

The `magicnum` package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2009/04/10 v1.1

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	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.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>	4
2	Implementation	4
2.1	Reload check and package identification	5
2.2	Catcodes	6
2.3	Check for previous definition	6
2.4	Without <code>LUA_TE_X</code>	6
2.5	With <code>LUA_TE_X</code>	7
2.6	Data	7
2.6.1	Plain data	7
2.6.2	Data for <code>T_EX</code>	9
2.6.3	Lua module	11
3	Test	14
3.1	Catcode checks for loading	14
3.2	Test data	15
4	Installation	16
4.1	Download	16
4.2	Bundle installation	17
4.3	Package installation	17
4.4	Refresh file name databases	17
4.5	Some details for the interested	17
5	History	18
	[2007/12/12 v1.0]	18
	[2009/04/10 v1.1]	18

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 L^AT_EX is available, they are stored in Lua tables.
- Without L^AT_EX 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`

`\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_EX'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^AT_EX.

```
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\@undefined
54 \xdef#1{#3}%
55 \fi
56 \ifx#1\relax
57 \xdef#1{#3}%
```

```

58      \fi
59    }%
60  \fi
61  \expandafter\x\csname ver@magicnum.sty\endcsname
62  \ProvidesPackage{magicnum}%
63  [2009/04/10 v1.1 Magic numbers (H0)]

```

2.2 Catcodes

```

64  \begingroup
65    \catcode123 1 % {
66    \catcode125 2 % }
67  \def\x{\endgroup
68    \expandafter\edef\csname magicnum@AtEnd\endcsname{%
69      \catcode35 \the\catcode35\relax
70      \catcode64 \the\catcode64\relax
71      \catcode123 \the\catcode123\relax
72      \catcode125 \the\catcode125\relax
73    }%
74  }%
75  \x
76  \catcode35 6 % #
77  \catcode64 11 % @
78  \catcode123 1 % {
79  \catcode125 2 % }
80  \def\TMP@EnsureCode#1#2{%
81    \edef\magicnum@AtEnd{%
82      \magicnum@AtEnd
83      \catcode#1 \the\catcode#1\relax
84    }%
85    \catcode#1 #2\relax
86  }
87  \TMP@EnsureCode{34}{12}% "
88  \TMP@EnsureCode{40}{12}% (
89  \TMP@EnsureCode{41}{12}% )
90  \TMP@EnsureCode{42}{12}% *
91  \TMP@EnsureCode{44}{12}% ,
92  \TMP@EnsureCode{45}{12}% -
93  \TMP@EnsureCode{46}{12}% .
94  \TMP@EnsureCode{47}{12}% /
95  \TMP@EnsureCode{60}{12}% <
96  \TMP@EnsureCode{62}{12}% >

```

2.3 Check for previous definition

```

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

```

2.4 Without LuaTeX

```

109  \begingroup\expandafter\expandafter\expandafter\endgroup
110  \expandafter\ifx\csname directlua\endcsname\relax

```

\magicnum

```

111 \begingroup\expandafter\expandafter\expandafter\endgroup
112 \expandafter\ifx\csname ifcsname\endcsname\relax
113   \def\magicnum#1{%
114     \expandafter\ifx\csname MG@#1\endcsname\relax
115       \else
116         \csname MG@#1\endcsname
117       \fi
118     }%
119   \else
120     \begingroup
121     \edef\x{\endgroup
122       \def\noexpand\magicnum##1{%
123         \expandafter\noexpand\csname
124           ifcsname\endcsname MG@##1\noexpand\endcsname
125           \noexpand\csname MG@##1%
126             \noexpand\expandafter\noexpand\endcsname
127           \expandafter\noexpand\csname fi\endcsname
128         }%
129       }%
130     \x
131   \fi
132 \else

```

2.5 With LuaTeX

```

133 \begingroup\expandafter\expandafter\expandafter\endgroup
134 \expandafter\ifx\csname RequirePackage\endcsname\relax
135   \input luatex.sty\relax
136 \else
137   \RequirePackage{luatex}[2009/04/10]%
138 \fi

```

\magicnum@directlua

```

139 \ifnum\luatexversion<36 %
140   \def\magicnum@directlua{\directlua0 }%
141 \else
142   \let\magicnum@directlua\directlua
143 \fi

