

# The `magicnum` package

Heiko Oberdiek  
<oberdiek@uni-freiburg.de>

2007/12/12 v1.0

## Abstract

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

## Contents

<b>1</b>	<b>Documentation</b>	<b>2</b>
1.1	Introduction	2
1.2	User interface	2
1.2.1	<code>\magicnum</code>	2
1.2.2	Properties	2
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.ifttype</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>	4
<b>2</b>	<b>Implementation</b>	<b>4</b>
2.1	Reload check and package identification	5
2.2	Catcodes	6
2.3	Check for previous definition	6
2.4	Without <code>LUAT<sub>E</sub>X</code>	6
2.5	With <code>LUAT<sub>E</sub>X</code>	7
2.6	Data	7
2.6.1	Plain data	7
2.6.2	Data for <code>T<sub>E</sub>X</code>	9
2.6.3	Lua module	11
<b>3</b>	<b>Test</b>	<b>14</b>
3.1	Catcode checks for loading	14
3.2	Test data	15
<b>4</b>	<b>Installation</b>	<b>16</b>
4.1	Download	16
4.2	Bundle installation	16
4.3	Package installation	17
4.4	Refresh file name databases	17
4.5	Some details for the interested	17
<b>5</b>	<b>History</b>	<b>18</b>
	[2007/12/12 v1.0]	18
<b>6</b>	<b>Index</b>	<b>18</b>

# 1 Documentation

## 1.1 Introduction

Especially since  $\varepsilon$ -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 L<sup>A</sup>T<sub>E</sub>X is available, they are stored in Lua tables.
- Without L<sup>A</sup>T<sub>E</sub>X 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:  $\varepsilon$ -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 T<sub>E</sub>X'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`

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

### 1.3.2 Category `etex.grouptype`

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

### 1.3.3 Category etex.ifttype

etex.ifttype.none	0
etex.ifttype.char	1
etex.ifttype.cat	2
etex.ifttype.num	3
etex.ifttype.dim	4
etex.ifttype.odd	5
etex.ifttype.vmode	6
etex.ifttype.hmode	7
etex.ifttype.mmode	8
etex.ifttype.inner	9
etex.ifttype.void	10
etex.ifttype.hbox	11
etex.ifttype.vbox	12
etex.ifttype.x	13
etex.ifttype.eof	14
etex.ifttype.true	15
etex.ifttype.false	16
etex.ifttype.case	17
etex.ifttype.defined	18
etex.ifttype.csname	19
etex.ifttype.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<sup>A</sup>T<sub>E</sub>X.

```
2 \begingroup
3 \catcode44 12 % ,
4 \catcode45 12 % -
5 \catcode46 12 % .
6 \catcode58 12 % :
7 \catcode64 11 % @
8 \expandafter\let\expandafter\x\csname ver@magicnum.sty\endcsname
9 \ifcase 0%
10 \ifx\x\relax % plain
11 \else
12 \ifx\x\empty % LaTeX
13 \else
14 1%
15 \fi
16 \fi
17 \else
18 \catcode35 6 % #
19 \catcode123 1 % {
20 \catcode125 2 % }
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 \endgroup
30 \expandafter\endinput
31 \fi
32 \endgroup
```

Package identification:

```
33 \begingroup
34 \catcode35 6 % #
35 \catcode40 12 % (
36 \catcode41 12 % )
37 \catcode44 12 % ,
38 \catcode45 12 % -
39 \catcode46 12 % .
40 \catcode47 12 % /
41 \catcode58 12 % :
42 \catcode64 11 % @
43 \catcode123 1 % {
44 \catcode125 2 % }
45 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
46 \def\x#1#2#3[#4]{\endgroup
47 \immediate\write-1{Package: #3 #4}%
48 \xdef#1{#4}%
49 }%
50 \else
51 \def\x#1#2[#3]{\endgroup
52 #2[#3]}%
53 \ifx#1\relax
54 \xdef#1{#3}%
55 \fi
56 }%
57 \fi
```

```

58 \expandafter\x\csname ver@magicnum.sty\endcsname
59 \ProvidesPackage{magicnum}%
60 [2007/12/12 v1.0 Magic numbers (H0)]

