



Chemo-informatics and computational drug design

Prof. Dr. Hans De Winter

University of Antwerp

Campus Drie Eiken, Building A

Universiteitsplein 1, 2610 Wilrijk, Belgium



**Gefinancierd door
de Europese Unie**

NextGenerationEU

Chapter 2. Representing molecules in the computer

1. Molecular formats

In order for computers to work with molecular information, molecules need to be converted in a format that computers can understand. During the last decades many different approaches have been developed, ranging from accurate quantum chemical representations with wavefunctions describing the position of electrons and the corresponding atom nuclei, up to the simplest representations in which only the molecular topology is described (Figure 1).

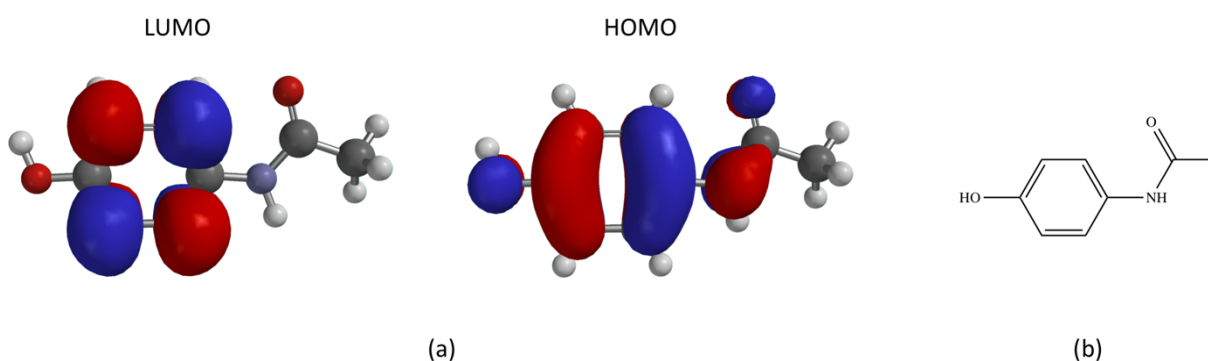


Figure 1. Different representations of the same molecule (paracetamol). (a) Two quantum chemical representations, the left one showing the lowest unoccupied molecular orbital (LUMO) and the right one showing the highest occupied molecular orbital (HOMO). (b) Example of a simple molecular representation in which the atom types and their connections are shown.

Although the fact that quantum chemical representations are considered to be the most accurate and contain the largest amount of *ab initio* information, in practice this approach is not very useful since it is extremely time consuming to generate and very expensive in terms of disk storage space. For this reason, alternative methods have been developed in which the majority of the chemical information is removed and considered to be implicitly present. For example, using the topological molecular representation shown in Figure 1b, it is implicitly assumed that:

- 1) each carbon, oxygen and nitrogen atom contain 6, 8 and 7 electrons, respectively,
- 2) the six-membered ring is aromatic,
- 3) each of the atoms is connected to an appropriate number of hydrogen atoms so that each atomic valence is correctly represented.
- 4) the oxygen atoms contain two lonepairs each and the nitrogen atom contains one lone pair,

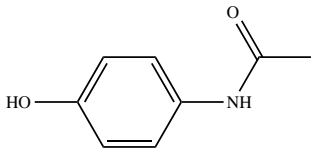
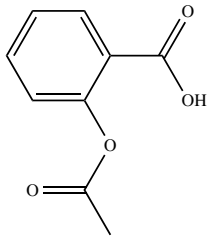
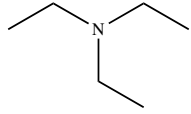
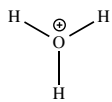
Also, the topological representation does not contain any information about the three-dimensional shape of the molecule, so in this case it is also implicitly assumed that the conformation of this molecule can be generated based on knowledge from other molecules and which has been collected using X-ray techniques (the Cambridge Structural Database is a comprehensive resource of almost a million X-ray structures of small molecules, see <https://www.ccdc.cam.ac.uk>). Hence, the majority of chemical representations assume some prior chemical knowledge; the main difference between most of the molecular formats is mainly in the amount of prior knowledge that is assumed. There are many chemical formats available (see https://en.wikipedia.org/wiki/Chemical_file_format for a complete overview), but the ones most frequently used are the SMILES, SDF and PDB formats.

1.1. SMILES

SMILES stands for Simplified Molecular Input Line Entry Specification and was developed in the 80's by David Weininger of Daylight Chemical Information Systems Inc. in the USA [1]. The format is a specification in the form of a line notation as it describes the structure of molecules using short sequences of characters (ASCII strings). A

complete format description is available at <http://www.daylight.com/dayhtml/doc/theory/theory.smiles.html>. Some examples are given in Table 1:

Table 1. Examples of SMILES representations

Name	Structure	Example SMILES
Paracetamol		<chem>Oc1ccc(NC(C)=O)cc1</chem>
Aspirin		<chem>O=C(O)c1ccccc1OC(C)=O</chem>
Triethylamine		<chem>CCN(CC)CC</chem>
Hydronium ion		<chem>[H][O+]([H]) [H]</chem> <chem>[OH3+]</chem>
Hydrogen cyanide	<chem>H—C≡N</chem>	<chem>N#C</chem>

SMILES notation consists of a series of characters containing no spaces. Hydrogen atoms may be omitted or included. Aromatic structures may be specified directly or in Kekulé form. There are five generic SMILES encoding rules, corresponding to specification of atoms, bonds, branches, ring closures, and disconnections.

Atoms

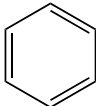
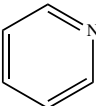
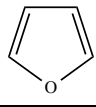
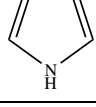
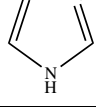
Atoms are represented by their standard abbreviation of the chemical elements and surrounded by square brackets, such as [Ag] for silver. In practice however, brackets are omitted in cases of atoms which:

- are in the organic subset of B, C, N, O, P, S, F, Cl, Br, or I, *and*
- have no formal charge, *and*
- have the number of hydrogens attached implied by their normal valences, *and*
- are the normal isotopes, *and*
- are not chiral centres.

All other elements or configurations must be enclosed in brackets, and have charges and hydrogens that are specified explicitly. For instance, the SMILES for water may be written as O or [OH2] or [H]O[H], but the hydronium ion can only be specified as [OH3+] or [H][O+]([H]) [H]. When brackets are used, the symbol H should be added if the atom in brackets is bonded to one or more hydrogen, followed by the number of hydrogen atoms if this is greater than 1, then by the sign '+' for a positive charge or by '-' for a negative charge. For example, [NH4+] should be used to represent ammonium, although that [H][N+]([H]) ([H]) [H] may also be used. If there is more than one charge, it is normally written as a digit like [Ca+2].

Aromatic atoms, like in the benzene ring, may also be represented in their lower-case form. For the B, C, N, O, P and S atoms this becomes 'b', 'c', 'n', 'o', 'p' and 's', respectively. Aromatic nitrogen bonded to hydrogen, as found in pyrrole, must be represented as [nH] (Table 2):

Table 2. SMILES representation of aromaticity

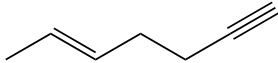
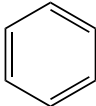
Name	Structure	Example SMILES
Benzene		<chem>c1ccccc1</chem> <chem>C1=CC=CC=C1</chem>
Pyridine		<chem>n1ccccc1</chem> <chem>C1=CC=CN=C1</chem>
Furan		<chem>o1ccccc1</chem> <chem>C1=COC=C1</chem>
Pyrrrole		<chem>c1ccc[nH]1</chem> <chem>C1=CNC=C1</chem>
Imidazole		<chem>n1c[nH]cc1</chem> <chem>C1=CNC=N1</chem>

Bonds

A single bond is represented using the symbol '-'. Bonds between non-aromatic atoms are assumed to be single unless specified otherwise and are implied by the adjacency of the atoms in the SMILES string. Although single bonds may be written as "-", this is usually omitted. For example, the SMILES for ethanol may be written as C-C-O, CC-O or C-CO, but is usually written as CCO or OCC.

