

## 2.2. SPECIFICATION OF THE CRYSTALLOGRAPHIC INFORMATION FILE (CIF)

(19) Matching square-bracket characters, '[' and ']', are reserved for possible future introduction as delimiters of multi-line data values. At this revision of the CIF specification, a data value may not begin with an unquoted left square-bracket character '['. (While not strictly necessary, the right square-bracket character ']' is restricted in the same way in recognition of its reserved use as a closing delimiter.)

(20) *Comment:* For example, the data value `foo` may be expressed equivalently as an unquoted string `foo`, as a quoted string `'foo'` or as a text field

```
;foo
;
```

By contrast, the value of the text field

```
;foo
  bar
;
```

is

```
foo<eol> bar
```

(where `<eol>` represents an end of line); the embedded space characters are significant.

(21) A comment in a CIF begins with an unquoted character '#' and extends to the end of the current line.

## 2.2.7.1.5. Character set

(22) Characters within a CIF are restricted to certain printable or white-space characters. Specifically, these are the ones located in the ASCII character set at decimal positions 09 (HT or horizontal tab), 10 (LF or line feed), 13 (CR or carriage return) and the letters, numerals and punctuation marks at positions 32–126.

*Comment:* The ASCII characters at decimal positions 11 (VT or vertical tab) and 12 (FF or form feed), often included in library implementations as white-space characters, are explicitly excluded from the CIF character set at this revision.

(23) *Comment:* The reference to the ASCII character set is specifically to identify characters in an established and widely available standard. It is understood that CIFs may be constructed and maintained on computer platforms that implement other character-set encodings. However, for maximum portability only the characters identified in the section above may be used. Other printable characters, even if available in an accessible character set such as Unicode, must be indicated by some encoding mechanism using only the permitted characters. At this revision, only the encoding convention detailed in Section 2.2.7.4(30)–(37) is recognized for this purpose.

## 2.2.7.1.6. White space

(24) Any of the white-space characters listed in paragraph (22) (*i.e.* HT, LF, CR) and the visible space character SP (position number 32 in the ASCII encoding) may be used interchangeably to separate tokens, with the exception that the semicolon characters delimiting multi-line text fields must be preceded by the white-space character or characters understood as indicating an end of line (see next paragraph).

## 2.2.7.1.7. End-of-line conventions

(25) The way in which a line is terminated is operating-system dependent. The STAR File specification does not address different operating-system conventions for encoding the end of a line of text in a text file. For a file generated and read in the same machine environment, this is rarely a problem, but increasingly applications on a network host may access files on different hosts through

protocols designed to present a unified view of a file system. In practice, for current common operating systems many applications may regard the ASCII characters LF or CR or the sequence CR LF as signalling an end of line, inasmuch as these represent the end-of-line conventions supported under the common operating systems Unix, MacOS or DOS/Windows. On platforms with record-oriented operating systems, applications must understand and implement the appropriate end-of-line convention. Care must be taken when transferring such files to other operating systems to insert the appropriate end-of-line characters for the target operating system. A more complete discussion is given in (42) below.

## 2.2.7.1.8. Case sensitivity

(26) Data names, block and frame codes, and reserved words are case-insensitive. The case of any characters within data values must be respected.

## 2.2.7.1.9. Implementation restrictions

(27) Certain allowed features of STAR File syntax have been expressly excluded or restricted from the CIF implementation.

## 2.2.7.1.9.1. Maximum line length and character set

(28) Lines of text may not exceed 2048 characters in length. This count excludes the character or characters used by the operating system to mark the line termination.

The ASCII characters decimal 11 (VT) and 12 (FF) are excluded from the allowed character set [see paragraph (22)].

## 2.2.7.1.9.2. Maximum data-name, block-code and frame-code lengths

(29) Data names may not exceed 75 characters in length.

(30) Data-block codes and save-frame codes may not exceed 75 characters in length (and therefore data-block headers and save-frame headers may not exceed 80 characters in length).

## 2.2.7.1.9.3. Single-level loop constructs

(31) Only a single level of looping is permitted.

## 2.2.7.1.9.4. Non-expansion of save-frame references

(32) Save frames are permitted in CIFs, but expressly for the purpose of encapsulating data-name definitions within data dictionaries. No reference to these save frames is envisaged, and the save-frame reference code permitted in STAR is not used. This means that unquoted character strings commencing with the \$ character may not be interpreted as save-frame codes in CIF. Use of such unquoted character strings is reserved to guard against subsequent relaxation of this constraint.

## 2.2.7.1.9.5. Exclusion of global\_ blocks

(33) In the full STAR specification, blocks of data headed by the special case-insensitive word `global_` are permitted before normal data blocks. They contain data names and associated values which are inherited in subsequent data blocks; the scope of a value extends from its point of declaration in a global block to the end of the file. Because rearrangements of the order of data blocks and concatenation of data blocks from different files are commonplace operations in many CIF applications, and because of the difficulty in properly tracking and implementing values implied by global blocks, use of the `global_` feature of STAR is expressly forbidden at this revision. To guard against its future introduction, the special case-insensitive word `global_` remains reserved in CIF.