The magicnum package

Heiko Oberdiek*

2019/11/29 v1.7

Abstract

This packages allows to access magic numbers by a hierarchical name system.

Contents

*Please report any issues at https://github.com/ho-tex/magicnum/issues

1 Documentation

1.1 Introduction

Especially since ε-TEX there are many integer values with special meanings, such as catcodes, group types, ... Package etex, enabled by options, defines macros in the user namespace for these values.

This package goes another approach for storing the names and values.

- If LuaTEX is available, they are stored in Lua tables.
- Without LuaTEX they are remembered using internal macros.

1.2 User interface

The integer values and names are organized in a hierarchical scheme of categories with the property names as leaves. Example: ε-TEX's `\currentgrouplevel` reports 2 for a group caused by `\hbox`. This package has choosen to organize the group types in a main category `etex` and its subcategory `grouptype`:

`etex.grouptype.hbox` = 2

The property name `hbox` in category `etex.grouptype` has value 2. Dots are used to separate components.

If you want to have the value, the access key is constructed by the category with all its components and the property name. For the opposite the value is used instead of the property name.

Values are always integers (including negative numbers).

1.2.1 `\magicnum`

`\magicnum` {⟨*access key*⟩}

Macro `\magicnum` expects an access key as argument and expands to the requested data. The macro is always expandable. In case of errors the expansion result is empty.

The same macro is also used for getting a property name. In this case the property name part in the access key is replaced by the value.

The catcodes of the resulting numbers and strings follow TEX's tradition of `\string`, `\meaning`, ...: The space has catcode 10 (`tex.catcode.space`) and the other characters have catcode 12 (`tex.catcode.other`).

Examples:

`\magicnum{etex.grouptype.hbox}` ⇒ 2
`\magicnum{tex.catcode.14}` ⇒ `comment`
`\magicnum{tex.catcode.undefined}` ⇒ ∅

1.2.2 Properties

- The components of a category are either subcategories or key value pairs, but not both.
- The full specified property names are unique and thus has one integer value exactly.
- Also the values inside a category are unique. This condition is a prerequisite for the reverse mapping of `\magicnum`.
- All names start with a letter. Only letters or digits may follow.

1.3 Data

1.3.1 Category `tex.catcode`

```
tex.catcode.escape        0
tex.catcode.begingroup    1
tex.catcode.endgroup      2
tex.catcode.math          3
tex.catcode.align         4
tex.catcode.eol           5
tex.catcode.parameter     6
tex.catcode.superscript   7
tex.catcode.subscript     8
tex.catcode.ignore        9
tex.catcode.space         10
tex.catcode.letter        11
tex.catcode.other         12
tex.catcode.active        13
tex.catcode.comment       14
tex.catcode.invalid       15
```

1.3.2 Category `etex.grouptype`

```
etex.grouptype.bottomlevel   0
etex.grouptype.simple        1
etex.grouptype.hbox          2
etex.grouptype.adjustedhbox  3
etex.grouptype.vbox          4
etex.grouptype.align         5
etex.grouptype.noalign       6
etex.grouptype.output        8
etex.grouptype.math          9
etex.grouptype.disc          10
etex.grouptype.insert        11
etex.grouptype.vcenter       12
etex.grouptype.mathchoice    13
etex.grouptype.semisimple    14
etex.grouptype.mathshift     15
etex.grouptype.mathleft      16
```

1.3.3 Category `etex.iftype`

```
etex.iftype.none      0
etex.iftype.char      1
etex.iftype.cat       2
etex.iftype.num       3
etex.iftype.dim       4
etex.iftype.odd       5
etex.iftype.vmode     6
etex.iftype.hmode     7
etex.iftype.mmode     8
etex.iftype.inner     9
etex.iftype.void      10
etex.iftype.hbox      11
etex.iftype.vbox      12
etex.iftype.x         13
etex.iftype.eof       14
etex.iftype.true      15
etex.iftype.false     16
etex.iftype.case      17
etex.iftype.defined   18
etex.iftype.csname    19
etex.iftype.fontchar  20
```