```

## 2.2 Catcodes

```

61 \begingroup
62 \catcode123 1 % {
63 \catcode125 2 % }
64 \def\x{\endgroup
65 \expandafter\edef\csname magicnum@AtEnd\endcsname{%
66 \catcode35 \the\catcode35\relax
67 \catcode64 \the\catcode64\relax
68 \catcode123 \the\catcode123\relax
69 \catcode125 \the\catcode125\relax
70 }%
71 }%
72 \x
73 \catcode35 6 % #
74 \catcode64 11 % @
75 \catcode123 1 % {
76 \catcode125 2 % }
77 \def\TMP@EnsureCode#1#2{%
78 \edef\magicnum@AtEnd{%
79 \magicnum@AtEnd
80 \catcode#1 \the\catcode#1\relax
81 }%
82 \catcode#1 #2\relax
83 }
84 \TMP@EnsureCode{34}{12}% "
85 \TMP@EnsureCode{40}{12}% (
86 \TMP@EnsureCode{41}{12}% )
87 \TMP@EnsureCode{42}{12}% *
88 \TMP@EnsureCode{44}{12}% ,
89 \TMP@EnsureCode{45}{12}% -
90 \TMP@EnsureCode{46}{12}% .
91 \TMP@EnsureCode{47}{12}% /
92 \TMP@EnsureCode{62}{12}% >

```

## 2.3 Check for previous definition

```

93 \begingroup\expandafter\expandafter\expandafter\endgroup
94 \expandafter\ifx\csname newcommand\endcsname\relax
95 \expandafter\ifx\csname magicnum\endcsname\relax
96 \else
97 \input infwarerr.sty\relax
98 \@PackageErrorNoLine{magicnum}{%
99 \string\magicnum\space is already defined%
100 }\@ehc
101 \fi
102 \else
103 \newcommand*{\magicnum}{}%
104 \fi

```

## 2.4 Without LuaTeX

```

105 \begingroup\expandafter\expandafter\expandafter\endgroup
106 \expandafter\ifx\csname luatexversion\endcsname\relax

\magicnum

107 \begingroup\expandafter\expandafter\expandafter\endgroup
108 \expandafter\ifx\csname ifcsname\endcsname\relax
109 \def\magicnum#1{%
110 \expandafter\ifx\csname MG@#1\endcsname\relax

```

```

111     \else
112     \csname MG@#1\endcsname
113     \fi
114   }%
115 \else
116   \begingroup
117   \edef\x{\endgroup
118     \def\noexpand\magicnum##1{%
119       \expandafter\noexpand\csname
120         ifcsname\endcsname MG@##1\noexpand\endcsname
121         \noexpand\csname MG@##1%
122           \noexpand\expandafter\noexpand\endcsname
123         \expandafter\noexpand\csname fi\endcsname
124       }%
125     }%
126   \x
127 \fi

128 \else

```

## 2.5 With LuaTeX

```

129 \begingroup\expandafter\expandafter\expandafter\endgroup
130 \expandafter\ifx\csname RequirePackage\endcsname\relax
131   \input luatex.sty\relax
132 \else
133   \RequirePackage{luatex}[2007/12/12]%
134 \fi

```

\magicnum

```

135 \def\magicnum#1{%
136   \directlua0{%
137     require("oberdiek.magicnum")%
138     oberdiek.magicnum.get(%
139       "\luaescapestring{#1}",\number\CatcodeTableString
140     )%
141   }%
142 }%

143 \magicnum@AtEnd
144 \expandafter\endinput
145 \fi
146 \</package>

```

## 2.6 Data

### 2.6.1 Plain data

```

147 \<data>
148 tex.catcode
149   escape = 0
150   begingroup = 1
151   endgroup = 2
152   math = 3
153   align = 4
154   eol = 5
155   parameter = 6
156   superscript = 7
157   subscript = 8
158   ignore = 9
159   space = 10
160   letter = 11
161   other = 12

