MYCHEMISTRY

v1.99b 2012/12/19

Create Reaction Schemes with LaTeX 2ε and Chemfig

Clemens Niederberger

https://bitbucket.org/cgnieder/mychemistry/
contact@mychemistry.eu

English documentation

Contents

1 Licence and Requirements

Permission is granted to copy, distribute and/or modify this software under the terms of the LaTeX Project Public License, version 1.3 or later (http://www.latex-project.org/lppl.txt). This package has the status "maintained."

MYCHEMISTRY needs and loads the packages etoolbox,[1] float,[2] xkeyval,[3] chemmacros[4] and chemfig.[5] It also loads the TikZ-libraries `arrows`, `positioning`, `decorations.pathmorphing`, `shapes`, `calc`, `matrix`, `chains`, `scopes` and `intersections`.

MYCHEMISTRY also loads translations[6] for proper language support.

2 Changes

With v1.99 (=v2.0beta) package dependencies have changed so that MYCHEMISTRY loads noticably less packages than before. Most important for you: it does *not* load mhchem[7] any more. This documentation uses chemmacros instead which is loaded by MYCHEMISTRY.

There are now considerably less package options as most of them are not needed any more.

MYCHEMISTRY now provides proper language support, e.g., when used together with babel.[8]

New is the command `\setelmove`. The command `\makevisible` now only has an effect if the new option `draft` is used.

3 Background

MYCHEMISTRY provides two environments within which the mechanisms are created. Both environments basically are `tikzpicture` environments. When this package first was written chemfig lacked the possibility of creating reaction schemes itself. This was the motivation for me to get started on MYCHEMISTRY which then rapidly grew and until now has its users.

The background has changed in August 2011, though. Then chemfig v1.0 was published and now provides simple but powerful macros to create reactions schemes. They don't seem to to be used very often but I am advertising as much as I can. Anyway, since then MYCHEMISTRY can be considered obsolete and is not actively developed any more. The current release 1.99 (= 2.0beta) was due to an incompatibilty with pdfpages[9] and a complete robustification of various parts of the code.

Other than bug fixes this package will very likely not *be updated any more*!

[1] CTAN: etoolbox [2] CTAN: float [3] CTAN: xkeyval [4] CTAN: chemmacros [5] CTAN: chemfig [6] Part of the exsheets package [7] CTAN: mhchem [8] CTAN: babel [9] CTAN: pdfpages

4 Package options

MYCHEMISTRY has a few package options.

`strict` → With this option all warning messages are turned into error messages.

`draft` → This option is an alias to `strict`. Additionally it enables the `\makevisible` command.

`final` → This option is the opposite to `draft`, i.e., most error messages will be issued as warnings. It also disables the `\makevisible` command.

`placement = <value>` → Changes the default floating behaviour of `\begin{rxnscheme}`, `\end{rxnscheme}` see section 7.15.

5 Usage

5.1 Basic Principle

Within the `tikzpicture` reactants and arrows are placed as nodes on a `chain`.[10]

```
\begin{tikzpicture}[start chain]
\node [on chain] {A};
\node [on chain] {B};
\node [on chain] {C};
\end{tikzpicture}
```

A B C

This way there are several possibilities to place the nodes relative to the others.

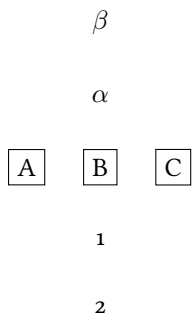
```
\begin{tikzpicture}[start chain=going right,node distance=5mm]
\node [draw,on chain] {Hello};
\node [draw,on chain] {World};
\node [draw,continue chain=going below,on chain] {,};
\node [draw,on chain] {this};
\node [draw,on chain] {is};
\end{tikzpicture}
```

Hello World , this is

Above all MYCHEMISTRY uses the possibility of creating branches to the chain.

[10] Provided by the tikzlibrary 'chains'

```
\begin{tikzpicture}[start chain=going right,node distance=5mm]
 \node [draw,on chain] {A};
 \node [draw,on chain] {B};
{ [start branch]
 \node [on chain=going below] {1};
 \node [on chain=going below] {2};
}
{ [start branch]
 \node [on chain=going above] {$\alpha$};
 \node [on chain=going above] {$\beta$};
}
 \node [draw,on chain] {C};
\end{tikzpicture}
```



You don't have to understand that mechanism in detail but you should remember the placement commands in the last example, because MYCHEMISTRY uses them in the same way.

In some of the examples in this documentation the nodes are boxed with a coloured frame (see section 7.8). This is done so one can see, which size they have and which impact changes of the alignment have on them.

5.2 How does it work?

5.2.1 Basic Commands

There are two basic commands:

`\reactant[<pos>,<name>,<tikz>]{<formula>}`

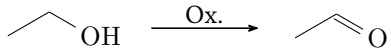
`\arrow[<pos>,<type>,<length factor>,<name>,both,<tikz>]{<above>}{<below>}`

Schemes are created within the `\begin{rxn} \end{rxn}` environment. There you place reactants and arrows.

```
\begin{rxn}
 \reactant{ \chemfig{-[::30]-[::-60]OH} }
 \arrow{Ox.}{}
 \reactant{ \chemfig{-[::30]=_[::-60]O} }
\end{rxn}
```

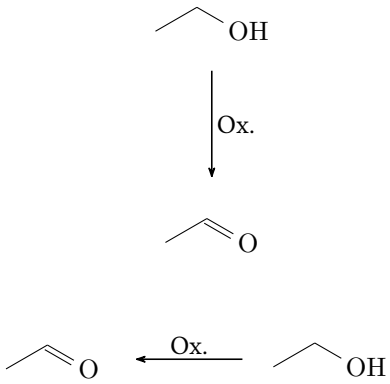
key word	pos. angle	neg. angle
`right`	0	± 360
`right above`	45	-315
`above`	90	-270
`above left`	135	-225
`left`	180	-180
`below left`	225	-135
`below`	270	-90
`below right`	315	-45

Table 1: key words for positioning



When you don't use options reactants and arrows always are put to the right of the last object. With the option <pos> you can change this behaviour.

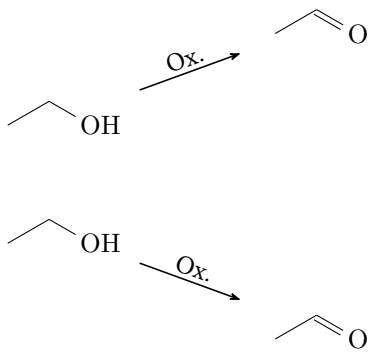
```
 % positioning using key words
\begin{rxn}
 \reactant{ \chemfig{-[::30]-[::-60]OH} }
 \arrow[below]{Ox.}{}
 \reactant[below]{ \chemfig{-[::30]=_[::-60]O} }
\end{rxn}
% positioning using an angle
\begin{rxn}
 \reactant{ \chemfig{-[::30]-[::-60]OH} }
 \arrow[180]{}{Ox.}
 \reactant[180]{ \chemfig{-[::30]=_[::-60]O} }
\end{rxn}
```



You can see in the last example that positioning can be realized through key words (see table 1) like `below` or by using the angle relative to the horizontal line. Every angle from the

interval $[-360°; 360°]$. 0° corresponds to `right` which is the default value. *Positive angles* mean a turn *counter clockwise*, negative ones a turn clockwise – like you're used to in mathematics.

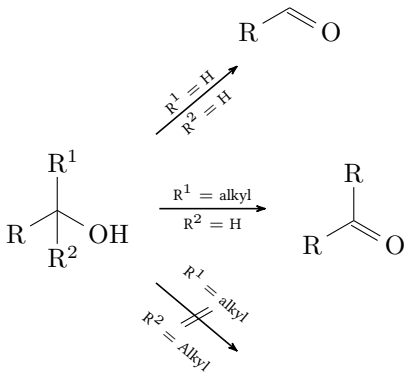
```
\begin{rxn}
 \reactant{ \chemfig{-[::30]-[::-60]OH} }
 \arrow[20]{Ox.}{}
 \reactant[20]{ \chemfig{-[::30]=_[::-60]O} }
\end{rxn}
\begin{rxn}
 \reactant{ \chemfig{-[::30]-[::-60]OH} }
 \arrow[-20]{Ox.}{}
 \reactant[-20]{ \chemfig{-[::30]=_[::-60]O} }
\end{rxn}
```



5.2.2 Positioning

Reactants and arrows cannot only be positioned through key words and angles. They can refer to another reactant or arrow.

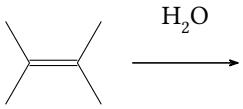
```
\begin{rxn}
 \reactant[,start]{ \chemfig{R-[::30](-[::60]R|^1) (-[::-120]R|^2)
 -[::-60]OH} }
 \arrow[40]{\tiny$\text{R}^1=\text{H}$} {\tiny$\text{R}^2=\text{H}$}
 \reactant[40]{ \chemfig{R-[::30]=_[::-60]O} }
 \arrow[start.0]{\tiny$\text{R}^1=\text{alkyl}$} {\tiny$\text{R}^2=\text{
 H}$}
 \reactant{ \chemfig{R-[::30](-[::60]R)=_[::-60]O} }
 \arrow[start.-40,-|>]{\tiny$\text{R}^1=\text{alkyl}$} {\tiny$\text{R
 }^2=\text{Alkyl}$}
\end{rxn}
```



In the last example the first reactant got the `<name> start`. The arrows in lines 5 and 7 now could refer to it. A *previously given name* can act as an anchor for later reactants or arrows, if the positioning is written like `<anchor>.<angle>`.

Arrows can be given names, too. The anchor point of an arrow always is in the middle of the arrow line and has *no* size.

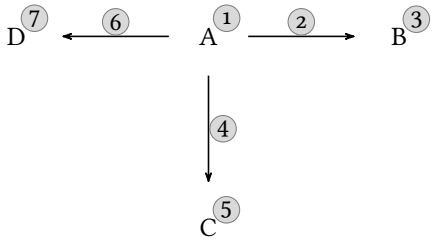
```
1  \begin{rxn}
2   \reactant{\chemfig{[:60]-(-[::60])=[::-60](-[::-60])-}}
3   \arrow[,,,arrow]{}{}
4   \reactant[arrow.90]{\ch{H2O}}
5  \end{rxn}
```



Using this kind of positioning does *not* break the chain.

```
1  \begin{rxn}
2   \reactant[,a]{A}
3   \arrow{}{}
4   \reactant{B}
5   \arrow[a.-90]{}{}
6   \reactant[-90]{C}
7   \arrow[a.180]{}{}
8   \reactant[180]{D}
9  \end{rxn}
```

All seven objects of this example are logically speaking part of the same chain. The next object is placed to the right of the last one if no positioning is used.



