

The `magicnum` package

Heiko Oberdiek*

2019/11/29 v1.7

Abstract

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

Contents

1 Documentation	2
1.1 Introduction	2
1.2 User interface	2
1.2.1 <code>\magicnum</code>	2
1.2.2 Properties	3
1.3 Data	3
1.3.1 Category <code>tex.catcode</code>	3
1.3.2 Category <code>etex.grouptype</code>	3
1.3.3 Category <code>etex.iftype</code>	4
1.3.4 Category <code>etex.nodetype</code>	4
1.3.5 Category <code>etex.interactionmode</code>	4
1.3.6 Category <code>luatex.pdfliteral.mode</code>	5
2 Implementation	5
2.1 Reload check and package identification	5
2.2 Catcodes	6
2.3 Check for previous definition	7
2.4 Without LuaTeX	7
2.5 With LuaTeX	8
2.6 Data	9
2.6.1 Plain data	9
2.6.2 Data for TeX	10
2.6.3 Lua module	12
3 Test	16
3.1 Catcode checks for loading	16
3.2 Test data	18
3.3 Small test for iniTeX	19
4 Installation	19
4.1 Download	19
4.2 Bundle installation	19
4.3 Package installation	19
4.4 Refresh file name databases	20
4.5 Some details for the interested	20

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

5 History	20
[2007/12/12 v1.0]	20
[2009/04/10 v1.1]	20
[2010/03/09 v1.2]	21
[2011/03/24 v1.3]	21
[2011/04/10 v1.4]	21
[2016/05/16 v1.5]	21
[2019/07/25 v1.6]	21
[2019/11/29 v1.7]	21
6 Index	21

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 \LaTeX is available, they are stored in \LaTeX tables.
- Without \LaTeX 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 chosen 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`

<code>\magicnum {\langle access key \rangle}</code>

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 L^AT_EX.

```
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 % @
```

```

46 \catcode91=12 %
47 \catcode93=12 %
48 \catcode123=1 %
49 \catcode125=2 %
50 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
51   \def\x#1#2#3[#4]{\endgroup
52     \immediate\write-1{Package: #3 #4}%
53     \xdef#1[#4]%
54   }%
55 \else
56   \def\x#1#2[#3]{\endgroup
57     #2[{#3}]%
58     \ifx#1\undefined
59       \xdef#1{#3}%
60     \fi
61     \ifx#1\relax
62       \xdef#1{#3}%
63     \fi
64   }%
65 \fi
66 \expandafter\x\csname ver@magicnum.sty\endcsname
67 \ProvidesPackage{magicnum}%
68 [2019/11/29 v1.7 Magic numbers (HO)]%

```

2.2 Catcodes

```

69 \begingroup\catcode61\catcode48\catcode32=10\relax%
70   \catcode13=5 % ^^M
71   \endlinechar=13 %
72   \catcode123=1 %
73   \catcode125=2 %
74   \catcode64=11 %
75   \def\x{\endgroup
76     \expandafter\edef\csname magicnum@AtEnd\endcsname{%
77       \endlinechar=\the\endlinechar\relax
78       \catcode13=\the\catcode13\relax
79       \catcode32=\the\catcode32\relax
80       \catcode35=\the\catcode35\relax
81       \catcode61=\the\catcode61\relax
82       \catcode64=\the\catcode64\relax
83       \catcode123=\the\catcode123\relax
84       \catcode125=\the\catcode125\relax
85     }%
86   }%
87 \x\catcode61\catcode48\catcode32=10\relax%
88 \catcode13=5 % ^^M
89 \endlinechar=13 %
90 \catcode35=6 % #
91 \catcode64=11 %
92 \catcode123=1 %
93 \catcode125=2 %
94 \def\TMP@EnsureCode#1#2{%
95   \edef\magicnum@AtEnd{%
96     \magicnum@AtEnd
97     \catcode#1=\the\catcode#1\relax
98   }%
99   \catcode#1=#2\relax
100 }

