Designing Universal Chemical Markup – Supplemental information

Jan Mokrý[1] and Miloslav Nič[2]

[1]**Department of Inorganic Chemistry, University of Chemistry and Technology Prague, Technická 5, 166 28, Prague 6, Czech Republic**
[2]**Department of Software Engineering, Czech Technical University in Prague, Thákurova 9, 160 00, Prague 6, Czech Republic**

ABSTRACT

Supplemental information for the article ”Designing Universal Chemical Markup (UCM) through the reusable methodology based on analyzing existing related formats” includes additional file 1 (Interactive references), 2 (Formats excluded from second stage) and 3 (UCM tree structure).

Keywords: designing UCM, supplemental information, interactive references, formats excluded from second stage, UCM tree structure

ADDITIONAL FILE 1 – INTERACTIVE REFERENCES

The interactive XHTML references generated in step 1 provide an overview of format XML structure and can be used to create useful documentation resources for the particular formats. Preparing such resources just requires a further manual editing of descriptions extracted from the format schema. The scale of this manual work during step 3 depends on the quality of documentation annotations in the schema, because these are used by our Python modules in step 1 to generate the description of each attribute, element and type. The manual corrections of descriptions in the interactive references are necessary to ensure correct cross-linking, as our simple algorithm for marking occurrences of names can sometimes mark the word as the attribute or element name even when actual meaning is different.

Unfortunately, majority of analyzed schemas did not include documentation annotations. In fact only CML and PDBML schemas contained sizable amount of annotations, but even these schemas contained too many cases of incomplete or unclear documentation. For the examples of incomplete and not yet finalized or unclear and vague documentation annotations in CML schema version 3 see the annotations for attributes (e.g. *atomRefGroup*, *constraint*, *convention*, *duration*, *symbol*, *tautomeric*, etc.) or elements (e.g. *identifier*, *object*, *system*, etc.). In PDBML 4.0-4.052 (or 4.2-4.052) schema the examples include incomplete documentation annotations for the attributes (e.g. *datablockName*, *units*, etc.), elements (e.g. *atom_site*, *datablock*, *pdbx_molecule*, *pdbx_version*, *space_group*, *valence_ref*, etc.) or types (e.g. *datablockType* and *em_helical_entityType*).

Mentioned complications did not allow us to automatically generate the documentation that would be complete after reasonable amount of manual editing. Therefore, we decided not to output documentation for analyzed formats in final interactive references. This greatly reduced the amount of manual editing we needed to perform during the analysis, while the interactive references still proved to be useful for quickly finding out how attributes, elements and types defined by a given format schema depend on each other.

Our interactive references offer a good overview of the format XML structure by providing a generated tree structure overview, which functions as the table of contents. There is also a floating side menu that stays on the screen while a user is scrolling. Thus a quick access to navigation at any time is ensured. Users can choose among sections listing attributes, elements and types with cross-linked dependencies extracted from the schema of the particular format.

All interactive XHTML references we generated are freely available at our website (http://www.universalchemicalmarkup.org).

ADDITIONAL FILE 2 – FORMATS EXCLUDED FROM SECOND STAGE

This additional file contains information about formats included only in the first stage of our analysis. We discuss here the Strengths and weaknesses of formats as well as provide the Overview of formats.

Strengths and weaknesses of formats

For all formats we briefly describe our findings categorized into groups that denote to which requirement the findings relate:

- Requirement 1 (FUNCTIONALITY): In the Overview of formats we used keywords to express briefly what functionality each format offers. While the functionality supported by the given format can be seen as its strength, going into details one could find various weaknesses, for example:
  - Some formats support only the certain types of structures or data (e.g. CIF is specialized for crystallography data,[1,2,3,4,5] InChI does not support more complex polymers and large biochemical structures,[6,7] etc.), and most formats cannot record electrons (i.e. we found only CML, which was included in the second stage of our analysis, supports this).
  - Other formats that can record properties support only predefined properties (e.g. PDB, PDBx/mmCIF or PDBML), or do not properly enforce associating scientific units with the property values (e.g. NCBI ASN.1, NCBI XML, PDB, PDBx/mmCIF or PDBML). Although one may often find the property units in the format documentation or in the database where the file in the particular format was obtained, it could be more clear to include either the units directly or add the reference pointing to them.
  - Formats with more complex structure seem to have various redundant parts. An example is the annotation functionality in NCBI XML and PDBML clearly implemented using various elements. One may easily verify this (e.g. by searching the element names containing strings such as ”annotation”, ”comment”, ”description” or ”descr”, ”text”, etc.) in NCBI XML 20141117 and PDBML 4.0-4.052 (or 4.2-4.052) schemas or using the interactive references we prepared for these formats.
- Requirement 2 (FUNCTIONALITY): Validation functionality differs among analyzed formats. Formats based on XML or similar standard syntax offer at least basic built-in validation capabilities. Thus, data in NCBI ASN.1, NCBI XML and PDBML formats can be checked using standard ASN.1 or XML validation tools. These validation tools utilize format specifications defined in a machine-parsable form according to ASN.1 or XML technology requirements. In the case of NCBI ASN.1, NCBI XML and PDBML formats the built-in validation focuses on the structure of the formats, but the validity of chemical data seems not to be checked precisely. The remaining formats, which do not have any built-in validation capabilities, can be divided into two groups. The first group that includes InChI, SMILES, SLN and Mol2 does not provide any dedicated validation functionality. However, chemical software may perform some checks for example when saving or generating data in these formats (e.g. InChI software checks if the input structure is ambiguous or contains errors and shows warnings accordingly[8]). For formats from the second group (i.e. CIF, PDB and PDBx/mmCIF) there are specialized software tools for validation including the online validation services.[9,10,11,12,13] Both chemical information (especially the crystallographic data) and format structure can be validated with such tools.
- Requirement 3 (FUNCTIONALITY): Annotation functionality in a form of classic plain text descriptions is supported by most formats (i.e. CIF, NCBI ASN.1, NCBI XML, PDB, PDBx/mmCIF, PDBML, Mol2). On the other hand XHTML or similar markup that would enable hyperlinks and other useful formatting features inside the annotations seems not to be actively encouraged in mentioned formats. In compact chemical formats annotation functionality is obviously limited. InChI and SMILES do not offer annotations, while SLN provides some restricted plain text annotations using appropriate predefined attributes. Although one could theoretically use custom SLN attributes to add more annotation possibilities, it would probably go against the concise nature of the format. Overall we believe it is much better idea to include annotations around the InChI, SMILES and SLN strings, and thus we do not see the limited annotation functionality of these compact formats as a significant weakness.

