

The `eventB` package*

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

February 26, 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}
4 \RequirePackage{etoolbox}
```

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.

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

```

41 \B@event[#1]{#2}{#3}{#4}{#5}{#6}{#7}
42 }
43
44 \renewcommand{\B@declaration}[2]{
45 \B@declarationbase{#1}{#2}
46 }
47
48 \renewcommand{\B@section}[3][ ]{
49 \B@sectionbase{#1}{#2}{#3}
50 }
51 }
52

```

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`.

```

53 \newcommand{\B@fontsize}{\normalsize} % The font size used in Bcode environment
54 \newcommand{\Bvspace}[1][2ex]{\[[#1]} % Vertical space
55 \newcommand{\Bhspace}[1][2em]{\hspace{#1}} % Horizontal space
56 \newcommand{\B@tab}{\quad} % A small separation space
57
58 \DeclareOption{small}{
59 \renewcommand{\B@fontsize}{\small}
60 \renewcommand{\Bvspace}[1][1ex]{\[[#1]}
61 \renewcommand{\Bhspace}[1][1em]{\hspace{#1}}
62 \renewcommand{\B@tab}{\ }
63 }
64 \DeclareOption{compact}{
65 \renewcommand{\B@fontsize}{\footnotesize}
66 \renewcommand{\Bvspace}[1][0ex]{\[[#1]}
67 \renewcommand{\Bhspace}[1][0.5em]{\hspace{#1}}
68 \renewcommand{\B@tab}{\ }
69 \ExecuteOptions{nobox}
70 }
71 \DeclareOption{tiny}{
72 \renewcommand{\B@fontsize}{\scriptsize}
73 \renewcommand{\Bvspace}[1][-0.5ex]{\[[#1]}
74 \renewcommand{\Bhspace}[1][0.5em]{\hspace{#1}}
75 \renewcommand{\B@tab}{\ }
76 \ExecuteOptions{nobox}
77 }
78

```

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.

```

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

```

```

83 \DeclareOption{colour}{
84   \newcommand{\setBKeywordColour}[1]{\colorlet{B@keywordcolor}{#1}}
85   \setBKeywordColour{blue}
86   \newcommand{\setBIdentifierColour}[1]{\colorlet{B@identifiercolor}{#1}}
87   \setBIdentifierColour{blue!50!red}
88   \newcommand{\setBLabelColour}[1]{\colorlet{B@labelcolor}{#1}}
89   \setBLabelColour{green!50!black}
90   \newcommand{\setBP0Colour}[1]{\colorlet{B@pocolor}{#1}}
91   \setBP0Colour{red}
92   \renewcommand{\B@keyword}[1]{
93     \ensuremath{\textcolor{B@keywordcolor}{\B@keywordbase{#1}}}\xspace
94   }
95   \renewcommand{\B@identifier}[1]{
96     \ensuremath{\textcolor{B@identifiercolor}{\B@identifier{#1}}}\xspace
97   }
98   \renewcommand{\B@label}[2][]{
99     \ensuremath{\textcolor{B@labelcolor}{\B@label{#1}{#2}}}\xspace
100  }
101  \renewcommand{\B@po}[1]{
102    \ensuremath{\textcolor{B@pocolor}{\B@po{#1}}}\xspace
103  }
104 }
105 \DeclareOption{color}{
106   \ExecuteOptions{colour}
107 }
108

```

After declaration of options, we execute them accordingly.

```

109 \ProcessOptions

```

3.3 Commands for Pretty-Print Event-B Models

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

```

110 \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.

```

111 \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.

