

# The `hobsub` package

Heiko Oberdiek  
<heiko.oberdiek at gmail.com>

2012/05/28 v1.13

## Abstract

Package `hobsub` implements the idea of loading several packages in one big collection package.

## Contents

<b>1</b>	<b>Documentation</b>	<b>2</b>
1.1	Usage	2
1.2	Supported features	3
1.3	Common limitations	3
1.4	Package <code>hobsub-generic</code>	3
1.5	Package <code>hobsub-hyperref</code>	3
1.6	Package <code>hobsub</code>	5
<b>2</b>	<b>Implementation</b>	<b>6</b>
2.1	Reload check and package identification	6
2.1.1	Package <code>hobsub</code>	6
2.1.2	Package <code>hobsub-generic</code>	7
2.1.3	Package <code>hobsub-hyperref</code>	8
2.2	Catcodes	9
2.3	Package <code>hobsub-hyperref</code> loads <code>hobsub-generic</code>	10
2.4	Preamble	10
2.5	End of preamble for subset packages	12
2.6	Package list	12
2.7	End of packages	13
<b>3</b>	<b>Test</b>	<b>13</b>
3.1	Catcode checks for loading	13
<b>4</b>	<b>Installation</b>	<b>14</b>
4.1	Download	14
4.2	Bundle installation	15
4.3	Package installation	15
4.4	Refresh file name databases	15
4.5	Some details for the interested	15
<b>5</b>	<b>Catalogue</b>	<b>16</b>
<b>6</b>	<b>References</b>	<b>17</b>

<b>7 History</b>	<b>17</b>
[2011/01/30 v1.0]	17
[2011/04/16 v1.1]	17
[2011/04/17 v1.2]	17
[2011/04/18 v1.3]	17
[2011/04/23 v1.4]	17
[2011/06/24 v1.5]	17
[2011/06/30 v1.6]	17
[2011/07/01 v1.7]	17
[2011/07/28 v1.8]	17
[2011/08/22 v1.9]	17
[2011/10/16 v1.10]	17
[2011/11/29 v1.11]	17
[2012/04/25 v1.12]	18
[2012/05/28 v1.13]	18
<b>8 Index</b>	<b>18</b>

## 1 Documentation

In January 2011 the mailing list `lualatex-dev` discussed “quietening lualatex console output” [1]. Inspired from this, I implemented this package `hobsub` to load several packages in one file. The package is applied for two packages collections that are subsets of my package bundle `oberdiek`.

**hobsub-generic** is a collection of packages that can also be used with plain `TeX`.

**hobsub-hyperref** is a collection of packages that is used by package `hyperref`.

Package `hobsub` provides the macros that are used in the collection packages and might be used in other projects as well. But there are many caveats, some of them are discussed below.

The most serious problem is consistency. A package that is available as standalone package and is part of a collection might have different versions. Because my packages are organized in a bundle and should be installed and updated as bundle, the risk seems low to me. However, if a single package is updated manually, for example for testing, then it should be loaded before the collection that contains the package with an older version. In a collection, the package will not be loaded again if it is already loaded.

### 1.1 Usage

The collection packages `hobsub-generic` and `hobsub-hyperref` are loaded as usual:

```
\usepackage{hobsub-generic}  LATEX
\input hobsub-generic.sty    plain TEX
```

or

```
\usepackage{hobsub-hyperref}
```

Of course, both `\usepackage` or `\RequirePackage` can be used.

If you need requirements on the version date of a package inside a collection, then specify them afterwards, for example:

```
\usepackage{hobsub-generic}
\usepackage{ltxcmds}[2010/12/04]% \ltx@ifblank is needed
```

Also it is not guaranteed that a package will be always part of a collection. Therefore it does not harm, to make the requirements explicite *after* a collection is loaded, see the previous example. `LATEX` knows that a package is already loaded and does not load it again.

## 1.2 Supported features

**No reloads:** If a package is already loaded then it is not reloaded again if it is part of a collection package.

**\listfiles:** Each package that is loaded in a collection is also added to the output of `\listfiles`.

**\@currname, \@current** are updated, they are used by L<sup>A</sup>T<sub>E</sub>X's `\ProvidesPackage`.

**\AtEndOfPackage:** L<sup>A</sup>T<sub>E</sub>X's hook mechanism that is used for `\AtEndOfPackage` is supported. However none of the packages in the collections are using `\AtEndOfPackage`, therefore this feature is completely untested.

## 1.3 Common limitations

**\endinput:** Often a package is using `\endinput`, either added by `docstrip`, or to stop package loading at an earlier time. Therefore `\endinput` must be caught during package loading to prevent that the collection package is closed too early before reading all packages it contains.

