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 ”Universal Chemical Markup (UCM) - A new format for common chemical data” includes additional file 1 (UCM examples and schemas) and 2 (UCM documentation).

Keywords: UCM, supplemental information, UCM examples and schemas, UCM documentation

ADDITIONAL FILE 1 – UCM EXAMPLES AND SCHEMAS

To ensure UCM 1-1-1 examples and schemas remain accessible, we provide these files in the compressed archive **UCM--1-1-1--SPECIFICATIONS.zip**, which is a part of this additional file, and can be freely downloaded at http://www.universalchemicalmarkup.org. The contents of the archive include:

- **SCHEMA--UCM--1-1-1.rnc** – The RELAX NG compact syntax version of the main UCM schema for grammar-based validation.
- **SCHEMA--UCM--1-1-1.rng** – The RELAX NG XML syntax version of the main UCM schema (automatically translated with Trang).
- **SCHEMA--UCM--1-1-1.xsd** – The main part of the automatically translated XSD version of the main UCM schema (remaining parts are in files **SCHEMA--UCM--1-1-1--*.xsd**).
- **SCHEMA--UCM--NAMESPACES--VALIDATION--1-1-1.nvdl** – The NVDL schema for validating the content from all namespaces enabled in UCM files. Note that validating by this schema requires the main UCM schema (**SCHEMA--UCM--1-1-1.rng**) and schemas for all XML formats integrated in UCM (see files **Schemas/SCHEMA--***).
- **SCHEMA--UCM--RULES--VALIDATION--1-1-1.sch** – The Schematron schema for checking more complex constraints between two or more UCM attributes, elements or values. Using this schema and the reference ”Skeleton” implementation of Schematron, we generated the validation stylesheets directly usable with a standard XSLT 2.0 processor:
  - **SCHEMA--UCM--RULES--VALIDATION--1-1-1.xsl** – Outputs all found errors to the console.
  - **SCHEMA--UCM--RULES--VALIDATION--1-1-1.svrl** – Outputs the validation report.
- **Examples/UCM--1-1-1--EXAMPLE--*.xml** – Files that contain UCM examples.
- **Examples/DEFINITIONS--*.xml** – Files that contain definitions used by UCM examples.

In order to illustrate how the provided UCM examples may be processed and validated using various UCM schemas, we include the following commands, which the readers can adapt to their computing environment:

- Define environment variables to avoid re-typing the paths to UCM files:

  ```
  export UCM_HOME='./UCM--1-1-1--SPECIFICATIONS'
  export UCM_EXAMPLE='UCM--1-1-1--EXAMPLE--01.xml'
  ```

- Validate the specified example using the RELAX NG XML syntax version of the main UCM schema (Requires: xmllint – available as a part of libxml2 from http://xmlsoft.org):

  ```
  xmllint --xinclude --relaxng
  "$UCM_HOME"/SCHEMA--UCM--1-1-1.rng
  "$UCM_HOME"/Examples/"$UCM_EXAMPLE"
  ```

- Validate the specified example using the RELAX NG compact syntax version of the main UCM schema (Requires: xmllint and jing – available from http://www.thaiopensource.com/relaxng/jing.html):

  ```
  xmllint --xinclude "$UCM_HOME"/Examples/"$UCM_EXAMPLE"
  --output "$UCM_HOME"/TMP.xml &&
  jing -c "$UCM_HOME"/SCHEMA--UCM--1-1-1.rnc
  "$UCM_HOME"/TMP.xml && rm "$UCM_HOME"/TMP.xml
  ```

- Perform the NVDL validation of the specified UCM example (Requires: xmllint and jing):

  ```
  xmllint --xinclude "$UCM_HOME"/Examples/"$UCM_EXAMPLE"
  --output "$UCM_HOME"/TMP.xml &&
  jing "$UCM_HOME"/SCHEMA--UCM--NAMESPACES--VALIDATION--1-1-1.nvdl
  "$UCM_HOME"/TMP.xml && rm "$UCM_HOME"/TMP.xml
  ```

- Perform the Schematron validation of the specified UCM example (Requires: xmllint and saxon – available from http://saxon.sourceforge.net):

  ```
  xmllint --xinclude "$UCM_HOME"/Examples/"$UCM_EXAMPLE"
  --output "$UCM_HOME"/TMP.xml &&
  saxon -ext:on -s:"$UCM_HOME"/TMP.xml
  -xsl:"$UCM_HOME"/SCHEMA--UCM--RULES--VALIDATION--1-1-1.xsl
  ```

- Perform the Schematron batch validation of selected UCM examples (Requires: xmllint and saxon):

  - For each selected example include the external parts (e.g. necessary definitions) by processing the XInclude inclusions:

    ```
    xmllint --xinclude "$UCM_HOME"/Examples/"$UCM_EXAMPLE"
    --output "$UCM_HOME"/Examples/Input/"$UCM_EXAMPLE"
    ```

  - Execute the Schematron batch validation and output the validation report:

    ```
    saxon -ext:on -s:"$UCM_HOME"/Examples/Input
    -xsl:"$UCM_HOME"/SCHEMA--UCM--RULES--VALIDATION--1-1-1.svrl
    -o:"$UCM_HOME"/Examples/Output
    ```

All commands were tested on Debian 8 and should work on similar Unix-based operating systems. On Windows and other operating systems commands should work after necessary syntax changes provided all required software is correctly installed.

