

myChemistry

v1.4

April 15th 2011

Clemens Niederberger

Creating reaction schemes with L^AT_EX and ChemFig

```
\begin{rxn}[,.7]
\setatomsep{1.5em}\footnotesize
\reactant[,a]{
\chemfig{C(-[4]*6(==--))(-[2]*6(==--))(-[6,,,2]HO)-C(-[2]CH_3)
(-[6]OH)-CH_3}
}
\branch[above right=of a]{
\arrow[above right,,.7]{}{}
\reactant[above right]{
\chemfig{C(-[4]*6(==--))(-[2]*6(==--))(-[6,,,2]H_2O@{e2}
\chembelow{O}{\oplus})-C(-[2]CH_3)-[6]OH)-CH_3}\elmove{e1}
{10:4mm}{-2}{-10:4mm}
}
\arrow{$-\text{H}_2\text{O}$}{}
\reactant[below right]{
\chemfig{\chembelow{C}{\oplus}(-[4]*6(==--))(-[2]*6(==--))-C
(-[2]CH_3)-[6]OH)-CH_3}
}
}
\branch[below right=of a]{
\arrow[below right,-|>,.7]{}{}
\reactant[below right]{
\chemfig{C(-[4]*6(==--))(-[2]*6(==--))(-[6,,,2]HO)-C(-[2]CH_3)
(-[6]OH)-CH_3}
\chembelow{O}{\oplus}H_2)-CH_3}\elmove{e3}
{170:4mm}{e4}{-170:4mm}
}
\arrow{$-\text{H}_2\text{O}$}{}
\reactant{
\chemfig{C(-[4]*6(==--))(-[2]*6(==--))(-[6,,,2]HO)-
\chembelow{C}{\oplus}(-[2]CH_3)-CH_3}
}
}
\end{rxn}
```

The diagram illustrates the dehydration of 1-phenyl-1-phenylpropan-1-ol to form a carbocation intermediate. The reactant is shown as a central carbon atom bonded to two phenyl rings, a methyl group, and a hydroxyl group. The reaction is shown in two ways: one with a curved arrow indicating the loss of water from the hydroxyl group, and another with a straight arrow indicating the loss of water from the hydroxyl group. The product is a carbocation intermediate, specifically a 1-phenyl-1-phenylpropan-1-yl cation, where the central carbon atom is bonded to two phenyl rings, a methyl group, and a positive charge. The reaction is labeled with $-\text{H}_2\text{O}$ above the arrow.

Contents

SECTION 1	About	3
1.1	Changes	3
	<i>Version 1.2, 3 • Version 1.3, 4 • Version 1.4, 4</i>	
1.2	Licence	5
1.3	Requirements	5
1.4	The Idea	5
SECTION 2	Usage	6
2.1	Background	6
2.2	Basic Principle	6
2.3	How does it work?	7
	<i>Basic Commands, 7 • Branches, 8 • Numbered Schemes, 11</i>	
2.4	Predefined Values	11
2.5	Package options	13
SECTION 3	Advanced Usage, Usage with TikZ	14
3.1	The Alignment Question	14
3.2	Using TikZ to Achieve Other Results	18
SECTION 4	Alphabetical Command Reference	18
4.1	anywhere	19
4.2	arrow	20
	<i>Options, 20 • Alignment, 22 • Appearance, 22</i>	
4.3	branch	22
	<i>Alignment problems, 24</i>	
4.4	chemand	25
4.5	dummy	25
4.6	elmove	26
4.7	makeinvisible	26
4.8	makevisible	26
4.9	marrow	27
4.10	mCsetup	27
4.11	merge	28
4.12	mesomeric	32
4.13	reactant	34
4.14	rxn (environment)	35
	<i>Options, 35</i>	
4.15	rxnscheme (environment)	37

<i>Options, 37 • Customizing rnscheme, 39</i>	
4.16	setarrowlength 41
4.17	setatomsizes 42
4.18	setbondlength 42
4.19	setarrowline 42
4.20	setbondshape 42
4.21	setmergelength 43
4.22	setrendist 43
4.23	setrxnalign/setschemealign 43
4.24	setschemename 44
4.25	transition 44
<hr/>	
SECTION 5	Examples 45
5.1	Addition Reaction 45
5.2	Mesomerism 47
5.3	Hydratisation 57
5.4	Esterification 59
5.5	Extensive Synthesis 60
<hr/>	
SECTION 6	Epilogue 65
<hr/>	
SECTION 7	Thanks 66
<hr/>	
INDEX	67

1 About

1.1 Changes

The most recent changes are marked with **new**.

1.1.1 Version 1.2

Since v1.2 there are some new features. The main part plays behind the scenes: the wrong arrow- and branch-alignments have been corrected. **Due to these changes my-Chemistry now needs Version 2.10 of TikZ (or pgf, actually, see section 1.3).**

There are some new "frontend" features as well. For one thing there are some new package options (see section 2.5). Additionally the arrows have got two new keys (see section 4.2).

The environments have gained a few features, with which they can be customized, too (see [section 4.14.1](#), [section 4.15.1](#) and [section 4.22](#)).

Last but not least since v1.2 **myChemistry** underlies LPPL version 1.3 or later.

1.1.2 Version 1.3

The commands `\branch`, `\mesomeric`, `\reactant` and `\transition` now cannot only accept alignment as optional argument but **TikZ** keys as well. Also the second mandatory Argument `<anchor>` now is an optional one. So the first argument still is alignment, the second is the anchor name and in the third you can use arbitrary **TikZ** keys.

```
1 \command[<alignment>,<anchor>,<tikz>]{}
```

Version 1.2 explicitly needed a given alignment, if one wanted to use **TikZ** keys, even with default alignment. Since v1.4 this isn't required any more.

```
1 % up to version 1.2:
2 \reactant{\ce{Br2}}{\arrow{$h\nu$}}{\reactant[right,
   draw,inner sep=5pt]{\ce{2 \lewis{0.,Br}}}}
3 % since version 1.3:
4 \reactant{\ce{Br2}}{\arrow{$h\nu$}}{\reactant[,draw,
   inner sep=5pt]{\ce{2 \lewis{0.,Br}}}}
```

The commands to customize **myChemistry** have been renamed and the command `\mCsetup` has been added. Now you can customize **myChemistry** using only one command. See [section 4.16](#), [section 4.17](#), [section 4.18](#), [section 4.20](#) and [section 4.10](#).

There is the new command `\chemand`, which produces a +, see [section 4.4](#).

If you're using **ChemFig** v0.4 or v0.4a, **myChemistry** inputs the file `bondwidth.tex` which provides the command `\setbondwidth{<width>}` with which you can change the line thickness of bonds.

And you might find this one nice: every **myChemistry** command in a listing in this documentation is a hyperlink referring to the corresponding entry in the command reference.

1.1.3 Version 1.4

NEW The command `\merge` has been rewritten. Now the arrow can be labeled.

The main purpose of the command `\dummy` now is obsolete. The command still exists, though.

Both environments `rxn` and `rxnscheme` have changed regarding the usage of their options, see [section 4.14.1](#) and [section 4.15.1](#).

The keys of the `\arrow` command now are options, so the syntax of the command now is consistent with the syntax of the other commands. Then there is a new arrow type, see [section 4.2](#). Also new is the appearance of the arrows and the possibility of customizing the line thickness of the arrows, see [section 4.19](#).

There is another new command `\anywhere` ([section 4.1](#)), with which text or formulæ can be placed off the chain.

1.2 Licence

myChemistry v1.4 underlies the The L^AT_EX project public license (<http://www.latex-project.org/lppl.txt>).

1.3 Requirements

In order to function properly myChemistry needs some packages to be available.

ChemFig without it why would you use myChemistry?

ifthen for internal queries;

calc for internal calculations;

xkeyval package options and command keys are created with this package;

float the **rxnscHEME** environment is defined with this package;

pgf/TikZ pgf actually isn't just one package but a whole bundle. They are the basis layer for TikZ. myChemistry needs at least the version from 09/08/2010¹. More precisely: the command **\pgfpositionnodelater** must exist. Even more precisely: only the **\arrow** key **both** (see section 4.2) needs **\pgfpositionnodelater**. If you don't use this option, myChemistry should work nicely with pgf v2.00. Older versions have not been tested.

1.4 The Idea

Since **ChemFig** was published August 2010, there is a flexible solution for creating organic structures. With **ChemFig** and 'mhchem' one is able to create nearly all structural and molecular formulars that a chemist needs. There is one thing, though, in which 'ochem' still beats **ChemFig**: creating reaction mechanisms. This is where myChemistry comes in.

myChemistry loads the packages

- **ChemFig**²,
- 'mhchem'³ in version 3, when ist exists,
- 'chemexec'⁴, when it exists, and
- 'chemcompounds'⁵, when ist exists.

¹<http://sourceforge.net/projects/pgf/files/>

²by Christian Tellechea, <http://www.ctan.org/tex-archive/macros/generic/chemfig/>

³by Martin Hensel, <http://www.ctan.org/tex-archive/macros/latex/contrib/mhchem/>

⁴by me, <http://www.ctan.org/tex-archive/macros/latex/contrib/chemexec/>

⁵by Stephan Schenk, <http://www.ctan.org/tex-archive/macros/latex/contrib/chemcompounds/>

How these packages work, you can read in their documentations. Commands provided by these packages are for example

- `\ce{}` (mhchem)
- `\ox{}`, `\om{}`, `\op{}`, `\Hyd`, `\Hpl` (chemexec)
- `\chemfig{}`, `\chemrel{}`, `\chemsign{}`, `\lewis{}` (ChemFig)
- `\declarecompound{}`, `\compound{}` (chemcompounds).

Some of these commands are used in the examples of this documentation *without being mentioned explicitly*.

Above all, **myChemistry** provides commands for creating reaction mechanisms.

2 Usage

2.1 Background

myChemistry provides two environments within which the mechanisms are created. Both environments basically are `tikzpicture` environments. One might ask oneself: why? You can do loads of stuff with **ChemFig** already. And **TikZ** provides all the possibilities one could ask for. But since I'm a lazy guy I wrote several macros providing **TikZ** commands, I often used. They got so many and got more and more possibilities to adjust, that I bundled them into this package. Of course you can still use **TikZ** and stay more flexible with it, if you like.

2.2 Basic Principle

Within the `tikzpicture` reactants and arrows are placed as nodes on a `chain`¹.

Example 1

```
1 \begin{tikzpicture}[start chain]
2 \node [on chain] {A};
3 \node [on chain] {B};
4 \node [on chain] {C};
5 \end{tikzpicture}
```

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

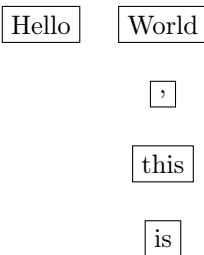
¹Provided by the tikzlibrary 'chains'

Example 2

```

1 \begin{tikzpicture}[start chain=
   going right,node distance=5mm]
2 \node [draw,on chain] {Hello};
3 \node [draw,on chain] {World};
4 \node [draw,continue chain=going
   below,on chain] {,};
5 \node [draw,on chain] {this};
6 \node [draw,on chain] {is};
7 \end{tikzpicture}

```



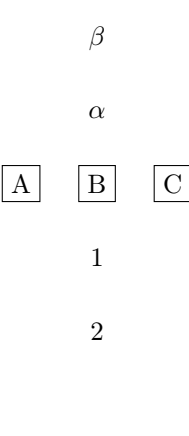
Above all **myChemistry** uses the possibility of creating branches to the chain.

Example 3

```

1 \begin{tikzpicture}[start chain=
   going right,node distance=5mm]
2 \node [draw,on chain] {A};
3 \node [draw,on chain] {B};
4 { [start branch
5 \node [on chain=going below]
   {1};
6 \node [on chain=going below]
   {2};
7 }
8 { [start branch
9 \node [on chain=going above]
   {1};
10 \node [on chain=going above]
   {2};
11 }
12 \node [draw,on chain] {C};
13 \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 4.8](#)). This is done so one can see, which size they have and which impact changes of the alignment have on them.

2.3 How does it work?**2.3.1 Basic Commands**

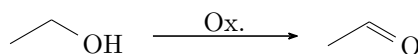
Let's take a look at an example first:

Example 4

```

1 \begin{rxn}
2 \reactant{ \chemfig
   {-[:30]-[:-60]OH} }
3 \arrow{Ox.}{}
4 \reactant{ \chemfig
   {-[:30]=_[:-60]O} }
5 \end{rxn}

```



In this example you see the most important commands of **myChemistry**:

```
1 \begin{rxn}[<align>,<scalefactor>]
```

The first of two environments. It creates a centered reaction scheme between two paragraphs (see [section 4.14](#)).

```
1 \reactant[<alignment>,<anchor>,<tikz>]{<formula>}
```

places a node on the chain, in which the chemical formulæ are written. The default alignment is to the right (see [section 4.13](#)).

```
1 \arrow[<direction>,<type>,<length factor>,<anchor>,<both>
   ]{<above>}{<below>}
```

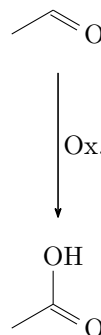
creates an arrow, which by default has a length of 5em pointing to the right (see [section 4.2](#)).

Example 5

```

1 \begin{rxn}
2 \reactant{ \chemfig
   {-[:30]=_[:-60]O} }
3 \arrow[below]{}{Ox.}
4 \reactant[below]{ \chemfig
   {-[:30](-[:-60]OH)=_[:-60]O} }
5 \end{rxn}

```



As you can see one can align the reaction scheme vertical as well. By giving **\reactant** the option **below**, the carboxylic acid is placed below the arrow. With **below** the arrow points down instead to the right.

2.3.2 Branches

With what we saw until now it is not yet clear, why one would use **myChemistry**. the horizontal reaction schemes can be realized with **ChemFig** and 'mhchem' just as well.

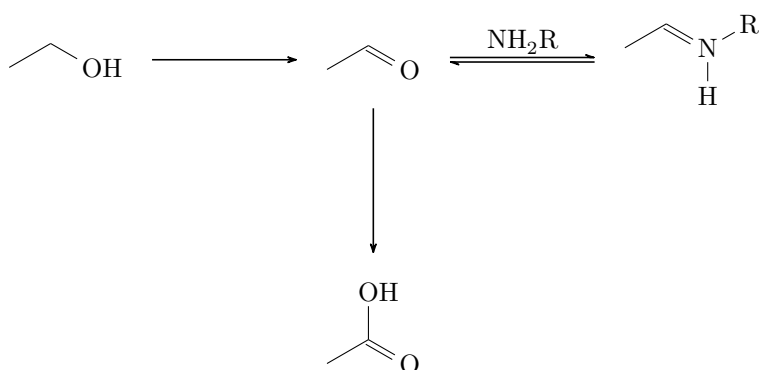
And why would anyone need a vertical reaction scheme? But what could make **myChemistry** interesting is the possibility of branched reaction schemes.

Example 6

