

# The adjcalc Package

Part of the adjustbox bundle

Martin Scharrer  
[martin.scharrer@web.de](mailto:martin.scharrer@web.de)

Version v1.1 – 2012/05/16

CTAN: <https://www.ctan.org/pkg/adjustbox>

## Abstract

This package provides macros to assign or to add algebraic expressions to  $\text{\TeX}$  lengths and counters. It either uses  $\epsilon$ - $\text{\TeX}$ , the `calc` package or the `pgfmath` package internally to achieve this. This package is part of the `adjustbox` bundle and was originally part of the `adjustbox` package before it was extracted to a dedicated package.

## 1 Introduction

$\text{\TeX}$  lengths and counters can be set using `\setlength` and `\setcounter`, and values can be added to these using `\addtolength` and `\addtocounter`, respectively. By default these macros await a single length or integer value to be assigned or added. However, often it is beneficial to use a short algebraic expression, like a sum of two values, instead. The `calc` package was created for this and redefines the above macros to parse the given expression. Another possibility is to use  $\epsilon$ - $\text{\TeX}$ 's `\dimexpr` or `\glueexpr` which also allows for several algebraic operations and is more efficient than the higher-level `calc` package. Finally PGF/TikZ provides a large math engine with the `pgfmath` package. It allows even for complicated functions like `log` or `sin`. However, `pgfmath` is rather large, especially because with PGF v2.10 it cannot be loaded on its own due to a bug, which forces the whole `pgf` package to be loaded.

Package authors which like to write macros which allow algebraic expressions for its length arguments now face the dilemma which math interface they should use. Either they exclude users which do not have an  $\epsilon$ - $\text{\TeX}$  distribution (even if they are very rare) or require to load a large package. This was the case with the `adjustbox` package and therefore own wrapper macros were used instead which will use one of the different math engine mentioned above. Because this functionality was deemed useful for other packages of the same and other authors it was extracted into the dedicated package `adjcalc`. The idea is that the user can choose the used math engine using package options.

## 2 Package Options

The following options define the way length values are processed by the provided macros. The first group can be either used as package options and as local setting

using `\adjcalcset` or as keys for `\adjustbox` or the `adjustbox` environment (but not for `\includegraphics` even if the `export` option of `adjustbox` was used). The only difference between using them as package options or keys is that as package option they also load required packages. Therefore all keys used in the document should be used as package options first or the required packages must be loaded manually.

**etex** Uses the  $\epsilon$ -TeX primitive `\glueexpr` to parse length values. This allows for additions, subtractions as well as multiplications and division by a numeric factor. See the official `etex_man` document for more details. This setting is the default if  $\epsilon$ -TeX is detected (which should be the case with all modern  $\LaTeX$  distributions).

**calc** Uses the `calc` package to parse length values. It supports all operations mentioned for `etex` and also some other operation like `\widthof{<text>}`. See the `calc` package manual for more details. This is the default setting if  $\epsilon$ -TeX is not detected.

**pgfmath** Uses the `pgfmath` package of the `pgf` bundle to parse length values. It supports all basic numeric operations and also advanced mathematical functions. See the `pgf` manual for more details. Because the `pgfmath` package can't be loaded independently in the current version (v2.10) the whole `pgf` package will be loaded.

**overwrite** This option will overwrite the standard macros with the macros of this package. This e.g. sets `\setlength` to be identical to `\adjsetlength`. Any package option used after `overwrite` will still take affect before the redefinition, i.e. the order in which `overwrite` is used is not meaningful.

**defaultunit=<unit>** This sets the default unit used for macros and keys which allow for unit-less values, e.g. the `trim` key of `\adjustbox` or the `\trimbox` macro of the `trimclip` package. The standard default unit is the same as for `\includegraphics`: "bp" (big points, PostScript points). However, for  $\LaTeX$  material TeX normal unit "pt" (TeX points) are better suited and will avoid rounding errors which otherwise get introduced by the internal conversion. The default unit is only used if the particular value is only a single number without unit, but not if any mathematical operations are used. If the special value `none` is used no default unit is applied and the internal check if the value is a single number is by-passed. This gives a small speed bonus and can be used to avoid potential issues with complex values. At this moment this setting will disable the default unit feature for the rest of the current group (i.e. all further `\adjustbox` keys or globally if used as a package option) and further usages of this option will have no affect. This might change in future versions of this package.

The following option can only be used as package option:

**none** Disables the features of this package and makes all macros use the normal  $\LaTeX$  equivalents, e.g. `\adjsetlength` will expand to `\setlength`, etc. This option also disables any previous set option as well as any future use of `overwrite`.

### 3 Macros

```
\adjcalcset{<key=value, ...>}
```

This configuration macro allows to change the above mentioned options during the document. Any changes are only locally to the current group/environment. Please see the descriptions of the options for any limitations.

```
\adjsetlength{<\lengthmacro>}{<length expression>}  
\adjaddtolength{<\lengthmacro>}{<length expression>}  
\adjsetcounter{<counter name>}{<integer expression>}  
\adjaddtocounter{<counter name>}{<integer expression>}
```

These macros can be used to set and add to  $\LaTeX$  length registers and counters like with the normal macros `\setlength`, `\addtolength`, `\setcounter` and `\addtocounter`. However, these macros allow the use of algebraic expressions, like sums and multiplications, while the standard macros only await a single value.

```
\adjsetlengthdefault{<\lengthmacro>}{<length expression with or without unit>}
```

This macro set the length macro but does also support length values without an explicit unit. If no unit is used the default unit set by the `defaultunit` option is used. The initial default unit is 'bp' (big points, i.e. PostScript/PDF points, 72bp=1inch).