In addition to UCM specifications, at http://www.universalchemicalmarkup.org, we also offer our online UCM VIEWER for an interactive preview of presented UCM examples.

ADDITIONAL FILE 2 – UCM DOCUMENTATION

This additional file contains documentation for all UCM attributes and UCM elements. We also offer the online version of UCM documentation at http://www.universalchemicalmarkup.org.

UCM attributes

Attribute charge

The formal charge on a *node* or *structure* element expressed as a decimal multiple of elementary charge. This attribute is optional and the value ”0” must be assumed if the attribute is not present.

The sum of charges of the *node* or *structure* child elements must be equal to the charge value on the parent *structure* element. Also the charge of the *node* element must correspond to the number of proton and electron particles inside that element.

Parents: node, structure

Attribute counts

One or more whitespace separated non-negative integer numbers, which express the counts of the entity represented by an UCM element with this attribute. This attribute is mandatory on *particle* elements.

For a *particle* element it can store the number of represented particles. More than one number specifying particle count can be used only if the parent *particle* element has the *type* attribute with the value of ”N”.

Parents: particle

Attribute format

The format of *define* and *structure* elements. This attribute is mandatory.

The following values are currently enabled:

- On a *define* element:
  - ”UCM” - Universal Chemical Markup
  - ”UNITSML” - Units Markup Language
  - ”BIBTEXML” - BibTeX Markup Language
- On a *structure* element:
  - ”UCM” - Universal Chemical Markup
  - ”IUPAC-PREFERRED-NAME-U” - Preferred IUPAC Name
  - ”IUPAC-GENERAL-NAME” - General IUPAC Name
  - ”CA-INDEX-NAME” - Chemical Abstracts Index Name
  - ”CAS-RN-U” - Chemical Abstracts Service Registry Number
  - ”REAXYS-RN-U” - Reaxys Registry Number
  - ”CHEMSPIDER-ID-U” - ChemSpider ID Number
  - ”PUBCHEM-CID-U” - PubChem Compound ID Number
  - ”PUBCHEM-SID” - PubChem Substance ID Number
  - ”INCHI” - InChI (International Chemical Identifier)
  - ”INCHI-KEY” - InChIKey
  - ”S-INCHI-U” - Standard InChI
  - ”S-INCHI-KEY” - Standard InChIKey
  - ”SMILES” - SMILES (Simplified Molecular Input Line Entry System)
  - ”SMARTS” - SMILES Arbitrary Target Specification
  - ”SLN” - SYBYL Line Notation

If the format attribute value is ”UCM” the child elements inside the element with this attribute must be from UCM namespace and must conform to the *version* of UCM implementation specified on the *ucm* root element (only UCM 1-1-1 is currently available). For the ”BIBTEXML” and ”UNITSML” values of the format attribute the child elements of this attribute's parent element must be from BibTeXML or UnitsML namespace and must conform to the BibTeXML or UnitsML implementation associated with the UCM *version* specified on the *ucm* root element (currently the BibTeXML 1.0 Extended and UnitsML 1.0 are associated with UCM 1-1-1). In the case of other format attribute values the content of the element with this attribute must match regular expression patterns, which specify the characters enabled for the particular format by the *version* of UCM implementation specified on the *ucm* root element.

The *structure* formats with the suffix ”-U” are unique in a sense that for one unique chemical structure the given format provides at most one unique representation. The *structure* formats without the suffix ”-U” can provide more than one unique representation for one unique chemical structure, or can provide one unique representation for more than one unique chemical structure.

Parents: define, structure

Attribute fractions

Two or more whitespace separated decimal numbers between zero and one expressing the ratio for UCM elements with attributes that store the specific lists of values. The sum of all decimal numbers in the attribute value must be equal to one. This attribute is mandatory on *share* elements. On *particle* elements the usage of this attribute depends on the context in which these elements are used.

For a *share* element this attribute stores the sharing ratio for each id reference representing a *node* element that shares the bonding electrons.

For a *particle* element with more than one number in the value of its *counts* attribute this attribute stores the occurrence ratio of each particle count.