- Requirements 4, 5 and 6 (MODIFIABILITY): In the modifiability requirements we mainly focused on how hard it is to modify either the given format (i.e. extend it) or its instance with data (i.e. transform it). For implementing the transformation of data stored in the analyzed formats, various programming languages can be used. Some of those programming languages are directly available in modern web browsers (e.g. XSLT and JavaScript, or other ECMAScript implementation) and may be utilized to transform the data from the analyzed formats into a form usable by web browsers. Although all formats we describe here are quite precisely defined and can be transformed into other formats or a web browser friendly form, XML technology brings various modifiability benefits for NCBI XML and PDBML. The examples of such benefits include: the possibility of using XML tool chain and especially XSLT to easily implement transformations, or the potential offered by XML namespaces for combining various XML formats in a single XML document (note that InChI, SMILES and SLN can be combined with other formats too). In addition XML technology may also increase the extensibility. However, some design choices tend to negate this, as we described in the article when discussing the XML benefits in detail. Therefore, the extensibility of NCBI XML and PDBML, which depends on NCBI ASN.1 and PDBx/mmCIF respectively, cannot match the extensibility of an independent XML format. Especially when any changes in NCBI ASN.1, PDB and PDBx/mmCIF depend on what is required by large databases using these formats. Both CIF with its dictionary mechanism and Mol2 formats seem to be relatively extensible, but independent XML formats usually offer even better extensibility. As explained in the article, XML formats may for example introduce new attributes and elements without breaking the existing functionality and the software working with the format can simply select just some of the supported attributes and elements it requires for the processing. In the case of compact formats, new or extended functionality may change the existing syntax ultimately leading to various versions with partly or completely incompatible features, as in distinct line notations based on SMILES (e.g. CurlySMILES has partially different syntax than Daylight SMARTS and has of course quite different features[14,15]). Both InChI and SLN try to avoid this. InChI has its mechanism of layered structure, while SLN uses the default predefined and custom user defined attributes.

- Requirements 7, 8 and 9 (USABILITY): For a format to fulfill our usability requirements it basically needs to be well structured, readable and properly documented, because then there is a higher probability that such a format will be searchable, easy to learn, simple to use and straightforward to implement. With the exception of compact formats (i.e. InChI, SMILES and SLN) the formats discussed here provide at least some self describing capabilities that help to achieve better readability, as in PDB and Mol2 formats. In the case of CIF, NCBI ASN.1 and PDBx/mmCIF the self describing capabilities are even similar to what is offered by XML formats like NCBI XML and PDBML. For InChI, SMILES and SLN the lack of self describing capabilities seems not to be a big weakness, because the data stored in these formats mostly record just the chemical graph of a structure. This structure is usually apparent in simpler cases even to a human user with only the basic knowledge of the format syntax and chemical software is often able to decode the structure and redraw it for the user (note we list the examples of software in the Overview of formats). Thus, all formats excluded from the second stage seem to be at least reasonably searchable, in the case of PDB and Mol2, or even adequately searchable in the case of remaining formats, which either benefit from good self describing capabilities or compact well defined syntax. As for how easy it is to understand and learn each format (e.g. to use it or implement it in software) we need to also look at other aspects that affect the overall readability of the format (i.e. its structure and the quality of documentation). With documentation it is quite straightforward, as most formats have adequate online documentation. Exceptions are NCBI XML and PDBML, which provide the detailed documentation of attributes and elements only in the source code of their schemas (though in PDBML 4.0-4.052 and 4.2-4.052 schemas some documentation annotations seem to be missing, as we describe in additional file 1). Fortunately one can usually find the relevant documentation also in NCBI ASN.1 and PDBx/mmCIF specifications respectively. The last format without the adequate online documentation is SLN. However, the published articles about SLN contain the thorough description of the format, and therefore, are sufficient substitute for online documentation. Now let us briefly discus the structure of formats described here. In InChI, SMILES and SLN, we think the structure is very well adapted to the compact nature of these formats, especially considering how the InChI layers or SLN attributes improve the modifiability of the format structure without disrupting its compactness. NCBI ASN.1 and NCBI XML use quite hierarchical structure in accordance with the ASN.1 syntax. And the remaining formats (especially those based on CIF such as PDBx/mmCIF or PDBML) seem to have mostly flat structure compared to the typical XML structure tree, which is often more hierarchical and enables grouping the similar or related parts together more clearly. During the first stage of our analysis we did not focus on details, but we noticed that NCBI XML and PDBML structure could be improved to better utilize the possibilities of XML technology. As we explain in our findings related to requirement 10, proper usage of XML attributes could increase the memory efficiency and readability of both formats. In addition it would also lead to more concise structure of these formats. On the other hand mechanisms automatically generating the NCBI XML and PDBML specifications would probably become more complex together with the translation of data between XML and non-XML formats used by NCBI and wwPDB (Worldwide Protein Data Bank). Finally it must be said that complex formats, such as those utilized by NCBI and wwPDB, will remain more difficult to understand and learn, although the available software continues to lower the usability barrier at least for users.

