

The luatexbase-cctb package

Manuel Pégourié-Gonnard & Élie Roux

Support: lualatex-dev@tug.org

v0.3 2010-05-27

Abstract

In addition to the registers existing in \TeX and $\varepsilon\text{-}\TeX$, Lua \TeX introduces a new concept: catcode tables. This package takes care of catcode table allocation just like Plain \TeX and LaTeX do for other registers.

Contents

1	Documentation	1
2	Implementation	2
2.1	\TeX package	2
2.1.1	Preliminaries	2
2.1.2	Load supporting Lua module	4
2.1.3	Primitives needed	4
2.1.4	User macros	4
2.1.5	Predefined tables	5
2.2	Lua module	6
3	Test files	7

1 Documentation

The main macro defined here is `\newluatexcatcodetable`. It behaves the same as `\newbox`. Additionally, the newly allocated catcode table is initialised to the catcodes of `Ini \TeX` . In order to help you define the catcode tables (once they are allocated), two helper macros are available.

```
\setcatcoderange{<from>}{<to>}{<value>}
```

Set all characters code in the range `<from>–<to>` to the given catcode `<value>`.

```
\setluatexcatcodetable{<table>}{<catcode statements>}
```

Set a previously allocated `<table>` to the catcodes given by executing `<catcode statements>`. Note that `<table>` must not be the current active catcode table. You may, however, load another catcode table in your `<catcode statements>`.

For your convenience, a few catcode tables are predefined:

- `\CatcodeTableIniTeX`: Ini \TeX catcodes.
- `\CatcodeTableString`: the catcode regime used by `\string` and `\meaning`: everything has catcode 12, except space (U+0020) that has catcode 10.
- `\CatcodeTableOther`: everything (included space) has catcode 12.
- `\CatcodeTableLaTeX`: basic $\LaTeX 2_{\epsilon}$ catcodes.
- `\CatcodeTableLaTeXAtLetter`: same as above, but `@` is a letter.
- `\CatcodeTableExpl`: catcodes used by $\LaTeX 3$ with `\ExplSyntaxOn`. Be aware that this does not provide the exact same environment as `\ExplSyntaxOn`: most noticeably, some booleans are not set, and `\endlinchar` is not adjusted (it should be 32).

Various Lua function accept a catcode table number as argument. In order to use them, the package writer needs to know the number of an allocated catcode table. Since `\chardef` is used for the definition of the control sequence, this is rather easy to do. However, for extra ease of use, the numbers are also directly accessible from Lua as the value of the table `luatexbase.catcodetables`, whose keys is the name of the control sequence (without any leading backslash). Moreover, nickames are available for the predefined catcode tables:

- `CatcodeTableIniTeX = ini`,
- `CatcodeTableString = string`,
- `CatcodeTableOther = other`,
- `CatcodeTableLaTeX = latex`,
- `CatcodeTableLaTeXAtLetter = latex-atletter = latex-package`,
- `CatcodeTableExpl = expl = expl3`,

2 Implementation

2.1 \TeX package

```
1 (*texpackage)
```

2.1.1 Preliminaries

Reload protection, especially for Plain \TeX .

```
2           \csname lltxb@cctb@loaded\endcsname
3 \expandafter\let\csname lltxb@cctb@loaded\endcsname\endinput

   Catcode defenses.

4 \begingroup
5   \catcode123 1 % {
6   \catcode125 2 % }
7   \catcode 35 6 % #
8   \toks0{}%
9   \def\x{}%
10  \def\y#1 #2 {%
11    \toks0\expandafter{\the\toks0 \catcode#1 \the\catcode#1}%

```

```

12   \edef\x{\x \catcode#1 #2}}%
13   \y 123 1 % {
14   \y 125 2 % }
15   \y 35 6 % #
16   \y 10 12 % ^^J
17   \y 34 12 % "
18   \y 36 3 % $ $
19   \y 39 12 % '
20   \y 40 12 % (
21   \y 41 12 % )
22   \y 42 12 % *
23   \y 43 12 % +
24   \y 44 12 % ,
25   \y 45 12 % -
26   \y 46 12 % .
27   \y 47 12 % /
28   \y 60 12 % <
29   \y 61 12 % =
30   \y 64 11 % @ (letter)
31   \y 62 12 % >
32   \y 95 12 % _ (other)
33   \y 96 12 % `
34   \edef\y#1{\endgroup\edef#1{\the\toks0\relax}\x}%
35 \expandafter\y\csname lltxb@cctb@AtEnd\endcsname

Package declaration.

36 \begingroup
37 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
38   \def\x#1[#2]{\immediate\write16{Package: #1 #2}}
39 \else
40   \let\x\ProvidesPackage
41 \fi
42 \expandafter\endgroup
43 \x{luatexbase-cctb}[2010/10/10 v0.3 Catcodetable allocation for LuaTeX]

Make sure LuaTeX is used.

44 \begingroup\expandafter\expandafter\expandafter\endgroup
45 \expandafter\ifx\csname RequirePackage\endcsname\relax
46   \input ifluatex.sty
47 \else
48   \RequirePackage{ifluatex}
49 \fi
50 \ifluatex\else
51   \begingroup
52     \expandafter\ifx\csname PackageError\endcsname\relax
53       \def\x#1#2#3{\begingroup \newlinechar10
54         \errhelp{#3}\errmessage{Package #1 error: #2}\endgroup}
55     \else
56       \let\x\PackageError
57     \fi
58   \expandafter\endgroup
59 \x{luatexbase-attr}{LuaTeX is required for this package. Aborting.}{%
60   This package can only be used with the LuaTeX engine^^J%
61   (command 'lualatex' or 'luatex').^^J%

