

The `dataref` package

Christian Dietrich 2013-2014

`stettberger@dokucode.de`

<https://github.com/stettberger/dataref>

2014/10/09 v0.3

1 Introduction

Writing scientific texts is a craft. It is the craft of communicating your results to your colleagues and to the curious world public. Often your conclusions are based upon facts and numbers that you gathered during your research for the specific topic. You might have done many experiments and produced lot of data. The craft of writing is to guide your reader through a narrative that is based upon that data. But there may be many versions of that data. Perhaps you found a problem in your experiment, while already writing, that forces you back into the laboratory. After a while, the moon has done its circle many times, you return from that dark place and your methodology has improved as significantly as your data has. But now you have to rewrite that parts of the data that reference the old data points.

The `dataref` is here to help you with managing your data points. It provides you with macro style keys that represent symbolic names for your data points. You can reference those symbolic names with `\dref`, use them in calculations to have always up-to-date percentage values, define projections between sets of data points and document them. `dataref` also introduces the notion of assertions (`\drefassert`) for your results to ensure that your prosa text references fit the underlying data.

2 Usage

The `dataref` package heavily uses `pgfkeys` and `pgfmath` to perform storage and operations upon data points. See `texdoc pgfmanual` for further information about those topics.

`\drefset{<name>}{<value>}`

The `\drefset` command is used to define the symbolic data points. The first argument is the symbolic name, the second argument is the value. The value can be a number, but it can also be arbitrary text. The key may contain virtually all characters, including spaces and slashes. It is good practice to use a hierarchy to structure you data point names.

```

\drefset{/control group/mice race}{Black Six}
\drefset{/control group/mice count}{32}
\drefset{/control group/dead after 24h}{3}
\drefset{/control group/dead after 48h}{7}
\drefset{/control group/recovered}{6}

\drefset{/med A/mice race}{Black Six}
\drefset{/med A/mice count}{32}
\drefset{/med A/dead after 24h}{6}
\drefset{/med A/dead after 48h}{1}
\drefset{/med A/recovered}{9}

```

The code snippet, which is best stored in an external file, and which might be auto-generated, is best read with `\input`. It defines 10 symbolic names, that are partitioned into two "directories" (`control group` and `medicament A`).

```

\dref*{<name>}
\dref[<format>]{<name>}

```

This macro is used to reference a single symbolic data point. The value stored in that datapoint is inserted into the text. `\dref` additionally marks the data point as used. It will then appear in the `dref` usage report. For undefined keys the default behaviour is to abort the compilation. But the package option `ignoremissing` just outputs a warning. All referenced/missing/found datapoints are noted in the aux file.

Macro	Expansion
<code>\dref*{/control group/mice race}</code>	Black Six
<code>\dref*{/control group/mice count}</code>	32
<code>\dref[sci,precision=2,zerosfill=true]{/med A/recovered}</code>	$9.00 \cdot 10^0$

`\dref` additionally takes an optional argument. This argument is interpreted as `/pgf/number format/` argument. See the `pgf/tikz` manual for more information. Only in the unstarred version this macro parses the value as a number. Be aware that `\dref` is **not** expandable.

```

\drefvalueof{<name>}

```

Since `\dref` is not expandable, this macro can be used to get the bare value of a symbolic data point. But use it with caution, since it bypasses all internal book keeping.

```

\drefvalueof{/med A/mice race}

```

```

\drefref{<name>}

```

This is complement of `\drefvalueof`, it does *only* the book keeping for a key (marking it as referenced etc.) So it might be used to compensate the usage of its bad sibling.

```

\drefref{/med A/mice race}

```

```

[ignoremissing]
[defaultvalue=1.0]

```

These two package options influence the behaviour regarding unknown keys. With `ignoremissing` each missing symbolic datapoint is replaced by the default-value. This behaviour might be useful when you use the `.aux` file, where the

unknown keys are noted to extract data points from a third source (e.g. database, wikidata, etc). In the future a secondary tool will be provided to resolve those references.

`\drefsethelp{<pattern>}{<text>}`
`\drefhelp{<name>}`

`dateref` comes with a simple method for defining documentation for data points. This help can for example be used to communicate what is the concrete semantics of the data point. This is of special interest when writer and data gatherer are not the same person. `\drefsethelp` takes two arguments: first a regular expression that matches the symbolic data point, second the help text.

`\drefsethelp{./mice race}{The mice race used for experiments heavily influences the outcome of the results}`

The helptext for a key is obtained by using the `\drefhelp` macro. It checks all defined helps (in linear order, first defined, first matched), and prints the first matching help text.

`\drefhelp{/med A/mice race}`

`\drefcalc[<format>]{<expr>}`
`data("<key>")`
`d(<key>)`

The `\drefcalc` command is the core function of calculating with data points. It is based on the `pgfmath` engine. It uses the required argument as a mathematical expression, but has additional features, that can be used.

`\drefcalc{(4+7)/12 * 100} ⇒ 91.67`

It adds support for the `data` function within `pgfmath`, which references symbolic data points. The keyname has to be in double quotes to indicate a string, but you can easily define an appropriate macro that abstracts from `data("")`. As a quote-free alternative to the `data` command, `\drefcalc` provides also `d(<key>)`.

