StructurePendium
Technologies GmbH

International chemical identifier for reactions

(RInChI)

Version 1.00

Amtsgericht Essen: HRB 22630 · Geschäftsführer Dr. Gerd Blanke · Konto Nr. (IBAN) DE68360501050008357741, Sparkasse Essen
(BIC: SPESDE3EXXX) · Steuernummer 112/5706/0199 beim Finanzamt Essen-Süd · U.St.-Nr. (VAT-ID): DE275049787

RInChI Version

Revision	Modifications	Date
Version 0.03.1	First test candidate of international chemical identifier for reactions, version 1.00 based on InChI version 1.04	28-Oct-2016
Version 0.03.8	Modifications introduced up to 28-Nov-2017	28-Nov-2016
Version 1.00.0	First RInChI release, release version 1.00	14-Mar-2017

Download

Revision	URL
Version 1.00.0	http://www.inchi-trust.org/downloads/

Document history

Version	Primary author	Modifications	Revision date
Draft 0.1	Gerd Blanke		10-Aug-2016
Draft 0.2	Gerd Blanke	Comments by Jan Holst Jensen added	26-Aug-2016
Draft 0.3	Günter Grethe, Gerd Blanke	1st Review by Günter Grethe	25-Sep-2016
Draft 0.4	Günter Grethe, Gerd Blanke	2nd Review by Günter Grethe	06-Oct-2016
Draft 0.5	Gerd Blanke	3rd Review by Jonathan Goodman, Hans Kraut	17-Oct-2016
Draft 0.6	Gerd Blanke	4th Review by Jan Holst Jensen	21-Oct-2016
Draft 0.7	Gerd Blanke	Preparation of test release	28-Oct-2016
1.00	Gerd Blanke	First release	14-Mar-2017

Content

Introduction

In 2011, the first prototype of RInChI was introduced as version 0.02 based on Python programming (Grethe, Goodman, & Allen, 2013). To make the resulting libraries comparable with InChI deliveries, to increase the search ability of RInChIKeys in the Web by the introduction of the Web-RInChIKey and to fix a number of bugs and development gaps in the former version this new version (version number 1.00) has been created. This version is supposed to be published for general usage by the InChI Trust.[1]

RInChI format

The RInChI format is a hierarchical, layered description of a reaction with different levels based on the Standard InChI representation (version 1.04) of each structural component participating in the reaction.

The version 1.00 of RInChI consists of six layers:

- The **first layer** is fixed, describing the versions used for the RInChI calculations. It starts with the acronym "RInChI", followed by an equality sign "=", the current RInChI version number ("1.00"), and the number of the InChI version used to generate the InChIs of the structural components of the reactions. InChIs are created by using the standard InChI calculator, version 1.04, leaving the acronym "1S" separated by a dot (".") from the RInChI version identifier. The end of the first layer is marked by a slash "/".

  These definitions specify the format of the first layer by "RInChI=1.00.1S/"

- **Layers 2 and 3** consist of InChIs representing the reactants and products of the reaction. The trailing version information "1S" for each InChI is omitted. Within each layer InChIs are separated by an exclamation mark "!". The layers are separated by "<>".

  In the first step, InChIs of all the reactants are ordered alphabetically to a common string in layer 2, while all InChIs of the products are kept alphabetically ordered as string in layer 3. To gain a unique representation, the string of layer 2 is compared with the string of layer 3. If the string of layer 3 is alphabetically seen "less" than that of layer 2, the contents of layer 2 and layer 3 are exchanged and the direction flag of layer 5 is reversed. Else, if the string of layer 2 is alphabetically "less" than the string of layer 3, both layers are kept as they are.[2]

  Note: In the case of half reactions or in those special cases where reactants and/or products are only described by no-structures one or both of these layers may vanish.[3]

- **Layer 4** is built from InChIs of the catalysts, solvents and reagents that are part of the reaction. The version flag "1S" within each InChI is left out. InChIs are sorted alphabetically. Exclamation marks "!" divide multiple InChIs within this layer.

  RInChI does not distinguish the individual roles of these compounds in the reaction (catalyst, solvent, reagent, etc.) and subsumes them as "agents".

  The fourth layer is separated from the third layer by "<>".

  The fourth layer is optional, because agents are not described for all of the reactions.[4]

- The **fifth layer** is the directional identifier. It is separated from group 4 by a slash"/" and begins with "d" followed by a plus sign "+" for forward reactions, "-" for backward reactions, or "=" to

[1] For more details of the changes between the former version 0.02 and the current version, see "Comparison of RInChI Version 0.02 with the current version" in the appendix of this paper.

[2] See reactions below for examples.

[3] See the chapters "No-structures and pseudoatoms representing components that cannot be structurally drawn" and "Half and empty reactions" in the appendix of this document.

[4] Note: If agents occur, the RXN file format is not sufficient. Instead, the reaction must be described in the RD file format that contains catalysts, solvents and other reagents as molfiles following the reaction section in RXN format within the RD file section.

describe an equilibrium reaction. If the fifth layer is empty, the direction of the reaction is unspecified.[5]

Accordingly, the values of the fifth layer are "/d+" (forward reaction), "/d-" (backward reaction), or "/d=" (equilibrium reaction).

If the rules for alphabetical ordering of the layers 2 and 3 enforce an exchange of these two layers, the direction flag must be exchanged as well, i.e. the forward reaction "/d+" becomes a backward one "/d-" and vice versa. The alphabetical re-ordering does not influence the value "/d=" because the equilibrium reaction incorporates forward and backward reaction.

- The **sixth layer** represents the "no-structure flag". It starts with a slash "/" followed by "u" for unknown structures and the counter of unknown structural elements in layer 2, 3 and 4 with each separated by a dash "-". That leads to the format "/u#2-#3-#4" with #2, #3, and #4 as number of unknown structures in the 2nd, 3rd and 4th layer. The 4th layer is only displayed if agents occur in the reaction.

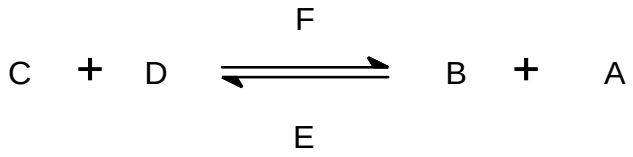
  In case of the reordering of level 2 and 3, the no-structure flag must be adapted accordingly by exchanging the occurrence numbers of the no-structures for the 2nd and 3rd level.

  The "no-structure" flag is optional and only displayed if one of its counters is not zero, i.e. if there is at least one no-structure found in the reaction.

Using italic characters for those layers that do not always contribute to the RInChI, it takes the following format:

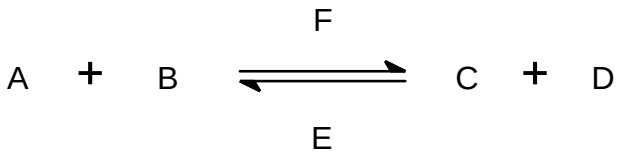
RInChI=1.00.1S/*layer2*<>*layer3*<>*layer4*/d(+,-,=)/u#2-#3-#4

To explain the structure of layers 2, 3 and 4, consider the following equilibrium reaction with compounds C and D as reactants, A and B as products and E and F as agents (catalysts or solvents).



Reaction 1

Note that the following depiction is an equivalent representation of the equilibrium "Reaction 1"



Reaction 2

To build the RInChI of Reaction 1 (and Reaction 2) let's assume that InChI(A) is the InChI describing the chemical structure A, InChI(B) that one of B and so forth. For simplification the InChIs are alphabetically sorted in the order of InChI(A) < InChI(B) < ….

Layer 6

All compounds (A, B, C, D, E, F) are assumed to have a full chemical structure representation, i.e. there are no no-structures given in this reaction. Therefore, the layer 6 (no-structure flag) is not set.

[5] Because RXN and RD files provide a forward reaction per definition, the default value of the 5th layer is "d+". The CT file format (Biovia, 2016) does neither support backward nor equilibrium nor unspecified reaction directions. To handle backward reactions they must be re-written as forward direction, i.e. reactants and products must be exchanged. In case of equilibrium reactions, a special parameter is provided for RInChI calculations that can be accessed programmatically. In the context of the usage of RXN/RD files unspecified reactions are assumed to be rare so that the direction identifier in layer 5 can only be handled manually by deletion.

Layer 5

Layer 5 is determined by the equilibrium reaction as "/d=".

Layer 4

Layer 4 is built from InChIs calculated from the structures representing the agents (catalysts, solvents, reagents) ordered alphabetically and separated by the exclamation mark:

Layer 4: InChI(E)!InChI(F)

Note that the alphabetical sorting of InChIs in group 4 is independent from the actual order of occurrences in Reaction 1 / Reaction 2

Layer 3 and Layer 2

The rules for the set-up of layers 2 and 3 must take into account that Reaction1 and Reaction 2 are equivalent depictions of the same reaction. To build a unique expression, all reactants are first grouped into layer 2 while all products go into layer 3. Using alphabetical sorting of InChIs in each layer with "!" as separator you receive

Layer 2 = InChI(C)!InChI(D)

Layer 3 = InChI(A)!InChI(B)

Layer 5 = /d=

To gain uniqueness, layer 2 and 3 are reordered alphabetically. Because "InChI(A)! InChI(B)" of layer 3 < "InChI(C)! InChI(D)" of layer 2, the content of the 2 layers must be exchanged and the direction indicated in layer 5 must be reversed:

Layer 2 = InChI(A)!InChI(B)

Layer 3 = InChI(C)!InChI(D)

Layer 5 = /d=

Note: In case of a forward reaction with layer 5 = "/d+", the exchange leads to a backward reaction with layer 5 = "/d-". Vice versa, a backward reaction must be described by layer 5 = "/d+" (forward reaction), if you have to exchange layer 2 and 3. Because equilibrium reactions represent a ratio of forward and backward reactions the reverse state of layer 5 continues to be /d= (equilibrium).

Therefore, Reaction 1 gets the following RInChI:

RInChI=1.00.1S/InChI(A)! InChI(B)<>InChI(C)! InChI(D)<>InChI(E)!InChI(F)/d=

Reaction 2:

While the layers 6, 5 and 4 are identical in Reaction 2, layers 2 and 3 are defined as

Layer 2 = InChI(A)!InChI(B)

Layer 3 = InChI(C)!InChI(D)

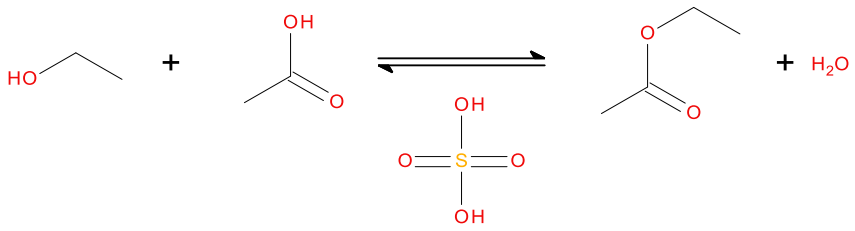
Because the alphabetical sorting shows that "InChI(A)!InChI(B)" < "InChI(C)! InChI(D)", i.e. layer 2 < layer 3, the resorting is not applicable and the layers are kept as they are. The RInChI of Reaction 2 is defined as

RInChI=1.00.1S/InChI(A)!InChI(B)<>InChI(C)!InChI(D)<>InChI(E)!InChI(F)/d=

and therefore identical to the RInChI of Reaction 1.

Examples

Example: Esterification of acetic acid

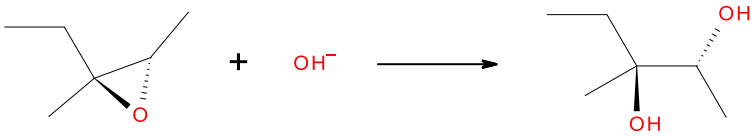


Reaction 3: Esterification

- Layer 6: This layer is omitted because the reaction does not contain any no-structures
- Layer 5: /d= to describe the equilibrium reaction
- Layer 4: The sulfuric acid as catalyst has the InChI=1S/H2O4S/c1-5(2,3)4/h(H2,1,2,3,4). Skipping the version leaves layer 4 with "H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)"
- Layer 3: The layer consists out of the products ethyl acetate (InChI=1S/C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3) and water (InChI=1S/H2O/h1H2). Skipping the version information and sorting alphabetically leaves layer 3 = "C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2"
- Layer 2: The reactants ethanol (InChI=1S/C2H6O/c1-2-3/h3H,2H2,1H3) and acetic acid (InChI=1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)) build the second layer. Omitting the version information and sorting alphabetically, layer 2 string becomes "C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3".[6]
- Alphabetical sorting of the strings for layers 2 and 3 show that the string of layer 2 is "less" than that of layer 3. Therefore, layer 2 and layer 3 must not to be exchanged.
- Layer 1: This layer is defined by "RInChI=1.00.1S/"

That compiles RInChI for Reaction 3: Esterification to "RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d="

Example: Alkaline ring opening



Reaction 4: Ring opening

- Layer 6: Because there is no "no-structure" involved in the reaction, this layer is skipped.
- Layer 5: /d+ to mark the forward reaction
- Layer 4: Layer 4 stays empty because there are no reagents described for the reaction.
- Layer 3: The layer contains the InChI for (2R,3S)-3-methylpentane-2,3-diol calculated to "InChI=1S/C6H14O2/c1-4-6(3,8)5(2)7/h5,7-8H,4H2,1-3H3/t5-,6+/m1/s1"
- Layer 2: By removing the strings "InChI=1S/", the reactants (2S,3S)-2-ethyl-2,3-dimethyl-oxirane with"InChI=1S/C6H12O/c1-4-6(3)5(2)7-6/h5H,4H2,1-3H3/t5-,6-/m0/s1" and hydroxide

[6] The alphabetical sorting reorders the reactants to acetic acid and ethanol

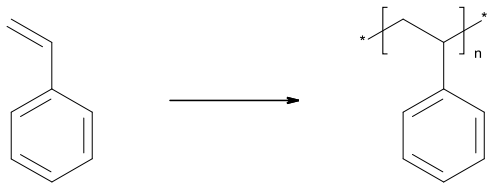
with "InChI=1S/H2O/h1H2/p-1" together with the "!" as separator, this layer contains the string "C6H12O/c1-4-6(3)5(2)7-6/h5H,4H2,1-3H3/t5-,6-/m0/s1!H2O/h1H2/p-1"

- Alphabetical sorting leaves layer 2 and 3 as they are.
- Layer 1: "RInChI=1.00.1S/" is the predefined string of this layer.

The resulting RInChI for the ring opening reaction is given as "RInChI=1.00.1S/C6H12O/c1-4-6(3)5(2)7-6/h5H,4H2,1-3H3/t5-,6-/m0/s1!H2O/h1H2/p-1<>C6H14O2/c1-4-6(3,8)5(2)7/h5,7-8H,4H2,1-3H3/t5-,6+/m1/s1/d+".

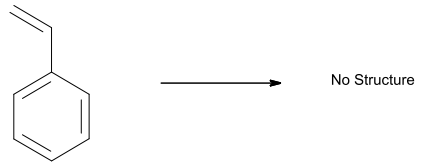
Example: Polymerisation of styrene

There are multiple ways to polymerize styrene to polystyrene. The chemical reaction is described by



Reaction 5: Polymerisation

Because the representation by structure repeating units is not supported by the standard InChI version 1.04 (see below "Unsupported molfile features" in the Appendix), the polystyrene structure must be represented by a no-structure that is described by the alphanumerical data section of the reaction:



Reaction 6: replacement for the representation of the styrene polymer

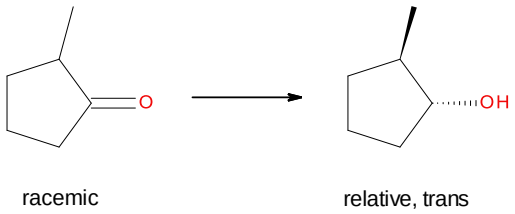
- Layer 6: The no-structure flag must be set to u0-1-0 according to the no-structure found as product.
- Layer 5: /d+ to describe the forward reaction
- Layer 4: The catalyst is not defined. The layer 4 is left empty and vanishes accordingly
- Layer 3: The no-structure of layer 3 is represented by InChI=1S/. The RInChI rules leave an empty string.
- Layer 2: The reactant styrene (InChI=1S/ C8H8/c1-2-8-6-4-3-5-7-8/h2-7H,1H2) defines layer 2 as "C8H8/c1-2-8-6-4-3-5-7-8/h2-7H,1H2".
- The alphabetical sorting of the strings for layer 2 and 3 has to handle the special case that layer 3 is empty. According to the sorting rules, the "empty string" is defined as the smallest entity of the alphanumerical sorting, i.e. layer 3 must take position 2, while layer 2 takes position 3. Hence, in layer 5 the reaction flag must be returned to the backward reaction flag "/d-" and the no-structure-flag of layer 6 must be written as "/u1-0-0"
- Layer 1: This layer is defined by "RInChI=1.00.1S/"

The RInChI for Reaction 5 is written as "RInChI=1.00.1S/<>C8H8/c1-2-8-6-4-3-5-7-8/h2-7H,1H2/d-/u1-0-0".

Special cases for the RInChI creation

Stereochemistry

The standard InChI (Version 1.04) always interprets structures with tetrahedral stereochemistry centers as absolutely defined stereo centers. Relative stereocenters are not handled.[7]



Reaction 7: Stereoselective reduction of cyclopentanones

The stereoselective reduction of cyclopentanones builds two stereocenters that are relative and trans orientated. Standard InChI interprets the compound as having absolute configuration so that the mixture of the relative trans centers cannot be represented by RInChI.

Workaround for the current RInChI / InChI version: do not use explicit stereochemistry for relative stereo centers or draw all compounds with absolute stereo centers that build the mixture that is represented by the relative stereorepresentation. For more information see "Representations of tetrahedral stereocenters by Standard InChIs" in the appendix.

Tautomerisations

The InChI algorithm is optimized to deliver identical InChIs for structures that represent different tautomers of the same structure class. That leads to limitations in the reaction representations that are described in the appendix "Tautomerization reactions".

Half reaction

For half-reactions where the reactant or product site is not defined, the string of the related layer is empty. See the appendix "Half and empty reactions" for more details and examples.

Reactions containing no-structures

No-structures and related pseudoatoms (R, X, A and * atom used standalone[8]) are placeholders for molecules that cannot be represented structurally but are described alphanumerically in the data set. These no-structures are represented by RInChI as "InChI=1S/". According to the rule that "InChI=1S/" is omitted in RInChI, only an empty string is left and a reaction with a no-structure as reactant or product would become identical to an half-reaction. Therefore, the no-structure flag in layer 6 has been introduced to distinguish the no-structures from the "half-reactions". For additional examples see the chapter "No-structures and pseudoatoms representing components that cannot be structurally drawn" in the appendix of this document.

Connectivity of metal atoms

The InChI algorithm ignores the connectivity of metal atoms in chemical structures. In case that the connectivity of an atom metal is changed by a chemical reaction, RInChI may not fully reflect this change. See "Connectivity of metal atoms" in the appendix for more details.

