What's New in InChI Software

Version 1.02 – final, with standard InChI/InChIKey

This is the release of the IUPAC standard International Chemical Identifier with InChIKey, version 1, software version 1.02.
(http://www.iupac.org/projects/2000/2000-025-1-800.html
http://www.iupac.org/inchi).

The release conforms to standard InChI and standard InChIKey definitions (see Standard InChI User's Guide) as established by IUPAC InChI Subcommittee at its September 15, 2008 meeting (for details, see http://sourceforge.net/mailarchive/message.php?msg_name=20081002155703.FXZB29597.aamtaout03-winn.ispmail.ntl.com%40ALAN).

Standard InChI

The standard InChI and InChIKey were defined by IUPAC InChI Subcommittee in response to user requests.

With respect to its internal (layered) structure, the standard InChI v.1 introduced in this v. 1.02 release of the standard InChI software is a subset of the IUPAC International Chemical Identifier v.1 introduced in previous InChI software releases (v. 1.01 in 2006 and v. 1.02-beta in 2007).

The standard InChI was defined to reach following goals.

- Standard InChI is for the purposes of interoperability/compatibility between large databases/web searching and information exchange.
- Standard InChI and non-standard InChI are always distinguishable.
- Standard InChI is a stable identifier; however, periodic updates may be necessary; they are reflected in the identifier version designation, which is included in the InChI string.
- Any shortcomings in standard InChI may be addressed using non-standard InChI.

The layered structure of the standard InChI conforms to the following requirements.

- Standard InChI organometallic representation should not include bonds to metal for the time being.
- 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; undefined and unknown stereo designations treated as the same;
    - isotopic composition means mass number of isotopic atoms (when specified)

In the light of the above requirements, the following options are selected for generation of standard InChI

- include tautomerism (i.e., turn mobile H perception on, exclude "fixed hydrogen atoms layer);
- omit reconnection of bonds to metal atoms;
- only a narrow end of a wedge points to a stereocenter;
- include all bug fixes (previous command line options "Fb", "Fb2", "Fnud") without a possibility to turn them off;
- exclude unknown/undefined stereo if no other stereo is present;
- include stereochemisty of phosphines and arsines;
- treat stereochemistry as absolute (not relative or racemic).

The standard InChI is designated by prefix:

"InChI=1S/……….. "
(that is, letter 'S' immediately follows the version number; standard InChI version numbers are always whole numbers).

Standard InChIKey

The InChIKey (introduced in InChI v.1 software v. 1.,02-beta) is a character signature based on a hash code of the InChI string. A hash code is a fixed length condensed digital representation of a variable length character string. Providing a hash derived from an InChI string should be helpful in search applications, including Web searching and chemical structure database indexing; also, this hash may serve as a checksum for verifying InChI, for example, after transmission over a network.

Standard InChIKey, introduced in v. 1.02 release of standard InChI software, is a key computed from (and only from) a standard InChI. It serves the principal purpose of a search-engine-style lookup of chemical information.

Standard InChIKey is a stable identifier; however, periodic updates may be necessary and are reflected in the version designation, which is included in the InChIKey string.

Note that the format of standard InChIKey is different from that in InChI software v. 1.02-beta.

Standard InChIKey has five distinct components.

(1) 14-character hash of the basic (Mobile-H) InChI layer;
(2) 8-character hash of the remaining layers (except for the "/p" segment, which accounts for added or removed protons: it is not hashed at all; the number of protons is encoded at the end of the standard InChIKey.)
(3) 1 flag character,
(4) 1 version character
(5) the last character is [de]protonation indicator.

The overall length of InChIKey is fixed at 27 characters, including separators (dashes):

AAAAAAAAAAAAAA-BBBBBBBBFV-P

This is significantly shorter than a typical InChI string.

Here

(1) **AAAAAAAAAAAAAA** is a 14-character hash.

(2) **BBBBBBBB** is an 8-character hash

(3) **F** is a flag indicating standard InChIKey (produced out of standard InChI): it always has the value 'S'.

(4) **V** is a flag for InChI version character: 'A' for version 1, 'B' for version 2, etc.

(5) **P** is an indicator for the number of protons; this number is not encoded in the hash but is indicated as a separate 2-character block at the end, where one character is a hyphen, as –N for neutral, -M for -1 hydrogen, -O for +1 hydrogen, etc. Namely:

Char	Protons	Char	Protons
N	0		
M	-1	O	+1
L	-2	P	+2
K	-3	Q	+3
J	-4	R	+4

I	-5	S	+5
H	-6	T	+6
G	-7	U	+7
F	-8	V	+8
E	-9	W	+9
D	-10	X	+10
C	-11	Y	+11
B	-12	Z	+12
A	< -12 or > +12		

The checksum character is not present.