```

```

101 \TMP@EnsureCode{34}{12}%
102 \TMP@EnsureCode{39}{12}%
103 \TMP@EnsureCode{40}{12}%
104 \TMP@EnsureCode{41}{12}%
105 \TMP@EnsureCode{42}{12}%
106 \TMP@EnsureCode{44}{12}%
107 \TMP@EnsureCode{45}{12}%
108 \TMP@EnsureCode{46}{12}%
109 \TMP@EnsureCode{47}{12}%
110 \TMP@EnsureCode{58}{12}%
111 \TMP@EnsureCode{60}{12}%
112 \TMP@EnsureCode{62}{12}%
113 \TMP@EnsureCode{91}{12}%
114 \TMP@EnsureCode{93}{12}%
115 \edef\magicnum@AtEnd{\magicnum@AtEnd\noexpand\endinput}

```

2.3 Check for previous definition

```

116 \begingroup\expandafter\expandafter\expandafter\endgroup
117 \expandafter\ifx\csname newcommand\endcsname\relax
118   \expandafter\ifx\csname magicnum\endcsname\relax
119   \else
120     \input infwareerr.sty\relax
121     \PackageError{magicnum}{%
122       \string\magicnum\space is already defined%
123     }\@ehc
124   \fi
125 \else
126   \newcommand*\{magicnum}{}%
127 \fi

```

2.4 Without LuaTeX

```

128 \begingroup\expandafter\expandafter\expandafter\endgroup
129 \expandafter\ifx\csname directlua\endcsname\relax

```

```

\magicnum
130 \begingroup\expandafter\expandafter\expandafter\endgroup
131 \expandafter\ifx\csname ifcsname\endcsname\relax
132   \def\magicnum#1{%
133     \expandafter\ifx\csname MG@#1\endcsname\relax
134     \else
135       \csname MG@#1\endcsname
136     \fi
137   }%
138 \else
139   \begingroup
140     \edef\x{\endgroup
141       \def\noexpand\magicnum##1{%
142         \expandafter\noexpand\csname
143           ifcsname\endcsname MG@##1\noexpand\endcsname
144           \noexpand\csname MG@##1%
145             \noexpand\expandafter\noexpand\endcsname
146           \expandafter\noexpand\csname fi\endcsname
147         }%
148       }%
149     \x
150   \fi

```

```
151 \else
```

2.5 With LuaTeX

```
152  \begingroup\expandafter\expandafter\expandafter\endgroup
153  \expandafter\ifx\csname RequirePackage\endcsname\relax
154    \input tex.sty\relax
155    \input infwarerr.sty\relax
156  \else
157    \RequirePackage{iftex}[2019/11/07]%
158    \RequirePackage{infwarerr}[2010/04/08]%
159  \fi

\magicnum@directlua
160  \ifnum\luatexversion<36 %
161    \def\magicnum@directlua{\directlua0 }%
162  \else
163    \let\magicnum@directlua\directlua
164  \fi

165  \magicnum@directlua{%
166    require("magicnum")%
167 }%
168  \begingroup
169    \def\x{2019/11/29 v1.7}%
170    \def\StripPrefix#1{%
171      \edef\x{\expandafter\StripPrefix\meaning\x}%
172      \edef\y{%
173        \magicnum@directlua{%
174          if oberdiek.magicnum.getversion then %
175            oberdiek.magicnum.getversion()%
176          end%
177        }%
178      }%
179      \ifx\x\y
180      \else
181        \PackageError{magicnum}{%
182          Wrong version of lua module.\MessageBreak
183          Package version: \x\MessageBreak
184          Lua module: \y
185        }\@ehc
186      \fi
187    \endgroup

\luaescapestring
188  \begingroup
189  \expandafter\ifx\csname luaescapestring\endcsname\relax
190    \directlua{%
191      if tex.enableprimitives then %
192        tex.enableprimitives('magicnum0', {'luaescapestring'})%
193      end%
194    }%
195    \global\let\luaescapestring\magicnum@luaescapestring
196  \fi
197  \expandafter\ifx\csname luaescapestring\endcsname\relax
198    \escapechar=92 %
199    \PackageError{magicnum}{%
200      Missing \string\luaescapestring
201    }\@ehc
202  \fi
203 \endgroup
```

```

\magicnum

204 \def\magicnum#1{%
205   \magicnum@directlua{%
206     oberdiek.magicnum.get("\luaescapestring{#1}")%
207   }%
208 }

209 \expandafter\magicnum@AtEnd
210 \fi%
211 
```

