

The nag package^{*}

Ulrich Michael Schwarz[†]

March 21, 2007

Abstract

Old habits die hard. All the same, there are commands, classes and packages which are outdated and superseded. `nag` provides routines to warn the user about the use of those. As an example, we provide an extension that detects many of the “sins” described in `l2tabu`.

Contents

1	User-side considerations.	2
1.1	Installation.	2
1.2	Usage.	2
1.3	<code>nag-l2tabu.cfg</code>	2
1.4	<code>nag-orthodox.cfg</code>	4
1.5	<code>nag-abort.cfg</code>	5
2	Author-side considerations and implementation.	5
2.1	Low-level tools.	6
2.2	Obsoletifying commands.	7
2.3	Obsoletifying packages and classes.	7
2.4	Common float errors and no-nos.	8
3	Switch vs. Environment	10
4	Compatibility issues	13
4.1	The caption package	13
4.2	The subfig package	13
4.3	The float package	17
5	Loading extensions	17

^{*}This document corresponds to `nag` 0.53, dated 2007/03/20.

[†]`ulmi@absatzen.de`

1 User-side considerations.

1.1 Installation.

Process `nag.ins` with \TeX to obtain some files: `nag.sty` and `nag-l2tabu.cfg` et al. must go to a place where \TeX will find them, like the local `TEXMF` tree. (If all else fails and you need it to work *right now*, having them in the same directory as the \TeX file you want to use them on may work under many circumstances.) You can, as usual, run \TeX on `nag.dtx` to obtain this documentation, including the implementation docs. (This is recommended if you plan to extend `nag` to handle your own packages.) `nagdemo.tex` is a horrible document that will show you many of the warnings that `nag` can generate.

1.2 Usage.

Add the following to the beginning your main document (Comments and `\listfiles` can be safely left before it, though):

```
\RequirePackage[l2tabu, orthodox]{nag}
```

This will check for many common mistakes, and give some hints on what to use instead. However, you should always refer to `l2tabu` for a more detailed explanation of the whats and whys: it gives more information than can be possibly pressed into two lines of error message. Orthodox checks for pitfalls that are not technically incorrect. If you know what you're doing, omit `orthodox`.

1.3 `nag-l2tabu.cfg`

In a nutshell, `nag-l2tabu.cfg` detects the following:

- Usage of the 2.09-style font commands `\it`, `\bf`, `\rm`, `\sc`, `\sl`, `\tt` and `\cal`.
- Usage of `\centerline`.
- Usage of the outdated packages `epsfig`, `psfig`, `epsf`, `doublespace`, `fancyheadings`, `scrpage`, `umlaut`, `isolatin`, `isolatin1`, `tlenc`, `caption2`, `psfonts`, `mathptm`, `times`, `palatino`, `mathppl`, `euler` and `utopia`, and of the outdated class `scrlltr`.
- Figures and tables without caption (this is not technically in `l2tabu`, but the people who have floats without captions tend to ask “Why is \TeX moving my pictures away from where I put them?”), labels within floats that do not reference the caption, and usage of the `center` environment within floats.

It is beyond the possibilities of this package to detect things like use of \TeX assignment syntax, or direct change of paper parameters, or reliable detection of user-issued `\sloppy`. Outdated maths environments are not treated because those can already be handled by Harald Harders' `onlyamsmath`.

Be warned, hence, that this package will possibly balk at legitimate use, and not find illegitimate use in all cases. It is a tool, not a replacement for study of `l2tabu`.

```

1 < *l2tabunag>
2 \ProvidesFile{nag-l2tabu.cfg}
3      [2007/03/21 v1.8b l2tabu rules for nag.sty (ulmi)]
4 %%
5 %% The sins.
6 %%
7 %% Section numbers refer to l2tabuen 1.7 revised/enlarged dated 2004OCT24
8 %% \S 1.1
9 \ObsoletePackage{a4wide}{the \lq a4paper\rq\space class option}
10 \ObsoletePackage{a4}{the \lq a4paper\rq\space class option}
11 %% \S 1.2--1.5 cannot reasonably be checked programmatically
12 %% \S 1.6 is handled by H.Harders' onlyamsmath package
13 %% \S 1.7 cannot reasonably be checked programmatically
14 %% \S 1.8 \sloppy is called by parbox, among others, and would
15 %% give many spurious warnings.
16 %% \S 2.1.1
17 \ObsoleteCS[an old LaTeX 2.09 command]{bf}
18      {\protect\bfseries\space or \protect\textbf}
19 \ObsoleteCS[an old LaTeX 2.09 command]{it}
20      {\protect\itshape\space or \protect\textit}
21 \ObsoleteCS[an old LaTeX 2.09 command]{rm}
22      {\protect\rmfamily\space or \protect\textrm}
23 \ObsoleteCS[an old LaTeX 2.09 command]{sc}
24      {\protect\scshape\space or \protect\textsc}
25 \ObsoleteCS[an old LaTeX 2.09 command]{sf}
26      {\protect\sffamily\space or \protect\textsf}
27 \ObsoleteCS[an old LaTeX 2.09 command]{sl}
28      {\protect\slshape\space or \protect\textsl}
29 \ObsoleteCS[an old LaTeX 2.09 command]{tt}
30      {\protect\ttfamily\space or \protect\texttt}
31 \ObsoleteCS[an old LaTeX 2.09 command]{cal}
32      {\protect\mathcal}% Hmm, this is not in l2tabu?
33 %% \S 2.1.2
34 %% Gone with 1.8 because this never worked for the kernel \frac anyway.
35 %% \ObsoleteCS[TeX]{over}{\protect\frac}
36 %% \ObsoleteCS[TeX]{choose}{\protect\frac\space or amsmath's \protect\binom}
37 %% \S 2.1.3
38 \ObsoleteCS[TeX]{centerline}{\protect\centering\space or center environment}
39 %% \S 2.2.1
40 \ObsoleteClass{scrlettr}{the scrlettr2 package}
41 %% \S 2.2.2
42 \ObsoletePackage{epsf}{the graphicx package}
43 \ObsoletePackage{psfig}{the graphicx package}
44 \ObsoletePackage[deprecated]{epsfig}{the graphicx package directly}
45 %% \S 2.2.3
46 \ObsoletePackage{doublespace}{the setspace package}
47 %% \S 2.2.4
48 \ObsoletePackage{fancyheadings}{the fancyhdr or scrpage2 packages}
49 \ObsoletePackage{scrpage}{the scrpage2 package}
50 %% \S 2.2.5