Double and triple bonds are represented by the symbols '=' and '#', respectively, as illustrated by the SMILES of carbon dioxide (O=C=O) and hydrogen cyanide (C#N). An aromatic 'one and a half' bond may be indicated with ':', although that the ':' may be omitted when lower-case atom symbols are used (Table 3).

Table 3. Different SMILES representations of bonds

Name	Structure	Example SMILES
5-hepten-1-yne		<chem>CC=CCCC#C</chem> <chem>C-C=C-C-C-C#C</chem>
Benzene		<chem>c1ccccc1</chem> <chem>C1=CC=CC=C1</chem> <chem>C1=C-C=C-C=C-1</chem> <chem>C1:C:C:C:C:C:1</chem>

Rings

Ring structures are written by breaking each ring at an arbitrary point to make an acyclic structure and adding numerical ring closure labels to show connectivity between non-adjacent atoms. For example, cyclohexane and dioxane may be written as C1CCCCC1 and O1CCOCC1 respectively. And methylcyclohexane and 1,2-dimethylcyclohexane are represented as CC1CCCCC1 and CC1CCCCC1C, respectively (Figure 2).

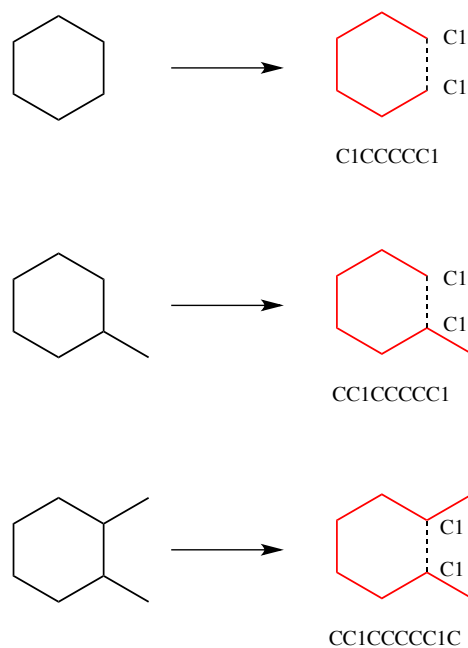


Figure 2. Illustration of how the ring labeling procedure works for the SMILES representation. Cyclic structures are represented by breaking one bond in each ring. The bonds are numbered in any order, designating ring opening (or ring closure) bonds by a digit immediately following the atomic symbol at each ring closure.

The choice of the numerical ring closure label is arbitrary, for example cyclohexane may just as well be represented as C2CCCCC2 or C0CCCCC0 (Table 4). For a second ring, the label will need to be different than the first label. For example, decalin (decahydronaphthalene) may be written as C1CCCC2C1CCCC2.

SMILES does not require that ring numbers be used in any particular order, and permits ring number zero, although this is rarely used. Also, it is permitted to re-use ring numbers after the first ring has closed, although this usually makes formulae harder to read. For example, bicyclohexyl is usually written as C1CCCCC1C2CCCCC2, but it may also be written as C0CCCCC0C0CCCCC0. Multiple digits after a single atom indicate multiple ring-closing bonds. For example, an alternative SMILES notation for decalin is C1CCCC2CCCC12, where the final carbon participates in both ring-closing bonds 1 and 2.

Ring-closing digits may be preceded by a bond type. For example, cyclopropene is usually written C1=CC1, but if the double bond is chosen as the ring-closing bond, it may be written as C=1CC1, C1CC=1, or C=1CC=1.

Table 4. Different SMILES representations of rings

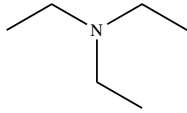
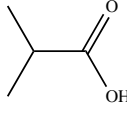
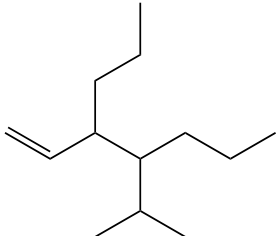
Name	Structure	Example SMILES
Cyclohexane		<chem>C1CCCCC1</chem> <chem>C0CCCCC0</chem> <chem>C2CCCCC2</chem>
Bicyclohexyl		<chem>C1CCCCC1C2CCCCC2</chem> <chem>C0CCCCC0C0CCCCC0</chem>
Decalin		<chem>C1CCCC2C1CCCC2</chem> <chem>C1CCCC2CCCC12</chem>

Branching

Branches are specified by enclosing them in parentheses, and can be nested or stacked. In all cases, the implicit connection to a parenthesized expression (a branch) is to the left. The first atom within the parentheses, and the first atom after the parenthesized group, are both bonded to the same branch point atom. Branches may be

written in any order. For example, bromochlorodifluoromethane may be written as FC (Br) (Cl) F, BrC (F) (F) Cl, C (F) (Cl) (F) Br, or the like. Examples of branches are given in Table 5.

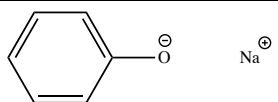
Table 5. Different SMILES representations of branches

Name	Structure	Example SMILES
Triethylamine		<chem>CCN(CC)CC</chem>
Isobutyric acid		<chem>CC(C)C(=O)O</chem>
3-Propyl-4-isopropyl-1-heptene		<chem>C=CC(CCC)C(C(C)C)CCC</chem>

Disconnected structures

Disconnected compounds are written as individual structures separated by a '.' (dot). The order in which the ions or ligands are listed is arbitrary. There is no implied pairing of one charge with another, nor is it necessary to have a net zero charge. If desired, the SMILES of one ion may be imbedded within another as shown in the example of sodium phenoxide (Table 6):

Table 6. Different SMILES representations sodium phenoxide illustrating disconnected structures.

Name	Structure	Example SMILES
Sodium phenoxide		<chem>[Na+].[O-]c1ccccc1 c1cc([O-].[Na+])ccc1</chem>

Stereochemistry

The SMILES representation permits, but does not require, specification of stereoisomers, which can be stereoisomerism around double bonds or stereoisomerism of asymmetric centres. The term *isomeric SMILES* collectively refers to SMILES written using these stereochemical rules.

Configuration around double bonds is specified using the characters '/' and '\' to show directional single bonds adjacent to a double bond. For example, F/C=C/F (see Table 7) is one representation of *trans*-1,2-difluoroethylene, in which the fluorine atoms are on opposite sides of the double bond, whereas F/C=C\F is one possible representation of *cis*-1,2-difluoroethylene, in which the F's are on the same side of the double bond.

Bond direction symbols always come in groups of at least two, of which the first is arbitrary. That is, F\C=C\F is the same as F/C=C/F. When alternating single-double bonds are present, the groups are larger than two, with the middle directional symbols being adjacent to two double bonds. For example, the common form of (2,4)-hexadiene is written C/C=C/C=C/C (Table 7).