1.3.4 Category `etex.nodetype`

```
etex.nodetype.none      -1
etex.nodetype.char      0
etex.nodetype.hlist     1
etex.nodetype.vlist     2
etex.nodetype.rule      3
etex.nodetype.ins       4
etex.nodetype.mark      5
etex.nodetype.adjust    6
etex.nodetype.ligature  7
etex.nodetype.disc      8
etex.nodetype.whatsit   9
etex.nodetype.math      10
etex.nodetype.glue      11
etex.nodetype.kern      12
etex.nodetype.penalty   13
etex.nodetype.unset     14
etex.nodetype.maths     15
```

1.3.5 Category `etex.interactionmode`

```
etex.interactionmode.batch      0
etex.interactionmode.nonstop    1
etex.interactionmode.scroll     2
etex.interactionmode.errorstop  3
```

1.3.6 Category `luatex.pdfliteral.mode`

```
luatex.pdfliteral.mode.setorigin  0
luatex.pdfliteral.mode.page       1
luatex.pdfliteral.mode.direct     2
```

2 Implementation

1 ⟨*package⟩

2.1 Reload check and package identification

Reload check, especially if the package is not used with LaTeX.

```
2 \begingroup\catcode61\catcode48\catcode32=10\relax%
3   \catcode13=5 % ^^M
4   \endlinechar=13 %
5   \catcode35=6 % #
6   \catcode39=12 % '
7   \catcode44=12 % ,
8   \catcode45=12 % -
9   \catcode46=12 % .
10  \catcode58=12 % :
11  \catcode64=11 % @
12  \catcode123=1 % {
13  \catcode125=2 % }
14  \expandafter\let\expandafter\x\csname ver@magicnum.sty\endcsname
15  \ifx\x\relax % plain-TeX, first loading
16  \else
17    \def\empty{}%
18    \ifx\x\empty % LaTeX, first loading,
19      % variable is initialized, but \ProvidesPackage not yet seen
20    \else
21      \expandafter\ifx\csname PackageInfo\endcsname\relax
22        \def\x#1#2{%
23          \immediate\write-1{Package #1 Info: #2.}%
24        }%
25      \else
26        \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
27      \fi
28      \x{magicnum}{The package is already loaded}%
29      \aftergroup\endinput
30    \fi
31  \fi
32 \endgroup%
```

Package identification:

```
33 \begingroup\catcode61\catcode48\catcode32=10\relax%
34   \catcode13=5 % ^^M
35   \endlinechar=13 %
36   \catcode35=6 % #
37   \catcode39=12 % '
38   \catcode40=12 % (
39   \catcode41=12 % )
40   \catcode44=12 % ,
41   \catcode45=12 % -
42   \catcode46=12 % .
43   \catcode47=12 % /
44   \catcode58=12 % :
45   \catcode64=11 % @
```

```
  \catcode91=12 % [
  \catcode93=12 % ]
  \catcode123=1 % {
  \catcode125=2 % }
  \expandafter\ifx\csname ProvidesPackage\endcsname\relax
    \def\x#1#2#3[#4]{\endgroup
      \immediate\write-1{Package: #3 #4}%
      \xdef#1{#4}%
    }%
  \else
    \def\x#1#2[#3]{\endgroup
      #2[{#3}]%
      \ifx#1\@undefined
        \xdef#1{#3}%
      \fi
      \ifx#1\relax
        \xdef#1{#3}%
      \fi
    }%
  \fi
\expandafter\x\csname ver@magicnum.sty\endcsname
\ProvidesPackage{magicnum}%
  [2019/11/29 v1.7 Magic numbers (HO)]%
```