```

\magicnum

```

144 \def\magicnum#1{%
145   \magicnum@directlua{%
146     require("oberdiek.magicnum")%
147     oberdiek.magicnum.get(%
148       "\luaescapestring{#1}",\number\CatcodeTableString
149     )%
150   }%
151 }%

152 \magicnum@AtEnd
153 \expandafter\endinput
154 \fi
155 \</package>

```

2.6 Data

2.6.1 Plain data

```

156 \<data>
157 \tex.catcode
158   escape = 0
159   begingroup = 1

```

```

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

```



```

222 ligature = 7
223 disc = 8
224 whatsit = 9
225 math = 10
226 glue = 11
227 kern = 12
228 penalty = 13
229 unset = 14
230 maths = 15
231 etex.interactionmode
232 batch = 0
233 nonstop = 1
234 scroll = 2
235 errorstop = 3
236 luatex.pdfliteral.mode
237 setorigin = 0
238 page = 1
239 direct = 2
240 </data>

```

2.6.2 Data for T_EX

```

241 <(*package)

\magicnum@add

242 \begingroup\expandafter\expandafter\expandafter\endgroup
243 \expandafter\ifx\csname detokenize\endcsname\relax
244   \def\magicnum@add#1#2#3{%
245     \expandafter\magicnum@@add
246       \csname MG@#1.#2\expandafter\endcsname
247       \csname MG@#1.#3\endcsname
248       {#3}{#2}%
249   }%
250   \def\magicnum@@add#1#2#3#4{%
251     \def#1{#3}%
252     \def#2{#4}%
253     \edef#1{%
254       \expandafter\strip@prefix\meaning#1%
255     }%
256     \edef#2{%
257       \expandafter\strip@prefix\meaning#2%
258     }%
259   }%
260   \expandafter\ifx\csname strip@prefix\endcsname\relax
261     \def\strip@prefix#1->{}%
262   \fi
263 \else
264   \def\magicnum@add#1#2#3{%
265     \expandafter\edef\csname MG@#1.#2\endcsname{%
266       \detokenize{#3}%
267     }%
268     \expandafter\edef\csname MG@#1.#3\endcsname{%
269       \detokenize{#2}%
270     }%
271   }%
272 \fi

273 \magicnum@add{tex.catcode}{escape}{0}
274 \magicnum@add{tex.catcode}{begingroup}{1}
275 \magicnum@add{tex.catcode}{endgroup}{2}
276 \magicnum@add{tex.catcode}{math}{3}
277 \magicnum@add{tex.catcode}{align}{4}
278 \magicnum@add{tex.catcode}{eol}{5}
279 \magicnum@add{tex.catcode}{parameter}{6}

```

```

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

```

```

342 \magicnum@add{etex.nodetype}{maths}{15}
343 \magicnum@add{etex.interactionmode}{batch}{0}
344 \magicnum@add{etex.interactionmode}{nonstop}{1}
345 \magicnum@add{etex.interactionmode}{scroll}{2}
346 \magicnum@add{etex.interactionmode}{errorstop}{3}
347 \magicnum@add{luatex.pdfliteral.mode}{setorigin}{0}
348 \magicnum@add{luatex.pdfliteral.mode}{page}{1}
349 \magicnum@add{luatex.pdfliteral.mode}{direct}{2}

350 \magicnum@AtEnd
351 \end{package}

```

2.6.3 Lua module

```

352 \lua
353 module("oberdiek.magicnum", package.seeall)

354 local data = {
355   ["tex.catcode"] = {
356     [0] = "escape",
357     [1] = "begingroup",
358     [2] = "endgroup",
359     [3] = "math",
360     [4] = "align",
361     [5] = "eol",
362     [6] = "parameter",
363     [7] = "superscript",
364     [8] = "subscript",
365     [9] = "ignore",
366     [10] = "space",
367     [11] = "letter",
368     [12] = "other",
369     [13] = "active",
370     [14] = "comment",
371     [15] = "invalid",
372     ["active"] = 13,
373     ["align"] = 4,
374     ["begingroup"] = 1,
375     ["comment"] = 14,
376     ["endgroup"] = 2,
377     ["eol"] = 5,
378     ["escape"] = 0,
379     ["ignore"] = 9,
380     ["invalid"] = 15,
381     ["letter"] = 11,
382     ["math"] = 3,
383     ["other"] = 12,
384     ["parameter"] = 6,
385     ["space"] = 10,
386     ["subscript"] = 8,
387     ["superscript"] = 7
388   },
389   ["etex.grouptype"] = {
390     [0] = "bottomlevel",
391     [1] = "simple",
392     [2] = "hbox",
393     [3] = "adjustedhbox",
394     [4] = "vbox",
395     [5] = "align",
396     [6] = "noalign",
397     [8] = "output",
398     [9] = "math",
399     [10] = "disc",
400     [11] = "insert",