```

```

162 active = 13
163 comment = 14
164 invalid = 15
165 etex.grouptype
166 bottomlevel = 0
167 simple = 1
168 hbox = 2
169 adjustedhbox = 3
170 vbox = 4
171 align = 5
172 noalign = 6
173 output = 8
174 math = 9
175 disc = 10
176 insert = 11
177 vcenter = 12
178 mathchoice = 13
179 semisimple = 14
180 mathshift = 15
181 mathleft = 16
182 etex.ifttype
183 none = 0
184 char = 1
185 cat = 2
186 num = 3
187 dim = 4
188 odd = 5
189 vmode = 6
190 hmode = 7
191 mmode = 8
192 inner = 9
193 void = 10
194 hbox = 11
195 vbox = 12
196 x = 13
197 eof = 14
198 true = 15
199 false = 16
200 case = 17
201 defined = 18
202 csname = 19
203 fontchar = 20
204 etex.nodetype
205 none = -1
206 char = 0
207 hlist = 1
208 vlist = 2
209 rule = 3
210 ins = 4
211 mark = 5
212 adjust = 6
213 ligature = 7
214 disc = 8
215 whatsit = 9
216 math = 10
217 glue = 11
218 kern = 12
219 penalty = 13
220 unset = 14
221 maths = 15
222 etex.interactionmode
223 batch = 0

```



```

224 nonstop = 1
225 scroll = 2
226 errorstop = 3
227 luatex.pdfliteral.mode
228 setorigin = 0
229 page = 1
230 direct = 2
231 </data>

```

## 2.6.2 Data for T<sub>E</sub>X

```

232 (*package)

\magicnum@add

233 \begingroup\expandafter\expandafter\expandafter\endgroup
234 \expandafter\ifx\csname detokenize\endcsname\relax
235   \def\magicnum@add#1#2#3{%
236     \expandafter\magicnum@@add
237       \csname MG@#1.#2\expandafter\endcsname
238       \csname MG@#1.#3\endcsname
239     {#3}{#2}%
240   }%
241   \def\magicnum@@add#1#2#3#4{%
242     \def#1{#3}%
243     \def#2{#4}%
244     \edef#1{%
245       \expandafter\strip@prefix\meaning#1%
246     }%
247     \edef#2{%
248       \expandafter\strip@prefix\meaning#2%
249     }%
250   }%
251   \expandafter\ifx\csname strip@prefix\endcsname\relax
252     \def\strip@prefix#1->{}%
253   \fi
254 \else
255   \def\magicnum@add#1#2#3{%
256     \expandafter\edef\csname MG@#1.#2\endcsname{%
257       \detokenize{#3}%
258     }%
259     \expandafter\edef\csname MG@#1.#3\endcsname{%
260       \detokenize{#2}%
261     }%
262   }%
263 \fi

264 \magicnum@add{tex.catcode}{escape}{0}
265 \magicnum@add{tex.catcode}{begingroup}{1}
266 \magicnum@add{tex.catcode}{endgroup}{2}
267 \magicnum@add{tex.catcode}{math}{3}
268 \magicnum@add{tex.catcode}{align}{4}
269 \magicnum@add{tex.catcode}{eol}{5}
270 \magicnum@add{tex.catcode}{parameter}{6}
271 \magicnum@add{tex.catcode}{superscript}{7}
272 \magicnum@add{tex.catcode}{subscript}{8}
273 \magicnum@add{tex.catcode}{ignore}{9}
274 \magicnum@add{tex.catcode}{space}{10}
275 \magicnum@add{tex.catcode}{letter}{11}
276 \magicnum@add{tex.catcode}{other}{12}
277 \magicnum@add{tex.catcode}{active}{13}
278 \magicnum@add{tex.catcode}{comment}{14}
279 \magicnum@add{tex.catcode}{invalid}{15}
280 \magicnum@add{etex.grouptype}{bottomlevel}{0}
281 \magicnum@add{etex.grouptype}{simple}{1}

