

3. CIF DATA DEFINITION AND CLASSIFICATION

3.6.7.6. Crystal symmetry

The categories describing symmetry are as follows:

SYMMETRY group
 SYMMETRY
 SYMMETRY_EQUIV
 SPACE_GROUP
 SPACE_GROUP_SYMOP

Data items in the SYMMETRY category are used to give details about the crystallographic symmetry. The equivalent positions for the space group are listed using data items in the SYMMETRY_EQUIV category. These categories are used in the same way in the core CIF and mmCIF dictionaries, and Section 3.2.4.4 can be consulted for details.

The current version of the mmCIF dictionary includes the SPACE_GROUP categories that were derived from the symmetry CIF dictionary (Chapter 3.8) and included in version 2.3 of the core CIF dictionary. At the time of writing, macromolecular applications have not yet begun to make use of these new categories.

Data items in these categories are as follows:

(a) SYMMETRY

- *_symmetry.entry_id*
 → *_entry_id*
 _symmetry.cell_setting
 _symmetry.Int_Tables_number
 _symmetry.space_group_name_Hall
 _symmetry.space_group_name_H-M

(b) SYMMETRY_EQUIV

- *_symmetry_equiv.id* (~ *_symmetry_equiv_pos_site_id*)
 _symmetry_equiv.pos_as_xyz

(c) SPACE_GROUP

- *_space_group.id*
 _space_group.crystal_system
 _space_group.IT_number
 _space_group.name_H-M_alt
 _space_group.name_Hall

(d) SPACE_GROUP_SYMOP

- *_space_group_symop.id*
 _space_group_symop.operation_xyz
 _space_group_symop.sg_id

The bullet (•) indicates a category key. The arrow (→) is a reference to a parent data item. Items in italics have aliases in the core CIF dictionary formed by changing the full stop (.) to an underscore (_) except where indicated by the ~ symbol.

The data item *_symmetry.entry_id* has been added to the SYMMETRY category to provide the formal category key required by the DDL2 data model.

3.6.7.7. Bond-valence information

The categories describing bond valences are as follows:

VALENCE group
 VALENCE_PARAM
 VALENCE_REF

These categories were introduced into version 2.2 of the core CIF dictionary to provide the information about bond valences required in inorganic crystallography. They appear in the mmCIF dictionary only for full compatibility with the core dictionary.

Data items in these categories are as follows:

(a) VALENCE_PARAM

- *_valence_param.atom_1*
- *_valence_param.atom_1_valence*
- *_valence_param.atom_2*
- *_valence_param.atom_2_valence*
 _valence_param.B
 _valence_param.details
 _valence_param.id

- _valence_param.ref_id*
 → *_valence_ref.id*
 _valence_param.Ro

(b) VALENCE_REF

- *_valence_ref.id*
 _valence_ref.reference

The bullet (•) indicates a category key. The arrow (→) is a reference to a parent data item. Items in italics have aliases in the core CIF dictionary formed by changing the full stop (.) to an underscore (_).

Information about the use of these data items in the core CIF dictionary is given in Section 3.2.4.5.

3.6.8. Publication

The results of the determination of the crystal structure of a biological macromolecule might be published in an academic journal and/or deposited in a structural database. The data items in the core CIF dictionary cover most of the requirements for constructing an article for publication from an mmCIF and the many well defined data fields in mmCIF allow an extensively annotated record of the structure to be deposited in a database. However, the formalism of two of the core CIF categories for publication did not fit the relational database model of mmCIF, so new categories were required. The core CIF category COMPUTING, which is used to list the programs used to determine the structure, is replaced by the mmCIF category SOFTWARE, and the core CIF category DATABASE, which is used to identify the records associated with the structure in various databases, is replaced by the mmCIF category DATABASE_2.

The category groups discussed here are: the CITATION group, which is used to give citations to the literature (Section 3.6.8.1); the COMPUTING group, which is used to cite software (Section 3.6.8.2); the DATABASE group for citing related database entries (Section 3.6.8.3), which includes a group of categories used to ensure compatibility with specific database records in the Protein Data Bank (Section 3.6.8.3.2); journal administration categories that might be used by a publisher (Section 3.6.8.4.1); and the PUBL family of categories used to store the text of an article for publication (Section 3.6.8.4.2).