```

1 \begin{rxn}
2 \reactant{ \chemfig{-[::30]-[::-60]OH} }
3 \arrow{}{}
4 \reactant[,carbonyl]{ \chemfig{-[::30]=_[::-60]O} }
5 \arrow[below]{}{}
6 \reactant[below]{ \chemfig{-[::30](-[::60]OH)=_[::-60]O} }
7 \branch[right=of carbonyl]{
8 \arrow[,<=>]{\ce{NH2R}}{}
9 \reactant{ \chemfig{-[::30]=_[::-60]N(-[6]H)-[::60]R} }
10 }
11 \end{rxn}

```



In the last example you've got to know another important command (see [section 4.3](#)):

```

1 \branch[<alignment>,<anchor>,<tikz>]{<branch commands>}

```

The branch was placed right of the first reactant with the anchor `carbonyl` by using the option `right=of carbonyl`. Inside the branch we used an arrow with the key `type=<=>` so we would get an equilibrium arrow. Other types would for example be `->` (default), `<-` or `<->`.

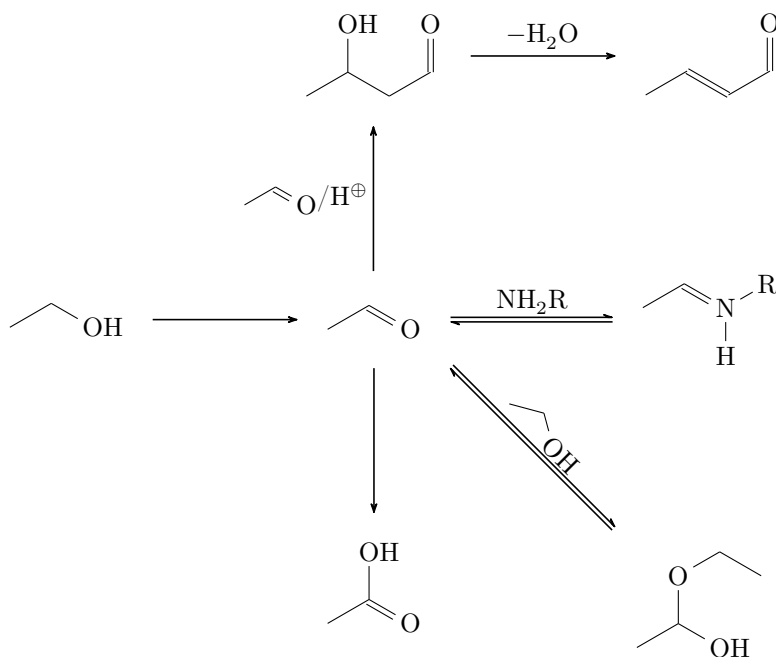
Extensive reaction schemes can be realized through multiple usage of `\branch`:

Example 7

```

1 \begin{rxn}
2 \reactant{ \chemfig{-[:30]-[:60]OH} }
3 \arrow{}{}
4 \reactant[,carbonyl]{ \chemfig{-[:30]=_[:60]O} }
5 \arrow[below]{}{}
6 \reactant[below]{ \chemfig{-[:30](-[:60]OH)=_[:60]O} }
7 \branch[right=of carbonyl,imin]{
8 \arrow[,<=>,1.12]{\ce{NH2R}}{}
9 \reactant{ \chemfig{-[:30]=_[:60]N(-[6]H)-[:60]R} }
10 }
11 \branch[below right=of carbonyl,halbacetal,yshift=-2pt,xshift=3pt]{
12 \arrow[below right,<=>,1.12]{ \chemfig{[,.75]-[:30]-[:60]OH} }{}
13 \reactant[below right]{ \chemfig{-[:30](-[:60]O-[:60]-[:60])
-[:60]OH} }
14 }
15 \branch[above=of carbonyl,aldol,xshift=5.2em]{
16 \arrow[above]{ \chemfig{[,.75]-[:30]=_[:60]O}/\Hp1 }{}
17 \reactant[above]{ \chemfig{-[:30](-[:60]OH)-[:60]-[:60]=[:60]O
} }
18 \arrow{$-\ce{H2O}$}{}
19 \reactant{ \chemfig{-[:30]=[:60]-[:60]=[:60]O} }
20 }
21 \end{rxn}

```



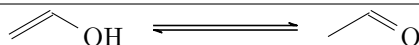
2.3.3 Numbered Schemes

The second environment provided by **myChemistry** works just as the first one. This time the scheme is placed within a numbered floating environment and is given a caption.

Example 8

```
1 \begin{rxnscheme}{Keto-enol tautomerization}
2 \reactant{\chemfig{=[:30]-[: -60]OH} }
3 \arrow[,<=>]{}{}
4 \reactant{\chemfig{-[:30]=[: -60]O} }
5 \end{rxnscheme}
```

Reaction scheme 1 Keto-enol tautomerization



Here we use the environment

```
1 \begin{rxnscheme}[<label>,<placement>,<align>,<
   scalefactor>,<name>]{<caption>}
2 ...
3 \end{rxnscheme}
```

In the command reference ([section 4.15](#)) you can read how to customize the scheme.

2.4 Predefined Values

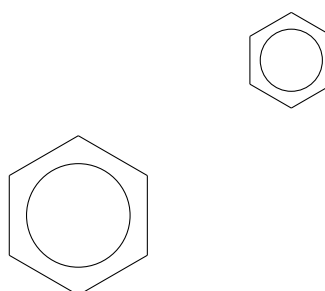
There are some predefined values, that are basically due to my personal taste. But of course you can change them according to your requirements. For **ChemFig**-formulae inside of **myChemistry** environments some values are predefined as follows:

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

Outside the **myChemistry** environments the defaults of **ChemFig** still are set.

Example 9

```
1 \begin{rxn}
2 \reactant{\chemfig{**6(-----)}}
3 \end{rxn}
4 \chemfig{**6(-----)}
```



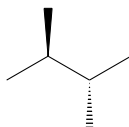
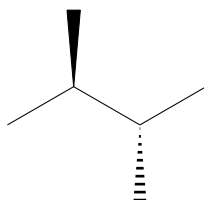
myChemistry's defaults can be changed with these commands:

```
1 \setbondlength{<length>}
2 \setbondshape{<base length>}{<dash thickness>}{<dash
   spacing>}
3 \setatomsiz{<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 `\setatomsiz` is `\small`.

Example 10

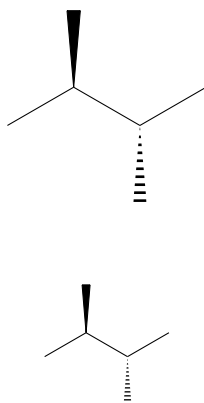
```
1 \setbondlength{2.1em}\setbondshape{5pt}{1pt}{2pt}\setatomsiz{\Large}
2 \begin{rxn}
3 \reactant{\chemfig{-[:30](<[:60])-[::-60](<[::-60])-[:60]}}
4 \end{rxn}
5 \setbondlength{}\setbondshape{}{}{}\setatomsiz{}
6 \begin{rxn}
7 \reactant{\chemfig{-[:30](<[:60])-[::-60](<[::-60])-[:60]}}
8 \end{rxn}
```



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*.

Example 11

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



The default length of an arrow is 5 em or $5\sqrt{2}$ em if it's a diagonal one. You can change that by using the following command:

```
1 \setarrowlength{<length>}
```

Then the values are <length> or $\text{<length>} \cdot \sqrt{2}$ respectively.

2.5 Package options

myChemistry has a number of package options.

chemstyle load the 'chemstyle' package, without conflicts with **myChemistry**.

color=<colour> This loads 'chemexec' with the options **color<colour>** & **shade=true**.

english With this option, **myChemistry** loads the english version of 'chemexec', if the package isn't loaded separately before. The name of the **rxnscheme** environment (see [section 4.15](#)) is changed into "Reaction scheme".

nochemexec prevent **myChemistry** from loading 'chemexec'.

nocolor 'chemexec' is loaded without colour and with the option **shade=false** (default behaviour of **myChemistry**).

nocompounds prevent **myChemistry** from loading 'chemcompounds'.

nomhchem prevent **myChemistry** from loading 'mhchem' lädt, if 'chemexec' isn't loaded either.

nopackages prevent **myChemistry** from loading *any* package (except **ChemFig**)¹.

placement=<position> The default placement behaviour of the **rxnscheme** environment is changed to <position>.

shade loads 'chemexec' with the **shade=true** option.

¹apart from the ones **myChemistry** needs to function (like **TikZ** etc.).

3 Advanced Usage, Usage with TikZ

The biggest problem with **myChemistry** usually is the correct positioning of reactants and arrows. [section 3.1](#) looks a little bit into this topic.

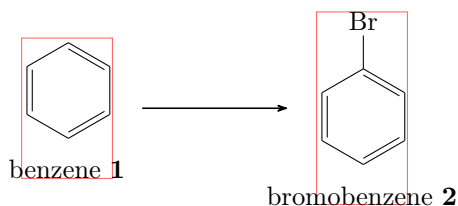
Some of the commands can be given **TikZ** code as third optional argument. More precisely you can use the same **TikZ** keys there as you would with a `\node` inside a `tikzpicture`. If a node is placed with `\node[<tikz>](<placement>){<anything>};`, then `<tikz>` is about the same in e.g. `\reactant[,,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

3.1 The Alignment Question

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

Example 12

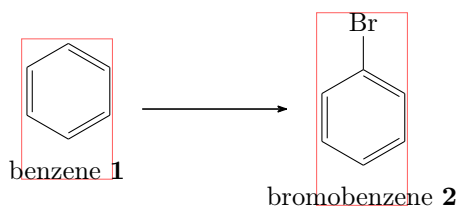
```
1 \makevisible
2 \begin{rxn}
3   \reactant{ \chemname{\chemfig{*6(==--=)}}{benzene \compound{benzene}}
4 }
5   \arrow{}}{}
6   \reactant{ \chemname{\chemfig{*6(==--(-Br)-)}}{bromobenzene \compound
7     {bromobenzene}}} }
8 \end{rxn}
```



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 **TikZ** code fails:

Example 13

```
1 \makevisible
2 \begin{rxn}
3   \reactant[, , yshift=-1em]{ \chemname{\chemfig{*6(==--=)}}{benzene \
4     \compound{benzene}}} }
5   \arrow{}}{}
6   \reactant{ \chemname{\chemfig{*6(==--(-Br)-)}}{bromobenzene \compound
7     {bromobenzene}}} }
8 \end{rxn}
```



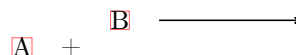
This is, because the first reactant is shifted with the respect to the object it refers to. Since it is the first object on the chain itself, it isn't shifted at all. The following arrow always is centered to the object before.

Example 14

```

1 \makevisible
2 \begin{rxn}
3 \reactant{A}
4 \chemand
5 \reactant[, , yshift=1em]{B}
6 \arrow{}{}
7 \end{rxn}

```



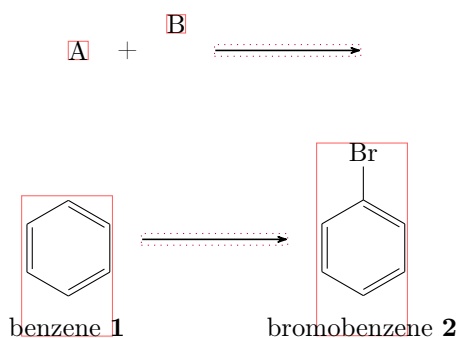
Since there is no possibility to change the alignment of the arrow itself (yet), what you can do is put it inside a branch.

Example 15

```

1 \makevisible
2 \begin{rxn}
3 \reactant{A}
4 \chemand
5 \reactant[, , yshift=1em]{B}
6 \branch[, , yshift=-1em]{\arrow{}{}}
7 \end{rxn}
8 \begin{rxn}
9 \reactant{ \chemname{\chemfig{*6(---)}}{benzene \compound{benzene}}
10 \branch[, , yshift=1em]{\arrow{}{}}
11 \reactant{ \chemname{\chemfig{*6(---(-Br)-)}}{bromobenzene \compound
12 \end{rxn}

```



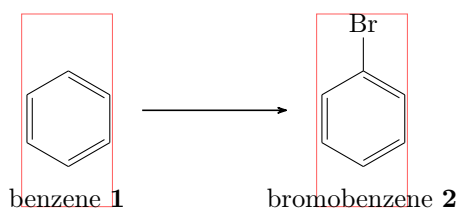
For the last example this isn't the best solution, though, because exact alignment needs lots of tries until you get the required result. There is another solution: an invisible bromine to the first benzene.

Example 16

```

1 \makevisible
2 \begin{rxn}
3   \reactant{ \chemname{\chemfig{*6(==(-[,,,draw=none]\phantom{Br})-)} }
4   \arrow{}{}
5   \reactant{ \chemname{\chemfig{*6(==(-Br)-)} }{bromobenzene \compound
6 \end{rxn}

```



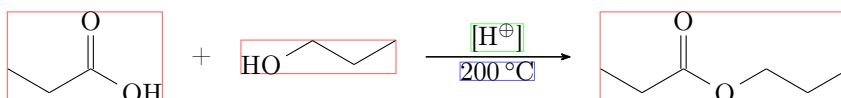
In other cases, too, an invisible substituent should be preferred over TikZ code, since it's easier and more precise:

Example 17

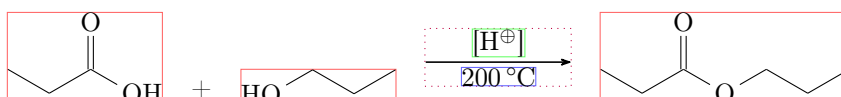
```

1 \makevisible
2 default:
3 \begin{rxn}
4 \reactant{\chemfig{-[: -30] -[:30] (= [2] O) -[: -30] OH}}
5 \chemand
6 \reactant{\chemfig{HO -[:30] -[: -30] -[:30]}}
7 \arrow{\[\Hpl]\}{\SI{200}{\celsius}}
8 \reactant{\chemfig{-[: -30] -[:30] (= [2] O) -[: -30] O -[:30] -[: -30] -[:30]}}
9 \end{rxn}
10 hydroxy groups at the same height through TikZ:
11 \begin{rxn}
12 \reactant{\chemfig{-[: -30] -[:30] (= [2] O) -[: -30] OH}}
13 \chemand[, , yshift=-1.2em]
14 \reactant[, , yshift=.12em]{\chemfig{HO -[:30] -[: -30] -[:30]}}
15 \branch[, , yshift=1.08em]{\arrow{\[\Hpl]\}{\SI{200}{\celsius}}}
16 \reactant{\chemfig{-[: -30] -[:30] (= [2] O) -[: -30] O -[:30] -[: -30] -[:30]}}
17 \end{rxn}
18 hydroxy groups at the same height through an invisible substituent:
19 \begin{rxn}
20 \reactant{\chemfig{-[: -30] -[:30] (= [2] O) -[: -30] OH}}
21 \chemand
22 \reactant{\chemfig{HO -[:30] (= [2, , , draw=none]\phantom{O})
23 -[: -30] -[:30]}}
24 \arrow{\[\Hpl]\}{\SI{200}{\celsius}}
25 \reactant{\chemfig{-[: -30] -[:30] (= [2] O) -[: -30] O -[:30] -[: -30] -[:30]}}
26 \end{rxn}
default:

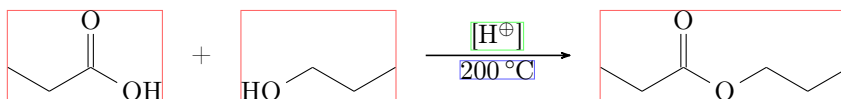
```



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. Maybe further versions of **myChemistry** will provide a more user friendly alignment syntax.