```

```

282 \magicnum@add{etex.grouptype}{hbox}{2}
283 \magicnum@add{etex.grouptype}{adjustedhbox}{3}
284 \magicnum@add{etex.grouptype}{vbox}{4}
285 \magicnum@add{etex.grouptype}{align}{5}
286 \magicnum@add{etex.grouptype}{noalign}{6}
287 \magicnum@add{etex.grouptype}{output}{8}
288 \magicnum@add{etex.grouptype}{math}{9}
289 \magicnum@add{etex.grouptype}{disc}{10}
290 \magicnum@add{etex.grouptype}{insert}{11}
291 \magicnum@add{etex.grouptype}{vcenter}{12}
292 \magicnum@add{etex.grouptype}{mathchoice}{13}
293 \magicnum@add{etex.grouptype}{semisimple}{14}
294 \magicnum@add{etex.grouptype}{mathshift}{15}
295 \magicnum@add{etex.grouptype}{mathleft}{16}
296 \magicnum@add{etex.ifttype}{none}{0}
297 \magicnum@add{etex.ifttype}{char}{1}
298 \magicnum@add{etex.ifttype}{cat}{2}
299 \magicnum@add{etex.ifttype}{num}{3}
300 \magicnum@add{etex.ifttype}{dim}{4}
301 \magicnum@add{etex.ifttype}{odd}{5}
302 \magicnum@add{etex.ifttype}{vmode}{6}
303 \magicnum@add{etex.ifttype}{hmode}{7}
304 \magicnum@add{etex.ifttype}{mmode}{8}
305 \magicnum@add{etex.ifttype}{inner}{9}
306 \magicnum@add{etex.ifttype}{void}{10}
307 \magicnum@add{etex.ifttype}{hbox}{11}
308 \magicnum@add{etex.ifttype}{vbox}{12}
309 \magicnum@add{etex.ifttype}{x}{13}
310 \magicnum@add{etex.ifttype}{eof}{14}
311 \magicnum@add{etex.ifttype}{true}{15}
312 \magicnum@add{etex.ifttype}{false}{16}
313 \magicnum@add{etex.ifttype}{case}{17}
314 \magicnum@add{etex.ifttype}{defined}{18}
315 \magicnum@add{etex.ifttype}{csname}{19}
316 \magicnum@add{etex.ifttype}{fontchar}{20}
317 \magicnum@add{etex.nodetype}{none}{-1}
318 \magicnum@add{etex.nodetype}{char}{0}
319 \magicnum@add{etex.nodetype}{hlist}{1}
320 \magicnum@add{etex.nodetype}{vlist}{2}
321 \magicnum@add{etex.nodetype}{rule}{3}
322 \magicnum@add{etex.nodetype}{ins}{4}
323 \magicnum@add{etex.nodetype}{mark}{5}
324 \magicnum@add{etex.nodetype}{adjust}{6}
325 \magicnum@add{etex.nodetype}{ligature}{7}
326 \magicnum@add{etex.nodetype}{disc}{8}
327 \magicnum@add{etex.nodetype}{whatsit}{9}
328 \magicnum@add{etex.nodetype}{math}{10}
329 \magicnum@add{etex.nodetype}{glue}{11}
330 \magicnum@add{etex.nodetype}{kern}{12}
331 \magicnum@add{etex.nodetype}{penalty}{13}
332 \magicnum@add{etex.nodetype}{unset}{14}
333 \magicnum@add{etex.nodetype}{maths}{15}
334 \magicnum@add{etex.interactionmode}{batch}{0}
335 \magicnum@add{etex.interactionmode}{nonstop}{1}
336 \magicnum@add{etex.interactionmode}{scroll}{2}
337 \magicnum@add{etex.interactionmode}{errorstop}{3}
338 \magicnum@add{luatex.pdfliteral.mode}{setorigin}{0}
339 \magicnum@add{luatex.pdfliteral.mode}{page}{1}
340 \magicnum@add{luatex.pdfliteral.mode}{direct}{2}

341 \magicnum@AtEnd
342 \endpackage

