union of Crystallography

Commission on Crystallographic Data
Commission on Journals
Working Party on Crystallographic Information

The Crystallographic Information File (CIF): a New Standard Archive File for Crystallography*

BY SYDNEY R. HALL

Crystallography Centre, University of Western Australia, Nedlands 6009, Australia

FRANK H. ALLEN

Crystallographic Data Centre, University Chemical Laboratory, Lensfield Road, Cambridge CB2 1EW, England

AND I. DAVID BROWN

Institute for Materials Research, McMaster University, Hamilton, Ontario L8S 4M1, Canada

(*Received* 8 *April* 1991; *accepted* 28 *June* 1991)

Abstract

The specification of a new standard Crystallographic Information File (CIF) is described. Its development is based on the Self-Defining Text Archive and Retrieval (STAR) procedure [Hall (1991). *J. Chem. Inf. Comput. Sci.* **31**, 326–333]. The CIF is a general, flexible and easily extensible free-format archive file; it is human and machine readable and can be edited by a simple text editor. The CIF is designed for the electronic transmission of crystallographic data between individual laboratories, journals and databases: it has been adopted by the International Union of Crystallography as the recommended medium for this purpose. The file consists of data names and data items, together with a loop facility for repeated items. The data names, constructed hierarchically so as to form data categories, are self-descriptive within a 32-character limit. The sorted list of data names, together with their precise definitions, constitutes the CIF Dictionary (Core Version 1991). The CIF Core Dictionary is presented in full and covers the fundamental and most commonly used data items relevant to crystal structure analysis. The Dictionary is also available as an electronic file suitable for CIF computer applications. Future extensions to the Dictionary will include data items used in more specialized areas of crystallography.

Introduction

There is an increasing need in many branches of science for a uniform but flexible method of archiving and exchanging data in electronic form. Rapid advances in computer technology, coupled with the expansion of local, national and international networks, have fuelled the need for such a facility. The variety and relative inflexibility of existing data exchange formats have inhibited their effective use. This is true even in fields where the basic data requirements are well defined. Problems of data exchange are further exacerbated if the number and nature of data types change rapidly and continuously. Under these conditions specialized and local file formats have proliferated. This diversity was tolerable when electronic data transfer was infrequent, or when data processing speeds required file formats finely tuned to specific applications. The developments cited above signal an end to this rationale. A general, flexible, rapidly extensible and universal file format protocol is now essential. It must be machine-independent and portable so that accessibility to data items is independent of their point of origin. It must allow new data items to be incorporated without the need to modify existing files.

In addition to archiving data, the use of a universal file would facilitate data exchange between software within a laboratory; between different laboratories; between authors and journals, providing electronic input to the publication process; and between researchers or journals and computerized databases.

Crystallography is not excepted from the need for a universal exchange file. Its activities are dominated by advanced computer-controlled equipment and sophisticated

---

* Reprints of this paper may be obtained from the Technical Editor, International Union of Crystallography, 5 Abbey Square, Chester CH1 2HU, England.

0108-7673/91/060655-31$03.00

software systems which measure and process data. In most cases, especially for small and medium-sized molecules, these data are clearly defined and standardized, and are generated in machine-readable form. The problem is, however, that there are too many different forms and, despite the fundamental role that computing plays in our discipline, only limited effort has been directed at devising a general and common format.

In the late seventies the IUCr Commissions on Crystallographic Data and Crystallographic Computing promoted the development of the Standard Crystallographic File Structure (Brown, 1983, 1988). The SCFS is based on the concept of formatted lines and keywords that identify blocks of data containing items in a specific order. The SCFS format satisfies some but not all of the requirements of a universal data exchange file.

At the XIV IUCr Congress in Perth it was proposed that *Acta Crystallographica* promote the submission of data in machine-readable form. This was seen as being particularly beneficial for Section C, which publishes about 1000 small-molecule and inorganic crystal structures a year. Each paper is currently prepared as a typed manuscript and converted to machine-readable text for computer typesetting. Some of this work is carried out by *Acta Crystallographica* staff. All steps involve more manual effort than is desirable. Both the data and text are prone to transcription errors in their passage from the computer and the author to the printed page. Machine-readable submissions would reduce input errors, minimize labour-intensive data entry and check procedures, and speed the publication process. The submitted data could also be transmitted directly to the relevant crystallographic databases. An IUCr Working Party on Crystallographic Information (WPCI) was set up to investigate the feasibility of such a submission process, and to coordinate the input of various IUCr Commissions that were involved in these types of activities.

It was soon recognized that submission of text and data to journals and databases required the use of a universal exchange file, some of whose properties could not be reconciled with the constraints imposed on the SCFS by its format. At a meeting of the IUCr WPCI, held in conjunction with the XI European Crystallographic Meeting (1988) in Vienna, it was decided to develop a universal file based on the Self-Defining Text Archive and Retrieval (STAR) procedure of Hall (1991*a*). The STAR File is intended for the electronic exchange of data and provides for text and numerical data in any order.

The WPCI commissioned the authors to develop a universal exchange file to be called the Crystallographic Information File (CIF). A preliminary report on this development was presented at the XV IUCr Congress and General Assembly (1990) in Bordeaux as part of the Open Meetings of the IUCr Commissions on Crystallographic Data and Computing. This paper is a detailed description of the CIF development.

A major feature of this work has been the development of a comprehensive Dictionary (Core Version 1991) of crystallographic data items. Each data item has been assigned a self-explanatory name for use in a CIF and each item is precisely defined within the Dictionary which appears in this paper as Appendix I. The Core Dictionary defines only those fundamental data items that are commonly used in a single-crystal structure analysis. Future extensions will encompass data items that are relevant to specialized areas of crystallography. The Core Dictionary is also available as an electronic file suitable for use with CIF computer applications.

Table 1. *STAR File syntax and terminology*

text string	string of characters bounded by blanks, single quotes (') double quotes ("), or by semi-colons (;) as the first character of a line
data name	a text string starting with an underline (_) character
data item	a text string not starting with an underline, but preceded by a data name to identify it
data loop	a list of data names, preceded by `loop_` and followed by a repeated list of data items
data block	a collection of data names (looped or not) and data items that are preceded by a `data_` code record. A data name must be unique within a data block. A data block is terminated by another `data_` statement or the end of file
data file	a collection of data blocks; the block codes must be unique within a data file

To aid the description of the CIF a brief introduction to the underlying concepts of the STAR File, on which the CIF application is based, will be given. Full details of STAR File specifications are available in the literature (Hall, 1991*a*).

STAR File concepts and syntax

A STAR File is composed of ASCII text that can be edited with a simple text editor. When viewed, its contents are easy to read and can be stored or transmitted electronically without conversion. The construction of a STAR File is simple. Each file contains a sequence of data blocks. Each data block contains a sequence of individual data items. There may be any number of data blocks and any number of data items within each data block. The data block represents the logical grouping of data that crystallographers normally associate with a 'data set', but it may be used for any other purpose. The identity of each data item within a data block is determined by a unique data name which precedes it in the file. Data items may be repeated in lists by placing them within a simple data loop structure.

An important property of a STAR File is that its syntax is defined by a few simple rules (see Table 1). This ensures maximum flexibility for data exchange and wide applicability. No assumptions are made about the order of the data blocks or data items, other than the requirement that the character strings which identify data blocks, or data names within a block, must be unique. There are no restrictions regarding the placement of data names or data items within a data block, other than the requirement that the name must precede the item. Data in a STAR File are accessed simply by requesting a specific data name within a specific data block. Prior knowledge about data type (*i.e.* text or numbers), whether the item is looped, or whether the item exists in the file at all, is unnecessary.

The basic syntax of a STAR File is best illustrated by examples. Each data block is identified by a unique character string starting with '`data_`'. The string `data_compound_B523` specifies the start of a data block identified by the block code '`compound_B523`'.

Each data item is identified by a unique data name composed of a character string starting with an underline character '_'. Three examples of data names and their associated data items are:

```
_cell_volume  2310(2)

_chemical_formula_moiety 'C23 H36 O7'
_publ_contact_author
;
   Prof Barry O'Connell
   Department of Chemistry
   Building #57-M5
   University of Kalamazoo
   Michigan        USA.
;
```

The data items above are of different types: numeric, character and text, respectively. The STAR File syntax makes no distinction between the type of data item. Each item is treated simply as a contiguous character string delimited by matching blanks, single quotes, double quotes, or semicolons as the first character of a line. The order and format of these strings in the file are irrelevant, except for the requirement that the data name precede the data item. Data on a line following a hash character '#' is considered to be a comment, except if it is contained within a text string. Examples of a hash character used as part of a data item and as a comment precursor are given above and below.

A data item, or a set of data items, may be repeated in a list. Such data items are preceded by a '`loop_`' string. Here is a list of data items specifying the habit of a crystal.

```
loop_
 _exptl_crystal_face_index_h
 _exptl_crystal_face_index_k
 _exptl_crystal_face_index_l
 _exptl_crystal_face_perp_dist
 _exptl_crystal_face_name         # not std name
 _exptl_crystal_face_description # not std name
     0   0  -1 0.012 A 'well formed'
     0   0   1 0.012 B *
    -1   0   0 0.023 C uneven
     1   0   0 0.027 D 'needs further grinding'
     0   1   0 0.016 E *
     0  -1   0 0.015 F pitted
```

Any data item, independent of its type, may be included in a loop. The only requirement is that the number of data items in a loop must be an exact multiple of the number of data names in the loop definition.

The CIF syntax

The archival facilities provided by the STAR File process are general and open-ended. There is no restriction on the number of loop levels, the length of the file records (*i.e.* the lines of data) or on the length of data names. Syntax of this generality is unlikely to be needed in crystallography. It was therefore considered reasonable to impose restrictions on the STAR File syntax which will simplify the software required to generate or access a CIF. The advantages offered by these restrictions were considered to be sufficiently important from a computing standpoint to compensate for a loss of generality in file attributes not critical to crystallography.

The CIF restrictions to the STAR File syntax are:

1. Lines may not exceed 80 characters.
2. Data names and block codes may not exceed 32 characters. All data names and block codes are case insensitive, *i.e.* `_ABS` and `_abs` are treated identically.
3. In a STAR File, a data item may be of any data type. However, it simplifies processing if data types are known in advance. The CIF Dictionary identifies whether a CIF data item is a *number* or a *character*. The character and text fields are considered interchangeable.
4. A data item is assumed to be a *number* if it starts with a digit '0'–'9', plus '+', minus '–' or a period '.' and it is not bounded by matching single or double quotes or semicolons as the first character on a line.
5. A number may be supplied as an integer, as a floating-point number, or in scientific notation. When concatenated with an integer in parentheses, that integer is assumed to be the estimated standard deviation in the final digit(s) of the number. For example: 34.5, 3.45E1, 34.5(12), 3.45E1(12) are all versions of 34·5 with and without an e.s.d. of 1·2.
6. A data item is assumed to be of data type *text* if it extends over more than one line, *i.e.* it starts and ends with a semicolon as the first character of a line.
7. A data item is assumed to be of data type *character* if it is not a *number* or *text*.
8. Only one level of `loop_` is permitted. Additional levels of repeated data must be stored as lists within a text field.
9. Many numeric fields contain data for which the units must be known. Each CIF data item has a default units code which is stated in the CIF Dictionary. If a data item is not stored in the default units, the units code is appended to the data name. For example, the default units for a crystal cell dimension are ångströms. If it is necessary to include this data item in a CIF with the units of picometres, the data name of `_cell_length_a` is replaced by `_cell_length_a_pm`. Only those units defined in the CIF Dictionary are acceptable. The default units, except for the ångström, conform to the SI Standard adopted by the IUCr. These default units should be used whenever possible.

Although the CIF data name and block code definitions are restricted to 32 characters, this is adequate for the construction of self-explanatory names. Data names defined for use in a CIF are separated into components to represent an internal hierarchy of data categories. The concept of data name categories is not explicit in the STAR File process, but it arises naturally as part of data name design. Thus data names of the form `_<category>_<topic>_<subtopic>` provide for hierarchical classifications and are used throughout the CIF definitions. Sorting on the basis of hierarchical names generates a logical ordering for data names in the Dictionary.

Certain abbreviation conventions have been adopted in this paper, and in the CIF Dictionary, when referring to groups of data names. Use of only the `_<category>_` or `_<category>_<topic>_` components of a data name, while retaining the trailing underline character, refers to a category or subcategory of data names. For example, `_refln_` refers to all data items which have data names starting with this text string. Another commonly used abbreviation replaces the leading components of a data name with an asterisk. This provides a convenient shorthand method for referring to specific members of a category of data names. For example, when discussing data items in the `_chemical_formula_` category, one can refer simply to the `*_moiety` and `*_sum` items rather than the full data names. This abbreviation aids in the identification of individual data names.

CIF Dictionary

Each data item in a data block is identified by a unique data name. The currently accepted CIF data names are listed and defined in Appendix I as the CIF Dictionary (Core Version 1991). These are the IUCr 'standard' data items currently accepted for the submission of machine-readable documents to the IUCr and to the crystallographic databases. The data items in the Core Dictionary are intended primarily for use in the description of most small-molecule and inorganic structures. Future extensions to this Dictionary will define data items used in more specialized areas of crystallography, such as powder diffraction and macromolecular studies. Appendix II contains an example of a CIF employing data names from the Core Dictionary in the submission of a manuscript and data to *Acta Crystallographica* Section C (Willis, Beckwith & Tozer, 1991).

The Dictionary (Core Version 1991) is also available† as an electronic file **cifdic.C91**. This file, which has been constructed using the STAR Dictionary Definition Language (DDL) proposal of Cook (1991), contains more information about the data items than the printed Dictionary in Appendix I. It represents the current master reference file for checking the application of CIF data items. The CIF applications programs *CIFER* (Allen & Edgington, 1992) and *CYCLOPS* (Hall, 1991*b*), which are described later, employ this Dictionary for the validation and manipulation of standard data names.

It must be emphasized that the CIF Dictionary only contains the definitions of data items recognized by the IUCr as *standard* crystallographic data. In the future, data items will be added to the Dictionary but the definitions of existing standard items must *never* be changed. This ensures perpetual access in long-term archive files. In the passage of time it can be expected that some data items will fall into disuse and be replaced by more useful data names, but the connection between a standard data name and its definition, once it has been entered into the CIF Dictionary, must remain sacrosanct.

The contents of a CIF are not restricted to data items defined in the standard Dictionary. The presence of *non-standard* data does not affect the logical integrity of a CIF nor the access to the standard data. A fundamental property of a STAR File is that its logical integrity is quite independent of the order or nature of the data items. This means that local *and* standard data can be mixed in the same file. The only potential problem that could arise is if a local data name is subsequently adopted as a standard data name. Careful choice of local data names, perhaps containing a unique code, should avoid this possibility.

A CIF need not contain every data item listed in the Dictionary. All data items are optional and should be included only if required. A list of data items that are essential for submissions to IUCr journals is included in a new *Notes for Authors* [*Acta Cryst.* (1991), **C47**, 2266–2275] which includes rules for machine-readable manuscripts. Although the CIF allows data items to be in any order, it is good practice to group the items according to the data categories described below. Thus, all `_exptl_` items would be entered together to simplify the task of visual searching and editing.

Data name categories

The definition of data categories has been given above. Each category represents a major logical group of crystallographic data, such as crystal symmetry, associated chemistry, diffraction data, manuscript text, and so on. Here is a brief introduction to the current CIF data categories. More detailed descriptions are given in the Dictionary (Appendix I).

`_audit_` data provide a record of the CIF creation and subsequent updating. These items usually precede all others in the CIF.

`_atom_` data are in two separate categories: those that describe atom sites in a crystal structure (*i.e.* `_atom_site_` data names) and those that describe the properties of the atom types that occupy these sites (*i.e.* `_atom_type_` data names).

`_cell_` data record the cell parameters, method of measurement, conditions *etc.*

`_chemical_` data specify the composition and chemical properties of the compound. The `_chemical_formula_` items must agree with those that specify the density, unit-cell and Z values.

`_chemical_conn_` data specify the 2D chemical structure for molecular species.

`_computing_` data record the computer programs used in the crystal structure analysis.

`_database_` data are only specified by database managers and should only appear in a CIF if they originate from this source.

---

† From the Technical Editor, International Union of Crystallography, 5 Abbey Square, Chester CH1 2HU, England. Fax: (44) 244 314888. Email: teched@iucr.ac.uk

`_diffrn_` data are the diffraction measurements.

`_exptl_` data record the crystal measurements, such as density, shape, size *etc*.

`_geom_` data describe the standard molecular and crystal geometry, as calculated from the contents of the `_atom_`, `_cell_` and `_symmetry_` data.

`journal_` data are entries used by the journals' staff in processing a CIF.

`_publ_` data are used when submitting CIF data to a journal for publication.

`_refine_` data describe the structure refinement parameters.

`_refln_` and `_reflns_` data specify the reflection items used to determine the `_atom_` data items. They are in two categories: `_refln_` and `_reflns_` items. The `_reflns_` data specify the parameters that apply to all reflections. The `_refln_` data refer to individual reflections and must be included in looped lists. The `_reflns_` data items are not looped.

`_symmetry_` data specify the space-group symmetry.

Looped and non-looped data items

The STAR File format permits any data item to be included in a `loop_` list. This is also the case in a CIF. However, only data items which need to be repeated should appear in a looped list. Looped lists are composed of data items with common properties. For example, atom site information (represented by `_atom_site_` items) may appear in one list and reflection data (represented by `_refln_` items) in another, but they cannot be mixed together. Appendix II should be consulted for examples of looped lists.