Table 7. Different SMILES representations of double bond stereochemistry

Name	Structure	Example SMILES
<i>Trans</i> -1,2-difluoroethylene		<chem>F/C=C/F</chem> <chem>F\C=C\F</chem>
<i>Cis</i> -1,2-difluoroethylene		<chem>F/C=C\F</chem> <chem>F/C=C\F</chem>
(2 <i>E</i> ,4 <i>E</i>)-hexa-2,4-diene		<chem>C/C=C/C=C/C</chem>
<i>Cis</i> -cyclooctene		<chem>C1CCC/C=C\CC1</chem>

Configuration at tetrahedral carbon is specified by '@' or '@@'. SMILES uses a very general type of chirality specification based on local chirality. Instead of using a rule-based numbering scheme to order neighbour atoms of a chiral centre, orientations are based on the order in which neighbours occur in the SMILES string.

The simplest and most common kind of chirality is tetrahedral; four neighbour atoms are evenly arranged about a central atom, known as the chiral centre. Consider the four bonds in the order in which they appear, left to right, in the SMILES form. Looking toward the central carbon from the perspective of the first bond, the other three are either clockwise or counter-clockwise. These cases are indicated with '@@' and '@', respectively, as the @ symbol itself is a counter-clockwise spiral. The symbol '@' indicates that the following neighbours are listed anticlockwise. '@@' indicates that the neighbours are listed clockwise.

Consider for example the amino acid alanine (Table 8). One of its SMILES forms is NC(C)C(=O)O, more fully written as N[CH](C)C(=O)O. This SMILES representation does not include stereochemical information. *L*-alanine however, the more common enantiomer, is written as N[C@@H](C)C(=O)O. Looking from the N-C bond, the hydrogen (H), methyl (C), and carboxylate (C(=O)O) groups appear clockwise. *D*-Alanine can be written as N[C@H](C)C(=O)O.

Table 8. Different SMILES representations of tetrahedral stereochemistry

Name	Structure	Example SMILES
Alanine		<chem>NC(C)C(=O)O</chem> <chem>N[CH](C)C(=O)O</chem> <chem>N[CH](C)C(=O)[OH]</chem>
<i>L</i> -Alanine		<chem>N[C@@H](C)C(=O)O</chem>
<i>D</i> -Alanine		<chem>N[C@H](C)C(=O)O</chem>

The SMILES specification includes elaborations on the @ symbol to indicate stereochemistry around more complex chiral centres, such as trigonal bipyramidal molecular geometry. For a complete reference, please refer to the online manual at <http://www.daylight.com/dayhtml/doc/theory/theory.smiles.html>.

Canonical SMILES

Any given molecule can be represented by a wide variety of SMILES representations, all depicting the same molecule. For example, methane can be represented as C, [H][C]([H])[H], or [CH4]. In order to overcome this 'random' behaviour of the SMILES representation, the concept of a canonical SMILES has been introduced. Canonicalization is a way to determine which of all possible SMILES will be used as the reference SMILES for a molecular graph. Suppose you want to find if a structure already exists in a data set. In graph theory this is the graph isomorphism problem. Using the canonical SMILES instead of the graphs reduces the problem to a simple text matching problem.

Hence canonicalization has some nice advantages but unfortunately there exists no universal canonical SMILES. Every toolkit or program uses a different algorithm, and sometimes the algorithm changes with different versions of the toolkit. There are even different forms of canonical SMILES, depending on if atomic properties like isotope are important for the result. Therefore, if one wants to compare molecules based on their canonical SMILES representations, care should be taken that all these canonical SMILES' have been generated by the same toolkit.

1.2. InChi

InChi stands for IUPAC International Chemical Identifier, and is a universal textual identifier or name for chemical compounds. It is designed to provide a unique and canonical way to encode the molecular topology into a text string, facilitating retrieval and storage of chemical information in molecular databases. The format and algorithms of the InChi format is freely available, ensuring thereby that there is only one standard. All information on the InChi standard can be found on <http://www.iupac.org/home/publications/e-resources/inchi.html>.

The InChi descriptor describes chemical substances in terms of layers of information: 1) the atoms and their bond connectivity, 2) tautomeric information, 3) isotope information, 4) stereochemistry, and 5) electronic charge information. Not all layers have to be provided; for instance, the tautomer or stereochemistry layers can be omitted if these types of information are not relevant or not defined. For example, the InChi description for ethanol is:

```
InChI=1S/C2H6O/c1-2-3/h3H,2H2,1H3
```

Every InChi starts with the string 'InChi=', followed by the version number (which is currently 1) and the letter 'S' to denote standard InChi. The remaining information is structured as a sequence of layers and sublayers. The standard four layers are the 'main layer', the 'charge layer', the 'stereochemical layer' and the 'isotopic layer'.

Main layer

The main layer provides information about the chemical formula (no prefix), the atom connections (prefix 'c/'), and the hydrogen atoms (prefix 'h/')

- The *chemical formula* sublayer is the only layer that must occur in every InChi. Without this sublayer an InChi string is invalid.
- The *atom connection* sublayer is indicated by the 'c/' prefix. The atoms defined in the chemical formula sublayer are numbered in sequence (excluding the hydrogens), and this sublayer defines which atoms are connected to which other atoms.
- The *hydrogen atoms* sublayer is indicated by the 'h/' prefix, and describes how many hydrogen atoms are connected to each of the non-hydrogen atoms.

An example of the main layer is given above for ethanol. The chemical formula sublayer states that there are two carbon atoms, six hydrogens and one oxygen. From this chemical formula, the atom numbering for the non-hydrogen atoms can be extracted: C1, C2, O3. The atom connection sublayer (prefix '/c') of the ethanol InChi states that atom 1 is connected to atom 2, and atom 2 is connected to atom 3. Hence the molecular topology, excluding hydrogens so far, is C1-C2-O3. Finally, hydrogen information is defined in the hydrogen atoms sublayer (prefix '/h'), stating that O3 is connected to one hydrogen ('3H'), C2 connected to two hydrogens ('2H2'), and C1

connected to three hydrogens ('1H3'). Branches and ring closures are given in parentheses, as exemplified in Figure 3.

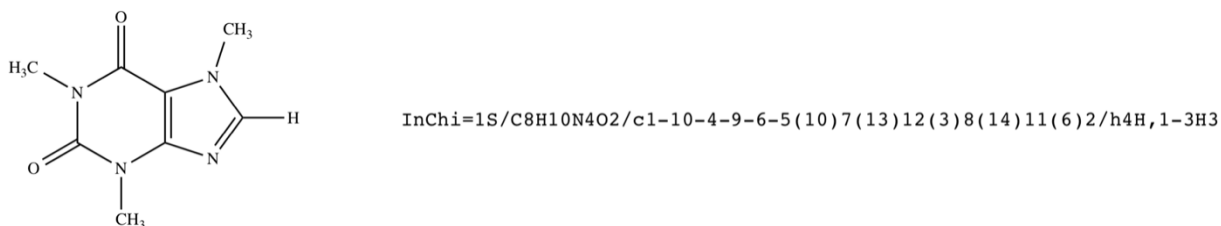


Figure 3. InChi description of caffeine. Atom sequence is determined by a canonicalization algorithm to ensure that the atom sequence is not depending on how the structure was entered or drawn (for more information, see <http://depth-first.com/articles/2006/08/12/inchi-canonicalization-algorithm/>).

