

The `tugboat` package*

The *TUGboat* team

2026-09-15

Contents

1 Document preambles	2
2 Introduction	2
2.1 Summary of control sequences	2
3 L^AT_EX 2_ε <i>TUGboat</i> class file	6
3.1 Setup and options	6
3.2 Resetting at start of paper	10
3.3 Helpful shorthands (common code with Plain styles)	11
3.4 Abbreviations and logos	12
3.5 General typesetting rules	17
3.6 Utility registers and definitions	19
3.7 Ragged right and friends	21
3.8 Assorted user-level markup	22
3.9 Reviews	27
3.10 Dates, volume and issue numbers, etc.	28
3.11 Page dimensions, glue, penalties, etc.	32
3.12 Messing about with the L ^A T _E X logo	33
3.13 Authors, contributors, addresses, signatures	34
3.14 Article title	40
3.15 Section titles	41
3.16 Section headings	45
3.17 Appendices	48
3.18 References	49
3.19 Title references	50
3.20 Float captions	51
3.21 Size changing commands	53
3.22 Lists and other text inclusions	53
3.23 Some fun with <code>verbatim</code>	54
3.24 Bibliography	57
3.25 Registration marks	60
3.26 Running headers and footers	60
3.27 Output routine	63
3.28 Font-related definitions and machinery	63

*This file has version number v2.39, last revised 2026-09-15

3.29	Editor's notes and other footnotes	64
3.30	Initialization	66
4	L^AT_EX 2_ε proceedings class (no longer used)	66
4.1	Proceedings titles	68
4.2	Section divisions	72
5	Plain T_EX styles	73
6	The L^AT_EX 2_ε compatibility-mode style files	73

1 Document preambles

```

1 <ltugboatcls | ltugproccls | ltugcomm>\NeedsTeXFormat{LaTeX2e}[1994/12/01]
2 <*dtx>
3 \ProvidesFile                {tugboat.dtx}
4 </dtx>
5 <ltugboatcls>\ProvidesClass  {ltugboat}
6 <ltugproccls>\ProvidesClass  {ltugproc}
7 <ltugboatsty>\ProvidesPackage{ltugboat}
8 <ltugprocsty>\ProvidesPackage{ltugproc}
9 <ltugcomm>   \ProvidesPackage{ltugcomm}
10             [2026-09-15 v2.39
11 <ltugboatcls>                TUGboat journal class%
12 <ltugproccls>                TUG conference proceedings class%
13 <ltugboatsty | ltugprocsty>  TUG compatibility package%
14 <ltugcomm>                  TUGboat 'common macros' package%
15 <*dtx>
16                               TUG macros source file%
17 </dtx>
18 ]

```

2 Introduction

This file contains all the macros for typesetting *TUGboat* with both plain T_EX and L^AT_EX 2_ε.

2.1 Summary of control sequences

A few of the abbreviations we define, with indications of expansion where that may not be obvious. For full definitions, see real code below (Section 3.4).

<code>\AllTeX</code>	(L ^A)T _E X
<code>\AMS</code>	American Mathematical Society
<code>\AmSTeX</code>	
<code>\aw</code>	A-W (abbreviation for Addison-Wesley)
<code>\API</code>	
<code>\AW</code>	Addison-Wesley
<code>\BibTeX</code>	
<code>\CandT</code>	Computers & Typesetting
<code>\ConTeXt</code>	ConT _E Xt

<code>\Cplusplus</code>	C++
<code>\DTD</code>	
<code>\DVD</code>	
<code>\DVI</code>	
<code>\DVIPDFMx</code>	DVIPDFM <i>x</i>
<code>\DVItoVDU</code>	DVItoVDU
<code>\ECMA</code>	
<code>\EPS</code>	
<code>\eTeX</code>	ε -T _E X
<code>\ExTeX</code>	ε_X T _E X
<code>\Ghostscript</code>	
<code>\Hawaii</code>	Hawai'i
<code>\HTML</code>	
<code>\ISBN</code>	ISBN
<code>\ISO</code>	
<code>\ISSN</code>	ISSN
<code>\JTeX</code>	
<code>\JoT</code>	The Joy of T _E X
<code>\LaTeX</code>	
<code>\LyX</code>	
<code>\macOS</code>	mac OS
<code>\MacOSX</code>	Mac OS X
<code>\MathML</code>	
<code>\Mc</code>	M with raised c
<code>\MF</code>	METAFONT
<code>\mf</code>	METAFONT
<code>\MFB</code>	The Metafontbook
<code>\MP</code>	METAPOST
<code>\mp</code>	MetaPost (in text only: still ‘ $\mp$ ’ in math)
<code>\OMEGA</code>	Omega ‘logo’ (Ω)
<code>\OCP</code>	Omega compiled process
<code>\OOXML</code>	
<code>\OTP</code>	Omega translation process
<code>\mtex</code>	multilingual T _E X
<code>\NTS</code>	New Typesetting System
<code>\pcMF</code>	pcMF
<code>\PCTeX</code>	
<code>\pcTeX</code>	
<code>\Pas</code>	Pascal
<code>\PiCTeX</code>	
<code>\plain</code>	plain (in typewriter font)
<code>\POBox</code>	P. O. Box
<code>\PS</code>	PostScript (with hyphenation)
<code>\SC</code>	Steering Committee
<code>\SGML</code>	SGML
<code>\SliTeX</code>	
<code>\slMF</code>	Metafont, slanted: deprecated: use <code>\textsl</code> instead
<code>\stTeX</code>	T _E X for the Atari ST
<code>\SVG</code>	

<code>\TANGLE</code>	
<code>\TB</code>	The $\TeX$ book
<code>\TeX</code>	(Although nearly every package defines this, most, including plain, are missing the spacefactor adjustment)
<code>\TeXhax</code>	
<code>\TeXMaG</code>	(defunct)
<code>\TeXtures</code>	
<code>\TeXXeT</code>	
<code>\Thanh</code>	
<code>\TFM</code>	TFM
<code>\TUB</code>	<i>TUGboat</i>
<code>\TUG</code>	$\TeX$ Users Group
<code>\UNIX</code>	
<code>\VAX</code>	
<code>\VnTeX</code>	
<code>\VorTeX</code>	
<code>\XeT</code>	
<code>\XeTeX</code>	reflected and lowered first ‘E’
<code>\XeLaTeX</code>	with extra space before ‘L’
<code>\XML</code>	
<code>\WEB</code>	
<code>\WEAVE</code>	
<code>\WYSIWYG</code>	

Macros for things that are slightly more significant.

<code>\NoBlackBoxes</code>	turns off marginal rules marking overfull boxes
<code>\BlackBoxes</code>	turns them back on
<code>\newline</code>	horizontal glue plus a break
<code>\ifundefined#1</code>	checks argument with <code>\csname</code> against <code>\relax</code>
<code>\topsmash</code>	smashes above baseline (from AMSTeX)
<code>\botsmash</code>	smashes below baseline (from AMSTeX)
<code>\smash</code>	smashes both (from plain)
<code>\ulap</code>	lap upwards
<code>\dlap</code>	lap downwards
<code>\xlap</code>	reference point at center horizontally; 0 width
<code>\ylap</code>	reference point at center vertically; 0 height, depth
<code>\zlap</code>	combination <code>\xlap</code> and <code>\ylap</code>
<code>\basezero</code>	to avoid insertion of <code>baselineskip</code> and <code>lineskip</code> glue
<code>\nullhrule</code>	empty <code>\hrule</code>
<code>\nullvrule</code>	empty <code>\vrule</code>
<code>\makestrut[#1;#2]</code>	ad hoc struts; #1=height, #2=depth
<code>\today</code>	today’s date
<code>\SetTime</code>	converts <code>\time</code> to hours, minutes
<code>\now</code>	displays time in hours and minutes
<code>\Now</code>	shows current date and time
<code>\ifPrelimDraft</code>	flag to indicate status as preliminary draft

<code>\rtitlex</code>	<i>TUGboat</i> volume and number info for running head
<code>\midrtitle</code>	information for center of running head
<code>\rtitlenexttopage</code>	next to page number in running head
<code>\HorzR@gisterRule</code>	pieces of registration marks ('trimmarks')
<code>\DownShortR@gisterRule</code>	
<code>\UpShortR@gisterRule</code>	
<code>\ttopregister</code>	top registration line with 'T' in center
<code>\tbotregister</code>	bottom registration line with inverted 'T' in center
<code>\topregister</code>	register actually used
<code>\botregister</code>	
<code>\raggedskip</code>	parameters used for ragged settings
<code>\raggedstretch</code>	
<code>\raggedparfill</code>	
<code>\raggedspaces</code>	
<code>\raggedright</code>	
<code>\raggedleft</code>	
<code>\raggedcenter</code>	
<code>\normalspaces</code>	
<code>\raggedbottom</code>	
<code>\bull</code>	square bullet
<code>\cents</code>	'cents' sign
<code>\Dag</code>	superscripted dagger
<code>\careof</code>	c/o
<code>\sfrac</code>	slashed fraction (arguments optionally separated by a slash)
<code>\cs</code>	control sequence name <code>\cs{name}→\name</code>
<code>\meta</code>	meta-argument name <code>\meta{name}→⟨name⟩</code>
<code>\dash</code>	en-dash surrounded by thinspaces; only breakable AFTER
<code>\Dash</code>	em-dash, as above
<code>\hyph</code>	permit automatic hyphenation after an actual hyphen
<code>\slash</code>	'breakable' slash
<code>\nth</code>	for obtaining '1 st ', '2 nd ', '3 rd ', etc.
<code>\tubissue</code>	gets \TUB followed by volume and issue numbers
<code>\xEdNote</code>	Editor's Note:
<code>\Review:</code>	Review: (for title of book review article)
<code>\reviewitem</code>	begin data for item being reviewed
<code>\revauth</code>	with one argument, author(s) of item being reviewed
<code>\revtitle</code>	with one argument, title of ...
<code>\revpubinfo</code>	with one argument, other info pertaining to ...
<code>\endreviewitem</code>	end data for item being reviewed
<code>\titleref</code>	one argument, format title as straight text (slanted, frenchspacing)

<code>\Input</code>	<code>\input</code> with some other bookkeeping for case where multiple articles are put together
<code>\TBremark</code>	reminder to <i>TUGboat</i> editorial staff
<code>\TEnableRemarks</code>	enable <code>\TBremarks</code> (normally suppressed)
<code>\pagexref</code>	used to write out page numbers to screen and external files
<code>\pagexrefON</code>	
<code>\pagexrefOFF</code>	
<code>\xref to</code>	used for symbolic cross-reference to other pages in <i>TUGboat</i>
<code>\xref toON</code>	
<code>\xref toOFF</code>	
<code>\TBdriver</code>	marks code which only takes effect when articles are run together in a driver file
<code>\signaturemark</code>	items for signatures
<code>\signaturewidth</code>	

3 $\text{\LaTeX} 2_{\epsilon}$ *TUGboat* class file

3.1 Setup and options

Occasionally we need to do different things when running under traditional (pdf)latex or a native Unicode engine. Since we don't need any fancier distinctions, instead of reading the `iftex` or another package, do the test directly.

```

19 <*common>
20 \newif\ifTBUnicodeengine
21 \ifx\Umathchardef\@thisisundefined % not (xetex|luatex)
22   \TBUnicodeenginefalse
23 \else
24   \TBUnicodeenginetrue
25 \fi
26 </common>

```

Check for reloading. Hmmm...Does this happen with $\text{\LaTeX} 2_{\epsilon}$ classes? Probably, in fact, as well that it doesn't, since the `\tugstyinit` referenced here doesn't exist; however, it's possible that we might need a similar mechanism in the future, so we retain its skeleton, without fleshing out the `\tugstyinit` bones.

```

27 <*tugboatcls>
28 \csname tugstyloaded@ \endcsname
29 \def\tugstyloaded@{\tugstyinit\endinput}

```

Acquire a name for this class if we don't already have one (by virtue of having been loaded by `tugproc.cls`). This name will be used in error messages and the like.

```

30 \providecommand{\@tugclass}{ltugboat}

```

Errors/warnings/information messages—if we're using $\text{\LaTeX} 2_{\epsilon}$ we can use the `\Class*` commands. `\TBDebug` is different from all the others, intended for temporary debugging messages; its messages start with `***` at the beginning of a line to make them stand out, and be parsable, e.g., by `texfot(1)`.

```

31 \def\TBError{\ClassError{\@tugclass}}
32 \def\TBWarning{\ClassWarning{\@tugclass}}
33 \def\TBWarningNL{\ClassWarningNoLine{\@tugclass}}

```

```

34 \def\TBInfo{\ClassInfo{\@tugclass}}
35 \def\TBDebug#1{\message{^^J*** #1}}

```

Unfortunately, L^AT_EX's `\loggingall` does not turn off `tracingonline`. And `microtype` outputs many verbose expansions to the terminal after `\loggingall`. So make our own:

```

36 \def\TBLoggingall{\loggingall \tracingonline=0 }
    Class options: draft vs. preprint vs. final.
37 \DeclareOption{draft}{% [draft], the default
38 % If the user loads hyperref, avoid passing on the global draft option
39 % (which would remove all links in the pdf).
40 \PassOptionsToPackage{final}{hyperref}
41 %
42 \AtEndOfClass{%
43   \setcounter{page}{901}%
44   \BlackBoxes
45   \def\MakeRegistrationMarks{}%
46   \PrelimDrafttrue
47 }%
48 }
49
50 \newif\ifpreprint
51 \def\preprint{\preprinttrue} % [preprint], hardly used
52 \DeclareOption{preprint}{%
53   \preprinttrue
54 }
55
56 \newif\iftubfinaloption % [final], manually inserted by us for processing
57 \DeclareOption{final}{%
58   \tubfinaloptiontrue
59 \AtEndOfClass{%
60   % Insert draft date into the header even with [final], if we are not
61   % doing a production run. (|tugboat.dates| sets up page numbers
62   % above 900 in such pseudo-draft mode.) We use [final] in the first
63   % place for this case because draft vs. final can change page
64   % layout, wrt registration marks, etc. (Not good, but too painful to
65   % change at this late date.)
66   \ifnum\value{page}>900 \PrelimDrafttrue \else \PrelimDraftfalse \fi
67   \@tubrunningfull
68 }%
69 }

```

We want to use `hyperref`'s `\texorpdfstring`, e.g., in the `draft` option above. If `hyperref` is not loaded, define our own trivial fallback to expand to the T_EX (first) argument.

Similarly, disable and more if we have `hyperref`, so section titles using them don't cause useless warnings.

```

70 \AtBeginDocument{%
71   \ifx\undefined\texorpdfstring
72     \DeclareRobustCommand{\texorpdfstring}[2]{#1}%
73   \fi
74   %
75   \ifx\undefined\pdfstringdefDisableCommands\else

```

```

76 \pdfstringdefDisableCommands{%
77 \let\acro\relax
78 \let\origDash=\Dash \def\Dash{\texorpdfstring{\origDash}{--}}%
79 % lots more could/should be added.
80 }%
81 \fi
82 }

```

TUGboat uses only 10pt for the main text.

```

83 \DeclareOption{11pt}{%
84 \TBWarning{The \@tugclass\space class only supports 10pt fonts:
85 \MessageBreak option \CurrentOption\space ignored}%
86 }
87 \DeclareOption{12pt}{\csname ds@11pt\endcsname}

```

Similarly, ignore one/two-side options.

```

88 \DeclareOption{oneside}{\TBWarning{Option \CurrentOption\space ignored}}
89 \DeclareOption{twoside}{\ds@oneside}

```

There are these people who seem to think `tugproc` is an option rather than a class... (Note that it's already been filtered out if we were calling from `ltugproc`.)

```

90 \DeclareOption{tugproc}{%
91 \TBWarning{Option \CurrentOption\space ignored: use class ltugproc
92 instead of \@tugclass}%
93 }

```

Option `rawcite` (the default) specifies the default citation mechanism (as built-in to $\LaTeX$); option `harvardcite` specifies the author-date citation mechanism defined in section 3.24 below.

```

94 \DeclareOption{rawcite}{\let\if@Harvardcite\iffalse}
95 \DeclareOption{harvardcite}{\let\if@Harvardcite\iftrue}

```

Option `extralabel` (the default) specifies that the publication years of two successive references with otherwise identical labels will be tagged with distinguishing letters; option `noextralabel` causes those letters to be suppressed. Note that (a) no two references will in any case have the same labels in the default (plain) `rawcite` setup, and that (b) the distinguishing letters appear in the labels themselves; the reader can work out the correspondence one with the other...

```

96 \DeclareOption{extralabel}{\let\UseExtraLabel\@firstofone}
97 \DeclareOption{noextralabel}{\let\UseExtraLabel\@gobble}

```

The section-numbering style, so that we can allow the same heading layout as in the plain macros.

```

98 \DeclareOption{numbersec}{\let\if@numbersec\iftrue}
99 \DeclareOption{nonumber}{\let\if@numbersec\iffalse}

```

Minimal running headers/footers contain just the *TUGboat* volume/issue identification and page numbers. ‘`runningfull`’ is the default, and includes title and author. ‘`runningoff`’ makes both headers and footers empty.

```

100 \DeclareOption{runningoff}{\AtEndOfClass{\@tubrunningoff}}
101 \DeclareOption{runningminimal}{\AtEndOfClass{\@tubrunningminimal}}
102 \DeclareOption{runningfull}{\AtEndOfClass{\@tubrunningfull}}

```

Usually we want to print the doi if [final], else not. But sometimes we want to omit it even if [final], namely when we're posting a review or other item early.

```
103 \newif\iftubomitdoioption
104 \DeclareOption{omitdoi}{%
105   \tubomitdoioptiontrue
106 }
```

`\iftubtwocolumn` Occasionally (tb107jackowski, and past conference preprints), we need the option `onecolumn`. For alternative approaches to one-column articles, see tb92hagen-euler and tb78milo.

```
107 \newif\iftubtwocolumn \@tubtwocolumntrue
108 \DeclareOption{onecolumn}{\@tubtwocolumnfalse}
```

`\ifsecondcolstart` Occasionally, we need to start an article in the second column of a page, due to splicing with a previous article. Let's try declaring that. Then, before `\maketitle`, we'll force the move to the second column.

And sometimes we need to add space at the top of that second column (e.g., tb147lode-realtime); there's no way to intervene in the article source, so define a hook `\tubsecondcolstartextra`.

```
109 \newif\iftubsecondcolstart
110 \DeclareOption{secondcolstart}{\tubsecondcolstarttrue}
111 \let\tubsecondcolstartextra\relax
```

Any other options, we pass on to `article.cls` before we load it:

```
112 \DeclareOption*{\PassOptionsToClass{\CurrentOption}{article}}
```

Request default options (draft mode, standard citation, numbered sections, etc.), process all options, and then get the base document class on top of which we reside, namely `article`. Always call `article` with the `twoside` option, since we want the ability to have odd/even headers/footers.

```
113 \ExecuteOptions{draft,extralabel,numbersec,rawcite,runningminimal}
114 \ProcessOptions
115 \LoadClass[twoside]{article}
```

Various fonts used throughout. Some effort has been made to suppress these things with explicit sizes in the macro name (`\tensl` is an example below), but keeping in step with the documentation is one thing that restricts such a move.

```
116 \def\sectitlefont{\fontfamily\sfddefault\fontseries{bx}\fontshape{n}%
117   \fontsize\@xvipt\stbaselineskip\selectfont}
118 \def\tensl{\fontseries{m}\fontshape{sl}\fontsize\@xpt\@xipt
119   \selectfont}
```

This font selection command is used *only* for the 'Editor's Note' introduction to notes; sadly it makes explicit reference to CMR, and Barbara Beeton has agreed that the reference may be constructed to use the current family such that, if no upright italic is defined, ordinary italics are used. A project for later...

```
120 \ifTBunicodeengine
121   % there is no "LM unslanted" in OpenType, so use the standard cmu
122   % scaled for the current text size. Not worth more effort.
123   \def\EdNoteFont{\font\ednotefont = cmu10 at 1em }
124 \else % traditional engine:
```

```

125 \def\EdNoteFont{\fontfamily{cmr}\fontseries{m}\fontshape{ui}\selectfont}
126 \fi
127 \ltugboatcls

```

If Ulrik Vieth's `mflogo.sty` is around, we'll use it. Otherwise (pro tem, at least) we'll warn the user and define the absolute minimum of machinery that *TUGboat* requires (that which was used prior to the invention of L^AT_EX 2_ε).

```

128 \*common
129 \IfFileExists{mflogo.sty}%
130   {\RequirePackage{mflogo}}%
131 \ltugcomn {\TBWarning
132 \tugcomn} {\PackageWarning{ltugcomn}
133   {Package mflogo.sty not available --\MessageBreak
134     Proceeding to emulate mflogo.sty}
135 \DeclareRobustCommand{\logofamily}{%
136   \not@math@alphabet\logofamily\relax
137   \fontencoding{U}\fontfamily{logo}\selectfont}
138 \DeclareTextFontCommand{\textlogo}{\logofamily}
139 \def\MF{\textlogo{META}\tubdisc\textlogo{FONT}\@}
140 \def\MP{\textlogo{META}\tubdisc\textlogo{POST}\@}
141 \DeclareFontFamily{U}{logo}{}
142 \DeclareFontShape{U}{logo}{m}{n}{%
143   <8><9>gen*logo%
144   <10><10.95><12><14.4><17.28><20.74><24.88>logo10%
145 }{}
146 \DeclareFontShape{U}{logo}{m}{sl}{%
147   <8><9>gen*logosl%
148   <10><10.95><12><14.4><17.28><20.74><24.88>logosl10%
149 }{}
150 \DeclareFontShape{U}{logo}{m}{it}{%
151   <->ssub*logo/m/sl%
152 }{}%
153 }

```

3.2 Resetting at start of paper

`\ResetCommands` We store a set of commands that should be executed at the start of each paper, `\AddToResetCommands` before any paper-specific customisation. These commands (stored in the token register `\ResetCommands`) include things such as resetting section and footnote numbers, re-establishing default settings of typesetting parameters, and so on. The user (or more typically, editor) may execute the commands by using the command `\StartNewPaper`. Things I've not yet thought of may be added to the list of commands, by

```

154 \newtoks\ResetCommands
155 \ResetCommands{%
156   \setcounter{part}{0}%
157   \setcounter{section}{0}%
158   \setcounter{footnote}{0}%
159   \authornumber\z@
160 }
161 \newcommand{\AddToResetCommands}[1]{%
162   \AddToResetCommands\expandafter{\AddToResetCommands#1}%
163 }

```

3.3 Helpful shorthands (common code with Plain styles)

`\makeescape`, ..., `\makecomment` allow users to change the category code of a single character a little more easily. These require that the character be addressed as a control sequence: e.g., `\makeescape\/` will make `'/'` an escape character.

```

164 <!!latex>
165 \def\makeescape#1{\catcode'#1=0 }
166 \def\makebgroup#1{\catcode'#1=1 }
167 \def\makeegroup#1{\catcode'#1=2 }
168 \def\makemath #1{\catcode'#1=3 }
169 </!!latex>
170 <*latex>
171 \def\makeescape#1{\catcode'#1=\z@}
172 \def\makebgroup#1{\catcode'#1=\@ne}
173 \def\makeegroup#1{\catcode'#1=\tw@}
174 \def\makemath #1{\catcode'#1=\thr@@}
175 </latex>
176 \def\makealign #1{\catcode'#1=4 }
177 \def\makeeol #1{\catcode'#1=5 }
178 \def\makeparm #1{\catcode'#1=6 }
179 \def\makesup #1{\catcode'#1=7 }
180 \def\makesub #1{\catcode'#1=8 }
181 \def\makeignore#1{\catcode'#1=9 }
182 \def\makespace #1{\catcode'#1=10 }
183 \def\makeletter#1{\catcode'#1=11 }
184 \chardef\other=12
185 \let\makeother\@makeother
186 \def\makeactive#1{\catcode'#1=13 }
187 \def\makecomment#1{\catcode'#1=14 }

```

`\savecat#1` and `\restorecat#1` will save and restore the category of a given character. These are useful in cases where one doesn't wish to localize the settings and therefore be required to globally define or set things.

```

188 \def\savecat#1{%
189   \expandafter\xdef\csname\string#1savedcat\endcsname{\the\catcode'#1}}
190 \def\restorecat#1{\catcode'#1=\csname\string#1savedcat\endcsname}
191 <!!latex>\savecat\@
192 </!!latex>\makeletter\@

```

`\SaveCS#1` and `\RestoreCS#1` save and restore 'meanings' of control sequences. Again this is useful in cases where one doesn't want to localize or where global definitions clobber a control sequence which is needed later with its 'old' definition.

```

193 \def\SaveCS#1{\expandafter\let\csname saved@@#1\expandafter\endcsname
194   \csname#1\endcsname}
195 \def\RestoreCS#1{\expandafter\let\csname#1\expandafter\endcsname
196   \csname saved@@#1\endcsname}

```

To distinguish between macro files loaded

```

197 \def\plaintubstyle{plain}
198 \def\largetubstyle{latex}

```

Control sequences that were first defined in L^AT_EX 2_ε of 1995/06/01 (or later), but which we merrily use. Only define if necessary:

```

199 \providecommand\hb@xt@{\hbox to}
200 \providecommand\textsuperscript[1]{\ensuremath{\m@th
201             ^{\mbox{\fontsize\sf@size\z@
202             \selectfont #1}}}}

```

(Note that that definition of `\textsuperscript` isn't robust, but probably doesn't need to be... What's more, it doesn't appear in the mythical 2.09 version of the package.)

