TOOLS FOR *DDLm* DICTIONARY MAINTENANCE

DRAFT GUIDE: 17 APRIL 2013

All queries, comments and corrections on this Guide and distributed software should be sent to Syd Hall, U of Western Australia, Crawley, 6009; sydney.hall@uwa.edu.au.

Four programs, written in Fortran, are intended to support the development and maintenance of STAR[1] data dictionaries written in the dictionary language *DDLm*[2,3].

The programs are:

TAGxref Records occurrences of STAR data names (i.e. tags) in a text file and cross-references them with primary and aliased tags defined in a dictionary (written in *DDLm*). The text file can also be a dictionary. IMPORT statements in the text file or dictionary are executed and expanded versions of these files are generated.

ALIGN Ensures compliance of STAR file data items with their attribute definitions in a dictionary (written in *DDLm*). The STAR file can also be a dictionary.

DICtree Maps the hierarchical relationships between the defined data categories and data names in a dictionary (written in *DDLm*).

DICalias Lists the defined data names with their aliases in a dictionary (written in *DDLm*).

TAGxref

TAGxref is a data tag cross-referencing tool ensuring data tags in a text file. It is especially useful for *DDLm* dictionary writers and maintainers.

It is used to record tag occurrences, by line number, in a text file, and cross-references them with the *primary* and *aliased tags* defined in a dictionary (written in *DDLm*). It identifies any undefined tags. The checked file may be *text* manuscript (i.e. with extension '`.txt`', not a '`.doc`' file) or another dictionary (i.e. a *text* file with extension '`.dic`').

An important by-product of the TAGxref checking process is that the files used to cross-reference the tags are automatically *populated* with imported data. This *import-expanded dictionary* is output with the filename extension '`.xrodic`'. If the checked text file is also a dictionary, it is output with extension '`.xrotex`'.

TAGxref Control files `XTAGS` and `XREF`

TAGxref uses two auxillary input files to control the cross-referencing process.

`XTAGS` is a text file containing those tags that will be *excluded* from the cross-referencing process. The `XTAGS` file distributed contains all of the DDLm attribute tags - this avoids their multiple referencing in the output report. `XTAGS` may edited but it must be done with caution.

`XREF` is a text file used to both control the cross-referencing process, and to enable a detailed print dump of the input file parsing and the import-population process. The content of this file is self-descriptive. For most cross-referencing tasks the contents of `XREF` should not need to be changed.

XREF controls the following.

(a) When a domain dictionary is being cross-referenced *against itself* it is beneficial to search for tag occurrences *inside dictionary descriptive values*, as tags are often referred to in such text. However, in its normal execution mode TAGxref will not cross-reference tags within item values enclosed by semicolons. To enable this to happen a '**;**' character is placed on the first line of the XREF file.

(b) Although the print-dump controls in XREF are intended primarily for debugging puposes, for some complex IMPORT nesting constructions it may be useful to print the steps taken by TAGxref to read and inject the imported information into a dictionary. This is achieved using the "dump" character codes described in XREF.

TAGxref Execution

TAGxref is distributed with an execution script 'xr' for use in a UNIX environment (equivalent scripts can easily be built for other OSs). To execute TAGxref enter the following:

xr *<STAR filename to be x-referenced> <conformance dictionary filename>*

Example applications:

`xr saly.cif cif.dic`	x-reference tags in a cif file with those in a domain dictionary
`xr mss.txt  cif.dic`	x-reference tags in a text manuscript with those in a domain dictionary
`xr cif.dic`	x-reference tags in a domain dictionary with its own definitions
`xr ddl.dic`	x-reference tags in the reference dictionary with its own definitions

TAGxref Output files

TAGxref generates four output files.

xrlis This output file contains a list of the cross-referenced tags and line numbers. The line numbers refer in the generated text file xrotex. In addition this list is retained as a separate file with the input text filename and an extension '.xrlis'. Using the 'xr' execution script any prior file of this name will be renamed as '.xrlis.old' .

