

The `eventB` package*

Thai Son Hoang
ETH-Zurich
<htson at inf dot ethz dot ch>

February 25, 2013

Abstract

This class provides a template for typesetting Event-B models. It was developed at the Swiss Federal Institute of Technology Zurich (ETH-Zurich).

Contents

1	Introduction	1
2	Usage	1
2.1	Package Options	1
3	Implementation	2
3.1	Package Loading	2
3.2	Declaration of Options for the Package	2
3.3	Commands for Pretty-Print Event-B Models	4

1 Introduction

This package was developed in order to ease the typesetting of Event-B models in \LaTeX .

2 Usage

See `sample-eventB.tex` for an example of how to use the package.

2.1 Package Options

The package offers the following options:

- `nobox`: to disable to bounding boxes for the Event-B modelling elements,
- `small`, `compact`, `tiny`: options for font size,
- `colour` (or `color`): to colour several modelling elements.

*This document corresponds to `eventB` v1.1.1, dated 2012/02/21.

3 Implementation

3.1 Package Loading

We begin by loading the required package `xspace` and `xcolor`.

```
1 \RequirePackage{xspace}
2 \RequirePackage{xcolor}
3 \RequirePackage{ifthen}
```

3.2 Declaration of Options for the Package

In this part various options for the package are defined.

Option for rounding boxes By default, Event-B modelling elements, e.g., invariants, events, etc., are displayed in a rounding box. This option enables them to be displayed without the rounding box.

```
4 \newlength{\B@oldfboxsep}
5 \newcommand{\event}[7] [] {
6   \setlength{\B@oldfboxsep}{\fboxsep}
7   \setlength{\fboxsep}{2ex}
8   \fbox{
9     \ensuremath{
10      \B@event[#1]{#2}{#3}{#4}{#5}{#6}{#7}
11    }
12  }
13  \setlength{\fboxsep}{\B@oldfboxsep}
14 }
15
16 \newcommand{\B@declaration}[2] {
17   \setlength{\B@oldfboxsep}{\fboxsep}
18   \setlength{\fboxsep}{2ex}
19   \fbox{
20     \ensuremath{
21      \B@declarationbase[#1]{#2}
22    }
23  }
24  \setlength{\fboxsep}{\B@oldfboxsep}
25 }
26
27 \newcommand{\B@section}[3] [] {
28   \setlength{\B@oldfboxsep}{\fboxsep}
29   \setlength{\fboxsep}{2ex}
30   \fbox{
31     \ensuremath{
32      \B@sectionbase[#1]{#2}{#3}
33    }
34  }
35  \setlength{\fboxsep}{\B@oldfboxsep}
36 }
37
38 \DeclareOption{nobox}{
39   \renewcommand{\event}[7] [] {
40     \B@event[#1]{#2}{#3}{#4}{#5}{#6}{#7}
```

```

41 }
42
43 \renewcommand{\B@declaration}[2]{
44   \B@declarationbase{#1}{#2}
45 }
46
47 \renewcommand{\B@section}[3][]{
48   \B@sectionbase{#1}{#2}{#3}
49 }
50 }
51

```

Options for font size and spacing We define the default values for font size and some spacing commands, and how they are redefined according to options `small`, `compact`, and `tiny`. In particular, option `compact` and `tiny` implies option `nobox`.

```

52 \newcommand{\B@fontsize}{\normalsize} % The font size used in Bcode environment
53 \newcommand{\Bvspace}[1][2ex]{\[[#1]} % Vertical space
54 \newcommand{\Bhspace}[1][2em]{\hspace{#1}} % Horizontal space
55 \newcommand{\B@tab}{\quad} % A small separation space
56
57 \DeclareOption{small}{
58   \renewcommand{\B@fontsize}{\small}
59   \renewcommand{\Bvspace}[1][1ex]{\[[#1]}
60   \renewcommand{\Bhspace}[1][1em]{\hspace{#1}}
61   \renewcommand{\B@tab}{\ }
62 }
63 \DeclareOption{compact}{
64   \renewcommand{\B@fontsize}{\footnotesize}
65   \renewcommand{\Bvspace}[1][0ex]{\[[#1]}
66   \renewcommand{\Bhspace}[1][0.5em]{\hspace{#1}}
67   \renewcommand{\B@tab}{\ }
68   \ExecuteOptions{nobox}
69 }
70 \DeclareOption{tiny}{
71   \renewcommand{\B@fontsize}{\scriptsize}
72   \renewcommand{\Bvspace}[1][-0.5ex]{\[[#1]}
73   \renewcommand{\Bhspace}[1][0.5em]{\hspace{#1}}
74   \renewcommand{\B@tab}{\ }
75   \ExecuteOptions{nobox}
76 }
77