Note that it is essential to loop repeated data items, rather than declaring them as a series of single data items. This is because the STAR format prohibits a data name from appearing more than once in a data block. The Dictionary identifies which individual data items will normally appear in looped lists. In most cases this is obvious from the nature of the data and its definition. Those data categories where individual items often appear in looped lists are summarized below:

`_atom_site_`	atomic site parameters
`_atom_site_aniso_`	atomic site anisotropic U values
`_atom_type_`	atom species
`_cell_measurement_refln_`	reflection data used in the unit-cell measurement
`_chemical_conn_`	chemical properties in connectivity table
`_diffrn_attenuator_`	diffraction attenuator scales
`_diffrn_orient_refln_`	reflections that define the diffractometer orientation matrix
`_diffrn_radiation_wavelength_`	discrete wavelengths used in diffraction measurements
`_diffrn_refln_`	diffraction intensity data
`_diffrn_scale_group_`	scale factors for groups of diffraction intensities
`_diffrn_standard_refln_`	standard reflections for scaling diffraction intensities
`_exptl_crystal_`	separate loops for crystals and crystal faces
`_geom_angle_`	interatomic angles
`_geom_bond_`	bond distances
`_geom_contact_`	contact distances
`_geom_torsion_`	torsion angles
`_publ_author_address`	authors' addresses
`_publ_author_name`	authors' names
`_refln_`	structure-factor reflection data
`_reflns_scale_`	scale factors for groups of structure-factor data
`_symmetry_equiv_pos_as_xyz`	space-group equivalent positions

CIF standard codes

It is particularly important that certain data items be specified in a CIF as standard codes. These codes aid in the simple and unambiguous interpretation of a data item, and should be used wherever applicable. For example, the data item which describes the method by which hydrogen atoms are treated in the least-squares refinement process is associated with the data name `_refine_ls_hydrogen_treatment` and is assigned the CIF standard codes:

`refall`	refined all H parameters
`refxyz`	refined H coordinates only
`refU`	refined H U's only
`noref`	no refinement of H parameters

If an assigned CIF standard code is inappropriate, it is normal practice to add a text field of the type `*_special_details`. A complete list of the currently defined CIF data names with standard codes is given in Section 3 of Appendix I.

Atomic site identification

The identification and labelling of atomic sites is vital in crystallography, both for the structure determination process, and for the description and interpretation of the final results. The atom labelling scheme adopted in the CIF embraces a very wide range of the common styles and conventions in current use. The scheme is fully detailed in Section 4 of the CIF Dictionary (Appendix I), but short introductory notes are also given here.

The labelling of atoms in most crystallographic studies serves two distinct purposes: (*a*) to identify a site in the crystal, and (*b*) to identify the chemical element that occupies that site. The former is used, for example, to link the site to the corresponding set of scattering-factor coefficients, whilst the latter is used in descriptions of the structure which are related to its chemistry, *e.g.* lists of geometrical data. The CIF Dictionary makes this distinction clear by defining `_atom_site_` and `_atom_type_` as separate data categories. The connection between the two is made through the identical data items `_atom_site_type_symbol` and `_atom_type_symbol`. However, normal crystallographic practice is to use a single label to define both the site and the chemical species occupying it. The CIF definition also makes provision for this. Thus, the `_atom_site_label` has a structure (described in detail in Section 4 of Appendix I) that can start with an element symbol (optionally followed by an oxidation state) followed by an identifier. Together these two parts uniquely define each site, with the first part, termed component 0, identically matching one of the `_atom_type_symbol` codes.

The `_atom_site_label` may contain up to six further components, in addition to component 0 described above, to provide for full and explicit site identification. Component 1, the atom number code, would normally be present and is simply concatenated with component 0. Other components, *e.g.* the residue code, the sequence code *etc.*, permit the construction of specialist `_atom_site_label` codes. These additional components are linked to previous ones by the underline character.

The link between the `_atom_site_` and `_atom_type_` data items can be given in two ways: (*a*) if the `_atom_site_type_symbol` is given, then it must match identically one of the `_atom_type_symbol` values, or (*b*) if it is not given, then the first part (component 0) of the `_atom_site_label` must match one of the `_atom_type_symbol` codes. In this way, the common practice of identifying atom sites as *e.g.* `C1`, `Hg7`, `H1C1` *etc.* is preserved.

This site labelling scheme can also be used for sites that are occupationally disordered, either by defining `_atom_type_` items with the appropriate average properties, or by defining two atom sites that have identical coordinates, but which point to different atom types.

Creating a CIF

A large proportion of the data included in a CIF will be generated by crystallographic software packages. The CIF example shown in Appendix II was created for publication in *Acta Crystallographica* Section C by the program *CIFIO* (Hall, 1990). However, it also contains data added manually by the authors (Willis, Beckwith & Tozer, 1991). The majority of data items in this example were requested specifically (from *CIFIO*) for a test *Acta Crystallographica* submission. Note that the missing items in this example are flagged with a '?'. Programs such as *CIFIO* can only output the data items present in their internal files and the '?' flag represents a simple mechanism for identifying missing items.

The use of a '?' as a missing data flag is important for two reasons: it signals the inaccessibility of a data item to the generating software, and it satisfies the STAR File requirement that each data name must be matched with a data value. The use of a '?' enables data items from other sources to be added later either manually or by software. For example, most `_publ_` items will be entered manually. Manuscripts can be entered by the authors either as ASCII text or as word-processed modules (see `_publ_manuscript_` fields in Appendix I). Journal staff will also, in consultation with authors, make changes in response to the detection of incorrect numerical data or to the comments of referees or Co-editors.

The provision for transmitting graphical publication data within a CIF is being investigated. The inclusion of ASCII graphical modules suitable for producing direct laser or plot diagrams (*e.g.* using HPGL or PostScript) has already been tested. It will also be possible for some diagrams to be generated directly from the `_atom_` and `_chemical_conn_` data by the journal staff. In the future it is expected that a CIF may contain encapsulated PostScript or $\TeX$ modules of text that can be transferred directly to the journals' word processing facility.

Apart from the generation of CIF's by crystallographic packages, specific standalone software has been developed to update, check and access CIF's. The Cambridge Crystallographic Data Centre is developing a program *CIFER* (Allen & Edgington, 1992) to facilitate the update and creation of a CIF *via* a menu-driven interface. This will enable a CIF to be upgraded interactively with 2D diagrams and other missing data. The program *EDCIF-J* (Osaki, 1991) is also designed for the interactive creation of a publication CIF. Another program, *CYCLOPS* (Hall, 1991*b*), uses the Dictionary file **cifdic.C91** to validate data names in any text file, including program source code. This is an essential requirement for software developers wishing to adhere to the standard data names. Finally, the program *QUASAR* (Hall & Sievers, 1990) is designed to extract requested data items from a CIF in the form of another STAR File (further details of this program are given below).

It should be emphasized that CIF's are not intended solely for publication purposes. They are well suited to local archiving and data exchange. Which data are stored in a CIF, and in what order, is very much up to the user. Any CIF, whether for local or global applications, may contain a mixture of standard (IUCr defined) and non-standard (locally defined) data. This will not affect access to either category of data.

It is possible to check or to extract data from a CIF using a text editor. However, a more systematic approach is to use specific software for these purposes, such as provided by the programs *CYCLOPS* and *QUASAR*. In the next section the program *QUASAR* is used to illustrate a typical approach to accessing data in a CIF.

Accessing a CIF: the program *QUASAR*

A data file conforming to the STAR File syntax is accessed by locating the appropriate data names. That is, search procedures rely on the grammatical rules of the STAR format to find the relevant data. This method of searching a file, referred to as parsing, is used by *QUASAR* (Hall &

Table 2. *Accessing a CIF using QUASAR*

(*a*) The *QUASAR* request file.

```
star_arc_toz.cif
star_out_toz.out

data_TOZ

_audit_creation_date

_chemical_name_systematic
_chemical_formula_sum

_cell_length_a
_cell_length_b
_cell_length_c
_cell_angle_alpha
_cell_angle_beta
_cell_angle_gamma

_reflns_scale_group_code
_reflns_scale_meas_F

_symmetry_equiv_pos_as_xyz
```

```
_atom_type_symbol
_atom_type_number_in_cell
_atom_type_scat_dispersion_real
_atom_type_scat_dispersion_imag


_atom_site_label
_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z
_atom_site_test_rubbish     #<<< ask for item not there
_atom_site_thermal_displace_type
_atom_site_U_iso_or_equiv

_refln_index_h
_refln_index_k
_refln_index_l
_refln_F_meas
_refln_F_sigma
_refln_observed_status
_refln_scale_group_code
```

(*b*) The *QUASAR* output file **toz.out**.

```
data_TOZ

_audit_creation_date                    91-03-20

_chemical_name_systematic
;trans-3-Benzoyl-2-(tert-butyl)-4-(iso-butyl)-
1,3-oxazolidin-5-one
;
_chemical_formula_sum                 'C18 H25 N O3'


_cell_length_a                          5.959(1)
_cell_length_b                          14.956(1)
_cell_length_c                          19.737(3)

_cell_angle_alpha                       90
_cell_angle_beta                        90
_cell_angle_gamma                       90

loop_
_reflns_scale_group_code
_reflns_scale_meas_F
              1 .960926

loop_
 symmetry_equiv_pos_as_xyz
        +x,+y,+z
  1/2-x,-y,1/2+z
  1/2+x,1/2-y,-z
  -x,1/2+y,1/2-z

loop_
_atom_type_symbol
_atom_type_number_in_cell
_atom_type_scat_dispersion_real
_atom_type_scat_dispersion_imag
  C  72 .017 .009
  H 100    0    0
  O  12 .047 .032
  N   4 .029 .018
```

```
loop_
 atom_site_label
 atom_site_fract_x
 atom_site_fract_y
_atom_site_fract_z
_atom_site_test_rubbish     # requested item not present
_atom_site_thermal_displace_type
_atom_site_U_iso_or_equiv
  C1     .4154(4) .5699(1) .3026(1) ? Uani  .060(1)
  C2     .5630(5) .5087(2) .3246(1) ? Uani  .060(2)
  C3     .5350(5) .4920(2) .3997(1) ? Uani  .048(1)
  N4     .3570(3) .5558(1) .4167(1) ? Uani  .039(1)
  C5     .3000(5) .6122(2) .3581(1) ? Uani  .045(1)
  O21    .6958(5) .4738(2) .2874(1) ? Uani  .090(2)
  C31    .4869(6) .3929(2) .4143(2) ? Uani  .059(2)
  C32    .2552(7) .3558(2) .3953(2) ? Uani  .073(2)
  C321    .209(1) .3542(4) .3211(3) ? Uani  .111(4)
  C322    .230(1) .2626(3) .4264(3) ? Uani  .149(5)
  C41    .2034(4) .5476(2) .4682(1) ? Uani  .041(1)
  H321C    .04(1)   .318(3)  .320(2) ? Uiso .14000
  H322A    .25(1)   .272(4)  .475(3) ? Uiso .19000
  H322B    .34976    .22118   .40954 ? Uiso .19000
  H322C    .08(1)   .234(4)  .397(3) ? Uiso .19000
  H412   -.007(6)   .447(2)  .552(2) ? Uiso .08000
  H513B   .115(7)   .757(3)  .426(2) ? Uiso .09000
  H513C   .329(6)   .817(2)  .430(2) ? Uiso .09000

loop_
_refln_index_h
 refln_index_k
_refln_index_l
_refln_F_meas
_refln_F_sigma
_refln_observed_status
_refln_scale_group_code
  ? ? ? ? ? ? ?

#               -----end-of-data-block-----
```

Sievers, 1990). It is described here as a suitable template for other CIF manipulation software.

QUASAR retrieves those data items contained in a request list of data names. As part of the access procedure it (*a*) outputs data items and data blocks in the order requested, (*b*) automatically checks the input file for logical integrity, (*c*) outputs the requested items in STAR File format (*i.e.* it spawns another CIF containing requested items), (*d*) permits multiple requests for a data item within the same data block, and (*e*) processes multiple data blocks in a single run.

An example of a *QUASAR* run is shown in Table 2.

The first two lines of the request list in Table 2(*a*) specify the input and output STAR File names. These are input as extensions to the strings `star_arc_` and `star_out_`, respectively. This example shows how specific data are extracted from the file **toz.cif** shown in Appendix II. The contents of the *QUASAR* output file, **toz.out**, are shown in Table 2(*b*). The contents of Tables 2(*a*) and (*b*) should be carefully compared with the data items in Appendix II.

Concluding remarks

Crystallography, along with other computer-intensive disciplines, has entered an era in which the archiving and electronic exchange of data are of paramount importance. It is these basic needs of storage and transportability of data in a machine-independent form that have driven the development of CIF. The CIF Core Dictionary provides a set of data names, data items and definitions that are fundamental to crystallography. Future versions of the Dictionary will contain extensions to this core that cover data items of importance in more specific areas of the subject, *e.g.* powder diffraction, macromolecular crystallography *etc.* Developments parallel to those of CIF have led to the definition of a standard molecular data (SMD) format (Barnard, 1990) for the exchange of chemical connectivity and reaction data. SMD addresses the need to transfer data between databases and the various software systems for molecular modelling and synthesis planning. Since there is an overlap of information between CIF and SMD, it is hoped that some degree of integration can be achieved between the two formats.

The provision of a common standard format for machine-readable submissions to *Acta Crystallographica* and to the crystallographic databases has been a particular aim of the CIF development. These submissions require only a restricted subset of CIF data names for manuscript preparation, many of which can be generated automatically. We would encourage those who are responsible for the development of crystallographic structure determination packages to provide output CIF files as soon as possible. Further, we envisage that software tools will rapidly become available to assist in the upgrading of a program-generated CIF to publication standard. The submission, processing and publication procedures for CIF manuscripts directed to *Acta Crystallographica* will be announced in the November issue of Section C [*Acta Cryst.* (1991), C**47**, 2266–2275]. The announcement will indicate those CIF data items that are normally regarded as essential in manuscript submission. The announcement will also indicate how authors without electronic mail or diskette facilities will benefit from the new procedures.

There is no doubt that the standardization of data exchange formats is overdue. Existing approaches are inadequate because they sacrifice portability, flexibility and extensibility for computational efficiency. Today's and tomorrow's computer technology relegate the latter consideration to a much lower priority. The STAR File, and its specific application to the CIF, are seen as forerunners in the development of new tools for local and global data exchange.

The authors wish to thank the IUCr Executive, members of various IUCr Commissions, the IUCr Working Party on Crystallographic Information, and a number of other crystallographers for their constructive comments and active encouragement during the course of this development. In particular we acknowledge the input and support of Enrique Abola, Mike Dacombe, Paula Fitzgerald, Howard Flack, Richard Goddard, Carl Krüger, Brian McMahon, Ted Maslen, George Sheldrick, Rolf Sievers and Jim Stewart which contributed directly to the CIF construction. Without this cooperation the relatively swift completion of the central core of this project would not have been possible. We are indebted to Hazel Woodley (Cambridge) for her careful editing of the many revisions to this manuscript.

Notes

The STAR File process is the subject of a patent application by the International Union of Crystallography and S. R. Hall. The CIF specifications and Dictionary are the Copyright of the International Union of Crystallography. These decisions protect STAR File and/or CIF from use for commercial gain without prior permission, and reflect the involvement of the IUCr in supporting this development. STAR File and CIF may be used freely by individuals for their intended purposes, and for the preparation of deposition documents for non-IUCr journals. Requests to use STAR File and/or CIF for any other purpose should be addressed to: Dr J. N. King, Executive Secretary, International Union of Crystallography, 5 Abbey Square, Chester CH1 2HU, England.

The computer program source code for *QUASAR* and *CYCLOPS*, and the CIF Dictionary (Core Version 1991) in printed form, and as the electronic file **cifdic.C91**, are available from the Technical Editor, International Union of Crystallography, 5 Abbey Square, Chester CH1 2HU, England (email: teched@iucr.ac.uk).

APPENDIX I

CIF Dictionary (Core Version 1991)

1. *Introduction*

This version of the CIF Dictionary contains the detailed definitions of data names which are acceptable in submissions to the IUCr and to the crystallographic databases. Data names are considered to be case insensitive: they may be given in upper- or lower-case letters, or in any combination of upper and lower case. The data name definitions are ordered alphabetically by the data category; general notes on these categories are given in Section 2 of this Appendix.

Certain abbreviation conventions have been adopted in the CIF Dictionary when referring to groups of data names. Use of only the `_<category>_` or `_<category>_<topic>_` components of a data name, while retaining the trailing underline character, refers to a category or subcategory of data names. For example, `_refln_` refers to all data items which have data names starting with this text string. Another commonly used abbreviation replaces the leading components of a data name with an asterisk. This provides a convenient shorthand method for referring to specific members of a category of data names. For example, when discussing data items in the `_chemical_formula_` category, one can refer simply to the `*_moiety` and `*_sum` items rather than the full data names. This abbreviation aids in the identification of individual data names.

Literature references that are required for the definition of a data item are included in full within the Dictionary, in order that it can be distributed as a standalone document.

The CIF Dictionary contains information about the permitted units for numerical data items. Default units do not require any extensions to be appended to the data name. These defaults, except for the ångström unit, conform to the SI standard adopted by the IUCr. Default units should be used wherever possible; they *must* be used in submissions to *Acta Crystallographica.*

Simple typesetting conventions have been adopted for use with CIF data. These are listed in a table below. These conventions are particularly important in text submissions to *Acta Crystallographica.* The list will be extended as need arises and reported in future versions of the Dictionary and in the *Notes for Authors* which will be published annually in *Acta.* Typesetting signals are important in the free-text fields, such as `_publ_section_`, and also in fields such as `_chemical_name_` or in the construction of atom labels for certain classes of compounds, *e.g.* amino acids and peptides.

Greek letters have been assigned a single-character ASCII alphabetic equivalent. As far as possible, this is the first letter of the fully spelled name of each Greek letter. The exceptions are marked * in the list below. Greek letter codes are preceded by a backslash '\'; lower-case Greek letters use the code in lower case, upper-case Greek letters use the code in upper case.

2. *CIF data categories*

`_audit_` *data names*

The `_audit_` data items provide a record of the CIF creation and subsequent updating. These items usually precede all others in the CIF.

`_atom_` *data names*

The `_atom_` data names are in two separate categories: those that describe atom sites in a crystal structure (*i.e.* `_atom_site_` data names), and those that describe the properties of the atom types that occupy these sites (*i.e.* `_atom_type_` data names).

The `_atom_type_` data provide information on the chemical identity, scattering factors, atomic radii and so on. The `_atom_site_` items describe specific information on atomic sites such as positional coordinates, atomic displacement parameters, magnetic moments and directions, and so on.

