

chemobabel.sty

— Chemical Structures from MDL Molfiles, ChemDraw Files or SMILES Notations —
化学構造式を MOL ファイルや ChemDraw ファイル、SMILES 表記法から自動生成

Acetaminophen (アセトアミノフェン)

June 30, 2015

This document describes the usage of chemobabel.sty and accompanying macros, a package bundle for **generating chemical structural formulas** to be inserted in your L^AT_EX documents. The formulas can be generated **from many kinds of chemical data formats**, including MDL Molfiles, ChemDraw files and even from SMILES notations, with the help of Open Babel and Inkscape. This document is written in both English and Japanese. The formulas below are generated using this method.

この文書では、L^AT_EX 文書中に挿入する化学構造式を生成するパッケージである chemobabel.sty と同梱リソースの使い方を説明します。構造式は Open Babel と Inkscape の機能を利用することにより、MDL Molfile, ChemDraw ファイル、さらには SMILES 表記法を含むさまざまな化学データ形式から生成することができます。説明は英語と日本語の両方で書かれています。以下の構造式はこの方法によって出力されたものです。

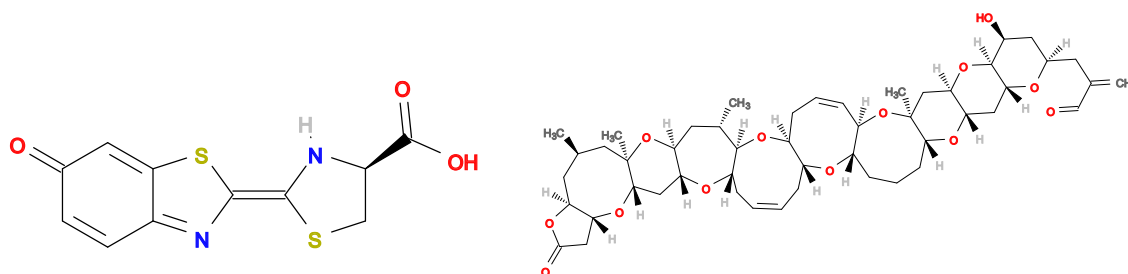
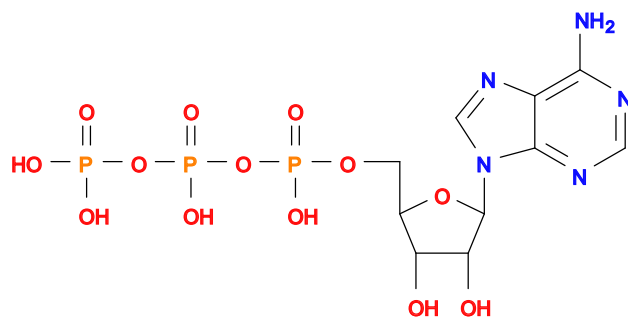


Fig. 1: Firefly luciferin & Brevetoxin A (from MDL Molfiles)

draw/ATP.cdx



draw/Glucose.cdx

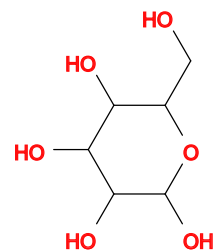


Fig. 2: ATP (Adenosine triphosphate) & Glucose (from ChemDraw files)

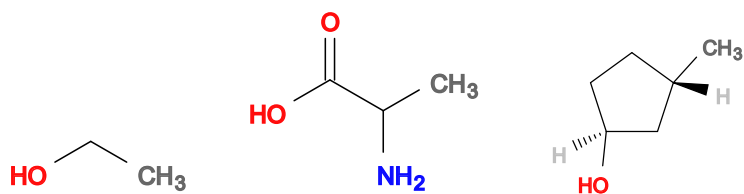


Fig. 3: Ethanol, Alanine & (1S,3S)-3-Methylcyclopentanol (from SMILES notations)

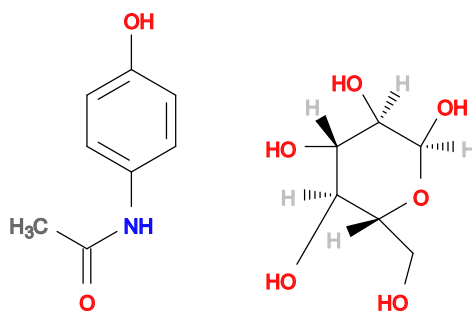


Fig. 4: Acetaminophen (Paracetamol) & α-D-Glucose (from SMILES notations)

Contents

1	Introduction (はじめに)	1
1.1	Motivation for Development (開発の動機)	1
1.1.1	English	1
1.1.2	日本語	2
1.2	Idea and Approach (着想とアプローチ)	3
1.2.1	English	3
1.2.2	日本語	3
2	Before you begin (使う前に)	4
2.1	English	4
2.2	日本語	4
3	Basic Usage (基本的な使い方)	5
3.1	At the beginning... (はじめに)	5
3.1.1	English	5
3.1.2	日本語	8
4	For Safer Typesetting... (安全なタイプセットのために)	11
4.1	Macro for Auto-extracting commands (コマンド自動抽出マクロ)	11
4.1.1	English	11
4.1.2	日本語	12
4.2	Note for Compatibility (互換性に関する注意)	13
4.2.1	English	13
4.2.2	日本語	14
5	Options for Depiction (描画オプション)	15
5.1	English	15
5.2	日本語	15
6	More complex structural formulas (より複雑な構造式)	18
6.1	English	18
6.2	日本語	21
7	Limitations and Alternatives (この方法の限界と代替案)	22
7.1	English	22
7.2	日本語	22

8	Technical information (技術情報)	23
9	Version History	24

1 Introduction (はじめに)

1.1 Motivation for Development (開発の動機)

1.1.1 English

As you already know, \LaTeX is being used all over the world. However, when it comes to drawing chemical structural formulas, ways of inserting formulas into \LaTeX documents are very limited. I think it is mainly due to lack of both reliable and simple methods of doing this that many of the researchers in chemistry are reluctant to use \LaTeX system.

Some \TeX / \LaTeX macros and packages are already available from CTAN:

- $\text{\X}\text{\LaTeX}$: a set of packages for drawing chemical structural formulas
- `chemfig`: a package which draws molecules using `TikZ`

These packages are reliable and almost all kinds of structural formulas can be drawn in a consistent way. However, a lot of practice will be required before one can come to make full use of them.

You can also insert structural formulas with `\includegraphics` of PDF or EPS files exported from special softwares for chemists, such as ChemDraw and other similar programs. In this case, you have to save files in both chemical formats (such as `.cdx` or `.mol`) and graphical formats (`.pdf` or `.eps`) manually.

This new package, `chemobabel.sty`, will offer a new choice for chemists who need chemical structures inserted in their \LaTeX documents. In this method, we use Open Babel for generating chemical structural formulas in SVG format, and Inkscape for converting SVG to PDF. Both programs are open source and cross-platform, and being actively maintained.

About Open Babel:

Open Babel is a toolbox specially designed to handle many kinds of chemical data. It is an open, collaborative project allowing anyone to search, convert, analyze, or store data from molecular modeling, chemistry, solid-state materials, biochemistry, or related areas. See Official Website for detail.