This redefinition of `\endinput` fails if the package loads other files that contain `\endinput`. To some degree it could be caught by redefining macros that load files. But this will not work for the primitive `\input`.

**Options:** L<sup>A</sup>T<sub>E</sub>X's package options are not supported. Therefore the packages that are put in a collection must not support options.

**Hook packages:** Some packages and classes provides hooks for packages, see packages `scrfile`, `filehook` or class `memoir`. These hooks are not supported for embedded packages inside a collection package. Because of the problem with `\endinput`, these hooks are not supported. The hooks might contain code that loads other files that execute `\endinput`.

## 1.4 Package hobsb-generic

Collection 'generic' contains some of the plain T<sub>E</sub>X compatible packages of my bundle, see table 1. It adds the following restrictions to any package that is part of this collection:

**Generic formats:** The following formats must be supported: L<sup>A</sup>T<sub>E</sub>X, plain T<sub>E</sub>X and even iniT<sub>E</sub>X. This applies to the collection package `hobsb-generic` the same way.

**Prefixed macros:** Any package of this collection should only define macros with prefixed names to avoid name clashes. Resources like registers must not be allocated and other global states should not be changed except for added macro definitions.

Thus the package `hobsb-generic` can also preloaded in the format file with iniT<sub>E</sub>X.

## 1.5 Package hobsb-hyperref

Collection 'hyperref' contains additional packages that are used by package `hyperref` and that does not fit in collection 'generic', see `tabletab:hyperref`. The latter collection package `hobsb-generic` is loaded via a conventional `\RequirePackage`.

Because most of the packages in the collection 'hyperref' needs L<sup>A</sup>T<sub>E</sub>X, the collection requires L<sup>A</sup>T<sub>E</sub>X.

Table 1: Overview hobsub-generic

<b>Package</b>	<b>Release</b>	<b>Dependencies</b>
infwarerr	2010/04/08 v1.3	
ltxcmds	2011/11/09 v1.22	
ifluatex	2010/03/01 v1.3	
ifvtex	2010/03/01 v1.5	
intcalc	2007/09/27 v1.1	
ifpdf	2011/01/30 v2.3	ifluatex
etexcmds	2011/02/16 v1.5	
kvsetkeys	2012/04/25 v1.16	
kvdefinekeys	2011/04/07 v1.3	
luatex-loader	2010/03/09 v0.4	
pdftexcmds	2011/11/29 v0.20	ifluatex, ifpdf, infwarerr, ltxcmds, luatex-loader
pdfescape	2011/11/25 v1.13	ltxcmds, pdftexcmds
bigintcalc	2012/04/08 v1.3	pdftexcmds
bitset	2011/01/30 v1.1	bigintcalc, infwarerr, intcalc
uniquecounter	2011/01/30 v1.2	bigintcalc, infwarerr

Table 2: Overview hobsub-hyperref

<b>Package</b>	<b>Release</b>	<b>Dependencies</b>
letltxmacro	2010/09/02 v1.4	
hopatch	2012/05/28 v1.2	
xcolor-patch	2011/01/30 v1.7	
atveryend	2011/06/30 v1.8	
atbegshi	2011/10/05 v1.16	ifpdf, infwarerr, ltxcmds
refcount	2011/10/16 v3.4	infwarerr, ltxcmds
hycolor	2011/01/30 v1.7	xcolor-patch

## 1.6 Package `hobsub`

This section is for advanced users that might want to build a collection package. Then it can be done by using `hobsub`:

```
...
\RequirePackage{hobsub}[2012/05/28]
...
\hobsub@StartPackage{foo1}
... package code of foo1.sty ...
\endinput
<hobsub>
\hobsub@StopPackage
\hobsub@StartPackage{bar2}
... package code of bar2.sty ...
\endinput
<hobsub>
\hobsub@StopPackage ...
```

But be aware of the caveats and limitations, see above. Only quite a small portion of the packages can be embedded like this.