α	A	`\a`	`\A`	alpha	
β	B	`\b`	`\B`	beta	
χ	X	`\c`	`\C`	chi	
δ	Δ	`\d`	`\D`	delta	
ε	E	`\e`	`\E`	epsilon	
φ	Φ	`\f`	`\F`	phi	*
γ	Γ	`\g`	`\G`	gamma	
η	H	`\h`	`\H`	eta	*
ι	I	`\i`	`\I`	iota	
κ	K	`\k`	`\K`	kappa	
λ	Λ	`\l`	`\L`	lambda	
μ	M	`\m`	`\M`	mu	
ν	N	`\n`	`\N`	nu	
ο	O	`\o`	`\O`	omicron	
π	Π	`\p`	`\P`	pi	
θ	Θ	`\q`	`\Q`	theta	*
ρ	R	`\r`	`\R`	rho	
σ	Σ	`\s`	`\S`	sigma	
τ	T	`\t`	`\T`	tau	
υ	Υ	`\u`	`\U`	upsilon	
ω	Ω	`\w`	`\W`	omega	*
ξ	Ξ	`\x`	`\X`	xi	
ψ	Ψ	`\y`	`\Y`	psi	*
ζ	Z	`\z`	`\Z`	zeta	

superscripts	`Csp^3^`	for	Csp^3
subscripts	`U~eq~`	for	U$_{eq}$
acute accent	`\'e`	for	é
grave accent	`` \`a ``	for	à
circumflex	`\^e`	for	ê
cedilla	`\,c`	for	ç
umlaut	`\"u`	for	ü
degree	`120\%`	for	120°
ångström	`1.54\%A`	for	1.54 Å

The `_atom_type_` data are global. They apply to one or more atom sites. The link to the atom site data is provided through the data names `_atom_type_symbol` and `_atom_site_type_symbol`. These items provide the common character codes which identify atom types. Normally these codes are element symbols but they can include the oxidation state or any other information that uniquely identifies the atom types present in the structure.

If the `_atom_site_type_symbol` data is supplied in an atom site list, it must match with one of the `_atom_type_symbol` codes. Alternatively, if the `_atom_site_type_symbol` is not supplied, the leading characters of the `_atom_site_label` must match with one of the `_atom_type_symbol` codes. Note that the `_atom_site_type_symbol` has precedence over the `_atom_site_label` for the purpose of linking with atom type data and, if the former is present, the latter need not contain an atom type code. The rules for specifying the `_atom_site_label` are given in Section 4. When several atom species share the same site, as is commonly found in mineral structures, two different approaches may be used. Atom types may be defined separately with unique symbol codes. A multiply occupied atom site is then specified as two or more atom sites with the same coordinates but different `_atom_site_type_symbol` (or `_atom_site_label`) codes. With this approach `_atom_site_occupancy` values must add up to unity or less. The alternative approach is to specify an `_atom_type_symbol` to identify the properties of the combined atomic species sharing the site. In this case only a single entry for each atom site is needed.

`_cell_` *data names*

These data specify the cell parameters, together with the method of measurement, experimental conditions, *etc*.

`_chemical_` *data names*

These data specify the composition and chemical properties of the compound. The formula data items must agree with those that specify the density, unit-cell and Z values.

The following rules apply to the construction of the data items `_chemical_formula_analytical`, `*_structural` and `*_sum`. For the data item `*_moiety` the formula construction is broken up into residues or moieties, *i.e.* groups of atoms that form a molecular unit or molecular ion. The rules given below apply within each moiety but different requirements apply to the way that moieties are connected (see `_chemical_formula_moiety`).

1. Only recognized element symbols may be used. The symbol D is used for deuterium.
2. Each element symbol is followed (without a space) by an integer or decimal 'count' number. A count of '1' may be omitted.
3. A space or parenthesis must separate each element symbol and its count from the next element symbol.
4. Where a group of elements is enclosed in parentheses, the multiplier for the group must follow the closing parentheses. That is, all element and group multipliers are assumed to be printed as subscripted numbers. An exception to this rule exists for `*_moiety` formula where pre- and post-multipliers are permitted for molecular units.
5. Unless the elements are ordered in a manner that corresponds to their chemical structure, as in `_chemical_formula_structural`, the order of the elements should be: C, H, followed by the other elements (including deuterium) in alphabetical order of their symbol. This is the Hill system used by *Chemical Abstracts*. This ordering is used in `*_analytical`, `*_sum` and within the molecular units of `*_moiety`.

`_chemical_conn_` *data names*

The `_chemical_conn_` data items specify the 2D chemical structure of the molecular species. They allow a 2D chemical diagram to be reconstructed for use in a publication or in a database search for structural and substructural relationships.

The chemical connectivity specification uses two related lists of looped data. These are the *atom* list and the *bond* list.

The *atom* data items provide information about the chemical properties of the atoms in the structure. In cases where crystallographic and molecular symmetry elements coincide it must also contain symmetry-generated atoms, so that the `_chemical_conn_` data items will always describe a complete chemical entity. The *bond* data items specify the connections between the atoms in the atom list and the nature of the chemical bond between these sites.

`_computing_` *data names*

These items identify the computer programs used in the crystal structure analysis.

`_database_` *data names*

These codes are assigned by database managers and should only appear in a CIF if they originate from this source.

`_diffrn_` *data names*

These items record details of the diffraction data and its measurement.

`_exptl_` *data names*

These items record experimental measurements on the crystal, such as shape, size, density *etc*.

`_geom_` *data names*

These data items provide information on the molecular and crystal geometry, as calculated from the contents of the `_atom_`, `_cell_` and `_symmetry_` data. Geometry data is therefore redundant in that it can be calculated from other more fundamental quantities in the CIF. It serves, however, the dual purpose of providing a check on the correctness of both sets of data, and of enabling the most important geometric data to be identified for publication by setting the `*_publ_flag`.

Geometry data types that are not defined explicitly in the CIF Dictionary may be entered as `_geom_special_details`.

`_journal_` *data names*

These are the book keeping entries used by the journals staff when processing a CIF submitted for publication. Normally the creator of a CIF will not specify these data. The data names are not defined in the Dictionary because they are for journal use only.

`_publ_` *data names*

These items are used when submitting a manuscript to a journal for publication.

`_refine_` *data names*

These items describe the structure refinement parameters.

`_refln_` *and* `_reflns_` *data names*

These names specify the reflection data used to determine the `_atom_` data items. They exist in two categories: `_refln_` and `_reflns_` items. The `_reflns_` data specify the parameters that apply to all reflections. The `_refln_` data refer to individual reflections and must be included in looped lists. The `_reflns_` data are not looped.

`_symmetry_` *data names*

These items specify the space-group symmetry.

3. *Standard codes table*

Recognized codes are provided for specific data items. Definitions of the codes are included within the CIF data name descriptions of Section 5. Only the codes shown there may be used. If one of these codes does not adequately identify the condition of a parameter, include this information in another data item (*e.g.* in `*_special_details` fields). For convenience we list here those data names for which standard codes are provided.

```
_atom_site_calc_flag
_atom_site_refinement_flags
_atom_site_thermal_displace_type
_atom_sites_solution_hydrogens
_atom_sites_solution_primary
_atom_sites_solution_secondary
_chemical_conn_bond_type
_diffrn_refln_scan_mode
_diffrn_refln_scan_mode_backgd
_exptl_absorpt_correction_type
_refine_ls_hydrogen_treatment
_refine_ls_matrix_type
_refine_ls_structure_factor_coef
_refine_ls_weighting_scheme
_refln_observed_status
_refln_refinement_status
_symmetry_cell_setting
```

4. *Atom label definition*

The `_atom_site_label` is the unique identifier of a specific site in the crystal structure which contains a particular atomic type or combination of atom types. There may be more than one `_atom_site_label` referring to the same position in the crystal structure. This is one approach to specifying shared atom sites. The other is to specify a single site containing a mixture of atom types in a fixed proportion defined by `_atom_type_description`.

The `_atom_site_label` may be constructed from up to seven distinct components, 0 to 6. These components are concatenated in sequential order from left to right. The `_atom_site_label` must contain a component 0 code. All other components are optional. Components 0 and 1 are concatenated; all other components are joined by an underline '_' character. These underlines must be included up to the highest-order component present (*i.e.* if a lower-order component is omitted the '_' separator must still be inserted in order to maintain the component ranking). An underline character can never be used within a component code itself.

For most applications component 0 of the atom label is a code that identifies the 'type' of atom, or atoms, at the atomic site. It must therefore match one of the specified `_atom_type_symbol` codes in the `_atom_type_` list. However, if the data item `_atom_site_type_symbol` is also specified, component 0 of the atom label is not used to identify the atom type and it may contain any code which is consistent with the construction rules cited below. In other words, the `_atom_site_type_symbol`, if specified, takes precedence over the `_atom_site_label` component 0 code in the role of linking the `_atom_site_` list to the `_atom_type_` list.

The `_atom_site_label` construction is flexible, visually decipherable and well suited to computer applications. The components can be easily identified and stripped with a single pass, from left to right, along the label string. Note that the underline separators are only used if higher-order components exist. If intermediate components are not used they may be omitted provided the underline separators are inserted. For example the label '`C233__ggg`' is acceptable and decodes as the components 0: '`C`', 1: '`233`', 2: '', and 3: '`ggg`'. There is no requirement that the same number of components be used in each label.

Components of _atom_site_label

Component 0: [optionally identical to an `_atom_type_symbol`] (*mandatory*)

A character string containing any character except a blank or an underline, with the proviso that each digit '0'–'9' be used only to designate an oxidation state and, as such, must be followed by a plus '+' or a minus '–' character. It is recommended that the element symbols be used when applicable. Permissible codes are: `Ni  Cu2+  dummy  Fe3+Ni2+  S-  H*  H(SDS)`

Component 1: [*atom number* code] (*optional*)

This string may contain any alphanumeric character except a blank or an underline '_' but the first character must be a digit '0'–'9' and the second character may not be a plus '+' or a minus '–'. It is intended primarily to differentiate sites containing the same atom type, but can be used for any purpose whatsoever. This string is concatenated directly with the `_atom_site_type_symbol`. Examples of combined component 0 and 1 codes are: `C1  C103g28  Fe3+17b  H*251  boron2a  Ni22+  Ni2+2  Fe2+Ni2+7`, where the component 0 is underlined to indicate how these labels are parsed.

An underline character is inserted if components beyond 1 are included in the label.

Component 2: [*identifier* code] (*optional*)

This string may contain any character except a blank or underline. It is intended primarily to identify specific structural information in a macromolecular fragment, but may be used for any other purpose as well.

An underline character is inserted if components beyond 2 are included in the label.

Components 3–6: [*residue, sequence number, chain-order, alternate* codes] (*optional*)

These strings may contain any character except a blank or an underline.

Underline characters are inserted after each component, 3 to 5, included in the label.

5. *CIF data name definitions*

The CIF definition of each data item in this dictionary contains: (*a*) an identifying data name, (*b*) a data type code, (*c*) a description, (*d*) optional parameters, and (*e*) optional example(s).

(*a*) The data name appears at the top of the definition in a bold typewriter face.

(*b*) The data type code appears to the right of the data name in italics and is bounded by parentheses. The possible type codes are 'char' and 'numb'. The 'char' code signals that the data item may be represented by *either* a single-line character string bounded by matching blanks, single quotes or double quotes, *or* multi-line text bounded by a semicolon as the first character of the bounding lines. The 'numb' code indicates that the data item is a number in integer, decimal or scientific notation.

(*c*) The description of the data item appears immediately below the data name in roman type. The description indicates the purpose of the data item and its relationship to other data items. References to the original definition of the data item are provided where appropriate.

(*d*) Parameters which specify the way in which the data item may be used follow the description, in a smaller roman typeface. These parameters appear in the definition as standard descriptive phrases. The meanings of these phrases are given below.

(i) "Appearance in list: [*yes* | *no* | *both*]." specifies whether the data item may be included in a repeated list of data items (that is, as a member of data item(s) preceded by a '`loop_`' command). Note that 'both' refers to a data item which is normally a single value but in special circumstances may also appear in a looped list.

(ii) "If looped, [*_data_name*] must be present in the same list." specifies another data item that *must* appear in the same looped list in order that the currently defined item may be correctly accessed.

(iii) "Where no value is given, the assumed value is '[*]'." specifies the string '*' which is assumed to be the default entry when this data item is absent from the CIF.

(iv) "The permitted range is [*min*]→[*max*]." specifies the minimum and maximum numbers permitted for this data item.

(v) "E.s.d. expected: [*yes* | *no*]." specifies whether an estimated standard deviation value, bounded by parentheses, is expected to be concatenated to a numerical data item.

(vi) "Default e.s.d. value: [*n*]." specifies the assumed estimated standard deviation '*n*' when a value is not appended to the data item.

(vii) "The units extensions are: ' ' ([*default units*] *1.0) '[*_ext*]' ([*alternative units*] [* | / | +][*con*])." specifies permitted units for dimensioned quantities. The first entry gives the default units which do not require a data name extension or a conversion factor. The second and succeeding entries specify the data name extension, the alternative units and the factor '*con*' needed to convert these units into the default. The symbol '*' signals a multiplication; the symbol '/' a division; and the '+' an addition of the value '*con*'.

(*e*) Examples of a data item may follow the parameter information in a typewriter face. Note that commas are used to separate different examples of the data item. A string containing blanks and bounded by quotes represents a single example. Sometimes an explanation of the example is provided in parentheses.

Version: **CIF Dictionary (Core 1991)**

`_atom_site_aniso_label` *(char)*

Anisotropic atomic displacement parameters are usually looped in a separate list. If this is the case, this code must match the `_atom_site_label` of the associated atom coordinate list and conform with the same rules described in `_atom_site_label`.

Appearance in list: yes.

`_atom_site_aniso_type_symbol` *(char)*

This `_atom_type_symbol` code links the anisotropic atom parameters to the atom type data associated with this site and must match one of the `_atom_type_symbol` codes in this list.

Appearance in list: yes. If looped, `_atom_site_aniso_label` must be present in the same list.

`_atom_site_aniso_U_11`
`_atom_site_aniso_U_12`
`_atom_site_aniso_U_13`
`_atom_site_aniso_U_22`
`_atom_site_aniso_U_23`
`_atom_site_aniso_U_33` *(numb)*

These are the standard anisotropic atomic displacement components which appear in the structure factor term: $\exp(-2\pi^2 \sum_i \sum_j U_{ij} h_i h_j a_i^* a_j^*)$. The components may be entered in any order.

Appearance in list: yes. If looped, `_atom_site_aniso_label` must be present in the same list. E.s.d. expected: yes. Default e.s.d. value: 0·0. The units extensions are: ' ' (ångströms squared *1.0) '`_pm`' (picometres squared /10000.) '`_nm`' (nanometres squared *100.).

`_atom_site_attached_hydrogens` *(numb)*

The number of hydrogen atoms attached to the atom at this site excluding any H atoms for which coordinates (measured or calculated) are given.

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. Where no value is given, the assumed value is '`0`'. The permitted range is 0→4.

Example(s): `2` (water oxygen), `1` (hydroxyl oxygen), `4` (ammonium nitrogen)

`_atom_site_calc_attached_atom` *(char)*

The `_atom_site_label` of the atom site to which the 'geometry-calculated' atom site is attached.

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. Where no value is given, the assumed value is '`.`'.

`_atom_site_calc_flag` *(char)*

A standard code to signal if the site data has been determined by diffraction data or calculated from the geometry of surrounding sites, or has been assigned dummy coordinates.

`d`	determined from diffraction measurements
`calc`	calculated from molecular geometry
`dum`	dummy site with meaningless coordinates

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. Where no value is given, the assumed value is `'d'`.

_atom_site_Cartn_x
_atom_site_Cartn_y
_atom_site_Cartn_z *(numb)*

The atom site coordinates specified according to a set of orthogonal Cartesian axes related to the cell axes as specified by the `_atom_sites_Cartn_transform_axes` description.

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. E.s.d. expected: yes. Default e.s.d. value: 0·0. The units extensions are: ‘ ’ (ångströms *1.0) ‘`_pm`’ (picometres /100.) ‘`_nm`’ (nanometres *10.).

_atom_site_chemical_conn_number *(numb)*

This number links an atom site to the chemical connectivity list. It must match a number specified by `_chemical_conn_atom_number`.

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. The permitted range is 1→ ∞.

_atom_site_constraints *(char)*

A description of the constraints applied to parameters at this site during refinement. See also `_atom_site_refinement_flags` and `_refine_ls_number_constraints`.

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. Where no value is given, the assumed value is `'.'`.

Example(s): `pop=1.0-pop(Zn3)`

_atom_site_description *(char)*

A description of special aspects of this site. See also `_atom_site_refinement_flags`.

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. Where no value is given, the assumed value is `'.'`.

Example(s): `'Ag/Si disordered'`

_atom_site_disorder_group *(char)*

A code to link disordered atom sites of a group that exist simultaneously in the crystal structure.

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. Where no value is given, the assumed value is `'.'`.

Example(s): `A`

_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z *(numb)*

Atom site coordinates as fractions of the `_cell_length_` values.

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. E.s.d. expected: yes. Default e.s.d. value: 0·0. Where no value is given, the assumed value is `'0·0'`.

_atom_site_label *(char)*

The `_atom_site_label` is a unique identifier for a particular site in the crystal. This code is made up of a sequence of up to seven components, `_atom_site_label_component_0` to `*_6`, which may be specified as separate data items. Component 0 usually matches one of the specified `_atom_type_symbol` codes. This is not mandatory if an `_atom_site_type_symbol` item is included in the atom site list. The `_atom_site_type_symbol` always takes precedence over an `_atom_site_label` in the identification of the atom type. The label components 1 to 6 are optional, and normally only components 0 and 1 are used. Note that components 0 and 1 are concatenated, while all other components, if specified, are separated by an underline character. Underline separators are only used if higher-order components exist. If an intermediate component is not used it may be omitted provided the underline separators are inserted. For example the label ‘C233__ggg’ is acceptable and represents the components C, 233, ‘’, and ggg. Each label may have a different number of components.