5.2.3 Branches

To break a chain you use the command

`\branch[<pos>,<name>,<tikz>]{<formulae>}`

The positioning of a branch is slightly different from earlier objects, although the syntax is similar. A branch has two additional ways of positioning. Every positioning that refers to an anchor cause that the branch is not a part of the chain but is a real branch.

`<angle>` on the chain

`<key>` on the chain

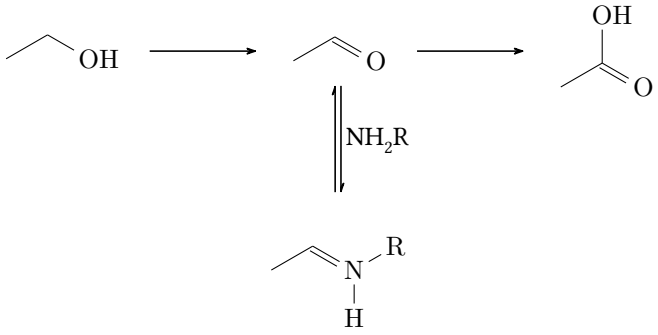
`<anchor>.<angle>` not on the chain

`on chain=going <key>` on the chain

`<key>=of <anchor>` not on the chain

As `<key>` you again use the values from table 1.

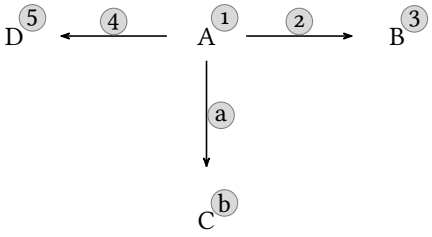
```
\begin{rxn}
 \reactant{ \chemfig{-[::30]-[::-60]OH} }
 \arrow{}{}
 \reactant[,carbonyl]{ \chemfig{-[::30]=_[::-60]O} }
 \branch[carbonyl.-90]{
   \arrow[-90,<=>]{\ch{NH2R}}{}
   \reactant[-90]{ \chemfig{-[::30]=_[::-60]N(-[6]H)-[::60]R} }
 }
 \arrow{}{}
 \reactant{ \chemfig{-[::30](-[::60]OH)=_[::-60]O} }
\end{rxn}
```



Please note that in the last example the arrow and the reactant placed after branch continue the original chain.

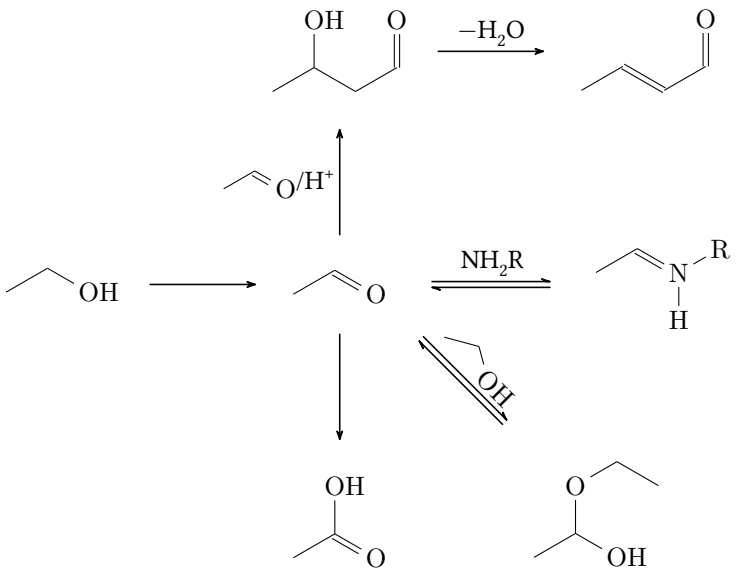
```
\begin{rxn}
 \reactant[,a]{A}
 \arrow{}{}
 \reactant{B}
 \branch[a.-90]{
   \arrow[-90]{}{}
   \reactant[-90]{C}
 }
 \arrow[a.180]{}{}
 \reactant[180]{D}
\end{rxn}
```

The chain is broken by the branch which starts a new chain itself.



Using \branch allows to create schemes with several branches:

```
\begin{rxn}
 \reactant{ \chemfig{-[::30]-[::-60]OH} }
 \arrow{}{}
 \reactant[,carbonyl]{ \chemfig{-[::30]=_[::-60]O} }
 \arrow[-90]{}{}
 \reactant[-90]{ \chemfig{-[::30](-[::60]OH)=_[::-60]O} }
 \branch[right=of carbonyl]{
   \arrow[,<=>,1.12]{\ch{NH2R}}{}
   \reactant{ \chemfig{-[::30]=_[::-60]N(-[6]H)-[::60]R} }
 }
 \branch[below right=of carbonyl]{
   \arrow[-45,<=>,1.12]{ \chemfig{[,.75]-[::30]-[::-60]OH} }{}
   \reactant[-45]{ \chemfig{-[::30](-[::60]O-[::-60]-[::-60])-[::-60]OH}
 }
 }
 \arrow[carbonyl.90]{ \chemfig{[,.75]-[::30]=_[::-60]O}/\Hpl }{}
 \reactant[90]{ \chemfig{-[::30](-[::60]OH)-[::-60]-[::60]=[::60]O} }
 \arrow{$-\ch{H2O}$}{}
 \reactant{ \chemfig{-[::30]=[::-60]-[::60]=[::60]O} }
\end{rxn}
```



5.2.4 Numbered Schemes

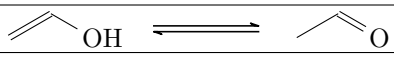
There is another environment to create schemes.

`\begin{rxnscheme}[<label>,<placement>,<alignment>,<skale factor>,<title>]` <caption>
`\end{rxnscheme}`

This is a floating environment for scheme, which can be given a `<caption>`, a `<label>` and the common placement identifiers like `htp` (default).

```
\begin{rxnscheme}{Keto-Enol-tautomerism}
 \reactant{ \chemfig{=[::30]-[::-60]OH} }
 \arrow[,<=>]{}{}
 \reactant{ \chemfig{-[::30]=[::-60]O} }
\end{rxnscheme}
```

Reaction scheme 1 Keto-Enol-tautomerism



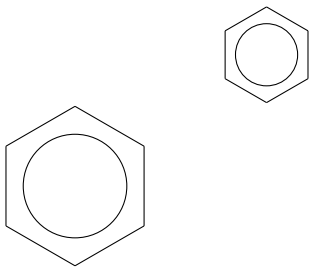
5.3 Predefined Values

There are some predefined values that are basically due to my personal taste. But of course you can change them according to your requirements. For chemfig-formulæ *inside of* MYCHEMISTRY *environments* some values are predefined as follows:

```
 \setatomsep{1.8em}
 \setcrambond{3pt}{0.5pt}{1pt}
```

Outside the MYCHEMISTRY environments the defaults of chemfig still are set.

```
\begin{rxn}
 \reactant{\chemfig{**6(------)}}
\end{rxn}
\chemfig{**6(------)}
```



MYCHEMISTRY's defaults can be changed with these commands:

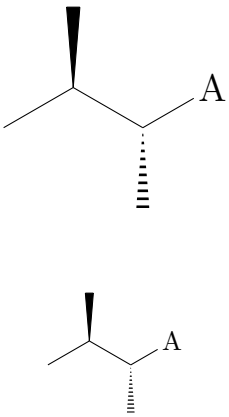
`\setbondlength{<length>}`

`\setbondshape{<base length>}{<dash thickness>}{<dash spacing>}`

`\setatomsize{<font size>}`

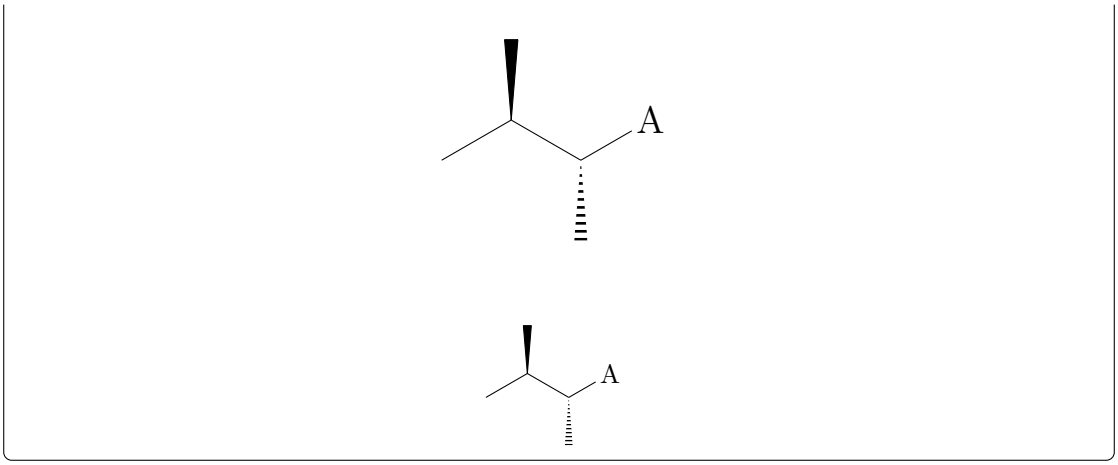
With these commands the parameters are changed *for all following* MYCHEMISTRY environments. If you leave the arguments empty default values are restored. Default for `\setatomsize` is `\small`.

```
\setbondlength{2.1em}\setbondshape{5pt}{1pt}{2pt}\setatomsize{\Large}
\begin{rxn}
 \reactant{\chemfig{-[::30](<[::60])-[::-60](<:[::-60])-[::60]A}}
\end{rxn}
\setbondlength{}\setbondshape{}{}{}\setatomsize{}
\begin{rxn}
 \reactant{\chemfig{-[::30](<[::60])-[::-60](<:[::-60])-[::60]A}}
\end{rxn}
```



If you only want to change the parameters of a single environment you can use chemfig's commands and LaTeX's fontsize commands *inside the environment.*

```
\begin{rxn}
 \setatomsep{2.1em}\setcrambond{5pt}{1pt}{2pt}\Large
 \reactant{\chemfig{-[::30](<[::60])-[::-60](<:[::-60])-[::60]A}}
\end{rxn}
\begin{rxn}
 \reactant{\chemfig{-[::30](<[::60])-[::-60](<:[::-60])-[::60]A}}
\end{rxn}
```



The default length of reaction arrows is 4 em, the default line width is `semithick` and the default label distance is 0.2 em. You can change these default values with the commands

`\setarrowlength{<length>}`

`\setarrowline{<line width>}`

`\setarrowlabel{<label distance>}`

or with

`\mCsetup{<options>}` → valid options are:

`arrowlength = <length>`

`arrowline = <line width>`

`arrowlabel = <label distance>`

6 Advanced Usage, Usage with Ti*k*Z

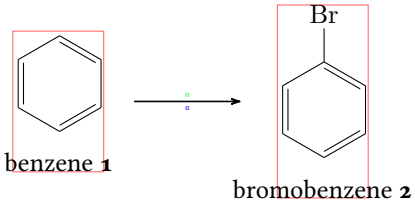
The biggest problem with MYCHEMISTRY usually is the correct positioning of reactants and arrows. Section 6.1 looks a little bit into this topic.