`\drefcalc{data("/med A/mice count") * 100} ⇒ 3,200` `\drefcalc{d(/med A/mice count) * 100} ⇒ 3,200`

The optional argument lets you give a number format, which is used for printing the result number (`/pgf/number format`).

`\drefcalc[precision=5,fixed]{1/3} ⇒ 0.33333`

`\drefcalc` works as well in a `/pgf/fpu` environment or a normal one. The FPU feature of `pgfmath` is used to handle large numbers, which may occur often when handling experiment data points.

`\dreflet{A=123456789, B=987654321, a=12, b=98}`

Macro	Inserted Text	<code>\drefresult</code>
<code>\drefcalc[/pgf/fpu]{A/B}</code>	0.12	0.1241
<code>\drefcalc{a/b}</code>	0.12	0.12244
<code>\drefcalc*[/pgf/fpu]{A/B}</code>		0.1241
<code>\drefcalc*{a/b}</code>		0.12244

`\drefcalc*`
`\drefresult`

`\drefformat{<number>}`

```

\drefcalc*{1/3} ABC: \drefresult ⇒ ABC: 0.33333
\drefformat[fixed,precision=1]{\drefresult}⇒ 0.3
\drefformat[sci]{100000} ⇒ 1 · 105

```

```

/dref/let={\lets}
\dreflet{\lets}

```

Since symbolic key names can get long, `dateref` has the possibility to define variables for use within mathematical expression from other expressions. These "let"-bindings can either be defined locally for a `\drefcalc` commando with a `pgf` key or globally with `\dreflet`.

```

\drefcalc[/dref/let{A=12*20,B=\cg{recovered}}]{A/B} ⇒ 40
\drefcalc[/dref/let={X=100}]{30/X} ⇒ 0.3

```

The bindings for `\drefcalc` are only local to that macro call. Defining a binding for the current group can be done with `\dreflet`.

```

\newcommand{\cg}[1]{data("/control group/#1")}
\dreflet{percent=data("/med A/mice count")/100}

```

The result clearly shows that a lorem ipsum kills
`\drefcalc{\cg{dead after 24h}/percent}` percent within 24 and
`\drefcalc{\cg{dead after 28h}/percent}` percent within 48 hours.
The result clearly shows that a lorem ipsum kills 9.38 percent within
24 and 21.88 percent within 48 hours.

```

\drefrel*[\<opts>]{\<key>}
\drefrel[\<opts>]{\<key>}

```

The `\drefrel` macro is used to calculate relations between a base value and a concrete key. A prominent example of such a relation is the percent relation. `\drefrel` allows you to write down intentionally what relation you want to express without thinking about a concrete formula. The starred version of this macro does not print anything, but sets only `\drefresult`.

```

\drefrel[base=/med A/mice count,factor]{/med A/recovered}
⇒ 28.13

```

The type of relation can be manipulated with various keys. Almost always the given argument `key` will be set in relation to a base value. The type of relation can be given as well as post-processing steps.

Like `\drefcalc`, `\drefrel` sets the `\drefresult` macro accordingly.

```

/dref/base
/dref/base plain
/dref/value plain

```

This specifies the key that will be used as a base. Without the **base plain** option, the value will be interpreted as a symbolic datapoint. With the option, base contains the plain value. When **value plain** is given, the mandatory argument is interpreted as a number and not as a symbolic name.

```

\drefrel[factor,base=50,base plain]{/med A/mice count} ⇒ 0.64
\drefrel[factor,base=50,base plain,value plain]{45} ⇒ 0.9

```

`/dref/factor`

Is a base relation type, which cannot be mixed with other relation types. It simply divides the given value by the base value.

$$\backslash\text{drefresult} = \frac{\text{value}}{\text{base}}$$

`/dref/increase`

`/dref/overhead`

Is a base relation type. It calculates the overhead factor a value show toward the base value. `increase` and `overhead` are synonyms.

$$\backslash\text{drefresult} = \frac{\text{value} - \text{base}}{\text{base}}$$

$$\backslash\text{drefrel}[\text{overhead}, \text{base}=50, \text{base plain}, \text{value plain}]{45} \Rightarrow -0.1$$

`/dref/delta`

Is a base relation type. It calculates the difference between value and base.

$$\backslash\text{drefresult} = \text{value} - \text{base}$$

$$\backslash\text{drefrel}[\text{delta}, \text{base}=50, \text{base plain}, \text{value plain}]{45} \Rightarrow -5$$

`/dref/scale`

`/dref/product`

Is a base-relation type. It calculates the product of value and base.

$$\backslash\text{drefresult} = \text{value} \cdot \text{base}$$

$$\backslash\text{drefrel}[\text{scale}, \text{base}=50, \text{base plain}, \text{value plain}]{45} \Rightarrow 45$$

`/dref/percent`

Is a post-processing type. It calculates the percent value from a fraction.

$$\backslash\text{drefresult} = \backslash\text{drefresult} \cdot 100.0$$

$$\backslash\text{drefrel}[\text{factor}, \text{percent}, \text{base}=\text{/med A/mice count}]{\text{/med A/recovered}} \Rightarrow 28.13$$

`/dref/abs`

Is a post-processing type. It takes the absolute value.

$$\backslash\text{drefresult} = | \backslash\text{drefresult} |$$

$$\backslash\text{drefrel}[\text{overhead}, \text{abs}, \text{base}=50, \text{base plain}, \text{value plain}]{45} \Rightarrow 0.1$$

