

# chemscheme — Support for chemical schemes \*

Joseph Wright †

Released 2007/08/05

## Abstract

The chemscheme package consists of two parts, both related to chemical schemes. The package adds a `scheme` float type to the LaTeX default types `figure` and `table`. The `scheme` float type acts in the same way as those defined by the LaTeX kernel, but is intended for chemical schemes. The package also provides a method for adding automatic chemical numbering to schemes.

## 1 Introduction

By default, LaTeX defines two float types, `figure` and `table`. Synthetic chemists make heavy use of schemes, which need a `scheme` float type. This is provided by chemscheme, in a manner consistent with the kernel floats.

Synthetic chemists also number compounds for ease of reference. There are a number of LaTeX packages which cover this area, most notably `bpchem` and `chemcompounds`. However, adding numbers automatically to schemes is not covered by any existing package. The chemscheme package seeks to rectify this.

## 2 Floating schemes

### 2.1 Basic use

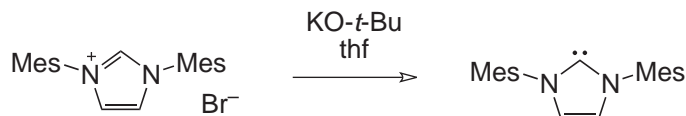
`scheme` The package provides a new float type, `scheme`, accessed in the usual way. So, issuing the following commands:

```
\begin{scheme}[ht]  
  \centering  
  \includegraphics{scheme-one}\\\
```

---

\*This file describes version v1.1, last revised 2007/08/05.

†E-mail: joseph.wright@morningstar2.co.uk



Scheme 1: A scheme with no compound numbers.

```
\caption{A scheme with no compound numbers.}
\label{scheme-one}
\end{scheme}
```

will lead to the appearance of a scheme as requested (Scheme 1). By default, the positioning of scheme floats follows the standard LaTeX system (positioning at the top of a page is preferred). As shown in the example, the use of positional modifiers is allowed. The float package is loaded by chemscheme, allowing the use of the [H] positional directive, meaning *really* here.<sup>1</sup> Labelling and referencing schemes follows the LaTeX conventions. So for example, to reference Scheme 1, you type

```
Scheme~\ref{scheme-one}
```

`\listofschemes`

To match the `\listoffigures` and `\listoftables` macros provided by the LaTeX kernel, chemscheme provides a `\listofschemes` command. This works in the same way as the kernel commands, taking a single optional argument [*Title*], which by default is “List of Schemes.” The output of the `\listofschemes` command is illustrated below.

## List of Schemes

1	A scheme with no compound numbers. . . . .	2
2	A scheme with temporary compound numbers. . . . .	4
3	A scheme with automated compound numbers. . . . .	4

## 2.2 Altering the defaults

For users of the standard class files or the KOMA-SCRIPT bundle, the float package is used to create the new float type. Thus the usual float commands can be used to modify the behaviour as desired. Users of memoir will have slightly different commands available, as memoir implements its own new float mechanism, which is used in that case. Schemes may be numbered within chapters (Scheme 1.1, Scheme 1.2, *etc.* by loading the package with the [chapter] option.

<sup>1</sup>memoir users should note that float cannot be used with this document class. The internal memoir system for float placement is thus in use in this case.

`\schemename`      Two macros are provided to allow the user to re-define the name of schemes and the title of the list of schemes in a consistent manner (independent of document class). Thus `\schemename` contains the text used in scheme captions (by default Scheme), and `\listschemes` contains the title of the scheme list (by default List of Schemes). Both may be freely re-defined.

