IUPAC International Chemical Identifier (InChI)

InChI version 1, Software version 1.05

User's Guide

Last revision date: January 31, 2017

This document is a part of the release of the IUPAC International Chemical Identifier with InChIKey, version 1, software version 1.05.

CONTENTS

This introductory User Guide is addressed to the novice user of InChI whose primary interest is to learn how to produce InChI/InChIKey identifiers of chemical compounds with InChI executables included in InChI Software distribution (note that alternatively one may simply use nearly any chemical drawing programs as, at the moment of this writing, they typically have a built-in InChI generation ability).

I. OVERVIEW

About InChI

The IUPAC International Chemical Identifier (InChI) provides unique labels for well-defined chemical substances. These labels are generated by converting an input chemical structure, in the form of a 'connection table', to a unique and predictable series of ASCII characters. They offer a means for representing chemical compounds in a manner that does not depend on how they were drawn. Note that they are re-expressions of chemical structures, they are not registry or registration numbers and do not require access to a database. They were developed primarily as a means of 'naming' a compound in digital media although they are expressed as simple text that may be manually interpreted. This document describes the operation and output of the present version of the program that generates this Identifier.

The Identifier is designed to process single, well-defined chemical compounds (which may be composed of multiple components).

InChI is a project of the International Union of Pure and Applied Chemistry (IUPAC) described at: http://www.iupac.org/inchi/

The IUPAC body which takes care of the current and future shape of InChI is the "IUPAC InChI Subcommittee" (IUPAC Division VIII InChI Subcommittee), which reports to IUPAC Division VIII and also to the IUPAC Committee on Publications and Cheminformatics Data Standards. There exist also InChI Subcommittee working groups made up of additional chemists who are developing rules for extending the capabilities of InChI. See: http://iupac.org/web/ins/802

Historically, the primary development of the InChI algorithm and software took place at NIST (US National Institute of Standards and Technology, USA) under the auspices of IUPAC. Since 2009, the responsibility for InChI technical development and promotion has been in the hands of the InChI Trust (http://www.inchi-trust.org/) – a not-for-profit organization which works in close contact with IUPAC (and of which IUPAC is a member).

Technical details are given in a separate document, the InChI Technical Manual. The basic algorithms were taken from the literature, with selection, testing and implementation done primarily at NIST; with modifications and additions by IUPAC and the InChI Trust.

In the several years of its development, many individuals contributed to the development of the InChI at meetings and through correspondence. The chemical rules employed are intended to represent a consensus view of the concept of chemical identity. The computer program described in this document applies these algorithms to input structures and generates both the Identifier and an annotated depiction of the structure.

Derivation of the InChI from an input chemical structure proceeds through three steps: 1) normalization – all input information not needed for structure identification is discarded and structure information is divided into 'layers'; 2) canonicalization – each atom is given a label that depends only on its position in the structure; 3) serialization – a string of characters, the Identifier, is generated from the canonical labels. All 'chemical' rules are applied in the first step.

The current version of the Identifier is 1; the current version of the InChI software is 1.05 (Winter 2017) release. Previously released versions 1.01 (2006), 1.02-beta (2007), 1.02-standard (2009), 1.03 (June 2010) and 1.04 (September 2011) as well as all earlier versions, are now considered obsolete.

Standard and non-standard InChI

InChI has a layered structure which allows one to represent molecular structure with a desired level of detail. Accordingly, the InChI Software may generate different InChI

strings for the same molecule, depending on the choice of a multitude of options (e.g., distinguishing or not distinguishing tautomers). This flexibility, however, may be considered a drawback with respect to standardization/interoperability. The standard InChI which is always produced with fixed options was defined by the IUPAC InChI Subcommittee in response to these concerns.

The standard InChI was defined to ensure interoperability/compatibility between large databases/web searching and information exchange. As related to its internal layered structure, standard InChI, introduced in v.1.02-standard (2009) release of InChI Software, is a subset of IUPAC International Chemical Identifier v.1. The layered structure of the standard InChI conforms to the following requirements.

- Standard InChI organometallic representation does 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; unknown stereo designations are treated as undefined;
    - isotopic composition means mass numbers of isotopic atoms (when specified)

Standard InChI v.1 was introduced in v. 1.02-standard release of the InChI Software in 2009 (this software version was able of generating only standard InChIs).

The present release of InChI Software, v. 1.05, has merged functionality. It allows one to produce both standard and non-standard InChI strings, as well as their hashed representation (InChIKey).

By default, InChI Software v. 1.05 produces standard InChI (for brevity, stdInChI below). In particular, the standard identifier is generated when the software is used without any

specifically added options. If some options are specified, and at least one of them qualifies as related to non-standard InChI (see section 'InChI Software Options' below), the program produces non-stdInChI/InChIKey.

The standard InChI is designated by the prefix: "InChI=1S/…….. " (that is, letter 'S' immediately follows the Identifier version number, '1'; Identifier version numbers should always be whole numbers).

Non-standard InChI is designated by the prefix: "InChI=1/…….. " (that is, letter 'S' is omitted).

InChI's obtained with the experimental features of v. 1.05 Software (support of polymers; support of "large" molecules) are designated by the prefix: "InChI=1B/…….. " ('B' for beta).

About InChIKey

The InChIKey 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 I`nChI 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.

The InChIKey consists of two blocks. The first block is always the same for the same molecular skeleton. All isotopic substitutions, changes in stereoconfiguration, tautomerism and protonation are reflected in the second block.

A standard InChIKey, which is a key produced from a standard InChI, does not account for tautomerism and may indicate only absolute stereo (or completely ignore stereo). It also does not account for the original structure's bonds to metal.

The two hash blocks of InChIKey are based on a truncated SHA-256 cryptographic hash function (http://en.wikipedia.org/wiki/SHA_hash_functions#SHA-2).

Note that due to the very essence of hash functions, collisions (the same InChIKey for different InChIs/structures) are unavoidable in very large collections. A theoretical – optimistic – estimate of collision resistance (i.e., the minimal size of a database at which a single collision is expected, that is, an event of the two hashes of two different InChI strings being the same) is 6.1×10^{9} molecular skeletons × 3.7×10^{5} stereo/ isotopomers per skeleton ≈ 2.2×10^{15}. To exemplify: the probability of a single first block collision in a database of 1 billion compounds is 1.3%. In other words, a single first block collision is expected in 1 out of 100/1.3 = 75 databases of 10^{9} compounds each. For 10^{8} (100 million) compounds in a database this probability is 0.014%.

A beta-version of the InChIKey was introduced in software v. 1.02-beta (2007). The standard InChIKey was introduced in v. 1.02-standard release (2009) as an InChIKey computed from the standard InChI and intended for the principal purpose of a search-engine-style lookup of chemical information.

The present release of the InChI Software, v. 1.05 (Fall 2016), has merged functionality. It allows one to produce both standard and non-standard InChIKey.

Note that the current format of InChIKey is different from that of the beta version (2007); the format of the standard InChIKey is the same as that of v. 1.02-standard (2009).

II. ABOUT InChI PROGRAMS

This document is accompanied by version 1.05 of the InChI generator executable. This program runs under 32/64 bit Microsoft Windows (`inchi-1.exe`) and Linux (`inchi-1`) operating systems. Also included is `winchi-1.exe`, which is a conventional Windows graphical-interface application.

The program `winchi-1` takes an input structure and generates both graphical and text output in a form designed to allow critical examination of the InChI. The Identifier and associated text output may be parsed and annotated.

As structure input, the program currently accepts standard SDfiles, Molfiles [see "Description of several chemical structure file formats used by computer programs developed at Molecular Design Limited" by Arthur Dalby, James G. Nourse, W. Douglas Hounshell, Ann K. I. Gushurst, David L. Grier, Burton A. Leland, and John Laufer, Journal of Chemical Information and Computer Sciences, 1992; 32(3); pp. 244-255]; a more recent description of V2000 and the latest V3000 formats may be downloaded from [http://accelrys.com/products/collaborative-science/biovia-draw/ctfile-no-fee.html], or its own output produced when the "Full auxiliary information" option is selected. Input may originate from individual disk files or through the Windows clipboard. From v. 1.05, a limited support of V3000 Molfiles is included.

InChI may be also generated by using Software Library/application programming interface (API). This is described later.