```

112 \newcommand{\carriersets}[1]{
113   \B@declaration{sets}{#1}
114 }
115 \newcommand{\constants}[1]{
116   \B@declaration{constants}{#1}
117 }
118 \newcommand{\axioms}[2][]{
119   \B@section[#1]{axioms}{#2}
120 }

```

```

121 \newcommand{\variables}[1]{
122   \B@declaration{variables}{#1}
123 }

124 \newcommand{\invariants}[2][ ]{
125   \B@section[#1]{invariants}{#2}
126 }

127 \newcommand{\variant}[1]{
128   \B@declaration{variant}{#1}
129 }

```

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

```

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

```

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

```

139 \newcommand{\Bctx}[1]{\ensuremath{\mathbf{#1}}\xspace}
140 \newcommand{\Bset}[1]{\Bidentifier{#1}}
141 \newcommand{\Bcst}[1]{\Bidentifier{#1}}
142 \newcommand{\Baxm}[1]{\Blabel{#1}}
143 \newcommand{\Bthm}[1]{\Blabel{thm}{#1}}
144
145 \newcommand{\Bmch}[1]{\ensuremath{\mathbf{#1}}\xspace}
146 \newcommand{\Bvrb}[1]{\Bidentifier{#1}}
147 \newcommand{\Binv}[1]{\Blabel{#1}}
148 \newcommand{\Bevt}[1]{\Blabel{#1}}
149 \newcommand{\Bpar}[1]{\Bidentifier{#1}}
150 \newcommand{\Bact}[1]{\Blabel{#1}}
151 \newcommand{\Bgrd}[1]{\Blabel{#1}}
152 \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.

```

153 \newcommand{\B@newmacro}[3][ ]{
154   \ifstrequal{#1}{\B@newmacro}{#2}{\B@newmacro}{#3}{#2}}
155   \expandafter\def\csname #2\endcsname{#3{#2}}
156 }{
157   \expandafter\def\csname #1\endcsname{#3{#2}}
158 }
159 }

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

```

```

163 \newcommand{\newBaxm}[2] [] {\B@newmacro[#1]{#2}{\Baxm}}
164 \newcommand{\newBthm}[2] [] {\B@newmacro[#1]{#2}{\Bthm}}
165 \newcommand{\newBmch}[2] [] {\B@newmacro[#1]{#2}{\Bmch}}
166 \newcommand{\newBvrb}[2] [] {\B@newmacro[#1]{#2}{\Bvrb}}
167 \newcommand{\newBinvt}[2] [] {\B@newmacro[#1]{#2}{\Binvt}}
168 \newcommand{\newBevt}[2] [] {\B@newmacro[#1]{#2}{\Bevt}}
169 \newcommand{\newBpar}[2] [] {\B@newmacro[#1]{#2}{\Bpar}}
170 \newcommand{\newBgrd}[2] [] {\B@newmacro[#1]{#2}{\Bgrd}}
171 \newcommand{\newBact}[2] [] {\B@newmacro[#1]{#2}{\Bact}}
172
173 %%%% Theorem Proof Obligation
174 %%%% Print the theorem proof obligation, given the theorem label.
175 %%%% Arguments:
176 %%%% 1. Theorem label
177 %%%%
178 %%%% Usage:
179 %%%% - \thmpo{thm} will produce "thm/THM"
180 \newcommand{\thmpo}[1]{\Bthm[#1]/\Bpo{THM}}
181
182 %%%% Axiom Well-definedness Proof Obligation
183 %%%% Print the axiom well-definedness proof obligation, given the
184 %%%% axiom label.
185 %%%% Arguments:
186 %%%% 1. Axiom label
187 %%%%
188 %%%% Usage:
189 %%%% - \axmwdpo{axm} will produce "axm/WD"
190 \newcommand{\axmwdpo}[1]{\Baxm[#1]/\Bpo{WD}}
191
192 %%%% Invariant Proof Obligation
193 %%%% Print the invariant proof obligation, given the event name and
194 %%%% invariant label
195 %%%% Arguments:
196 %%%% 1. Event name
197 %%%% 2. Invariant label
198 %%%%
199 %%%% Usage:
200 %%%% - \invpo{evt}{inv} will produce "evt/inv/INV"
201 \newcommand{\invpo}[2]{\Bevt[#1]/\Binvt[#2]/\Bpo{INV}}
202
203 %%%% Theorem (in guard) Proof Obligation
204 %%%% Print the simulation proof obligation, given the event name and
205 %%%% the theorem (in guard) label.
206 %%%% Arguments:
207 %%%% 1. Event name
208 %%%% 2. Theorem (in guard) label
209 %%%%
210 %%%% Usage:
211 %%%% - \grdthmpo{evt}{thm} will produce "evt/thm/THM"

