

2.6. SPECIFICATION OF A RELATIONAL DICTIONARY DEFINITION LANGUAGE (DDL)

2.6.6.1.6. *ITEM_ENUMERATION*

The DDL category *ITEM_ENUMERATION* holds lists of permissible values for a data item. Each enumerated value is specified in item *_item_enumeration.value*, each of which may have an associated description item *_item_enumeration.detail*. The combination of items *_item_enumeration.name* and *_item_enumeration.value* form the key for this category. The parent definition of the former item is defined in the category *ITEM*. Multiple unique enumeration values may be specified for each data item.

2.6.6.1.7. *ITEM_EXAMPLES*

The DDL category *ITEM_EXAMPLES* is provided to hold examples associated with individual data items. An example specification consists of the text of the example, *_item_examples.case*, and an optional comment item, *_item_examples.detail*, which can be used to qualify the example. Multiple examples may be provided for each item.

2.6.6.1.8. *ITEM_LINKED*

The *ITEM_LINKED* category defines parent–child relationships between data items. This provides the mechanism for specifying the relationships between data items that may exist in multiple categories. Link relationships are most commonly defined between key items, which form the keys for many different categories.

In the DDL definition, all child relationships are expressed within the parent category.

Because the item *_item_linked.parent_name* has been defined as an implicit item, the child relationships can be specified most economically in the parent category where the parent item name can be automatically inferred. If link relationships are specified in a child category, then both parent and child item names must be specified.

Both parent and child item names in this category are children of *_item.name*, which ensures that all link relationships can be properly resolved. However, it is possible to define cyclical link relationships within this category. Any implementation of this DDL category should include a method to check for the existence of such pathological cases.

2.6.6.1.9. *ITEM_METHODS*

The *ITEM_METHODS* category is used to associate method identifiers with data items. Any number of unique method identifiers may be associated with a data item. The method identifiers reference the full method definitions in the parent *METHOD_LIST* category.

2.6.6.1.10. *ITEM_RANGE*

The *ITEM_RANGE* category defines a restricted range of permissible values for a data item. The restrictions are specified as one or more sets of the items *_item_range.minimum* and *_item_range.maximum*. These items give the lower and upper bounds for a permissible range. To specify that an item value may be equal to the upper or lower bound or a range, the minimum and maximum values of the range are equated. The special STAR value indicating that a data value is not appropriate (denoted by a period, ‘.’) can be used to avoid expressing an upper or lower bound value. When limits are applied to character data, comparisons are made following the collating sequence of the character set. When limits are applied to abstract data types, methods must be provided to

define any comparison operations that must be performed to check the boundary conditions.

2.6.6.1.11. *ITEM_RELATED*

The *ITEM_RELATED* category describes specific relationships that exist between data items. These relationships are distinct from the parent–child relationships that are expressed in the category. The related item is identified as the item *_item_related.related_name* that is a child of *_item.name*.

Item relationships defined by *_item_related.function_code* in this category include some of the following (Table 2.6.5.1): an item is related to another item by a conversion factor; an item is a replacement for another item; an item is replaced by another item; an item is an alternative expression of an item; items which differ only in some convention of their expression; and items which express a set of related characteristics. One can also identify whether the declaration of an item is mutually exclusive with its alternative item. Multiple related items can be associated with each data item and multiple relationship codes can be specified for each related item.

2.6.6.1.12. *ITEM_STRUCTURE*

The *ITEM_STRUCTURE* category holds a code which identifies a structure definition that is associated with a data item. A structure in this context is a reusable matrix or vector definition declared in category *ITEM_STRUCTURE_LIST*. The data item *_item_structure.code* is a child of the item *_item_structure_list.code*. The item *_item_structure.code* provides an indirect reference into the list of structure-type definitions in category *ITEM_STRUCTURE_LIST*. The *_item_structure.organization* item describes the row/column precedence of the matrix organization.

2.6.6.1.13. *ITEM_STRUCTURE_LIST*

The *ITEM_STRUCTURE_LIST* category holds definitions of matrices and vectors that can be associated with data items. A component of the key for this category is *_item_type_list.code*, which is referenced by *_item_structure.code* to assign a structure type to a data item. The definition of a structure involves the specification of a length for each dimension of the matrix structure. The combination of items *_item_structure_list.code* and *_item_structure_list.index* forms the key for this category. The latter index item is the identifier for the dimension, hence multiple unique dimensions can be specified for each structure code. The length of each dimension is assigned to *_item_structure_list.dimension*.

2.6.6.1.14. *ITEM_SUB_CATEGORY*

The *ITEM_SUB_CATEGORY* category is used to assign subcategory membership for data items. A data item may belong to any number of subcategories. Each subcategory must be defined in a category named *SUB_CATEGORY*.

2.6.6.1.15. *ITEM_TYPE*

The *ITEM_TYPE* category holds a code that identifies the data type of each data item. The data item *_item_type.code* is a child of the item *_item_type_list.code*. Data-type definitions are actually made in the *ITEM_TYPE_LIST* parent category. The item *_item_type.code* provides an indirect reference into the list of data-type definitions in category *ITEM_TYPE_LIST*. This indirect