1.1.2 日本語

ご存じのとおり、 \LaTeX は世界中で使われています。しかし、化学構造式を描くという点で考えると、 \LaTeX 文書中に構造式を挿入する手段は非常に限られています。この目的を達成する確実かつ容易な方法が存在しないことは、化学分野における多くの研究者が \LaTeX システムを使わない要因になっていると考えられます。

すでに CTAN からいくつかのパッケージが利用可能です：

- \XeLaTeX : 化学構造式を描くためのパッケージ集
- `chemfig`: `TikZ` を利用して構造式を描画するパッケージ

これらのパッケージは確実で、あらゆる構造式を一貫性をもって描画することができます。しかし、十分にその機能を活用できるようになるには相当の熟練を要します。

化学者向けの専用ツール (ChemDraw など) で PDF や EPS 形式で出力し、`\includegraphics` によって構造式を取り込むという手段もあります。この場合、化学用のフォーマット (`.cdx` や `.mol` など) と画像フォーマット (`.pdf` や `.eps` など) の両方で保存する必要があります。

新しいパッケージ `chemobabel.sty` は、 \LaTeX 文書中に化学構造式を挿入する必要がある化学者に新たな選択肢を提供することでしょう。この方法では、Open Babel によって SVG 形式の構造式画像を生成し、さらに Inkscape によって SVG から PDF に変換します。どちらのプログラムもオープンソースかつクロスプラットフォームであり、盛んに開発が行われています。

Open Babel について

Open Babel はさまざまな化学データを扱うために特別に設計されたツールです。オープンな共同プロジェクトで、分子モデリング・化学・固体物性・生化学や関連分野のデータを誰もが検索・変換・分析・保存できることを目指しています。詳細は公式サイトを参照してください。

1.2 Idea and Approach (着想とアプローチ)

1.2.1 English

First I found a post on Noel O’Blog [1], [2] and the comment [3] about generating chemical structural formulas from SMILES notations using Open Babel. I thought the method described in these posts were very interesting, but I didn’t like all the programs could be executed by `-shell-escape` option. At the same time, I found a macro for copying all `\includegraphics` from one source file to another [4], which made me conceive of an idea of extracting minimal source code which needs `-shell-escape` option.

I thought this method would be applicable to converting ChemDraw files into graphics, but I could not find any project attempting to realize this idea. I found a project named Chemdraw in L^AT_EX on sourceforge, but now it was closed. This drove me to develop a new package to simplify the process for inserting chemical structural formulas into L^AT_EX documents.

1.2.2 日本語

最初に Open Babel を用いて SMILES 表記から化学構造式を生成するという Noel O’Blog の記事 [1], [2] とコメント [3] を見つけました。これらの投稿に示された手法は非常におもしろいと思いましたが、すべてのプログラムが `-shell-escape` オプションによって実行されてしまうことがあまり好きではありませんでした。それと同じころ、あるソース中の `\includegraphics` を別のソースに書き出すというマクロを発見 [4] し、このとき私は `-shell-escape` が必要な最小のコードだけを抽出するという着想を得ました。

この方法は ChemDraw ファイルを画像に変換する目的にも応用できると考えたのですが、この発想を実現したプロジェクトが見当たらず、ただ一つ見つけた Sourceforge の Chemdraw in L^AT_EX はすでに終了していました。そこで、化学構造式を L^AT_EX 文書中に挿入する手順を簡略化するパッケージを開発しようと考えました。

2 Before you begin (使う前に)

2.1 English

First, you have to install Open Babel and Inkscape on your computer, and export PATH to the command-line binary of both programs, obabel and inkscape. You can confirm by executing following commands:

In UNIX operating system:

```
$ which obabel
```

```
$ which inkscape
```

In Windows operating system:

```
> where obabel
```

```
> where inkscape
```

And when you get some command echo, installation should be successful.

2.2 日本語

初めにコンピュータに Open Babel と Inkscape をインストールし、コマンドライン実行ファイルである obabel と inkscape に PATH を通します。正しく PATH が通っているかどうかは以下のコマンドによって確認できます：

UNIX 系の場合：

```
$ which obabel
```

```
$ which inkscape
```

Windows の場合：

```
> where obabel
```

```
> where inkscape
```

何らかのディレクトリ名が返ってきた場合は、正しくインストールできているはずです。

3 Basic Usage (基本的な使い方)

3.1 At the beginning... (はじめに)

3.1.1 English

In the preamble of your document, declare

```
\usepackage{graphicx} % for platex (NOT pdflatex), dvipdfmx option needed
\usepackage{chemobabel}
```

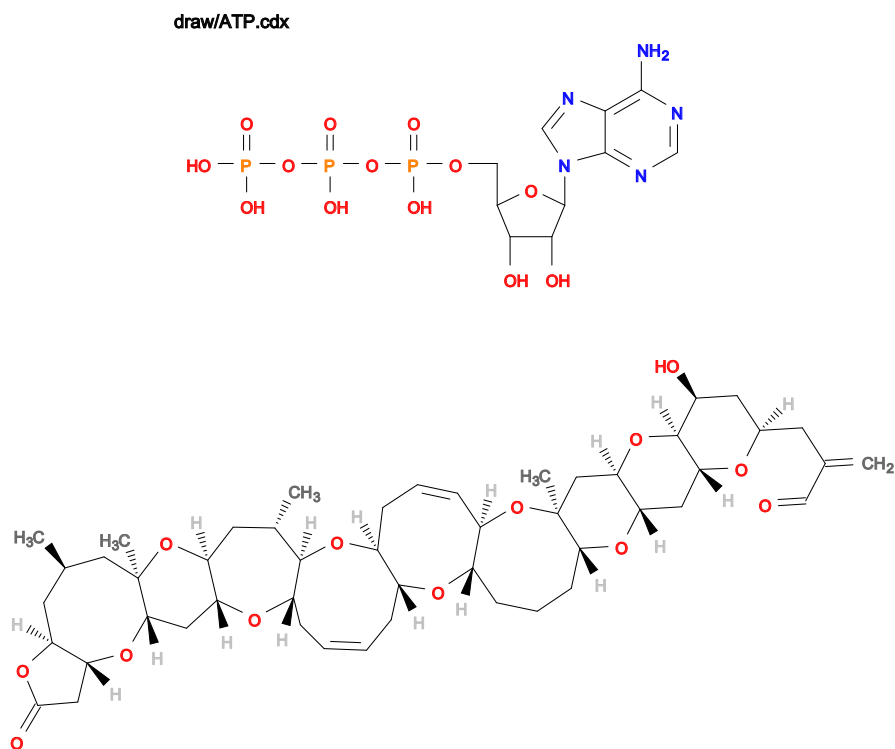
to load chemobabel package. Use `\chemobabel` command as follows^{*1}:

```
\chemobabel[scale=0.4]{draw/ATP.cdx}{}
\chemobabel[scale=0.4]{draw/Brevetoxin A.mol}{} 
```

and typeset with `-shell-escape` option:

```
$ pdflatex -shell-escape test.tex
```

You will see in `chemobabelimgdir` subdirectory “chemobabelimg[*NUM*].pdf”, and output file `test.pdf` will be:



^{*1} I put original files in a subdirectory `draw`, so added `draw/` before the actual filename. Of course you can put in the same directory with your `LATEX` file.