## 2 Implementation

### 2.1 Reload check and package identification

#### 2.1.1 Package hobsup

```
1 (*package)
```

Reload check, especially if the package is not used with L<sup>A</sup>T<sub>E</sub>X.

```
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@hobsub.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{hobsub}{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@hobsub.sty\endcsname
67 \ProvidesPackage{hobsub}%
68 [2012/05/28 v1.13 Construct package bundles (H0)]%
69 \endpackage)

```

### 2.1.2 Package hobsub-generic

```
70 <*generic>
```

Reload check, especially if the package is not used with L<sup>A</sup>T<sub>E</sub>X.

```

71 \begingroup\catcode61\catcode48\catcode32=10\relax%
72 \catcode13=5 % ^~M
73 \endlinechar=13 %
74 \catcode35=6 % #
75 \catcode39=12 % '
76 \catcode44=12 % ,
77 \catcode45=12 % -
78 \catcode46=12 % .
79 \catcode58=12 % :
80 \catcode64=11 % @
81 \catcode123=1 % {
82 \catcode125=2 % }
83 \expandafter\let\expandafter\x\csname ver@hobsub-generic.sty\endcsname
84 \ifx\x\relax % plain-TeX, first loading
85 \else
86   \def\empty{}%
87   \ifx\x\empty % LaTeX, first loading,
88     % variable is initialized, but \ProvidesPackage not yet seen
89   \else
90     \expandafter\ifx\csname PackageInfo\endcsname\relax
91       \def\x#1#2{%
92         \immediate\write-1{Package #1 Info: #2.}%
93       }%
94     \else
95       \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
96     \fi
97     \x{hobsub-generic}{The package is already loaded}%
98     \aftergroup\endinput
99   \fi
100 \fi
101 \endgroup%

```

Package identification:

```

102 \begingroup\catcode61\catcode48\catcode32=10\relax%
103 \catcode13=5 % ^~M
104 \endlinechar=13 %
105 \catcode35=6 % #
106 \catcode39=12 % '
107 \catcode40=12 % (
108 \catcode41=12 % )
109 \catcode44=12 % ,
110 \catcode45=12 % -
111 \catcode46=12 % .

```

```

112 \catcode47=12 % /
113 \catcode58=12 % :
114 \catcode64=11 % @
115 \catcode91=12 % [
116 \catcode93=12 % ]
117 \catcode123=1 % {
118 \catcode125=2 % }
119 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
120   \def\x#1#2#3[#4]{\endgroup
121     \immediate\write-1{Package: #3 #4}%
122     \xdef#1{#4}%
123   }%
124 \else
125   \def\x#1#2[#3]{\endgroup
126     #2[#{3}]%
127     \ifx#1\undefined
128       \xdef#1{#3}%
129     \fi
130     \ifx#1\relax
131       \xdef#1{#3}%
132     \fi
133   }%
134 \fi
135 \expandafter\x\csname ver@hobsub-generic.sty\endcsname
136 \ProvidesPackage{hobsub-generic}%
137 [2012/05/28 v1.13 Bundle oberdiek, subset generic (H0)]%
138 </generic>

```

### 2.1.3 Package hobsub-hyperref

```
139 <*hyperref>
```

Reload check, especially if the package is not used with L<sup>A</sup>T<sub>E</sub>X.

```

140 \begingroup\catcode61\catcode48\catcode32=10\relax%
141   \catcode13=5 % ^^M
142   \endlinechar=13 %
143   \catcode35=6 % #
144   \catcode39=12 % '
145   \catcode44=12 % ,
146   \catcode45=12 % -
147   \catcode46=12 % .
148   \catcode58=12 % :
149   \catcode64=11 % @
150   \catcode123=1 % {
151   \catcode125=2 % }
152 \expandafter\let\expandafter\x\csname ver@hobsub-hyperref.sty\endcsname
153 \ifx\x\relax % plain-TeX, first loading
154 \else
155   \def\empty{}%
156   \ifx\x\empty % LaTeX, first loading,
157     % variable is initialized, but \ProvidesPackage not yet seen
158   \else
159     \expandafter\ifx\csname PackageInfo\endcsname\relax
160       \def\x#1#2{%
161         \immediate\write-1{Package #1 Info: #2.}%
162       }%
163     \else
164       \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
165     \fi
166     \x{hobsub-hyperref}{The package is already loaded}%
167   \aftergroup\endinput
168 \fi
169 \fi
170 \endgroup%

```

Package identification:

```
171 \begingroup\catcode61\catcode48\catcode32=10\relax%
172 \catcode13=5 % ^^M
173 \endlinechar=13 %
174 \catcode35=6 % #
175 \catcode39=12 % '
176 \catcode40=12 % (
177 \catcode41=12 % )
178 \catcode44=12 % ,
179 \catcode45=12 % -
180 \catcode46=12 % .
181 \catcode47=12 % /
182 \catcode58=12 % :
183 \catcode64=11 % @
184 \catcode91=12 % [
185 \catcode93=12 % ]
186 \catcode123=1 % {
187 \catcode125=2 % }
188 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
189 \def\x#1#2#3[#4]{\endgroup
190 \immediate\write-1{Package: #3 #4}%
191 \xdef#1{#4}%
192 }%
193 \else
194 \def\x#1#2[#3]{\endgroup
195 #2[#{#3}]%
196 \ifx#1\undefined
197 \xdef#1{#3}%
198 \fi
199 \ifx#1\relax
200 \xdef#1{#3}%
201 \fi
202 }%
203 \fi
204 \expandafter\x\csname ver@hobsub-hyperref.sty\endcsname
205 \ProvidesPackage{hobsub-hyperref}%
206 [2012/05/28 v1.13 Bundle oberdiek, subset hyperref (H0)]%
207 </hyperref>
```