```

```

62   Package loading has been stopped to prevent additional errors.}
63   \llt@ccctb@AtEnd
64   \expandafter\endinput
65 \fi

```

2.1.2 Load supporting Lua module

First load `luatexbase-loader` (hence `luatexbase-compat`), then the supporting Lua module.

```

66 \begingroup\expandafter\expandafter\expandafter\endgroup
67 \expandafter\ifx\csname RequirePackage\endcsname\relax
68   \input luatexbase-loader.sty
69 \else
70   \RequirePackage{luatexbase-loader}
71 \fi
72 \luatexbase@directlua{require('luatexbase.cctb')}

```

2.1.3 Primitives needed

Load `luatexbase-compat`.

```

73 \begingroup\expandafter\expandafter\expandafter\endgroup
74 \expandafter\ifx\csname RequirePackage\endcsname\relax
75   \input luatexbase-compat.sty
76 \else
77   \RequirePackage{luatexbase-compat}
78 \fi

```

Make sure the primitives we need are available.

```

79 \luatexbase@ensure@primitive{luaescapestring}
80 \luatexbase@ensure@primitive{catcodetable}
81 \luatexbase@ensure@primitive{initcatcodetable}
82 \luatexbase@ensure@primitive{savecatcodetable}

```

2.1.4 User macros

The allocation macro. Allocate tables starting with 1, since table 0 is reserved for IniTeX catcodes by LuaTeX.

```

83 \newcount\llt@catcodetable@alloc
84 \llt@catcodetable@alloc\z@
85 \def\newluatexcatcodetable#1{%
86   \ifnum\llt@catcodetable@alloc<65535\relax
87     \global\advance\llt@catcodetable@alloc\@ne
88     \allocationnumber\llt@catcodetable@alloc
89     \global\chardef#1\allocationnumber
90     \luatexinitcatcodetable\allocationnumber
91     \begingroup\escapechar\m@ne
92     \luatexbase@directlua{luatexbase.catcodetabledef_from_tex(
93       '\luatexluaescapestring{\string#1}', '\number\allocationnumber')}%
94     \endgroup
95     \wlog{\string#1=\string\luatexcatcodetable\the\allocationnumber}%
96   \else
97     \errmessage{No room for a new \string\luatexcatcodetable}%
98   \fi}

```

A small patch to get two new counters in Plain too.

```
99 \expandafter\ifx\csname @tempcnta\endcsname\relax
100 \csname newcount\endcsname\@tempcnta
101 \fi
102 \expandafter\ifx\csname @tempcntb\endcsname\relax
103 \csname newcount\endcsname\@tempcntb
104 \fi
```

Set the catcodes for a range of characters.

```
105 \def\setcatcoderange#1#2#3{%
106 \edef\luaSCR@temp{%
107 \noexpand\@tempcnta=\the\@tempcnta
108 \noexpand\@tempcntb=\the\@tempcntb
109 \noexpand\count@=\the\count@
110 \relax}%
111 \@tempcnta=#1\relax
112 \@tempcntb=#2\relax
113 \count@=#3\relax
114 \loop\unless\ifnum\@tempcnta>\@tempcntb
115 \catcode\@tempcnta=\count@
116 \advance\@tempcnta\@ne
117 \repeat
118 \luaSCR@temp}
```

Set a catcode table.

```
119 \def\setluatexcatcodetable#1#2{%
120 \begingroup
121 #2%
122 \luatexsavecatcodetable#1%
123 \endgroup}
```

2.1.5 Predefined tables

The IniTeX catcode table needs no extra initialisation.

```
124 \newluatexcatcodetable\CatcodeTableIniTeX
```

The String and Other catcode tables.

```
125 \newluatexcatcodetable\CatcodeTableString
126 \setluatexcatcodetable\CatcodeTableString{%
127 \luatexcatcodetable\CatcodeTableIniTeX
128 \catcode0 12 % nul
129 \catcode13 12 % carriage return
130 \catcode37 12 % percent
131 \setcatcoderange{65}{90}{12}% A-Z
132 \setcatcoderange{97}{122}{12}% a-z
133 \catcode92 12 % backslash
134 \catcode127 12 }

135 \newluatexcatcodetable\CatcodeTableOther
136 \setluatexcatcodetable\CatcodeTableOther{%
137 \luatexcatcodetable\CatcodeTableString
138 \catcode32 12 }
```