ATP.cdx is the file with chemical structural formula of ATP (Adenosine triphosphate) drawn in ChemDraw, and Brevetoxin A.mol is the file which can be obtained from ChemSpider (originally 9041092.mol).

Remember to use `-shell-escape` option, or it won't allow L^AT_EX to call any command-line programs, such as obabel and inkscape!

The syntax is:

`\chemobabel[options]{filename}{obabel options}`

As parameters, *filename* should be placed in the first braces, and *obabel options* (see Section 5 for some examples) can be added in the second braces. You can leave the second parameter empty.

You can give *options* to be passed to `\includegraphics` command as optional parameters in the box brackets. This option is not necessary, so can be omitted. In this case the default option “scale=1” is passed to `\includegraphics` command.

You can also write SMILES notations directly in your L^AT_EX file to be converted into structural formulas. Use `\smilesobabel` command as follows:

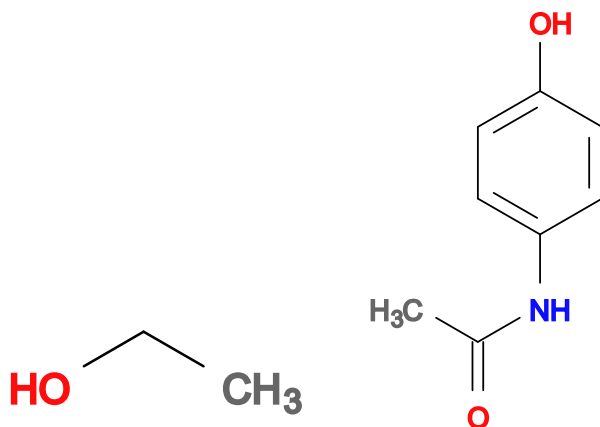
```
\smilesobabel{CCO}{}
```

```
\smilesobabel[scale=0.7]{CC(=O)Nc1ccc(cc1)O}{}
```

and typeset with `-shell-escape` option:

```
$ pdflatex -shell-escape test.tex
```

This time you will get “smilesobabelimg[*NUM*].pdf” and “test.pdf”:



CCO is the SMILES notation of ethanol, and CC(=O)Nc1ccc(cc1)O is that of acetaminophen (paracetamol).

The syntax is:

```
\smilesobabel[options]{SMILES notation}{obabel options}
```

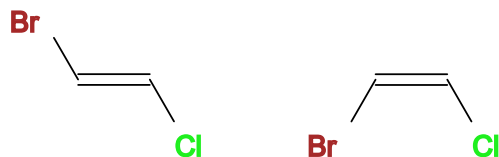
The only difference from `\chemobabel` is that *SMILES notation* should be placed in the first braces.

Important: You can put any kinds of SMILES notations as a parameter of `smilesobabel`. However, there are some cases when you should pay another attention.

In the SMILES syntax, a backslash (`\`) and a percent symbol (`%`) sometimes appear. As you know however, these characters have a special meaning in \LaTeX . To solve this problem, you have to change category codes of these characters temporarily and set an alternative. I recommend a tilde (`~`) to work as a escape character in replace of a backslash `\`, because a tilde does not appear in SMILES syntax (and there is a support in the method described in Section 4 for this reason).

You can do this as follows (of course, you can also apply this method to notations without such characters!):

```
\begingroup
\catcode'\~ = 0
\catcode'\% = 11
\catcode'\ = 11
~smilesobabel[scale=0.7]{Cl/C=C/Br}{~}
~smilesobabel[scale=0.7]{Cl/C=C\Br}{~}
~endgroup
```



In short, a `~` becomes an escape character, and `%` and `\` become normal letters similar to alphabets inside the group between `\begingroup` and `~endgroup` (NOT `\endgroup`, because at this point `\` is not a escape character and `~` is). This change is no longer effective outside this group. An alternative for the comment character `%` will be almost unnecessary in such a short term (see Section 8 for detail).

3.1.2 日本語

プリアンブルに

```
\usepackage[dvipdfmx]{graphicx} % platex + dvipdfmx の場合
```

```
\usepackage{chemobabel}
```

と宣言して chemobabel パッケージを読み込みます。\\chemobabel コマンドは

```
\chemobabel[scale=0.4]{draw/ATP.cdx}{}
```

```
\chemobabel[scale=0.4]{draw/Brevetoxin A.mol}{}
```

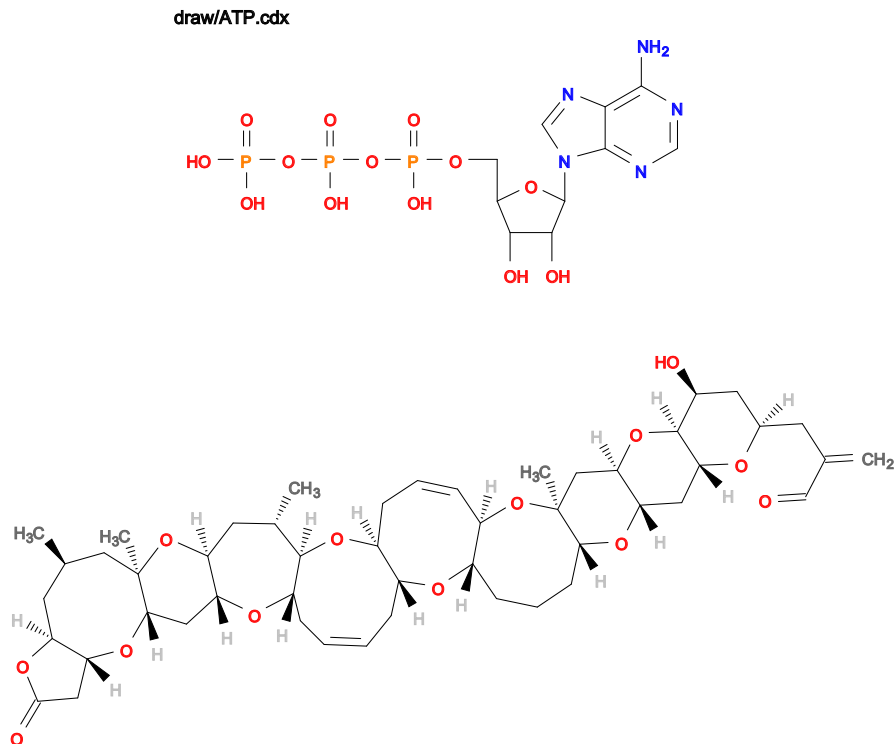
のようにして使います*2。これを -shell-escape オプションを付けてタイプセットします。

```
$ platex -shell-escape test.tex
```

すると chemobabelingdir 以下に「chemobabeling[NUM].pdf」というファイルが生成します。得られた test.dvi に対して

```
$ dvipdfmx test.dvi
```

と実行すると、出力される「test.pdf」は以下のようになります：



*2 ここでは元のファイルを **draw** というサブディレクトリに置いたので、ファイル名の前に **draw/** を付けました。もちろん L^AT_EX ソースと同一ディレクトリに置くこともできます。

ATP.cdx は ATP (アデノシン三リン酸) の構造式を ChemDraw で描画したファイル、Brevetoxin A.mol は ChemSpider から入手できるファイルです (もとのファイル名は 9041092.mol となっています)。