3.6.8.1. Literature citations

The categories describing literature citations are as follows:

CITATION group
 CITATION
 CITATION_AUTHOR
 CITATION_EDITOR

Data items in these categories are as follows:

(a) CITATION

- *_citation.id*
 _citation.abstract
 _citation.abstract_id_CAS
 _citation.book_id_ISBN
 _citation.book_publisher
 _citation.book_publisher_city
 _citation.book_title
 _citation.coordinate_linkage
 _citation.country
 _citation.database_id_CSD
 _citation.database_id_Medline
 _citation.journal_abbrev
 _citation.journal_full
 _citation.journal_id_ASTM
 _citation.journal_id_CSD
 _citation.journal_id_ISSN
 _citation.journal_issue
 _citation.journal_volume
 _citation.language
 _citation.page_first

3.6. CLASSIFICATION AND USE OF MACROMOLECULAR DATA

```

_citation.page_last
_citation.details (~ _citation_special_details)
_citation.title
_citation.year

```

(b) CITATION_AUTHOR

```

• _citation_author.citation_id
  → _citation.id
_citation_author.name
_citation_author.ordinal

```

(c) CITATION_EDITOR

```

• _citation_editor.citation_id
  → _citation.id
_citation_editor.name
_citation_editor.ordinal

```

The bullet (●) indicates a category key. The arrow (→) is a reference to a parent data item. Items in italics have aliases in the core CIF dictionary formed by changing the full stop (.) to an underscore (_).

The original core CIF dictionary contained the data item `_publ_section_references` for citations of journal articles, book chapters and monographs. The authors of the mmCIF dictionary felt that a more detailed and structured approach to literature citations was required. This is provided by the mmCIF categories `CITATION`, `CITATION_AUTHOR` and `CITATION_EDITOR`. These categories were subsequently included in the core CIF dictionary and are used in the same way in both dictionaries. Section 3.2.5.1 may be consulted for details. Although `_publ_section_references` remains a valid mmCIF data item, it is expected that the `CITATION`, `CITATION_AUTHOR` and `CITATION_EDITOR` categories will be used for literature citations in mmCIFs.

3.6.8.2. Citation of software packages

The categories describing software citations are as follows:

```

COMPUTING group
  COMPUTING
  SOFTWARE

```

It is expected that citations of software packages in an mmCIF will be made using data items in the `SOFTWARE` category. However, in some cases, a particular publisher or database may require that this information is given using data items in the `COMPUTING` category instead (see Section 3.2.5.2 for details).

Data items in these categories are as follows:

(a) COMPUTING

```

• _computing.entry_id
  → _entry.id
_creating.cell_refinement
_creating.data_collection
_creating.data_reduction
_creating.molecular_graphics
_creating.publication_material
_creating.structure_refinement
_creating.structure_solution

```

(b) SOFTWARE

```

• _software.name
  _software.version
  _software.citation_id
    → _citation.id
  _software.classification
  _software.compiler_name
  _software.compiler_version
  _software.contact_author
  _software.contact_author_email
  _software.date
  _software.dependencies
  _software.description
  _software.hardware
  _software.language
  _software.location
  _software.mods

```

```

_software.os
_software.os_version
_software.type

```

The bullet (●) indicates a category key. Where multiple items within a category are marked with a bullet, they must be taken together to form a compound key. The arrow (→) is a reference to a parent data item. Items in italics have aliases in the core CIF dictionary formed by changing the full stop (.) to an underscore (_).

The data item `_computing.entry_id` has been added to the `COMPUTING` category to provide the formal category key required by the DDL2 data model.

The data items in the `SOFTWARE` category are used to cite the software packages used in the structure analysis. The software can be described in great detail if necessary. However, for most applications a small subset of these data items, for example just `_software.name` and `_software.version`, could be used (see Example 3.6.8.1).

Most data items in the `SOFTWARE` category are self-explanatory, but a few require further comment. The data item `_software.citation_id` provides a way to link the details of a program to the citation of an article in the literature that describes the program; this data item must match a value of `_citation.id` in the `CITATION` category. The name and e-mail address of the author of the software can also be given using `_software.contact_author` and `_software.contact_author_email`, respectively. (This may be the original author or someone who subsequently modifies or maintains the software; these data items would generally refer to the person most closely associated with the maintenance of the code at the time it was used.) The release date of the software may be recorded in `_software.date`. As far as possible, the date should be that of the version recorded in `_software.version`. The data item `_software.location` may be used to supply a URL from which the software may be downloaded or where it is described in detail.