```

```

212 \newcommand{\grdthmpo}[2]{\Bevt{#1}/\Bthm{#2}/\Bpo{THM}}
213
214 %%%% Feasibility Proof Obligation
215 %%%% Print the feasibility proof obligation, given the event name and
216 %%%% the action label
217 %%%% Arguments:
218 %%%% 1. Event name
219 %%%% 2. Action label
220 %%%%
221 %%%% Usage:
222 %%%% - \fispo{evt}{act} will produce "evt/act/FIS"
223 \newcommand{\fispo}[2]{\Bevt{#1}/\Bact{#2}/\Bpo{FIS}}
224
225 %%%% Variant finiteness Proof Obligation
226 %%%% Print the Variant finiteness proof obligation
227 %%%% Arguments: No arguments
228 %%%%
229 %%%% Usage:
230 %%%% - \finpo will produce "FIN"
231 \newcommand{\finpo}{\Bpo{FIN}}
232
233 %%%% Variant Proof Obligation
234 %%%% Print the guard strengthen proof obligation, given the event name
235 %%%% Arguments:
236 %%%% 1. Event name
237 %%%%
238 %%%% Usage:
239 %%%% - \grdpo{evt} will produce "evt/VAR"
240 \newcommand{\varpo}[1]{\Bevt{#1}/\Bpo{VAR}}
241
242 %%%% Simulation Proof Obligation
243 %%%% Print the simulation proof obligation, given the event name and
244 %%%% the action label.
245 %%%% Arguments:
246 %%%% 1. Event name
247 %%%% 2. Action label
248 %%%%
249 %%%% Usage:
250 %%%% - \simpo{evt}{act} will produce "evt/act/SIM"
251 \newcommand{\simpo}[2]{\Bevt{#1}/\Bact{#2}/\Bpo{SIM}}
252
253 %%%% Guard Strengthen Proof Obligation
254 %%%% Print the guard strengthen proof obligation, given the event
255 %%%% name and the guard label
256 %%%% Arguments:
257 %%%% 1. (Abstract) Event name
258 %%%% 2. (Abstract) Guard label
259 %%%%
260 %%%% Usage:
261 %%%% - \grdpo{evt}{grd} will produce "evt/grd/GRD"
262 \newcommand{\grdpo}[2]{\Bevt{#1}/\Bgrd{#2}/\Bpo{GRD}}
263

```

```

264 %%%% Variant Natural Number Proof Obligation
265 %%%% Print the Variant Natural Number proof obligation, given the event name
266 %%%% Arguments:
267 %%%% 1. Event name
268 %%%%
269 %%%% Usage:
270 %%%% - \natpo{evt} will produce "evt/NAT"
271 \newcommand{\natpo}[1]{\Bevt{#1}/\Bpo{NAT}}
272
273 \newcommand{\B@keywordbase}[1]{\mathbf{#1}}
274 \newcommand{\B@identifier}[1]{\mathit{#1}}
275 \newcommand{\B@label}[2][]{
276   \ifstrequal{#1}{}{
277     \mathsf{#2}
278   }{
279     \mathit{#2}
280   }
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   \ifstrequal{#1}{}{
305     \begin{array}{l}
306       \B@keyword{#2:} \\
307       \begin{array}{l@{\B@tab}l}
308         #3
309       \end{array}
310     \end{array}
311   }{
312     \begin{array}{l@{\B@tab}l}
313       #3
314     \end{array}
315   }
316 }
317