III. RUNNING InChI PROGRAMS

Graphical Interface Program (winchi-1)

Introduction

The InChI generation program is provided along with sample chemical structures in a 'zip' file INCHI-1-BIN.zip. To use this program, first extract the contents of the file to a directory of your choice. To start the program, run the file `winchi-1.exe` that was extracted from the zip file. Figure 1 then appears on your monitor.

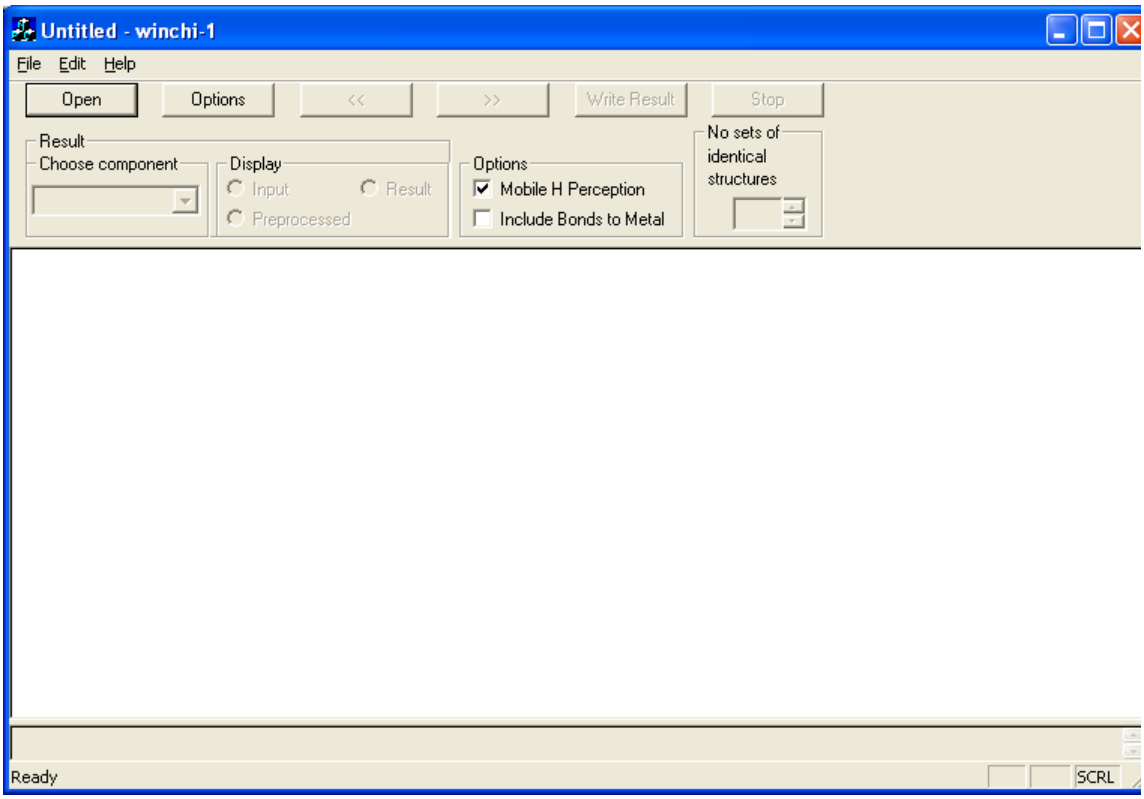

Figure 1

Generating an InChI begins with the selection of an input structure file. The simplest way is to drag the input structure file from Windows Explorer directory list into the InChI window. Structures also may be copied from certain chemical structure editors (ISIS/Draw with "Copy Mol/Rxnfile to the Clipboard" option or from ACD/ChemSketch) and pasted into the InChI window (Select Edit → Paste from InChI menu). The input structure file pathname may be provided as a command line option when you start `winchi-1`. Selection of the input structure file may also be done by first clicking on the 'Open' button (top left corner of Figure 1) and then, in the dialog box that appears (as shown in Figure 2),

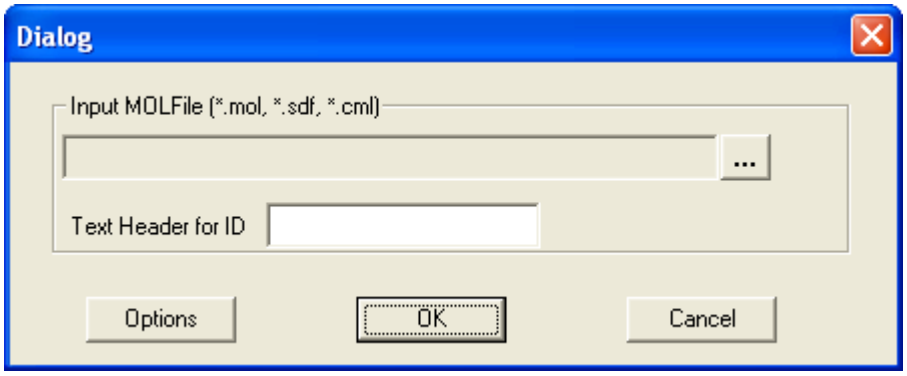

Figure 2

selecting a structure file using the '…' button on the right of the 'Input Structure File' field. You may select any of the sample .mol or .sdf files for initial testing. In this dialog you may also enter "Text Header for ID"; this will simply add to the InChI header a structure ID if it is present in an input SDfile (from other input formats the header and ID are extracted automatically). Ignore this box for now.

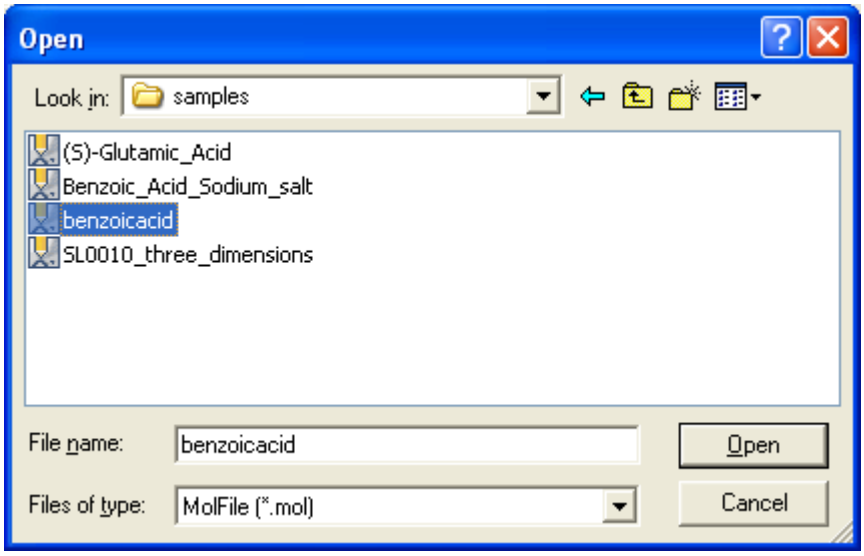

Figure 3

Figure 3 shows the selection of a structure file. In this case it is entitled benzoicacid.mol, which was prepared by a separate structure-drawing program. Clicking the file name copies it into "File name:" line. After that click "Open" to close the dialog.

At this point you may also change InChI processing options. (The choices for the options that can be changed are shown in Figure 4, but no changes are made in this example.)