2.6 Data

2.6.1 Plain data

```

212 /*data)
213 tex.catcode
214 escape = 0
215 begingroup = 1
216 endgroup = 2
217 math = 3
218 align = 4
219 eol = 5
220 parameter = 6
221 superscript = 7
222 subscript = 8
223 ignore = 9
224 space = 10
225 letter = 11
226 other = 12
227 active = 13
228 comment = 14
229 invalid = 15
230 etex.grouptype
231 bottomlevel = 0
232 simple = 1
233 hbox = 2
234 adjustedhbox = 3
235 vbox = 4
236 align = 5
237 noalign = 6
238 output = 8
239 math = 9
240 disc = 10
241 insert = 11
242 vcenter = 12
243 mathchoice = 13
244 semisimple = 14
245 mathshift = 15
246 mathleft = 16
247 etex.iftype
248 none = 0
249 char = 1
250 cat = 2
251 num = 3
252 dim = 4
253 odd = 5
254 vmode = 6

```

```

255   hmode = 7
256   mmode = 8
257   inner = 9
258   void = 10
259   hbox = 11
260   vbox = 12
261   x = 13
262   eof = 14
263   true = 15
264   false = 16
265   case = 17
266   defined = 18
267   csname = 19
268   fontchar = 20
269 etex.nodetype
270   none = -1
271   char = 0
272   hlist = 1
273   vlist = 2
274   rule = 3
275   ins = 4
276   mark = 5
277   adjust = 6
278   ligature = 7
279   disc = 8
280   whatsit = 9
281   math = 10
282   glue = 11
283   kern = 12
284   penalty = 13
285   unset = 14
286   maths = 15
287 etex.interactionmode
288   batch = 0
289   nonstop = 1
290   scroll = 2
291   errorstop = 3
292 luatex.pdfliteral.mode
293   setorigin = 0
294   page = 1
295   direct = 2
296 </data>

2.6.2 Data for TEX

297 /*package)

\magicnum@add
298 \begingroup\expandafter\expandafter\expandafter\endgroup
299 \expandafter\ifx\csname detokenize\endcsname\relax
300   \def\magicnum@add#1#2#3{%
301     \expandafter\magicnum@add
302       \csname MG@#1.#2\expandafter\endcsname
303       \csname MG@#1.#3\endcsname
304       {#3}{#2}%
305   }%
306   \def\magicnum@@add#1#2#3#4{%
307     \def#1{#3}%
308     \def#2{#4}%
309     \edef#1{%