---

[7] The standard InChI uses the parameter /SAbs for the handling of chemical structures with tetrahedral stereo centers, so that relative centers are not correctly seen as absolute isomers. The InChI parameter /SUCF (Use Chiral Flag) interprets the Chiral flag settings in the molfile correctly differentiating between compound with relative and absolute centers.

[8] If any of the following atoms R, X, A or * is found within a molecule, the InChI algorithm returns an error so that RInChI cannot be calculated.

Unsupported molfile features

Generally, query features and V3 molfile elements are not supported by InChI and therefore not available for RInChI. Details can be found in the chapter "Limitations for RInChI caused by InChI restrictions" of the appendix.

RInChI-AuxInfos (RAuxInfos)

As described above, RInChI is based on InChIs of the components participating in the reaction. However, each InChI itself only provides the connectivity information of a molecule while the InChI-AuxInfo of the molecule contains all remaining data necessary to reconstruct the full molecule including atom numbering and atom coordinates.

In order to fully rebuild a reaction from RInChI to the RXN/RD file format, RAuxInfo (RInChI-AuxInfo) has been introduced. RAuxInfo consists of four layers corresponding to the first four[9] layers of RInChI. Layers 2, 3 and 4 of RAuxInfo are compiled from the AuxInfo strings of InChIs following the rules developed for RInChI using the order of the components determined for the RInChI calculation:

- The **first layer** is rigid and contains the version identifier for the RInChI version (V 1.00) and for the version of the InChI AuxInfo that is "1". The separator to the next layer is the slash "/".

  That leaves the first layer with "RAuxInfo=1.00.1/"

- Layers 2 to 4 consist of the InChI-AuxInfos of the components that build layers 2, 3 and 4 in the final stage of the RInChI creation process. To build each layer out of the InChI-AuxInfos, the trailing InChI version number "1/" of each AuxInfo string is omitted. Multiple AuxInfos are separated by exclamation marks ("!"). They are not ordered alphabetically, but the arrangement of the AuxInfos follows the order of InChIs in the related layer, i.e. the first AuxInfo corresponds to the first InChI in RInChI and so forth.

  Layers 2, 3 and 4 are separated from each other by "<>".

  Layer 4 is only displayed if it is not empty. In case it is omitted, the separator "<>" is skipped as well. If layer 4 and 3 are empty, both layers and their separators are left open. Finally, only layer 1 is displayed if there are no InChI-AuxInfos available for layers 2 to 4.[10]

RAuxInfo=1.00.1/*layer2*<>*layer3*<>*layer4*

Examples

Example: Esterification of acetic acid

RInChI from the example "Example: Esterification of acetic acid" is defined by "RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d="

RAuxInfo is defined by taking the AuxInfo of InChIs in order of their appearance in each layer, i.e. layer 2 must contain the AuxInfo for acetic acid and ethanol, layer 3 that of ethyl acetate and water and layer 4 that tof the agent sulfuric acid:

"RAuxInfo=1.00.1/1/N:1,2,3,4/E:(3,4)/rA:4nCCOO/rB:s1;s2;d2;/rC:-3.8549,-.5552,0;-2.8321,.0354,0;-2.8321,1.2168,0;-1.8089,-.5554,0;!0/N:3,2,1/rA:3nOCC/rB:s1;s2;/rC:2.499,-.1614,0;1.4762,.4292,0;.453,-.1615,0;<>0/N:5,1,4,2,6,3/rA:6nCCOCCO/rB:s1;s2;s3;s4;d2;/rC:7.2384,-1.0475,0;8.2613,-.4569,0;8.2613,.7246,0;9.2844,1.3153,0;10.3076,.7246,0;9.2844,-1.0476,0;!0/N:1/rA:1nO/rB:/rC:12.5696,.0354,0;<>1/N:2,3,4,5,1/E:(1,2,3,4)/CRV:5.6/rA:5nSOOOO/rB:s1;s1;d1;d1;/rC:6.4257,-2.7792,0;7.1757,-2.7792,0;5.6757,-2.7792,0;6.4257,-3.5292,0;6.4257,-2.0292,0;"

[9] Layer 5 and 6 of RInChI are inapplicable because they do not contain InChI strings.

[10] For examples see chapter "Half and empty reactions" in the appendix of this document.

Example: Alkaline ring opening

As described above, the RInChI string is given as "RInChI=1.00.1S/C6H12O/c1-4-6(3)5(2)7-6/h5H,4H2,1-3H3/t5-,6-/m0/s1!H2O/h1H2/p-1<>C6H14O2/c1-4-6(3,8)5(2)7/h5,7-8H,4H2,1-3H3/t5-,6+/m1/s1/d+"

The RAuxInfo begins with the version identifier for the RInChI version (V 1.00) and for the InChI Version that is "1" ("RAuxInfo=1.00.1/") followed by AuxInfos of the reactants' InChIs ordered like the InChIs in layer 2 and separated by "!" plus the AuxInfo of the InChi of the product "RAuxInfo=1.00.1/0/N:2,5,7,1,4,3,6/it:im/rA:7cCCCCCOC/rB:s1;s1;s3;s4;P3N4;s3;/rC:1.6851,-6.1198,0;.4352,-6.1248,0;2.3146,-7.1997,0;3.4875,-6.7675,0;4.287,-5.8066,0;3.2753,-7.9993,0;1.3603,-8.0071,0;!1/N:1/rA:1nO-/rB:/rC:7.5313,-6.9069,0;<>0/N:2,6,8,1,4,3,5,7/it:im/rA:8cCCCCOCOC/rB:s1;s1;s3;N4;s4;P3;s3;/rC:13.954,-5.7406,0;12.704,-5.7406,0;14.579,-6.8231,0;15.829,-6.8231,0;16.454,-5.7406,0;16.454,-7.9056,0;14.579,-8.0731,0;13.4965,-7.4481,0;"

Example: Polymerisation of styrene

Because of the no-structure, layer 2 of RInChI for this reaction is empty while the InChI of styrene gives the layer 3. Layer 4 is not defined, while layer 5 characterizes the backward reaction "/d-" and layer 6 counts the no-structure related to layer 2 with "/u1-0-0". That leads to

"RInChI=1.00.1S/<>C8H8/c1-2-8-6-4-3-5-7-8/h2-7H,1H2/d-/u1-0-0"

The AuxInfo of the styrene molecule used in the example is given as AuxInfo=1/0/N:8,7,5,4,6,3,1,2/E:(4,5)(6,7)/rA:8nCCCCCCCC/rB:d1;s2;d3;s4;s1d5;s2;d7;/rC:4.6504,-6.2569,0;5.7332,-5.6321,0; 6.8119,-6.2563,0;6.8119,-7.5066,0;5.7359,-8.1306,0;4.6504,-7.5122,0;5.7332,-4.3828,0;4.6512,-3.7582,0

Based on the order of RInChI, AuxInfo defines layer 3 while layer 2 is left empty, used as a placeholder for the non-existing AuxInfo of the no-structure.

"RAuxInfo=1.00.1/<>0/N:8,7,5,4,6,3,1,2/E:(4,5)(6,7)/rA:8nCCCCCCCC/rB:d1;s2;d3;s4;s1d5;s2;d7;/rC:4.6504,-6.2569,0;5.7332,-5.6321,0;6.8119,-6.2563,0;6.8119,-7.5066,0;5.7359,-8.1306,0;4.6504,-7.5122,0;5.7332,-4.3828,0;4.6512,-3.7582,0;"

Additional information is presented in the chapter "RXN and RD file regeneration from RInChI" in the appendix of this document.

Recalculation of RXN/RD files from RInChIs and RInChI-Auxinfos

Because RInChI is based on InChIs of each of the reaction components and each InChI represents the connectivity table of one component of the reaction, RXN or RD files can be (re)built out of InChIs of a RInChI.

In case RAuxInfo is given, the actual chirality of the molecule, the tautomeric format and the atom coordinates of the reaction components can be rebuilt. Otherwise, any tetrahedral stereochemistry of the reaction components is always returned as relative (without chiral flag), the preferred and not the original tautomeric format is returned and all atom coordinates are set to 0.0000. In the second case, a third party program is needed to recalculate the atom positions of each reaction components. The recalculation may lead to a different atom order around the stereo center while the original stereo bond is kept. That may reverse the parity of the stereo center. For more details, see appendix "RXN and RD file regeneration from RInChI".

A RXN file can be fully reconstructed from the related RInChI and RAuxInfo strings. But the comparison of the original RD file with the recalculated RD file shows that only the structural elements are kept and that the specific roles of the agents (catalysts, solvents, etc.) are lost together with all other alphanumerical information of the original RD file.

For more details, see chapter "RXN and RD file regeneration from RInChI" in the appendix of this document.

RInChIKeys

RInChIKeys are hashed representations of the RInChI strings. The hashing process creates shorter strings that are unique representations of the original (longer) RInChIs. However, RInChIs (and therefore the original RXN and/or RD file) cannot be rebuilt out of the hashed string. That makes RInChIKeys an encrypted unique depiction of chemical reactions especially suitable for database processes and web operations.

For the anticipated usage, three different types of RInChIKeys have been developed:

- Long-RInChIKeys are strings concatenated from the InChIKeys of each reaction component. The length of Long-RInChIKeys is flexible.
- Short-RInChIKeys are created by hashing the major and minor layers of InChIs for each group of RInChI to a fixed length string
- Web-RInChIKeys deduplicate InChIs over all groups and hash all major and minor InChI layers into a fixed length string ignoring the specific role of the reaction components.

All RInChIKeys are generated using the sha2 hashing functionality used by and provided with the InChI algorithm.[11]

Long-RInChIKeys

While RInChI is built by concatenating InChIs of each reaction component, the Long-RInChIKey is composed from InChIKeys of the components. Based on the component order of RInChI, the Long-RInChIKey consists of 5 layers each of them separated by one or two hyphens "-":

- The first layer is rigid and contains the version identifier for the Long-RInChIKey "SA" pointing to the usage of the Standard InChI version 1.04 (="A"). That delivers the fixed length string "Long-RInChIKey=SA" followed by "-" as separator to the next layer.
- The second layer consists of five letters. The first letter describes the direction of the reaction with "F" for forward reaction, "B" for backward reaction, "E" for equilibrium reaction and "U" for unspecified reaction direction. Therefore, "F" corresponds to "d+" of the 5th RInChI layer, "B" to "d-", "E" to "d="and "U" stands for undefined with the 5th RInChI layer being empty (i.e. no value for the 5th RInChI layer given).[12]

  The remaining four letters are not in use in the current RInChI version but are intended to describe reaction conditions like temperature, yield, etc. in future releases. The hash code of this empty string leads to the acronym "UHFF" corresponding to the first four letters of the hash for an empty string.

  Therefore, the format of the second level is defined by a fixed string with "(F,B,E,U)UHFF" where "F,B,E,U" means to take one out of the four letters "F", "B", "E", or "U". The separator to the next layer is a single hyphen "-".

- Layers 3, 4, and 5 of the Long-RInChIKey correspond to layers 2, 3, and 4 of RInChI. Within each of the RInChI layers, the InChI of each component in the layer is replaced by the InChIKey of this compound keeping the order as defined in RInChI.

  Multiple InChIKeys in the same layer are separated by a single hyphen "-".

  Each No-structure or one of the related pseudoatoms (R, X, A and *-arom) is represented by the hash value for an empty string ("MOSFIJXAXDLOML-UHFFFAOYSA-N")

  Layers 3, 4, and 5 are separated by a double hash "--".

  Layers are only displayed if they contain at least one InChIKey. If the last layer of the string is empty the separator "--" is omitted as well.

[11] For more details see …/INCHI-1-API/INCHI_API/inchi_dll/sha2.c and sha2.h and as reference (downloaded 09-Aug-2016) http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf

[12] Note that the unspecified reaction direction is not implemented in the current RInChI release.

Using italic characters for those layers that do not always contribute to RInChI it takes the following format:

Long-RInChIKey=SA-(F,B,E,U)UHFF-*layer3*--*layer4*--*layer5*

with (F,B,E,U) = F, B, E, or U

Notes:

Like the size of RInChIs, the length of the Long-RInChIKey depends on the number of components participating in the reaction. Because InChIKeys are generally shorter than InChIs, it is expected that the Long-RInChIKey is generally shorter than RInChI.

Although redundant, the version information "SA" at the end of the second layer of the InChIKey is kept in the Long-RInChIKey. Because most of the drawing tools allow creating a Standard InChIKey of a chemical structure, the usage of the unmodified InChIKey in the Long-RInChIKey simplifies searches for reaction components by starting with the Standard InChIKey of the component and running a text search over Long-RInChIKeys of interest.

Each occurring no-structure is represented in the Long-RInChIKey by "MOSFIJXAXDLOML-UHFFFAOYSA-N" and empty layers by an empty string "" (e.g. for half-reactions). That leaves the no-structure flag of the sixth RInChI layer obsolete.

Anticipated usage:

Long-RInChIKeys are a valuable tool for the database storage of reactions. Beside uniqueness checks, they allow the identification of each reaction component by simple text searches based on Standard InChIKeys.

Examples

Example: Esterification of acetic acid

RInChI of the esterification example is defined by "RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d="

InChIKeys of the components have the following values

Component	InChIKey
Acetic acid	QTBSBXVTEAMEQO-UHFFFAOYSA-N
Ethanol	LFQSCWFLJHTTHZ-UHFFFAOYSA-N
Ethyl acetate	XEKOWRVHYACXOJ-UHFFFAOYSA-N
Water	XLYOFNOQVPJJNP-UHFFFAOYSA-N
Sulfuric acid	QAOWNCQODCNURD-UHFFFAOYSA-N

- The first layer of the Long-RInChIKey consists of the identifier "Long-RInChIKey=SA".
- The 5th layer of RInChI defines the reaction as equilibrium reaction. Therefore, the 2nd layer of the Lon-RInChIKey begins with "E" followed by the fixed values "UHFF", so that the 2nd layer gets the value "EUHFF".
- The 3rd layer of the Long-RInChKey is derived from InChIKeys related to the compound defined in the 2nd layer of RInChI keeping the order of the compounds as defined in RInChI: Accordingly, InChIKeys of acetic acid and ethanol have to be linked by a hash "-" as separator: "QTBSBXVTEAMEQO-UHFFFAOYSA-N-LFQSCWFLJHTTHZ-UHFFFAOYSA-N".
- The 4th layer of the RInChIKey is built from InChIKeys according to the order of compounds in the 3rd layer of RInChI. The InChIKeys of ethyl acetate and water fill the 4th layer as "XEKOWRVHYACXOJ-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N"
- The 5th layer of the Long-RInChIKey is defined by the InChIKey of sulfuric acid as agent: "QAOWNCQODCNURD-UHFFFAOYSA-N"

- Behind the first 2 layers a single hyphen is used as separator; Layers 3 to 5 are separated by a double hypen "--".

The Long-RInChIKey for the esterification is defined as "Long-RInChIKey=SA-EUHFF-QTBSBXVTEAMEQO-UHFFFAOYSA-N-LFQSCWFLJHTTHZ-UHFFFAOYSA-N--XEKOWRVHYACXOJ-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N--QAOWNCQODCNURD-UHFFFAOYSA-N"

Notes:

- Because none of the compounds is chiral or shows any other special representational issues, all minor layers are identical.
- Because none of the compounds is in a special protonation state, all protonation states at the end of a molecule show "N".

Example: Alkaline ring opening

RInChI of the reaction is given as "RInChI=1.00.1S/C6H12O/c1-4-6(3)5(2)7-6/h5H,4H2,1-3H3/t5-,6-/m0/s1!H2O/h1H2/p-1<>C6H14O2/c1-4-6(3,8)5(2)7/h5,7-8H,4H2,1-3H3/t5-,6+/m1/s1/d+"

The InChI keys of the reactants and the product have following values:

Component	InChI key
(2S,3S)-2-ethyl-2,3-dimethyl-oxirane	ZISUZIXPPXXNPC-WDSKDSINSA-N
Hydroxide	XLYOFNOQVPJJNP-UHFFFAOYSA-M
(2R,3S)-3-methylpentane-2,3-diol	RLWWHEFTJSHFRN-RITPCOANSA-N

- The first layer of the Long-RInChIKey is the identifier "Long-RInChIKey=SA".
- Because the 5th layer of RInChI defines the reaction as a forward reaction, the 2nd layer of the Long-RInChIKey begins with "F" followed by the fixed values "UHFF", leaving the 2nd layer with the value "FUHFF".
- The 3rd layer of the Long-RInChIKey is built from InChIKeys of (2S,3S)-2-ethyl-2,3-dimethyl-oxirane with ("ZISUZIXPPXXNPC-WDSKDSINSA-N") and Hydroxide ("XLYOFNOQVPJJNP-UHFFFAOYSA-M") separated by a dash "-", leaving the 3rd layer with "ZISUZIXPPXXNPC-WDSKDSINSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-M"
- The 4th layer of the Long-RInChIKey consists of the InChIKey for (2R,3S)-3-methylpentane-2,3-diol shown as "RLWWHEFTJSHFRN-RITPCOANSA-N".
- The 5th layer is not defined in RInChI. Therefore, the agent layer is empty.
- Behind the first 2 layers a single hyphen is used as separator and layers 3 and 4 are separated by a double hyphen "--".

That results in the following Long-RInChIKey: "Long-RInChIKey=SA-FUHFF-ZISUZIXPPXXNPC-WDSKDSINSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-M--RLWWHEFTJSHFRN-RITPCOANSA-N"

Notes:

- Caused by the stereochemistry, the minor layers of (2R,3S)-3-methylpentane-2,3-diol and (2S,3S)-2-ethyl-2,3-dimethyl-oxirane are not represented by the hash for an empty string as e.g. in the minor layer of the Hydroxyde.
- The Hydroxide provides another protonation level identified by "M".

Example: Polymerisation of styrene

RInChI for the polymerization of styrene is defined as "RInChI=1.00.1S/<>C8H8/c1-2-8-6-4-3-5-7-8/h2-7H,1H2/d-/u1-0-0"

InChIKeys of the components have following values

Component	InChI key
Styrene	PPBRXRYQALVLMV-UHFFFAOYSA-N
No-structure as representation for the polystyrene	MOSFIJXAXDLOML-UHFFFAOYSA-N

- The first layer of the Long-RInChIKey consists of the identifier "Long-RInChIKey=SA".
- Because the 5th layer of the RInChI defines the reaction as a backward reaction, the 2nd layer of the Long-RInChIKkey begins with "B" followed by the fixed values "UHFF", leaving the 2nd layer with the value "BUHFF".
- The 3rd layer of the Long-RInChIKey is built from the InChIKey of the no-structure described in the 2nd layer of RInChI. Therefore, the 3rd layer is represented by "MOSFIJXAXDLOML-UHFFFAOYSA-N"[13]
- The 4th layer of the RInChIKey is built from InChIKeys for styrene given as "PPBRXRYQALVLMV-UHFFFAOYSA-N".
- The 5th layer is not defined in RInChI. Thus it is skipped.
- Behind the first 2 layers a single hyphen is used as separator; layers 3 and 4 are separated by a double hyphen "--".