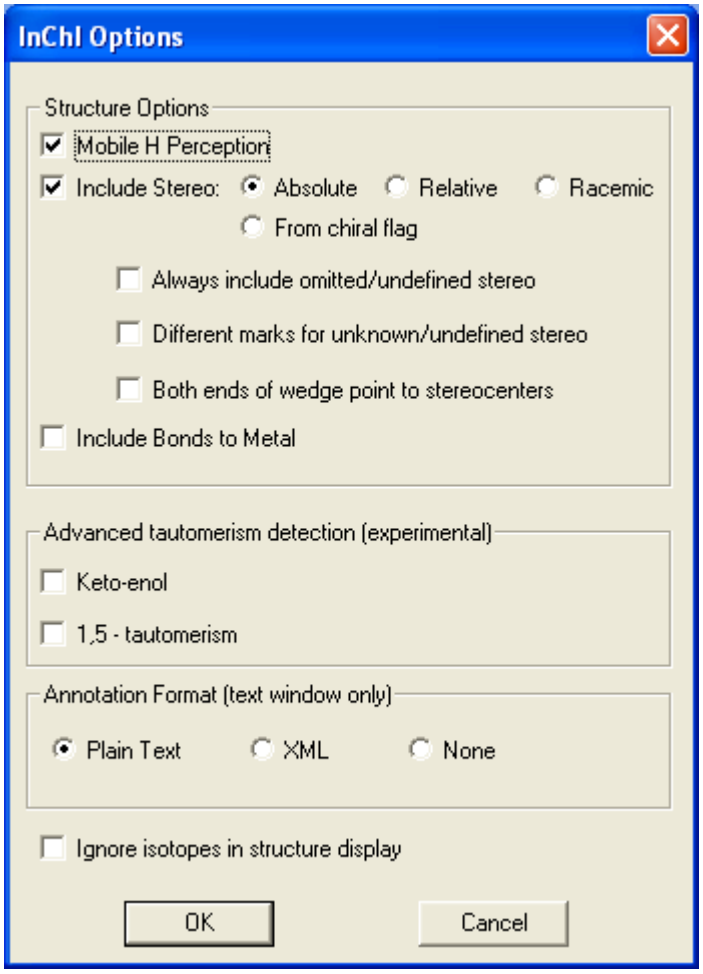

Figure 4

Close InChI Options dialog if you opened it and select OK in the dialog (Fig. 2) when done; the result is Figure 5.

The main output window is composed of two sections: the upper section (shown in white in Figure 5) shows structural information graphically and the lower section (shown in gray in Figure 5) shows text output.

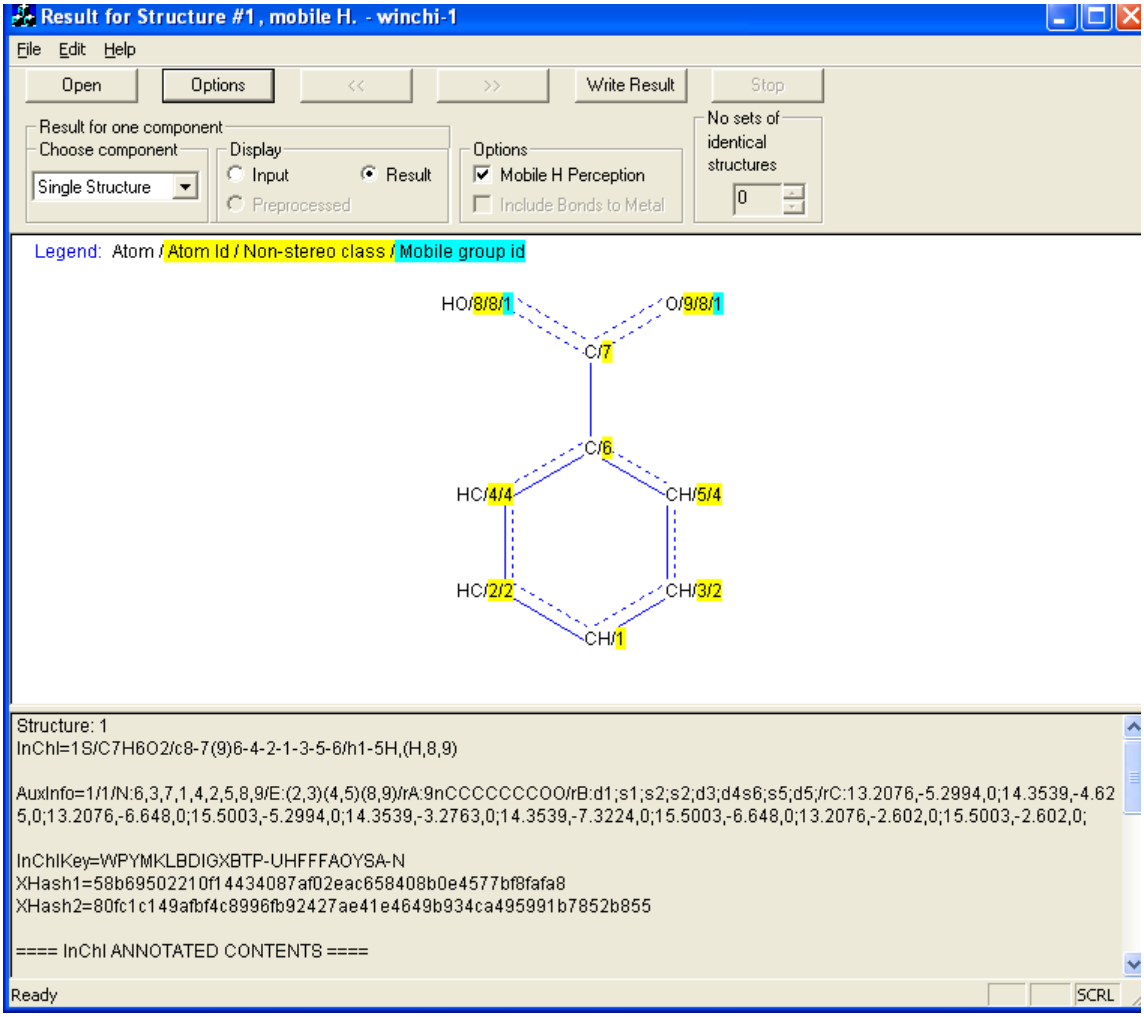

Figure 5

Upper section

The structure is displayed along with labels generated by InChI algorithms. In cases where an SDF file is input, the first structure shown is the first entry in the input file. The example shown in Figure 5 is a single component example. If more than one component (independent structure) is found in the first structure file (such as benzoic acid, sodium salt shown in Figure 6), each may be separately examined using the "Choose component" 'combo box' on the upper left of the screen, although they are treated as part of a single compound by InChI (Figures 7 and 8).

The buttons under “Display” permit viewing of the input structure and the preprocessed structure if it differs from the input structure. . The buttons under “Options” are the same as in the “Options” dialog box. “Mobile H Perception” removes the “fixed-H” part of the identifier. Figure 9 shows the same structure with the option “Mobile H Perception” off.