```

```

51\ObsoletePackage{isolatin}{the inputenc package with option latin1}
52\ObsoletePackage{umlaut}{the inputenc package with suitable option
53  (latin1, utf8 ...)}
54\ObsoletePackage{isolatin1}{the inputenc package with option latin1}
55%% \S 2.2.6
56\ObsoletePackage{tlenc}{the fontenc package with option T1}
57%% \S 2.2.7 we don't check for bst yet.
58%% (This is in l2tabu 1.8)
59\ObsoletePackage{caption2}{the caption package v3.0 or later}
60%% \S 2.3.1-3
61\ObsoletePackage{times}
62      {the mathptmx, helvet (option scaled=.9), courier packages}
63\ObsoletePackage{pslatex}
64      {the mathptmx, helvet (option scaled=.9), courier packages}
65\ObsoletePackage{mathptm}
66      {the mathptmx package}
67%% \S 2.3.4-5
68\ObsoletePackage{palatino}
69      {the mathpazo, helvet (option scaled=.95), courier packages}
70\ObsoletePackage{mathpple}{the mathpazo package}
71%% \S 2.3.6 can't be checked
72%% \S 2.3.7
73\ObsoletePackage{euler}{the eulervm package}
74\ObsoletePackage{utopia}{the fourier package}
75%% \S 3.1
76\NagDeclareFloat{figure}\NagDeclareFloat{table}%
77\g@addto@macro\nag@labels{,label}%
78%% The latter two are used by KOMA-Script, the last by hypcap.
79% \changes{0.53}{2007/03/21}{hypcap support. (H.G.Krauth"auser)}
80% \changes{0.53}{2007/03/21}{topcapt support.}
81\g@addto@macro\nag@captions{,caption,captionabove,captionbelow,hc@caption,topcaption}%
82
83%% \S 3.2
84\NotAnEnvironment{appendix}%
85%% In the same vein:
86\@for\sectioning:=frontmatter,mainmatter,backmatter\do{%
87  \expandafter\NotAnEnvironment\expandafter{\sectioning}%
88}
89%% \S 3.3 handled by onlyamsmath.
90%% \S 3.4 -- nothing to be done --
91\l2tabunag

```

1.4 nag-orthodox.cfg

nag-orthodox.cfg warns about usage that is not technically incorrect, but will mostly do things an unwary user may not expect. This includes in particular the usage of font size and style switches as environments (line spacing will be off if the environment does not contain a trailing \par, spurious spaces might occur since the switches don't ignore spaces), and, conversely, the usage of center etc. environments as unclosed

switches. (Detection of the latter might still be somewhat brittle.)

```

92 (*orthodoxnag)
93 \ProvidesFile{nag-orthodox.cfg}
94     [2006/04/19 v1.8 strict rules for nag.sty (ulmi)]
95 \@for\fontcmd:=tiny,small,footnotesize,normalsize,large,Large,%
96     LARGE,huge,Huge\do{%
97     \expandafter\NotAnEnvironment\expandafter{\fontcmd}%
98 }%
99 \@for\fontcmd:=sffamily,rmfamily,ttfamily,%
100     bfseries,mdseries,scshape,%
101     itshape,upshape\do{%
102     \expandafter\NotAnEnvironment\expandafter{\fontcmd}%
103 }%
104 \@for\justsw:=centering,raggedleft,raggedright,%
105     RaggedLeft,RaggedRight\do{%
106     \expandafter\NotAnEnvironment\expandafter{\justsw}%
107 }
108 \@for\justenv:=center,flushleft,flushright\do{%
109     \expandafter\NotASwitch\expandafter{\justenv}%
110 }
111 \</orthodoxnag>

```

1.5 nag-abort.cfg

Requesting this nag file will turn all complaints into errors.

```

112 (*abortnag)
113 \ProvidesFile{nag-abort.cfg}
114     [2005/05/08 v0.1 treat complaints as errors (ulmi)]
115 \renewcommand\nag@warn[1]{%
116     \addtocounter{nag@sins}{1}%
117     \PackageError{nag}{#1}{#1}%
118 }
119 \renewcommand\nag@warnNoLine[1]{%
120     \addtocounter{nag@sins}{1}%
121     \PackageError{nag}{#1}{#1}%
122 }
123 \</abortnag>

<*nag>

```

2 Author-side considerations and implementation.

If you are a package or class author and want to extend the range of nag (or prevent nag from criticizing your macros), please see the description below, in sections 2.2 and following. It is probably wise to group new rules in a separate nag file: users can request nag files by passing their name as a package parameter, as shown above for the example of l2tabu.