Appearance in list: yes.

Example(s): `C12`, `Ca3g28`, `Fe3+17`, `H*251`, `boron2a`, `C_a_phe 83_a_0`, `Zn_Zn_301_A_0`

_atom_site_label_component_0
_atom_site_label_component_1
_atom_site_label_component_2
_atom_site_label_component_3
_atom_site_label_component_4
_atom_site_label_component_5
_atom_site_label_component_6 *(char)*

Component 0 is normally a code which matches identically with one of the `_atom_type_symbol` codes. If this is the case then the rules governing the `_atom_type_symbol` code apply. If, however, the data item `_atom_site_type_symbol` is also specified in the atom site list, component 0 need not match this symbol or adhere to any of the `_atom_type_symbol` rules. Component 1 is referred to as the ‘atom number’. When component 0 is the atom type code, it is used to number the sites with the same atom type. This component code must start with at least one digit which is not followed by a + or − sign (to distinguish it from the component 0 rules). Components 2 to 6 contain the identifier, residue, sequence, chain order and alternate codes, respectively. These codes may be composed of any characters except an underline.

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list.

_atom_site_occupancy *(numb)*

The fraction of the atom type present at this site. The sum of the occupancies of all the atom types at this site may not significantly exceed 1·0 unless it is a dummy site.

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. E.s.d. expected: yes. Default e.s.d. value: 0·0. Where no value is given, the assumed value is `'1·0'`. The permitted range is 0·0→1·0.

_atom_site_refinement_flags *(char)*

A concatenated series of single-letter codes which indicate the refinement restraints or constraints applied to this site.

.	no refinement constraints
S	special position constraint on site
G	rigid group refinement of site
R	riding atom site attached to non-riding atom
D	distance or angle restraint on site
T	thermal displacement constraints
U	$U_{\rm iso}$ or U_{ij} restraint (rigid bond)
P	partial occupancy constraint

Appearance in list: yes. If looped, `_atom_site_label` must be present in the same list. Where no value is given, the assumed value is `'.'`.

_atom_site_restraints *(char)*

A description of restraints applied to specific parameters at this site during refinement. See also **_atom_site_refinement_flags** and **_refine_ls_number_restraints**.

Appearance in list: yes. If looped, **_atom_site_label** must be present in the same list. Where no value is given, the assumed value is '.'.

Example(s): `'restrained to planar ring'`

_atom_site_symmetry_multiplicity *(numb)*

The multiplicity of a site due to the space-group symmetry as is given in *International Tables for Crystallography*, Vol. A (1987).

Appearance in list: yes. If looped, **_atom_site_label** must be present in the same list. The permitted range is 1→192.

_atom_site_thermal_displace_type *(char)*

A standard code used to describe the type of atomic displacement parameters used for the site.

Uani	anisotropic U_{ij}
Uiso	isotropic U
Uovl	overall U
Umpe	multipole expansion U

Appearance in list: yes. If looped, **_atom_site_label** must be present in the same list.

_atom_site_type_symbol *(char)*

A code to identify the atom specie(s) occupying this site. This code must match a corresponding **_atom_type_symbol**. The specification of this code is optional if component 0 of the **_atom_site_label** is used for this purpose. See **_atom_type_symbol**.

Appearance in list: yes. If looped, **_atom_site_label** must be present in the same list.

Example(s): `Cu` , `Cu2+` , `dummy` , `Fe3+Ni2+` , `S-` , `H*` , `H(SDS)`

_atom_site_U_iso_or_equiv *(numb)*

Isotropic atomic displacement parameter, or equivalent isotropic atomic displacement parameter calculated from anisotropic atomic displacement parameters. The latter must be calculated as $U_{(equiv)} = (1/3)\sum_i[\sum_j(U_{ij}a_i^*a_j^*A_i.A_j)]$ where A are the real-cell and a^* the reciprocal-cell lengths [see Fischer, R. X. & Tillmanns, E. (1988). *Acta Cryst.* C**44**, 775–776].

Appearance in list: yes. If looped, **_atom_site_label** must be present in the same list. E.s.d. expected: yes. Default e.s.d. value: 0·0. The permitted range is 0·0→10·0. The units extensions are: ' ' (ångströms squared *1.0) '_pm' (picometres squared /10000.) '_nm' (nanometres squared *100.).

_atom_site_Wyckoff_symbol *(char)*

The Wyckoff symbol (letter) as listed in the space-group section of *International Tables for Crystallography*, Vol. A (1987).

Appearance in list: yes. If looped, **_atom_site_label** must be present in the same list.

_atom_sites_Cartn_tran_matrix_11
_atom_sites_Cartn_tran_matrix_12
_atom_sites_Cartn_tran_matrix_13
_atom_sites_Cartn_tran_matrix_21
_atom_sites_Cartn_tran_matrix_22
_atom_sites_Cartn_tran_matrix_23
_atom_sites_Cartn_tran_matrix_31
_atom_sites_Cartn_tran_matrix_32
_atom_sites_Cartn_tran_matrix_33 *(numb)*

Matrix elements used to transform fractional coordinates to orthogonal Cartesian coordinates. The axial alignments of this transformation are described in **_atom_sites_Cartn_transform_axes**.

$$\begin{pmatrix} 11 & 12 & 13 \\ 21 & 22 & 23 \\ 31 & 32 & 33 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}_{\text{fractional}} = \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix}_{\text{Cartesian}}$$

_atom_sites_Cartn_transform_axes *(char)*

A description of the relative alignment of the crystal cell axes to the Cartesian orthogonal axes as applied in the transformation matrix **_atom_sites_Cartn_tran_matrix_**.

Example(s): `'a parallel to x; b in the plane of y & z'`

_atom_sites_solution_primary
_atom_sites_solution_secondary
_atom_sites_solution_hydrogens *(char)*

Codes which identify the methods used to locate the initial atomic sites. The ***_primary** code identifies how the first atom sites were determined; the ***_secondary** code identifies how the remaining non-hydrogen sites were located; and the ***_hydrogens** code identifies how the hydrogens were located.

difmap	difference Fourier map
vecmap	real-space vector search
heavy	heavy-atom method
direct	structure-invariant direct methods
geom	inferred from neighbouring sites
disper	anomalous-dispersion techniques
isomor	isomorphous structure methods

_atom_type_analytical_mass_% *(numb)*

Mass percentage of this atom type derived from chemical analysis.

Appearance in list: yes. If looped, **_atom_type_symbol** must be present in the same list. The permitted range is 0·0→ ∞.

_atom_type_description *(char)*

A description of the atom(s) designated by this atom type. In most cases this will be the element name and oxidation state of a single atom species. For disordered or nonstoichiometric structures it will describe a combination of atom species.

Appearance in list: yes. If looped, **_atom_type_symbol** must be present in the same list.

Example(s): `deuterium`, `0·34Fe+0·66Ni`

_atom_type_number_in_cell *(numb)*

Total number of atoms of this atom type in the unit cell.

Appearance in list: yes. If looped, **_atom_type_symbol** must be present in the same list. The permitted range is 0→ ∞.

_atom_type_oxidation_number *(numb)*

Formal oxidation state of this atom type in the structure.

Appearance in list: yes. If looped, **_atom_type_symbol** must be present in the same list. Where no value is given, the assumed value is '0'. The permitted range is −6→6.

_atom_type_radius_bond
_atom_type_radius_contact *(numb)*

The effective intra- and intermolecular bonding radii of this atom type.

Appearance in list: yes. If looped, `_atom_type_symbol` must be present in the same list. The permitted range is 0·0→4·0. The units extensions are: ' ' (ångströms *1.0) '`_pm`' (picometres /100.) '`_nm`' (nanometres *10.).

_atom_type_scat_Cromer_Mann_a1
_atom_type_scat_Cromer_Mann_a2
_atom_type_scat_Cromer_Mann_a3
_atom_type_scat_Cromer_Mann_a4
_atom_type_scat_Cromer_Mann_b1
_atom_type_scat_Cromer_Mann_b2
_atom_type_scat_Cromer_Mann_b3
_atom_type_scat_Cromer_Mann_b4
_atom_type_scat_Cromer_Mann_c *(numb)*

The Cromer–Mann scattering-factor coefficients used to calculate the scattering factors for this atom type. May be entered in any order. See *International Tables for X-ray Crystallography*, Vol. IV, Table 2.2B (1974); or *International Tables for Crystallography*, Vol. C, Tables 6.1.1.4 and 6.1.1.5 (1991).

Appearance in list: yes. If looped, `_atom_type_symbol` must be present in the same list.

_atom_type_scat_dispersion_imag
_atom_type_scat_dispersion_real *(numb)*

The imaginary and real components of the anomalous dispersion scattering factors, f'' and f' (in electrons) for this atom type and the radiation given in `_diffrn_radiation_wavelength`.

Appearance in list: yes. If looped, `_atom_type_symbol` must be present in the same list. Where no value is given, the assumed value is '0·0'.

_atom_type_scat_source *(char)*

Reference to source of scattering factors used for this atom type.

Appearance in list: yes. If looped, `_atom_type_symbol` must be present in the same list.

Example(s): `'International Tables Vol. IV Table 2.4.6B'`

_atom_type_scat_versus_stol_list *(char)*

A table of scattering factors as a function of $(\sin\theta)/\lambda$. This table should be well commented to indicate the items present. Regularly formatted lists are strongly recommended

Appearance in list: yes. If looped, `_atom_type_symbol` must be present in the same list.

_atom_type_symbol *(char)*

The code used to identify the atom specie(s) representing this atom type. Normally this code is the element symbol. The code may be composed of any character except an underline with the additional proviso that digits designate an oxidation state and must be followed by a + or − character.

Appearance in list: yes.

Example(s): `C`, `Cu2+`, `H(SDS)`, `dummy`, `FeNi`

_audit_creation_date *(char)*

A date that the CIF was created. The date format is yy-mm-dd.

Example(s): `90-07-12`

_audit_creation_method *(char)*

A description of how data was entered into the CIF.

Example(s): `'spawned by the program QBEE'`

_audit_update_record *(char)*

A record of any changes to the CIF. The update format is a date (yy-mm-dd) followed by a description of the changes. The latest update entry is added to the bottom of this record.

Example(s): `'90-07-15 Updated by the Co-editor'`

_cell_angle_alpha
_cell_angle_beta
_cell_angle_gamma *(numb)*

Unit-cell angles in degrees of the reported structure. The values of `_refln_index_h`, `*_k`, `*_l` must correspond to the cell defined by these values and `_cell_length_a`, `*_b` and `*_c`. The values of `_diffrn_refln_index_h`, `*_k`, `*_l` may not correspond to these values if a cell transformation took place following the measurement of diffraction intensities. See also `_diffrn_reflns_transf_matrix_`.

E.s.d. expected: yes. Default e.s.d. value: 0·0. Where no value is given, the assumed value is '90·0'. The permitted range is 0·0→180·0.

_cell_formula_units_Z *(numb)*

The number of the formula units in the unit cell as specified by `_chemical_formula_structural`, `_chemical_formula_moiety` or `_chemical_formula_sum`.

The permitted range is 1→ ∞.

_cell_length_a
_cell_length_b
_cell_length_c *(numb)*

Unit-cell lengths corresponding to the structure reported. The values of `_refln_index_h`, `*_k`, `*_l` must correspond to the cell defined by these values and `_cell_angle_` values. The values of `_diffrn_refln_index_h`, `*_k`, `*_l` may not correspond to these values if a cell transformation took place following the measurement of diffraction intensities. See also `_diffrn_reflns_transf_matrix_`.

E.s.d. expected: yes. Default e.s.d. value: 0·0. The permitted range is 0·0→ ∞. The units extensions are: ' ' (ångströms *1.0) '`_pm`' (picometres /100.) '`_nm`' (nanometres *10.).

_cell_measurement_pressure *(numb)*

The pressure at which the unit-cell parameters were measured (not the pressure used to synthesize the sample).

E.s.d. expected: yes. Default e.s.d. value: 0·0. The units extensions are: ' ' (kilopascals *1.0) '`_GPa`' (gigapascals *1.0E+6).

_cell_measurement_radiation *(char)*

Description of the radiation used to measure the unit-cell data. See also `_cell_measurement_wavelength`.

Example(s): `neutron`, `'Cu K\a'`, `synchrotron`

_cell_measurement_refln_index_h
_cell_measurement_refln_index_k
_cell_measurement_refln_index_l *(numb)*

Miller indices of a reflection used for unit-cell measurements.

Appearance in list: yes.

_cell_measurement_refln_theta *(numb)*

Theta angle in degrees for the reflection used for unit-cell measurement with the indices _cell_measurement_refln_index_.

Appearance in list: yes. If looped, _cell_measurement_refln_index_ must be present in the same list. The permitted range is 0·0→90·0.

_cell_measurement_reflns_used *(numb)*

The total number of reflections used to determine the unit cell. These reflections may be specified as _cell_measurement_refln_ data items.

_cell_measurement_temperature *(numb)*

The temperature at which the unit-cell parameters were measured (not the temperature of synthesis).

E.s.d. expected: yes. Default e.s.d. value: 0·0. The permitted range is 0·0→ ∞. The units extensions are: ‘ ’ (Kelvin +0) ‘_C’ (Celsius +273.0).

_cell_measurement_theta_max
_cell_measurement_theta_min *(numb)*

The maximum and minimum theta angles in degrees of reflections used to measure the unit cell.

The permitted range is 0·0→90·0.

_cell_measurement_wavelength *(numb)*

The wavelength of the radiation used to measure the unit cell. If this is not specified, the wavelength is assumed to be the same as that given in _diffrn_radiation_wavelength.

The permitted range is 0·0→ ∞. The units extensions are: ‘ ’ (ångströms *1.0) ‘_pm’ (picometres /100.) ‘_nm’ (nanometres *10.).

_cell_special_details *(char)*

A description of special aspects of the cell choice, noting possible alternative settings.

Example(s): `pseudo-orthorhombic`, `'standard setting from 45 deg rotation around c'`

_cell_volume *(numb)*

Volume calculated from _cell_length_ and _cell_angle_ values.

E.s.d. expected: yes. Default e.s.d. value: 0·0. The permitted range is 0·0→ ∞. The units extensions are: ‘ ’ (cubic ångströms *1.0) ‘_pm’ (cubic picometres /1.0E+6) ‘_nm’ (cubic nanometres *1000.).

_chemical_compound_source *(char)*

Description of the source 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 product.

Example(s): `'From Norilsk (USSR)'`, `'Extracted from the bark of Cinchona Naturalis'`

_chemical_conn_atom_charge *(numb)*

The net integer charge assigned to this atom. This is the formal charge assignment normally found in chemical diagrams.

Appearance in list: yes. If looped, _chemical_conn_atom_type_symbol must be present in the same list. Where no value is given, the assumed value is ‘0’. The permitted range is −6→6.

Example(s): 1 (for an ammonium nitrogen), −1 (for a chloride ion)

_chemical_conn_atom_display_x
_chemical_conn_atom_display_y *(numb)*

The 2D Cartesian coordinates (x, y) of the position of this atom in a recognizable chemical diagram. The coordinate origin is at the lower left corner, the x axis is horizontal and the y axis is vertical. The coordinates must lie in the range 0·0 to 1·0. These coordinates can be obtained from projections of a suitable uncluttered view of the molecular structure. If absent, values will be assigned by the journals' or database staff.

Appearance in list: yes. If looped, _chemical_conn_atom_type_symbol must be present in the same list. The permitted range is 0·0→1·0.

_chemical_conn_atom_NCA *(numb)*

The number of connected atoms excluding terminal hydrogen atoms.

Appearance in list: yes. If looped, _chemical_conn_atom_type_symbol must be present in the same list. The permitted range is 0→ ∞.

_chemical_conn_atom_NH *(numb)*

The total number of hydrogen atoms attached to this atom, regardless of whether they are included in the refinement or the _atom_site_ list. This number will be the same as _atom_site_attached_hydrogens only if none of the hydrogen atoms appear in the _atom_site_ list.

Appearance in list: yes. If looped, _chemical_conn_atom_type_symbol must be present in the same list. The permitted range is 0→ ∞.

_chemical_conn_atom_number *(numb)*

The chemical sequence number to be associated with this atom.

Appearance in list: yes. If looped, _chemical_conn_atom_type_symbol must be present in the same list. The permitted range is 1→ ∞.

_chemical_conn_atom_type_symbol *(char)*

A code identifying the atom type. This code must match an _atom_type_symbol code in the _atom_type_ list; or be a recognizable element symbol.

Appearance in list: yes.

_chemical_conn_bond_atom_1
_chemical_conn_bond_atom_2 *(numb)*

Atom numbers which must match with chemical sequence numbers specified as _chemical_conn_atom_number values. These link the bond connection to the chemical numbering and atom sites.

Appearance in list: yes. The permitted range is 1→ ∞.

`_chemical_conn_bond_type` *(char)*

The chemical bond type associated with the connection between the two sites `_chemical_conn_bond_atom_1` and `*_2`.

`sing`	single bond
`doub`	double bond
`trip`	triple bond
`quad`	quadruple bond
`arom`	aromatic bond
`poly`	polymeric bond
`delo`	delocalized double bond
`pi`	pi bond

Appearance in list: yes. If looped, `_chemical_conn_bond_atom_1` must be present in the same list. Where no value is given, the assumed value is `'sing'`.

`_chemical_formula_appendix` *(These are notes only)*

`_chemical_formula_` items specify the composition and chemical properties of the compound. The formula data items must agree with those that specify the density, unit-cell and *Z* values.

The following rules apply to the construction of the data items `_chemical_formula_analytical`, `*_structural` and `*_sum`. For the data item `*_moiety` the formula construction is broken up into residues or moieties, *i.e.* groups of atoms that form a molecular unit or molecular ion. The rules given below apply within each moiety but different requirements apply to the way that moieties are connected (see `_chemical_formula_moiety`).