タイプセット時に `-shell-escape` オプションを与えるのを忘れないようにしてください。このオプションを与えることで \LaTeX が外部プログラムである obabel や inkscape を呼んで実行することができます。

構文は以下のとおりです。

`\chemobabel[options]{filename}{obabel options}`

引数として、中括弧の 1 つめにファイル名を与え、2 つめに obabel に送るオプション (例は第 5 章を参照) を指定してください。2 つめの引数は空にすることもできます。

また、オプションとして角括弧に `\includegraphics` に渡すオプションを指定することができます。このオプションは省略可能で、何も指定しない場合は `scale=1` が適用されます。

\LaTeX ソースファイル中に直接 SMILES 表記を書き込んで、構造式に変換することもできます。`\smilesobabel` コマンドを以下のように使います：

`\smilesobabel{CCO}{}`

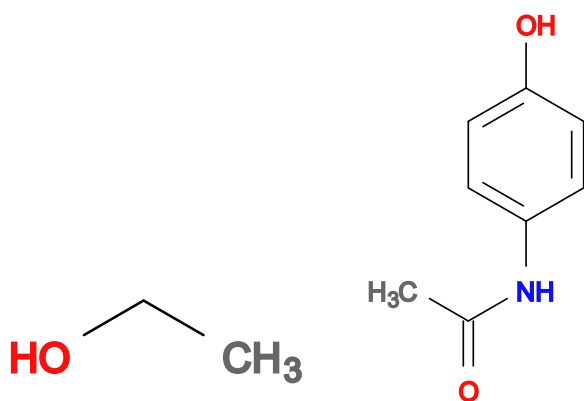
`\smilesobabel[scale=0.65]{CC(=O)Nc1ccc(cc1)O}{}`

これを `-shell-escape` オプションを付けてタイプセットします。

`$ platex -shell-escape test.tex`

`$ dvipdfmx test.dvi`

すると今度は「smilesobabelimg[NUM].pdf」のようなファイルが生成し、出力される「test.pdf」は以下ようになります：



`CCO` はエタノールの SMILES 表記、`CC(=O)Nc1ccc(cc1)O` はアセトアミノフェンの SMILES 表記です。

構文は以下のとおりです。

```
\smilesobabel[options]{SMILES notation}{obabel options}
```

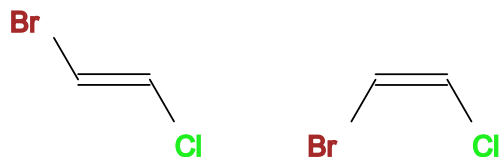
`\chemobabel` との唯一の違いは、中括弧の 1 つめに SMILES 表記法のテキストを与えることだけです。

重要： `smilesobabel` の引数としては任意の SMILES 表記を用いることができます。ただし、特別な注意を払う必要が生じる場合があります。

SMILES の文法では、バックスラッシュ (`\`) やパーセント記号 (`%`) が時々現れます。しかしご存じのとおり、これらの文字は \LaTeX では特別な意味を持っています。この問題を回避するには、一時的にこれらの文字のカテゴリコードを変更し、代わりとなる文字を設定する必要があります。作者としては、チルダ (`~`) をバックスラッシュ `\` の代わりにエスケープ文字として用いることを推奨します。というのは、チルダが SMILES の文法で現れないためです（さらに、この理由から第 4 章で説明する方法でもサポートしています）。

これは実際には以下のようにして行います（もちろん、この方法は特殊文字を含まない場合にも適用できます）。

```
\begingroup
  \catcode'\~ = 0
  \catcode'\% = 11
  \catcode'\ = 11
  ~smilesobabel[scale=0.7]{Cl/C=C/Br}{}
  ~smilesobabel[scale=0.7]{Cl/C=C\Br}{}
~endgroup
```



簡単に説明すれば、チルダ `~` がエスケープ文字となり、パーセント `%` とバックスラッシュ `\` がアルファベットと同じ通常の文字になります。ただし、この変更は `\begingroup` と `~endgroup` (`\endgroup` ではない；この時点では `\` はまだエスケープ文字ではなく、`~` が該当するから) の間のグループ内でだけ有効で、グループの外では無効になります。コメント文字の `%` の代わりとなる文字は設定しなくても、このような短い間では不要であることがほとんどであると思われます（詳細は第 8 章をみてください）。

4 For Safer Typesetting... (安全なタイプセットのために)

4.1 Macro for Auto-extracting commands (コマンド自動抽出マクロ)

4.1.1 English

With the basic method described in Section 3, you will get desired output in almost any situation. However, it is sometimes NOT desirable to use `-shell-escape` option, because with that option \LaTeX can call ANY external command-line programs whether you like it or not. This means that there is a chance for unknown programs to be executed, and this can be dangerous especially in case that you don't write your own \LaTeX code. To avoid this problem, I wrote a macro for extracting all `\chemobabel` and `\smilesobabel` commands to another \LaTeX file (Reference: \TeX Forum [4]).

All you have to do is to add

```
\input{chemobabel-extract.tex}
```

at the end of the preamble (or at least after loading `chemobabel` package) in your document. If you are using `chemobabel.sty` version 0.5 or later, you can simplify this process by giving `extract` option when loading the package^{*3}:

```
\usepackage[extract]{chemobabel}
```

When typesetting, usual command

```
$ pdflatex test.tex
```

can be used. Of course you will see some error messages in your document, but you will get “ChemFigFile.tex” in the same directory, a minimal \LaTeX source file which includes all `\chemobabel` and `\smilesobabel` commands written in the original \LaTeX file. You can simply typeset it by

```
$ pdflatex -shell-escape ChemFigFile.tex
```

and you will get PDF figures in the same way as described in Section 3. After this, typeset your own document again:

```
$ pdflatex test.tex
```

This time you will get desired output “test.pdf” which includes figures properly. This method will also save you a lot of time for typesetting.

^{*3} Notice: You must NOT use these two methods at the same time. It will execute an endless loop!

4.1.2 日本語

第 3 章で述べた基本的な使い方で、ほとんどの場合問題ないでしょう。しかし、`-shell-escape` オプションを付けてタイプセットするため、好むと好まざるとにかかわらず \LaTeX がどんな外部コマンドでも実行できてしまいます。つまり、知らないうちにプログラムが実行されてしまう可能性を示唆しており、これはとりわけ自分で書いた \LaTeX ソースでない場合には危険を伴います。この問題を回避するために、`\chemobabel` と `\smilesobabel` コマンドだけを別の \LaTeX ファイルに抽出するマクロを作成しました（参考： \TeX Forum [4]）。

プリアンブルの末尾（あるいは少なくとも `chemobabel` パッケージを読み込んだ後）に

```
\input{chemobabel-extract.tex}
```

と書き加えるだけで十分です。この操作は `chemobabel.sty` の version 0.5 以降を使用している場合、単にパッケージを読み込む際に `extract` オプションを付与することでも簡単に代用可能です*4：

```
\usepackage[extract]{chemobabel}
```