```

```

401 [12] = "vcenter",
402 [13] = "mathchoice",
403 [14] = "semisimple",
404 [15] = "mathshift",
405 [16] = "mathleft",
406 ["adjustedhbox"] = 3,
407 ["align"] = 5,
408 ["bottomlevel"] = 0,
409 ["disc"] = 10,
410 ["hbox"] = 2,
411 ["insert"] = 11,
412 ["math"] = 9,
413 ["mathchoice"] = 13,
414 ["mathleft"] = 16,
415 ["mathshift"] = 15,
416 ["noalign"] = 6,
417 ["output"] = 8,
418 ["semisimple"] = 14,
419 ["simple"] = 1,
420 ["vbox"] = 4,
421 ["vcenter"] = 12
422 },
423 ["etex.ifttype"] = {
424 [0] = "none",
425 [1] = "char",
426 [2] = "cat",
427 [3] = "num",
428 [4] = "dim",
429 [5] = "odd",
430 [6] = "vmode",
431 [7] = "hmode",
432 [8] = "mmode",
433 [9] = "inner",
434 [10] = "void",
435 [11] = "hbox",
436 [12] = "vbox",
437 [13] = "x",
438 [14] = "eof",
439 [15] = "true",
440 [16] = "false",
441 [17] = "case",
442 [18] = "defined",
443 [19] = "csname",
444 [20] = "fontchar",
445 ["case"] = 17,
446 ["cat"] = 2,
447 ["char"] = 1,
448 ["csname"] = 19,
449 ["defined"] = 18,
450 ["dim"] = 4,
451 ["eof"] = 14,
452 ["false"] = 16,
453 ["fontchar"] = 20,
454 ["hbox"] = 11,
455 ["hmode"] = 7,
456 ["inner"] = 9,
457 ["mmode"] = 8,
458 ["none"] = 0,
459 ["num"] = 3,
460 ["odd"] = 5,
461 ["true"] = 15,
462 ["vbox"] = 12,

```

```

463     ["vmode"] = 6,
464     ["void"] = 10,
465     ["x"] = 13
466 },
467 ["etex.nodetype"] = {
468     [-1] = "none",
469     [0] = "char",
470     [1] = "hlist",
471     [2] = "vlist",
472     [3] = "rule",
473     [4] = "ins",
474     [5] = "mark",
475     [6] = "adjust",
476     [7] = "ligature",
477     [8] = "disc",
478     [9] = "whatsit",
479     [10] = "math",
480     [11] = "glue",
481     [12] = "kern",
482     [13] = "penalty",
483     [14] = "unset",
484     [15] = "maths",
485     ["adjust"] = 6,
486     ["char"] = 0,
487     ["disc"] = 8,
488     ["glue"] = 11,
489     ["hlist"] = 1,
490     ["ins"] = 4,
491     ["kern"] = 12,
492     ["ligature"] = 7,
493     ["mark"] = 5,
494     ["math"] = 10,
495     ["maths"] = 15,
496     ["none"] = -1,
497     ["penalty"] = 13,
498     ["rule"] = 3,
499     ["unset"] = 14,
500     ["vlist"] = 2,
501     ["whatsit"] = 9
502 },
503 ["etex.interactionmode"] = {
504     [0] = "batch",
505     [1] = "nonstop",
506     [2] = "scroll",
507     [3] = "errorstop",
508     ["batch"] = 0,
509     ["errorstop"] = 3,
510     ["nonstop"] = 1,
511     ["scroll"] = 2
512 },
513 ["luatex.pdfliteral.mode"] = {
514     [0] = "setorigin",
515     [1] = "page",
516     [2] = "direct",
517     ["direct"] = 2,
518     ["page"] = 1,
519     ["setorigin"] = 0
520 }
521 }

522 function get(name, catcodetable)
523     local startpos, endpos, category, entry =
524         string.find(name, "^(%a[%a%d%.]*)%.(?[%a%d]+)$")