Some of the commands can be given Ti*k*Z code as third optional argument. More precisely you can use the same Ti*k*Z keys there as you would with a `\node` inside a `tikzpicture`. If a node is placed with `\node[<tikz>](<placement>){<anything>}` then `<tikz>` is about the same in e.g. `\reactant[,,<tikz>]{}`. With this you can customize your scheme in many ways.

6.1 The Alignment Question

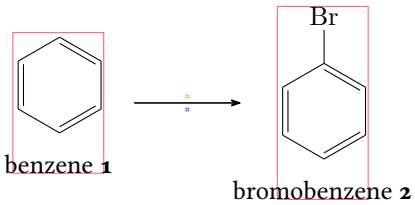
Since reactants, arrows and branches are aligned centered to the referred object the default alignment not always produces nice results.

```
\makevisible
\begin{rxn}
 \reactant{ \chemname{\chemfig{*6(-=-=-=)}}{benzene \cmpd{benzene}} }
 \arrow{}{}
 \reactant{ \chemname{\chemfig{*6(-=-=(-Br)-=)}}{bromobenzene \cmpd{
 bromobenzene}} }
\end{rxn}
```



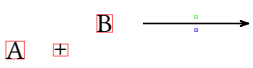
As you can see both reactants are not aligned equally to the arrow as far as the benzene ring is concerned. The first reactant seems to be shifted up. Trying to solve this with Ti*k*Z code fails:

```
\makevisible
\begin{rxn}
 \reactant[,,yshift=-1em]{ \chemname{\chemfig{*6(-=-=-=)}}{benzene \cmpd{
 benzene}} }
 \arrow{}{}
 \reactant{ \chemname{\chemfig{*6(-=-=(-Br)-=)}}{bromobenzene \cmpd{
 bromobenzene}} }
\end{rxn}
```



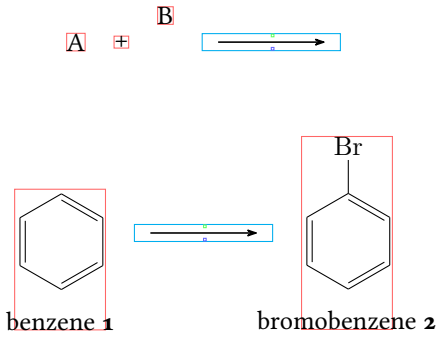
This is, because the first reactant is shifted with the respect to the object it refers to. Since it is the first object on the chain itself it isn't shifted a all. The following arrow always is centered to the object before.

```
\makevisible
\begin{rxn}
 \reactant{A}
 \chemand
 \reactant[,,yshift=1em]{B}
 \arrow{}{}
\end{rxn}
```



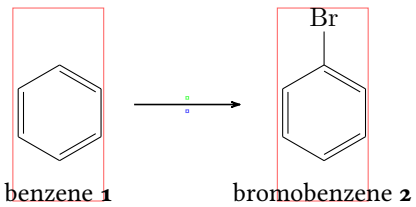
Since there is no possibility to change the alignment of the arrow itself what you can do is put it inside a branch.

```
\makevisible
\begin{rxn}
 \reactant{A}
 \chemand
 \reactant[,,yshift=1em]{B}
 \branch[,,yshift=-1em]{\arrow{}{}}
\end{rxn}
\begin{rxn}
 \reactant{ \chemname{\chemfig{*6(-=-=-=)}}{benzene \cmpd{benzene}} }
 \branch[,,yshift=1em]{\arrow{}{}}
 \reactant{ \chemname{\chemfig{*6(-=-=(-Br)-=)}}{bromobenzene \cmpd{
 bromobenzene}} }
\end{rxn}
```



For the last example this isn't the best solution though because exact alignment needs lots of tries until you get the required result. There is another solution: an invisible bromine to the first benzene.

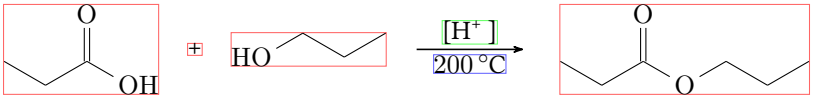
```
\makevisible
\begin{rxn}
 \reactant{ \chemname{\chemfig{*6(-=-=(-[,,,,draw=none]\phantom{Br})-=)
 }}{benzene \cmpd{benzene}} }
 \arrow{}{}
 \reactant{ \chemname{\chemfig{*6(-=-=(-Br)-=)}}{bromobenzene \cmpd{
 bromobenzene}} }
\end{rxn}
```



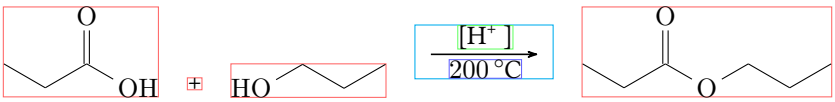
In other cases, too, an invisible substituent should be preferred over Ti*k*Z code since it's easier and more precise:

```
\makevisible
default:
\begin{rxn}
 \reactant{\chemfig{-[:-30]-[:30](=[2]O)-[:-30]OH}}
 \chemand
 \reactant{\chemfig{HO-[:30]-[:-30]-[:30]}}
 \arrow{[\Hpl]}{\SI{200}{\celsius}}
 \reactant{\chemfig{-[:-30]-[:30](=[2]O)-[:-30]O-[:30]-[:-30]-[:30]}}
\end{rxn}
hydroxy groups at the same height through TikZ:
\begin{rxn}
 \reactant{\chemfig{-[:-30]-[:30](=[2]O)-[:-30]OH}}
 \chemand[,,yshift=-1.2em]
 \reactant[,,yshift=.12em]{\chemfig{HO-[:30]-[:-30]-[:30]}}
 \branch[,,yshift=1.08em]{\arrow{[\Hpl]}{\SI{200}{\celsius}}}
 \reactant{\chemfig{-[:-30]-[:30](=[2]O)-[:-30]O-[:30]-[:-30]-[:30]}}
\end{rxn}
hydroxy groups at the same height through an invisible substituent:
\begin{rxn}
 \reactant{\chemfig{-[:-30]-[:30](=[2]O)-[:-30]OH}}
 \chemand
 \reactant{\chemfig{HO-[:30](=[2,,,,draw=none]\phantom{O})-[:-30]-[:30]}}
 \arrow{[\Hpl]}{\SI{200}{\celsius}}
 \reactant{\chemfig{-[:-30]-[:30](=[2]O)-[:-30]O-[:30]-[:-30]-[:30]}}
\end{rxn}
```

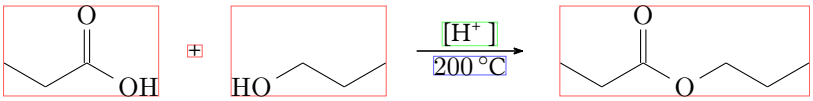
default:



hydroxy groups at the same height through TikZ:



hydroxy groups at the same height through an invisible substituent:

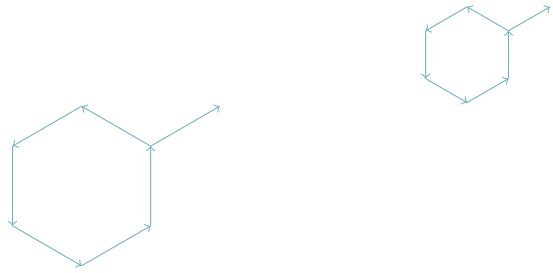


I'm afraid that in many other cases you'll have to play with `xshift` and `yshift`, though, until the scheme looks the way you want.

6.2 Using Ti*k*Z to Achieve Other Results

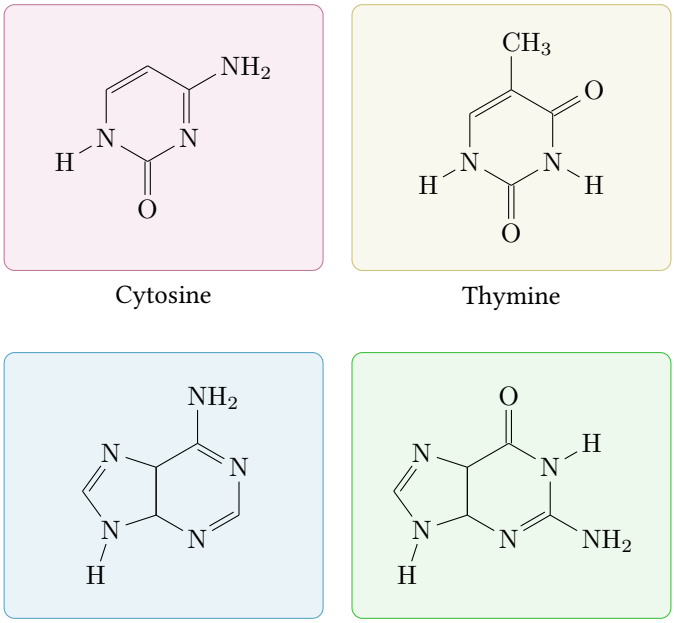
You could, just for fun?, change the looks of a molecule with Ti*k*Z.

```
\begin{rxn}
 \reactant[,,->,green!45!blue!55]{ \chemfig{*6(---(-)---)} }
\end{rxn}
\chemfig[->,green!45!blue!55]{*6(---(-)---)}
```



