

chemstyle — Writing chemistry with style*

Joseph Wright[†]

Released 2008/07/17

Abstract

The chemstyle package provides a “one-stop shop” for setting up formatting of L^AT_EX documents following the editorial policies of various chemical journals. It provides a number of handy chemistry-related commands, and loads several support packages to aid the chemist.

Contents

| | | | | | |
|----------|---|----------|-----------|---|-----------|
| 1 | Introduction | 2 | 7 | The package code | 8 |
| | | | 7.1 | Setup code | 8 |
| | | | 7.2 | Option handling | 8 |
| | | | 7.3 | Extra units | 10 |
| | | | 7.4 | Standard state symbol | 10 |
| | | | 7.5 | Alkyl radicals | 11 |
| | | | 7.6 | Float formatting | 12 |
| | | | 7.7 | Cross-references | 12 |
| | | | 7.8 | Latin phrases | 12 |
| | | | 7.9 | Loading the style definition | 14 |
| | | | 7.10 | Handling reference section naming | 14 |
| 2 | Style options for chemstyle | 2 | 8 | Configuration files | 16 |
| 2.1 | Creating styles | 3 | 8.1 | RSC style | 16 |
| 2.2 | Package options | 3 | 8.2 | <i>Angew. Chem.</i> style | 17 |
| | | | 8.3 | <i>J. Organomet. Chem.</i> style | 17 |
| 3 | Naming of the references section | 3 | 8.4 | <i>Tetrahedron Lett.</i> style | 17 |
| | | | 8.5 | <i>J. Am. Chem. Soc.</i> style | 17 |
| 4 | Additional macros | 4 | 8.6 | <i>Inorg. Chem.</i> style | 18 |
| 4.1 | Additional units | 4 | 8.7 | <i>J. Phys. Chem.</i> style | 18 |
| 4.2 | The standard state symbol | 4 | 8.8 | <i>Org. Lett.</i> style | 18 |
| 4.3 | Alkyl radicals | 4 | 8.9 | The empty style — none | 20 |
| 4.4 | Latin phrases | 5 | 9 | Change History | 20 |
| 5 | Additional information | 5 | 10 | Index | 21 |
| 5.1 | Advice for users of the rsc package | 5 | | | |
| 5.2 | Interactions with other packages | 6 | | | |
| 5.3 | Captions above floats | 6 | | | |
| 6 | A template for chemical articles | 7 | | | |

*This file describes version v1.3, last revised 2008/07/17.

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

Table 1: Styles provided by chemstyle

| Option | Journals using this style |
|---------|--|
| none | Not applicable |
| angew | <i>Angew. Chem., Chem. Eur. J.</i> |
| jomc | <i>J. Organomet. Chem., Coord. Chem. Rev.</i> |
| ic | <i>Inorg. Chem.</i> |
| jacs | <i>J. Am. Chem. Soc.</i> |
| jcp | <i>J. Phys. Chem. A, J. Phys. Chem. B</i> |
| orglett | <i>Org. Lett.</i> |
| rsc | <i>Chem. Commun., Org. Biomol. Chem.</i> <i>Dalton Trans.</i> |
| tetlett | <i>Tetrahedron, Tetrahedron Lett.</i> |

1 Introduction

The aim of chemstyle is to provide a quick method to set up various document parameters (such as caption formatting), simply by specifying the model journal. The package has also been designed to allow rapid addition of new journal styles. Each style definition is a separate file, and new styles can be added very readily. chemstyle has grown out of the rsc package, which had a similar aim but was much more limited (and less robustly implemented). The chemstyle package is also designed with the use of biblatex in mind: the rsc package is closely bound to traditional BibTeX use.

As a successor to the rsc package, chemstyle provides a range of chemistry-related additional macros. The set provided here is an extended version of those provide by rsc. Everything that can be done using the rsc L^AT_EX package is therefore possible using the chemstyle package.

The formatting system provided by chemstyle are intended for writing a variety of documents. Thus the stylistic changes made by the package do not seek to reproduce the appearance of printed journal articles. The package aims to be suitable for use in preparing drafts of papers, but also for writing reports, theses and so on.

2 Style options for chemstyle

journal chemstyle should be loaded with a package option specifying which journal style to follow. Currently, chemstyle is aware of a the styles listed in Table [Table 1](#). New styles can be developed by creating a new file modelled on the existing definitions; chemstyle will automatically search for correctly-named styles. The style files provided with chemstyle have been derived from current practice in the target journals. It is not always easy to pick the correct stylistic settings from (sometimes inconsistent) real-world examples. The package author welcomes feedback on the styles provided.

The none style is notable as it is not based on a journal. Instead this is a minimal style, which provides the additional commands without making formatting changes. It is also the default style if no option is given. This

document has been compiled implicitly using the `none` option, for example.

2.1 Creating styles

The process of creating a new style for `chemstyle` is intended to be relatively easy. New styles should be saved as files with the extension `.jdf` (standing for Journal Definition File), and should be saved somewhere in the path searched by \TeX .¹ The definition files included in the package should provide a guide to the basic options available for producing new styles. Arbitrary \TeX commands can be included, if they are necessary for a particular style. For example, other packages can be loaded in `\usepackage`.

The maintainer of `chemstyle` is happy to add new styles to the package, either by contribution by users or on request (when he has sufficient time!). If you have a new style to add (or corrections to an existing one), please contact the package author.

2.2 Package options

| | |
|---|---|
| <code>journal</code> | As of version 1.2, <code>chemstyle</code> uses the <code>keyval</code> system for option management. The choice of journal style can therefore be given as a normal option or by using the key <code>journal=</code> . The other package options provided by the package are described as they arise in the documentation. |
| <code>\cstsetup</code> | Some of the package options can only be set when loading the package. The choice of journal style is one of these. Others can be altered anywhere in the document, using the <code>\cstsetup</code> macro. This accepts a <code>keyval</code> list and processes it as needed. |
| <code>chemcompounds</code> <code>floatrow</code> | <code>chemstyle</code> loads the <code>chemscheme</code> package, and therefore will pass through the <code>optchemcompounds</code> option. It can also request floats are created either by <code>float</code> or <code>floatrow</code> ; this is controlled by the <code>floatrow</code> option. Setting <code>floatrow=false</code> means that <code>float</code> is used to generate floats; the default value is <code>true</code> . |

3 Naming of the references section

`chemstyle` alters the naming of the references section of a document. By default, `chemstyle` alters the value of `\bibname` or `\refname` (as appropriate) to the form of words chosen by the target journal for the “References” section.

The “Notes and References” naming commands are language-aware, *via* the `babel` interface. Currently, `chemstyle` includes appropriate labels for `babel` languages `english`, `UKenglish`, `ngerman` and `french`. Other languages can be added if appropriate wordings are provided to the author. The naming system is designed to work correctly with both `natbib` and `biblatex`.

| | |
|--|---|
| <code>notes</code> <code>nonotes</code> | The package recognises the <code>notes</code> option for controlling how the references section is named. It accepts the values <code>true</code> , <code>false</code> and <code>auto</code> . The Boolean values either enable or disable the addition of “Notes and” to the “References” of the section title. The <code>auto</code> option works in conjunction with the <code>notes2bib</code> package. If notes are added, “Notes and” is included in the section title, whereas |
|--|---|

¹Depending on your \TeX distribution, you may need to rebuild your file database after creating a new style. For Mik \TeX users, this can be done graphically or by typing `mpm -update-db` at the command line; for TeXLive, run `texhash`.

if no notes are given the section title remains as “References”. The `nonotes` option is equivalent to `notes=false`.
`notesbefore` The second option for this area is `notesbefore`. This takes `true` and `false` only, and sets whether “Notes and References” or “References and Notes” is produced.