2.1 Low-level tools.

Identify ourselves.

```
124 \NeedsTeXFormat{LaTeX2e}
125 \ProvidesPackage{nag}[2007/03/20 0.53 warning about old commands (ulmi)]
126 \let\@xa\expandafter
127 \let\@nx\noexpand
```

First of all, two counters we need. The first is used to generate running numbers for replacement macros, the latter is stepped for each complaint we have, so that the user gets a frighteningly high number, showing how sinful he or she is.

```
128 \newcounter{nag@c}
129 \newcounter{nag@sins}
```

`\nag@prepend` `\nag@prepend{<cs>}{<something>}`: Prepend *<something>* to the macro definition of *<cs>*.

In reality, we do call indirection: save old macro away, redefine macro to do the something, call old macro. (With thanks to Juergen Goebel, Heiko Oberdiek and Rolf Niepraschk (savesym))

Don't define the macro if it's not there. This confuses caption, which loads ragged2e AtBeginDocument, at which point, RaggedLeft et al. were already defined by us. ... but *do* log a message.

```
130 \newcommand\nag@prepend[2]{%
131   \@ifundefined{#1}{%
132     % if it doesn't exist, don't do anything.
133     \PackageInfo{nag}{%
134       Command \@backslashchar#1\space not defined, skipping amendment
135     }
136   }{%
137     \@xa\let
138     \csname nag@@#1@\thenag@c\@xa\endcsname
139     \csname #1\endcsname
140     \@xa\nag@pr@p@nd\csname #1\@xa\endcsname
141     \csname nag@@#1@\thenag@c\endcsname{#2}%
142     \stepcounter{nag@c}%
143   }
144 }
145 \newcommand\nag@pr@p@nd[3]{%
146   \def#1{#3#2}%
147 }
```

`\nag@warn` All complaints to the user run through one of these two macros, with or without source line.

```
148 \newcommand\nag@warn{%
149   \addtocounter{nag@sins}{1}%
150   \PackageWarning{nag}%
151 }
152 \newcommand\nag@warnNoLine{%
153   \addtocounter{nag@sins}{1}%
154 }
```

```

154 \PackageWarningNoLine{nag}%
155 }

```

2.2 Obsoletifying commands.

(No, I do not think that is a proper word either.)

`\ObsoleteCS` Usage: `\ObsoleteCS[⟨reason⟩]{⟨CS⟩}{⟨suggestions⟩}` Mark `\⟨CS⟩` as obsolete. `⟨reason⟩` defaults to obsolete. When the macro is used anyway, the following warning is logged: Command `\⟨CS⟩` is `⟨reason⟩`. Use `⟨suggestions⟩` instead.

```

156 \newcommand\ObsoleteCS[3][obsolete]{%
157   \AtBeginDocument{%
158     \nag@prepend{#2}{%
159       \nag@warn{%
160 Command \backslashchar#2 is #1.
161 \MessageBreak
162 Use #3 instead}%
163   }%
164 }%
165 }

```

2.3 Obsoletifying packages and classes.

Checking for packages and classes is done by looking for `ver@foo.sty`, which holds the version information that is also displayed by `\listfiles`. This means that we're out of luck if fontenc ever becomes obsolete, because that won't be detected.

First, define a macro to check if a control sequence is defined. Unlike `\@ifundefined`, this will not define the control sequence to `\relax`, but the arguments will be executed in a group. For our purposes, this doesn't matter, because we only give a warning (and `\addtocounter` already is `\global`).

```

166 \newcommand\nag@ifcsname[3]{%
167   \begingroup\@ifundefined{#1}{#3}{#2}\endgroup
168 }

```

Just because we can, use $\epsilon\text{T}_{\text{X}}$ `\ifcsname` if we can. This bootstrapping gives me a big grin... Note we add an extra group for compatibility with the non- ϵ case.

```

169 \nag@ifcsname{ifcsname}{%
170   \renewcommand*\nag@ifcsname[3]{%
171     \begingroup
172       % assume it won't be there.
173       \let\tmp@a\@secondoftwo
174       \ifcsname #1\endcsname
175       % It still might be relax from some other test. Thanks to J"org
176       % Sommer for finding this bug.
177       \expandafter\ifx\csname #1\endcsname\relax
178       \else
179         % it's there after all
180         \let\tmp@a\@firstoftwo

```

```

181     \fi
182     \fi
183     \tmp@a{#2}{#3}%
184     \endgroup
185 }%

```

This way of escaping the grouping gives me an even bigger grin.

```

186 \global\let\nag@ifcsname\nag@ifcsname
187 {}

```

\ObsoletePackage Usage: `\ObsoletePackage[⟨reason⟩]{⟨package⟩}{⟨alternative⟩}`. Mark `⟨package⟩` as obsolete. `⟨reason⟩` defaults to `obsolete`. If the `⟨package⟩` is used anyway, at the end of the compilation, the following warning will be displayed:
Package `⟨package⟩` is `⟨reason⟩`. Use `⟨alternative⟩` instead.

```

188 \newcommand\ObsoletePackage[3][obsolete]{%
189   \AtEndDocument{%
190     |\@clsextension| is onlypreamble, for some reason.
191     \nag@ifcsname{ver@#2.sty}{%
192       \nag@warnNoLine{%
193         Package #2 is #1.\MessageBreak
194         Use #3 instead}%
195       }{}%
196     }%
197 }