- Requirement 10 (PERFORMANCE): With regards to the performance of formats, we noticed only the lower memory efficiency of NCBI XML and PDBML. The exact testing of performance was not the focus of our analysis. However, when we saw how often NCBI XML and PDBML schemas utilize element nodes with quite long names, it was clear the memory efficiency of both formats can suffer. For example in NCBI XML 20141117 schemas (or in the interactive reference we prepared for the format) one may find elements such as *Atomic-coordinates*, *Atomic-coordinates_atoms*, *Atomic-coordinates_number-of-points*, *PC-StereoPentagonalBiPyramid*, *PC-StereoPentagonalBiPyramid_center* and so on. This seems to be the limitation of the automatic translation from NCBI ASN.1 specifications. Although the functionality of NCBI formats is defined in a modular way, modules are designed to be combined into one huge specification, where some modules depend on each other instead of being easily usable as standalone parts. Moreover, a single namespace is used for all NCBI XML modules. Consequently the names of parent elements are used as prefixes in the names of child elements to maintain the uniqueness of names. Elements with very long names can be also seen in PDBML 4.0-4.052 and 4.2-4.052 schemas (or in our interactive reference for the format). Some examples include *atom_site_auth_asym_id_1*, *exptl_crystal_grow_compCategory*, *hydrogen_bond_constraints_total_count*, *maximum_torsion_angle_constraint_violation*, *pdbx_exptl_crystal_cryo_treatmentCategory* and so on. The reason for such names seems to be again the automatic translation of specifications, which utilizes the names from PDB Exchange Dictionary. Another, problem is the fact that elements in both NCBI XML and PDBML are often used even for storing small data chunks like coordinates and other numeric values. This lowers the memory efficiency and readability further, as can be best seen from practical examples. For NCBI XML just download a simple chemical structure (e.g. methylbenzene) from NCBI PubChem database. Then, it is possible to check how some elements (e.g. *PC-Atoms_aid_E*, *PC-Bonds_aid1_E*, *PC-BondType*, *PC-Coordinates_aid_E*, *PC-Conformer_x_E*, etc.) are repeatedly used for storing small chunks of data. In the case of PDBML simply download a chemical structure (e.g. 2LZ5) from the RCSB (Research Collaboratory for Structural Bioinformatics) PDB database (RCSB PDB interface enables one to quickly view the raw PDBML file). Then, see how elements that are especially inside the *atom_site* element (e.g. *Cartn_x*, *Cartn_y*, *Cartn_z*, *auth_atom_id*, *occupancy*, etc.) repeatedly store small chunks of data.

- Requirements 11 and 12 (AVAILABILITY): As can be seen from the Overview of formats, specialized chemical software required for the practical usage of formats described in this additional file is available at least for Windows, Mac and Linux platforms. The specifications of most formats are freely available, although some under a proprietary license or policy (e.g. CIF,[16] Mol2,[17] some versions of SMILES except OpenSMILES,[18] etc.). The only exception is SLN, which seems to be thoroughly described only by published articles[19,20] that are well structured but not openly accessible for public.

Overview of formats

The following sections contain the basic information we gathered for all formats included only in the first stage of our analysis.

Crystallographic Information File (CIF)

CIF is the standard interchange format for representing crystallographic information for chemical structures.[1,2,3,4,5] Closely related is macromolecular CIF for macromolecular structures.[21,22] The CIF format was developed by the Working Party on Crystallographic Information in an effort sponsored by the International Union of Crystallography.[1] It is widely adopted format supported by chemical software and as the submission format for Acta Crystallographica and other journals.[1,4,23]

Updated: 2003-02-23
Version: 1.1
Website: http://www.iucr.org/resources/cif
Software (CIF): cif2cif,[9] CIFEDIT,[9,24] CIFLIB,[9,25] CIFtbx,[9,26] enCIFer,[9,27] Jmol,[9,28,29] Open Babel,[30,31] publCIF,[9,32] RasMol,[9,33] Xtal[9]
Keywords: CIF, Crystallographic Information File, chemical graph, 2D structure, 3D structure, crystal structure, structure property data
Links: http://www.iucr.org/resources/cif/spec/version1.1
http://www.iucr.org/resources/cif/software