4 Additional macros

4.1 Additional units

| | |
|---------------------|--|
| <code>\Hz</code> | Both the <code>Slunits</code> and <code>unitsdef</code> packages provide methods for handling a range of units, but do not include a small number of unit macros useful to chemists. In contrast, the <code>siunitx</code> package provides all of these using the <code>synchem</code> module. If <code>Slunits</code> or <code>unitsdef</code> are loaded, <code>chemstyle</code> provides the missing units. On the other hand, if no unit package is loaded, <code>siunitx</code> is loaded and provides the units itself. |
| <code>\mmHg</code> | |
| <code>\molar</code> | |
| <code>\Molar</code> | |
| <code>\cmc</code> | |

| | |
|-------------------------|-------------------------------|
| 10 Hz | <code>\SI{10}{\Hz}\</code> |
| 20 mmHg | <code>\SI{20}{\mmHg}\</code> |
| 30 mol dm ⁻³ | <code>\SI{30}{\molar}\</code> |
| 40 M | <code>\SI{40}{\Molar}\</code> |
| 50 cm ³ | <code>\SI{50}{\cmc}\</code> |

4.2 The standard state symbol

`\standardstate` Related to the above, but not exactly a unit is the `\standardstate` command.² This generates the tricky `\standardstate` symbol.

| | |
|--|---|
| the standard conditions are indicated: | the standard conditions are indicated: <code>\standardstate\</code> |
| \ominus | <code>\$\Delta S\standardstate\$</code> |
| ΔS^\ominus | |

4.3 Alkyl radicals

| | |
|-------------------|---|
| <code>\nPr</code> | There are a few alkyl radicals that come up all of the time. No one seems to have put these into a package, so they are provided here. As you would expect, <code>\iPr</code> gives <i>i</i> -Pr, <code>\iBu</code> gives <i>i</i> -Bu and <code>\tBu</code> gives <i>t</i> -Bu, and so on. The style of the output depends on the journal style specified; most journals seem to favour one version of the abbreviation. |
| <code>\iPr</code> | |
| <code>\nBu</code> | |
| <code>\iBu</code> | |
| <code>\sBu</code> | |

| | |
|--|--|
| The alkyl group could be <i>n</i> -Pr, <i>i</i> -Pr or <i>n</i> -Bu without affecting the selectivity. | The alkyl group could be <code>\nPr</code> , <code>\iPr</code> or <code>\nBu</code> without affecting the selectivity. |
|--|--|

`xspace` When `chemstyle` is loaded using the `xspace` option, the `xspace` package is automatically used to add space after the command names, so that `\iPr` group will result in “*i*-Pr group” being typeset.

`radhyphen` The appearance of these radical abbreviations is controlled by the package options `radhyphen`, `rademph`, `radsuper` and `radprefix`. The journal styles set these automatically, but they can be redefined at any point. The options work as might be expected, and are Boolean switches.

`rademph`

`radsuper`

`radprefix`

²The `\standardstate` macro is only defined if the user does not have their own version.

iPr *tBu*
 Bu-*i* Bu-*s*

```
\cstsetup{radhyphen=false,radsuper=true}\\
\iPr \tBu
\cstsetup{radhyphen=true,radsuper=false,%
radprefix=false,rademph=false}\\
\iBu \sBu
```

4.4 Latin phrases

`\latin` The various Latin phrases commonly used in chemistry are made available as the obvious commands. By altering the definition of `\latin`, this allows ready switching from italic to Roman typesetting. Notice that `\etc`, `\ie` and `\eg` are aware of trailing periods, and so doubling-up should not occur. Once again, these macros use `xspace` when given as a package option to handle automatic addition of spaces after these phrases.

`\invacuo` The use of italic for these abbreviations is set by altering the package option `abbremph`, which takes values `true` and `false`.

et al. or *in vacuo* plus some text
i.e. equals *e.g.* perhaps *etc.*

```
\etal or \invacuo plus \latin(some text)\\
\cstsetup{abbremph=false}
\ie equals \eg perhaps \etc
```

`abbrcomma` For American journals, where it is obligatory to follow “*e.g.*” and “*i.e.*” with a comma, the package provides a mechanism for handling this automatically. Thus, when using an appropriate journal style, `\eg`, `\eg.` and `\eg,` will all result in typesetting “*e.g.,*”. The Boolean package option `abbrcomma` controls this.

e.g. this
i.e., that

```
\eg this\\
\cstsetup{abbrcomma=true}
\ie that
```

The `\etc` and `\etal` commands are set up on the assumption that they come at the end of a sentence. Hence the spacing after these will default to an inter-sentence space. If you desire an inter-word space, use the normal methods

etc. more text
et al. have shown

```
\etc\ more text \\
\etal~have shown
```

`nophrases` The definitions of all of the phrases are designed not to overwrite any given by the user *in the preamble*. So, if you have your own `\latin` macro, it will be used even if you load `chemstyle`. If you encounter any problems, try loading the package with the `nophrases` option; this option prevents the package even trying to define any of the phrase macros.

`\latinemphon` For backward-compatibility with previous versions of `chemstyle`, the macros `\latinemphon` and `\latinemphoff` are provided. These work as would be expected, to alter the formatting produced by `\latin`. The more general `keyval` method is now preferred for making this change.

5 Additional information

5.1 Advice for users of the `rsc` package

The `chemstyle` package is intended as a replacement for the `rsc` package. As such, it covers almost everything the the `rsc` L^AT_EX package does, and more. Users of `rsc`

are strongly encouraged to update to using `chemstyle`. The bibliography styles provided by `rsc` will continue to be required, of course. Migration of these styles to `biblatex` is an on-going project.

5.2 Interactions with other packages

The `chemstyle` package has been designed to avoid, as far as possible, clashes with other packages. The package requires the presence of the standard `graphicx` and `varioref` packages. If these packages need to be loaded with explicit options, this should be done *before* loading `chemstyle`. The `chemscheme` package is also needed, as it provides the floating `scheme` environment essential in synthetic chemistry documents.

5.3 Captions above floats

The `scheme` float type is generated using either the `float` or `floatrow` package. This has the side-effect that the placement of captions for floats does not depend on where the `\caption` command comes inside the floating environment.³ If you wish to alter the placement of captions, the mechanism of the underlying package will be needed. There are some subtle differences between the two: although `floatrow` provides the `float` macros, they are not all 100% compatible. This document is compiled using `floatrow`, and so to fix the position of captions the following code is appropriate.

```
\begin{table} [ht]
  The float contents
  \caption{A caption below the float contents in the source}
\end{table}
\floatsetup[table]{style=plain} % When using floatrow
% \floatstyle{plain}
% \restylefloat{table} When using float
\begin{table} [ht]
  The second float contents, which should be above the caption
  \caption{A second caption below the float contents
    in the source}
\end{table}
```

Table 2: A caption below the float contents in the source

The float contents

The second float contents, which should be above the caption

Table 3: A second caption below the float contents in the source

³Normally this is a good thing.

