

1 The Breton language

The file `breton.dtx`¹ defines all the language-specific macros for the Breton language.

There are not really typographic rules for the Breton language. It is a local language (it's one of the celtic languages) which is spoken in Brittany (West of France). So we have a synthesis between french typographic rules and english typographic rules. The characters `:`, `;`, `!` and `?` are made active in order to get a whitespace automatically before these characters.

The macro `\LdfInit` takes care of preventing that this file is loaded more than once, checking the category code of the `@` sign, etc.

```
1 {*code}
2 \LdfInit{breton}\captionsbreton
```

When this file is read as an option, i.e. by the `\usepackage` command, `breton` will be an 'unknown' language in which case we have to make it known. So we check for the existence of `\l@breton` to see whether we have to do something here.

```
3 \ifx\l@breton\@undefined
4 \@nopatterns{Breton}
5 \adddialect\l@breton0\fi
```

The next step consists of defining commands to switch to the English language. The reason for this is that a user might want to switch back and forth between languages.

`\captionsbreton` The macro `\captionsbreton` defines all strings used in the four standard document classes provided with L^AT_EX.

```
6 \addto\captionsbreton{%
7 \def\prefacename{Rakskrid}%
8 \def\refname{Daveenno\`u}%
9 \def\abstractname{Dvierra\~n}%
10 \def\bibname{Lennadurezh}%
11 \def\chaptername{Pennad}%
12 \def\appendixname{Stagadenn}%
13 \def\contentsname{Taolenn}%
14 \def\listfigurename{Listenn ar Figurenno\`u}%
15 \def\listtablename{Listenn an taolenno\`u}%
16 \def\indexname{Meneger}%
17 \def\figurename{Figurenn}%
18 \def\tablename{Taolenn}%
19 \def\partname{Lodenn}%
20 \def\enclname{Diello\`u kevret}%
21 \def\ccname{Eilskrid da}%
22 \def\headtoname{evit}
23 \def\pagename{Pajenn}%
24 \def\seename{Gwelout}%
```

¹The file described in this section has version number v1.0h and was last revised on 2005/03/29.

```

25 \def\alsoname{Gwelout ivez}%
26 \def\proofname{Proof}% <-- needs translation
27 \def\glossaryname{Glossary}% <-- Needs translation
28 }

```

`\datebreton` The macro `\datebreton` redefines the command `\today` to produce Breton dates.

```

29 \def\datebreton{%
30 \def\today{\ifnum\day=1\relax 1/\${\rm a\tilde{n}}$\else
31 \number\day\fi \space a\space viz\space\ifcase\month\or
32 Genver\or C'hwevrer\or Meurzh\or Ebrel\or Mae\or Mezheven\or
33 Gouere\or Eost\or Gwengolo\or Here\or Du\or Kerzu\fi
34 \space\number\year}}

```

`\extrasbreton` The macro `\extrasbreton` will perform all the extra definitions needed for the Breton language. The macro `\noextrasbreton` is used to cancel the actions of `\extrasbreton`.

The category code of the characters `:`, `;`, `!` and `?` is made `\active` to insert a little white space.

```

35 \initiate@active@char{:}
36 \initiate@active@char{:;}
37 \initiate@active@char{:!}
38 \initiate@active@char{:?}

```

We specify that the breton group of shorthands should be used.

```

39 \addto\extrasbreton{\languageshorthands{breton}}

```

These characters are ‘turned on’ once, later their definition may vary.

```

40 \addto\extrasbreton{%
41 \bbl@activate{:}\bbl@activate{:;}%
42 \bbl@activate{:!}\bbl@activate{:?}}

```

Don’t forget to turn the shorthands off again.

```

43 \addto\noextrasbreton{%
44 \bbl@deactivate{:}\bbl@deactivate{:;}%
45 \bbl@deactivate{:!}\bbl@deactivate{:?}}

