NAME

rtftool - RTF template expander for chemistry data

VERSION

1.11, 2011-3-18

SYNOPSYS

rtftool [-noinput] [-contractsupergroups 0/1] [-hydrogens asis/none/hetero/all] [-redolayout 0/1] [-security 0/1] [-style stylename] [-version] templatefile ?outputfile? ?inputfile..?

DESCRIPTION

rtftool is a program to process RTF template files. It recognizes placeholder tags in an RTF file, replaces them with chemical structure images, OLE objects or property data, and outputs an RTF result file suitable for further editing with standard word processors, including Microsoft Word.

The only required argument is the template file, which needs to be in RTF format. If no output file is specified, the result data is written to standard output. If no input file is specified, structure data is read from standard input. Reading of structure input can be completely suppressed by setting the *-noinput* option.

The RTF template is expected to contain tags in a pseudo-XML format. Only a single tag named *chemtag* is recognized. Other XML-style mark-up in the template is passed on unchanged. The chemistry tags can either be pairs of start/end tags, as in

<chemtag>....</chemtag>

or empty tags as in

<chemtag/>

The latter tag style is more common, since the only tags where data between the start and end tags is useful are hyperlinks. It is possible to use a template without any tags, but since no substitution occurs, this is not really useful

The exact function of the embedded tags is determined by their attribute set. The only required attribute is *class*. All other attributes have reasonable defaults. In the absence of an explicit structure source (i.e. the tag contains attributes which specify an external file name, or in-line data), the structure and reaction data used to expand the tags comes from the input file(s). The set of input files is opened as a single virtual multi-record file. The associated record number starts with one. Record two is then either the second record from the first file on the argument line (in case that file has more than one record), or the first record from the second input file. Tags which cannot be resolved, for example because they refer to records not present in the input data set, are output unchanged and could potentially be filled in a second processing step. The format of input files is automatically detected. The standard executable recognizes several dozen chemistry file formats. It is not required that the input data contains atomic 2D layout coordinates. If these are missing, they are automatically computed, but if they are present, they are used, possibly after suitable re-scaling.

The following attributes are decoded from the tags:

class This is the only required attribute. The associated attribute value can be one of:

png (or *image*): The structure or reaction data is resolved into a static PNG image. The image contains an invisible embedded SMILES or Reaction SMILES field, so the chemistry information can be recovered without OCR if necessary using for example the Cactvs toolkit.

emf: The structure of reaction data is resolved into a static EMF vector drawing. The image contains an invisible embedded CDX structure record. These can for example be read with ChemDraw just like a native CDX file. EMF images print nicer than pixel images. The RTF encoding of interactive OLE objects also contains an EMF image as fallback. In case the responsible OLE handler (i.e. ChemDraw or ISISDraw) is not installed on a computer where the RTF file is read, the depiction is still visible and printable.

cdx (or *chemdraw*): The structure or reaction data is resolved into an editable ChemDraw OLE object. If ChemDraw is installed on the computer, a double click on the image changes it into an interactive window, and the Word toolbar displays the standard ChemDraw menus and icons.

skc (or *isisdraw*): The structure or reaction data is resolved into an editable ISISDraw OLE object. If ISISDraw is installed on the computer, a double click on the image opens it up in the ISISDraw program. The drawing remains linked to the document, so editing the data in ISISDraw updates the document when the editor is closed.

symyx (or *symyxdraw*): This is essentially the same as *skc*, with the exception that upon loading the RTF file into Word the rendering is not immediately updated by the drawing application. Only upon opening and re-transmitting the drawing is replaced. This is required by a bug in the current version of SymyxDraw not found in ISISDraw. SymyxDraw does not honor any size and scale information in the OLE context when called automatically, so that the immediate redraw results in renderings that are much too large. They use the default unscaled SymyxDraw bond length, which is to make things worse significantly larger than that of ISISDraw. Strangely, the scaling constraints are properly taken into account when the drawing is manually opened. Since the initial rendering was not performed by the original software, its visual appearance is not exactly the same as after opening and updating.

property: The template tag is replaced by computed or extracted data from the referring structure or reaction which is displayed as text.

link: The template tag is replaced by a clickable hyperlink where the target address is computed or extracted from the referring structure or reaction. The data is expected to be a properly formatted hyperlink as a string. No further processing takes place. This tag is usually coded as an start/end tag pair in order to supply a complementary link text. The data used with this class is not incompatible to those used for in the context of the *property* tag class, so in order to create a link with its full target address visible, simply nest a *property* tag within the *link* tag pair, with both tags referring to the same data.

ref: The template tag is replaced by whatever data an externally specified secondary tag resolves to. With this class, the *file* attribute needs to be specified. This file is expected to contain the XML text, with all attributes, of a single empty <*chemtag/*> tag this application can parse. Parsing of the file contents is performed exactly as the parsing of a tag embedded in the primary RTF document. This feature can be useful, for example, in order to avoid the insertion of voluminous in-line data for repeated identical tags.