Charge layer

The charge layer consists of two sublayers:

- The *total charge* sublayer '/q' is the net charge of the core parent structure.
- The *protonation/deprotonation* sublayer '/p' indicates the net number of protons that need to be removed from (or added to) the structure when derived from its core parent. An example illustrating this is given in Figure 4:

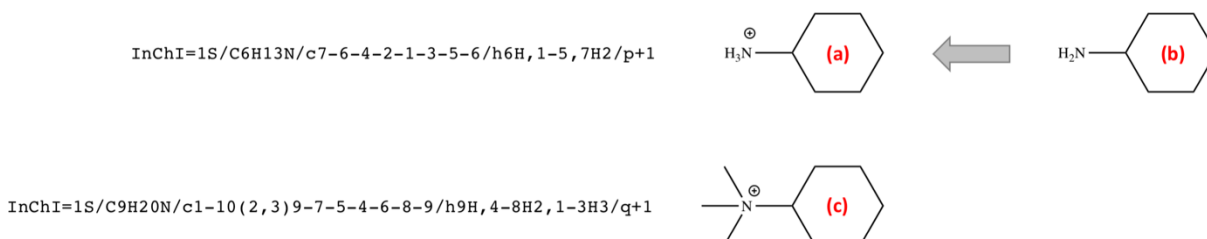


Figure 4. InChi notations of two charged compounds. The first compound (a) is the protonated form of the neutral core (b). For this reason, the InChi string of (a) is not containing a total charge sublayer '/q', since the total charge of its core structure (b) equals zero. However, the InChi of compound (a) contains a protonation/deprotonation sublayer '/p' to indicate that the compound contains an extra proton compared to its core structure (b). It is up to the processing software to determine where this proton should be positioned, as this is not defined in the hydrogen atoms sublayer '/h' (which indicates the presence of a single hydrogen on atom 6, two hydrogens on atoms 1-5 and 7, and two hydrogens on atom 7, the latter being the N). Compound (c) contains a formal charge which cannot be removed by deprotonation; hence the InChi of this compound contains a total charge sublayer '/q' defining a total charge of +1, and the InChi is lacking a protonation/deprotonation sublayer '/p'.

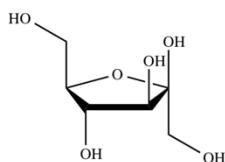
Other layers

The two main remaining layers include the stereochemical layer and the isotopic layer. The stereochemical layer contains sublayers representing double bond sp^2 stereochemistry ('/b') and tetrahedral sp^3 stereochemistry ('/t'). The isotopic layer (signified with the prefix '/i') identifies different isotopically labelled atoms. Exchangeable isotopic hydrogen atoms (deuterium and tritium) are listed separately.

1.3. InChIKey

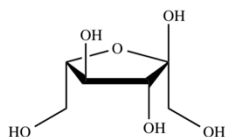
The length of an InChi string depends on the number of atoms in the molecule. Due to this variable length, InChi strings are not always the preferred manner to be used as a key in database queries. In order to tackle this shortcoming, the InChIKey was introduced in 2007. An InChIKey is a fixed-length (27 characters) condensed digital representation. The first part is 14 characters long and encodes the molecular skeleton (connectivity). After a hyphen, there is a second string of 10 characters, the first eight of which encode stereochemistry and isotopes, and after that, 'S' indicates that the key was produced from standard InChI and 'A' indicates that version 1 of InChI was used. The final character, 'N', means 'neutral' (Figure 5). Both parts of the InChIKey are based on a truncated SHA-256 hash of the corresponding InChI layers. For encoding of the data, only uppercase ASCII letters are used

which ensures that the indexing engines will not split the data and also avoids case-sensitivity problems. There is a finite, but extremely small probability of finding two structures with the same InChIKey.



D-Fructose

InChI=1S/C6H12O6/c7-1-3-4(9)5(10)6(11,2-8)12-3/h3-5,7-11H,1-2H2/t3-,4-,5+,6-/m1/s1
RFSUNEUAIZKAJO-ARQDHWQXSA-N



L-Fructose

InChI=1S/C6H12O6/c7-1-3-4(9)5(10)6(11,2-8)12-3/h3-5,7-11H,1-2H2/t3-,4-,5+,6+/m0/s1
RFSUNEUAIZKAJO-UNTFVMJOSA-N

Figure 5. Illustration of the InChIKey concept. Shown are D- and L-fructose with their corresponding InChi and InChiKey representations. Indicated in red are the differences between InChi and InChiKey of the two molecules. Since D- and L-fructose are stereoisomers of each other's, differences are only present in the stereochemistry sublayer of the InChi string. Use of InChiKey allows searches based solely on atom connectivity (the first 14 characters), as the stereoisomers D-fructose and L-fructose both have the same first block of 14 characters, RFSUNEUAIZKAJO.

1.4. MOL-block

The MOL-file (sometimes also called MOL-block) is a chemical file format capable of representing a single chemical structure and its associated data fields. The MOL format was developed by Molecular Design Limited (MDL). The molfile consists of a header and a connection table (CT). Below is a sample chemical record in MOL format of L-alanine, of which the corresponding structure is shown in **Error! Reference source not found.** below:

```
L-Alanine
ChemDraw
This is a comment line
..6..5..0..0..1..0.....999.V2000
...-0.6622...0.5342...0.0000.C...0..0.....0..0..0..0
...0.6222...-0.3000...0.0000.C...0..0.....0..0..0..0
...-0.7207...2.0817...0.0000.C...1..0.....0..0..0..0
...-1.8622...-0.3695...0.0000.N...0..3.....0..0..0..0
...0.6220...-1.8037...0.0000.O...0..0.....0..0..0..0
...1.9464...0.4244...0.0000.O...0..5.....0..0..0..0
..1..2..1..0..0..0
..1..3..1..1..0..0
..1..4..1..0..0..0
..2..5..2..0..0..0
..2..6..1..0..0..0
M..CHG..2...4...1...6...-1
M..ISO..1...3...13
M..END
```

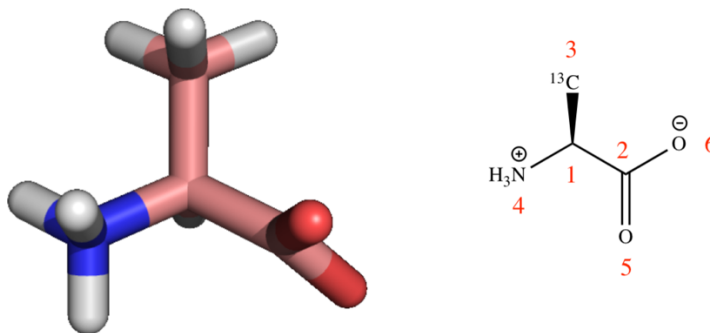


Figure 6. Structure of L-alanine of which the corresponding MOL representation is given above. This structure also shows the hydrogen atoms for clarity, but these hydrogen atoms have been omitted in the MOL-file above (in this example, only the non-H atoms are stored in the MOL-file).

There are currently two main versions that differ considerably, namely V2000 and V3000. V2000 is the oldest, yet still widely used. For that reason, we will only consider V2000 (the [V3000 format is available online](#)).

The V2000 version of a MOL-block is comprised of three parts: the header block, the connection table and the data items.

Header block

The first part of the SDF file consist of a three-line **header block** (the first three black lines in the example above). These three lines **may** contain:

- Name of the molecule (**L - a l a n i n e**)
- Details of the software used to generate the structure (**ChemDraw**)
- Comment (**Comment**)

The three lines in the header block may be empty; however, in that specific case there should be three empty lines (the software that reads MOL-files counts on the fact that there are three lines preceding the connection table, see below).

