*International Tables for Crystallography* (2006). Vol. G, Section 3.1.5.4.1, pp. 77–78.

Example 3.1.5.1. *A DDL1 dictionary identification block.*

```
 data_on_this_dictionary
    _dictionary_name           cif_core.dic
    _dictionary_version        2.3.1
    _dictionary_update         2005-06-27
    _dictionary_history
;  1991-05-27  Created from CIF Dictionary text. SRH
   1991-05-30  Validated with CYCLOPS & CIF ms. SRH
   ...
;
```

be seen that there is no sure way of working out the category from the complete data name except by referring to its `_category` attribute in the associated dictionary. This differs from the DDL2 convention of including an explicit separator (a full stop) between the category name and the remainder of a data name.

While it is not mandatory that a data name should incorporate its category name as a leading component, authors are strongly encouraged to adopt this convention. A small number of core data items that did not conform to this convention have been deprecated in later releases of the core dictionary. However, in the powder dictionary the convention has been broken so that one can present data sets separately or merge them together. In this dictionary, some data names beginning with the strings `_pd_calc`, `_pd_meas` and `_pd_proc` all belong formally to the category PD_DATA. This allows calculated data values to be tabulated with raw and processed measurements if this is useful.

One other case where a data name does not begin with its associated category name is that of the pseudo data names such as `_exptl_[]` that appear in the dictionary to describe the purpose of a category (Section 3.1.5.3). Such data names are always assigned the category CATEGORY_OVERVIEW and are further differentiated from other data names by having a data type of 'null'.

### 3.1.5.1. The dictionary identification block

As mentioned above, the dictionary file must contain information that unambiguously states its identity and version. In DDL1-based CIF dictionaries, this is achieved by itemizing the full set of dictionary attributes (see Section 2.5.6.5) within a data block named `data_on_this_dictionary`, as in Example 3.1.5.1 from the core dictionary.

### 3.1.5.2. Irreducible sets of data items

In general, a dictionary data block defines a single data item. However, there are instances where several related data names are defined in the same data block. Sometimes this has been done for convenience, to produce a compact listing of similar data names that have common attributes and whose small differences in meaning can best be expressed by a single definition. Such groupings are discouraged, except where they represent components of a larger entity that has no sensible meaning in the absence of any of the components. For example, the data block `data_refln_index_` defines the three data items `_refln_index_h`, `_refln_index_k` and `_refln_index_l` that represent the Miller indices of a reflection. All three indices must have a value in order to specify a reflection and so each has no meaning in isolation.

Note that there is no formal method of expressing this close relationship within DDL1 except by grouping the definitions in the same data block in this way. In DDL2 dictionaries, it is common to assign the components of an irreducible set to a specific subcategory.

Example 3.1.5.2. *A category description in a DDL1 dictionary.*

```
data_exptl_[]
    _name                          '_exptl_[]'
    _category                        category_overview
    _type                            null
    loop_ _example
    _example_detail
# - - - - - - - - - - - - - - - - - - - - - - - - -
;   _exptl_absorpt_coefficient_mu     0.962
    _exptl_absorpt_correction_type    psi-scan
    _exptl_absorpt_process_details
                'North, Phillips & Mathews (1968)'
    _exptl_absorpt_correction_T_min   0.929
    _exptl_absorpt_correction_T_max   0.997
;
;   Example 1 - based on a paper by Steiner [Acta
                Cryst. (1996), C52, 2554-2556].
;
# - - - - - - - - - - - - - - - - - - - - - - - - -
    _definition
;      Data items in the EXPTL category record
       details about the experimental work prior
       to the intensity measurements and details
       about the absorption-correction technique
       employed.
;
```

### 3.1.5.3. Category descriptions

As discussed above, categories in DDL1 are intended as 'natural groupings' of data items. To document the purpose of a category within a dictionary, 'pseudo' data names are used. All pseudo data names are assigned a `_category` attribute of `category_overview` and have an associated `_type` value of 'null'. They are also named by convention as `_category_name_[dictionarycode]`, for example `_pd_data_[pd]` for the description of the PD_DATA category in the powder dictionary (indicated by the code 'pd' in square brackets). For the core dictionary, *dictionarycode* is not given, resulting in names like `_exptl_[]` to describe the EXPTL category.

Example 3.1.5.2 is a slightly edited extract from the core dictionary showing how a data block for a category description is composed, including the presence of an example.

Note that the *dictionarycode* extension allows a dictionary to include comments on items that it defines in a category already established in the core dictionary. For example, the modulated structures dictionary includes the category overview item `_audit_link_[ms]`. This describes the convention adopted to express the relationship between data blocks in a modulated structures data file using the `_audit_link_` data names already defined in the core dictionary.

### 3.1.5.4. Data-item definitions

The data blocks described in Sections 3.1.5.1 and 3.1.5.3 are used to identify the dictionary and to describe the nature and purpose of a category. The remaining data blocks in a dictionary provide the attributes of data values in a form suitable for machine extraction and validation. The following examples show how this is done for various types of data.