3.6.8.3. Citation of related database entries

Categories describing related database entries are as follows:

```

DATABASE group
  Related database entries (§3.6.8.3.1)
  DATABASE
  DATABASE_2

```

Example 3.6.8.1. The refinement program *Prolsq* described with data items in the `SOFTWARE` category.

```

_software.name
_software.version
_software.date
_software.type
_software.contact_author
_software.contact_author_email
_software.location
_software.classification
_software.citation_id
_software.language
_software.compiler_name
_software.compiler_version
_software.hardware
_software.os
_software.os_version
_software.dependencies
_software.mods
_software.description
  Prolsq unknown . program
  'Wayne A. Hendrickson' ?
  'ftp://rosebud.sdsc.edu/pub/sdsc/xtal/CCP4/ccp4/'
  refinement ref5 Fortran
  'Convex Fortran' v8.0 'Convex C220' ConvexOS v10.1
  'Requires that Protin be run first' optimized
  'restrained least-squares refinement'

```

3. CIF DATA DEFINITION AND CLASSIFICATION

Compatibility with PDB format files (§3.6.8.3.2)

DATABASE_PDB_CAVEAT
DATABASE_PDB_MATRIX
DATABASE_PDB_REMARK
DATABASE_PDB_REV
DATABASE_PDB_REV_RECORD
DATABASE_PDB_TVECT

The purpose of entries in the DATABASE category group is to provide pointers that link the mmCIF to all database entries that result from the deposition of the file. For mmCIF, the relevant category is DATABASE_2, which replaces the DATABASE category of the core dictionary.

Note the distinction between the database pointers provided here and those in the STRUCT_REF family of categories. The latter are intended to provide links to external database entries for any aspect of any subset of the structure that the author may wish to record, including previous determinations of the same structure, other structures containing the same ligand or references to the sequence(s) of the macromolecule(s) in sequence databases. In contrast, the links provided in DATABASE_2 refer to the entire contents of the mmCIF and are designed to cover situations in which the entire file is deposited in more than one database (for example, in the PDB and in a database for protein kinases).

3.6.8.3.1. Related database entries

Data items in these categories are as follows:

(a) DATABASE

- `_database.entry_id`
→ `_entry.id`
`_database.code_CAS`
`_database.code_CSD`
`_database.code_ICSD`
`_database.code_MDF`
`_database.code_NBS`
`_database.code_PDB`
`_database.code_PDF`
`_database.code_depnum_ccdc_archive`
`_database.code_depnum_ccdc_fiz`
`_database.code_depnum_ccdc_journal`
`_database.CSD_history`
`_database.journal_ASTM`
`_database.journal_CSD`

(b) DATABASE_2

- `_database_2.database_id`
- `_database_2.database_code`

The bullet (•) indicates a category key. Where multiple items within a category are marked with a bullet, they must be taken together to form a compound key. The arrow (→) is a reference to a parent data item. Items in italics have aliases in the core CIF dictionary formed by changing the full stop (.) to an underscore (_).

The DATABASE category is retained in the mmCIF dictionary, but only for consistency with the core dictionary.

The role of the data items in the DATABASE_2 category is to store identifiers assigned by one or more databases to the structure described in the mmCIF. In the data model used in the core CIF dictionary, each database has an individual data item. The data model in mmCIF is more general. It comprises the data items `_database_2.database_id`, which identifies the database, and `_database_2.database_code`, which is the code assigned by the database to the entry. Thus a new database can be referred to without needing to add an additional data item to the dictionary. If a structure has been deposited in more than one database, the values of `_database_2.database_id` and `_database_2.database_code` can be looped.