```

\ObsoleteClass Usage: `\ObsoleteClass[⟨reason⟩]{⟨class⟩}{⟨alternative⟩}`. Mark `⟨class⟩` as obsolete. `⟨reason⟩` defaults to `obsolete`. If the `⟨class⟩` is used anyway, at the end of the compilation, the following warning will be displayed:
Class `⟨class⟩` is `⟨reason⟩`. Use `⟨alternative⟩` instead.

```

198 \newcommand\ObsoleteClass[3][obsolete]{%
199   \AtEndDocument{%
200     |\@clsextension| is onlypreamble, for some reason.
201     \nag@ifcsname{ver@#2.cls}{%
202       \nag@warnNoLine{%
203         Class #2 is #1.\MessageBreak
204         Use #3 instead}%
205       }{}%
206     }%
207 }

```

2.4 Common float errors and no-nos.

We do the following:

- check for presence of a caption
- check for absence of the center environment
- check that a label comes only after a caption

First of all, we define two ifs to memorize whether we have a label and/or a caption in the float already. Package writers may want to set these manually behind nag's back. In this way, they can suppress possible warnings if they know what they're doing – we only check at the end of the float environment, which gives them plenty of time to call `\csname nag@haslabeltrue\endcsname` et al. (Thanks to Markus Kohm for pointing out this need.)

```
208 \newif\ifnag@haslabel
209 \newif\ifnag@hascaption
```

Now, to the work proper: first, add the endcenter check, then, prepare to set up the caption/label checks locally to the floats, and finally, add the code that generates the warning.

```
210 \newcommand\nag@hackfloat[1]{%
211   \nag@prepend{#1}{%
212     \nag@prepend{endcenter}{%
213       \nag@warn%
214       {\lq center\rq\space environment in #1.\MessageBreak
215        Maybe you want \protect\centering\space instead}
216     }%
217   }
218   \nag@prepend{#1}{%
```

Add checks to all macros named by `\nag@labels` and `\nag@captions`, respectively. Scoping of presence-of-caption information: Well, maybe I should do it the way the kernel does, which means a label is just as local as `\refstepcounter's` `@currentlabel` information as of v0.4. I think we can leave captions global.

Note: we cannot exchange the order of the for loops here: if a cs generates both a label and a caption, it shouldn't get complained about.

```
219   \@for\labelprovider:=\nag@labels\do{%
220     \ifx\labelprovider@empty\else
221       \nag@prepend{\labelprovider}{%
222         {\nag@captioncheck\nag@haslabeltrue}
223       \fi
224     }%
225   \@for\captionprovider:=\nag@captions\do{%
226     \ifx\captionprovider@empty\else
227       \nag@prepend{\captionprovider}{\global\nag@hascaptiontrue}%
228     \fi
229   }%
230   \global\nag@haslabelfalse\global\nag@hascaptionfalse
231 }%
232 \nag@prepend{end#1}{%
233   \ifnag@hascaption\relax\else
234     \nag@warn%
235 {#1 with no \protect\caption}%
236   \fi
237 }%
238 }
239 \newcommand\nag@captioncheck{%
```

```

240 \ifnag@hascaption\else
241 \nag@warn{\protect\label\space in float, but not after
242 \protect\caption}%
243 \fi
244 }

```

Define the lists of commands that are floats, generate labels, and generate captions, respectively. We don't start with defined floats (that is for nag-l2tabu.cfg to set up). Since v0.52, we handle an empty name, so the lists may be empty. Also, no labels and captions are provided by default since v0.52. This has been moved to nag-l2tabu.cfg. See also \NagDeclareFloat, which is the user-level wrapper for new floats. Since there are no packages to define new caption or label commands on an user level, there is no wrapper for those.

```

245 \def\nag@floats{}
246 \def\nag@labels{}
247 \def\nag@captions{}

```

We call the above for each float environment named via \nag@floats:

```

248 \newcommand\nag@floatsetup{%
249 \@for\flo:=\nag@floats\do{%
250 \ifx\flo\@empty\else
251 \xa\nag@hackfloat\@xa{\flo}%
252 \fi
253 }%
254 }

```

but only after all other packages get their chance to add to the list:

```

255 \AtBeginDocument{%
256 \nag@floatsetup
257 }

```

At the very end, we will display a running total of complaints.

```

258 \AtBeginDocument{%
259 \AtEndDocument{%
260 \ifnum\value{nag@sins}>0%
261 \PackageWarningNoLine{nag}{\arabic{nag@sins} complaints
262 in total}%
263 \else
264 \typeout{No complaints by nag.}%
265 \fi
266 }%
267 }

```

3 Switch vs. Environment

People often use switches as environments and vice versa. This is dangerous in because it tends to *almost* work. (Consider font size commands in particular, but also \centering vs. center environment.) As usual, “it’s not an error if you know what

you’re doing”. In particular, it is perfectly valid code to use the `\foo...\endfoo` syntax. So, `\NotASwitch` needs to trace the calls to `\foo` and see if they match with corresponding `\endfoos` with its own stack. This might still be brittle. Fortunately, it is currently only needed for `nag-orthodox`, where it checks for the justification environments.