### 3 Horizontal positioning of all floats

`\floatcontentscentre`    The LaTeX default is to position all float contents flush-left. There is no “hook”  
`\floatcontentscenter`    provided to alter this. The `chemscheme` packages therefore provides com-  
`\floatcontentsleft`      mands to align all float contents automatically. As the macro names make  
`\floatcontentsright`     clear, `\floatcontentscentre` will make all floats centred (for users speaking  
U.S. English, the alternative spelling `\floatcontentscenter` is also available).  
The default behaviour is restored using the command `\floatcontentsleft`.  
Finally, `\floatcontentsright` is provided for use if needed. notice that the  
float positioning commands should be given *outside* floating environments, and  
apply to all subsequent floats.

It is important to note that the positioning mechanism used here relies on a low-level hack of the LaTeX kernel. This has been tested with the standard LaTeX classes, the memoir class and the KOMA-SCRIPT bundled. Other document classes may not give the desired behaviour.

## 4 Reference numbers in graphics

### 4.1 Background

There are a number of packages available on CTAN for tracking compound reference numbers. The two with the most up to date and comprehensive features are `bpchem` and `chemcompounds`. Both allow in-text numbering to be handled automatically. However, neither will allow the use of these numbers directly in schemes, figures, *etc.* Both leave it to the user to manually adapt schemes to match any changes in numbering.

The `chemscheme` package provides a mechanism for rectifying this issue, provided that *compilation occurs in LaTeX (DVI) mode and not using pdfLaTeX*.<sup>2</sup>

### 4.2 Usage

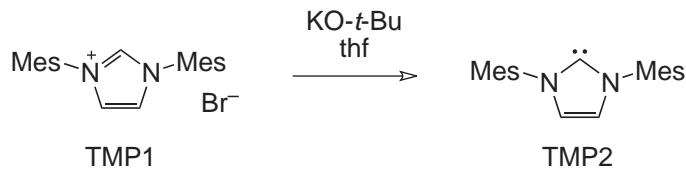
Getting automated numbers into schemes is a two step procedure. In the first step, schemes (or other graphics) should be prepared as normal and saved as encapsulated postscript (EPS) files. The most popular chemistry drawing package, CHEMDRAW, is able to do this from the Save As . . . dialog. The positions where the auto-labels should be positioned have to be marked in the EPS file. The marker should consist of an “indicator” that the text is to be replaced, followed

`\chemschemerefmarker`

---

<sup>2</sup>Of course, both may be using the pdfTeX engine; it is the output mode that is important.

by a reference number or letter. The “indicator” text should be the same in all graphics; it is stored in `\chemschemerefmarker`, and has default value `TMP`. Thus the graphics should contain labels `TMP1`, `TMP2`, etc. A suitable unmodified graphic is shown in Scheme 2.

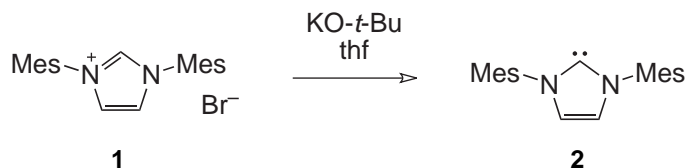


Scheme 2: A scheme with temporary compound numbers.

The value of `\chemschemerefmarker` may be altered as normal using `\renewcommand`, so for example to use `XXX` as the indicator for replacement, you would execute `\renewcommand{\chemschemerefmarker}{XXX}`.

`\chemschemeref` In the second step, the command `\chemschemeref` is used to indicate the mapping of the temporary markers to the automatically-managed numbering. The syntax of the command is `\chemschemeref{<temp-marker>}{<label>}`, where `<temp-marker>` is the unique marker used in the graphic, and `<label>` is the name assigned to the compound by the user. Each replacement requires a separate `\chemschemeref`, all of which should appear before the relevant `\includegraphics` command. Notice that `<temp-marker>` only contains the unique part of the temporary marker. Thus using the default indicator `TMP`, the EPS file might contain `TMP1` and `<temp-marker>` would be `1`.