International Chemical Identifier (InChI)

InChI is a non-proprietary identifier for chemical substances that can be used in printed and electronic data sources thus enabling easier linking of diverse data compilations.[34,8] It is designed to provide a standard and machine-readable way to encode molecular information and to facilitate the search for such information in databases and on the web.[35,8]

Updated: 2011-09-13
Version: 1.04
Website: http://www.iupac.org/home/publications/e-resources/inchi.html
Software (InChI): ChemDoodle,[36] ChemSketch,[37] Marvin Applets, Marvin Beans,[38] Open Babel[30,31]
Keywords: InChI, International Chemical Identifier, InChIKey, chemical identifier, chemical graph, structure
Links: http://www.inchi-trust.org

NCBI Abstract Syntax Notation 1 (NCBI ASN.1) and NCBI Extensible Markup Language (NCBI XML)

NCBI ASN.1 is used for the storage and retrieval of data such as nucleotide and protein sequences, biochemical structures, genomes, and MEDLINE records.[39] It permits computers and software systems of all types to reliably exchange both the data structure and content to achieve interoperability between platforms.[39]

NCBI XML can be regarded as a specification for the group of XML-based formats (each defined by a module in a separate XSD or DTD file), which provide a representation of various NCBI data in XML format.[40,41,42]

In NCBI data specifications one can find ASN, XSD or DTD files that describe ASN.1 and XML formats for various chemical data such as: MMDB (Molecular Modeling Database) Chemical Graph ASN.1/XML for NCBI MMDB chemical graph data; MMDB Structural Model ASN.1/XML for NCBI MMDB structural model data; or PubChem Substance ASN.1/XML for NCBI PubChem substance data. All XSD or DTD files, automatically generated from ASN.1 files,[40,41] are designed to be included together in one complex XSD or DTD module.[40,42]

Updated: NCBI ASN.1 and NCBI XML: 2014-11-17
Version: NCBI ASN.1 and NCBI XML: 20141117
Website: http://www.ncbi.nlm.nih.gov/data_specs
Namespace: http://www.ncbi.nlm.nih.gov (only for NCBI XML)
Schema: http://www.ncbi.nlm.nih.gov/data_specs/ver/20141117/schema/NCBI_all_20141117.xsd (only for NCBI XML)
Schema – Language: XSD

Software (NCBI ASN.1 and NCBI XML): NCBI Databases, NCBI Entrez, NCBI Blast, NCBI Toolkit[43,39,42,41]

Keywords: NCBI ASN.1, National Center for Biotechnology Information Abstract Syntax Notation One, NCBI XML, National Center for Biotechnology Information Extensible Markup Language, MMDB Chemical Graph ASN.1/XML, MMDB Structural Model ASN.1/XML, MMDB ASN.1/XML, PubChem Substance ASN.1/XML, PubChem ASN.1/XML, chemical graph, 2D structure, 3D structure, nucleotide sequence, peptide sequence, structured sequence, structure property data

Links: http://www.ncbi.nlm.nih.gov/Structure/asn1.html
http://www.ncbi.nlm.nih.gov/IEB/ToolBox/SDKDOCS/INDEX.HTML
http://www.ncbi.nlm.nih.gov/IEB/ToolBox/SDKDOCS/ASNLIB.HTML
http://www.ncbi.nlm.nih.gov/data_specs/NCBI_data_in_XML.html
http://www.ncbi.nlm.nih.gov/IEB/ToolBox/XML/ncbixml.txt

Protein Data Bank (PDB), Protein Data Bank Exchange Dictionary Macromolecular Crystallographic Information File (PDBx/mmCIF) and Protein Data Bank Markup Language (PDBML)

PDB format can store all data contained in the wwPDB archive.[44,45] The data contained in the archive include atomic coordinates, crystallographic structure factors and nuclear magnetic resonance experimental data.[44,45] Aside from coordinates, each deposition also includes the names of molecules, primary and secondary structure information, sequence database references, where appropriate, and ligand and biological assembly information, details about data collection and structure solution, and bibliographic citations.[44,45]

The PDB file format will be phased out in 2016, because as of 2014 it is being replaced with PDBx/mmCIF format, which uses macromolecular CIF syntax and is based on Protein Data Bank Exchange Dictionary.[12]

PDBML provides a representation of data from the Worldwide Protein Data Bank in XML format.[46,47] The schema of the format is automatically generated from the Protein Data Bank Exchange Dictionary.[46,47]

Updated: PDB: 2012-11-21; PDBx/mmCIF: 2015-02-28; PDBML: 2015-03-10

Version: PDB: 3.3; PDBx/mmCIF: 4.052; PDBML: 4.0-4.052 (and 4.2-4.052)

Website: http://www.wwpdb.org/documentation/file-format

Namespace: http://pdbml.pdb.org/schema/pdbx-v40.xsd or
http://pdbml.pdb.org/schema/pdbx-v42.xsd (only for PDBML)

Schema: http://pdbml.pdb.org/schema/pdbx-v40.xsd or
http://pdbml.pdb.org/schema/pdbx-v42.xsd (only for PDBML)