xrlog This output file contains a record of the parsing of the primary input files, and all files that are imported. If the print record of this process is expanded (using the XREF controls) this will appear in this file. In addition this record is retained as a separate file with the input text filename and an extension '.xrlog'. Using the 'xr' execution script any prior file of this name will be renamed as '.xrlog.old' .

xrodic This output file is a copy of the input dictionary file but with all IMPORT directives (i.e. _import.get [ ]) executed and expanded. This file is also retained separately with the input dictionary filename and an extension '.xrodic'. Using the 'xr' execution script any prior file of this name will be renamed as '.xrodic.old' .

xrotex This output file is a copy of the input text file but with all IMPORT directives (if any exist) executed and expanded. This file is also retained separately with the input text filename and an extension '.xrotex'. Using the 'xr' execution script any prior file of this name will be renamed as '.xrotex.old' . All line numbers listed in xrlis refer to this output file.

ALIGN

ALIGN is a dictionary compliance tool that is specially designed for *DDLm* dictionary writers and maintainers.

Its use ensures a close compliance between the data items (both tags and values) in a STAR file and the dictionary in which they are defined. Most importantly, the checked STAR file can be a dictionary and thus allows attributes in a domain dictionaries (written in DDLm) to be aligned with their defined properties in the DDLM reference dictionary.

NB Depending on the scope of the input dictionary(s), exhaustive compliance tests using ALIGN may only be possible if IMPORT statements (i.e. `import.get [ ]`) in the dictionary(s) have been replaced by the imported data. The program TAGxref should be used to populate dictionaries with imported information prior to using ALIGN. TAGxref outputs files (with filename extensions '`.xrotex`' and '`.xrodic`') containing import-expanded data, and these are appropriate for input to ALIGN.

ALIGN can also be used as a remediation tool. By entering 'o' in the `XALIGN` control file (see (b) below) ALIGN outputs a copy of the input STAR file with any *alias tags* (as defined in the dictionary with attribute `_alias.definition_id`) replaced with the *primary tags* (as defined by the attribute `_definition.id`). The output file has the same name as the input STAR file but with the extension '**`.aln`**'

ALIGN Control file **`XALIGN`**

Several aspects of the ALIGN checking process may be controlled using the auxiliary input file `XALIGN`. The content of this file is self-descriptive. For most checking instances the contents of `XALIGN` should not need to be changed.

`XALIGN` controls the following.

(a) When a domain dictionary is being checked against the DDLM reference dictionary, it is usually beneficial to exclude alert messages generated by any *domain-specific* states assigned to the attribute `_type.contents` in the domain dictionary (e.g. '`symop`' in `cif.dic`). Because domain-specific states are not defined in the reference dictionary they cause alerts to be generated. These can obfuscate and mask more serious errors- they may be nullified by placing up to ten domain-specific *state codes* into lines 2 to 11 of `XALIGN`.

(b) The way in which ALIGN processes and prints data can be controlled by a sequence of one-letter characters entered on the first line of `XALIGN`. This included the facility to output a copy of the input STAR file with alias tags replaced by the primary defined tags (by entering the character '**`o`**'), and changing the way in which measurand values are written (by entering '**`u`**'). This is described further in (c) below. Most character controls are intended primarily for debugging purposes, in some complex file constructions it may be useful to print the internal tables used by ALIGN to store the attribute information from the STAR file and the dictionary. This is achieved using the "dump" character codes described in `XALIGN`.

(c) Sometimes in a STAR files it is more convenient to express numerical values and their standard uncertainty values as separate items; the value with a tag <*dataname*> and the standard uncertainty value with a separate tag <*dataname*>`_su`. To facilitate this style of STAR file presentation, ALIGN can convert *appended SU formats* (of the form `23.57(3)` where the SU value is `0.03`) into two separate items in the output STAR file. To facilitate this, the character code '**u**' is entered on the first line of `XALIGN`.