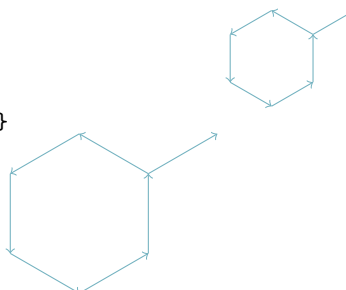
3.2 Using TikZ to Achieve Other Results

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

Example 18

```

1 \begin{rxn}
2   \reactant[,,->,green!45!blue
   !55]{ \chemfig{*6(---(-)---)} }
3 \end{rxn}
4 \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:

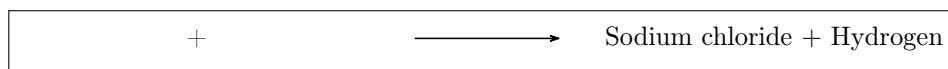
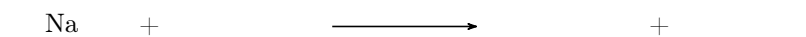
Example 19

```

1 \newcommand{\emptyreactant}{\reactant[,minimum width=5em]{ \rule[-1em
   ]{1em}{.5pt}};\rule[-1em]{3em}{.5pt} }{} }
2 \newcommand{\stoich}{\rule[-3pt]{1em}{.5pt}}
3 \begin{rxn}
4   \reactant{\bf\Large Ionic compounds (I)}
5   \reactant[below,a,yshift=1em]{Fill in the missing parts}
6   \branch[below=of a,b]{ \reactant[,minimum width=5em]{ \stoich\ Na }\chemand \emptyreactant \arrow{}{} \emptyreactant \chemand \emptyreactant }
7   \branch[below=of b,,draw,inner sep=3pt]{\reactant[,minimum width=5em ]{} \chemand \reactant[,minimum width=5em ]{} \arrow{}{} \reactant[,minimum width=5em ]{Sodium chloride $+$ Hydrogen}}
8 \end{rxn}
```

Ionic compounds (I)

Fill in the missing parts



4 Alphabetical Command Reference

In the following section every command is explained.

4.1 anywhere

NEW Sometimes it is useful to be able to place a reactant outside of the chain, e. g. see [scheme 12](#).

```
1 \anywhere[<anchor>]{<tikz>}{<formula>}
```

The command `\anywhere` makes that possible. It is placed through `<tikz>` in a similar way as `\branch`.

Example 20

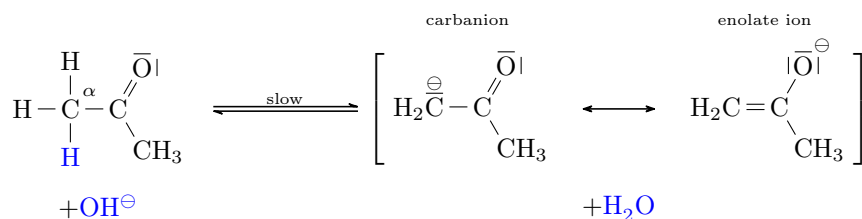
```
1 \begin{rxn}
2 \reactant[,carbonyl_A]{\chemfig{R_2C=O}}
3 \anywhere{above=of carbonyl_A}{\chemfig{H-[: -30]O-[:30]H}};
4 \end{rxn}
```

$\text{H}-\text{O}-\text{H}$
 $\text{R}_2\text{C}=\text{O}$

This command can be used for naming reaction partners.

Example 21

```
1 \begin{rxn}
2 \reactant[,ketone]{\chemfig{H-\chemabove{C}{\hspace*{5mm}\scriptstyle\alpha}}(-[2]H)(-[6,,,2]{\textcolor{blue}{H}})-C(=[:60]\lewis{02,0})-[: -60]C|H_3}}
3 \anywhere{below=of ketone}{\textcolor{blue}{Hyd}}
4 \arrow[,<=>]{\tiny slow}{}
5 \mesomeric[,mesomer]{
6 \reactant[,carbanion]{\chemfig{H_2|\chemabove[3pt]{\lewis{2,C}}{\scriptstyle\ominus}}-C(=[:60]\lewis{02,0})-[: -60]C|H_3}}
7 \marrow
8 \reactant[,enolate]{\chemfig{H_2C=C(-[:60]\chemabove{\lewis{024,0}}{\hspace*{5mm}\scriptstyle\ominus})-[: -60]C|H_3}}
9 }
10 \anywhere{above=of enolate}{\tiny enolate ion}
11 \anywhere{above=of carbanion}{\tiny carbanion}
12 \anywhere{below=of mesomer}{\textcolor{blue}{ce{H2O}}}
13 \end{rxn}
```



4.2 arrow

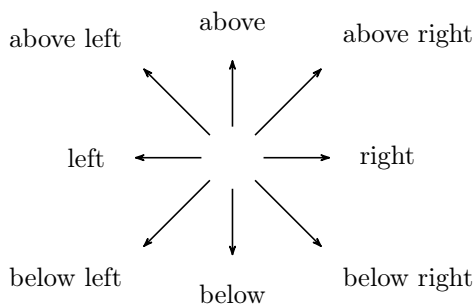
Reaction arrows are created with `\arrow`.

```
1 \arrow[<direction>,<type>,<length factor>,<anchor>,<both>
   ]{<above>}{<below>}
```

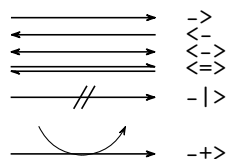
4.2.1 Options

There are five options to customize the arrows. You've got to use them in the right order, separated by commas.

1. `<direction>` – possible values are:



2. `<type>` – possible values are:



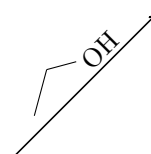
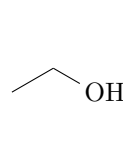
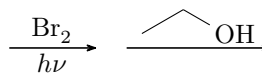
3. `<length factor>` – the length (5.0em, factor = 1.0, default) is multiplied with this factor.
4. `<anchor>` – this anchor can be used to refer to the arrow, e.g. with a branch.
5. `<both>` – this gives both arrow argument nodes the same width and height.

Example 22

```

1 \begin{rxn}
2 \arrow[,,,6]{\ce{Br2}}{\$h\nu$}
3 \arrow{\chemfig{-[:30]-[: -60]OH}}{}}
4 \end{rxn}
5 \begin{rxn}
6 \arrow[above,,,6]{\ce{Br2}}{\$h\nu$}
7 \end{rxn}
8 \begin{rxn}
9 \arrow[above]{\chemfig{-[:30]-[: -60]OH}}{}}
10 \end{rxn}
11 \begin{rxn}
12 \arrow[above right,,,5]{\ce{Br2}}{\$h\nu$}
13 \end{rxn}
14 \begin{rxn}
15 \arrow[above right]{\chemfig{-[:30]-[: -60]OH}}{}}
16 \end{rxn}

```



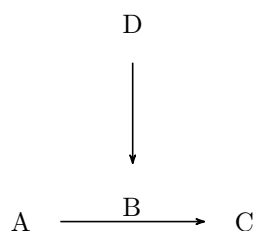
Most keys in action:

Example 23

```

1 \begin{rxn}
2 \reactant{A}
3 \arrow[,,,arrow_a]{B}{}
4 \branch[above=of arrow_a,,
5 yshift=1em]{
6 \arrow[above,<-,7]{}{}
7 \reactant[above]{D}
8 }
9 \reactant{C}
10 \end{rxn}

```

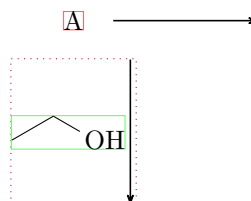


4.2.2 Alignment

If an arrow is placed inside a branch (see [section 4.3](#)) the alignment of the branch possibly is determined by the width and height of the arrow arguments. If these arguments have different sizes, the alignment can go wrong.

Example 24

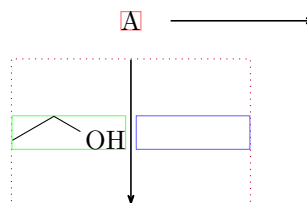
```
1 \makevisible
2 \begin{rxn}
3 \reactant[,a]{A}
4 \arrow{}{}
5 \branch[below=of a]{
6 \arrow[below]{\chemfig
7 }
8 \end{rxn}
9 \makeinvisible
```



By using the key `both`, both argument nodes have the same size, which can correct the alignment.

Example 25

```
1 \makevisible
2 \begin{rxn}
3 \reactant[,a]{A}
4 \arrow{}{}
5 \branch[below=of a]{
6 \arrow[below,,,both]{\chemfig
7 }
8 \end{rxn}
9 \makeinvisible
```



There is more about the alignment problem in [section 4.3.1](#).

4.2.3 Appearance

With the commands `\setarrowlength` ([section 4.16](#)) and `\setarrowline` ([section 4.19](#)) the general appearance of the arrows can be changed.

4.3 branch

`\branch` is used to, well, create a branch to a reaction. If you used earlier versions of **myChemistry** please be aware, that the command syntax has changed with **v1.3**.

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

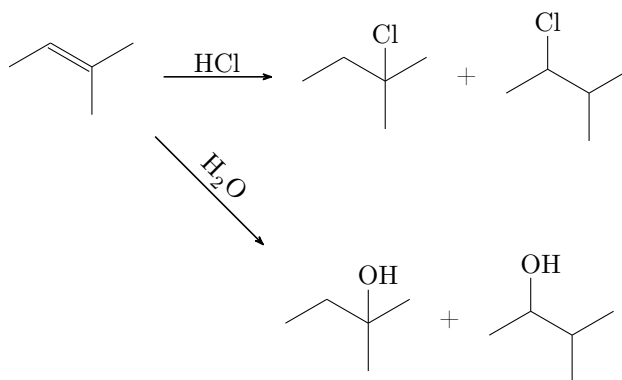
For `\branch` alignment an anchor is important. Let's take a look at an example:

Example 26

```

1 \begin{rxn}
2 \reactant[,start]{\chemfig{-[:30]=_[: -60](-[: -60]) -[:60]}}
3 \arrow[,,.75]{\ce{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[below right, ,.75]{\ce{H2O}}{}
9   \reactant[below right]{\chemfig{-[:30] -[: -60](-[:120]OH)
10    (-[: -60]) -[:60]}}
11   \chemand
12   \reactant{\chemfig{-[:30](-[:60]OH) -[: -60](-[: -60]) -[:60]}}
13 }
13 \end{rxn}

```



The first reactant got the anchor `start` (line 2, also see [section 4.13](#)).

```
2 \reactant[,start]{ ... }
```

`\branch` now refers to it in its alignment (line 7):

```
7 \branch[below right=of start]{ ... }
```

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`.

Example 27

```

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

```

$\text{CH}_2=\text{CH}-\text{OH} \rightleftharpoons \text{CH}_3-\text{CH}=\text{O}$

You have several options for alignment: you can either put the `\branch` on a `chain` or place it relative to a object.

chain In this case you use: `on chain=going <value>`.

relative In this case you use: `<value>=of <anchor>`.

As possible `<value>` you have the same choices as for `\reactant`, see [section 4.13](#). Default is on `chain=going right`.

4.3.1 Alignment problems

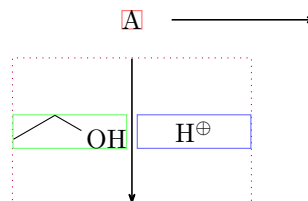
If an arrow has two arguments with different sizes and is placed inside a branch, the alignment of the branch can go wrong. In this case the `\arrow` key `both` isn't a solution, since the smaller argument then is not placed next to the arrow but is centered in its node.

Example 28

```

1 \makevisible
2 \begin{rxn}
3 \reactant[,a]{A}
4 \arrow{}{}
5 \branch[below=of a]{
6 \arrow[below,,,both]{\chemfig
7 }
8 \end{rxn}
9 \makeinvisible

```



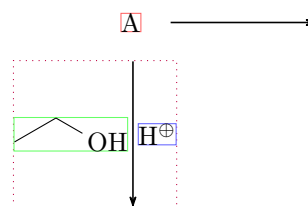
What you have to do is shift the branch using the *TikZ* keys `xshift` and `yshift`.

Example 29

```

1 \makevisible
2 \begin{rxn}
3 \reactant[,a]{A}
4 \arrow{}{}
5 \branch[below=of a,,xshift
   =-1.35em]{
6 \arrow[below]{\chemfig
   {-[::30]-[::-60]OH}}{\Hpl}
7 }
8 \end{rxn}
9 \makeinvisible

```

**4.4 chemand**

The command

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

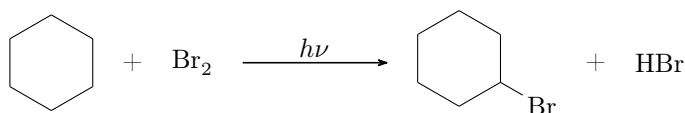
produces and places a + in the same way `\reactant` places arbitrary text.

Example 30

```

1 \begin{rxn}
2 \reactant{\chemfig{*6(-----)}}
3 \chemand
4 \reactant{\ce{Br2}}
5 \arrow{$h\nu$}{}
6 \reactant{\chemfig{*6(--(-Br)----)}}
7 \chemand
8 \reactant{\ce{HBr}}
9 \end{rxn}

```



The optional arguments for `\chemand` and `\reactant` are the same, see [section 4.13](#) for `\reactant`'s options.

4.5 dummy

NEW

`\dummy` creates an empty node. Up to v1.3 `\arrow` needed to follow after a node, because `\arrow` internally uses `\tikzchainprevious`. If there was no node on the chain *before* `\arrow` was used, it caused an error. With `\branch` it was similar. By using `\dummy` you could start a scheme with an arrow anyway.

```
1 \begin{rxn}
```

```

2   \dummy\arrow{}{}
3   \end{rxn}

```

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

4.6 elmove

`\elmove` just is a shortcut for `ChemFig`'s `\chemmove`.

```

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

```

This is expanding the command

```

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

```

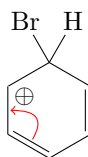
using `[->,red,shorten <=3pt,shorten >=1pt]` as default for `<tikz>`. How you use `\chemmove` is described in the documentation for `ChemFig`.

Example 31

```

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

```



4.7 makeinvisible

`\makeinvisible` restores the normal `myChemistry` behaviour after `\makevisible` (see section 4.8) has been used. `\makeinvisible` only changes the looks of nodes following after it.

4.8 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 for `\makevisible` in section 4.2. Every kind of nodes is emphasized with a different colour:

`\reactant{}`, `\arrow{above}{}`, `\arrow{below}{}` und `\branch{}`. Also see section 4.7.

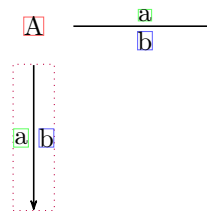
`\makevisible` only changes the looks of nodes following after it.

Example 32

```

1 \makevisible
2 \begin{rxn}
3 \reactant[,a]{A}
4 \arrow{a}{b}
5 \branch[below=of a]{
6   \arrow[below,,,both]{a}{b}
7 }
8 \end{rxn}
9 \makeinvisible

```



4.9 marrow

`\marrow` creates a double-headed arrow.