```

```

310      \expandafter\strip@prefix\meaning#1%
311  }%
312  \edef#2{%
313      \expandafter\strip@prefix\meaning#2%
314  }%
315 }%
316 \expandafter\ifx\csname strip@prefix\endcsname\relax
317     \def\strip@prefix#1->{}%
318 \fi
319 \else
320 \def\magicnum@add#1#2#3{%
321     \expandafter\edef\csname MG#1.#2\endcsname{%
322         \detokenize{#3}}%
323 }%
324 \expandafter\edef\csname MG#1.#3\endcsname{%
325     \detokenize{#2}}%
326 }%
327 }%
328 \fi

329 \magicnum@add{tex.catcode}{escape}{0}
330 \magicnum@add{tex.catcode}{begingroup}{1}
331 \magicnum@add{tex.catcode}{endgroup}{2}
332 \magicnum@add{tex.catcode}{math}{3}
333 \magicnum@add{tex.catcode}{align}{4}
334 \magicnum@add{tex.catcode}{eol}{5}
335 \magicnum@add{tex.catcode}{parameter}{6}
336 \magicnum@add{tex.catcode}{superscript}{7}
337 \magicnum@add{tex.catcode}{subscript}{8}
338 \magicnum@add{tex.catcode}{ignore}{9}
339 \magicnum@add{tex.catcode}{space}{10}
340 \magicnum@add{tex.catcode}{letter}{11}
341 \magicnum@add{tex.catcode}{other}{12}
342 \magicnum@add{tex.catcode}{active}{13}
343 \magicnum@add{tex.catcode}{comment}{14}
344 \magicnum@add{tex.catcode}{invalid}{15}
345 \magicnum@add{etex.grouptype}{bottomlevel}{0}
346 \magicnum@add{etex.grouptype}{simple}{1}
347 \magicnum@add{etex.grouptype}{hbox}{2}
348 \magicnum@add{etex.grouptype}{adjustedhbox}{3}
349 \magicnum@add{etex.grouptype}{vbox}{4}
350 \magicnum@add{etex.grouptype}{align}{5}
351 \magicnum@add{etex.grouptype}{noalign}{6}
352 \magicnum@add{etex.grouptype}{output}{8}
353 \magicnum@add{etex.grouptype}{math}{9}
354 \magicnum@add{etex.grouptype}{disc}{10}
355 \magicnum@add{etex.grouptype}{insert}{11}
356 \magicnum@add{etex.grouptype}{vcenter}{12}
357 \magicnum@add{etex.grouptype}{mathchoice}{13}
358 \magicnum@add{etex.grouptype}{semisimple}{14}
359 \magicnum@add{etex.grouptype}{mathshift}{15}
360 \magicnum@add{etex.grouptype}{mathleft}{16}
361 \magicnum@add{etex.iftype}{none}{0}
362 \magicnum@add{etex.iftype}{char}{1}
363 \magicnum@add{etex.iftype}{cat}{2}
364 \magicnum@add{etex.iftype}{num}{3}
365 \magicnum@add{etex.iftype}{dim}{4}
366 \magicnum@add{etex.iftype}{odd}{5}
367 \magicnum@add{etex.iftype}{vmode}{6}

```

```

368 \magicnum@add{etex.iftype}{hmode}{7}
369 \magicnum@add{etex.iftype}{mmode}{8}
370 \magicnum@add{etex.iftype}{inner}{9}
371 \magicnum@add{etex.iftype}{void}{10}
372 \magicnum@add{etex.iftype}{hbox}{11}
373 \magicnum@add{etex.iftype}{vbox}{12}
374 \magicnum@add{etex.iftype}{x}{13}
375 \magicnum@add{etex.iftype}{eof}{14}
376 \magicnum@add{etex.iftype}{true}{15}
377 \magicnum@add{etex.iftype}{false}{16}
378 \magicnum@add{etex.iftype}{case}{17}
379 \magicnum@add{etex.iftype}{defined}{18}
380 \magicnum@add{etex.iftype}{csname}{19}
381 \magicnum@add{etex.iftype}{fontchar}{20}
382 \magicnum@add{etex.nodetype}{none}{-1}
383 \magicnum@add{etex.nodetype}{char}{0}
384 \magicnum@add{etex.nodetype}{hlist}{1}
385 \magicnum@add{etex.nodetype}{vlist}{2}
386 \magicnum@add{etex.nodetype}{rule}{3}
387 \magicnum@add{etex.nodetype}{ins}{4}
388 \magicnum@add{etex.nodetype}{mark}{5}
389 \magicnum@add{etex.nodetype}{adjust}{6}
390 \magicnum@add{etex.nodetype}{ligature}{7}
391 \magicnum@add{etex.nodetype}{disc}{8}
392 \magicnum@add{etex.nodetype}{whatsit}{9}
393 \magicnum@add{etex.nodetype}{math}{10}
394 \magicnum@add{etex.nodetype}{glue}{11}
395 \magicnum@add{etex.nodetype}{kern}{12}
396 \magicnum@add{etex.nodetype}{penalty}{13}
397 \magicnum@add{etex.nodetype}{unset}{14}
398 \magicnum@add{etex.nodetype}{maths}{15}
399 \magicnum@add{etex.interactionmode}{batch}{0}
400 \magicnum@add{etex.interactionmode}{nonstop}{1}
401 \magicnum@add{etex.interactionmode}{scroll}{2}
402 \magicnum@add{etex.interactionmode}{errorstop}{3}
403 \magicnum@add{luatex.pdfliteral.mode}{setorigin}{0}
404 \magicnum@add{luatex.pdfliteral.mode}{page}{1}
405 \magicnum@add{luatex.pdfliteral.mode}{direct}{2}

406 \magicnum@AtEnd%
407 </package>