```

### 2.6.3 Lua module

```
343  $\langle$ *lua $\rangle$ 
344 module("oberdiek.magicnum", package.seeall)
345 local data = {
346   ["tex.catcode"] = {
347     [0] = "escape",
348     [1] = "begingroup",
349     [2] = "endgroup",
350     [3] = "math",
351     [4] = "align",
352     [5] = "eol",
353     [6] = "parameter",
354     [7] = "superscript",
355     [8] = "subscript",
356     [9] = "ignore",
357     [10] = "space",
358     [11] = "letter",
359     [12] = "other",
360     [13] = "active",
361     [14] = "comment",
362     [15] = "invalid",
363     ["active"] = 13,
364     ["align"] = 4,
365     ["begingroup"] = 1,
366     ["comment"] = 14,
367     ["endgroup"] = 2,
368     ["eol"] = 5,
369     ["escape"] = 0,
370     ["ignore"] = 9,
371     ["invalid"] = 15,
372     ["letter"] = 11,
373     ["math"] = 3,
374     ["other"] = 12,
375     ["parameter"] = 6,
376     ["space"] = 10,
377     ["subscript"] = 8,
378     ["superscript"] = 7
379   },
380   ["etex.grouptype"] = {
381     [0] = "bottomlevel",
382     [1] = "simple",
383     [2] = "hbox",
384     [3] = "adjustedhbox",
385     [4] = "vbox",
386     [5] = "align",
387     [6] = "noalign",
388     [8] = "output",
389     [9] = "math",
390     [10] = "disc",
391     [11] = "insert",
392     [12] = "vcenter",
393     [13] = "mathchoice",
394     [14] = "semisimple",
395     [15] = "mathshift",
396     [16] = "mathleft",
397     ["adjustedhbox"] = 3,
398     ["align"] = 5,
399     ["bottomlevel"] = 0,
400     ["disc"] = 10,
401     ["hbox"] = 2,
402     ["insert"] = 11,
```

```

403     ["math"] = 9,
404     ["mathchoice"] = 13,
405     ["mathleft"] = 16,
406     ["mathshift"] = 15,
407     ["noalign"] = 6,
408     ["output"] = 8,
409     ["semisimple"] = 14,
410     ["simple"] = 1,
411     ["vbox"] = 4,
412     ["vcenter"] = 12
413 },
414 ["etex.ifttype"] = {
415     [0] = "none",
416     [1] = "char",
417     [2] = "cat",
418     [3] = "num",
419     [4] = "dim",
420     [5] = "odd",
421     [6] = "vmode",
422     [7] = "hmode",
423     [8] = "mmode",
424     [9] = "inner",
425     [10] = "void",
426     [11] = "hbox",
427     [12] = "vbox",
428     [13] = "x",
429     [14] = "eof",
430     [15] = "true",
431     [16] = "false",
432     [17] = "case",
433     [18] = "defined",
434     [19] = "csname",
435     [20] = "fontchar",
436     ["case"] = 17,
437     ["cat"] = 2,
438     ["char"] = 1,
439     ["csname"] = 19,
440     ["defined"] = 18,
441     ["dim"] = 4,
442     ["eof"] = 14,
443     ["false"] = 16,
444     ["fontchar"] = 20,
445     ["hbox"] = 11,
446     ["hmode"] = 7,
447     ["inner"] = 9,
448     ["mmode"] = 8,
449     ["none"] = 0,
450     ["num"] = 3,
451     ["odd"] = 5,
452     ["true"] = 15,
453     ["vbox"] = 12,
454     ["vmode"] = 6,
455     ["void"] = 10,
456     ["x"] = 13
457 },
458 ["etex.nodetype"] = {
459     [-1] = "none",
460     [0] = "char",
461     [1] = "hlist",
462     [2] = "vlist",
463     [3] = "rule",
464     [4] = "ins",

```

```

465     [5] = "mark",
466     [6] = "adjust",
467     [7] = "ligature",
468     [8] = "disc",
469     [9] = "whatsit",
470     [10] = "math",
471     [11] = "glue",
472     [12] = "kern",
473     [13] = "penalty",
474     [14] = "unset",
475     [15] = "maths",
476     ["adjust"] = 6,
477     ["char"] = 0,
478     ["disc"] = 8,
479     ["glue"] = 11,
480     ["hlist"] = 1,
481     ["ins"] = 4,
482     ["kern"] = 12,
483     ["ligature"] = 7,
484     ["mark"] = 5,
485     ["math"] = 10,
486     ["maths"] = 15,
487     ["none"] = -1,
488     ["penalty"] = 13,
489     ["rule"] = 3,
490     ["unset"] = 14,
491     ["vlist"] = 2,
492     ["whatsit"] = 9
493 },
494 ["etex.interactionmode"] = {
495     [0] = "batch",
496     [1] = "nonstop",
497     [2] = "scroll",
498     [3] = "errorstop",
499     ["batch"] = 0,
500     ["errorstop"] = 3,
501     ["nonstop"] = 1,
502     ["scroll"] = 2
503 },
504 ["luatex.pdfliteral.mode"] = {
505     [0] = "setorigin",
506     [1] = "page",
507     [2] = "direct",
508     ["direct"] = 2,
509     ["page"] = 1,
510     ["setorigin"] = 0
511 }
512 }

513 function get(name, catcodetable)
514     local startpos, endpos, category, entry =
515         string.find(name, "^(%a[%a%d%.]*)%.(-%?[%a%d]+)$")
516     if not entry then
517         return
518     end
519     local node = data[category]
520     if not node then
521         return
522     end
523     local num = tonumber(entry)
524     local value
525     if num then
526         value = node[num]