```
1 \marrow[<direction>]
```

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

4.10 mCsetup

With

```
1 \mCsetup{<keys>}
```

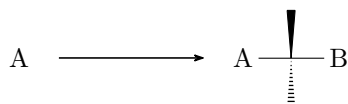
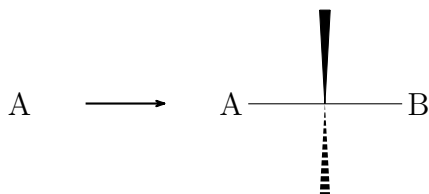
you can fully customize **myChemistry**. For each of **myChemistry**'s commands going like `\set<command>` except `\setbondshape` there is a key `<command>=<value>`. Additionally there is the key `align=<value>`, with which you can change the alignment behaviour of both `rxn` and `rxnscheme`, and the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

Example 33

```

1 \mCsetup{
2   align=left,
3   arrowlength=3em,
4   arrowline=thick,
5   atomsize=\large,
6   bondlength=3em,
7   mergelength=4em,
8   rcndist=2em
9   %rxnalign=right,
10  %schemealign=left
11 }
12 \setbondshape{4pt}{2pt}{1pt}
13 \begin{rxn}
14 \reactant{A}\arrow{}{}\reactant{\chemfig{A-(<[2])(<:[6])-B}}
15 \end{rxn}
16 \mCsetup{reset}
17 \begin{rxn}
18 \reactant{A}\arrow{}{}\reactant{\chemfig{A-(<[2])(<:[6])-B}}
19 \end{rxn}

```

**4.11 merge****NEW**

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 [section 4.13](#) and [section 4.3](#)) and be placed *before* `\merge`.

```

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

```

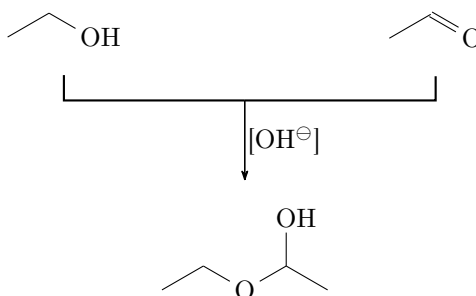
`\merge` has three optional and three mandatory arguments. The latter are the anchor names of the reactants, that are to be merged.

Example 34

```

1 \begin{rxn}
2   \branch[,first]{
3     \reactant[,start_a]{\chemfig{-[:30]-[: -30]OH}}
4     \reactant[,start_b,xshift=9em]{\chemfig{-[:30]=[: -30]O}}
5   }
6   \branch[below=of first,target,yshift=-5em]{
7     \reactant{\chemfig{-[:30]-[: -30]O-[:30](-[2]OH)-[: -30]}}
8   }
9   \merge[\ce{[\Hyd]}]{target}{start_a}{start_b}
10 \end{rxn}

```



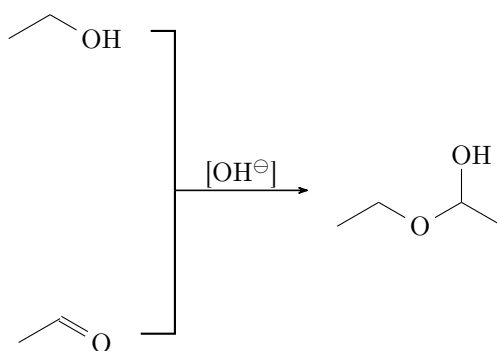
The default <direction> is **below**, other possible values are **right**, **left** or **above**. With <length> the length of the arrow from the point of merging to the tip can be changed. The default length is 3em. The default length can be changed with `\setmergelength` or `\mCsetup`.

Example 35

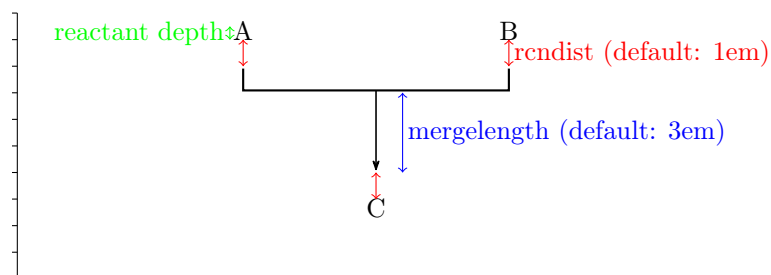
```

1 \begin{rxn}
2   \reactant[,start_a]{\chemfig{-[:30]-[: -30]OH}}
3   \branch[below=of start_a,start_b,yshift=-9em]{
4     \reactant{\chemfig{-[:30]=[: -30]O}}
5   }
6   \branch[right=of start_a,target,xshift=7em,yshift=-6em]{
7     \reactant{\chemfig{-[:30]-[: -30]O-[:30](-[2]OH)-[: -30]}}
8   }
9   \merge[\ce{[\Hyd]}],right,5em{target}{start_a}{start_b}
10 \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 4.22](#)) and the length of the **\merge** arrow **mergelength** (also see [section 4.21](#)).



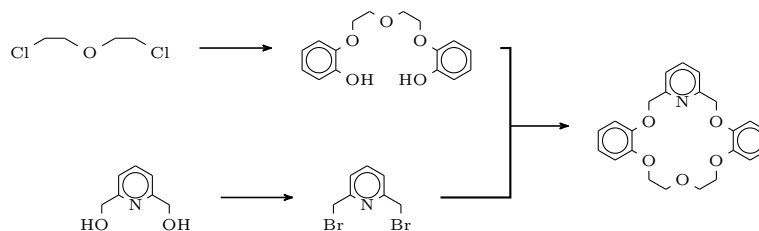
At last an example where two chains are merged.

Example 36

```

1 \begin{rxn}
2 \setatomsep{1em}\tiny
3 % chain 1
4 \reactant[,first]{ \chemfig{C1-[:30,1.5]--[: -30,1.5]O
   -[:30,1.5]--[: -30,1.5]C1}{ } }
5 \arrow[,,.5]{ }
6 \reactant[,start_above]{ \chemfig{O(-[: -150]**6(-----(-OH)-))
   -[:90]-[:30]-[: -30]O-[:30]-[: -30]-[: -90]O-[: -30]**6(-(-HO)-----)} }
7 % chain 2
8 \branch[below=of first,start_below,xshift=8em,yshift=-4em]{
9 \reactant{ \chemfig{**6((--[6,,2]HO)-N(--[6]OH)-----)} }
10 \arrow[,,.5]{ }
11 \reactant{ \chemfig{**6((--[6]Br)-N(--[6]Br)-----)} }
12 }
13 % target
14 \branch[right=of start_above,target,xshift=5em,yshift=-4em]{
15 \reactant[,c]{ \chemfig{O(-[: -150]**6(-----(-O?) -)) -[:90]-[:30]**6(-N
   -(--[: -90]O-[: -30]**6(-(-O-[6]-[: -150]-[:150]O-[: -150]-[:150]? )-----))
   -----)} }
16 }
17 % merging:
18 \merge[,right]{target}{start_above}{start_below}
19 \end{rxn}

```



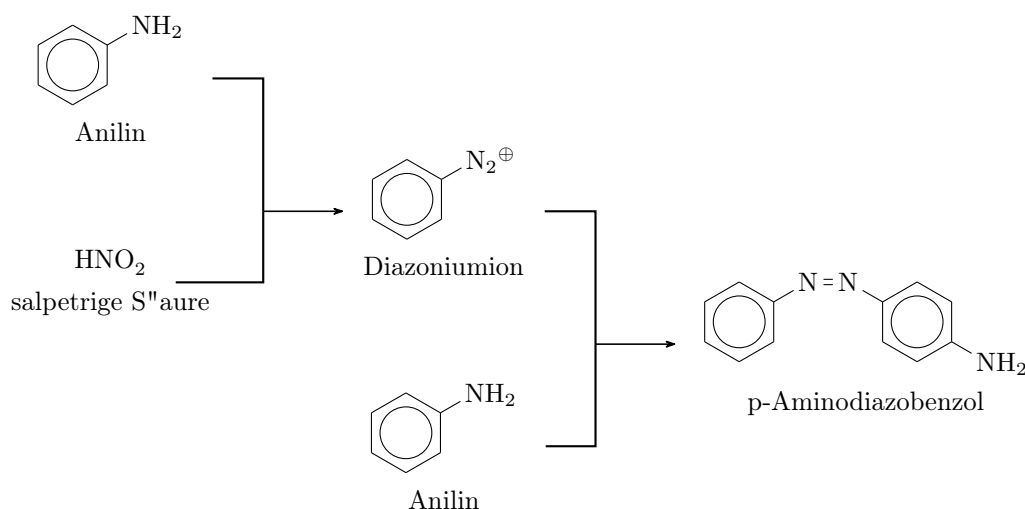
The usage of `\merge` may require some playing with branches, `xshift` and `yshift`, until you get the desired result.

Example 37

```

1 \begin{rxn}
2 \setatomsep{1.5em}
3 \reactant[,start_aa]{ \chemname{\chemfig{**6(---(-NH_2)---)}}{Anilin}
4 \reactant[below,start_ab,yshift=-3em]{ \chemname{\ce{HNO2}}{salpetrige
  S"aure} }
5 \branch[right=of start_aa,target_a,xshift=6em,yshift=-5em]{
6 \reactant{ \chemname{\chemfig{**6(---(-N|_2\op)---)}}{Diazoniumion}
7 }% = start_ba
8 \branch[below=of target_a,start_bb,yshift=-3em]{
9 \reactant{ \chemname{\chemfig{**6(---(-NH_2)---)}}{Anilin} }
10 }
11 \branch[right=of target_a,target_b,xshift=6em,yshift=-5em]{
12 \reactant{ \chemname{\chemfig{N(-[: -150]**6(-----))=N
  -[: -30]**6(---(-NH_2)---)}}{p-Aminodiazobenzol} }
13 }
14 \merge[,right]{target_a}{start_aa}{start_ab}
15 \merge[,right]{target_b}{target_a}{start_bb}
16 \end{rxn}

```

**4.12 mesomeric**

The `\mesomeric` command works just like `\branch` (see [section 4.3](#)) but places the formulæ into square brackets. If you used earlier versions of **myChemistry** please be aware, that the command syntax has changed with v1.3.

```

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

```

The resonance formulæ are written into `<formula>`. With `\marrow` (see [section 4.9](#)) you create the resonance arrows. If needed you can give an anchor (`<anchor>`) to `\mesomeric`

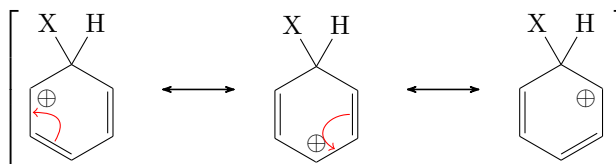
(also see [section 4.3](#)). Alignment is used the same way as with `\reactant`.

Example 38

```

1 \begin{rxn}
2 \mesomeric{
3   \reactant{
4     \chemfig{*6(=[@{e1}]--(-[:120]X)(-[:60]H)-(-[:30],.4,,white)\
      oplus)-[@{e2}]})}
5     \elmove{e1}{60:4mm}{e2}{0:4mm}
6   }
7   \marrow
8   \reactant{
9     \chemfig{*6(-(-[:90],.4,,white)\oplus)-[@{e4}]=[@{e3}]-(-[:120]X)
      (-[:60]H)-=)}
10    \elmove{e3}{180:4mm}{e4}{150:4mm}
11  }
12  \marrow
13  \reactant{
14    \chemfig{*6(--(-[:150],.4,,white)\oplus)-(-[:120]X)(-[:60]H)-=)}
15  }
16 }
17 \end{rxn}

```



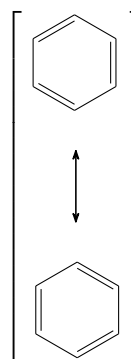
Or vertical, too:

Example 39

```

1 \begin{rxn}
2 \mesomeric{
3   \reactant{ \chemfig
4     {*6(==--==)} }
5   \marrow[below]
6   \reactant[below]{ \chemfig
7     {*6(==--==)} }
8 }
9 \end{rxn}

```



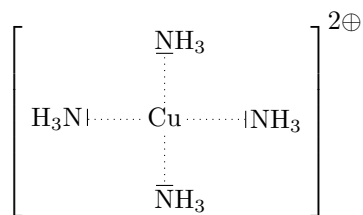
Or maybe a coordination complex?

Example 40

```

1 \begin{rxn}
2 \setatomsep{3em}
3 \mesomeric[,a]{
4   \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} }
5 }
6 \node[above right=of a,yshift=-1em] {$2\oplus$};
7 \end{rxn}

```

**4.13 reactant****NEW**

The command `\reactant1` is somehow the basic command of **myChemistry**. If you used earlier versions of **myChemistry** please be aware, that the command syntax has changed with v1.3.

```
1 \reactant[<alignment>,<anchor>,<tikz>]{<formula>}
```

In this command the actual formulæ are written (<formula>). If needed, they can be given an anchor (<anchor>). The optional argument <alignment> can have 8 different values:

- a) right,
- b) above right,
- c) above,
- d) above left,
- e) left,
- f) below left,
- g) below,
- h) below right

¹In older versions this command was called `\reactand`, which is still available.

Default is `right`. You use this argument to place the reactant relatively to the reactant or arrow right before.

Example 41

1 <code>horizontal:</code>	horizontal:
2 <code>\begin{rxn}</code>	
3 <code>\reactant{\ce{Br2}}</code>	Br_2
4 <code>\reactant[below]{\ce{Cl2}}</code>	Cl_2
5 <code>\end{rxn}</code>	
6	
7 <code>more than one reactant:</code>	more than one reactant:
8 <code>\begin{rxn}</code>	
9 <code>\reactant{\ce{Br2}}</code>	Br_2
10 <code>\reactant[below]{\ce{I2}}</code>	I_2
11 <code>\reactant{\ce{Cl2}}</code>	Cl_2
12 <code>\end{rxn}</code>	
13	vertical reaction:
14 <code>vertical reaction:</code>	$\text{Br}-\text{Br}$
15 <code>\begin{rxn}</code>	
16 <code>\reactant{\ce{Br-Br}}</code>	$\text{Br}-\text{Br}$
17 <code>\arrow[below, ,.5]{\h\nu}</code>	\downarrow
18 <code>\reactant[below]{\ce{2 ~\lewis{0.,Br}}}</code>	$2 \text{ Br}\cdot$
19 <code>\end{rxn}</code>	

4.14 rxn (environment)

`rxn` is a non-floating not numbered environment for reaction schemes. Per default all schemes are centered.

```

1 \begin{rxn}[<align>,<scalefactor>]
2   ...
3 \end{rxn}
```

4.14.1 Options

NEW `rxn` has two options, which are to be used in the following order, separated by a comma:

1. `<alignment>` alignment behaviour of the `rxn` environment; default is ‘center’.
2. `<scalefactor>` factor by which the `rxn` environment is scaled; default: ‘1.0’. Same behaviour as with `rxnscheme`, see [section 4.15](#).

