

## 2.6. SPECIFICATION OF A RELATIONAL DICTIONARY DEFINITION LANGUAGE (DDL2)

Table 2.6.5.1. Relationships defined by `_item_related.function_code`

Code	Meaning
alternate	The item identified in <code>_item_related.related_name</code> is an alternative expression in terms of its application and attributes to the item in this definition
alternate_exclusive	The item identified in <code>_item_related.related_name</code> is an alternative expression in terms of its application and attributes to the item in this definition; only one of the alternative forms may be specified
convention	The item identified in <code>_item_related.related_name</code> differs from the defined item only in terms of a convention in its expression
conversion_constant	The item identified in <code>_item_related.related_name</code> differs from the defined item only by a known constant
conversion_arbitrary	The item identified in <code>_item_related.related_name</code> differs from the defined item only by an arbitrary constant
replaces	The defined item replaces the item identified in <code>_item_related.related_name</code>
replacedby	The defined item is replaced by the item identified in <code>_item_related.related_name</code>
associated_value	The item identified in <code>_item_related.related_name</code> is meaningful when associated with the defined item
associated_esd	The item identified in <code>_item_related.related_name</code> is the standard uncertainty (estimated standard deviation) of the defined item

Example 2.6.5.4 illustrates the definition of a pair of mmCIF categories, CITATION and CITATION\_AUTHOR, which share a common data item, `_citation.id`. This example illustrates how an item that occurs in multiple categories may be defined. In the case of the citation identifier, `_citation.id`, the ITEM category is preceded by a `loop_` directive and within this loop all of the definitions of the citation identifier are listed. For instance, the citation identifier is also an item in category CITATION\_AUTHOR, where it has the item name `_citation_author.citation_id`. For conformity with the manner in which the core CIF dictionary has been organized, a skeleton definition of the child data item `_citation_author.citation_id` has been included in the dictionary. In fact, this skeleton definition is formally unnecessary.

As a matter of style, the mmCIF dictionary generally defines all of the instances of a data item within the parent definition. Items that are related to the parent definition are also listed in the ITEM\_LINKED category. The repetition of a data item in multiple categories gives rise to parent–child relationships between such definitions. These relationships are stored in the ITEM\_LINKED category. In Example 2.6.5.4, this category stores the list of data items that are children of the citation identifier `_citation.id`. These include `_citation_author.citation_id`, `_citation_editor.citation_id` and `_software.citation_id`.

### 2.6.6. Detailed DDL2 specifications

DDL2 is presented here (Chapter 4.10) in the form of a dictionary that is defined in terms of its own definitional elements. This self-consistent description not only provides a prototype for other application dictionaries, but also provides a mechanism by which

Example 2.6.5.3. Definition of an mmCIF category.

```

save_CELL
  _category.description
; Data items in the CELL category record details
  about the crystallographic cell parameters.
;
  _category.id                cell
  _category.mandatory_code    no
  _category.key.name          ' _cell.entry_id'
  loop_
  _category_group.id          'inclusive_group'
                                'cell_group'
  _category_examples.detail
# -----
;
Example 1 - based on PDB entry 5HVP and laboratory
           records for the structure corresponding
           to PDB entry 5HVP
;
  _category_examples.case
;
  _cell.entry_id              '5HVP'
  _cell.length_a              58.39
  _cell.length_a_esd          0.05
  _cell.length_b              86.70
  _cell.length_b_esd          0.12
  _cell.length_c              46.27
  _cell.length_c_esd          0.06
  _cell.angle_alpha           90.00
  _cell.angle_beta            90.00
  _cell.angle_gamma           90.00
  _cell.volume                 234237
  _cell.details
; The cell parameters were refined every twenty
  frames during data integration. The cell
  lengths given are the mean of 55 such
  refinements; the esds given are the root mean
  square deviations of these 55 observations
  from that mean.
;
;
save_

```

the consistency and relational integrity of the DDL data model can be independently verified. DDL2 defines a relatively simple set of organizational elements including data blocks, categories, category groups, subcategories and items. Data dictionaries (*e.g.* mmCIF) apply these elements provided by the DDL to describe the knowledge base of an application domain. The following sections provide detailed specifications of each definitional element of DDL2.

#### 2.6.6.1. DDL2 definitions describing data items

In this section, the DDL2 categories that describe the properties of data items are presented. Figs 2.6.4.1 and 2.6.4.2 illustrate the organization of definitional elements in these categories.

##### 2.6.6.1.1. ITEM

The category named ITEM is used to assign membership of data items to categories. This category forms the bridge between the category and data-item levels of abstraction. The key data item in this category is the full data-item name, `_item.name`. This name contains both the category and data-item identifiers, and is thus a unique identifier for the data item. The category identifier, `_item.category_id`, is included in this category as a separate mandatory data item. This has been done to provide an explicit reference to those categories that use the category identifier as a unique identifier.