Then, the item:

```
        _measured.value          23.57(3)
```

will be output as…

```
        _measured.value          23.57
        _measured.value_su        0.03
```

ALIGN Execution

ALIGN is distributed with an execution script '**al**' for use in a UNIX environment (equivalent scripts can easily be built for other OSs). To execute ALIGN enter the following:

al *<STAR filename to be checked>* *<conformance dictionary filename>*

Examples:

`al saly.cif cif.dic.xrodic`	checks contents of cif file against its domain dictionary
`al cif.dic.xrodic ddl.dic.xrodic`	checks domain dictionary against reference dictionary
`al ddl.dic.xrodic`	checks the reference dictionary against itself

ALIGN Output files

ALIGN generates two output files.

The output file with the extension '`.log`' is a record of the ALIGN run with the error alerts reported. Using the '`al`' execution script any prior file of this name will be renamed as '`.log.old`'.

If the control character '**o**' is entered into `XALIGN`, the input STAR filename is copied to a file with the filename extension '`.aln`'. Provided there are no serious syntax errors in the input file, the output file will be a fully compliant STAR file. Using the '`al`' execution script any prior file of this name will be renamed as '`.aln.old`'.

DICtree

DICtree is a data hierarchy-mapping tool for *DDLm* dictionary writers and maintainers.

It is used to map the hierarchical (i.e. parent and child tree) relationships of all primary items, and their categories, (i.e. those defined with the attribute `_definition.id`) in a dictionary written in DDLm. Aliased tags are not shown in this mapping.

NB Depending on the scope of the input dictionary, it may only be possible for DICtree to generate a complete hierarch map if IMPORT statements in the dictionary have been replaced by the imported data. The program TAGxref may be used to populate dictionaries with imported information and outputs a dictionary with the filename extension '`.xrodic`') that is suitable for use with DICtree.

DICtree Execution

DICtree is distributed with an execution script '`dt`' for use in a UNIX environment (equivalent scripts can easily be built for other OSs). To execute ALIGN enter the following:

dt *<dictionary filename>*

Examples:

`dt cif.dic.xrodic`	map the item/category hierarchies for a domain dictionary
`dt ddl.dic.xrodic`	map the item/category hierarchies for the reference dictionary

DICtree Output file

`tree` This output file contains a text map of the hierarchy of defined data items and categories. The map is also retained as a separate file with the input text filename and an extension '`.tree`'. Using the '`dt`' execution script any prior file of this name will be renamed as '`.tree.old`'.

DICalias

DICalias lists the hierarchy of primary tags, and the defined aliased tags, in a *DDLm* dictionary.

NB Depending on the scope of the input dictionary, it may only be possible fo[[r DICalias to generate a complete listing of tags if the IMPORT statements in the dictionary have been replaced with the imported data. The program TAGxref may be used to populate dictionaries with imported information, and its output dictionary (with the filename extension '`.xrodic`') is suitable for use with DICalias.

DICalias Execution

DICalias is distributed with an execution script '`da`' for use in a UNIX environment (equivalent scripts can easily be built for other OSs). To execute ALIGN enter the following:

da *<dictionary filename>*

Examples:

`da cif.dic.xrodic`	list the tag hierarchies, along with aliased tags, in a domain dictionary

DICalias Output file

`alias` This output file contains a text listing of the hierarchy of defined data items and categories, along with the corresponding aliased tags. This listing is also retained as a separate file with the input text filename and an extension '`.alias`'. Using the '`da`' execution script any prior file of this name will be renamed as '`.alias.old`'.

References

(1) Spadaccini, N.; Hall, S. R. *Extensions to the STAR File Syntax*. J Chem. Inf. Model. 2012, **52**, 1901-1906.

(2) Spadaccini, N.; Hall, S. R. *DDLm: A New Dictionary Definition Language*. J Chem. Inf. Model. 2012, **52**, 1907-1916.

(3) Spadaccini, N.; Castleden, I. R.; du Boulay, D.; Hall, S. R. *dREL Relational Expression Language for Dictionary Methods*. J Chem. Inf. Model. 2012, **52**, 1917-1925.