## 2.2 Catcodes

```
208 <*package j generic j hyperref>
209 \begingroup\catcode61\catcode48\catcode32=10\relax%
210 \catcode13=5 % ^^M
211 \endlinechar=13 %
212 \catcode123=1 % {
213 \catcode125=2 % }
214 \catcode64=11 % @
215 \def\x{\endgroup
216 \expandafter\edef\csname HOBsub@AtEnd\endcsname{%
217 \endlinechar=\the\endlinechar\relax
218 \catcode13=\the\catcode13\relax
219 \catcode32=\the\catcode32\relax
220 \catcode35=\the\catcode35\relax
221 \catcode61=\the\catcode61\relax
222 \catcode64=\the\catcode64\relax
223 \catcode123=\the\catcode123\relax
224 \catcode125=\the\catcode125\relax
225 }%
226 }%
227 \x\catcode61\catcode48\catcode32=10\relax%
228 \catcode13=5 % ^^M
229 \endlinechar=13 %
```

```

230 \catcode35=6 % #
231 \catcode64=11 % @
232 \catcode123=1 % {
233 \catcode125=2 % }
234 \def\TMP@EnsureCode#1#2{%
235   \edef\HOBsub@AtEnd{%
236     \HOBsub@AtEnd
237     \catcode#1=\the\catcode#1\relax
238   }%
239   \catcode#1=#2\relax
240 }
241 \TMP@EnsureCode{39}{12}% '
242 \TMP@EnsureCode{40}{12}% (
243 \TMP@EnsureCode{41}{12}% )
244 \TMP@EnsureCode{45}{12}% -
245 \TMP@EnsureCode{46}{12}% .
246 \TMP@EnsureCode{47}{12}% /
247 \TMP@EnsureCode{58}{12}% :
248 \TMP@EnsureCode{60}{12}% <
249 \TMP@EnsureCode{62}{12}% >
250 \TMP@EnsureCode{96}{12}% `
251 \edef\HOBsub@AtEnd{\HOBsub@AtEnd\noexpand\endinput}
252 </package j generic j hyperref>

```

### 2.3 Package hobsup-hyperref loads hobsup-generic

```

253 <*hyperref>
254 \NeedsTeXFormat{LaTeX2e}
255 \RequirePackage{hobsup-generic}[2012/05/28]
256 </hyperref>

```

### 2.4 Preamble

```

257 <*preamble>
258 \begingroup\expandafter\expandafter\expandafter\endgroup
259 \expandafter\ifx\csname ver@hobsup.sty\endcsname\relax
260 \else
261   \expandafter\hobsup@GobbleRemainingPackage
262 \fi
263 </preamble>
264 <*package j preamble>

```

\HOBsub@OrgEndinput

```

265 \begingroup\expandafter\expandafter\expandafter\endgroup
266 \expandafter\ifx\csname HOBsub@OrgEndinput\endcsname\relax
267   \let\HOBsub@OrgEndinput\endinput
268 \fi

```

hobsup@GobbleRemainingPackage

```

269 \def\hobsup@GobbleRemainingPackage{%
270   \begingroup
271   \catcode92=14 % backslash: comment
272   \catcode32=14 % space: comment
273   \catcode35=14 % hash: comment
274   \catcode123=14 % left brace: comment
275   \catcode125=14 % right brace: comment
276   \catcode60=3 % less: math
277   \catcode62=4 % greater: align
278   \endlinechar=-1 %
279   \HOBsub@GobbleRemainingPackage
280 }

```

HOBsub@GobbleRemainingPackage

```

281 \catcode60=3 % less: dollar
282 \catcode62=4 % greater: align
283 \long\def\HOBsub@GobbleRemainingPackage#1<hobsub>{%
284   \endgroup
285 }
286 \catcode60=12 % less: other
287 \catcode62=12 % greater: other