2.2 Catcodes

```
\begingroup\catcode61\catcode48\catcode32=10\relax%
  \catcode13=5 % ^^M
  \endlinechar=13 %
  \catcode123=1 % {
  \catcode125=2 % }
  \catcode64=11 % @
  \def\x{\endgroup
    \expandafter\edef\csname magicnum@AtEnd\endcsname{%
      \endlinechar=\the\endlinechar\relax
      \catcode13=\the\catcode13\relax
      \catcode32=\the\catcode32\relax
      \catcode35=\the\catcode35\relax
      \catcode61=\the\catcode61\relax
      \catcode64=\the\catcode64\relax
      \catcode123=\the\catcode123\relax
      \catcode125=\the\catcode125\relax
    }%
  }%
\x\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode64=11 % @
\catcode123=1 % {
\catcode125=2 % }
\def\TMP@EnsureCode#1#2{%
  \edef\magicnum@AtEnd{%
    \magicnum@AtEnd
    \catcode#1=\the\catcode#1\relax
  }%
  \catcode#1=#2\relax
}
```

```
\TMP@EnsureCode{34}{12}% "
\TMP@EnsureCode{39}{12}% '
\TMP@EnsureCode{40}{12}% (
\TMP@EnsureCode{41}{12}% )
\TMP@EnsureCode{42}{12}% *
\TMP@EnsureCode{44}{12}% ,
\TMP@EnsureCode{45}{12}% -
\TMP@EnsureCode{46}{12}% .
\TMP@EnsureCode{47}{12}% /
\TMP@EnsureCode{58}{12}% :
\TMP@EnsureCode{60}{12}% <
\TMP@EnsureCode{62}{12}% >
\TMP@EnsureCode{91}{12}% [
\TMP@EnsureCode{93}{12}% ]
\edef\magicnum@AtEnd{\magicnum@AtEnd\noexpand\endinput}
```

2.3 Check for previous definition

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname newcommand\endcsname\relax
  \expandafter\ifx\csname magicnum\endcsname\relax
  \else
    \input infwarerr.sty\relax
    \@PackageError{magicnum}{%
      \string\magicnum\space is already defined%
    }\@ehc
  \fi
\else
  \newcommand*{\magicnum}{}%
\fi
```

2.4 Without LuaTEX

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname directlua\endcsname\relax
```

`\magicnum`

```
  \begingroup\expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname ifcsname\endcsname\relax
    \def\magicnum#1{%
      \expandafter\ifx\csname MG@#1\endcsname\relax
      \else
        \csname MG@#1\endcsname
      \fi
    }%
  \else
    \begingroup
      \edef\x{\endgroup
        \def\noexpand\magicnum##1{%
          \expandafter\noexpand\csname
          ifcsname\endcsname MG@##1\noexpand\endcsname
            \noexpand\csname MG@##1%
                 \noexpand\expandafter\noexpand\endcsname
          \expandafter\noexpand\csname fi\endcsname
        }%
      }%
    \x
  \fi

\else
```

2.5 With LuaTEX

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname RequirePackage\endcsname\relax
  \input tex.sty\relax
  \input infwarerr.sty\relax
\else
  \RequirePackage{iftex}[2019/11/07]%
  \RequirePackage{infwarerr}[2010/04/08]%
\fi
```

\magicnum@directlua

```
\ifnum\luatexversion<36 %
  \def\magicnum@directlua{\directlua0 }%
\else
  \let\magicnum@directlua\directlua
\fi

\magicnum@directlua{%
  require("magicnum")%
}%
\begingroup
  \def\x{2019/11/29 v1.7}%
  \def\StripPrefix#1>{}%
  \edef\x{\expandafter\StripPrefix\meaning\x}%
  \edef\y{%
    \magicnum@directlua{%
      if oberdiek.magicnum.getversion then %
        oberdiek.magicnum.getversion()%
      end%
    }%
  }%
  \ifx\x\y
  \else
    \@PackageError{magicnum}{%
      Wrong version of lua module.\MessageBreak
      Package version: \x\MessageBreak
      Lua module: \y
    }\@ehc
  \fi
\endgroup
```

\luaescapestring

```
\begingroup
  \expandafter\ifx\csname luaescapestring\endcsname\relax
    \directlua{%
      if tex.enableprimitives then %
        tex.enableprimitives('magicnum@', {'luaescapestring'})%
      end%
    }%
    \global\let\luaescapestring\magicnum@luaescapestring
  \fi
  \expandafter\ifx\csname luaescapestring\endcsname\relax
    \escapechar=92 %
    \@PackageError{magicnum}{%
      Missing \string\luaescapestring
    }\@ehc
  \fi
\endgroup
```

\magicnum

```
  \def\magicnum#1{%
    \magicnum@directlua{%
      oberdiek.magicnum.get("\luaescapestring{#1}")%
    }%
  }%