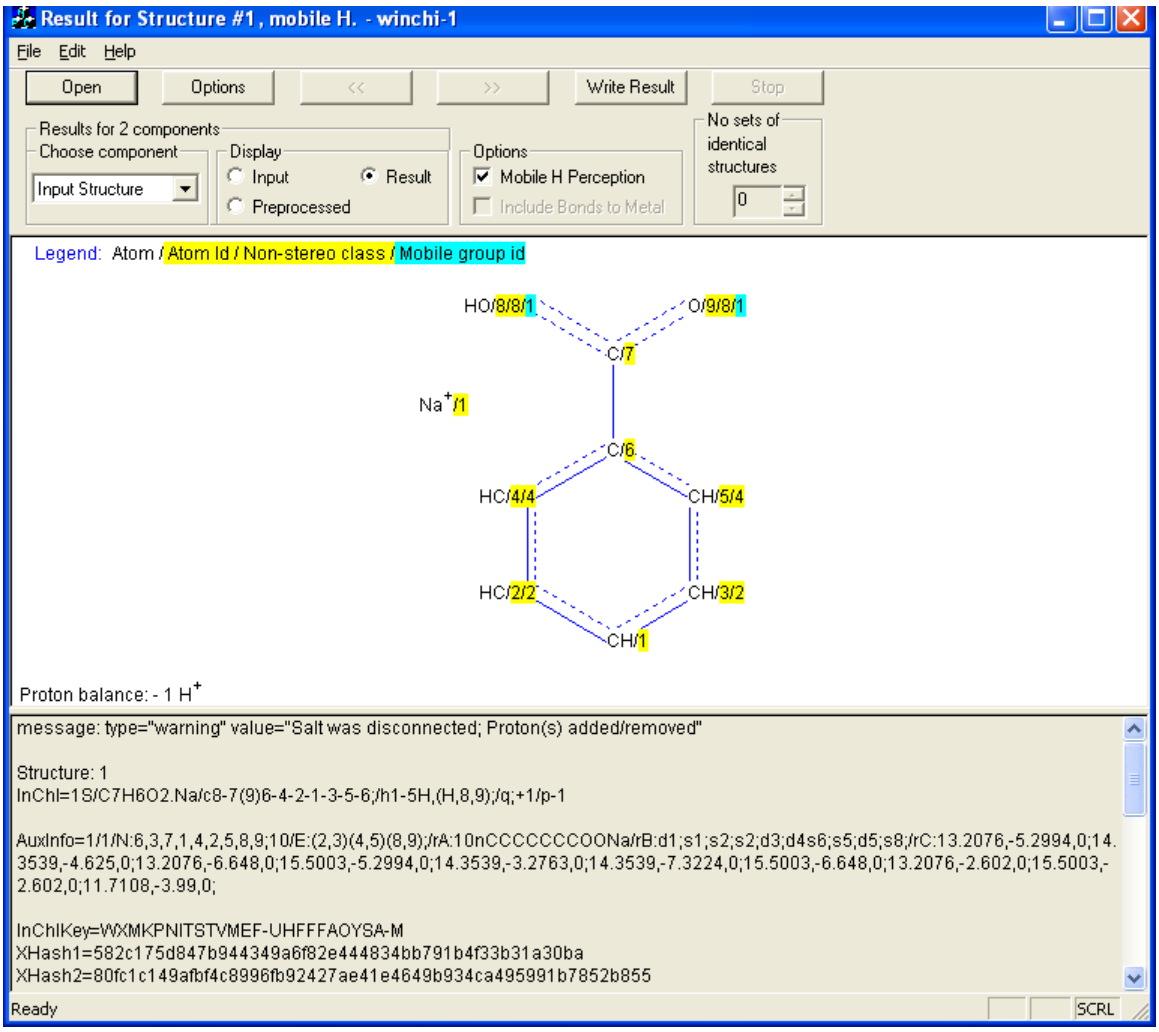

Figure 6

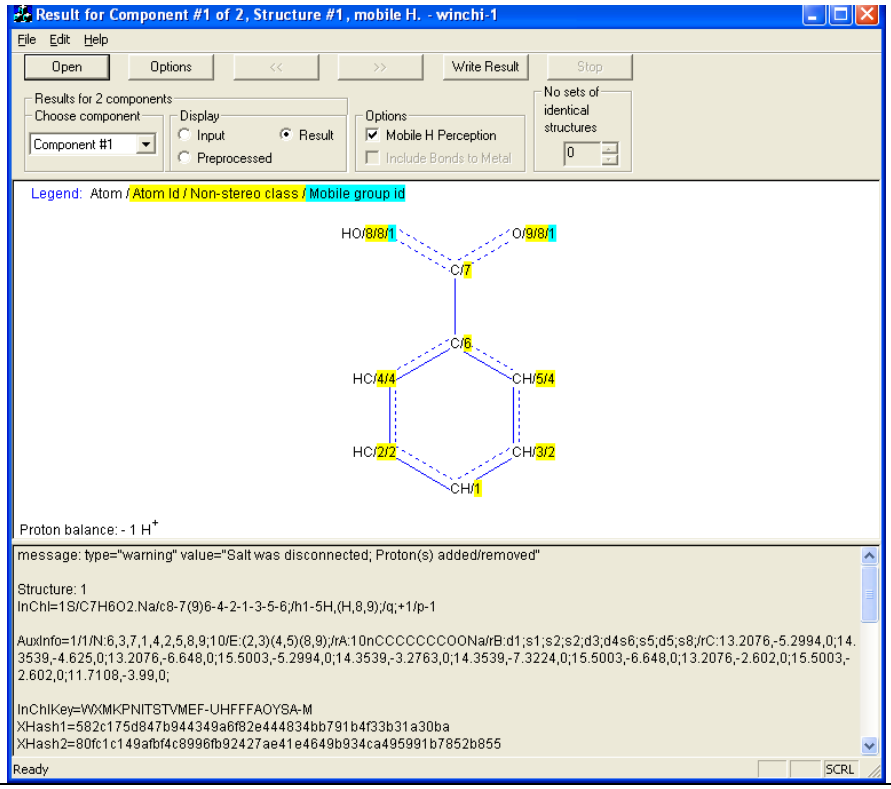

Figure 7

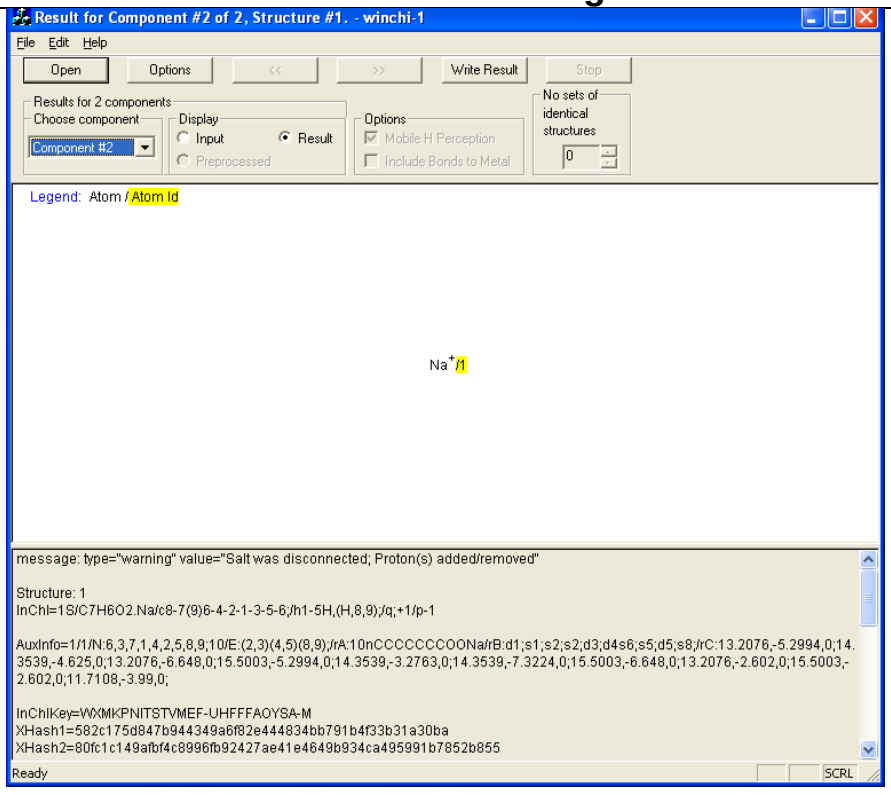

Figure 8

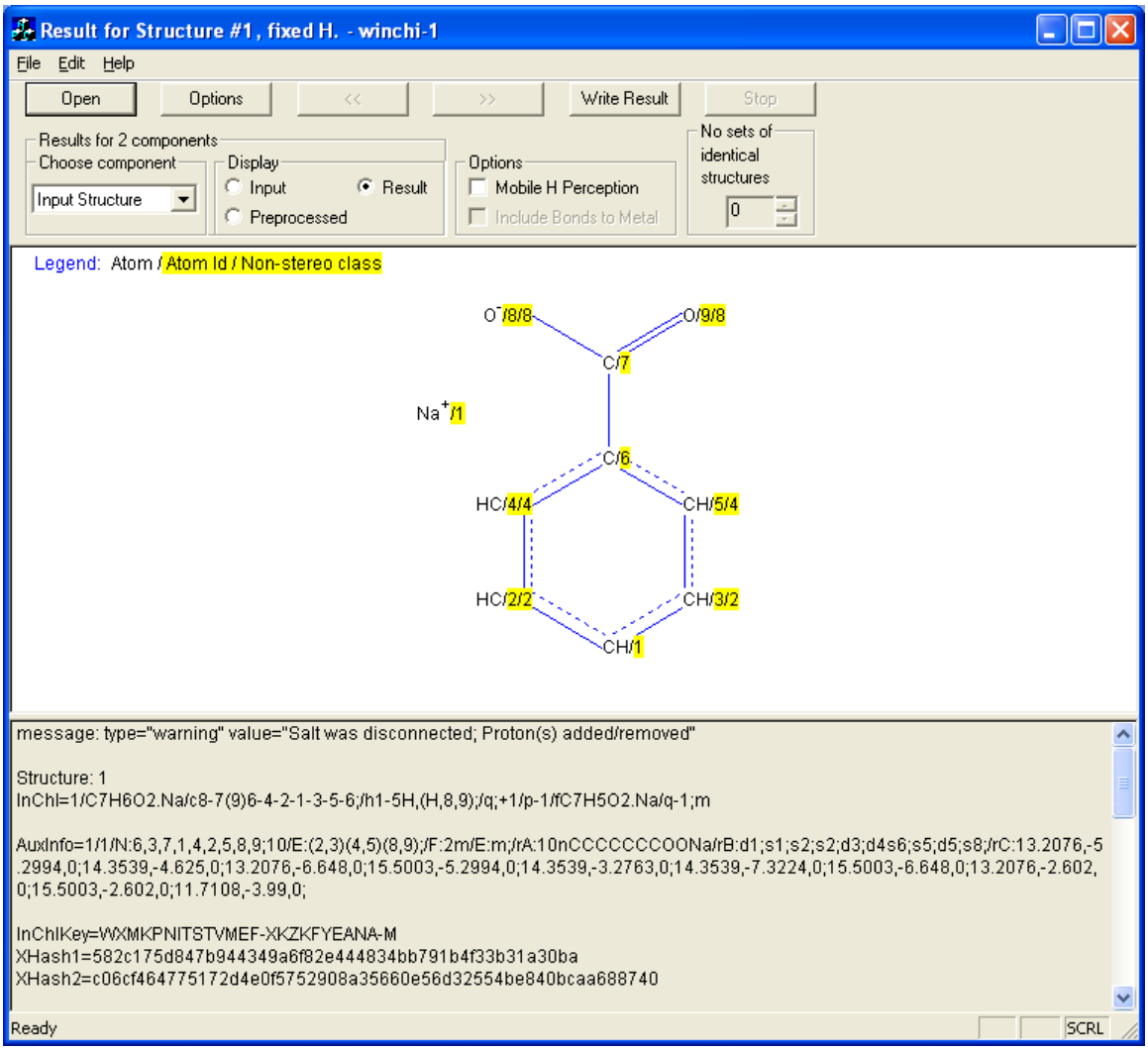

Figure 9.

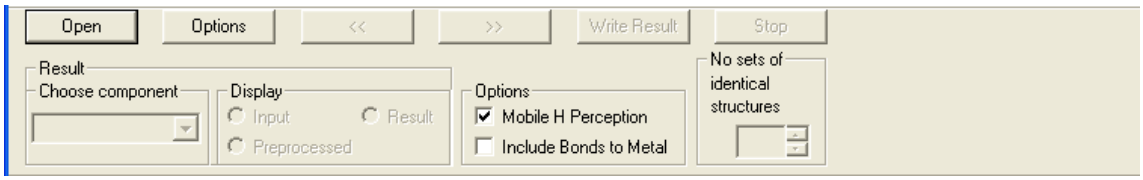

Figure 10. InChI Toolbar

On the InChI Toolbar the rightmost box displays the number of sets of equivalent components. When equivalent components are found, they may be highlighted by making a selection in the box. This provides a quick way to determine if two depictions of the same compound are considered to be the same by InChI algorithms, although the actual InChI generated will represent the collection of structures as a single compound.

The structure display shows the canonical identification number of each atom along with the non-stereo equivalence class number assigned to that atom. The canonical number is the unique number given to an atom and used for 'serialization' (creation of the actual InChI). The non-stereo equivalence class number is a number assigned to each set of equivalent atoms (all atoms having the same equivalence class number are indistinguishable, *ignoring stereochemistry*; the equivalence class number is the smallest canonical identification number in the class of equivalent atoms). This information is only intended to assist in the understanding of results of InChI processing and is not directly used in InChI generation except in the processing of stereochemistry.

Stereochemical parities of bonds and atoms are also displayed. A question mark symbol indicates that stereoisomerism is possible, but the configuration has not been specified. Bonds that have been found to be variable by alternation or movement of mobile H-atoms or charges are shown by dotted lines. This information is used only for deciding which bonds may exhibit double bond (*Z/E*) isomerism. By design, the Identifier does not explicitly represent bond types.

Lower Section

The InChI along with auxiliary data and explanatory information is shown in the lower section of the output window, such as seen in Figure 6 (see Section VI). Unlike the graphical display, even if more than one disconnected component is found, all textual results for a single input structure file are shown together. This reflects the important point that all components of a submitted structure are considered by InChI to be part of a single compound. Results for different (disconnected) components of a single substance are

separated by semicolons, except for chemical formulas, which, in keeping with common conventions, are separated by dots.

Options

Pressing the Options Button opens the InChI Options Dialog Box. The following options are then available (as seen in Figure 4):

- Mobile H Perception – turning Off will fix all H-atoms (disallow H-migration), this allows the generation of a fixed-H section of the Identifier (and makes the resulting InChI non-standard).
- Include Stereo (Absolute, Relative, Racemic, From chiral flag) – include stereo layer and choose its type or exclude all stereo information from the identifier. If the last option is selected then in presence of a chiral flag stereochemistry is considered absolute, otherwise relative.