6 A template for chemical articles

This is a very simple template for chemistry-related documents. Hopefully it contains a few extra hints for getting well-formatted documents quickly. For simplicity, the template assumes that the user is writing a thesis for a U.K. university. Hence it uses U.K. defaults and RSC-based styling. Most of the packages used have good documentation, but a brief summary of why they are recommended follows.

```
1 \documentclass[fontsize=10pt,paper=a4,english,UKenglish]
2 {scrreprt}
3 \usepackage{geometry,upgreek,booktabs,babel}
4 \usepackage[journal=rsc,xspace=true]{chemstyle}
5 \usepackage[version=3]{mhchem}
6 \usepackage[footnotes]{notes2bib}
7 \usepackage[final]{microtype}
8 \usepackage[extrarun=on]{auto-pst-pdf}
9 \usepackage[colorlinks]{hyperref}
10 \begin{document}
11   Document contents go here
12 \end{document}
```

The versatile KOMA-script bundle provides more advanced versions of the standard document classes. If you want paragraphs separated out, with no indents (a common style for theses), add the `parskip` option to the font and paper size ones given here.

The `babel` system is loaded to sort out hyphenation and so on, and could be useful if there are any foreign-language quotes.

`geometry` allows the users to alter page layout with ease: much better than trying to hack the raw L^AT_EX system. `booktabs` gives much nicer looking tables than the L^AT_EX 2_ε default. `upgreek` provides non-italic lowercase Greek letters, which should be used for things such as bond descriptions.

π -bond σ^* -orbital

`\uppi` π -bond `\upsigma^*` σ^* -orbital

Load `chemstyle` (of course) to give not only some easy formatting, but also to automatically provide a float type for schemes, thanks to the `chemscheme` package. This also loads either `chemcompounds` or (optionally) `bpchem` to track compound numbers.

The `mhchem` package provides the `\ce` command for rapidly typesetting formulas, so that you can type `\ce{H2SO4}` and get H₂SO₄.⁴

The `microtype` package improves formatting when used with the pdfL^AT_EX engine. By giving the `final` option, it is active even when using `draft` as a global option.

Using `notes2bib` allows the user to automatically add notes to the bibliography from within the document body. So you can put `\bibnote{A note}` in the source, and this will move into the References section without any further effort. The `footnotes` option means that footnotes do the same.

Finally, the `hyperref` package makes headings, citations and so on into hyperlinks.

⁴There is a slight cheat here, as this document uses lower-case numerals in the text. The example is written as `\ce{H2SO4}`.

7 The package code

7.1 Setup code

`\cst@id` First of all, the package identifies itself and loads other packages needed to
`\cst@ver` function. Loading chemscheme is deferred until later, to allow processing of options at the correct point.

```
13 \NeedsTeXFormat{LaTeX2e}
14 \def\cst@id$#1: #2.#3 #4 #5-#6-#7 #8 #9${%
15     \def\cst@ver{#5/#6/#7\space v1.3\space}}
16 \cst@id $Id: chemstyle.dtx 36 2008-07-17 07:39:53Z joseph $
17 \ProvidesPackage{chemstyle}
18 [\cst@ver Writing chemistry with style]
```

Packages that are needed under all circumstances are loaded here. There are some limitations on the age of xkeyval.

```
19 \RequirePackage{xkeyval}[2005/05/07]
20 \RequirePackage{graphicx,varioref,caption,xspace}
```

`\cst@tempa` A couple of temporary private macros.

```
\cst@tempb 21 \newcommand*\cst@tempa{}
22 \newcommand*\cst@tempb{}
```

7.2 Option handling

`\cst@boolkey` Option handling is processed using xkeyval. As a number of Boolean keys are needed, a shortcut is handy.

```
23 \newcommand*\cst@boolkey}[1]{
24     \define@boolkey[cst]{opt}[cst@]{#1}[true]{} }
```

`\ifcst@chemcompounds` The Boolean keys are now defined.

```
\ifcst@floatrow 25 \cst@boolkey{chemcompounds}
\ifcst@xspace 26 \cst@boolkey{floatrow}
\ifcst@phrases 27 \cst@boolkey{xspace}
\ifcst@radprefix 28 \cst@boolkey{phrases}
\ifcst@rademph 29 \cst@boolkey{radprefix}
\ifcst@radsuper 30 \cst@boolkey{rademph}
\ifcst@radhyphen 31 \cst@boolkey{radsuper}
\ifcst@abbremph 32 \cst@boolkey{radhyphen}
\ifcst@abbrcomma 33 \cst@boolkey{abbremph}
\ifcst@notesbefore 34 \cst@boolkey{abbrcomma}
35 \cst@boolkey{notesbefore}
```

`\ifcst@notes` For notes, some choices are available. These need a bit of testing.

```
\ifcst@autonotes 36 \newif\ifcst@notes
\cst@tempa 37 \newif\ifcst@autonotes
\cst@tempb 38 \define@choicekey*[cst]{opt}{notes}[\cst@tempa
39 {true,false,auto}[true]
40 {\cst@notesfalse
41 \cst@autonotesfalse
42 \edef\cst@tempb{auto}
43 \ifx\cst@tempa\cst@tempb
```



```

44     \cst@notesttrue
45     \cst@autonotesttrue
46   \fi
47   \edef\cst@tempb{true}
48   \ifx\cst@tempa\cst@tempb
49     \cst@notesttrue
50   \fi}

```

Some old options are no longer needed at all.

```

51 \DeclareOptionX[cst]<opt>{siunits}
52   {\PackageInfo{chemstyle}{Option siunits no longer required}}
53 \DeclareOptionX[cst]<opt>{SIunits}
54   {\PackageInfo{chemstyle}{Option siunits no longer required}}
55 \DeclareOptionX[cst]<opt>{nonotes}
56   {\ExecuteOptionsX[cst]<opt>{notes=false}}
57 \DeclareOptionX[cst]<opt>{nophrases}
58   {\ExecuteOptionsX[cst]<opt>{phrases=false}}

```

\cst@journal The journal option can be given with or without journal=.

\cst@language 59 \define@cmdkeys[cst]{opt}[cst@]{journal,language}

60 \DeclareOptionX*{\edef\cst@journal{\CurrentOption}}

The default options are executed, and the user options are processed.

```

61 \setkeys[cst]{opt}{
62   notes=true,
63   phrases=true,
64   chemcompounds=true,
65   xspace=true,
66   floatrow=true,
67   journal=none,
68   radprefix=true,
69   rademph=true,
70   radsuper=false,
71   radhyphen=true,
72   abbrevemph=true,
73   abbrcomma=false,
74   language=english,
75   notesbefore=true}
76 \ProcessOptionsX[cst]<opt>

```

With the options processed, those to be passed to chemscheme are sorted out.

```

77 \ifcst@chemcompounds
78   \PassOptionsToPackage{chemcompounds}{chemscheme}
79 \fi
80 \ifcst@floatrow
81   \RequirePackage[floats=floatrow]{chemscheme}
82 \else
83   \RequirePackage[floats=float]{chemscheme}
84 \fi

```

\cst@disablekey Some of the options are no longer relevant once the package is loaded. These are disabled here with a suitable message.

```

85 \newcommand*{\cst@disablekey}[1]{
86   \define@key[cst]{opt}{#1}{\PackageWarning{chemstyle}