  \expandafter\magicnum@AtEnd
\fi%
```

⟨/package⟩

2.6 Data

2.6.1 Plain data

```
⟨*data⟩
tex.catcode
  escape = 0
  begingroup = 1
  endgroup = 2
  math = 3
  align = 4
  eol = 5
  parameter = 6
  superscript = 7
  subscript = 8
  ignore = 9
  space = 10
  letter = 11
  other = 12
  active = 13
  comment = 14
  invalid = 15
etex.grouptype
  bottomlevel = 0
  simple = 1
  hbox = 2
  adjustedhbox = 3
  vbox = 4
  align = 5
  noalign = 6
  output = 8
  math = 9
  disc = 10
  insert = 11
  vcenter = 12
  mathchoice = 13
  semisimple = 14
  mathshift = 15
  mathleft = 16
etex.iftype
  none = 0
  char = 1
  cat = 2
  num = 3
  dim = 4
  odd = 5
  vmode = 6
```

```
  hmode = 7
  mmode = 8
  inner = 9
  void = 10
  hbox = 11
  vbox = 12
  x = 13
  eof = 14
  true = 15
  false = 16
  case = 17
  defined = 18
  csname = 19
  fontchar = 20
etex.nodetype
  none = -1
  char = 0
  hlist = 1
  vlist = 2
  rule = 3
  ins = 4
  mark = 5
  adjust = 6
  ligature = 7
  disc = 8
  whatsit = 9
  math = 10
  glue = 11
  kern = 12
  penalty = 13
  unset = 14
  maths = 15
etex.interactionmode
  batch = 0
  nonstop = 1
  scroll = 2
  errorstop = 3
luatex.pdfliteral.mode
  setorigin = 0
  page = 1
  direct = 2
⟨/data⟩
```

2.6.2 Data for TEX

