# Species Identification

**Document Information**

| | |
|---|---|
| **Editors:** | K. Smith, M. Doronin |
| **Authors:** | K. Smith |
| **Contributors:** | VAMDC WP6 working group |
| **Type of document:** | standards documentation |
| **Status:** | draft |
| **Distribution:** | public |
| **Work package:** | WP6 |
| **Version:** | 11.12 |
| **Date:** | 23/12/2011 |
| **Document code:** | |
| **Document URL:** | http://www.vamdc.org/documents/VAMDC-InChI_v11.12.pdf |

**Abstract:** This document describes the use of InChI and InChIKey for species identification within VAMDC infrastructure

## Version History

| Version | Date | Modified By | Description of Change |
|---|---|---|---|
| V11.12 | 23/12/2011 | K.Smith | first version |
| | | | |

## Disclaimer

The information in this document is subject to change without notice. Company or product names mentioned in this document may be trademarks or registered trademarks of their respective companies.

## License



## All rights reserved



## Acknowledgements

VAMDC is funded under the "Combination of Collaborative Projects and Coordination and Support Actions" Funding Scheme of The Seventh Framework Program. Call topic: INFRA-2008-1.2.2 Scientific Data Infrastructure. Grant Agreement number: 239108.

# CONTENTS

CHAPTER

ONE

# SPECIES IDENTIFICATION: THE IUPAC INTERNATIONAL CHEMICAL IDENTIFIER (INCHI™)

In order to uniquely identify common species across participant VAMDC databases, the Standard IUPAC International Chemical Identifier, and in particular a hash (based on SHA-1) of this identifier (the Standard InChIKey) must be generated for each species (i.e. atom or molecule) within each participant VAMDC node.

This is a brief overview of InChI and InChIKey. For futher informaiton see the documentation at IUPAC.

## 1.1 InChI(Key) Generation for VAMDC - Quick Summary

To ensure compatibility with external databases, and to give VAMDC members the widest choice of tools:

- The structure of all species must be specified in a form that can be converted into InChI(Key). The preferred forms are .mol, .sdf (which can be converted directly by the InChI Trust software), CML or SMILES (which can be converted using a tool such as Openbabel).
- All InChI(Key)s within VAMDC must be Standard InChI(Key)s.
- When generating InChI(Key)s, the most abundant isotopes of elements *must not* be explicitly specified (though see the possible exceptions below).
- When generating InChI(Key)s, *only isotopes that differ from the most abundant must* be explicitly specified (e.g. Carbon-13, Oxygen-18).

## 1.2 Scope of the InChI

InChI is defined as "a series of characters derived by applying a set of rules to a chemical structure to provide a unique digital 'signature' for a compound."

*Included in the scope of InChI:*

- Elements
- Well defined covalently bonded organic molecules
- Organometallic molecules

*Excluded from the scope of InChI:*

- Polymers
- Electronic States
- Conformations

- Nuclear Spin isomers

## 1.3 Structure of an InChI

The InChI has a layered structure and up to 6 layers can be specified:

```
{InChI version}
1. Main Layer (M):
/{formula}
/c{connections}
/h{H_atoms}
2. Charge Layer
/q{charge}
/p{protons}
3. Stereo Layer
/b{stereo:dbond}
/t{stereo:sp3}
/m{stereo:sp3:inverted}
/s{stereo:type (1=abs, 2=rel, 3=rac)}
4. Isotopic Layer (MI):
/i{isotopic:atoms}*
/h{isotopic:exchangeable_H}
/b{isotopic:stereo:dbond}
/t{isotopic:stereo:sp3}
/m{isotopic:stereo:sp3:inverted}
/s{isotopic:stereo:type (1=abs, 2=rel, 3=rac)}
5. Fixed H Layer (F):
/f{fixed_H:formula}*
/h{fixed_H:H_fixed}
/q{fixed_H:charge}
/b{fixed_H:stereo:dbond}
/t{fixed_H:stereo:sp3}
/m{fixed_H:stereo:sp3:inverted}
/s{fixed_H:stereo:type (1=abs, 2=rel, 3=rac)}
(6.) Fixed/Isotopic Combination (FI)
/i{fixed_H:isotopic:atoms}*
/b{fixed_H:isotopic:stereo:dbond}
/t{fixed_H:isotopic:stereo:sp3}
/m{fixed_H:isotopic:stereo:sp3:inverted}
/s{fixed_H:isotopic:stereo:type (1=abs, 2=rel, 3=rac)}
/o{transposition}
```

See the InChI Technical Manual for more details.

## 1.4 Standard vs Non-Standard

**Standard InChI** was defined to ensure interoperability/compatibility between large databases/web searching & information exchange. It is a *subset* of InChI.