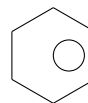
If you use the `scalefactor` option, you might see strange effects on `ChemFig` formulae.

Example 42

```

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

```



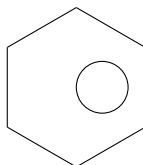
Scaling does in general not affect the size of **ChemFig** formulæ, but does scale the aromaticity ring of benzene and similar molecules. This is due to a possible bug in **ChemFig** itself.

Example 43

```

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

```



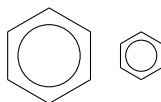
This can be solved either by using the *first* optional argument of **\chemfig** to undo the scaling or by using the *second* optional argument to scale the rest of the molecule.

Example 44

```

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

```



Alignment examples:

Example 45

```

1 \begin{rxn}[center]
2 \reactant{center}\arrow{}}{\}\reactant{centered}
3 \end{rxn}
4 \begin{rxn}[right]
5 \reactant{right}\arrow{}}{\}\reactant{raggedleft}
6 \end{rxn}
7 \begin{rxn}[left]
8 \reactant{left}\arrow{}}{\}\reactant{raggedright}
9 \end{rxn}

```

center \longrightarrow centered

right \longrightarrow raggedleft

left \longrightarrow raggedright

4.15 rxnscheme (environment)

`rxnscheme` is a floating environment for reaction schemes.

```

1 \begin{rxnscheme}[<label>,<placement>,<alignment>,<
   scalefactor>,<name>]{<caption>}
2 ...
3 \end{rxnscheme}

```

4.15.1 Options

NEW `rxnscheme` has five options, which are to be used in the following order, separated by commas:

1. `<label>` Like every other floating environment `rxnscheme` can be given a label. To do that, you need to use the option `<label>`. For example if you use

```

1 \begin{rxnscheme}[rs:schema]{<caption>}
2 ...
3 \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 `htp`. The default value is `H` (exactly here). An example for this option is [scheme 13](#) in [section 5.5](#).

3. `<alignment>` This option changes the alignment of the scheme. You can choose between `left`, `center` and `right`.
4. `<scalefactor>` `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` formulæ. You can have strange effects on `ChemFig` formulæ if you use this key, though. See [section 4.14.1](#) for more information.
5. `<name>` This option changes the name of the actual scheme from "Reaktionschema" or "Reaction scheme" into `<name>`.

```

1  \begin{rxnscheme}[,<placement>]{<caption>}
2      ...
3  \end{rxnscheme}

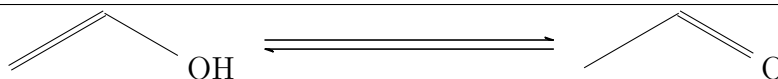
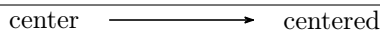
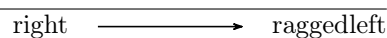
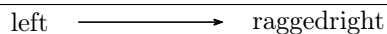
```

Example 46

```

1  \begin{rxnscheme}[,,2]{Big scheme}
2  \large\setatomsep{3.5em}
3  \reactant{ \chemfig{=[::30]-[::-60]OH} }
4  \arrow{,<=>}{}{}
5  \reactant{ \chemfig{-[::30]=[::-60]O} }
6  \end{rxnscheme}
7  \begin{rxnscheme}[,,,5]{Small scheme}
8  \tiny\setatomsep{1em}
9  \reactant{ \chemfig{=[::30]-[::-60]OH} }
10 \arrow{,<=>}{}{}
11 \reactant{ \chemfig{-[::30]=[::-60]O} }
12 \end{rxnscheme}
13 \begin{rxnscheme}{center}
14 \reactant{center}\arrow{}{}\reactant{centered}
15 \end{rxnscheme}
16 \begin{rxnscheme}[,,right]{right}
17 \reactant{right}\arrow{}{}\reactant{raggedleft}
18 \end{rxnscheme}
19 \begin{rxnscheme}[,,left]{left}
20 \reactant{left}\arrow{}{}\reactant{raggedright}
21 \end{rxnscheme}

```

Reaction scheme 2 Big scheme**Reaction scheme 3** Small scheme**Reaction scheme 4** center**Reaction scheme 5** right**Reaction scheme 6** left**4.15.2 Customizing rxnscheme**

Style If you don't like the style of `rxnscheme` you can change it by using

```
1 \floatstyle{<new style>}
2 \restylefloat{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 `rxnscheme`

Example 47

```

1 \begin{rxnscheme}{ruled}
2 \reactant{default style}
3 \end{rxnscheme}
4 \floatstyle{boxed}
5 \restylefloat{rxnfloat}
6 \begin{rxnscheme}{boxed}
7 \reactant{framed object}
8 \end{rxnscheme}
9 \floatstyle{plain}
10 \restylefloat{rxnfloat}
11 \begin{rxnscheme}{plain}
12 \reactant{without any special formatting}
13 \end{rxnscheme}

```

Reaction scheme 7 ruled

default style

framed object

Reaction scheme 8: boxed

without any special formatting

Reaction scheme 9: plain

Placement Usually floating environments have an optional argument for their placement. `rxnscheme`'s default placement is H which means, it is placed *exactly here*. If you want to change it into htp or something, you can use

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

It's easier, though, loading **myChemistry** with the 'placement' option:

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

This will change the default placement behaviour from H to <placement>. You can also change the placement behaviour of just one `rxnscheme` environment by using the placement option:

```

1 \begin{rxnscheme}[,<placement>]{<caption>}
2 ...
3 \end{rxnscheme}

```

Name If you want to change the name of reaction scheme¹, you can do that with

¹You probably do. You reading the English documentation means probably, that you're not German.


```
1 \setschemename{<new name>}
```

The default name is "Reaktionschema" or, with package option 'english', "Reaction scheme".

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

```
1 \makeatletter
2 \@addtoreset{rxnfloat}{section}
3 \makeatletter
4 \renewcommand{\therxnscheme}{\arabic{section}.\arabic{
  rxnscheme}}
```

the counter is reset with every new section and looks like `section.rxnscheme`. Please be aware, that you have to write `\@addtoreset` between `\makeatletter` and `\makeatother` because of the @.

List of schemes By writing

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

you can create a list of all schemes created with `rxnscheme`.

Example 48

Reaction schemes

1	Keto-enol tautomerization . . .	11
2	Big scheme	39
3	Small scheme	39
4	center	39
5	right	39
6	left	39
7	ruled	40
8	boxed	40
9	plain	40
10	addition reaction	45
11	electrophilic substitution	48
12	Hydratisation	57
13	synthesis of chrysanthemum acid	61

4.16 setarrowlength

The default length of an reaction arrow is 5.0em or $5.0 \cdot \sqrt{2}$ em for the diagonal ones. You can change these values using

```
1 \setarrowlength{<length>}
```

into `<length>` or `<length>·√2`, respectively. Mind the fact that you have to use a length unit. If you leave the argument empty, the length is reset to default.

This command replaces `\arrowlength` from earlier versions.

4.17 setatomsiz

With

```
1 \setatomsiz{<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.

This command replaces `\atomsiz` from earlier versions.

4.18 setbondlength

With

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

you can change `\setatomsep{<length>}` for all **ChemFig** formulæ *inside* of the **myChemistry** environments. Default value is 1.8em. If you leave the argument empty, the length is reset to default.

This command replaces `\bondlength` from earlier versions.

4.19 setarrowline

NEW

With the command

```
1 \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
```

The values `very thick` and `ultra thick` should not be used.

This command also applies to `\merge`.

4.20 setbondshape

With

```
1 \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.

This command replaces `\bondshape` from earlier versions.

4.21 setmergelength

NEW

With

```
1 \setmergelength{<länge>}
```

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 4.11](#)). If you leave an argument empty, the value is reset to default (3 em).

4.22 setrcndist

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

```
1 \setrcndist{<länge>}
```

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

Example 49

```
1 \setrcndist{2em}
2 \begin{rxn}
3 \reactant{A}\arrow{>}{}
4 \end{rxn}
5 \setrcndist{}
6 \begin{rxn}
7 \reactant{A}\arrow{>}{}
8 \end{rxn}
```

A \longrightarrow

A \longrightarrow

4.23 setrxnalign/setschemealign

With the commands

```
1 \setrxnalign{<alignment>}
2 \setschemealign{<alignment>}
```

The default alignment behaviour of `rxn` and `rxnscheme` (see [section 4.14.1](#) & [section 4.15.1](#)) can be set. You can choose between `left`, `center` and `right`.

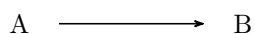
If you leave the argument empty, **myChemistry**'s default behaviour (`center`) is restored.

Example 50

```

1 \setrxnalign{right}
2 \begin{rxn}
3 \reactant{A}\arrow{ }{}\reactant{B}
4 \end{rxn}
5 \setrxnalign{ }
6 \begin{rxn}
7 \reactant{A}\arrow{ }{}\reactant{B}
8 \end{rxn}

```

**4.24 setschemename**

See [section 4.15.2](#).

4.25 transition

`\transition` works exactly like `\reactant` (see [section 4.13](#)). If you used earlier versions of **myChemistry** please be aware, that the command syntax has changed with v1.3.

```

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

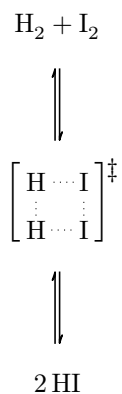
```

Example 51

```

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

```

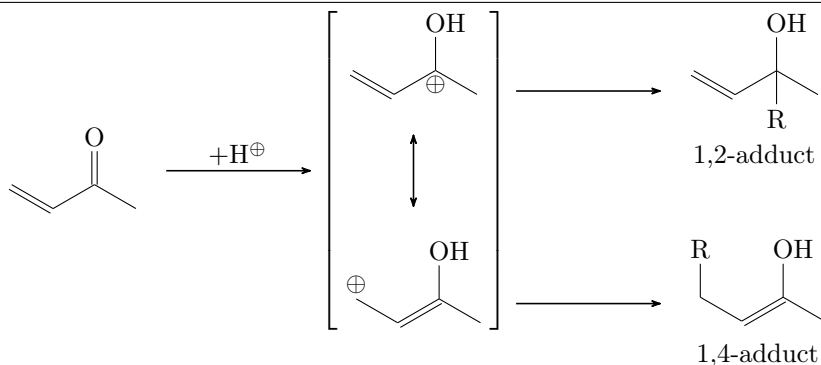


5 Examples

5.1 Addition Reaction

A simple reaction scheme with two different products.

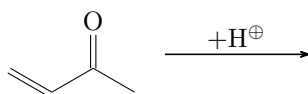
Reaction scheme 10 addition reaction



Let's take a closer look, step after step. At first we write the first reactant and the reaction arrow.

```

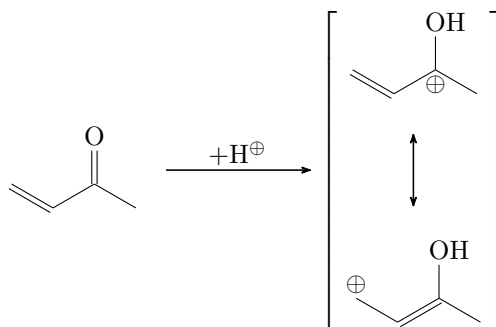
1 \reactant{ \chemfig{=_[::-30] -[::60] (=[::60] O) -[::-60]} }
2 \arrow{ $+ \Hpl$ }{}
```



Then we write the resonance formulæ. The `\mesomeric` gets the anchor `rf` (line 7).

```

3 \mesomeric[,rf]{
4   \reactant{ \chemfig{=_[::-30] -[::60] (-[::60] OH)
5     (-[::-120,.3,,,white]\oplus) -[::-60]} }
6   \marrow[below]
7   \reactant[below]{ \chemfig{\oplus -[6,.3,,,white
8     ] -[::-30] =_[[::60] (-[::60] OH) -[::-60]} }
9 }
```

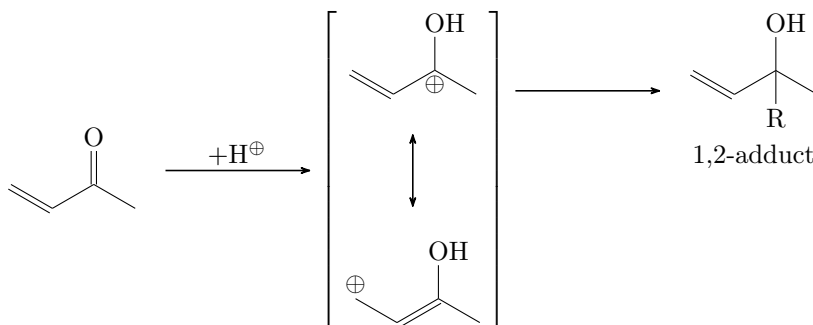


Now comes the 1,2-adduct, placed in a branch referring `rf`, shifted above with `yshift`:

```

8   \branch[right=of rf,,yshift=3em]{
9     \arrow{}}{}
10    \reactant{ \chemname{\chemfig{=_{[: -30] -[: :60] (-[: :60] OH
11              ) (-[: : -120] R) -[: : -60]}}}{1,2-adduct} }

```

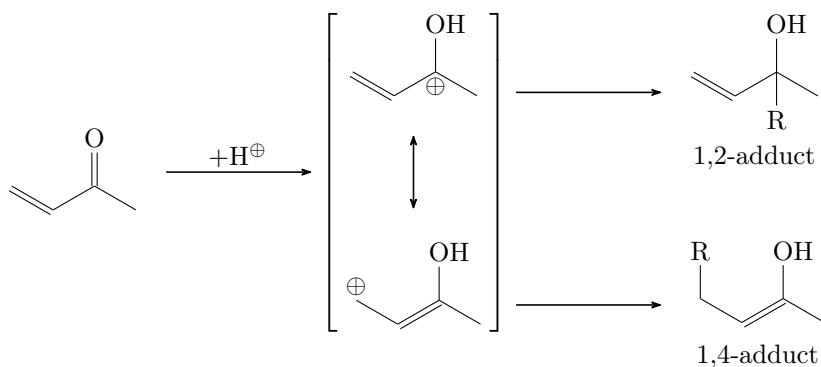


At last we write the branch containing the 1,4-adduct, also referencing `rf`, shifted below with `yshift`:

```

12   \branch[right=of rf,,yshift=-5em]{
13     \arrow{}}{}
14    \reactant{ \chemname{\chemfig{R
15              -[6] -[: -30] =_{[: :60] (-[: :60] OH) -[: : -60]}}}{1,4-adduct}