Schema – Language: XSD

Software (PDB, PDBx/mmCIF and PDBML): MMCIF Dictionary Suite,[48] Worldwide Protein Data Bank[44,45,46,47]

Software (PDB and PDBx/mmCIF): Jmol,[28,29] RasMol[33]

Software (PDB): ChemDoodle,[36] Marvin Applets, Marvin Beans,[38] Open Babel,[30,31] PerlMol,[49] PyMOL[50,51]

Software (PDBML): PDBjViewer,[52] Protein Molecular Viewer,[53,54] PDBML2CIF[48]

Keywords: PDB, Protein Data Bank, PDBx/mmCIF, Protein Data Bank Exchange Dictionary Macromolecular Crystallographic Information File, PDBML, Protein Data Bank Markup Language, chemical graph, 2D structure, 3D structure, crystal structure, nucleotide sequence, peptide sequence, structure property data

Links: http://www.wwpdb.org/documentation/format33/v3.3.html
http://mmcif.wwpdb.org
http://pdbml.pdb.org

Simplified Molecular Input Line Entry System (SMILES)

SMILES is a line notation for describing chemical structures using short ASCII (American Standard Code for Information Interchange) strings.[55,56,57] SMILES strings are widely supported by the chemical software, which usually supports generating chemically correct structure depictions from the molecules encoded as SMILES strings.[56,57] Canonical SMILES can be used as unique chemical identifiers[58,56,57] (although this is primarily useful when all canonical SMILES were created using a single canonicalizer, because different canonical SMILES may be produced by different algorithms[18]). The original SMILES specifications were published in 1988 and 1989.[55,58] It has since been modified and extended by others. Examples of two well known SMILES-based extensions are Daylight SMARTS and SMIRKS.[56] In 2007, an open standard called OpenSMILES was developed by the Blue Obelisk open source chemistry community.[59,60,18]

Updated: 2012-11-17
Version: 1.0
Website: http://www.opensmiles.org
Software (SMILES): ChemDoodle,[36] ChemSketch,[37] JME Molecular Editor,[61,62] Marvin Applets, Marvin Beans,[38] Open Babel,[30,31] PerlMol[49]
Keywords: SMILES, Simplified Molecular Input Line Entry System, OpenSMILES, Daylight SMILES, Daylight SMARTS, Daylight SMIRKS, chemical line notation, chemical identifier, chemical graph, structure, substructure, polymer structure, chemical reaction, chemical query
Links: http://www.opensmiles.org/opensmiles.html
http://www.opensmiles.org/spec/open-smiles.html
http://www.daylight.com/smiles
http://www.daylight.com/dayhtml/doc/theory/theory.smiles.html
http://www.daylight.com/dayhtml/doc/theory/theory.smarts.html
http://www.daylight.com/dayhtml/doc/theory/theory.smirks.html

SYBYL Line Notation (SLN)

SLN is an ASCII language used to represent chemical structures, including common organic molecules, macromolecules, polymers, and combinatorial libraries.[19,20] SLN is also used to express substructural queries, reactions and includes a complete facility for Markush representation.[19,20] This concise language is ideal for database storage of chemical entities as well as for network communication of structures and queries.[19,20]

Updated: 2008-11-11
Version: UNAVAILABLE
Website: UNAVAILABLE
Software (SLN): ChemDoodle,[36] Concord,[63] PerlMol[49]
Keywords: SLN, SYBYL Line Notation, chemical line notation, chemical graph, 2D structure, 3D structure, substructure, polymer structure, chemical reaction, chemical query
Links: UNAVAILABLE

Tripos Mol 2 File (Mol2)

Mol2 format offers a complete and portable representation of chemical structures used by SYBYL-X Suite.[17,64] It is written out as a free format ASCII file according to Mol2 format specifications to avoid the restrictions created by fixed text formats.[17,64]

Updated: UNAVAILABLE
Version: UNAVAILABLE
Website: http://tripos.com/index.php?family=modules,SimplePage,Mol2_File_Format2009
Software (Mol2): ChemDoodle,[36] Jmol,[28,29] Marvin Applets, Marvin Beans,[38] Open Babel,[30,31] RasMol,[33] SYBYL-X Suite[17,64]
Keywords: Mol2, Tripos Mol 2 File, SYBYL Mol 2 File, chemical graph, 2D structure, 3D structure, substructure, crystal structure
Links: http://www.tripos.com/mol2/mol2_format3.html

ADDITIONAL FILE 3 – UCM TREE STRUCTURE

The resulting basic XML tree structure, which we iteratively developed for UCM using concepts described in the article, is in tree structure scheme 1. The scheme uses the following simple syntax:

- `element` – Denotes an UCM element.
- `@attribute` – Denotes an UCM attribute.

- `(ATTRIBUTES)` – Specifies the enabled attributes of an UCM element (e.g. `point (@id, @x, @y, @z)` means the *point* element with the *id*, *x*, *y* and *z* attributes).
- Quantifiers ”?”, ”*” and ”+” are used to express 0 or 1, 0 or more, and 1 or more respectively.
- Keyword ”OR” has its literal meaning.
- Element contents are indented by four spaces.
- Ellipsis means the attributes and contents of the element are in its definition.