```

Options for colouring Keywords, labels and identifiers in Event-B can be coloured. We define several commands and redefine them accordingly for colouring. When `colour` (or `color`) option is enabled, one can customise the colours for Event-B keywords, labels or identifier or proof obligation labels.

```

78 \newcommand{\B@keyword}[1]{\ensuremath{\B@keywordbase{#1}}\xspace}
79 \newcommand{\B@identifier}[1]{\ensuremath{\B@identifier{#1}}\xspace}
80 \newcommand{\B@label}[2][]{\ensuremath{\B@label{#1}{#2}}\xspace}
81 \newcommand{\B@po}[1]{\ensuremath{\B@po{#1}}\xspace}
82 \DeclareOption{colour}{

```

```

83 \newcommand{\setBKeywordColour}[1]{\colorlet{B@keywordcolor}{#1}}
84 \setBKeywordColour{blue}
85 \newcommand{\setBIdentifierColour}[1]{\colorlet{B@identifiercolor}{#1}}
86 \setBIdentifierColour{blue!50!red}
87 \newcommand{\setBLabelColour}[1]{\colorlet{B@labelcolor}{#1}}
88 \setBLabelColour{green!50!black}
89 \newcommand{\setBP0Colour}[1]{\colorlet{B@pocolor}{#1}}
90 \setBP0Colour{red}
91 \renewcommand{\B@keyword}[1]{
92   \ensuremath{\textcolor{B@keywordcolor}{\B@keywordbase{#1}}}\xspace
93 }
94 \renewcommand{\B@identifier}[1]{
95   \ensuremath{\textcolor{B@identifiercolor}{\B@identifier{#1}}}\xspace
96 }
97 \renewcommand{\B@label}[2][ ]{
98   \ensuremath{\textcolor{B@labelcolor}{\B@label{#1}{#2}}}\xspace
99 }
100 \renewcommand{\B@po}[1]{
101   \ensuremath{\textcolor{B@pocolor}{\B@po{#1}}}\xspace
102 }
103 }
104 \DeclareOption{color}{
105   \ExecuteOptions{colour}
106 }
107

```

After declaration of options, we execute them accordingly.

```

108 \ProcessOptions

```

3.3 Commands for Pretty-Print Event-B Models

We start with the definition of the `\eventB` macro.

```

109 \newcommand{\eventB}{Event-B\xspace}

```

The `Bcode` environment for displaying Event-B models. The environment has an optional argument for specifying the font size. By default, it is the same as the `\B@fontsize` controlled by the package option.

```

110 \newenvironment{Bcode}[1][\B@fontsize]{\begin{center}#1}{\end{center}}

```

Declarations and Collections Event-B modelling elements are organised into declarations (e.g., variables, constants, etc.) or collections (e.g., invariants, axioms). For each declaration, the input is a comma-separated list of elements. For each collection, the input is a newly(`\`)-separated list of elements.

```

111 \newcommand{\carriersets}[1]{
112   \B@declaration{sets}{#1}
113 }
114 \newcommand{\constants}[1]{
115   \B@declaration{constants}{#1}
116 }
117 \newcommand{\axioms}[2][ ]{
118   \B@section{#1}{axioms}{#2}
119 }

```

```

120 \newcommand{\variables}[1]{
121   \B@declaration{variables}{#1}
122 }

123 \newcommand{\invariants}[2][]{
124   \B@section[#1]{invariants}{#2}
125 }

126 \newcommand{\variant}[1]{
127   \B@declaration{variant}{#1}
128 }

```

Event-B keywords We define the keywords for pretty-print Event-B models.

```

129 \newcommand{\Bany}{\B@keyword{any}}
130 \newcommand{\Bbegin}{\B@keyword{begin}}
131 \newcommand{\Bend}{\B@keyword{end}}
132 \newcommand{\Brefines}{\B@keyword{refines}}
133 \newcommand{\Bstatus}{\B@keyword{status}}
134 \newcommand{\Bthen}{\B@keyword{then}}
135 \newcommand{\Bwhen}{\B@keyword{when}}
136 \newcommand{\Bwhere}{\B@keyword{where}}
137 \newcommand{\Bwith}{\B@keyword{with}}