Parents: particle, share

Attribute id

The unique id for UCM elements. It must start with one or more letters (lower or upper case), followed by zero or more patterns consisting of an optional hyphen or underscore and one or more letters (lower or upper case) or digits. Possible values are for example: ”A-1”, ”Structure-1-V13” or ”H2O_G”. This attribute is mandatory on *bond*, *node*, *point*, *property* and *structure* elements and optional on *join*, *particle*, *share*, *stereo*, *ucm* and *values* elements. On *description* elements the usage of this attribute depends on the context in which these elements are used.

Parents: bond, define, description, join, node, particle, point, property, share, stereo, structure, ucm, values

Attribute idrefs

One or more whitespace separated id references to UCM elements. The attribute must contain each id reference at most once. This attribute is mandatory on *join* and *stereo* elements. On *bond*, *description*, *node*, *particle* and *property* elements usage of this attribute depends on the context in which these elements are used.

Parents: bond, description, join, node, particle, property, share, stereo

Attribute litrefs

One or more whitespace separated id references to the literature for a *description* element. This attribute is optional.

The value of the litrefs attribute must refer to one or more literature references defined using BibTeXML inside a *define* element. The attribute must contain each id reference at most once.

Parents: description

Attribute order

The formal bond order for a *bond* element. This attribute is mandatory.

The following values are currently enabled:

- ”PS” - Partial single bond (One bonding electron must be specified using a *particle* element.)
- ”S” - Single bond (Two bonding electrons may be specified using *particle* elements.)
- ”PD” - Partial double bond (Three bonding electrons must be specified using *particle* elements.)
- ”D” - Double bond (Four bonding electrons may be specified using *particle* elements.)
- ”PT” - Partial triple bond (Five bonding electrons must be specified using *particle* elements.)
- ”T” - Triple bond (Six bonding electrons may be specified using *particle* elements.)
- ”PQ” - Partial quadruple bond (Seven bonding electrons must be specified using *particle* elements.)
- ”Q” - Quadruple bond (Eight bonding electrons may be specified using *particle* elements.)
- ”A” - Aromatic bond (Bonding electrons must be specified using *particle* elements.)
- ”DL” - Delocalized or other bond (Bonding electrons must be specified using *particle* elements.)
- ”I” - Ionic bond
- ”H” - Hydrogen bond
- ”DIP” - Dipole or other interaction

Number of bonding electrons specified using *particle* elements inside the *bond* element must correspond with its order attribute value. Electrons participating in the *bond* must be specified if the order attribute has one of the values ”PS”, ”PD”, ”PT”, ”PQ”, ”A” or ”DL”. There must not be any *particle* elements inside the *bond* element if the order attribute has one of the values ”I”, ”H” or ”DIP”. For the other values of the order attribute specifying electrons participating in the *bond* is optional.

Parents: bond

Attribute quantity

The quantity expressed in a *property* element. Usage of this attribute depends on the context in which the parent *property* element is used.

The value of the quantity attribute must refer to the unique xml:id of a quantity defined using UnitsML inside a *define* element.

Parents: property

Attribute sense

The sense sign of a *stereo* element. This attribute is mandatory.

The interpretation of the sense attribute depends on the position of the parent *stereo* element and on the sequence and number of id references specified on that *stereo* element. There are seven possible contexts with different interpretation.

For the *stereo* element, which is inside a *node* element and has the *idrefs* attribute with four id references, describing the stereochemistry of a chirality centre, the following applies. If the rotation from the highest to lower priority substituents is clockwise, the sense sign value must be ”+” (and if the rotation is counterclockwise, the sense sign value must be ”-”). The lowest priority substituent is always assumed to be pointed away from the observer. If the priority of substituents is assigned according to the Cahn-Ingold-Prelog system of priority rules, then the sense sign value ”+” corresponds to the ”R” and the ”-” to the ”S” configuration of substituents on the *node* element.

For the *stereo* element, which is inside a *bond* element and has the *idrefs* attribute with four id references, describing the stereochemistry of a bond, the following applies. If both higher priority substituents are on the same side of the reference plane, the sense sign value must be ”+” (and if both higher priority substituents are on the opposite side, the sense sign value must be ”-”). For a double bond, the reference plane contains nodes participating in the *bond* and is perpendicular to the plane containing these nodes and the nodes directly bonded to them. For a bond in cyclic compounds, the reference plane is the plane to which the cycle skeleton approximates. If the priority of substituents is assigned according to the Cahn-Ingold-Prelog system of priority rules, then the sense sign value ”+” corresponds to the ”Z” and the ”-” to the ”E” configuration of substituents on the *bond* element.