An example will make usage clearer. In the example used in this document, the starting material is given label `IMesHCl` and the product is called `IMes`. As is shown in Scheme 2, in the EPS file these are labelled `TMP1` and `TMP2`, respectively. To relabel these as `1` and `2`, the following is needed:



Scheme 3: A scheme with automated compound numbers.

```
\begin{scheme}[ht]
\centering
\chemschemeref{1}{IMesHCl}
\chemschemeref{2}{IMes}
\includegraphics[scheme-two]{\}
\caption{A scheme with automated compound numbers.}
\label{scheme-two}
\end{scheme}
```

It is important to notice that the chemical labels must be defined *before* the use of `\chemschemeref`, or odd things may happen (most likely your document will not compile with strange errors). Also notice that the new label is centred on the middle of the temporary marker, with the same baseline. This should allow the user to obtain good alignment of labels and structures.

`\chemschemerefs`

As described in Section 4.3, `chemscheme` supports `bpchem`, which allows tracking of sub-labels (**1a**, **1b**, *etc.*). To allow use of these in schemes, the `\chemschemerefs` command is provided. This takes an additional argument `{\sub-label}`, which is used to generate the appropriate text. When used with `chemcompounds`, this command will gobble its arguments and issue a warning; no substitution will take place.

`\chemschemerefformat`

The format of chemical references is controlled by the underlying package, `bpchem` or `chemcompounds`. However, it is useful to be able to specify additional formatting for schemes. By default, `chemscheme` formats all reference numbers in a sans serif font. This is controlled by `\chemschemerefformat`. The additional formatting applied within schemes may be altered by redefining `\chemschemerefformat`. By careful choice of the font commands given here, good visual matching should be obtained between the automatically-generated labels and other text in the scheme. In this document, the CHEMDRAW source uses 10 point Arial, with the LaTeX sans serif font provided by the `helvet` package, loaded scaled to 95 %, *i.e.*

```
\usepackage[scaled=0.95]{helvet}
```

The CHEMDRAW file used to generate the example schemes is included with the package, as `chemscheme.cdx`.

### 4.3 Managing chemical numbering

The `chemscheme` package can use one of two packages for management of chemical numbering. Both `bpchem` and `chemcompounds` are supported: `bpchem` is the default, and `chemcompounds` is requested by loading `chemscheme` with the `[chemcompounds]` option. Both packages have advantages: `bpchem` allows the tracking of sub-references (very common in organic chemistry), whilst `chemcompounds` has a very well thought-out interface. It is technically feasible to support both simultaneously, but this is unlikely to have wide application. For this reason, `chemscheme` loads only one package (depending on the user option given), and uses this package to provide numbering management.

Both `bpchem` and `chemcompounds` provide a variety of methods for defining chemical labels. In either case, the user should define the chemical label *before* it is referenced by `\chemschemeref`. Errors are very likely to occur if this is not done.

## 5 Known issues

The number replacement code relies on the use of `psfrag`, which can only be applied to EPS graphics. Thus, users of pdfLaTeX will not benefit from the automated number replacement. For final compilation directly to PDF, it may be preferable to modify the graphics files and use `epstopdf` to include the files. The alternative is to use the more traditional route  $\text{DVI} \rightarrow \text{PS} \rightarrow \text{PDF}$ .

Currently, the names of schemes are only available in English. If the correct translations are provided, the package author will aim to incorporate the appropriate code.

## 6 Implementation

### 6.1 Setup code

The initial code goes through the usual steps of identifying the package.

```
1 \NeedsTeXFormat{LaTeX2e}
2 \ProvidesPackage{chemscheme}%
3 [2007/08/05 v1.1 Support for chemical schemes]
```

The `psfrag` package is required to carry out the inclusion of chemical numbers in graphics. If the `memoir` package is not being used, the `float` package is used to provide support for new float types. However, if `memoir` is in use, `float` should not be loaded.

```
4 \RequirePackage{psfrag}
5 \@ifclassloaded{memoir}{}
6 {\RequirePackage{float}}
```