```

Event-B modelling elements We define several macros for pretty-print Event-B modelling elements.

```

138 \newcommand{\Bctx}[1]{\ensuremath{\mathbf{#1}}\xspace}
139 \newcommand{\Bset}[1]{\Bidentifier{#1}}
140 \newcommand{\Bcst}[1]{\Bidentifier{#1}}
141 \newcommand{\Baxm}[1]{\Blabel{#1}}
142 \newcommand{\Bthm}[1]{\Blabel{thm}{#1}}
143
144 \newcommand{\Bmch}[1]{\ensuremath{\mathbf{#1}}\xspace}
145 \newcommand{\Bvrb}[1]{\Bidentifier{#1}}
146 \newcommand{\Binv}[1]{\Blabel{#1}}
147 \newcommand{\Bevt}[1]{\Blabel{#1}}
148 \newcommand{\Bpar}[1]{\Bidentifier{#1}}
149 \newcommand{\Bact}[1]{\Blabel{#1}}
150 \newcommand{\Bgrd}[1]{\Blabel{#1}}
151 \newcommand{\Bbap}[1]{\hbox{\sl\bfseries #1}}

```

Meta-macros for creating macros for modelling elements We define meta-macros to create macros for different modelling elements.

```

152 \newcommand{\B@newmacro}[3][]{
153   \ifthenelse{\equal{#1}{}}{
154     \expandafter\def\csname #2\endcsname{#3{#2}}
155   }{
156     \expandafter\def\csname #1\endcsname{#3{#2}}
157   }
158 }

159 \newcommand{\newBctx}[2][]{\B@newmacro{#1}{#2}{\Bctx}}
160 \newcommand{\newBset}[2][]{\B@newmacro{#1}{#2}{\Bset}}
161 \newcommand{\newBcst}[2][]{\B@newmacro{#1}{#2}{\Bcst}}

```

```

162 \newcommand{\newBaxm}[2] [] {\B@newmacro[#1]{#2}{\Baxm}}
163 \newcommand{\newBthm}[2] [] {\B@newmacro[#1]{#2}{\Bthm}}
164 \newcommand{\newBmch}[2] [] {\B@newmacro[#1]{#2}{\Bmch}}
165 \newcommand{\newBvrb}[2] [] {\B@newmacro[#1]{#2}{\Bvrb}}
166 \newcommand{\newBinvt}[2] [] {\B@newmacro[#1]{#2}{\Binvt}}
167 \newcommand{\newBevt}[2] [] {\B@newmacro[#1]{#2}{\Bevt}}
168 \newcommand{\newBpar}[2] [] {\B@newmacro[#1]{#2}{\Bpar}}
169 \newcommand{\newBgrd}[2] [] {\B@newmacro[#1]{#2}{\Bgrd}}
170 \newcommand{\newBact}[2] [] {\B@newmacro[#1]{#2}{\Bact}}
171
172 %%%% Theorem Proof Obligation
173 %%%% Print the theorem proof obligation, given the theorem label.
174 %%%% Arguments:
175 %%%% 1. Theorem label
176 %%%%
177 %%%% Usage:
178 %%%% - \thmpo{thm} will produce "thm/THM"
179 \newcommand{\thmpo}[1]{\Bthm[#1]/\Bpo{THM}}
180
181 %%%% Axiom Well-definedness Proof Obligation
182 %%%% Print the axiom well-definedness proof obligation, given the
183 %%%% axiom label.
184 %%%% Arguments:
185 %%%% 1. Axiom label
186 %%%%
187 %%%% Usage:
188 %%%% - \axmwdpo{axm} will produce "axm/WD"
189 \newcommand{\axmwdpo}[1]{\Baxm[#1]/\Bpo{WD}}
190
191 %%%% Invariant Proof Obligation
192 %%%% Print the invariant proof obligation, given the event name and
193 %%%% invariant label
194 %%%% Arguments:
195 %%%% 1. Event name
196 %%%% 2. Invariant label
197 %%%%
198 %%%% Usage:
199 %%%% - \invpo{evt}{inv} will produce "evt/inv/INV"
200 \newcommand{\invpo}[2]{\Bevt[#1]/\Binvt[#2]/\Bpo{INV}}
201
202 %%%% Theorem (in guard) Proof Obligation
203 %%%% Print the simulation proof obligation, given the event name and
204 %%%% the theorem (in guard) label.
205 %%%% Arguments:
206 %%%% 1. Event name
207 %%%% 2. Theorem (in guard) label
208 %%%%
209 %%%% Usage:
210 %%%% - \grdthmpo{evt}{thm} will produce "evt/thm/THM"