#### 3.1.5.4.1. *Definitions of single quantities*

Example 3.1.5.3 is the core dictionary definition of the data name for the ambient temperature during the experiment. Because this is a single (non-looped) value, the relevant data name is one among several discrete items in the DIFFRN category. No further description of its relationship to other data items is required.

The type of the associated data value (*numb* for numerical) is specified, together with any constraint on its legal value. The range

Example 3.1.5.3. *A simple definition of a data item describing a physical quantity.*

```
data_diffrn_ambient_temperature
    _name               '_diffrn_ambient_temperature'
    _category                    diffrn
    _type                        numb
    _type_conditions             esd
    _enumeration_range           0.0:
    _units                       K
    _units_detail                kelvin
    _definition
;      The mean temperature in kelvins at which the
       intensities were measured.
;
```

specified (`0.0:`) indicates that it may be any non-negative real number. The physical units of the quantity are also indicated.

The `_definition` attribute is a concise human-readable documentation of the meaning associated with the data name.

Example 3.1.5.4 is taken from the powder dictionary and illustrates a data item that can have only one of a limited set of values. This data item indicates the geometry of the experiment. The associated data value is of type *char* and may legally take only one of the two possible values listed.

### 3.1.5.4.2. *Looped data*

Many of the attributes of looped data items, such as their physical units or valid numerical values, may be defined in exactly the same way as for non-looped data. However, more care needs to be taken to describe the relationships between different looped data items.

Consider the following example listing of some three-dimensional atom-site coordinates and displacement parameters.

```
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
    O1 .4154(4) .56990(10) .3026000   .0600(10) Uani
    C2 .5630(5) .5087(2)   .32460(10) .060(2)   Uani
    C3 .5350(5) .4920(2)   .39970(10) .0480(10) Uani
    N4 .3570(3) .55580(10) .4167000   .0390(10) Uani
    C5 .3000(5) .6122(2)   .35810(10) .0450(10) Uani

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)

loop_
     _geom_bond_atom_site_label_1
     _geom_bond_atom_site_label_2
     _geom_bond_distance
      O1  C2   1.342(4)
      O1  C5   1.439(3)
      C2  C3   1.512(4)
      C2  O21  1.199(4)
```

These loops, or tables of values, are properties of atom sites, each identified by a label such as O1. The definition of a data name such as `_atom_site_U_iso_or_equiv` expresses this by using the DDL1 `_list_reference` attribute (Example 3.1.5.5).

Example 3.1.5.4. *A data item that can take only one of a discrete set of allowed values.*

```
data_pd_spec_mount_mode
    _name                          '_pd_spec_mount_mode'
    _category                       pd_spec
    _type                           char
    loop_ _enumeration              reflection
                                    transmission
    _definition
;      A code describing the beam path through
       the specimen.
;
```

Example 3.1.5.5. *Definition relating a looped data item to the item used to identify a 'loop packet', or row of entries in a table.*

```
data_atom_site_U_iso_or_equiv
    _name                '_atom_site_U_iso_or_equiv'
    _category                       atom_site
    _type                           numb
    _type_conditions                esd
    _list                           yes
    _list_reference                '_atom_site_label'
```

Example 3.1.5.6. *Definition of a mandatory item within a loop.*

```
data_atom_site_label
    _name                          '_atom_site_label'
    _category                       atom_site
    _type                           char
    _list                           yes
    _list_mandatory                 yes
    loop_ _list_link_child
           '_atom_site_aniso_label'
           '_geom_bond_atom_site_label_1'
           '_geom_bond_atom_site_label_2'
```

For an entry in the table to make sense, the site identifier must be present, so the definition for `_atom_site_label` declares it a mandatory item within its list (Example 3.1.5.6).

It is common for an atom-site identifier to be used in several related tabulations in a particular crystal structure description, and in a CIF description this means that it may occur in several different looped lists. The dictionary definition gives a formal account of this by listing the data names in other looped lists which are just different manifestations of this same item. This is done using the `_list_link_child` attribute, which identifies the data names to which the one being currently defined is 'parent'. In Example 3.1.5.6 (which is a subset of the full list in the core dictionary), `_atom_site_aniso_label`, `_geom_bond_atom_site_label_1` and `_geom_bond_atom_site_label_2` are identified as children of `_atom_site_label`.

It can been seen immediately that `_atom_site_aniso_label` is the atom-site identification label appearing in the second table in the example listing above, and the `_geom_bond_` items are clearly atom-site labels in a table of bonding properties between specified sites. There is, however, a difference between the two secondary tables: the bond-properties table is described by data items in the GEOM_BOND category, but the table of anisotropic displacement parameters includes data names that have the same `_category` attribute as the coordinate data items, namely ATOM_SITE. The latter is an example of multiple lists or tables belonging to the same category, a feature permitted only in DDL1-based data files.

### 3.1.5.4.3. *Units*

The physical units in which a quantitative data item must be expressed are identified by the DDL1 attributes `_units` and

references