`/dref/negate`

Is a post-processing type. It negates the value.

$$\backslash\text{drefresult} = \backslash\text{drefresult} \cdot -1.0$$

$$\backslash\text{drefrel}[\text{factor}, \text{negate}, \text{base}=\text{/med A/mice count}]{\text{/med A/recovered}} \Rightarrow -0.28$$

`/dref/divide`

Is a post-processing type. Divides the result by a constant factor. The argument must be a plain number.

```
\drefresult = \drefresult · {divide}
```

```
\drefrel[value plain,divide=1e6]{1453342654} ⇒ 1,453.34
```

```
\drefprojection{<from>}{<to>}{<projection>}
```

Sometimes one or multiple sets of data have to be projected/mixed into a new set of data that is fully dependent on those values. This is achieved with `\drefprojection`. It projects one data set (subdirectory) into another one. Within the projection three different operations are possible: `\id`, `\rename` and `\calc`.

identity function renaming of points 10

```
\drefprojection{/control group}{/projection}{
  \id{mice race} % identity function
  \rename{mice count}{count} % renaming of points
  \calc{data("/dead after 24h")+data("/dead after 48h")}{died}
}
```

```
\dref*/{/projection/died} ⇒ 10
```

```
\dref*/{/projection/mice race} ⇒ Black Six
```

```
\dref*/{/projection/count} ⇒ 32
```

```
\drefrow{<list>}{<macro>}
```

```
\drefrow*
```

Often different columns in a table have to be obtained from your data points. Often those rows and columns are similar. Generating parts of tables within \LaTeX is very tricky, so `dateref` provides you with `\drefrow`. This macro iterates over a comma-separated list of values and fills out a macro which is interpreted as a symbolic data point. The entries are separated with `&` and printed. In the starred variant the resulting text is not interpreted as symbolic name, but as a macro. The symbolic name is expanded with `\drefvalueof`.

The second argument is the macro, and can have two macro replacements. The first replacement `#1` is the value of the list item, the second `#2` is the index in the list.

```
\begin{tabular}{lccc}
Group & <24h & <48h & recovered \\
Control Group & \drefrow{dead after 24h,dead after 48h,recovered}%
& {/control group/#1} \\
Medicament A & \drefrow{dead after 24h,dead after 48h,recovered}%
& {/med A/#1} \\
Starred Variant & \drefrow*{B,C,D}{\#1=#1,\#2=#2} \\
\end{tabular}
```

Group	< 24h	<48h	recovered
Control Group	3	7	6
Medicament A	6	1	9
Starred Variant	#1=B,#2=1	#1=C,#2=2	#1=D,#2=3

`\drefassert{expr}`
`[noassert]`

Sometimes the underlying data changes while you are writing. But what if your prose text relies on certain characteristics of the data. `\drefassert` uses a `pgfmath` expression that evaluates to `true` or `false`. When the assertion holds (`true`) nothing happens, only a terminal message is printed. When it does not hold (`false`) the compilation is aborted.

```
\drefassert{data("/control group/mice count") > 30}
Of the more than thirty infected mice...
```

The `noassert` package options disables the latex abortion. In that case only a warning message is printed on the terminal.

`[annotate=none]`
`[annotate=footnote]`
`[annotate=pdfcomment]`
`\drefannotate{style}`

While writing a document it is desirable to know, what key is used, while writing the text and generating the document. Therefore `dateref` provides the possibility to annotate values. The default package option `none` disables this kind of annotation. The `pdfcomment` option uses pdf annotations. Be aware that those annotations work properly only on a few selected PDF readers¹. `\drefannotate` sets the annoation style for the current group.

```
\drefannotate{none}
Black Six, 32, 33.33

\drefannotate{footnote}
Black Six2, 323, 33.334

\drefannotate{pdfcomment}
Black Six, 32, 33.33
```

`\drefusagereport`
`[usagereport]`
`[refall]`

With the `usagereport` package option enabled, `\drefusagereport` generates a usagereport of all referenced keys. The usage report groups the keys by the help texts. If the `refall` package option is given, all keys are marked as referenced.

Datagraphy

	Page	Value
<code>/control group/mice race</code>	2, 6, 7	Black Six
<code>/projection/mice race</code>	6	Black Six
The mice race used for experiments heavily influences the outcome of the results		
	Page	Value
<code>/med A/recovered</code>	2, 4, 5, 6	9

¹In doubt use Acrobat

²`\dref*{/control group/mice race}`

³`\dref{/control group/mice count}`

⁴`\drefcalc{100/3}`

	Page	Value
/control group/recovered	4, 6	6
/control group/dead after 24h	4, 6	3
/control group/dead after 48h	4, 6	7
/med A/dead after 24h	6	6
/med A/dead after 48h	6	1
Of all infected mice, a certain number died within a specified period of time. A certain recovered from the infection. Each mouse is in exactly one category.		
Keys without Description	Page	Value
/control group/mice count	2, 6, 7	32
/med A/mice count	3, 4, 5	32
/.DUMMY	5	1
/projection/died	6	10
/projection/count	6	32