For the *stereo* element, which is inside a *node* element and has the *idrefs* attribute with five id references, describing the stereochemistry of a square planar complex, the following applies. If the line connecting both higher or both lower priority substituents does not go approximately through the central *node* element, the sense sign value must be ”+” (and if the line goes approximately through the central *node* element, the sense sign value must be ”-”). The value ”+” corresponds to the ”cis” and the ”-” to the ”trans” configuration of substituents on the *node* element.

For the *stereo* element, which is inside a *node* element and has the *idrefs* attribute with seven id references, describing the stereochemistry of an octahedral complex, the following applies. If the reference planes defined by all three higher and all three lower priority substituents are approximately parallel and do not contain the central *node* element, the sense sign value must be ”+” (and if the reference planes are approximately perpendicular and approximately contain the central *node* element, the sense sign value must be ”-”). The value ”+” corresponds to the ”fac” and the ”-” to the ”mer” configuration of substituents on the *node* element.

For the *stereo* element, which is inside a *structure* element and has the *idrefs* attribute with six id references, describing the stereochemistry of a chiral axis, the following applies. If the rotation around the axis from the highest to lower priority substituents is clockwise, the sense sign value must be ”+” (and if the rotation is counterclockwise, the sense sign value must be ”-”). If the priority of substituents on the axis is assigned according to the Cahn-Ingold-Prelog system of priority rules, than the sense sign value ”+” corresponds to the ”R” and the ”-” to the ”S” configuration of substituents inside the *structure* element.

For the *stereo* element, which is inside a *structure* element and has the *idrefs* attribute with five id references, describing the twist conformation of a bidentate ligand, the following applies. Using the reference plane described in the *stereo* element documentation (for this context), if the first *node* element that forms the twist is above the reference plane, the sense sign value must be ”+” (and if the first *node* element that forms the twist is below the reference plane, the sense sign value must be ”-”). The first *node* element forming the twist is seen by the observer on the left side. It assumed that the ligand structure is oriented towards the observer and horizontally (seen from the observer point of view). The sense sign value ”+” corresponds to the ”delta” and the ”-” to the ”lambda” twist ligand conformation inside the *structure* element.

For the *stereo* element, which is inside a *structure* element and has the *idrefs* attribute with seven id references, describing the absolute configuration of three bidentate ligands, the following applies. Using reference planes described in the *stereo* element documentation (for this context), if the *node* elements in the first reference plane are to the right from the *node* elements in the second reference plane, when observing *node* elements from the same ligand structure, the sense sign value must be ”+” (and if the *node* elements in the first reference plane are to the left from the *node* elements in the second reference plane, the sense sign value must be ”-”). It is assumed that the observed ligand structure is oriented towards the observer in such a way that the first reference plane is above and the second reference plane is below the central *node* element (seen from the observer point of view). The sense sign value ”+” corresponds to the ”Delta” and the ”-” to the ”Lambda” absolute configuration inside a *structure* element.

The *stereo* element documentation describes how to order id references in the *idrefs* attribute based on what these id references denote.

Parents: stereo

Attribute type

The type of *particle*, *property* and *structure* elements. This attribute is mandatory on *particle* and *structure* elements. On *property* elements the usage of this attribute depends on the context in which the *property* element is used.

The following values are currently enabled:

- On a *particle* element:
  - ”P” - Proton
  - ”N” - Neutron
  - ”E” - Electron
  - ”BE” - Bonding electron (The electron defined to be used in a *bond* element.)
  - ”NBE” - Non-bonding electron (The electron defined not to be used in any *bond* element.)
- On a *property* element:
  - ”PR” - Property (For the *property* element that describes measured or calculated property.)
  - ”CN” - Condition (For the *property* element that describes the required condition for the *values* of the parent *property* element.)
  - ”ER” - Error (For the *property* element that describes errors in *values* of the parent *property* element.)
- On a *structure* element:
  - ”ST” - Structure
  - ”SBST” - Substructure
  - ”STQR” - Structure query
  - ”STID” - Structure identifier

The enabled values of this attribute also depend on the position of its parent element.

On *particle* elements there are two possible contexts with different enabled values. For the *particle* element inside a *bond* element only the ”BE” value of the type attribute is enabled. For the *particle* element inside a *node* element the following applies. The parent *node* element cannot contain more than one *particle* child element with the same value of the type attribute. If the parent *node* element contains the *particle* child element that has the ”NBE” value of the type attribute, it must also contain the *particle* element with the ”BE” value of the type attribute. If the parent *node* element contains the *particle* child element that has the ”BE” or ”NBE” value of the type attribute, it must not contain any *particle* element with the ”E” value of the type attribute.

On *property* elements if the type attribute has the value ”PR”, its parent *property* element must be in a *bond*, *node*, *particle*, *point* or *structure* element or in a *define* element. If the type attribute has the value ”CN”, its parent *property* element must be inside another *property* element or in a *define* element. If the type attribute has the value ”ER”, its parent *property* element must be inside another *property* element.