```

```

211 \newcommand{\grdthmpo}[2]{\Bevt{#1}/\Bthm{#2}/\Bpo{THM}}
212
213 %%%% Feasibility Proof Obligation
214 %%%% Print the feasibility proof obligation, given the event name and
215 %%%% the action label
216 %%%% Arguments:
217 %%%% 1. Event name
218 %%%% 2. Action label
219 %%%%
220 %%%% Usage:
221 %%%% - \fispo{evt}{act} will produce "evt/act/FIS"
222 \newcommand{\fispo}[2]{\Bevt{#1}/\Bact{#2}/\Bpo{FIS}}
223
224 %%%% Variant finiteness Proof Obligation
225 %%%% Print the Variant finiteness proof obligation
226 %%%% Arguments: No arguments
227 %%%%
228 %%%% Usage:
229 %%%% - \finpo will produce "FIN"
230 \newcommand{\finpo}{\Bpo{FIN}}
231
232 %%%% Variant Proof Obligation
233 %%%% Print the guard strengthen proof obligation, given the event name
234 %%%% Arguments:
235 %%%% 1. Event name
236 %%%%
237 %%%% Usage:
238 %%%% - \grdpo{evt} will produce "evt/VAR"
239 \newcommand{\varpo}[1]{\Bevt{#1}/\Bpo{VAR}}
240
241 %%%% Simulation Proof Obligation
242 %%%% Print the simulation proof obligation, given the event name and
243 %%%% the action label.
244 %%%% Arguments:
245 %%%% 1. Event name
246 %%%% 2. Action label
247 %%%%
248 %%%% Usage:
249 %%%% - \simpo{evt}{act} will produce "evt/act/SIM"
250 \newcommand{\simpo}[2]{\Bevt{#1}/\Bact{#2}/\Bpo{SIM}}
251
252 %%%% Guard Strengthen Proof Obligation
253 %%%% Print the guard strengthen proof obligation, given the event
254 %%%% name and the guard label
255 %%%% Arguments:
256 %%%% 1. (Abstract) Event name
257 %%%% 2. (Abstract) Guard label
258 %%%%
259 %%%% Usage:
260 %%%% - \grdpo{evt}{grd} will produce "evt/grd/GRD"
261 \newcommand{\grdpo}[2]{\Bevt{#1}/\Bgrd{#2}/\Bpo{GRD}}
262

```

```

263 %%%% Variant Natural Number Proof Obligation
264 %%%% Print the Variant Natural Number proof obligation, given the event name
265 %%%% Arguments:
266 %%%% 1. Event name
267 %%%%
268 %%%% Usage:
269 %%%% - \natpo{evt} will produce "evt/NAT"
270 \newcommand{\natpo}[1]{\Bevt{#1}/\Bpo{NAT}}
271
272 \newcommand{\B@keywordbase}[1]{\mathbf{#1}}
273 \newcommand{\B@identifier}[1]{\mathit{#1}}
274 \newcommand{\B@label}[2][]{
275   \def\is@thm{#1}
276   \ifx\is@thm\@empty
277     \mathsf{#2}
278   \else
279     \mathit{#2}
280   \fi
281 }
282
283
284
285 \newcommand{\eventinline}[7][]{
286   \setlength{\B@oldfboxsep}{\fboxsep}
287   \setlength{\fboxsep}{2ex}
288   \fbox{
289     \ensuremath{
290       \B@eventinline[#1]{#2}{#3}{#4}{#5}{#6}{#7}
291     }
292   }
293   \setlength{\fboxsep}{\B@oldfboxsep}
294 }
295
296
297 \newcommand{\B@declarationbase}[2]{
298   \begin{array}{l@{\B@tab}l}
299     \B@keyword{#1:} & #2
300   \end{array}
301 }
302
303 \newcommand{\B@sectionbase}[3][]{
304   \def\no@title{#1}
305   \ifx\no@title\@empty
306     \begin{array}{l}
307       \B@keyword{#2:} \\\
308       \begin{array}{l@{\B@tab}l}
309         #3
310       \end{array}
311     \end{array}
312   \else
313     \begin{array}{l@{\B@tab}l}
314       #3
315     \end{array}
316   \fi