タイプセットの際には通常の

```
$ platex test.tex
```

というコマンドを用いてください。当然ながらこの出力結果にはエラーメッセージが表示されますが、同時に同じディレクトリに「ChemFigFile.tex」というファイルが生成します。これは元のソースに書かれていた `\chemobabel` と `\smilesobabel` コマンドをすべて含む最小の \LaTeX ソースで、これは直接

```
$ pdflatex -shell-escape ChemFigFile.tex
```

とタイプセットできます*5。すると、第 3 章同様に PDF 形式の図のファイルが生成しますので、あとは再度

```
$ platex test.tex
```

```
$ dvipdfmx test.dvi
```

と実行すれば、適切に図を取り込んだ望みどおりの「test.pdf」が得られます。この方法を用いれば、タイプセットに要する時間を大幅に短縮することも期待できます。

*4 ただし、くれぐれも 2 つの方法を同時に書き込んではいけません。無限ループになってしまい危険です。

*5 欧文と同じ仕様に統一したため、「ChemFigFile.tex」では `graphicx` パッケージに `dvipdfmx` オプションを与えずに読み込んでいます。したがって、このタイプセットだけは `pdflatex` を用いてください。

4.2 Note for Compatibility (互換性に関する注意)

4.2.1 English

In order to keep compatibility, in “chemobabel-extract-compatible.tex” I reserved macros for extracting `\smiles` and `\obabel` at the same time. If you have a L^AT_EX source file which has `\smiles` and/or `\obabel` in it, you can use this by adding

```
\input{chemobabel-extract-compatible.tex}
```

instead of “chemobabel-extract.tex”. When you use `\smilesobabel` command only, you should input “chemobabel-extract.tex” as described above.

`\smiles` and `obabel` are defined as:

```
% ----- for \smiles command -----
\newcounter{smilescounter}
\setcounter{smilescounter}{1}
\newcommand{\smiles}[1]{
  \immediate\write18{obabel -:"#1" -O smilesimg\arabic{smilescounter}.png}
  \includegraphics{smilesimg\arabic{smilescounter}.png}
  \addtocounter{smilescounter}{1}
}
% ----- for \obabel command -----
\usepackage{graphvizObabel}
%% -----
```

`\smiles` is a macro written by Noel O’Boyle [1] which embeds PNG graphics, and `\obabel` depends on `graphvizObabel.sty` by Jakob Lykke Andersen’s [3]. When these two definitions are added in your preamble section, these two commands can be used as follows:

```
\smiles{CCO}
\obabel[scale=0.6]{CCO}
```

However, `\smiles` command creates raster graphics and has no options to pass to `\includegraphics`, and `\obabel` command sometimes does not work because of lack of file extension (.pdf). In addition, when you leave optional parameter of `\obabel` empty, you would get an error like:

```
! Package keyval Error:  undefined.
```

This is not so critical because `\includegraphics` can handle with empty options, so you can typeset successfully. However, when using my “chemobabel-extract-compatible.tex” macro this can cause a fatal problem:

```
! Use of \obabel doesn't match its definition.
```

These problems are solved in `\smilesobabel`, so `\smilesobabel` command recommended.

4.2.2 日本語

互換性を維持するため、「chemobabel-extract-compatible.tex」には `\smiles` と `\obabel` も同時に抽出するマクロを残してあります。手元に `\smiles` や `\obabel` を含む \LaTeX ソースがある場合は、「chemobabel-extract.tex」の代わりに

```
\input{chemobabel-extract-compatible.tex}
```

を使うことができます。仮に `\smilesobabel` だけを使うのであれば「chemobabel-extract.tex」を使うべきです。

`\smiles` と `\obabel` の定義は先ほどの英語版に示してありますので、そちらを参照してください。

`\smiles` は Noel O’Blog [1] の修正版であり、`\obabel` は Jakob Lykke Andersen’s [3] の `graphvizObabel.sty` に依存します。これらの定義を任意の \LaTeX ソースのプリアンブルに追加すれば、これら 2 つのコマンドを以下のように用いることができます：

```
\smiles{CCO}
```

```
\obabel[scale=0.6]{CCO}
```

ただし、`\smiles` はラスター画像を生成してかつ `\includegraphics` に渡すオプションを扱えず、`\obabel` は場合によっては拡張子 (.pdf) が欠落しているためにうまくはたきません。さらに、仮に `\obabel` でオプションを空にした場合

```
! Package keyval Error: undefined.
```

というエラーが出ます。これを無視してタイプセットすると `\includegraphics` はオプションを空のままと解釈できるため、実質的には単なる警告レベルでタイプセットは可能でした。しかし、自動抽出マクロを実行した場合にはこの問題が原因となって

```
! Use of \obabel doesn't match its definition.
```

というエラーを誘発していました。これらの問題は `\smilesobabel` では解決されていますので、特別な理由がない限り `\smilesobabel` コマンドを推奨します。

5 Options for Depiction (描画オプション)

5.1 English

As I have already mentioned, in both `\chemobabel` and `\smilesobabel`, you can give some *obabel options* in the second braces. Here I will introduce some examples.

- `\smilesobabel[scale=0.6]{CC(=O)Nc1ccc(cc1)O}{-xa}`

executes a command which gives a formula with all carbon atoms shown:

```
$ obabel -:"CC(=O)Nc1ccc(cc1)O" -O smilesobabelimg[NUM].svg -xa
```

- `\smilesobabel[scale=0.6]{CC(=O)Nc1ccc(cc1)O}{-xu -xC}`

gives one without element-specific atom coloring or terminal carbon atoms:

```
$ obabel -:"CC(=O)Nc1ccc(cc1)O" -O smilesobabelimg[NUM].svg -xu -xC
```

- `\chemobabel[scale=0.6]{ATP.cdx}{-xd}`

You can add `-xd` option to remove filenames from ChemDraw figures.

```
$ obabel -:"ATP.cdx" -O chemobabelimg[NUM].svg -xd
```

5.2 日本語

既に述べたとおり、`\chemobabel` と `\smilesobabel` の両方について、2 つめの中括弧に *obabel options* を与えることができます。ここで、いくつかの有用な例を挙げておきます。

- `\smilesobabel[scale=0.6]{CC(=O)Nc1ccc(cc1)O}{-xa}`

構造式中の全ての炭素原子を省略せずに描画するコマンドを実行します：

```
$ obabel -:"CC(=O)Nc1ccc(cc1)O" -O smilesobabelimg[NUM].svg -xa
```

- `\smilesobabel[scale=0.6]{CC(=O)Nc1ccc(cc1)O}{-xu -xC}`

元素の着色を行わず、末端の炭素原子を明示しない構造式を出力します：

```
$ obabel -:"CC(=O)Nc1ccc(cc1)O" -O smilesobabelimg[NUM].svg -xu -xC
```

- `\chemobabel[scale=0.6]{ATP.cdx}{-xd}`

ChemDraw の図からファイル名を取り除くには、`-xd` オプションを付けます：

```
$ obabel -:"ATP.cdx" -O chemobabelimg[NUM].svg -xd
```