Long-RinchiKey for the polymerization is shown as "Long-RInChIKey=SA-BUHFF-MOSFIJXAXDLOML-UHFFFAOYSA-N--PPBRXRYQALVLMV-UHFFFAOYSA-N"

Short-RInChIKeys

The disadvantage of the Long-RInChIKey is the variable length of the key that makes it less suitable for index operations, for exact match searches in databases, or for web searches. Based on the RInChI string, the fixed length Short-RInChIKey is built out of nine layers each separated by hyphens "-" as described in the following steps

- Similar to the Long-RInChIKey, the first layer is rigid and contains the version identifier for the Short-RInChIKey "SA" (Standard InChI version 1.04 ="A"). That delivers the fixed length string "Short-RInChIKey=SA".
- The second layer of the Short-RInChIKey is identical to the second layer of the Long-RInChIKey. The first of the five letters describes the direction of the reaction with "F" for forward reaction, "B" for backward reaction, "E" for equilibrium reaction and "U" for unspecified reaction direction.[14] The hash string "UHFF" follows the "direction letter". The intention of this string is to describe reaction conditions like temperature, yield, etc. in future RInChI releases.

  Therefore, the format of the second layer is defined by a fixed string with "(F,B,E,U)UHFF" where "F,B,E,U" means to take one out of the four letters "F", "B", "E", or "U".

- The creation of layers 3 to 5 and 6 to 8 is based on InChIs of each RInChI group. Each InChI itself consists of the following three groups: major layer – minor layer – protonation state. For each layer of RInChI, the major InChI layers are concatenated into one string in alphabetical order as defined in the related RinChI layer. This string is hashed by the InChI hashing algorithm keeping the first 10 letters.

  The hash of the InChI major layers of the second section of RInChI goes into the Short-RInChIKey layer 3. The hash of InChI major layers of the fourth group in RInChI is found in the Short-RInChIKey layer 4, and eventually the hash over InChI major layers in the fifth section of RInChI is set into the Short-RInChIKey layer 5. Any empty layer results in the hash for an empty string shortened to 10 letters ("UHFFFADPSC").

  Short-RInChIKey layers 6, 7, and 8 are built from the minor InChI layers and the protonation states of the reactants, products and agents as defined in the RInChI layers 3, 4 and 5.

  For each layer of the RInChI, the protonation states are added up leaving … L for -2 protons, M for -1 proton, N for 0 proton, O for +1 proton, P for +2 protons … in the range of maximal ± 12

[13] Note: In case of multiple no-structure representation each of the structures are represented by MOSFIJXAXDLOML-UHFFFAOYSA-N separately. See the chapter "No-structures and pseudoatoms representing components that cannot be structurally drawn" in the appendix for more examples.
[14] Note that the unspecified reaction direction is not implemented in the current RInChI release.

protons[15]. The added protonation states of InChIs are taken to the first position in the related layer of the Short-RInChIKey.

All minor InChI layers of each RInChI section are joined keeping the order of InChIs in the related layer of RInChI and are eventually hashed. From the resulting string, the first four letters are joined with the hashed protonation state building the base for the minor layers of the Short-RInChIKey kept in sections 6 to 8 of the Short-RInChIKey. The hashes from RInChI layer 2 are found in section 6 of the Short-RInChIKey, the hashes from section 3 of RInChI are set into layer 7 of the Short-RInChIKey, and the hashes of group 4 of RInChI go into layer 8 of the Short-RInChIKey.

An empty minor layer in RInChI is constructed out of the first four letters of the hash for an empty string with the protonation state N (no change in protons) in front. That leaves the representation of the empty string with "NUHFF".

- The ninth layer of the Short-RInChIKey consists of three letters, each of them representing the count of no-structures found in the sixth layer of RInChI. To simplify reading, "Z" corresponds to a count of zero no-structures (i.e. no no-structure), while "A" stands for one no-structure, B for two and so forth. The format is given by ### with # = Z, A, B, …

That defines the Short-RInChIKey as

Short-RInChIKey=SA-(F,B,E,U)UHFF-major layer3-major layer4-major layer5-minor layer6- minor layer7-minor layer8-### *with* (F,B,E,U) = F, B, E, or U and # = *Z, A, B*…

Notes:

Short-RInChIKey has a fixed length of 55 letters plus 8 hyphens as separators resulting in a total of 63 characters.

Although hash strings are shortened quite considerably, collision tests by Guenter Grethe, Jonathan M Goodman and Chad HG Allen (Grethe, Goodman, & Allen, 2013) showed that the results are still sufficient to avoid hash clashes.

Anticipated usage:

The fixed length of Short-RInChIKey makes it suitable for exact searches of reactions in databases and in the WEB as well as indexing reactions in databases or linking over the same reaction in different databases.

Examples

Example: Esterification of acetic acid

RInChI for the esterification is defined as "RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d="

- The first layer of the Short-RInChIKey consists of the identifier "Short-RInChIKey=SA".
- The fifth layer of the RInChI defines the reaction as equilibrium reaction. Therefore, the 2nd layer of the Short-RInChIKey begins with "E" followed by the fixed values "UHFF", so that the 2nd layer is written as "EUHFF".
- The 3rd layer consists of the first 10 characters of the hashed string of the major layers for InChIKeys of acetic acid ("C2H4O2/c1-2(3)4/h1H3,(H,3,4)") and ethanol ("C2H6O/c1-2-3/h3H,2H2,1H3Z") taken from the second group of RInChI . It shows as "JJFIATRHOH".

[15] See IUPAC International Chemical Identifier (InChI) - InChI version 1, software version 1.04 (2011) - Technical Manual (Stein, Heller, Tchekhovskoi, & Pletnev, 2011), p.58 for more details

- The 4th layer is built out of the first 10 characters of the hashed string concatenated from the InChI major layer for ethyl acetate "C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3" and water "H2O/h1H2" of the third group of RInChI. It is calculated to give "UDXZTNISGZ".
- The 5th layer comprises the first 10 letters of the hashed string from the major layer of InChI for sulfuric acid ("H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)") in the 4th section of the RInChI. It becomes "QAOWNCQODC"[16].
- The first letter of the 6th layer is built by summing up the protonation states of InChIs of acetic acid (not defined) and ethanol (not defined) from the 2nd group of RInChI. As "N" represents no protonation the resulting protonation state is "N".

  The other four letters of this layer are the first 4 letters of the hash of the concatenated strings from the minor layers of InChIs for acetic acid and ethanol which is empty for these compounds. The first 4 letters of the hash string are kept: "UHFF".

  That leaves the 6th layer of the Short-RInChIKey with "NUHFF".
- The first letter of the 7th layer is the sum over the protonation states of InChIKeys of ethyl acetate (not defined) and water (not defined) from the third group of RInChI resulting in the state "N".

  The remaining four letters are the first 4 letters of the hash of the concatenated strings from the minor layers of InChI for ethyl acetate as found in the fourth group of RInChI. Because these strings are empty, it is defined by "UHFF".

  Therefore, the 7th layer of the Short-RInChIKey lists as "NUHFF".
- The first letter of the 8th layer is taken from the protonation state of the InChIKey of sulfuric acid (not defined) from the 4th group of RInChI contributing the state "N".

  The continuing four letters are the first 4 letters of the hash of the minor layer of InChI for sulfuric acid as found in the 4th group of RInChI. Because there is no information given, it is defined as "UHFF".

  Therefore, the 8th layer of the Short-RInChIKey is written as "NUHFF".
- As the sixth layer of RInChI describes, there are no no-structures involved in the reaction representation. Because the letter "Z" stands for zero no-structures, the 9th layer becomes "ZZZ".
- All layers of the Short-RInChIKey are separated by an hyphen "-" from each other.

The Short-RInChIKey of the esterification reaction is "Short-RInChIKey=SA-EUHFF-JJFIATRHOH-UDXZTNISGZ-QAOWNCQODC-NUHFF-NUHFF-NUHFF-ZZZ"

Example: Alkaline ring opening

RinChI of this reaction has the following format: "RInChI=1.00.1S/C6H12O/c1-4-6(3)5(2)7-6/h5H,4H2,1-3H3/t5-,6-/m0/s1!H2O/h1H2/p-1<>C6H14O2/c1-4-6(3,8)5(2)7/h5,7-8H,4H2,1-3H3/t5-,6+/m1/s1/d+"

- The first layer of the Short-RInChIkey is generally written as "Short-RInChIKey=SA".
- Because the 5th layer of RInChI states that the reaction is a forward reaction, the 2nd layer of the Short-RInChIKey begins with "F" followed by the fixed values "UHFF" letting the 2nd layer of the Short-RInChIKey become "FUHFF".
- The second layer of RInChI is defined by the major layers of InChIs for (2S,3S)-2-ethyl-2,3-dimethyl-oxirane and hydroxide ("H2O/h1H2"). Following the order in the second layer of RInChI, both strings are concatenated, hashed by the sha2 implementation of the InChI algorithm and the first 10 letters of the resulting string are kept for the 3rd layer of the Short-RInChIKey as "KXNHVTRUIV".
- The 4th layer is built out of the major layer of the 3rd group of RInChI calculated for (2R,3S)-3-methylpentane-2,3-diol as "C6H14O2/c1-4-6(3,8)5(2)7/h5,7-8H,4H2,1-3H3". The string is hashed and the first 10 characters are kept as "RLWWHEFTJS".

[16] If the layer only consists of one component, the content of the 3rd,4th and 5th layer is identical to the first ten letters of the first layer of the InChIKey representing this single compound. In the example of sulfuric acid, the first layer of the InChIKey is calculated as "QAOWNCQODCNURD". The first ten characters "QAOWNCQODC"are taken over into the 5th layer of the esterification Short-RInChIKey

- There are no agents defined for this reaction. The first 10 letters for the hash representing the empty string are "UHFFFADPSC" filling the 5th layer of the Short-RInChIKey.
- The first letter of the 6th layer results from the protonation states of (2S,3S)-2-ethyl-2,3-dimethyl-oxirane (not defined) and Hydroxide ("p-1") from the second layer of RInChI. Adding the protonation states 0 and -1 you get M (= -1). The remaining 4 letters are generated by concatenating the minor layers of InChIs of (2R,3S)-3-methylpentane-2,3-diol ("t5-,6-/m0/s1") and hydroxide (not defined) from the second layer in RInChI and keeping the first 4 letters of the resulting hash string with "CHCV". The 6th layer becomes "MCHCV".
- The protonation state of the 7th layer is given in the third layer of RInChI by the protonation state of InChI of (2R,3S)-3-methylpentane-2,3-diol which is not explicitly defined and therefore 0 corresponding to "N". Together with the first 4 characters of the hash of the minor layer ("/t5-,6+/m1/s1") this layer is described as "NMHYF".
- Because there are no agents defined for this reaction, the 8th layer is empty and therefore represented by "NHUFF".
- Because none of the layers contains a "no-structure", the 9th layer of the Short-RInChIKey becomes "ZZZ".
- All layers of the Short-RInChIKey are separated by hyphens "-" from each other.

The Short-RInChIKey of the ring opening reaction is calculated as "Short-RInChIKey=SA-FUHFF-KXNHVTRUIV-RLWWHEFTJS-UHFFFADPSC-MCHCV-NMHYF-NUHFF-ZZZ".

Example: Polymerization of styrene

RInChI for the polymerization is given as "RInChI=1.00.1S/<>C8H8/c1-2-8-6-4-3-5-7-8/h2-7H,1H2/d-/u1-0-0"

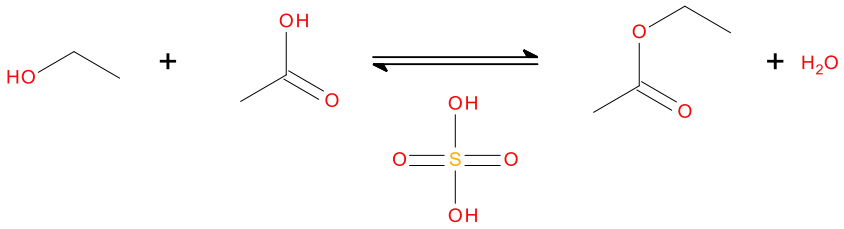
- The first layer of the Short-RInChIKey is defined by "Short-RInChIKey=SA".
- Because the 5th layer of the RInChI defines the reaction as backward reaction, the 2nd layer of the Short-RInChIKey begins with "B" followed by the fixed values "UHFF" presenting the 2nd layer of the Short-RInChIKey as "BUHFF".
- The second layer of RInChI is empty. The first 10 characters for the related hash are "UHFFFADPSC" which is placed into the 3rd layer of the Short-RInChIKey.
- The 4th layer is built from the first 10 characters of the hashed string of the major layer of InChI for styrene with "C8H8/c1-2-8-6-4-3-5-7-8/h2-7H,1H2" from the third RInChI layer and is shown as "PPBRXRYQAL".
- There is no information given for the fourth RInChI layer. The first 10 letters for the hash representing the empty string are "UHFFFADPSC" filling the 5th layer of the Short-RInChIKey.
- The 6th to 8th layer of the Short-RInChIKey follow the same set-up: There is no definition given in the second, third and fourth layer of RInChI for the protonation states leaving the sum over all protonation states with 0 corresponding to "N". Together with the first 4 letters for the empty minor layers of InChIs in the RInChI layers 2, 3, and 4, the Short-RInChIKey layers 6,7, and 8 get the value "NUHFF".
- The sixth layer of RInChI with "u1-0-0" indicates that the second layer of RInChI contains a no-structure. Because "A" represents one no-structure and "Z" zero no-structures, the 9th layer of the Short-RInChIKey becomes "AZZ"
- All layers of the Short-RInChIKey are separated by hyphens "-" from each other.

The Short-RInChIKey of the polymerization reaction becomes "Short-RInChIKey= SA-BUHFF-UHFFFADPSC-PPBRXRYQAL-UHFFFADPSC-NUHFF-NUHFF-NUHFF-AZZ"

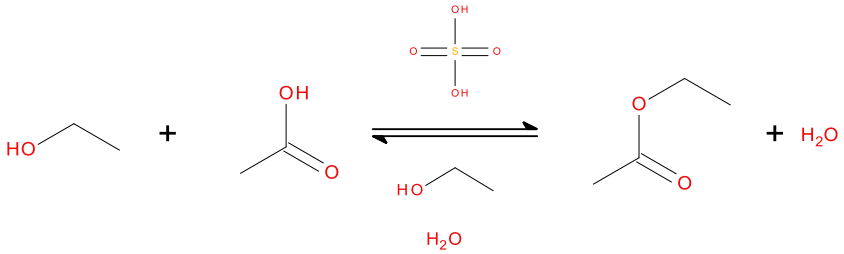
Web-RInChIKeys

RInChI, Long-RInChIKey and Short-RInChIKey are exact representations of a reaction as defined by its author(s). Unfortunately, there is no generally shared definition of how the roles of compounds

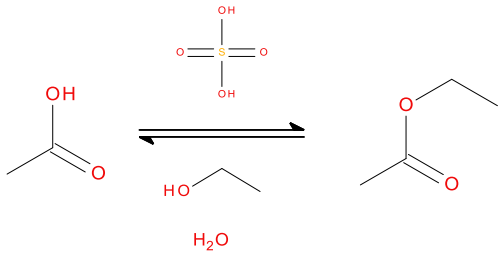
participating in reactions must be defined. The following three examples describe the same esterification reaction:



Reaction 8: Esterification



Reaction 9: Esterification with explicit representation of solvents



Reaction 10: Esterification focusing to amino acid

While the first description (Reaction 8) follows the rule that those compounds that undergo a change in their bond and/or atom count define the reactants and products with the sulfuric acid (over the arrow) as catalyst, the second one (Reaction 9) considers that ethanol and water have a second role as solvent. The third reaction (Reaction 10) puts the major focus of the reaction on acetic acid, while H2SO4 stays in its role as catalyst (above the arrow) and ethanol and water are just solvents (below the arrow in our example).

If you have to search reactions, for example in web searches, over multiple reaction databases or in databases with unknown reaction component definitions, a more "open" key is needed to bypass the hurdles of role identifications. The "Web-RInChIKey" has been defined to address this issue and to overcome a problem with some web search engines that replace hyphens "-" in search strings by blanks. This causes problems especially for searches with Short-RInChIKeys which consist of nine layers that become nine separate search strings in these web search engines resulting in a major loss of accuracy in the search results. On the other side, longer strings make the results in web searches more precise. Therefore, the optimal RInChIKey for web searches is defined by a fixed character length with only a very few separated strings, with each of the strings long enough to provide specific results in the web search processes. The format chosen for the "Web-RInChIKey is 47 characters long including a string of 17 characters to hash the mayor layers of all reaction components and a second string of 15 characters composed of the sum of all protonation states and the hash over all minor layers of all components extended by "SA" as InChI related versioning identification.

The creation of the "Web-RInChIKey" is based on the simplification that InChIs of all components of the reactions are considered without keeping the actual role of each constituent within the reaction and that the reaction direction is ignored. In order to create the key the following steps are taken:

1. All InChIs of the reaction components are ordered alphabetically. The role of the components are ignored.
2. Any duplicates are deleted from the list.
3. All major InChI layers are concatenated in the order defined in the first step and hashed using the InChI hash functionality. The first 17 characters of the hash string define the "major" layer of the Web-RInChIKey.
4. The protonation states of all participating InChIs are summed up with 0 as protonation state for those that are not explicitly defined in the InChI string. L = -2, M = -1, N = 0, O = +1, P = +2 … as 1-letter representation for the protonation state[17] become the first character of the "minor" layer of the Web-RInChIKey.

   The alphabetical order is used to combine all minor layers of the InChIs. The resulting string is hashed by the sha2 method provided by the InChI library. The first 12 characters of the minor layers hash are joined with the one letter for the protonation state.

   The minor layer is completed by the version identification for the InChI algorithm by adding the letter combination "SA" to the end of the minor layer of the Web-RInChIKey. The total minor Web-RInChIKey is a fixed string with15 characters.
5. Both layers are separated by a hyphen "-".
6. The 1st layer of the Web-RInChIKey is the fixed string "Web-RInChIKey=".

With these steps, the Web-RInChIKey has the following format:

Web-RInChIKey=major layer-minor layer

Notes:

The Web-RInChIKey has a fixed length of 33 characters including the separator for the structural representation plus 14 letters for the first layer leaving a total of 47 characters. Therefore, the Web-RInChIKey is a fixed length key.

With lengths of 17 letters for the major layer and 15 characters for the minor layer, web searches deliver precise result sets.

The Web-RInChIKey is a component-based representation of reactions that is agnostic about the role of any component within a reaction. Reactions with identical components but different roles are represented by identical Web-RInChIKeys.

While the InChI for a missing reaction component (in half-reactions etc.) is an empty string, the no-structure is represented by "MOSFIJXAXDLOML-UHFFFAOYSA-N". Consequently, no-structures and missing reaction components are distinguished so that Web-RInChIKey does not need a no-structure flag. (See appendix "No-structures and pseudoatoms representing components that cannot be structurally drawn" and "Half and empty reactions" for more details.)

Because some web search engines ignore the hyphen "-" in their search algorithms, they run a search with two separated strings. Caused by the length and unique character combination, especially of the major layer, it is expected to find the requested reactions among the top hits of the web search.

For the results of duplicate checks in the American Patent database for the years 2008 to 2011 provided by NextMove see "Results of Web-RInChIkey comparisons" in the appendix of this documentation.