```

```

525 if not entry then
526   return
527 end
528 local node = data[category]
529 if not node then
530   return
531 end
532 local num = tonumber(entry)
533 local value
534 if num then
535   value = node[num]
536   if not value then
537     return
538   end
539 else
540   value = node[entry]
541   if not value then
542     return
543   end
544   value = "" .. value
545 end
546 tex.print(catcodetable, value)
547 end
548  $\langle$ /lua $\rangle$ 

```

3 Test

3.1 Catcode checks for loading

```

549  $\langle$ *test1 $\rangle$ 
550 \catcode'\{=1 %
551 \catcode'\}=2 %
552 \catcode'\#=6 %
553 \catcode'\@=11 %
554 \expandafter\ifx\csname count@\endcsname\relax
555   \countdef\count@=255 %
556 \fi
557 \expandafter\ifx\csname @gobble\endcsname\relax
558   \long\def@gobble#1{}%
559 \fi
560 \expandafter\ifx\csname @firstofone\endcsname\relax
561   \long\def@firstofone#1{#1}%
562 \fi
563 \expandafter\ifx\csname loop\endcsname\relax
564   \expandafter@firstofone
565 \else
566   \expandafter@gobble
567 \fi
568 {%
569   \def\loop#1\repeat{%
570     \def\body{#1}%
571     \iterate
572   }%
573   \def\iterate{%
574     \body
575     \let\next\iterate
576   \else
577     \let\next\relax
578   \fi
579   \next
580 }%
581 \let\repeat=\fi

```

```

582 }%
583 \def\RestoreCatcodes{}
584 \count@=0 %
585 \loop
586   \edef\RestoreCatcodes{%
587     \RestoreCatcodes
588     \catcode\the\count@=\the\catcode\count@\relax
589   }%
590 \ifnum\count@<255 %
591   \advance\count@ 1 %
592 \repeat
593
594 \def\RangeCatcodeInvalid#1#2{%
595   \count@=#1\relax
596   \loop
597     \catcode\count@=15 %
598   \ifnum\count@<#2\relax
599     \advance\count@ 1 %
600   \repeat
601 }
602 \expandafter\ifx\csname LoadCommand\endcsname\relax
603   \def\LoadCommand{\input magicnum.sty\relax}%
604 \fi
605 \def\Test{%
606   \RangeCatcodeInvalid{0}{47}%
607   \RangeCatcodeInvalid{58}{64}%
608   \RangeCatcodeInvalid{91}{96}%
609   \RangeCatcodeInvalid{123}{255}%
610   \catcode'\@=12 %
611   \catcode'\=0 %
612   \catcode'\{=1 %
613   \catcode'\}=2 %
614   \catcode'\#=6 %
615   \catcode'\[=12 %
616   \catcode'\]=12 %
617   \catcode'\%=14 %
618   \catcode'\ =10 %
619   \catcode13=5 %
620   \LoadCommand
621   \RestoreCatcodes
622 }
623 \Test
624 \csname @@end\endcsname
625 \end
626 </test1>

```

3.2 Test data

```

627 <*testplain>
628 \input magicnum.sty\relax
629 \def\Test#1#2{%
630   \edef\result{\magicnum{#1}}%
631   \edef\expect{#2}%
632   \edef\expect{\expandafter\stripprefix\meaning\expect}%
633   \ifx\result\expect
634   \else
635     \errmessage{%
636       Failed: [#1] % hash-ok
637       returns [\result] instead of [\expect]%
638     }%
639   \fi
640 }
641 \def\stripprefix#1->{}