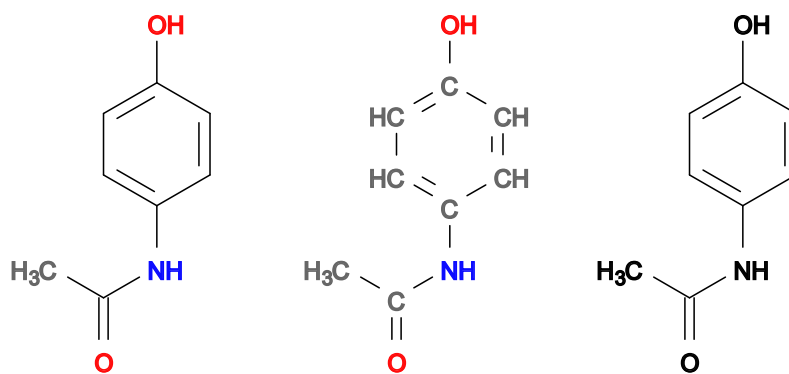


Fig. 5: Acetaminophen (Paracetamol): Nothing, -xa, -xu

SMILES: CC(=O)Nc1ccc(cc1)O

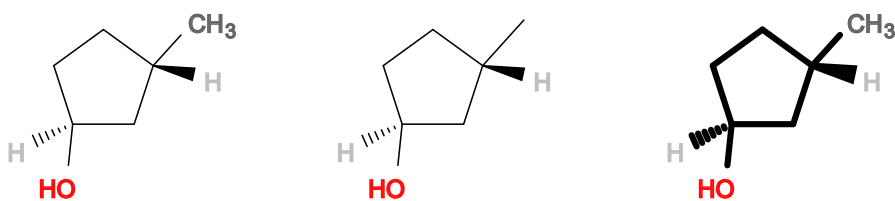


Fig. 6: (1*S*,3*S*)-3-Methylcyclopentanol: Nothing, -xC, -xt

SMILES: C1[C@H](C)C[C@H](O)C1

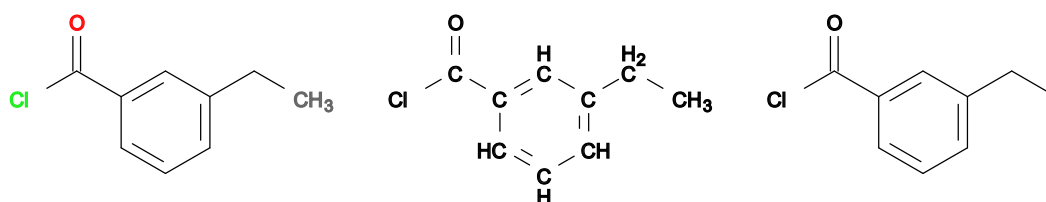


Fig. 7: 3-Ethylbenzoyl chloride: Nothing, -xa -xu, -xu -xC

SMILES: CCc1cccc(c1)C(=O)Cl

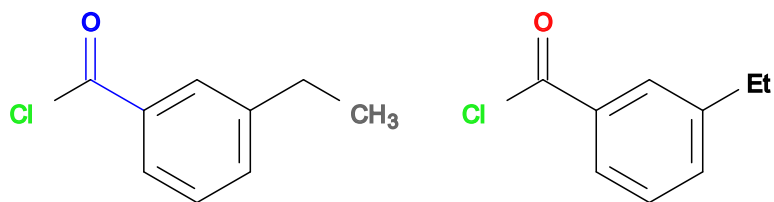


Fig. 8: 3-Ethylbenzoyl chloride: --highlight "cC=O blue", -xA --genalias

SMILES: CCc1cccc(c1)C(=O)Cl

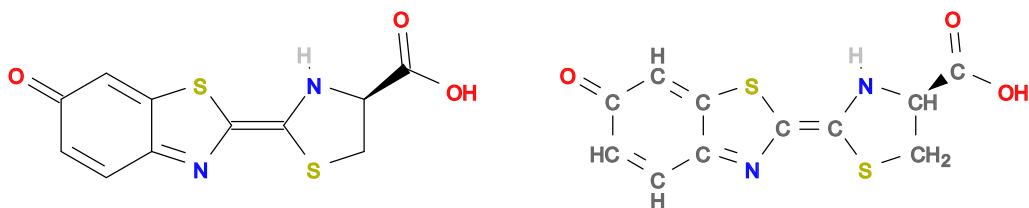


Fig. 9: Firefly luciferin (from MDL Molfiles): Nothing, -xa

draw/ATP.cdx

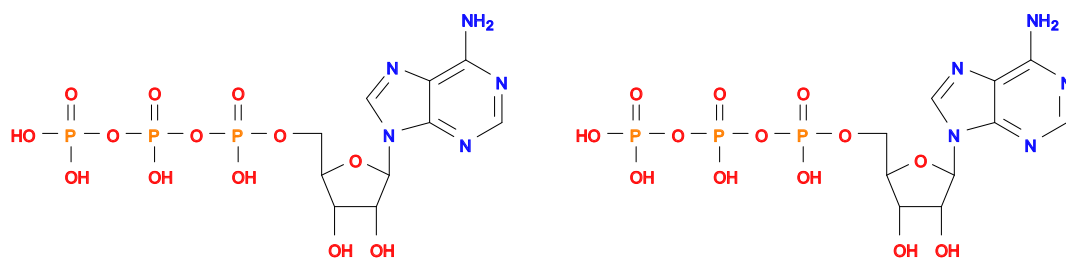


Fig. 10: ATP (from ChemDraw files): Nothing, -xd

For more details, see obabel and babel and SVG depiction (svg) in Open Babel documentation.

より詳細には Open Babel documentation の obabel and babel または SVG depiction (svg) を参照してください。

6 More complex structural formulas (より複雑な構造式)

6.1 English

Open Babel can generate more complex structural formulas from SMILES notations.

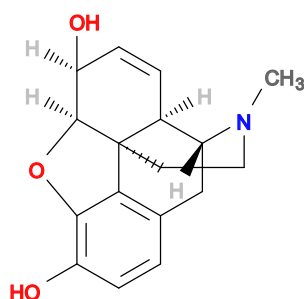


Fig. 11: (-)-Morphine

SMILES: CN1CC[C@]23c4c5ccc(c4O[C@H]2[C@H](C=C[C@H]3[C@H]1C5)O)O

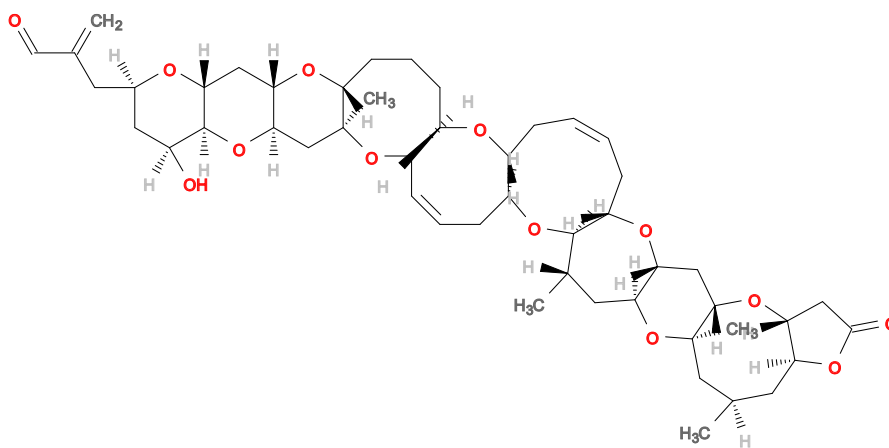


Fig. 12: Brevetoxin A

SMILES (Sorry but too long):