```



The complete code looks like this:

```

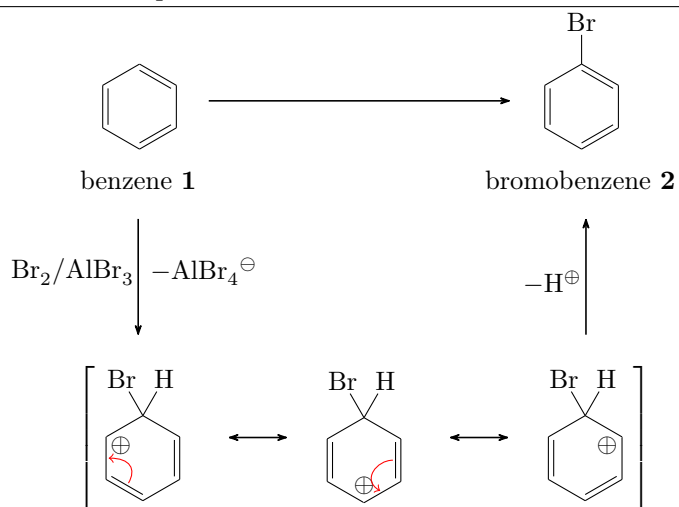
1  \begin{rxnscheme}{addition reaction}
2  \reactant{ \chemfig{=[:-30]-[:60](=[:60]O)-[:60]}
3  }
4  \arrow{ $+ \text{Hpl}$ }{}
5  \mesomeric[,rf]{
6  \reactant{ \chemfig{=[:-30]-[:60](-[:60]OH)
7  (-[:-120,.3,,,white]\oplus)-[:60]} }
8  \marrow[below]
9  \reactant[below]{ \chemfig{\oplus-[6,.3,,,white
10 ]-[:-30]=[:60](-[:60]OH)-[:60]} }
11 }
12 \branch[right=of rf,,yshift=3em]{
13 \arrow{}}{}
14 \reactant{ \chemname{\chemfig{=[:-30]-[:60](-[:60]
15 OH)([:-120]R)-[:60]}}{1,2-adduct} }
16 }
17 \branch[right=of rf,,yshift=-5em]{
18 \arrow{}}{}
19 \reactant{ \chemname{\chemfig{R
20 -[6]-[:-30]=[:60](-[:60]OH)-[:60]}}{1,4-adduct
21 } }
22 }
23 \end{rxnscheme}

```

5.2 Mesomerism

We want to display the following reaction scheme:

Reaction scheme 11 electrophilic substitution

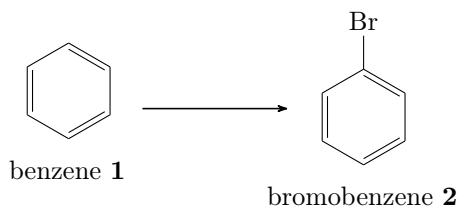


First we write the main reaction. To do so, we use the commands `\reactant`, `\arrow` and the environment `\begin{rxnscheme} ... \end{rxnscheme}`.

```

1 \begin{rxnscheme}{electrophilic substitution}
2   \reactant{
3     \chemname{\chemfig{*6(==-=)}}{benzene \compound{
4       benzene}}
5   }
6   \arrow{}{}
7   \reactant{
8     \chemname{\chemfig{*6(==(-Br)-)}}{bromobenzene \
9     compound{bromobenzene}}
10  }
11 \end{rxnscheme}

```



Now we make it a little bit smaller:

```

1 \begin{rxnscheme}[,,, .8]{electrophilic substitution}
2   \setatomsep{1.6em}
3   \reactant{
4     \chemname{\chemfig{*6(==-=)}}{benzene \compound{
5       benzene}}

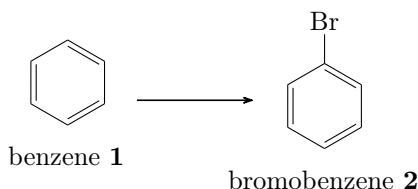
```



```

5      }
6      \arrow{ }{}
7      \reactant{
8          \chemname{\chemfig{*6(==(-Br)-=)}}{bromobenzene \
              compound{bromobenzene}}
9      }
10     \end{rxnscheme}

```



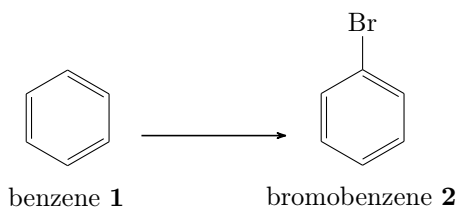
We have two possibilities to align both benzene rings at the same height. Either we shift the second one up using *TikZ* code:

```

7      \reactant[, , yshift=1em]{
8          \chemname{\chemfig{*6(==(-Br)-=)}}{bromobenzene \
              compound{bromobenzene}}
9      }

```

This is not the best solution, because the arrow isn't centered with respect to the rings.

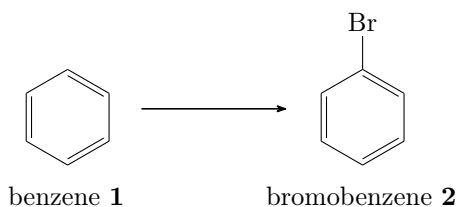


The second possibility would be to shift the first ring down. We can't achieve that by using *TikZ* code, because the following arrow and reactant align themselves with respect to the reactant or arrow directly before. But we can write an invisible bromine to the first benzene to do the trick:

```

3      \reactant{
4          \chemname{\chemfig{*6(==(-[, , , white]\phantom{Br})-=)}}{benzene \compound{benzene}}
5      }

```



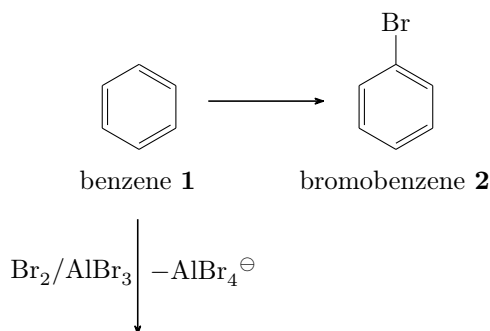
We need to give the first reactant an anchor in order to refer to it with the branch.

```

1 \begin{rxnscheme}[,,,8]{electrophilic substitution}
2 \setatomsep{1.6em}
3 \reactant[,start]{
4 \chemname{\chemfig{*6(==(-[,,,,white]\phantom{Br})-=)
   }}{benzene \compound{benzene}}
5 }
6 \branch[below=of start]{
7 \arrow[below,,,,both]{ \ce{Br2 / AlBr3} }{ $-\ce{AlBr
   4\om}$ }
8 }
9 \arrow{}{}
10 \reactant{
11 \chemname{\chemfig{*6(==(-Br)-=)}}{bromobenzene \
   compound{bromobenzene}}
12 }
13 \end{rxnscheme}

```

So the first reactant gets the anchor `start` and the branch refers to it with `below=of start`. For the reaction arrow to point below, we need to use the option `below`. Now we get:



Next we write the resonance formulæ of the Wheland intermediate. To do that we use three further commands: `\mesomeric`, `\marrow` and `\elmove`.

```

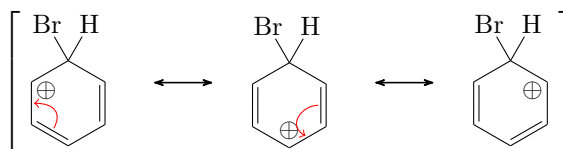
1 \mesomeric{
2 \reactant{
3 \chemfig{*6(=[@{e1}]--(-[:120]Br)(-[:60]H)
   -(-[:30,.4,,,white]\oplus)-[@{e2}]})}
4 \elmove{e1}{60:4mm}{e2}{0:4mm}
5 }
6 \marrow
7 \reactant{
8 \chemfig{*6(-(-[:90,.4,,,white]\oplus)-[@{e4}]=[@{e
   3}] -(-[:120]Br)(-[:60]H)-=)}

```

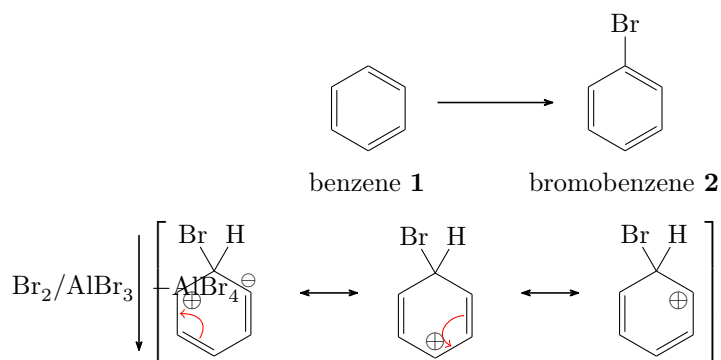
```

9      \elmove{e3}{180:4mm}{e4}{150:4mm}
10    }
11    \marrow
12    \reactant{
13      \chemfig{*6(==(-[:150,.4,,,white]\oplus)-(-[:120]Br
14        )(-[:60]H)-=)}
15    }

```



When we write the code *inside* of the branch, directly after the arrow, we get the following:



This obviously messes everything up. We can try this, though:

```

1  \begin{rxn}[,.8]
2  \setatomsep{1.6em}
3  \reactant[,start]{
4    \chemname{\chemfig{*6(==(-[:120]Br)-(-[:60]H)
5      )}}{benzene \compound{benzene}}
6  }
7  \branch[below=of start]{
8    \arrow[below,,,both]{\ce{Br2 / AlBr3}}{\$-\ce{AlBr4\
9      om}}\$}
10   \mesomeric[below]{
11     \reactant{
12       \chemfig{*6(=[@{e1}]-(-[:120]Br)-(-[:60]H)
13         -(-[:30,.4,,,white]\oplus)-[@{e2}])}
14       \elmove{e1}{60:4mm}{e2}{0:4mm}

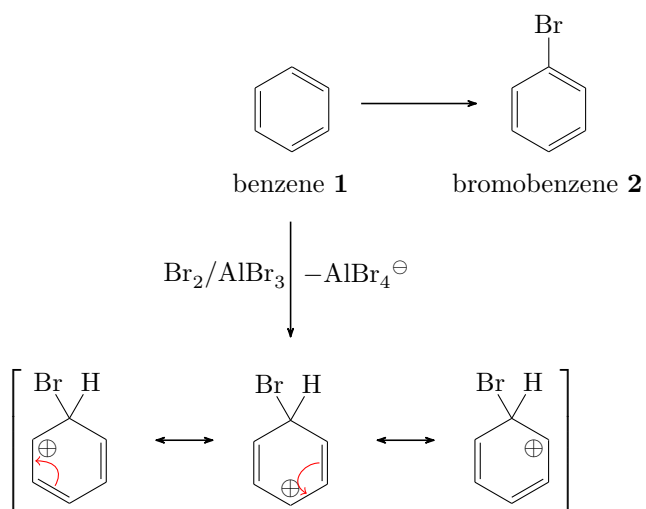
```

```

12      }
13      \marrow
14      \reactant{
15          \chemfig{*6(-(-[:90,.4,,,white]\oplus)-[@{e4}]=[@
16              {e3}]-(-[:120]Br)(-[:60]H)-=)}
17          \elmove{e3}{180:4mm}{e4}{150:4mm}
18      }
19      \marrow
20      \reactant{
21          \chemfig{*6(==(-[:150,.4,,,white]\oplus)
22              -(-[:120]Br)(-[:60]H)-=)}
23      }
24      \arrow{}{}
25      \reactant{
26          \chemname{\chemfig{*6(==(-Br)-=)}}{bromobenzene \
27              compound{bromobenzene}}
28      }
29      \end{rxn}

```

The result is better:



It isn't really what we want, though, because the intermediate is centered below the arrow. In order to be able to shift the whole thing, we place it into a branch of its own.

```

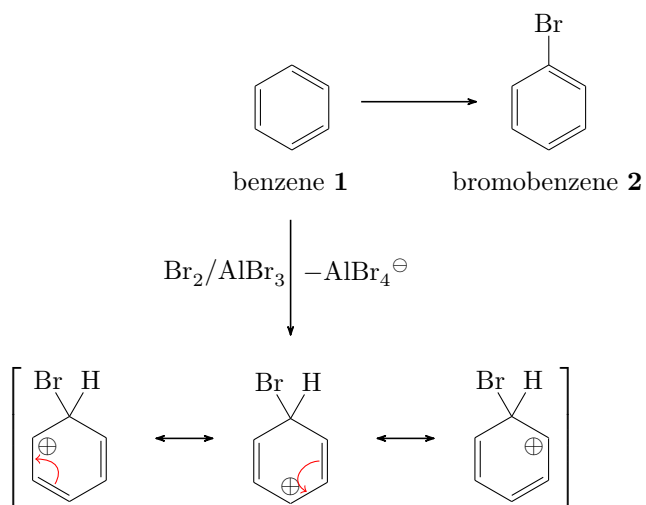
4      ...
5      \branch[below=of start]{
6          \arrow[below,,,arrow_a,both]{\ce{Br2 / AlBr3}}{\$-\ce{
7              AlBr4\om}}\$}
8      }

```

```

8   \branch[below=of arrow_a]{
9     \mesomeric{
10      \reactant{
11        \chemfig{*6(=[@{e1}]--(-[:120]Br)(-[:60]H)
12          -(-[:30,.4,,,white]\oplus)-[@{e2}])}
13        \elmove{e1}{60:4mm}{e2}{0:4mm}
14      }
15      \marrow
16      \reactant{
17        \chemfig{*6(-(-[:90,.4,,,white]\oplus)-[@{e4}]=[@{e
18          3}]-(-[:120]Br)(-[:60]H)-=)}
19        \elmove{e3}{180:4mm}{e4}{150:4mm}
20      }
21      \marrow
22      \reactant{
23        \chemfig{*6(==(-[:150,.4,,,white]\oplus)-(-[:120]
24          Br)(-[:60]H)-=)}
25      }
26    }
27  }
28  ...

```



In first sight this isn't better. But by shifting the branch with `xshift`, we get what we want:

```

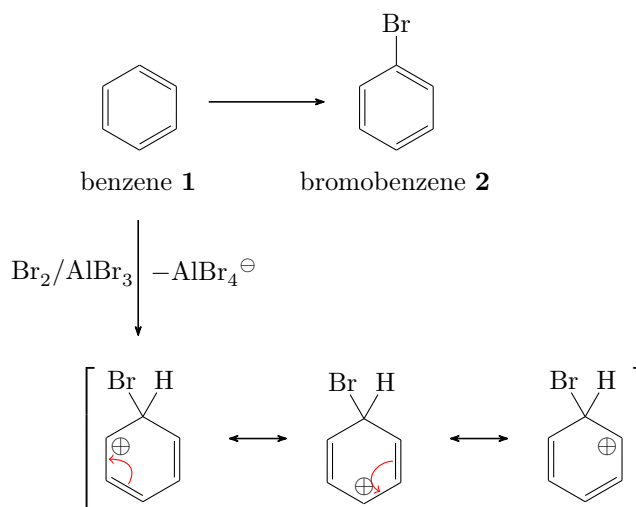
4   ...
5   \branch[below=of start]{
6     \arrow[below,,,arrow_a,both]{\ce{Br2 / AlBr3}}{\$-\ce{
7       AlBr4\om}}\$}

```

```

7   }
8   \branch[below=of arrow_a,,xshift=8.5em]{
9     \mesomeric{
10      \reactant{
11        \chemfig{*6(=[@{e1}]--(-[:120]Br)(-[:60]H)
12          -(-[: -30,.4,,,white]\oplus)-[@{e2}])}
13        \elmove{e1}{60:4mm}{e2}{0:4mm}
14      }
15      \marrow
16      \reactant{
17        \chemfig{*6(-(-[:90,.4,,,white]\oplus)-[@{e4}]=[@{e
18          3}]-(-[:120]Br)(-[:60]H)-=)}
19        \elmove{e3}{180:4mm}{e4}{150:4mm}
20      }
21      \marrow
22      \reactant{
23        \chemfig{*6(==(-[: -150,.4,,,white]\oplus)-(-[:120]
24          Br)(-[:60]H)-=)}
25      }
26    }
27  }
28  ...