On *structure* elements there are again two possible contexts with different enabled values. For the *structure* elements that are the children of an *ucm* element the type attribute must have the ”ST” or ”STQR” value. For the *structure* elements inside another *structure* element the enabled values of the type attribute are just ”SBST” or ”STID”.

Parents: particle, property, structure

Attribute version

The version of UCM used in the *ucm* element. Version number format is X-Y-Z, where X is major version (adds new parts and removes or changes present parts if necessary), Y is extended version (adds new parts) and Z is bugfix version (fixes small bugs without adding new parts or changing present parts). Currently the only enabled value is ”1-1-1”. This attribute is mandatory.

Parents: ucm

Attribute x

The ”x” coordinate of a *node* or *point* element in 3-dimensional space expressed in nanometers as a decimal number. This attribute is mandatory on a *point* element. On a *node* element this attribute is optional and the value must be calculated if the attribute is not present. Note that if the ”x” coordinate is specified ”*y*” and ”*z*” coordinates must be specified too.

Parents: node, point

Attribute y

The ”y” coordinate of a *node* or *point* element in 3-dimensional space expressed in nanometers as a decimal number. This attribute is mandatory on a *point* element. On a *node* element this attribute is optional and the value must be calculated if the attribute is not present. Note that if the ”y” coordinate is specified ”*x*” and ”*z*” coordinates must be specified too.

Parents: node, point

Attribute z

The ”z” coordinate of a *node* or *point* element in 3-dimensional space expressed in nanometers as a decimal number. This attribute is mandatory on a *point* element. On a *node* element this attribute is optional and the value must be calculated if the attribute is not present. Note that if the ”z” coordinate is specified ”*x*” and ”*y*” coordinates must be specified too.

Parents: node, point

UCM elements

Element bond

A container for information about a bond in a chemical *structure*. The bond element can represent various bonds from simple ones that connect two *node* elements to delocalized or other bonds, in which multiple *node* or *point* elements participate.

There are two possible contexts where different attributes and child elements are enabled for this element.

For the bond element without the *idrefs* attribute the following applies. It is mandatory to specify the value of the *id* and *order* attribute, but the *idrefs* attribute must be omitted. All id references must be specified using one or more *join* child elements. The *particle* element(s) must represent enough bonding electrons, which can be used by all bond elements that refer to this element. This element must contain zero or one *description* element as the first child followed by zero or more *property*, one or more *join* and zero or more *particle* child elements, and by zero or one *stereo* child element (the sequence of child elements is mandatory).

For the bond element with the *idrefs* attribute the following applies. It is mandatory to specify the value of the *id*, *idrefs* and *order* attribute. The value of the *idrefs* attribute must contain two id references. Each id reference must refer to the *node* or *point* element inside a *structure* element. If electrons participating in the bond are not expressed inside this bond element by the *particle* element(s) each id reference must refer only to the *node* element inside a *structure* element. The *particle* element(s) must represent enough bonding electrons, which can be used by all bond elements that refer to this element. This element must contain zero or one *description* element as the first child followed by zero or more *property* and *particle* child elements, and by zero or one *stereo* child element (the sequence of child elements is mandatory and *join* child elements must be omitted). Usage of this context is mandatory when the bond element has the *order* attribute with the value ”PS”, ”S”, ”PD”, ”D”, ”PT”, ”T”, ”PQ” or ”Q”.

Attributes: id, idrefs, order
Children: description, join, particle, property, stereo
Parents: structure

Element define

A container for various definitions.

It is optional to specify the value of the *id* attribute and mandatory to specify the *format* for this element. Then, inside this element UCM can be used to define descriptions, nodes and properties;

UnitsML can be used to define quantities with appropriate units; and literature references can be defined using BibTeXML.

This element can contain UCM, UnitsML or BibTeXML content, which must use the appropriate namespace. The UCM namespace is ”http://www.universalchemicalmarkup.org”, UnitsML namespace is ”urn:oasis:names:tc:unitsml:schema:xsd:UnitsMLSchema-1.0” and BibTeXML namespace is ”http://bibtexml.sf.net/”. Usually the most advantageous approach is to specify these using namespace prefixes on the *ucm* element or on relevant define elements. The UCM content is restricted to zero or more *description*, *property* and *node* child elements (the sequence of child elements is mandatory). The UnitsML and BibTeXML content must be valid according to the UnitsML or BibTeXML schema.

Attributes: format, id
Children: description, node, property
Parents: ucm

Element description

A container for the description with optional literature reference(s).

There are two possible contexts where attributes are enabled differently for this element.

For the description element that is a child of the *define* element the following applies. It is mandatory to specify the value of the *id* attribute and optional to specify one or more id references using the *idrefs* or *litrefs* attribute.