1. Only recognized element symbols may be used.
2. Each element symbol is followed by a 'count' number. A count of '1' may be omitted.
3. A space or parenthesis must separate each element symbol and its count.
4. Where a group of elements is enclosed in parentheses, the multiplier for the group must follow the closing parentheses. That is, all element and group multipliers are assumed to be printed as subscripted numbers. [An exception to this rule exists for `*_moiety` formulae where pre- and post-multipliers are permitted for molecular units].
5. Unless the elements are ordered in a manner that corresponds to their chemical structure, as in `_chemical_formula_structural`, the order of the elements within any group or moiety should be: C, H followed by the other elements in alphabetical order of their symbol. This is the 'Hill' system used by *Chemical Abstracts*. This ordering is used in `_chemical_formula_moiety` and `_chemical_formula_sum`.

`_chemical_formula_analytical` *(char)*

Formula determined by standard chemical analysis including trace elements. See `_chemical_formula_appendix` for rules for writing chemical formulae. Parentheses are used only for e.s.d.'s.

Example(s): `'Fe2.45(2) Ni1.60(3) S4'`

`_chemical_formula_moiety` *(char)*

Formula with each discrete bonded residue or ion shown as a separate moiety. See above `_chemical_formula_appendix` for rules for writing chemical formulae. In addition to the general formulae requirements, the following rules apply:

1. Moieties are separated by commas ', '.
2. The order of elements within a moiety follows general rule 5 in `_chemical_formula_appendix`.
3. Parentheses are not used within moieties but may surround a moiety. Parentheses may not be nested.
4. Charges should be placed at the end of the moiety. The charge '+' or '−' may be preceded by a numerical multiplier and should be separated from the last (element symbol + count) by a space. Pre- or post-multipliers may be used for individual moieties.

Example(s): `'C7 H4 Cl Hg N O3 S'`
`'C12 H17 N4 O S 1+, C6 H2 N3 O7 1-'`
`'C12 H16 N2 O6, 5(H2 O1)'`
`'(Cd 2+)3, (C6 N6 Cr 3-)2, 2(H2 O)'`

`_chemical_formula_structural` *(char)*

See above `_chemical_formula_appendix` for the rules for writing chemical formulae for inorganics, organometallics, metal complexes *etc.*, in which bonded groups are preserved as discrete entities within parentheses, with post-multipliers as required. The order of the elements should give as much information as possible about the chemical structure. Parentheses may be used and nested as required. This formula should correspond to the structure as actually reported, *i.e.* trace elements not included in atom type and atom site data should not be included in this formula (see also `_chemical_formula_analytical`).

Example(s): `'Ca ((Cl O3)2 O)2 (H2 O)6'`
`'(Pt (N H3)2 (C5 H7 N3 O)2) (Cl O4)2'`

`_chemical_formula_sum` *(char)*

See above `_chemical_formula_appendix` for the rules for writing chemical formulae in which all discrete bonded residues and ions are summed over the constituent elements, following the ordering given in general rule 5 in `_chemical_formula_appendix`. Parentheses are not normally used.

Example(s): `'C18 H19 N7 O8 S'`

`_chemical_formula_weight` *(numb)*

Formula mass in daltons. This mass should correspond to the formulae given under `_chemical_formula_structural`, `*_moiety` or `*_sum` and, together with the *Z* value and cell parameters, should yield the density given as `_exptl_crystal_density_diffrn`.

The permitted range is 1·0→ ∞.

`_chemical_formula_weight_meas` *(numb)*

Formula mass in daltons measured by a non-diffraction experiment.

The permitted range is 1·0→ ∞.

`_chemical_melting_point` *(numb)*

The melting point of the crystal.

The permitted range is 0·0→ ∞. The units extensions are: `' '` (Kelvin +0) `'_C'` (Celsius +273·0).

`_chemical_name_common` *(char)*

Trivial name by which compound is commonly known.

Example(s): `1-bromoestradiol`

`_chemical_name_mineral` *(char)*

Mineral name accepted by the International Mineralogical Association. Use only for natural minerals. See also `_chemical_compound_source`.

Example(s): `chalcopyrite`

_chemical_name_structure_type *(char)*

Commonly used structure-type name. Usually only applied to minerals or inorganic compounds.

Example(s): `perovskite, sphalerite, A15`

_chemical_name_systematic *(char)*

IUPAC or *Chemical Abstracts* full name of compound.

Example(s): `1-bromoestra-1,3,5(10)-triene-3,17\b-diol`

_computing_cell_refinement
_computing_data_collection
_computing_data_reduction
_computing_molecular_graphics
_computing_publication_material
_computing_structure_refinement
_computing_structure_solution *(char)*

Software used in the processing of this data. Give the program or package name and a brief reference.

Example(s): `'CAD4 (Enraf-Nonius)'`
`'DIFDAT, SORTRF, ADDREF (XTAL3.0, 1990)'`
`'FRODO (Jones, 1986) & ORTEP (Johnson, 1965)'`
`'CRYSTALS (Watkin, 1988)'`
`'SHELX85 (Sheldrick, 1985)'`

_database_code_CAS
_database_code_CSD
_database_code_ICSD
_database_code_MDF
_database_code_NBS
_database_code_PDF *(char)*

The codes are assigned by databases: *Chemical Abstracts*; Cambridge Structural (organic and metal-organic compounds); Inorganic Crystal Structure; Metals Data File (metal structures); NBS (NIST) Crystal Data Database (lattice parameters) and the Powder Diffraction File (JCPDS/ICDD).

_database_journal_ASTM
_database_journal_CSD *(char)*

The ASTM coden for a journal as given in the Chemical Source List and the journal code used in the Cambridge Structural Database.

_diffrn_ambient_pressure *(numb)*

The pressure at which the diffraction data were measured.

E.s.d. expected: yes. Default e.s.d. value: 0·0. The permitted range is 0·0→ ∞. The units extensions are: ‘ ’ (kilopascals *1.0) ‘_GPa’ (gigapascals *1.0E+6).

_diffrn_ambient_temperature *(numb)*

The mean temperature at which the diffraction data were measured.

E.s.d. expected: yes. Default e.s.d. value: 0·0. The permitted range is 0·0→ ∞. The units extensions are: ‘ ’ (Kelvin +0) ‘_C’ (Celsius +273·0).

_diffrn_attenuator_code *(char)*

A code associated with a particular attenuator setting. This code is referenced by the `_diffrn_refln_attenuator_code` which is stored with the diffraction data. See `_diffrn_attenuator_scale`.

Appearance in list: yes.

_diffrn_attenuator_scale *(numb)*

The intensity scale associated with a particular attenuator setting identified by `_diffrn_attenuator_code`.

Appearance in list: yes. If looped, `_diffrn_attenuator_code` must be present in the same list. The permitted range is 1·0→ ∞.

_diffrn_measurement_device *(char)*

Description of the diffractometer or camera used to measure the diffraction intensities.

Example(s): `'Gandolfi 114mm powder camera'`

_diffrn_measurement_method *(char)*

Method used to measure diffraction data.

Example(s): `'profile data from theta/2theta scans'`

_diffrn_orient_matrix_type *(char)*

A description of the orientation matrix type and how it should be applied to define the orientation of the crystal precisely with respect to the diffractometer axes.

_diffrn_orient_matrix_UB_11
_diffrn_orient_matrix_UB_12
_diffrn_orient_matrix_UB_13
_diffrn_orient_matrix_UB_21
_diffrn_orient_matrix_UB_22
_diffrn_orient_matrix_UB_23
_diffrn_orient_matrix_UB_31
_diffrn_orient_matrix_UB_32
_diffrn_orient_matrix_UB_33 *(numb)*

The elements of the diffractometer orientation matrix. These define the dimensions of the reciprocal cell and its orientation to the local diffractometer axes. See `_diffrn_orient_matrix_type`.

_diffrn_orient_refln_angle_chi
_diffrn_orient_refln_angle_kappa
_diffrn_orient_refln_angle_phi
_diffrn_orient_refln_angle_psi *(numb)*

Diffractometer angles in degrees of a reflection used to define the orientation matrix. See `_diffrn_orient_matrix_UB_` and `_diffrn_orient_refln_index_h`, `*_k` and `*_l`.

Appearance in list: yes. If looped, `_diffrn_orient_refln_index_` must be present in the same list.

_diffrn_orient_refln_index_h
_diffrn_orient_refln_index_k
_diffrn_orient_refln_index_l *(numb)*

The indices of a reflection used to define the orientation matrix. See `_diffrn_orient_matrix_type` and `_diffrn_orient_matrix_`

Appearance in list: yes.

_diffrn_radiation_detector *(char)*

The detector used to measure the diffraction intensities.

Appearance in list: both.

Example(s): `scintillation`, `LiI`, `'video tube'`, `'Kodak II film'`

_diffrn_radiation_detector_dtime *(numb)*

The deadtime in microseconds of **_diffrn_radiation_detector**.

Appearance in list: both. The permitted range is 0·0→ ∞.

_diffrn_radiation_filter_edge *(numb)*

Absorption edge of the radiation filter used.

Appearance in list: both. The permitted range is 0·0→ ∞. The units extensions are: ' ' (ångströms *1.0) '_pm' (picometres /100.) '_nm' (nanometres *10.).

_diffrn_radiation_inhomogeneity *(numb)*

Half-width in millimetres of the incident beam in the perpendicular direction with respect to the diffraction plane.

Appearance in list: both. The permitted range is 0·0→ ∞.

_diffrn_radiation_monochromator *(char)*

The method used to obtain monochromatic radiation. If a monochromator crystal is used the material and the indices of the Bragg reflection are specified.

Appearance in list: both.

Example(s): `'Zr filter'`, `'Ge 220'`, `none`, `'equatorial mounted graphite'`

_diffrn_radiation_polarisn_norm *(numb)*

The angle in degrees of the perpendicular polarisation component to the diffraction plane. See **_diffrn_radiation_polarisn_ratio**.

Appearance in list: both. The permitted range is 0·0→ ∞.

_diffrn_radiation_polarisn_ratio *(numb)*

Polarisation ratio of the diffraction beam incident on the crystal. It is the ratio of the perpendicularly polarised to the parallel polarised component of the radiation. The perpendicular component forms an angle of **_diffrn_radiation_polarisn_norm** to the normal to the diffraction plane of the sample (*i.e.* the plane containing the incident and reflected beams).

Appearance in list: both. The permitted range is 0·0→ ∞.

_diffrn_radiation_source *(char)*

The source of radiation.

Appearance in list: both.

Example(s): `'RU2 Rigaku Denki rotating Cu anode'`, `'fine focus Philips Mo tube'`, `'5MeV synchrotron'`, `'HIFAR reactor'`

_diffrn_radiation_type *(char)*

The nature of the radiation.

Appearance in list: both.

Example(s): `'Cu K\a'`, `neutron`, `electron`

_diffrn_radiation_wavelength *(numb)*

The radiation wavelength.

Appearance in list: both. The permitted range is 0·0→ ∞. The units extensions are: ' ' (ångströms *1.0) '_pm' (picometres /100.) '_nm' (nanometres *10.).

_diffrn_radiation_wavelength_id *(char)*

The code identifying each value of **_diffrn_radiation_wavelength**. The **_diffrn_radiation_** data is looped when multiple wavelengths are used. This code is used to link with the **_diffrn_refln_** list. It must match with one of the **_diffrn_refln_wavelength_id** codes.

Appearance in list: yes. If looped, **_diffrn_radiation_wavelength** must be present in the same list.

Example(s): `x1`, `x2`, `neut`

_diffrn_radiation_wavelength_wt *(numb)*

The relative weight of a wavelength identified by the code **_diffrn_radiation_wavelength_id** in the list of wavelengths.

Appearance in list: yes. If looped, **_diffrn_radiation_wavelength_id** must be present in the same list. The permitted range is 0·0→1·0. Where no value is given, the assumed value is '1·0'.

_diffrn_refln_angle_chi
_diffrn_refln_angle_kappa
_diffrn_refln_angle_omega
_diffrn_refln_angle_phi
_diffrn_refln_angle_psi
_diffrn_refln_angle_theta *(numb)*

The diffractometer angles in degrees of a reflection. These correspond to the specified orientation matrix and the original measured cell before any subsequent cell transformations.

Appearance in list: yes. If looped, **_diffrn_refln_index_** must be present in the same list.

_diffrn_refln_attenuator_code *(char)*

The code identifying the attenuator setting for this reflection. This code must match one of the **_diffrn_attenuator_code** values.

Appearance in list: yes. If looped, **_diffrn_refln_index_** must be present in the same list.

_diffrn_refln_counts_bg_1
_diffrn_refln_counts_bg_2
_diffrn_refln_counts_net
_diffrn_refln_counts_peak
_diffrn_refln_counts_total *(numb)*

The diffractometer counts for the measurements: background before the peak, background after the peak, net counts after background removed, counts for peak scan or position, and the total counts (background plus peak).

Appearance in list: yes. If looped, **_diffrn_refln_index_** must be present in the same list. The permitted range is 0→ ∞.

_diffrn_refln_crystal_id *(char)*

Code identifying each crystal if multiple crystals are used. Is used to link with _exptl_crystal_id in the _exptl_crystal_ list.

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list.

_diffrn_refln_detect_slit_horiz
_diffrn_refln_detect_slit_vert *(numb)*

Total horizontal and vertical slit apertures in degrees.

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list. The permitted range is 0·0→90·0.

_diffrn_refln_elapsed_time *(numb)*

Elapsed time from the start of diffraction measurement to the measurement of this intensity.

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list. The permitted range is 0·0→ ∞. The units extensions are: ' ' (minutes *1.0) '_sec' (seconds /60.) '_hr' (hours *60.).

_diffrn_refln_index_h
_diffrn_refln_index_k
_diffrn_refln_index_l *(numb)*

Miller indices of a diffraction reflection. These need not match the _refln_index_h, *_k, *_l values if a transformation of the original measured cell has taken place. Details of the cell transformation are described in _diffrn_reflns_reduction_process. See also _diffrn_reflns_transf_matrix_.

Appearance in list: yes.

_diffrn_refln_intensity_net
_diffrn_refln_intensity_sigma *(numb)*

Net intensity and e.s.d. calculated from the diffraction counts after the attenuator and standard scales have been applied.

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list. The permitted range is 0→ ∞.

_diffrn_refln_scale_group_code *(char)*

The code identifying the scale appling to this reflection. This code must match with a specified _diffrn_scale_group_code value.

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list.

_diffrn_refln_scan_mode *(char)*

The code identifying the mode of scanning with a diffractometer. See _diffrn_refln_scan_width and _diffrn_refln_scan_mode_backgd.

om	ω scan
ot	$\omega/2\theta$ scan

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list.

_diffrn_refln_scan_mode_backgd *(char)*

The code identifying the mode of scanning a reflection to measure the background intensity.

st	stationary counter background
mo	moving counter background

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list.

_diffrn_refln_scan_width *(numb)*

The scan width in degrees of the scan mode defined by the code _diffrn_refln_scan_mode.

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list. The permitted range is 0·0→90·0.

_diffrn_refln_sint/lambda *(numb)*

The sin θ over wavelength value for this reflection.

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list. The permitted range is 0·0→ ∞. The units extensions are: ' ' (reciprocal ångströms *1.0) '_pm' (reciprocal picometres *100.) '_nm' (reciprocal nanometres /10.).

_diffrn_refln_standard_code *(char)*

The code identifying that this reflection was measured as a standard intensity. This is the case if the code matched one of the _diffrn_standard_refln_code values.

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list.

Example(s): `1, 2, 3, s1, s2, s3, A, B, C`

_diffrn_refln_wavelength *(numb)*

The mean wavelength of radiation used to measure diffraction for this reflection. This is an important parameter for data collected using energy dispersive detectors or the Laue method.

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list. The permitted range is 0·0→ ∞. The units extensions are: ' ' (ångströms *1.0) '_pm' (picometres /100.) '_nm' (nanometres *10.).

_diffrn_refln_wavelength_id *(char)*

Code identifying the wavelength in the _diffrn_radiation_ list.

Appearance in list: yes. If looped, _diffrn_refln_index_ must be present in the same list.

Example(s): `x1, x2, neut`

_diffrn_reflns_av_R_equivalents *(numb)*

The residual $[\sum|av\Delta(I)|/\sum|av(I)|]$ for symmetry-equivalent reflections used to calculate the average intensity $av(I)$. The $av\Delta(I)$ term is the average difference between $av(I)$ and the individual intensities.

The permitted range is 0·0→ ∞.

_diffrn_reflns_av_sigmaI/netI *(numb)*

Measure $[\sum|\sigma(I)|/\sum|net(I)|]$ for all measured reflections.

The permitted range is 0·0→ ∞.

_diffrn_reflns_limit_h_max
_diffrn_reflns_limit_h_min
_diffrn_reflns_limit_k_max
_diffrn_reflns_limit_k_min
_diffrn_reflns_limit_l_max
_diffrn_reflns_limit_l_min *(numb)*

The index limits of the diffraction reflection data specified by _diffrn_refln_index_h, *_k, *_l.

_diffrn_reflns_number *(numb)*

The total number of measured diffraction data.

The permitted range is 0→ ∞.

_diffrn_reflns_reduction_process *(char)*

A description of the process used to reduce the intensity data into structure-factor magnitudes.

Example(s): `'data averaged using Fisher test'`

_diffrn_reflns_theta_max
_diffrn_reflns_theta_min *(numb)*

Theta angle limits in degrees for the measured diffraction data.

The permitted range is 0·0→90·0.

_diffrn_reflns_transf_matrix_11
_diffrn_reflns_transf_matrix_12
_diffrn_reflns_transf_matrix_13
_diffrn_reflns_transf_matrix_21
_diffrn_reflns_transf_matrix_22
_diffrn_reflns_transf_matrix_23
_diffrn_reflns_transf_matrix_31
_diffrn_reflns_transf_matrix_32
_diffrn_reflns_transf_matrix_33 *(numb)*

Elements of the matrix used to transform the diffraction reflection indices `_diffrn_refln_index_h`, `*_k`, `*_l` into the `_refln_index_h`, `*_k`, `*_l` indices.

$$(h \quad k \quad l)_{\rm diffraction}\begin{pmatrix}11 & 12 & 13\\ 21 & 22 & 23\\ 31 & 32 & 33\end{pmatrix} = (h' \quad k' \quad l')$$

_diffrn_scale_group_code *(char)*

The code identifying a specific measurement group (*e.g.* for multi-film or multi-crystal data). The code must match a `_diffrn_refln_scale_group_code` in the reflection list.