[17] See IUPAC International Chemical Identifier (InChI) - InChI version 1, software version 1.04 (2011) - Technical Manual (Stein, Heller, Tchekhovskoi, & Pletnev, 2011), p.58 for more details

Anticipated usage:

The Web-RInChIKey has been optimized for web searches and for the comparison of reaction databases with unknown component roles.

Example: Esterification of acetic acid

Considering the three variations of the esterification reaction, the RInChI algorithm calculates the following RInChIs:

Reaction 8:

RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d=

Long-RInChIKey=SA-EUHFF-QTBSBXVTEAMEQO-UHFFFAOYSA-N-LFQSCWFLJHTTHZ-UHFFFAOYSA-N--XEKOWRVHYACXOJ-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N--QAOWNCQODCNURD-UHFFFAOYSA-N

Short-RInChIKey=SA-EUHFF-JJFIATRHOH-UDXZTNISGZ-QAOWNCQODC-NUHFF-NUHFF-NUHFF-ZZZ

Reaction 9:

RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>C2H6O/c1-2-3/h3H,2H2,1H3!H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d=

Long-RInChIKey=SA-EUHFF-QTBSBXVTEAMEQO-UHFFFAOYSA-N-LFQSCWFLJHTTHZ-UHFFFAOYSA-N--XEKOWRVHYACXOJ-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N--LFQSCWFLJHTTHZ-UHFFFAOYSA-N-QAOWNCQODCNURD-UHFFFAOYSA-N

Short-RInChIKey=SA-EUHFF-JJFIATRHOH-UDXZTNISGZ-UAUFKIWNBD-NUHFF-NUHFF-NUHFF-ZZZ

Reaction 10:

RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3<>C2H6O/c1-2-3/h3H,2H2,1H3!H2O/h1H2!H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d=

Long-RInChIKey=SA-EUHFF-QTBSBXVTEAMEQO-UHFFFAOYSA-N--XEKOWRVHYACXOJ-UHFFFAOYSA-N--LFQSCWFLJHTTHZ-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N-QAOWNCQODCNURD-UHFFFAOYSA-N

Short-RInChIKey=SA-EUHFF-QTBSBXVTEA-XEKOWRVHYA-DNBJJWMYJT-NUHFF-NUHFF-NUHFF-ZZZ

Web-RInChIKey

In order to calculate the Web-RInChIkey the InChIs of each component are listed:

Component	InChI / Major
Ethanol	C2H6O/c1-2-3/h3H,2H2,1H3
Acetic acid	C2H4O2/c1-2(3)4/h1H3,(H,3,4)
Ethyl acetate	C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3
Water	H2O/h1H2
Sulfuric acid	H2O4S/c1-5(2,3)4/h(H2,1,2,3,4

The InChIs are ordered alphabetically in ascending order and any duplicates are deleted:

Component	InChI / Major
Acetic acid	C2H4O2/c1-2(3)4/h1H3,(H,3,4)
Ethanol	C2H6O/c1-2-3/h3H,2H2,1H3
Ethyl acetate	C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3
Water	H2O/h1H2
Sulfuric acid	H2O4S/c1-5(2,3)4/h(H2,1,2,3,4

All major layers of the alphabetically ordered InChIs are joined to a common string that is hashed based on the InChI algorithm. The first 17 characters of the hash string are kept as major layer of the Web-RInChIKey: "SMUHAWIQPXIVCEVKG"

All protonation states are added together. Because the protonation states are not listed at any of the InChIs, the total protonation state is 0 which is represented by "N".

All minor layers of the InChIKeys are joined in the previously defined order and hashed keeping the first 12 characters of the hash string. Because none of the minor layers is filled in this example, the hash becomes the representation of the empty sting: "UHFFFADPSCTJ"

Finally the version of the underlying InChI is defined with "SA" as described before.

The minor layer of the Web-RInChIKey becomes "NUHFFFADPSCTJSA".

All three reactions (Reaction 8 to 10) use the same components although the roles differ. Therefore, all three of them are identified by the same Web-RInChIKey:

"Web-RInChIKey=SMUHAWIQPXIVCEVKG-NUHFFFADPSCTJSA"

Example: Alkaline ring opening

RInChI is given as "RInChI=1.00.1S/C6H12O/c1-4-6(3)5(2)7-6/h5H,4H2,1-3H3/t5-,6-/m0/s1!H2O/h1H2/p-1<>C6H14O2/c1-4-6(3,8)5(2)7/h5,7-8H,4H2,1-3H3/t5-,6+/m1/s1/d+", RInChIKeys as "Long-RInChIKey=SA-FUHFF-ZISUZIXPPXXNPC-WDSKDSINSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-M--RLWWHEFTJSHFRN-RITPCOANSA-N" and "Short-RInChIKey=SA-FUHFF-KXNHVTRUIV-RLWWHEFTJS-UHFFFADPSC-MCHCV-NMHYF-NUHFF-ZZZ"

Component	InChI		
	Major	Protonation state	Minor
2S,3S)-2-ethyl-2,3-dimethyl-oxirane	C6H12O/c1-4-6(3)5(2)7-6/h5H,4H2,1-3H3		t5-,6-/m0/s1
Hydroxide	H2O/h1H2	p-1	
(2R,3S)-3-methylpentane-2,3-diol	C6H14O2/c1-4-6(3,8)5(2)7/h5,7-8H,4H2,1-3H3		t5-,6+/m1/s1

The InChI of (2S,3S)-2-ethyl-2,3-dimethyl-oxirane, hydroxide, and (2R,3S)-3-methylpentane-2,3-diol are ordered alphabetically.

Component	InChI		
	Major	Protonation state	Minor
2S,3S)-2-ethyl-2,3-dimethyl-oxirane	C6H12O/c1-4-6(3)5(2)7-6/h5H,4H2,1-3H3		t5-,6-/m0/s1
(2R,3S)-3-methylpentane-2,3-diol	C6H14O2/c1-4-6(3,8)5(2)7/h5,7-8H,4H2,1-3H3		t5-,6+/m1/s1
Hydroxide	H2O/h1H2	p-1	

The strings of the three major layers are concatenated and hashed keeping the first 17 characters with "ZHLKMEWITROQDDAWW".

The added protonation states with the empty states handled as 0 become p-1 which is represented by "M". The first 12 characters of the hashed concatenated string of the three minor layers become "GWJVGYOGVOSO" followed by "SA" to point to the InChI version. That leaves the minor layer with "M GWJVGYOGVOSOSA".

The resulting key is defined as

"Web-RInChIKey=ZHLKMEWITROQDDAWW-MGWJVGYOGVOSOSA"

Example: Polymerisation of styrene

RInChIs are calculated as "RInChI=1.00.1S/<>C8H8/c1-2-8-6-4-3-5-7-8/h2-7H,1H2/d-/u1-0-0", "Long-RInChIKey=SA-BUHFF-MOSFIJXAXDLOML-UHFFFAOYSA-N--PPBRXRYQALVLMV-UHFFFAOYSA-N", and "Short-RInChIKey=SA-BUHFF-UHFFFADPSC-PPBRXRYQAL-UHFFFADPSC-NUHFF-NUHFF-NUHFF-AZZ"

For the calculation of the Web-RInChIKey, list the InChIs for each reaction component

Component	InChI / Major
Styrene	C8H8/c1-2-8-6-4-3-5-7-8/h2-7H,1H2
No-structure	

In this example the minor layer and the protonation layer are not defined.

Order InChIs in ascending alphabetical order and delete any duplicates:

Component	InChI / Major
No-structure	
Styrene	C8H8/c1-2-8-6-4-3-5-7-8/h2-7H,1H2

Concatenate all the major layers and hash them using the InChI hash functionality. Keep the first 17 characters of the hash string leaving "MMBMJDIYKORFMRQKP".

Neither the protonation states nor the minor layers of the reaction components are defined. The sum of the protonation states is 0 = "N" and the first 12 characters of the hash over the empty minor layers is calculated as "UHFFFADPSCTJ". Adding the "SA" for the version, the minor layer of the Web-RInChIKey is given as "NUHFFFADPSCTJSA".

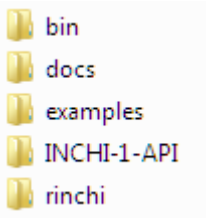
Therefore, the styrene polymerization is described by

"Web-RInChIKey=MMBMJDIYKORFMRQKP-NUHFFFADPSCTJSA"

Scope of delivery

Directory structure

RInChI provides the following directory structure:



with

Directory	Subdirectory	Content
bin	rinchi_cmdline	Exe files for the RInChI command line tools for Linux (32 bit, 64 bit), PowerMac, and Windows (32 bit, 64 bit)
	rinchi_lib	Compile libraries for Linux (32 bit, 64 bit), and Windows (32 bit, 64 bit)
docs		Documentation
examples		All RXN/RD files with the related RInChIs that are described in this document
INCHI-1-API	INCHI	InChI release 1.04
	INCHI-API	
rinchi	example_apps	Code examples for the RInChI command line tools and for rxn_from_molfiles with the directory "data" demonstrating the creation of RInChI out of molfiles and their component roles instead of RXN or RD files
	lib	Code for the InChI and RInChI libraries
	parsers	Code of parser components
	rinchi	Code for reaction handling
	rinchi_lib	Make files
	writers	Code for the RXN/RD file writer

For code compilation, make sure that the InChI libraries and the RInChI directory are located under the same directory like in the above shown directory tree.

Function calls

C-style call interface: rinchi_lib.cpp / rinchi_lib.h

For maximum control you will want to use the C++ call interface described in the next section. However, for easy interfacing to 3rd party tools and languages a high-level C-style interface is provided. The C-style interface is convenient when calling RInChI functionality from e.g. Python or database functions.

The C-style interface is found in the files /rinchi_lib/rinchi_lib.h and /rinchi_lib/rinchi_lib.cpp. The header file defines six callable functions, described below. All functions, except the rinchilib_latest_err_msg(), return an integer error code. The error code can be 0 (success) or 1 (error occurred).

Results are returned as pointers to C-strings. The result pointers reference C-strings managed by RInChI. You should *not* attempt to free the memory of these strings – this is handled by RInChI.

Also note that result pointers are valid for the current call only. The strings will be overwritten on the next rinchi_lib call, so copy or evaluate the result before doing the next rinchi_lib call. Note that this also has implications if you want to use the C-style interface in a multi-threaded environment. In that case, you need to serialize access to rinchi_lib functions, both to get consistent result pointers and also to get consistent error messages.

rinchilib_latest_err_msg

Returns a pointer to a C-string with the latest error message produced. The memory of this C-string is managed by RInChI – do not attempt to free this yourself.

(const char rinchilib_latest_err_msg())*

rinchilib_rinchi_from_file_text

Generate RInChI and RAuxInfo from a RD or RXN file text block.

Parameter:

input_format	‘‘AUTO’’, "RD" or "RXN" (with ‘‘AUTO’’ as default value)
in_file_text	Pointer to C-string containing a text block with RD or RXN file contents
in_force_equilibrium	Force interpretation of reaction as equilibrium reaction (boolean)
out_rinchi_string	Returned pointer to C-string with generated RInChI
out_rinchi_auxinfo	Returned pointer to C-string with generated RAuxInfo

Return value:

Standard error code.

Function:

int rinchilib_rinchi_from_file_text (const char input_format, const char* in_file_text, bool in_force_equilibrium, const char** out_rinchi_string, const char** out_rinchi_auxinfo)*

rinchilib_rinchikey_from_file_text

Generate a RInChI key from an RD or RXN file text block.

Parameter:

input_format	"RD" or "RXN"
in_file_text	Pointer to C-string containing a text block with RD or RXN file contents
in_force_equilibrium	Force interpretation of reaction as equilibrium reaction (boolean)
key_type	Pointer to C-string with 1 letter controlling the type of key generated with • ‘‘L’’ for the Long-RInChIKey • ‘‘S’’ for the Short key (Short-RInChIKey) • ‘‘W’’ for the Web key (Web-RInChIKey)
out_rinchi_key	Returned pointer to C-string with generated key

Return value:

Standard error code

Function:

int rinchilib_rinchikey_from_file_text (const char input_format, const char* in_file_text, const char* key_type, bool in_force_equilibrium, const char** out_rinchi_key)*

rinchilib_file_text_from_rinchi

Reconstruct a RD or RXN file from RInChI and (optionally) RAuxInfo.

Parameter:

rinchi_string	Pointer to C-string with RInChI string
rinchi_auxinfo	Pointer to C-string with RAuxInfo. May point to a blank string but may not be a NULL pointer.
output_format	"RD" or "RXN".
out_file_text	Returned pointer to C-string with produced file text block.

Return value:

Standard error code

Function

int rinchilib_file_text_from_rinchi(const char rinchi_string, const char* rinchi_auxinfo, const char* output_format, const char** out_file_text*);

rinchilib_inchis_from_rinchi

Splits RInChI and (optionally) RAuxInfo into InChI+AuxInfo pairs of individual component moeities.

Parameter:

rinchi_string	Pointer to C-string with RInChI string
rinchi_auxinfo	Pointer to C-string with RAuxInfo. May point to a blank string but may not be a NULL pointer.
out_inchis_text	Returned pointer to C-string with InChIs and RAuxInfo-s

Return value:

Standard error code

Function

int rinchilib_inchis_from_rinchi (const char rinchi_string, const char* rinchi_auxinfo, const char** out_inchis_text);*

rinchilib_rinchikey_from_rinchi

Generates a RInChI key from a RInChI string

Parameter:

rinchi_string	Pointer to C-string with RInChI string
key_type	Pointer to C-string with 1 letter controlling the type of key generated with • "L" for the Long-RInChIKey • "S" for the Short key (Short-RInChIKey) • "W" for the Web key (Web-RInChIKey)
out_rinchi_key	Returned pointer to C-string with generated key

Return value:

Standard error code

Function

int rinchilib_rinchikey_from_rinchi(const char rinchi_string, const char* key_type, const char** out_rinchi_key);*

C++ call interface: rinchi_reaction.cpp / rinchi_reaction.h

The C++ call interface is found in the files /rinchi/rinchi_reaction.h and /rinchi/rinchi_reaction.cpp.

The C++ interface models a reaction as a set of molecular structures, each of which have a role of either Reactant, Product, or Agent. The C++ Reaction class represents the reaction and each structure in the reaction is represented by a ReactionComponent instance.

The Reaction class has the following methods:

add_reactant

Creates a new reaction component as reactant and adds it to the Reaction instance. Returns a pointer to the newly created ReactionComponent.

Function

ReactionComponent add_reactant()*

add_product

Creates a new reaction component as product and adds it to the Reaction instance. Returns a pointer to the newly created ReactionComponent.

Function

ReactionComponent add_product()*

add_agent

Creates a new reaction component as agent and adds it to the Reaction instance. Returns a pointer to the newly created ReactionComponent..

Function

ReactionComponent add_agent()*

rinchi_string

Returns the RInChI string

Function

const std::string Reaction::rinchi_string()

rinchi_auxinfo

Returns the RInChI AuxInfo as string

Function

const std::string Reaction::rinchi_auxinfo()

rinchi_long_key

Returns Long-RInChIKey as string

Function

const std::string Reaction::rinchi_long_key()

rinchi_short_key

Returns Short-RInChIKey as string

Function

const std::string Reaction::rinchi_short_key()

rinchi_web_key

Returns Web-RInChIKey as string

Function

const std::string Reaction::rinchi_web_key()

mdl_molfile_reader.cpp / mdl_molfile_reader.h

The ReactionComponent instances represent a structure through the standard InChI input structure – a C-style structure that contains an array of atoms and their neighbors. In order to calculate RInChI, all ReactionComponents must have this input structure populated correctly. The /parsers/mdl_molfile_reader.h and /parsers/mdl_molfile_reader.cpp files provide code that can read MDL molfiles and populate the InChI input structure accordingly.

The MdlMolfileReader class has two variants of the read_molecule() function: One to read an MDL molfile from a standard stream and one to read an MDL molfile from a named file.

read_molecule

Function

MdlMolfileReader::read_molecule(const std::string& file_name, ReactionComponent& result, int lines_already_read)

Both functions have the optional argument "lines_already_read". This is useful for providing better error messages if you use the MdlMolfileReader as part of processing a larger file; e.g. if you pass in the value 100 and there is an error in line 5 of the molfile, the error message will reflect that an error occurred in line 105 overall.

RInChI programming example

RInChI cmdline tool

This tool is a command line call to calculate RInChIs and RInChIKeys from RXN or RD files or RD files or RXN files from a file containing RInChIs.

The RInChI command line tools provide the following parameters

File name	• RXN or RD file • File containing an RInChI string optionally followed by a linefeed and RAuxInfo data.
Options	
/force-equilibrium	Forces the reaction to be interpreted as an equilibrium reaction. Only relevant for RD or RXN file input.
/rd or /rxn	Force output file format of reconstructed reaction file to be RD or RXN. If the output file format is not forced, this utility will choose to write RD files if the reaction has agents, and RXN files if not. Only relevant for RInChI string input.