```

```

527     if not value then
528         return
529     end
530 else
531     value = node[entry]
532     if not value then
533         return
534     end
535     value = "" .. value
536 end
537 tex.print(catcodetable, value)
538 end
539  $\langle$ /lua $\rangle$ 

```

## 3 Test

### 3.1 Catcode checks for loading

```

540  $\langle$ *test1 $\rangle$ 
541 \catcode'\{=1 %
542 \catcode'\}=2 %
543 \catcode'\#=6 %
544 \catcode'\@=11 %
545 \expandafter\ifx\csname count@\endcsname\relax
546     \countdef\count@=255 %
547 \fi
548 \expandafter\ifx\csname @gobble\endcsname\relax
549     \long\def\@gobble#1{}%
550 \fi
551 \expandafter\ifx\csname @firstofone\endcsname\relax
552     \long\def\@firstofone#1{#1}%
553 \fi
554 \expandafter\ifx\csname loop\endcsname\relax
555     \expandafter\@firstofone
556 \else
557     \expandafter\@gobble
558 \fi
559 {%
560     \def\loop#1\repeat{%
561         \def\body{#1}%
562         \iterate
563     }%
564     \def\iterate{%
565         \body
566         \let\next\iterate
567     \else
568         \let\next\relax
569     \fi
570     \next
571 }%
572 \let\repeat=\fi
573 }%
574 \def\RestoreCatcodes{}
575 \count@=0 %
576 \loop
577     \edef\RestoreCatcodes{%
578         \RestoreCatcodes
579         \catcode\the\count@=\the\catcode\count@\relax
580     }%
581 \ifnum\count@<255 %
582     \advance\count@ 1 %
583 \repeat

```

```

584
585 \def\RangeCatcodeInvalid#1#2{%
586   \count@=#1\relax
587   \loop
588     \catcode\count@=15 %
589     \ifnum\count@<#2\relax
590       \advance\count@ 1 %
591     \repeat
592 }
593 \expandafter\ifx\csname LoadCommand\endcsname\relax
594   \def\LoadCommand{\input magicnum.sty\relax}%
595 \fi
596 \def\Test{%
597   \RangeCatcodeInvalid{0}{47}%
598   \RangeCatcodeInvalid{58}{64}%
599   \RangeCatcodeInvalid{91}{96}%
600   \RangeCatcodeInvalid{123}{255}%
601   \catcode'\@=12 %
602   \catcode'\=0 %
603   \catcode'\{=1 %
604   \catcode'\}=2 %
605   \catcode'\#=6 %
606   \catcode'\[=12 %
607   \catcode'\]=12 %
608   \catcode'\%=14 %
609   \catcode'\ =10 %
610   \catcode13=5 %
611   \LoadCommand
612   \RestoreCatcodes
613 }
614 \Test
615 \csname @@end\endcsname
616 \end
617 </test1>

```

## 3.2 Test data

```

618 (*testplain)
619 \input magicnum.sty\relax
620 \def\Test#1#2{%
621   \edef\result{\magicnum{#1}}%
622   \edef\expect{#2}%
623   \edef\expect{\expandafter\stripprefix\meaning\expect}%
624   \ifx\result\expect
625     \else
626       \errmessage{%
627         Failed: [#1] % hash-ok
628         returns [\result] instead of [\expect]%
629       }%
630   \fi
631 }
632 \def\stripprefix#1->{}
633 </testplain>
634 (*testlatex)
635 \NeedsTeXFormat{LaTeX2e}
636 \documentclass{minimal}
637 \usepackage{magicnum}[2007/12/12]
638 \usepackage{qstest}
639 \IncludeTests{*}
640 \LogTests{log}{*}{*}
641 \newcommand*\Test[2]{%
642   \Expect*\magicnum{#1}{#2}%
643 }