```



```

317 }
318
319 \newcommand{\B@po}[1]{\ensuremath{\mathsf{#1}}\xspace}
320
321 %%%% (BEGIN) Macros for Pretty-Print Event-B Components %%%
322 \newcommand{\SKIP}{\textsc{skip}}
323
324
325 %%%% Pretty print an general Event-B event
326 %%%% Arguments:
327 %%%% 1. (Optional) convergence status.
328 %%%% 2. Name of the event.
329 %%%% 3. Name of the abstract event.
330 %%%% 4. (Comma-separated) list of parameters.
331 %%%% 5. (Newline(\))-separated) list of guards.
332 %%%% 6. (Newline(\))-separated) list of witness predicates.
333 %%%% 7. (Newline(\))-separated) list of assignments.
334 %%%%
335 %%%% Usage: \B@event[conv]{conc}{abs}{x,y}{G1(x,y)\G2(x,y)}{W1\W2}{S1(v,x,y)\S2(w,x,y)}
336 %%%% will produce the following
337 %%%%
338 %%%% conc
339 %%%% refines abs
340 %%%% status conv
341 %%%% any x, y where
342 %%%% G1(x, y)
343 %%%% G2(x, y)
344 %%%% with
345 %%%% W1
346 %%%% W2
347 %%%% then
348 %%%% S1(v, x, y)
349 %%%% S2(w, x, y)
350 %%%% end
351 %%%%
352 %%%% Special case:
353 %%%% - Empty abstract event --> refines clause is omitted.
354 %%%% - Empty convergence status --> status clause is omitted.
355 %%%% - Empty witness --> with clause is omitted.
356 %%%% - Empty parameters, empty guards --> begin ... end
357 %%%% - Empty parameters --> when ... then ... end
358 %%%% - Empty actions --> \SKIP
359 \newcommand{\B@event}[7][]{
360   \def\evt@sts{#1}
361   \def\evt@name{#2}
362   \def\evt@absevt{#3}
363   \def\evt@pars{#4}
364   \def\evt@grds{#5}
365   \def\evt@wits{#6}
366   \def\evt@acts{#7}
367   %% Pretty-print convergence status
368   \ifx\evt@sts\@empty
369     \def\pretty@sts{

```

```

370 \else
371 \def\pretty@sts{\B@tab\Bstatus \B@tab \evt@sts \}
372 \fi
373 % Pretty-print abstract events
374 \ifx\evt@absevt\@empty
375 \def\pretty@absevt{}
376 \else
377 \def\pretty@absevt{\B@tab\Brefines \B@tab \evt@absevt \}
378 \fi
379 % Pretty-print parameters
380 \ifx\evt@pars\@empty
381 \def\pretty@pars{}
382 \else
383 \def\pretty@pars{\B@tab\Bany \B@tab \evt@pars \B@tab \Bwhere \}
384 \fi
385 % Pretty-print guards
386 \ifx\evt@grds\@empty
387 \def\pretty@grds{}
388 \else
389 \def\evt@grds@tmp{
390   \begin{array}{@{\B@tab\B@tab}l@{\B@tab}l}
391     \evt@grds
392   \end{array}\}
393 }
394 \ifx\evt@pars\@empty
395 \def\pretty@grds{
396   \B@tab \Bwhen \}
397   \evt@grds@tmp
398 }
399 \else
400 \def\pretty@grds{\evt@grds@tmp}
401 \fi
402 \fi
403 % Pretty-print witnesses
404 \ifx\evt@wits\@empty
405 \def\pretty@wits{}
406 \else
407 \def\pretty@wits{
408   \B@tab\Bwith\}
409   \begin{array}{@{\B@tab\B@tab}ll}
410     \evt@wits
411   \end{array}\}
412 }
413 \fi
414 % Pretty-print actions
415 \ifx\evt@acts\@empty
416 \def\evt@acts{\SKIP}
417 \else
418 \fi
419 \def\evt@acts@tmp{
420   \begin{array}{@{\B@tab\B@tab}l@{\B@tab}l}
421     \evt@acts
422   \end{array}\}
423 }