The institutions and individual databases recognized in the DATABASE_2 category in the current version of the mmCIF dictionary are CAS (Chemical Abstracts Service), CSD (Cam-

bridge Structural Database), ICSD (Inorganic Crystal Structure Database), MDF (Metals Data File), NDB (Nucleic Acid Database), NBS (the Crystal Data database of the National Institute of Standards and Technology, formerly the National Bureau of Standards), PDB (Protein Data Bank), PDF (Powder Diffraction File), RCSB (Research Collaboratory for Structural Bioinformatics) and EBI (European Bioinformatics Institute). It is intended that new databases will be added to this list on an ongoing basis; the purpose of specifying a list of possible databases in the dictionary is to ensure that each database is referenced consistently.

3.6.8.3.2. Compatibility with PDB format files

Data items in these categories are as follows:

(a) DATABASE_PDB_REV

- `_database_PDB_rev.num`
`_database_PDB_rev.author_name`
`_database_PDB_rev.date`
`_database_PDB_rev.date_original`
`_database_PDB_rev.mod_type`
`_database_PDB_rev.replaced_by`
`_database_PDB_rev.replaces`
`_database_PDB_rev.status`

(b) DATABASE_PDB_REV_RECORD

- `_database_PDB_rev_record.rev_num`
→ `_database_PDB_rev.num`
- `_database_PDB_rev_record.type`
`_database_PDB_rev_record.details`

(c) DATABASE_PDB_MATRIX

- `_database_PDB_matrix.entry_id`
→ `_entry.id`
`_database_PDB_matrix.origx[1][1]`
`_database_PDB_matrix.origx[1][2]`
`_database_PDB_matrix.origx[1][3]`
`_database_PDB_matrix.origx[2][1]`
`_database_PDB_matrix.origx[2][2]`
`_database_PDB_matrix.origx[2][3]`
`_database_PDB_matrix.origx[3][1]`
`_database_PDB_matrix.origx[3][2]`
`_database_PDB_matrix.origx[3][3]`
`_database_PDB_matrix.origx_vector[1]`
`_database_PDB_matrix.origx_vector[2]`
`_database_PDB_matrix.origx_vector[3]`
`_database_PDB_matrix.scale[1][1]`
`_database_PDB_matrix.scale[1][2]`
`_database_PDB_matrix.scale[1][3]`
`_database_PDB_matrix.scale[2][1]`
`_database_PDB_matrix.scale[2][2]`
`_database_PDB_matrix.scale[2][3]`
`_database_PDB_matrix.scale[3][1]`
`_database_PDB_matrix.scale[3][2]`
`_database_PDB_matrix.scale[3][3]`
`_database_PDB_matrix.scale_vector[1]`
`_database_PDB_matrix.scale_vector[2]`
`_database_PDB_matrix.scale_vector[3]`

(d) DATABASE_PDB_TVECT

- `_database_PDB_tvect.id`
`_database_PDB_tvect.details`
`_database_PDB_tvect.vector[1]`
`_database_PDB_tvect.vector[2]`
`_database_PDB_tvect.vector[3]`

(e) DATABASE_PDB_CAVEAT

- `_database_PDB_caveat.id`
`_database_PDB_caveat.text`

(f) DATABASE_PDB_REMARK

- `_database_PDB_remark.id`
`_database_PDB_remark.text`

The bullet (•) indicates a category key. Where multiple items within a category are marked with a bullet, they must be taken together to form a compound key. The arrow (→) is a reference to a parent data item.

3.6. CLASSIFICATION AND USE OF MACROMOLECULAR DATA

A major goal of the design of the mmCIF data model was that a file could be transformed from Protein Data Bank (PDB) format to mmCIF format and back again without loss of information. This required the creation of mmCIF data items whose sole purpose is to capture PDB-specific records that do not map onto mmCIF data items. These records would never be created for a *de novo* mmCIF. This family of categories also belongs to the PDB category group (see Section 3.6.9.3).

The items in the categories `DATABASE_PDB_MATRIX` and `DATABASE_PDB_TVECT` are derived from the elements of transformation matrices and vectors used by the Protein Data Bank. The items in the categories `DATABASE_PDB_REV` and `DATABASE_PDB_REV_RECORD` record details about the revision history of the data block as archived by the Protein Data Bank.