First of all, a helper macro we hinge upon:

```
268 \DeclareRobustCommand\nag@ifCurrentEnvironment[3]{%
269   \bgroup
270   \def\tmp@a{#1}%
271   \ifx\@currenenv\tmp@a
272     #2%
273   \else
274     #3%
275   \fi
276   \egroup
277 }
```

And now, the two variations there are:

`\NotAnEnvironment` Usage: `\NotAnEnvironment{<command>}` Issue an error if the user calls `\begin{command}` and not `\command` directly.

```
278 \newcommand\NotAnEnvironment[1]{%
279   \AtBeginDocument{%
280     \nag@prepend{#1}{%
281       \nag@ifCurrentEnvironment{#1}{%
282         \nag@warn{%
283           There is no environment ‘‘#1’’. \MessageBreak
284           Maybe you want a grouped \@backslashchar#1
285         }%
286       }% OK case.
287     }%
288   }%
289 }
```

`\NotASwitch` is a bit more involved:

`\NotASwitch` Usage: `\NotASwitch{<command>}` Issue an error if the user calls `\command` and not `\begin{command}` and mis-nests calls or doesn’t call `\endcommand` at all.

```
291 % we need to maintain a stack of environments that are used in the
292 % \foo...\endfoo way.
293 \newcommand\nag@envstack{\relax}
294
295 \DeclareRobustCommand\nag@beginenv[1]{%
296   % push a begin-entry onto the stack. Form is
297   % |\foo{lineno}| for environment foo.
298   \xdef\nag@envstack{%
299     \@nx{%
300       \xa\@nx\csname #1\endcsname
301       \@nx{\the\inputlineno}\@nx}%
302 }
```

```

302 \@nx}%
303 \@xa\@nx\nag@envstack
304 }%
305 }
306 \DeclareRobustCommand\nag@endenv[1]{%
307 % extract the first entry.
308 \@xa\nag@endenv\nag@envstack\@nil #1\@nil
309 }
310
311 \def\nag@endenv#1#2\@nil #3\@nil{%
312 \def\tmp@a{#1}%
313 \def\tmp@b{\relax}%
314 \ifx\tmp@a\tmp@b
315 % This was the end-of-stack flag.
316 \nag@warn{'\@backslashchar end#3'} without matching
317 '\@backslashchar #3'}
318 \else
319 % We may assume this is a proper entry. See if the begin-token on
320 % the stack matches what |\nag@endenv| was passed.
321 \@xa\ifx\csname #3\@xa\endcsname\@firstoftwo #1%
322 %OK case, just pop the entry.
323 \gdef\nag@envstack{#2}%
324 \else
325 % error case
326 \nag@warn{%
327 You cannot close '\@xa\string\@firstoftwo #1' on line
328 \@secondoftwo #1 with '\@backslashchar end#3'%
329 }%
330 % leave it on the stack. Some case of misnesting will always cause
331 % horrible amounts of follow-up errors. Also, scare them!
332 \fi
333 \fi
334 }

```

At the end, we complain about all the entries that are still on the stack.

```

335 \AtEndDocument{%
336 \@xa\@tfor\@xa\looseends\@xa:\@xa=\nag@envstack\do{%
337 \@xa\ifx\looseends\relax\else
338 \nag@warnNoLine{Unmatched
339 '\@xa\@xa\@xa\string\@xa\@firstoftwo\looseends'
340 command on line
341 \@xa\@xa\@xa\string\@xa\@secondoftwo\looseends%
342 }%
343 \fi
344 }%
345 }

```

Now, the user-side command is easy.

```

346 \newcommand\NotASwitch[1]{%
347 \AtBeginDocument{%

```

```

348 \nag@prepend{#1}{%
349 \nag@beginenv{#1}%
350 }%
351 \nag@prepend{end#1}{%
352 \nag@endenv{#1}%
353 }%
354 }%
355 }

```

4 Compatibility issues

4.1 The caption package

Axel Sommerfeldt's caption package loads the ragged2e package AtBeginDocument (regardless of whether it is needed). This is too late for us to amend the `\RaggedFoo` commands with `\NotAnEnvironment`. Since v0.51 of nag, they will then be skipped (with information in the log). Earlier versions would fail because by time ragged2e was loaded, the commands were already defined by the amendment process. To make sure the commands *are* amended, load ragged2e explicitly yourself.

4.2 The subfig package

Starting with v0.52 of nag, we recognize the fact that the `\subfloat` command from Steven D. Cochran's subfig package is a caption-provider for its fourth argument. Earlier versions would flag use of `\label` as inappropriate. The current implementation works with versions close enough to v1.3 of subfig. Since the change is a one-liner, I hope it will be integrated into future versions of subfig.

```

356 \AtBeginDocument{%
357 \nag@ifcsname{ver@subfig.sty}{%
358 \PackageInfo{nag}{Attempting subfig hack\@gobble}%
359 \nag@maybehacksubfig
360 }{%
361 }%
362 }
363 \def\nag@maybehacksubfig{%
364 %
365 % of course, i need to touch the single longest definition in
366 % subfig.sty, to amend one single command...
367 %
368 % The definition is taken from subfig.sty 1.3 dated 2005/07/05 by
369 % S.D. Cochran, where it is called sf@@@subfloat, and appears here
370 % under the conditions of section 6 of the LPPL 1.3. The subfig
371 % package is available on a CTAN mirror near you.
372 %
373 \long\def\nag@@original@@sf@@@subfloat##1[##2][##3]##4{%
374 \@ifundefined{FBsc@max}{%
375 }{%