We end up wanting this fairly often, and L^AT_EX removed `\line`.

```

203 \def\tubline{\hbox to \hsize}

```

3.4 Abbreviations and logos

```

204 \def\tubdisc{\texorpdfstring{-}} % avoid warning about \- in section titles
205 \DeclareRobustCommand{\AllTeX}{%
206   \texorpdfstring{(La\kern-.075em)\kern-.05emTeX}{(La)TeX}}
207 \DeclareRobustCommand{\AI}{\acro{AI}}
208 \def\AMS{American Mathematical Society}
209 \def\AmS{\mathcal{A}\kern-.1667em\lower.5ex\hbox
210   {\mathcal{M}}\kern-.125em\mathcal{S}}
211 \def\AmSLaTeX{\AmS-LaTeX}
212 \def\AmSTeX{\AmS-TeX}
213 \def\ANSI{\acro{ANSI}}
214 \def\API{\acro{API}}
215 \def\ASCII{\acro{ASCII}}
216 \def\aw{\acro{A\kern.04em\raise.115ex\hbox{-}W}}
217 \def\AW{Addison\kern.1em-\penalty\z@\hskip\z@skip Wesley}
218 %
219 % make BibTeX work in slanted contexts too; it's common in titles, and
220 % especially burdensome to hack in .bib files.
221 \def\Bib{%
222   \ifdim \fontdimen1\font>0pt
223     B{\SMC\SMC IB}%
224   \else
225     B\textsc{ib}% LaTeX has more kerns, but they are too much to our eyes
226   \fi
227 }
228 \def\BibLaTeX{\texorpdfstring{\Bib\tubdisc\kern.02em LaTeX}{BibLaTeX}}
229 \def\BibTeX{\texorpdfstring{\Bib\tubdisc\kern-.04em TeX}{BibTeX}}
230 % no good way to determine bold font, and we want to lose the kern, too:
231 % (we \let BibTeX to this in maketitle)
232 \def\bfBib{B{\SMC\SMC IB}}
233 \def\bfBibTeX{\texorpdfstring{\bfBibTeX}{BibTeX}}
234 \def\bfBibLaTeX{\texorpdfstring{\bfBibLaTeX}{BibLaTeX}}
235 %
236 \def\BSD{\acro{BSD}}
237 \def\CandT{\textsl{Computers\& Typesetting}}
238 % must not define \CJK, because the CJK package does.

```

We place our `\kern` after `\-` so that it disappears if the hyphenation is taken:

```

239 \def\ConTeXt{\texorpdfstring{C\kern-.0333em\kern-.0667emTeX\kern-.0333emt}
240   {ConTeXt}}
241 \def\CMkIV{\ConTeXt\ MkIV}
242 \def\Cplusplus{Cplusplus}

```

```

243 %
244 % Turns out this original TUB definition has different output under
245 % pdfTeX and lua/xetex, because, ultimately, the axis in the OTF math
246 % cm fonts (newcm, latinmodern-math) fonts is at 57 units instead of 70
247 % units. Why Jacko did this is unknown, but can't be changed now.
248 %original TUGboat: \def\plusplus{\raisebox{.7ex}{$_{++}$}}
249 %
250 % We can't avoid OTF math in general. So we change the definition not to
251 % use math. The results are the same within a couple of decimal places
252 % (didn't seem to matter to make it exact), and it's simpler besides.
253 \def\plusplus{\texorpdfstring{\raise .351ex \hbox{\scriptsize ++}}{++}}
254 %
255 % consider rm vs. bold + tb139may-automata.ltx
256 \def\CPU{\acro{CPU}}
257 \def\CSzabbr{\ensuremath{\cal C}\kern-.1667em\lower.5ex\hbox{$\cal S$}}
258 \def\CSS{\acro{CSS}}
259 \def\CSTUG{\CSzabbr\kern.05em\acro{TUG}}
260 \def\CSV{\acro{CSV}}
261 \def\CTAN{\acro{CTAN}}
262 \def\DTD{\acro{DTD}}
263 \def\DTK{\acro{DTK}}
264 \def\DVD{\acro{DVD}}
265 \def\DVI{\acro{DVI}}
266 \def\DVIPDFMx{\acro{DVIPDFM}$x$}
267 \def\DVitoVDU{DVito\kern-.12em VDU}
268 \def\ECMA{\acro{ECMA}}
269 \def\EPS{\acro{EPS}}
270 % no line break at this hyphen please, and try to get a bold \varepsilon.
271 \def\TUBdefaultTeX{\ensuremath{\varepsilon}\mbox{-}\kern-.125em\TeX}%
272 \DeclareRobustCommand{\eTeX}{%
273   \ifx\f@series\bfseries@rm
274     \ifx\boldsymbol\undefined % \boldsymbol is from amsmath; also support bm?
275       \TBWarning{bold varepsilon for \string\eTeX\space not available; load amsmath}%
276       \TUBdefaultTeX
277     \else
278       \ensuremath{\boldsymbol{\varepsilon}}\mbox{-}\kern-.125em\TeX
279     \fi
280   \else
281     \TUBdefaultTeX
282   \fi
283 }
284 \DeclareRobustCommand{\ExTeX}{%
285   \ensuremath{\textstyle\varepsilon_{\kern-0.15em\cal{X}}}\kern-.2em\TeX}
286 \def\FAQ{\acro{FAQ}}
287 \def\FTP{\acro{FTP}}
288 \def\Ghostscript{Ghost\tubdisc script}
289 \def\GNU{\acro{GNU}}
290 \def\GUI{\acro{GUI}}
291 \DeclareRobustCommand{\HarfBuzz}{Harf\discretionary{-}{-}{\kern.02em}Buzz}
292 \def\Hawaii{Hawai'i}
293 \def\HTML{\acro{HTML}}
294 \def\HTTP{\acro{HTTP}}
295 \def\HTTPS{\acro{HTTPS}}
296 \def\iOS{i\acro{OS}}

```

```

297 \def\IDE{\acro{IDE}}
298 \def\IEEE{\acro{IEEE}}
299 \def\ISBN{\acro{ISBN}}
300 \def\ISO{\acro{ISO}}
301 \def\ISSN{\acro{ISSN}}
302 \def\JPEG{\acro{JPEG}}
303 \def\JTeX{\leavevmode\hbox{\lower.5ex\hbox{J}\kern-.18em\TeX}}
304 \def\JoT{\textsl{The Joy of \TeX}}
305 \DeclareRobustCommand{\KOMAScript}{\textsf{K}\kern.05em O\kern.05em%
306     M\kern.05em A\kern.1em\hyph\kern.1em Script}}
307 \def\LAMSTeX{L\raise.42ex\hbox{\kern-.3em
308     $\m@th$\fontsize\sf@size\z@\selectfont
309     $\m@th\mathcal{A}$}%
310     \kern-.2em\lower.376ex\hbox{$\m@th\mathcal{M}$}\kern-.125em
311     {\m@th\mathcal{S}$}-\TeX}
312 % This code is hacked from its definition of \cs{LaTeX}; it allows
313 % slants (for example) to propagate into the raised (small) 'A':
314 % \begin{macrocode}
315 \DeclareRobustCommand{\La}%
316     {\L\kern-.36em
317     {\setbox0\hbox{T}%
318     \vbox to\ht0{\hbox{$\m@th$%
319         \csname S@\fontsize\endcsname
320         \fontsize\sf@size\z@
321         \math@fontsfalse\selectfont
322         A}%
323         \vss}%
324     }}

```

We started with the intention that we wouldn't redefine `\LaTeX` when we're running under it, so as not to trample on an existing definition. However, this proves less than satisfactory; a single logo may be OK for the run of documents, but for *TUGboat*, we find that something noticeably better is necessary; see section 3.12.

```

325 \iflatex\def\LaTeX{\La\kern-.15em\TeX}
326 \def\LMTX{\acro{LMTX}}
327 \def\LuaHBTeX{Lua\acro{HB}\tubdisc\TeX}%
328 \def\LuaHBLaTeX{Lua\acro{HB}\tubdisc\LaTeX}%
329 \def\LuaLaTeX{Lua\tubdisc\LaTeX}% dtk-logos defines it
330 \def\LuaTeX{Lua\tubdisc\TeX}% ditto
331 \def\luatex{\LuaTeX}% ditto
332 \def\LyX{L\kern-.1667em\lower.25em\hbox{Y}\kern-.125emX}
333 \def\macOS{mac\acro{OS}}
334 \def\MacOSX{Mac\,\acro{OS},X}}
335 \def\MathML{Math\acro{ML}}
336 \def\Mc{\setbox\TestBox=\hbox{M}M\vbox
337     to\ht\TestBox{\hbox{c}\vfil}} % for Robert McGaffey

```

If we're running under $\text{\LaTeX} 2_{\varepsilon}$, we use Ulrik Vieth's `mflogo.sty` if it's present. Otherwise, we're using a short extract of Vieth's stuff. Either way, we don't need to specify `\MF` or `\MP`.

```

338 \def\mf{\textsc{Metafont}}
339 \def\MTFB{\textsl{The \MF\kern.1em\tubdisc book}}
340 \DeclareRobustCommand{\MiKTeX}{MiK\TeX}
341 \def\MkIV{Mk\acro{IV}}

```

```

342 \let\TB@@mp\mp
343 \DeclareRobustCommand{\mp}{\ifmmode\TB@@mp\else MetaPost\fi}
344 \def\mtex{T\kern-.1667em\lower.424ex\hbox{\^E}\kern-.125emX\@}
345 %
346 % In order that the \cs{OMEGA} command will switch to using the TS1
347 % variant of the capital Omega character if \texttt{textcomp.sty} is
348 % loaded, we define it in terms of the \cs{textohm} command. Note
349 % that this requires us to interpose a level of indirection, rather
350 % than to use \cs{let}\dots
351 % Revised definition of \cs{NTS} based on that used by Phil Taylor.
352 %
353 % \begin{macrocode}
354 \DeclareRobustCommand{\NTG}{\acro{NTG}}
355 \DeclareRobustCommand{\NTS}{\ensuremath{\mathcal{N}\mkern-4mu
356 \raisebox{-0.5ex}{\mathcal{T}}}\mkern-2mu \mathcal{S}}}
357 \DeclareTextSymbol{\textohm}{OT1}{'012}
358 \DeclareTextSymbolDefault{\textohm}{OT1}
359 \newcommand{\OMEGA}{\textohm}
360 \DeclareRobustCommand{\OCP}{\OMEGA\acro{CP}}
361 \DeclareRobustCommand{\OOXML}{\acro{OOXML}}
362 \DeclareRobustCommand{\OTF}{\acro{OTF}}
363 \DeclareRobustCommand{\OTP}{\OMEGA\acro{TP}}
364 \DeclareRobustCommand{\OpTeX}{\texorpdfstring{Op\kern-.05em\TeX}{OpTeX}}

365 \def\Pas{Pascal}
366 \def\pcMF{\leavevmode\raise.5ex\hbox{p\kern-.3\p@ c}MF\@}
367 \def\PCTeX{PC\thinspace\TeX}
368 \def\pcTeX{\leavevmode\raise.5ex\hbox{p\kern-.3\p@ c}\TeX}
369 \def\pdfLaTeX{pdf\slash\tubdisc\LaTeX}% dtk-logos
370 \def\pdflatex{\pdfLaTeX}
371 \def\pdfTeX{pdf\slash\tubdisc\TeX}% dtk-logos
372 \def\pdftex{\pdfTeX}
373 \def\PDF{\acro{PDF}}
374 \def\PDFUA{\acro{PDF/UA}}
375 \def\PGF{\acro{PGF}}
376 \def\PHP{\acro{PHP}}
377 \def\PiC{P\kern-.12em\lower.5ex\hbox{I}\kern-.075emC\@}
378 \def\PiCTeX{\PiC\kern-.11em\TeX}
379 \def\plain{\texttt{plain}}
380 \def\PNG{\acro{PNG}}
381 \def\POBox{P.\thinspace O.\thinspace Box }
382 \def\PS{{Post\tubdisc Script}}
383 \def\PSTricks{\acro{PST}ricks}
384 \def\RIT{\acro{RIT}}
385 \def\RTF{\acro{RTF}}
386 \def\SC{Steering Committee}
387 \def\SGML{\acro{SGML}}
388 \def\SliTeX{\textrm{S\kern-.06em\textsc{l}\kern-.035emi}%
389 \kern-.06em\TeX}}
390 \def\sLMF{\textsl{MF}} % should never be used
391 \def\SQL{\acro{SQL}}
392 \def\stTeX{\textsc{st}\kern-0.13em\TeX}
393 \def\STIX{\acro{STIX}}
394 \def\SVG{\acro{SVG}}
395 \def\TANGLE{\texttt{TANGLE}\@}

```

```

396 \def\TB{\textsl{The \TeX\tubdisc book}}
397 \def\TIFF{\acro{TIFF}}
398 \def\TP{\textsl{\TeX: The Program}}
399 \DeclareRobustCommand{\TeX}{T\kern-.1667em\lower.5ex\hbox{E}\kern-.125emX\@}
400 \def\TeXhax{\TeX hax}
401 \DeclareRobustCommand{\TeXLive}{\TeX Live}
402 \def\TeXMaG{\TeX M\kern-.1667em\lower.5ex\hbox{A}\kern-.2267emG\@}
403 \def\TeXtures{\textit{Textures}}
404 \let\Textures=\TeXtures
405 \def\TeXworks{\TeX\kern-.07em works}
406 \def\TeXXeT{\TeX-{}-\XeT}
407 \def\TFM{\acro{TFM}}
408 % \Thanh is defined below.
409 \def\TikZ{Ti/\em k}Z}
410 \def\TTN{\textsl{TTN}\@}
411 \def\TTN{\textsl{\TeX} and TUG News}}
412 \def\TUB{\texttub{TUGboat}}\def\texttub{\textsl} % redefined in some situations
413 \def\TUG{\TeX \UG}
414 \def\tug{\acro{TUG}}
415 \def\UG{Users Group}
416 \def\UNIX{\acro{UNIX}}
417 % Don't define \UTF, since other packages use it for Unicode character access.
418 % On the other hand, we want a macro for UTF-8 that doesn't break at a
419 % following -, as in \tbUTF-8.
420 \def\tbUTF{\acro{UTF}\futurelet\@nextchar\@tbUTFcheck}
421 \def\@tbUTFcheck{\ifx\@nextchar-%
422 \mbox{-}\let\next=\tbgobbledash
423 \else
424 \let\next=\empty
425 \fi\next}
426 \def\tbgobbledash-{}
427 \def\VAX{V\kern-.12em A\kern-.1em X\@}
428 \def\VnTeX{V\kern-.03em n\kern-.02em \TeX}
429 \def\VorTeX{V\kern-2.7\p@\lower.5ex\hbox{0}\kern-1.4\p@ R}\kern-2.6\p@\TeX}
430 \def\XeT{\texorpdfstring{X\kern-.125em\lower.424ex\hbox{E}\kern-.1667emT\@}{XeT}}
431 \def\XML{\acro{XML}}
432 \def\XMP{\acro{XMP}}
433 \def\WEB{\texorpdfstring{\texttt{WEB}\@}{WEB}}
434 \def\WEAVE{\texttt{WEAVE}\@}
435 \def\WYSIWYG{\acro{WYSIWYG}}

```

XeTeX requires reflecting the first E, hence we complain if the graphics package is not present. (For plain documents, this can be loaded via `miniltx` or `Eplain`.) Also, at Barbara's suggestion, if the current font is slanted, we rotate by 180 instead of reflecting so there is a better chance to look ok. (The magic values here seem more or less ok for `cmsl` and `cmti`.)

```

436 \def\tubreflect#1{%
437 \@ifundefined{reflectbox}{%
438 \TBEError{A graphics package must be loaded to use \string\XeTeX}
439 {Load graphicx or graphics.}%
440 }{%
441 \ifdim \fontdimen1\font>0pt
442 \raise 1.75ex \hbox{\kern.1em\rotatebox{180}{#1}}\kern-.1em
443 \else

```

```

444     \reflectbox{#1}%
445     \fi
446   }%
447 }
448 \def\tubhideheight#1{\setbox0=\hbox{#1}\ht0=0pt \dp0=0pt \box0 }
449 \def\XekernbeforeE{-.125em}
450 \def\XekernafterE{-.1667em}
451 % From Max, mail of 13sep24:
452 % hyperref is trying to expand \Xe to get a string for
453 % the embedded PDF table of contents, but \Xe is unsafe in an
454 % expansion-only context [even when defined with \DeclareRobustCommand,
455 % for reasons unknown].
456 % An easy way to fix this is to replace \DeclareRobustCommand with
457 % \NewDocumentCommand, which defines the macro as ‘‘\protected’’ instead
458 % as ‘‘\protect’’ed.
459 \NewDocumentCommand\tub@Xe{}{\leavevmode
460   \tubhideheight{\hbox{X%
461     \setbox0=\hbox{\TeX}\setbox1=\hbox{E}}%
462     \ifdim \fontdimen1\font>0pt
463       % XeTeX logo needs tinkering when slanted/italic font,
464       % so make kerns changeable
465       \def\XekernbeforeE{-.11em}%
466       \def\XekernafterE{-.16em}%
467       \dp1=-.17ex
468     \fi
469     \lower\dp0\hbox{\raise\dp1\hbox{\kern\XekernbeforeE\tubreflect{E}}}%
470     \kern\XekernafterE}}
471 % [But then,] For hyperref to be able to see the \texorpdfstring, it
472 % needs to be inside of a non-protected macro, but we still want the
473 % graphics commands to be protected, so we need to make a wrapper command:
474 \newcommand\Xe{\texorpdfstring{\tub@Xe}{Xe}}
475 \def\XeTeX{\texorpdfstring{\Xe\TeX}{XeTeX}}
476 \def\XeLaTeX{\texorpdfstring{\Xe{\kern.11em \LaTeX}}{XeLaTeX}}
477 %
478 \def\XHTML{\acro{XHTML}}
479 \def\XSL{\acro{XSL}}
480 \def\XSLF0{\acro{XSL}\raise.08ex\hbox{-}\acro{F0}}
481 \def\XSLT{\acro{XSLT}}
482 \def\YAML{\acro{YAML}}

```

3.5 General typesetting rules

```

483 \newlinechar='^^J
484 \clubpenalty=10000
485 \widowpenalty=10000
486 \def\NoParIndent{\parindent=\z@}
487 \newdimen\normalparindent \normalparindent=20\p@
488 \def\NormalParIndent{\global\parindent=\normalparindent}
489 \NormalParIndent
490 \def\BlackBoxes{\overfullrule=5\p@}
491 \def\NoBlackBoxes{\overfullrule=\z@}
492 \def\newline{\hskip\z@\@plus\pagewd\break}
493 \def\tubnewline{\texorpdfstring{\newline}{ }}

```

`\Thanh` Han The Thanh's name is complicated because it has a double-accented character over the e, U+1EBF (it also has grave accents over the a's, but they aren't a problem): Hàn Thế Thành. Many fonts do not have this character; hardly any, in the old days, and definitely not Computer Modern.