```



```

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

```

```

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

```

```

425 \B@ifstrequal{\evt@pars}{}{
426 \B@ifstrequal{\evt@grds}{}{
427 \renewcommand\pretty@acts@keyword{\B@tab\Bbegin \\\
428 }{}
429 }{}
430 \newcommand\pretty@acts{
431 \pretty@acts@keyword
432 \pretty@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
448 %%%% Pretty print an general Event-B event
449 %%%% Arguments:
450 %%%% 1. (Optional) convergence status.
451 %%%% 2. Name of the event.
452 %%%% 3. Name of the abstract event.
453 %%%% 4. (Comma-separated) list of parameters.
454 %%%% 5. (Newline(\)-separated) list of guards.
455 %%%% 6. (Newline(\)-separated) list of witness predicates.
456 %%%% 7. (Newline(\)-separated) list of assignments.
457 %%%%
458 %%%% Usage: \B@event[conv]{conc}{abs}{x,y}{G1(x,y)\G2(x,y)}{W1\W2}{S1(v,x,y)\S2(w,x,y)}
459 %%%% will produce the following
460 %%%%
461 %%%% conc
462 %%%% refines abs
463 %%%% status conv
464 %%%% any x, y where
465 %%%% G1(x, y)
466 %%%% G2(x, y)
467 %%%% with
468 %%%% W1
469 %%%% W2
470 %%%% then
471 %%%% S1(v, x, y)
472 %%%% S2(w, x, y)
473 %%%% end
474 %%%%
475 %%%% Special case:
476 %%%% - Empty abstract event --> refines clause is omitted.
477 %%%% - Empty convergence status --> status clause is omitted.
478 %%%% - Empty witness --> with clause is omitted.

```

```

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

```

```

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

```

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 56, 62, 68, 75, 298, 307, 312, 372, 378, 384, 391, 397, 409, 410, 420, 424, 427, 443	C
\@empty 493, 499, 505, 511, 521, 530, 538, 539	\Bact . 150, 171, 223, 251	\carriersets 112
\ 54, 60, 66, 73, 306, 330–332, 334, 372, 378, 384, 393, 397, 409, 412, 422, 424, 427, 436, 454– 456, 458, 568, 570	\Bany 130, 384, 502	\colorlet 84, 86, 88, 90
_ 62, 68, 75	\Baxm 142, 163, 190	\constants 115
A	\Bbap 152	\csname 155, 157
\axioms 118	\Bbegin . . . 131, 427, 540	D
\axmwdpo 189, 190	\Bcst 141, 162	\DeclareOption . 39, 58, 64, 71, 83, 105
B	\Bctx 139, 160	\def 155, 157, 483–489, 491, 494, 496, 500, 502, 506, 508, 512, 517, 522, 524, 531, 534, 537, 540, 545
\B@declaration 17, 44, 113, 116, 122, 128	\begin 111, 298, 305, 307, 312, 391, 410, 420, 435, 550	E
\B@declarationbase 22, 45, 297	\Bend 132, 443, 558	\else 495, 501, 507, 516, 523, 532, 541, 543
\B@event 11, 41, 334, 359, 458	\Bevt 148, 168, 201, 212, 223, 240, 251, 262, 271, 436, 551	\end . . 111, 300, 309, 310, 314, 393, 412, 422, 444, 559
\B@eventinline 290, 482	\bfseries 152	\endcsname . . . 155, 157
\B@fontsize 53, 59, 65, 72, 111	\Bgrd 151, 170, 262	\ensuremath . . . 10, 21, 32, 79–82, 93, 96, 99, 102, 139, 145, 289, 318
\B@identifier 80, 96, 274	\Bhspace . 55, 61, 67, 74	\event 6, 40, 580
\B@ifstrequal 358, 369, 375, 381, 387, 395, 405, 416, 425, 426	\Bidentifier . 80, 95, 140, 141, 146, 149	\eventB 110
\B@keyword . . 79, 92, 130–138, 299, 306	\Bin 147, 167, 201	\eventinline 285
\B@keywordbase 79, 93, 273	\Blabel 81, 98, 142, 143, 147, 148, 150, 151	\evt@absevt 363, 375, 378, 485, 493, 496
\B@label . . . 81, 99, 275	\Bmch 145, 165	\evt@acts 367, 416, 417, 421, 489, 530, 531, 535
\B@newmacro 153, 160–171	\Bpar 149, 169	\evt@acts@keyword 537, 540, 546
\B@oldfboxsep 5, 7, 14, 18, 25, 29, 36, 286, 293	\Bpo 82, 101, 180, 190, 201, 212, 223, 231, 240, 251, 262, 271	\evt@acts@tmp . 534, 547
\B@po 82, 102, 318	\Brefines . 133, 378, 496	\evt@grds 365, 387, 392, 426, 487, 505, 509, 539
\B@section 28, 48, 119, 125	\Bset 140, 161	\evt@grds@tmp 508, 514, 517
\B@sectionbase 33, 49, 303	\Bstatus 134, 372	\evt@name 362, 436, 484, 551
	\Bthen 135, 424, 537	\evt@pars 364, 381, 384,
	\Bthm . 143, 164, 180, 212	
	\Bvrb 146, 166	
	\Bvspace . 54, 60, 66, 73	
	\Bwhen 136, 397, 513	
	\Bwhere . . . 137, 384, 502	
	\Bwith 138, 409, 525	