Tree Structure Scheme 1 The basic XML tree structure developed for UCM 1-1-1 on the basis of our concept analysis.

```
ucm (@id?, @version)
    description? ...

    define (@id?, @format)*
        description* ...
        property* ...
        node* ...
        OR
        UNITSML*
        OR
        BIBTEXML*

    structure (@id, @format, @type, @charge?)*
        description? ...
        structure* ...
        property* ...
        node* ...
        bond* ...
        point (@id, @x, @y, @z)*
            description? ...
            property* ...
        stereo* ...
        OR IUPAC-PREFERRED-NAME-U OR IUPAC-GENERAL-NAME OR CA-INDEX-NAME
        OR CAS-RN-U OR REAXYS-RN-U
        OR CHEMSPIDER-ID-U OR PUBCHEM-CID-U OR PUBCHEM-SID
        OR INCHI OR INCHI-KEY OR S-INCHI-U OR S-INCHI-KEY
        OR SMILES OR SMARTS OR SLN

description (@id?, @idrefs?, @litrefs?)
    XHTML* OR PLAINTEXT*

property (@id, @idrefs?, @type?, @quantity?)
    description? ...
    property* ...
    values (@id?)?

node (@id, @idrefs?, @charge?, @x?, @y?, @z?)
    description? ...
    property* ...
    particle* ...
    stereo? ...

bond (@id, @idrefs?, @order)
    description? ...
    property* ...
    join (@id?, @idrefs)*
    particle* ...
    stereo? ...

particle (@id?, @idrefs?, @type, @counts, @fractions?)
    description? ...
    property* ...
    share (@id?, @idrefs, @fractions)*

stereo (@id?, @idrefs, @sense)
    description? ...
```

REFERENCES

1 *Crystallographic Information Framework*. The International Union of Crystallography (IUCr). Available at: ⟨http://www.iucr.org/resources/cif⟩.

2 *CIF Version 1.1 Working specification*. The International Union of Crystallography (IUCr). Available at: ⟨http://www.iucr.org/resources/cif/spec/version1.1⟩.

3 HALL, S. R.; ALLEN, F. H.; BROWN, I. D. The crystallographic information file (CIF): a new standard archive file for crystallography. *Acta Crystallographica Section A Foundations of Crystallography*. 1991, vol. 47, pp. 655–685. Available also at: ⟨http://dx.doi.org/10.1107/S010876739101067X⟩.

4 BROWN, I. D. CIF (crystallographic information file): A standard for crystallographic data interchange. *Journal of Research of the National Institute of Standards and Technology*. 1996, vol. 101, pp. 341–346. Available also at: ⟨http://nistdigitalarchives.contentdm.oclc.org/cdm/ref/collection/p13011coll6/id/48323⟩.

5 BROWN, I. D.; MCMAHON, B. CIF: the computer language of crystallography. *Acta Crystallographica Section B Structural Science*. 2002, vol. 58, pp. 317–324. Available also at: ⟨http://dx.doi.org/10.1107/S0108768102003464⟩.

6 NITSCHE, C. *What's up InChI?* Chemical Information BULLETIN, 2011. Available at: ⟨http://bulletin.acscinf.org/node/263⟩.

7 *Project Details: InChI Requirements for Representation of Polymers*. International Union of Pure and Applied Chemistry (IUPAC). Available at: ⟨http://www.iupac.org/nc/home/projects/project-db/project-details.html?tx_wfqbe_pi1%5Bproject_nr%5D=2009-042-1-800⟩.

8 *The InChI Standard - Technical FAQ*. InChI Trust. Available at: ⟨http://www.inchi-trust.org/technical-faq⟩.

9 *Software for CIF and STAR*. The International Union of Crystallography (IUCr). Available at: ⟨http://www.iucr.org/resources/cif/software⟩.

10 *IUCr checkCIF*. The International Union of Crystallography (IUCr). Available at: ⟨http://checkcif.iucr.org⟩.

11 *RCSB PDB Data Validation and Deposition Services*. Research Collaboratory for Structural Bioinformatics (RCSB). Available at: ⟨http://deposit.rcsb.org⟩.

12 *PDBx/mmCIF General FAQ*. Available at: ⟨http://mmcif.wwpdb.org/docs/faqs/pdbx-mmcif-faq-general.html⟩.

13 READ, R. J. et al. A New Generation of Crystallographic Validation Tools for the Protein Data Bank. *Structure*. 2011, vol. 19, pp. 1395–1412. Available also at: ⟨http://dx.doi.org/10.1016/J.STR.2011.08.006⟩.

14 DREFAHL, A. CurlySMILES: a chemical language to customize and annotate encodings of molecular and nanodevice structures. *Journal of Cheminformatics*. 2011, vol. 3, pp. 1. Available also at: ⟨http://dx.doi.org/10.1186/1758-2946-3-1⟩.

15 *Daylight Theory Manual - 4. SMARTS - A Language for Describing Molecular Patterns*. Daylight Chemical Information Systems, 2008. Available at: ⟨http://www.daylight.com/dayhtml/doc/theory/theory.smarts.html⟩.

16 *The IUCr policy for the protection and the promotion of the STAR File and CIF standards for exchanging and archiving electronic data*. The International Union of Crystallography (IUCr), 2000. Available at: ⟨http://www.iucr.org/resources/cif/comcifs/policy⟩.