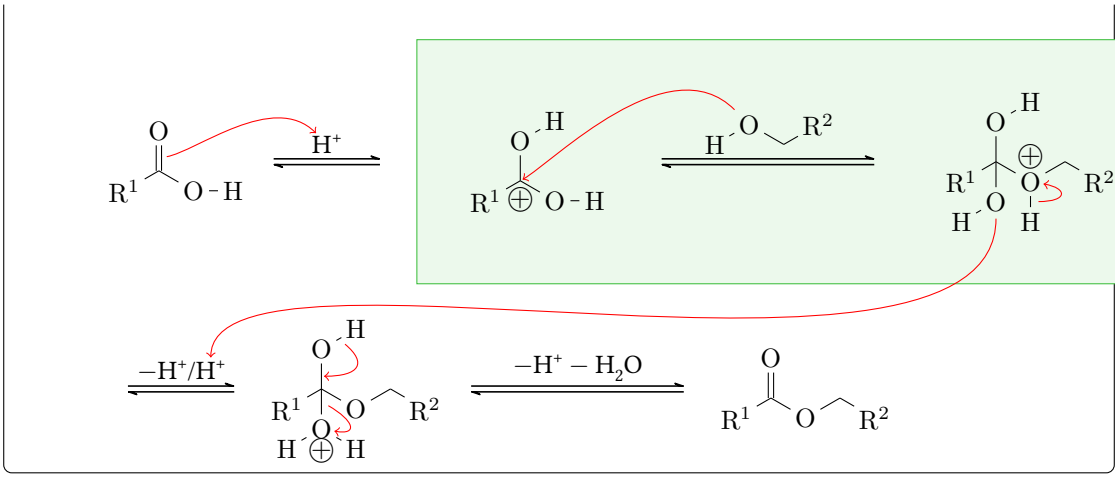
The last example is not very good, of course, since you can achieve the same result using chemfig's own possibilities. But other cases are imaginable: for example one could define a style with which reactants are shown:

```
\colorlet{mCgreen}{green!50!gray}
\colorlet{mCblue}{cyan!50!gray}
\colorlet{mCred}{magenta!50!gray}
\colorlet{mCyellow}{yellow!50!gray}
\begin{rxn}
 \tikzset{reactant/.style={draw=#1,fill=#1!10,inner sep=1em,minimum
 height=10em,minimum width=12em,rounded corners}}
 \reactant[,cytosine,reactant=mCred]{\chemfig{H-[:30]N*6(-(=O)-N=(-NH_2)
 -=-)}}
 \anywhere{cytosine.-90,,yshift=-2mm}{Cytosine}
 \reactant[,thymine,reactant=mCyellow]{\chemfig{H-[:30]N*6(-(=O)-N(-H)-(=
 O)-(-CH_3)=-)}}
 \anywhere{thymine.-90,,yshift=-2mm}{Thymine}
 \reactant[cytosine.-90,adenine,yshift=-2em,reactant=mCblue]{\chemfig
 {[:-36]*5(-N(-H)-*6(-N=-N=(-NH_2)--)--N=)}}
 \anywhere{adenine.-90,Guanin,yshift=-2mm}{Adenine}
 \reactant[,guanine,reactant=mCgreen]{\chemfig{[:-36]*5(-N(-H)-*6(-N=(-NH
 _2)-N(-H)-(=O)--)--N=)}}
 \anywhere{guanine.-90,,yshift=-2mm}{Guanine}
\end{rxn}
```



This way certain parts of a scheme could be emphasized:

```
\begin{rxn}
 \setatomsep{1.5em}
 \colorlet{mCgreen}{green!50!gray}
 \tikzset{emph/.style={draw=mCgreen,fill=mCgreen!10,inner sep=1em}}
 \reactant[,lineOne]{\chemfig{R^1-[:30](=[@{b1}2]O)-[:-30]O-H}}
 \arrow[,<=>]{\chemfig{@{Hpl1}\Hpl}}{}
 \branch[,,emph]{
   \reactant{\chemfig{R^1-[:30]@{C1}(-[2]O-[:30]H)(-[6,.5,,,draw=none]\
 fplus)-[:-30]O-H}}
   \arrow[,<=>,2]{\chemfig{[:30]H-@{O1}O-[::-60]-R^2}}{}
   \reactant{\chemfig{R^1-[:30](-[2]O-[:30]H)(-[6]@{O2}O-[:-150]H)-[:-30]
 @{O3}\chemabove{O}{\fplus}(-[@{b2}6]H)-[:30]-[:-30]R^2}}
 }
 \anywhere{lineOne.-90,lineTwo,xshift=-3em,yshift=-7em}{}
 \arrow[lineTwo.0,<=>]{$-$\Hpl/\chemfig{@{Hpl2}\Hpl}}{}
 \reactant{\chemfig{R^1-[:30](-[@{b3}2]O-[@{b4}:30]H)(-[@{b5}6]@{O4}\
 chembelow{O}{\fplus}(-[:-30]H)-[:-150]H)-[:-30]O-[:30]-[:-30]R^2}}
 \arrow[,<=>,2]{$- \ch{H+} - \ch{H2O}$}{}
 \reactant{\chemfig{R^1-[:30](=[2]O)-[:-30]O-[:30]-[:-30]R^2}}
 \anywhere{lineTwo.-90}{
   \elmove{b1}{10:5mm}{Hpl1}{135:1cm}
   \elmove{O1}{135:1.5cm}{C1}{30:5mm}
   \elmove{O2}{-90:3cm}{Hpl2}{90:2cm}
   \elmove{b2}{0:5mm}{O3}{-10:5mm}
   \elmove{b4}{-40:5mm}{b3}{0:5mm}
   \elmove{b5}{-30:5mm}{O4}{-10:5mm}
 }
\end{rxn}
```



7 Alphabetical Command Reference

In the following section every command is explained.

7.1 anywhere

Sometimes it might be useful to be able to place a reactant or anything else off the chain.

`\anywhere{<pos>,<name>,<tikz>}{<something>}`

For this case there's `\anywhere`. It is positioned with `<pos>` similar to `\branch`.

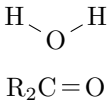
`<anchor>.<angle>` not on the chain

`on chain=going <key>` on the chain

`<key>=of <anchor>` not on the chain

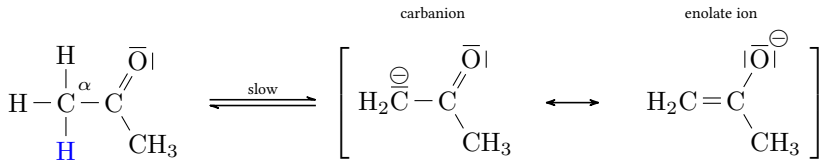
Please be aware that `<pos>` cannot be left empty.

```
\begin{rxn}
 \reactant[,carbonyl_A]{\chemfig{R_2C=O}}
 \anywhere{above=of carbonyl_A}{\chemfig{H-[:-30]O-[:30]H}};
\end{rxn}
```



You can use this command e.g. for placing labels next to reactants.

```
\begin{rxn}
 \reactant[,ketone]{\chemfig{H-\chemabove{C}{\hspace*{5mm}\scriptstyle\
 alpha}(-[2]H)(-[6,,,2]{}|{\textcolor{blue}H})-C(=[:60]\lewis{02,O})
 -[:-60]C|H_3}}
 \anywhere{below=of ketone}{$+ \ch[format=\color{blue}]{OH-}$}
 \arrow[,<=>]{\tiny slow}{}
 \mesomeric[,mesomer]{
   \reactant[,carbanion]{\chemfig{H_2|\chemabove[3pt]{\lewis{2,C}}{\fscrm
 }-C(=[:60]\lewis{02,O})-[:-60]C|H_3}}
   \marrow
   \reactant[,enolate]{\chemfig{H_2C=C(-[:60]\chemabove{\lewis{024,O}}{\
 hspace*{5mm}\fscrm})-[:-60]C|H_3}}
 }
 \anywhere{above=of enolate}{\tiny enolate ion}
 \anywhere{above=of carbanion}{\tiny carbanion}
 \anywhere{below=of mesomer}{$+ \ch[format=\color{blue}]{H2O}$}
\end{rxn}
```



You can find many further examples for the usage of `\anywhere` in `examples.tex` or `examples.pdf`, respectively.

7.2 `\arrow`

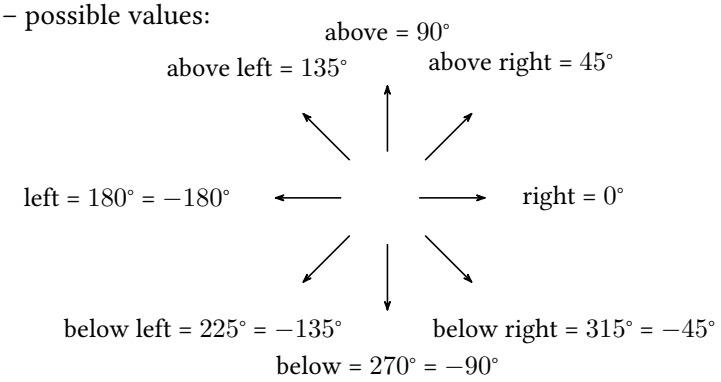
Reaction arrows are created with `\arrow`.

`\arrow[<pos>,<type>,<length factor>,<name>,both,<tikz>]{<above>}{<below>}`

7.2.1 Options

There are a number of options to customize the arrows. The options must be used in the right order, separated with commas.

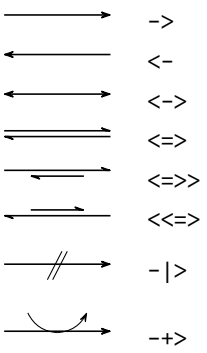
1. <pos> – possible values:



Additionally you can use every angle from the interval $[-360°; 360°]$.

You can also use an angle with respect to an object that's been named with <name>: <name>.<angle>. This means the arrow is placed in an angle of <angle> next to <name>. You can find lots of examples in `examples.tex` or`examples.pdf`, respectively.

2. <type> – possible values:



3. <length factor> – this factor multiplies with the arrow length (4 em with factor = 1.0, default)

4. <name> – can be used to give the arrow an anchor, to which another object can refer.

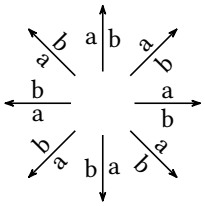
5. `both`– this gives both nodes within which the labels are written the same size.

6. <tikz> – you can modify the arrow with Ti***k***Z keys. Not all Ti***k***Z keys have an effect, though. You can't shift arrows with `shift=`<coord>, for example.

```
\begin{rxn}
\arrow[,,.25]{M}{}\arrow[,,.5]{MM}{}\arrow[,,.75]{MMM}{}\arrow{MMMM}{}\
arrow[,,1.125]{MMMMM}{}\arrow[,,1.25]{MMMMM}{}
\end{rxn}
```


Please note, that the labels <above> and <below> are rotated with the arrow. At an angle of 180° <above> actually is *below* the arrow.

```
\begin{rxn}
 \setarrowlength{2.5em}
 \dummy[a]
 \arrow{a}{b}
 \arrow[a.45]{a}{b}
 \arrow[a.90]{a}{b}
 \arrow[a.135]{a}{b}
 \arrow[a.180]{a}{b}
 \arrow[a.225]{a}{b}
 \arrow[a.270]{a}{b}
 \arrow[a.315]{a}{b}
\end{rxn}
```

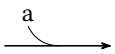


You need to be aware of one or two points, when using the arrow type `-+>`: if you leave the labels empty, the arrow is the same as `->`. The first label is the one added to the reaction, the second the one subtracted. If you use only one of the labels only the corresponding arrow part is drawn.