Appearance in list: yes.

Example(s): `1, 2, 3 , s1, A, B, c1, c2, c3`

_diffrn_scale_group_I_net *(numb)*

The intensity scale for a specific measurement group identified by `_diffrn_scale_group_code`.

Appearance in list: yes. The permitted range is 0·0→ ∞.

_diffrn_special_details *(char)*

Special details of the diffraction measurement process. Should include information about source instability, crystal motion, degradation and so on.

_diffrn_standard_refln_code *(char)*

The code identifying a reflection measured as a standard reflection with the indices `_diffrn_standard_refln_index_`. This is the same code as the `_diffrn_refln_standard_code` in the `_diffrn_refln_` list.

Appearance in list: yes. If looped, `_diffrn_standard_refln_index_` must be present in the same list.

Example(s): `1, 2, 3, s1, A, B`

_diffrn_standard_refln_index_h
_diffrn_standard_refln_index_k
_diffrn_standard_refln_index_l *(numb)*

Miller indices of standard reflections used in the diffraction measurement process.

Appearance in list: yes.

_diffrn_standards_decay_% *(numb)*

The percentage variation of the mean intensity for all standard reflections.

The permitted range is 0·0→ ∞.

_diffrn_standards_interval_count
_diffrn_standards_interval_time *(numb)*

The number of reflection intensities, or the time in minutes, between the measurement of standard reflection intensities.

The permitted range is 0→ ∞.

_diffrn_standards_number *(numb)*

The number of unique standard reflections used in the diffraction measurements.

The permitted range is 0→ ∞.

_diffrn_standards_scale_sigma *(numb)*

The e.s.d. of the individual mean standard scales applied to the intensity data.

The permitted range is 0·0→ ∞.

_exptl_absorpt_coefficient_mu *(numb)*

The absorption coefficient μ calculated from atomic content of the cell, the density and the radiation wavelength.

The permitted range is 0·0→ ∞. The units extensions are: `' '` (reciprocal millimetres *1.0) `'_cm'` (reciprocal centimetres /10.).

_exptl_absorpt_correction_T_max
_exptl_absorpt_correction_T_min *(numb)*

The maximum and minimum transmission factors for the crystal and radiation. These factors are also referred to as the absorption correction A or $1/A^*$.

The permitted range is 0·0→1·0.

_exptl_absorpt_correction_type *(char)*

The absorption correction type and method.

analytical	analytical from crystal shape
integration	integration from crystal shape
empirical	empirical from diffraction data
refdelf	refined from ΔF
sphere	spherical
cylinder	cylindrical
none	no absorption correction applied

_exptl_absorpt_process_details *(char)*

Description of the absorption process applied to the data.

Example(s): `'Tompa analytical'`

_exptl_crystal_colour *(char)*

The colour of the crystal.

Example(s): `'Dark green'`

_exptl_crystal_density_diffrn *(numb)*

Density values calculated from crystal cell and contents. The units are megagrams per cubic metre (grams per cubic centimetre).

The permitted range is 0·0→ ∞.

_exptl_crystal_density_meas *(numb)*

Density values measured using standard chemical and physical methods. The units are megagrams per cubic metre (grams per cubic centimetre).

The permitted range is 0·0→ ∞.

_exptl_crystal_density_meas_temp *(numb)*

Temperature in Kelvin that _exptl_crystal_density_meas was determined at.

The permitted range is 0·0→ ∞.

_exptl_crystal_density_method *(char)*

The method used to measure _exptl_crystal_density_meas.

_exptl_crystal_description *(char)*

A description of the crystal quality and habit. Dimensional data is better placed in the _exptl_crystal_face_ data items.

_exptl_crystal_F_000 *(numb)*

The number of electrons in the crystal unit cell $F(000)$.

The permitted range is 1→ ∞.

_exptl_crystal_face_diffr_chi
_exptl_crystal_face_diffr_kappa
_exptl_crystal_face_diffr_phi
_exptl_crystal_face_diffr_psi *(numb)*

The diffractometer angle settings in degrees for a specific crystal face associated with _exptl_crystal_face_perp_dist.

Appearance in list: yes. If looped, _exptl_crystal_face_perp_dist must be present in the same list.

_exptl_crystal_face_index_h
_exptl_crystal_face_index_k
_exptl_crystal_face_index_l *(numb)*

Miller indices of the crystal face associated with the value _exptl_crystal_face_perp_dist.

Appearance in list: yes. If looped, _exptl_crystal_face_perp_dist must be present in the same list.

_exptl_crystal_face_perp_dist *(numb)*

The perpendicular distance of the face to centre of rotation of the crystal.

Appearance in list: yes. The permitted range is 0·0→ ∞. The units extensions are: ' ' (millimetres *1.0) '_cm' (centimetres *10.0).

_exptl_crystal_id *(char)*

Code identifying each crystal if multiple crystals are used. Is used to link with _diffrn_refln_crystal_id in diffraction data and with _refln_crystal_id in the _refln_ list.

Appearance in list: yes.

_exptl_crystal_preparation *(char)*

Details of crystal growth and preparation of the crystal (*e.g.* mounting) prior to the diffraction measurements.

Example(s): 'mounted in an argon-filled quartz capillary'

_exptl_crystal_size_max
_exptl_crystal_size_mid
_exptl_crystal_size_min
_exptl_crystal_size_rad *(numb)*

The maximum, medial and minimum dimensions of the crystal. If the crystal is a sphere or a cylinder then the *_rad item is the radius. These may appear in a list with _exptl_crystal_id if multiple crystals used in the experiment.

Appearance in list: both. If looped, _exptl_crystal_id must be present in the same list. The permitted range is 0·0→ ∞. The units extensions are: ' ' (millimetres *1.0) '_cm' (centimetres *10.0).

_exptl_crystals_number *(numb)*

The total number of crystals used in the data measurement.

The permitted range is 1→ ∞.

_exptl_special_details *(char)*

Any special information about the experimental work prior to the diffraction measurement. See also _exptl_crystal_preparation.

_geom_angle *(numb)*

Angle in degrees bounded by the _geom_angle_atom_site_label_1, *_2 and *_3. Site at *_2 is at the apex of the angle.

Appearance in list: yes. If looped, _geom_angle_atom_site_label_ must be present in the same list. E.s.d. expected: yes. Default e.s.d. value: 0·0.

_geom_angle_atom_site_label_1
_geom_angle_atom_site_label_2
_geom_angle_atom_site_label_3 *(char)*

The labels of the three atom sites which define the angle specified by _geom_angle. These must match labels specified as _atom_site_label in the atom list. Label 2 identifies the site at the apex of the angle.

Appearance in list: yes.

_geom_angle_publ_flag *(char)*

This code signals if the angle is referred to in a publication or should be placed in a table of significant angles.

no	do not include angle in special list
yes	do include angle in special list

Appearance in list: yes. If looped, _geom_angle_atom_site_label_ must be present in the same list.

_geom_angle_site_symmetry_1
_geom_angle_site_symmetry_2
_geom_angle_site_symmetry_3 *(char)*

The symmetry code of each atom site as the symmetry equivalent position number 'n' and the cell translation number 'mmm'. These numbers are combined to form the code 'n mmm' or n_mmm. 'n' is the sequence number of the symmetry elements as listed in _symmetry_equiv_pos_as_xyz. 'mmm' are the concatenated cell translations along x, y, z with respect to the base number 555. The symmetry transformation is applied to the coordinates given by _atom_site_fract_x, *_y, *_z identified by _atom_site_label. If there are no cell translations the translation number may be omitted. If no symmetry operations or translations are applicable then a single period '.' may be used.

Appearance in list: yes. If looped, **_geom_angle_atom_site_label_** must be present in the same list.

Example(s): . (no symmetry or translation to site), 4 (4th symmetry operation applied), 7_645 (7th symm. posn; +a on x; $-b$ on y)

_geom_bond_atom_site_label_1
_geom_bond_atom_site_label_2 *(char)*

The labels of two atom sites that form a bond. These must match labels specified as **_atom_site_label** in the atom list.

Appearance in list: yes.

_geom_bond_distance *(numb)*

The intramolecular bond distance.

Appearance in list: yes. If looped, **_geom_bond_atom_site_label_** must be present in the same list. E.s.d. expected: yes. Default e.s.d. value: 0·0. The permitted range is 0·0→ ∞. The units extensions are: ' ' (ångströms *1.0) The units extensions are: '_pm' (picometres /100.) '_nm' (nanometres *10.).

_geom_bond_publ_flag *(char)*

Signals if the bond distance is referred to in a publication or should be placed in a list of special bond distances.

no	do not include bond in special list
yes	do include bond in special list

Appearance in list: yes. If looped, **_geom_bond_atom_site_label_** must be present in the same list.

_geom_bond_site_symmetry_1
_geom_bond_site_symmetry_2 *(char)*

The symmetry code of each atom site as the symmetry-equivalent position number 'n' and the cell translation number 'mmm'. These numbers are combined to form the code 'n mmm' or n_mmm. 'n' is the sequence number of the symmetry elements as listed in **_symmetry_equiv_pos_as_xyz**. 'mmm' are the concatenated cell translations along x, y, z with respect to the base number 555. The symmetry transformation is applied to the coordinates given by **_atom_site_fract_x**, ***_y**, ***_z** identified by **_atom_site_label**. If there are no cell translations the translation number may be omitted. If no symmetry operations or translations are applicable then a single period '.' may be used.

Appearance in list: yes. If looped, **_geom_bond_atom_site_label_** must be present in the same list.

Example(s): (no symmetry or translation to site), 4 (4th symmetry operation applied), 7_645 (7th symm. posn; +a on x; $-b$ on y)

_geom_contact_atom_site_label_1
_geom_contact_atom_site_label_2 *(char)*

The labels of two atom sites that are within contact distance. The labels must match **_atom_site_label** codes in the atom list.

Appearance in list: yes.

_geom_contact_distance *(numb)*

The interatomic contact distance.

Appearance in list: yes. If looped, **_geom_contact_atom_site_label_** must be present in the same list. E.s.d. expected: yes. Default e.s.d. value: 0·0. The permitted range is 0·0→ ∞. The units extensions are: ' ' (ångströms *1.0) The units extensions are: '_pm' (picometres /100.) '_nm' (nanometres *10.).

_geom_contact_publ_flag *(char)*

Signals if the contact distance is referred to in a publication or should be placed in a list of special contact distances.

no	do not include distance in special list
yes	do include distance in special list

Appearance in list: yes. If looped, **_geom_contact_atom_site_label_** must be present in the same list.

_geom_contact_site_symmetry_1
_geom_contact_site_symmetry_2 *(char)*

The symmetry code of each atom site as the symmetry equivalent position number 'n' and the cell translation number 'mmm'. These numbers are combined to form the code 'n mmm' or n_mmm. 'n' is the sequence number of the symmetry elements as listed in **_symmetry_equiv_pos_as_xyz**. 'mmm' are the concatenated cell translations along x, y, z with respect to the base number 555. The symmetry transformation is applied to the coordinates given by **_atom_site_fract_x**, ***_y**, ***_z** identified by **_atom_site_label**. If there are no cell translations the translation number may be omitted. If no symmetry operations or translations are applicable then a single period '.' may be used.

Appearance in list: yes. If looped, **_geom_contact_atom_site_label_** must be present in the same list.

Example(s): . (no symmetry or translation to site), 4 (4th symmetry operation applied), 7_645 (7th symm. posn; +a on x; $-b$ on y)

_geom_special_details *(char)*

The description of geometrical information not covered by the existing **_geom_** data names, such as least-squares planes.

_geom_torsion *(numb)*

The torsion angle in degrees bounded by the four atom sites identified by the **_geom_torsion_atom_site_label_** codes. These must match labels specified as **_atom_site_label** in the atom list. The torsion angle definition should be that of Klyne, W. & Prelog, V. (1960). *Endeavour*, **16**, 521–528.

Appearance in list: yes. If looped, **_geom_torsion_atom_site_label_** must be present in the same list. E.s.d. expected: yes. Default e.s.d. value: 0·0.

_geom_torsion_atom_site_label_1
_geom_torsion_atom_site_label_2
_geom_torsion_atom_site_label_3
_geom_torsion_atom_site_label_4 *(char)*

The labels of the four atom sites which define the torsion angle specified by **_geom_torsion**. These must match codes specified as **_atom_site_label** in the atom list. The torsion angle definition should be that of Klyne, W. & Prelog, V. (1960). *Endeavour*, **16**, 521–528. The vector direction ***_label_2** to ***_label_3** is the viewing direction, and the torsion angle is the angle of twist required to superimpose the projection of the vector site 2–site 1 onto the projection of the vector site 3–site 4. Clockwise torsions are positive, anticlockwise torsions are negative.

Appearance in list: yes.

_geom_torsion_publ_flag *(char)*

This code signals if the angle is referred to in a publication or should be placed in a table of significant angles.

no	do not include distance in special list
yes	do include distance in special list

Appearance in list: yes. If looped, **_geom_torsion_atom_site_label_** must be present in the same list.

_geom_torsion_site_symmetry_1
_geom_torsion_site_symmetry_2
_geom_torsion_site_symmetry_3
_geom_torsion_site_symmetry_4 *(char)*

The symmetry code of each atom site as the symmetry-equivalent position number 'n' and the cell translation number 'mmm'.

These numbers are combined to form the code 'n mmm' or n_mmm. 'n' is the sequence number of the symmetry elements as listed in `_symmetry_equiv_pos_as_xyz`. 'mmm' are the concatenated cell translations along x, y, z with respect to the base number 555. The symmetry transformation is applied to the coordinates given by `_atom_site_fract_x`, `*_y`, `*_z` identified by `_atom_site_label`. If there are no cell translations the translation number may be omitted. If no symmetry operations or translations are applicable then a single period '.' may be used.

Appearance in list: yes. If looped, `_geom_torsion_atom_site_label_` must be present in the same list.

Example(s): `.` (no symmetry or translation to site), `4` (4th symmetry operation applied), `7_645` (7th symm. posn; $+a$ on x; $-b$ on y)

_journal_coden_ASTM
_journal_coden_Cambridge
_journal_coeditor_address
_journal_coeditor_code
_journal_coeditor_email
_journal_coeditor_fax
_journal_coeditor_name
_journal_coeditor_notes
_journal_coeditor_phone
_journal_date_accepted
_journal_date_from_coeditor
_journal_date_to_coeditor
_journal_date_printers_final
_journal_date_printers_first
_journal_date_proofs_in
_journal_date_proofs_out
_journal_date_recd_copyright
_journal_date_recd_electronic
_journal_date_recd_hard_copy
_journal_issue
_journal_name_full
_journal_page_first
_journal_page_last
_journal_suppl_publ_number
_journal_suppl_publ_pages
_journal_techeditor_address
_journal_techeditor_code
_journal_techeditor_email
_journal_techeditor_fax
_journal_techeditor_name
_journal_techeditor_notes
_journal_techeditor_phone
_journal_volume
_journal_year *(char)*

Data items specified by the journal staff.

_publ_author_address *(char)*

The address of a publication author. If there is more than one author this will be looped with `_publ_author_name`.

Appearance in list: both.

Example(s):

```
; Department
  Institute
  Street
  City and postcode
  COUNTRY
;
```

_publ_author_name *(char)*

The name of a publication author. If there are multiple authors they will be looped with `_publ_author_address`. The family name(s) followed by a comma, precedes the first names or initials.

Appearance in list: both.

Example(s): `'Bleary, Percival R.'`
`"O'Neil, F.K."`
`'Van den Bossche, G.'`
`'Yang, D.-L.'`
`'Simonov, Yu.A'`

_publ_contact_author *(char)*

The name and address of the author submitting the manuscript and CIF. This is the person contacted by the journal editorial staff.

Example(s):

```
; Professor Dr. J.U. Blogs
  Department of Structural Chemistry
  RRDD Institute of Technology
  Building #6-M57
  Highho Street
  Citytown 64664
  COUNTRYHERE
;
```

_publ_contact_author_email *(char)*

Email address in a form recognizable to international networks.

Example(s): `name@host.domain.country`, `uur5@banjo.bitnet`

_publ_contact_author_fax *(char)*

Facsimile telephone number with international code in parentheses.

Example(s): `'(12) 34 947 7334'`

_publ_contact_author_phone *(char)*

Telephone number with international code in parentheses and any extension number preceded by 'ext'.

Example(s): `'(12) 34 947 7330 ext 5543'`

_publ_contact_letter *(char)*

A letter submitted to the journal editor by the contact author.

_publ_manuscript_creation *(char)*

A description of the wordprocessor package and computer used to create the word processed manuscript stored as `_publ_manuscript_processed`.

Example(s): `'Tex file created by FrameMaker on a Sun 3/280'`

_publ_manuscript_incl_extra_item
_publ_manuscript_incl_extra_info
_publ_manuscript_incl_extra_defn *(char)*

These data items are used to specify the need to include specific data into a manuscript which is not normally requested by the journal. `*_item` specifies the data name; `*_info` provides the reasons for the inclusion; and `*_defn` flags whether this is a standard definition or not (flags are 'yes' or 'no'). The example shows how these three items are looped. Note that `*_item` names *must* be enclosed in single quotes.

Appearance in list: yes.

Example(s):

```
'_atom_site_symmetry_multiplicity'   'special sites' yes
'_chemical_compound_source'          'rare material' yes
'_reflns_d_resolution_high' 'limited data a problem' yes
'_crystal_magnetic_permeability'      'new quantity'  no
```

_publl_manuscript_processed *(char)*

The full manuscript of a paper (excluding possibly the figures and the tables) output in ASCII characters from a word processor. Information about the generation of this data item must be specified in the data item **_publ_manuscript_creation**.

_publ_manuscript_text *(char)*

The full manuscript of a paper (excluding figures and possibly the tables) output as standard ASCII text.