```

```

376     \FB@readaux{\let\FBsuboheight\relax}%
377 }%
378 \@tempcnta=\@ne
379 \if@minipage
380     \@tempcnta=\z@
381 \else\ifdim \lastskip=\z@ \else
382     \@tempcnta=\tw@
383 \fi\fi
384 \ifmaincaptiontop
385     \sf@top=\sf@nearskip
386     \sf@bottom=\sf@farskip
387 \else
388     \sf@top=\sf@farskip
389     \sf@bottom=\sf@nearskip
390 \fi
391 \leavevmode
392 \setbox\@tempboxa \hbox{%
393     ##4}%
394 \@tempdima=\wd\@tempboxa
395 \ifundefined{FBsc@max}{%
396 }{%
397     \global\advance\Xhsize-\wd\@tempboxa
398     \dimen@=\ht\@tempboxa
399     \advance\dimen@\dp\@tempboxa
400     \ifdim\dimen@>\FBso@max
401     \global\FBso@max\dimen@
402     \fi
403 }%
404 \vtop\bgroup
405     \vbox\bgroup
406         \ifcase\@tempcnta
407         \@minipagefalse
408         \or
409             \vskip\sf@top
410         \or
411             \ifdim \lastskip=\z@ \else
412                 \@tempskipb\sf@top\relax\@xaddvskip
413             \fi
414         \fi
415         \sf@ifpositiontop{%
416             \ifx \@empty##3\relax \else
417                 \sf@subcaption{##1}{##2}{##3}%
418                 \vskip\sf@capskip
419                 \vskip\sf@captopadj
420             \fi\egroup
421             \hrule width0pt height0pt depth0pt
422             \box\@tempboxa
423         }{%
424             \ifundefined{FBsc@max}{%
425                 \box\@tempboxa

```

```

426     }{%
427     \ifx\FBsuboheight\relax
428     \box\@tempboxa
429     \else
430     \vbox to \FBsuboheight{\FBafil\box\@tempboxa\FBbfil}%
431     \fi}%
432     \egroup
433     \ifx \@empty##3\relax \else
434     \vskip\sf@capskip
435     \hrule width0pt height0pt depth0pt
436     \sf@subcaption{##1}{##2}{##3}%
437     \fi
438     }%
439     \vskip\sf@bottom
440     \egroup
441     \@ifundefined{FBsc@max}{%
442     }{%
443     \addtocounter{FRobj}{-1}%
444     \ifnum\c@FRobj=0\else
445     \subfloatrowsep
446     \fi
447     }%
448     \ifmaincaptiontop\else
449     \global\advance\@nameuse{c@\@captype}\m@ne
450     \fi
451     \endgroup\ignorespaces}%
452 %
453 \expandafter\ifx\csname sf@@@subfloat\endcsname\nag@@original@s f@@@subfloat
454 % yup, that's it.
455 \PackageInfo{nag}{OK, equivalent to subfig 1.3, redefining
456 \backslashchar sf@@@subfloat\@gobble}%
457 \global\long\def\s f@@@subfloat##1[##2][##3]##4{%
458 \@ifundefined{FBsc@max}{%
459 }{%
460 \FB@readaux{\let\FBsuboheight\relax}%
461 }%
462 \@tempcnta=\@ne
463 \if@minipage
464 \@tempcnta=\z@
465 \else\ifdim \lastskip=\z@ \else
466 \@tempcnta=\tw@
467 \fi\fi
468 \ifmaincaptiontop
469 \sf@top=\sf@narskip
470 \sf@bottom=\sf@farskip
471 \else
472 \sf@top=\sf@farskip
473 \sf@bottom=\sf@narskip
474 \fi
475 \leavevmode

```

```

476 \setbox\@tempboxa \hbox{%
477   %% ulmi: new 2007/02/25: #4 may contain label command
478   \csname nag@hascaptiontrue\endcsname
479   %% and that was it.
480   ##4}%
481 \@tempdima=\wd\@tempboxa
482 \@ifundefined{FBsc@max}{%
483 }{%
484   \global\advance\Xhsize-\wd\@tempboxa
485   \dimen@=\ht\@tempboxa
486   \advance\dimen@\dp\@tempboxa
487   \ifdim\dimen@>\FBso@max
488   \global\FBso@max\dimen@
489   \fi
490 }%
491 \vtop\bgroup
492   \vbox\bgroup
493     \ifcase\@tempcnta
494     \@minipagefalse
495     \or
496     \vskip\sf@top
497     \or
498     \ifdim \lastskip=\z@ \else
499       \@tempskipb\sf@top\relax\@xaddvskip
500     \fi
501   \fi
502   \sf@ifpositiontop{%
503     \ifx \@empty##3\relax \else
504       \sf@subcaption{##1}{##2}{##3}%
505       \vskip\sf@capskip
506       \vskip\sf@captopadj
507     \fi\egroup
508     \hrule width0pt height0pt depth0pt
509     \box\@tempboxa
510   }{%
511     \@ifundefined{FBsc@max}{%
512       \box\@tempboxa
513     }{%
514       \ifx\FBsuboheight\relax
515         \box\@tempboxa
516       \else
517         \vbox to \FBsuboheight{\FBafil\box\@tempboxa\FBbfil}%
518       \fi}%
519   \egroup
520   \ifx \@empty##3\relax \else
521     \vskip\sf@capskip
522     \hrule width0pt height0pt depth0pt
523     \sf@subcaption{##1}{##2}{##3}%
524   \fi
525 }%