3 Implementation

Guard against reading twice

```
1 \ifx\drefloaded\undefined
2   \let\drefloaded=\relax
3 \else
4   \expandafter\endinput
5 \fi
6 \ifx\PackageError\undefined
7   \def\dref@error#1{\immediate\write-1{Package dref: Error! #1.}}%
8 \else
9   \def\dref@error#1{\PackageError{dref}{#1}{}}%
10 \fi
11 % \end{macrocode}
12 %
13 % \begin{macrocode}
14 \RequirePackage{pgf}
15 \RequirePackage{kvoptions}
16 \usepgflibrary{fpu}
17 \usepackage{etoolbox}
18 \let\origforlistloop\forlistloop
19 \usepackage{etextools}
20 \let\forlistloop\origforlistloop
21 \RequirePackage{xcolor}
22
23 \SetupKeyvalOptions{
24   family=dref,
25   prefix=dref@
26 }
27 \DeclareStringOption[/data]{datapath}
28 \DeclareStringOption[1]{defaultvalue}
29 \DeclareStringOption[none]{annotate}
30 \DeclareBoolOption{usagereport}
31 \DeclareBoolOption{refall}
32 \DeclareBoolOption{ignoremissing}
33 \DeclareBoolOption{noassert}
34 \ProcessKeyvalOptions*
\
\dref@set
35 \def\dref@set#1#2{%
36   \pgfkeys@temptoks{#2}%
37   \expandafter\xdef\csname
38     pgfk@\dref@datapath#1\endcsname{\the\pgfkeys@temptoks}%
39   \ifdref@refall%
40     \expandafter\dref@found\expandafter{\dref@datapath#1}{0}
41     \expandafter\dref@referenced\expandafter{\dref@datapath#1}{0}%
42   \fi%
43 }
```

```

\drefset
44 \def\drefset#1#2{\dref@set{#1}{#2}}

\dref@expandable
45 \def\dref@expandable#1{%
46   \pgfkeysifdefined{\dref@datapath\csuse{dref@prefix}#1}{%
47     \pgfkeysvalueof{\dref@datapath\csuse{dref@prefix}#1}%
48   }{%
49     \ifdref@ignoremissing%
50       \dref@defaultvalue%
51     \else%
52       \typeout{Dref error: undefined key ‘#1’}\QUIT%
53     \fi%
54   }%
55 }

\dref@unexpandable
56 \def\dref@unexpandable#1{%
57   \def\drefcurrentkey{\dref@datapath\csuse{dref@prefix}#1}%
58   \pgfkeysifdefined{\drefcurrentkey}{%
59     \edef\dref@thepage{\arabic{page}}%
60     \immediate\write\@auxout{\noexpand\dref@found{\drefcurrentkey}{\dref@thepage}}%
61   }{%
62     \immediate\write\@auxout{\noexpand\dref@notfound{\drefcurrentkey}{\dref@thepage}}%
63     \ifdref@ignoremissing%
64       \typeout{Dref warning: undefined key ‘\drefcurrentkey’}%
65       \dref@mkanotate{UNDEFINED: \drefcurrentkey}%
66     \else%
67       \dref@error{Dref error: undefined key ‘\drefcurrentkey’}%
68     \fi%
69   }%
70   \immediate\write\@auxout{\noexpand\dref@referenced{\drefcurrentkey}{\dref@thepage}}%
71 }

\drefifdefined
72 \newcommand{\drefifdefined}[3]{
73   \def\drefcurrentkey{\dref@datapath\csuse{dref@prefix}#1}%
74   \pgfkeysifdefined{\drefcurrentkey}{#2}{#3}%
75 }

\dref
76 \def\dref{\@ifstar\@@dref\@dref}
77 \newcommand{\@dref}[2] []{% Unstarred
78   \edef\dref@argument{#2}%
79   \expandafter\dref@unexpandable\expandafter{\dref@argument}%
80   \pgfmathparse{\dref@expandable{#2}}%
81   \dref@format[#1]{\pgfmathresult}%
82   \dref@mkanotate{\textbackslash dref\{#2\}}%
83 }

```

```

84 \newcommand{\@@dref}[2] [] {% Starred
85   \edef\dref@argument{#2}%
86   \expandafter\dref@unexpandable\expandafter{\dref@argument}%
87   \expandafter\gdef\expandafter\dref@dref@output\expandafter{\expandafter\dref@expandable\expa
88   \dref@dref@output%
89   \dref@mkannotate{\textbackslash dref*{\#2\}}%
90 }

\drefvalueof
91 \def\drefvalueof#1{%
92   \dref@expandable{#1}%
93 }

\drefref
94 \def\drefref#1{%
95   \dref@unexpandable{#1}%
96 }

\dref@help@match
97 \newcommand{\dref@help@match}[2] {%
98   \ifstrmatch{#1}{#2}%
99 }

\dref@help
100 \newcommand{\dref@help}[2] [] {%
101   \pgfkeysifdefined{#2/help}{%
102     \pgfkeysvalueof{#2/help}%
103   }{#1}%
104 }

\drefsethelp
105 \csdef{dref@helps}{}
106 \newcommand{\drefsethelp}[2] {
107   \csdef{dref@help@#1}{#2}%
108   \listcsadd{dref@helps}{#1}%
109 }

\drefhelp
110 \newcommand{\drefhelp}[1] {
111   \renewcommand{\do}[1] {%
112     \dref@help@match{##1}{#1}{%
113       \csuse{dref@help@##1}%
114       \listbreak}{}%
115   }%
116   \ifcvoid{dref@helps}{}{%
117     \dolistcsloop{dref@helps}%
118   }%
119 }