Option processing now takes place. This requires two Boolean values, which are declared here. By default, `chemscheme` uses `bpchem` for managing chemical compound references. The options reflect this.

```
7 \newif \ifCHEMSCH@chapter \CHEMSCH@chapterfalse
8 \newif \ifCHEMSCH@bpchem \CHEMSCH@bpchemtrue
9 \DeclareOption{chapter}{\global\CHEMSCH@chaptertrue}
10 \DeclareOption{chemcompounds}{\global\CHEMSCH@bpchemfalse}
11 \DeclareOption{bpchem}{}
12 \ProcessOptions
```

### 6.2 Support for schemes

`\schemename`    The new float type for schemes is produced using the `float` package, or the appropriate `memoir` code. First, names for schemes and the list of schemes are set up. These are provided for ease of modification, as demonstrated in the `memoir` manual.

`\listschemes`

```
13 \newcommand*{\schemename}{Scheme}
14 \newcommand*{\listschemes}{List of Schemes}
```

scheme The `\newfloat` command is used to generate the new float type, first checking if numbering within chapters is requested. A suitable warning is issued if no chapters exist in the document. The syntax required depends on whether `float` or `memoir` is used for the definition.

```

15 \ifCHEMSCH@chapter
16   \ifx\chapter\@undefined
17     \PackageWarning{chemscheme}%
18       {No chapters in this document \MessageBreak%
19         Ignoring 'chapter' option}
20     \@ifclassloaded{memoir}
21       {\newfloat{scheme}{los}{\schemename}}
22       {\newfloat{scheme}{tbp}{los}}
23   \else
24     \@ifclassloaded{memoir}
25       {\newfloat[chapter]{scheme}{los}{\schemename}}
26       {\newfloat{scheme}{tbp}{los}[chapter]}
27   \fi
28 \else
29   \@ifclassloaded{memoir}
30     {\newfloat{scheme}{los}{\schemename}}
31     {\newfloat{scheme}{tbp}{los}}
32 \fi

```

`\listofschemes` To ensure that things are labelled correctly, the new float type is given an appropriate name. Notice that `memoir` does this at the float-definition stage. In common with the standard float types, a `\listof` command is provided for schemes.

```

33 \@ifclassloaded{memoir}
34   {\newlistof{listofschemes}{los}{\listofschemes}}
35   {\floatname{scheme}{\schemename}}
36   \newcommand*{\listofschemes}[1][\listofschemes]%
37     {\listof{scheme}{#1}}

```

### 6.3 Positioning float contents

`\floatcontentscentre` In order to centre the content of all floats, a method is needed to break into the mechanism. None is provided by default, but it can be achieved by patching `\floatcontentscenter` `\@floatboxreset`. User space switching commands are defined to turn centring on and off.

```

38 \newcommand*{\floatcontentscentre}%
39   {\let\CHEMSCH@everyfloat\centering}
40 \let\floatcontentscenter\floatcontentscentre
41 \newcommand*{\floatcontentsleft}%
42   {\let\CHEMSCH@everyfloat\relax}
43 \newcommand*{\floatcontentsright}%
44   {\let\CHEMSCH@everyfloat\raggedleft}
45 \let\CHEMSCH@floatboxreset\@floatboxreset
46 \floatcontentsleft
47 \def\@floatboxreset{\CHEMSCH@everyfloat\CHEMSCH@floatboxreset}