Standard InChI distinguishes between chemical substances at the level of “connectivity, “stereochemistry”, and “isotopic composition”, where:

- connectivity means tautomer-invariant valence-bond connectivity; different tautomers have the same connectivity/hydrogen layer
- stereochemistry means configuration of stereogenic atoms and bonds; only absolute stereo or no stereo at all is allowed; unknown stereo designations are treated as undefined;
- isotopic composition means mass number of isotopic atoms (when specified)

Standard InChI prefix: `InChI=1S/.....`

Non-standard InChI prefix: `InChI=1/.....`

The Standard InChI organometallic representation does not include bonds to metal for the time being. This has important implications for some species - e.g. *metal cyanides and isocyanides are currently indistinguishable with Standard InChI*. (Depending on how many molecules this affects, we may need to make some exceptions to the Standard InChI rule.)

## 1.5 Internal InChI Generation Algorithm

The process of generating an InChI takes the following structure normalization steps:

Step 1. Alter the structure drawing

Step 2. Disconnect "salts"

Step 3. Disconnect metals

Step 4. Eliminate radicals if possible

Step 5. Process variable protonation (charges and mobile H)

Step 5.1. Remove protons from charged heteroatoms

Step 5.2. Remove protons from neutral heteroatoms

Step 5.3. Add protons to reduce negative charge

Step 6. Process charges and mobile H

Step 6, procedure 1: Simple tautomerism detection

Step 6, procedure 2. Moveable positive charge detection

Step 6, procedure 3. Additional normalization

See the InChI Technical Manual for more details.

## 1.6 The InChIKey

The InChIKey is a fixed length SHA-256 hash of InChI (27 characters, including two hyphens). Its fixed length makes it easy to index and it is thus designed for databases and web searching.

The InChIKey also serves as a checksum for verifying an InChI, for example, after transmission over a network.

The structure of the InChIKey is illustrated thus:

`AAAAAAAAAAAAAA-BBBBBBBBFV-P`

It consists of:

14 character hash of basic InChI layer - encodes molecular skeleton (should be the same for all isotopologues)

8 character hash of remaining layers (except protonation)

F = S or N (standard or non-standard)

V = A (InChI version 1)

P = (de) protonation indicator = N for neutral, M for -1, O for +1 proton, etc

## 1.7 Standard InChIKey vs InChIKey

Standard:

```
InChI=1S/...........
AAAAAAAAAAAAAA-BBBBBBBBSA-P
```

Non-standard:

```
InChI=1/...........
AAAAAAAAAAAAAA-BBBBBBBBNA-P
```

As with InChI, Standard InChIKeys do not account for tautomerism & indicates only absolute stereo (or completely ignores stereo). Also does not account for original structure's bonds to metal.

## 1.8 How to Generate InChI(Key)s

In all cases, within VAMDC, the **Standard** InChI(Key) must be generated.

The species must be written in a chemoinformatic form which specifies its structure. The core version 1.04 InChI Tools only support the .mol and .sdf formats. CML was supported by InChI version 1.03, but this was withdrawn in version 1.04 (though OpenBabel supports this and many other input formats - e.g. SMILES).

*Use the InChI Trust Software*

http://www.inchi-trust.org/

Input must be in the form of .MOL or .SDFile. Version 1.03 accepts CML format as well.

*Use an online converter:*

InChI Trust Experimental Converter

(experimental converter powered by OASA/BKChem)

QUB Experimental Converter

(experimental converter powered by Openbabel)

*Use conversion tools:*

E.g. Openbabel. Openbabel facilitates conversions from many different formats (e.g. .mol, .sdf, SMILES, CML)

*Use a chemical drawing package:*

E.g. Chemsketch

*Web Based Lookup:*

NIST Webbook

ChemSpider

Cactus

## 1.9 Example Conversion

The example below is for Methane:

SMILES:

```
C
```

or (explicitly specifying hydrogen):

```
[C]([H])([H])([H])[H]
```

CML:

```
<molecule id="CH4-1">
<atomArray>
  <atom id="C1" elementType="C"/>
  <atom id="H1" elementType="H"/>
  <atom id="H2" elementType="H"/>
  <atom id="H3" elementType="H"/>
  <atom id="H4" elementType="H"/>
</atomArray>
<bondArray>
  <bond atomRefs2="C1 H1" id="C1_H1" order="S"/>
  <bond atomRefs2="C1 H2" id="C1_H2" order="S"/>
  <bond atomRefs2="C1 H3" id="C1_H3" order="S"/>
  <bond atomRefs2="C1 H4" id="C1_H4" order="S"/>
</bondArray>
</molecule>
```

Both inputs will result in the following InChI and InChIKey:

```
InChI=1S/CH4/h1H4
VNWKTOKETHGBQD-UHFFFAOYSA-N
```

## 1.10 Standard InChI/InChIKey, Isomers and Isotopologues

Some, but not all, isomerism is supported in Standard InChI(Key).