```

```

644 \begin{qstest}{magicnum}{magicnum}
645 \end{qstest}

646 \begin{testdata}
647 \Test{tex.catcode.escape}{0}
648 \Test{tex.catcode.invalid}{15}
649 \Test{tex.catcode.unknown}{-1}
650 \Test{tex.catcode.0}{escape}
651 \Test{tex.catcode.15}{invalid}
652 \Test{etex.iftype.true}{15}
653 \Test{etex.iftype.false}{16}
654 \Test{etex.iftype.15}{true}
655 \Test{etex.iftype.16}{false}
656 \Test{etex.nodetype.none}{-1}
657 \Test{etex.nodetype.-1}{none}
658 \Test{luatex.pdfliteral.mode.direct}{2}
659 \Test{luatex.pdfliteral.mode.1}{page}
660 \Test{}{}
661 \Test{unknown}{}
662 \Test{unknown.foo.bar}{}
663 \Test{unknown.foo.4}{}
664 \end{testdata}

665 \begin{testplain}
666 \csname @@end\endcsname
667 \end
668 \end{testplain}
669 \end{testlatex}
670 \end{qstest}
671 \csname @@end\endcsname
672 \end{testlatex}

```

## 4 Installation

### 4.1 Download

**Package.** This package is available on CTAN<sup>1</sup>:

[CTAN:macros/latex/contrib/oberdiek/magicnum.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/magicnum.pdf](#) Documentation.

**Bundle.** All the packages of the bundle ‘oberdiek’ 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/oberdiek.tds.zip](#)

*TDS* refers to the standard “A Directory Structure for T<sub>E</sub>X Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

### 4.2 Bundle installation

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

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

---

<sup>1</sup><http://ftp.ctan.org/tex-archive/>



**Script installation.** Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

### 4.3 Package installation

**Unpacking.** The `.dtx` file is a self-extracting `docstrip` archive. The files are extracted by running the `.dtx` through plain- $\text{\TeX}$ :

```
tex magicnum.dtx
```

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

<code>magicnum.sty</code>	→ <code>tex/generic/oberdiek/magicnum.sty</code>
<code>magicnum.lua</code>	→ <code>scripts/oberdiek/magicnum.lua</code>
<code>oberdiek.magicnum.lua</code>	→ <code>scripts/oberdiek/oberdiek.magicnum.lua</code>
<code>magicnum.pdf</code>	→ <code>doc/latex/oberdiek/magicnum.pdf</code>
<code>magicnum.txt</code>	→ <code>doc/latex/oberdiek/magicnum.txt</code>
<code>test/magicnum-test1.tex</code>	→ <code>doc/latex/oberdiek/test/magicnum-test1.tex</code>
<code>test/magicnum-test2.tex</code>	→ <code>doc/latex/oberdiek/test/magicnum-test2.tex</code>
<code>test/magicnum-test3.tex</code>	→ <code>doc/latex/oberdiek/test/magicnum-test3.tex</code>
<code>magicnum.dtx</code>	→ <code>source/latex/oberdiek/magicnum.dtx</code>

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  $\text{\TeX}$  distribution (`te $\text{\TeX}$` , `mik $\text{\TeX}$` , ...) relies on file name databases, you must refresh these. For example, `te $\text{\TeX}$`  users run `texhash` or `mktextlsr`.

### 4.5 Some details for the interested

**Attached source.** The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is `pdftk`, e.g. unpack the file into the current directory:

```
pdftk magicnum.pdf unpack_files output .
```