```

```

424 \def\evt@acts@keyword{\B@tab\Bthen \}
425 \ifx\evt@pars@empty
426 \ifx\evt@grds@empty
427 \def\evt@acts@keyword{\B@tab\Bbegin \}
428 \else
429 \fi
430 \else
431 \fi
432 \def\pretty@acts{
433   \evt@acts@keyword
434   \evt@acts@tmp
435 }
436 % Really do it now
437 \begin{array}{l}
438   \Bevt{\evt@name} \\\
439   \pretty@sts
440   \pretty@absevt
441   \pretty@pars
442   \pretty@grds
443   \pretty@wits
444   \pretty@acts
445   \B@tab\Bend
446 \end{array}
447 }
448
449 %%%% Pretty print an general Event-B event
450 %%%% Arguments:
451 %%%% 1. (Optional) convergence status.
452 %%%% 2. Name of the event.
453 %%%% 3. Name of the abstract event.
454 %%%% 4. (Comma-separated) list of parameters.
455 %%%% 5. (Newline(\)-separated) list of guards.
456 %%%% 6. (Newline(\)-separated) list of witness predicates.
457 %%%% 7. (Newline(\)-separated) list of assignments.
458 %%%%
459 %%%% Usage: \B@event[conv]{conc}{abs}{x,y}{G1(x,y)\G2(x,y)}{W1\W2}{S1(v,x,y)\S2(w,x,y)}
460 %%%% will produce the following
461 %%%%
462 %%%% conc
463 %%%% refines abs
464 %%%% status conv
465 %%%% any x, y where
466 %%%% G1(x, y)
467 %%%% G2(x, y)
468 %%%% with
469 %%%% W1
470 %%%% W2
471 %%%% then
472 %%%% S1(v, x, y)
473 %%%% S2(w, x, y)
474 %%%% end
475 %%%%
476 %%%% Special case:
477 %%%% - Empty abstract event --> refines clause is omitted.

```

```

478 %%%% - Empty convergence status --> status clause is omitted.
479 %%%% - Empty witness --> with clause is omitted.
480 %%%% - Empty parameters, empty guards --> begin ... end
481 %%%% - Empty parameters --> when ... then ... end
482 %%%% - Empty actions --> \SKIP
483 \newcommand{\B@eventinline}[7][]{
484   \def\evt@sts{#1}
485   \def\evt@name{#2}
486   \def\evt@absevt{#3}
487   \def\evt@pars{#4}
488   \def\evt@grds{#5}
489   \def\evt@wits{#6}
490   \def\evt@acts{#7}
491   %% Ignore convergence status
492   \def\pretty@sts{
493     % Pretty-print abstract events
494     \ifx\evt@absevt\@empty
495       \def\pretty@absevt{
496         \else
497         \def\pretty@absevt{\Brefines~\evt@absevt~}
498         \fi
499         % Pretty-print parameters
500         \ifx\evt@pars\@empty
501           \def\pretty@pars{
502             \else
503             \def\pretty@pars{\Bany~\evt@pars~\Bwhere~}
504             \fi
505             % Pretty-print guards
506             \ifx\evt@grds\@empty
507               \def\pretty@grds{
508                 \else
509                 \def\evt@grds@tmp{
510                   \evt@grds
511                 }
512                 \ifx\evt@pars\@empty
513                   \def\pretty@grds{
514                     \Bwhen~
515                     \evt@grds@tmp~
516                   }
517                 \else
518                 \def\pretty@grds{\evt@grds@tmp~}
519                 \fi
520                 \fi
521                 % Pretty-print witnesses
522                 \ifx\evt@wits\@empty
523                   \def\pretty@wits{
524                     \else
525                     \def\pretty@wits{
526                       \Bwith~
527                       \evt@wits~
528                     }
529                     \fi
530                     % Pretty-print actions
531                     \ifx\evt@acts\@empty

```

```

532 \def\evt@acts{\SKIP}
533 \else
534 \fi
535 \def\evt@acts@tmp{
536   \evt@acts
537 }
538 \def\evt@acts@keyword{\Bthen}
539 \ifx\evt@pars\@empty
540 \ifx\evt@grds\@empty
541 \def\evt@acts@keyword{\Bbegin}
542 \else
543 \fi
544 \else
545 \fi
546 \def\pretty@acts{
547   \evt@acts@keyword~
548   \evt@acts@tmp~
549 }
550 % Really do it now
551 \begin{array}{l}
552   \Bevt{\evt@name}\widehat{=}\sim
553   \pretty@sts
554   \pretty@absevs
555   \pretty@pars
556   \pretty@grds
557   \pretty@wits
558   \pretty@acts
559   \Bend
560 \end{array}
561 }
562
563 %%%% INITIALISATION label
564 \newBevt{init}
565
566 %%%% Pretty print the initialisation: no ‘refines’ clause. no parameters, no
567 %%%% guards
568 %%%% Arguments:
569 %%%% 1. (Newline(\))-separated) list of assignments.
570 %%%%
571 %%%% Usage: \init{S1(v,x,y)\S2(w,x,y)}
572 %%%% will produce the following
573 %%%%
574 %%%% init
575 %%%% begin
576 %%%% S1(v, x, y)
577 %%%% S2(w, x, y)
578 %%%% end
579 %%%%
580 \newcommand{\initialisation}[1]{
581   \event{\init}{-}{-}{-}{-}{#1}
582 }