```

2.6.3 Lua module

```

408 /*lua)
409 oberdiek = oberdiek or {}
410 local magicnum = oberdiek.magicnum or {}
411 oberdiek.magicnum = magicnum
412 function magicnum.getversion()
413   tex.write("2019/11/29 v1.7")
414 end
415 local data = {
416   ["tex.catcode"] = {
417     [0] = "escape",
418     [1] = "begingroup",
419     [2] = "endgroup",
420     [3] = "math",
421     [4] = "align",

```

```

422     [5] = "eol",
423     [6] = "parameter",
424     [7] = "superscript",
425     [8] = "subscript",
426     [9] = "ignore",
427     [10] = "space",
428     [11] = "letter",
429     [12] = "other",
430     [13] = "active",
431     [14] = "comment",
432     [15] = "invalid",
433     ["active"] = 13,
434     ["align"] = 4,
435     ["begingroup"] = 1,
436     ["comment"] = 14,
437     ["endgroup"] = 2,
438     ["eol"] = 5,
439     ["escape"] = 0,
440     ["ignore"] = 9,
441     ["invalid"] = 15,
442     ["letter"] = 11,
443     ["math"] = 3,
444     ["other"] = 12,
445     ["parameter"] = 6,
446     ["space"] = 10,
447     ["subscript"] = 8,
448     ["superscript"] = 7
449 },
450 ["etex.grouptype"] = {
451     [0] = "bottomlevel",
452     [1] = "simple",
453     [2] = "hbox",
454     [3] = "adjustedhbox",
455     [4] = "vbox",
456     [5] = "align",
457     [6] = "noalign",
458     [8] = "output",
459     [9] = "math",
460     [10] = "disc",
461     [11] = "insert",
462     [12] = "vcenter",
463     [13] = "mathchoice",
464     [14] = "semisimple",
465     [15] = "mathshift",
466     [16] = "mathleft",
467     ["adjustedhbox"] = 3,
468     ["align"] = 5,
469     ["bottomlevel"] = 0,
470     ["disc"] = 10,
471     ["hbox"] = 2,
472     ["insert"] = 11,
473     ["math"] = 9,
474     ["mathchoice"] = 13,
475     ["mathleft"] = 16,
476     ["mathshift"] = 15,
477     ["noalign"] = 6,
478     ["output"] = 8,
479     ["semisimple"] = 14,

```

```

480     ["simple"] = 1,
481     ["vbox"] = 4,
482     ["vcenter"] = 12
483   },
484   ["etex.itype"] = {
485     [0] = "none",
486     [1] = "char",
487     [2] = "cat",
488     [3] = "num",
489     [4] = "dim",
490     [5] = "odd",
491     [6] = "vmode",
492     [7] = "hmode",
493     [8] = "mmode",
494     [9] = "inner",
495     [10] = "void",
496     [11] = "hbox",
497     [12] = "vbox",
498     [13] = "x",
499     [14] = "eof",
500     [15] = "true",
501     [16] = "false",
502     [17] = "case",
503     [18] = "defined",
504     [19] = "csname",
505     [20] = "fontchar",
506     ["case"] = 17,
507     ["cat"] = 2,
508     ["char"] = 1,
509     ["csname"] = 19,
510     ["defined"] = 18,
511     ["dim"] = 4,
512     ["eof"] = 14,
513     ["false"] = 16,
514     ["fontchar"] = 20,
515     ["hbox"] = 11,
516     ["hmode"] = 7,
517     ["inner"] = 9,
518     ["mmode"] = 8,
519     ["none"] = 0,
520     ["num"] = 3,
521     ["odd"] = 5,
522     ["true"] = 15,
523     ["vbox"] = 12,
524     ["vmode"] = 6,
525     ["void"] = 10,
526     ["x"] = 13
527   },
528   ["etex.nodetype"] = {
529     [-1] = "none",
530     [0] = "char",
531     [1] = "hlist",
532     [2] = "vlist",
533     [3] = "rule",
534     [4] = "ins",
535     [5] = "mark",
536     [6] = "adjust",
537     [7] = "ligature",