```

```

87     {Option '#1' only valid when loading package}}
88 \AtBeginDocument{
89   \cst@disablekey{floatrow}
90   \cst@disablekey{journal}
91   \cst@disablekey{phrases}
92   \cst@disablekey{nophrases}
93   \cst@disablekey{notes}
94   \cst@disablekey{nonotes}
95   \cst@disablekey{chemcompounds}}

```

`\cstsetup` A macro for changing the settings at run-time without needing to know the internals of the package.

```

96 \newcommand*{\cstsetup}[1]{%
97   \iffalse{\fi\ifnum0='}\fi
98   \setkeys[cst]{opt}{#1}%
99   \ifnum0='{ \fi\iffalse}\fi}

```

7.3 Extra units

`\cmc` The `siunitx` package provides all of the units desired here. So if it is loaded, the module `synchem` is all that needs to be loaded. Otherwise, a series of tests are needed for `SIunits` and `unitsdef`. If either of these are loaded, they are used; if not, then `siunitx` is loaded in any case.

```

\Hz
\cubiccentimeter
\Molar
\molar
\mmHg
\mol
100 \AtBeginDocument{
101   \@ifpackageloaded{siunitx}
102     {\requiresiconfigs{synchem}}
103     {\@ifpackageloaded{SIunits}
104       {\newcommand*{\cubiccentimeter}{\centi\metre\cubed}
105        \newcommand*{\Molar}{\textsc{m}}
106        \newcommand*{\molar}{\mole\usk\deci\metre\rpcubed}
107        \newcommand*{\mmHg}{\milli\metre\Hg}
108        \newcommand*{\mol}{\mole}
109        \newcommand*{\cmc}{\cubiccentimeter}
110        \newcommand*{\Hz}{\hertz}}
111     {\@ifpackageloaded{unitsdef}
112       {\newunit{\cubiccentimeter}{%
113         \cm\unitsuperscript{3}}
114        \newunit{\Molar}{\textsc{m}}
115        \newunit{\molar}{%
116          \mole\unitsep\dm\unitsuperscript{--3}}
117        \newunit{\mmHg}{mmHg}
118        \newunit{\mol}{\mole}
119        \newcommand*{\cmc}{\cubiccentimeter}
120        \newcommand*{\Hz}{\hertz}}
121       {\RequirePackage[alsoload=synchem]{siunitx}}}}

```

7.4 Standard state symbol

`\cst@varnothing` The “standard state” symbol is handy. This is produced by rotating the “varnothing” symbol from the AMS set. Note that the rotation angle here has been carefully checked, but is set by eye. The symbol is loaded directly here, rather than using the `amssymb` package, to avoid any clashes.

`\standardstate`

```

122 \DeclareSymbolFont{CSTAMS}{U}{msb}{m}{n}
123 \DeclareMathSymbol{\cst@varnothing}{\mathord}{CSTAMS}{"3F}
124 \providecommand*\standardstate{%
125   {\textsuperscript{\rotatebox[origin=c]{138.8}%
126     {\ensuremath{\cst@varnothing}}}}

```

7.5 Alkyl radicals

`\cst@radical` The `\cst@radical` macro does the hard work of declaring each abbreviation. The commands are made robust so they can (hopefully) be used anywhere.

```

127 \newcommand*\cst@radical}[2]{%
128   \expandafter\DeclareRobustCommand\expandafter*\expandafter{%
129     \csname #1#2\endcsname}%

```

`\cst@emph` To allow redefinition while running, the various component parts of the system are defined each time an abbreviation is used.

```

\cst@hyphen
\cst@super
130   \ifcst@radhyphen
131     \def\cst@hyphen{-}%
132   \else
133     \let\cst@hyphen\relax
134   \fi
135   \ifcst@radsuper
136     \let\cst@super\textsuperscript
137   \else
138     \let\cst@super\relax
139   \fi
140   \ifcst@rademph
141     \let\cst@emph\emph
142   \else
143     \let\cst@emph\relax
144   \fi

```

The final check is for a prefix, which is easiest to do here.

```

145   \ifcst@radprefix
146     \cst@super{\cst@emph{#1}}\cst@hyphen#2%
147   \else
148     #2\cst@hyphen\cst@super{\cst@emph{#1}}%
149   \fi
150   \cst@xspace}}

```

`\cst@xspace` A re-useable `\xspace`-based macro.

```

151 \newcommand*\cst@xspace{%
152   \ifcst@xspace
153     \expandafter\xspace%
154   \fi}

```

`\nPr` With a mechanism in place, the abbreviations are declared. The format of the resulting output will depend upon the configuration file used.

```

\iPr
\nBu 155 \cst@radical{n}{Pr}%
\iBu 156 \cst@radical{i}{Pr}%
\sBu 157 \cst@radical{n}{Bu}%
\tBu 158 \cst@radical{i}{Bu}%
      159 \cst@radical{s}{Bu}%
      160 \cst@radical{t}{Bu}%

```

7.6 Float formatting

The next step is to format the floats correctly. The standard float types are now restyled to place the captions correctly (for most journals). Normally in chemical documents the author expects the float to be “here” if possible; this is therefore set as the default.

```
161 \floatstyle{plaintop}
162 \restylefloat{table}
163 \floatstyle{plain}
164 \restylefloat{figure}
165 \floatplacement{table}{htbp}
166 \floatplacement{figure}{htbp}
```

Work with schemes needs to be done after hyperref might be loaded.

```
167 \AtBeginDocument{
168   \restylefloat{scheme}
169   \floatplacement{scheme}{htbp}}
```

The contents of floats are centred by default, using the hook from the chemscheme package.

```
170 \floatcontentscentre
```

7.7 Cross-references

The naming for cross-references is sorted out properly using the varioref package.

```
171 \labelformat{figure}{\figurename~#1}
172 \labelformat{table}{\tablename~#1}
173 \AtBeginDocument{\labelformat{scheme}{\schemename~#1}}
```

7.8 Latin phrases

| | |
|---|--|
| <pre>\latin \cst@latin \latinemphon \latinemphoff</pre> | A series of Latin phrases are provided, with a quick switch to print them in Roman letters if needed. A mechanism is needed to alter the effect of the <code>\latin</code> command <i>only</i> if the user does not have their own version. This is achieved here, with precautions taken to ensure the user can define their own <code>\latin</code> command <i>after</i> loading <code>chemstyle</code> and still have everything work properly. |
|---|--|

```
174 \newcommand*{\cst@latin}{%
175   \ifcst@abbremph
176     \expandafter\emph%
177   \fi}
178 \newcommand*{\latinemphon}{\cstsetup{abbremph=true}}
179 \newcommand*{\latinemphoff}{\cstsetup{abbremph=false}}
180 \AtBeginDocument{\providecommand*\latin{\cst@latin}}
```

| | |
|--|---|
| <pre>\etc \invacuo \etal \eg \ie</pre> | For the macros themselves, care is taken about trailing full stops. The <code>\xspace</code> command deals with any problems of spacing. Things could go wrong with complex punctuation, as no other checks are performed. All of these functions use <code>\providecommand</code> to avoid standing on the user’s own versions, if they exist. |
|--|---|

```
181 \ifcst@phrases
182 \AtBeginDocument{
183   \providecommand*\etc{
```

```

184     {\@ifnextchar.{\cst@etc}{\cst@etc.\cst@xspace}}
185 \providecommand*\invacuo{
186     {\latin{in vacuo}\cst@xspace}
187 \providecommand*\etal{
188     {\@ifnextchar.{\cst@etal}{\cst@etal.\cst@xspace}}
189 \providecommand*\eg{
190     {\ifcst@abbrcomma
191         \expandafter\cst@commaabbr%
192     \else
193         \expandafter\cst@nocommaabbr%
194     \fi
195     {eg}}}
196 \providecommand*\ie{
197     {\ifcst@abbrcomma
198         \expandafter\cst@commaabbr%
199     \else
200         \expandafter\cst@nocommaabbr%
201     \fi
202     {ie}}}
```