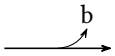
```
corresponds \verb"->":
\begin{rxn}
 \arrow[,-+>]{}{}
\end{rxn}
add:
\begin{rxn}
 \arrow[,-+>]{a}{}
\end{rxn}
subtract:
\begin{rxn}
 \arrow[,-+>]{}{b}
\end{rxn}
both add and subtract:
\begin{rxn}
 \arrow[,-+>]{a}{b}
\end{rxn}
Blanks are \emph{not} an empty label:
\begin{rxn}
 \arrow[,-+>]{ }{ }
\end{rxn}
```

corresponds ->:

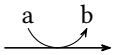
add:



subtract:



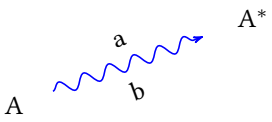
both add and subtract:



Blanks are *not* an empty label:

You can change the appearance of an arrow with Ti*k*Z keys:

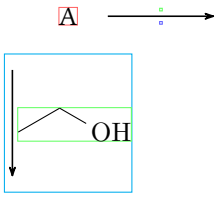
```
\begin{rxn}
 \mCsetup{arrowlabel=.7em,arrowlength=6em}
 \reactant{A}
 \arrow[20,,,,,decorate,decoration=snake,blue]{a}{b}
 \reactant[20]{A$^*$}
\end{rxn}
```



7.2.2 Alignment

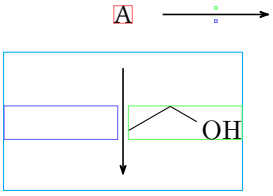
If an arrow is placed inside a branch the alignment of the branch may be determined by the size of the label nodes. If the two labels have a different size, alignment can go wrong.

```
\makevisible
\begin{rxn}
 \reactant[,a]{A}
 \arrow{}{}
 \branch[below=of a]{
   \arrow[-90]{\chemfig{-[::30]-[::-60]OH}}{}
 }
\end{rxn}
\makeinvisible
```



By using the option `both` both nodes get the same size. This can correct the alignment.

```
\makevisible
\begin{rxn}
 \reactant[,a]{A}
 \arrow{}{}
 \branch[below=of a]{
  \arrow[-90,,,,both]{\chemfig{-[::30]-[::-60]OH}}{}
 }
\end{rxn}
\makeinvisible
```



There's more to the question of alignment in section 7.3.2.

7.2.3 Appearance

You can change the general appearance of arrows with `\setarrowlength` (section 7.17) and `\setarrowline` (section 7.20).

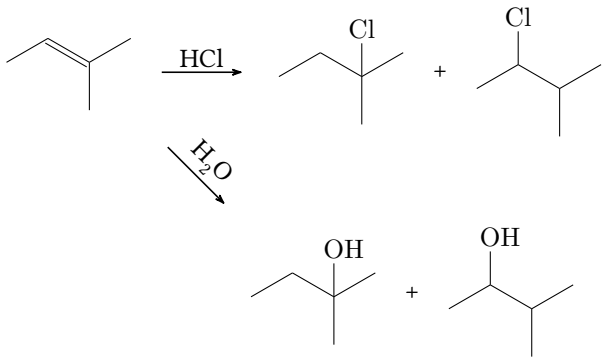
7.3 `\branch`

`\branch` is used to, well, create a branch to a reaction.

`\branch[<pos>,<anchor>,<tikz>]{<branch code>}`

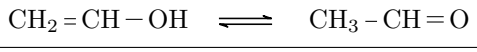
For \branch positioning an anchor is important. Let's take a look at an example:

```
1  \begin{rxn}
2   \reactant[,start]{\chemfig{-[::30]=_[::-60](-[::-60])-[::60]}}
3   \arrow[,,.75]{\ch{HCl}}{}
4   \reactant{\chemfig{-[::30]-[::-60](-[::120]Cl)(-[::-60])-[::60]}}
5   \chemand
6   \reactant{\chemfig{-[::30](-[::60]Cl)-[::-60](-[::-60])-[::60]}}
7   \branch[below right=of start]{
8     \arrow[-45,,.75]{\ch{H2O}}{}
9     \reactant[-45]{\chemfig{-[::30]-[::-60](-[::120]OH)(-[::-60])-[::60]}}
10    \chemand
11    \reactant{\chemfig{-[::30](-[::60]OH)-[::-60](-[::-60])-[::60]}}
12  }
13 \end{rxn}
```



The first reactant got the anchor `start` (line 2, also see section 7.13). \branch now refers to it in its alignment (line 7). If you don't use the alignment reference to an anchor you automatically refer to the last \reactant or \arrow. If you don't use alignment at all, then the branch is aligned to the right of the last \reactant or \arrow.

```
1  \begin{rxn}
2   \reactant{ \chemfig{CH_2=CH-OH} }
3   \arrow[,<=>,.5]{}{}
4   \branch{ \reactant{ \chemfig{CH_3-CH=O} } }
5  \end{rxn}
```



7.3.1 Positioning

There are several ways to position a branch. It can either be part of the chain or be a real branch.

`<angle>` on the chain

<key> on the chain

<anchor>.<angle> not on the chain

on chain=going <key> on the chain

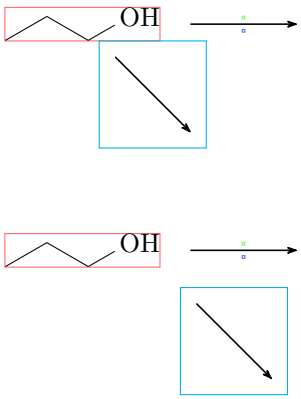
<key>=of <anchor> not on the chain

The default behaviour is equal to \branch[0]{}. In different situations different ways can be favoured. For example if you want to use \branch to shift an arrow it can still be part of the chain.

```
\begin{rxn}
 \reactant{A}
 \branch[,,yshift=1em]{\arrow{}{}}
 \reactant[,,yshift=-1em]{B}
\end{rxn}
```

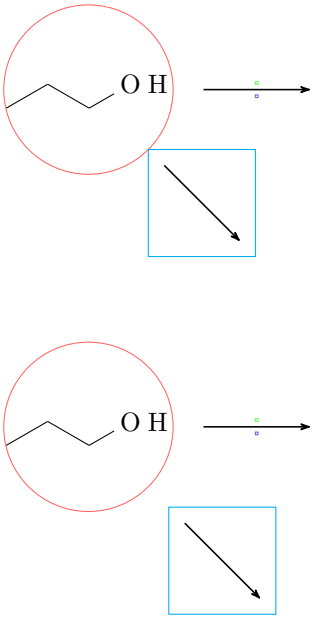
If you like to start a real branch it *does* matter which way you use.

```
\makevisible
\begin{rxn}
 \reactant[,a]{\chemfig{[:30]--[::-60]-OH}}
 \arrow{}{}
 \branch[a.-45]{\arrow[-45]{}{}}
\end{rxn}
\begin{rxn}
 \reactant[,a]{\chemfig{[:30]--[::-60]-OH}}
 \arrow{}{}
 \branch[below right=of a]{\arrow[-45]{}{}}
\end{rxn}
```



If a reactant isn't squared or circled $-45°$ does not mean the same as "below right."

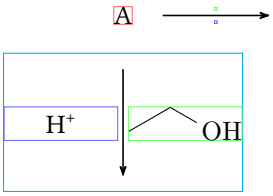
```
\makevisible
\begin{rxn}
 \reactant[,a,circle]{\chemfig{[:30]--[::-60]-OH}}
 \arrow{}{}
 \branch[a.-45]{\arrow[-45]{}{}}
\end{rxn}
\begin{rxn}
 \reactant[,a,circle]{\chemfig{[:30]--[::-60]-OH}}
 \arrow{}{}
 \branch[below right=of a]{\arrow[-45]{}{}}
\end{rxn}
```



7.3.2 Alignment problems

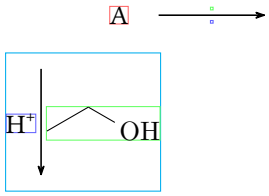
If an arrow has two arguments with different sizes and is placed inside a branch the alignment of the branch can go wrong. In this case the `\arrow` key `both` isn't a solution since the smaller argument then is not placed next to the arrow but is centered in its node.

```
\makevisible
\begin{rxn}
 \reactant[,a]{A}
 \arrow{}{}
 \branch[below=of a]{
  \arrow[below,,,,both]{\chemfig{-[::30]-[::-60]OH}}{\Hpl}
 }
\end{rxn}
```



What you have to do is shift the branch using the Ti*k*Z keys `xshift` and `yshift`.

```
\makevisible
\begin{rxn}
 \reactant[,a]{A}
 \arrow{}{}
 \branch[below=of a,,xshift=-1.35em]{
  \arrow[below]{\chemfig{-[::30]-[::-60]OH}}{\Hpl}
 }
\end{rxn}
```



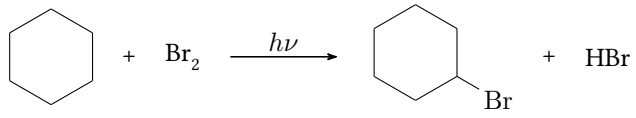
7.4 \chemand

The command

`\chemand[<alignment>,<anchor>,<tikz>]`

produces and places a + in the same way `\reactant` places arbitrary text.

```
\begin{rxn}
 \reactant{\chemfig{*6(------)}}
 \chemand
 \reactant{\ch{Br2}}
 \arrow{$h\nu$}{}
 \reactant{\chemfig{*6(--(-Br)----)}}
 \chemand
 \reactant{\ch{HBr}}
\end{rxn}
```



The optional arguments for \chemand and \reactant are the same, see section 7.13 for \reactant's options.

7.5 \dummy

\dummy creates an empty node. Up to v1.3 \arrow needed to follow after a node since \arrow internally uses \tikzchainprevious. If there was no node on the chain *before* \arrow was used an error was raised. The same thing happened with \branch. By using \dummy you could start a scheme with an arrow anyway.

```
\begin{rxn}
 \dummy\arrow{}{}
\end{rxn}
```

This is not *necessary any more.* In some (alignment) cases an empty node still might be useful so the command still exists.