```

`\dref@referenced`

```
120 \def\dref@notfound#1#2{
121   \ifdref@usagereport%
122     \dref@usagereport@notfound{#1}{#2}%
123   \else\relax\fi%
124 }
125 \def\dref@found#1#2{
126   \ifdref@usagereport%
127     \dref@usagereport@found{#1}{#2}%
128   \else\relax\fi%
129 }
130 \def\dref@referenced#1#2{
131   \ifdref@usagereport%
132     \dref@usagereport@referenced{#1}{#2}%
133   \else\relax\fi%
134 }
```

`\dref@let`

```
135 \def\dref@let#1{%
136   \def\@tmp##1=##2;{\pgfmathdeclarefunction*{##1}{0}{\pgfmathparse{##2}}}%
137   \renewcommand*\do}[1]{\@tmp##1;}%
138   \ifstrempy{#1}{}%
139     \docsvlist{#1}%
140   }%
141 }
142
143 % \end{macro}
144 %
145 %
146 % \begin{macro}{\dreflet}
147 %   \begin{macrocode}
148 \def\dreflet#1{%
149   \dref@let{#1}%
150 }
```

`\drefcalc`

```
151
152 \def\dref@parser#1#2{%
153   \edef\@tempa{#1}%
154   \csdef{dref@parser@result}{}%
155   \csdef{dref@parser@state}{}%
156   \expandafter\dref@parser@parse\@tempa\@nnil%
157   \xdef#2{\csuse{dref@parser@result}}%
158 }
159
160 \def\dref@parser@parse#1#2\@nnil{%
161   %\typeout{#1 State: \csuse{dref@parser@state}}%
162   \ifblank{#1#2}{%
163     \csxdef{dref@parser@result}{\csuse{dref@parser@result}\csuse{dref@parser@state}}%
```

```

164 }{%
165   \ifcsdef{dref@parser@\csuse{dref@parser@state}@#1}{%
166     \csuse{dref@parser@\csuse{dref@parser@state}@#1}#2\@nnil%
167   }{%
168     \csxdef{dref@parser@result}{\csuse{dref@parser@result}\csuse{dref@parser@state}#1}%
169     \csdef{dref@parser@state}{}%
170     \ifblank{#2}{}{%
171       \dref@parser@parse#2\@nnil%
172     }%
173   }%
174 }%
175 }
176
177 \csdef{dref@parser@d}{\csdef{dref@parser@state}{d}\dref@parser@parse}
178 \csdef{dref@parser@d@a}{\csdef{dref@parser@state}{da}\dref@parser@parse}
179 \csdef{dref@parser@da@t}{\csdef{dref@parser@state}{dat}\dref@parser@parse}
180 \csdef{dref@parser@dat@a}{\csdef{dref@parser@state}{data}\dref@parser@parse}
181 \csdef{dref@parser@data@(){\csdef{dref@parser@state}{data()}\dref@parser@parse}
182 \csdef{dref@parser@data@@"}{\dref@parser@tillquote}
183 \csdef{dref@parser@d@(){\dref@parser@tillparen}
184
185 \def\dref@parser@tillquote#1"#2\@nnil{%
186   \drefref{\dref@data@math@prefix #1}%
187   \csxdef{dref@parser@result}{\csuse{dref@parser@result}(\drefvalueof{\dref@data@math@prefix #1}
188   \csdef{dref@parser@state}{}%
189   \ifblank{#2}{}{\dref@parser@parse#2\@nnil}}
190
191 \def\dref@parser@tillparen#1)#2\@nnil{%
192   \drefref{\dref@data@math@prefix #1}%
193   \csxdef{dref@parser@result}{\csuse{dref@parser@result}(\drefvalueof{\dref@data@math@prefix #1}
194   \csdef{dref@parser@state}{}%
195   \ifblank{#2}{}{\dref@parser@parse#2\@nnil}}%
196
197
198 \def\dref@parser@end#1#2\@nnil{}
199 \csdef{dref@parser@@}{\typeout{end}\dref@parser@end}
200
201 \newcommand{\dref@calc}[1]{%
202   %\typeout{Before: '#1'}%
203   \dref@parser{#1}{\dref@calc@@argA}%
204   %\typeout{After: \dref@calc@@argA}%
205   \pgfmathparse{\dref@calc@@argA}%
206 }
207
208 \pgfset{/dref/let/.code={\dref@let{#1}}}
209 \pgfset{/dref/prefix/.code={\csdef{dref@prefix}{#1}}}
210 \def\drefprefix#1{\csdef{dref@prefix}{#1}}
211 \def\drefresult{0}
212 \def\drefcalc{\@ifstar\@@drefcalc@drefcalc}
213 \newcommand{\@drefcalc}[2][ ]{% Unstarred