Connection table

The second part consists of the **connection table** (CT) which is composed of a *counts line* (highlighted in red in the example above), an *atom block* (blue), a *bond blocks* (green), and followed by a *properties block* (black).

1) The **counts line** is made up of twelve fixed-length fields - the first eleven are three characters long, and the last field is six characters long:

```
aaabbb111fffcccsssxxrrrpppiiimmmvvvvvv
```

in which:

aaa	= number of atoms
bbb	= number of bonds
111	= number of atoms lists (not used)
fff	= not used anymore
ccc	= chiral flag (0 = not chiral; 1 = chiral)
sss	= number of stext entries (not used)
xxx, rrr, ppp, iii	= not used anymore
mmm	= not used anymore, default set to 999
vvvvvv	= version (should be V2000 in V2000 format)

The first two fields are the most critical, and specify the number of atoms and bonds of the compound. In the example given above, the *L*-alanine compound consists of 6 atoms and 5 bonds. Actually, the compound consists of more than 6 atoms and 5 bonds (13 atoms and 12 bonds), but the given numbers in the file denote the actual number of atoms and bonds presented in the SDF file (implicit hydrogen atoms are not provided in this example). *L*-Alanine is a chiral molecule, so the chiral flag is set to *on*. **V2000** in the counts line denotes the SDF version, with V2000 being the most widely used one and V3000 (the extended connection table) being a more recent version.

2) The **atom block** is made up of atom lines, one line per atom, with the following format:

```
xxxxxxxxxyyyyyyyzzzzzzzzz·aaaddcccssshhbbbvvv
```

in which:

xxxxxxxxxx	= x-coordinate of atom	(10 characters)
yyyyyyyyyy	= y-coordinate of atom	(10 characters)
zzzzzzzzzz	= z-coordinate of atom	(10 characters)
·	= space	(1 character)
aaa	= atom symbol	(3 characters)
dd	= mass difference from the value in the periodic table (default = 0)	(2 characters)
ccc	= charge (0 = 0, 1 = +3, 2 = +2, 3 = +1, 5 = -1, 6 = -2, 7 = -3)	(3 characters)
sss	= atom stereo parity (ignored when read)	(3 characters)
hhh	= ignored	(3 characters)
bbb	= ignored	(3 characters)
vvv	= valence (default = 0)	(3 characters)

Of these, the atom coordinates, atom symbol, mass difference and charge fields are the most important ones; the other fields are often ignored.

3) The **bond block** is made up of bond lines, one line per bond, with the following format:

```
111222tttsss
```

in which:

111	= first atom number (counting starts from 1)	(3 characters)
222	= second atom number	(3 characters)
ttt	= bond type (1 = S; 2 = D; 3 = T; 4 = A; 5 = S or D; 6 = S or A; 7 = D or A; 8 = any)	(3 characters)
sss	= bond stereo (default 0)	(3 characters)

The bond type symbols denote the following: S is a single bond, D is a double bond, T is a triple bond, A is an aromatic bond.

4) The **properties block** is made up of a number of additional properties, and is read line-per-line until an **M END** line is encountered. Each line in the property block is identified by a prefix in the form of **M XXX** with two spaces separating the **M** and **XXX**. The prefix **M END** terminates the properties block. The most important properties are:

- **M·CHGnn8·aaa·vvv ...**

Denotes atomic formal **charges** [with **nn8** denoting the number of entries (1-8), **aaa** the atom number, and **vvv** the charge]. When present, this property supersedes all charge values in the atom block, forcing a zero charge on all atoms not listed in an **M CHG** line.

- M· ·RADnn8·aaa·vvv ...
Denotes atomic **radicals** [with nn8 denoting the number of entries (1-8), aaa the atom number, and vvv the radical property (0 = no radical; 1 = singlet; 2 = doublet; 3 = triplet)]. When present, this property supersedes all charge values in the atom block, forcing a zero charge on all atoms not listed in an M RAD line.
- M· ·ISOnn8·aaa·vvv ...
Denote atomic isotopes [with nn8 denoting the number of entries (1-8), aaa the atom number, and vvv the absolute mass of the atom isotope as a positive integer]. When present, this property supersedes all isotope values in the atom block. Default (no entry) means natural abundance.
- M· ·END
Should be the last line in a MOL-file.

1.5. SDF

The Structure Data Format (SDF) is a chemical file format to represent multiple chemical structure records and associated data fields. It wraps the MOL-block format and extends this with a chemical data block. SDF has been developed by Molecular Design Limited (MDL) and has become the most widely used standard for importing and exporting information on chemicals. Below is a sample chemical record in SDF format of L-alanine, of which the structure is shown in Figure 6 above:

```
L-Alanine
ChemDraw
Comment
..6..5..0..0..1..0.....999.V2000
...-0.6622...0.5342...0.0000.C...0..0.....0..0..0..0
...0.6222...-0.3000...0.0000.C...0..0.....0..0..0..0
...-0.7207...2.0817...0.0000.C...1..0.....0..0..0..0
...-1.8622...-0.3695...0.0000.N...0..3.....0..0..0..0
...0.6220...-1.8037...0.0000.O...0..0.....0..0..0..0
...1.9464...0.4244...0.0000.O...0..5.....0..0..0..0
..1..2..1..0..0..0
..1..3..1..1..0..0
..1..4..1..0..0..0
..2..5..2..0..0..0
..2..6..1..0..0..0
M..CHG..2...4...1...6...-1
M..ISO..1...3..13
M..END
>..<Sample Ref.>
OC101-12

>..<Melting Point>
41.00 - 43.00

>..<B1 Record No.>
304

>..<ID>
304

$$$$
```

The SDF format is comprised of three parts: the header block, the connection table and the data items. The header block and connection table together form the MOL-block as described in the section before. The SDF-format to describe a single molecule consists therefore of the combination of a MOL-block followed by a number of data items (Figure 7):

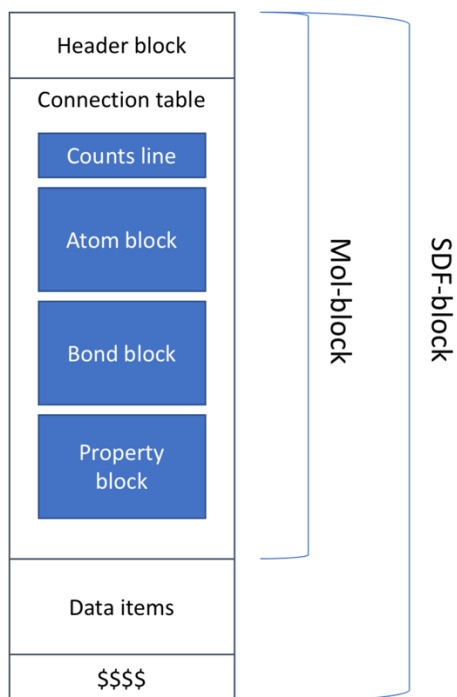


Figure 7. The relation between the different formats and blocks. The connection table consists of a counts line, and an atom, bond and property block (the MOL-block). The SDF-block consists of a MOL-blocks amended with data items and a closing \$\$\$\$ line.

Data items

The SDF format can include associated data items (the orange lines in the previous example of *L*-alanine). These data fields start with a header, which begins with a > character. On the same line, the name of the data field is written in angular brackets. After the header, a data field contains one or more lines of up to 200 characters of free text, which is the value of the data field. Each data field ends with a blank line.

It should be noted that the data items section is not obligatory. In cases when there are no data items to include, then this section may be omitted.