`\cst@etal` Internal macros are used for items ending in a full stop, to allow clean handling of spacing. Notice that `\ie` and `\eg` cannot come at the end of a sentence, they are designed to give only an inter-word space.

```

\cst@etc
\cst@eg
\cst@ie 203 \newcommand*\cst@etal{\latin{et~al}}
204 \newcommand*\cst@etc{\latin{etc}}
205 \newcommand*\cst@ie{\latin{i.e\spacefactor999\relax}}
206 \newcommand*\cst@eg{\latin{e.g\spacefactor999\relax}}
```

`\cst@commaabbr` To handle the comma issue, two general macros are used.

```

\cst@nocommaabbr 207 \newcommand*\cst@commaabbr[1]{%
208     \csname cst@#1\endcsname%
209     \cst@addpunct}
210 \newcommand*\cst@nocommaabbr[1]{%
211     \@ifnextchar.%
212     {\csname cst@#1\endcsname}
213     {\csname cst@#1\endcsname.\cst@xspace}}
```

`\cst@addpunct` The following macros are very closely based on those in the `cite` package used for moving citations after punctuation. The first macro is used as an initial hook. Notice that `\relax` is essential here, as it provides an argument for `\cst@add@punct` in the first round of checking.

```

214 \newcommand*\cst@addpunct{%
215     \cst@add@punct\relax}
```

`\cst@add@punct` Here, a plain `TeX` `\futurelet` is used to test the next character. Notice that this macro takes a single argument, which is used to recursively gobble up punctuation.

```

216 \newcommand*\cst@add@punct[1]{%
217     \futurelet\cst@tempa\cst@add@punct@}
```

`\cst@add@punct@` The checking occurs here. If a match is made, then the process is repeated to allow the punctuation to be gobbled.

```

\cst@tempa
\cst@tempb 218 \newcommand*\cst@add@punct@{%
```

```

219 \ifx.\cst@tempa
220 \let\cst@tempb\cst@add@punct
221 \else
222 \ifx,\cst@tempa
223 \let\cst@tempb\cst@add@punct

```

No match, and so new punctuation is to be added and the loop ended.

```

224 \else
225 \let\cst@tempb\cst@commapunct
226 \fi
227 \fi
228 \cst@tempb}

```

`\cst@commapunct` The `\cst@punct` macro holds the comma-containing punctuation to be added.

```

229 \newcommand*{\cst@commapunct}{.,\cst@xspace}

```

7.9 Loading the style definition

The style definition is loaded here, once everything necessary is in place. A style must be loaded, so a default is provided to be on the safe side. The journal style file must have extension `.jdf`.

```

230 \InputIfFileExists{\cst@journal.jdf}
231 {\PackageInfo{chemstyle}
232 {Loaded \cst@journal.jdf}}
233 {\PackageWarning{chemstyle}
234 {Requested style '\cst@journal' does not exist\MessageBreak
235 Loading default style}
236 \InputIfFileExists{none.jdf}}{}
237 {\PackageError{chemstyle}{Failed to load none.jdf}
238 {The default journal definition file was not
239 found\MessageBreak This is a core part of
240 chemstyle\MessageBreak Something is wrong with
241 the installation}}

```

7.10 Handling reference section naming

`\cst@refsec` To enable the renaming of `\bibname` or `\refname` to the desired value, some work is needed. Both `natbib` and `biblatex` alter the behaviour here subtly. The `babel` package also requires careful handling. The initial step is to find whether the document class defines chapters, as this affects whether `\refname` or `\bibname` should be changed. This is done inside a group so that a false definition is not left around.⁵ A second check is needed, in case the document class does not define any reference section at all (for example, `minimal` does not).

```

242 \begingroup
243 \ifundefined{chapter}
244 {\ifundefined{refname}
245 {\PackageInfo{chemstyle}
246 {No bibliography name command found}
247 \cstsetup{notes=false}}
248 {\gdef\cst@refsec{\refname}}}

```

⁵If `\@undefined` is ever fixed, this will no longer be needed. If ϵ -TeX was required, the `\ifdefined` would be a choice here; however, ϵ -TeX is not going to be required just for this.

```

249     {\gdef\cst@refsec{\bibname}}
250 \endgroup

\c@bibnote The business-end of the work is only carried out if the notes option is set to
true or auto. The check above means that a reference section is also defined.
First, some checking for the auto option occurs; the counter used by notes2bib
needs to be defined even if the package is not loaded.

251 \ifcst@autonotes
252   \AtBeginDocument{
253     \ifpackageloaded{notes2bib}{}
254     {\newcounter{bibnote}}
255 \fi

\cst@refname The internal macros to actually display the reference section name deals with the
\cst@ref@name various options. The \expandafters are for caution here: nothing should go
\cst@ref@name@ wrong with the nested \if statements.

256 \newcommand*{\cst@refname}{%
257   \ifcst@notes
258     \ifcst@autonotes
259       \ifnum\the\value{bibnote}>\z@
260         \expandafter\expandafter\expandafter\expandafter
261         \expandafter\expandafter\expandafter\cst@ref@name
262       \else
263         \expandafter\expandafter\expandafter\expandafter
264         \expandafter\expandafter\expandafter\cst@ref@name@
265       \fi
266     \else
267       \expandafter\expandafter\expandafter\cst@ref@name
268     \fi
269   \else
270     \expandafter\cst@ref@name@
271   \fi}
272 \newcommand*{\cst@ref@name}[3]{%
273   \ifcst@notesbefore
274     #3\space#2\space#1%
275   \else
276     #1\space#2\space#3%
277   \fi}
278 \newcommand*{\cst@ref@name@}[3]{#1}

\cst@setrefname Setting up the section name is a tricky affair. If no section is defined, then
the whole process is skipped. Otherwise, the first check made s if the default
language matches the definition given. If so, the heading is changed now.

279 \@ifundefined{cst@refsec}
280   {\let\cst@setrefname\@gobblefour}
281   {\newcommand*{\cst@setrefname}[4]{
282     \edef\cst@tempa{#1}%
283     \ifx\cst@tempa\cst@language
284       \expandafter\renewcommand\expandafter*\expandafter{%
285         \cst@refsec}
286       {\cst@refname{#2}{#3}{#4}}
287     \fi