395, 425, 486, 499, 502, 511, 538	\newBctx 160	\pretty@sts . . . 370, 372, 437, 491, 552
\evt@sts	\newBevt 168, 563	\pretty@wits 406, 408, 441, 522, 524, 556
. 361, 369, 372, 483	\newBgrd 170	\ProcessOptions . . . 109
\evt@wits . 366, 405, 411, 488, 521, 526	\newBinu 167	
\ExecuteOptions . . .	\newBmch 165	Q
. 69, 76, 106	\newBpar 169	\quad 56
\expandafter	\newBset 161	
. . . . 155, 157, 358	\newBthm 164	R
	\newBvrb 166	\renewcommand 40, 44, 48, 59–62, 65– 68, 72–75, 92, 95, 98, 101, 417, 427
	\newcommand 6, 17, 28, 53–56, 79–82, 84, 86, 88, 90, 110, 112, 115, 118, 121, 124, 127, 130– 143, 145–153, 160–171, 180, 190, 201, 212, 223, 231, 240, 251, 262, 271, 273–275, 285, 297, 303, 318, 321, 358, 359, 361–367, 370, 372, 376, 378, 382, 384, 388, 390, 396, 401, 406, 408, 419, 424, 430, 482, 579	\RequirePackage . . . 1–4
F	\newenvironment . . . 111	S
\fbox 9, 20, 31, 288	\newlength 5	\scriptsize 72
\fboxsep . . . 7, 8, 14, 18, 19, 25, 29, 30, 36, 286, 287, 293	\normalsize 53	\setBIdentifierColour 86, 87
\fi 497, 503, 518, 519, 528, 533, 542, 544		\setBKeywordColour 84, 85
\finpo 230, 231		\setBLabelColour 88, 89
\fispo 222, 223		\setBPOColour . . . 90, 91
\footnotesize 65		\setlength . 7, 8, 14, 18, 19, 25, 29, 30, 36, 286, 287, 293
		\simpo 250, 251
G		\SKIP 321, 357, 417, 481, 531
\grdpo 239, 261, 262		\sl 152
\grdthmpo 211, 212		\small 59
H	P	T
\hbox 152	\pretty@absevt 376, 378, 438, 494, 496, 553	\textcolor 93, 96, 99, 102
\hspace . . . 55, 61, 67, 74	\pretty@acts 430, 442, 545, 557	\textsc 321
	\pretty@acts@keyword 424, 427, 431	\thmpo 179, 180
I	\pretty@acts@tmp 419, 432	
\ifstrequal 154, 276, 304, 358	\pretty@grds . . 388, 396, 401, 440, 506, 512, 517, 555	V
\ifx 493, 499, 505, 511, 521, 530, 538, 539	\pretty@grds@tmp 390, 398, 401	\variables 121
\init 570, 580	\pretty@pars 382, 384, 439, 500, 502, 554	\variant 127
\initialisation . . . 579		\varpo 240
\invariants 124		
\invpo 200, 201		W
		\widehat 551
M		
\mathbf . . . 139, 145, 273		X
\mathit 274, 279		\xspace 79–82, 93, 96, 99, 102, 110, 139, 145, 318
\mathsf 277, 318		
N		
\natpo 270, 271		
\newBact 171		
\newBaxm 163		
\newBcst 162		

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