**Unpacking with  $\text{\LaTeX}$ .** The `.dtx` chooses its action depending on the format:

**plain- $\text{\TeX}$ :** Run `docstrip` and extract the files.

**$\text{\LaTeX}$ :** Generate the documentation.

If you insist on using  $\text{\LaTeX}$  for `docstrip` (really, `docstrip` does not need  $\text{\LaTeX}$ ), 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 pdfL<sup>A</sup>T<sub>E</sub>X:

```
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.

## 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; numbers in *roman* refer to the code lines where the entry is used.

Symbols		
\#	543, 605	\countdef ..... 546
\%	608	\csname ..... 8, 21,
\@	544, 601	45, 58, 65, 94, 95, 106, 108, 110,
\@PackageErrorNoLine	98	112, 119, 121, 123, 130, 234,
\@ehc	100	237, 238, 251, 256, 259, 545,
\@firstofone	552, 555	548, 551, 554, 593, 615, 666, 671
\@gobble	549, 557	
\[	606	<b>D</b>
\]	602	\detokenize ..... 257, 260
\{	541, 603	\directlua ..... 136
\}	542, 604	\documentclass ..... 636
\]	607	
		<b>E</b>
		\empty ..... 12
		\end ..... 616, 667, 670
		\endcsname ..... 8, 21,
		45, 58, 65, 94, 95, 106, 108, 110,
		112, 120, 122, 123, 130, 234,
		237, 238, 251, 256, 259, 545,
		548, 551, 554, 593, 615, 666, 671
		\endinput ..... 30, 144
		\errmessage ..... 626
		\Expect ..... 642
		\expect ..... 622, 623, 624, 628
		<b>I</b>
		\ifcase ..... 9
		\ifnum ..... 581, 589
		\ifx ..... 10, 12, 21, 45, 53,
		94, 95, 106, 108, 110, 130, 234,
		251, 545, 548, 551, 554, 593, 624
		\immediate ..... 23, 47
		\IncludeTests ..... 639
		\input ..... 97, 131, 594, 619
<b>A</b>		
\advance	582, 590	
<b>B</b>		
\begin	644	
\body	561, 565	
<b>C</b>		
\catcode	3, 4, 5, 6, 7, 18, 19, 20,	
	34, 35, 36, 37, 38, 39, 40, 41, 42,	
	43, 44, 62, 63, 66, 67, 68, 69, 73,	
	74, 75, 76, 80, 82, 541, 542, 543,	
	544, 579, 588, 601, 602, 603,	
	604, 605, 606, 607, 608, 609, 610	
\CatcodeTableString	139	
\count@	546, 575,	
	579, 581, 582, 586, 588, 589, 590	

<code>\iterate</code> .....	562, 564, 566	<code>\ProvidesPackage</code> .....	59
<b>L</b>		<b>R</b>	
<code>\LoadCommand</code> .....	594, 611	<code>\RangeCatcodeInvalid</code> .....	585, 597, 598, 599, 600
<code>\LogTests</code> .....	640	<code>\repeat</code> .....	560, 572, 583, 591
<code>\loop</code> .....	560, 576, 587	<code>\RequirePackage</code> .....	133
<code>\luaescapestring</code> .....	139	<code>\RestoreCatcodes</code> ..	574, 577, 578, 612
<b>M</b>		<code>\result</code> .....	621, 624, 628
<code>\magicnum</code> 2, 99, 103, <u>107</u> , <u>135</u> , 621, 642		<b>S</b>	
<code>\magicnum@@add</code> .....	236, 241	<code>\space</code> .....	99
<code>\magicnum@add</code> <u>233</u> , 264, 265, 266, 267,		<code>\strip@prefix</code> .....	245, 248, 252
268, 269, 270, 271, 272, 273,		<code>\stripprefix</code> .....	623, 632
274, 275, 276, 277, 278, 279,		<b>T</b>	
280, 281, 282, 283, 284, 285,		<code>\Test</code> .....	596, 614,
286, 287, 288, 289, 290, 291,		620, 641, 647, 648, 649, 650,	
292, 293, 294, 295, 296, 297,		651, 652, 653, 654, 655, 656,	
298, 299, 300, 301, 302, 303,		657, 658, 659, 660, 661, 662, 663	
304, 305, 306, 307, 308, 309,		<code>\the</code> .....	66, 67, 68, 69, 80, 579
310, 311, 312, 313, 314, 315,		<code>\TMP@EnsureCode</code> .....	77,
316, 317, 318, 319, 320, 321,		84, 85, 86, 87, 88, 89, 90, 91, 92	
322, 323, 324, 325, 326, 327,		<b>U</b>	
328, 329, 330, 331, 332, 333,		<code>\usepackage</code> .....	637, 638
334, 335, 336, 337, 338, 339, 340		<b>W</b>	
<code>\magicnum@AtEnd</code> .....	78, 79, 143, 341	<code>\write</code> .....	23, 47
<code>\meaning</code> .....	245, 248, 623	<b>X</b>	
<b>N</b>		<code>\x</code> .....	8, 10, 12, 22,
<code>\NeedsTeXFormat</code> .....	635	26, 28, 46, 51, 58, 64, 72, 117, 126	
<code>\newcommand</code> .....	103, 641	<b>P</b>	
<code>\next</code> .....	566, 568, 570	<code>\PackageInfo</code> .....	26
<code>\number</code> .....	139	<b>R</b>	