Examples

To calculate RInChI and RInChkey for the esterification of the example above enter

>rinchi_cmdline Esterification_01.rdf /force-equilibrium

```
RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>H2O4S/c1-
5(2,3)4/h(H2,1,2,3,4)/d=

RAuxInfo=1.00.1/1/N:1,2,3,4/E:(3,4)/rA:4nCCOO/rB:s1;s2;d2;/rC:-3.8549,-.5552,0;-2.8321,.0354,0;-2.8321,1.2168,0;-1.8089,-
.5554,0;!0/N:3,2,1/rA:3nOCC/rB:s1;s2;/rC:2.499,-.1614,0;1.4762,.4292,0;.453,-
.1615,0;<>0/N:5,1,4,2,6,3/rA:6nCCOCCO/rB:s1;s2;s3;s4;d2;/rC:7.2384,-1.0475,0;8.2613,-
.4569,0;8.2613,.7246,0;9.2844,1.3153,0;10.3076,.7246,0;9.2844,-
1.0476,0;!0/N:1/rA:1nO/rB:/rC:12.5696,.0354,0;<>1/N:2,3,4,5,1/E:(1,2,3,4)/CRV:5.6/rA:5nSOOOO/rB:s1;s1;d1;d1;/rC:6.4257,-
2.7792,0;7.1757,-2.7792,0;5.6757,-2.7792,0;6.4257,-3.5292,0;6.4257,-2.0292,0;
```

```
Long-RInChIKey=SA-EUHFF-QTBSBXVTEAMEQO-UHFFFAOYSA-N-LFQSCWFLJHTTHZ-UHFFFAOYSA-N--XEKOWRVHYACXOJ-UHFFFAOYSA-N-XLYOFNOQVPJJNP-
UHFFFAOYSA-N--QAOWNCQODCNURD-UHFFFAOYSA-N
Short-RInChIKey=SA-EUHFF-JJFIATRHOH-UDXZTNISGZ-QAOWNCQODC-NKDYL-NKDYL-NMOSF-ZZZ
Web-RInChIKey=SMUHAWIQPXIVCEVKG-NJYTGTKWPQNQUSA
```

If you save the strings of RInChI and RAuxInfo (if applicable) in a file (e.g. Esterification_01_RInChI.txt) run the following command to return the RD file

>rinchi_cmdline Esterification_01_RInChI.txt

```
$RDFILE 1
$DATM 04/16/2016 18:58:40
$RFMT
$RXN

      RInChI1.00
NOTE: Reaction is an equilibrium reaction.
  2  2
$MOL
Reactant1
  InChIV10

  4  3  0  0  0  0  0  0  0  0  1 V2000
   -3.8549   -0.5552    0.0000 C   0  0  0     0  0  0  0  0  0
   -2.8321    1.0054    0.0000 C   0  0  0     0  0  0  0  0  0
   -2.8321    1.2168    0.0000 O   0  0  0     0  0  0  0  0  0
   -1.8089   -0.5554    0.0000 O   0  0  0     0  0  0  0  0  0
  1  2  1  0  0  0  0
  2  3  1  0  0  0  0
  2  4  2  0  0  0  0
M  END
$MOL
Reactant2
  InChIV10

  3  2  0  0  0  0  0  0  0  0  1 V2000
    2.4990   -0.1614    0.0000 O   0  0  0     0  0  0  0  0  0
    1.4762    0.4292    0.0000 C   0  0  0     0  0  0  0  0  0
    0.4530   -0.1615    0.0000 C   0  0  0     0  0  0  0  0  0
  1  2  1  0  0  0  0
  2  3  1  0  0  0  0
M  END
$MOL
Product1
  InChIV10

  6  5  0  0  0  0  0  0  0  0  1 V2000
    7.2384   -1.0475    0.0000 C   0  0  0     0  0  0  0  0  0
    8.2613   -0.4569    0.0000 C   0  0  0     0  0  0  0  0  0
    8.2613    0.7246    0.0000 O   0  0  0     0  0  0  0  0  0
    9.2844    1.3153    0.0000 C   0  0  0     0  0  0  0  0  0
   10.3076    0.7246    0.0000 C   0  0  0     0  0  0  0  0  0
    9.2844   -1.0476    0.0000 O   0  0  0     0  0  0  0  0  0
  1  2  1  0  0  0  0
  2  3  1  0  0  0  0
  2  6  2  0  0  0  0
  3  4  1  0  0  0  0
  4  5  1  0  0  0  0
M  END
$MOL
Product2
  InChIV10

  1  0  0  0  0  0  0  0  0  0  1 V2000
   12.5696    1.0054    0.0000 O   0  0  0     0  0  0  0  0  0
M  END
$DTYPE RXN:AGENTS(1):MOLECULES(1):MOLSTRUCTURE
$DATUM $MFMT
Agent1
  InChIV10

  5  4  0  0  0  0  0  0  0  0  1 V2000
    6.4257   -2.7792    0.0000 S   0  0  0     0  0  0  0  0  0
    7.1757   -2.7792    0.0000 O   0  0  0     0  0  0  0  0  0
    5.6757   -2.7792    0.0000 O   0  0  0     0  0  0  0  0  0
    6.4257   -3.5292    0.0000 O   0  0  0     0  0  0  0  0  0
    6.4257   -2.0292    0.0000 O   0  0  0     0  0  0  0  0  0
  1  2  1  0  0  0  0
  1  3  1  0  0  0  0
  1  4  2  0  0  0  0
  1  5  2  0  0  0  0
M  END
```

Notes:

- With "rinchi_cmdline Esterification_01_RInChI.txt /RXN" you return the RXN section of the RInChI only.
- In case the RInChI was created from a RXN file, rinchi_cmdline *RInChI_Filename.txt /RD* returns the RXN section with a RD file header, e.g. to make it compatible to a RD file output together with other RD files.

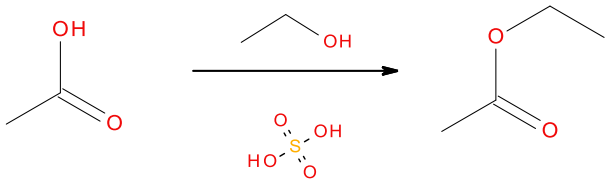
- If the ASCII file containing the RInChI does not contain the RAuxInfo section all coordinates of the returned RXN or RD file are set to 0.0000 and need additional re-calculation. (To see an example call rinchi_cmdline Esterification_01_RInChI_No_RAux.txt

The Rinchi cmdline tool is found in the RInChI subdirectory example_apps/rinchi_cmdline. The code is available for the compilation on 32 and 64 Bit MS Windows, Linux, and Mac OS X machines.

example_apps/rxn_from_molfiles

This example shows how you can create RInChIs and RInChIKeys from molfiles with their reaction role (Reactant, product agent) instead of using RXN or RD files. The example calls are found in rxn_from_molfiles.cpp

```
// Load reactant(s).
rc = rxn.add_reactant();
mol_reader.read_molecule("../data/reactant_01.mol", *rc);
// Load product(s).
rc = rxn.add_product();
mol_reader.read_molecule("../data/product_01.mol", *rc);
// Load agent(s).
rc = rxn.add_agent();
mol_reader.read_molecule("../data/agent_01.mol", *rc);
rc = rxn.add_agent();
mol_reader.read_molecule("../data/agent_02.mol", *rc);

std::cout << "RInChI string  = " << rxn.rinchi_string() << std::endl;
std::cout << "RInChI long key  = " << rxn.rinchi_long_key() << std::endl;
std::cout << "RInChI short key = " << rxn.rinchi_short_key() << std::endl;
std::cout << "RInChI web key   = " << rxn.rinchi_web_key() << std::endl;

Please note that the paths are hard coded in this example
```



To run the example, it is assumed that the compiled executable is found in the subdirectory example_apps/rxn_from_molfiles. Using the esterification example with acetic acid as reactant_01, ethyl acetate as product_01, ethanol as agent_01 (solvent), and sulfuric acid as agent_02 (catalyst) you will get for
> rxn_from_molfiles:

```
RInChI string    =  RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3<>C2H6O/c1-2-3/h3H,2H2,1H3!H2O4S/c1-
                    5(2,3)4/h(H2,1,2,3,4)/d+
RInChI long key  =  Long-RInChIKey=SA-FUHFF-QTBSBXVTEAMEQO-UHFFFAOYSA-N--XEKOWRVHYACXOJ-UHFFFAOYSA-N--LFQSCWFLJHTTHZ-UHFFFAOYSA-N-
                    QAOWNCQODCNURD-UHFFFAOYSA-N
RInChI short key =  Short-RInChIKey=SA-FUHFF-QTBSBXVTEA-XEKOWRVHYA-UAUFKIWNBD-NMOSF-NMOSF-NKDYL-ZZZ
RInChI web key   =  Web-RInChIKey=CZRWKHMMVCVIJYGKV-NIQWWHAMGEBWZSA
```

example_apps/data

RXN and RD files that can be used for testing and development.

Appendix

RXN and RD file definition

The format of RXN files and RD files is defined in the documentation "CTFILE FORMATS - Biovia databases 2017" (Biovia, 2016).

RXN files represent reactions based on molfiles of reactants and products.

RD files (Reaction Data files) contain alphanumerical data, molecules and reactions describing the properties of one or more reactions. Each reaction section begins with "$RFMT" (necessary key word for the RInChI calculation). The reaction information may be hierarchical, e.g. one reaction may consists of multiple variations with each variation providing alphanumerical data about reaction conditions, yields, literature, etc., combined with structural information (molfiles) about catalysts, solvents and other reagents." $MFMT" is used by RInChI to identify single molfiles in the RD file sections.
While "CTFILE FORMATS" describe the general syntax of the RD files, the actual format depends on the data model that is underlying the data source from which the RD file has been exported. Consequently, the same reaction may be found in different RD formats.

Following formats are supported:

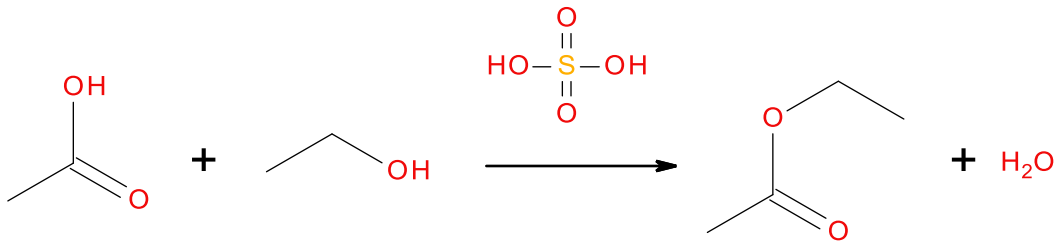
Flat RD file (no hierarchy)

Within each reaction section a RXN file is defined that may be followed by information about agents (catalysts, solvents). All structure data for one reaction are evaluated for the RInChI calculation by calculating the reactant and product section from the RXN part and adding all remaining molfiles to the agent section.

Note

To recognize the molfile describing the structure of the catalyst, you must comply with the molfile format for RD files with the header line $DATUM $MFMT as defined in the CT file format documentation (Biovia, 2016).

Example



Reaction 11: Esterification (Esterification_01)

The following RD file (*Esterification_ 01_flat.rdf*) was written as output from Biovia's Pipeline Pilot using the standard RD file writer with shortened V2 molfiles but manually adapted molfile section for the catalyst. Per default, the RD file writer of Pipeline Pilot 9.5 does not add the line $DATUM $MFMT automatically and does not keep the first (in most of the cases empty name) line of the molfile:

```
$RDFILE 1
$DATM  SciTegic  02/05/16 13:19:12
$RFMT
$RXN

SciTegic     0205161319

2  2
$MOL

SciTegic02051613192D
```

```
4  3  0  0  0  0            999 V2000
2.2188   -5.2500    0.0000 C   0  0
3.2416   -4.6594    0.0000 C   0  0
3.2416   -3.4780    0.0000 O   0  0
4.2648   -5.2502    0.0000 O   0  0
2  1  0
3  1  0
4  2  0
M  END
$MOL

SciTegic02051613192D

3  2  0  0  0  0            999 V2000
7.5313   -5.0313    0.0000 O   0  0
8.5541   -4.4407    0.0000 C   0  0
9.5773   -5.0314    0.0000 C   0  0
2  1  0
3  1  0
M  END
$MOL

SciTegic02051613192D

6  5  0  0  0  0            999 V2000
14.5625   -5.4688    0.0000 C   0  0
15.5854   -4.8782    0.0000 C   0  0
15.5854   -3.6967    0.0000 O   0  0
16.6085   -5.4689    0.0000 O   0  0
16.6085   -3.1060    0.0000 C   0  0
17.6317   -3.6967    0.0000 C   0  0
2  1  0
3  1  0
4  2  0
5  1  0
6  1  0
M  END
$MOL

SciTegic02051613192D

1  0  0  0  0  0            999 V2000
20.6563   -4.6250    0.0000 O   0  0
M  END
$DTYPE ID
$DATUM 1
$DTYPE CATALYST
$DATUM $MFMT

SciTegic02051613192D

5  4  0  0  0  0            999 V2000
4.8321   -5.7375    0.0000 O   0  0
4.8321   -4.9875    0.0000 S   0  0
4.8321   -4.2375    0.0000 O   0  0
5.5821   -4.9875    0.0000 O   0  0
4.0821   -4.9875    0.0000 O   0  0
2  1  0
3  1  0
4  2  0
5  2  0
M  END
```

```
RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d+
Long-RInChIKey=SA-FUHFF-QTBSBXVTEAMEQO-UHFFFAOYSA-N-LFQSCWFLJHTTHZ-UHFFFAOYSA-N--XEKOWRVHYACXOJ-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N--QAOWNCQODCNURD-UHFFFAOYSA-N
Short-RInChIKey=SA-FUHFF-JJFIATRHOH-UDXZTNISGZ-QAOWNCQODC-NKDYL-NKDYL-NMOSF-ZZZ
Web-RInChIKey=SMUHAWIQPXIVCEVKG-NJYTGTKWPQNQUSA
```

Hierarchical RD files

Most of the hierarchical RD files have in common that a variation level is defined: Each reaction is described by one or more variations where each variation is defined by its conditions, yield, catalysts, solvents, etc.

The RInChi calculation only handles one set of agents. In case of multiple variations for one reaction, multiple RInChI and RInChIKeys must be calculated to represent all combinations of the reaction and all related variations.

The standard RInChI calculator only evaluates the first variation of any reaction in the RD file neglecting the remaining variations..

Note

In reaction databases containing the variation level of chemical reactions the RInChI and RInChIKeys are properties of the variation level and not of the reaction in the top level of the data hierarchy. If you have to add a RInChI/RInChIKey to the top level of a hierarchical database, RInChI must only be calculated from the RXN section of the RD file omitting the agent information.

Example

For the esterification reaction above (Reaction 11) you get the following RD file exported from a local ISIS reaction database using the data model of the local reaction database of MDL/ISIS version 2.5sp5 (See example file *Esterification_01.rdf*). The catalyst is assumed to be a property of the first variation of this reaction:

```
$RDFILE 1
$DATM 2/11/2016 0:47:3
$RFMT $RIREG 180
$RXN

ISIS     021120160047
2  2
$MOL

-ISIS-  02111600472D

3  0  0  0  0  0  0  0  0999 V2000
-3.8549   -0.5552    0.0000 C   0  0  0  0  0  0  0  0  0  1  0  0
-2.8321    0.0354    0.0000 C   0  0  0  0  0  0  0  0  0  2  0  0
-2.8321    1.2168    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
-1.8089   -0.5554    0.0000 O   0  0  0  0  0  0  0  0  0  4  0  0
2  1  0  0  0  2
3  1  0  0  0  6
4  2  0  0  0  2
M  END
$MOL

-ISIS-  02111600472D

2  0  0  0  0  0  0  0  0999 V2000
2.4990   -0.1614    0.0000 O   0  0  0  0  0  0  0  0  0  8  0  0
1.4762    0.4292    0.0000 C   0  0  0  0  0  0  0  0  0  6  0  0
0.4530   -0.1615    0.0000 C   0  0  0  0  0  0  0  0  0  7  0  0
2  1  0  0  0  4
3  1  0  0  0  2
M  END
$MOL

-ISIS-  02111600472D

5  0  0  0  0  0  0  0  0999 V2000
7.2384   -1.0475    0.0000 C   0  0  0  0  0  0  0  0  0  1  0  0
8.2613   -0.4569    0.0000 C   0  0  0  0  0  0  0  0  0  2  0  0
8.2613    0.7246    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
9.2844    1.3153    0.0000 C   0  0  0  0  0  0  0  0  0  6  0  0
10.3076    0.7246    0.0000 C   0  0  0  0  0  0  0  0  0  7  0  0
9.2844   -1.0476    0.0000 O   0  0  0  0  0  0  0  0  0  4  0  0
2  1  0  0  0  2
3  1  0  0  0  6
4  1  0  0  0  4
5  1  0  0  0  2
6  2  0  0  0  2
M  END
$MOL

-ISIS-  02111600472D

0  0  0  0  0  0  0  0  0999 V2000
12.5696    0.0354    0.0000 O   0  0  0  0  0  0  0  0  0  8  0  0
M  END
$DTYPE RXN:RXNREGNO
$DATUM 180
$DTYPE RXN:VARIATION(1):VARIATION_NO
$DATUM 1
$DTYPE RXN:VARIATION(1):CATALYST(1):CATALYST_NO
$DATUM 1
$DTYPE RXN:VARIATION(1):CATALYST(1):MOL(1):MOLSTRUCTURE
$DATUM $MFMT

-ISIS-  02111600472D

4  0  0  0  0  0  0  0  0999 V2000
6.4257   -2.7792    0.0000 S   0  0  3  0  0  0  0  0  0  0  0  0
7.1757   -2.7792    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
5.6757   -2.7792    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
6.4257   -3.5292    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
6.4257   -2.0292    0.0000 O   0  0  0  0  0  0  0  0  0  0  0  0
2  1  0  0  0  0
3  1  0  0  0  0
4  2  0  0  0  0
5  2  0  0  0  0
M  END
```

RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d+

Long-RInChIKey=SA-FUHFF-QTBSBXVTEAMEQO-UHFFFAOYSA-N-LFQSCWFLJHTTHZ-UHFFFAOYSA-N--XEKOWRVHYACXOJ-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N--QAOWNCQODCNURD-UHFFFAOYSA-N

Short-RInChIKey=SA-FUHFF-JJFIATRHOH-UDXZTNISGZ-QAOWNCQODC-NKDYL-NKDYL-NMOSF-ZZZ

Web-RInChIKey=SMUHAWIQPXIVCEVKG-NJYTGTKWPQNQUSA

Comparison of the RInChIs / RinChIKeys shows that the flat and hierarchical files deliver identical results. Although the underlying data model does differ (flat versus hierarchical model), RInChI and RInChIKeys are equal.

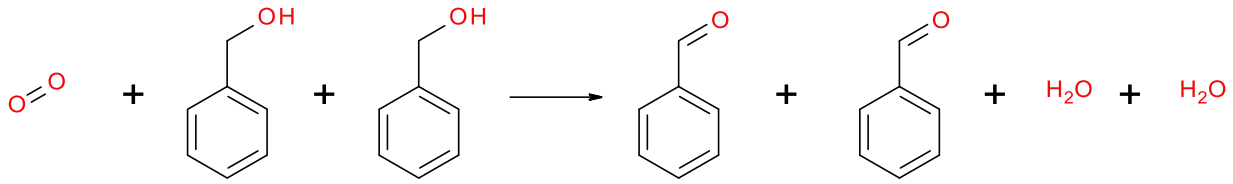
Multiple steps or stages

Each variation may consists of one or more "steps" or "stages" indicating that multiple steps must be taken to synthesize the products of this variation. These data are comprised on variation level or in one or more sublevels of the variation.

The RInChI algorithm interprets all molfiles of the first variation as agents regardless whether they are stored on the variation level or any sublevel of the variation layer. Because InChIs of these molfiles are ordered alphabetically, the original order of the molecules in the "steps"/"stages" is lost, i.e. these steps cannot be reconstructed from the RInChI any more.

Example

The following reaction is described by Wang Feng, Ueda Wataru, and Xu Jie (Feng, Wataru, & Jie, 2012) taken from the SPRESI database, reaction regno: 4407650.



Reaction 12: Oxidation of phenlymethanol (Multiplesteps.rdf)

Step/Stage 1: 3-4 h, 20 degree, trifluoromethanesulfonic acid silver salt, Iron(II) chloride, dichloromethane, acetonitrile, C6H15N diisopropylamine |
Step/Stage 2: 14 h, 50 degree, tetrahydrofuran, acetonitrile, CK2O3 potassium carbonate, O2 oxygen, CK2O3 potassium carbonate

Although two steps/stages are involved in this reaction, all components being part of the steps/stages are equally listed as agents ordered alphabetically. Therefore, the original assignments to the two stages are lost.