```

```

214 \begingroup%
215 \pgfset{/pgf/number format/.cd, #1}%
216 \dref@calc{#2}%
217 \pgfmathprintnumber to [fixed, assume math mode=true, precision=10, 1000 sep={}] {\pgfmathresult} {\
218 \xdef\drefresult{\drefresult}%
219 \dref@format{\pgfmathresult}%
220 \dref@mkanotate{\textbackslash drefcalc\{#2\}}%
221 \endgroup%
222 }
223 \newcommand{\@@drefcalc}[2][ ]{ % Starred
224 \begingroup%
225 \pgfset{/pgf/number format/.cd, #1}%
226 \dref@calc{#2}%
227 \pgfmathprintnumber to [fixed, assume math mode=true, precision=10, 1000 sep={}] {\pgfmathresult} {\
228 \xdef\drefresult{\drefresult}%
229 \endgroup%
230 }

\drefformat
231 \newcommand{\dref@format}[2][ ]{ %
232 \pgfmathprintnumber [#1]{#2}%
233 }
234 \newcommand{\drefformat}[2][ ]{\dref@format [#1]{#2}}

data()
235 \gdef\dref@data@math@prefix{}
236 \pgfmathdeclarefunction{data}{1}{%
237 \begingroup%
238 \dref@unexpandable{\dref@data@math@prefix#1}%
239 \pgfmathparse{\dref@expandable{\dref@data@math@prefix#1}}%
240 \pgfmath@smuggleone\pgfmathresult%
241 \endgroup%
242 }
243 \long\def\drefprojection#1#2#3{%
244 \begingroup%
245 \def\dref@data@math@prefix{#1}%
246 \def\rename##1##2{\dref@unexpandable{#1/#1}\drefset{#2/#2}{\dref@expandable{#1/#1}}}%
247 \def\id##1{\rename{##1}{##1}}%
248 \def\calc##1##2{%
249 \begingroup%
250 \drefcalc{##1}%
251 \xdef\dref@project@result{\drefresult}
252 \endgroup%
253 \drefset{#2/#2}{\dref@project@result}%
254 }%
255 #3%
256 \endgroup%
257 }

\dref@makerow

```

```

258
259 \newtoks\dref@toks
260 \newcount\drefcellcount
261
262 \newcommand{\dref@makerow}[2]{%
263   {\global\dref@toks={}%
264     \drefcellcount=\z@%
265     \def\do##1{%
266       \advance\drefcellcount\@ne%
267       \def\@tempa{\doX{##1}}%
268       \expandafter\@tempa\expandafter{\the\drefcellcount}%
269     }%
270     \def\doX##1##2{%
271       \csxdef{cell\the\drefcellcount}{\detokenize{%
272         #2%
273       }}%
274     }%
275     \expandafter\def\expandafter\arglist\expandafter{##1}%
276     \expandafter\docsvlist\expandafter{\arglist}%
277     \@tempcntb=0\relax
278     {\loop\ifnum\@tempcntb<\drefcellcount
279       \advance\@tempcntb by 1\relax%
280       \ifnum \@tempcntb = 1%
281         \edef\@@next{\csuse{cell\the\@tempcntb}}%
282         \else%
283           \edef\@@next{&\csuse{cell\the\@tempcntb}}%
284         \fi%
285         \global%
286         \dref@toks%
287         \expandafter=%%
288         \expandafter{%
289           \the%
290           \expandafter\dref@toks%
291           \@@next}%
292         \repeat}%
293     }%
294     \typeout{LINE: \the\dref@toks}%
295     \expandafter\scantokens\expandafter{\the\dref@toks}}
296
297 \long\def\drefrow{\@ifstar\@@drefrow\@drefrow}
298 \def\@drefrow#1#2{\dref@makerow{#1}{\dref{#2}}} % Unstarred
299 \def\@@drefrow#1#2{\dref@makerow{#1}{#2}} % Starred

```

\dref@mkannotate

```

300
301 \expandafter\ifstrequal\expandafter{\dref@annotate}{pdfcomment}{
302   \RequirePackage{pdfcomment}
303 }
304
305 \def\dref@mkannotate@none#1{\relax}

```

```

306 \def\dref@mkannotate@footnote#1{\footnote{\texttt{#1}}}
307 \def\dref@mkannotate@pdfcomment#1{\pdfcomment[opacity=0.4,voffset=2ex]{#1}}
308
309 \newcommand{\dref@mkannotate}[1]{%
310   \ifcsdef{dref@mkannotate@\dref@annotate}{%
311     \csuse{dref@mkannotate@\dref@annotate}{#1}%
312   }{%
313     \dref@error{Value for annotate not supported: '\dref@annotate'}%
314   }%
315 }
316
317 \newcommand{\dref@annotate}[1]{%
318   \renewcommand{\dref@annotate}{#1}%
319 }

```

Usagereport

```

320 \ifdref@usagereport
321   \RequirePackage{xtab}
322   \RequirePackage{booktabs}
323 \fi

```

\dref@usagereport@referenced

```

324 \newcommand{\dref@usagereport@notfound}[2]{}
325 \newcommand{\dref@usagereport@found}[2]{}
326
327 \csdef{pgfdat@usagereport@keys}{}
328 \csdef{pgfdat@usagereport@matchedkeys}{}
329
330 \newcommand{\dref@usagereport@referenced}[2]{
331   \ifinlistcs{#2}{\dref@usagereport@referenced@#1}{}{
332     \listcsadd{dref@usagereport@referenced@#1}{#2}
333   }
334   \ifinlistcs{#1}{\dref@usagereport@keys}{}{
335     \listcsadd{dref@usagereport@keys}{#1}
336   }
337 }