Structural isomers (same molecular formula, different connectivity) always yield different Standard InChI(Key)s.

*Some* stereoisomers (same molecular formula, different spatial orientation), such as cis- and trans- versions of a species *can* also yield distinct Standard InChI(Key)s. Note, however, that this is not always true. Two examples are cis- and trans-hydroxymethylene and cis- and trans-difluoroethene. The former yields only one distinct InChI(Key). The latter yields two distinct InChI(Key)s.

Different isotopologues (same molecule, same structure, different constituent isotopes) also yield different Standard InChI(Key)s. Note that in the case of isotopologues, ONLY the elements in the species that differ from the most abundant isotopes should have their isotopes explicitly specified. (See also the last section of this document.)

The example below is for C-13 Methane:

SMILES:

```
[13CH4]
```

or (explicitly specifying hydrogen):

```
[13C]([H])([H])([H])[H]
```

CML:

```
<molecule id="CH4-2">
<atomArray>
  <atom id="C1" elementType="C" isotopeNumber="13"/>
  <atom id="H1" elementType="H"/>
  <atom id="H2" elementType="H"/>
  <atom id="H3" elementType="H"/>
  <atom id="H4" elementType="H"/>
</atomArray>
<bondArray>
  <bond atomRefs2="C1 H1" id="C1_H1" order="S"/>
  <bond atomRefs2="C1 H2" id="C1_H2" order="S"/>
  <bond atomRefs2="C1 H3" id="C1_H3" order="S"/>
  <bond atomRefs2="C1 H4" id="C1_H4" order="S"/>
```

```
</bondArray>
</molecule>
```

Both inputs will result in the following InChI and InChIKey:

```
InChI=1S/CH4/h1H4/i1+1
VNWKTOKETHGBQD-OUBTZVSYSA-N
```

Note that the first 14 characters of the InChIKey are identical to the one generated above for C-12 methane.

## 1.11 User Specification of InChIs

In principle, simple InChIs can be hand-produced (e.g. for elements) and the InChI Trust Software API used to generate the InChIKey. However, use of this mechanism to generate InChI(Key)s is unwise. A good illustration of the problem is the generation of an InChI for the Hydrogen Ion (i.e. the proton):

INCORRECT:

```
InChI=1S/H/q+1
ASSFXGJQJOXDAB-UHFFFAOYSA-N
```

CORRECT:

```
InChI=1S/p+1
GPRLSGONYQIRFK-UHFFFAOYSA-N
```

InChI uses a defined algorithm (see earlier) to generate IDs for complex structures. These must not be hand-generated or guessed.

## 1.12 InChI and Average vs Most Abundant Isotope

InChI assumes the average (terrestrial) abundance when the isotope is not specified in the originating format.

This affects the 31 elements in the table below.

Species that contain the most abundant elements should NOT specify the isotope. This ensures compatibility of InChI(Key)s with external databases (e.g. NIST).

If specificity is required in any of the 31 exceptions, the affected element (and only that element) should have its isotope specified when generating the InChI and InChIKey.

**Table of InChI Assumed Isotope Masses when isotope not explicitly specified**

| Element | Symbol | Most Abundant Isotope Mass | InChI Assumed Mass |
|---|---|---|---|
| Nickel | Ni | 58 | 59 |
| Copper | Cu | 63 | 64 |
| Zinc | Zn | 64 | 65 |
| Gallium | Ga | 69 | 70 |
| Germanium | Ge | 74 | 73 |
| Selenium | Se | 80 | 79 |
| Bromine | Br | 79 | 80 |
| Zirconium | Zr | 90 | 91 |
| Molybdenum | Mo | 98 | 96 |
| Ruthenium | Ru | 102 | 101 |
| Silver | Ag | 107 | 108 |
| Cadmium | Cd | 114 | 112 |
| Tin | Sn | 120 | 119 |
| Antimony | Sb | 121 | 122 |

Continued on next page

**Table 1.1 – continued from previous page**

| | | | |
|---|---|---|---|
| Tellurium | Te | 130 | 128 |
| Xenon | Xe | 132 | 131 |
| Barium | Ba | 138 | 137 |
| Neodymium | Nd | 142 | 144 |
| Samarium | Sm | 152 | 150 |
| Europium | Eu | 153 | 152 |
| Gadolinium | Gd | 158 | 157 |
| Dysprosium | Dy | 164 | 163 |
| Erbium | Er | 166 | 167 |
| Ytterbium | Yb | 174 | 173 |
| Hafnium | Hf | 180 | 178 |
| Rhenium | Re | 187 | 186 |
| Osmium | Os | 192 | 190 |
| Iridium | Ir | 193 | 192 |
| Mercury | Hg | 202 | 201 |
| Thallium | Tl | 205 | 204 |
| Lead | Pb | 208 | 207 |

## 1.13 Further Information

The release notes, user's guide, technical manual and API reference can all be found here.