```



```

526     \vskip\sf@bottom
527   \egroup
528   \@ifundefined{FBsc@max}{%
529   }{%
530     \addtocounter{FRobj}{-1}%
531     \ifnum\c@FRobj=0\else
532       \subfloatrowsep
533     \fi
534   }%
535   \ifmaincaptiontop\else
536     \global\advance\@nameuse{c@\@captype}\m@ne
537   \fi
538   \endgroup\ignorespaces}%
539 \else
540   \PackageInfo{nag}{Not redefining sf@@subfloat, it looks odd\@gobble}
541 \fi
542 }

```

4.3 The float package

Sorry, there is no way for nag to automatically add new float types to check them for captions. However, since v0.52, there is an user-level command `\NagDeclareFloat` that will do the bookkeeping for you, i.e. after your call to `\newfloat`, you call `\NagDeclareFloat` with the first argument to `\newfloat`.

```

543 \newcommand*\NagDeclareFloat[1]{\g@addto@macro\nag@floats{,#1}}

```

5 Loading extensions

Finally, we deal with package options. This is simple: just try to input appropriate nag files.

```

544 \DeclareOption*{%
545   \InputIfFileExists{nag-\CurrentOption.cfg}{%
546     \PackageInfo{nag}{%
547       Loaded nag-\CurrentOption.cfg
548     }
549   }{%
550     \InputIfFileExists{\CurrentOption.nag}{%
551       \PackageWarningNoLine{nag}{%
552         Loaded old-style config file \CurrentOption.nag.\MessageBreak
553         Consider renaming the file to nag-\CurrentOption.cfg
554       }%
555     }{%
556       \PackageWarningNoLine{nag}{Required ruleset
557         \CurrentOption, and it wasn't there}
558     }%
559   }
560 }
561 \ProcessOptions*

```

</nag>

Change History

0.1	General: First official version. 1	0.5	General: Handle the case that somebody else relaxes the ver@-commands. Stack-based No-tASwitch. 1
0.2	General: Added abort.nag, suggested by Michael Zedler 1		
	Rephrased umlaut.sty warning, suggested by Patrick Happel. 1	0.51	\nag@prepend: bugfix 6
0.3	General: Fixed missing globals 9	0.52	General: Command NagDeclareFloat added 17
	New ifdefined that won't relax the commands 1		made eTeX-ifcsname more robust . 7
0.4	General: bugfix 9		twiddle subfig's bowels 13
	config file names changed to free extension 17	\nag@prepend: info 6	
	Handling command vs. environment; bugfixes 1	0.53	General: bugfix: more Robustness. (Jörg Sommer) 11

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		A
\ " 79, 175	\@secondoftwo 173, 328, 341	\advance 397,
\@backslashchar 134, 160,	\@tempboxa 392,	399, 449, 484, 486, 536
284, 316, 317, 328, 456	394, 397–399, 422,	\AtBeginDocument 157,
\@capttype 449, 536	425, 428, 430,	255, 258, 279, 347, 356
\@clsextension . 190, 200	476, 481, 484–	\AtEndDocument
\@currrentvir 271	486, 509, 512, 515, 517	... 189, 199, 259, 335
\@empty 220, 226,	\@tempcnta	
250, 416, 433, 503, 520	... 378, 380, 382,	B
\@firstoftwo	406, 462, 464, 466, 493	\bfseries 18
... 180, 321, 327, 339	\@tempdima 394, 481	\bgroup 269,
\@gobble 358, 456, 540	\@tempskipb 412, 499	404, 405, 491, 492
\@minipagefalse 407, 494	\@tfor 336	\binom 36
\@nameuse 449, 536	\@xa 126,	\box 422, 425, 428,
\@ne 378, 462	137, 138, 140, 251,	430, 509, 512, 515, 517
\@nil 308, 311	300, 303, 308, 321,	C
\@nx 127, 299–303	327, 336, 337, 339, 341	\c@FRobj 444, 531
	\@xaddvskip 412, 499	\caption 235, 242