17 *Tripos Mol2 File Format*. Tripos. Available at: ⟨http://tripos.com/index.php?family=modules,SimplePage,Mol2_File_Format2009⟩.

18 CRAIG, A. J. et al. *OpenSMILES specification*. Blue Obelisk community, 2012. Available at: ⟨http://www.opensmiles.org/opensmiles.html⟩.

19 ASH, S. et al. SYBYL Line Notation (SLN): A Versatile Language for Chemical Structure Representation. *J. Chem. Inf. Comput. Sci.* 1997, vol. 37, pp. 71–79. Available also at: ⟨http://dx.doi.org/10.1021/ci960109j⟩.

20 HOMER, R. W. et al. SYBYL Line Notation (SLN): A Single Notation To Represent Chemical Structures, Queries, Reactions, and Virtual Libraries. *J. Chem. Inf. Model.* 2008, vol. 48, pp. 2294–2307. Available also at: ⟨http://dx.doi.org/10.1021/ci7004687⟩.

21 BOURNE, P. E. et al. The Macromolecular Crystallographic Information File (mmCIF) in *Macromolecular Crystallography Part B*. San Diego: Academic Press, 1997, pp. 571–590. Available also at: ⟨http://mmcif.wwpdb.org/docs/pubs/methods-enzymology-paper-1997.html⟩. ISBN 9780121821784.

22 WESBROOK, J. D.; BOURNE, P. E. STAR/mmCIF: An Ontology for Macromolecular Structure. *Bioinformatics*. 2000, vol. 16, pp. 159–168. Available also at: ⟨http://dx.doi.org/10.1093/BIOINFORMATICS/16.2.159⟩.

23 BROWN, I. D.; MCMAHON, B. The Crystallographic Information File (CIF). *Data Science Journal*. 2006, vol. 5, pp. 174–177. Available also at: ⟨http://www.jstage.jst.go.jp/article/dsj/5/0/5_0_174/_article⟩.

24 TOBY, B. H. CIF applications. XIII. CIFEDIT, a program for viewing and editing CIFs. *Journal of Applied Crystallography*. 2003, vol. 36, pp. 1288–1289. Available also at: ⟨http://dx.doi.org/10.1107/S0021889803016790⟩.

25 WESBROOK, J. D.; HSIEH, S. H.; FITZGERALD, P. M. D. CIF Applications. VI. CIFLIB: an application program interface to CIF dictionaries and data files. *Journal of Applied Crystallography*. 1997, vol. 30, pp. 79–83. Available also at: ⟨http://dx.doi.org/10.1107/S0021889896008643⟩.

26 HALL, S. R. CIF applications. IV.CIFtbx: a tool box for manipulating CIFs. *Journal of Applied Crystallography*. 1993, vol. 26, pp. 482–494. Available also at: ⟨http://dx.doi.org/10.1107/S0021889893050897⟩.

27 ALLEN, F. H. et al. CIF applications. XV. enCIFer: a program for viewing, editing and visualizing CIFs. *Journal of Applied Crystallography*. 2004, vol. 37, pp. 335–338. Available also at: ⟨http://dx.doi.org/10.1107/S0021889804003528⟩.

28 *Jmol: an open-source Java viewer for chemical structures in 3D*. Available at: ⟨http://www.jmol.org⟩.

29 WILLIGHAGEN, E. L. Processing CML conventions in Java. *Internet Journal of Chemistry*. 2001, vol. 4, pp. article 4. Available also at: ⟨http://www.openscience.org/~egonw/cml/cml_conventions.html⟩.

30 O'BOYLE, N. M. et al. Open Babel: An open chemical toolbox. *Journal of Cheminformatics*. 2011, vol. 3, pp. 33. Available also at: ⟨http://dx.doi.org/10.1186/1758-2946-3-33⟩.

31 *Supported File Formats and Options*. Open Babel. Available at: ⟨http://openbabel.org/docs/current/FileFormats/Overview.html⟩.

32 WESTRIP, S. P. publCIF: software for editing, validating and formatting crystallographic information files. *Journal of Applied Crystallography*. 2010, vol. 43, pp. 920–925. Available also at: ⟨http://dx.doi.org/10.1107/S0021889810022120⟩.

33 *RasMol and OpenRasMol - Molecular Graphics Visualisation Tool*. Available at: ⟨http://www.openrasmol.org⟩.

34 *The IUPAC International Chemical Identifier (InChI)*. International Union of Pure and Applied Chemistry (IUPAC). Available at: ⟨http://www.iupac.org/home/publications/e-resources/inchi.html⟩.

35 *About the InChI Standard*. InChI Trust. Available at: ⟨http://www.inchi-trust.org/about-the-inchi-standard⟩.

36 *ChemDoodle: Features - Universality*. iChemLabs. Available at: ⟨http://www.chemdoodle.com/features/universality⟩.

37 *ChemSketch: A complete software package for drawing chemical structures.* ACD Labs. Available at: ⟨http://acdlabs.com/products/draw_nom/draw/chemsketch⟩.

38 *File formats in Marvin*. ChemAxon. Available at: ⟨https://docs.chemaxon.com/display/marvinsketch/File+formats+in+Marvin⟩.