```



The last arrow is also placed in its own branch, so we can shift it where we want.

```

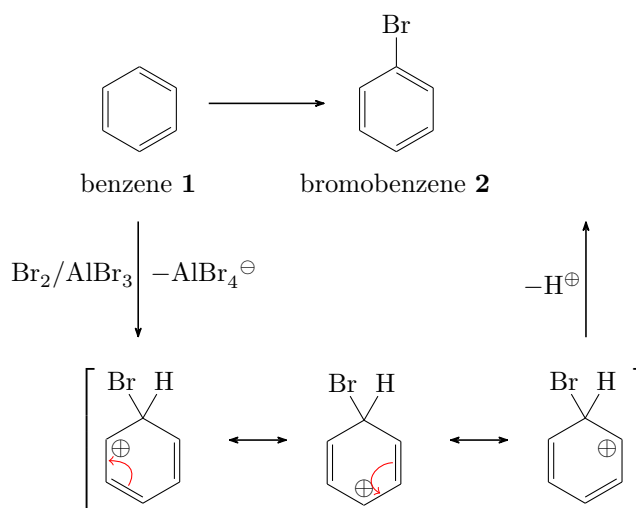
4   ...
5   \branch[below=of start]{
6     \arrow[below,,arrow_a,both]{\ce{Br2 / AlBr3}}{\$-\ce{
7       AlBr4\om}}\$}

```

```

7   }
8   \branch[below=of arrow_a,mesomerism,xshift=8.5em]{
9     \mesomeric{
10      \reactant{
11        \chemfig{*6(=[@{e1}]--(-[:120]Br)(-[:60]H)
12          -(-[:30,.4,,,white]\oplus)-[@{e2}])}
13        \elmove{e1}{60:4mm}{e2}{0:4mm}
14      }
15      \marrow
16      \reactant{
17        \chemfig{*6(-(-[:90,.4,,,white]\oplus)-[@{e4}]=[@{e
18          3}]-(-[:120]Br)(-[:60]H)-=)}
19        \elmove{e3}{180:4mm}{e4}{150:4mm}
20      }
21      \marrow
22      \reactant{
23        \chemfig{*6(-=-([:-150,.4,,,white]\oplus)-(-[:120]
24          Br)(-[:60]H)-=)}
25      }
26    }
27  }
28  ...

```



Now we're almost there: the arrow of the main reaction still is too short. The complete code:

```

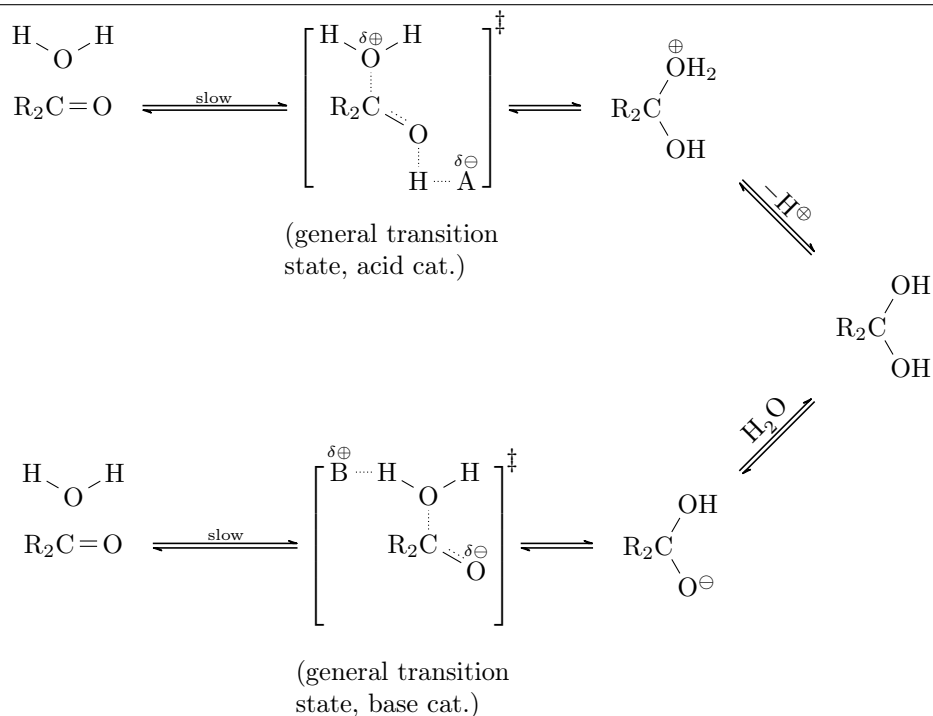
1 \begin{rxnscheme}[,,,8]{electrophilic substitution}
2 \setatomsep{1.6em}
3 \reactant[,start]{\chemname{\chemfig{*6(==(-[,,,white
   ]\phantom{Br})-=)}}{benzene \compound{benzene}}} }
4
5 \branch[below=of start]{
6   \arrow[below,,,arrow_a,both]{\ce{Br2 / AlBr3}}{\$-\ce{
   AlBr4\om}}\$}
7 }
8 \branch[below=of arrow_a,mesomerism,xshift=8.5em]{
9   \mesomeric{
10    \reactant{
11      \chemfig{*6(=[@{e1}]--(-[:120]Br)(-[:60]H)
12        -(-[: -30,.4,,,white]\oplus)-[@{e2}]})}
13      \elmove{e1}{60:4mm}{e2}{0:4mm}
14    }
15    \marrow
16    \reactant{
17      \chemfig{*6(-(-[:90,.4,,,white]\oplus)-[@{e4}]=[@{e
18        3}]-(-[:120]Br)(-[:60]H)-=)}
19      \elmove{e3}{180:4mm}{e4}{150:4mm}
20    }
21    \marrow
22    \reactant{
23      \chemfig{*6(==(-[: -150,.4,,,white]\oplus)-(-[:120]
24        Br)(-[:60]H)-=)}
25    }
26  }
27 }
28
29 \arrow[, ,2.6]{}{}
30 \reactant{ \chemname{\chemfig{*6(==(-Br)-=)}}{
   bromobenzene \compound{bromobenzene}}} }
31 \end{rxnscheme}

```


5.3 Hydratisation

A scheme with transition states.

Reaction scheme 12 Hydratisation



For this example we first declare a style for the delocalized double bonds:

```

1 \pgfdeclaredecoration{ddbond}{initial}{%
2   \state{initial}[width=2pt]{%
3     \pgfpathlineto{\pgfpoint{2pt}{0pt}}%
4     \pgfpathmoveto{\pgfpoint{1.5pt}{2pt}}%
5     \pgfpathlineto{\pgfpoint{2pt}{2pt}}%
6     \pgfpathmoveto{\pgfpoint{2pt}{0pt}}%
7   }%
8   \state{final}{%
9     \pgfpathlineto{\pgfpointdecoratedpathlast}%
10  }%
11 }%
12 \tikzset{lddbond/.style={decorate,decoration=ddbond}}%
13 \tikzset{rddbond/.style={decorate,decoration={ddbond,
    mirror}}}%

```

Now the delocalized double bond can be used via **ChemFig**'s fifth option (see the **ChemFig** manual):

Example 52

```
1 \chemfig{-[,,,,lddbond]-[,,,,
      rddbond]}
```

Further we define the two commands

```
1 \newcommand*\delm{\ensuremath{\text{\tiny$\delta\ominus$}}}
2 \newcommand*\delp{\ensuremath{\text{\tiny$\delta\oplus$}}}
```

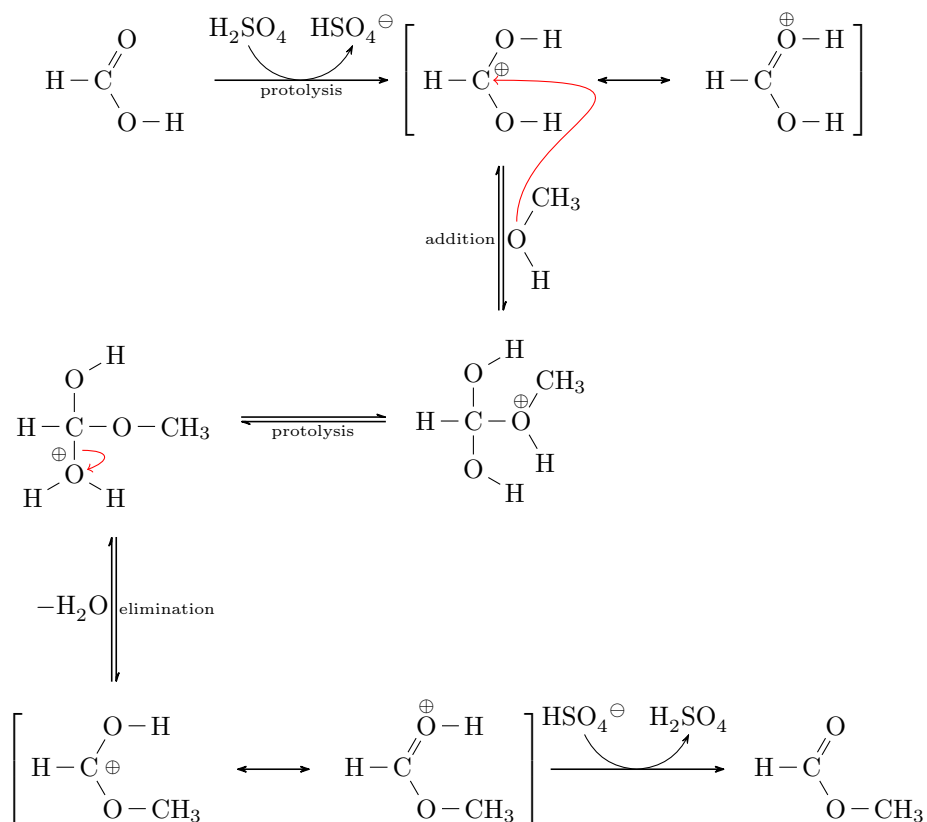
to use the partial charges without effort.

Then the whole code looks like follows:

```
1 \begin{rxnscheme}[rs:hydratisation,htp]{Hydratisation}
2 \reactant[,carbonyl_A]{\chemfig{R_2C=O}}
3 \anywhere{above=of carbonyl_A}{\chemfig{H-[: -30]O-[:30]H
  }}
4 \arrow[,<=>]{\tiny slow}{}
5 \transition[,transition_A]{\chemfig{R_2C(-[2,,2,,densely
  dotted]\chemabove{O}{\delp}(-[:150]H)-[:30]H)
  -[: -30,1.15,,,lddbond]O-[6,,,densely dotted]H-[,,,,
  densely dotted]\chemabove{A}{\delm}}}}
6 \anywhere{below=of transition_A,text width=3cm}{(general
  transition state, acid cat.)}
7 \arrow[,<=>,.5]{}{}
8 \reactant{\chemfig{R_2C(-[:60]\chemabove{O}{\scriptstyle
  \oplus}H_2)-[: -60]OH}}
9 \arrow[below right,<=>,.5]{\Hpl}{}
10 \reactant[below right]{\chemfig{R_2C(-[:60]OH)-[: -60]OH
  }}
11 \arrow[below left,<=>,.5]{\ce{H2O}}{}
12 \reactant[below left,zw]{\chemfig{R_2C(-[:60]OH)-[: -60]O
  |\om}}
13 \arrow[left,<=>,.5]{}{}
14 \transition[left,transition_B]{\chemfig{R_2C(-[2,,2,,
  densely dotted]O(-[:150]H-[4,,,densely dotted]\
  chemabove{B}{\delp})-[:30]H)-[: -30,1.15,,,lddbond]\
  chemabove{O}{\delm}-[6,,,draw=none]\phantom{H}}}}
15 \anywhere{below=of transition_B,text width=3cm}{(general
  transition state, base cat.)}
16 \arrow[left,<=>]{\tiny langsam}{}
17 \reactant[left,carbonyl_B]{\chemfig{R_2C=O}}
18 \anywhere{above=of carbonyl_B}{\chemfig{H-[: -30]O-[:30]H
  }}
```

19 \end{rxnscheme}

5.4 Esterification



```

1 \begin{rxn}
2   \reactant{\chemfig{H-C(=[:60]O)-[: -60]O-H}}
3   \arrow[, -+>, 1.2, protolysis]{\ce{H2SO4}}{\ce{HSO4\om}}
4   \anywhere{below=of protolysis, yshift=1em}{\tiny
      protolysis}
5   \mesomeric{
6     \reactant{\chemfig{H-@{a2}C(-[:60]O-H)(-[:30,.5,,,draw
          =none]{\scriptstyle\oplus})-[: -60]O-H}}
7     \marrow
8     \reactant{\chemfig{H-C(=[:60]\chemabove{O}{\scriptstyle\oplus})-H)-[: -60]O-H}}
9   }
10  \branch[on chain=going below,, xshift=-5em]{
11    \arrow[below, <=>]{\tiny addition}{\chemfig{H-[:120]@{a
      1}O-[:60]CH_3}}

```

```

12      \reactant[below]{\chemfig{H-C(-[2]O-[:30]H)(-\
      chemabove{0}{\scriptstyle\oplus})(-[:60]CH_3)-[: -60]
      H)-[6]O-[: -30]H}}
13    }
14    \elmove{a1}{90:1.5cm}{a2}{0:3cm}
15    \branch[on chain=going left,,yshift=-3.5em]{
16      \arrow[left,<=>]{}{\tiny protolysis}
17    }
18    \reactant[left]{\chemfig{H-C(-[2]O-[:30]H)(-O-CH_3)-[@{b
      1}6]@{a3}\chemabove{0}{\hspace*{-4mm}\scriptstyle\
      oplus})(-[: -150]H)-[: -30]H}}
19    \elmove{b1}{0:5mm}{a3}{20:5mm}
20    \arrow[below,<=>]{\ce{- H2O}}{\tiny elimination}
21    \mesomeric[below,,xshift=6em]{
22      \reactant{\chemfig{H-C(-[:60]O-H)(-[,.5,,,draw=none]{\
      scriptstyle\oplus})-[: -60]O-CH_3}}
23      \marrow
24      \reactant{\chemfig{H-C(=[:60]\chemabove{0}{\
      scriptstyle\oplus})-H)-[: -60]O-CH_3}}
25    }
26    \arrow[,+>,1.2]{\ce{HSO4\om}}{\ce{H2SO4}}
27    \reactant{\chemfig{H-C(=[:60]O)-[: -60]O-CH_3}}
28    \end{rxn}