  For standard InChI the only allowed choice is absolute stereochemistry or omission of all stereo; other choices make InChI non-standard.
- Always include omitted/undefined stereo – by default, InChI does not include unknown/undefined stereo unless at least one defined stereo is present in the input structure. Turning this option On results in inclusion of unknown/undefined stereo in all cases.
- Different marks for unknown/undefined stereo – turning this option On will result in usage of the two different signs, ‘u’ and ‘?’, for “unknown” and “undefined” stereo. Briefly: “undefined” means not given while “unknown” means explicitly marked as unknown, e.g., with “wavy” bonds. By default, this option is turned off and the twoh signs are merged to ‘?’ (that is, “unknown” stereo treated as “undefined”).
- Both ends of wedge point to stereocenters – by default, this option is turned Off. This means that that a stereo bond depicted by a wedge affects the stereochemistry of only the atom ‘pointed to’ by the narrow end of that wedge. However, it may be turned On if the user is completely sure that a stereobond affects both atoms it

connects (that is, for 2D structures complying to the legacy "perspective" stereochemistry drawing style).

- Include Bonds to Metal - turning On will add a layer that includes specific bonding to metals (in case of salts the bonds between a metal and an acid cannot be reconnected – as seen in Figures 6-9 where that choice is "grayed out" and cannot be ticked or checked).
- Annotation Format (Plain Text; XML, None) – choose appropriate format for explanatory information.
- Ignore Isotopes in Structure Display – this does not change the identifier, it only affects the structure appearance and the display of sets of equivalent components.

Note that the above options form a subset of a full options set available in the command-line executable `inchi-1` (see section 'InChI Software Options' below).

Text File Output

At any time you may select 'Write Result' to analyze the input file and write all textual results to an output file located in the same directory as the program. The name of this file is derived from the name of the input structure file and is displayed when it is created (the name has extension .txt). Figure 11 is an example of this for benzoic acid. It shows the directory/location on the computer as well as the file names given to the three (3) output files. Two other files to assist in diagnosing problems, should they occur, are created and their names displayed. One of them is a log file; it contains names of input and output files, a list of selected options, warning and error messages, number of processed structures, processing time, etc. The name of this file has extension .log. Another file – a problem file -- contains input structure file records that caused errors. This file (its name has extension .prb) may be important to determine reasons for the errors. A listing of errors and warnings is given in the Appendix 1.

Figures 12-14 show the content of the three output files. The .prb file is, of course, empty, since there were no problems encountered in generating the InChI for benzoic acid.