So, when the e-circumflex-acute character in Thanh's name is not in the font, we'll use the following definition. Until fall 2025, we used a definition that put the acute vertically over the e. But then Barbara said, and Thanh agreed, that it is preferable nowadays to put the acute to the right of the circumflex. So that's what this definition does:

```
494 \def\TBecircacute{\~e\llap{\raise 0.3ex\hbox{\'{}\kern-0.4ex}}}%
```

For posterity, this was our previous definition, used for many years, which put the acute directly over the circumflex:

```
%\def\Thanh{H\~{an}~Th\~e\llap{\raise 0.5ex\hbox{\'{}\kern-0.4ex}}~Th\~{anh}}
```

Here's a way to get at our new definition in case we overwrite it below. This is because the `lm` fonts, which are the $\text{\LaTeX}$ default with Unicode engines, put the acute vertically over the circumflex (perhaps since *TUGboat* did), and we'll want to change that.

```
495 \let\TBecircacutebuilt=\TBecircacute
```

If running under `pdf $\text{\LaTeX}$` , allow the actual UTF-8 character to be used in text. `\DeclareUnicodeCharacter` is not defined under the Unicode engines. A possible definition is given here: <https://tex.stackexchange.com/questions/195458>, but let's wait to see if we need it.

```
496 \ifx\DeclareUnicodeCharacter\undefined\else
497   \DeclareUnicodeCharacter{1EBF}{\TBecircacute} % for the literal character
498 \fi
```

Under Unicode engines, by default, we'll use the character from the font, if it's there.

```
499 \ifTBunicodeengine
500   \AtBeginDocument{% in case a different font gets loaded
501     % \iffontchar is from e-TeX; safe to use under Unicode engines.
502     \iffontchar\font"1EBF
503       % Use the font's version by default; perhaps we're using newcm or
504       % another font that has the acute offset to the right.
505       \def\TBecircacute{\char"1EBF }%
506     \fi
507   }%
508 \fi % (end TBunicodeengine)
509
510 % Finally, we can define \Thanh itself, using \TBecircacute.
511 \def\Thanh{H\~{an}~Th\TBecircacute~Th\~{anh}}%
```

One more complication: add tagging for Thanh using the UTF-8 characters. This comes from Frank Mittelbach, written for <https://tug.org/TUGboat/tb47-1/tb145sojka-doctor-mittelbach.pdf>.

```
512 \ExplSyntaxOn
513 \IfPDFManagementActiveT{
514   \AddToHook{cmd/Thanh/before}
515   {
516     \mode_leave_vertical:
517     \tag_mc_end_push:
```

```

518 % We could use ^^ notation with the UTF-8 bytes (no ^^^ since that
519 % doesn't work under pdftex). Maybe later.
520 \tag_struct_begin:n {tag=Span,actualtext=Hàn~ Thế~ Thành}
521 \tag_mc_begin:n{ }
522 }
523 \AddToHook{cmd/Thanh/after}
524 {
525   \tag_mc_end:
526   \tag_struct_end:
527   \tag_mc_begin_pop:n{ }
528 }
529 } % (end PDFManagementactiveT)
530 \ExplSyntaxOff

```

See the TUGboat test file `tubibthanh.tex` (and the Makefile) to exercise all this.

\tubsentencespace Occasionally, notably after citations that need to come after a sentence-ending period, we want to tell T_EX that it's still at the end of a sentence. As in: `... whatever. \cite{foo}\tubsentencespace` This happens when, e.g., the reference applies to more than the final sentence. Also can be needed when `\@` cannot be used because the sentence-ending punctuation itself occurs inside a control sequence that prevents it.

```

531 \def\tubsentencespace{\spacefactor=3000{}\space\ignorespaces}

```

\tubdots Latin Modern and many other fonts irritatingly make the Unicode ellipsis character (U+2026) a single character's width, typically more squashed together than three period characters. This just looks wrong. It is too painful to try to redefine in general, but provide the normal definition to reset in individual papers with, e.g.: `\ifx\tubdots\undefined \else \let\dots\tubdots \let\ldots\tubdots \fi`

The `plain.tex` definition does not have the small space before the first dot, but that space makes the result look better in cases like `[\tubdots]` where something other than a space comes before the ellipsis.

```

532 \def\tubdots{\ifmmode\mathellipsis\else
533   \kern\fontdimen3\font % space before first dot
534   .\kern\fontdimen3\font
535   .\kern\fontdimen3\font
536   .\kern\fontdimen3\font\fi}

```

\allowhyphens Hyphen control: first, we save (via `\edef`) the hyphenpenalties in `\allowhyphens`. This allows us to permit hyphens temporarily in things like `\netaddresses`, which typically occur when `\raggedright` is set, but which need to be allowed to break at their artificial discretionaries.

```

537 \edef\allowhyphens{\noexpand\hyphenpenalty\the\hyphenpenalty\relax
538   \noexpand\exhyphenpenalty\the\exhyphenpenalty\relax}
539 \def\nohyphens{\hyphenpenalty\@M\exhyphenpenalty\@M}

```

3.6 Utility registers and definitions

We define a few scratch registers (and the like) for transient use; they're all paired: an internal one (`\T@st*`) and an external one (`\Test*`).

Comment: Exercise for an idle day: find whether all these are necessary, or whether we can use the L^AT_EX temporaries for some (or all) of the \T_{st}* ones.

Comment: (bb) All these registers are used in the plain version, tugboat.sty.

```
540 \newbox\T@stBox          \newbox\TestBox
541 \newcount\T@stCount      \newcount\TestCount
542 \newdimen\T@stDimen      \newdimen\TestDimen
543 \newif\ifT@stIf          \newif\ifTestIf
```

Control sequence existence test, stolen from T_EXbook exercise 7.7 (note that this provides functionality that in some sense duplicates something within L^AT_EX).

```
544 \def\ifundefined#1{\expandafter\ifx\csname#1\endcsname\relax }
```

L^AT_EX conventions which are also useful here.

```
545 <!!latex>
546   \let\@@input\input
547   \def\iinput#1{\@@input#1 }
548   \def\@inputcheck{\if\@nextchar\bgroup
549     \expandafter\iinput\else\expandafter\@input\fi}
550   \def\input{\futurelet\@nextchar\@inputcheck}
551 </!!latex>
```

Smashes repeated from AMS-T_EX; plain T_EX implements only full \smash.

```
552 \newif\iftop@            \newif\ifbot@
553 \def\topsmash{\top@true\bot@false\smash@}
554 \def\botsmash{\top@false\bot@true\smash@}
555 \def\smash{\top@true\bot@true\smash@}
556 \def\smash@{\relax\ifmmode\def\next{\mathpalette\mathsm@sh}%
557   \else\let\next\makesm@sh\fi \next }
558 \def\finsm@sh{\iftop@\ht\z@ \z@\fi\ifbot@\dp\z@ \z@\fi\box\z@}
```

Vertical ‘laps’; cf. \llap and \rlap

```
559 \long\def\ulap#1{\vbox to \z@{\vss#1}}
560 \long\def\dlap#1{\vbox to \z@{\#1\vss}}
```

And centered horizontal and vertical ‘laps’

```
561 \def\xlap#1{\hb@xt@\z@{\hss#1\hss}}
562 \long\def\ylap#1{\vbox to \z@{\vss#1\vss}}
563 \long\def\zlap#1{\ylap{\xlap{#1}}}
```

Avoid unwanted vertical glue when making up pages.

```
564 \def\basezero{\baselineskip\z@skip \lineskip\z@skip}
```

Empty rules for special occasions

```
565 \def\nullhrule{\hrule \@height\z@ \@depth\z@ \@width\z@ }
566 \def\nullvrule{\vrule \@height\z@ \@depth\z@ \@width\z@ }
```

Support ad-hoc strut construction.

```
567 \def\makestrut[#1;#2]{\vrule \@height#1 \@depth#2 \@width\z@ }
```

Construct box for figure pasteup, etc.; height = #1, width = #2, rule thickness = #3

```
568 \def\drawoutlinebox[#1;#2;#3]{\T@stDimen=#3
569   \vbox to#1{\hrule \@height\T@stDimen \@depth\z@
```

```

570          \vss\hb@xt@#2{\vrule \@width\T@stDimen
571              \hfil\makestrut[#1;\z@]}%
572              \vrule \@width\T@stDimen}\vss
573          \hrule \@height\T@stDimen \@depth\z@}}

```

Today's date, to be printed on drafts. Based on T_EXbook, p.406.

```

574 <!!latex>
575 \def\today{\number\day\space \ifcase\month\or
576     Jan \or Feb \or Mar \or Apr \or May \or Jun \or
577     Jul \or Aug \or Sep \or Oct \or Nov \or Dec \fi
578     \number\year}
579 </!!latex>

```

Current time; this may be system dependent!

```

580 \newcount\hours
581 \newcount\minutes
582 \def\SetTime{\hours=\time
583     \global\divide\hours by 60
584     \minutes=\hours
585     \multiply\minutes by 60
586     \advance\minutes by-\time
587     \global\multiply\minutes by-1 }
588 \SetTime
589 \def\now{\ifnum\hours<10 0\fi\number\hours:%
590     \ifnum\minutes<10 0\fi\number\minutes}
591 \def\Now{\today\ \now}
592 \newif\ifPrelimDraft % true if ([draft] or [preprint] or pageno>900)
593 \def\midrttitle{} % center of running heads
594 \def\rtitlenexttopage{\ifPrelimDraft \textsl{\small draft: \Now}\fi}
595 %\def\rtitlenexttopage{\ifnum\value{page}>900 \textsl{\small draft: \Now}\fi}

```

Sometimes we want to refer to the pages of another article in the same issue. `tugboat.dates` makes the real definition; here we define a placeholder so it won't be undefined when we send the source back to the author.

```

596 \let\thisissuepageref\empty

```

3.7 Ragged right and friends

`\raggedskip` Plain T_EX's definition of `\raggedright` doesn't permit any stretch, and results in too many overfull boxes. We also turn off hyphenation. This code lies somewhere between that of Plain T_EX and of L^AT_EX.

`\raggedspaces`

```

597 \newdimen\raggedskip \raggedskip=\z@
598 \newdimen\raggedstretch \raggedstretch=5em % ems of font set now (10pt)
599 \newskip\raggedparfill \raggedparfill=\z@\@plus 1fil
600 \def\raggedspaces{\spaceskip=.3333em \relax \xspaceskip=.5em \relax }

```

`\raggedright` Some applications may have to add stretch, in order to avoid all overfull boxes.

`\raggedleft` We define the following uses of the above skips, etc.

`\raggedcenter`

```

601 \def\raggedright{%
602     \nohyphens \raggedspaces
603     \rightskip=\raggedskip\@plus\raggedstretch
604     \parfillskip=\raggedparfill
605 }
606 \def\raggedleft{%

```

`\normalspaces`

```

607 \nohyphens \raggedspaces
608 \leftskip=\raggedskip\@plus\raggedstretch
609 \parfillskip=\z@skip
610 \let\ \@centercr % else tabulararray fails,
611                % https://github.com/lvjr/tabulararray/issues/348
612 }
613 \def\raggedcenter{%
614 \nohyphens \raggedspaces
615 \leftskip=\raggedskip\@plus\raggedstretch
616 \rightskip=\leftskip
617 \parindent=\z@
618 \parfillskip=\z@skip
619 }
620 %
621 % Undo |\raggedspaces|.
622 \def\normalspaces{\spaceskip\z@skip \xspaceskip\z@skip}

\tubjustifiedpar Undo the \raggedright (or other such) settings, restoring normality.
623 \def\tubjustifiedpar{\rightskip=0pt \parfillskip=0pt plus1fil
624                \allowhyphens \normalspaces}

```

3.8 Assorted user-level markup

We provide a new definition of `~` by redefining `\` (`\DeclareRobustCommand` doesn't mind redefinition, fortunately). This is based on the version in AMS-`TeX`—the `LATeX 2ε` version (`ltspace.dtx`) has `\leavevmode` and does not do anything with the surrounding space(s). Our version messes up with the `\pfill` used in doc-generated indexes (github.com/latex3/latex2e/issues/75), but later (2018++) versions of doc should protect against our redefinition.

```

625 \let\latexnobreakspace=\nobreakspace
626 \DeclareRobustCommand{\nobreakspace}{\unskip\nobreak\ \ignorespaces}

```

Plain `TeX` defines `\newbox` as `\outer`. We solemnly preserve the following, which removes the `\outerness`; of course, we carefully exclude it from what we generate... (`\outerness` is a spawn of the devil, is it not? Barbara Beeton responded to the previous sentence “`\outerness` has its place: it avoids register buildup, hence running out of memory”. In another context, David Carlisle remarked that an error control mechanism that causes more confusing errors than it prevents is rather a poor one. This is perhaps not the place to conduct a serious debate...)

```

627 \def\boxcs#1{\box\csname#1\endcsname}
628 \def\setboxcs#1{\setbox\csname#1\endcsname}
629 \def\newboxcs#1{\expandafter\newbox\csname#1\endcsname}
630 \let\gobble\@gobble
631 \def\vellipsis{%
632 \leavevmode\kern0.5em
633 \raise\p@\vbox{\baselineskip6\p@\vskip7\p@\hbox{.}\hbox{.}\hbox{.}}
634 }
635 % \bull doesn't work with tagging; requires ActualText using, e.g.,
636 % accsup, but the ActualText is ignored since it's just a rule.
637 % (Lots of our other commands also are not properly tagged.)
638 % https://github.com/latex3/tagging-project/pull/535

```

```

639 \def\bull{\vrule \@height 1ex \@width .8ex \@depth -.2ex }
640 \DeclareRobustCommand{\cents}{\textcent}
641 \def\tubcentsold{{\rm\raise.2ex\rlap{\kern.05em$\scriptstyle/$}c}}
642 \def\careof{\leavevmode\hbox{\raise.75ex\hbox{c}\kern-.15em
643         /\kern-.125em\smash{\lower.3ex\hbox{o}}}\ignorespaces}
644 \def\Dag{\raise .6ex\hbox{$\scriptstyle\dagger$}}
645 %
646 \DeclareRobustCommand{\sfrac}[1]{\@ifnextchar/{\@sfrac{#1}}%
647         {\@sfrac{#1}/}}
648 \def\@sfrac#1/#2{\leavevmode\kern.1em\raise.5ex
649         \hbox{$\m@th\mbox{\fontsize\sf@size\z@
650             \selectfont#1}$}\kern-.1em
651         /\kern-.15em\lower.25ex
652         \hbox{$\m@th\mbox{\fontsize\sf@size\z@
653             \selectfont#2}$}}
654 %
655 % don't stay bold in description items, bold italic is too weird.
656 \DeclareRobustCommand\meta[1]{%
657     \ensuremath{\langle}%
658     \ifmmode \expandafter\mbox \fi % if in math
659     {\it #1/}% no typewriter italics, please
660     \ensuremath{\rangle}%
661 }
662 %
663 % Use \tt rather than \texttt because italic typewriter is just too strange
664 % and upright works well enough in both italic and bold contexts.
665 % Would be nice to change catcode of _ for \LaTeX3, but we don't.
666 %
667 % By the way, it would be possible to substitute typewriter slanted for
668 % typewriter italic in general:
669 % \url{https://tex.stackexchange.com/questions/692277}.
670 % But it feels like that is too intrusive a change, even though in
671 % practice we always prefer tt slanted.
672 \DeclareRobustCommand{\cs}[1]{\texorpdfstring
673     {\tt \char'\@#1}\@}%
674     {\textbackslash #1}%
675 }
676 %
677 % This command was defined much later than the others around here, so
678 % let's not conflict with any existing definitions that might be out there.
679 % Don't allow hyphenations or other line breaks.
680 \DeclareRobustCommand{\tubbraced}[1]{\texorpdfstring
681     {\mbox{\texttt{\char'\@#1\char'\@}}}%
682     {\textbraceleft #1\textbraceright}%
683 }
684 %
685 % Literal text, such as class names, package names, filenames, etc,
686 % Trying to define separate commands for each seems impossible and pointless.
687 % Usually we don't want hyphenation or any other kind of break.
688 \DeclareRobustCommand{\tbcodes}[1]{\mbox{\texttt{#1}}}
689 %
690 % On the other hand, sometimes we need to break such code fragments.
691 % If |hyperref| is loaded, we want |\nolinkurl|, else just |\url|.
692 \AtBeginDocument{%

```

```

693 \ifx\nolinkurl\undefined
694   \DeclareRobustCommand{\tbcodebreak}{\url}
695 \else
696   \DeclareRobustCommand{\tbcodebreak}{\nolinkurl}
697 \fi
698 }
699 %
700 % Not sure why we ever want this instead of LaTeX's \, (using \kern),
701 % but fine, just keeping it.
702 \DeclareRobustCommand{\thinskip}{\hskip 0.16667em\relax}
703 %
704 % Ah, urls. Nowadays, we like the visible url to not have any protocol,
705 % if it is \texttt{http://} or \texttt{https://}. But we need to include
706 % the protocol if we are making live links, since a string like
707 % \texttt{tug.org/whatever} will be taken as a local filename by
708 % browsers and PDF readers. Since we need to check for
709 % \texttt{hyperref}, make the definition \cs{AtBeginDocument}. In the
710 % end, \cs{tbsurl}\tubbraced{foo} produces \texttt{https://foo} and
711 % \cs{tbhurl}\tubbraced{foo} produces \texttt{http://foo}.
712 %
713 \AtBeginDocument{%
714 \ifx\hyper@normalise\undefined
715   \ifx\url\undefined      % define our own simplistic non-hyperref \url
716     \def\url{\begingroup  % might as well catch common special chars
717       \catcode'\#=12 \catcode'\$=12 \catcode'\%=12 \catcode'\^=12
718       \catcode'\&=12 \catcode'\_ =12 \catcode'\~=12
719       \finish@tub@url}
720   \def\finish@tub@url#1{\tt #1\endgroup}
721   \fi
722   \let\tburl\url  % no hyperref, so just \url is fine;
723   \let\tbsurl\url % \let instead of \def so we can still
724   \let\tbhurl\url % use \def\url{\tbsurl} without infloop.
725 \else
726   % This hyperref hook-in is due to Ulrike Fischer.
727   % \url{https://github.com/latex3/hyperref/issues/125}.
728   % \tb[sh]url@ are defined next.
729   \DeclareRobustCommand*\tburly{\tbsurl}%
730   \DeclareRobustCommand*\tbsurly{\hyper@normalise\tbsurl}%
731   \DeclareRobustCommand*\tbhurly{\hyper@normalise\tbhurl}%
732 \fi
733 }
734 %
735 % Outside \AtBeginDocument, back at the top level of the dtx, we
736 % turn on expl syntax for the main definitions of \tb[sh]url. We want
737 % to auto-remove an explicit protocol in case it
738 % was given.
739 %
740 % Giving \verb|https://| to \cs{tbhurl} generates an invalid link; in
741 % practice there's no use for that so we don't bother to check for it.
742 %
743 \ExplSyntaxOn
744 % Helper function to test whether #1 matches the leading characters of #2.
745 \cs_new:Npn \str_if_starts_with:nnTF #1#2
746 {

```

```

747 % #1 = prefix to test
748 % #2 = full string
749 %
750 % compare the first N characters (tokens) of the full string #2,
751 % where N is the length of the prefix #1,
752 % to the prefix.
753 \str_if_eq:eeTF
754 { \str_range:nnn {#2} {1} { \str_count:n {#1} } }
755 {#1}
756 }
757
758 % the main function to handle url #1.
759 \def\tbsurl@#1 % https
760 {
761 \group_begin:
762 %
763 % \url uses math mode for typesetting; fvextra sets \everymath to
764 % play with the catcode of -, among others, which ends up causing an
765 % infinite loop when we try to typeset the DOI url.
766 % https://github.com/gpoore/fvextra/issues/39
767 %
768 % There's no sensible way to use \everymath with \url, so clear it
769 % (suggested to the url.sty author). url.sty also doesn't handle an
770 % active hyphen in general: https://tex.stackexchange.com/questions/114704
771 \everymath = {}
772 %
773 % URL encoding for when pdf management (\DocumentMetadata) is active,
774 % else a % in the input url ends up as a %25 in the active link.
775 % From Ulrike, 4apr26.
776 \IfPDFManagementActiveT{
777 \bool_if:NTF \l__hyp_href_url_encode_bool
778 {
779 \tl_set:Nn \l__hyp_text_enc_uri_print_tl {utf8/URI}
780 }
781 {
782 \tl_set:Nn \l__hyp_text_enc_uri_print_tl {utf8/string}
783 }
784 }% end pdf management
785 %
786 \str_set:Nn \l_tmpa_str {#1}
787 %
788 % This checks if http:// is contained anywhere within the argument url,
789 % but that was not good enough, since http:// might occur inside the url,
790 % e.g., https://web.archive.org/web/20090809184749/http://www.eco-log.de/
791 % \str_if_in:NnTF \l_tmpa_str {http://}
792 %
793 % Instead we check for "starts with http://".
794 % Another fix would have been to insert a constant string "foo" before
795 % the argument and the prefix, but since the clean solution is at hand,
796 % might as well use it.
797 \str_if_starts_with:nnTF {http://} {#1}
798 {
799 \tbhurl@{#1} % if http, redirect to remove protocol
800 % this version prints the http, as we originally thought was better.