```

\dref@usagereport@strippath

```

338 \expandafter\def\expandafter\dref@usagereport@strippath@\dref@datapath#1\blanktest{#1}
339
340 \newcommand{\dref@usagereport@strippath}[1]{%
341   \expandafter\ifstrmatch\expandafter{\expandafter^\dref@datapath.*$}{#1}%
342   {\dref@usagereport@strippath@#1\blanktest}%
343   {#1}%
344 }

```

usagereport@formatreferencelist

```

345 \newcommand{\dref@usagereport@formatreferencelist}[1]{%
346   \begingroup%

```



```

347 \def\sep{}%
348 \renewcommand{\do}[1]{\sep\ifdef{\hyperlink}{\hyperlink{page.##1}{##1}}{##1}\def\sep{, }}%
349 \dolistcsloop{dref@usagereport@referenced@#1}%
350 \endgroup%
351 }

```

\dref@usagereport@keyheader

```

352 \newif\ifdref@usagereport@keyheader@first
353 \dref@usagereport@keyheader@firsttrue
354 \newcommand{\dref@usagereport@keyheader}[1]{%
355 \ifdref@usagereport@keyheader@first%
356 \global\dref@usagereport@keyheader@firstfalse%
357 \else%
358 \\%
359 \fi%
360 \textbf{\ifdef{\hypertarget}%
361 {\hypertarget{##1}{\dref@usagereport@strippath{##1}}}%
362 {\dref@usagereport@strippath{##1}}}%
363 & \dref@usagereport@formatreferencelist{##1}%
364 & \pgfkeysifdefined{##1}{\pgfkeysvalueof{##1}}{\textbf{\color{red}undefined}}}%
365 }

```

\dref@usagereport@forhelp

```

366 \newif\ifdref@withhelp
367 \errorcontextlines=23
368 \newlength{\dreflinewidth}%
369 \newcommand{\dref@usagereport@forhelp}[1]{%
370 \begingroup%
371 \dref@withhelpfalse%
372 \renewcommand{\do}[1]{%
373 \dref@help@match{##1}{##1}{%
374 \dref@withhelpttrue%
375 }{}%
376 }%
377 \dolistcsloop{dref@usagereport@keys}%
378 \dref@usagereport@keyheader@firsttrue%
379 \renewcommand{\do}[1]{%
380 \dref@help@match{##1}{##1}{%
381 \dref@usagereport@keyheader{##1}%
382 \ifinlistcs{##1}{dref@usagereport@matchedkeys}{}%
383 \listcsadd{dref@usagereport@matchedkeys}{##1}%
384 }%
385 }{}%
386 }%
387 \ifdref@withhelp
388 \tablehead{\hline & Page & Value \\ \hline}%
389 \setlength\tabcolsep{3pt}%
390 \dreflinewidth=\linewidth%
391 \advance\dreflinewidth by -6\tabcolsep%
392 \begin{xtabular}{|p{0.7\dreflinewidth}|p{0.15\dreflinewidth}|p{0.15\dreflinewidth}|}%

```

```

393     \dolistcsloop{dref@usagereport@keys}\\\hline
394     \multicolumn{3}{|p{\linewidth}|}{\csuse{dref@help@#1}}\\\hline
395   \end{xtabular}%
396   \fi%
397
398   \endgroup%
399 }

```

\dref@usagereport@withouthelp

```

400 \newif\ifdref@withouthelp
401 \newcommand{\dref@usagereport@withouthelp}{%
402   \begingroup%
403   \dref@withouthelpfalse%
404   \renewcommand{\do}[1]{%
405     \ifinlistcs{##1}{dref@usagereport@matchedkeys}{}%
406     \dref@withouthelptrue%
407   }%
408 }%
409 \dolistcsloop{dref@usagereport@keys}%
410 \dref@usagereport@keyheader@firsttrue%
411 \renewcommand{\do}[1]{%
412   \ifinlistcs{##1}{dref@usagereport@matchedkeys}{}%
413   \dref@usagereport@keyheader{##1}%
414 }%
415 }%
416 \ifdref@withouthelp%
417   \tablehead{\hline Keys without Description      & Page & Value \\\hline}%
418   \setlength\tabcolsep{3pt}%
419   \dreflinewidth=\linewidth%
420   \advance\dreflinewidth by -6\tabcolsep%
421   \begin{xtabular}{|p{0.7\dreflinewidth}|p{0.15\dreflinewidth}|p{0.15\dreflinewidth}|}%
422     \dolistcsloop{dref@usagereport@keys}\\\hline
423   \end{xtabular}%
424   \fi%
425   \endgroup%
426 }

```

\drefusagereport

```

427 \newcommand{\drefusagereport}{%
428   \ifdref@usagereport%
429   \ifcvoid{dref@usagereport@keys}{\typeout{EMPTY}}{%
430     \begingroup%
431     \renewcommand{\do}[1]{%
432       \ifinlistcs{##1}{dref@usagereport@matchedkeys}{}%
433       \dref@usagereport@forhelp{##1}%
434     }%
435   }%
436   \dolistcsloop{dref@helps} % For all help text
437   \dref@usagereport@withouthelp\relax
438   \endgroup%