For the description element that is not a child of the *define* element the usage of *id*, *idrefs* and *litrefs* attributes is optional.

In both contexts each id reference in the *idrefs* attribute must refer to the external description (i.e. the description element inside a *define* element), while each id reference in the *litrefs* attribute must refer to the literature reference defined using BibTeXML inside a *define* element.

This element can contain a simple plain text or XHTML content, which must use the ”http://www.w3.org/1999/xhtml” namespace. Usually the most advantageous approach is to specify it using a namespace prefix on the *ucm* element. The XHTML content must be valid according to the XHTML schema.

Attributes: id, idrefs, litrefs
Parents: bond, define, node, particle, point, property, share, stereo, structure, ucm

Element join

A container for information about how to join *node* or *point* elements that participate in the *bond*.

It is optional to specify the value of the *id* attribute and mandatory to specify two or more id references using the *idrefs* attribute. Each id reference must refer to the *node* or *point* element inside a *structure* element. If electrons participating in the bond are not expressed inside the parent *bond* element by the *particle* element(s) each id reference must refer only to the *node* element inside a *structure* element. The value of this element must specify the interpretation of id references in the *idrefs* attribute.

The following values are currently enabled:

- ”SQ” - Sequential interpretation
- ”CC” - Cyclic interpretation
- ”CT” - Centered interpretation

The sequential interpretation means that the *node* or *point* elements referenced by the neighboring id references are bonded by the bond expressed in the parent *bond* element. The cyclic interpretation only extends the sequential interpretation by regarding also the first and last id reference as neighboring. The centered interpretation means the first id reference refers to the *node* or *point* element that is bonded, by the bond expressed in the parent *bond* element, to all *node* or *point* elements specified by the remaining id references.

Attributes: id, idrefs
Parents: bond

Element node

A container for information about a chemical node that represents a monoatomic particle (usually an atom or a monoatomic ion) composed of protons, neutrons or electrons.

There are three possible contexts where different attributes and child elements are enabled for this element.

For the node element inside a *define* element the following applies. It is mandatory to specify the value of the *id* attribute, but the *idrefs* and *charge* attribute must be omitted as well as coordinates (”*x*”, ”*y*”, ”*z*”) in 3-dimensional space. This element must contain zero or one *description* element as the first child followed by zero or more *property* and one or more *particle* child elements (the sequence of child elements is mandatory and the *stereo* child element must be omitted). The *particle* child elements must define the number of protons, neutrons and electrons.

For the node element that is inside a *structure* element and has the *idrefs* attribute, the following applies. It is mandatory to specify the value of the attribute *id*, *idrefs* and *charge* (if not zero), and optional to specify coordinates (”*x*”, ”*y*”, ”*z*”) in 3-dimensional space. The value of the *idrefs* attribute must be one id reference that refers to the node definition (i.e. the node element inside a *define* element). The node definition must contain enough bonding electrons (expressed by the *particle* element), which can be used by all bonds of the node element that uses this definition. This element must contain zero or one *description* element as the first child followed by zero or more *property* child elements and by zero or one *stereo* child element (the sequence of child elements is mandatory and *particle* child elements must be omitted).

For the node element that is inside a *structure* element and without the *idrefs* attribute, the following applies. It is mandatory to specify the value of the attribute *id* and *charge* (if not zero), and optional to specify coordinates (”*x*”, ”*y*”, ”*z*”) in 3-dimensional space, but the *idrefs* attribute must be omitted. Inside the element must be enough bonding electrons (expressed by the *particle* element), which can be used by all bonds of this element. This element must contain zero or one *description* element as the first child followed by zero or more *property* and one or more *particle* child elements, and by zero or one *stereo* child element (the sequence of child elements is mandatory). The *particle* child elements must define the number of protons, neutrons and electrons.

Attributes: charge, id, idrefs, x, y, z
Children: description, particle, property, stereo
Parents: define, structure

Element particle

A container for information about one or more subatomic particles.

There are two possible contexts where different attributes and child elements are enabled for this element.

For the particle element inside a *bond* element the following applies. It is optional to specify the value of the *id* attribute and mandatory to specify one id reference using the *idrefs* attribute. The id reference must refer to the *bond* or *structure* element or to the *node* element inside a *structure* element. The referenced *bond*, *structure* or *node* element must contain enough subatomic particles (of the given *type*), which can be used by the particle element. In addition it is also mandatory to provide the value of the *type* and *counts* attribute, but the *fractions* attribute must be omitted. This element must contain zero or one *description* element as the first child followed by zero or more *property* and *share* child elements (the sequence of child elements is mandatory).