```

```

801 % \expandafter\hyper@linkurl
802 % \expandafter{\expandafter\Hurl\expandafter{\l_tmpa_str}}{\l_tmpa_str}
803 }
804 {
805   \str_remove_once:Nn \l_tmpa_str {https://}
806   \expandafter\hyper@linkurl
807   \expandafter{\expandafter\Hurl\expandafter{\l_tmpa_str}}
808   {https://\l_tmpa_str}
809 }
810 \group_end:
811 } % end \tbsurl@
812 %
813 % explicit http, handle similarly but don't bother checking for https
814 \def\tbhurl@#1
815 {
816   \str_set:Nn\l_tmpa_str{#1}
817   \str_remove_once:Nn \l_tmpa_str {http://}
818   \expandafter\hyper@linkurl\expandafter{\expandafter\Hurl\expandafter
819   {\l_tmpa_str}}{http://\l_tmpa_str}
820 }
821 \ExplSyntaxOff
822 %
823 % Now let's use those macros for putting a url into a simple
824 % ragged-right footnote.
825 \def\tburlfootnote{\tbsurlfootnote}
826 \def\tbsurlfootnote#1{\unskip\footnote{\raggedright\tbsurl{#1}}}
827 \def\tbhurlfootnote#1{\unskip\footnote{\raggedright\tbhurl{#1}}}
828 %
829 % Close up space between footnote mark and punctuation ('pre-punctuation').
830 \DeclareRobustCommand{\tbppkernfoot}{\tubthinnerespace}
831
832 % Make \! work in text mode, for older LaTeX.
833 \DeclareRobustCommand{\!}{\ifmmode\mskip-\thinmuskip \else\kern-0.16667em \fi}
834 %
835 % Half a thinspace, positive and negative. Should have named these
836 % \cs{tb} instead of \cs{tub}, but not worth changing now.
837 \DeclareRobustCommand{\tubthinnerespace}
838   {\ifmmode\mskip.5\thinmuskip \else\kern0.08333em \fi}
839 \DeclareRobustCommand{\tubthinnerespaceneg}
840   {\ifmmode\mskip-.5\thinmuskip \else\kern-0.08333em \fi}
841 %
842 % Half a smallskip.
843 \DeclareRobustCommand{\tubsmallerskip}
844   {\vskip 1.5pt plus .75pt minus .75pt\relax}
845 %

```

We play a merry game with dashes, providing all conceivable options of breakability before and after.

```

846 \def\endash{--}
847 \def\emdash{\endash-}
848 \def\d@sh#1#2{\unskip#1\thinspace#2\thinspace\ignorespaces}
849 \def\dash{\d@sh\nobreak\endash}
850 \def\Dash{\d@sh\nobreak\emdash}
851 \def\ldash{\d@sh\empty{\hbox{\endash}\nobreak}}

```

```

852 \def\rdash{\d@sh\nobreak\endash}
853 \def\Ldash{\d@sh\empty{\hbox{\emdash}\nobreak}}
854 \def\Rdash{\d@sh\nobreak\emdash}

```

Hacks to permit automatic hyphenation after an actual hyphen, or after a slash.

```

855 \def\hyph{-\penalty\z@\hskip\z@skip }
856 \def\slash{/\penalty\z@\hskip\z@skip }

```

Adapted from comp.text.tex posting by Donald Arseneau, 26 May 93.
 $\text{\LaTeX 2}_{\epsilon}$ -isation added by Robin Fairbairns. Destroys both the TestCounts.

```

857 \def\nth#1{%
858   \def\reserved@a##1##2\@nil{\ifcat##1n%
859     0%
860     \let\reserved@b\ensuremath
861     \else##1##2%
862     \let\reserved@b\relax
863     \fi}%
864   \TestCount=\reserved@a#1\@nil\relax
865   \ifnum\TestCount < 0 \multiply\TestCount by\m@ne \fi % subdue negatives
866   \T@stCount=\TestCount
867   \divide\T@stCount by 100 \multiply\T@stCount by 100
868   \advance\TestCount by-\T@stCount % n mod 100
869   \ifnum\TestCount > 20 \T@stCount=\TestCount
870     \divide\T@stCount by 10 \multiply\T@stCount by 10
871     \advance\TestCount by-\T@stCount % n mod 10
872   \fi
873   \reserved@b{#1}%
874   \textsuperscript{\ifcase\TestCount th%      0th
875     \or st%      1st
876     \or nd%      2nd
877     \or rd%      3rd
878     \else th%    nth
879   \fi}%
880 }

```

3.9 Reviews

Format information on reviewed items for book review articles. For the $\text{\LaTeX 2}_{\epsilon}$ version, we follow Fairbairns' maxim, and define something that can even look like a $\text{\LaTeX}$ macro...

```

881 \def\Review{\@ifnextchar:{\@Review}{\@Review:}}
882 \def\@Review:{\@ifnextchar[%
883   {\@Rev}%
884   {\@Rev[Book review]}}
885 \def\@Rev[#1]#2{{\ignorespaces#1\unskip:\enspace\ignorespaces
886   \slshape\mdseries#2}}
887 \def\reviewitem{\addvspace{\BelowTitleSkip}%
888   \def\revauth##1{\def\therevauth{##1, }\ignorespaces}%
889   \def\revtitle##1{\def\therevtitle{{\slshape##1}. }\ignorespaces}%
890   \def\revpubinfo##1{\def\therevpubinfo{##1.}\ignorespaces}%
891 }
892 \def\endreviewitem{{\noindent\interlinepenalty=10000

```

```

893 \therevauth\therevtitle\therevpubinfo\endgraf}%
894 \vskip\medskipamount
895 }
896 \def\titleref#1{\slshape\frenchspacing#1\nocorr}}
897 \let\booktitle=\titleref % older name

```

3.10 Dates, volume and issue numbers, etc.

Dates and other items which identify the volume and issue. `\issueseqno` is a sequential issue number starting from the first issue published; volume 15,4 has `\issueseqno=45`.

```
\vol 19, 1.
```

To use: `\issdate March 1998`.

```
\issueseqno=58
```

Starting with volume 23 (nominal 2002), we have `\issyear` instead of `\issdate`, because issues don't have months any more.

For production, these are set in a separate file, `tugboat.dates`, which is issue-specific.

```

898 \newcount\issueseqno \issueseqno=-1
899 \def\volx{\gdef\volx{Volume~\volno~(\volyr), No.~\issno}}
900 \def\volyr{}
901 \def\volno{}
902 \def\vol#1, #2.{%
903     \gdef\volno{#1}%
904     \gdef\issno{#2}%
905     \setbox\TestBox=\hbox{\volyr}%
906     \ifdim \wd\TestBox > .2em \volx \fi }
907 \def\issyear#1.{%
908     \gdef\issdt{#1}\gdef\volyr{#1}%
909     \gdef\bigissdt{#1}%
910     \setbox\TestBox=\hbox{\volno}%
911     \ifdim \wd\TestBox > .2em \volx \fi }
912 \def\issdate#1#2 #3.{%
913     \gdef\issdt{#1#2 #3}\gdef\volyr{#3}%
914     \gdef\bigissdt{#1{\smc\uppercase{#2}} #3}%
915     \setbox\TestBox=\hbox{\volno}%
916     \ifdim \wd\TestBox > .2em \volx \fi }
917 % The \vol command must be invoked precisely like this, including spaces.
918 % Since we are the only ones who write it, we can be strict.
919 \vol 0, 0.
920 \issdate Thermidor, 9999.

```

(The curious may like to know that *Thermidor* was one of the French revolutionary month names.)

For L^AT_EX use, define a version of the issue declaration that can take or leave the old plain syntax

```

921 <!\latex>\def\tubissue#1(#2)%
922 <*\latex>
923 \def\tubissue#1{\@ifnextchar(%)
924     {\@tubissue@b{#1}}
925     {\@tubissue@a{#1}}}
926 \def\@tubissue@b#1(#2){\@tubissue@a{#1}{#2}}

```

```

927 \def\tubissue@a#1#2%
928 \</latex>
929 {\TUB~#1, no.~#2}

```

TUGboat conventions include the sequential issue number in the file name. Permit this to be incorporated into file names automatically. If issue number = 11, `\Input filnam` will read `tb11filnam.tex`

```

930 \def\infil@{\jobname}
931 \def\Input #1 {\ifnum\issueseqno<0
932   \def\infil@{#1}%
933   \else
934   \def\infil@{tb\number\issueseqno#1}
935   \fi
936   \edef\jobname{\infil@}\@readFLN
937   @@input \infil@relax
938   \if@RMKopen
939   \immediate\closeout\@TBremarkfile\@RMKopenfalse
940   \fi
941 }

```

`\TBremarks` are things that need to be drawn to the attention of the editors; the conscientious author will include such things in the article file. By default, remarks are suppressed, but their appearance may be enabled by the `\TBenableRemarks` command, which can be included in the configuration file `ltugboat.cfg` (or `ltugproc.cfg`, if that's what we're at).

```

942 \newif\if@RMKopen \@RMKopenfalse
943 \newwrite\@TBremarkfile
944 \def\@TBremark#1{%
945   \if@RMKopen
946   \else
947   \@RMKopentrue\immediate\openout\@TBremarkfile=\infil@.rmk
948   \fi
949   \toks@={#1}%
950   \immediate\write\@TBremarkfile{^^J\the\toks@}%
951   \immediate\write16{^^JTBRemark:: \the\toks@^^J}%
952 }

```

We initialise `\TBremark` to ignore its argument (this used to involve a `\TBremarkOFF` which was cunningly defined exactly the same as `\gobble`)

```

953 \let\TBremark=\gobble

```

`\TBenableRemarks` simply involves setting `\TBremark` to use the functional `\@TBremark` defined above.

```

954 \def\TBenableRemarks{\let\TBremark\@TBremark}

```

For marking locations in articles that pertain to remarks in another file of editorial comments

```

955 \def\TUBedit#1{}

```

For using different filenames in the production process than those supplied by authors

```

956 \def\TUBfilename#1#2{\expandafter\def\csname file@@#1\endcsname{#2}}
957 \newread\@altfilenames

```

```

958 \def\@readFLN{\immediate\openin\@altfilenames=\jobname.fln
959 \ifeof\@altfilenames\let\@result\relax\else
960 \def\@result{\@input\jobname.fln }\fi
961 \immediate\closein\@altfilenames
962 \@result}
963 \@readFLN
964 \everyjob=\expandafter{\the\everyjob\@readFLN}
965 \InputIfFileExists{\jobname.fln}%
966 {\TBInfo{Reading alternative file \jobname.fln}}
967 {}

```

The following needs to work entirely in T_EX's mouth

```

968 \def\@tubfilename#1{\expandafter\ifx\csname file@@#1\endcsname\relax
969 #1\else\csname file@@#1\endcsname\fi}
970 \def\fileinput#1{\@input\@tubfilename{#1} }

```

Write out (both to a file and to the log) the starting page number of an article, to be used for cross references and in contents. `\pagexref` is used for articles fully processed in the *TUGboat* run. `\PageXref` is used for 'extra' pages, where an item is submitted as camera copy, and only running heads (at most) are run.

```

971 <!!latex>
972 \def\pagexrefON#1{%
973     \write-1{\def\expandafter\noexpand\csname#1\endcsname{\number\pageno}}%
974     \write\ppoutfile{%
975         \def\expandafter\noexpand\csname#1\endcsname{\number\pageno}}%
976     }
977 \def\PageXrefON#1{%
978     \immediate\write-1{\def\expandafter
979         \noexpand\csname#1\endcsname{\number\pageno}}%
980     \immediate\write\ppoutfile{\def\expandafter
981         \noexpand\csname#1\endcsname{\number\pageno}}}
982 </!!latex>
983 <!!latex>
984 \def\pagexrefON#1{%
985     \write-1{\def\expandafter\noexpand\csname#1\endcsname{\number\c@page}}%
986     \write\ppoutfile{%
987         \def\expandafter\noexpand\csname#1\endcsname{\number\c@page}}%
988     }
989 \def\PageXrefON#1{%
990     \immediate\write-1{\def\expandafter
991         \noexpand\csname#1\endcsname{\number\c@page}}%
992     \immediate\write\ppoutfile{\def\expandafter
993         \noexpand\csname#1\endcsname{\number\c@page}}}
994 </!latex>
995 \def\pagexrefOFF#1{}
996 \let\pagexref=\pagexrefOFF
997 \def\PageXrefOFF#1{}
998 \let\PageXref=\PageXrefOFF
999 \def\xreftoON#1{%
1000 \ifundefined{#1}%
1001     ???\TBremark{Need cross reference for #1.}%
1002 \else\csname#1\endcsname\fi}
1003 \def\xreftoOFF#1{???}
1004 \let\xrefto=\xreftoOFF

```

`\TBdriver` ‘marks code for use when articles are run together in a driver file’. Since we don’t yet have a definition of that arrangement, we don’t have a definition of `\TBdriver`. Its argument (which one presumes was intended as the code for this unusual state) is just gobbled.

```
1005 \let\TBdriver\gobble
```

Hyphenation exceptions. We read our own full `ushyphex.tex` (generated from `tb0hyf.tex`) if it’s available. The additional exceptions are nearly all included in the file, but keep defining them anyway, since we have for many years.

But do not define any exceptions if `\tubomithyphenations` is defined. This is needed for the `hyf` articles themselves.

```
1006 \ifx\tubomithyphenations\@thisisundefined
1007 \InputIfFileExists{ushyphex.tex}{\}{} % ok if it's missing
1008 \hyphenation{Del-a-ware Dijk-stra Duane Eijk-hout
1009 Flor-i-da Free-BSD Ghost-script
1010 Hara-lam-bous Jac-kow-ski Ja-pa-nese Karls-ruhe Lua-Meta
1011 Mac-OS Math-Sci-Net
1012 Net-BSD Open-BSD Open-Office
1013 Pak-i-stan Post-Script Rich-ard Skoup South-all
1014 Vieth VM-ware Win-Edt
1015 acro-nym acro-nyms analy-sis ap-pen-di-ces ap-pen-dix asyn-chro-nous
1016 bib-lio-graph-i-cal bit-map bit-mapped bit-maps buf-fer buf-fers bool-ean
1017 col-umns com-put-able com-put-abil-ity
1018 data-base data-bases
1019 de-allo-cate de-allo-cates de-allo-cated de-allo-ca-tion
1020 de-riv-a-tive de-riv-a-tives de-riv-a-ble der-i-va-tion dis-trib-ut-able
1021 es-sence
1022 fall-ing
1023 half-way
1024 in-fra-struc-ture
1025 key-note
1026 long-est
1027 ma-gyar man-u-script man-u-scripts meta-table meta-tables
1028 mne-mon-ic mne-mon-ics mono-space mono-spaced
1029 name-space name-spaces
1030 off-line over-view
1031 pal-ettes par-a-digm par-a-dig-matic par-a-digms
1032 pipe-line pipe-lines
1033 plug-in plug-ins pres-ent-ly pro-gram-mable
1034 re-allo-cate re-allo-cates re-allo-cated re-printed
1035 set-ups se-vere-ly spell-ing spell-ings stand-alone strong-est
1036 sub-ex-pres-sion sub-tables sur-gery syn-chro-ni-city syn-chro-nous
1037 text-height text-length text-width
1038 time-stamp time-stamped time-stamps
1039 vis-ual vis-ual-ly
1040 which-ever white-space white-spaces wide-spread wrap-around
1041 }
1042 \fi
1043 <!!latex>\restorecat\@
1044 </common>
1045 <*clashtail>
1046 \PrelimDrafttrue
```

3.11 Page dimensions, glue, penalties, etc.

```

1047 \textheight 54pc      % 648pt = 645.58bp = 8.97in
1048 \textwidth 39pc       % 468pt = 466.25bp = 6.48in
1049 \columnsep 1.5pc      % 18pt = 17.93bp = .249in
1050 \columnwidth 18.75pc % 225pt = 224.16bp = 3.11in
1051 \hfuzz 1pt
1052 \parindent \normalparindent % 20pt
1053 \parskip \z@ % \@plus\p@
1054 \leftmargini 2em
1055 \leftmarginv .5em
1056 \leftmarginvi .5em
1057 \oddsidemargin \z@
1058 \evensidemargin \z@
1059 \topmargin -2.5pc      % 30pt = 29.89bp = .415in
1060 \headheight 12\p@
1061 \headsep 20\p@
1062 \marginparwidth 48\p@
1063 \marginparsep 10\p@
1064 \partopsep=\z@
1065 \topsep=3\p@\@plus\p@\@minus\p@
1066 \parsep=3\p@\@plus\p@\@minus\p@
1067 \itemsep=\parsep
1068 %
1069 % The width of one column plus gutter (=243pt =242.09bp) is useful sometimes.
1070 \newdimen\tubcolwidthandgutter
1071 \tubcolwidthandgutter=\columnwidth
1072 \advance\tubcolwidthandgutter by \columnsep
1073 %
1074 % Ordinarily we typeset in two columns, but the onecolumn option
1075 % goes to one. In which case we want to center the text block on an
1076 % 8.5in width, given the default 72.27pt offset with margins of zero.
1077 % We are always in LaTeX's twoside mode because of how we load article,
1078 % and this is a good thing, since we want different headings.
1079 \if@tubtwocolumn \twocolumn \else
1080   \onecolumn
1081   \textwidth=34pc
1082   \oddsidemargin=30.8775pt
1083   \evensidemargin=\oddsidemargin
1084 \fi
1085 %
1086 \newdimen\pagewd        \pagewd=\textwidth
1087 \newdimen\trimwd        \trimwd=\pagewd
1088 \newdimen\trimlgt        \trimlgt=11in
1089 \newdimen\headmargin    \headmargin=3.5pc

```

Don't go to a float page so soon. Not all of these are relevant to all articles, but we may as well set them all.

```

1090 \renewcommand{\topfraction}{.9} % don't go to a float page so soon
1091 \renewcommand{\dbltopfraction}{.9}
1092 \renewcommand{\bottomfraction}{.7}
1093 \renewcommand{\textfraction}{.1}
1094 \renewcommand{\floatpagefraction}{.8}
1095 \renewcommand{\dblfloatpagefraction}{.8} % the most common one used

```

3.12 Messing about with the L^AT_EX logo

Barbara Beeton's pleas for L^AT_EX logos that look right in any font shape provoked me to generate the following stuff that is configurable.

Here's the command for the user to define a new version. The arguments are font family, series and shape, and then the two kern values used in placing the raised 'A' of L^AT_EX.

```
1096 \newcommand{\DeclareLaTeXLogo}[5]{\expandafter\def
1097   \csname @LaTeX@#1/#2/#3\endcsname{#{4}{#5}}}
```

The default values are as used in the source of L^AT_EX itself:

```
1098 \def\@LaTeX@default{.36}{.15}
```

More are defined in the initial version, for bold CM sans (which is used as `\SecTitleFont`), and CM italic medium and bold, and Bitstream Charter (which Nelson Beebe likes to use). Duplicate for Latin Modern.

```
1099 \DeclareLaTeXLogo{cmss}{bx}{n}{.3}{.15}
1100 \DeclareLaTeXLogo{lmss}{bx}{n}{.3}{.15}
1101 %
1102 \DeclareLaTeXLogo{cmr}{m}{it}{.29}{.2}
1103 \DeclareLaTeXLogo{lmr}{m}{it}{.29}{.2}
1104 %
1105 \DeclareLaTeXLogo{cmr}{m}{sl}{.29}{.15}
1106 \DeclareLaTeXLogo{lmr}{m}{sl}{.29}{.15}
1107 %
1108 \DeclareLaTeXLogo{cmr}{bx}{it}{.29}{.2}
1109 \DeclareLaTeXLogo{lmr}{bx}{it}{.29}{.2}
1110 %
1111 \DeclareLaTeXLogo{cmr}{bx}{sl}{.29}{.2}
1112 \DeclareLaTeXLogo{lmr}{bx}{sl}{.29}{.2}
1113 %
1114 \DeclareLaTeXLogo{bch}{m}{n}{.2}{.08}
1115 \DeclareLaTeXLogo{bch}{m}{it}{.2}{.08}
```

Redefine `\LaTeX` to choose the parameters for the current font, or to use the default value otherwise:

```
1116 \DeclareRobustCommand{\LaTeX}{\expandafter\let\expandafter\reserved@a
1117   \csname @LaTeX@f@family/\f@series/\f@shape\endcsname
1118   \ifx\reserved@a\relax\let\reserved@a\@LaTeX@default\fi
1119   \expandafter\@LaTeX\reserved@a}
```

Here's the body of what was originally `\LaTeX`, pulled out with its roots dripping onto the smoking ruin of original L^AT_EX, and then bits stuck in on the side.

`\@LaTeX@default` provides parameters as one finds in the original; other versions are added as needed.

```
1120 \newcommand{\@LaTeX}[2]{%
1121   %\wlog{latex logo family=\f@family/\f@series/\f@shape -> #1, #2.}%
1122   L\kern-#1em
1123   {\sbox\z@ T%
1124     \vbox to\ht0{\hbox{$\m@th$%
1125       \csname S@\f@size\endcsname
1126       \fontsize\sf@size\z@
```

```

1127             \math@fontsfalse\selectfont
1128             A}%
1129         \vss}%
1130     }%
1131     \kern-#2em%
1132     \TeX}

```

3.13 Authors, contributors, addresses, signatures

An article may have several authors (of course), so we permit an `\author` command for each of them. The names are then stored in a set of `\csnames` called `\author1`, `\author2`, ... Similarly, there are several `\address<n>` and `\netaddress<n>` and `\PersonalURL<n>` and `\ORCID<n>` commands set up for each article.

Comment: [RF] I would like to make provision for several authors at the same address, but (short of preempting the `*` marker, which it would be nice to retain so as to preserve compatibility with the `plain` style) I'm not sure how one would signal it. [KB] Current kludges for multiple author affiliations in `tb143rishi-xml-first`, `tb140rishi-elsarticle`, `tb128ruckert-hint`, etc.

```

1133 \def\theauthor#1{\csname theauthor#1\endcsname}
1134 \def\theaddress#1{\csname theaddress#1\endcsname}
1135 \def\thenetaddress#1{\csname thenetaddress#1\endcsname}
1136 \def\thePersonalURL#1{\csname thePersonalURL#1\endcsname}
1137 \def\theORCID#1{\csname theORCID#1\endcsname}

```

The standard way of listing authors is to iterate from 1 to `\count@` and to pick the author names as we go.

```

1138 <!\latex>\newcount\@tempcnta
1139 \def\@defaultauthorlist{%
1140     \@getauthorlist\@firstofone
1141 }

```

`\@getauthorlist` processes the author list, passing every bit of stuff that needs to be typeset to the macro specified as its argument.

```

1142 \def\@getauthorlist#1{%
1143     \count@\authornumber
1144     \advance\count@ by -2
1145     \@tempcnta0

```

Loop to output the first $n - 2$ of the n authors (the loop does nothing if there are two or fewer authors)

```

1146     \loop
1147         \ifnum\count@>0
1148             \advance\@tempcnta by \@ne
1149             #1{\ignorespaces\theauthor{\number\@tempcnta}\unskip, }%
1150             \advance\count@ by \m@ne
1151     \repeat
1152     \count@\authornumber
1153     \advance\count@ by -\@tempcnta
1154     \ifnum\authornumber>0

```

If there are two or more authors, we output the penultimate author's name here, followed by 'and'

```

1155 \ifnum\count@>1
1156 \count@\authornumber
1157 \advance\count@ by \m@ne
1158 #1{\ignorespaces\theauthor{\number\count@}\unskip\@tubauthorlastsep}%
1159 \fi

```

Finally (if there were any authors at all) output the last author's name:

```

1160 #1{\ignorespaces\theauthor{\number\authornumber}\unskip}
1161 \fi
1162 }
1163 %
1164 \def\@tubauthorlastsep{, }% until 2018, was: "\ and "

```

Signature blocks. The author can (in principle) define a different sort of signature block using `\signature`, though this could well cause the editorial group to have collective kittens (unless it had been discussed in advance...)

```

1165 \def\signature#1{\def\@signature{#1}}
1166 \def\@signature{\@defaultsignature}

```

`\@defaultsignature` loops through all the authors, outputting the details we have about that author, or (if we're in a sub-article) outputs the contributor's name and closes the group opened by `\contributor`. It is (as its name implies) the default body for `\makesignature`

```

1167 \def\@defaultsignature{%
1168 \let\thanks\@gobble
1169 \frenchspacing
1170 %
1171 \ifnum\authornumber<0

```

if `\authornumber < 0`, we are in a contributor's section

```

1172 \medskip
1173 \signaturemark
1174 \theauthor{\number\authornumber}\\
1175 \theaddress{\number\authornumber}\\
1176 \allowhyphens
1177 \thenetaddress{\number\authornumber}\\
1178 \thePersonalURL{\number\authornumber}\\
1179 \theORCID{\number\authornumber}\\

```

`\authornumber ≥ 0`, so we are in the body of an ordinary article:

```

1180 \else
1181 \count@=0
1182 \loop
1183 \ifnum\count@<\authornumber
1184 \medskip
1185 \advance\count@ by \@ne
1186 \signaturemark
1187 \theauthor{\number\count@}\\
1188 \theaddress{\number\count@}\\
1189 {%
1190 \allowhyphens
1191 \thenetaddress{\number\count@}\\
1192 \thePersonalURL{\number\count@}\\
1193 \theORCID{\number\count@}\\

```

```

1194         }%
1195     \repeat
1196 \fi
1197 }%
1198 }
1199 \newdimen\signaturewidth \signaturewidth=12pc

```

The optional argument to `\makesignature` is useful in some circumstances (e.g., multi-contributor articles)

```

1200 \newcommand{\makesignature}[1][\medskipamount]{%
    check the value the user has put in \signaturewidth: it may be at most
    1.5pc short of \columnwidth

```

```

1201 \@tempdima\signaturewidth
1202 \advance\@tempdima 1.5pc
1203 \ifdim \@tempdima>\columnwidth
1204     \signaturewidth \columnwidth
1205     \advance\signaturewidth -1.5pc
1206 \fi
1207 \par
1208 \penalty9000
1209 \vspace{#1}%
1210 \rightline{%
1211     \vbox{\hsize\signaturewidth \ninepoint \raggedright
1212         \parindent \z@ \everypar={\hangindent 1pc }%
1213         \parskip \z@skip
1214         \def\|{\unskip\hfil\break}%
1215         \def\\{\endgraf}%
1216         \def\phone{\rm Phone: }%
1217         \def\tubmultipleaffilauthor{\unskip,\\ \hspace*{1em}}%
1218         \rm\@signature}%
1219 }%
1220 \ifnum\authornumber<0 \endgroup\fi
1221 }
1222 \def\signaturemark{\leavevmode\llap{$\diamond$\enspace}}

```

The idea here is that if multiple authors share affiliation information, we need only typeset the affiliation once. We separate by commas for the `\maketitle`, and put on separate lines, also with commas, in the `\makesignature`.