Multiple molecule entries

An interesting property of the SDF-format is that it can contain multiple molecule entries. Each molecule is represented as a separate SDF-block. The end of each molecule entry is denoted with a \$\$\$\$ line.

1.6. PDB

The Protein Data Bank (PDB) format provides a standard representation for macromolecular structure data derived from X-ray diffraction and NMR studies. It is mainly focussed on the representation of protein structures but it may be used for other ligands as well. This representation was created in the 1970's and a large amount of software using it has been written in the meantime. The format is quite extensive and elaborated but the most important aspect is highlighted here.

PDB format consists of lines of information in a text file. Each line of information in the file is called a record. A PDB file generally contains several different types of records, arranged in a specific order to describe a structure. The PDB format of *L*-alanine (**Error! Reference source not found.**) is given here:

```

HETATM...1..N...LIG.....-0.944...-0.430...-0.803...0.00...0.00.....N1+
HETATM...2..C...LIG.....0.362...0.206...-0.576...0.00...0.00.....C
HETATM...3..C...LIG.....0.357...1.563...-1.160...0.00...0.00.....C
HETATM...4..C...LIG.....1.494...-0.664...-1.172...0.00...0.00.....C
HETATM...5..O...LIG.....0.534...2.579...-0.450...0.00...0.00.....O
HETATM...6..O...LIG.....0.159...1.741...-2.486...0.00...0.00.....O1-
CONNECT...1...2
CONNECT...2...1...3...4
CONNECT...3...2
CONNECT...3...5
CONNECT...3...6
CONNECT...4...2
CONNECT...5...3
CONNECT...6...3
END

```

In general, the most important record types are:

- **ATOM**: atomic coordinate record containing the X, Y, Z orthogonal Å coordinates for atoms in standard residues (amino acids and nucleic acids).
- **HETATM**: atomic coordinate record containing the X, Y, Z orthogonal Å coordinates for atoms in nonstandard residues. Nonstandard residues include inhibitors, cofactors, ions, and solvent. The only functional difference from **ATOM** records is that **HETATM** residues are by default not connected to other residues.
- **CONNECT**: specify connectivity between atoms for which coordinates are supplied. The connectivity is described using the atom serial number as shown in the entry. **CONNECT** records are not mandatory for standard protein residues.
- **END**: marks the end of the PDB file.
- **TER**: indicates the end of a chain of residues. For example, a haemoglobin molecule consists of four subunit chains that are not connected. **TER** indicates the end of a chain and prevents the display of a connection to the next chain.

The **ATOM** record is made up, each atom on a separate line, of the following format (not all fields have to be supplied):

```

ATOM...nnnnn...mmmmArrr...csss...xxxxxxxxxyyyyyyyzzzzzzzo0000otttttt.....ee

```

in which:

ATOM	= required field	(4 characters followed by two spaces)
nnnnn	= atom serial number	(5 characters)
..	= 2 spaces	(2 characters)
mmmm	= atom name	(4 characters)
.	= 1 space	(1 character)
A	= alternate location indicator	(1 character)
rrr	= residue name	(3 characters)
.	= 1 space	(1 character)
c	= chain identifier	(1 character)
ssss	= residue sequence number	(4 characters)
....	= 4 spaces	(4 characters)
xxxxxxxx	= X orthogonal Å coordinate	(8 characters)
yyyyyyyy	= Y orthogonal Å coordinate	(8 characters)

zzzzzzzz	= Z orthogonal Å coordinate	(8 characters)
oooooo	= occupancy	(6 characters)
tttttt	= temperature factor	(6 characters)
.....	= 10 spaces	(10 characters)
ee	= element symbol	(2characters)

The **HETATM** is identical as the **ATOM** record, with the exception that the first 6 characters of the **ATOM** record are replaced with **HETATM**.

The **CONNECT** record obeys the following format:

```
CONNECTnnnnnnmmmmmmoooooppvvvvvv
```

in which (all six characters wide):

CONNECT	= required field
nnnnnn	= atom serial number
mmmmmm	= serial number of bonded atom
oooooo	= serial number of bonded atom
pppppp	= serial number of bonded atom
vvvvvv	= serial number of bonded atom

These **CONNECT** records occur in increasing order of the atom serial numbers they carry in columns 7-11. The target-atom serial numbers carried on these records also occur in increasing order. The connectivity list given by the **CONNECT** records is redundant in that each bond indicated is given twice, once with each of the two atoms involved specified in columns 7-11.

The **END** record marks the end of the PDB file. Most software packages do not require an **END** record.

The **TER** record marks the end of a protein or nucleic acid chain. The **TER** records occur in the coordinate section of the entry, and indicate the last residue presented for each polypeptide and/or nucleic acid chain for which there are determined coordinates. For proteins, the residue defined on the **TER** record is the carboxy-terminal residue; for nucleic acids it is the 3'-terminal residue. According the official specifications, the format of a **TER** record is as follows:

```
TER...nnnnn.....rrr.ciiiiix
```

in which:

TER	= required field	(3 characters)
...	= 3 spaces	(3 characters)
nnnnnn	= serial number	(6 characters)
.....	= 6 spaces	(6 characters)
rrr	= residue name	(3 characters)
.	= 1 space	(1 character)
c	= chain identifier	(1 character)
iiii	= residue sequence number	(4 characters)
x	= insertion code	(1 character)

However, in practice often only the **TER** keyword field is specified without any of the other fields.

The PDB format specifies many more keywords, many of these being specific for describing the primary and secondary structures of the polypeptide chains, information about the crystallographic setup and experimental details, and another set of keywords to specify the authors of the PDB file.

2. Molecular graphics

With molecular graphics, molecules and their properties are represented on graphical display devices such as a computer screen. Ever since Dalton's atoms and Kekulé's benzene, there has been a rich history of hand-drawn atoms and molecules, and these representations have had an important influence on modern molecular graphics. Many molecular graphics programs and systems exist and the majority of those graphics systems include editing commands or calculations for use in molecular modelling.

2.1. Visualisation software

There are many molecular graphics programs available, some of these being commercial while other programs are open source and/or free of charge for academic purposes. It is impossible to list all of these tools; hence we will limit ourselves to those open source tools that are most commonly used in our experience.

PyMol

PyMol (<https://pymol.org/2/>) is a free and open source molecular graphics system for visualization, animation, editing, and publication-quality imagery. PyMOL is scriptable and can be extended using the Python language. Supports Windows, Mac OSX, Unix, and Linux (Figure 8).