The items in the `DATABASE_PDB_CAVEAT` category record comments about the data block flagged as 'CAVEATS' by the Protein Data Bank at the time the original PDB archive file was created. A PDB CAVEAT record indicates that the entry contains severe errors. In PDB format, extended comments were stored as a sequence of fixed-length (80-character) format records, columns 9 and 10 being reserved for continuation sequence numbering. The mmCIF representation retains each record as a separate data value and does not attempt to merge continuation records to provide more readable running text. Hence the PDB CAVEAT entry

```
CAVEAT 1ABC THE CRYSTAL TRANSFORMATION IS WRONG
CAVEAT 2 1ABC BUT IS UNCORRECTABLE AT THIS TIME
```

would be represented in mmCIF as

```
loop_
  _database_PDB_caveat.id
  _database_PDB_caveat.text
  1
; THE CRYSTAL TRANSFORMATION IS WRONG
;
  2
; BUT IS UNCORRECTABLE AT THIS TIME
;
```

The PDB format used 'REMARK' records to store information relating to several aspects of the structure in free or loosely structured text. In some cases, the conventions used for individual types of REMARK record allow structured data to be extracted automatically and translated to specific mmCIF data items. Where this is not possible, the `DATABASE_PDB_REMARK` category may be used to retain the information that appeared in these parts of PDB format files. Unlike the CAVEAT records, it is possible to collect together several REMARK records sharing a common numbering into a single free-text field. For example, PDB practice has been to repeat the contents of CAVEAT records (see above) as records of type 'REMARK 5'. While each separate CAVEAT record is converted to a separate mmCIF data value, the complete text of a REMARK 5 record may be gathered into a single mmCIF data value. Hence the CAVEAT example above would also appear in a PDB file as part of a 'REMARK 5' as

```
REMARK 5 THE CRYSTAL TRANSFORMATION IS WRONG
REMARK 5 BUT IS UNCORRECTABLE AT THIS TIME
```

and would appear in an mmCIF as

```
loop_
  _database_PDB_remark.id
  _database_PDB_remark.text
  5
; THE CRYSTAL TRANSFORMATION IS WRONG
; BUT IS UNCORRECTABLE AT THIS TIME
;
```

Note that by convention the value of `_database_PDB_remark.id` matches the class of the REMARK record in the PDB file.

3.6.8.4. Article publication

Categories used during the publication of an article are as follows:

```
IUCR group
  Journal housekeeping and reference entries (§3.6.8.4.1)
  JOURNAL
  JOURNAL_INDEX
  Contents of a publication (§3.6.8.4.2)
  PUBL
  PUBL_AUTHOR
  PUBL_BODY
  PUBL_MANUSCRIPT_INCL
```

These categories cover both the metadata for the article (information about the article) and the text of the article itself.

3.6.8.4.1. Journal housekeeping and citation entries

Data items in these categories are as follows:

- (a) JOURNAL
- `_journal.entry_id`
 - `_entry.id`
 - `_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.data_validation_number`
 - `_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.language`
 - `_journal.name_full`
 - `_journal.page_first`
 - `_journal.page_last`
 - `_journal.paper_category`
 - `_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`
- (b) JOURNAL_INDEX
- `_journal_index.subterm`
 - `_journal_index.term`
 - `_journal_index.type`

The bullet (•) indicates a category key. The arrow (→) is a reference to a parent data item. Items in italics have aliases in the core CIF dictionary formed by changing the full stop (.) to an underscore (_).

In mmCIF, the families of categories used to contain the text of an article for publication and to record information about the handling and processing of the article by a publisher are assigned to the IUCR category group. The name arose from the fact that CIF is sponsored by the International Union of Crystallography and several of the journals of the IUCr can handle articles submitted for publication in CIF format. However, these data items may be

3. CIF DATA DEFINITION AND CLASSIFICATION

freely used by other publishers who wish to handle articles submitted in CIF format. The JOURNAL and JOURNAL_INDEX categories are used in the same way in the core CIF and mmCIF dictionaries, and Section 3.2.5.4 can be consulted for details.