Similarly, within `\netaddress`, `!tubmultipleaffilnet` separates with a space before and after the comma. (All this per bb.) See `tb122childs-trotter.ltx`, `tb131sojka-czech.ltx` for examples.

```

1223 \def\tubmultipleaffilauthor{\unskip, \ignorespaces}%
1224 \def\tubmultipleaffilnet{\unskip\textrm{\,,\ \ignorespaces}}

```

Now all the awful machinery of author definitions. `\authornumber` records the number of authors we have recorded to date.

```

1225 \newcount\authornumber
1226 \authornumber=0

```

`\author` ‘allocates’ another author name (by bumping `\authornumber`) and also sets up the address and netaddress for this author to produce a warning and to prevent oddities if they’re invoked. This last assumes that invocation will be in the context of `\signature` (`ltugboat.cls`) or `\maketitle` (`ltugproc.cls`);

in both cases, invocation is followed by a line break (tabular line break `\\` in `ltugproc`, `\endgraf` in `\makesignature` in `ltugboat`).

```
1227 \def\author{%
1228   \global\advance\authornumber\@ne
1229   \TB@author
1230 }
```

`\contributor` is for a small part of a multiple-part article; it begins a group that will be ended in `\makesignature`.

```
1231 \def\contributor{%
1232   \begingroup
1233   \authornumber\m@ne
1234   \TB@author
1235 }
```

Both ‘types’ of author fall through here to set up the author name and to initialise author-related things. `\EDITORno*` commands allow the editor to record that there’s good reason for an *address* or *netaddress* not to be there, but nowadays, we consider all address information optional.

```
1236 \def\TB@author#1{%
1237   \expandafter\def\csname theauthor\number\authornumber\endcsname
1238     {\ignorespaces#1\unskip}%
1239 %   \expandafter\def\csname theaddress\number\authornumber\endcsname
1240 %     {\TBWarningNL{Address for #1\space missing}\@gobble}%
1241 %   \expandafter\def\csname thenetaddress\number\authornumber\endcsname
1242 %     {\TBWarningNL{Net address for #1\space missing}\@gobble}%
1243   \expandafter\let\csname thePersonalURL\number\authornumber\endcsname
1244     \@gobble
1245   \expandafter\let\csname theORCID\number\authornumber\endcsname
1246     \@gobble
1247 }
1248 \def\EDITORnoaddress{%
1249   \expandafter\let\csname theaddress\number\authornumber\endcsname
1250     \@gobble
1251 }
1252 \def\EDITORnonetaddress{%
1253   \expandafter\let\csname thenetaddress\number\authornumber\endcsname
1254     \@gobble
1255 }
```

`\address` copies its argument into the `\theaddress<n>` for this author.

```
1256 \def\address#1{%
1257   \expandafter\def\csname theaddress\number\authornumber\endcsname
1258     {\leavevmode\ignorespaces#1\unskip}}
```

`\network` is for use within the optional argument of `\netaddress`; it defines the *name* of the network the user is on.

Comment: I think this is a fantasy, since everyone (in practice, nowadays) quotes an internet address. In principle, there are people who will quote X.400 addresses (but they’re few and far between) and I have (during 1995!) seen an address with an UUCP bang-path component on `comp.text.tex`, but *really*!

```
1259 \def\network#1{\def\@network{#1: }}}
```

`\netaddress` begins a group, executes an optional argument (which should not, presumably, contain global commands) and then relays to `\@relay@netaddress` with both `@` and `%` made active (so that they can be discretionary points in the address). If we're using L^AT_EX 2_ε, we use the default-argument form of `\newcommand`; otherwise we write it out in all its horribleness.

```
1260 \newcommand{\netaddress}[1][\relax]{%
1261   \begingroup
1262   \def\@network{}}%
```

Unfortunately, because of the catcode hackery, we have still to do one stage of relaying within our own code, even if we're using L^AT_EX 2_ε.

```
1263   #1\@sanitize\makespace\ \makeactive\@%
1264   \makeescape! \makebgroup[ \makeegroup]% seems more useful than literals
1265   \makeactive\.\makeactive\%\@relay@netaddress}%
```

`\@relay@netaddress` finishes the job. It sets `\thenetaddress` for this author to contain the network name followed by the address. As a result of our kerfuffle above, `@` and `%` are active at the point we're entered. We ensure they're active when `\thenetaddress` gets expanded, too. (*WOT?!*)

```
1266 \def\@relay@netaddress#1{%
1267   \ProtectNetChars
1268   \expandafter\protected@xdef
1269     \csname thenetaddress\number\authornumber\endcsname
1270     {\protect\leavevmode\textrm{\@network}}%
1271     {\protect\NetAddrChars\net
1272       \ignorespaces#1\unskip}}%
1273 \endgroup
1274 }
```

We `\personalURL` quite differently from `\netaddress`: it is set up to simply call `\tburl`, which makes a live link if possible, and also removes a leading protocol. Thus the argument has to be a true url, not just a random string, but that restriction seems ok to get the benefits. Since `\tburl` handles all the catcoding, no need to do any of that here.

```
1275 \def\personalURL#1{%
1276   % define \cs{thePersonalURL}\meta{n} for author \meta{n}'s \personalURL.
1277   \expandafter\protected@xdef
1278     \csname thePersonalURL\number\authornumber\endcsname{%
1279     \protect\leavevmode
1280       \ignorespaces
1281       \protect\tburl{#1}%
1282       \unskip
1283   }%
1284 }
```

Previously: `\personalURL` was similar to `\netaddress`, apart from (1) the lack of the eccentric optional argument, (2) the activation of `'/'`. This is the old definition, no longer used (left here just for posterity); new definition is just above.

```
1285 %\def\personalURL{\begingroup
1286 %  \@sanitize\makespace\ \makeactive\@%
1287 %  \makeactive\.\makeactive\%\makeactive\/%
1288 %  \@personalURL}%
1289 %\def\@personalURL#1{%
```

```

1290 % \ProtectNetChars
1291 % % define \cs{thePersonalURL}\meta{n} for author \meta{n}.
1292 % \expandafter\protected@xdef
1293 % \csname thePersonalURL\number\authornumber\endcsname{%
1294 % \protect\leavevmode
1295 % {%
1296 % \protect\URLchars
1297 % \net
1298 % \ignorespaces\protect\tburl{#1}\unskip
1299 % }%
1300 % }%
1301 % \endgroup
1302 %}

```

Define the activation mechanism for ‘@’, ‘%’, ‘.’ and ‘/’, for use in the above. Note that, since the code has ‘%’ active, we have ‘*’ as a comment character, which has a tendency to make things look peculiar...

```

1303 {%
1304 \makecomment\*
1305 \makeactive\@
1306 \gdef\netaddrat{\makeactive\@*
1307 \def\{\discretionary{\char"40}{\char"40}}
1308 \makeactive\%
1309 \gdef\netaddrpercent{\makeactive\%*
1310 \def\{\discretionary{\char"25}{\char"25}}
1311 \makeactive\.
1312 \gdef\netaddrdot{\makeactive\.*
1313 \def\{\discretionary{\char"2E}{\char"2E}}

```

`\NetAddrChars` is what *we* use (we’re constrained to retain the old interface to this stuff, but it *is* clunky...). Since URLs are a new idea, we are at liberty not to define a separate `\netaddrslash` command, and we only have `\URLchars`.

```

1314 \gdef\NetAddrChars{\netaddrat \netaddrpercent \netaddrdot}
1315 \makeactive\/
1316 \gdef\URLchars{*
1317 \NetAddrChars
1318 \makeactive\/*
1319 \def/{\discretionary{\char"2F}{\char"2F}}

```

`\ProtectNetChars` includes protecting ‘/’, since this does no harm in the case of net addresses (where it’s not going to be active) and we thereby gain by not having yet another `csname`.

```

1320 \gdef\ProtectNetChars{*
1321 \def\{\protect@}*
1322 \def\{\protect%}*
1323 \def\{\protect.}*
1324 \def/{\protect/}*
1325 }
1326 }

```

L^AT_EX 2_ε (in its wisdom) suppresses `\DeclareOldFontCommand` when in compatibility mode, so that in that circumstance we need to use a declaration copied from `latex209.def` rather than the way we would normally do the thing (using the command L^AT_EX 2_ε defines for the job).

```

1327 \if@compatibility
1328   \DeclareRobustCommand{\net}{\normalfont\ttfamily\mathgroup\sympewriter}
1329 \else
1330   \DeclareOldFontCommand{\net}{\ttfamily\upshape\mdseries}{\mathtt}
1331 \fi
1332 \def\authorlist#1{\def\@author{#1}}
1333 \def\@author{\@defaultauthorlist}

```

\ORCID inserts ‘ORCID’ and then argument into the \theORCID<n> for this author. Also, we want \small for this.

```

1334 \def\ORCID#1{%
1335   \expandafter\def\csname theORCID\number\authornumber\endcsname
1336     {\leavevmode \ignorespaces {\SMC ORCID} #1\unskip}}

```

For the online re-publication (as of 2009) by Mathematical Sciences Publishers <http://mathscipub.org>, lots and lots of metadata is needed, much of it redundant with things we already do. They are flexible enough to allow us to specify it in any reasonable way, so let’s make one command \mspmetavar which takes two arguments. Example: \mspmetavar{volumenumber}{30}. For our purposes, it is just a no-op. And this initiative never came to anything, so it is not used at all.

\mspmetavar

```

1337 \def\mspmetavar#1#2{}

```

3.14 Article title

\if@articletitle \maketitle takes an optional “*”; if present, the operation is not defining the \maketitle title of a paper, merely that of a “business” section (such as the participants at \@r@maketitle a meeting) that has no credited author or other title. In this case, the command flushes out the latest \sectitle (or whatever) but does nothing else.

Provide machinery (\PreTitleDrop to skip extra space, even one or more full columns, above the top of an article to leave space to paste up a previous article that has finished on the same page. This is a fall back to accommodate the fact that multiple articles cannot be run together easily.

In addition, if the secondcolstart option was specified, do \null\newpage to move over. This is separate from \PreTitleDrop, for no particular reason.

```

1338 \newif\if@articletitle
1339 \def\maketitle{\@ifstar
1340   {\@articletitlefalse\@r@maketitle}%
1341   {\@articletitletrue\@r@maketitle}%
1342 }
1343 \def\@r@maketitle{\par
1344   \iftubsecondcolstart \null\newpage\tubsecondcolstartextra \fi
1345   \ifdim\PreTitleDrop > \z@
1346     \loop
1347       \ifdim \PreTitleDrop > \textheight
1348         \vbox{}\vfil\eject
1349         \advance\PreTitleDrop by -\textheight
1350       \repeat
1351     \vbox to \PreTitleDrop{\vfil}%
1352     \global\PreTitleDrop=\z@
1353   \fi
1354   \begingroup

```

```

1355 \setcounter{footnote}{0}
1356 \global\@topnum\z@ % disallow floats above the title
1357 \def\thefootnote{\fnsymbol{footnote}}
1358 \@maketitle
1359 \@thanks
1360 \endgroup
1361 \setcounter{footnote}{0}
1362 \gdef\@thanks{}
1363 }

```

`\title` We redefine the `\title` command, so as to set the `\rhTitle` command at the same time. While we're at it, we redefine it to have optional arguments for use as 'short' versions, thus obviating the need for users to use the `\shortTitle` command.

```

1364 \def\rhTitle{}% avoid error if no author or title
1365 \renewcommand{\title}{\@dblarg\TB@title}
1366 \def\TB@title[#1]#2{\gdef\@title{#2}%
1367   \bgroup
1368     \let\thanks\@gobble
1369     \def\{\{\unskip\space\ignorespaces}%
1370     \protected@xdef\rhTitle{#1}%
1371   \egroup
1372 }

```

`\shortTitle` The `\rh*` commands are versions to be used in the running head of the article.
`\ifshortAuthor` Normally, they are the same things as the author and title of the article, but in the
`\shortAuthor` case that there are confusions therein, the text should provide substitutes, using the `\short*` commands.

```

1373 \def\shortTitle #1{\def\rhTitle{#1}}
1374 \newif\ifshortAuthor
1375 \def\shortAuthor #1{\def\rhAuthor{#1}\shortAuthortrue}

```

3.15 Section titles

The following macros are used to set the large *TUGboat* section heads (e.g. "General Delivery", "Fonts", etc.)

Define the distance between articles which are run together:

```

1376 \def\secsep{\vskip 5\baselineskip}

```

Note that `\stbaselineskip` is used in the definition of `\sectitlefont`, in L^AT_EX 2_ε, so that it has (at least) to be defined before `\sectitlefont` is used (we do the whole job).

```

1377 \newdimen\stbaselineskip \stbaselineskip=18\p@
1378 \newdimen\stfontheight
1379 \settoheight{\stfontheight}{\sectitlefont 0}

```

Declaring section titles; the conditional `\ifSecTitle` records the occurrence of a `\sectitle` command. If (when) a subsequent `\maketitle` occurs, the section title box will get flushed out; as a result of this, one could in principle have a set of `\sectitle` commands in a semi-fixed steering file, and inclusions of files inserted only as and when papers have appeared. Only the last `\sectitle` will actually be executed.

```

1380 \newif\ifWideSecTitle

```

```

1381 \newif\iftubtitlerulefullwidth
1382 \newif\ifSecTitle \SecTitlefalse
1383 \newcommand{\sectitle}{%
1384   \SecTitletrue
1385   \@ifstar
1386     {\WideSecTitletrue\def\s@ctitle}%
1387     {\WideSecTitlefalse\def\s@ctitle}%
1388 }

```

`\PreTitleDrop` records the amount of column-space we need to eject before we start any given paper. It gets zeroed after that ejection has happened.

```

1389 \newdimen\PreTitleDrop \PreTitleDrop=\z@

```

The other parameters used in `\@sectitle`; I don't think there's the slightest requirement for them to be registers (since they're constant values, AFAIK), but converting them to macros would remove the essentially useless functionality of being able to change them using assignment, which I'm not about to struggle with just now...

`\AboveTitleSkip` is glue above the article title; `\BelowTitleSkip` is glue below the authors in the title block. `\strulethickness` is the value to use for `\fboxrule` when setting the title, and for the rule above titles when there is no box.

For `\BelowTitleSkip`, add some stretch and shrink since the first column of an article often needs it; otherwise, a first column of all text will come out underfull. Use `plus2pt` since that is the same as the glue above sections, but `minus1pt` since we'd usually prefer to shrink somewhere else if possible.

```

1390 \newskip\AboveTitleSkip \AboveTitleSkip=12pt
1391 \newskip\BelowTitleSkip \BelowTitleSkip=8pt plus2pt minus1pt
1392 \newdimen\strulethickness \strulethickness=.6pt

```

`\@sectitle` actually generates the section title (in a rather generous box). It gets called from `\maketitle` under conditional `\ifSecTitle`; by the time `\@sectitle` takes control, we already have `\SecTitlefalse`. This implementation uses L^AT_EX's `\framebox` command, on the grounds that one doesn't keep a dog and bark for oneself...

```

1393 \def\@sectitle #1{%
1394   \par
1395   \penalty-1000

```

If we're setting a wide title, the stuff will be at the top of a page (let alone a column) but inside a box, so that the separator won't be discardable: so don't create the separator in this case.

```

1396   \ifWideSecTitle\else\secsep\fi
1397   {%
1398     \fboxrule\strulethickness
1399     \fboxsep\z@
1400     \noindent\framebox[\hsize]{%
1401       \vbox{%
1402         \raggedcenter
1403         \let\\\@sectitle@newline
1404         \sectitlefont
1405         \makestrut[2\stfontheight;\z@]%
1406         #1%

```

```

1407         \makestrut[\z@;\stfontheight]\endgraf
1408     }%
1409 }%
1410 }%
1411 \nobreak
1412 \vskip\baselineskip
1413 }

```

`\@sectitle@newline` For use inside `\sectitle` as `\\`. Works similarly to `\\` in the “real world”—uses an optional argument

```

1414 \newcommand{\@sectitle@newline}[1][\z@]{%
1415   \ifdim#1>\z@
1416     \makestrut[\z@;#1]%
1417   \fi
1418   \unskip\break
1419 }

```

We need to trigger the making of a section title in some cases where we don’t have a section title proper (for example, in material taken over from TTN).

```

1420 \def\@makesectitle{\ifSecTitle
1421   \global\SecTitlefalse
1422   \ifWideSecTitle
1423     \twocolumn[\@sectitle{\s@ctitle}]%
1424     \global\WideSecTitlefalse
1425   \else
1426     \@sectitle{\s@ctitle}%
1427   \fi
1428 \else
1429   \vskip\AboveTitleSkip
1430   \kern\topskip
1431   \hrule \@height\z@ \@depth\z@ \@width 10\p@
1432   \kern-\topskip
1433   \kern-\strulethickness
1434   \iftubtitlerulefullwidth
1435     \hrule \@height\strulethickness \@depth\z@ width\textwidth
1436   \else
1437     \hrule \@height\strulethickness \@depth\z@
1438   \fi
1439   \kern\medskipamount
1440   \nobreak
1441 \fi
1442 }

```

`\@maketitle` Finally, the body of `\maketitle` itself.

```

1443 \def\@maketitle{%
1444   \@makesectitle
1445   \if@articletitle{%
1446     \nohyphens \interlinepenalty\@M
1447     \setbox0=\hbox{%
1448       \let\thanks\@gobble
1449       \let\=\quad
1450       \let\and=\quad
1451       \ignorespaces\@author}%
1452     {%

```

```

1453      \noindent\bf\raggedright\ignorespaces\frenchspacing
1454      \let\BibTeX=\bfBibTeX % else LaTeX Font Warning:
1455                        %      Font shape 'OT1/cmr/bx/sc' undefined
1456      \@title\endgraf
1457  }%
1458  \ifdim \wd0 < 5\p@                % omit if author is null
1459  \else

```

Since we have $\text{\BelowTitleSkip} + 4\text{pt} = \text{\baselineskip}$, we skip by 4pt here. However, an all-text first column still comes out underfull, maybe because of the top rule? Thus $\text{\BelowTitleSkip}$ is given a little stretch and shrink.

```

1460      \nobreak \vskip 4\p@
1461      {%
1462          \leftskip=\normalparindent
1463          \raggedright
1464          \def\and{\unskip\}%
1465          \noindent\@author\endgraf
1466      }%
1467  \fi
1468  \nobreak
1469  \vskip\BelowTitleSkip
1470 }\fi%
1471 \global\@afterindentfalse
1472 \aftergroup\@afterheading
1473 }

```

Dedications are ragged right, in italics.

```

1474 \newenvironment{dedication}%
1475   {\raggedright\noindent\itshape\ignorespaces}%
1476   {\endgraf\medskip}

```

The `abstract` and `longabstract` environments both use $\text{\section*}$. For one-column articles (or in `ltugproc` class), indent the abstract. This is done in the usual bizarre L^AT_EX way, by treating it as a one-item list with an empty item marker.

```

1477 \def\@tubonecolumnabstractstart{%
1478     \list{}\{\listparindent\normalparindent
1479         \itemindent\z@ \leftmargin\@tubfullpageindent
1480         \rightmargin\leftmargin \parsep \z@\}\item[]\ignorespaces
1481 }
1482 \def\@tubonecolumnabstractfinish{%
1483     \endlist
1484 }
1485 \renewenvironment{abstract}%
1486   {\begin{SafeSection}%
1487     \section*{%
1488         \if@tubtwocolumn\else \hspace*\{\@tubfullpageindent\}\fi
1489         Abstract}%
1490     \if@tubtwocolumn\else \@tubonecolumnabstractstart \fi
1491   }%
1492   {\if@tubtwocolumn\else \@tubonecolumnabstractfinish \fi
1493   \end{SafeSection}}
1494 \newenvironment{longabstract}%
1495   {\begin{SafeSection}%

```

```

1496     \section*{Abstract}%
1497     \bgroup\small
1498   }%
1499   {\endgraf\egroup
1500     \end{SafeSection}%
1501   \vspace{.25\baselineskip}
1502   \begin{center}
1503     {\$--*--\$}
1504   \end{center}
1505   \vspace{.5\baselineskip}}

```

3.16 Section headings

Redefine style of section headings to match plain *TUGboat*. Negative before skip suppresses following parindent. (So negate the stretch and shrink too).

These macros are called `\*head` in the plain styles.

Relaying via `\TB@startsection` detects inappropriate use of `\section*`. Of course, if (when) *we* use it, we need to avoid that relaying; this can be done by `\letting \TB@startsection to \TB@safe@startsection`, within a group.

First the version for use in the default case, when class option `NUMBERSEC` is in effect.