RInChI=1.00.1S/C7H6O/c8-6-7-4-2-1-3-5-7/h1-6H!C7H6O/c8-6-7-4-2-1-3-5-7/h1-6H!H2O/h1H2!H2O/h1H2<>C7H8O/c8-6-7-4-2-1-3-5-7/h1-5,8H,6H2!C7H8O/c8-6-7-4-2-1-3-5-7/h1-5,8H,6H2!O2/c1-2<>2C1H.Fe/h2*1H;/q;;+2/p-2!C2H3N/c1-2-3/h1H3!C2H3N/c1-2-3/h1H3!C3H6Cl2/c1-3(2,4)5/h1-2H3!C4H8O/c1-2-4-5-3-1/h1-4H2!C6H15N/c1-5(2)7-6(3)4/h5-7H,1-4H3!C7H5F3O3S.Ag/c8-7(9,10)5-1-3-6(4-2-5)14(11,12)13;/h1-4H,(H,11,12,13);/q;+1/p-1!CH2O3.2K/c2-1(3)4;;/h(H2,2,3,4);;/q;2*+1/p-2/d-

Long-RInChIKey=SA-BUHFF-HUMNYLRZRPPJDN-UHFFFAOYSA-N-HUMNYLRZRPPJDN-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N--WVDDGKGOMKODPV-UHFFFAOYSA-N-WVDDGKGOMKODPV-UHFFFAOYSA-N-MYMOFIZGZYHOMD-UHFFFAOYSA-N--NMCUIPGRVMDVDB-UHFFFAOYSA-L-WEVYAHXRMPXWCK-UHFFFAOYSA-N-WEVYAHXRMPXWCK-UHFFFAOYSA-N-ZEOVXNVKXIPWMS-UHFFFAOYSA-N-WYURNTSHIVDZCO-UHFFFAOYSA-N-UAOMVDZJSHZZME-UHFFFAOYSA-N-SLXXHLHFINYNPJ-UHFFFAOYSA-M-BWHMMNNQKKPAPP-UHFFFAOYSA-L

Short-RInChIKey=SA-BUHFF-XIWYQVNKGG-IKDRUMAWLO-WGDMUZGRJH-NUHFF-NUHFF-IUHFF-ZZZ

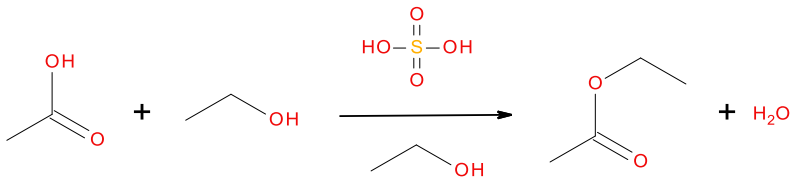
Web-RInChIKey=BYOIFZLGJDOQEXHJI-IUHFFFADPSCTJSA

Separation between agents and reactants or products

Some of the reactants and products may be agents of the reaction as well. RInChI calculation does not unify the reaction in these cases but keeps the role of the compounds as the author defines it with the exception of the Web-RInChIKey where each component only occurs once alphabetically ordered and without the original role in the reaction.

Example

The esterification above is run in ethanol where ethanol plays the role of a solvent as well (see *Esterification_02.rdf*)



Reaction 13: Esterification (Esterification_02.rdf)

RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>**C2H6O/c1-2-3/h3H,2H2,1H3**!H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d+

Long-RInChIKey=SA-FUHFF-QTBSBXVTEAMEQO-UHFFFAOYSA-N-LFQSCWFLJHTTHZ-UHFFFAOYSA-N--XEKOWRVHYACXOJ-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N--**LFQSCWFLJHTTHZ-UHFFFAOYSA-N**-QAOWNCQODCNURD-UHFFFAOYSA-N

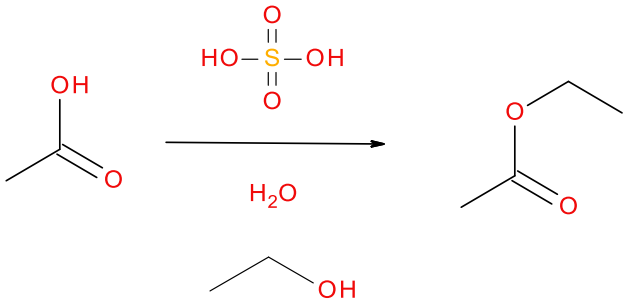
Short-RInChIKey=SA-FUHFF-JJFIATRHOH-UDXZTNISGZ-**UAUFKIWNBD**-NKDYL-NKDYL-NKDYL-ZZZ

Web-RInChIKey=SMUHAWIQPXIVCEVKG-NJYTGTKWPQNQUSA

Compared with the RInChI of the first esterification example (Reaction 3, Esterification_01.rdf), the InChI for ethanol is additionally found in the agent sections of RInChI; consequently, the InChIkey of ethanol is written into the agent section of the Long-RINChIKey and the agent section of the Short-InChIKey reflects the additional C_2H_5OH as well. The Web-RInChIKey for both reactions is identical, because it does not reflect the number of occurrences and the roles of the molecules participating in the reaction,

Example

Reaction 10 is only described by the main reactant and product of the ersterification reaction:



Reaction 10 (Esterification_03.rdf)

RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3<>**C2H6O/c1-2-3/h3H,2H2,1H3!H2O/h1H2**!H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d+

Long-RInChIKey=SA-FUHFF-QTBSBXVTEAMEQO-UHFFFAOYSA-N--XEKOWRVHYACXOJ-UHFFFAOYSA-N--**LFQSCWFLJHTTHZ-UHFFFAOYSA-N-XLYOFNOQVPJJNP-UHFFFAOYSA-N**-QAOWNCQODCNURD-UHFFFAOYSA-N

Short-RInChIKey=SA-FUHFF-QTBSBXVTEA-XEKOWRVHYA-**DNBJJWMYJT**-NMOSF-NMOSF-**NYAOM**-ZZZ

Web-RInChIKey=SMUHAWIQPXIVCEVKG-NJYTGTKWPQNQUSA

While RInChI, Long-RInChIKey and Short-RInChIKey reflect the changes (more agents less reactants and products), the Web-RInChIKey is identical for the esterification reactions 10, 11 and 13 described in this paper.

Limitations for the technical recognition of RXN/RD files

To recognize a file being a RD or RXN file, the first 1000 lines are scanned for the occurrence of $RFMT (RD file) or $RXN (RXN file). If these keywords are not found, the RInChI calculator will return an error message.

RXN and RD file regeneration from RInChI

The generation of a RXN or RD-Files from RInChI depends on the availability of the RInChI AuxInfo containing the coordinates of the structures from the original reaction in the RXN or RD file.

The original RXN file can be fully regenerated from RInChI and RInChI AuxInfo.

The coordinates and the original stereochemistry (if available)[18] of the reaction components and all agents are rebuilt form RInChI and RInChI AuxInfo, but the original role of an agent (catalyst, solvent, etc.) is lost together with all other alphanumeric data stored in the original RD file. The returned RD file provides a hierarchical format with "Agents" as first sub-hierarchy level irrespective of the original RD file format being flat or hierarchical.

The following RD file is returned for the esterification shown under Reaction 11: Esterification (example *Esterification_01_RInChI.txt*):

```
$RDFILE 1
$DATM 03/14/2016 21:37:04
$RFMT
$RXN

      RInChI1.00

  2  2
$MOL
Reactant1
  InChIV10

  4  3  0  0  0  0  0  0  0  0  1 V2000
    2.2188   -5.2500    0.0000 C   0  0  0     0  0  0  0  0  0
    3.2416   -4.6594    0.0000 C   0  0  0     0  0  0  0  0  0
    3.2416   -3.4780    0.0000 O   0  0  0     0  0  0  0  0  0
    4.2648   -5.2502    0.0000 O   0  0  0     0  0  0  0  0  0
  1  2  1  0  0  0  0
  2  3  1  0  0  0  0
  2  4  2  0  0  0  0
M  END
$MOL
Reactant2
  InChIV10

  3  2  0  0  0  0  0  0  0  0  1 V2000
    7.5313   -5.0313    0.0000 O   0  0  0     0  0  0  0  0  0
    8.5541   -4.4407    0.0000 C   0  0  0     0  0  0  0  0  0
    9.5773   -5.0314    0.0000 C   0  0  0     0  0  0  0  0  0
  1  2  1  0  0  0  0
  2  3  1  0  0  0  0
M  END
$MOL
Product1
  InChIV10

  6  5  0  0  0  0  0  0  0  0  1 V2000
   14.5625   -5.4688    0.0000 C   0  0  0     0  0  0  0  0  0
   15.5854   -4.8782    0.0000 C   0  0  0     0  0  0  0  0  0
   15.5854   -3.6967    0.0000 O   0  0  0     0  0  0  0  0  0
   16.6085   -5.4689    0.0000 O   0  0  0     0  0  0  0  0  0
   16.6085   -3.1060    0.0000 C   0  0  0     0  0  0  0  0  0
   17.6317   -3.6967    0.0000 C   0  0  0     0  0  0  0  0  0
  1  2  1  0  0  0  0
```

[18] RInChI follows the rules InChI uses for the rebuild of the stereochemistry into molfiles. Because Standard InChI interprets all tetrahedral stereocenters as absolute centers (with Chiral flag), InChI and therefore all RInChIs and RInChIKeys differ from the original chirality for all molecules with tetrahedral stereocenters with relative stereochemistry (i.e. without Chiral flag). But using RAuxInfo to reconstruct the RXN and molfiles, the original status is fully recovered on RXN-/mol file level, i.e. the different representation in the RInChIs is only hardly to detect.

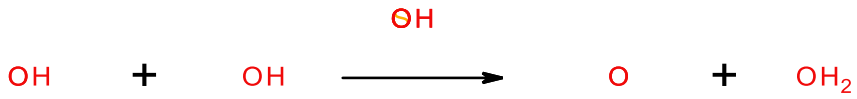
```
  2  3  1  0  0  0  0
  2  4  2  0  0  0  0
  3  5  1  0  0  0  0
  5  6  1  0  0  0  0
M  END
$MOL
Product2
  InChIV10

  1  0  0  0  0  0  0  0  0  0  1 V2000
   20.6563   -4.6250    0.0000 O   0  0  0     0  0  0  0  0  0
M  END
$DTYPE RXN:AGENTS(1):MOLECULES(1):MOLSTRUCTURE
$DATUM $MFMT
Agent1
  InChIV10

  5  4  0  0  0  0  0  0  0  0  1 V2000
    4.8321   -5.7375    0.0000 O   0  0  0     0  0  0  0  0  0
    4.8321   -4.9875    0.0000 S   0  0  0     0  0  0  0  0  0
    4.8321   -4.2375    0.0000 O   0  0  0     0  0  0  0  0  0
    5.5821   -4.9875    0.0000 O   0  0  0     0  0  0  0  0  0
    4.0821   -4.9875    0.0000 O   0  0  0     0  0  0  0  0  0
  1  2  1  0  0  0  0
  2  3  1  0  0  0  0
  2  4  2  0  0  0  0
  2  5  2  0  0  0  0
M  END
```

Without the data from RinChI AuxInfo, the RXN/RD file recalculator only rebuilds the connectivity tables of each molecule but sets all coordinates to 0.0000. A third party tool is needed to recalculate the coordinates of the molecules. The following example shows the recalculated RD file from above but without RInChI AuxInfo (example *Esterification_01_RInChI_No_RAux.txt*):

```
$RDFILE 1
$DATM 03/14/2016 21:37:04
$RFMT
$RXN

      RInChI1.00

  2  2
$MOL
Reactant1
  InChIV10

  4  3  0  0  0  0  0  0  0  0  1 V2000
    0.0000    0.0000    0.0000 C   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 C   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 O   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 O   0  0  0     0  0  0  0  0  0
  1  2  1  0  0  0  0
  2  3  2  0  0  0  0
  2  4  1  0  0  0  0
M  END
$MOL
Reactant2
  InChIV10

  3  2  0  0  0  0  0  0  0  0  1 V2000
    0.0000    0.0000    0.0000 C   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 C   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 O   0  0  0     0  0  0  0  0  0
  1  2  1  0  0  0  0
  2  3  1  0  0  0  0
M  END
$MOL
Product1
  InChIV10

  6  5  0  0  0  0  0  0  0  0  1 V2000
    0.0000    0.0000    0.0000 C   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 C   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 C   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 C   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 O   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 O   0  0  0     0  0  0  0  0  0
  1  3  1  0  0  0  0
  2  4  1  0  0  0  0
  3  6  1  0  0  0  0
  4  5  2  0  0  0  0
  4  6  1  0  0  0  0
M  END
$MOL
Product2
  InChIV10

  1  0  0  0  0  0  0  0  0  0  1 V2000
    0.0000    0.0000    0.0000 O   0  0  0     0  0  0  0  0  0
M  END
$DTYPE RXN:AGENTS(1):MOLECULES(1):MOLSTRUCTURE
$DATUM $MFMT
Agent1
  InChIV10
```

```
  5  4  0  0  0  0  0  0  0  0  1 V2000
    0.0000    0.0000    0.0000 O   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 O   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 O   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 O   0  0  0     0  0  0  0  0  0
    0.0000    0.0000    0.0000 S   0  0  0     0  0  0  0  0  0
  1  5  1  0  0  0  0
  2  5  1  0  0  0  0
  3  5  2  0  0  0  0
  4  5  2  0  0  0  0
M  END
```

The RD file is displayed in Biovia/Draw with all atoms of each molecule being overlaid on top of each other:

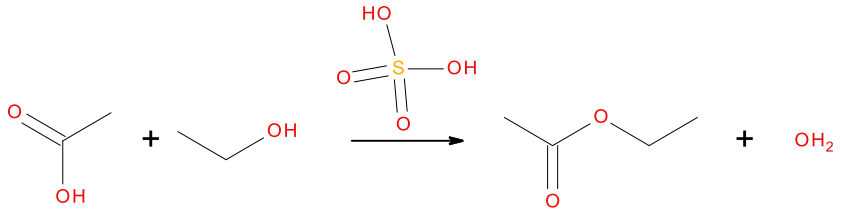


Reaction 11: rebuilt from RInChI without RAuxInfo

(Note: the RXN file and the agent H2SO4 were joined manually for this display of the reaction.)

Although all coordinates are set to 0.0000, drawing editors like ChemDraw, Biovia/Draw, or Marvin rebuild a reaction by moving the molecules according to its position in the RXN section of the RD file as translation along the x-axes of the canvas but they do not automatically recalculate all the coordinates of each compound.

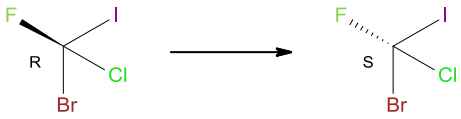
To return to the fully displayed molecules you may manually re-sketch each molecule or, for example, use the "clean" function of Biovia/Draw to recalculate the coordinates:



Reaction 11: rebuilt from RInChI without RAuxInfo with rebuilt coordinates

The test website http://proteax.dk/rinchi/rinchi.py/demo uses RDKit for the recalculation of the coordinates.

Without AuxInfo the InChI algorithm does not return the parity of a stereocenter into the recalculated molfile. Consequently, the RInChI calculator cannot return the stereochemistry for reactions from RInChI without RAuxInfo as demonstrated in the following example:

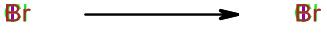


Reaction 14. Inversion reaction (Inverted_stereochemistry.rxn)

RInChi is calculated to

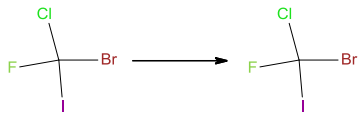
```
RInChI=1.00.1S/CBrClFI/c2-1(3,4)5/t1-/m0/s1<>CBrClFI/c2-1(3,4)5/t1-/m1/s1/d+
Long-RInChIKey=SA-FUHFF-XEGUVFFZWHRVAV-SFOWXEAESA-N--XEGUVFFZWHRVAV-PVQJCKRUSA-N
Short-RInChIKey=SA-FUHFF-XEGUVFFZWH-XEGUVFFZWH-UHFFFADPSC-NYRHR-NAYUW-NUHFF-ZZZ
Web-RInChIKey=OMADXNIUVSMMAGAIO-NUAXZUCYELSRHSA
```

Based on the RInChI (without RAuxInfo) the reaction is rebuilt with 0.0000 coordinates:



Reaction 14. Inversion reaction (Inverted_stereochemistry_No_Aux_recalc.rxn)

For example, Biovia/Draw 2017 recalculates the coordinates as shown in the following depiction:



Reaction 15: Reaction 14 rebuilt without RAuxInfo

Notes:

The stereocenters have vanished because the InChI algorithm does not return the centers from an InChI string without AuxInfo, so that the recalculated reaction by RInChI is returned without centers.

The halogenide atoms around the C-atom have been reordered by the calculation.

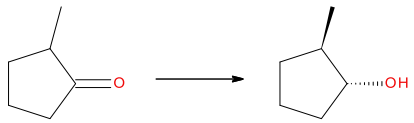
Limitations for RInChI caused by InChI restrictions

The calculation of RInChI is based on the standard version of InChI, version 1.04. Restrictions of InChIs will limit the functionality of RInChI as shown above for the recalculation of reactions from RInChI without RAuxInfo.

Representations of tetrahedral stereocenters by Standard InChIs

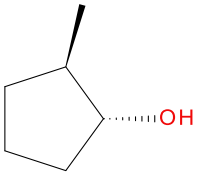
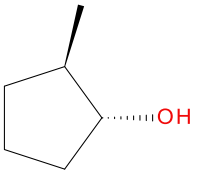
The Standard InChI algorithm uses the parameter /SAbs to calculate the InChI of structures with tetrahedral stereocenters independently from the actual chirality settings in the molfile. Consequently, all tetrahedral stereocenters are interpreted as absolute centers.[19]

That leads to serious limitations in the representation of reactions with components where only the relative stereochemistry is known, i.e. the related molfiles do not provide a Chiral flag as shown in the following example illustrating the reduction of 2-methylcyclopentanone:



Reaction 16: Reduction of 2-methylcyclopentanone

The rac-(1R,2R)-2-methylcyclopentanol does not have "absolute" stereocenters, i.e. the molfiile does not contain the Chiral flag. The InChI algorithm returns the following values for the Standard InChI:

Structure	InChI
 relative, trans	InChI=1S/C6H12O/c1-5-3-2-4-6(5)7/h5-7H,2-4H2,1H3/t5-,6-/m1/s1 AuxInfo=1/0/N:1,6,5,7,4,3,2/it:im/rA:7**n**COCCCCC/rB:;n2;p1s3;s4;s5;s3s6;/rC:9.2596,-4.1618,0;10.9199,-6.3794,0;9.6706,-6.3794,0;8.9363,-5.3685,0;7.748,-5.7546,0;7.748,-7.004,0;8.9362,-7.3901,0; InChIKey=BVIJQMCYYASIFP-PHDIDXHHSA-N
 absoulte, trans	InChI=1S/C6H12O/c1-5-3-2-4-6(5)7/h5-7H,2-4H2,1H3/t5-,6-/m1/s1 AuxInfo=1/0/N:1,6,5,7,4,3,2/it:im/rA:7**c**COCCCCC/rB:;n2;p1s3;s4;s5;s3s6;/rC:12.4945,-6.0577,0;14.1548,-8.2753,0;12.9055,-8.2753,0;12.1712,-7.2644,0;10.9829,-7.6505,0;10.9829,-8.8999,0;12.1711,-9.286,0; InChIKey=BVIJQMCYYASIFP-PHDIDXHHSA-N

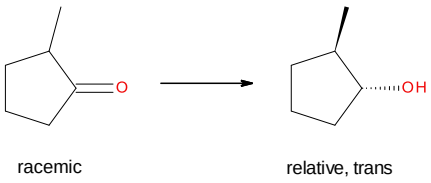
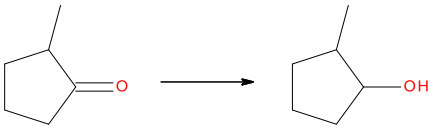
[19] The recalculation of a structure from its InChI and its AuxInfo always rebuilds the original structure.