```

The last thing `\extrasbreton` needs to do is to make sure that `\frenchspacing` is in effect. If this is not the case the execution of `\noextrasbreton` will switch it off again.

```

46 \addto\extrasbreton{\bbl@frenchspacing}
47 \addto\noextrasbreton{\bbl@nonfrenchspacing}

```

`\breton@sh@;` We have to reduce the amount of white space before `;`, `:` and `!` when the user types a space in front of these characters. This should only happen outside mathmode, hence the test with `\ifmmode`.

```

48 \declare@shorthand{breton}{;}{;%
49 \ifmmode
50 \string;\space
51 \else\relax

```

In horizontal mode we check for the presence of a ‘space’ and replace it by a `\thinspace`.

```
52     \ifhmode
53         \ifdim\lastskip>\z@
54             \unskip\penalty\@M\thinspace
55         \fi
56     \fi
57     \string;\space
58 \fi}%
```

`\breton@sh@:` Because these definitions are very similar only one is displayed in a way that the
`\breton@sh@!` definition can be easily checked.

```
59 \declare@shorthand{breton}{:}{%
60     \ifmmode\string;\space
61     \else\relax
62     \ifhmode
63         \ifdim\lastskip>\z@\unskip\penalty\@M\thinspace\fi
64     \fi
65     \string;\space
66 \fi}
67 \declare@shorthand{breton}{!}{%
68     \ifmmode\string!\space
69     \else\relax
70     \ifhmode
71         \ifdim\lastskip>\z@\unskip\penalty\@M\thinspace\fi
72     \fi
73     \string!\space
74 \fi}
```

`\breton@sh@?` For the question mark something different has to be done. In this case the amount
of white space that replaces the space character depends on the dimensions of the
font.

```
75 \declare@shorthand{breton}{?}{%
76     \ifmmode
77         \string?\space
78     \else\relax
79     \ifhmode
80         \ifdim\lastskip>\z@
81             \unskip
82             \kern\fontdimen2\font
83             \kern-1.4\fontdimen3\font
84         \fi
85     \fi
86     \string?\space
87 \fi}
```

All that is left to do now is provide the breton user with some extra utilities.
Some definitions for special characters.

```
88 \DeclareTextSymbol{\at}{OT1}{64}
```

```

89 \DeclareTextSymbol{\at}{T1}{64}
90 \DeclareTextSymbolDefault{\at}{OT1}
91 \DeclareTextSymbol{\boi}{OT1}{92}
92 \DeclareTextSymbol{\boi}{T1}{16}
93 \DeclareTextSymbolDefault{\boi}{OT1}
94 \DeclareTextSymbol{\circonflexe}{OT1}{94}
95 \DeclareTextSymbol{\circonflexe}{T1}{2}
96 \DeclareTextSymbolDefault{\circonflexe}{OT1}
97 \DeclareTextSymbol{\tild}{OT1}{126}
98 \DeclareTextSymbol{\tild}{T1}{3}
99 \DeclareTextSymbolDefault{\tild}{OT1}
100 \DeclareTextSymbol{\degre}{OT1}{23}
101 \DeclareTextSymbol{\degre}{T1}{6}
102 \DeclareTextSymbolDefault{\degre}{OT1}

```

The following macros are used in the redefinition of $\hat{\ }$ and $\" \$ to handle the letter i.

```

103 \AtBeginDocument{%
104   \DeclareTextCompositeCommand{\^}{OT1}{i}{\^i}
105   \DeclareTextCompositeCommand{\"}{OT1}{i}{\"}i}

```

And some more macros for numbering.

```

106 \def\kentan{1\/$}{\rm a\tilde{n}}$}
107 \def\eil{2\/$}{\rm l}$}
108 \def\re{\\/$}{\rm re}$}
109 \def\trede{3\re}
110 \def\pevare{4\re}
111 \def\vet{\\/$}{\rm vet}$}
112 \def\pempvet{5\vet}

```

The macro `\ldf@finish` takes care of looking for a configuration file, setting the main language to be switched on at `\begin{document}` and resetting the category code of `@` to its original value.

```

113 \ldf@finish{breton}
114 \code

```