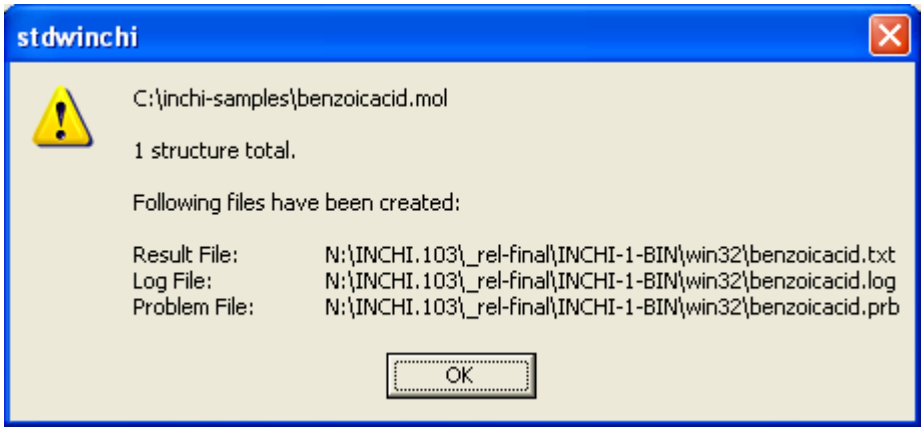

Figure 11

```
* Input_File: "C:\inchi-samples\benzoicacid.mol"
Structure: 1
InChI=1S/C7H6O2/c8-7(9)6-4-2-1-3-5-6/h1-5H,(H,8,9)
AuxInfo=1/1/N:6,3,7,1,4,2,5,8,9/E:(2,3)(4,5)(8,9)/rA:9nCCCCCCCOO/rB:d1;s1;s2;s2;d3;d4s6;s
5;d5;/rC:13.2076,-5.2994,0;14.3539,-4.625,0;13.2076,-6.648,0;15.5003,-5.2994,0;14.3539,-
3.2763,0;14.3539,-7.3224,0;15.5003,-6.648,0;13.2076,-2.602,0;15.5003,-2.602,0;

InChIKey=WPYMKLBDIGXBTP-UHFFFAOYSA-N
XHash1=58b69502210f14434087af02eac658408b0e4577bf8fafa8
XHash2=80fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855
```

Figure 12. File benzoicacid.txt

```
InChI version 1, Software version 1.04 Build of September 9, 2011
Opened log file 'benzoicacid.log'
Opened input file 'benzoicacid.mol'
Opened output file 'benzoicacid.txt'
Opened problem file 'benzoicacid.prb'
The command line used:
"C:\inchi-samples\inchi-1.exe benzoicacid.mol benzoicacid.txt benzoicacid.log benzoicacid.prb"
Generating standard InChI
Input format: MOLfile
Output format: Plain text
Full Aux. info
Timeout per structure: 60.000 sec
Up to 1024 atoms per structure

End of file detected after structure #1.
Finished processing 1 structure: 0 errors, processing time 0:00:00.00
```

Figure 13. File benzoicacid.log

Figure 14. File benzoicacid.prb

Command Line Executable inchi-1

An executable program (`inchi-1.exe under Windows, inchi-1 under Linux)`. uses 'command line' arguments that are shown by invoking the program without any arguments. The principal use of the program is to allow batch processing processing of multiple structure files, primarily SDF files. The Linux version does not display chemical structures.

Standard redirection may be used to suppress `inchi-1` console output.

Under Windows:

```
inchi-1 /AuxNone input.sdf output.txt logfile.log NUL 2>NUL
```

Under Linux:

```
inchi-1 -AuxNone input.sdf output.txt logfile.log NUL
2>/dev/null
```

">" or "1>" redirects standard output, "2>" redirects standard error output.

To process files greater than 2 GB with `inchi-1`, the output of a problem file should be suppressed. To do that, the output and log file names should be included in the command line; the name of the problem file should be NUL, for example:

```
inchi-1 input.sdf  output.txt logfile.log NUL
```

Note that the graphical program `winchi-1.exe` cannot process files greater than 2 GB.

AMI (Allow Multiple Inputs) mode

Since InChI Software v. 1.04, the possibility of processing multiple input files at a single run was added to the inchi-1 executable (both Windows and Linux versions).

This mode is activated by the inchi-1 command line option "/AMI" (Windows) or "–AMI" (Linux; AMI stands for "Allow Multiple Inputs"). In this mode, all the file names supplied in the command line are considered as the names of separate input files.

For further convenience, the common file name wildcards ("*" and "?") are supported.

For example, issuing a command

```
inchi-1 *.mol /AMI (Windows)
```

```
inchi-1 *.mol -AMI (Linux)
```

will instruct the executable to process all the mol-files in the current directory.

Note, that omitting the switch "AMI" assumes working in a conventional single-input mode which may result in undesired treatment of wildcards[1].

In AMI mode, the names of output, log and problem files could not be individually specified. Instead, they are formed, for each of multiple inputs, by appending the file name with suffixes ".txt", ".log" and ".prb". However, to partially mimic the behavior of inchi-1 in conventional single-input mode, three additional command line options are introduced (see section "Availability of InChI Software options", Table 6). They allow one to redirect the output to stdout, log to stderr, as well as to suppress creation of problem files.

Examples *(Windows, Linux)*:

```
inchi-1 nci*.mol /AMI /AMIOutStd /AMIPrbNone /AuxNone /Key
```

```
./inchi-1 /home/me/mol/nci/*.mol -AMI -AMILogStd -AMIPrbNone
-RecMet –FixedH
```

As indicated by tests, processing of multiple MOL files in AMI mode may be several times faster (the exact speed-up depends on many details; anyway the processing time is still significantly longer than that for a single SDF file containing the same data).

[1] There is an important difference in wildcard expansion under Windows and Linux.
Under Windows, inchi-1 executable makes an expansion itself (if "AMI" switch is specified). That is, if "AMI" is omitted, no expansion occurs and "*.mol" is just considered as an invalid file name.
Under Linux, wildcards are always expanded by shell. That is, if "AMI" is omitted, "*.mol" will be expanded to the list of file names; the first four of them will be treated by inchi-1, according to single-input rules, as input, output, log and problem file names (which means that the last three files will be over-written).

New enhancements of v. 1.05

InChI Software v. 1.05 introduced -experimental support of InChI/InChIKey for regular single-strand polymers and experimental support of large molecules containing up to 32767 atoms was added.

By default, the executable inchi-1 ignores polymer-specific data (which also ensures compatibility with the behaviour of previous versions); to allow treatment of polymers, one should explicitly use the new command line option `Polymers` (`-Polymers` under Linux or `/Polymers` under Windows). Analogously, switch `LargeMolecules` is necessary to enable processing molecules having more than 1024 atoms.

InChI Software Library (libinchi)

For advanced users who may want to create the Identifier in their own software the InChI Software Library (InChI API) is provided in a separate package. The package contains 'C' source code for `inchi-1.exe`, 'C' source code for the InChI Library that may be compiled into a Dynamic Link Library (DLL) `libinchi.dll` under Windows or Shared Object (SO) `libinchi.so` under Linux; also, there are 'C' and Python examples of simple applications that read input Molfile and use the InChI Library to produce Identifiers.

The InChI Library does not display structures and is not able to read chemical structural data from the input file. It uses specially formatted input binary data and produces three strings: InChI, the Auxiliary Information, and, if necessary, an error or warning message. Also, there are procedure to calculate InChIKey and other service routines. The source code is accompanied with makefiles tested with gcc under Windows and Linux.

The InChI Library allows one to generate both standard and non-standard InChIs/InChIKeys. For example, an API function `GetINCHI()` produces standard InChI by default and non-standard InChI if some "InChI creation option" is specified in input

parameters. However, for compatibility with the previous v. 1.02-standard (2009) release, the procedures which deal only with stdInChI – for example, `GetStdINCHI()` - are retained.

The InChI API calls are documented in the separate "InChI API Reference" document and source code header file "inchi_api.h" included in the package.

InChI Software Options

The exact set of InChI Software options has been changing from release to release. The description below refers to the current, v. 1.05 (Fall 2016) release.

The options are available in graphical program `winchi-1`, command line executable `inchi-1` and through InChI API. Not all the options are available for all the parts of software; the maximal set of options is available for the `inchi-1` program.

Structure perception and InChI creation options

Options affecting generation of InChI are divided on "structure perception" options and "InChI creation" options.

The perception options are considered drawing style/edit flags which affect the input structure interpretation and are not memorized. It is assumed that the user may deliberately use these options to account for the specific features of structure collections. Whence, perception options may be used while generating standard InChI without loss of its "standardness".

Perception options are listed in the following table. Presented here are command line switches available (they should be used with the appropriate prefix - i.e., 'NEWPSOFF' should be entered as '/NEWPSOFF' under Windows and '-NEWPSOFF' under Linux).

Table 1. Structure perception options.

Structure perception option	Meaning	Default behavior (standard; if no option supplied)

NEWPSOFF	Both ends of a wedge (which indicates stereochemistry) point to stereocenters	Only the narrow end of a wedge points to a stereocenter
DoNotAddH	All hydrogens in input structure are explicit	Add H according to usual valences
SNon	Ignore stereo	Use absolute stereo

There are several options (Table 2) which modify the interpretation of input stereochemical data. In principle, they also may be considered "structure perception" options. However, as the standard InChI, by definition, requires the use of absolute stereo (or no stereo at all), these "perception" options assume generation of non-standard InChI.