```

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

Symbols	\B@tab 55, 61, 67, 74, 298, 308, 313, 371, 377, 383, 390, 396, 408, 409, 420, 424, 427, 445	C
\@empty 276, 305, 368, 374, 380, 386, 394, 404, 415, 425, 426, 494, 500, 506, 512, 522, 531, 539, 540	\Bact . 149, 170, 222, 250	\carriersets 111
\@ 53, 59, 65, 72, 307, 331–333, 335, 371, 377, 383, 392, 396, 408, 411, 422, 424, 427, 438, 455– 457, 459, 569, 571	\Bany 129, 383, 503	\colorlet 83, 85, 87, 89
	\Baxm 141, 162, 189	\constants 114
	\Bbap 151	\csname 154, 156
	\Bbegin . . . 130, 427, 541	D
	\Bcst 140, 161	\DeclareOption . 38, 57, 63, 70, 82, 104
	\Bctx 138, 159	\def 154, 156, 275, 304, 360–366, 369, 371, 375, 377, 381, 383, 387, 389, 395, 400, 405, 407, 416, 419, 424, 427, 432, 484–490, 492, 495, 497, 501, 503, 507, 509, 513, 518, 523, 525, 532, 535, 538, 541, 546
	\begin 110, 298, 306, 308, 313, 390, 409, 420, 437, 551	
_ 61, 67, 74	\Bend 131, 445, 559	
A	\Bevt 147, 167, 200, 211, 222, 239, 250, 261, 270, 438, 552	E
\axioms 117	\bfseries 151	\else . 278, 312, 370, 376, 382, 388, 399, 406, 417, 428, 430, 496, 502, 508, 517, 524, 533, 542, 544
\axmwdpo 188, 189	\Bgrd 150, 169, 261	\end . . 110, 300, 310, 311, 315, 392, 411, 422, 446, 560
B	\Bhspace . 54, 60, 66, 73	\endcsname . . . 154, 156
\B@declaration 16, 43, 112, 115, 121, 127	\Bidentifier . 79, 94, 139, 140, 145, 148	\ensuremath 9, 20, 31, 78–81, 92, 95, 98, 101, 138, 144, 289, 319
\B@declarationbase 21, 44, 297	\Bin 146, 166, 200	\equal 153
\B@event 10, 40, 335, 359, 459	\Blabel 80, 97, 141, 142, 146, 147, 149, 150	\event 5, 39, 581
\B@eventinline 290, 483	\Bmch 144, 164	\eventB 109
\B@fontsize 52, 58, 64, 71, 110	\Bpar 148, 168	\eventinline 285
\B@identifier 79, 95, 273	\Bpo 81, 100, 179, 189, 200, 211, 222, 230, 239, 250, 261, 270	\evt@absevt s 362, 374, 377, 486, 494, 497
\B@keyword . . 78, 91, 129–137, 299, 307	\Brefines . 132, 377, 497	\evt@acts 366, 415, 416, 421, 490, 531, 532, 536
\B@keywordbase 78, 92, 272	\Bset 139, 160	
\B@label . . . 80, 98, 274	\Bstatus 133, 371	
\B@newmacro 152, 159–170	\Bthen 134, 424, 538	
\B@oldfboxsep 4, 6, 13, 17, 24, 28, 35, 286, 293	\Bthm . 142, 163, 179, 211	
\B@po 81, 101, 319	\Bvrb 145, 165	
\B@section 27, 47, 118, 124	\Bvspace . 53, 59, 65, 72	
\B@sectionbase 32, 48, 303	\Bwhen 135, 396, 514	
	\Bwhere . . . 136, 383, 503	
	\Bwith 137, 408, 526	