The `\tubsecfmt` macro defines our standard formatting for section titles: ragged right, french spacing, no hyphenation. The `\tubruninsecfmt` macro is the simpler form for run-in section headings (when the afterskip is negative), with the afterskip glue given by `\tubruninglue`. The `\tubsechook` macro allows overriding the defaults.

```

1506 \def\tubsechook{}
1507 \def\tubsecfmt{\normalsize\bf\raggedright\frenchspacing\nohyphens\tubsechook}
1508 \def\tubruninglue{-1em plus-2\fontdimen3\font minus-\fontdimen4\font}
1509 \def\tubruninsecfmt{\normalsize\bf\tubsechook}
1510 %
1511 % We make before skip+afterskip = 1 normal linespace if there's no
1512 % stretching or shrinking of the glue, to help retain a small chance of
1513 % keeping baselines aligned.
1514 %
1515 \if@numbersec
1516   \def\section{\TB@startsection{%
1517     {section}                % name of counter
1518     {1}                      % level
1519     {0pt}                    % indent
1520     {-8pt plus-2pt minus-2pt} % before skip; negative -> \noindent after
1521     {4pt}                    % afterskip; negative -> hspace for run-in
1522     {\tubsecfmt}}}           % style
1523 %
1524 \def\subsection{\TB@startsection{%
1525   {subsection}%
1526   2%
1527   \z@
1528   {-8\p@ \@plus-2\p@ \@minus-2\p@}%
1529   {4\p@}%
1530   {\tubsecfmt}}}
1531 %

```

```

1532 \def\subsubsection{\TB@startsection{%
1533   {subsubsection}%
1534   3%
1535   \z@
1536   {-8\p@ \@plus-2\p@ \@minus-2\p@}%
1537   {4\p@}%
1538   {\tubsecfmt}}}}
1539 %
1540 \def\paragraph{\TB@startsection{%
1541   {paragraph}%
1542   4%
1543   \z@
1544   {4\p@ \@plus1\p@ \@minus1\p@}%
1545   {\tubruninglue}
1546   {\tubruninsecfmt}}}}

```

Now the version if class option `nonumber` is in effect, i.e., if `\if@numbersec` is false.

```

1547 \else
1548   \setcounter{secnumdepth}{0}
1549   \def\section{\TB@nolimlabel\TB@startsection{%
1550     {section}% same as numbered
1551     1%
1552     \z@
1553     {-8\p@ \@plus-2\p@ \@minus-2\p@}%
1554     {4\p@}%
1555     {\tubsecfmt}}}}
1556 %
1557   \def\subsection{\TB@nolimlabel\TB@startsection{%
1558     {subsection}%
1559     2%
1560     \z@
1561     {-8\p@ \@plus-2\p@ \@minus-2\p@}%
1562     {\tubruninglue}
1563     {\tubruninsecfmt}}}}
1564 %
1565   \def\subsubsection{\TB@nolimlabel\TB@startsection{%
1566     {subsubsection}%
1567     3%
1568     \parindent
1569     {-8\p@ \@plus-2\p@ \@minus-2\p@}%
1570     {\tubruninglue}
1571     {\tubruninsecfmt}}}}
1572 \fi

```

`\TB@startsection` used to warn about * versions of sectioning commands when numbering wasn't in effect. But that eventually seemed a useless complaint, since it can be useful to switch back and forth between numbered and unnumbered can be useful during article development. So now `\TB@startsection` is just a synonym for `\@startsection`.

```

1573 \def\TB@startsection#1{\@startsection#1}%

```

`\TB@safe@startsection` is to be used where `\section*` (etc.) appear in places where the request is OK (because it's built in to some macro we don't fiddle with).

```
1574 \def\TB@safe@startsection#1{\@startsection#1}
```

The `SafeSection` environment allows use of `*`-forms of sectioning environments. It's not documented for the general public: it's intended as an editor's facility.

```
1575 \newenvironment{SafeSection}%
1576 {\let\TB@startsection\TB@safe@startsection}%
1577 {}
```

And now for the exciting sectioning commands that L^AT_EX defines but we don't have a definition for (whatever else, we don't want Lamport's originals, which come out 'like the blare of a bugle in a lullaby'¹).

The three inappropriate ones are subparagraph (indistinguishable from paragraph), and chapter and part. The last seemed almost to be defined in an early version of these macros, since there was a definition of `\l@part`. I've not got down to where that came from (or why). If class option `NONUMBER` is in effect, we also suppress `\paragraph`, since it has no parallel in the plain style.

```
1578 \if@numbersec
1579 \def\subparagraph{\TB@nosection\subparagraph\paragraph}
1580 \else
1581 \def\paragraph{\TB@nosection\paragraph\subsubsection}
1582 \def\subparagraph{\TB@nosection\subparagraph\subsubsection}
1583 \fi
1584 \def\chapter{\TB@nosection\chapter\section}
1585 \def\part{\TB@nosection\part\section}
1586 \def\TB@nosection#1#2{\TBwarning{class does not support \string#1,
1587 \string#2\space used instead}\#2}
```

`\l@<sectioninglevel>` is for table of contents (of an article). We define new macros to allow easily changing the font used for toc entries (for *TUGboat*, we usually want roman, not bold), and the space between entries. Nelson Beebe and Frank Mittelbach's articles often have toc's (and few others). Also turn off microtype protrusion after

Contents

, or leaders get messed up.

```
1588 \def\TBtocsectionfont{\normalfont}
1589 \newskip\TBtocsectionspace \TBtocsectionspace=1.0ex\@plus\p@
1590 % |#1| is both the section number and title, as in
1591 % |{\numberline {1}Introduction}|.
1592 % |#2| is the page number.
1593 %
1594 % Per Ulrike, the hook calls are for tagging, introduced with the
1595 % June 2023 \LaTeX.
1596 % qq need to also do subsections like tb137carlisle to avoid hyphenation
```

¹Thurber, *The Wonderful O*

```

1597 \def\l@section#1#2{%
1598   \addpenalty{\@secpenalty}%
1599   \addvspace{\TBtocsectionspace}%
1600   \@tempdima 1.5em
1601   \beginngroup
1602     \parindent\z@
1603     \rightskip=0pt plus2em
1604     \parfillskip\z@
1605     \hyphenpenalty=10000
1606     \TBtocsectionfont
1607     \leavevmode
1608     \advance\leftskip by \@tempdima % space between section number and text
1609     \hskip-\leftskip
1610     %
1611     \ifx\UseHookWithArguments\undefined\else % hook before number and text
1612       \UseHookWithArguments{contentsline/text/before}{4}
1613       {\toclevel@part}{#1}{#2}{\@contentsline@destination}%
1614     \fi
1615     %
1616     % don't worry if this cs is not defined, hence the \csname.
1617     % If it doesn't exist, we just typeset #1 as text.
1618     \csname contentsline@text@1@format\endcsname
1619     {#1}% number and title
1620     \unskip % avoid extra space just in case
1621     \csname pdfmakespace\endcsname % fake space if pdftex
1622     ~% ensure at least a word space between text and page number
1623   }
1624   %
1625   \ifx\UseHookWithArguments\undefined\else % hook after number and text
1626     \UseHookWithArguments{contentsline/text/after}{4}
1627     {\toclevel@part}{#1}{#2}{\@contentsline@destination}%
1628   \fi
1629   \nobreak\hfil
1630   \nobreak
1631   % page number
1632   \hb@xt@\@pnumwidth{\hfil
1633     \ifx\UseHookWithArguments\undefined\else
1634       \UseHookWithArguments{contentsline/page/before}{4}
1635       {\toclevel@part}{#1}{#2}{\@contentsline@destination}%
1636     \fi
1637     \tutypesetpageno{#2}%
1638     \ifx\UseHookWithArguments\undefined\else
1639       \UseHookWithArguments{contentsline/page/after}{4}
1640       {\toclevel@part}{#1}{#2}{\@contentsline@destination}%
1641     \fi
1642   }\par
1643 \endgroup}

```

3.17 Appendices

Appendices (which are really just another sort of section heading) raise a problem: if the sections are unnumbered, we plainly need to restore the section numbering, which in turn allows labelling of section numbers again (`\TBnolimelabel` happens

before the `\refstepcounter`, so its effects get lost ... what a clever piece of design that was). So here we go:

```
1644 \renewcommand{\appendix}{\par
1645   \renewcommand{\thesection}{\@Alph\c@section}%
1646   \setcounter{section}{0}%
1647   \if@numbersec
1648   \else
1649     \setcounter{secnumdepth}{1}%
1650   \fi
```

Now: is this the start of an appendix environment? This can be detected by looking at `\@currentenv`; if we are, we need to relay to `\@appendix@env` to pick up the optional argument.

```
1651 \def\@tempa{appendix}
1652 \ifx\@tempa\@currentenv
1653   \expandafter\@appendix@env
1654 \fi
1655 }
```

Here we deal with `\begin{appendix}[\langle app-name \rangle]`

```
1656 \newcommand{\app@prefix@section}{}
1657 \newcommand{\@appendix@env}[1][Appendix]{%
1658   \renewcommand{\@seccntformat}[1]{\csname app@prefix@##1\endcsname
1659     \csname the##1\endcsname\quad}%
1660   \renewcommand{\app@prefix@section}{#1 }%
1661 }
```

Ending an appendix environment is pretty trivial...

```
1662 \let\endappendix\relax
```

3.18 References

If the sections aren't numbered, the natural tendency of the author to cross-reference (which, after all, is one of the things $\text{\LaTeX}$ is for ever being advertised as being good at) can cause headaches.

The following command is used by each of the sectioning commands to make a following `\ref` command bloop at the author. Even if the author then ignores the complaint, the poor old editor may find the offending `\label` rather more easily.

(Note that macro name is to be read as “*noli me label*” (I don't know the mediæval Latin for ‘label’).

Comment To come (perhaps): detection of the act of labelling, and an analogue of `\ifG@refundefined` for this sort of label

```
1663 \def\TB@nolimelabel{%
1664   \def\@currentlabel{%
1665     \protect\TBWarning{%
1666       Invalid reference to numbered label on page \thepage
1667       \MessageBreak made%
1668     }%
1669     \textbf{?!?}%
1670   }%
1671 }
```

3.19 Title references

This is a first cut at a mechanism for referencing by the title of a section; it employs the delightfully simple idea Sebastian Rahtz has in the `nameref` package (which is part of `hyperref`). As it stands, it lacks some of the bells and whistles of the original, but they could be added; this is merely proof-of-concept.

The name label comes from the moveable bit of the section argument; we subvert the `\@sect` and `\@ssect` commands (the latter deals with starred section commands) to grab the relevant argument.

As of the June 2023 L^AT_EX (or somewhat earlier, but this is good enough), there are hooks that allow us to avoid redefining `\@sect` and `\@ssect`.

```

1672 \@ifl@t@r\fmtversion{2023-06-01}{-}{%
1673 \let\TB@@sect\@sect
1674 \let\TB@@ssect\@ssect
1675 \def\@sect#1#2#3#4#5#6[#7]#8{%
1676   \def\@currentlabelname{#7}%
1677   \TB@@sect{#1}{#2}{#3}{#4}{#5}{#6}[{#7}]{#8}%
1678 }
1679 \def\@ssect#1#2#3#4#5{%
1680   \def\@currentlabelname{#5}%
1681   \TB@@ssect{#1}{#2}{#3}{#4}{#5}%
1682 }
1683 } % LaTeX earlier than June 2023

```

We output the name label as a second `\newlabel` command in the `.aux` file. That way, packages such as `varioref` which also read the `.aux` information can still work. So we redefine `\label` to first call the standard L^AT_EX `\label` and then write our named label as `nr<label>`.

Similarly, we only need this with pre-June 2023 L^AT_EX. With more recent LaTeX, define `currentlabelname` via hooks.

```

1684 \@ifl@t@r\fmtversion{2023-06-01}{-}{%
1685   \RequirePackage{getttitlestring}
1686   \AddToHookWithArguments{cmd/@sect/before}{%
1687     \GetTitleString{#7}%
1688     \let\@currentlabelname\GetTitleStringResult}%
1689   \AddToHookWithArguments{cmd/@ssect/before}{%
1690     \GetTitleString{#5}%
1691     \let\@currentlabelname\GetTitleStringResult}%
1692 }{% else older latex:
1693   \let\@savelatexlabel=\label % so save original LaTeX command
1694   %
1695   \def\label#1{%
1696     \@savelatexlabel{#1}%
1697     \@bsphack
1698     \if@filesw
1699       \protected@write\@auxout{%
1700         {\string\newlabel{nr@#1}{\@currentlabel}{\@currentlabelname}}}%
1701     \fi
1702     \@esphack}
1703   % in case there are no sectioning commands:
1704   \let\@currentlabelname\@empty
1705 }

```

Getting named references is then just like getting page references in the L^AT_EX kernel (see `ltxref.dtx`).

The above was written by RobinF decades ago; the macros in *TUGboat* were never changed. Meanwhile, the `\nameref` in *hyperref* has changed many times, and we want to use its version if available. So we provide our `\nameref` `\AtBeginDocument`, so as not to overwrite any previous version. Until May 2022, *hyperref* silently overwrote an existing definition, that is, *TUGboat*'s. But now it is no longer silent.

It seems that all the internal definitions above do not cause problems, so just let them alone.

```

1706 \AtBeginDocument{%
1707   \ifl@t@r\fmtversion{2023-06-01}%
1708   { % after June 2023, LaTeX stores the label name; use that.
1709     \long\def\@thirdoffive#1#2#3#4#5{#3}
1710     \providecommand\nameref[1]{%
1711       \expandafter\@setref
1712       \csname r@#1\endcsname\@thirdoffive{#1}}%
1713   }
1714   { % for earlier versions, still avoid overwriting \nameref per above.
1715     % but if not otherwise defined, use the "nr" label defined by our \label.
1716     \providecommand\nameref[1]{%
1717       \expandafter\@setref
1718       \csname r@nr@#1\endcsname\@secondoftwo{#1}}%
1719   }%
1720 }
```

3.20 Float captions

By analogy with what we've just done to section titles and the like, we now do our best to discourage hyphenation within captions. We also typeset them in `\small` (actually `\tubcaptionfonts`).

First, let's define a dimension by which we will indent full-page captions. We'll also use this to indent abstracts in proceedings style.

`\@tubfullpageindent`

```

1721 \newdimen\@tubfullpageindent
1722 \@tubfullpageindent = \if@tubtwocolumn 4.875pc \else 3.875pc \fi
```

One-line captions are normally centered, but sometimes we want to set them flush left for consistency with other nearby figures.

`\tubcaptionleftglue`

```

1723 \let\tubcaptionleftglue=\hfil
```

For *TUGboat*, we like 9pt captions to help differentiate from the main text. Make a macro so we can easily override.

```

1724 \def\tubcaptionfonts{\small}%
```

One more *TUGboat* thing for `\@makecaption`: we prefer to have captions below figures (normal), but above tables (not what L^AT_EX expects). We've defined `\abovecaptionskip` to be nonzero and `\belowcaptionskip` to be zero, which is thus correct for tables, but incorrect for figures; thus we use the values reversed, so that `\abovecaptionskip` is

```

1725 \def\tbtablestring{table}
1726 \long\def\@makecaption#1#2{%
1727   %\TBDebug{dtx.capttype=\@capttype,str=\@tablestring}%
1728   \ifx\@capttype\tbtablestring
1729     \vskip\belowcaptionskip % table reversed (zero)
1730   \else
1731     \vskip\abovecaptionskip % non-table normal above (nonzero)
1732   \fi
1733   % try in an hbox:
1734   \sbox\@tempboxa{\tubcaptionfonts \frenchspacing \tubmakecaptionbox{#1}{#2}}%
1735   \ifdim \wd\@tempboxa > \hsize
1736     {% caption doesn't fit on one line; set as a paragraph.
1737       \tubcaptionfonts \raggedright \hyphenpenalty=\@M \parindent=1em
1738       % indent full-width captions {figure*}, but not single-column {figure}.
1739       \ifdim\hsize = \textwidth
1740         \leftskip=\@tubfullpageindent \rightskip=\leftskip
1741         \advance\rightskip by 0pt plus2em % increase acceptable raggedness
1742       \fi
1743       \noindent \tubmakecaptionbox{#1}{#2}\par}%
1744   \else
1745     % fits on one line; use the hbox, usually centered. Do not reset its glue.
1746     \global\@minipagefalse
1747     \hb@xt@\hsize{\tubcaptionleftglue\box\@tempboxa\hfil}%
1748   \fi
1749   %
1750   % Caption below the caption (space wanted for tables).
1751   \ifx\@capttype\tbtablestring
1752     \vskip\abovecaptionskip % table reversed (nonzero)
1753   \else
1754     \vskip\belowcaptionskip % non-table normal below (zero)
1755   \fi
1756 }

```

Also use `\tubcaptionfonts` for the caption labels, and put the label (e.g., “Figure 1”) in bold. Use a macro so we can override.

```

1757 \def\tubmakecaptionbox#1#2{{\tubcaptionfonts\textbf{#1}}:\ #2}%

```

We used to switch the labels into bold this way, but it’s better to do it as part of `\@makecaption` since then it will apply to other floating types, such as those created by the `newfloat` package. (E.g., `tb142duck-pylatex`.)

```

\def\fnun@figure{{\tubcaptionfonts \bf \figurename\nobreakspace\thefigure}}
\def\fnun@table{{\tubcaptionfonts \bf \tablename\nobreakspace\thetable}}

```

If the `listings` package is being used, bold for its label too; this `\def` is too early, but maybe `listings` will play nice later.

```

1758 \def\lstlistingnamestyle{\bfseries}

```

Let’s reduce the default space above captions, and give it some flexibility. The default is 10pt, which seems rather much.

```

1759 \setlength\abovecaptionskip{4pt plus2pt minus1pt} % default 10pt

```

Let’s also reduce the space between floats, and between floats and text. Strangely, it seems to be these that count, rather than `\dbl...`, at least sometimes.

```

1760 \setlength\floatsep { 8pt plus3pt minus2pt} % default 12pt plus2pt minus2pt
1761 \setlength\textfloatsep{12pt plus4pt minus3pt} % default 20pt plus2pt minus4pt

```

We want to allow more floats at the top/bottom/everywhere on a page; all depends on their content.

```
1762 \setcounter{bottomnumber}{2} % default 1
1763 \setcounter{topnumber}{4} % default 2
1764 \setcounter{totalnumber}{6} % default 3
```

3.21 Size changing commands

In addition to their ‘normal’ effects, these commands change the glue around displays to match that of being around lists. Otherwise, in a column that contains both displays and “lists”, including verbatim blocks, etc., the different spacing can become noticeable. For example, `tb145beisert-eqnlines`, page 4, column 1. $\text{\LaTeX}$ ’s default for `\abovedisplayskip` and `\belowdisplayskip` are much larger (and not equal).

```
\normalsize
\small 1765 \renewcommand{\normalsize}{%
\footnotesize 1766 \setfontsize\normalsize\@xpt\@xipt
\abovedisplayskip 1767 \abovedisplayskip=\topsep % same as space above lists/verbatim/etc.
\belowdisplayskip 1768 \belowdisplayskip=\abovedisplayskip
1769 \abovedisplayshortskip=\z@\@plus 3\p@
1770 \belowdisplayshortskip=\p@\@plus 3\p@\@minus\p@
1771 }
1772
1773 \renewcommand{\small}{%
1774 \setfontsize\small\@ixpt{11}%
1775 \abovedisplayskip=2pt plus.66pt minus.66pt % 2/3 \topsep
1776 \belowdisplayskip=\abovedisplayskip
1777 \abovedisplayshortskip=\z@\@plus 2\p@
1778 \belowdisplayshortskip=\p@\@plus 2\p@\@minus\p@
1779 }
1780
1781 \renewcommand{\footnotesize}{%
1782 \setfontsize\footnotesize\@viipt{9.5}%
1783 \abovedisplayskip=1pt plus.33pt minus.33pt % 1/3 \topsep
1784 \belowdisplayskip=\abovedisplayskip
1785 \abovedisplayshortskip=\z@\@plus 3\p@
1786 \belowdisplayshortskip=\p@\@plus 3\p@\@minus\p@
1787 }
```

3.22 Lists and other text inclusions

```
1788 \def\@listi{%
1789 \leftmargin\leftmarginii\parsep=\p@\@plus\p@\@minus\p@
1790 \itemsep=\parsep
1791 \listparindent=1em
1792 }
1793
1794 \def\@listii{%
1795 \leftmargin\leftmarginii
1796 \labelwidth=\leftmarginii \advance\labelwidth-\labelsep
1797 \topsep=2\p@\@plus\p@\@minus\p@ % space between first item and preceding
1798 \parsep=\p@\@plus\p@\@minus\p@
```

```

1799 \itemsep=\parsep % space between successive items
1800 \listparindent=1em % indentation of subsequent paragraphs
1801 }
1802
1803 \def\@listiii{%
1804 \leftmargin=\leftmarginiii
1805 \labelwidth=\leftmarginiii \advance\labelwidth-\labelsep
1806 \topsep=\p@\@plus\p@\@minus\p@
1807 \parsep=\z@
1808 \itemsep=\topsep
1809 \listparindent=1em
1810 }
1811 \def\quote{\list{}{\rightmargin.5\leftmargin}\item[]}

```

From Dominik Wujastyk's font article. First paragraph of a quotation will not be indented, and right margin is decreased for narrow columns.

```

1812 \renewcommand{\quotation}{\list{}{\listparindent 1.5em
1813 \rightmargin.5\leftmargin\parsep \z@\@plus\p@}\item[]

```

The `compactitemize`, `compactenumerate`, and `compactdescription` environments, without space between the items.

```

1814 \newenvironment{compactitemize}%
1815 {\begin{itemize}%
1816 \setlength{\itemsep}{0pt}%
1817 \setlength{\parskip}{0pt}%
1818 \setlength{\parsep}{0pt}%
1819 }%
1820 {\end{itemize}}
1821 %
1822 \newenvironment{compactenumerate}%
1823 {\begin{enumerate}%
1824 \setlength{\itemsep}{0pt}%
1825 \setlength{\parskip}{0pt}%
1826 \setlength{\parsep}{0pt}%
1827 }%
1828 {\end{enumerate}}
1829 %
1830 \newenvironment{compactdescription}%
1831 {\begin{description}%
1832 \setlength{\itemsep}{0pt}%
1833 \setlength{\parskip}{0pt}%
1834 \setlength{\parsep}{0pt}%
1835 }%
1836 {\end{description}}
1837 %

```

3.23 Some fun with verbatim

The plain *TUGboat* style allows [optional] arguments to its `\verbatim` command. This will allow the author (or editor) to specify a range of exciting features; we would definitely like the numbered verbatim style for code (that facility is reserved for a future version of this package), and the present little bit of code imposes the `\ruled` option on the built-in `verbatim` environment. (Note that we don't yet deal with `verbatim*`, which is in itself an option to the plain original.)

We start by saving various bits and bobs whose operation we're going to subvert, then starting our `\def\verbatim`.