```

5.5 Creating an extensive synthesis using myChemistry and ChemFig

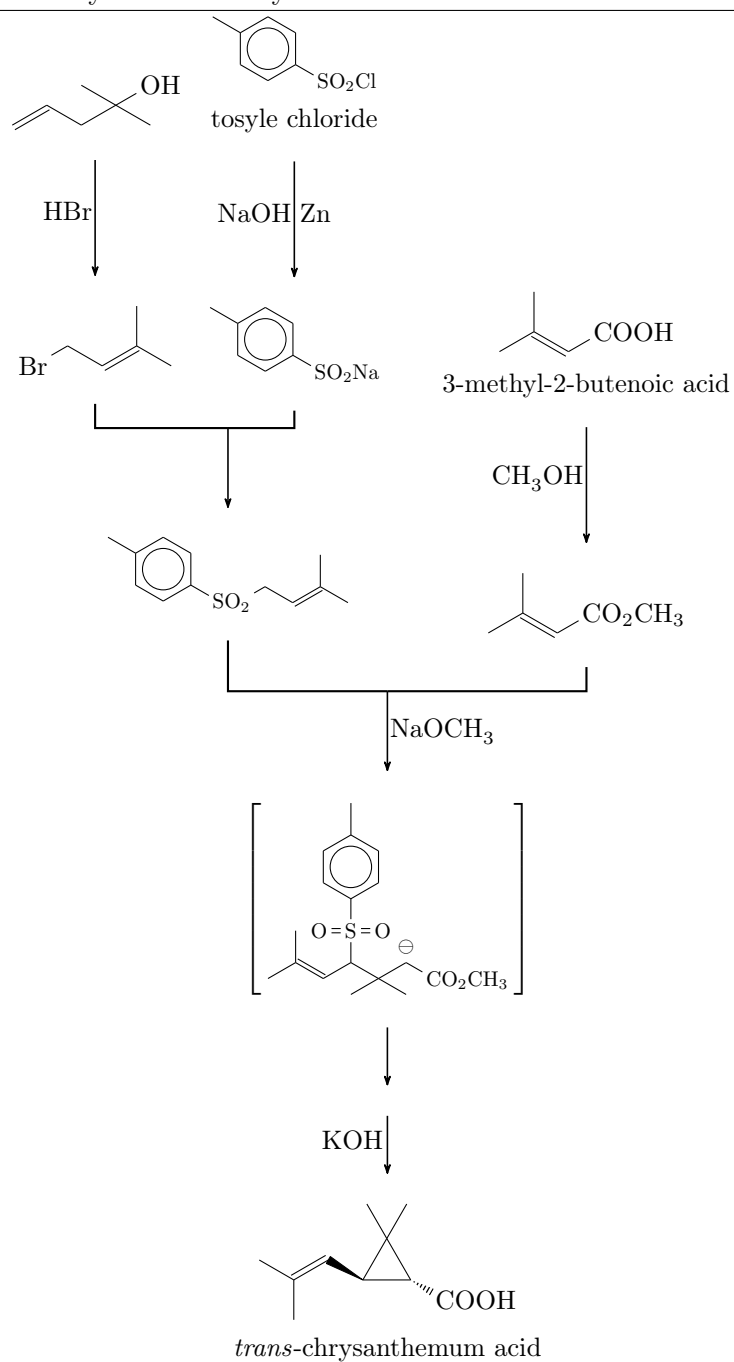
As last example we can create extensive syntheses, using the `\merge` command.

```

1    \begin{rxnscheme}[,htp,,.8]{synthesis of chrysanthemum
      acid}
2    \setatomsep{1.5em}
3    \branch[,start_left]{
4      \reactant{\chemfig{=_[::30]-[: -60]-[:60](-[: -60])
      (-[:120])-[:0]OH}}
5      \arrow[below]{\ce{HBr}}{}
6      \reactant[below]{\chemfig{Br
      -[:30]-[: -60]=[:60](-[: -60])-[:60]}}
7    }
8    \branch[right=of start_left,start_center,yshift=1em]{
9      \reactant{\chemname{\chemfig{[scale=.8]**6(--(-SO_2
      Cl)---(-)-)}}{tosyle chloride}}
10     \arrow[below]{\ce{NaOH}}{\ce{Zn}}
11     \reactant[below]{\chemfig{[scale=.8]**6(--(-SO_2Na)
      ---(-)-)}}

```

Reaction scheme 13 synthesis of chrysanthemum acid



```

12 }
13 \branch[right=of start_center,start_right,xshift=3em,
    yshift=-10em]{
14   \reactant{\chemname{\chemfig{-[:30](-[:60])
    =_[: -60] -[:60] COOH}}}{3-methyl-2-butenic acid}}
15   \arrow[below]{\ce{CH3OH}}{}
16   \reactant[below]{\chemfig{-[:30](-[:60])
    =_[: -60] -[:60] CO_2CH_3}}
17 }
18 \branch[below=of start_left,target_one,xshift=5em,yshift
    =-5em]{
19   \reactant{\chemfig{}}[scale=.8]**6(--(-S0
    _2-[:30]-[: -60]=_[:60](-[:60]) -[: -60]) ---(-) -)
    )}
20 }
21 \branch[below=of target_one,target_two,xshift=6em,yshift
    =-6em]{
22   \mesomeric{\chemfig{}}[scale=.8]{-[:30](-[:60])
    =^[: -60] -[:60](-[:60]S(=[:90]O)(=[: -90]O)
    -[:0]**6(---(-)---)) -[: -60](-[:0])(-[: -120])
    -[:60](-[:60,.5,,,white]\ominus) -[: -60] CO_2CH
    _3}}
23   \arrow[below,,.5]{}{}
24   \arrow[below,,.5]{\ce{KOH}}{}
25   \reactant[below]{\chemname{\chemfig{-[: -30](-[: -60])
    =^[:60]>[: -60](-[:90,1.2])
    -[:30,1.2](-[:120,1.2](-[: -60]) -[:0])<[: -30]
    COOH}}}{\emph{trans}-chrysanthemum acid}}
26 }
27 \merge{target_one}{start_left}{start_center}
28 \merge[\ce{NaOCH3}]{target_two}{target_one}{start_right}
29 \end{rxnscheme}

```

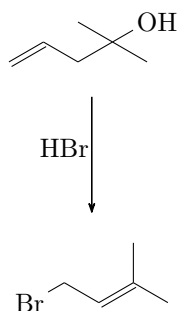
Let's go through the code, piece by piece.

```

1 \begin{rxnscheme}[,htp,,.8]{synthesis of chrysanthemum
    acid}
2   \setatomsep{1.5em}
3   \branch[,start_left]{
4     \reactant{\chemfig{=_[:30] -[: -60] -[:60](-[: -60])
    (-[:120]) -[:0] OH}}
5     \arrow[below]{\ce{HBr}}{}
6     \reactant[below]{\chemfig{Br
    -[:30] -[: -60]=_[:60](-[: -60]) -[:60]}}
7   }

```

In lines 1 and 2 we begin the environment and make sure, that the formulæ don't become too big. In lines 3 – 7 the first two reactants are written (lines 4 and 6) and connected with an arrow (line 5).

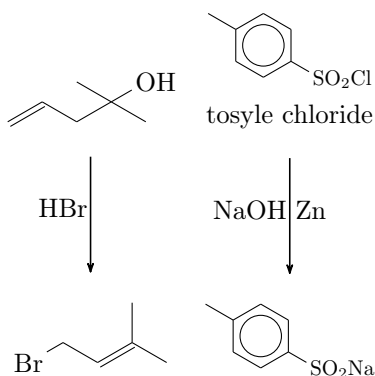


```

8   \branch[right=of start_left,start_center,yshift=1em]{
9     \reactant{\chemname{\chemfig{[scale=.8]**6(-(-SO_2
      Cl)---(-)-)}}{tosyle chloride}}
10    \arrow[below]{\ce{NaOH}}{\ce{Zn}}
11    \reactant[below]{\chemfig{[scale=.8]**6(-(-SO_2Na)
      ---(-)-)}}
12  }

```

In the following lines 8 – 12, we create the second branch of the synthesis.



```

13  \branch[right=of start_center,start_right,xshift=3em,
      yshift=-10em]{
14    \reactant{\chemname{\chemfig{-[:30](-[:60])
      =_[:60]-[:60]COOH}}}{3-methyl-2-butenic acid}}
15    \arrow[below]{\ce{CH3OH}}{}
16    \reactant[below]{\chemfig{-[:30](-[:60])
      =_[:60]-[:60]CO_2CH_3}}
17  }
18  \branch[below=of start_left,target_one,xshift=5em,yshift
      =-5em]{

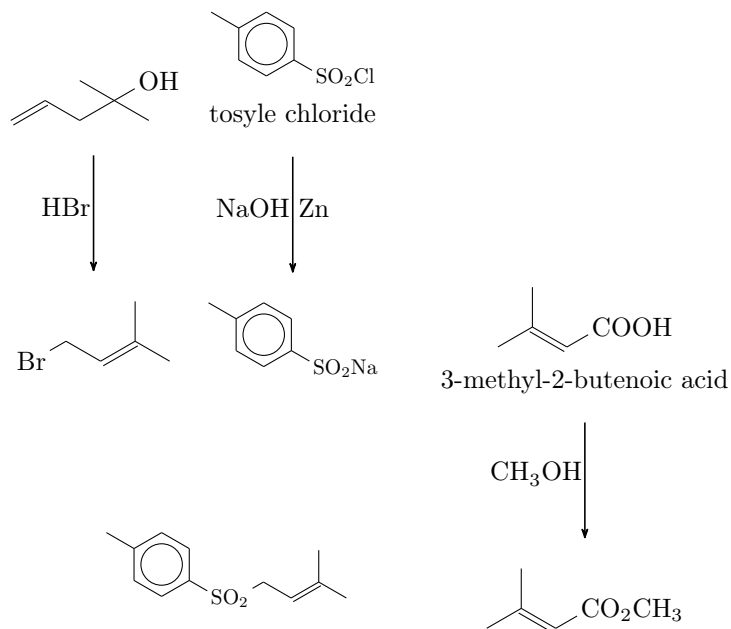
```

```

19      \reactant{\chemfig{[scale=.8]**6(--(-SO
      _2-[:30]-[::-60]=[::60](-[::60])-[::-60])---(-)-)
      }}
20    }

```

In lines 13 – 20 we create the third branch and the product of the first two branches.

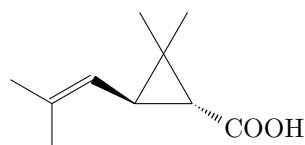
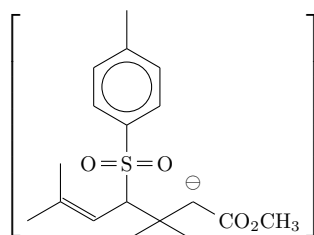
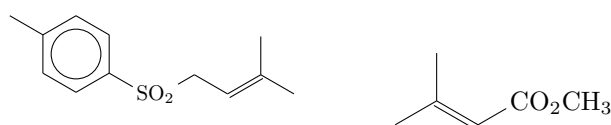


```

21    \branch[below=of target_one,target_two,xshift=6em,yshift
      =-6em]{
22      \mesomeric{\chemfig{[scale=.8]{-[::30](-[::60])
      =^[::-60]-[::60](-[::60])S(=[::90]O)(=[::-90]O)
      -[::0]**6(---(-)---)-[::-60](-[::0])(-[::-120])
      -[::60](-[::60,.5,,,white)\ominus)-[::-60]CO_2CH_3}}}
23      \arrow[below,,,5]{\ce{KOH}}
24      \arrow[below,,,5]{\ce{KOH}}
25      \reactant[below]{\chemname{\chemfig{-[::-30](-[::-60])
      =^[::60]>[::-60](-[::90,1.2])
      -[::30,1.2](-[::120,1.2](-[::-60])-[::0])<[::-30]
      COOH)}}{\emph{trans}-chrysanthemum acid}}
26    }

```

In lines 21 – 26 we create the last branch.

*trans*-chrysanthemum acid

Finally, the different branches are merged, the second merging arrow gets a label and the environment is ended.

```

27   \merge{target_one}{start_left}{start_center}
28   \merge[\ce{NaOCH3}]{target_two}{target_one}{start_right}
29   \end{rxnscheme}

```

6 Epilogue

myChemistry is still very new. This means there are probably a number of bugs I haven't discovered yet. There also might be missing one or two features, that would be useful. Since I only can test and work on myChemistry in my spare time, I'd be very glad about *every* kind of feedback. If you like myChemistry, why don't you help me improve it by telling me your experiences?

I tried using real chemical reactions but I didn't make sure, that they all make sense chemically. So you shouldn't trust the examples in respect to chemistry but rather take a look into a real chemistry teaching book.

I apologize for any bad or wrong English. I hope you understood the documentation anyway.

Have fun with **myChemistry**!

Clemens Niederberger, Berlin, April 15th 2011

7 Thanks

I owe thanks for bug reports and suggestions to:

F. Chervet, Ferghun, V. Garibal.

Index

A

alignment 14 - 17
anywhere 19, 58
arrow 8, 20 ff., 48 f.
 alignment 22
 appearance 22
 both 22
 options 20 f.
 anchor 20
 both 20
 direction 20
 length 20
 type 20

B

Befehle

 anywhere 19
 setarrowline 42
branch 9, 22 - 25, 46, 50 ff.
 alignment 24 f.

C

chemand 25
chemcompounds 5 f.
chemexec 5 f.
ChemFig 5 f.
commands
 arrow 20
 branch 22
 chemand 25
 dummy 25
 elmove 26
 makeinvisible 26
 makevisible 26
 marrow 27
 mCsetup 27
 merge 28
 mesomeric 32
 reactant 34

rxn *see* rxn
rxnscheme *see* rxnscheme
setarrowlength 41
setatomsizes 42
setbondlength 42
setbondshape 42
setmergelength 43
setrcndist 43
setrxnalign 43
setschemealign 43
setschemename 44
transition 44

D

default values 11 ff.
dummy 25 f.

E

elmove 26, 50
environment
 rxn *see* rxn
 rxnscheme *see* rxnscheme

M

makeinvisible 26
makevisible 26 f.
marrow 27, 45, 50
mCsetup 27 ff.
 align 27
 arrowlength 27
 atomsizes 27
 bondlength 27
 mergelength 27
 rcndist 27
 reset 27
 rxnalign 27
 schemealign 27
merge 28 - 32, 42, 60, 65
mesomeric 32 ff., 45, 50
mhchem 5 f.

O

options 13
 chemstyle 13
 color 13
 english 13
 nochemexec 13
 nocolor 13
 nocompounds 13
 nomhchem 13
 nopackages 13
 placement 13
 shade 13

R

reactant 8, 34f., 48f.
 tikz 14
 requirements 5
 calc 5
 ChemFig 5
 float 5
 ifthen 5
 pgf 5
 TikZ 5
 xkeyval 5
 rxn 8, 35 ff.
 options 35 ff.
 alignment 35
 scale 35
 rxnscheme 11, 37 - 41, 48
 customize 39 ff.
 counter 41
 list 41
 name 40
 placement 40
 style 39
 options 37 ff.
 alignment 38
 label 37
 name 38
 placement 37
 scale 38

S

setarrowlength 13, 41 f.
 setarrowline 42
 setatomsizes 12, 42
 setbondlength 12, 42
 setbondshape 12, 42 f.
 setmergelength 29, 43
 setrcndist 43
 setrxnalign 43 f.
 setschemealign 43 f.
 setschemename 44

T

TikZ 49, 57
 transition 44, 57

X

xshift 17, 24, 31, 53

Y

yshift 17, 24, 31, 46