commissioning and maintenance cycle. Information about COMCIFS activities and other CIF developments may be obtained from http://www.iucr.org/iucr-top/cif/.

1.1.8. Diversification: the Molecular Information File and dictionary definition language

While the primary thrust of these activities was the development of an exchange mechanism for crystal-structure reports, there was also interest in enriching the description of the chemical properties and behaviour of the compounds under study. Some work was therefore done to broaden the descriptions of bond order that were present in rudimentary form in the core CIF dictionary and to develop more detailed two-dimensional graphical representations of chemical molecules. The result of this work was the Molecular Information File (MIF), described in Chapter 2.4. As with CIF, the specific data items required for MIF were defined in a dictionary.

This work has extended the STAR File approach into chemistry. It is envisaged that later modules could describe spectroscopic data, reaction schemes and much more. Particular requirements of chemical structural databases are the need to query for generic structures, and the need to allow for the labelling and comparison of libraries of substructural components. Both requirements can be met by features of the STAR File, but they are features omitted from CIF. In practice, therefore, data files in MIF format cannot be readily accessed by most crystallographic applications, and the format is at present little used by crystallographers.

An important outcome of the work on MIF was the recognition that attributes of the data items needed for a particular application can be recorded using the same formalism as the data files themselves. This gave rise to the idea of a dictionary definition language (DDL), a set of tags for describing the names and attributes of data items. The dictionaries (the collections of data names for CIF and MIF applications) could then be constructed as STAR Files themselves, with the immediate result that software written to parse data files could equally easily parse the associated dictionaries. Now it became feasible to build into applications the ability to validate data by dynamically reading and interpreting the properties associated with a data tag in an accompanying dictionary.

The idea of a DDL was proposed by Tony Cook during early discussions on MIF and was adopted while the original CIF paper (Hall *et al.*, 1991) was in the press. The original core CIF dictionary was therefore produced with an early version of a DDL that was never fully documented (Fig. 1.1.8.1). Building on early experience with the core dictionary and the technical evolution of MIF, Hall & Cook (1995) worked through several revisions before publishing DDL version 1.4, the stable version described in Chapter 2.5 of this volume. Because of the circumstances in which it was developed, this dictionary definition language is able to accommodate both the flat-file quasi-relational structure of CIF and the more hierarchical multiple-looped data model of MIF.

1.1.9. The macromolecular Crystallographic Information File

The original goal of CIF was the creation of an archive format for the description and results of experiments in small-molecule and inorganic crystallography. The data names and their definitions were embodied in the *core* dictionary, so called because most of the terms in it were considered common to any crystallographic application. In 1990, the IUCr formed a working group, chaired by Paula Fitzgerald, to expand this dictionary to meet the additional requirements of macromolecular crystallography. The resulting expanded dictionary was to be known as the macromolecular CIF (mmCIF) dictionary.

```
##############################################################################
#
#                        DDL Data Name Descriptions
#                        --------------------------
#
# _compliance           The dictionary version in which the item is defined.
#
# _definition           The description of the item.
#
# _enumeration          A permissible value for an item. The value 'unknown'
#                       signals that the item can have any value.
#
# _enumeration_default  The default value for an item if it is not specified
#                       explicitly. 'unknown' means the default is not known.
#
# _enumeration_detail   The description of a permissible value for an item.
#                       Note that the code '.' normally signals a null
#                       or 'not applicable' condition.
#
# _enumeration_range    The range of values for a numerical item. The
#                       construction is 'min:max'. If 'max' is omitted then the
#                       item can have any value greater than or equal to 'min'.
#
# _esd                  Signals whether an estimated standard deviation is
#                       expected to be appended (enclosed within brackets)
#                       to a numerical item. May be 'yes' or 'no'.
#
# _esd_default          The default value for the esd of a numerical item
#                       if a value is not appended.
#
# _example              An example of the item.
#
# _example_detail       A description of the example.
#
# _list                 Signals whether an item is expected to occur in a looped
#                       list. Possible values: 'yes','no' or 'both'.
#
# _list_identifier      Identifies a data item that MUST appear in the list
#                       containing the currently defined data item.
#
# _name                 The data name of the item defined.
#
# _type                 The data type 'numb' or 'char' (latter includes 'text').
#
# _units_extension      The data-name extension code used to specify the units
#                       of a numerical item.
#
# _units_description    A description of the units.
#
# _units_conversion     The method of converting the item into a value based
#                       on the default units. Each conversion number is
#                       preceded by an operator code *, /, +, or - which
#                       indicates how the conversion number is applied.
#
# _update_history       A record of the changes to this file.
#
#-eof-eof-eof-eof-eof-eof-eof-eof-eof-eof-eof-eof-eof-eof-eof-eof-eof-eof-eof
```

Fig. 1.1.8.1. Informal DDL used in the initial version of the core CIF dictionary (now superseded by DDL1.4).

The group's original short-term goal was to fulfil the IUCr mandate of defining data names for an adequate description of a macromolecular crystallographic experiment and its results. Longer-term goals were also determined: to provide sufficient data names so that the experimental section of a structure paper could be written automatically and to facilitate the development of tools so that computer programs could easily interface with CIF data files. A number of informal and formal meetings were held to describe the progress of this project and to solicit community feedback.

An important meeting took place at the University of York in April 1993. The attendees included the mmCIF working group, structural biologists and computer scientists. Vigorous discussion arose on whether the formal structure of the dictionary implemented in the then-current dictionary definition language (DDL1) could deal with the complexity of macromolecular data sets. There were criticisms that the data typing was not strong enough and that there were no formal links expressing relationships between data items. A working group was formed to address these issues, resulting in a second workshop in Tarrytown, New York, in October 1993. The discussions at this second meeting focused on the development of software tools and the requirements of an enhanced DDL. Such a DDL was proposed during a third workshop at the Free University of Brussels in October 1994. This new DDL (DDL2; Chapter 2.6) was designed by John Westbrook and sought to address the various problems identified at the preceding workshops, while retaining compatibility with existing CIF data files.