```

cst@tempa At the beginning of the document, a check needs to be made for babel and biblatex. For the later, a low-level hack is made to attach to the mechanism. For babel the appropriate commands are added to the `\captions...` macro.

```

288 \AtBeginDocument{
289   \ifpackageloaded{biblatex}
290     {\def\cst@tempa{\csgappto{bib@strings@#1}}
291      \expandafter\cst@tempa\expandafter{\expandafter
292        \renewcommand\expandafter*\expandafter{%
293          \cst@refsec}{%
294            \cst@refname{#2}{#3}{#4}}}}
295   {\ifpackageloaded{babel}
296     {\def\cst@tempa{\expandafter\addto\expandafter{%
297       \csname captions#1\endcsname}}
298      \expandafter\cst@tempa\expandafter
299        {\expandafter\renewcommand\expandafter*%
300          \expandafter{\cst@refsec}{%
301            \cst@refname{#2}{#3}{#4}}}}

```

To make sure the changes are applied *now*, the current language is re-selected.

```

302   \expandafter\selectlanguage\expandafter{%
303     \language}
304   {}

```

Default names are loaded for a range of languages. Most journals stick to the same words, with only the order changing.

```

305 \cst@setrefname{english}{References}{and}{Notes}
306 \cst@setrefname{UKenglish}{References}{and}{Notes}
307 \cst@setrefname{ngerman}{Literatur}{und}{Notizen}
308 \cst@setrefname{german}{Literatur}{und}{Notizen}
309 \cst@setrefname{french}{R\'{e}f\'{e}rences}{et}{Notes}
310 \cst@setrefname{frenchb}{R\'{e}f\'{e}rences}{et}{Notes}

```

8 Configuration files

Each journal style needs slightly differing commands to get the formatting just right. This is handled here, with each style in a separate file. There is not a lot happening in most of these files, as the information is by its nature quite repetitive.

8.1 RSC style

```

311 \ProvidesFile{rsc.jdf}
312 [\cst@ver RSC journal style]
313 \AtBeginDocument{%
314   \renewcommand{\figurename}{Fig.}}
315 \captionsetup{labelsep=quad,labelfont=bf}
316 \setkeys{cst}{opt}{
317   radprefix=true,
318   rademph=true,
319   radsuper=false,
320   radhyphen=true,
321   abbrevemph=true,

```



```
322 abbrcomma=false}
```

8.2 *Angew. Chem.* style

```
323 \ProvidesFile{angew.def}
324 [\cst@ver Angew. Chem. journal style]
325 \captionsetup{labelsep=period,labelfont={bf,it},font=sf,
326   singlelinecheck=off}
327 \captionsetup[table]{labelsep=colon}
328 \setkeys[cst]{opt}{
329   radprefix=true,
330   rademph=true,
331   radsuper=false,
332   radhyphen=false,
333   abbremph=false,
334   abbrcomma=false}
```

8.3 *J. Organomet. Chem.* style

```
335 \ProvidesFile{jomc.def}
336 [\cst@ver J. Organomet. Chem. journal style]
337 \captionsetup{labelsep=period}
338 \captionsetup[table]{labelsep=newline,singlelinecheck=off}
339 \AtBeginDocument{%
340   \renewcommand{\figurename}{Fig.}}
341 \setkeys[cst]{opt}{
342   radprefix=true,
343   rademph=true,
344   radsuper=true,
345   radhyphen=false,
346   abbremph=false,
347   abbrcomma=false,
348   notesbefore=true}
```

8.4 *Tetrahedron Lett.* style

```
349 \ProvidesFile{tetlett.def}
350 [\cst@ver Tetrahedron Lett. journal style]
351 \captionsetup{labelsep=period,singlelinecheck=off,labelfont=bf}
352 \setkeys[cst]{opt}{
353   radprefix=true,
354   rademph=true,
355   radsuper=false,
356   radhyphen=true,
357   abbremph=false,
358   abbrcomma=true,
359   notesbefore=true}
```

8.5 *J. Am. Chem. Soc.* style

```
360 \ProvidesFile{jacs.jdf}
361 [\cst@ver J. Am. Chem. Soc. journal style]
362 \DeclareCaptionLabelSeparator{perquad}{.\quad}
363 \captionsetup{labelfont={bf,it,sf},textfont=sf,
364   labelsep=perquad}
365 \captionsetup[figure]{textfont=rm}
366 \captionsetup{singlelinecheck=off}
```

```

367 \setkeys[cst]{opt}{
368   radprefix=true,
369   rademph=true,
370   radsuper=true,
371   radhyphen=false,
372   abbrevemph=false,
373   abbrcomma=true,
374   notesbefore=true}
375 \floatstyle{plaintop}
376 \restylefloat{scheme}
377 \floatstyle{plain}
378 \cst@notesbeforefalse

```

8.6 *Inorg. Chem.* style

Almost exactly the same as for *J. Am. Chem. Soc.*, so most of the work is left to `jacs.jdf`.

```

379 \ProvidesFile{ic.jdf}
380 [\cst@ver Inorg. Chem. journal style]
381 \input {jacs.jdf}
382 \captionsetup{textfont=rm}

```

8.7 *J. Phys. Chem.* style

```

383 \ProvidesFile{jpc.jdf}
384 [\cst@ver J. Phys. Chem. journal style]
385 \DeclareCaptionFormat{labelcaps}{\MakeUppercase{#1}#2#3}
386 \captionsetup{font=bf,labelsep=colon,format=labelcaps}
387 \captionsetup[figure]{format=plain,textfont=md,labelsep=period}
388 \captionsetup{singlelinecheck=off}
389 \setkeys[cst]{opt}{
390   radprefix=true,
391   rademph=true,
392   radsuper=false,
393   radhyphen=false,
394   abbrevemph=false,
395   abbrcomma=true,
396   notesbefore=true}
397 \floatstyle{plaintop}
398 \restylefloat{scheme}
399 \floatstyle{plain}
400 \cst@notesbeforefalse

```

8.8 *Org. Lett.* style

```

401 \ProvidesFile{orglett.jdf}
402 [\cst@ver Org. Lett. journal style]
403 \RequirePackage{xcolor}

```

Currently, this style only works with float.

```

404 \@ifpackageloaded{floatrow}
405   {\PackageError{chemstyle}
406     {Org. Lett. style only works with float}
407     {Load chemstyle with the floatrow=false option
408       to\MessageBreak use the orglett style}}