For the particle element inside a *node* element the following applies. It is optional to specify the value of the *id* attribute and mandatory to specify the value of the *type* and *counts* attribute, but the *idrefs* attribute must be omitted. If the *type* attribute value is ”N” and the *counts* attribute contains more than one particle count in its value it is mandatory to specify the occurrence ratio of each count using the *fractions* attribute. In other cases the *fractions* attribute must be omitted. This element must contain zero or one *description* element as the first child followed by zero or more *property* child elements (the sequence of child elements is mandatory and *share* child elements must be omitted).

Attributes: counts, fractions, id, idrefs, type
Children: description, property, share
Parents: bond, node

Element point

A container for information about a point in a chemical *structure*.

It is mandatory to specify the value of the *id* attribute as well as coordinates (”*x*”, ”*y*”, ”*z*”) in 3-dimensional space.

The intended usage of the point element is to describe important places in the *structure* that are outside the scope of *node* elements. An example may be the places to which other *node* or *structure* (via its point child element) is or can be bonded.

This element must contain zero or one *description* element as the first child followed by zero or more *property* child elements (the sequence of child elements is mandatory).

Attributes: id, x, y, z
Children: description, property
Parents: structure

Element property

A container for information about a measured or calculated property. The property element can be used in a *bond*, *node*, *particle*, *point* or *structure* element or for a property definition in a *define* element.

There are two possible contexts where different attributes and child elements are enabled for this element.

For the property element inside a *define* element, or the property element that does not have the *idrefs* attribute, the following applies. It is mandatory to specify the value of the *id*, *type* and *quantity* attribute, but the *idrefs* attribute must be omitted. This element must contain zero or one *description* element as the first child followed by zero or more property (it can be nested) child elements and one *values* child element (the sequence of child elements is mandatory).

For the property element that has the *idrefs* attribute, the following applies. It is mandatory to specify the value of the *id* and *idrefs* attribute, but the *type* and *quantity* attribute must be omitted. The value of the *idrefs* attribute must be one id reference referencing the property definition (i.e. the property element inside a *define* element). This element must contain zero or one *description* element as the only child (property and *values* child elements must be omitted).

Attributes: id, idrefs, quantity, type
Children: description, property, values
Parents: bond, define, node, particle, point, property, structure

Element share

A container for information about how bonding electrons are shared between *node* elements that participate in the *bond*.

It is optional to specify the value of the *id* attribute and mandatory to specify two or more id references using the *idrefs* attribute. Each id reference must refer to the *node* element inside a *structure* element. For each id reference it is also mandatory to specify the sharing ratio using the *fractions* attribute.

This element must contain zero or one *description* child element.

Attributes: fractions, id, idrefs
Children: description
Parents: particle

Element stereo

A container for describing the stereochemistry of a *structure* element or around a *node* or *bond* element.

It is optional to specify the value of the *id* attribute and mandatory to specify the id references using the *idrefs* attribute. Each id reference must refer to the *node* or *point* element inside a *structure* element. In addition it is also mandatory to provide the value of the *sense* attribute.

The interpretation of the stereo element depends on the sequence and number of id references. There are five possible contexts with different interpretation.

For the stereo element, which is inside a *node* or *bond* element and has the *idrefs* attribute with four id references, describing the stereochemistry of a chirality centre or bond, the following applies. The id references indicate substituents on the *node* or *bond* element ordered by descending priority.

For the stereo element, which is inside a *node* element and has the *idrefs* attribute with five or seven id references, describing the stereochemistry of a square planar or octahedral complex, the following applies. The first id reference denotes the central *node* element and the remaining id references indicate substituents on the central *node* element ordered by descending priority.

For the stereo element, which is inside a *structure* element and has the *idrefs* attribute with six id references, describing the stereochemistry of a chiral axis, the following applies. The first two id references refer to the *node* or *point* elements defining the axis and the remaining id references indicate substituents on the axis ordered by descending priority.

For the stereo element, which is inside a *structure* element and has the *idrefs* attribute with five id references, describing the twist conformation of a bidentate ligand, the following applies. The first id reference denotes the central *node* element, the next two id references refer to the *node* elements forming the reference plane together with the central *node* element and the last two id references refer to the *node* elements forming the twist. The sequence of id references for *node* elements forming the twist is from left to right side (as seen from the observer point of view described in the *sense* attribute documentation).

For the stereo element, which is inside a *structure* element and has the *idrefs* attribute with seven id references, describing the absolute configuration of three bidentate ligands, the following applies. The first id reference denotes the central *node* element, the next three id references refer to the *node* elements forming the first reference plane (above the central *node* element) and the last three id references refer to the *node* elements forming the second reference plane (below the central *node* element). The first reference plane can be chosen arbitrarily.