record This attribute defines the virtual or physical file record a tag refers to. The attribute value is a positive integer. It can be used in combination with a filename for direct access to a specific record in a file, or without a filename to refer to the virtual file formed by the input file command arguments. The default value of this attribute is one. Record numbering starts with one.

file Define a reference file. If a tag has a *file* attribute, it imports its data no longer from the file(s) listed as program arguments, but from this specific file. The file name may be a path relative to the current directory. This argument can be combined with the *record* argument to extract a specific record from a larger file.

inlinestructure Specify a structure source directly, without referring to an explicit or anonymous file. The attribute value is a line notation which the program can decode. In standard program releases, this includes plain or hex-encoded SMILES strings, InChI-strings, base64-encoded serialized Cactvs toolkit ensemble object strings, hex- or base64-encoded Cactvs toolkit Minimols, zlib-compressed base64-encoded MDL Molfiles, PubChem CID numbers and CAS numbers (which are looked up at PubChem, so this may or may not work, and may or may not be correct). The format is automatically determined. When an in-line structure is decoded, any record number specification is ignored. In-line data is supposed to contain only a single record.

inlinereaction Specify a reaction source directly, without referring to an explicit or anonymous file. The attribute value is a line notation which the program can decode. In standard program releases, this includes plain or hex-encoded Reaction SMILES strings, base64-encoded serialized Cactvs toolkit reaction object strings, and zlib-compressed base-64 encoded MDL RXN files. When an in-line reaction is decoded, any record number specification is ignored. In-line data is supposed to contain only a single record.

property Specify the name of a property. This argument is required for the *property* and *link* tag classes, and ignored on all others. The property name can either be specified in Cactvs toolkit notation, or as written in a data source (i.e. the content of the <..> part of the data line in an SD file, without the bracket characters, and with all spaces and punctuation within the brackets). The program can compute a fair number of standard derived property values from structures or reactions. The exact scope of this capability is dependent on the program configuration. Example useful computable structure properties are E_WEIGHT (molecular weight), E_SMILES, E_INCHI (SMILES and InChI strings), E_CID (PubChem CID), E_CAS (CAS number) and E_FORMULA (molecular formula). Names and registration IDs found in the header portion of an MDL Molfile/SDfile are accessible as properties E_NAME and E_IDENT. In case a property name contains blanks, or characters which might interfere with XML parsing, it must be quoted.

height and **width** These attributes set the *maximum* size of a structure or reaction image. Since the depiction is clipped, the actual size is often smaller. The values are positive integers and the units are points (1/72 of an inch). The defaults are 200x200 pts for structures and 200x400 pts for reactions. In case a depiction were to exceed this size with the standard or configured bond length, the bond length is automatically shrunk so that the depiction remains within these boundaries.

bondlength Set a standard bond length in points (1/72 of an inch) as a positive integer value. If this attribute is not set, the default bond length of the standard software associated with the expanded tag object format is used (i.e. ChemDraw for *cdx* OLE, ISISDraw for *skc* OLE, Cactvs for PNG and EMF). In any case, a user-specified bond length is overridden and rescaled in case the resulting sketch would exceed the box given by the *height* and *width* attributes, or their respective defaults.

hydrogens Set a local hydrogen mode for this tag. This attribute overides the global option of the same name for this tag only.

arrow Set an arrow style for reaction depictions. The possible attribute values are *left*, *right*, *resonance*, *equilibrium*, and *none*. If the displayed object is a structure, or this attribute is used in a *property* or *link* tag, it is ignored. In case the attribute is not set, and the input format contains arrow style information (for example, if the input is a cdx/cdxml or skc/tgf file), this information is preserved. Otherwise, a default arrow style is chosen which is dependent on the display format.

over and **under** Set annotation texts to be added over or under a reaction arrow. If the displayed object is a structure, or this attribute is used in a *property* or *link* tag, it is ignored. If the attribute is repeated, every instance of the attribute starts a new line. In case there a multiple annotations, those specified first are drawn closest to the arrow and additional lines are added while moving away from the arrow line to the top or bottom.

redolayout This tag is followed by a boolean value (0 or any positive number). It the value is not zero, the 2D layout coordinates of the structure or reaction object are recalculated before the depiction is generated. By default, pre-existing layout coordinates are preserved and only scaled to suit the output format. It is possible to use original and recomputed layouts of the same source, such as a specific file record, together in a single document. This attribute overides the global option of the same name for this tag only.

Tag attributes other then these may be present, but are ignored.

Some care must be taken when preparing RTF template files with tags. The tag parser is rather robust and tolerates a fair amount of RTF mark-up within a tag specification (see the sample template file) and automatically filter it out. Nevertheless, this is not completely foolproof. It is not allowed to have grouped RTF markup sequences in curly brackets which cross the start or end of the tag text. On expansion, this can lead to RTF output with a corrupted bracketing structure which may not be readable, or have a corrupted layout. Manually typing RTF template files works reasonably well on older MS Word versions, but MS Word 2007 is really dangerous in this respect. Generally, it is not advisable to correct or change anything within an existing tag by editing it within Word. It is safer to delete the tag completely, and then to retype is carefully in one go, without any intervening mouse clicks, formatting commands, etc. Of course, using an ASCII text editor to prepare or clean up a template file is a preferred solution in any case. If such a tool is used, the tags should be encoded cleanly, without any RTF markup in their text. Any RTF layout or format specifications which wrap around a tag are preserved and apply to the formatting, so there is no need for tag attributes to specify font styles, text colors, and the like for *property* tags.