Table 2. Stereo interpretation options (lead to generation of non-standard InChI).

Stereo option	Meaning	Default behavior (standard; if no option supplied)
SRel	Use relative stereo	Use absolute stereo
SRac	Use racemic stereo	Use absolute stereo
SUCF	Use Chiral Flag in MOL/SD file record: if On – use absolute stereo, Off – relative	Use absolute stereo (or another option if requested by SRel /SRac/SNon switches)

The creation options affects the InChI algorithm, not structure perception. They modify the defaults which are specified for standard InChI and significantly affect the final appearance (e.g., additional InChI layers may appear). Whence, using any of the creation options qualifies the resulting identifier as non-standard.

Creation options used for generation of a particular non-standard InChI may be appended to the created identifier, see below.

InChI creation options are listed in the following table.

Table 3. InChI creation options.

InChI creation option	Meaning	Default behavior (if no option supplied)
SUU	Always indicate unknown/undefined stereo	Does not indicate unknown/undefined stereo

		unless at least one defined stereocenter is present
SLUUD	Stereo labels for "unknown" and "undefined" are different, 'u' and '?', resp. (new option; see explanation)	Stereo labels for "unknown" and "undefined" are the same ('?')
RecMet	Include reconnected metals results	Do not include
FixedH	Include Fixed H layer	Do not include
KET	Account for keto-enol tautomerism (experimental extension to InChI v. 1)	Ignore keto-enol tautomerism
15T	Account for 1,5-tautomerism (experimental extension to InChI v. 1)	Ignore 1,5-tautomerism
LargeMolecules	Accept molecules containing more than 1024 (but less than 32767) atoms	Reject molecules containing more than 1024 atoms
Polymers	Accept polymer data in input V2000 Molfiles.	Ignore polymer data in input V2000 Molfiles.
OutErrInChI	Output empty InChI and corresponding InChIKey if error occurs[2]	Output nothing

The standard InChI is always generated if no InChI creation/stereo modification options are specified. This means:

- include tautomerism (i.e., turn mobile H perception on, exclude "fixed hydrogen atoms" layer) except for keto-enol and 1,5-tautomerism;
- omit reconnection of bonds to metal atoms;
- only the narrow end of a wedge points to a stereocenter;
- exclude unknown/undefined stereo if no other stereo is present;
- treat stereochemistry as absolute (not relative or racemic).

[2] The Standard InChI and InChIKey for empty entity are:
InChI=1S//
InChIKey=MOSFIJXAXDLOML-UHFFFAOYSA-N
The Non-standard InChI and InChIKey are:
InChI=1//
InChIKey=MOSFIJXAXDLOML-UHFFFAOYNA-N

Inversely, if any of SUU/SLUUD/RecMet/FixedH/Ket/15T/SRel/SRac/SUCF options are specified in the command line, the generated InChI will be non-standard.

Saving InChI creation options

Since the software v. 1.03, the command-line option "/SaveOpt" ("-SaveOpt" under Linux) was introduced. It allows one to append saved InChI creation options to a non-standard InChI string.

The "SaveOpt appendix" currently consists of the two capital Latin letters which are separated from the InChI string by a backslash '\'. Note that this appendix is not considered as an integral part (layer) of InChI itself; rather, it is an optional complement. It may or may not be present after the end of an InChI string (by default – no "SaveOpt" option – it is absent). To signify this, the appendix is separated from the previous sequence of symbols by a character which may not appear in any other place, a backslash.

Note also that the InChI generation option "/SaveOpt" (and the saved-options appendix) is not available for standard InChI as the latter is always created with the same options.

As for the encoding of saved options, the first SaveOpt letter encodes whether RecMet/FixedH/SUU/SLUUD switches were activated. Each of them is a binary switch ON/OFF, giving a total of 2*2*2*2=16 values which are encoded by capital letters 'A' through 'P'.

The second letter encodes experimental (InChI 1 extension) options KET and 15T. Each of these options is a binary switch ON/OFF, so there are 2*2=4 combinations, encoded by 'A' through 'D'. Note that anything but 'A' here would indicate "extended" InChI 1. Note that here is some reservation for future needs: the 2nd memorization character may accommodate two more ON/OFF binary options (at 26-base encoding).

The exact encoding scheme is specified in the tables below.

Table 4. Meaning of the 1st SaveOpt letter.

Letter	RecMet	FixedH	SUU	SLUUD

A	OFF	OFF	OFF	OFF
B	OFF	OFF	OFF	ON
C	OFF	OFF	ON	OFF
D	OFF	OFF	ON	ON
E	OFF	ON	OFF	OFF
F	OFF	ON	OFF	ON
G	OFF	ON	ON	OFF
H	OFF	ON	ON	ON
I	ON	OFF	OFF	OFF
J	ON	OFF	OFF	ON
K	ON	OFF	ON	OFF
L	ON	OFF	ON	ON
M	ON	ON	OFF	OFF
N	ON	ON	OFF	ON
O	ON	ON	ON	OFF
P	ON	ON	ON	ON

Table 5. Meaning of the 2nd SaveOpt letter.

Letter	Ket	15T
A	OFF	OFF
B	OFF	ON
C	ON	OFF
D	ON	ON

Examples:

```
InChI=1/C9H11NO2.Na/c1-3-5(7(3)9(10)12)6-4(2)8(6)11;/h5-
6,11H,1-2H3,(H2,10,12);/q;+1/p-
1/t5?,6?;/i/hD/fC9H10NO2.Na/h11h,10H2;/q-1;m/i10D;\OA
```

(this identifier was created with options /RecMet /FixedH /SUU and /SaveOpt)

```
InChI=1/C9H11NO2.Na/c1-3-5(7(3)9(10)12)6-4(2)8(6)11;/h5-
6,11H,1-2H3,(H2,10,12);/q;+1/p-1/t5?,6?;/i/hD\KA
```

(this identifier was created for the same input structure with options /RecMet /SUU and /SaveOpt)

```
InChI=1S/C9H11NO2.Na/c1-3-5(7(3)9(10)12)6-4(2)8(6)11;/h5-
6,11H,1-2H3,(H2,10,12);/q;+1/p-1/i/hD
```

(this identifier was created for the same input structure with no InChI creation options)

The next table summarizes the availability of various options in the various parts of the InChI Software.

Table 6. Availability of InChI Software options.

Options availability			Command line option (without / or – prefix)	Explanation
winchi	inchi-1	API calls		
Input				
–	Yes	–	STDIO	Use standard input/output streams
–	Yes	–	InpAux	Input structures in InChI default aux. info format (for use with STDIO)
Yes	Yes	Yes	SDF:*name*	Read from the input SDfile the ID under the named data header
–	Yes	–	AMI	Allow multiple input files
Output				
–	Yes	Yes	AuxNone	Do not produce Auxiliary Information
–	Yes	–	NoLabels	Omit structure number, DataHeader and ID from InChI output
–	Yes	Yes	SaveOpt	Save custom InChI creation options
–	Yes	–	Tabbed	Separate structure number, InChI, and AuxIndo with tabs
–	Yes	–	OutErrInChI	On fail, print empty InChI (default: nothing)
Always	Yes	–	D	Display the structure
Yes	Yes	–	Equ	Display sets of identical components
–	Yes	–	F*number*	Set display font size (points)
–	Yes	Yes	OutputSDF	Convert InChI created with default auxiliary info to a SDfile
–	Yes	Yes	SdfAtomsDT	Output Hydrogen Isotopes to SDfile as Atoms D and T
–	Yes	–	AMIOutStd	Write output to stdout (in AMI mode only)

-	Yes	-	AMILogStd	Write log messages to stderr (in AMI mode only)
-	Yes	-	AMIPrbNone	Suppress creation of problem files (in AMI mode only)
Structure perception				
Yes	Yes	Yes	NEWPSOFF	Both ends of wedge point to stereocenters
-	Yes	Yes	DoNotAddH	Do not add H according to usual valences
Yes	Yes	Yes	SNon	Ignore stereo information in input structures
Stereo perception modifiers (non-standard InChI)				
Yes	Yes	Yes	SRel	Relative stereo
Yes	Yes	Yes	SRac	Racemic stereo
Yes	Yes	Yes	SUCF	Use Chiral Flag: On means Absolute stereo, Off - Relative
Customizing InChI creation (non-standard InChI)				
Yes	Yes	Yes	SUU	Always include omitted unknown/undefined stereo
Yes	Yes	Yes	SLUUD	Make labels for unknown and undefined stereo different
Yes	Yes	Yes	RecMet	Include reconnected metals results
Yes	Yes	Yes	FixedH	Include Fixed H layer
Yes	Yes	Yes	KET	Account for keto-enol tautomerism (experimental)
Yes	Yes	Yes	15T	Account for 1,5-tautomerism (experimental)
Always	Yes	Always	Polymers	Experimental support of simple polymers
Always	Yes	Yes	LargeMolecules	Experimental support of molecules up to 32767 atoms
No	Yes	No	OutErrInChI	Output empty InChI and corresponding InChIKey if error occurs
Generation				