3.6.8.4.2. Contents of a publication

Data items in these categories are as follows:

(a) PUBL

- `_publ.entry_id`
→ `_entry.id`
- `_publ.contact_author`
- `_publ.contact_author_address`
- `_publ.contact_author_email`
- `_publ.contact_author_fax`
- `_publ.contact_author_name`
- `_publ.contact_author_phone`
- `_publ.contact_letter`
- `_publ.manuscript_creation`
- `_publ.manuscript_processed`
- `_publ.manuscript_text`
- `_publ.requested_category`
- `_publ.requested_coeditor_name`
- `_publ.requested_journal`
- `_publ.section_abstract`
- `_publ.section_acknowledgements`
- `_publ.section_comment`
- `_publ.section_discussion`
- `_publ.section_experimental`
- `_publ.section_exptl_prep`
- `_publ.section_exptl_refinement`
- `_publ.section_exptl_solution`
- `_publ.section_figure_captions`
- `_publ.section_introduction`
- `_publ.section_references`
- `_publ.section_synopsis`
- `_publ.section_table_legends`
- `_publ.section_title`
- `_publ.section_title_footnote`

(b) PUBL_AUTHOR

- `_publ.author.address`
- `_publ.author.email`
- `_publ.author.footnote`
- `_publ.author.id_iucr`
- `_publ.author.name`

(c) PUBL_BODY

- `_publ.body.contents`
- `_publ.body.element`
- `_publ.body.format`
- `_publ.body.label`
- `_publ.body.title`

(d) PUBL_MANUSCRIPT_INCL

- `_publ.manuscript_incl.entry_id`
→ `_entry.id`
- `_publ.manuscript_incl.extra_defn`
- `_publ.manuscript_incl.extra_info`
- `_publ.manuscript_incl.extra_item`

The bullet (•) indicates a category key. The arrow (→) is a reference to a parent data item. Items in italics have aliases in the core CIF dictionary formed by changing the full stop (.) to an underscore (_).

The categories PUBL, PUBL_AUTHOR, PUBL_BODY and PUBL_MANUSCRIPT_INCL are also members of the IUCR group in the mmCIF dictionary. They are used in the same way in the core CIF and mmCIF dictionaries, and Section 3.2.5.5 can be consulted for details.

3.6.9. File metadata

As in the core CIF dictionary, information about the source and the revision history of an mmCIF may be given in the AUDIT group

of categories: AUDIT, AUDIT_AUTHOR, AUDIT_CONTACT_AUTHOR and AUDIT_CONFORM (Section 3.6.9.1). However, the mmCIF dictionary differs from the core CIF dictionary in the way it expresses relationships between data blocks: instead of the core AUDIT_LINK category, mmCIF has two categories, ENTRY and ENTRY_LINK, that essentially fulfil the same role but are classified in a distinct category group (Section 3.6.9.2).

3.6.9.1. History of a data block

The categories describing the history of a data block are as follows:

- AUDIT group
 - AUDIT
 - AUDIT_AUTHOR
 - AUDIT_CONFORM
 - AUDIT_CONTACT_AUTHOR

Data items in these categories are as follows:

(a) AUDIT

- `_audit.revision_id`
- `_audit.creation_date`
- `_audit.creation_method`
- `_audit.update_record`

(b) AUDIT_AUTHOR

- `_audit.author.name`
- `_audit.author.address`

(c) AUDIT_CONFORM

- `_audit.conform.dict_name`
- `_audit.conform.dict_version`
- `_audit.conform.dict_location`

(d) AUDIT_CONTACT_AUTHOR

- `_audit.contact_author.name`
- `_audit.contact_author.address`
- `_audit.contact_author.email`
- `_audit.contact_author.fax`
- `_audit.contact_author.phone`

The bullet (•) indicates a category key. Where multiple items within a category are marked with a bullet, they must be taken together to form a compound key. Items in italics have aliases in the core CIF dictionary formed by changing the full stop (.) to an underscore (_).

The data items in these categories are used in the same way in the mmCIF dictionary as in the core CIF dictionary (see Section 3.2.6). The data item `_audit.revision_id` has been added to the AUDIT category to provide the formal category key required by the DDL2 data model. The core data item `_audit_block_code` has been replaced by `_entry.id` (see Section 3.6.9.2).

3.6.9.2. Links between data blocks

The categories describing links between data blocks are as follows:

- ENTRY group
 - ENTRY
 - ENTRY_LINK
- AUDIT group
 - AUDIT_LINK

Data items in these categories are as follows:

(a) ENTRY

- `_entry.id`

(b) ENTRY_LINK

- `_entry_link.entry_id`
→ `_entry.id`
- `_entry_link.id`
- `_entry_link.details`