39 *ASN.1 File Format (Summary)*. National Center for Biotechnology Information (NCBI), 2014. Available at: ⟨http://www.ncbi.nlm.nih.gov/Structure/asn1.html⟩.

40 *NCBI data specifications*. National Center for Biotechnology Information (NCBI), 2014. Available at: ⟨http://www.ncbi.nlm.nih.gov/data_specs⟩.

41 *NCBI Data in XML*. National Center for Biotechnology Information (NCBI), 2005. Available at: ⟨http://www.ncbi.nlm.nih.gov/data_specs/NCBI_data_in_XML.html⟩.

42 *NCBI Data in XML - NCBI ToolBox*. National Center for Biotechnology Information (NCBI), 2004. Available at: ⟨http://www.ncbi.nlm.nih.gov/IEB/ToolBox/XML/ncbixml.txt⟩.

43 VAKATOV, D. (Ed.) *The NCBI C++ Toolkit Book*. Bethesda (MD): National Center for Biotechnology Information (US), 2011. 1120 pp. Available also at: ⟨http://www.ncbi.nlm.nih.gov/toolkit/doc/book⟩.

44 *Protein Data Bank Contents Guide: Atomic Coordinate Entry Format Description*. The Worldwide Protein Data Bank (wwPDB), 2011. Available at: ⟨http://www.wwpdb.org/documentation/format33/v3.3.html⟩.

45 *The Worldwide Protein Data Bank File Format Documentation*. The Worldwide Protein Data Bank (wwPDB). Available at: ⟨http://www.wwpdb.org/documentation/file-format⟩.

46 WESBROOK, J. et al. PDBML: the representation of archival macromolecular structure data in XML. *Bioinformatics*. 2005, vol. 21, pp. 988–992. Available also at: ⟨http://dx.doi.org/10.1093/BIOINFORMATICS/BTI082⟩.

47 *PDBML Schema Resources*. The Worldwide Protein Data Bank (wwPDB). Available at: ⟨http://pdbml.pdb.org⟩.

48 *RCSB Software Tools*. Research Collaboratory for Structural Bioinformatics (RCSB). Available at: ⟨http://sw-tools.rcsb.org⟩.

49 *PerlMol - Perl Modules for Molecular Chemistry*. Available at: ⟨http://www.perlmol.org⟩.

50 *PyMOL - View 3D Molecular Structures*. Available at: ⟨http://www.pymol.org/view⟩.

51 *PyMOLWiki - Command Line Options*. Available at: ⟨http://pymolwiki.org/index.php/Command_Line_Options⟩.

52 *PDBjViewer (jV) - interactive molecular viewer*. Available at: ⟨http://www.pdbj.org/jv/index.html⟩.

53 *Protein Molecular Viewer*. Available at: ⟨http://www.zti.aei.polsl.pl/w3/dmrozek/pmView.htm⟩.

54 MROZEK, D.; MASTEJ, A.; MALYSIAK, B. Protein Molecular Viewer for Visualizing Structures Stored in the PDBML Format in PIETKA, E.; KAWA, J. (Eds.) *Information Technologies in Biomedicine*. Berlin, Heidelberg: Springer, 2008, pp. 377–386. Available also at: ⟨http://dx.doi.org/10.1007/978-3-540-68168-7_42⟩. ISBN 9783540681687.

55 WEININGER, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* 1988, vol. 28, pp. 31–36. Available also at: ⟨http://dx.doi.org/10.1021/CI00057A005⟩.

56 *SMILES – Simplified Molecular Input Line Entry System*. Daylight Chemical Information Systems, 2008. Available at: ⟨http://www.daylight.com/smiles⟩.

57 *Daylight Theory Manual - 3. SMILES - A Simplified Chemical Language*. Daylight Chemical Information Systems, 2008. Available at: ⟨http://www.daylight.com/dayhtml/doc/theory/theory.smiles.html⟩.

58 WEININGER, D.; WEININGER, A.; WEININGER, J. L. SMILES. 2. Algorithm for generation of unique SMILES notation. *J. Chem. Inf. Comput. Sci.* 1989, vol. 29, pp. 97–101. Available also at: ⟨http://dx.doi.org/10.1021/CI00062A008⟩.

59 *OpenSMILES Home Page*. Blue Obelisk community. Available at: ⟨http://www.opensmiles.org⟩.

60 CRAIG, A. J. et al. *OpenSMILES Specification DRAFT*. Blue Obelisk community, 2007. Available at: ⟨http://www.opensmiles.org/spec/open-smiles.html⟩.

61 ERTL, P. Molecular structure input on the web. *Journal of Cheminformatics*. 2010, vol. 2, pp. 1. Available also at: ⟨http://dx.doi.org/10.1186/1758-2946-2-1⟩.

62 ERTL, P. *JME Molecular Editor*. 2012. Available at: ⟨http://www.molinspiration.com/jme⟩.

63 *Concord - Generate accurate 3D coordinates*. Tripos. Available at: ⟨http://www.tripos.com/data/SYBYL/Concord_072505.pdf⟩.

64 *Tripos Mol2 File Format - Sample Mol2 File*. Tripos. Available at: ⟨http://www.tripos.com/mol2/mol2_format3.html⟩.