Standard InChIKey is supported by a new command line option /Key of the software and by InChI API function **GetStdINCHIKeyFromStdINCHI** (which accepts only standard InChI as input).

Executables

The primary command line executable is stdinchi-1. When invoked without command line options:

```
stdinchi-1 InputFile.sdf OutputFile.txt LogFile.txt ProblemFile.txt
```

it generates standard InChI with auxiliary information (AuxInfo) from the InputFile (in SDF/MOL format).

The software does not accept options leading to generation of a non-standard InChI, only minor adjustments of “structure perception features” are allowed.

The new command line option “InpAux” has been introduced for use in combination with option STDIO if the input stream contains structure(s) in AuxInfo form (without InpAux option, the program presumes that the structure is represented in MOL/SDF format).

Another program, stdwinchi-1.exe, may be run only under MS Windows. It generates standard InChI/InChIKey and displays the chemical structures and details of the generated identifiers.

For more details, refer to Standard InChI User Guide.

Standard InChI Library

The package contains code and pre-built binaries for standard InChI/InChIKey generation library (available under MS Windows and i386 Linux, files libstdinchi.dll and libstdinchi.so.1.02.00.gz). Note that the API (Application Program Interface to libstdinchi) has been changed with respect to previous InChI software versions to account for the new features and avoid confusion with the non-standard InChI entry point names.

In particular, as compared to the InChI software v.1.01, functions have been added to support generation of the standard InChIKey and modularized generation of the standard InChI.

Standard InChI software is not able to generate non-standard InChI or InChIKey.

Conversion of the standard InChI to the non-standard form is not available by design (support for option InChI2InChI has been deliberately removed from the standard InChI software).

More details are provided in Standard InChI API Reference document.

Bugs Fixed

Several bugs have been fixed in v. 1.02 beta release and in this final release.

The purpose of several fixes implemented in v. 1.02 beta software is to withstand malicious attempts to attack a Web server by providing a specially designed InChI string input to InChI binaries.

The standard InChI ver. 1 software version 1.02 contains fixes to bugs discovered by W. D. Ihlenfeldt (4/11/2007), A. Dalke (until 6/30/2007), Anonymous (03/29/2007, req. 1690823), and a few other minor fixes. For descriptions of the bugs see http://sourceforge.net/mailarchive/forum.php?thread_name=028b01c77cac%24c721f8e0%248801a8c0%40xempc3&forum_name=inchi-discuss and http://sourceforge.net/tracker/?atid=741489&group_id=136669&func=browse

The following preprocessor directive activates most of the fixes

#define FIX_DALKE_BUGS 1

included in mode.h header files. A few other bug fixes have been applied unconditionally.

Some of the features implemented in these fixes are:

in reading InChI string (inchi2struct mode):

- restriction on the number of atoms in a component (not more than 1024);
- restriction on the number of components (not more than 1024);
- restriction on max. component charge (not more than +255 and not less than -255)

In converting InChI to structure (inchi2struct mode):

- Fixed false detection of “Stereo centers/allenes: Falsely inverted”

The fix to the bug discovered by W. D. Ihlenfeldt may affect structure to InChI conversion output in case of the normalization described in InChI Technical Manual, p.13, the last row, 3, in "2-3. Non-terminal fragments - 3 atoms" table, as mentioned in http://sourceforge.net/mailarchive/forum.php?thread_name=028b01c77cac%24c721f8e0%248801a8c0%40xempc3&forum_name=inchi-discuss)
This may affect coordination compounds with substructure C=N-C where N has an additional bond to a metal atom (so that the total formal valence of N is greater than 3).The reason for the possible output change is the nature of the bug: in cases when the bug does not cause a crash it might produce results different from the cited description on page 13 of the InChI Technical Manual. The fix should produce results compliant to the InChI Technical Manual.

Changes to the reading InChI string code may lead to a rejection of an input InChI in important cases where the v1.01 would crash on such an input. However, one may create an InChI that could be processed by the current v1.01 and refused by this fixed version. An example may contain, for example, a charge=+256 in the /q layer.

Changes in the reading InChI string code may cause a rejection of certain invalid InChI that would be accepted by v.1.01.

Also, the fix is implemented for minor issues related to halogen oxoanions (e.g., ClO4- with 4 double bonds and "minus" on Cl) and amidinium cations (-C(-NHR)(-NR'R") drawn with "plus" on C rather than on N. The impact of this fix is rather small: as tested with ChemSpider database (courtesy of Antony Williams), it changed 209 InChI out of ~17M.

There are three bug fixes implemented after v. 1.02-beta release. The combined impact of these bug fixes (that is, the number of affected InChI's) is less than 100 for a dataset of ~17M different molecules.

Several other bugs have been fixed in Inchi2Struct software mode that do not affect InChI generation from MOL/SDF/CML structures.