2.5. Specification of the core CIF dictionary definition language (DDL1)

BY S. R. HALL AND A. P. F. COOK

2.5.1. Introduction

The CIF approach to data representation described in Chapter 2.2 is based on the STAR File universal data language (Hall, 1991; Hall & Spadaccini, 1994) detailed in Chapter 2.1. An important advantage of the CIF approach is the self-identification of data items through the use of tag–value tuples. This syntax removes the need for preordained data ordering in a file or stream of data and enables appropriate parsing tools to automate access independently of the data source. In this chapter, we will show that the CIF syntax also provides a higher level of abstraction for managing data storage and exchange – that of defining the meaning of data items (*i.e.* their properties and characteristics) as attribute descriptors linked to the identifying data tag.

Each attribute descriptor specifies a particular characteristic of a data item, and a collection of these attributes can be used to provide a unique definition of the data item. Moreover, placing the definitions of a selected set of data items into a CIF-like file provides an electronic dictionary for a particular subject area. In the modern parlance of knowledge management and the semantic web, such a data dictionary represents a *domain ontology*.

In most respects, data dictionaries serve a role similar to spoken-word dictionaries and as such are an important adjunct to the CIF data-management approach by providing semantic information that is necessary for automatic validation and compliance procedures. That is, prior lexical knowledge of the nature of individual items ensures that each item in a CIF can be read and interpreted correctly *via* the unique tag that is the link to descriptions in a data dictionary file. Because the descriptions in the dictionary are machine-parsable, the semantic information they contain forms an integral part of a data-handling process. In other words, machine-interpretable semantic knowledge embedded in data dictionaries leads directly to the automatic validation and manipulation of the relevant items stored in any CIF.

2.5.1.1. The concept of a dictionary definition language (DDL)

The structure or arrangement of data in a CIF is well understood and predictable because the CIF syntax may be specified succinctly (see Chapter 2.2 for CIF syntax expressed using extended Backus–Naur form). In contrast, the meaning of individual data values in a file is only known if the nature of these items is understood. For CIF data this critical link between the value and the meaning of an item is achieved using an electronic 'data dictionary' in which the definitions of relevant data items are catalogued according to their data name and expressed as attribute values, one set of attributes per item.

The dictionary definitions describe the characteristics of each item, such as data type, enumeration states and relationships between data. The more precise the definitions, the higher the level of semantic knowledge of defined items and the better the efficiency achievable in their exchange. To a large degree, the precision achievable hinges on the attributes selected for use in dictionary definitions, these being the semantic vocabulary of a dictionary. Within this context, attribute descriptors constitute a dictionary definition language (DDL). Definitions of the attributes described in this chapter are provided in Chapter 4.9.

The main purpose of this chapter is to describe the DDL attributes used to construct the core, powder, modulated structures and electron density CIF dictionaries detailed in Chapters 3.2–3.5 and 4.1–4.4. We shall see that each item definition in these dictionaries is constructed separately by appropriate choice from the attribute descriptors available, and that a sequence of definition blocks (one block per item) constitutes a CIF dictionary file. The organization of attributes and definition blocks in a dictionary file need not be related to the syntax of a data CIF, but in practice there are significant advantages if they are. Firstly, a common syntax for data and dictionary files enables the same software to be used to parse both. Secondly, and of equal importance, data descriptions and dictionaries need continual updating and additions, and the CIF syntax provides a high level of extensibility and flexibility, whereas most other formats do not. Finally, the use of a consistent syntax permits the dictionary attribute descriptors themselves to be described in their own DDL dictionary file.

While the basic functionality and flexibility of a dictionary is governed by the CIF syntax, the precision of the data definitions contained within it is determined entirely by the scope and number of the attribute descriptors representing the DDL. Indeed, the ability of a DDL to permit the simple and seamless evolution of data definitions and the scope of the DDL to precisely define data items both play absolutely pivotal roles *via* the supporting dictionaries in determining the power of the CIF data-exchange process.

2.5.2. The organization of a CIF dictionary

The precision and efficiency of a data definition language are directly related to the scope of the attribute vocabulary. In other words, the lexical richness of the DDL depends on the number and the specificity of the available language attributes. The breadth of these attributes, in terms of the number of separate data characteristics that can be specified, largely controls the precision of data definitions. However, it is the functionality of attributes that determines the information richness and enables higher-level definition complexity. For example, the attributes that define child–parent relationships between data and key pointers to items in list packets are essential to understanding the data hierarchy and to its validation. Attributes provide the semantic tools of a dictionary.

The choice and scope of attributes in the DDL are governed by both semantic and technical considerations. Attributes need to have a clear purpose to facilitate easy definition and comprehension, and their routine application in automatic validation processes. A CIF dictionary is much more than a list of unrelated data definitions. Each definition conforms to the CIF syntax, which requires each data block in the dictionary to be unique. However, the functionality of a dictionary involves elements of both relational and object-oriented processes. For example, attributes in one definition may refer to another definition *via* `_list_link_parent` or `_list_link_child` attributes, so as to indicate the dependency

Affiliations: SYDNEY R. HALL, School of Biomedical and Chemical Sciences, University of Western Australia, Crawley, Perth, WA 6009, Australia; ANTHONY P. F. COOK, BCI Ltd, 46 Uppergate Road, Stannington, Sheffield S6 6BX, England.

references