```

```

538     [8] = "disc",
539     [9] = "whatsit",
540     [10] = "math",
541     [11] = "glue",
542     [12] = "kern",
543     [13] = "penalty",
544     [14] = "unset",
545     [15] = "maths",
546     ["adjust"] = 6,
547     ["char"] = 0,
548     ["disc"] = 8,
549     ["glue"] = 11,
550     ["hlist"] = 1,
551     ["ins"] = 4,
552     ["kern"] = 12,
553     ["ligature"] = 7,
554     ["mark"] = 5,
555     ["math"] = 10,
556     ["maths"] = 15,
557     ["none"] = -1,
558     ["penalty"] = 13,
559     ["rule"] = 3,
560     ["unset"] = 14,
561     ["vlist"] = 2,
562     ["whatsit"] = 9
563   },
564   ["etex.interactionmode"] = {
565     [0] = "batch",
566     [1] = "nonstop",
567     [2] = "scroll",
568     [3] = "errorstop",
569     ["batch"] = 0,
570     ["errorstop"] = 3,
571     ["nonstop"] = 1,
572     ["scroll"] = 2
573   },
574   ["lualatex.pdfliteral.mode"] = {
575     [0] = "setorigin",
576     [1] = "page",
577     [2] = "direct",
578     ["direct"] = 2,
579     ["page"] = 1,
580     ["setorigin"] = 0
581   }
582 }

583 function magicnum.get(name)
584   local startpos, endpos, category, entry =
585     string.find(name, "^(%a[%a%d%..]*%)%.(-?[%a%d]+)$")
586   if not entry then
587     return
588   end
589   local node = data[category]
590   if not node then
591     return
592   end
593   local num = tonumber(entry)
594   local value
595   if num then

```

```

596     value = node[num]
597     if not value then
598         return
599     end
600 else
601     value = node[entry]
602     if not value then
603         return
604     end
605     value = "" .. value
606 end
607 tex.write(value)
608 end
609 </lua>

```

3 Test

3.1 Catcode checks for loading

```

610 <*test1>
611 \catcode`{\=1 %
612 \catcode`}=2 %
613 \catcode`\#=6 %
614 \catcode`\@=11 %
615 \expandafter\ifx\csname count@\endcsname\relax
616   \countdef\count@=255 %
617 \fi
618 \expandafter\ifx\csname @gobble\endcsname\relax
619   \long\def@\gobble#1{}%
620 \fi
621 \expandafter\ifx\csname @firstofone\endcsname\relax
622   \long\def@\firstofone#1{#1}%
623 \fi
624 \expandafter\ifx\csname loop\endcsname\relax
625   \expandafter@\firstofone
626 \else
627   \expandafter@\gobble
628 \fi
629 {%
630   \def\loop#1\repeat{%
631     \def\body{#1}%
632     \iterate
633   }%
634   \def\iterate{%
635     \body
636     \let\next\iterate
637   \else
638     \let\next\relax
639   \fi
640   \next
641 }%
642   \let\repeat=\fi
643 }%
644 \def\RestoreCatcodes{}
645 \count@=0 %
646 \loop
647   \edef\RestoreCatcodes{%
648     \RestoreCatcodes