Expanded output files may display slightly differently depending on whether a suitable manager program associated with the underlying OLE format of an image is installed on the computer where it is viewed or not. The generated RTF code tries to invoke the responsible OLE manager once immediately upon loading the file into MS Word or another RTF viewer (except in the case of SymyxDraw, see above), so on computers where the software is present the depiction is essentially regenerated with the original chemical drawing software, using only the most fundamental layout information such as atomic coordinates from the embedded data. On computers which do not have a suitable OLE manager application installed, the OLE component is ignored, and the precomputed preview and fallback EMF image is displayed and printed instead. Since this image was not generated by the original chemical drawing software, there may be unavoidable differences in fonts, line thicknesses, etc.. Nevertheless, the fallback image should be reasonably close to a regenerated version in normal cases. Once an OLE object was successfully opened or automatically updated on a computer with the proper manager application installed, the preview/fallback image is also regenerated and further viewers of a re-saved file always render the exact display style of the original chemical drawing application.

EXAMPLES

rtftool -security 0 v1template.rtf v1output.rtf a_chemical.sdf

Above command reads the template file *v1template.rtf*, write processed output to file *v1output.rtf*, and use the multi-record file *a_chemical.sdf* as chemical structure data source.

OPTIONS

-contractsupergroups 0/1

This flags controls how groups of atoms which are marked as a superatom in the input data are drawn. If the flag is on, the individual atoms of the superatom group are not drawn and instead the group label is shown. If the flag is off, the group label is omitted and the individual atoms are plotted. The default value for this option is 1, i.e. superatom groups are drawn contracted. In ChemDraw OLE objects, the transmitted structure data contains the full atom set even if the groups are contracted. The group labels can be selected and expanded via the atom menu, using the original atom coordinates. This option only controls expansion of groups for which the full atom set is part of the input data. Groups which are encoded as a single pseudo atom are always plotted with the superatom label, and expansion may or my not work in ChemDraw, depending on whether ChemDraw understand the label. Even if expansion succeeds, coordinates for such a group are newly computed.

Some tools, such as the Marvin tools and ChemDraw, encode a full set of atom coordinates for contracted groups in their respective output files, but these coordinates appear frequently to have been computed without regard for the environment of the expanded group. Expansion may therefore result in atom overlaps and generally unsatisfactory layout - but these are the original input coordinates. If the layout is recomputed with the **-redolayout** option, new coordinates are also generated for the hidden group atoms.

-hydrogens asis/none/hetero/all

Control whether hydrogens are shown or not. The *asis* mode takes the hydrogens as they are specified in the input data. In mode *none* (or *suppress*), all hydrogens are stripped. In mode *hetero* (or *special*), hydrogens are implicitly added and shown on hetero atoms and in some other standard environments, such as with aldehydes. In mode *all*, a complete hydrogen set is added, and all hydrogens are shown.

-noinput

Do not read any input files. This can be useful in case the template file only contains named file references or in-line data. Alternatively, it is also possible to redirect standard input to /dev/null for the same effect.

-redolayout 0/1

If this flag is set, all 2D layout coordinates of the objects read from the input file(s) are recomputed before the objects are drawn. However, this applies only to objects read form the command line input files, not to any in-line data or file references embedded in the tags. If these are to be re-layouted, please use the *redolayout* tag attribute for tag-level control. By default this flag is not set and original coordinates are preserved if present. In case the input data does not contain 2D coordinates, they are always automatically computed.

-style *stylename*

By default, embedded OLE objects for ChemDraw and ISIS/Draw use the default style parameters, such as bond length, font sizes and line thickness, of the underlying chemical drawing program. With this option, these default style parameters can be globally changed to a preconfigured bundle. This corresponds to the *“File/Apply Document Settings from”* menu item in ChemDraw and *“Options/Read Journal or Custom Settings”* menu item in ISIS/Draw. At this time, the only recognized style name is *acs*, which corresponds to the “*JACS*” style in ISIS/Draw and “*ACS Document 1996*” in ChemDraw. Passing an empty string, or an unrecognized style name, is equivalent to omitting the option altogether.

-security 0/1

The software contains a comprehensive set of basic chemical computation routines for standard data, such as molecular weight, etc. These can be referenced as computed property fields in the template. However, some of these routines contact Internet databases and services for their functionality. Examples are the routines to obtain PubChem reference IDs from structures. In a commercial context, submission of confidential structure data is usually not desired. If the security flag is set, all functions which would submit structure data to external sources are suppressed. Properties which can be computed locally are still accessible. By default, structure security is enabled.

-version

Print version and licensing information and then exit.

COPYRIGHT

This program was designed and implemented for the CACTVS system by W. D. Ihlenfeldt. All rights reserved. This program is not part of the standard CACTVS toolkit distribution and must not be used without a license in any context.