

2.2. SPECIFICATION OF THE CRYSTALLOGRAPHIC INFORMATION FILE (CIF)

to allow for recognition of the fact that no terminal line termination is intended – see below.)

In order to understand this scheme, suppose the CIF fragment (1) below were considered to have long lines. They could be transformed into (2) as follows:

(1) *Initial CIF*

```
#####
### CIF submission form for Rietveld refinements
###
###           Version 14 December 1998
###
#####
data_znvddata
_chemical_name_systematic
; zinc dihydroxide divanadate dihydrate
;
_chemical_formula_moiety      'H2 O9 V2 Zn3, 2(H2 O)'
_chemical_formula_sum         'H6 O11 V2 Zn3'
_chemical_formula_weight      480.05
```

(2) *Transformed CIF*

```
#\
#####\
#####
### CIF submission form for Rietveld refinements
###
###           Version 14 December 1998
###
#####
data_znvddata
_chemical_name_systematic
;\
zinc dihydroxide divan\
adate dihydrate
;
_chemical_formula_moiety
;\
H2 O9 V2 Zn3, 2(H2 O)\
;
_chemical_formula_sum      'H6 O11 V2 Zn3'
_chemical_formula_weight   480.05
```

In making the transformation from the backslash-folded form to long lines, it is very important to strip trailing blanks before attempting to recognize a backslash as the last character. When reassembling text-field lines, no reassembly should be done except in text fields that begin with the special sequence described above, line termination–semicolon–backslash–line termination, (<eol>;\<eol>), so that text fields that happen to contain backslashes but which were not created by folding long lines are not changed. It is also important to remove the trailing backslashes when reassembling long lines. The final line termination–semicolon sequence of a text field takes priority over the reassembly process and ends it, but a trailing backslash on the last line of a text field very nicely conveys the information that no trailing line termination is intended to be included within the character string.

Similarly, when reassembling long-line comments, the reassembly begins with a comment of the form hash–backslash–line termination. The initial hash mark is retained and then a forward scan is made through line terminations and blanks for the next comment, from which the initial hash mark is stripped and then the contents of the comment are appended. If that comment ends with a backslash, the trailing backslash is stripped and the process repeats. Note that the process will be ended by intervening tags, values, data blocks or other non-white-space information, and that the process will not start at all without the special hash–backslash–line termination comment.

Since there are very few, if any, CIFs that contain text fields and comments beginning this way, in most cases it is reasonable to adopt the policy of doing this processing unless it is disabled.

Here is another example of folding. The following three text fields would be equivalent:

```
;C:\foldername\filename
;
;\
C:\foldername\filename
;
and
;\
C:\foldername\file\
name
;
```

but the following example would be a two-line value where the first line had the value C:\foldername\file\ and the second had the value name:

```
;
C:\foldername\file\
name
;
```

Note that backslashes should not be used to fold lines outside of comments and text fields. That would introduce extraneous characters into the CIF and violate the basic syntax rules. In any case, such action is not necessary.

2.2.7.4.12. *Dictionary compliance*

(27) Dictionary files containing the definitions and attribute sets for the data items contained in a CIF should be identified within the CIF by some or all of the data items

```
_audit_conform_dict_name
_audit_conform_dict_version
_audit_conform_dict_location
```

corresponding to DDL1 dictionaries or

```
_audit_conform.dict_name
_audit_conform.dict_version
_audit_conform.dict_location
```

for DDL2 dictionaries. Where no such information is provided, it may be assumed that the file should conform against the core CIF dictionary.

(28) The `_audit_conform` data items may be looped in cases where more than one dictionary is used to define the items in a CIF and they may include dictionaries of local data items provided such dictionary files have been prepared in accordance with the rules of the appropriate DDL.

(29) A detailed protocol exists for locating, merging and overlaying multiple dictionary files (McMahon *et al.*, 2000) (see Section 3.1.9).

2.2.7.4.13. *CIF markup conventions*

(30) If permitted by the relevant dictionary and if no other indication is present, the contents of a text or character field are assumed to be interpretable as text in English or some other human language. Certain special codes are used to indicate special characters or accented letters not available in the ASCII character set, as listed below.