```
⟨*package⟩
```

`\magicnum@add`

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname detokenize\endcsname\relax
  \def\magicnum@add#1#2#3{%
    \expandafter\magicnum@@add
        \csname MG@#1.#2\expandafter\endcsname
        \csname MG@#1.#3\endcsname
       {#3}{#2}%
  }%
  \def\magicnum@@add#1#2#3#4{%
    \def#1{#3}%
    \def#2{#4}%
    \edef#1{%
```

```
      \expandafter\strip@prefix\meaning#1%
    }%
    \edef#2{%
      \expandafter\strip@prefix\meaning#2%
    }%
  }%
  \expandafter\ifx\csname strip@prefix\endcsname\relax
    \def\strip@prefix#1->{}%
  \fi
\else
  \def\magicnum@add#1#2#3{%
    \expandafter\edef\csname MG@#1.#2\endcsname{%
      \detokenize{#3}%
    }%
    \expandafter\edef\csname MG@#1.#3\endcsname{%
      \detokenize{#2}%
    }%
  }%
\fi

\magicnum@add{tex.catcode}{escape}{0}
\magicnum@add{tex.catcode}{begingroup}{1}
\magicnum@add{tex.catcode}{endgroup}{2}
\magicnum@add{tex.catcode}{math}{3}
\magicnum@add{tex.catcode}{align}{4}
\magicnum@add{tex.catcode}{eol}{5}
\magicnum@add{tex.catcode}{parameter}{6}
\magicnum@add{tex.catcode}{superscript}{7}
\magicnum@add{tex.catcode}{subscript}{8}
\magicnum@add{tex.catcode}{ignore}{9}
\magicnum@add{tex.catcode}{space}{10}
\magicnum@add{tex.catcode}{letter}{11}
\magicnum@add{tex.catcode}{other}{12}
\magicnum@add{tex.catcode}{active}{13}
\magicnum@add{tex.catcode}{comment}{14}
\magicnum@add{tex.catcode}{invalid}{15}
\magicnum@add{etex.grouptype}{bottomlevel}{0}
\magicnum@add{etex.grouptype}{simple}{1}
\magicnum@add{etex.grouptype}{hbox}{2}
\magicnum@add{etex.grouptype}{adjustedhbox}{3}
\magicnum@add{etex.grouptype}{vbox}{4}
\magicnum@add{etex.grouptype}{align}{5}
\magicnum@add{etex.grouptype}{noalign}{6}
\magicnum@add{etex.grouptype}{output}{8}
\magicnum@add{etex.grouptype}{math}{9}
\magicnum@add{etex.grouptype}{disc}{10}
\magicnum@add{etex.grouptype}{insert}{11}
\magicnum@add{etex.grouptype}{vcenter}{12}
\magicnum@add{etex.grouptype}{mathchoice}{13}
\magicnum@add{etex.grouptype}{semisimple}{14}
\magicnum@add{etex.grouptype}{mathshift}{15}
\magicnum@add{etex.grouptype}{mathleft}{16}
\magicnum@add{etex.iftype}{none}{0}
\magicnum@add{etex.iftype}{char}{1}
\magicnum@add{etex.iftype}{cat}{2}
\magicnum@add{etex.iftype}{num}{3}
\magicnum@add{etex.iftype}{dim}{4}
\magicnum@add{etex.iftype}{odd}{5}
\magicnum@add{etex.iftype}{vmode}{6}
```

```
\magicnum@add{etex.iftype}{hmode}{7}
\magicnum@add{etex.iftype}{mmode}{8}
\magicnum@add{etex.iftype}{inner}{9}
\magicnum@add{etex.iftype}{void}{10}
\magicnum@add{etex.iftype}{hbox}{11}
\magicnum@add{etex.iftype}{vbox}{12}
\magicnum@add{etex.iftype}{x}{13}
\magicnum@add{etex.iftype}{eof}{14}
\magicnum@add{etex.iftype}{true}{15}
\magicnum@add{etex.iftype}{false}{16}
\magicnum@add{etex.iftype}{case}{17}
\magicnum@add{etex.iftype}{defined}{18}
\magicnum@add{etex.iftype}{csname}{19}
\magicnum@add{etex.iftype}{fontchar}{20}
\magicnum@add{etex.nodetype}{none}{-1}
\magicnum@add{etex.nodetype}{char}{0}
\magicnum@add{etex.nodetype}{hlist}{1}
\magicnum@add{etex.nodetype}{vlist}{2}
\magicnum@add{etex.nodetype}{rule}{3}
\magicnum@add{etex.nodetype}{ins}{4}
\magicnum@add{etex.nodetype}{mark}{5}
\magicnum@add{etex.nodetype}{adjust}{6}
\magicnum@add{etex.nodetype}{ligature}{7}
\magicnum@add{etex.nodetype}{disc}{8}
\magicnum@add{etex.nodetype}{whatsit}{9}
\magicnum@add{etex.nodetype}{math}{10}
\magicnum@add{etex.nodetype}{glue}{11}
\magicnum@add{etex.nodetype}{kern}{12}
\magicnum@add{etex.nodetype}{penalty}{13}
\magicnum@add{etex.nodetype}{unset}{14}
\magicnum@add{etex.nodetype}{maths}{15}
\magicnum@add{etex.interactionmode}{batch}{0}
\magicnum@add{etex.interactionmode}{nonstop}{1}
\magicnum@add{etex.interactionmode}{scroll}{2}
\magicnum@add{etex.interactionmode}{errorstop}{3}
\magicnum@add{luatex.pdfliteral.mode}{setorigin}{0}
\magicnum@add{luatex.pdfliteral.mode}{page}{1}
\magicnum@add{luatex.pdfliteral.mode}{direct}{2}

\magicnum@AtEnd%
⟨/package⟩
```