```

\hobsub@StartPackage

```

288 \def\hobsub@StartPackage#1{%
289   \begingroup\expandafter\expandafter\expandafter\endgroup
290   \expandafter\ifx\csname ver@#1.sty\endcsname\relax
291     \let\HOBsub@OrgCurrName\@currname
292     \let\HOBsub@OrgCurrExt\@current
293     \csname @pushfilename\endcsname
294     \def\@currname{#1}%
295     \def\@current{sty}%
296     \expandafter\def\csname\@currname.\@current-h@k\endcsname{%
297       \let\endinput\hobsub@GobbleRemainingPackage
298       \def\hobsub@StopPackage{%
299         \let\hobsub@StopPackage\relax
300         \HOBsub@StopPackage
301 %       \hobsub@Info{hobsub}{Package `#1' loaded}%
302       }%
303       \hobsub@AddToFileList{#1.sty}%
304   \else
305     \hobsub@Info{hobsub}{Skipping package `#1' (already loaded)}%
306     \let\hobsub@StopPackage\relax
307     \expandafter\hobsub@GobbleRemainingPackage
308   \fi
309 }

```

\hobsub@StopPackage

```
310 \let\hobsub@StopPackage\relax
```

\hobsub@Info

```

311 \def\hobsub@Info#1#2{%
312   \begingroup\expandafter\expandafter\expandafter\endgroup
313   \expandafter\ifx\csname @PackageInfoNoLine\endcsname\relax
314     \immediate\write-1{Package #1 Info: #2.}%
315   \else
316     \let\hobsub@Info\@PackageInfoNoLine
317     \hobsub@Info{#1}{#2}%
318   \fi
319 }

```

\HOBsub@StopPackage

```

320 \def\HOBsub@StopPackage{%
321   \csname\@currname.\@current-h@k\endcsname
322   \let\endinput\HOBsub@OrgEndinput
323   \csname @popfilename\endcsname
324   \let\@currname\HOBsub@OrgCurrName
325   \let\@current\HOBsub@OrgCurrExt
326 }

```

\hobsub@AddToFileList

```

327 \begingroup\expandafter\expandafter\expandafter\endgroup
328 \expandafter\ifx\csname @addtofilelist\endcsname\relax
329   \def\hobsub@AddToFileList#1{%
330   \else
331     \def\hobsub@AddToFileList#1{%
332       \@addtofilelist{#1}%

```

```

333 }%
334 \fi

335 </package j preamble>

```

## 2.5 End of preamble for subset packages

Add the end marker for possible skipping of previous macros, if package `hobsub` was already loaded.

```

336 <*preamble>
337 \hobsub@GobbleRemainingPackage
338 <hobsub>

All macros of package hobsub are now available. Now we officially load the
package hobsub inline for proper package identification.

339 \hobsub@StartPackage{hobsub}%
340 \begingroup\expandafter\expandafter\expandafter\endgroup
341 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
342 \immediate\write-1{%
343   Package: hobsub 2012/05/28 v1.13 Construct package bundles (HO)%
344 }%
345 \else
346 \ProvidesPackage{hobsub}%
347 [2012/05/28 v1.13 Construct package bundles (HO)]%
348 \fi
349 </preamble>

350 <*post j preamble>
351 \endinput
352 <hobsub>
353 \hobsub@StopPackage
354 </post j preamble>

```

## 2.6 Package list

```

355 <infwarerr>\hobsub@StartPackage{infwarerr}
356 <ltxcmds>\hobsub@StartPackage{ltxcmds}
357 <ifluatex>\hobsub@StartPackage{ifluatex}
358 <ifvtex>\hobsub@StartPackage{ifvtex}
359 <intcalc>\hobsub@StartPackage{intcalc}
360 <ifpdf>\hobsub@StartPackage{ifpdf}
361 <etexcmds>\hobsub@StartPackage{etexcmds}
362 <kvsetkeys>\hobsub@StartPackage{kvsetkeys}
363 <kvdefinekeys>\hobsub@StartPackage{kvdefinekeys}
364 <*luatex-loader>
365 \ifluatex
366 \else
367 \expandafter\hobsub@GobbleRemainingPackage
368 \fi
369 \hobsub@StartPackage{luatex-loader}
370 </luatex-loader>
371 <pdftexcmds>\hobsub@StartPackage{pdftexcmds}
372 <pdfescape>\hobsub@StartPackage{pdfescape}
373 <bigintcalc>\hobsub@StartPackage{bigintcalc}
374 <bitset>\hobsub@StartPackage{bitset}
375 <uniquecounter>\hobsub@StartPackage{uniquecounter}
376 <letltxmacro>\hobsub@StartPackage{letltxmacro}
377 <hopatch>\hobsub@StartPackage{hopatch}
378 <xcolor-patch>\hobsub@StartPackage{xcolor-patch}
379 <atveryend>\hobsub@StartPackage{atveryend}
380 <atbegshi>\hobsub@StartPackage{atbegshi}
381 <refcount>\hobsub@StartPackage{refcount}
382 <hycolor>\hobsub@StartPackage{hycolor}