The priority of substituents should be (thus it is not mandatory) assigned according to the Cahn-Ingold-Prelog system of priority rules. The *sense* attribute documentation describes how to assign the value describing the stereo configuration based on the sequence of id references and the position of referenced elements. Note that the stereo configuration described by this element should correspond with the 3-dimensional coordinates (”*x*”, ”*y*”, ”*z*”) of referenced elements, but UCM validation currently does not verify this. It is assumed that software tools working with the stereochemistry in UCM should implement the necessary checks and if required also algorithms that can asign the stereo configuration using the 3-dimensional coordinates (”*x*”, ”*y*”, ”*z*”) automatically or semiautomatically (e.g. when the user decides the substituents priority manually).

This element must contain zero or one *description* child element.

Attributes: id, idrefs, sense
Children: description
Parents: bond, node, structure

Element structure

A container for the chemical structure. The structure element can represent a complete chemical structure or just some relevant part of it. Thus, using this element it is possible to express structures and substructures as well as structure queries and identifiers.

There are two possible contexts where different attributes and child elements are enabled for this element.

For the structure element that has the *format* attribute with the value of ”UCM”, the following applies. It is mandatory to specify the value of the *id*, *format* and *type* attribute, and optional to specify the value of the *charge* attribute. This element must contain zero or one *description* element as the first child followed by zero or more structure (it can be nested), *property*, *node*, *bond*, *point* and *stereo* child elements (the sequence of child elements is mandatory).

For the structure element that has the *format* attribute with other value than ”UCM”, the following applies. It is mandatory to specify the value of the *id*, *format* and *type* attribute, but the *charge* attribute must be omitted. The content of this element must conform to regular expression patterns, which check whether the used characters and content structures follow the basic specifications of the given *format*. Note that for ”SMILES”, ”SMARTS” and ”SLN” *format* values the chemical reactions syntax is not enabled in structure elements. Additionally, for the content of structure elements using other *format* value than ”UCM” the validation in UCM does not attempt to verify that it represents chemically correct or otherwise valid data. It is assumed that software tools working with such data (e.g. structure queries and identifiers) should implement the necessary algorithms to check and parse the data and where required and possible obtain the corresponding chemical structures. These structures could be later stored in UCM format (i.e. in the structure elements with the *format* attribute value of ”UCM”).

In both contexts the *type* attribute value further restricts the values of the *format* attribute, which in turn restricts the possible content of this element. If the *type* attribute has the value of ”ST”, the *format* attribute can have any value listed in its documentation as enabled on structure elements. In other words just the ”UNITSML” and ”BIBTEXML” values are not enabled. If the *type* attribute value is ”SBST”, the *format* attribute must have the value of ”UCM”. In the case of ”STQR” *type* value the *format* attribute can have any value enabled for structure elements except for the ”UCM” value. Finally, for the *type* value ”STID” the *format* attribute can have any value enabled for structure elements with the exception of ”UCM”, ”SMILES”, ”SMARTS” and ”SLN” values.

Attributes: charge, format, id, type
Children: bond, description, node, point, property, stereo, structure
Parents: structure, ucm

Element ucm

The root element of UCM. The ucm element is the main container for all UCM attributes and elements.

It is optional to specify the value of the *id* attribute and mandatory to specify the *version* and namespace of UCM used for this element. Usually the most advantageous approach is to specify the UCM namespace ”http://www.universalchemicalmarkup.org” as the default one using the xmlns attribute on the ucm element. Optionally also other namespace prefixes can be specified on the ucm element. This is useful to enable the XHTML usage inside *description* elements or BibTeXML and UnitsML usage inside *define* elements.

This element must contain zero or one *description* element as the first child followed by zero or more *define* and *structure* child elements (the sequence of child elements is mandatory).

Note that special xml:base and xml:lang attributes may be used on all UCM elements. The xml:base attribute is utilized by the xi:include mechanism, as it provides the base URI (Uniform Resource Identifier) for interpreting any relative URI in the scope of the parent element. The xml:lang attribute denotes a language code for the natural language in the scope of the parent element. The values of both xml:base and xml:lang attributes are inherited. Other special attributes from XML namespace ”http://www.w3.org/XML/1998/namespace.html” are not currently supported, but if the need arises we may add them. Special attributes from XML namespace are implemented in UCM according to their definitions in the official schema, which is available at ”http://www.w3.org/2009/01/xml.xsd”.

Attributes: id, version
Children: define, description, structure

Element values

A container for numeric values representing a measured or calculated *property*.

It is optional to specify the value of the *id* attribute. Then, this element must contain one or more whitespace separated double precision floating point numbers. The special values ”-INF” (negative infinity), ”INF” (positive infinity) and ”NaN” (not a number) are supported as well as optional scientific notation. The decimal separator must be ”.” and no thousands separator may be used.

Attributes: id
Parents: property