2.6.3 Lua module

```
⟨*lua⟩

oberdiek = oberdiek or {}
local magicnum = oberdiek.magicnum or {}
oberdiek.magicnum = magicnum

function magicnum.getversion()
  tex.write("2019/11/29 v1.7")
end

local data = {
  ["tex.catcode"] = {
    [0] = "escape",
    [1] = "begingroup",
    [2] = "endgroup",
    [3] = "math",
    [4] = "align",
```

```
    [5] = "eol",
    [6] = "parameter",
    [7] = "superscript",
    [8] = "subscript",
    [9] = "ignore",
    [10] = "space",
    [11] = "letter",
    [12] = "other",
    [13] = "active",
    [14] = "comment",
    [15] = "invalid",
    ["active"] = 13,
    ["align"] = 4,
    ["begingroup"] = 1,
    ["comment"] = 14,
    ["endgroup"] = 2,
    ["eol"] = 5,
    ["escape"] = 0,
    ["ignore"] = 9,
    ["invalid"] = 15,
    ["letter"] = 11,
    ["math"] = 3,
    ["other"] = 12,
    ["parameter"] = 6,
    ["space"] = 10,
    ["subscript"] = 8,
    ["superscript"] = 7
  },
  ["etex.grouptype"] = {
    [0] = "bottomlevel",
    [1] = "simple",
    [2] = "hbox",
    [3] = "adjustedhbox",
    [4] = "vbox",
    [5] = "align",
    [6] = "noalign",
    [8] = "output",
    [9] = "math",
    [10] = "disc",
    [11] = "insert",
    [12] = "vcenter",
    [13] = "mathchoice",
    [14] = "semisimple",
    [15] = "mathshift",
    [16] = "mathleft",
    ["adjustedhbox"] = 3,
    ["align"] = 5,
    ["bottomlevel"] = 0,
    ["disc"] = 10,
    ["hbox"] = 2,
    ["insert"] = 11,
    ["math"] = 9,
    ["mathchoice"] = 13,
    ["mathleft"] = 16,
    ["mathshift"] = 15,
    ["noalign"] = 6,
    ["output"] = 8,
    ["semisimple"] = 14,
```

```
    ["simple"] = 1,
    ["vbox"] = 4,
    ["vcenter"] = 12
  },
  ["etex.iftype"] = {
    [0] = "none",
    [1] = "char",
    [2] = "cat",
    [3] = "num",
    [4] = "dim",
    [5] = "odd",
    [6] = "vmode",
    [7] = "hmode",
    [8] = "mmode",
    [9] = "inner",
    [10] = "void",
    [11] = "hbox",
    [12] = "vbox",
    [13] = "x",
    [14] = "eof",
    [15] = "true",
    [16] = "false",
    [17] = "case",
    [18] = "defined",
    [19] = "csname",
    [20] = "fontchar",
    ["case"] = 17,
    ["cat"] = 2,
    ["char"] = 1,
    ["csname"] = 19,
    ["defined"] = 18,
    ["dim"] = 4,
    ["eof"] = 14,
    ["false"] = 16,
    ["fontchar"] = 20,
    ["hbox"] = 11,
    ["hmode"] = 7,
    ["inner"] = 9,
    ["mmode"] = 8,
    ["none"] = 0,
    ["num"] = 3,
    ["odd"] = 5,
    ["true"] = 15,
    ["vbox"] = 12,
    ["vmode"] = 6,
    ["void"] = 10,
    ["x"] = 13
  },
  ["etex.nodetype"] = {
    [-1] = "none",
    [0] = "char",
    [1] = "hlist",
    [2] = "vlist",
    [3] = "rule",
    [4] = "ins",
    [5] = "mark",
    [6] = "adjust",
    [7] = "ligature",
```

```
    [8] = "disc",
    [9] = "whatsit",
    [10] = "math",
    [11] = "glue",
    [12] = "kern",
    [13] = "penalty",
    [14] = "unset",
    [15] = "maths",
    ["adjust"] = 6,
    ["char"] = 0,
    ["disc"] = 8,
    ["glue"] = 11,
    ["hlist"] = 1,
    ["ins"] = 4,
    ["kern"] = 12,
    ["ligature"] = 7,
    ["mark"] = 5,
    ["math"] = 10,
    ["maths"] = 15,
    ["none"] = -1,
    ["penalty"] = 13,
    ["rule"] = 3,
    ["unset"] = 14,
    ["vlist"] = 2,
    ["whatsit"] = 9
  },
  ["etex.interactionmode"] = {
    [0] = "batch",
    [1] = "nonstop",
    [2] = "scroll",
    [3] = "errorstop",
    ["batch"] = 0,
    ["errorstop"] = 3,
    ["nonstop"] = 1,
    ["scroll"] = 2
  },
  ["luatex.pdfliteral.mode"] = {
    [0] = "setorigin",
    [1] = "page",
    [2] = "direct",
    ["direct"] = 2,
    ["page"] = 1,
    ["setorigin"] = 0
  }
}
function magicnum.get(name)
  local startpos, endpos, category, entry =
      string.find(name, "^(%a[%a%d%.]*)%.(-?[%a%d]+)$")
  if not entry then
    return
  end
  local node = data[category]
  if not node then
    return
  end
  local num = tonumber(entry)
  local value
  if num then
```

```
    value = node[num]
    if not value then
      return
    end
  else
    value = node[entry]
    if not value then
      return
    end
    value = "" .. value
  end
  tex.write(value)
end
```