<code>\evt@acts@keyword</code> .	500, 506, 512,	<code>\pretty@pars</code> 381, 383,
.... 424, 427,	522, 531, 539, 540	441, 501, 503, 555
433, 538, 541, 547	<code>\init</code> 571, 581	<code>\pretty@sts</code> ... 369,
<code>\evt@acts@tmp</code>	<code>\initialisation</code> ... 580	371, 439, 492, 553
. 419, 434, 535, 548	<code>\invariants</code> 123	<code>\pretty@wits</code> 405, 407,
<code>\evt@grds</code> 364,	<code>\invpo</code> 199, 200	443, 523, 525, 557
386, 391, 426,	<code>\is@thm</code> 275, 276	<code>\ProcessOptions</code> ... 108
488, 506, 510, 540		
<code>\evt@grds@tmp</code>	M	Q
.... 389, 397,	<code>\mathbf</code> ... 138, 144, 272	<code>\quad</code> 55
400, 509, 515, 518	<code>\mathit</code> 273, 279	
<code>\evt@name</code>	<code>\mathsf</code> 277, 319	
. 361, 438, 485, 552	N	R
<code>\evt@pars</code>	<code>\natpo</code> 269, 270	<code>\renewcommand</code>
. 363, 380, 383,	<code>\newBact</code> 170	39, 43, 47, 58–
394, 425, 487,	<code>\newBaxm</code> 162	61, 64–67, 71–
500, 503, 512, 539	<code>\newBcst</code> 161	74, 91, 94, 97, 100
<code>\evt@sts</code>	<code>\newBctx</code> 159	<code>\RequirePackage</code> ... 1–3
. 360, 368, 371, 484	<code>\newBevt</code> 167, 564	
<code>\evt@wits</code> . 365, 404,	<code>\newBgrd</code> 169	S
410, 489, 522, 527	<code>\newBinv</code> 166	<code>\scriptsize</code> 71
<code>\ExecuteOptions</code> ...	<code>\newBmch</code> 164	<code>\setBIdentifierColour</code>
..... 68, 75, 105	<code>\newBpar</code> 168 85, 86
<code>\expandafter</code> .. 154, 156	<code>\newBset</code> 160	<code>\setBKeywordColour</code> .
	<code>\newBthm</code> 163 83, 84
F	<code>\newBvrb</code> 165	<code>\setBLabelColour</code> 87, 88
<code>\fbox</code> 8, 19, 30, 288	<code>\newcommand</code> 5,	<code>\setBPOColour</code> ... 89, 90
<code>\fboxsep</code> ... 6, 7, 13,	16, 27, 52–55,	<code>\setlength</code> . 6, 7, 13,
17, 18, 24, 28,	78–81, 83, 85,	17, 18, 24, 28,
29, 35, 286, 287, 293	87, 89, 109, 111,	29, 35, 286, 287, 293
<code>\fi</code> .. 280, 316, 372,	114, 117, 120,	<code>\simpo</code> 249, 250
378, 384, 401,	123, 126, 129–	<code>\SKIP</code> 322,
402, 413, 418,	142, 144–152,	358, 416, 482, 532
429, 431, 498,	159–170, 179,	<code>\sl</code> 151
504, 519, 520,	189, 200, 211,	<code>\small</code> 58
529, 534, 543, 545	222, 230, 239,	
<code>\finpo</code> 229, 230	250, 261, 270,	T
<code>\fispo</code> 221, 222	272–274, 285,	<code>\textcolor</code> 92, 95, 98, 101
<code>\footnotesize</code> 64	297, 303, 319,	<code>\textsc</code> 322
	322, 359, 483, 580	<code>\thmpo</code> 178, 179
G	<code>\newenvironment</code> ... 110	
<code>\grdpo</code> 238, 260, 261	<code>\newlength</code> 4	V
<code>\grdthmpo</code> 210, 211	<code>\no@title</code> 304, 305	<code>\variables</code> 120
	<code>\normalsize</code> 52	<code>\variant</code> 126
H	P	<code>\varpo</code> 239
<code>\hbox</code> 151	<code>\pretty@absepts</code> ...	W
<code>\hspace</code> .. 54, 60, 66, 73 375, 377,	<code>\widehat</code> 552
	440, 495, 497, 554	
I	<code>\pretty@acts</code>	X
<code>\ifthenelse</code> 153	. 432, 444, 546, 558	<code>\xspace</code> 78–81,
<code>\ifx</code> .. 276, 305, 368,	<code>\pretty@grds</code> .. 387,	92, 95, 98, 101,
374, 380, 386,	395, 400, 442,	109, 138, 144, 319
394, 404, 415,	507, 513, 518, 556	
425, 426, 494,		

Change History

v1.0		v1.1	
General: Initial version	1	General: Re-implement how options are defined, added options 'box'	1
v1.0.1		v1.1.1	
General: Ensure that the keywords, labels are in math-mode	1	General: Updated documentation . .	1