_publ_requested_coeditor_name *(char)*

The Co-editor's name requested to process the submitted manuscript.

_publ_requested_journal *(char)*

The journal's name requested for publication.

_publ_section_title
_publ_section_abstract
_publ_section_comment
_publ_section_introduction
_publ_section_experimental
_publ_section_discussion
_publ_section_acknowledgements
_publ_section_references
_publ_section_figure_captions
_publ_section_table_legends *(char)*

The sections of a manuscript if submitted in parts. As an alternative see **_publ_manuscript_text** and **_publ_manuscript_processed**.

_refine_diff_density_max
_refine_diff_density_min *(numb)*

The largest and smallest density in the final difference Fourier map.

E.s.d. expected: yes. Default e.s.d. value: 0·0. The units extensions are: ' ' (electrons per cubic ångström *1.0) '**_pm**' (electrons per cubic picometre *1.0E+6) '**_nm**' (electrons per cubic nanometre /1000.).

_refine_ls_abs_structure_details *(char)*

The nature of the absolute structure and how it was determined. For example, to describe the nature of the Friedel data used.

_refine_ls_abs_structure_Flack *(numb)*

This measure of absolute structure (enantiomorph or polarity) is defined in the paper by Flack, H. D. (1983). *Acta Cryst.* **A39**, 876–881.

E.s.d. expected: yes. Default e.s.d. value: 0·0. The permitted range is 0·0→1·0.

_refine_ls_abs_structure_Rogers *(numb)*

This measure of absolute structure (enantiomorph or polarity) is defined in the paper by Rogers, D. (1981). *Acta Cryst.* **A37**, 734–741.

E.s.d. expected: yes. Default e.s.d. value: 0·0.

_refine_ls_extinction_coef *(numb)*

The extinction coefficient used to calculate the correction factor applied to the structure-factor data. The nature of the extinction coefficient is given in the definitions of **_refine_ls_extinction_expression** and **_refine_ls_extinction_method**. For the 'Zachariasen' method it will be the r^* value; for the 'B–C type 1 isotropic' method it is the 'g' value, and for 'B–C type 2 isotropic' corrections it is the 'ρ' value. Note that the magnitude of these values is usually of the order of 10000.

E.s.d. expected: yes. Default e.s.d. value: 0·0.

Example(s): `3472(52)` (Zachariasen coefficient $r^* = 0{\cdot}347(5) \times 10^4$)

_refine_ls_extinction_expression *(char)*

A description or reference of the extinction correction equation used to apply the data item **_refine_ls_extinction_coef**. This information must be sufficient to reproduce the extinction correction factors applied to the structure factors.

Example(s): `'Equ (22) p292 "Crystallographic Computing" Munksgaard 1970'`

_refine_ls_extinction_method *(char)*

A description of the extinction correction method applied with the data item **_refine_ls_extinction_coef**. This description should include information about the correction method 'Becker–Coppens' [Becker, P. J. & Coppens, P. (1974). *Acta Cryst.* **A30**, 129–153] or 'Zachariasen' [Zachariasen, W.H. (1967). *Acta Cryst.* **23**, 558–564]. The latter is sometimes referred to as the 'Larson' method [Larson, A. C. (1967). *Acta Cryst.* **23**, 664–665] even though it employs Zachariasen's formula. The Becker–Coppens procedure is referred to as 'type 1' when correcting secondary extinction dominated by the mosaic spread; as 'type 2' when secondary extinction is dominated by particle size and includes a primary extinction component; and as 'mixed' when there is a mixture of types 1 and 2. For the B–C method it is also necessary to set the mosaic distribution as either 'Gaussian' or 'Lorentzian'; and the nature of the extinction as 'isotropic' or 'anisotropic'. Note that if either the 'mixed' or 'anisotropic' corrections are applied the multiple coefficients cannot be contained in ***_extinction_coef** and must be listed in **_refine_special_details**.

Where no value is given, the assumed value is '**Zachariasen**'.

Example(s): `'B-C type 2 Gaussian isotropic'`

_refine_ls_goodness_of_fit_all
_refine_ls_goodness_of_fit_obs *(numb)*

The least-squares goodness-of-fit parameter S for all data, and for observed data, after the final cycle of refinement. Ideally account should be taken of parameters restrained in the least squares. The goodness-of-fit parameter S is defined as $S = [\sum(w|Y_m - Y_c|^2)/(N_{ref} - N_{param})]^{1/2}$ where the sum is over the specified reflection data; N_{ref} is the number of reflections used in the refinement; N_{param} is the number of refined parameters; Y_m and Y_c are the measured and calculated coefficients specified in `_refine_ls_structure_factor_coef`; and w is the least-squares weight [1/(e.s.d. squared)]. See also `_refine_ls_restrained_S_` definitions.

E.s.d. expected: yes. Default e.s.d. value: 0·0. The permitted range is 0·0→ ∞.

`_refine_ls_hydrogen_treatment` *(char)*

Treatment of hydrogen atoms in the least-squares refinement.

`refall`	refined all H parameters
`refxyz`	refined H coordinates only
`refU`	refined H U only
`noref`	no refinement of H parameters

Where no value is given, the assumed value is '**`refxyz`**'.

`_refine_ls_matrix_type` *(char)*

Type of matrix used to accumulate the least-squares derivatives.

`full`	full
`fullcycle`	full with fixed elements per cycle
`atomblock`	block diagonal per atom
`userblock`	user-defined blocks
`diagonal`	diagonal elements only
`sparse`	selected elements only

Where no value is given, the assumed value is '**`full`**'.

`_refine_ls_number_constraints` *(numb)*

The number of constrained (non-refined or dependent) parameters in the least-squares process. These may be due to symmetry or any other constraint process (*e.g.* rigid-body refinement). See also `_atom_site_constraints` and `_atom_site_refinement_flags`. A general description of constraints may appear in `_refine_special_details`.

Where no value is given, the assumed value is '`0`'. The permitted range is 0→ ∞.

`_refine_ls_number_parameters` *(numb)*

The number of parameters refined in the least-squares process. If possible this number should include some contribution from the restrained parameters.The restrained parameters are distinct from the constrained parameters (where one or more parameters are linearly dependent on the refined value of another). Least-squares restraints often depend on geometry or energy considerations and this makes their direct contribution to this number, and to the goodness-of-fit calculation, difficult to assess.

The permitted range is 0→ ∞.

`_refine_ls_number_reflns` *(numb)*

Number of reflections contributing to least-squares derivatives.

The permitted range is 0→ ∞.

`_refine_ls_number_restraints` *(numb)*

The number of restrained parameters. These are parameters which are not directly dependent on another refined parameter. Often restrained parameters involve geometry or energy dependencies. See also `_atom_site_constraints` and `_atom_site_refinement_flags`. A general description of refinement constraints may appear in `_refine_special_details`.

The permitted range is 0→ ∞.

`_refine_ls_R_factor_all`
`_refine_ls_R_factor_obs` *(numb)*

Residual factors for all reflection data, and for reflection data classified as 'observed' (see `_reflns_observed_criterion`). $R = (\sum||F_m| - |F_c||/\sum|F_m|)$; F_m and F_c are measured and calculated structure factors. This is the conventional R factor. See also `_refine_ls_wR_factor_` definitions.

The permitted range is 0·0→ ∞.

`_refine_ls_restrained_S_all`
`_refine_ls_restrained_S_obs` *(numb)*

The least-squares goodness-of-fit parameter S' for all data, and for observed data, after the final cycle of least squares. This parameter explicitly includes the restraints applied in the least-squares process. $S' = \{[\sum(w|Y_m - Y_c|^2) + \sum_r(w_r|P_c - P_t|^2)]/(N_{ref} + N_{restr} - N_{param})\}^{1/2}$ where the sum $\sum$ is over the specified reflection data; $\sum_r$ is over the restraint data; N_{ref} is the number of reflections used in the refinement (see `_refine_ls_number_reflns`); N_{param} is the number of refined parameters (see `_refine_ls_number_parameters`); N_{restr} is the number of restraints (see `_refine_ls_number_restraints`); Y_m and Y_c are the measured and calculated coefficients specified in `_refine_ls_structure_factor_coef`; P_c and P_t are the calculated and target restraint values; w is the least-squares reflection weight [1/(e.s.d. squared)] and w_r is the restraint weight. See also `_refine_ls_goodness_of_fit_` definitions.

The permitted range is 0·0→ ∞.

`_refine_ls_shift/esd_max`
`_refine_ls_shift/esd_mean` *(numb)*

The largest and the average ratios of the final least-squares parameter shift divided by the final estimated standard deviation.

The permitted range is 0·0→ ∞.

`_refine_ls_structure_factor_coef` *(char)*

Structure-factor coefficient $|F|$, F^2 or I, used in the least-squares refinement process.

`Inet`	net intensity
`Fsqd`	structure factor squared
`F`	structure-factor magnitude

Where no value is given, the assumed value is '**`F`**'.

`_refine_ls_weighting_scheme` *(char)*

The weighting scheme applied in the least-squares process. The standard code may be followed by a description of the weight.

`sigma`	based on measured e.s.d.'s
`unit`	unit or no weights applied
`calc`	calculated weights applied

Where no value is given, the assumed value is '**`sigma`**'.

`_refine_ls_wR_factor_all`
`_refine_ls_wR_factor_obs` *(numb)*

Residual factors for all reflection data, and for reflection data classified as 'observed' (see `_reflns_observed_criterion`). $wR = [\sum(w|Y_m - Y_c|^2)/\sum(wY_m^2)]^{1/2}$ where Y_m and Y_c are the measured and calculated coefficients specified by the `_refine_ls_structure_factor_coef`; w is the least-squares weight. See also the `_refine_ls_R_factor_` definitions.

The permitted range is 0·0→ ∞.

_refine_special_details *(char)*

Description of special aspects of the refinement process.

_refln_A_calc
_refln_A_meas *(numb)*

Calculated, measured structure-factor component $A = |F|\cos(\text{phase})$ in electrons.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list.

_refln_B_calc
_refln_B_meas *(numb)*

Calculated, measured structure-factor component $B = |F|\sin(\text{phase})$ in electrons.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list.

_refln_crystal_id *(char)*

Code identifying each crystal if multiple crystals are used. Is used to link with **_exptl_crystal_id** in the **_exptl_crystal_** list.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list.

_refln_F_calc
_refln_F_meas
_refln_F_sigma *(numb)*

The calculated, measured and standard deviation (derived from measured data) of the structure factors, in electrons.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list.

_refln_F_squared_calc
_refln_F_squared_meas
_refln_F_squared_sigma *(numb)*

Calculated, measured and estimated standard deviations of the squared structure factors, in electrons squared.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list.

_refln_index_h
_refln_index_k
_refln_index_l *(numb)*

Miller indices of the reflection. .

Appearance in list: yes.

_refln_intensity_calc
_refln_intensity_meas
_refln_intensity_sigma *(numb)*

The calculated, measured and standard deviation (derived from measured data) of the intensity, in the measured units.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list.

_refln_mean_path_length_tbar *(numb)*

Mean path length through the crystal for this reflection.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list. The units extensions are: ' ' (millimetres *1.0) '**_cm**' (centimetres *10.0).

_refln_observed_status *(char)*

Classification of a reflection. *E.g.* 'observed' or 'unobserved' according to a criterion specified in **_reflns_observed_criterion**.

o	observed by **_reflns_observed_criterion**
<	unobserved by **_reflns_observed_criterion**
–	systematically absent reflection
x	unreliable measurement – not used

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list. Where no value is given, the assumed value is '**o**'.

_refln_phase_calc
_refln_phase_meas *(numb)*

The calculated and measured structure-factor phase in degrees.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list.

_refln_refinement_status *(char)*

Status of reflection in the structure refinement process.

incl	included in ls process
excl	excluded from ls process
extn	excluded due to extinction

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list. Where no value is given, the assumed value is '**incl**'.

_refln_scale_group_code *(char)*

Code identifying the structure-factor scale. This code must correspond to one of the **_reflns_scale_group_code** values.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list.

Example(s): 1, 2, 3, s1, A, B, c1, c2, c3

_refln_sint/lambda *(numb)*

The $(\sin\theta)/\lambda$ for this reflection.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list. The permitted range is 0·0→ ∞. The units extensions are: ' ' (reciprocal ångströms *1.0) '**_pm**' (reciprocal picometres *100.) '**_nm**' (reciprocal nanometres /10.).

_refln_symmetry_epsilon *(numb)*

The symmetry reinforcement factor corresponding to the number of times the reflection indices are generated identically from the space-group symmetry operations.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list. The permitted range is 1→32.

_refln_symmetry_multiplicity *(numb)*

The number of symmetry-equivalent reflections. The equivalent reflections have the same structure-factor value because of the space-group symmetry and the Friedel relationship.

Appearance in list: yes. If looped, **_refln_index_** must be present in the same list. The permitted range is 1→24.

`_refln_wavelength` (*numb*)

The mean wavelength of radiation used to measure this reflection. This is an important parameter for data collected using energy-dispersive detectors or the Laue method.

Appearance in list: yes. If looped, `_refln_index_` must be present in the same list. The permitted range is 0·0→ ∞. The units extensions are: ' ' (ångströms *1.0) '`_pm`' (picometres /100.) '`_nm`' (nanometres *10.).

`_refln_wavelength_id` (*char*)

Code identifying the wavelength in the `_diffrn_radiation_` list. See `_diffrn_radiation_wavelength_id`.

Appearance in list: yes. If looped, `_refln_index_` must be present in the same list.

`_reflns_d_resolution_high`
`_reflns_d_resolution_low` (*numb*)

The highest and lowest resolution for the interplanar spacings in the reflection data. These are the smallest and largest *d* values.

The permitted range is 0·0→ ∞. The units extensions are: ' ' (ångströms *1.0) '`_pm`' (picometres /100.) '`_nm`' (nanometres *10.).

`_reflns_limit_h_max`
`_reflns_limit_h_min`
`_reflns_limit_k_max`
`_reflns_limit_k_min`
`_reflns_limit_l_max`
`_reflns_limit_l_min` (*numb*)

Miller indices limits for the reflection data. These need not be the same as the `_diffrn_reflns_limit_` values.

`_reflns_number_total`
`_reflns_number_observed` (*numb*)

The total number of reflections, and the number of 'observed' reflections, in the `_refln_` list (not the `_diffrn_refln_` list). The observed reflections satisfy the `_reflns_observed_criterion`. This may contain Friedel equivalent reflections according to the nature of the structure and the procedures used. The item `_reflns_special_details` describes the reflection data.

The permitted range is 0→ ∞.

`_reflns_observed_criterion` (*char*)

The criterion used to classify a reflection as 'observed'. This criterion is usually expressed in terms of an e.s.d. threshold.

Example(s): `>2sigma(I)`

`_reflns_scale_group_code` (*char*)

The code identifying a scale `_reflns_scale_meas_`. These are linked to the `_refln_` list by the `_refln_scale_group_code`.

Appearance in list: yes. If looped, `_reflns_scale_meas_` must be present in the same list.

`_reflns_scale_meas_F`
`_reflns_scale_meas_F_squared`
`_reflns_scale_meas_intensity` (*numb*)

Scales associated with `_reflns_scale_group_code`. These codes may not correspond to those in the `_diffrn_scale_` list.

Appearance in list: yes. If looped, `_reflns_scale_group_code` must be present in the same list. The permitted range is 0·0→ ∞.

`_reflns_special_details` (*char*)

A description of reflection data not covered by the other data names. It should include details of the Friedel reflection data.

`_symmetry_cell_setting` (*char*)

The cell settings for this space-group symmetry.

```
triclinic
monoclinic
orthorhombic
tetragonal
rhombohedral
trigonal
hexagonal
cubic
```

`_symmetry_equiv_pos_as_xyz` (*char*)

Symmetry equivalent position in the '*xyz*' representation. Except for the space group $P1$, this data will be repeated in a loop. The format of the data item is as per *International Tables for Crystallography*, Vol. A (1987). All equivalent positions should be entered, including those for lattice centring and a centre of symmetry, if present.

Appearance in list: yes.

Example(s): `'-y+x,-y,1/3+z'`

`_symmetry_Int_Tables_number` (*numb*)

Space-group number from *International Tables for Crystallography*, Vol. A (1987).

`_symmetry_space_group_name_Hall` (*char*)

Hall space-group symbol [Hall, S. R. (1981). *Acta Cryst.* A**37**, 517–525]. This symbol gives the space-group setting explicitly. Leave spaces between the separate components of the symbol.

Example(s): `'-P 2ac 2n'`,`'-R 3 2"'`,`'P 61 2 2 (0 0 -1)'`

`_symmetry_space_group_name_H-M` (*char*)

Hermann–Mauguin space-group symbol. Note that the H–M symbol does not necessarily contain complete information about the symmetry and the space-group origin. If used always supply the *full* symbol from *International Tables for Crystallography*, Vol. A (1987) and indicate the origin and the setting if it is not implicit. If there is any doubt that the equivalent positions can be uniquely deduced from this symbol specify the `_symmetry_equiv_pos_as_xyz` or `*_Hall` data items as well. Leave spaces between symbols referring to different axes.