⟨/lua⟩

3 Test

3.1 Catcode checks for loading

⟨*test1⟩

```
\catcode`\{=1 %
\catcode`\}=2 %
\catcode`\#=6 %
\catcode`\@=11 %
\expandafter\ifx\csname count@\endcsname\relax
  \countdef\count@=255 %
\fi
\expandafter\ifx\csname @gobble\endcsname\relax
  \long\def\@gobble#1{}%
\fi
\expandafter\ifx\csname @firstofone\endcsname\relax
  \long\def\@firstofone#1{#1}%
\fi
\expandafter\ifx\csname loop\endcsname\relax
  \expandafter\@firstofone
\else
  \expandafter\@gobble
\fi
{%
  \def\loop#1\repeat{%
    \def\body{#1}%
    \iterate
  }%
  \def\iterate{%
    \body
      \let\next\iterate
    \else
      \let\next\relax
    \fi
    \next
  }%
  \let\repeat=\fi
}%
\def\RestoreCatcodes{}
\count@=0 %
\loop
  \edef\RestoreCatcodes{%
    \RestoreCatcodes
```

```
    \catcode\the\count@=\the\catcode\count@\relax
  }%
\ifnum\count@<255 %
  \advance\count@ 1 %
\repeat

\def\RangeCatcodeInvalid#1#2{%
  \count@=#1\relax
  \loop
    \catcode\count@=15 %
  \ifnum\count@<#2\relax
    \advance\count@ 1 %
  \repeat
}
\def\RangeCatcodeCheck#1#2#3{%
  \count@=#1\relax
  \loop
    \ifnum#3=\catcode\count@
    \else
      \errmessage{%
        Character \the\count@\space
        with wrong catcode \the\catcode\count@\space
        instead of \number#3%
      }%
    \fi
  \ifnum\count@<#2\relax
    \advance\count@ 1 %
  \repeat
}
\def\space{ }
\expandafter\ifx\csname LoadCommand\endcsname\relax
  \def\LoadCommand{\input magicnum.sty\relax}%
\fi
\def\Test{%
  \RangeCatcodeInvalid{0}{47}%
  \RangeCatcodeInvalid{58}{64}%
  \RangeCatcodeInvalid{91}{96}%
  \RangeCatcodeInvalid{123}{255}%
  \catcode`\@=12 %
  \catcode`\\=0 %
  \catcode`\%=14 %
  \LoadCommand
  \RangeCatcodeCheck{0}{36}{15}%
  \RangeCatcodeCheck{37}{37}{14}%
  \RangeCatcodeCheck{38}{47}{15}%
  \RangeCatcodeCheck{48}{57}{12}%
  \RangeCatcodeCheck{58}{63}{15}%
  \RangeCatcodeCheck{64}{64}{12}%
  \RangeCatcodeCheck{65}{90}{11}%
  \RangeCatcodeCheck{91}{91}{15}%
  \RangeCatcodeCheck{92}{92}{0}%
  \RangeCatcodeCheck{93}{96}{15}%
  \RangeCatcodeCheck{97}{122}{11}%
  \RangeCatcodeCheck{123}{255}{15}%
  \RestoreCatcodes
}
\Test
\csname @@end\endcsname
```

```
\end
```
⟨/test1⟩

3.2 Test data

⟨*testplain⟩
```
\input magicnum.sty\relax
\def\Test#1#2{%
  \edef\result{\magicnum{#1}}%
  \edef\expect{#2}%
  \edef\expect{\expandafter\stripprefix\meaning\expect}%
  \ifx\result\expect
  \else
    \errmessage{%
      Failed: [#1] % hash-ok
      returns [\result] instead of [\expect]%
    }%
  \fi
}
\def\stripprefix#1->{}
```
⟨/testplain⟩

⟨*testlatex⟩
```
\NeedsTeXFormat{LaTeX2e}
\documentclass{minimal}
\usepackage{magicnum}[2016/05/16]
\usepackage{qstest}
\IncludeTests{*}
\LogTests{log}{*}{*}
\newcommand*{\Test}[2]{%
  \Expect*{\magicnum{#1}}{#2}%
}
\begin{qstest}{magicnum}{magicnum}
```
⟨/testlatex⟩