7.6 \elmove

\elmove just is a wrapper for chemfig's \chemmove.

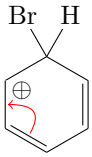
`\elmove[<tikz>]{<start>}{<start direction>}{<end>}{<end direction>}`

This is expanding to

```
\chemmove{\draw[<tikz>](<start>).. controls +(<start direction
 >) and +(<end direction>)..(<end>);}
```

using `[->,red,shorten <=3pt,shorten >=1pt]` as default for `<tikz>`. How you use \chemmove is described in the documentation of .

```
\begin{center}
 \setatomsep{1.8em}
 \chemfig{*6(=[@{e1}]-=-(-[:120]Br)(-[:60]H)-(-[:-30,.4,,,white]\oplus)-[
 @{e2}])}
 \elmove{e1}{60:4mm}{e2}{0:4mm}
\end{center}
```



You should use \elmove *only inside* of \anywhere, \reactant or \transition. Otherwise you might get alignment errors.

7.7 \makeinvisible

\makeinvisible

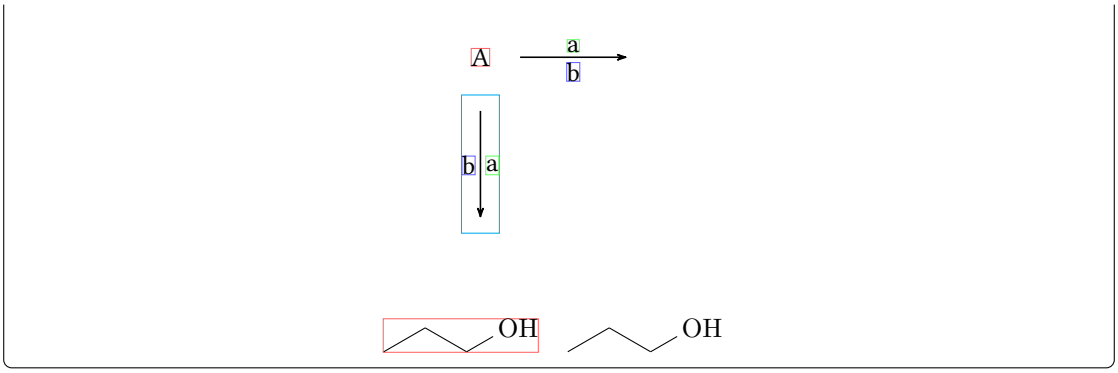
\makeinvisible restores the normal MYCHEMISTRY behaviour after \makevisible (see section 7.8) has been used. \makeinvisible only changes the looks of nodes following after it.

7.8 \makevisible

\makevisible

With \makevisible you can visualize the nodes within which reactants, arrows and branches are set. This is useful when you're aligning branches, for example. You can see an example of the usage of \makevisible in section 7.2. Every kind of nodes is emphasized with a different colour:
\reactant{}, \arrow{above}, \arrow{below} and \branch{}. Also see section 7.7.
\makevisible only changes the looks of nodes following after it ond only effects the group whithin which it is used.

```
\begin{rxn}
 \makevisible
 \reactant[,a]{A}
 \arrow{a}{b}
 \branch[below=of a]{
   \arrow[-90,,,,both]{a}{b}
 }
\end{rxn}
\begin{rxn}
 {\makevisible
 \reactant{\chemfig{[:30]--[::-60]-OH}}
 }
 \reactant{\chemfig{[:30]--[::-60]-OH}}
\end{rxn}
```



Please beware that `\makevisible` only *has an effect if the package option* `draft` *is active*!

7.9 `\marrow`

`\marrow` creates a double-headed arrow.

`\marrow[<direction>]`

It is a shortcut for `\arrow[<direction>,<->,.5]{}{}`.

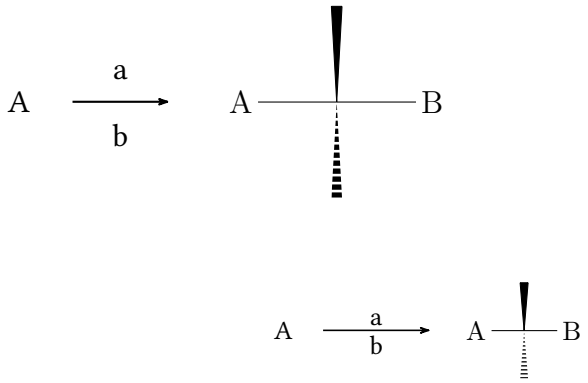
7.10 `\mCsetup`

With

`\mCsetup{<keys>}`

you can fully customize MYCHEMISTRY. For each of MYCHEMISTRY's commands going like `\set<command>` except `\setbondshape` there is a key `<command> = <value>`. Additionally there is the key `align = <value>` with which you can change the alignment behaviour of both `\begin{rxn} \end{rxn}` and `\begin{rxnscheme} \end{rxnscheme}`, and the key `reset` with whith all values are reset to default.

```
\mCsetup{
 align=left,
 arrowlabel=.7em,
 arrowlength=3em,
 arrowline=thick,
 atomsize=\large,
 bondlength=3em,
 mergelength=4em,
 rcndist=2em
 %rxnalign=right,
 %schemealign=left
}
\setbondshape{4pt}{2pt}{1pt}
\begin{rxn}
 \reactant{A}\arrow{a}{b}\reactant{\chemfig{A-(<[2])(<:[6])-B}}
\end{rxn}
\mCsetup{reset}
\begin{rxn}
 \reactant{A}\arrow{a}{b}\reactant{\chemfig{A-(<[2])(<:[6])-B}}
\end{rxn}
```



\mCsetup as well as every other setup command only effects the group whithin which it is used.

Here is a complete list of valid keys:

align

arrowlabel

arrowlength

arrowline

atomsize

bondlength

elmove

mergelength

rcndist

reset

rxnalign

schemealign

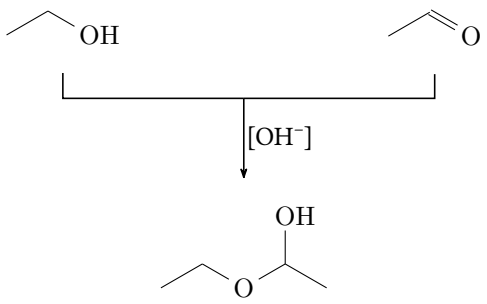
7.11 \merge

The \merge command is used to merge different reaction chains. In order to do that the reactants that are to be merged must have an anchor name (\reactant[,<anchor>]{}), similar with branches, see sections 7.13 and 7.3) and be placed *before* \merge.

\merge[<label>,<direction>,<length>]{<target>}{<start a>}{<start b>}

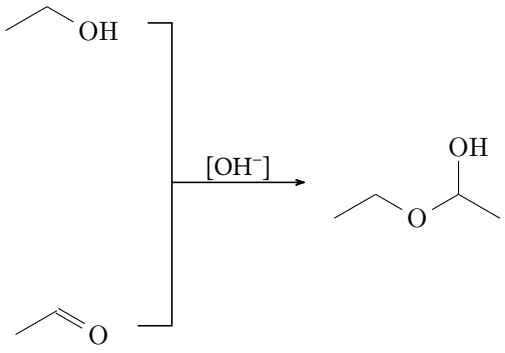
\merge has three optional and three mandatory arguments. The latter are the anchor names of the reactants that are to be merged.

```
\begin{rxn}
 \branch[,first]{
  \reactant[,start_a]{\chemfig{-[:30]-[:-30]OH}}
  \reactant[,start_b,xshift=9em]{\chemfig{-[:30]=[:-30]O}}
 }
 \branch[below=of first,target,yshift=-5em]{
  \reactant{\chemfig{-[:30]-[:-30]O-[:30](-[2]OH)-[:-30]}}
 }
 \merge[\ch{[OH^-]}]{target}{start_a}{start_b}
\end{rxn}
```

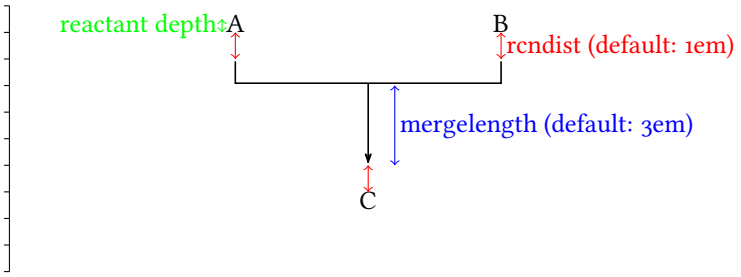


The default <direction> is below, other possible values are right, left and above. With <length> the length of the arrow from the point of merging to the tip can be changed. The default length is 3 em. The default length can be changed with \setmergelength or \mCsetup.

```
\begin{rxn}
 \reactant[,start_a]{\chemfig{-[:30]-[:-30]OH}}
 \branch[below=of start_a,start_b,yshift=-9em]{
  \reactant{\chemfig{-[:30]=[:-30]O}}
 }
 \branch[right=of start_a,target,xshift=7em,yshift=-6em]{
  \reactant{\chemfig{-[:30]-[:-30]O-[:30](-[2]OH)-[:-30]}}
 }
 \merge[\ch{[OH^-]},right,5em]{target}{start_a}{start_b}
\end{rxn}
```

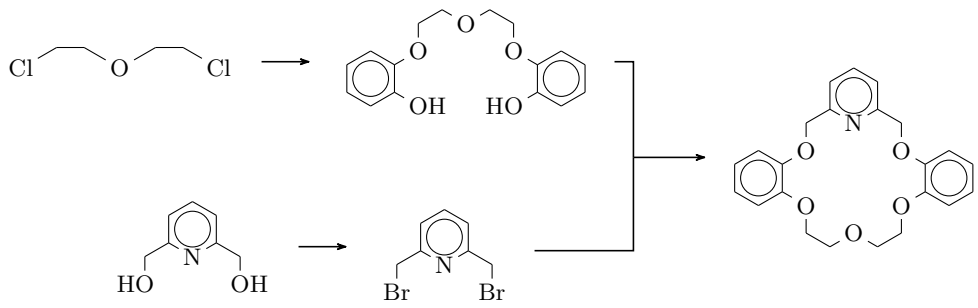


Since you have to place the reactants first it might be useful to know a little bit about involved lengths. There are three values which influence the needed distance between start points and the target reactant. There is the *depth* of the involved reactants, the distance of the "reaction nodes" *rcndist* (see section 7.24) and the length of the `\merge` arrow *mergelength* (also see section 7.23).