```

```

439 }% cempty @keys
440 \fi%
441 }

\drefassert
442 \newcommand{\drefassert}[1]{%
443 \begingroup%
444 \drefcalc*{#1}%
445 \expandafter\ifstrequal\expandafter{\drefresult}{1}{%
446 \typeout{Assertion holds: #1}%
447 }{%
448 \ifdref@noassert%
449 \typeout{Assertion failed: #1}%
450 \else%
451 \dref@error{Assertion failed: #1}%
452 \fi%
453 }%
454 \endgroup%
455 }

\drefrel
456 \newif\if@dref@valuemustderef%
457 \newif\if@dref@basemustderef%
458 \newif\if@dref@increase%
459 \newif\if@dref@product%
460 \newif\if@dref@factor%
461 \newif\if@dref@delta%
462 \newif\if@dref@percent%
463 \newif\if@dref@abs%
464 \newif\if@dref@neg%
465 \pgfkeys{%
466 \dref@datapath/.DUMMY/.initial=1
467 }
468 \pgfkeys{%
469 /dref/.cd,%
470 value/.initial = /.DUMMY,%
471 base/.initial = /.DUMMY,%
472 divide/.initial = 1,%
473 value plain/.is if=@dref@valuemustderef,%
474 value plain/.default=false,%
475 value plain=true,%
476 base plain/.is if=@dref@basemustderef,%
477 base plain/.default=false,%
478 base plain=true,%
479 factor/.is if=@dref@factor,%
480 factor/.default=true,%
481 factor=false,%
482 delta/.is if=@dref@delta,%
483 delta/.default=true,%
484 delta=false,%

```

```

485 scale/.is if=@dref@product,%
486 scale/.default=true,%
487 scale=false,%
488 product/.is if=@dref@product,%
489 product/.default=true,%
490 product=false,%
491 increase/.is if=@dref@increase,%
492 increase/.default=true,%
493 increase=false,%
494 overhead/.is if=@dref@increase,%
495 overhead/.default=true,%
496 overhead=false,%
497 percent/.is if=@dref@percent,%
498 percent/.default=true,%
499 percent=false,%
500 abs/.is if=@dref@abs,%
501 abs/.default=true,%
502 abs=false,%
503 negate/.is if=@dref@neg,%
504 negate/.default=true,%
505 negate=false,%
506 }
507
508 \def\drefrel{\@ifstar\@@drefrel@drefrel}
509
510 \newcommand{\@drefrel}[2] [] {%
511   \@@drefrel[#1]{#2}%
512   \@@drefrel@result%
513   \dref@mkanotate{\textbackslash{}drefrel[#1]\{#2\}}%
514 }
515
516 \newcommand{\@@drefrel}[2] [] {%
517   \begingroup%
518   \pgfkeys{/pgf/fpu}%
519   \pgfkeys{/dref/.cd,#1}%
520   \pgfkeys{/dref/value=#2}%
521   \if@dref@valuemustderef%
522     \drefref{\pgfkeysvalueof{/dref/value}}%
523     \edef\drefvalue{\drefvalueof{\pgfkeysvalueof{/dref/value}}}%
524   \else%
525     \def\drefvalue{\pgfkeysvalueof{/dref/value}}%
526   \fi%
527   \if@dref@basemustderef%
528     \drefref{\pgfkeysvalueof{/dref/base}}%
529     \def\drefbase{\drefvalueof{\pgfkeysvalueof{/dref/base}}}%
530   \else%
531     \def\drefbase{\pgfkeysvalueof{/dref/base}}%
532   \fi%
533   \xdef\drefresult{\drefvalue}%
534   \if@dref@increase%

```

```

535     \pgfmathparse{((\drefvalue) - (\drefbase)) / (\drefbase)}%
536     \def\drefresult{\pgfmathresult}%
537 \else%
538     \if@dref@factor%
539         \pgfmathparse{(\drefvalue) / (\drefbase)}%
540         \def\drefresult{\pgfmathresult}%
541     \else%
542         \if@dref@delta%
543             \pgfmathparse{(\drefvalue) - (\drefbase)}%
544             \def\drefresult{\pgfmathresult}%
545         \else%
546             \if@dref@product%
547                 \pgfmathparse{(\drefvalue) * (\drefbase)}%
548                 \def\drefresult{\pgfmathresult}%
549             \else
550                 \def\drefresult{\drefvalue}%
551             \fi
552         \fi%
553     \fi%
554 \fi%
555 % Percent
556 \if@dref@percent%
557     \pgfmathparse{(\drefresult)*100.0}%
558     \def\drefresult{\pgfmathresult}%
559 \fi%
560 % Absolute Value
561 \if@dref@abs%
562     \pgfmathparse{abs(\drefresult)}%
563     \def\drefresult{\pgfmathresult}%
564 \fi%
565 % Negative Value
566 \if@dref@neg%
567     \pgfmathparse{-1.0*(\drefresult)}%
568     \def\drefresult{\pgfmathresult}%
569 \fi%
570 \pgfmathparse{\drefresult/\pgfkeysvalueof{/dref/divide}}%
571 \pgfmathprintnumberto[fixed,assume math mode=true,precision=10,1000 sep={}]{\pgfmathresult}{\
572 \pgfmathprintnumberto{\pgfmathresult}{\@@drefrel@result}%
573 \xdef\drefresult{\drefresult}%
574 \xdef\@@drefrel@result{\@@drefrel@result}%
575 \endgroup%
576 }
577
578 %

```