60 sec	Yes *)	Yes*)	Wnumber	Set time-out per structure in seconds
–	Yes	Yes	WarnOnEmptyStructure	Warn and produce empty InChI for empty structure
Always	Yes	– **)	Key	Generate InChIKey
Always	Yes	– **)	XHash1	Generate hash extension (to 256 bits) for 1st block of InChIKey
Always	Yes	– **)	XHash2	Generate hash extension (to 256 bits) for 2nd block of InChIKey
Conversion				
–	Yes	–	InChI2Struct	Convert standard InChI string(s) into structure(s)
–	Yes	–	InChI2InChI	Convert InChI string(s) into InChI string(s)

*) W0 means unlimited time. In InChI Library the default is W0, in inchi the default is 60 seconds (W60).
**) In InChI Library, generation of InChIKey/hash extensions is performed via a separate API call.

Test Files

A number of Molfiles (*.mol) and two SDfiles (*.sdf) are included with the program for illustrative purposes. Some Molfiles contain more than one fragment – each may be viewed separately using the 'combo-box' on the upper left of the screen. Multiple structures are given in the SDfiles, which may be viewed in order by pressing the 'Next Structure' (">>") and 'Previous Structure' ("<<") buttons. File Samples.sdf contains all of the individual Molfiles from Samples.zip. These SDfiles contain names of the structures. To display them enter word "name" (without quotes) in "Structure ID Header" field (Fig. 2).

IV. CHEMICAL STRUCTURE INPUT

Molfile structures may be submitted either as a single Molfile or as a series of concatenated Molfiles (an SDfile). A number of programs, some of them freely available, may be used to create these Molfiles. Information on how to produce and convert If an input structure

contains more than one independent structure, each component is individually shown in the graphical output section of the program, though this has no effect on the InChI. Text results are given for all layers and all components (different components of a single substance are separated by semicolons in each layer, except for chemical formulas, which, by convention, are separated by dots.).

While structure normalization methods built into the program perceive a range of different structure drawing conventions, it is possible that other conventions may not be properly recognized. Examination of the graphical results of InChI processing, especially for equivalent atom classes and stereo labeling, should reveal such problems.

If an SDfile is 'labeled', the program can supply these labels in its output. If the tag name is 'Name' and the data field is '2-methylanthracene', this information would appear in the SDfile as 3 lines (the last line is blank):

```
>   <Name>
 2-methylanthracene
```

In this case, if the tag 'Name' is entered in the 'Structure ID Header' field in the input dialog box, '2-methylanthracene' will appear in the output text.

A variety of structure files are provided for testing. Individual Molfiles have extension .MOL, concatenated Molfiles have extension .SDF.

Since v. 1.05, the ability to read and parse large (up to 32767 atoms) input files in Molfile V3000 format was added to the inchi-1 executable and the API procedure MakeINCHIFromMolfileText(). This is necessary for treating large molecules (previous versions supported only V2000 format limited to not more than 1000 atoms).

In addition, provisional support for extended features of Molfile V3000 was also added, both to inchi-1 and the InChI Software Library, API. This means that extended data (on haptic coordination bonds and stereo collections) are read and parsed; however, they are not used currently (as this requires significant modification of the Identifier itself, not just the Software).

InChI for polymers

Since v. 1.05, InChI supports regular single-strand polymers. Both structure-based and source-based representation and encoding of polymers are supported.

Executable inchi-1 supports reading input Molfile files containing polymer description lines[3]. This support is also built into the API procedure MakeINCHIFromMolfileText() and the demo program mol2inchi included in this distribution. To generate InChI from molecular data stored in other formats, one may use InChI API Library polymer-aware procedures (GetINCHIEx() and new IXA calls) which accept specifically polymer-extended input data structures

Note that support of polymers is an experimental feature. To emphasize this, InChI/InChIKey for a polymer uses the 'B' flag character (for "Beta"), instead of 'S' or 'N' for standard/non-standard InChI. It is supposed that this flag will be replaced by common standard/non-standard conventions if and when InChI for polymers is finally adopted. Also, by default the executable inchi-1 ignores polymer-specific data (which also ensures compatibility with the behaviour of previous versions); to allow treatment of polymers, one should explicitly use the new command line option `Polymers` (`-Polymers` under Linux or `/Polymers` under Windows).

An additional optional "modification" layer has been added to the InChI layout to encode polymeric structures. This layer starts from two symbols '/z' and is located immediately before the stereo sub-layer (if any) of the main InChI layer (for metal-containing structures polymer layer may appear twice).

Quick examples:

InChI for styrene-butadiene block copolymer, source-based representation (entries 11 and 12 in Table 2 below):

[3] Properties block of V2000 format, lines: "M STY", "M SAL", "M SBL", "M SST", "M SCN", "M SLB", "M SDI", "M SMT"; see [CTFile Formats. Accelrys, December 2011. http://accelrys.com/products/collaborative-science/biovia-draw/ctfile-no-fee.html]

```
InChI=1B/C8H8.C4H6/c1-2-8-6-4-3-5-7-8;1-3-4-2/h2-7H,1H2;3-
4H,1-2H2/z200-9-12;200-1-8;330-1-12

InChIKey=MTAZNLWOLGHBHU-ZNVYRHKRBA-N
```

InChI for polycaprolactam, structure-based representation (entry 37 in Table 2):

```
InChI=1B/C6H10O2/c7-6-4-2-1-3-5-8-6/h1-5H2/z101-1-
8(1,2,1,3,2,4,3,5,4,6,5,8,6,8)

InChIKey=PAPBSGBWRJIAAV-CMRMDLKMBA-N
```

InChI for coordination polymer, zinc-containing fungicide Zineb, structure-based representation produced with RecMet option (entry 64 in Table 2):

```
InChI=1B/C4H8N2S4.Zn/c7-3(8)5-1-2-6-4(9)10;/h1-
2H2,(H2,5,7,8)(H2,6,9,10);/q;+2/p-2/z101-1-
11(1,5)/rC4H6N2S4Zn/c1-2-6-4-9-11(10-4)7-3(5-1)8-11/h5-
6H,1-2H2/z101-1-11(1,5)

InChIKey=AMHNZOICSMBGDH-ZPQHTIIXBA-L
```

For more details on InChI for polymers and related drawing/preparing input data rules please refert to InChI v. 1.05 Release Notes document RelNotes.pdf accompanying this distribution.

V. Further reading and contacts

In addition to this introductory User Guide, a number of materials concerning InChI is currently available.

In particular, this distribution contains separate documents (PDF files) with InChI Technical Manual (InChI_TechMan.pdf) and InChI API Reference (InChI_API_Reference.pdf). For more brief and less technical description, look at: *Heller, S., McNaught, A., Pletnev, I., Stein, S., and Tchekhovskoi, D. InChI, the IUPAC international chemical identifier. Journal of Cheminformatics 7 (2015), 23–23*. DOI:

10.1186/s13321-015-0068-4 For much more brief description, as well as background and history, look at*: Heller, S., McNaught, A., Stein, S., Tchekhovskoi, D., and Pletnev, I. InChI - the worldwide chemical structure identifier standard. Journal of Cheminformatics 5 (2013), 7–7*. DOI: 10.1186/1758-2946-5-7

For InChI FAQ, address to: http://www.inchi-trust.org/technical-faq/

Contacts:

The InChI Trust
8 Cavendish Avenue
Cambridge CB1 7US
UK

Alan D. McNaught (Company Secretary, InChI Trust)
alan@inchi-trust.org

Steve Heller (InChI project director, InChI Trust)
steve@hellers.com

Igor V. Pletnev (InChI developer)
igor.pletnev@gmail.com