C[C@@H]1C[C@H]2[C@@H](CC(=O)O2)O[C@H]3C[C@@H]4[C@H](C[C@@H]([C@@H]5[C@@H](O4)C/C=C\C

However, the results are sometimes undesirable and may also be different depending on version numbers. Here are some examples:

Cephalostatin-1 (Example from Wikipedia):

C[C@@](C)(O1)C[C@@H](O)[C@@]1(O2)[C@@H](C)[C@@H]3CC=C4[C@]3(C2)C(=O)C[C@H]5[C@H]4CC[

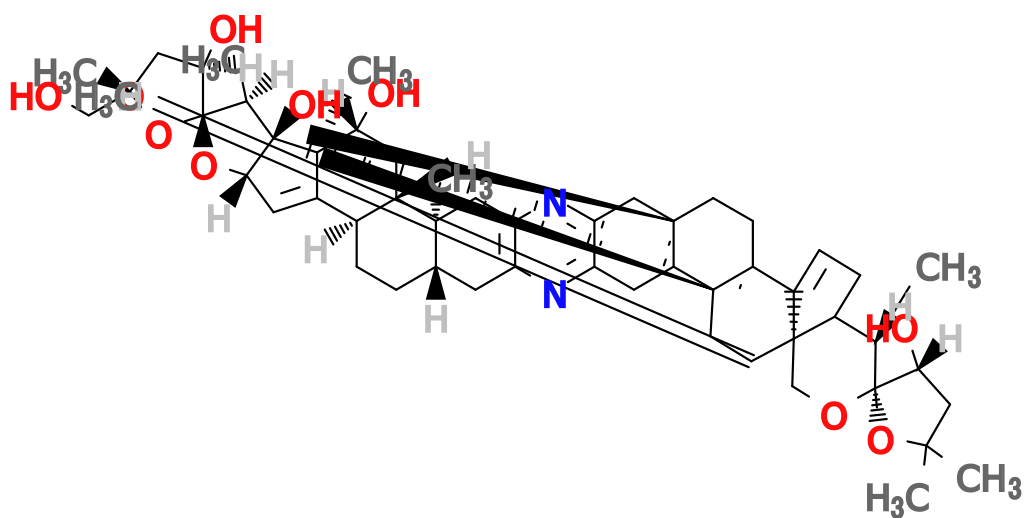


Fig. 13: Cephalostatin-1 (figure from Open Babel for Mac 2.3.1)

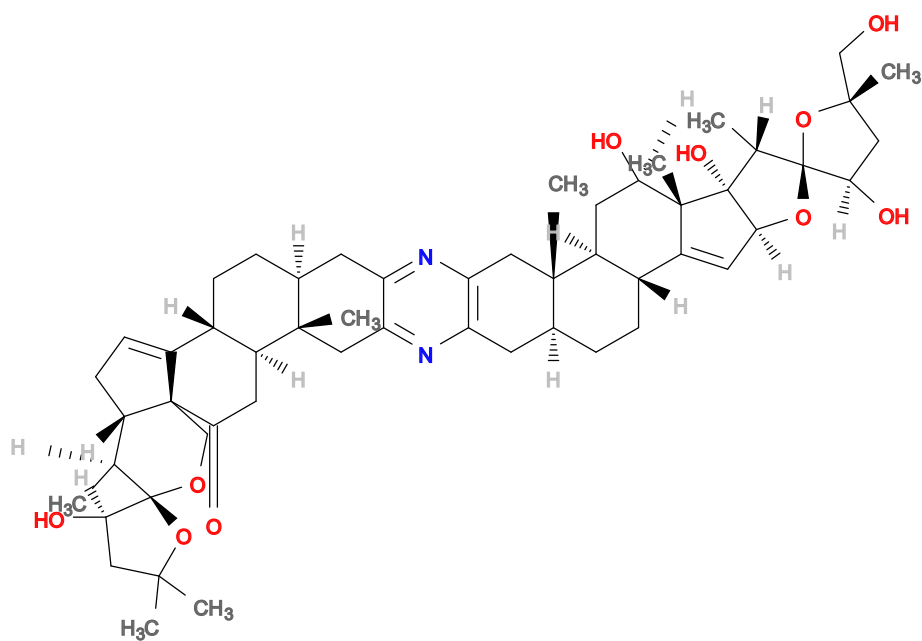


Fig. 14: Cephalostatin-1 (figure from Open Babel for Win 2.3.2)

Sesamin: c1cc2c(cc1C3C4COC(C4C03)c5ccc6c(c5)OC06)OC02

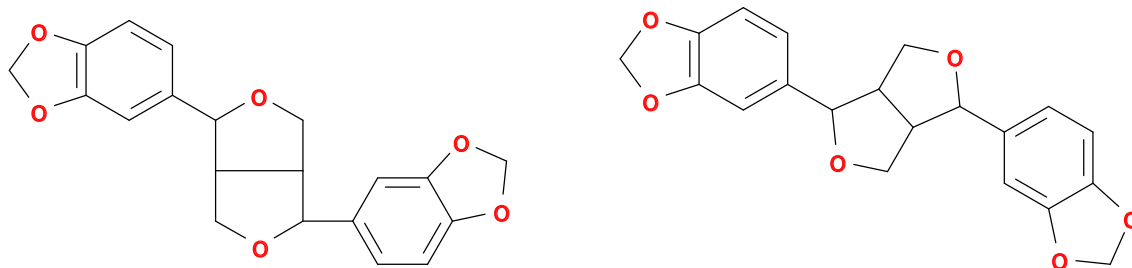


Fig. 15: Sesamin (figures from Open Babel for Mac 2.3.1; Normal and `--gen2d`)

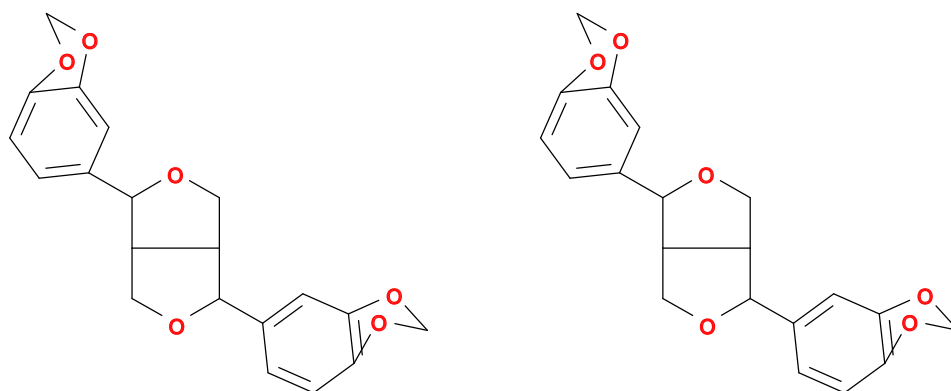


Fig. 16: Sesamin (figures from Open Babel for Win 2.3.2; Normal and `--gen2d`)

Also, Open Babel can generate wrong structures from SMILES notations. Here we can see the formula generated from a SMILES notation lacks one double bond.