Both compounds only differ in the AuxInfo ("n" versus "c") but not in the InChI string nor in the InChIKey.[20]

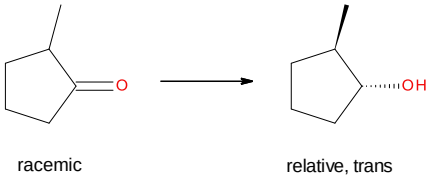
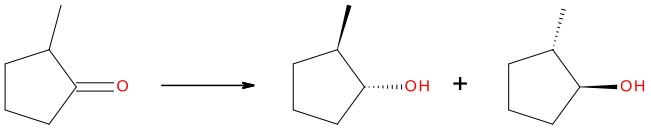
Because the RInChI algorithm is based on InChI and InChIKey, **RInChI is not capable to handle any reaction with components containing relative tetrahedral stereocenters** like the reduction of 2-methylcyclopentanone.

Potential Workarounds:

Skip the relative stereochemistry

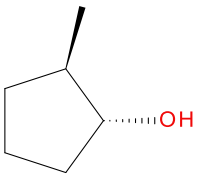
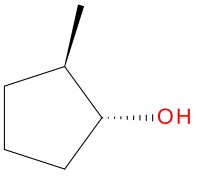
Original reaction	Reaction without stereocenters
 racemic relative, trans	

or draw all relative components explicitly

Original reaction	Reaction with all stereoactive components drawn explicitly
 racemic relative, trans	

Alternative approach by leaving the Standard InChI definition

While leaving the Standard InChI definition makes the RInChI less searchable by those widely spread tools building (standard) InChIs, the InChI algorithm provides an out-of-the-box solution for reactions with components containing relative tetrahedral stereocenters by using the InChI parameter /SUCF (Stereo Use Chiral Flag) parameter. This parameter let InChI distinguish between the absolute (with Chiral flag) and relative (without Chiral flag) stereocenters automatically:

Structure	InChI calculated with the parameter /SUCF
 relative, trans	InChI=1/C6H12O/c1-5-3-2-4-6(5)7/h5-7H,2-4H2,1H3/t5-,6-/s2 AuxInfo=1/0/N:1,6,5,7,4,3,2/rA:7nCOCCCCC/rB:;n2;p1s3;s4;s5;s3s6;/rC:9.2596,-4.1618,0;10.9199,-6.3794,0;9.6706,-6.3794,0;8.9363,-5.3685,0;7.748,-5.7546,0;7.748,-7.004,0;8.9362,-7.3901,0; InChIKey=BVIJQMCYYASIFP-IOMOGOHMNA-N
 absoulte, trans	InChI=1/C6H12O/c1-5-3-2-4-6(5)7/h5-7H,2-4H2,1H3/t5-,6-/m1/s1 AuxInfo=1/0/N:1,6,5,7,4,3,2/it:im/rA:7cCOCCCCC/rB:;n2;p1s3;s4;s5;s3s6;/rC:12.4945,-6.0577,0;14.1548,-8.2753,0;12.9055,-8.2753,0;12.1712,-7.2644,0;10.9829,-7.6505,0;10.9829,-8.8999,0;12.1711,-9.286,0; InChIKey=BVIJQMCYYASIFP-PHDIDXHHNA-N

[20] Because AuxInfo keeps the information, both compounds can be fully rebuilt from InChI and AuxInfo.

Both InChI and InChIKey differ in this case so that RInChI using the parameter /SUCF is capable to differentiate between relative and absolute centers. (See marks in red). Because the area of Standard InChI is left, the versioning is modified as shown in blue.

This approach has not been implemented for the Standard RInChI algorithm.

No support of V3 molfiles

Because the Standard InChI version 1.04 does not support V3-specific molfile features, RInChI is not capable to handle V3 RXN files or RD files with V3 components. That includes big molecules (with more than 999 heavy atoms), any structures with coordinative or hydrogen bonds, and Biovia's enhanced stereochemistry with AND# and OR# stereocenters.

Query bonds

InChI does not support the following query bond types: single or double, single or aromatic, and double or aromatic.

As InChI is capable to analyze structures with aromatic bonds, RInChI and all related keys are calculated for reactions with aromatic bonds although these bonds are seen as query bonds in the Biovia (MDL) context.

Support of pseudo and query atoms

InChI does not support R, X, * and the query atoms A and Q. Therefore, RInChIs cannot be calculated for reactions with molecules containing these atoms.

Note: In case that A, R, X, or * atoms are "stand alone" atoms that are not bound to any other atom, they are treated like No-structures being seen as placeholders for compounds that cannot be represented by chemical structures. (see rules for No-structures in "No-structures and pseudoatoms representing components that cannot be structurally drawn")

Sgroups, abbreviations (shortcuts, residues) and Sgroup data

Sgroup brackets are generally ignored and InChI is calculated for the underlying full structural unit. That leads to false results for SRUs (polymer representation) because the polymer consists of n units while the drawn chemical structure only represents the fraction of 1/n of the full polymer. Multiple groups can be seen as exceptions in this case because technically seen a multiple group "physically" overlays the structures defining the multiple group m times with m representing the factor of the multiple group. Because the InChI is calculated over the full underlying structure of the group, the InChI correctly represents the multiple group.

Abbreviations (residues, shortcuts) are a special case of Sgroups. InChI always represents the underlying structure of the Sgroup and ignores the text of the abbreviation that is set over this structural section.

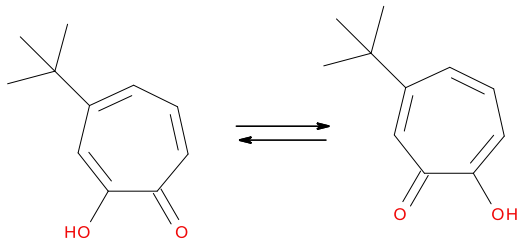
The Biovia extended SCSR V3 molfile format is not recognized by InChI.

Sgroup data are ignored for the InChI calculation, and only the related chemical structure is reflected in the InChI.

Tautomerization reactions

InChIs are optimized to build identical InChIs for different tautomers of the same compound. Consequently, in some of the cases the InChI for reactants and products of tautomerization reactions may become identical though they have to represent two different entities like in the case of 4-tert-butyl-2-

hydroxy-cyclohepta-2,4,6-trien-1-one and 6-tert-butyl-2-hydroxy-cyclohepta-2,4,6-trien-1-one (example: *Tautomerization_01.rxn*):



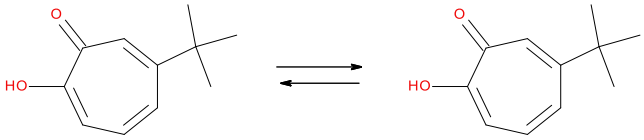
RInChI=1.00.1S/C11H14O2/c1-11(2,3)8-5-4-6-9(12)10(13)7-8/h4-7H,1-3H3,(H,12,13)<>C11H14O2/c1-11(2,3)8-5-4-6-9(12)10(13)7-8/h4-7H,1-3H3,(H,12,13)/d=

Long-RInChIKey=SA-EUHFF-YCJXIOAKVCHNQZ-UHFFFAOYSA-N--YCJXIOAKVCHNQZ-UHFFFAOYSA-N

Short-RInChIKey=SA-EUHFF-YCJXIOAKVC-YCJXIOAKVC-UHFFFADPSC-NMOSF-NMOSF-NUHFF-ZZZ

Web-RInChIKey=YCJXIOAKVCHNQZAOJ-NMOSFIJXAXDLOSA

Note: If you rebuild this reaction based on RInChI only (without RAuxInfo), the reaction will only show one of the 2 components that plays the role of reactant and product:



Connectivity of metal atoms

InChIs do not reflect the connectivity of metal atoms in chemical structures. In special cases like rearrangement reactions, the reactants and products may become identical.

No-structures and pseudoatoms representing components that cannot be structurally drawn

No structures, R, X, A and * atoms representing a compound that participates in a reactions but cannot be represented as chemical structure.

Examples:

The following reaction (*nostruct_-_X.rxn*) contains a reactant and a product that cannot be represented by a chemical structure but are uniquely described somewhere else.

RInChI=1.00.1S//d+/u1-1-0

Long-RInChIKey=SA-FUHFF-MOSFIJXAXDLOML-UHFFFAOYSA-N--MOSFIJXAXDLOML-UHFFFAOYSA-N

Short-RInChIKey=SA-FUHFF-UHFFFADPSC-UHFFFADPSC-UHFFFADPSC-NUHFF-NUHFF-NUHFF-AAZ

Web-RInChIKey=MOSFIJXAXDLOMLMKR-NMOSFIJXAXDLOSA

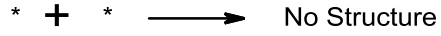
Accordingly, the reactant and product side of RInChI does not contain InChI strings while the counter of no structures displays one no structure for reactants and one no structure in the product section. The Long-RInChIKey contains the hash value for empty strings for the reactant and the product, the Short-RInChIKey contains the hashed no-structures for reactant, product and agent. The Web-RInChIKey represent the hash value for one single no-structure.

Because no structures, R, X, A and * atoms are not distinguished, the following reaction (*R-_-A.rxn*) returns identical RInChIs:

RInChI=1.00.1S//d+/u1-1-0

```
Long-RInChIKey=SA-FUHFF-MOSFIJXAXDLOML-UHFFFAOYSA-N--MOSFIJXAXDLOML-UHFFFAOYSA-N
Short-RInChIKey=SA-FUHFF-UHFFFADPSC-UHFFFADPSC-UHFFFADPSC-NUHFF-NUHFF-NUHFF-AAZ
Web-RInChIKey=MOSFIJXAXDLOMLMKR-NMOSFIJXAXDLOSA
```

Similarly, the following reaction (*star_star_-_nostruct.rxn*) only differs in the number of no structures for RInChI and Short-RInChIKey while the Long-RInChIKey contains an additional hashed no structure representation. The Web-RInChIKey is identical to the other two reactions because it contains each type of molecule only once.

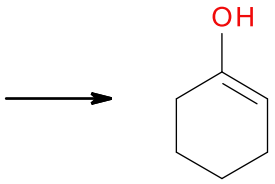



```
RInChI=1.00.1S//d+/u2-1-0
Long-RInChIKey=SA-FUHFF-MOSFIJXAXDLOML-UHFFFAOYSA-N-MOSFIJXAXDLOML-UHFFFAOYSA-N--MOSFIJXAXDLOML-
UHFFFAOYSA-N
Short-RInChIKey=SA-FUHFF-UHFFFADPSC-UHFFFADPSC-UHFFFADPSC-NUHFF-NUHFF-NUHFF-BAZ
Web-RInChIKey=MOSFIJXAXDLOMLMKR-NMOSFIJXAXDLOSA
```

Note: R, X, A and * atoms being part of a chemical structure (i.e. they are bound to other atoms) are not handled by InChI and will invoke an error message

Half and empty reactions

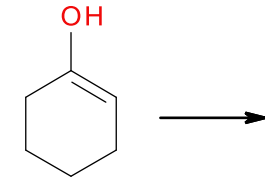
There are a number of reactions without any knowledge about the chemical structure of any reactant or product like in the following example (*No_reactant_-_1_product.rxn*) without reactant:



RInChIs are calculated by leaving the related position being empty. Because there is no structure defined (even not a no-structure), the no-structure counter at the end of RInChI is set to zero.

```
RInChI=1.00.1S/<>C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2/d+
Long-RInChIKey=SA-FUHFF---QHDHNVFIKWGRJR-UHFFFAOYSA-N
Short-RInChIKey=SA-FUHFF-UHFFFADPSC-QHDHNVFIKW-UHFFFADPSC-NUHFF-NMOSF-NUHFF-ZZZ
Web-RInChIKey=QHDHNVFIKWGRJRNLA-NMOSFIJXAXDLOSA
```

A similar case exists for a reactant leading to undefined product(s) like in *1_reactant_-_no_product.rxn*:




```
RInChI=1.00.1S/C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2/d+
Long-RInChIKey=SA-BUHFF---QHDHNVFIKWGRJR-UHFFFAOYSA-N
Short-RInChIKey=SA-FUHFF-QHDHNVFIKW-UHFFFADPSC-UHFFFADPSC-NMOSF-NUHFF-NUHFF-ZZZ
Web-RInChIKey=QHDHNVFIKWGRJRNLA-NMOSFIJXAXDLOSA
```

In case neither the reactants nor the products are known (example*: No_reactant_-_no_product.rxn*) you get

with the following keys

```
RInChI=1.00.1S//d+
RAuxInfo=1.00.1/
Long-RInChIKey=SA-FUHFF
Short-RInChIKey=SA-FUHFF-UHFFFADPSC-UHFFFADPSC-UHFFFADPSC-NUHFF-NUHFF-NUHFF-ZZZ
Web-RInChIKey=UHFFFADPSCTJAUYIS-NUHFFFADPSCTJSA
```

Notes for the Long-RInChIKey:

The InChIKeys for empty products or reactants are omitted. Because of the sorting rules, the format of reactions with missing reactant and those with missing product becomes identical except that the direction of both reaction types differs as described in the two examples taken from above:

Half-reaction Example I: no reactant but product is defined

```
Long-RInChIKey=SA-FUHFF---QHDHNVFIKWGRJR-UHFFFAOYSA-N as Forward reaction
```

Half-reaction Example I1: reactant but product is empty

```
Long-RInChIKey=SA-BUHFF---QHDHNVFIKWGRJR-UHFFFAOYSA-N as Backward reaction
```

In case of the no reactant to no product reaction only the basic parts of the Long-RInChIKey are kept.

Notes for Short-RInChhIkey:

Short-RInChIkeys of half-reactions and reactions with no-structures as reactant or product at the related positions do only differ in the no-structure count at the end of the key

Equilibrium Reactions

To handle a reaction as equilibrium reaction, the related parameter "in_force_equilibrium" of the function rinchi_from_file_text or rinchikey_from_file_text must be set to TRUE (default FALSE).

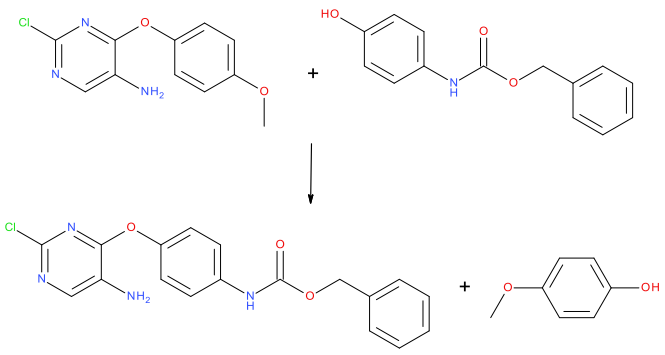
Note: There is no defined place for the equilibrium parameter in the RXN file definition.

For equilibrium reactions the comment "NOTE: Reaction is an equilibrium reaction." is returned in the comment section of the rebuilt RXN/RD file

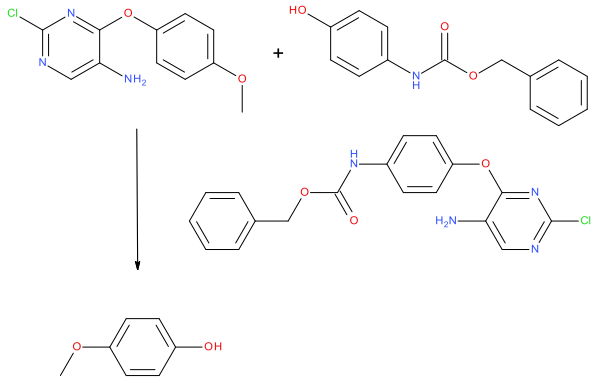
```
$RXN

      RInChI1.00
NOTE: Reaction is an equilibrium reaction.
  2  2
$MOL
…
```

Results of Web-RInChIkey comparisons

Out of the 424.626 reactions published in the patent database by NextMove based on the material of the US Patent Office from 2008 to 2011, RInChIKeys could be calculated for 406.138 reactions. All Long-RInChIKeys and Short-RInChIKeys were unique, but 3613 reactions showed duplicates for Web-RInChIKeys like in the following two examples where one of the two major products of the first example becomes an agent in the second example



'paragraphNum': '0673', 'documentId': 'US20130309196A1'



'paragraphNum': '1164', 'documentId': 'US20110105509A1'

Comparison of RInChI Version 0.02 with the current version

While version 0.02 was programmed in Python using a Python based sha hashing algorithm, the current version is programmed in C++ and is fully based on the InChI release 1.04 (standard version). The most obvious change is caused by moving to the InChI hashing method that delivers different hash codes than the Python method of version 0.02. Consequently, the values of the RInChIKeys for version 0.02 are not identical to the RInChIKeys of the current version for identical reactions.

Unlike the Python version 0.02, version 1.00 does not automatically convert any agents into other roles; it entirely follows the role assignments by the author of the reaction. Accordingly, in version V1.00 all agent information is handled in the third section of RInChI and RInChIKey only.

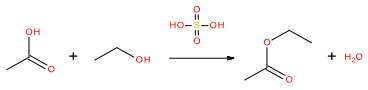
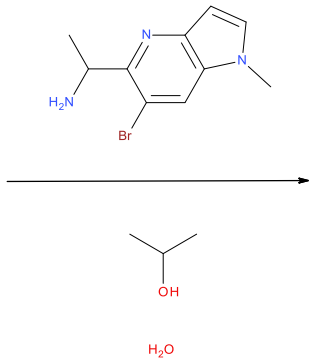
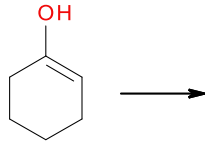
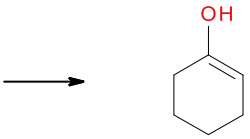
To include "half-reactions", no-structures, and pseudoatoms representing full structures and to enhance the search- and readability of RInChI and RInChIKeys some minor modifications had to be added to the current RInChI version:

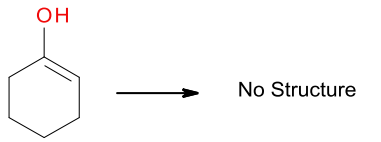
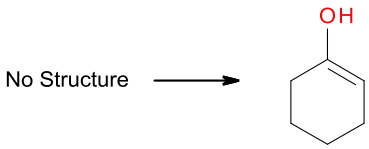
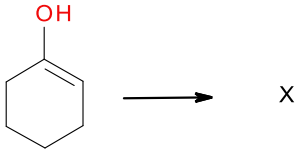
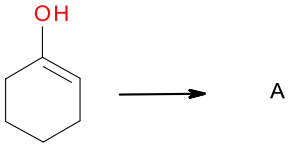
RInChI

The molecule and group separators were modified to enhance the readability and to avoid ambiguities in case of no-structures and half-reactions. The no-structure layer has been added at the end of RInChI if applicable.