Tables for L^AT_EX 2_ε.

```
139 \newluatexcatcodetable\CatcodeTableLaTeX
140 \setluatexcatcodetable\CatcodeTableLaTeX{%
141   \luatexcatcodetable\CatcodeTableIniTeX
142   \setcatcoderange{0}{31}{15}%
143   \catcode9   10 % tab
144   \catcode12  13 % form feed
145   \catcode13  5  % carriage return
146   \catcode35  6  % hash
147   \catcode36  3  % dollar
148   \catcode38  4  % ampersand
149   \catcode94  7  % circumflex
150   \catcode95  8  % underscore
151   \catcode123 1  % brace left
152   \catcode125 2  % brace right
153   \catcode126 13 % tilde
154   \catcode127 15 }

155 \newluatexcatcodetable\CatcodeTableLaTeXAtLetter
156 \setluatexcatcodetable\CatcodeTableLaTeXAtLetter{%
157   \luatexcatcodetable\CatcodeTableLaTeX
158   \catcode64=11 }
```

A table for expl3.

```
159 \newluatexcatcodetable\CatcodeTableExpl
160 \setluatexcatcodetable\CatcodeTableExpl{%
161   \luatexcatcodetable\CatcodeTableLaTeX
162   \catcode126 10 % tilde is a space char
163   \catcode32  9  % space is ignored
164   \catcode9   9  % tab also ignored
165   \catcode95  11 % underscore letter
166   \catcode58  11 % colon letter
167 }
```

Finally do the shortcuts.

```
168 \luatexbase@directlua{luatexbase.catcodetable_do_shortcuts()}

    That's all, folks!

169 \lltxb@cctb@AtEnd
170 </texpackage>
```

2.2 Lua module

```
171 <!*luamodule>
172 module('luatexbase', package.seeall)
```

The number associated to a CS name is remembered in the `catcodetables` table.

```
173 catcodetables = {}
174 function catcodetabledef_from_tex(name, number)
175   catcodetables[name] = tonumber(number)
176 end
```

The next function creates some shortcuts for better readability in lua code. This makes `luatexbase.catcodetables.latex` equivalent to `luatexbase.catcodetables.CatcodeTableLaTeX`.

```

177 function catcodetable_do_shortcuts()
178     local cat = catcodetables
179     cat['latex']           = cat.CatcodeTableLaTeX
180     cat['latex-package']  = cat.CatcodeTableLaTeXAtLetter
181     cat['latex-atletter'] = cat.CatcodeTableLaTeXAtLetter
182     cat['ini']            = cat.CatcodeTableIniTeX
183     cat['expl3']          = cat.CatcodeTableExpl
184     cat['expl']           = cat.CatcodeTableExpl
185     cat['string']         = cat.CatcodeTableString
186     cat['other']          = cat.CatcodeTableOther
187 end
188 </luamodule>

```

3 Test files

The tests done are very basic: we just make sure that the package loads correctly and the macros don't generate any error, under both LaTeX en Plain TeX.

```

189 <testplain,testlatex>\catcode00 15
190 <testplain>\input luatexbase-cctb.sty
191 <testlatex>\RequirePackage{luatexbase-cctb}
192 <*testplain,testlatex>
193 \begingroup \catcode64 11 \global\let\lua\luatexbase@directlua \endgroup

```

Also check that the catcodetable's number is remembered well, independently of the current value of `\escapechar`.

```

194 \newluatexcatcodetable\testcctb
195 \lua{assert(luatexbase.catcodetables.testcctb)}
196 \begingroup
197 \escapechar64
198 \newluatexcatcodetable\anothercctb
199 \endgroup
200 \lua{assert(luatexbase.catcodetables.anothercctb)}

```

Now, play a little bit with predefined tables. Be careful to change catcodes only inside a group to avoid chaos.

```

201 \def\test#1#2#3{%
202     \begingroup
203     \ifcsname CatcodeTable#1\endcsname \else \INTERR \fi
204     \luatexcatcodetable\csname CatcodeTable#1\endcsname
205     \ifnum\catcode#2=#3 \else \ERROR \fi
206     \endgroup}
207 \test {IniTeX}      {00} {09}
208 \test {IniTeX}      {92} {00}
209 \test {IniTeX}      {64} {12}
210 \test {IniTeX}      {65} {11}
211 \test {String}      {92} {12}
212 \test {String}      {65} {12}
213 \test {String}      {32} {10}
214 \test {Other}       {92} {12}
215 \test {Other}       {65} {12}
216 \test {Other}       {32} {12}
217 \test {LaTeXAtLetter} {64} {11}

```

```
218 \test {LaTeX}          {64} {12}
219 \test {Expl}           {32} {09}
220 (testlatex)\documentclass{minimal}
221 \lua{%
222   tex.sprint('\string\setbox0=\string\hbox{')
223   tex.sprint(luatexbase.catcodetables.string, "\string\undef # _^&")
224   tex.sprint('}')
225 }

226 </testplain, testlatex>
227 (testplain)\bye
228 (testlatex)\stop
```