Example(s): `'P 1 21/m 1'`,`'P 2/n 2/n 2/n (origin at -1)'`, `'R -3 2/m'`

APPENDIX II

Extract from a publication CIF

This is an extract from a CIF, generated by the program *CIFIO* (Hall, 1990), and submitted to *Acta Crystallographica* Section C for publication (Willis, Beckwith & Tozer, 1991). This CIF contains more data than required for publication purposes. Items requested for inclusion in this CIF but not present in the *Xtal* binary file are flagged with a '?'.

```
data_TOZ

_audit_creation_date                  91-03-20
_audit_creation_method
     from_xtal_archive_file_using_CIFIO
_audit_update_record
; 91-04-09  text and data added by Tony Willis.
  91-04-15  rec'd by co-editor with diagram as ms. HL7.
  91-04-17  adjustments based on first referees report.
  91-04-18  adjustments based on second referees report.
;

_chemical_name_systematic
;
trans-3-Benzoyl-2-(tert-butyl)-4-(isobutyl)-
1,3-oxazolidin-5-one
;

_chemical_formula_moiety              'C18 H25 N O3'
_chemical_formula_sum                 'C18 H25 N O3'
_chemical_formula_weight              303.40
_chemical_melting_point               ?

_computing_data_collection
    'Philips PW1100/20 software 1976'
_computing_cell_refinement
    'LATCON (Xtal: Schwarzenbach & King)'
_computing_data_reduction
   'pwredu (McLaughlin) ADDREF SORTRF (Xtal)'
_computing_structure_solution
    'SHELXS86 (Sheldrick)'
_computing_structure_refinement
   'SHELX76 (Sheldrick) CRYLSQ (Xtal: Olthof)'
_computing_publication_material
   'BONDLA CIFIO (Xtal: Hall)

_cell_length_a                        5.959(1)
_cell_length_b                        14.956(1)
_cell_length_c                        19.737(3)
_cell_angle_alpha                     90
_cell_angle_beta                      90
_cell_angle_gamma                     90
_cell_volume                          1759.0(3)
_cell_formula_units_Z                 4
_cell_measurement_temperature         293
_cell_measurement_reflns_used          25
_cell_measurement_theta_min            25
_cell_measurement_theta_max            31

_symmetry_cell_setting                orthorhombic
_symmetry_space_group_name_H-M        'P 21 21 21'
_symmetry_space_group_name_Hall       P_2ac_2ab
loop_
_symmetry_equiv_pos_as_xyz
       +x,+y,+z  1/2-x,-y,1/2+z
       1/2+x,1/2-y,-z  -x,1/2+y,1/2-z

_exptl_crystal_description            prism
_exptl_crystal_colour                 colourless
_exptl_crystal_size_max               0.32
_exptl_crystal_size_mid               0.27
_exptl_crystal_size_min               0.10
_exptl_crystal_density_diffrn          1.146
_exptl_crystal_density_meas           ?
_exptl_crystal_F_000                  656

_exptl_absorpt_coefficient_mu         5.9
_exptl_absorpt_correction_type        'shelx76 gaussian'
_exptl_absorpt_correction_T_min       .933
_exptl_absorpt_correction_T_max       .824


_diffrn_special_details
;
\q scan width (1.0 + 0.14tan\q)\%, \q scan rate
1.2\% per min.Background counts for 5 s on each
side every scan.
;
_diffrn_ambient_temperature           293
_diffrn_radiation_wavelength          1.5418
_diffrn_radiation_type                'Cu K\a'
_diffrn_radiation_source              'X-ray tube'
_diffrn_radiation_monochromator       'graphite'
_diffrn_radiation_detector             ?
_diffrn_measurement_device
       'Philips PW1100/20 diffractometer'
_diffrn_measurement_method            \q/2\q

_diffrn_standards_number              3
_diffrn_standards_interval_time       120
_diffrn_standards_decay_%             0

loop_
_diffrn_standard_refln_index_h
_diffrn_standard_refln_index_k
_diffrn_standard_refln_index_l
       3 2 4    1 9 1    3 0 10

loop_
_diffrn_attenuator_code
_diffrn_attenuator_scale
       1     16.976

_diffrn_reflns_number                 1592
_diffrn_reflns_av_R_equivalents       0
_diffrn_reflns_av_sigmaI/netI         .027
_diffrn_reflns_limit_h_min            0
_diffrn_reflns_limit_h_max            6
_diffrn_reflns_limit_k_min           -17
_diffrn_reflns_limit_k_max            0
_diffrn_reflns_limit_l_min            0
_diffrn_reflns_limit_l_max            22
_diffrn_reflns_theta_min              3.71
_diffrn_reflns_theta_max              61.97
_diffrn_reflns_reduction_process       ?

loop_
_atom_type_symbol
_atom_type_oxidation_number
_atom_type_number_in_cell
_atom_type_scat_dispersion_real
_atom_type_scat_dispersion_imag
_atom_type_scat_source
  C  0  72  .017  .009
  International_Tables_Vol_IV_Table_2.2B
  H  0  100  0  0
  International_Tables_Vol_IV_Table_2.2B
  O  0  12  .047  .032
  International_Tables_Vol_IV_Table_2.2B
  N  0  4  .029  .018
  International_Tables_Vol_IV_Table_2.2B

loop_
_atom_site_label
_atom_site_fract_x
_atom_site_fract_y
_atom_site_fract_z
_atom_site_U_iso_or_equiv
_atom_site_thermal_displace_type
_atom_site_calc_flag
_atom_site_calc_attached_atom
O1  .4154(4)  .5699(1)  .3026(1)  .060(1) Uani ? ?
C2  .5630(5)  .5087(2)  .3246(1)  .060(2) Uani ? ?
C3  .5350(5)  .4920(2)  .3997(1)  .048(1) Uani ? ?
N4  .3570(3)  .5558(1)  .4167(1)  .039(1) Uani ? ?
C5  .3000(5)  .6122(2)  .3581(1)  .045(1) Uani ? ?
O21  .6958(5)  .4738(2)  .2874(1)  .090(2) Uani ? ?
C31  .4869(6)  .3929(2)  .4143(2)  .059(2) Uani ? ?
C32  .2552(7)  .3558(2)  .3953(2)  .073(2) Uani ? ?
C321  .209(1)  .3542(4)  .3211(3)  .111(4) Uani ? ?
C322  .230(1)  .2626(3)  .4264(3)  .149(5) Uani ? ?
C41  .2034(4)  .5476(2)  .4682(1)  .041(1) Uani ? ?
# ................................... data omitted for brevity
H321C  .04(1)  .318(3)  .320(2)  .14000 Uiso ? ?
H322A  .25(1)  .272(4)  .475(3)  .19000 Uiso ? ?
H322B  .34976  .22118  .40954  .19000 Uiso  calc C322
H322C  .08(1)  .234(4)  .397(3)  .19000 Uiso ? ?
```

```
H412   -.007(6)   .447(2)   .552(2)   .08000 Uiso ? ?
H513B  .115(7)   .757(3)   .426(2)   .09000 Uiso ? ?
H513C  .329(6)   .817(2)   .430(2)   .09000 Uiso ? ?

loop_
_atom_site_aniso_label
_atom_site_aniso_U_11
_atom_site_aniso_U_22
_atom_site_aniso_U_33
_atom_site_aniso_U_12
_atom_site_aniso_U_13
_atom_site_aniso_U_23
O1 .071(1) .076(1) .0342(9) .008(1) .0051(9) -.0030(9)
C2 .060(2) .072(2) .047(1) .002(2) .013(1) -.009(1)
C3 .038(1) .060(2) .044(1) .007(1) .001(1) -.005(1)
N4 .037(1) .048(1) .0325(9) .0025(9) .0011(9) -.0011(9)
C5 .043(1) .060(1) .032(1) .001(1) -.001(1) .001(1)
# ................................. data omitted for brevity
O21 .094(2) .109(2) .068(1) .023(2) .038(1) -.010(1)
C51 .048(2) .059(2) .049(1) .002(1) -.000(1) .007(1)
C511 .048(2) .071(2) .097(3) -.008(2) -.003(2) .010(2)
C512 .078(2) .083(2) .075(2) .009(2) -.005(2) .033(2)
C513 .074(2) .055(2) .075(2) .004(2) .001(2) -.010(2)

_refine_special_details
      sfls:_F_calc_weight_full_matrix

_refine_ls_structure_factor_coef  F
_refine_ls_matrix_type            full
_refine_ls_weighting_scheme
      'calc w=1/(\s^2^(F)+0.0004F^2^)'
_refine_ls_hydrogen_treatment
      'refxyz except H332B noref'
_refine_ls_extinction_method      Zachariasen
_refine_ls_extinction_coef        3514(42)
_refine_ls_extinction_expression
    'equ(22) p292 "Crystallographic Computing" (1970)'

_refine_ls_abs_structure_details
;
The absolute configuration was assigned to agree
with the known chirality at C3 arising from its
precursor l-leucine.
;
_refine_ls_abs_structure_Flack      0
_refine_ls_number_reflns            1408
_refine_ls_number_parameters        272
_refine_ls_number_restraints        0
_refine_ls_number_constraints       0
_refine_ls_R_factor_all                .038
_refine_ls_R_factor_obs                .034
_refine_ls_wR_factor_all               .044
_refine_ls_wR_factor_obs               .042
_refine_ls_goodness_of_fit_all        1.462
_refine_ls_goodness_of_fit_obs        1.515
_refine_ls_shift/esd_max               .164
_refine_ls_shift/esd_mean              .044
_refine_diff_density_min              -.108
_refine_diff_density_max               .131

_geom_special_details                 ?

loop_
_geom_bond_atom_site_label_1
_geom_bond_atom_site_label_2
_geom_bond_distance
_geom_bond_site_symmetry_1
_geom_bond_site_symmetry_2
_geom_bond_publ_flag
  O1  C2  1.342(4)  .  .  yes
  O1  C5  1.439(3)  .  .  yes
  C2  C3  1.512(4)  .  .  yes
  C2  O21  1.199(4)  .  .  yes
  C3  N4  1.465(3)  .  .  yes
  C3  C31  1.537(4)  .  .  yes
  C3  H3  1.00(3)  .  .  ?
  N4  C5  1.472(3)  .  .  yes
  N4  C41  1.374(3)  .  .  yes
  C5  C51  1.528(4)  .  .  yes
  C5  H5  1.00(3)  .  .  ?
  C31  C32  1.535(5)  .  .  yes
  C31  H31A  .95(3)  .  .  ?
```

```
  C31  H31B  .96(3)  .  .  ?
  C32  C321  1.490(7)  .  .  yes
  C32  C322  1.531(6)  .  .  yes
  C32  H32  1.10(4)  .  .  ?
# ................................. data omitted for brevity
  C513  H513C  1.02(4)  .  .  ?

loop_
_geom_angle_atom_site_label_1
_geom_angle_atom_site_label_2
_geom_angle_atom_site_label_3
_geom_angle
_geom_angle_site_symmetry_1
_geom_angle_site_symmetry_2
_geom_angle_site_symmetry_3
_geom_angle_publ_flag
  C2  O1  C5  111.6(2)  .  .  .  yes
  O1  C2  C3  110.9(2)  .  .  .  yes
  O1  C2  O21  122.2(3)  .  .  .  yes
  C3  C2  O21  127.0(3)  .  .  .  yes
  C2  C3  N4  101.3(2)  .  .  .  yes
  C2  C3  C31  111.3(2)  .  .  .  yes
  C2  C3  H3  107(1)  .  .  .  ?
  N4  C3  C31  116.7(2)  .  .  .  yes
  N4  C3  H3  110(1)  .  .  .  ?
  C31  C3  H3  110(1)  .  .  .  ?
  C3  N4  C5  111.2(2)  .  .  .  yes
  C3  N4  C41  126.4(2)  .  .  .  yes
  C5  N4  C41  118.6(2)  .  .  .  yes
# ................................. data omitted for brevity
  H513B  C513  H513C  104(1)  .  .  .  ?

_reflns_limit_h_min              0
_reflns_limit_h_max              6
_reflns_limit_k_min              0
_reflns_limit_k_max              17
_reflns_limit_l_min              0
_reflns_limit_l_max              22
_reflns_number_total             1592
_reflns_number_observed          1408
_reflns_observed_criterion
          refl_observed_if_F_>_6.0_sigma(F)
_reflns_d_resolution_high         0.8733
_reflns_d_resolution_low          11.9202

loop_
_reflns_scale_group_code
_reflns_scale_meas_F            1  .960926

loop_
_refln_index_h
_refln_index_k
_refln_index_l
_refln_F_meas
_refln_F_calc
_refln_F_sigma
_refln_observed_status
_refln_scale_group_code
              ?  ?  ?  ?  ?  ?  ?  ?

#----------------------------------------------

_publ_requested_journal
           'Acta Crystallographica C'

_publ_contact_author
;
      Dr Anthony C. Willis
      Research School of Chemistry
      Australian National University
      GPO Box 4
      Canberra, A.C.T.
      Australia    2601
;
_publ_contact_letter
;
Please consider this CIF submission for
publication as a Short Format Paper in Acta
Crystallographica C.  The figure and structure
factor listings will be sent by normal mail.
;
```

```
_publ_requested_coeditor_name   'Prof. S.R.Hall'

_publ_contact_author_phone      '616 249 4109'
_publ_contact_author_fax        '616 249 0750'
_publ_contact_author_email      willis@RSC3.anu.oz.au

loop_
_publ_author_name
_publ_author_address

     'Willis, Anthony C.'
;     Research School of Chemistry
      Australian National University
      GPO Box 4
      Canberra, A.C.T.
      Australia    2601
;
     'Beckwith, Athelstan L.J.'
;     Research School of Chemistry
      Australian National University
      GPO Box 4
      Canberra, A.C.T.
      Australia    2601
;
     'Tozer, Matthew J.'
;     Research School of Chemistry
      Australian National University
      GPO Box 4
      Canberra, A.C.T.
      Australia    2601
;

_publ_section_title
;
 trans-3-Benzoyl-2-(tert-butyl)-4-(isobutyl)-
1,3-oxazolidin-5-one
;

_publ_section_abstract
;
 The oxazolidinone ring is a shallow envelope
conformation with the tert-butyl and isobutyl groups
occupying trans-positions with respect to the ring.
The angles at the N atom sum to 356.2\%, indicating a
very small degree of pyramidalization at this atom.
This is consistent with electron delocalization between
the N atom and the carbonyl centre [N-C=O = 1.374(3)\%A].
;

_publ_section_experimental
;
 The \q scan width was (1.0+0.14tan\q)\% with a \q scan
rate of 1.2\% per min. and background counts for 5s on
each side of every scan. The absolute configuration has
been assigned to agree with the known chirality at C3
arising from its precursor l-leucine. Refinement was by
full-matrix least-squares methods.
;

_publ_section_comment
;
The structure of the title compound was undertaken
to establish whether an isobutyl group would be formed
cis or trans to the tert-butyl group as part of a
study of diastereoselectivities of free radical
reactions.  Details of the synthetic work will be
published elsewhere (Beckwith, Chai & Tozer, 1991).

The X-ray analysis shows that the tert-butyl and
isobutyl groups are trans, as expected from nmr
evidence.  The oxazolidinone ring is in an envelope
conformation with C5 the out-of-plane atom.  The angle
between the least-squares plane through atoms O1, C2,
C3 and N4 and the plane of N4, C5 and O1 is 10.9(2)\%.
Intra-annular torsion angles (starting with C5-O1-C2-C3
and proceeding around the ring) are 9.5, -2.3, -5.4,
10.7 and -12.3(3)\%. A search of the January 1990
Version of the Cambridge Structure Database (Allen,
Kennard & Taylor, 1983) revealed three structure
determinations with 1,3-oxazolidin-5-one fragments
(Seebach, Boes, Naef & Schweizer 1983, Karady, Amato
& Weinstock 1984, Weber, Aeschimann, Maetzke & Seebach,
1986). Bond lengths and angles in these compounds are
comparable with those in the present study; the
magnitudes of their respective maximum intra-annular
torsion angle are 10.2, 1.9 and 10.2\%.  Diagrams and
most calculations were performed with the Xtal3.0
package (Hall & Stewart, 1990), as was the generation
of the Crystallographic Information File used for the
submission of this paper.
;

_publ_section_references
;
Allen, F.H., Kennard, O. and Taylor, R. (1983).
Accounts Chem. Res., 16, 146-153.

Beckwith, A.L.J., Chai, C.L.L. & Tozer, M.J. (1991).
Manuscript in preparation.

Hall, S.R. and Stewart, J.M. (1990). Eds. Xtal3.0
Reference Manual. University of Western Australia
and University of Maryland.

Karady, S., Amato, J.S. & Weinstock, L.M. (1984).
Tetrahedron Lett., 25, 4337-4340.

McLaughlin, G.M. (1983).  PWREDU. Program for data
reduction for Philips PW1100/20 diffractometer.
Australian National Univ., Canberra.

Seebach, D., Boes,M., Naef, W.B. & Schweizer, W.B.
(1983). J Am. Chem. Soc., 105,5390-5398.

Sheldrick, G.M. (1976).  SHELX76. Program for crystal
structure determination.  Univ. of Cambridge, England.

Sheldrick, G.M. (1985).  SHELXS86. Program for the
solution of crystal structures.  Univ. of Gottingen,
Federal Republic of Germany.

Weber, T., Aeschimann, R., Maetzke, T. & Seebach, D.
(1986). Helv. Chim. Acta, 69,1365-1377.
;

_publ_section_figure_captions
;
 View of C~18~H~25~NO~3~ showing the labelling of the
non-H atoms. Thermal ellipsoids are shown at 50%
probability levels; H atoms are drawn as small circles
of arbitrary radius.
;
```

References

ALLEN, F. H. & EDGINGTON, P. R. (1992). *CIFER: a Program for CIF Generation*. Crystallographic Data Centre, Cambridge, England.

BARNARD, J. M. (1990). *J. Chem. Inf. Comput. Sci.* **30**, 81–96.

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

BROWN, I. D. (1988). *Acta Cryst.* A**44**, 232.

COOK, A. P. F. (1991). *Implementing SMD in STAR: Dictionary Definition Language*. ORAC Ltd. report (March 8), Leeds, England.

HALL, S. R. (1990). *CIFIO: XTAL3.0 Crystallographic Program System*, edited by S. R. HALL & J. M. STEWART. Univs. of Western Australia, Australia, and Maryland, USA.

HALL, S. R. (1991*a*). *J. Chem. Inf. Comput. Sci.* **31**, 326–333.

HALL, S. R. (1991*b*). *CYCLOPS: a Program for Validating Data Names in a Text File*. Univ. of Western Australia, Australia.

HALL, S. R. & SIEVERS, R. (1990). *QUASAR: a Program for Accessing a STAR File*. Univs. of Western Australia, Australia, and Bonn, Germany.

OSAKI, K. (1991). *EDCIF-J: an Interactive Program for Creating a CIF*. Personal communication.

WILLIS, A. C., BECKWITH, A. L. J. & TOZER, M. J. (1991). *Acta Cryst.* C**47**, 2276–2277.