globalvals – Resuable variables for use in large projects

Charles Baynham

2019/02/06 (v 1.1)

This file is maintained by Charles Baynham.
Bug reports can be opened at
`https://github.com/charlesbaynham/globalvals`.

The globalvals package allows the user to declare a variable which can then be used anywhere else in a document, including before it was declared. This can be useful in large projects, where values can be entered once and automatically updated throughout the document, without having to maintain a separate file full of definitions.

This is done by putting the definitions into the .aux files, therefore requiring two runs to get it right.

It implements two commands: `\defVal{`⟨*key*⟩`}{`⟨*value*⟩`}` and `\useVal{<key>}`. `\defVal` sets up a global variable and `\useVal` recalls it.

Using `\defVal` twice with the same ⟨*key*⟩ will result in an error. Using `\useVal` for an undefined value will output the text "??".

1 Defining a value

`\defVal` Defining a macro can be done using the command

`\defVal{`⟨*key*⟩`}{`⟨*value*⟩`}`

For example, you might call

```
\defVal{software_version}{v1.65}
```

⟨*value*⟩s will be expanded, so you can also embed macros within your variables, e.g.:

```
\defVal{fractional_stability}{\SI{1E-16}{\per\sqrt\second}}
```

2 Using a variable

`\useVal` To use a defined variable, use the command

`\useVar{`⟨*key*⟩`}`

For instance, the values saved in section 1 could be recalled using

```
\useVal{software_version}
```

and

```
\useVal{fractional_stability}
```

to give "v1.65" and "$1 \times 10^{-16}/\sqrt{\mathrm{s}}$".

Importantly, values may be used **before they are defined**. This is handy if you e.g. would like to refer to a quantity in your abstract but it's most sensible defined in a later chapter. Like so:

```
The clock's fractional accuracy is estimated as
\useVal{an_important_quantity}.
...
\defVal{an_important_quantity}{\num{1E-18}}
```

to give

> The clock's fractional accuracy is estimated as 1×10^{-18}.