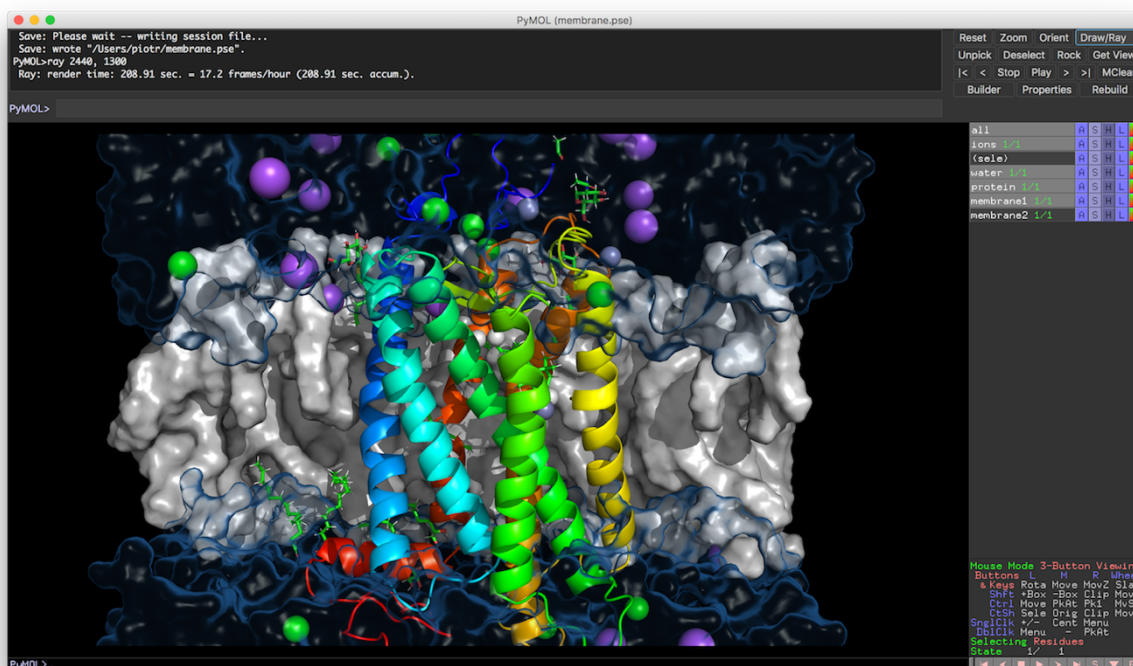


Figure 8. Screenshot of a PyMol session showing a cross-membrane protein represented as ribbons. The membrane part is colored gray, while the protein is colored yellow/green/blue.

PyMol can be extended with many add-ons and public-available scripts, downloadable from the PyMolWiki page (https://pymolwiki.org/index.php/Main_Page). This forum is a powerful resource for questions and answers, and illustrates the power of open source software that is supported by a broad user community.

For advanced visualization problems, PyMol also supports hardware stereographics using active shutters systems. Stereoscopy (also called stereoscopies, or stereo imaging) is a technique for creating or enhancing the illusion of depth in an image by means of stereopsis for binocular vision]. Most stereoscopic methods present two offset images separately to the left and right eye of the viewer. These two-dimensional images are then combined in the

brain to give the perception of 3D depth. An active shutter 3D system is a technique of displaying stereoscopic 3D images. It works by only presenting the image intended for the left eye while blocking the right eye's view, then presenting the right-eye image while blocking the left eye, and repeating this so rapidly that the interruptions do not interfere with the perceived fusion of the two images into a single 3D image. Modern active shutter 3D systems generally use liquid crystal shutter glasses (also called active shutter glasses). Each eye's glass contains a liquid crystal layer which has the property of becoming opaque when voltage is applied, being otherwise transparent. The glasses are controlled by a timing signal that allows the glasses to alternately block one eye, and then the other, in synchronization with the refresh rate of the screen. The timing synchronization to the video equipment may be achieved via a wired signal, or wirelessly by either an infrared or radio frequency transmitter. A well-known system for active shutter 3D visualization is the **3D Vision kit** of NVIDIA. It comes with 3D shutter glasses, a transmitter, and special graphics driver software. While regular LCD monitors run at 60 Hz, a 120 Hz monitor is required to use 3D Vision (Figure 9). A special 3D-enable graphics card is also required.



Figure 9. The 3D Vision kit from NVIDIA to display hardware-enable active stereo. A special 120 Hz monitor, 3D shutter glasses and a transmitter are needed.

VMD

VMD is a molecular visualization program for displaying, animating, and analysing large biomolecular systems using 3-D graphics and built-in scripting. It is an open-source software package that can be downloaded free of charge from <http://www.ks.uiuc.edu/Development/Download/download.cgi?PackageName=VMD> for Mac OS X, Unix or Windows. It is especially suited for advanced visualisation of molecular dynamics trajectories and can be fine-tuned by means of Tcl-Tk-based scripting (Figure 10).

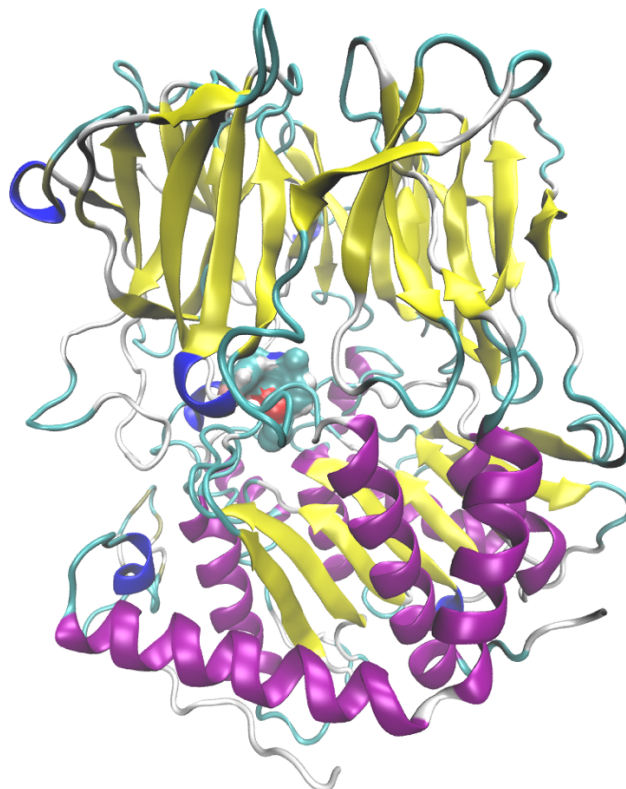


Figure 10. Structure of a propyl endopeptidase protein rendered by the VMD program. The protein peptide backbone is rendered as a ribbon-diagram (see below) and colored by secondary structure (helices in purple, β -sheets in yellow). The GSK-522 ligand is shown in the center of the cavity.

2.2. Molecular representations

CPK

CPK is a three-dimensional space-filling representation in which atoms are represented by spheres. The name CPK refers to the scientists Corey, Pauling and Koltun, who developed this representation concept into a useful form. The radius of each atoms sphere is normally correlated to its atomic number, but sometimes the spheres are drawn with a smaller radius and the bonds are drawn in a cylinder-type of representation (Figure 11).

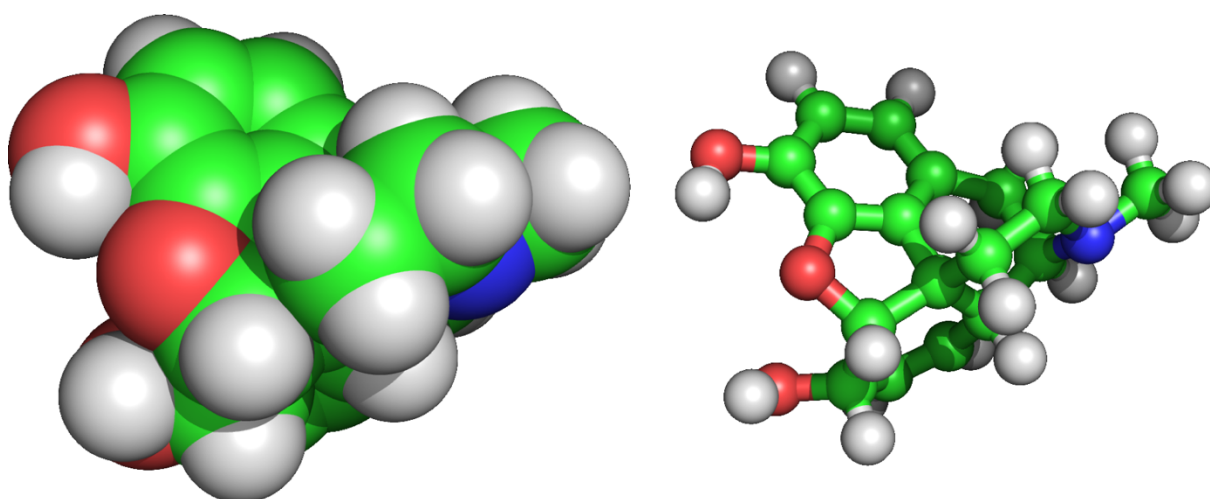


Figure 11. CPK representations of morphine. Left picture is a typical CPK representation in which the atoms are shown as spheres. The right picture shows a CPK model in which the atom radii are reduced and in which the bonds are drawn as cylinders.