```

```

642 </testplain>
643 <*testlatex>
644 \NeedsTeXFormat{LaTeX2e}
645 \documentclass{minimal}
646 \usepackage{magicnum}[2009/04/10]
647 \usepackage{qstest}
648 \IncludeTests{*}
649 \LogTests{log}{*}{*}
650 \newcommand*{\Test}[2]{%
651   \Expect*{\magicnum{#1}}{#2}%
652 }
653 \begin{qstest}{magicnum}{magicnum}
654 </testlatex>

655 <*testdata>
656 \Test{tex.catcode.escape}{0}
657 \Test{tex.catcode.invalid}{15}
658 \Test{tex.catcode.unknown}{}
659 \Test{tex.catcode.0}{escape}
660 \Test{tex.catcode.15}{invalid}
661 \Test{etex.iftex.true}{15}
662 \Test{etex.iftex.false}{16}
663 \Test{etex.iftex.15}{true}
664 \Test{etex.iftex.16}{false}
665 \Test{etex.nodetype.none}{-1}
666 \Test{etex.nodetype.-1}{none}
667 \Test{luatex.pdfliteral.mode.direct}{2}
668 \Test{luatex.pdfliteral.mode.1}{page}
669 \Test{}{}
670 \Test{unknown}{}
671 \Test{unknown.foo.bar}{}
672 \Test{unknown.foo.4}{}
673 </testdata>

674 <*testplain>
675 \csname @@end\endcsname
676 \end
677 </testplain>
678 <*testlatex>
679 \end{qstest}
680 \csname @@end\endcsname
681 </testlatex>

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

[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_EX Files” ([CTAN:tds/tds.pdf](#)). Directories with `texmf` in their name are usually organized this way.

¹<http://ftp.ctan.org/tex-archive/>

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
```

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-`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 `TEX` distribution (te`TEX`, mi`kTEX`, ...) relies on file name databases, you must refresh these. For example, te`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 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 L^AT_EX 0.40.

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		553, 588, 597, 610, 611, 612,
\#	552, 614	613, 614, 615, 616, 617, 618, 619
\%	617	\CatcodeTableString 148
\@	553, 610	\count@ 555, 584,
\@PackageErrorNoLine	102	588, 590, 591, 595, 597, 598, 599
\@ehc	104	\countdef 555
\@firstofone	561, 564	\csname 8, 21,
\@gobble	558, 566	45, 61, 68, 98, 99, 110, 112, 114,
\@undefined	53	116, 123, 125, 127, 134, 243,
\[615	246, 247, 260, 265, 268, 554,
\]	611	557, 560, 563, 602, 624, 675, 680
\{	550, 612	
\}	551, 613	
\]	616	
		D
_	618	\detokenize 266, 269
		\directlua 140, 142
		\documentclass 645
A		E
\advance	591, 599	\empty 12
		\end 625, 676, 679
B		\endcsname 8, 21,
\begin	653	45, 61, 68, 98, 99, 110, 112, 114,
\body	570, 574	116, 124, 126, 127, 134, 243,
		246, 247, 260, 265, 268, 554,
C		557, 560, 563, 602, 624, 675, 680
\catcode	3, 4, 5, 6, 7, 18, 19, 20,	\endinput 30, 153
	34, 35, 36, 37, 38, 39, 40, 41, 42,	\errmessage 635
	43, 44, 65, 66, 69, 70, 71, 72, 76,	\Expect 651
	77, 78, 79, 83, 85, 550, 551, 552,	\expect 631, 632, 633, 637

I		\newcommand 107, 650	
\ifcase	9	\next	575, 577, 579
\ifnum	139, 590, 598	\number	148
\ifx	10, 12, 21, 45, 53, 56, 98, 99, 110, 112, 114, 134, 243, 260, 554, 557, 560, 563, 602, 633	P	
\immediate	23, 47	\PackageInfo	26
\IncludeTests	648	\ProvidesPackage	62
\input	101, 135, 603, 628	R	
\iterate	571, 573, 575	\RangeCatcodeInvalid	
L		594, 606, 607, 608, 609
\LoadCommand	603, 620	\repeat	569, 581, 592, 600
\LogTests	649	\RequirePackage	137
\loop	569, 585, 596	\RestoreCatcodes ..	583, 586, 587, 621
\luaescapestring	148	\result	630, 633, 637
\luatexversion	139	S	
M		\space	103
\magicnum 2, 103, 107, <u>111</u> , <u>144</u> , 630, 651		\strip@prefix	254, 257, 261
\magicnum@@add	245, 250	\stripprefix	632, 641
\magicnum@add <u>242</u> , 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349		T	
\magicnum@AtEnd	81, 82, 152, 350	\Test	605, 623, 629, 650, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672
\magicnum@directlua	<u>139</u> , 145	\the	69, 70, 71, 72, 83, 588
\meaning	254, 257, 632	\TMP@EnsureCode	80, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96
N		U	
\NeedsTeXFormat	644	\usepackage	646, 647
O		W	
\newcommand		\write	23, 47
X		X	
\x	8, 10, 12, 22, 26, 28, 46, 51, 61, 67, 75, 121, 130		