⟨*testdata⟩
```
\Test{tex.catcode.escape}{0}
\Test{tex.catcode.invalid}{15}
\Test{tex.catcode.unknown}{}
\Test{tex.catcode.0}{escape}
\Test{tex.catcode.15}{invalid}
\Test{etex.iftype.true}{15}
\Test{etex.iftype.false}{16}
\Test{etex.iftype.15}{true}
\Test{etex.iftype.16}{false}
\Test{etex.nodetype.none}{-1}
\Test{etex.nodetype.-1}{none}
\Test{luatex.pdfliteral.mode.direct}{2}
\Test{luatex.pdfliteral.mode.1}{page}
\Test{}{}
\Test{unknown}{}
\Test{unknown.foo.bar}{}
\Test{unknown.foo.4}{}
```
⟨/testdata⟩

⟨*testplain⟩
```
\csname @@end\endcsname
\end
```
⟨/testplain⟩
⟨*testlatex⟩

```
\end{qstest}
\csname @@end\endcsname
⟨/testlatex⟩
```

3.3 Small test for iniTEX

```
⟨*test4⟩
\catcode`\{=1
\catcode`\}=2
\catcode`\#=6
\input magicnum.sty\relax
\edef\x{\magicnum{tex.catcode.15}}
\edef\y{invalid}
\def\Strip#1>{}
\edef\y{\expandafter\Strip\meaning\y}
\ifx\x\y
  \immediate\write16{Ok}%
\else
  \errmessage{\x<>\y}%
\fi
\csname @@end\endcsname\end
⟨/test4⟩
```

4 Installation

4.1 Download

Package. This package is available on CTAN[1]:

CTAN:macros/latex/contrib/magicnum/magicnum.dtx The source file.

CTAN:macros/latex/contrib/magicnum/magicnum.pdf Documentation.

Bundle. All the packages of the bundle 'magicnum' are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

CTAN:install/macros/latex/contrib/magicnum.tds.zip

TDS refers to the standard "A Directory Structure for TEX Files" (CTAN:pkg/tds). Directories with `texmf` in their name are usually organized this way.

4.2 Bundle installation

Unpacking. Unpack the `magicnum.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip magicnum.tds.zip -d ~/texmf
```

Script installation. Check the directory `TDS:scripts/magicnum/` for scripts that need further installation steps.

4.3 Package installation

Unpacking. The `.dtx` file is a self-extracting docstrip archive. The files are extracted by running the `.dtx` through plain TEX:

```
tex magicnum.dtx
```

[1] CTAN:pkg/magicnum

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
magicnum.sty          → tex/generic/magicnum/magicnum.sty
magicnum.lua          → scripts/magicnum/magicnum.lua
magicnum.magicnum.lua → scripts/magicnum/magicnum.magicnum.lua
magicnum.pdf          → doc/latex/magicnum/magicnum.pdf
magicnum.txt          → doc/latex/magicnum/magicnum.txt
magicnum.dtx          → source/latex/magicnum/magicnum.dtx
```

If you have a `docstrip.cfg` that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

4.4 Refresh file name databases

If your TEX distribution (TEX Live, mikTEX, ...) relies on file name databases, you must refresh these. For example, TEX Live users run `texhash` or `mktexlsr`.

4.5 Some details for the interested

Unpacking with LATEX. The `.dtx` chooses its action depending on the format:

plain TEX: Run docstrip and extract the files.

LATEX: Generate the documentation.

If you insist on using LATEX for docstrip (really, docstrip does not need LATEX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{magicnum.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfLATEX:

```
pdflatex magicnum.dtx
makeindex -s gind.ist magicnum.idx
pdflatex magicnum.dtx
makeindex -s gind.ist magicnum.idx
pdflatex magicnum.dtx
```

5 History

[2007/12/12 v1.0]

- First public version.

[2009/04/10 v1.1]

- Adaptation to LuaTEX 0.40.

[2010/03/09 v1.2]

- Adaptation to package luatex 0.4.

[2011/03/24 v1.3]

- Catcode fixes.

[2011/04/10 v1.4]

- Compatibility for iniTEX.
- Dependency from package luatex removed.
- Version check for lua module.

[2016/05/16 v1.5]

- Documentation updates.

[2019/07/25 v1.6]

- remove uses of module function, see PR70

[2019/11/29 v1.7]

- Documentation updates.
- Use iftex directly.

6 Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; plain numbers refer to the code lines where the entry is used.