Stick representation

In the CPK representation, focus is on the atoms and the corresponding volume the molecule occupies. In a stick representation, focus is rather on the connectivity between the atoms by representing the bonds as cylinders (Figure 12). For computational drug design, this representation is probably the most useful as it permits the researcher to investigate potential interactions between ligand and protein.

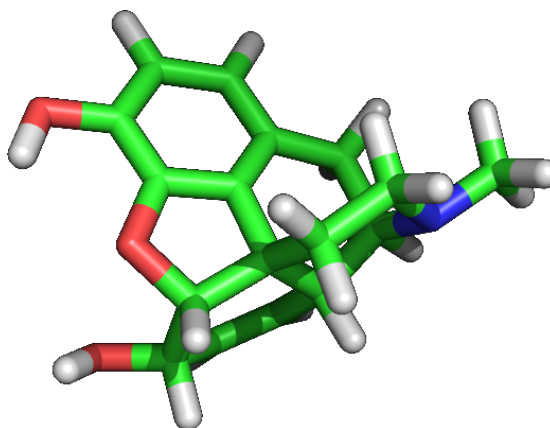


Figure 12. Stick representation of morphine. Only the bonds between atoms are shown, and colored according the participating atom types (C is green, O is red, H white and N blue).

Ribbon diagram

Ribbon diagrams, also known as Richardson diagrams, are 3D schematic representations of protein structure and are one of the most common methods of protein depiction used today. The ribbon shows the overall path and organization of the protein backbone in 3D, and are generated by interpolating a smooth curve through the polypeptide backbone. α -helices are shown as coiled ribbons or thick tubes, β -strands as arrows, and lines or thin tubes for non-repetitive coils or loops. The direction of the polypeptide chain is shown locally by the arrows, and may be indicated overall by a colour ramp along the length of the ribbon.



Figure 13. PyMol ribbon of the structure of the mouse brain tubby protein, which has a characteristic β -barrel fold with a central α -helix (PDB: 1C8Z). β -sheets are shown in yellow, and α -helices in red.

Ribbon diagrams are simple, yet powerful, in expressing the visual basics of a molecular structure (twist, fold and unfold). This method has successfully portrayed the overall organization of the protein structure, reflecting its 3-

dimensional information, and allowing for better understanding of a complex object both by the expert structural biologists and also by other scientists, students, and the general public.

Solvent-accessible surface

The solvent-accessible surface area (SASA) is the surface area of a molecule that is accessible to a solvent, in most cases water. The SASA was first described by Lee & Richards in 1971.² The SASA is typically calculated using the 'rolling ball' algorithm developed by Shrake & Rupley in 1973.³ This algorithm uses a sphere (of solvent) of a particular radius to 'probe' the surface of the molecule. In the case of water as solvent, the radius that is most commonly used is 1.4 Å (Figure 14 and Figure 15).

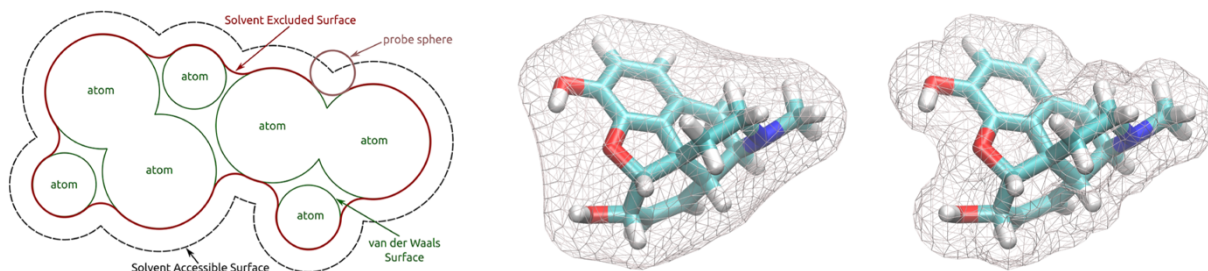


Figure 14. Left: illustration of the principle of the SASA, showing the relationship between probe radius, atom radii. Solvent-excluded surface and van der Waals surface. The solvent-accessible surface (black dotted line) is defined by the area that the center of probe sphere is traversing, while the solvent-excluded area (SES, red line) corresponds to the part of the van der Waals surface that can be touched by the probe, plus the reentrant surface. Middle and right: example of the solvent-excluded surface calculated using two different probe radii (8 Å for the left figure and 1 Å for the right figure).

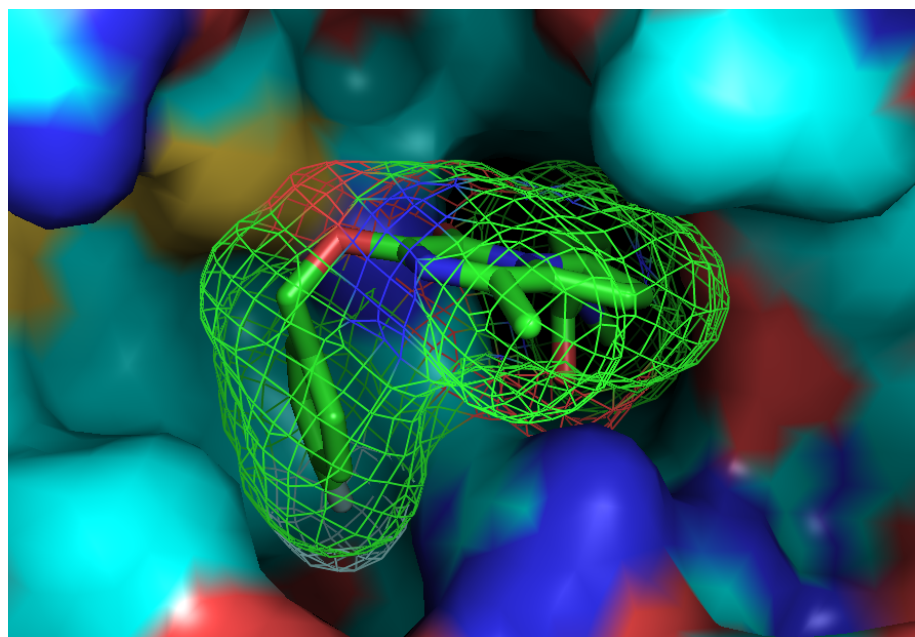


Figure 15. Illustrating the use of the SASA in analyzing shape complementary between ligand (wireframe SASA) and its surrounding receptor (solid surface).

Many programs are available to calculate SASA's, including the PyMol software. The SASA for a typical druglike molecule as shown in Figure 15 is around 300-500 Å², while for a protein like prolyl endopeptidase this area becomes 75,000 Å².