

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   \ifthenelse{\equal{#1}{}}{
276     \mathsf{#2}
277   }{
278     \mathit{#2}
279   }
280 }
281
282
283
284 \newcommand{\eventinline}[7][]{
285   \setlength{\B@oldfboxsep}{\fboxsep}
286   \setlength{\fboxsep}{2ex}
287   \fbox{
288     \ensuremath{
289       \B@eventinline[#1]{#2}{#3}{#4}{#5}{#6}{#7}
290     }
291   }
292   \setlength{\fboxsep}{\B@oldfboxsep}
293 }
294
295
296 \newcommand{\B@declarationbase}[2]{
297   \begin{array}{l@{\B@tab}l}
298     \B@keyword{#1:} & #2
299   \end{array}
300 }
301
302 \newcommand{\B@sectionbase}[3][]{
303   \ifthenelse{\equal{#1}{}}{
304     \begin{array}{l}
305       \B@keyword{#2:} \\
306       \begin{array}{l@{\B@tab}l}
307         #3
308       \end{array}
309     \end{array}
310   }{
311     \begin{array}{l@{\B@tab}l}
312       #3
313     \end{array}
314   }
315 }
316

```



```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

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, 297, 306, 311, 369, 375, 381, 388, 394, 406, 407, 418, 422, 425, 443	C
\@empty 366, 372, 378, 384, 392, 402, 413, 423, 424, 492, 498, 504, 510, 520, 529, 537, 538	\Bact . 149, 170, 222, 250	\carriersets 111
\ 53, 59, 65, 72, 305, 329–331, 333, 369, 375, 381, 390, 394, 406, 409, 420, 422, 425, 436, 453– 455, 457, 567, 569	\Bany 129, 381, 501	\colorlet 83, 85, 87, 89
	\Baxm 141, 162, 189	\constants 114
	\Bbap 151	\csname 154, 156
	\Bbegin . . . 130, 425, 539	D
	\Bcst 140, 161	\DeclareOption . 38, 57, 63, 70, 82, 104
	\Bctx 138, 159	\def 154, 156, 358–364, 367, 369, 373, 375, 379, 381, 385, 387, 393, 398, 403, 405, 414, 417, 422, 425, 430, 482–488, 490, 493, 495, 499, 501, 505, 507, 511, 516, 521, 523, 530, 533, 536, 539, 544
	\begin 110, 297, 304, 306, 311, 388, 407, 418, 435, 549	
\sqcup 61, 67, 74	\Bend 131, 443, 557	
A	\Bevt 147, 167, 200, 211, 222, 239, 250, 261, 270, 436, 550	E
\axioms 117	\bfseries 151	\else 368, 374, 380, 386, 397, 404, 415, 426, 428, 494, 500, 506, 515, 522, 531, 540, 542
\axmwdpo 188, 189	\Bgrd 150, 169, 261	\end . . 110, 299, 308, 309, 313, 390, 409, 420, 444, 558
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, 288, 317
\B@declarationbase 21, 44, 296	\Bin 146, 166, 200	\equal . . . 153, 275, 303
\B@event 10, 40, 333, 357, 457	\Blabel 80, 97, 141, 142, 146, 147, 149, 150	\event 5, 39, 579
\B@eventinline 289, 481	\Bmch 144, 164	\eventB 109
\B@fontsize 52, 58, 64, 71, 110	\Bpar 148, 168	\eventinline 284
\B@identifier 79, 95, 273	\Bpo 81, 100, 179, 189, 200, 211, 222, 230, 239, 250, 261, 270	\evt@absevt 360, 372, 375, 484, 492, 495
\B@keyword . . 78, 91, 129–137, 298, 305	\Brefines . 132, 375, 495	\evt@acts . . . 364, 413, 414, 419, 488, 529, 530, 534
\B@keywordbase 78, 92, 272	\Bset 139, 160	
\B@label . . . 80, 98, 274	\Bstatus 133, 369	
\B@newmacro 152, 159–170	\Bthen 134, 422, 536	
\B@oldfboxsep 4, 6, 13, 17, 24, 28, 35, 285, 292	\Bthm . 142, 163, 179, 211	
\B@po 81, 101, 317	\Bvrb 145, 165	
\B@section 27, 47, 118, 124	\Bvspace . 53, 59, 65, 72	
\B@sectionbase 32, 48, 302	\Bwhen 135, 394, 512	
	\Bwhere . . . 136, 381, 501	
	\Bwith 137, 406, 524	