```

## 6.4 Reference numbers in graphics

<code>\chemschemerefmarker</code>	The two macros <code>\chemschemerefmarker</code> and <code>\chemschemerefformat</code> are used to allow customisation of the behaviour of the package. Here defaults are provided.
<code>\chemschemerefformat</code>	<pre> 48 \newcommand{\chemschemerefmarker}{TMP} 49 \newcommand{\chemschemerefformat}{\textsf{}} </pre>
<code>\CHEMSCH@label</code>	<p>Depending on the user options provided, either <code>bpchem</code> or <code>chemcompounds</code> is loaded to manage chemical citations. The macro <code>\CHEMSCH@label</code> is defined as the labelling command of the appropriate package.</p> <pre> 50 \ifCHEMSCH@bpchem 51   \RequirePackage{bpchem} 52   \let\CHEMSCH@label\CNlabel 53 \else 54   \RequirePackage{chemcompounds} 55   \let\CHEMSCH@label\compound 56 \fi </pre>
<code>\chemschemeref</code>	<p>The referencing macro is a wrapper for the appropriate <code>\psfrag</code> command. The first argument of <code>\chemschemeref</code> is the temporary code assigned to a compound, with the second containing the reference used for the compound in the citation scheme. The use of <code>[b][b]</code> places the new label at the centre of the position of the temporary one, at the same vertical position.</p> <pre> 57 \newcommand{\chemschemeref}[2]{% 58   \psfrag{\chemschemerefmarker#1}[b][b]% 59   {\chemschemerefformat{\CHEMSCH@label{#2}}}% 60 } </pre>
<code>\chemschemerefsub</code>	<p>The sub version of the referencing command is needed as <code>bpchem</code> allows tracking of sub-compounds (<i>e.g.</i> <b>1a</b>, <b>1b</b>, <i>etc.</i>). As <code>chemcompounds</code> does not do this, the command in that case gobbles up the third argument and gives the user a suitable warning.</p> <pre> 61 \ifCHEMSCH@bpchem 62   \newcommand{\chemschemerefsub}[3]{% 63     \psfrag{\chemschemerefmarker#1}[b][b]% 64     {\chemschemerefformat{\CNlabelsub{#2}{#3}}}% 65   } 66 \else 67   \newcommand{\chemschemerefsub}[3]{% 68     \PackageWarning{chemscheme}% 69     {'chemcompound' option active\MessageBreak 70     Command \protect\chemschemerefsub\space not used \MessageBreak 71     Please alter your source to \MessageBreak 72     \protect\chemschemeref}} 73 \fi </pre>



# Change History

v1.0	v1.1
General: Initial public release . . . . . 1	General: Added float centring code 1

## Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

<b>Symbols</b>	<code>\chemschemeref</code> ..	<code>\floatcontentsright</code>
<code>\@floatboxreset</code> 45, 47	..... 4, <u>57</u> , 72	..... 3, <u>38</u>
	<code>\chemschemerefformat</code>	
	..... 5, <u>48</u> , 59, 64	
<b>C</b>	<code>\chemschemerefmarker</code>	<b>I</b>
<code>\centering</code> ..... 39	..... 3, <u>48</u> , 58, 63	<code>\ifCHEMSCH@bpchem</code>
<code>\CHEMSCH@bpchemfalse</code>	<code>\chemschemereffsub</code>	<code>\ifCHEMSCH@chapter</code>
..... 10	..... 5, <u>61</u>	..... 7, 15
<code>\CHEMSCH@bpchemtrue</code>		
..... 8	<b>E</b>	<b>L</b>
<code>\CHEMSCH@chapterfalse</code>	environments:	<code>\listofschemes</code> . 2, <u>33</u>
..... 7	scheme ..... 1, <u>15</u>	<code>\listschemes</code> ....
<code>\CHEMSCH@chaptertrue</code>	<b>F</b>	..... 2, <u>13</u> , 34, 36
..... 9	<code>\floatcontentscenter</code>	
<code>\CHEMSCH@everyfloat</code>	..... 3, <u>38</u>	<b>S</b>
..... 39, 42, 44, 47	<code>\floatcontentscentre</code>	scheme (environment)
<code>\CHEMSCH@floatboxreset</code>	..... 3, <u>38</u>	..... 1, <u>15</u>
..... 45, 47	<code>\floatcontentsleft</code>	<code>\schemename</code> ....
<code>\CHEMSCH@label</code> . <u>50</u> , 59	..... 3, <u>38</u>	. 2, <u>13</u> , 21, 25, 30, 35