```

## 2.7 End of packages

```
383 <package j at end>\HOBsub@AtEnd%
```

## 3 Test

### 3.1 Catcode checks for loading

```
384 <*test1>
385 \catcode`\{=1 %
386 \catcode`\}=2 %
387 \catcode`\#=6 %
388 \catcode`\@=11 %
389 \expandafter\ifx\csname count@\endcsname\relax
390 \countdef\count@=255 %
391 \fi
392 \expandafter\ifx\csname @gobble\endcsname\relax
393 \long\def@gobble#1{}%
394 \fi
395 \expandafter\ifx\csname @firstofone\endcsname\relax
396 \long\def@firstofone#1{#1}%
397 \fi
398 \expandafter\ifx\csname loop\endcsname\relax
399 \expandafter@firstofone
400 \else
401 \expandafter@gobble
402 \fi
403 {%
404 \def\loop#1\repeat{%
405 \def\body{#1}%
406 \iterate
407 }%
408 \def\iterate{%
409 \body
410 \let\next\iterate
411 \else
412 \let\next\relax
413 \fi
414 \next
415 }%
416 \let\repeat=\fi
417 }%
418 \def\RestoreCatcodes{}
419 \count@=0 %
420 \loop
421 \edef\RestoreCatcodes{%
422 \RestoreCatcodes
423 \catcode\the\count@=\the\catcode\count@\relax
424 }%
425 \ifnum\count@<255 %
426 \advance\count@ 1 %
427 \repeat
428
429 \def\RangeCatcodeInvalid#1#2{%
430 \count@=#1\relax
431 \loop
432 \catcode\count@=15 %
433 \ifnum\count@<#2\relax
434 \advance\count@ 1 %
435 \repeat
436 }
437 \def\RangeCatcodeCheck#1#2#3{%
438 \count@=#1\relax
```

```

439 \loop
440   \ifnum#3=\catcode\count@
441   \else
442     \errmessage{%
443       Character \the\count@\space
444       with wrong catcode \the\catcode\count@\space
445       instead of \number#3%
446     }%
447   \fi
448 \ifnum\count@<#2\relax
449   \advance\count@ 1 %
450 \repeat
451 }
452 \def\space{ }
453 \expandafter\ifx\csname LoadCommand\endcsname\relax
454 \def\LoadCommand{\input hobsup.sty\relax}%
455 \fi
456 \def\Test{%
457   \RangeCatcodeInvalid{0}{47}%
458   \RangeCatcodeInvalid{58}{64}%
459   \RangeCatcodeInvalid{91}{96}%
460   \RangeCatcodeInvalid{123}{255}%
461   \catcode`\@=12 %
462   \catcode`\=0 %
463   \catcode`\%=14 %
464   \LoadCommand
465   \RangeCatcodeCheck{0}{36}{15}%
466   \RangeCatcodeCheck{37}{37}{14}%
467   \RangeCatcodeCheck{38}{47}{15}%
468   \RangeCatcodeCheck{48}{57}{12}%
469   \RangeCatcodeCheck{58}{63}{15}%
470   \RangeCatcodeCheck{64}{64}{12}%
471   \RangeCatcodeCheck{65}{90}{11}%
472   \RangeCatcodeCheck{91}{91}{15}%
473   \RangeCatcodeCheck{92}{92}{0}%
474   \RangeCatcodeCheck{93}{96}{15}%
475   \RangeCatcodeCheck{97}{122}{11}%
476   \RangeCatcodeCheck{123}{255}{15}%
477   \RestoreCatcodes
478 }
479 \Test
480 \csname @@end\endcsname
481 \end
482 </test1)

```

## 4 Installation

### 4.1 Download

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

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

[CTAN:macros/latex/contrib/oberdiek/hobsup.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](#)

<sup>1</sup>[ftp://ftp.ctan.org/tex-archive/](http://ftp.ctan.org/tex-archive/)

*TDS* refers to the standard “A Directory Structure for  $\TeX$  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
```

**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 hobsub.dtx
```

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

```
hobsub.sty           → tex/generic/oberdiek/hobsub.sty
hobsub-generic.sty  → tex/generic/oberdiek/hobsub-generic.sty
hobsub-hyperref.sty → tex/generic/oberdiek/hobsub-hyperref.sty
hobsub.pdf          → doc/latex/oberdiek/hobsub.pdf
test/hobsub-test1.tex → doc/latex/oberdiek/test/hobsub-test1.tex
hobsub.dtx          → source/latex/oberdiek/hobsub.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 (te $\TeX$ , mi $\TeX$ , ...) relies on file name databases, you must refresh these. For example, te $\TeX$  users run `texhash` or `mktexlsr`.

## 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 hobsub.pdf unpack_files output .
```