```

```

649      \catcode\the\count@=\the\catcode\count@\relax
650  }%
651 \ifnum\count@<255 %
652   \advance\count@ 1 %
653 \repeat
654
655 \def\RangeCatcodeInvalid#1#2{%
656   \count@=#1\relax
657   \loop
658     \catcode\count@=15 %
659   \ifnum\count@<#2\relax
660     \advance\count@ 1 %
661   \repeat
662 }
663 \def\RangeCatcodeCheck#1#2#3{%
664   \count@=#1\relax
665   \loop
666     \ifnum#3=\catcode\count@
667     \else
668       \errmessage{%
669         Character \the\count@\space
670         with wrong catcode \the\catcode\count@\space
671         instead of \number#3%
672       }%
673     \fi
674   \ifnum\count@<#2\relax
675     \advance\count@ 1 %
676   \repeat
677 }
678 \def\space{ }
679 \expandafter\ifx\csname LoadCommand\endcsname\relax
680   \def\LoadCommand{\input magicnum.sty\relax}%
681 \fi
682 \def\Test{%
683   \RangeCatcodeInvalid{0}{47}%
684   \RangeCatcodeInvalid{58}{64}%
685   \RangeCatcodeInvalid{91}{96}%
686   \RangeCatcodeInvalid{123}{255}%
687   \catcode`@=12 %
688   \catcode`\\=0 %
689   \catcode`\%=14 %
690   \LoadCommand
691   \RangeCatcodeCheck{0}{36}{15}%
692   \RangeCatcodeCheck{37}{37}{14}%
693   \RangeCatcodeCheck{38}{47}{15}%
694   \RangeCatcodeCheck{48}{57}{12}%
695   \RangeCatcodeCheck{58}{63}{15}%
696   \RangeCatcodeCheck{64}{64}{12}%
697   \RangeCatcodeCheck{65}{90}{11}%
698   \RangeCatcodeCheck{91}{91}{15}%
699   \RangeCatcodeCheck{92}{92}{0}%
700   \RangeCatcodeCheck{93}{96}{15}%
701   \RangeCatcodeCheck{97}{122}{11}%
702   \RangeCatcodeCheck{123}{255}{15}%
703   \RestoreCatcodes
704 }
705 \Test
706 \csname @@end\endcsname

```

```

707 \end
708 </test1>

```

3.2 Test data

```

709 (*testplain)
710 \input magicnum.sty\relax
711 \def\Test#1#2{%
712   \edef\result{\magicnum{#1}}%
713   \edef\expect{#2}%
714   \edef\expect{\expandafter\stripprefix\meaning\expect}%
715   \ifx\result\expect
716   \else
717     \errmessage{%
718       Failed: [#1] % hash-ok
719       returns [\result] instead of [\expect]%
720     }%
721   \fi
722 }
723 \def\stripprefix#1->{}
724 </testplain>

725 (*testlatex)
726 \NeedsTeXFormat{LaTeX2e}
727 \documentclass{minimal}
728 \usepackage[magicnum][2016/05/16]
729 \usepackage[qstest]
730 \IncludeTests{*}
731 \LogTests{log}{*}{*}
732 \newcommand*\Test[2]{%
733   \Expect*{\magicnum{#1}}{#2}%
734 }
735 \begin{qstest}[magicnum]{magicnum}
736 </testlatex>

737 (*testdata)
738 \Test{tex.catcode.escape}{0}
739 \Test{tex.catcode.invalid}{15}
740 \Test{tex.catcode.unknown}{}
741 \Test{tex.catcode.0}{escape}
742 \Test{tex.catcode.15}{invalid}
743 \Test{etex.iftype.true}{15}
744 \Test{etex.iftype.false}{16}
745 \Test{etex.iftype.15}{true}
746 \Test{etex.iftype.16}{false}
747 \Test{etex.nodetype.none}{-1}
748 \Test{etex.nodetype.-1}{none}
749 \Test{lualatex.pdfliteral.mode.direct}{2}
750 \Test{lualatex.pdfliteral.mode.1}{page}
751 \Test{}{}
752 \Test{unknown}{}
753 \Test{unknown.foo.bar}{}
754 \Test{unknown.foo.4}{}
755 </testdata>

756 (*testplain)
757 \csname @@end\endcsname
758 \end
759 </testplain>
760 (*testlatex)

```

```

761 \end{qstest}
762 \csname @@end\endcsname
763 
```

3.3 Small test for iniTeX

```

764 /*test4)
765 \catcode`\#=1
766 \catcode`\#=2
767 \catcode`\#=6
768 \input magicnum.sty\relax
769 \edef\x{\magicnum{tex.catcode.15}}
770 \edef\y{invalid}
771 \def\Strip#1>{}
772 \edef\y{\expandafter\Strip\meaning\y}
773 \ifx\x\y
774   \immediate\write16{0k}%
775 \else
776   \errmessage{\x<>\y}%
777 \fi
778 \csname @@end\endcsname\end
779 
```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

[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 L^AT_EX. The `.dtx` chooses its action depending on the format:

plain `TeX`: Run `docstrip` and extract the files.