<code>\evt@acts@keyword</code> .	498, 504, 510,	<code>\pretty@pars</code> 379, 381,
.... 422, 425,	520, 529, 537, 538	439, 499, 501, 553
431, 536, 539, 545	<code>\init</code> 569, 579	<code>\pretty@sts</code> ... 367,
<code>\evt@acts@tmp</code>	<code>\initialisation</code> ... 578	369, 437, 490, 551
. 417, 432, 533, 546	<code>\invariants</code> 123	<code>\pretty@wits</code> 403, 405,
<code>\evt@grds</code> 362,	<code>\invpo</code> 199, 200	441, 521, 523, 555
384, 389, 424,		<code>\ProcessOptions</code> ... 108
486, 504, 508, 538		
<code>\evt@grds@tmp</code>	M	Q
.... 387, 395,	<code>\mathbf</code> ... 138, 144, 272	<code>\quad</code> 55
398, 507, 513, 516	<code>\mathit</code> 273, 278	
<code>\evt@name</code>	<code>\mathsf</code> 276, 317	
. 359, 436, 483, 550		R
<code>\evt@pars</code>	N	<code>\renewcommand</code>
. 361, 378, 381,	<code>\natpo</code> 269, 270	39, 43, 47, 58–
392, 423, 485,	<code>\newBact</code> 170	61, 64–67, 71–
498, 501, 510, 537	<code>\newBaxm</code> 162	74, 91, 94, 97, 100
<code>\evt@sts</code>	<code>\newBcst</code> 161	<code>\RequirePackage</code> ... 1–3
. 358, 366, 369, 482	<code>\newBctx</code> 159	
<code>\evt@wits</code> . 363, 402,	<code>\newBevt</code> 167, 562	S
408, 487, 520, 525	<code>\newBgrd</code> 169	<code>\scriptsize</code> 71
<code>\ExecuteOptions</code> ...	<code>\newBin</code> 166	<code>\setBIdentifierColour</code>
..... 68, 75, 105	<code>\newBmch</code> 164 85, 86
<code>\expandafter</code> .. 154, 156	<code>\newBpar</code> 168	<code>\setBKeywordColour</code> .
	<code>\newBset</code> 160 83, 84
	<code>\newBthm</code> 163	<code>\setBLabelColour</code> 87, 88
	<code>\newBvrb</code> 165	<code>\setBPOColour</code> ... 89, 90
F	<code>\newcommand</code> 5,	<code>\setlength</code> . 6, 7, 13,
<code>\fbox</code> 8, 19, 30, 287	16, 27, 52–55,	17, 18, 24, 28,
<code>\fboxsep</code> ... 6, 7, 13,	78–81, 83, 85,	29, 35, 285, 286, 292
17, 18, 24, 28,	87, 89, 109, 111,	<code>\simpo</code> 249, 250
29, 35, 285, 286, 292	114, 117, 120,	<code>\SKIP</code> 320,
<code>\fi</code> 370, 376, 382, 399,	123, 126, 129–	356, 414, 480, 530
400, 411, 416,	142, 144–152,	<code>\sl</code> 151
427, 429, 496,	159–170, 179,	<code>\small</code> 58
502, 517, 518,	189, 200, 211,	
527, 532, 541, 543	222, 230, 239,	T
<code>\finpo</code> 229, 230	250, 261, 270,	<code>\textcolor</code> 92, 95, 98, 101
<code>\fispo</code> 221, 222	272–274, 284,	<code>\textsc</code> 320
<code>\footnotesize</code> 64	296, 302, 317,	<code>\thmpo</code> 178, 179
	320, 357, 481, 578	
G	<code>\newenvironment</code> ... 110	V
<code>\grdpo</code> 238, 260, 261	<code>\newlength</code> 4	<code>\variables</code> 120
<code>\grdthmpo</code> 210, 211	<code>\normalsize</code> 52	<code>\variant</code> 126
		<code>\varpo</code> 239
H	P	W
<code>\hbox</code> 151	<code>\pretty@absevt</code> ...	<code>\widehat</code> 550
<code>\hspace</code> .. 54, 60, 66, 73 373, 375,	
	438, 493, 495, 552	X
I	<code>\pretty@acts</code>	<code>\xspace</code> 78–81,
<code>\ifthenelse</code> 153, 275, 303	. 430, 442, 544, 556	92, 95, 98, 101,
<code>\ifx</code> 366, 372, 378, 384,	<code>\pretty@grds</code> .. 385,	109, 138, 144, 317
392, 402, 413,	393, 398, 440,	
423, 424, 492,	505, 511, 516, 554	

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