```

```

409 {}
\OrgLettColour To allow the user to control the colour of the “bars” in this style, an additional
macro is provided. The purple colour looks about right for matching the printed
journal, but if anyone has a better suggestion please let the package author know.
410 \newcommand*{\OrgLettColour}{purple}

\fs@orglett In order to get the distinctive coloured bars used by Org. Lett., a new style for
floats is needed. This is based on the ruled style from the float package.
411 \newcommand*{\fs@orglett}{%
412   \def\@fs@cfont{\bfseries}%
413   \let\@fs@capt\floatc@ruled
414   \def\@fs@pre{\begingroup\color{\OrgLettColour}
415     \hrule height12pt depth0pt \kern2pt\endgroup}%
This is the same \@fs@mid as is used in the float package for plaintop floats.
416   \def\@fs@mid{\vspace\belowcaptionskip\relax}%
417   \def\@fs@post{\begingroup\color{\OrgLettColour}\kern2pt
418     \hrule height1.5pt depth0pt\endgroup}%
419   \let\@fs@iftopcapt\iftrue}

\fs@orglettfig Figures need to be slightly different, so an almost identical command is needed.
420 \newcommand*{\fs@orglettfig}{%
421   \def\@fs@cfont{\bfseries}%
422   \let\@fs@capt\floatc@plain
423   \def\@fs@pre{\begingroup\color{\OrgLettColour}%
424     \hrule height12pt depth0pt \kern2pt\endgroup}%
425   \def\@fs@mid{\vspace\abovecaptionskip\relax}%
426   \def\@fs@post{\begingroup\color{\OrgLettColour}\kern2pt
427     \hrule height1.5pt depth0pt\endgroup}%
428   \let\@fs@iftopcapt\iffalse}

The new style is now applied. Users can change back to normal floats by changing
back to the plain style.
429 \floatstyle{orglettfig}
430 \restylefloat{figure}
431 \floatstyle{orglett}
432 \restylefloat{scheme}
433 \restylefloat{table}

The more usual style commands now occur.
434 \DeclareCaptionLabelSeparator{perquad}{.\quad}
435 \captionsetup{labelfont=bf,labelsep=perquad}
436 \setkeys{cst}{opt}{
437   radprefix=true,
438   rademph=true,
439   radsuper=true,
440   radhyphen=false,
441   abbrevemph=false,
442   abbrcomma=true,
443   notesbefore=true}
444 \cst@notesbeforefalse

```

8.9 The empty style — none

To allow the user to load the extra macros provided here without any style changes, a “do nothing” style is provided. It simply makes sure that very little changes compared to the L^AT_EX kernel. This requires undoing the defaults provided above. For commands where a default is needed (*e.g.* the `\latin` command) the style of the RSC is followed. As the float package has been loaded, notice that captions will be placed below floats even if the `\caption` command appears above the contents of the floating environment.

```

445 \ProvidesFile{none.jdf}
446 [\cst@ver Empty journal style]
447 \floatstyle{plain}
448 \restylefloat{table}
449 \labelformat{figure}{#1}
450 \labelformat{table}{#1}
451 \floatplacement{table}{tbp}
452 \floatplacement{figure}{tbp}
453 \AtBeginDocument{
454   \floatplacement{scheme}{tbp}
455   \labelformat{scheme}{#1}}

```

9 Change History

| | | | | |
|-------|--|----|----------------------------------|--|
| v1.0 | General: Initial release of package . . . | 1 | \cst@add@punct@: New macro . . . | 13 |
| v1.0a | General: Added <i>J. Phys. Chem.</i> style | 18 | \cst@addpunct: New macro . . . | 13 |
| | Added <i>Org. Lett.</i> style | 18 | \cst@latin: New macro | 12 |
| | Fixed incorrect Latin formatting | | \cst@varnothing: New macro . . . | 10 |
| | for <i>angew</i> option | 17 | \eg: Adds comma for ACS journals | 12 |
| | No longer load <i>fixltx2e</i> pack- | | \ie: Adds comma for ACS journals | 12 |
| | age | 8 | | |
| v1.0b | General: Added <i>microtype</i> and | | v1.1a | General: Fixed error in documenta- |
| | <i>notes2bib</i> to template | 1 | | tion compilation under L ^A T _E X . . . |
| | Fixed formatting of alkyls | 16 | v1.1b | General: Load caption even when |
| | Sorted a problem with <i>babel</i> | | | journal style does not exist . . . |
| | and figure name format | 16 | v1.1c | General: Added <i>Slunits</i> option |
| v1.1 | General: Added <i>phrases</i> option . . | 1 | | Added <i>frenchb</i> alias for |
| | Added <i>siunits</i> option | 1 | | <i>french</i> |
| | Added <i>xspace</i> option | 1 | | Added <i>german</i> alias for |
| | Fixed (another) error in alkyl for- | | | <i>ngerman</i> |
| | matting | 16 | | Documentation improved |
| | Fixed error with spacing after | | | Fixed problems with capitalisa- |
| | <i>e.g.</i> and <i>i.e.</i> | 1 | | tion of <i>Slunits</i> |
| | License changed from GPL to LPPL | 1 | | \standardstate: Altered angle of |
| | Phrases modified to better avoid | | | rotation for better appearance . . |
| | clash with user’s own com- | | | Definition only occurs if user does |
| | mands | 1 | v1.1d | not have own version |
| | \cst@add@punct: New macro . . | 13 | | General: Require caption in all cases |
| | | | | 8 |

| | | | | | |
|-------|---|----|--|---|----|
| v1.1e | General: Fixed packaging problem . . . | 1 | \cst@commapunct: Name change from \cst@punct | 14 | |
| v1.1f | \cst@addpunct: No longer used when comma not required . . . | 13 | \cst@latin: Change of definition to allow better working with run-time options | 12 | |
| | \cst@eg: New macro | 13 | \cst@nocommaabbr: New macro | 13 | |
| | \cst@etal: New macro | 13 | \cst@radical: Re-write to allow definition to change at run-time | 11 | |
| | \cst@etc: New macro | 13 | \cst@ref@name: New macro | 15 | |
| | \cst@ie: New macro | 13 | \cst@ref@name@: New macro | 15 | |
| | \eg: Definition comma-mode dependent | 12 | \cst@refname: New macro | 15 | |
| | Fixed problem with full stop spacing | 12 | \cst@refsec: New macro | 14 | |
| | \etal: Fixed problem with full stop spacing | 12 | \cst@setrefname: New macro | 15 | |
| | \ie: Definition comma-mode dependent | 12 | \cst@tempa: New macro | 8 | |
| | Fixed problem with full stop spacing | 12 | \cst@tempb: New macro | 8 | |
| v1.1g | | | \cst@xspace: New definition | 11 | |
| | General: Added jomc style | 17 | \eg: Dynamic definition for comma mode | 12 | |
| | Fixed problem with cross-referencing when babel is not loaded | 12 | \ie: Dynamic definition for comma mode | 12 | |
| v1.1h | | | \latinemphoff: New definition to work with xkeyval system | 12 | |
| | General: Added tetlett style | 17 | \latinemphon: New definition to work with xkeyval system | 12 | |
| v1.2 | | | v1.2a | General: Fixed error with restyling scheme floats, caused by change in chemscheme | 1 |
| | General: Added xkeyval interface | 8 | v1.2b | General: Change due to name change of si package to siunitx | 10 |
| | Re-write of reference-section re-naming code | 14 | v1.2c | General: Altered call to siunitx function | 10 |
| | Re-write of units section | 10 | v1.4 | General: Default to loading siunitx | 10 |
| | Removed obsolete ch and chapter options | 1 | | | |
| | Various internal tweaks | 1 | | | |
| | \cst@add@punct@: Internal tweaks | 13 | | | |
| | \cst@commaabbr: New macro | 13 | | | |

10 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.