<code>\captionprovider</code> 225–227	J	<code>\nag@warnNoLine</code> 119,
<code>\changes</code> 79, 80	<code>\justenv</code> 108, 109	152, 192, 202, 338
D	<code>\justsw</code> 104, 106	<code>\NagDeclareFloat</code> 76, 543
<code>\DeclareRobustCommand</code>	L	<code>\NeedsTeXFormat</code> 124
..... 268, 295, 306	<code>\label</code> 241	<code>\newcommand</code> . 130, 145,
<code>\dimen@</code> . 398–401, 485–488	<code>\labelprovider</code> . 219–221	148, 152, 156, 166,
<code>\dp</code> 399, 486	<code>\lastskip</code> 381, 411, 465, 498	188, 198, 210, 239,
E	<code>\leavevmode</code> 391, 475	248, 278, 293, 346, 543
<code>\egroup</code> 276, 420,	<code>\long</code> 373, 457	<code>\newcounter</code> 128, 129
432, 440, 507, 519, 527	<code>\looseends</code>	<code>\newif</code> 208, 209
<code>\endfoo</code> 292	... 336, 337, 339, 341	<code>\noexpand</code> 127
F	M	<code>\NotAnEnvironment</code> 84,
<code>\FB@readaux</code> 376, 460	<code>\m@ne</code> 449, 536	87, 97, 102, 106, 278
<code>\FBafil</code> 430, 517	<code>\mathcal</code> 32	<code>\NotASwitch</code> . 109, <u>291</u> , 346
<code>\FBbfil</code> 430, 517	N	O
<code>\FBso@max</code> 400, 401, 487, 488	<code>\nag@@original@sff@subfloat</code>	<code>\ObsoleteClass</code> .. 40, <u>198</u>
<code>\FBsuboheight</code> .. 376, 373, 453	<code>\ObsoleteCS</code> 17,
427, 430, 460, 514, 517	<code>\nag@beginenv</code> .. 295, 349	<code>\ObsoletePackage</code> ...
<code>\fontcmd</code> ... 95, 97, 99, 102	<code>\nag@captioncheck</code> 9, 10, 42–44,
<code>\foo</code> 292, 297 222, 239	46, 48, 49, 51, 52,
<code>\frac</code> 34–36	<code>\nag@captions</code> 81, 225, 247	54, 56, 59, 61, 63,
G	<code>\nag@end@nv</code> 308, 311	65, 68, 70, 73, 74, <u>188</u>
<code>\g@addto@macro</code> 77, 81, 543	<code>\nag@endenv</code> . 306, 320, 352	<code>\or</code> 408, 410, 495, 497
<code>\gdef</code> 323	<code>\nag@envstack</code> .. 293,	P
<code>\global</code> 186,	298, 303, 308, 323, 336	<code>\PackageError</code> .. 117, 121
227, 230, 397, 401,	<code>\nag@floats</code> . 245, 249, 543	<code>\PackageInfo</code> ... 133,
449, 457, 484, 488, 536	<code>\nag@floatsetup</code> 248, 256	358, 455, 540, 546
H	<code>\nag@hackfloat</code> . 210, 251	<code>\ProvidesFile</code> . 2, 93, 113
<code>\hbox</code> 392, 476	<code>\nag@hascaptionfalse</code> 230	<code>\ProvidesPackage</code> ... 125
<code>\hrule</code> .. 421, 435, 508, 522	<code>\nag@hascaptiontrue</code> 227	R
<code>\ht</code> 398, 485	<code>\nag@haslabelfalse</code> . 230	<code>\renewcommand</code> 115, 119, 170
I	<code>\nag@haslabeltrue</code> .. 222	<code>\rmfamily</code> 22
<code>\if@minipage</code> ... 379, 463	<code>\nag@ifcsname</code> 166, 169,	S
<code>\ifcase</code> 406, 493	170, 186, 191, 201, 357	<code>\S</code> 8, 11–14, 16, 33,
<code>\ifdim</code> 381,	<code>\nag@ifCurrentEnvironment</code>	37, 39, 41, 45, 47,
400, 411, 465, 487, 498 268, 281	50, 55, 57, 60, 67,
<code>\ifmaincaptiontop</code> ..	<code>\nag@labels</code> .. 77, 219, 246	71, 72, 75, 83, 89, 90
... 384, 448, 468, 535	<code>\nag@maybehacksubfig</code>	<code>\scshape</code> 24
<code>\ifnag@hascaption</code> 359, 363	<code>\sectioning</code> 86, 87
..... 209, 233, 240	<code>\nag@pr@p@nd</code> ... 140, 145	<code>\setbox</code> 392, 476
<code>\ifnag@haslabel</code> 208	<code>\nag@prepend</code> <u>130</u> , 158,	<code>\sf@subfloat</code> 457
<code>\ignorespaces</code> .. 451, 538	211, 212, 218, 221,	<code>\sf@bottom</code> 386,
<code>\inputlineno</code> 301	227, 232, 280, 348, 351	389, 439, 470, 473, 526
<code>\itshape</code> 20	<code>\nag@warn</code> 115,	
	<u>148</u> , 159, 213, 234,	
	241, 282, 316, 326	

<code>\sf@capskip</code>	<code>\string</code> 327, 339, 341	V
... 418, 434, 505, 521	<code>\subfloatrowsep</code> 445, 532	<code>\vbox</code> ... 405, 430, 492, 517
<code>\sf@capttopadj</code> .. 419, 506		<code>\vskip</code> 409,
<code>\sf@farskip</code>	T	418, 419, 434, 439,
... 386, 388, 470, 472	<code>\textbf</code> 18	496, 505, 506, 521, 526
<code>\sf@ifpositiontop</code> ..	<code>\textit</code> 20	<code>\vtop</code> 404, 491
..... 415, 502	<code>\textrm</code> 22	
<code>\sf@nearskip</code>	<code>\textsc</code> 24	W
... 385, 389, 469, 473	<code>\textsf</code> 26	<code>\wd</code> 394, 397, 481, 484
<code>\sf@subcaption</code>	<code>\textsl</code> 28	
... 417, 436, 504, 523	<code>\texttt</code> 30	X
<code>\sf@top</code> . 385, 388, 409,	<code>\thenag@c</code> 138, 141	<code>\xdef</code> 298
412, 469, 472, 496, 499	<code>\tmp@a</code> 173, 180,	<code>\Xhsize</code> 397, 484
	183, 270, 271, 312, 314	
<code>\sffamily</code> 26	<code>\tmp@b</code> 313, 314	Z
<code>\sloppy</code> 14	<code>\ttfamily</code> 30	<code>\z@</code> 380,
<code>\slshape</code> 28	<code>\tw@</code> 382, 466	381, 411, 464, 465, 498