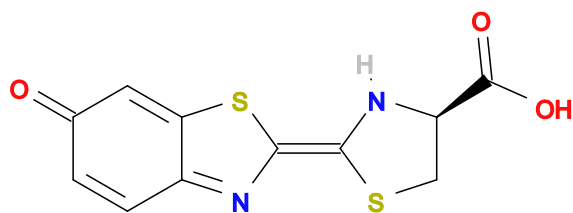


Fig.17: Firefly luciferin: exact structure from ChemSpider ID 4588411

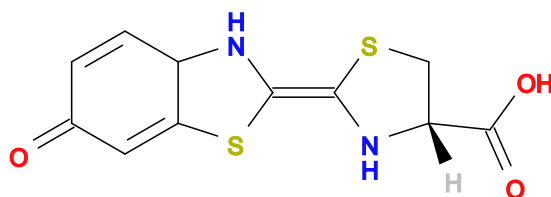


Fig.18: Firefly luciferin?: output from Open Babel for Win 2.3.2

SMILES: C1[C@@H](N/C(=c\2\nc3c(=CC(=O)C=C3)s2)/S1)C(=O)O

We can avoid all these problems by using `\chemobabel` with `.mol` or `.cdx` files, instead of `\smilesobabel`. However, the fact that many complex structures can be generated from only one character string is very interesting, isn't it?

6.2 日本語

図はすべて英語版を参照してください。

Open Babel は SMILES 表記から複雑な構造式を生成することができます。

しかし、時には結果が望ましくないことがあり、また使用するバージョン番号によって出力が異なることがあります。いくつかの例を示します：

また、Open Babel は SMILES 表記から誤った構造式を生成することもあります。ホタルルシフェリンの図で、SMILES 表記から生成した構造式には二重結合が一つ欠けています。

このような問題点は全て `.mol` または `.cdx` を用意して `\chemobabel` を利用することで回避できます。とはいえ、ただの文字列から複雑な構造式を生成しうること自体、おもしろいとは思いませんか？

7 Limitations and Alternatives (この方法の限界と代替案)

7.1 English

This method relies on Open Babel for generating structural formulas, even without drawing anything by yourself. Of course you can modify and customize these structures to some extent, but this method will not be suitable for fine control of the output. Also, computer-generated formulas may sometimes be unnatural and undesirable. Therefore this method will be useful only when you need a simple method for inserting structural formulas with minimal modification. If you are not satisfied with the output, consider using \LaTeX or chemfig packages instead.

I also found following packages which can be used for similar purpose:

- mol2chemfig: convert chemical structures from MDL molfile format to chemfig source code

7.2 日本語

この方法は Open Babel に依存して構造式を生成しており、一切自分で構造式を描画することなく済ませることさえ可能です。もちろんある程度はそれらの構造を修正したりカスタマイズしたりできますが、この方法は出力の緻密な調節には不向きでしょう。また、コンピュータによって生成される構造式は時に見た目が不自然で、好ましくない場合もあるかもしれません。したがって、この方法は構造式の出力を完全に制御したいという意図がなく、構造式を簡便に挿入したい場合にのみ。出力に満足できない場合は、 \LaTeX や chemfig のようなパッケージを代わりに使うことをご検討ください。

同様の目的を達成するために使えそうなパッケージとして、以下のようなものもあります：

- mol2chemfig: convert chemical structures from MDL molfile format to chemfig source code

8 Technical information (技術情報)

I read online SMILES Tutorial, and checked what kinds of characters are used in SMILES syntax.

- Roman alphabets: A-Z, a-z
- Numbers: 1-10
- Brackets: [] ()
- Others:
 - * (unspecified atomic number)
 - . (disconnection)
 - + (charge sign)
 - - (single bond or charge sign)
 - = (double bond)
 - # (triple bond)
 - \$ (quadruple bond)
 - : (aromatic bond)
 - % (used when more than 10 ring closures)
 - /, \ (configuration around double bonds)
 - @ (tetrahedral chirality)
 - > (reaction)

It is important to prevent these characters from being translated as L^AT_EX control sequence. However, a backslash (\) is a default escape character in L^AT_EX, and a percent symbol (%) also has a special meaning as a comment character. To solve this problem, I chose a tilde (~) to work as an escape character in replace of a backslash by changing category codes.

本パッケージを制作するにあたり、オンラインの SMILES Tutorial を読んで SMILES 表記法に用いられる文字の種類を調べました (一覧は英語部分参照)。

これらの文字が L^AT_EX におけるコントロールシーケンスとして解釈されることを防ぐ必要がありますが、バックスラッシュ \ はデフォルトのエスケープ文字であり、またパーセント記号 % もコメント文字としての特殊な意味を持っています。この問題を回避するため、今回カテゴリーコードを変更することで「チルダ ~」をバックスラッシュの代わりにエスケープ文字として用いるようにしました。

To do (if possible):

- Check whether `obabel` and `inkscape` are successfully installed or not, before running programs.
- The method for changing category codes is not sophisticated (users have to change them on their own!), so further improvement needed.

9 Version History

Version 0.1 (December 1, 2014):	Made public (as <code>smilesobabel.sty</code>)
Version 0.2 (December 2, 2014):	Add options which can be passed to <code>obabel</code> .
Version 0.3 (December 7, 2014):	Change name of package: <code>chemobabel.sty</code> Images are stored in <code>chemobabelimgdir</code> . Add <code>\chemobabel</code> command.
Version 0.4 (December 9, 2014):	Fix a bug: (Thanks: Yusuke Terada) Extra spaces at the end of lines are removed.
Version 0.5 (December 20, 2014):	Add an option: <code>extract</code> option can load the macro easily.
Version 0.6 (June 29, 2015):	Improve warning messages.

Important!

- In Version 0.2, the number of parameters set in `\smilesobabel` is changed!
- From Version 0.3, the package name is changed to `chemobabel.sty`.

References

- [1] Cheer up your L^AT_EX with SMILES support – Noel O’Blog
- [2] Cheer up your L^AT_EX with SMILES support II – Noel O’Blog
- [3] L^AT_EX: Graphviz and OpenBabel – Jakob Lykke Andersen
- [4] 文章内の画像のみを表示する方法 – T_EX Forum
(How to extract only figures in L^AT_EX source)
- [5] 化学構造式を T_EX で (1) : 自動化による簡単生成 – Acetaminophen’s diary
(Chemical Structural Formula in L^AT_EX (1): An Easy Method by Auto-generation)
- [6] 化学構造式を T_EX で (2) : 自動化の注意点と解消法 – Acetaminophen’s diary
(Chemical Structural Formula in L^AT_EX (2): Important Notes for Auto-generation and Solution)
- [7] 化学構造式を T_EX で (3) : 補足事項 – Acetaminophen’s diary
(Chemical Structural Formula in L^AT_EX (3): Supplement)

Note: Acetaminophen’s diary is my own blog! (Sorry, but only in Japanese)

I added a post about this package on the 8th day of T_EX & L^AT_EX Advent Calendar 2014 (in Japanese):

- 誰でも簡単！ 化学構造式を L^AT_EX に取り込むパッケージ – Acetaminophen’s diary
(An easy way to insert chemical structural formulas into L^AT_EX documents)