| | |
|----------------------------------|----------------------------|
| A | |
| abbrcomma (option) | 5 |
| abbremph (option) | 5 |
| C | |
| \c@bibnote | <u>251</u> |
| chemcompounds (option) | 3 |
| \cmc | <u>4</u> , <u>100</u> |
| \cst@add@punct | 215, <u>216</u> , 220, 223 |
| \cst@add@punct@ | 217, <u>218</u> |
| \cst@addpunct | 209, <u>214</u> |
| \cst@autonotesfalse | 41 |
| \cst@autonotesttrue | 45 |
| \cst@boolkey | <u>23</u> , 25–35 |
| \cst@commaabbr | 191, 198, <u>207</u> |
| \cst@commapunct | 225, <u>229</u> |
| \cst@disablekey | <u>85</u> |

| | | | |
|--|---|---|--|
| <code>\cst@eg</code> | <u>203</u> | <code>\ifcst@chemcompounds</code> | <u>25</u> , <u>77</u> |
| <code>\cst@emph</code> | <u>130</u> , <u>146</u> , <u>148</u> | <code>\ifcst@floatrow</code> | <u>25</u> , <u>80</u> |
| <code>\cst@etal</code> | <u>188</u> , <u>203</u> | <code>\ifcst@notes</code> | <u>36</u> , <u>257</u> |
| <code>\cst@etc</code> | <u>184</u> , <u>203</u> | <code>\ifcst@notesbefore</code> | <u>25</u> , <u>273</u> |
| <code>\cst@hyphen</code> | <u>130</u> , <u>146</u> , <u>148</u> | <code>\ifcst@phrases</code> | <u>25</u> , <u>181</u> |
| <code>\cst@id</code> | <u>13</u> | <code>\ifcst@rademph</code> | <u>25</u> , <u>140</u> |
| <code>\cst@ie</code> | <u>203</u> | <code>\ifcst@radhyphen</code> | <u>25</u> , <u>130</u> |
| <code>\cst@journal</code> | <u>59</u> , <u>230</u> , <u>232</u> , <u>234</u> | <code>\ifcst@radprefix</code> | <u>25</u> , <u>145</u> |
| <code>\cst@language</code> | <u>59</u> , <u>283</u> | <code>\ifcst@radsuper</code> | <u>25</u> , <u>135</u> |
| <code>\cst@latin</code> | <u>174</u> | <code>\ifcst@xspace</code> | <u>25</u> , <u>152</u> |
| <code>\cst@nocommaabbr</code> | <u>193</u> , <u>200</u> , <u>207</u> | <code>\invacuo</code> | <u>5</u> , <u>181</u> |
| <code>\cst@notesbeforefalse</code> | <u>378</u> , <u>400</u> , <u>444</u> | <code>\iPr</code> | <u>4</u> , <u>155</u> |
| <code>\cst@notesfalse</code> | <u>40</u> | | |
| <code>\cst@notesttrue</code> | <u>44</u> , <u>49</u> | J | |
| <code>\cst@radical</code> | <u>127</u> , <u>155</u> – <u>160</u> | <code>journal (option)</code> | <u>2</u> , <u>3</u> |
| <code>\cst@ref@name</code> | <u>256</u> | | |
| <code>\cst@ref@name@</code> | <u>256</u> | L | |
| <code>\cst@refname</code> | <u>256</u> , <u>286</u> , <u>294</u> , <u>301</u> | <code>\latin</code> | <u>5</u> , <u>174</u> , <u>186</u> , <u>203</u> – <u>206</u> |
| <code>\cst@refsec</code> | <u>242</u> , <u>285</u> , <u>293</u> , <u>300</u> | <code>\latinemphoff</code> | <u>5</u> , <u>174</u> |
| <code>\cst@setrefname</code> | <u>279</u> , <u>305</u> – <u>310</u> | <code>\latinemphon</code> | <u>5</u> , <u>174</u> |
| <code>\cst@super</code> | <u>130</u> , <u>146</u> , <u>148</u> | | |
| <code>\cst@tempa</code> | <u>21</u> , <u>36</u> , <u>217</u> , <u>218</u> , <u>282</u> , <u>283</u> , <u>288</u> , <u>290</u> , <u>291</u> , <u>296</u> , <u>298</u> | M | |
| <code>\cst@tempb</code> | <u>21</u> , <u>36</u> , <u>218</u> | <code>\mmHg</code> | <u>4</u> , <u>100</u> |
| <code>\cst@varnothing</code> | <u>122</u> | <code>\mol</code> | <u>100</u> |
| <code>\cst@ver</code> | <u>13</u> , <u>312</u> , <u>324</u> , <u>336</u> , <u>350</u> , <u>361</u> , <u>380</u> , <u>384</u> , <u>402</u> , <u>446</u> | <code>\Molar</code> | <u>4</u> , <u>100</u> |
| <code>\cst@xspace</code> | <u>150</u> , <u>151</u> , <u>184</u> , <u>186</u> , <u>188</u> , <u>213</u> , <u>229</u> | <code>\molar</code> | <u>4</u> , <u>100</u> |
| <code>\cstsetup</code> | <u>3</u> , <u>96</u> , <u>178</u> , <u>179</u> , <u>247</u> | | |
| <code>\cubiccentimeter</code> | <u>100</u> | N | |
| | | <code>\nBu</code> | <u>4</u> , <u>155</u> |
| E | | <code>nonotes (option)</code> | <u>3</u> |
| <code>\eg</code> | <u>5</u> , <u>181</u> | <code>nophrases (option)</code> | <u>5</u> |
| <code>\etal</code> | <u>5</u> , <u>181</u> | <code>notes (option)</code> | <u>3</u> |
| <code>\etc</code> | <u>5</u> , <u>181</u> | <code>notesbefore (option)</code> | <u>4</u> |
| | | <code>\nPr</code> | <u>4</u> , <u>155</u> |
| F | | O | |
| <code>floatrow (option)</code> | <u>3</u> | <code>options:</code> | |
| <code>\fs@orglett</code> | <u>411</u> | <code>abbrcomma</code> | <u>5</u> |
| <code>\fs@orglettfig</code> | <u>420</u> | <code>abbremph</code> | <u>5</u> |
| | | <code>chemcompounds</code> | <u>3</u> |
| H | | <code>floatrow</code> | <u>3</u> |
| <code>\Hz</code> | <u>4</u> , <u>100</u> | <code>journal</code> | <u>2</u> , <u>3</u> |
| | | <code>nonotes</code> | <u>3</u> |
| I | | <code>nophrases</code> | <u>5</u> |
| <code>\iBu</code> | <u>4</u> , <u>155</u> | <code>notes</code> | <u>3</u> |
| <code>\ie</code> | <u>5</u> , <u>181</u> | <code>notesbefore</code> | <u>4</u> |
| <code>\ifcst@abbrcomma</code> | <u>25</u> , <u>190</u> , <u>197</u> | <code>rademph</code> | <u>4</u> |
| <code>\ifcst@abbremph</code> | <u>25</u> , <u>175</u> | <code>radhyphen</code> | <u>4</u> |
| <code>\ifcst@autonotes</code> | <u>36</u> , <u>251</u> , <u>258</u> | <code>radprefix</code> | <u>4</u> |
| | | <code>radsuper</code> | <u>4</u> |
| | | <code>xspace</code> | <u>4</u> |
| | | <code>\OrgLettColour</code> | <u>410</u> |

| | | | |
|---------------------------------------|---|------------------------------------|---|
| | R | <code>\standardstate</code> | 4 , 122 |
| <code>rademph</code> (option) | 4 | | |
| <code>radhyphen</code> (option) | 4 | T | |
| <code>radprefix</code> (option) | 4 | <code>\tBu</code> | 4 , 155 |
| <code>radsuper</code> (option) | 4 | | |
| | S | X | |
| <code>\sBu</code> | 4 , 155 | <code>xspace</code> (option) | 4 |