In version 0.02 the pseudoatom X has been used to represent structures that could not be calculated by the RInChI algorithm. That includes no-structures. In the current version the no-structure representation follows the standard InChI representation

Version	Version 0.02	Current version
Version layer	RInChI = 0.02.1.S/	RInChI=1.00.1S/
Component separator	//	!
Group separator	///	<>
Reaction direction layer	/d+, /d-, or /d=	/d+, /d-, or /d=
No-structure layer, if applicable		/u#1-#2-#3 with #1 number of no-structures in the first group (mostly reactants), #2 number of no-structures in the second group (mostly products), and #3 number of no-structures in the third group (agents). This layer is only added in case that any no-structure is found in the reaction.

Examples	Version 0.02	Current version
 (Esterification_01.rdf)	RInChI=0.02.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)//C2H6O/c1-2-3/h3H,2H2,1H3///C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3//H2O/h1H2///H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d+	RInChI=1.00.1S/C2H4O2/c1-2(3)4/h1H3,(H,3,4)!C2H6O/c1-2-3/h3H,2H2,1H3<>C4H8O2/c1-3-6-4(2)5/h3H2,1-2H3!H2O/h1H2<>H2O4S/c1-5(2,3)4/h(H2,1,2,3,4)/d+
 (No_Structure_0-02.rdf)	RInChI=0.02.1S///////C10H12BrN3/c1-6(12)10-7(11)5-9-8(13-10)3-4-14(9)2/h3-6H,12H2,1-2H3//C3H8O/c1-3(2)4/h3-4H,1-2H3//H2O/h1H2/d+	RInChI=1.00.1S/<><>C10H12BrN3/c1-6(12)10-7(11)5-9-8(13-10)3-4-14(9)2/h3-6H,12H2,1-2H3!C3H8O/c1-3(2)4/h3-4H,1-2H3!H2O/h1H2/d+
 (1_reactant_-_no_product.rxn)	RInChI=0.02.1S////C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2/d-	RInChI=1.00.1S/<>C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2/d-
 (No_reactant_-_1_product.rxn)	RInChI=0.02.1S////C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2/d+	RInChI=1.00.1S/<>C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2/d+

Examples	Version 0.02	Current version
 (1_reactant_-_no_structure.rxn)	RInChI=0.02.1S/C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2///X/d+	RInChI=1.00.1S/<>C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2/d-/u1-0-0
 (no_structure_-_1_product.rxn)	RInChI=0.02.1S/C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2///X/d-	RInChI=1.00.1S/<>C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2/d+/u1-0-0
 (1_reactant_-_X.rxn)	RInChI=0.02.1S/C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2///X/d+	RInChI=1.00.1S/<>C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2/d-/u1-0-0
 (1_reactant_-_A.rxn)	RInChI=0.02.1S/C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2///X/d+	RInChI=1.00.1S/<>C6H10O/c7-6-4-2-1-3-5-6/h4,7H,1-3,5H2/d-/u1-0-0

RInChIKeys

Due to the switch from the Python based SHA-2 hashing algorithm to the InChI hash functions all RInChIKeys of the current version differ from those of version 0.02.

Version 0.02 provides version A and version B. Because the reaction direction is an important information for the key, version A (direction is encoded in the hashed string) was skipped for this version so that only version B is used in the current release.

Long-RInChKeys

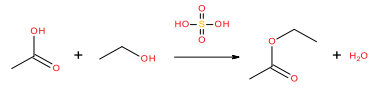
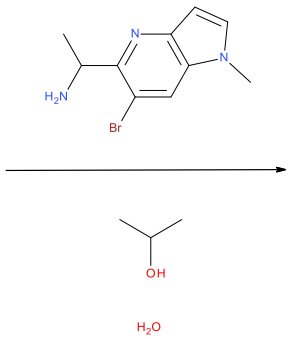
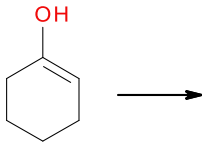
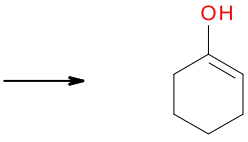
In the first section of version 0.02 the first character described the key version A or B. Because version A is not continued, the first letter becomes obsolete. 'SA' continues to point to the usage of the standard InChI functionality (S) of the InChI release, version 1.0(4) (=A).

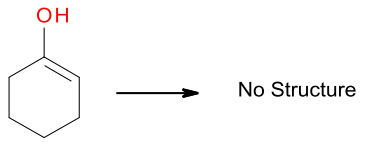
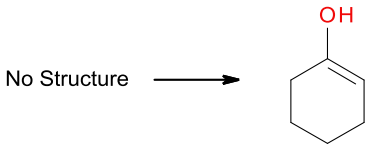
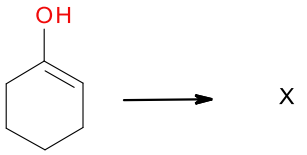
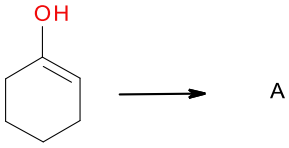
In the second layer, "U" for the unspecified reaction was skipped. The RXN/RD file format only allows forward reactions; backward reactions must be saved as forward reactions in RXN/RD files. That leaves "d+" or "F" as default, unless group 1 and group 2 are not reordered because of the ascending ordering (case of "d-" or "B"), or the equilibrium parameter is explicitly used ("d=" or "E"). Because the RXN/RD files do not know "unspecified" this parameter is not used in the current RInChI version.

The last 4 characters of the second layer are not used and left open for the potential later integration of alphanumerical data of reactions like temperature or yield. The "UHFF" of the current versions (former "EANN") represents the first 4 characters of the hash value for an empty string.

The Long-RInChIKey consists of the InChIKeys of the participating molecules. Unlike in the RInChI version 0.02, the current version uses full InChIKeys including the version letters "SA" at the end of the second layer of InChIKey although this information is redundant in the context of the Long-RInChIKey with the "SA" identification for the InChI version at the beginning of the string.
Note: This extension simplifies searches of single molecules in reactions because the InChIKey string of any InChI creator can be used for the text search over Long-RInChIKeys without the need to modify the InChIKey by deleting the "SA" at the end of the second layer of the InChIKey.

Version	Version 0.02	Current version
Version layer	Long-RInChIKey=aSA-**FEANN (Forward reaction)** Long-RInChIKey=aSA-**BEANN (Backward reaction)** Long-RInChIKey=aSA-**EEANN (Equilibrium reaction)** Long-RInChIKey=aSA-**UEANN (Unspecified reaction)**	Long-RInChIKey=SA-**FUHFF (Forward reaction)** Long-RInChIKey=SA-**BUHFF (Backward reaction)** Long-RInChIKey=SA-**EUHFF (Equilibrium reaction)**
Component separator	-	-
Group separator	--	--
Example: esterification component acetic acid	QTBSBXVTEAMEQO-UHFFFAOY-N	QTBSBXVTEAMEQO-UHFFFAOYSA-N

Examples	Version 0.02	Current version
 (Esterification_01.rdf)	Long-RInChIKey=bSA-FEANN- QTBSBXVTEAMEQO-UHFFFAOY-N- LFQSCWFLJHTTHZ-UHFFFAOY-N-- XEKOWRVHYACXOJ-UHFFFAOY-N- XLYOFNOQVPJJNP-UHFFFAOY-N-- QAOWNCQODCNURD-UHFFFAOY-N	Long-RInChIKey=SA-FUHFF- QTBSBXVTEAMEQO-UHFFFAOYSA-N- LFQSCWFLJHTTHZ-UHFFFAOYSA-N-- XEKOWRVHYACXOJ-UHFFFAOYSA-N- XLYOFNOQVPJJNP-UHFFFAOYSA-N-- QAOWNCQODCNURD-UHFFFAOYSA-N
 (No_Structure_0-02.rdf)	Long-RInChIKey=bSA-FEANN----- UTURCBLXOVRHBL-UHFFFAOY-N- KFZMGEQAYNKOFK-UHFFFAOY-N- XLYOFNOQVPJJNP-UHFFFAOY-N	Long-RInChIKey=SA-FUHFF----- UTURCBLXOVRHBL-UHFFFAOYSA-N- KFZMGEQAYNKOFK-UHFFFAOYSA-N- XLYOFNOQVPJJNP-UHFFFAOYSA-N
 (1_reactant_-_no_product.rxn)	Long-RInChIKey=bSA-BEANN--- QHDHNVFIKWGRJR-UHFFFAOY-N	Long-RInChIKey=SA-BUHFF--- QHDHNVFIKWGRJR-UHFFFAOYSA-N
 (No_reactant_-_1_product.rxn)	Long-RInChIKey=bSA-FEANN--- QHDHNVFIKWGRJR-UHFFFAOY-N	Long-RInChIKey=SA-FUHFF--- QHDHNVFIKWGRJR-UHFFFAOYSA-N

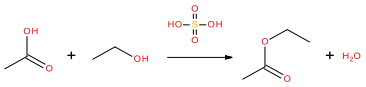
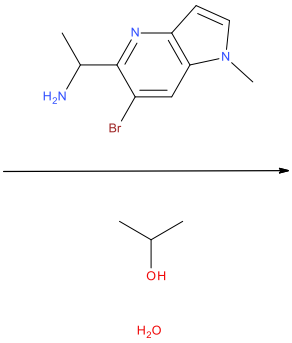
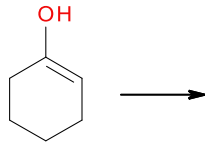
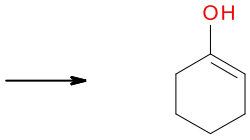
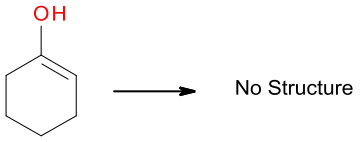
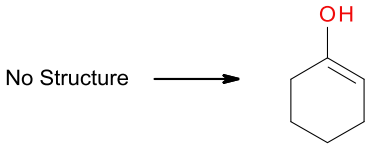
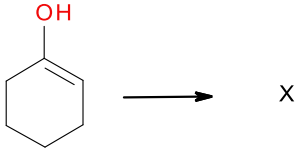
Examples	Version 0.02	Current version
 (1_reactant_-_no_structure.rxn)	Long-RInChIKey=bSA-FEANN-QHDHNVFIKWGRJR-UHFFFAOY-N--X	Long-RInChIKey=SA-BUHFF-MOSFIJXAXDLOML-UHFFFAOYSA-N--QHDHNVFIKWGRJR-UHFFFAOYSA-N
 (no_structure_-_1_product.rxn)	Long-RInChIKey=bSA-BEANN-QHDHNVFIKWGRJR-UHFFFAOY-N--X	Long-RInChIKey=SA-FUHFF-MOSFIJXAXDLOML-UHFFFAOYSA-N--QHDHNVFIKWGRJR-UHFFFAOYSA-N
 (1_reactant_-_X.rxn)	Long-RInChIKey=bSA-FEANN-QHDHNVFIKWGRJR-UHFFFAOY-N--X	Long-RInChIKey=SA-BUHFF-MOSFIJXAXDLOML-UHFFFAOYSA-N--QHDHNVFIKWGRJR-UHFFFAOYSA-N
 (1_reactant_-_A.rxn)	Long-RInChIKey=bSA-FEANN-QHDHNVFIKWGRJR-UHFFFAOY-N--X	Long-RInChIKey=SA-BUHFF-MOSFIJXAXDLOML-UHFFFAOYSA-N--QHDHNVFIKWGRJR-UHFFFAOYSA-N

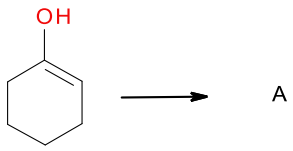
Short-RInChIKeys

Because the first 2 layers of Short-RInChIKeys are identical to the Long-RInChIKeys see section Long-RInChKeys for modifications in this section.

Compared with version 0.02 the hash for the no-structure count is added at the end of the Short-RInChIKey in the format ### for the first, second, and third layer with Z for 0 no-structures, A for 1 no-structure, B for 2 no-structures etc.

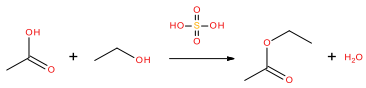
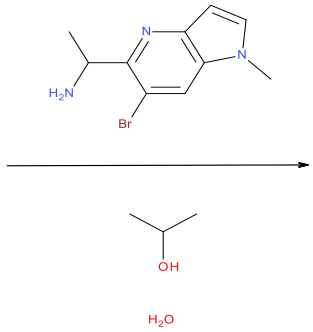
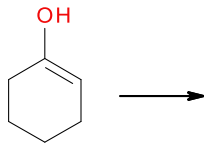
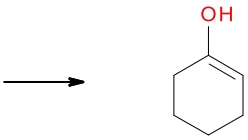
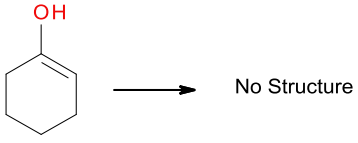
Version	Version 0.02	Current version
Version layer	Long-RInChIKey=aSA-**FEANN (Forward reaction)** Long-RInChIKey=aSA-**BEANN (Backward reaction)** Long-RInChIKey=aSA-**EEANN (Equilibrium reaction)** Long-RInChIKey=aSA-**UEANN (Unspecified reaction)**	Long-RInChIKey=SA-**FUHFF (Forward reaction)** Long-RInChIKey=SA-**BUHFF (Backward reaction)** Long-RInChIKey=SA-**EUHFF (Equilibrium reaction)**
Group separator	-	-
No-structure layer at the end of the string		-### with # = Z for 0 no-structures, A for 1 no-structure, B for 2 no-structures … in group1, 2 or 3 of . Example: -ZZZ reaction without no-structures

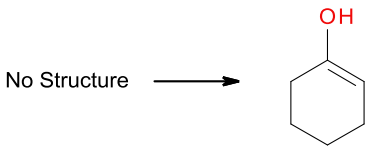
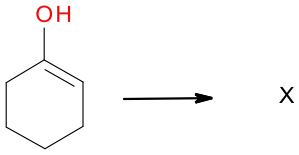
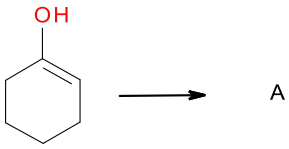
Examples	Version 0.02	Current version
 (Esterification_01.rdf)	Short-RInChIKey=bSA-FEANN-CGNSUPQNQD-CICVPLFIBO-DACVCVAGUF-NEANN-NEANN-NEANN	Short-RInChIKey=SA-FUHFF-JJFIATRHOH-UDXZTNISGZ-QAOWNCQODC-NKDYL-NKDYL-NMOSF-ZZZ
 (No_Structure_0-02.rdf)		Short-RInChIKey=SA-FUHFF-UHFFFADPSC-UHFFFADPSC-MKJSQESANL-NUHFF-NUHFF-NYAOM-ZZZ
 (1_reactant_-_no_product.rxn)	Short-RInChIKey=bSA-BEANN-EANNATPGMB-BNLICLBODK-EANNATPGMB-NEANN-NEANN-NEANN	Short-RInChIKey=SA-BUHFF-UHFFFADPSC-QHDHNVFIKW-UHFFFADPSC-NUHFF-NMOSF-NUHFF-ZZZ
 (No_reactant_-_1_product.rxn)	Short-RInChIKey=bSA-FEANN-EANNATPGMB-BNLICLBODK-EANNATPGMB-NEANN-NEANN-NEANN	Short-RInChIKey=SA-FUHFF-UHFFFADPSC-QHDHNVFIKW-UHFFFADPSC-NUHFF-NUHFF-NUHFF-ZZZ
 (1_reactant_-_no_structure.rxn)	Short-RInChIKey=bSA-FEANN-BNLICLBODK-BIQAGPPMEE-EANNATPGMB-NEANN-NEANN-NEANN	Short-RInChIKey=SA-BUHFF-UHFFFADPSC-QHDHNVFIKW-UHFFFADPSC-NUHFF-NMOSF-NUHFF-AZZ
 (no_structure_-_1_product.rxn)	Short-RInChIKey=bSA-BEANN-BNLICLBODK-BIQAGPPMEE-EANNATPGMB-NEANN-NEANN-NEANN	Short-RInChIKey=SA-FUHFF-UHFFFADPSC-QHDHNVFIKW-UHFFFADPSC-NUHFF-NMOSF-NUHFF-AZZ
 (1_reactant_-_X.rxn)	Short-RInChIKey=bSA-FEANN-BNLICLBODK-BIQAGPPMEE-EANNATPGMB-NEANN-NEANN-NEANN	Short-RInChIKey=SA-BUHFF-UHFFFADPSC-QHDHNVFIKW-UHFFFADPSC-NUHFF-NMOSF-NUHFF-AZZ

Examples	Version 0.02	Current version
 (1_reactant_-_A.rxn)	Short-RInChIKey=bSA-FEANN-BNLICLBODK-BIQAGPPMEE-EANNATPGMB-NEANN-NEANN-NEANN	Short-RInChIKey=SA-BUHFF-UHFFFADPSC-QHDHNVFIKW-UHFFFADPSC-NUHFF-NMOSF-NUHFF-AZZ

Web-RInChIKeys

Web-RInChIKey did not exist in version 0.02.

Examples	Version 0.02	Current version
 (Esterification_01.rdf)		Web-RInChIKey= SMUHAWIQPXIVCEVKG-NJYTGTKWPQNQUSA
 (No_Structure_0-02.rdf)		Web-RInChIKey= MKJSQESANLJZFMCUO-NYAOMGKQNBYDOSA
 (1_reactant_-_no_product.rxn)		Web-RInChIKey= QHDHNVFIKWGRJRNLA-NMOSFIJXAXDLOSA
 (No_reactant_-_1_product.rxn)		Web-RInChIKey= QHDHNVFIKWGRJRNLA-NMOSFIJXAXDLOSA
 (1_reactant_-_no_structure.rxn)		Web-RInChIKey= NVRPNKTYVJSLDPIVQ-NKDYLRUABBWBUSA

Examples	Version 0.02	Current version
 (no_structure_-_1_product.rxn)		Web-RInChIKey= NVRPNKTYVJSLDPIVQ-NKDYLRUABBWBUSA
 (1_reactant_-_X.rxn)		Web-RInChIKey= NVRPNKTYVJSLDPIVQ-NKDYLRUABBWBUSA
 (1_reactant_-_A.rxn)		Web-RInChIKey= NVRPNKTYVJSLDPIVQ-NKDYLRUABBWBUSA

Literature

Biovia, D. S. (2016). *Biovia CT file formats 2017.*

Feng, W., Wataru, U., & Jie, X. (2012). Detection and Measurement of Surface Electron Transfer on Reduced Molybdenum Oxides (MoO[x]) and Catalytic Activities of Au/MoO[x]. *Angewandt Chemie, Int. Ed. Engl., 51*(Int. Ed. Engl.), S. 3883-3887.

Grethe, G., Goodman, J. M., & Allen, C. H. (2013). International chemical identifier for reactions (RInChI). *Journal of Cheminformatics*, 3.59.

Stein, S. E., Heller, S. R., Tchekhovskoi, D. V., & Pletnev, I. V. (2011). *IUPAC International Chemical Identifier (InChI) - InChI version 1, software version 1.04 - Technical Manual.* Gaithersburg, Maryland, U.S: National Institute of Standards and Technology.