L^AT_EX: Generate the documentation.

If you insist on using L^AT_EX for `docstrip` (really, `docstrip` does not need L^AT_EX), 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 pdfL^AT_EX:

```
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 Lua`TeX` 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.

Symbols	C
\#	613, 767
\%	689
\@	614, 687
\@PackageError	121, 181, 199
\@ehc	123, 185, 201
\@firstofone	622, 625
\@gobble	619, 627
\@undefined	58
\\"	688
\{	611, 765
\}	612, 766
A	
\advance	652, 660, 675
\aftergroup	29
B	
\begin	735
\body	631, 635
C	
\catcode	2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 33, 34, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 69, 70, 72, 73, 74, 78, 79, 80, 81, 82, 83, 84, 87, 88, 90, 91, 92, 93, 97, 99, 611, 612, 613, 614, 649, 658, 666, 670, 687, 688, 689, 765, 766, 767
\count@	616, 645,
\countdef	649, 651, 652, 656, 658, 659, 660, 664, 666, 669, 670, 674, 675
\countdef	616
\csname	14, 21, 50, 66, 76, 117, 118, 129, 131, 133, 135, 142, 144, 146, 153, 189, 197, 299, 302, 303, 316, 321, 324, 615, 618, 621, 624, 679, 706, 757, 762, 778
D	
\detokenize	322, 325

\directlua	161, 163, 190	\magicnum@luaescapestring	195
\documentclass	727	\meaning	171, 310, 313, 714, 772
E			
\empty	17, 18	\MessageBreak	182, 183
\end	707, 758, 761, 778	N	
\endcsname	14, 21, 50, 66, 76, 117, 118, 129, 131, 133, 135, 143, 145, 146, 153, 189, 197, 299, 302, 303, 316, 321, 324, 615, 618, 621, 624, 679, 706, 757, 762, 778	\NeedsTeXFormat	726
\endinput	29, 115	\newcommand	126, 732
\newlinechar	4, 35, 71, 77, 89	\next	636, 638, 640
\errmessage	668, 717, 776	\number	671
\escapechar	198	P	
\Expect	733	\PackageInfo	26
\expect	713, 714, 715, 719	\ProvidesPackage	19, 67
I			
\ifnum	160, 651, 659, 666, 674	\RangeCatcodeCheck	663, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702
\ifx	15, 18, 21, 50, 58, 61, 117, 118, 129, 131, 133, 153, 179, 189, 197, 299, 316, 615, 618, 621, 624, 679, 715, 773	\RangeCatcodeInvalid	655, 683, 684, 685, 686
\immediate	23, 52, 774	\repeat	630, 642, 653, 661, 676
\IncludeTests	730	\RequirePackage	157, 158
\input	120, 154, 155, 680, 710, 768	\RestoreCatcodes	644, 647, 648, 703
\iterate	632, 634, 636	\result	712, 715, 719
L			
\LoadCommand	680, 690	S	
\LogTests	731	\space	122, 669, 670, 678
\loop	630, 646, 657, 665	\Strip	771, 772
\luaescapestring	188, 206	\strip@prefix	310, 313, 317
\luatexversion	160	\StripPrefix	170, 171
M			
\magicnum	2, 122, 126, 130, 204, 712, 733, 769	\stripprefix	714, 723
\magicnum@add	301, 306	T	
\magicnum@add	298, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405	\Test	682, 705, 711, 732, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754
\magicnum@AtEnd	95, 96, 115, 209, 406	\the	77, 78, 79, 80, 81, 82, 83, 84, 97, 649, 669, 670
\magicnum@directlua	160, 165, 173, 205	\TMP@EnsureCode	94, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114
U			
\usepackage	728, 729	W	
\write	23, 52, 774	X	
\x	14, 15, 18, 22, 26, 28, 51, 56, 66, 75, 87, 140, 149, 169, 171, 179, 183, 769, 773, 776	Y	
\y	172, 179, 184, 770, 772, 773, 776		