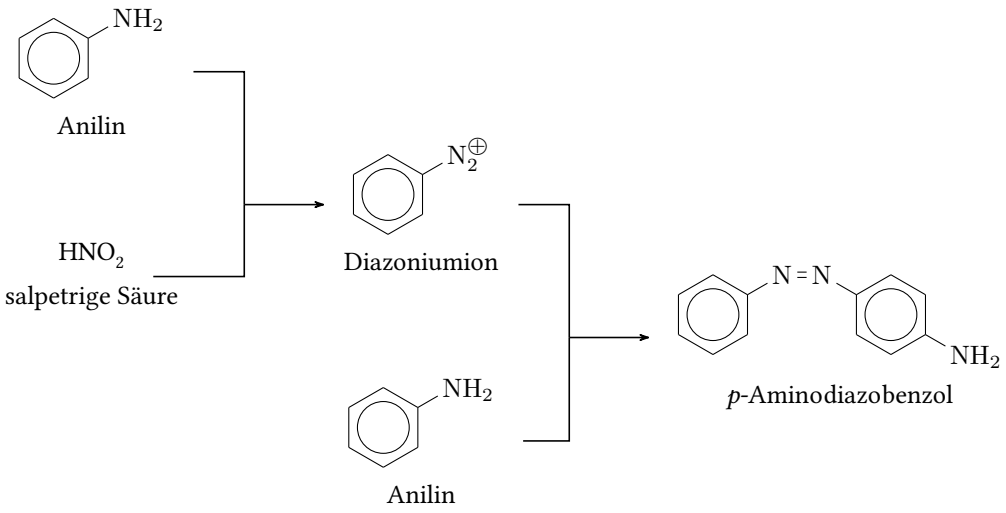
At last an example where two chains are merged.

```
\begin{rxn}
 \setatomsep{1.15em}\footnotesize
 % chain 1
  \reactant[,first]{ \chemfig{Cl-[:30,1.5]--[:-30,1.5]O
 -[:30,1.5]--[:-30,1.5]Cl}{} }
  \arrow[,,.5]{}{}
  \reactant[,start_above]{ \chemfig{O(-[:-150]**6(-----(-OH)-))
 -[:90]-[:30]-[:-30]O-[:30]-[:-30]-[:-90]O-[:-30]**6(-(-HO)-----)} }
 % chain 2
 \branch[below=of first,start_below,xshift=8em,yshift=-4em]{
  \reactant{ \chemfig{**6((--[6,,,2]HO)-N-(--[6]OH)----)} }
  \arrow[,,.5]{}{}
  \reactant{ \chemfig{**6((--[6]Br)-N-(--[6]Br)----)} }
 }
 % target
 \branch[right=of start_above,target,xshift=5em,yshift=-4em]{
  \reactant[,c]{ \chemfig{O(-[:-150]**6(-----(-O?)-))-[:90]-[:30]**6(-N
 -(--[:-90]O-[:-30]**6(-(-O-[6]-[:-150]-[:150]O-[:-150]-[:150]?)-----))
 ----)} }
 }
 % merging:
 \merge[,right]{target}{start_above}{start_below}
\end{rxn}
```



The usage of `\merge` may require some playing with branches, `xshift` and `yshift` until you get the desired result.

```
\begin{rxn}
 \setatomsep{1.5em}
 \reactant[,start_aa]{ \chemname{\chemfig{**6(---(-NH_2)---)}}{Anilin} }
 \reactant[below,start_ab,yshift=-3em]{ \chemname{\ch{HNO2}}{salpetrige S
 \"aure} }
 \branch[right=of start_aa,target_a,xshift=6em,yshift=-5em]{
   \reactant{ \chemname{\chemfig{**6(---(-N|_2\fpch)---)}}{Diazoniumion}
 }
 }% = start_ba
 \branch[below=of target_a,start_bb,yshift=-3em]{
   \reactant{ \chemname{\chemfig{**6(---(-NH_2)---)}}{Anilin} }
 }
 \branch[right=of target_a,target_b,xshift=6em,yshift=-5em]{
   \reactant{ \chemname{\chemfig{N(-[:-150]**6(------))=N-[:-30]**6(---(-
 NH_2)---)}}{\iupac{\para\-Amino\|diazo\|benzol}} }
 }
 \merge[,right]{target_a}{start_aa}{start_ab}
 \merge[,right]{target_b}{target_a}{start_bb}
\end{rxn}
```



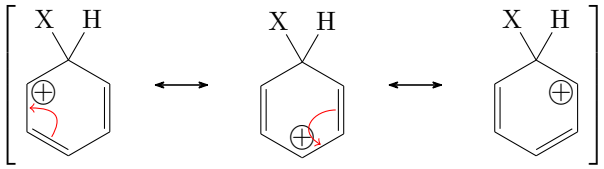
7.12 \mesomeric

The \mesomeric command works just like \branch (see section 7.3) but places the formulæ into square brackets.

\mesomeric[<pos>,<anchor>,<tikz>]{<formula>}

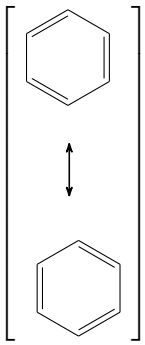
The resonance formulæ are written into <formula>. With \marrow (see section 7.9) you can create the resonance arrows. If needed you can give an anchor (<anchor>) to \mesomeric (also see section 7.3). Alignment is used the same way as with \reactant.

```
\begin{rxn}
 \mesomeric{
   \reactant{
     \chemfig{*6(=[@{e1}]-=-(-[:120]X)(-[:60]H)-(-[:-30,.4,,,white]\fplus
 )-[@{e2}])}
     \elmove{e1}{60:4mm}{e2}{0:4mm}
   }
   \marrow
   \reactant{
     \chemfig{*6(-(-[:90,.4,,,white]\fplus)-[@{e4}]=[@{e3}]-(-[:120]X)
 (-[:60]H)-=)}
     \elmove{e3}{180:4mm}{e4}{150:4mm}
   }
   \marrow
   \reactant{
     \chemfig{*6(-=-(-[:-150,.4,,,white]\fplus)-(-[:120]X)(-[:60]H)-=)}
   }
  }
\end{rxn}
```



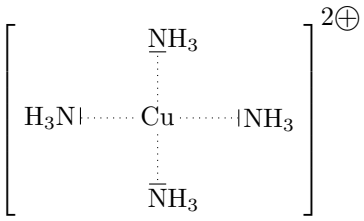
Or vertical, too:

```
\begin{rxn}
 \mesomeric{
   \reactant{ \chemfig{*6(=-=-=-)} }
   \marrow[below]
   \reactant[below]{ \chemfig{*6(-=-=-=)} }
 }
\end{rxn}
```



Or maybe a coordination complex?

```
\begin{rxn}
 \setatomsep{3em}
 \mesomeric[,a]{
   \reactant{ \chemfig{H_3\lewis{0,N}-[,1.35,,,dotted]{Cu}(-[2,,,,dotted
 ]\lewis{6,N}H_3)(-[6,,,,dotted]\lewis{2,N}H_3)-[,1.2,,,dotted]\lewis{4,N
 }H_3} }
 }
 \node[above right=of a,yshift=-1em] {$2\fplus$};
\end{rxn}
```



7.13 \reactant

\reactant is something like the basic command.

\reactant[<pos>,<name>,<tikz>]{<formula>}

Formulæ (<formula>) are written inside this command and can be named (<name>) for other objects to refer to. <pos> can either be one of these key words

1. right,
2. above right,
3. above,
4. above left,
5. left,
6. below left,
7. below,
8. below right

or be an angle from the intervall $[-360°; 360°]$. You also can position it relativ to another object via <anchor>.<angle>. The default equals \reactant[0]{}.

```
below:
\begin{rxn}
  \reactant{\ch{Br2}}
  \reactant[-90]{\ch{Cl2}}
\end{rxn}

more than one reactant:
\begin{rxn}
  \reactant{\ch{Br2}}
  \reactant[-90]{\ch{I2}}
  \reactant{\ch{Cl2}}
\end{rxn}

going down:
\begin{rxn}
 \reactant{\ch{Br-Br}}
 \arrow[-90,,.5]{$h\nu$}{}
 \reactant[-90]{\ch{2 "\Lewis{0.,Br}"}}
\end{rxn}
```

below:

$$\mathrm{Br_2}$$

$$\mathrm{Cl_2}$$

more than one reactant:

$$\mathrm{Br_2}$$

$$\mathrm{I_2} \quad \mathrm{Cl_2}$$

going down:

$$\mathrm{Br{-}Br} \downarrow h\nu \; 2\,\mathrm{Br^{\bullet}}$$

You can find many more examples in the file `examples.tex` or `examples.pdf`, respectively.

7.14 rxn (environment)

`\begin{rxn} \end{rxn}` is a non-floating not numbered environment for reaction schemes. Per default all schemes are centered.

`\begin{rxn}[<alignment>,<scalefactor>] ... \end{rxn}`

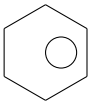
7.14.1 Options

\begin{rxn} \end{rxn} has two options which are to be used in the following order, separated by a comma:

1. <alignment>: alignment behaviour of the \begin{rxn} \end{rxn} environment; default is center.
2. <scalefactor>: factor by which the \begin{rxn} \end{rxn} environment is scaled; default: 1.0. Same behaviour as with \begin{rxnscheme} , \end{rxnscheme} see section 7.15.

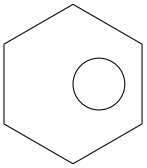
If you use the <scalefactor> option you might see strange effects on chemfig formulæ.

```
\begin{rxn}[,.5]
 \reactant{\chemfig{**6(------)}}
\end{rxn}
```



Scaling does in general not affect the size of chemfig formulæ but does scale the aromaticity ring of benzene and similar molecules. This is not MYCHEMISTRY's fault:

```
\chemfig[scale=.5]{**6(------)}
```

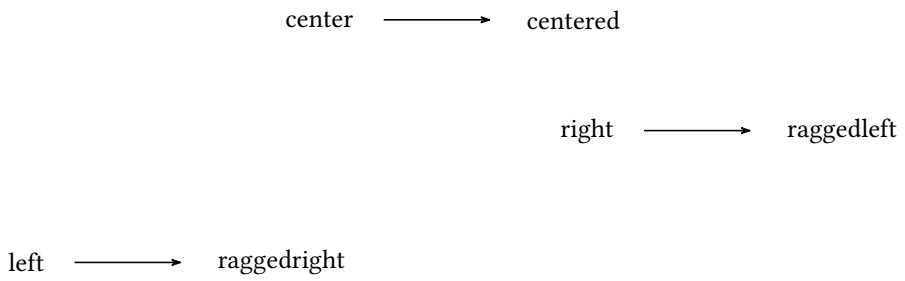


This issue can be solved either by using the *first* optional argument of \chemfig to undo the scaling or by using the *second* optional argument to scale the rest of the molecule.

```
\begin{rxn}[,.5]
 \reactant{\chemfig[scale=2]{**6(------)}}
 \reactant{\chemfig[][scale=.5]{**6(------)}}
\end{rxn}
\chemfig[scale=.5][scale=.5]{**6(------)}
```

Alignment examples:

```
\begin{rxn}[center]
 \reactant{center}\arrow{}{}\reactant{centered}
\end{rxn}
\begin{rxn}[right]
 \reactant{right}\arrow{}{}\reactant{raggedleft}
\end{rxn}
\begin{rxn}[left]
 \reactant{left}\arrow{}{}\reactant{raggedright}
\end{rxn}
```



7.15 rxnscheme (environment)

`\begin{rxnscheme} \end{rxnscheme}` is a floating environment for reaction schemes.