With tagging enabled, this entire definition is irrelevant; L^AT_EX makes its own definition. So we have to alter things with hooks (run below) instead of inline in the code.

```
1838 %\let\@TB@verbatim\@verbatim
1839 \let\@TBverbatim\verbatim
1840 \let\@TBendverbatim\endverbatim
1841 %
1842 \def\verbatim{%
```

This `\obeylines` prevents the `\futurelet` from propagating into the body of the verbatim, thus causing lines that start with odd characters (like `#` or even `\`) to behave peculiarly.

```
1843 \obeylines
1844 \futurelet\reserved@a\@switch@sqbverbatim
1845 }% end \def\verbatim
1846 %
1847 \def\@switch@sqbverbatim{\ifx\reserved@a[%]
1848 \expandafter\@sqbverbatim\else
1849 \def\reserved@b{\@sqbverbatim[]}\expandafter\reserved@b\fi}
1850 %
1851 \def\@sqbverbatim[#1]{%
```

The command `\ruled` is only defined inside the `verbatim` optional argument. It specifies that there should be rules above and below the verbatim block. This doesn't work with tagging, but we don't expect that's ever desired nowadays; `\ruled` is rarely used.

```
1852 \def\ruled{\let\if@ruled\iftrue}%
```

Execute the arguments we've got, and relay to a (hacked) copy of the L^AT_EX verbatim environment defined next.

```
1853 #1\@TBverbatim}
```

The built-in environment itself relays to `\@verbatim`, which we've subverted to impose our views on appearance.

```
1854 \def\@verbatim{%
```

Deal with `\ruled`:

```
1855 \if@ruled\trivlist\item\hrule\kern5\p@\nobreak\fi
```

Now, the code out of the original verbatim environment:

```
1856 \trivlist \item\relax
1857 \if@minipage\else\vskip\parskip\fi
1858 \leftskip\@totalleftmargin\rightskip\z@skip
1859 \parindent\z@\parfillskip\@flushglue\parskip\z@skip
1860 \@@par
1861 \@tempswafalse
1862 \def\par{%
1863 \if@tempswa
1864 \leavevmode \null \@@par\penalty\interlinepenalty
1865 \else
1866 \@tempswatrue
```

```

1867     \ifhmode\@@par\penalty\interlinepenalty\fi
1868     \fi}%
1869     \obeylines \verbatim@font \@noligs
1870     \let\do\@makeoother \dospecials
1871     \everypar \expandafter{\the\everypar \unpenalty}%
1872 }% end |\@sqbverbatim|

```

To end the environment, we do everything in reverse order: relay via the copy we made of `\endverbatim`, and then finish off the option changes (again `\ruled` only, so far).

```

1873 \def\endverbatim{\@TBendverbatim
1874   \if@ruled\kern5\p@\hrule\endtrivlist\fi}

```

Define the `\if` used by the `\ruled` option:

```

1875 \let\if@ruled\iffalse

```

As another option for `verbatim`, the command `\makevmeta` says to make `!j...j` do `<...>`. We define it outside `verbatim` so that it will work with tagging, as `\tubmakevmeta` to avoid namespace pollution. See the test file `tubvmetatest.tex`. Ideas from Frank Mittelbach and Tim Prescott.

```

1876 \def\tubmakevmeta{\makeescape\! \let<\tubverb@meta \tubverb@clearliglist}
1877 \def\tubverb@meta#1>{\meta{#1}}

```

The default `verbatim` defines “`j̣,-`” as active characters to stop ligatures; remove `j̣` from the list so we get normal characters. Just hope that the CM `j̣` ligatures aren’t actually desired.

```

1878 \def\tubverb@clearliglist{%
1879   \def\verbatim@nolig@list{\do\‘\do\,\do\’\do\~}%
1880 }

```

Then, use a hook to define `\makevmeta` as `\tubmakevmeta`, so it will work with tagging.

```

1881 \AddToHook{env/verbatim/begin}{\let\makevmeta\tubmakevmeta}

```

For older $\text{\LaTeX}$ versions we start the macro with `\par` to avoid a common error: if the optional argument is `\small`, and the document has no blank line before the `verbatim` block, we don’t want that preceding paragraph to be set with `\small`’s line spacing.

As of the 2026-11-01 $\text{\LaTeX}$ release, $\text{\LaTeX}$ itself fixes this problem. Then we don’t want the `\par`, because it interferes with correct tagging. We have to add this in a hook also because the tagging redefines `\verbatim`. <https://github.com/latex3/tagging-project/issues/1503>

```

1882 \AddToHook{env/verbatim/begin}{%
1883   \IfFormatAtLeastTF{2026-11-01}
1884     {\relax}% new enough, no need for \par
1885     {\par}% older, needs \par to avoid baselineskip leakage
1886 }

```

Finally, if `microtype` is loaded, we want it to be deactivated in `verbatim` blocks. It often manipulates a leading `\` rather too much, thus messing with the visible fixed-width alignment.

```

1887 \AtBeginDocument{%
1888   \@ifpackageloaded{microtype}
1889     {\g@addto@macro\@verbatim{\microtypesetup{activate=false}}}{ }
1890 }

```

3.24 Bibliography

This is more or less copied verbatim from Glenn Paulley's *chicago.sty* (gnpaulle@bluebox.uwaterloo.ca). It produces an author-year citation style bibliography, using output from the BIBTEX style file based on that by Patrick Daly. It needs extra macros beyond those in standard L^AT_EX to function properly. The form of the bibitem entries is:

```
\bibitem[\protect\citeauthoryear{Jones, Baker, and Smith}
{Jones et al.}{1990}{key}...
```

The available citation commands are:

```
\cite{key}      → (Jones, Baker, and Smith 1990)
\citeA{key}     → (Jones, Baker, and Smith)
\citeNP{key}    → Jones, Baker, and Smith 1990
\citeANP{key}   → Jones, Baker, and Smith
\citeN{key}     → Jones, Baker, and Smith (1990)
\shortcite      → (Jones et al. 1990)
\citeyear       → (1990)
\citeyearNP     → 1990
```

First of all (after checking that we're to use Harvard citation at all), make a copy of L^AT_EX's default citation mechanism.

```
1891 \if@Harvardcite
1892 \let\@internalcite\cite
```

Normal forms.

```
1893 \def\cite{\def\@citesep{-1000}%
1894   \def\@cite##1##2{##1\if@tempswa , ##2\fi}}%
1895   \def\citeauthoryear##1##2##3{##1, ##3\@internalcite}
1896 \def\citeNP{\def\@citesep{-1000}%
1897   \def\@cite##1##2{##1\if@tempswa , ##2\fi}%
1898   \def\citeauthoryear##1##2##3{##1, ##3\@internalcite}
1899 \def\citeN{\def\@citesep{-1000}%
1900   \def\@cite##1##2{##1\if@tempswa , ##2\else{}}\fi}%
1901   \def\citeauthoryear##1##2##3{##1 (##3)\@citedata}
1902 \def\citeA{\def\@citesep{-1000}%
1903   \def\@cite##1##2{##1\if@tempswa , ##2\fi}}%
1904   \def\citeauthoryear##1##2##3{##1\@internalcite}
1905 \def\citeANP{\def\@citesep{-1000}%
1906   \def\@cite##1##2{##1\if@tempswa , ##2\fi}%
1907   \def\citeauthoryear##1##2##3{##1\@internalcite}
```

Abbreviated forms (using *et al.*)

```
1908 \def\shortcite{\def\@citesep{-1000}%
1909   \def\@cite##1##2{##1\if@tempswa , ##2\fi}}%
1910   \def\citeauthoryear##1##2##3{##2, ##3\@internalcite}
1911 \def\shortciteNP{\def\@citesep{-1000}%
1912   \def\@cite##1##2{##1\if@tempswa , ##2\fi}%
1913   \def\citeauthoryear##1##2##3{##2, ##3\@internalcite}
1914 \def\shortciteN{\def\@citesep{-1000}%
1915   \def\@cite##1##2{##1\if@tempswa , ##2\else{}}\fi}%
1916   \def\citeauthoryear##1##2##3{##2 (##3)\@citedata}
1917 \def\shortciteA{\def\@citesep{-1000}%
```

```

1918 \def\cite##1##2{##1\if@tempswa , ##2\fi}}%
1919 \def\citeauthoryear##1##2##3{##2}\@internalcite}
1920 \def\shortciteANP{\def\citesep{-1000}%
1921 \def\cite##1##2{##1\if@tempswa , ##2\fi}%
1922 \def\citeauthoryear##1##2##3{##2}\@internalcite}

```

When just the year is needed:

```

1923 \def\citeyear{\def\citesep{-1000}%
1924 \def\cite##1##2{##1\if@tempswa , ##2\fi}}%
1925 \def\citeauthoryear##1##2##3{##3}\@citedata}
1926 \def\citeyearNP{\def\citesep{-1000}%
1927 \def\cite##1##2{##1\if@tempswa , ##2\fi}%
1928 \def\citeauthoryear##1##2##3{##3}\@citedata}

```

Place commas in-between citations in the same `\citeyear`, `\citeyearNP`, `\citeN`, or `\shortciteN` command. Use something like `\citeN{ref1,ref2,ref3}` and `\citeN{ref4}` for a list.

```

1929 \def\@citedata{%
1930 \ifnextchar [{\@tempswatrue\@citedatax}%
1931 \ifnextchar [%{\@tempswafalse\@citedatax[]}%
1932 }
1933
1934 \def\@citedatax[#1]#2{%
1935 \if@files\immediate\write\@auxout{\string\citation{#2}}\fi%
1936 \def\@citea{\@cite{\@for\citeb:=#2\do%
1937 {\@citea\def\@citea{ }\@ifundefined% by Young
1938 {b@\citeb}{\bf ?}%
1939 \@warning{Citation '\@citeb' on page \thepage \space undefined}}}%
1940 {\csname b@\citeb\endcsname}}{#1}}%

```

Don't box citations, separate with ; and a space; Make the penalty between citations negative: a good place to break.

```

1941 \def\@citex[#1]#2{%
1942 \if@files\immediate\write\@auxout{\string\citation{#2}}\fi%
1943 \def\@citea{\@cite{\@for\citeb:=#2\do%
1944 {\@citea\def\@citea{ }\@ifundefined% by Young
1945 {b@\citeb}{\bf ?}%
1946 \@warning{Citation '\@citeb' on page \thepage \space undefined}}}%
1947 {\csname b@\citeb\endcsname}}{#1}}%

```

No labels in the bibliography.

```

1948 \def\@biblabel#1{}

```

Set length of hanging indentation for bibliography entries.

```

1949 \newlength{\bibhang}
1950 \setlength{\bibhang}{2em}

```

Indent second and subsequent lines of bibliographic entries. Stolen from openbib.sty: `\newblock` is set to `{}`.

```

1951 \newdimen\bibindent
1952 \bibindent=1.5em
1953 \@ifundefined{refname}%
1954 {\newcommand{\refname}{References}}%
1955 {}%

```

For safety's sake, suppress the `\TB@startsection` warnings here...

```

1956 \def\thebibliography#1{% for harvardcite
1957   \let\TB@startsection\TB@safe@startsection
1958   \section*{\refname
1959     \@mkboth{\uppercase{\refname}}{\uppercase{\refname}}}%
1960   \list{[\arabic{enumi}]}{%
1961     \labelwidth\z@ \labelsep\z@
1962     \leftmargin\bibindent
1963     \itemindent -\bibindent
1964     \listparindent \itemindent
1965     \parsep \z@
1966     \usecounter{enumi}}%
1967   \def\newblock{}%
1968   \BibJustification
1969   \frenchspacing % more than just period, see comments below
1970 }

```

`etal` Other bibliography odds and ends.

```

\bibentry 1971 \def\etal{et\,al.\@}
1972 \def\bibentry{%
1973   \smallskip
1974   \hangindent=\parindent
1975   \hangafter=1
1976   \noindent
1977   \sloppy
1978   \clubpenalty500 \widowpenalty500
1979   \frenchspacing
1980 }

```

`\bibliography` Changes made to accommodate TUB file naming conventions

```

\bibliographystyle 1981 \def\bibliographystyle#1{%
1982   \if@filesw
1983     \immediate\write\@auxout{\string\bibdata{\@tubfilename{#1}}}%
1984   \fi
1985   \@input{\jobname.bbl}%
1986 }
1987 \def\bibliographystyle#1{%
1988   \if@filesw
1989     \immediate\write\@auxout{\string\bibstyle{\@tubfilename{#1}}}%
1990   \fi
1991 }

```

`\thebibliography` If the user has asked to use L^AT_EX's default citation mechanism (using the `rawcite` option), we still need to patch `\sloppy` to support justification of the body of the bibliography. We kludge in a call to `\frenchspacing` too, since there is no reason to change only period's `\sfcode`, as L^AT_EX's original `thebibliography` (in `classes.dtx`) does.

By the way, `amsgen.sty` changes `\frenchspacing` to set the `\sfcode` of punctuation character to successively decreasing integers ending at 1001 for comma. Thus its 1006 for period is overwritten to 1000 for `thebibliography`, making `amsgen's \@addpunct` ineffective. Don't know what that means in practice, if anything.

Back here, we also play with *The T_EXbook* since we always have, though that is no longer needed.

```

1992 \else % not harvardcite
1993 \let\TB@origthebibliography\thebibliography
1994 \def\thebibliography{%
1995   \let\TB@startsection\TB@safe@startsection
1996   \def\sloppy{\frenchspacing\BibJustification}%
1997   \TB@origthebibliography} % latex's thebibliography now reads args.
1998 \fi % not harvardcite

```

`\BibJustification` defines how the bibliography is to be justified. The Lamport default is “`\sloppy`”, but we find some sort of ragged right setting is almost always preferred, so (as of 2025) make that the default. (`\BibJustification` is nevertheless reset to its default value at the start of a paper.)

```

1999 \let\TB@sloppy\sloppy
2000 \let\BibJustification\raggedright
2001 \newcommand{\SetBibJustification}[1]{%
2002   \renewcommand{\BibJustification}{#1}%
2003 }
2004 \ResetCommands\expandafter{\the\ResetCommands
2005   \let\BibJustification\TB@sloppy
2006 }

```

3.25 Registration marks

We no longer use these since Cadmus does not want them.

```

2007 \def\HorzR@gisterRule{\vrule \@height 0.2\p@ \@depth\z@ \@width 0.5in }
2008 \def\DownShortR@gisterRule{\vrule \@height 0.2\p@ \@depth 1pc \@width 0.2\p@ }
2009 \def\UpShortR@gisterRule{\vrule \@height 1pc \@depth\z@ \@width 0.2\p@ }

```

“T” marks centered on top and bottom edges of paper

```

2010 \def\ttopregister{\dlap{%
2011   \hb@xt@\trimwd{\HorzR@gisterRule \hfil \HorzR@gisterRule
2012     \HorzR@gisterRule \hfil \HorzR@gisterRule}%
2013   \hb@xt@\trimwd{\hfil \DownShortR@gisterRule \hfil}}}
2014 \def\tbotregister{\ulap{%
2015   \hb@xt@\trimwd{\hfil \UpShortR@gisterRule \hfil}%
2016   \hb@xt@\trimwd{\HorzR@gisterRule \hfil \HorzR@gisterRule
2017     \HorzR@gisterRule \hfil \HorzR@gisterRule}}}
2018 \def\topregister{\ttopregister}
2019 \def\botregister{\tbotregister}

```

3.26 Running headers and footers

```

2020 \def\rtitlex{\def\texttub##1{{\normalsize\textrm{##1}}}\TUB, \volx}

```

registration marks; these are temporarily inserted in the running head

```

2021 \def\MakeRegistrationMarks{}
2022 \def\UseTrimMarks{%
2023   \def\MakeRegistrationMarks{%
2024     \ulap{\rlap{%
2025       \vbox{\dlap{\vbox to\trimlgt{\vfil\botregister}}}%
2026       \topregister\vskip \headmargin \vskip 10\p@}}}%

```

```

2027 }
2028 % put issue identification and page number in header.
2029 \def\@oddhead{\MakeRegistrationMarks
2030   \frenchspacing
2031   \normalsize\csname normalshape\endcsname\rm \tubheadhook
2032   \rtitlex\qqquad \midrttitle\hfil
2033   \rtitlenexttopage\quad\tubtypesetpageno{\thepage}}
2034 \def\@evenhead{\MakeRegistrationMarks
2035   \frenchspacing
2036   \normalsize\csname normalshape\endcsname\rm \tubheadhook
2037   \tubtypesetpageno{\thepage}\quad\rtitlenexttopage
2038   \hfil\midrttitle \qqquad\rtitlex}
2039
2040 % Put a ? into the page number in the headers in all but a final run, so
2041 % people aren't tempted to cite it.
2042 %
2043 \newcommand{\tubtypesetpageno}[1]{%
2044   \ifnum #1>900
2045     % in CM, numerals are exactly .5em.
2046     %
2047     % The \texorpdfstring avoids the usual hyperref warning:
2048     % Token not allowed in a PDF string ... removing '\@ifnextchar'
2049     \texorpdfstring{\makebox[.5em][l]{\small ?}}{?}%
2050     %
2051     \textsl{\@arabic{\numexpr#1-900\relax}}}% assuming e-tex
2052   \else
2053     \@arabic{#1}%
2054   \fi
2055 }
2056 %
2057 % The above changes the page number in the headers and tocs. It does not
2058 % change the page number in cross-references, which will still show up
2059 % as '901' instead of '?1'. In order to do that, we'd have to redefine
2060 % |\thepage| (https://tex.stackexchange.com/questions/687258).
2061 %
2062 % The problem is that |\thepage| is not expected to contain typesetting
2063 % commands like |\makebox| and |\textsl|, but to expand to the simple
2064 % page number (in whatever form). For example, when redefining
2065 % |\thepage| to the above, terminal warnings then look like:
2066 % |LaTeX Warning: Citation 'foo' on page \makebox [.5em][l]{...|
2067 % losing the actual page number.
2068 %
2069 % So apparently there is no way to add the ? correctly in all contexts.
2070 %
2071 % BTW, such a custom page number format would also break makeindex,
2072 % etc., but for that we could provide the information. Per Ulrike:
2073 %\usepackage{index}
2074 %\newcommand\specialthepage{\interval{\value{page}-900}}
2075 %\newindex[specialthepage]*{default}{idx}{ind}{Index}
2076
2077 % can be used to reset the font, e.g., tb98kuester.
2078 \def\tubheadhook{}
2079
2080 % in case the official \author is too verbose for the footline.

```

```

2081 % (the \shortauthor / \rhAuthor stuff is only enabled for proceedings, fix!)
2082 \def\tubrunningauthor{\@author}
2083
2084 % put title and author in footer.
2085 \def\@tubrunningfull{%
2086   \def\@oddfoot{% make line break commands produce a normal space
2087     \def\{\unskip\ \ignorespaces}%
2088     \let\newline=\%
2089     \tubtypesetdoi
2090     \frenchspacing\hfil\rhTitle}
2091   \def\@evenfoot{%
2092     \let\thanks\@gobble
2093     \tubtypesetdoi
2094     \frenchspacing\tubrunningauthor\hfil}
2095 }
2096
2097 % empty footer.
2098 \def\@tubrunningminimal{%
2099   \def\@oddfoot{\tubtypesetdoi\hfil}%
2100   \def\@evenfoot{\tubtypesetdoi\hfil}%
2101 }
2102
2103 % empty footer and header.
2104 \def\@tubrunningoff{%
2105   \@tubrunningminimal
2106   \def\@oddhead{\hfil}%
2107   \def\@evenhead{\hfil}%
2108 }
2109
2110 \def\ps@headings{}
2111 \pagestyle{headings}

```

Typeset the doi. The format we decided on looks like: <https://doi.org/10.47397/tb/41-3/tb129> where the last element is the \jobname.

We put this below the footline. The footer definitions above specify that it is always called, even if the regular footer is empty.

If the article started in the second column (option [secondcolstart]), we manually move the doi over.

We do not check for validity of \volno, \issno, \jobname. For testing, etc., seems simpler to just typeset what we've got. Other scripts will verify consistency.

```

2112 %
2113 \def\tubdoiprefix{10.47397/tb} % assigned to us by crossref.
2114 \def\tubabovedoi{} % fudge spacing or whatever.
2115 %
2116 \def\tubtypesetdoi{%
2117   \iftubomitdoioption\else % do if not explicit omission ...
2118     \ifnum\volno>0 % and if being run for production ...
2119       \iftubfinaloption % and if [final], even if pageno>900
2120         \vbox to 0pt{% don't impact normal layout
2121           \edef\thedoi{% but make url invalid if >900
2122             \ifnum\count0>900 example.org%
2123               \else doi.org\fi
2124             /\tubdoiprefix/\volno-\issno/\jobname
2125           }%

```

```

2126         \scriptsize
2127         \vskip\baselineskip
2128         \tubabovedoi
2129         \iftubsecondcolstart \moveright \tubcolwidthandgutter \fi
2130         \rlap{\expandafter\tbsurl\expandafter{\thedoi}}}%
2131         \vss
2132     }%
2133     \fi % tubfinaloption
2134     \fi % volno>0
2135     \fi % !tubomitdoioption
2136     \global\let\tubtypesetdoi\empty % only do it once, no matter what.
2137 }
2138 %
2139 %

```

3.27 Output routine

Modified to alter `\brokenpenalty` across columns

Comment We’re playing with fire here: for example, `\outputdblcol` has changed in $\text{\LaTeX 2}_\epsilon$ for 1995/06/01 (with the use of `\hb@xt@`). *This* time there’s no semantic change, but...

```

2140 \def\outputdblcol{\if@firstcolumn \global\@firstcolumnfalse
2141     \global\setbox\@leftcolumn\box\@outputbox
2142     \global\brokenpenalty10000
2143 \else \global\@firstcolumntrue
2144     \global\brokenpenalty100
2145     \setbox\@outputbox\vbox{\hb@xt@\textwidth{\hb@xt@\columnwidth
2146         {\box\@leftcolumn \hfil}\hfil \vrule \@width\columnseprule\hfil
2147         \hb@xt@\columnwidth{\box\@outputbox \hfil}}}\@combinedblfloats
2148     \outputpage \begingroup \@dblfloatplacement \@startdblcolumn
2149     \@whiles\if@fcolmade \fi{\outputpage\@startdblcolumn}\endgroup
2150     \fi}

```

3.28 Font-related definitions and machinery

These are mostly for compatibility with plain `tugboat.sty`

```

2151 \newif\ifFirstPar      \FirstParfalse
2152 \def\smc{\sc}
2153 \def\ninepoint{\small}
2154 \</classtail>

```

`\SMC` *isn’t* small caps—Barbara Beeton says she thinks of it as “big small caps”. She says (modulo capitalisation of things...):

For the things it’s used for, regular small caps are not appropriate—they’re too small. Real small caps are appropriate for author names (and are so used in continental bibliographies), section headings, running heads, and, on occasion, words to which some emphasis is to be given. `\SMC` was designed to be used for acronyms and all-caps abbreviations, which look terrible in small caps, but nearly as bad in all caps in the regular text size. The principle of using “one size smaller” than the text size is similar to the design of caps in German—where they

are smaller relative to lowercase than are caps in fonts intended for English, to improve the appearance of regular text in which caps are used at the heads of all nouns, not just at the beginnings of sentences.