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

**plain  $\TeX$ :** Run `docstrip` and extract the files.

**$\LaTeX$ :** Generate the documentation.

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

```
latex \let\install=y\input{hobsub.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 hobsub.dtx
makeindex -s gind.ist hobsub.idx
pdflatex hobsub.dtx
makeindex -s gind.ist hobsub.idx
pdflatex hobsub.dtx
```

## 5 Catalogue

The following XML file can be used as source for the [T<sub>E</sub>X Catalogue](#). The elements `caption` and `description` are imported from the original XML file from the Catalogue. The name of the XML file in the Catalogue is `hobsub.xml`.

```
483 (*catalogue)
484 <?xml version='1.0' encoding='us-ascii'?>
485 <!DOCTYPE entry SYSTEM 'catalogue.dtd'>
486 <entry datestamp='$Date$' modifier='$Author$' id='hobsub'>
487   <name>hobsub</name>
488   <caption>Construct package bundles</caption>
489   <authorref id='auth:oberdiek' />
490   <copyright owner='Heiko Oberdiek' year='2011,2012' />
491   <license type='lppl1.3' />
492   <version number='1.13' />
493   <description>
494     The package offers a means of loading a bundle of (other) packages
495     &#x201C;at once&#x201D;, offering a means of reducing the amount of log output
496     as the packages are loaded, and an aide-memoire for complicated
497     requirements. In some ways, loading a package via hobsub behaves
498     much like loading the package in the normal way; for example,
499     reloads are suppressed, as in LaTeX proper, and correct
500     <tt>\listfiles</tt> output is produced.
501     <p />
502     Examples provided are a &#x2018;generic&#x2019; bundle (comprising the author&#x2019;s
503     packages that can be used with Plain TeX) and a &#x2018;hyperref&#x2019; bundle
504     (comprising packages useful when generating hypertext output with
505     LaTeX).
506     <p />
507     The package is part of the <xref refid='oberdiek'>oberdiek</xref> bundle.
508   </description>
509   <documentation details='Package documentation'
510     href='ctan:/macros/latex/contrib/oberdiek/hobsub.pdf' />
511   <ctan file='true' path='/macros/latex/contrib/oberdiek/hobsub.dtx' />
512   <miktex location='oberdiek' />
513   <texlive location='oberdiek' />
514   <install path='/macros/latex/contrib/oberdiek/oberdiek.tds.zip' />
515 </entry>
516 </catalogue>
```

## 6 References

- [1] Will Robertson, *[ltx] quietening luatex console output*; mailing list `luatex-dev at tug.org`, 2011-01-29; <http://tug.org/pipermail/luatex-dev/2011-January/001031.html>.

## 7 History

**[2011/01/30 v1.0]**

- First public version.

**[2011/04/16 v1.1]**

- Package updates.

**[2011/04/17 v1.2]**

- White spaces at line begins are removed or reduced in the generated collection packages.

**[2011/04/18 v1.3]**

- Package updates.

**[2011/04/23 v1.4]**

- Package updates.

**[2011/06/24 v1.5]**

- Package updates.

**[2011/06/30 v1.6]**

- Package updates.

**[2011/07/01 v1.7]**

- Package updates.

**[2011/07/28 v1.8]**

- Package updates.

**[2011/08/22 v1.9]**

- Package updates.

**[2011/10/16 v1.10]**

- Package updates.

**[2011/11/29 v1.11]**

- Package updates.

[2012/04/25 v1.12]

- Package updates.

[2012/05/28 v1.13]

- Package updates.