One could alternatively use the category and item identifiers as the basis for this category rather than the concatenated form of the item name, and thus eliminate the redundant specification of the

Example 2.6.5.4. *Related categories linked by parent-child relationships.*

```

save_CITATION
_category.description
; Data items in the CITATION category record details
about the literature cited as being relevant to
the contents of the data block.
;
_category.id                citation
_category.mandatory_code    no
_category_key.name          '_citation.id'
_loop__category_group.id    'inclusive_group'
                           'citation_group'
# ----- abbreviated definition -----
save_

save__citation.id
_item.description.description
; The value of _citation.id must uniquely identify a
record in the CITATION list. The _citation.id
'primary' should be used to indicate the citation
that the author(s) consider to be the most
pertinent to the contents of the data block.
;
_loop__item.name
      _item.category_id
      _item.mandatory_code
'_citation.id'                citation          yes
'_citation_author.citation_id' citation_author yes
'_citation_editor.citation_id' citation_editor yes
'_software.citation_id'       software          yes
_item_aliases.alias.name     '_citation_id'
_item_aliases.dictionary      cif_core.dic
_item_aliases.version         2.0.1
_loop__item_linked.child_name
      _item_linked.parent_name
'_citation_author.citation_id' '_citation.id'
'_citation_editor.citation_id' '_citation.id'
'_software.citation_id'       '_citation.id'
_item_type.code               code
_loop__item_examples.case     'primary' '1' '2'
save_

save_CITATION_AUTHOR
_category.description
; Data items in the CITATION_AUTHOR category record
details about the authors associated with the
citations in the CITATION list.
;
_category.id                citation_author
_category.mandatory_code    no
_loop__category_key.name    '_citation_author.citation_id'
                           '_citation_author.name'
_loop__category_group.id    'inclusive_group'
                           'citation_group'
# ----- abbreviated definition -----
save_

save__citation_author.citation_id
_item.description.description
; This data item is a pointer to _citation.id in the
CITATION category.
;
_item.name                   '_citation_author.citation_id'
_item.mandatory_code         yes
_item_aliases.alias.name     '_citation_author.citation_id'
_item_aliases.dictionary      cif_core.dic
_item_aliases.version         2.0.1
save_

```

category identifier. The full name has been used here in order to provide compatibility with existing applications.

The item category also includes a code to indicate whether a data item is mandatory in a category and therefore must be included in any tuple of items in the category. This code, `_item.mandatory_code`, may have three values: `yes`, `no` and `implicit`. This last named value indicates that the item is manda-

tory, but that the value of this item may be derived from the context. In the case of an item name or a category identifier, these values can be obtained from the current save-frame name. Implicit specification dramatically simplifies the appearance of each dictionary definition because it avoids the repeated declaration of item names and category identifiers that are basis components or the unique identifiers for most categories.

Although the data item `_item.name` is the basis for all of the item-level categories, its definition and properties need only be specified at a single point. Here, the data items that occur in multiple categories are defined only in the parent category. In certain situations, a child data item may be used in a manner which requires a description distinct from the parent data item. For instance, `_item_linked.parent_name` and `_item_linked.child_name` are both data-item names as well as children of `_item.name`, but clearly the manner in which these items are used in the `ITEM_LINKED` category requires additional description. It is important to note that although the design of this DDL supports the definition of data items in multiple categories within the parent category, it is also possible to provide separate complete definitions within each category.

#### 2.6.6.1.2. *ITEM\_ALIASES*

The DDL category `ITEM_ALIASES` defines the alias names that can be substituted for a data-item name. The alias mechanism also provides a means of identifying items by names other than those that follow the naming conventions used in this DDL. This feature should be used primarily to guarantee the stability of names defined in previously published dictionaries. The items `_item_aliases.name`, `_item_aliases.dictionary` and `_item_aliases.version` form the key for this category. The items `_item_aliases.dictionary` and `_item_aliases.version` are provided to distinguish between dictionaries and different versions of the same dictionary. Any number of unique alias names can be defined for a data item.

#### 2.6.6.1.3. *ITEM\_DEFAULT*

The DDL category `ITEM_DEFAULT` holds default values assigned to data items. Default data values are specified in item `_item_default.value`. Default values are assigned to data items that are not declared within a category. The key item for this category, `_item_default.name`, is a child of `_item.name`. A single default value may be specified for a data item.

#### 2.6.6.1.4. *ITEM\_DEPENDENT*

The `ITEM_DEPENDENT` category defines dependency relationships among data items within a category. Each data item on which a particular data item depends is specified as an item `_item_dependent.dependent_name`. For a data item to be considered completely defined, each of its dependent data items must also be specified.

#### 2.6.6.1.5. *ITEM\_DESCRIPTION*

The DDL category `ITEM_DESCRIPTION` holds a description for each data item. The key item for this category is `_item_description.name`, which is defined in the parent category `ITEM`. The text of the item description is held by data item `_item_description.description`. A single description may be provided for each data item.