2026 note from Karl: it's not so much "German" fonts in particular (few fonts nowadays are language-specific anyway), as a design choice in general. Some typefaces (e.g., Lucida, but there are many more) make capitals smaller than the ascender height; others (e.g., Computer Modern) don't. For the former, reducing further is neither necessary nor desirable, but we don't try to handle that case here, since *TUGboat* rarely uses any font other than CM and derivatives.

We define this in terms of the memory of the size currently selected that's maintained in `\@currsize`: if the user does something silly re. selecting fonts, we'll get the wrong results. The following code is adapted from an old version of `relsize.sty` by Donald Arseneau and Matt Swift. The order of examination of `\@currsize` is to do the most common cases first.

This is (I think) not a robust command since it's a conditional. <https://tex.stackexchange.com/questions/1274#issuecomment-4078534466>

```

2155 <*common>
2156 \newcommand{\SMC}{%
2157   \ifx\@currsize\normalsize\small\else
2158   \ifx\@currsize\small\footnotesize\else
2159   \ifx\@currsize\footnotesize\scriptsize\else
2160   \ifx\@currsize\large\normalsize\else
2161   \ifx\@currsize\Large\large\else
2162   \ifx\@currsize\LARGE\Large\else
2163   \ifx\@currsize\scriptsize\tiny\else
2164   \ifx\@currsize\tiny\tiny\else
2165   \ifx\@currsize\huge\LARGE\else
2166   \ifx\@currsize\Huge\huge\else
2167   \small\SMC@unknown@warning
2168 \fi\fi\fi\fi\fi\fi\fi\fi\fi\fi
2169 }
2170 \newcommand{\SMC@unknown@warning}{\TBWarning{\string\SMC: nonstandard
2171   text font size command -- using \string\small}}
2172 \newcommand{\textSMC}[1]{\{\SMC #1\}}

```

The `\acro` command uses `\SMC` as it was originally intended. Since the argument never contains lowercase letters (except in error), it resets the `spacefactor` to 1000 after inserting the text. We define it using `\NewDocumentCommand` as an easy way to make the command robust, and thus be usable in titles when tagging is active. Suggestion from Ulrike Fischer: <https://github.com/latex3/tagging-project/issues/1274#issuecomment-4078534466>.

```

2173 \NewDocumentCommand\acro{m}{\textSMC{#1}\@}
2174 </common>

```

3.29 Editor's notes and other footnotes

`\EdNote` allows the editor to enter notes in the text of a paper. If the command is given something that appears like an optional argument, the entire text of the note is placed in square brackets. (Yes, it really is!)

```

2175 <*classtail>
2176 \def\xEdNote{\EdNoteFont Editor's note:\enspace}
2177 \def\EdNote{\@ifnextchar[%]

```

```

2178 {%
2179   \ifvmode
2180     \smallskip\noindent\let\@EdNote@\@EdNote@v
2181   \else
2182     \unskip\quad\def\@EdNote@{\unskip\quad}%
2183   \fi
2184   \@EdNote
2185 }%
2186 \xEdNote
2187 }
2188 \long\def\@EdNote[#1]{%
2189   [\thinspace\xEdNote\ignorespaces
2190     #1%
2191     \unskip\thinspace]%
2192   \@EdNote@
2193 }
2194 \def\@EdNote@v{\par\smallskip}

```

Macros for Mittelbach's self-documenting style

```

2195 \def\SelfDocumenting{%
2196   \setlength\textwidth{31pc}
2197   \onecolumn
2198   \parindent \z@
2199   \parskip 2\p@\@plus\p@\@minus\p@
2200   \oddsidemargin 8pc
2201   \evensidemargin 8pc
2202   \marginparwidth 8pc
2203   \toks@\expandafter{\@oddhead}%
2204   \xdef\@oddhead{\hss\hb@xt@\pagewd{\the\toks@}}%
2205   \toks@\expandafter{\@evenhead}%
2206   \xdef\@evenhead{\hss\hb@xt@\pagewd{\the\toks@}}%
2207   \def\ps@titlepage{%
2208   }
2209   \def\ps@titlepage{}
2210
2211   % Let's have a little space after the footnote marker,
2212   % also a little more space for the marker itself.
2213   % The argument is the footnote text.
2214   \long\def\@makefnmark#1{%
2215     \parindent 1em
2216     \noindent
2217     \hb@xt@2em{}%
2218     \llap{\@makefnmark}\null
2219     $\mskip5mu$% space after marker
2220     #1% footnote text
2221   }
2222   %
2223   % For comparison, here is the original definition from classes.dtx:
2224   % \long\def\@makefnmark#1{%
2225     \parindent 1em
2226     \noindent
2227     \hb@xt@2em{\hss\@makefnmark}#1}

```

`\tubraggedfoot` Produce a ragged-right footnote.

```

2228 \newcommand{\tubraggedfoot}{\rightskip=\raggedskip plus\raggedstretch\relax}

\creditfootnote Sometimes we want the label “Editor’s Note:”, sometimes not.
\supportfootnote 2229 \def\creditfootnote{\nomarkfootnote\xEdNote}
2230 \def\supportfootnote{\nomarkfootnote\relax}

General macro \nomarkfootnote to make a footnote without a reference
mark, etc. #1 is an extra command to insert, #2 the user’s text.
2231 \gdef\nomarkfootnote#1#2{\begingroup
2232 \def\thefootnote{}%
2233 % no period, please, also no fnmark. Also no hyperref warning.
2234 \def\@makefntext##1{##1}%
2235 \def\Hy@Warning##1{}%
2236 \footnotetext{\noindent #1#2}%
2237 \endgroup}

```

3.30 Initialization

If we’re going to use Harvard-style bibliographies, we set up the bibliography style: the user doesn’t get any choice. (Not recommended.)

```

2238 \if@Harvardcite
2239 \AtBeginDocument{%
2240 \bibliographystyle{ltugbib}}%
2241 }
2242 \fi
2243 \authornumber\z@
2244 \let\@signature\@defaultsignature
2245 \InputIfFileExists{ltugboat.cfg}
2246 {\TBInfo{Loading ltugboat.cfg configuration information}}
2247 {}
2248 </classtail>

```

4 L^AT_EX 2_ε proceedings class (no longer used)

\@tugclass Make the code of ltugboat.cls (when we load it) say it’s really us:

```

2249 < *ltugproccls>
2250 \def\@tugclass{ltugproc}

```

\if@proc@sober TUG’96 proceedings switched to more sober headings still; so the tug95 option establishes the original state. In the absence of any other guidance, we use the ’96 for TUG’97 proceedings, but also allow numbering of sections.

```

2251 \newif\if@proc@sober
2252 \newif\if@proc@numerable
2253 \DeclareOption{tug95}{%
2254 \@proc@soberfalse
2255 \@proc@numerablefalse
2256 }
2257 \DeclareOption{tug96}{%
2258 \@proc@sobertrue
2259 \@proc@numerablefalse
2260 }

```

```

2261 \DeclareOption{tug97}{%
2262   \@proc@sobertrue
2263   \@proc@numerabletrue
2264 }
2265 \DeclareOption{tug2002}{%
2266   \@proc@sobertrue
2267   \@proc@numerabletrue
2268   \let\if@proc@numbersec\iftrue
2269   \PassOptionsToClass{numbersec}{ltugboat}%
2270 }

```

`\if@proc@numbersec` If we're in a class that allows section numbering (the actual check occurs after `\ProcessOptions`, we can have the following:

```

2271 \DeclareOption{numbersec}{\let\if@proc@numbersec\iftrue
2272   \PassOptionsToClass{numbersec}{ltugboat}%
2273 }
2274 \DeclareOption{nonumber}{\let\if@proc@numbersec\iffalse
2275   \PassOptionsToClass{nonumber}{ltugboat}%
2276 }

```

`\ifTB@title` If we have a paper for which we want to create a detached title, with an editor's note, and then set the paper separately, we use option `notitle`.

```

2277 \newif\ifTB@title
2278 \DeclareOption{title}{\TB@titletrue}
2279 \DeclareOption{notitle}{\TB@titlefalse}
2280 \AtBeginDocument{\stepcounter{page}}

```

There are these people who seem to think `tugproc` is an option as well as a class...

```

2281 \DeclareOption{tugproc}{%
2282   \ClassWarning{@tugclass}{Option \CurrentOption\space ignored}%
2283 }

```

All other options are simply passed to `ltugboat`...

```

2284 \DeclareOption*{\PassOptionsToClass{\CurrentOption}{ltugboat}}

```

If there's a `tugproc` defaults file, input it now: it may tell us which year we're to perform for... (Note: this code *is* millenium-proof. It's not terribly classy for years beyond 2069, but then I'm not going to be around then—this will be an interesting task for a future `TeX`ie...)

```

2285 \InputIfFileExists{@tugclass.cfg}{\ClassInfo{ltugproc}%
2286   {Loading ltugproc.cfg configuration information}}{}
2287 \@ifundefined{TUGprocExtraOptions}%
2288   {\let\TUGprocExtraOptions\empty}%
2289   {\edef\TUGprocExtraOptions{\TUGprocExtraOptions}}

```

`\tugProcYear` Now work out what year it is

```

2290 \@tempcnta\year
2291 \ifnum\@tempcnta<2000
2292   \divide\@tempcnta by100
2293   \multiply\@tempcnta by100
2294   \advance\@tempcnta-\year
2295   \@tempcnta-\@tempcnta
2296 \fi

```

And use that for calculating a year for us to use.

```

2297 \edef\@tempa{\noexpand\providecommand\noexpand\tugProcYear
2298             {\ifnum10>\@tempcnta0\fi\the\@tempcnta}}
2299 \@tempa
2300 \ClassInfo{ltugproc}{Class believes year is
2301 \expandafter\ifnum\tugProcYear<2000 19\fi\tugProcYear
2302 \gobble}

```

Check that this is a “sensible year” (one for which we have a class option defined). If not, make it a ‘suitable’ year, in particular, one that allows numbering sections.

```

2303 \expandafter\ifx\csname ds@tug\tugProcYear\endcsname\relax
2304 \def\tugProcYear{2002}\fi

```

Now execute the default ‘year’ option and get on with processing. Note that this command gets ignored if the configuration file specifies a silly year.

```

2305 \ExecuteOptions{tug\tugProcYear,title\TUGprocExtraOptions}
2306 \ProcessOptions
2307 \if@proc@numbersec
2308 \if@proc@numerable
2309 \else
2310 \ClassWarning{\@tugclass}{This year’s proceedings may not have
2311 numbered sections}%
2312 \fi
2313 \fi

```

Call ltugboat, adding whichever section numbering option is appropriate

```

2314 \LoadClass[\if@proc@numbersec numbersec\else nonumber\fi]{ltugboat}

```

4.1 Proceedings titles

`\maketitle` There’s no provision for ‘section titles’ in proceedings issues, as there are in *TUG-*
`\ifTB@madetitle` *boat* proper. Note the tedious L^AT_EX bug-avoidance in the `\@TB@test@document` macro.

```

2315 \def\maketitle{%
2316 \begingroup

```

first, a bit of flim-flam to generate an initial value for `\rhAuthor` (unless the user’s already given one with a `\shortAuthor` comand).

```

2317 \ifshortAuthor\else
2318 \global\let\rhAuthor\@empty
2319 \def\g@addto@rhAuthor##1{%
2320 \begingroup
2321 \toks@{\expandafter{\rhAuthor}}%
2322 \let\thanks\gobble
2323 \protected@xdef\rhAuthor{\the\toks@##1}%
2324 \endgroup
2325 }%
2326 \@getauthorlist\g@addto@rhAuthor
2327 \fi

```

now, the real business of setting the title

```

2328 \ifTB@title

```

```

2329      \setcounter{footnote}{0}%
2330      \renewcommand{\thefootnote}{\@fnsymbol\c@footnote}%
2331      \if@tubtwocolumn
2332          \twocolumn[\@maketitle]%
2333      \else
2334          \onecolumn
2335          \global\@topnum\z@
2336          \@maketitle
2337      \fi
2338      \@thanks
2339      \thispagestyle{TBproctitle}
2340      \fi
2341  \endgroup
2342  \TB@madetitletrue
2343 }
2344 \newif\ifTB@madetitle \TB@madetitlefalse

\@TB@test@document \@TB@test@document checks to see, at entry to \maketitle, if we've had
\begin{document}. See LATEX bug report latex/2212, submitted by Robin Fair-
bairns, for details.
2345 \def\@TB@test@document{%
2346     \edef\@tempa{\the\everypar}
2347     \def \@tempb{\@nocument}
2348     \ifx \@tempa\@tempb
2349         \@nocument
2350     \fi
2351 }

\AUTHORfont Define the fonts for titles and things
\TITLEfont 2352 \def\AUTHORfont {\large\rmfamily\mdseries\upshape}
\addressfont 2353 \def\TITLEfont {\Large\rmfamily\mdseries\upshape}
\netaddrfont 2354 \def\addressfont{\small\rmfamily\mdseries\upshape}
2355 \def\netaddrfont{\small\ttfamily\mdseries\upshape}

\aboveauthorskip Some changeable skips to permit variability in page layout depending on the par-
\belowauthorskip ticular paper's page breaks.
\belowabstractskip 2356 \newskip\aboveauthorskip \aboveauthorskip=18\p@ \@plus4\p@
2357 \newskip\belowauthorskip \belowauthorskip=\aboveauthorskip
2358 \newskip\belowabstractskip \belowabstractskip=12\p@ \@plus2\p@ \@minus2\p@

\@maketitle The body of \maketitle (for proceedings).
2359 \def\@maketitle{%
2360     {\parskip\z@
2361     \frenchspacing
2362     \TITLEfont\raggedright\noindent\@title\par
2363     \count@=0
2364     \loop
2365     \ifnum\count@<\authornumber
2366         \vskip\aboveauthorskip
2367         \advance\count@\@ne
2368         {\AUTHORfont\theauthor{\number\count@}\endgraf}%
2369         \addressfont\theaddress{\number\count@}\endgraf
2370     }%

```

```

2371         \allowhyphens
2372         \hangindent1.5pc
2373         \netaddrfont\thenetaddress{\number\count@}\endgraf
2374         \hangindent1.5pc
2375         \thePersonalURL{\number\count@}\endgraf
2376     }%
2377     \repeat
2378     \vskip\belowauthorskip}%
2379     \if@abstract
2380         \centerline{\bfseries Abstract}%
2381         \vskip.5\baselineskip\rmfamily
2382         \@tubonecolumnabstractstart
2383             \the\abstract@toks
2384         \@tubonecolumnabstractfinish
2385         \global\@ignoretrue
2386     \fi
2387     \vskip\belowabstractskip
2388     \global\@afterindentfalse\aftergroup\@afterheading
2389 }

```

abstract (*env.*) Save the contents of the abstract environment in the token register `\abstract@toks`.
`\if@abstract` We need to do this, as otherwise it may get ‘typeset’ (previously, it got put in a
`\abstract@toks` box) before `\begin{document}`, and experiments prove that this means our shiny new `\SMC` doesn’t work in this situation.

If you need to understand the ins and outs of this code, look at the place I lifted it from: `tabularx.dtx` (in the tools bundle). The whole thing pivots on having stored the name of the ‘abstract’ environment in `\@abstract@`

```

2390 \newtoks\abstract@toks \abstract@toks{}
2391 \let\if@abstract\iffalse
2392 \def\abstract{%

```

we now warn unsuspecting users who provide an `abstract` environment *after* the `\maketitle` that would typeset it...

```

2393 \ifTB@madetitle
2394     \TBWarning{abstract environment after \string\maketitle}
2395 \fi
2396 \def\@abstract@{abstract}%
2397 \ifx\@currenvir\@abstract@
2398 \else
2399     \TBEError{\string\abstract\space is illegal:%
2400         \MessageBreak
2401         use \string\begin{\@abstract@} instead}%
2402     {\@abstract@\space may only be used as an environment}
2403 \fi
2404 \global\let\if@abstract\iftrue
2405 {\ifnum0='}\fi
2406 \@abstract@getbody}
2407 \let\endabstract\relax

```

`\@abstract@getbody` gets chunks of the body (up to the next occurrence of `\end`) and appends them to `\abstract@toks`. It then uses `\@abstract@findend` to detect whether this `\end` is followed by `{abstract}`

```

2408 \long\def\@abstract@getbody#1\end{%
2409     \global\abstract@toks\expandafter{\the\abstract@toks#1}%

```

2410 \end{@abstract@findend}

Here we've got to \end in the body of the abstract. \end{@abstract@findend} takes the 'argument' of the \end do its argument.

2411 \def\end{@abstract@findend#1}{%

2412 \def\@tempa{#1}%

If we've found an 'end' to match the 'begin' that we started with, we're done with gathering the abstract up; otherwise we stuff the end itself into the token register and carry on.

2413 \ifx\@tempa\@abstract@

2414 \expandafter\@abstract@end

2415 \else

It's not \end{abstract}—check that it's not \end{document} either (which signifies that the author's forgotten about ending the abstract)

2416 \def\@tempb{document}%

2417 \ifx\@tempa\@tempb

2418 \TBError{\string\begin{\@abstract@}

2419 ended by \string\end{\@tempb}}%

2420 {You've forgotten \string\end{\@abstract@}}

2421 \else

2422 \global\abstract@toks\expandafter{\the\abstract@toks\end{#1}}%

2423 \expandafter\expandafter\expandafter\@abstract@getbody

2424 \fi

2425 \fi}

In our case, the action at the 'proper' \end is a lot simpler than what appears in tabularx.dtx ... don't be surprised!

2426 \def\@abstract@end{\ifnum0='{ \fi}%

2427 \expandafter\end\expandafter{\@abstract@}}

\makesignature \makesignature is improper in proceedings, so we replace it with a warning (and a no-op otherwise)

2428 \renewcommand{\makesignature}{\TBWarning

2429 {\string\makesignature\space is invalid in proceedings issues}}

\ps@TBproctitle Now we define the running heads in terms of the \rh* commands.

\ps@TBproc 2430 \def\ps@TBproctitle{\let\@oddhead\MakeRegistrationMarks

\dopagecommands 2431 \let\@evenhead\MakeRegistrationMarks

\setpagecommands 2432 \TB@definefeet

\TB@definefeet 2433 }

\pfoottext 2434 \def\ps@TBproc{%

\rfoottext 2435 \def\@oddhead{\MakeRegistrationMarks

2436 {%

2437 \hfil

2438 \def\{\unskip\ignorespaces}%

2439 \rmfamily\rhTitle

2440 }%

2441 }%

2442 \def\@evenhead{\MakeRegistrationMarks

2443 {%

2444 \def\{\unskip\ignorespaces}%

2445 \rmfamily\rhAuthor

```

2446     \hfil
2447   }%
2448 }%
2449 \TB@definefeet
2450 }
2451
2452 \advance\footskip8\p@    % for deeper running feet
2453
2454 \def\dopagecommands{\csname @pagecommands\number\c@page\endcsname}
2455 \def\setpagecommands#1#2{\expandafter\def\csname @pagecommands#1\endcsname
2456   {#2}}
2457 \def\TB@definefeet{%
2458   \def\@oddfoot{\ifpreprint\pfoottext\hfil\Now\hfil\thepage
2459     \else\rfoottext\hfil\thepage\fi\dopagecommands}%
2460   \def\@evenfoot{\ifpreprint\thepage\hfil\Now\hfil\pfoottext
2461     \else\thepage\hfil\rfoottext\fi\dopagecommands}%
2462 }
2463
2464 \def\pfoottext{{\smc Preprint}:
2465   Proceedings of the \volyr{} Annual Meeting}
2466 \def\rfoottext{\normalfont\TUB, \volx\Dash
2467   {Proceedings of the \volyr{} Annual Meeting}}
2468
2469 \pagestyle{TBproc}

```

4.2 Section divisions

Neither sections nor subsections are numbered by default in the proceedings style: note that this puts a degree of stress on authors' natural tendency to reference sections, which is a matter that needs attention. The class option `NUMBERSEC` once again numbers the sections (and noticeably changes the layout).

```

2470 \if@proc@numbersec
2471 \else
2472   \setcounter{secnumdepth}{0}
2473 \fi

```

Otherwise, the `\section` command is pretty straightforward. However, the `\subsection` and `\subsubsection` are run-in, and we have to remember to have negative stretch (and shrink if we should in future choose to have one) on the `\afterskip` parameter of `\@startsection`, since the whole skip is going to end up getting negated. We use `\TB@startsection` to detect inappropriate forms.

```

2474 \if@proc@numbersec
2475 \else
2476   \if@proc@sober
2477     \def\section
2478       {\TB@nolimelabel
2479         \TB@startsection{{section}%
2480           1%
2481           \z@%
2482           {-8\p@\@plus-2\p@\@minus-2\p@}%
2483           {6\p@}%
2484           {\normalsize\bfseries\raggedright}}}
2485   \else

```

```

2486 \def\section
2487     {\TB@nolimelabel
2488      \TB@startsection{{section}%
2489                       1%
2490                       \z@%
2491                       {-8\p@\@plus-2\p@\@minus-2\p@}%
2492                       {6\p@}%
2493                       {\large\bfseries\raggedright}}}
2494 \fi
2495 \def\subsection
2496     {\TB@nolimelabel
2497      \TB@startsection{{subsection}%
2498                       2%
2499                       \z@%
2500                       {6\p@\@plus 2\p@\@minus2\p@}%
2501                       {-5\p@\@plus -\fontdimen3\the\font}%
2502                       {\normalsize\bfseries}}}
2503 \def\subsubsection
2504     {\TB@nolimelabel
2505      \TB@startsection{{subsubsection}%
2506                       3%
2507                       \parindent%
2508                       \z@%
2509                       {-5\p@\@plus -\fontdimen3\the\font}%
2510                       {\normalsize\bfseries}}}
2511 \fi
2512 </tugproccls>

```

5 Plain T_EX styles

```

2513 <*tugboatsty>
2514 % err...
2515 </tugboatsty>
2516 <*tugprocsty>
2517 % err...
2518 </tugprocsty>

```

6 The L^AT_EX 2_ε compatibility-mode style files

```

2519 <*ltugboatsty>
2520 \obsoletefile{ltugboat.cls}{ltugboat.sty}
2521 \LoadClass{ltugboat}
2522 </ltugboatsty>
2523 <*ltugprocsty>
2524 \obsoletefile{ltugproc.cls}{ltugproc.sty}
2525 \LoadClass{ltugproc}
2526 </ltugprocsty>

```