`\begin{rxnscheme}[<label>,<placement>,<alignment>,<scalefactor>,<name>]{<caption>}`
... `\end{rxnscheme}`

7.15.1 Options

`\begin{rxnscheme} \end{rxnscheme}` has five options which are to be used in the following order, separated by commas:

1. `<label>`: Like every other floating environment `\begin{rxnscheme} \end{rxnscheme}` can be given a label. In order to do that you need to use the option `<label>`. For example if you use

   ```
   \begin{rxnscheme}[rs:schema]{<caption>}
    ...
   \end{rxnscheme}
   ```

   you can refer to it by using `\ref{rs:schema}` as usual.

2. `<placement>`: With this option you can change the placement of the float, e.g. with `h`. The default value is `htp`.

3. <alignment>: This option changes the alignment of the scheme. You can choose between `left`, `center` and `right`.

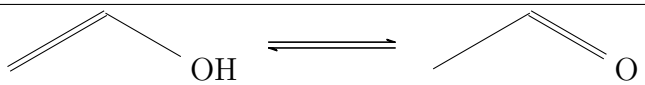
4. <scalefactor>: `\begin{rxnscheme} \end{rxnscheme}` has another key with which the scheme can be scaled. Please keep in mind that it doesn't affect the font size and the size of chemfig's formulæ. You can have strange effects on chemfig's formulæ if you use this key, though. See section 7.14.1 for more information.

5. <name> This option changes the name of the actual scheme from "Reaction scheme" into <name>.

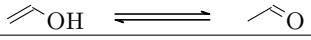
```
\begin{rxnscheme}[,<placement>]{<caption>}
 ...
\end{rxnscheme}
```

```
\begin{rxnscheme}[,,,2]{Big scheme}
 \large\setatomsep{3.5em}
 \reactant{ \chemfig{=[::30]-[::-60]OH} }
 \arrow[,<=>]{}{}
 \reactant{ \chemfig{-[::30]=[::-60]O} }
\end{rxnscheme}
\begin{rxnscheme}[,,,.5]{Small scheme}
 \setatomsep{1.15em}\footnotesize
 \reactant{ \chemfig{=[::30]-[::-60]OH} }
 \arrow[,<=>]{}{}
 \reactant{ \chemfig{-[::30]=[::-60]O} }
\end{rxnscheme}
\begin{rxnscheme}{center}
 \reactant{center}\arrow{}{}\reactant{centered}
\end{rxnscheme}
\begin{rxnscheme}[,,right]{right}
 \reactant{right}\arrow{}{}\reactant{raggedleft}
\end{rxnscheme}
\begin{rxnscheme}[,,left]{left}
 \reactant{left}\arrow{}{}\reactant{raggedright}
\end{rxnscheme}
```

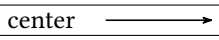
Reaction scheme 2 Big scheme



Reaction scheme 3 Small scheme



Reaction scheme 4 center



7.15.2 Customizing rxnscheme

Style If you don't like the style of `\begin{rxnscheme} \end{rxnscheme}` you can change it by using

```
\floatstyle{<new style>}
```

```
\rxnfloat
```

There are different possible styles, provided by the float package:

`plain` without any special formatting, the caption is below the object.

`plaintop` like `plain` but the caption is placed above the object.

`boxed` the object is boxed, the caption placed below.

`ruled` the caption is placed above the object framed by two rules, one above and one below, another rule frames the object below; default for `\begin{rxnscheme} \end{rxnscheme}`.

```
\begin{rxnscheme}{ruled}
 \reactant{default style}
\end{rxnscheme}
\floatstyle{boxed}
\restylefloat{rxnfloat}
\begin{rxnscheme}{boxed}
 \reactant{framed object}
\end{rxnscheme}
\floatstyle{plain}
\restylefloat{rxnfloat}
\begin{rxnscheme}{plain}
 \reactant{without any special formatting}
\end{rxnscheme}
```

Placement Usually floating environments have an optional argument for their placement. The default placement is `htp`. If you want to change it into `H` or something, you can use

```
\floatplacement{rxnfloat}{<placement>}
```

It's easier, though, loading MYCHEMISTRY with the `placement` option:

Reaction scheme 5 right

right ⟶ raggedleft

Reaction scheme 6 left

left ⟶ raggedright

```
\usepackage[placement=<placement>]{mychemistry}
```

This will change the default placement behaviour from `htp` to `<placement>`. You can also change the placement behaviour of just one `\begin{rxnscheme} \end{rxnscheme}` by using the placement option:

```
\begin{rxnscheme}[,<placement>]{<caption>}
 ...
\end{rxnscheme}
```

Name If you want to change the name of reaction scheme you can do that with

`\setschemename{<new name>}`

The default name is "Reaction scheme" but can change depending on the language you've loaded with babel.

Counter The counter can be changed just as usual. For example by using

```
\makeatletter
\@addtoreset{rxnfloat}{section}
\makeatletter
\renewcommand*\therxnscheme{\arabic{section}.\arabic{rxnscheme
 }}
```

the counter is reset with every new section an looks like `section.rxnscheme`.

List of Schemes By writing

`\listof{rxnfloat}{<title>}`

you can create a list of all schemes created with `\begin{rxnscheme} \end{rxnscheme}`.

Reaction scheme 7 ruled

default style

framed object

Reaction scheme 8: boxed

```
\listof{rxnfloat}{Reaction schemes}
```

Reaction schemes

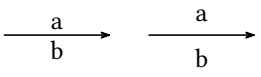
7.16 \setarrowlabel

The distance of arrow labels to the arrow has a default value of 0.2 em. You can change this with

`\setarrowlabel{<distance>}`

into `<distance>`.

```
\begin{rxn}
 \arrow{a}{b}
 \setarrowlabel{.5em}
 \arrow{a}{b}
\end{rxn}
```



7.17 \setarrowlength

The default length of an reaction arrow is 4 em. You can change this value with

`\setarrowlength{<length>}`

into `<length>`. Mind the fact that you have to use a length unit. If you leave the argument empty the length is reset to default.

without any special formatting

Reaction scheme 9: plain

7.18 \setatomsize

With

```
\setatomsize{<font size>}
```

you can change the font size of the atom groups. Default value is \small. If you leave the argument empty the size is reset to default.

7.19 \setbondlength

With

```
\setbondlength{<length>}
```

you can change \setatomsep{<length>}= for all chemfig formulæ *inside* of the MYCHEMISTRY environments. The default value is 1.8 em. If you leave the argument empty the length is reset to default.

7.20 \setarrowline

With the command

```
\setarrowline{<value>}
```

the thickness of the arrows can be customized. Possible values are

——— ultra thin
——— very thin
——— thin
——— semithick (default)
——— thick
——— very thick
——— ultra thick

This command also applies to \merge.

7.21 \setbondshape

With

```
\setbondshape{<base length>}{<dash thickness>}{<dash spacing>}
```

you can change \setcrambond{<base length>}{<dash thickness>}{<dash spacing>} for all chemfig formulæ *inside* of the MYCHEMISTRY environments. Default values are (in this order) 3 pt, 0.5 pt and 1 pt. If you leave an argument empty the value is reset to default.

7.22 \setelmove

The command

\setelmove{<tikz>}

sets the default style that is used for the lines drawn by \elmove. An empty argument resets to ->,red,shorten <=3pt,shorten >=1pt.

7.23 \setmergelength

With

\setmergelength{<length>}

you can change the length of the \merge arrow. More precisely you can change the length of the arrow from the point of line crossing to the arrow tip (see section 7.11). If you leave the argument empty the value is reset to default (3 em).

7.24 \setrcndist

The nodes within which the reactants an arrows are set have a certain distance between them. The default distance is 1 em. If you want to change that you can use

\setrcndist{<length>}

If you leave the argument empty the distance is reset to 1 em.

```
\setrcndist{2em}
\begin{rxn}
 \reactant{A}\arrow{}{}
\end{rxn}
\setrcndist{}
\begin{rxn}
 \reactant{A}\arrow{}{}
\end{rxn}
```

A ⟶

A ⟶

7.25 \setrxnalign/\setschemealign

With the commands

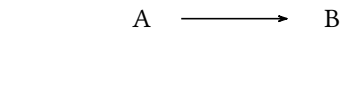
\setrxnalign{<alignment>}

\setschemealign{<alignment>}

The default alignment behaviour of \begin{rxn} \end{rxn} and \begin{rxnscheme} \end{rxnscheme} (see sections 7.14.1 and 7.15.1) can be set. You can choose between left, center and right.

If you leave the argument empty MYCHEMISTRY's default behaviour (center) is restored.

```
\setrxnalign{right}
\begin{rxn}
 \reactant{A}\arrow{}{}\reactant{B}
\end{rxn}
\setrxnalign{}
\begin{rxn}
 \reactant{A}\arrow{}{}\reactant{B}
\end{rxn}
```



7.26 \setschemename

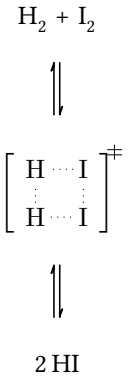
See page 44.

7.27 \transition

\transition works exactly like \reactant (see section 7.13).

\transition[<pos>,<anchor>,<tikz>]{<formula>}

```
\begin{rxn}
 \reactant{ \ch{H2 + I2} }
 \arrow[below,<=>,.5]{}{}
 \transition[below]{ \chemfig[dotted][]{H?-I-[2]I-[4]H?} }
 \arrow[below,<=>,.5]{}{}
 \reactant[below]{ \ch{2 HI} }
\end{rxn}
```



Index