## 8 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	
<code>\#</code> .....	387
<code>\%</code> .....	463
<code>\@</code> .....	388, 461
<code>\@PackageInfoNoLine</code> .....	316
<code>\@addtofilelist</code> .....	332
<code>\@currrent</code> .....	292, 295, 296, 321, 325
<code>\@currname</code> .....	291, 294, 296, 321, 324
<code>\@firstofone</code> .....	396, 399
<code>\@gobble</code> .....	393, 401
<code>\@undefined</code> .....	58, 127, 196
<code>\@</code> .....	462
<code>\{</code> .....	385
<code>\}</code> .....	386
<b>A</b>	
<code>\advance</code> .....	426, 434, 449
<code>\aftergroup</code> .....	29, 98, 167
<b>B</b>	
<code>\body</code> .....	405, 409
<b>C</b>	
<code>\catcode</code> .....	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, 71, 72, 74, 75, 76, 77, 78, 79, 80, 81, 82, 102, 103, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 140, 141, 143, 144, 145, 146, 147, 148, 149, 150, 151, 171, 172, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 209, 210, 212, 213, 214, 218, 219, 220, 221, 222, 223, 224, 227, 228, 230, 231, 232, 233, 237, 239, 271, 272, 273, 274, 275, 276, 277, 281, 282, 286, 287, 385, 386, 387, 388, 423, 432, 440, 444, 461, 462, 463
<code>\count@</code> .....	390, 419, 423, 425, 426, 430, 432, 433, 434, 438, 440, 443, 444, 448, 449
<code>\countdef</code> .....	390
<code>\csname</code> .....	14, 21, 50, 66, 83, 90, 119, 135, 152, 159, 188, 204, 216, 259, 266, 290,
	293, 296, 313, 321, 323, 328, 341, 389, 392, 395, 398, 453, 480
<b>E</b>	
<code>\empty</code> .....	17, 18, 86, 87, 155, 156
<code>\end</code> .....	481
<code>\endcsname</code> .....	14, 21, 50, 66, 83, 90, 119, 135, 152, 159, 188, 204, 216, 259, 266, 290, 293, 296, 313, 321, 323, 328, 341, 389, 392, 395, 398, 453, 480
<code>\endinput</code> .....	29, 98, 167, 251, 267, 297, 322, 351
<code>\endlinchar</code> .....	4, 35, 73, 104, 142, 173, 211, 217, 229, 278
<code>\errmessage</code> .....	442
<b>H</b>	
<code>\hobsub@AddToFileList</code> .....	303, <u>327</u>
<code>\HOBsub@AtEnd</code> .....	235, 236, 251, 383
<code>\HOBsub@GobbleRemainingPackage</code> ..	279, <u>281</u>
<code>\hobsub@GobbleRemainingPackage</code> ..	261, <u>269</u> , 297, 307, 337, 367
<code>\hobsub@Info</code> .....	301, 305, <u>311</u>
<code>\HOBsub@OrgCurrExt</code> .....	292, 325
<code>\HOBsub@OrgCurrName</code> .....	291, 324
<code>\HOBsub@OrgEndinput</code> .....	<u>265</u> , <u>322</u>
<code>\hobsub@StartPackage</code> .....	288, 339, 355, 356, 357, 358, 359, 360, 361, 362, 363, 369, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382
<code>\HOBsub@StopPackage</code> .....	300, <u>320</u>
<code>\hobsub@StopPackage</code> .....	298, 299, 306, <u>310</u> , 353
<b>I</b>	
<code>\ifluatex</code> .....	365
<code>\ifnum</code> .....	425, 433, 440, 448
<code>\ifx</code> .....	15, 18, 21, 50, 58, 61, 84, 87, 90, 119, 127, 130, 153, 156, 159, 188, 196, 199, 259, 266, 290, 313, 328, 341, 389, 392, 395, 398, 453
<code>\immediate</code> .....	23, 52, 92, 121, 161, 190, 314, 342
<code>\input</code> .....	454

<code>\iterate</code> .....	406, 408, 410	<code>\repeat</code> .....	404, 416, 427, 435, 450
<b>L</b>		<code>\RequirePackage</code> .....	255
<code>\listfiles</code> .....	500	<code>\RestoreCatcodes</code> ..	418, 421, 422, 477
<code>\LoadCommand</code> .....	454, 464	<b>S</b>	
<code>\loop</code> .....	404, 420, 431, 439	<code>\space</code> .....	443, 444, 452
<b>N</b>		<b>T</b>	
<code>\NeedsTeXFormat</code> .....	254	<code>\Test</code> .....	456, 479
<code>\next</code> .....	410, 412, 414	<code>\the</code> .....	217, 218, 219, 220, 221, 222, 223, 224, 237, 423, 443, 444
<code>\number</code> .....	445	<code>\TMP@EnsureCode</code>	234, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250
<b>P</b>		<b>W</b>	
<code>\PackageInfo</code> .....	26, 95, 164	<code>\write</code>	23, 52, 92, 121, 161, 190, 314, 342
<code>\ProvidesPackage</code> .....	..... 19, 67, 88, 136, 157, 205, 346		
<b>R</b>		<b>X</b>	
<code>\RangeCatcodeCheck</code> .....	. 437, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476	<code>\x</code> .....	14, 15, 18, 22, 26, 28, 51, 56, 66, 83, 84, 87, 91, 95, 97, 120, 125, 135, 152, 153, 156, 160, 164, 166, 189, 194, 204, 215, 227
<code>\RangeCatcodeInvalid</code> .....	..... 429, 457, 458, 459, 460		