

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 % \ifx\evt@acts\empty
417 % \def\evt@acts{\SKIP}
418 % \else
419 % \fi
420 % \def\evt@acts@tmp{
421 %   \begin{array}{@{\B@tab\B@tab}l@{\B@tab}l}
422 %     \evt@acts
423 %   \end{array}\\
424 % }

```

```

425 % \def\evt@acts@keyword{\B@tab\Bthen \\\}
426 % \ifx\evt@pars\@empty
427 % \ifx\evt@grds\@empty
428 % \def\evt@acts@keyword{\B@tab\Bbegin \\\}
429 % \else
430 % \fi
431 % \else
432 % \fi
433 % \def\pretty@acts{
434 %   \evt@acts@keyword
435 %   \evt@acts@tmp
436 % }
437 % Really do it now
438 \begin{array}{l}
439   \Bevt{\evt@name} \\\
440   \pretty@sts
441   \pretty@absevt
442   \pretty@pars
443   \pretty@grds
444   \pretty@wits
445   % \pretty@acts
446   \B@tab\Bend
447 \end{array}
448 }
449 }
450
451 %%%% Pretty print an general Event-B event
452 %%%% Arguments:
453 %%%% 1. (Optional) convergence status.
454 %%%% 2. Name of the event.
455 %%%% 3. Name of the abstract event.
456 %%%% 4. (Comma-separated) list of parameters.
457 %%%% 5. (Newline(\)-separated) list of guards.
458 %%%% 6. (Newline(\)-separated) list of witness predicates.
459 %%%% 7. (Newline(\)-separated) list of assignments.
460 %%%%
461 %%%% Usage: \B@event[conv]{conc}{abs}{x,y}{G1(x,y)\G2(x,y)}{W1\W2}{S1(v,x,y)\S2(w,x,y)}
462 %%%% will produce the following
463 %%%%
464 %%%% conc
465 %%%% refines abs
466 %%%% status conv
467 %%%% any x, y where
468 %%%% G1(x, y)
469 %%%% G2(x, y)
470 %%%% with
471 %%%% W1
472 %%%% W2
473 %%%% then
474 %%%% S1(v, x, y)
475 %%%% S2(w, x, y)
476 %%%% end
477 %%%%
478 %%%% Special case:

```

```

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

```

```

533 \ifx\evt@acts\@empty
534 \def\evt@acts{\SKIP}
535 \else
536 \fi
537 \def\evt@acts@tmp{
538   \evt@acts
539 }
540 \def\evt@acts@keyword{\Bthen}
541 \ifx\evt@pars\@empty
542 \ifx\evt@grds\@empty
543 \def\evt@acts@keyword{\Bbegin}
544 \else
545 \fi
546 \else
547 \fi
548 \def\pretty@acts{
549   \evt@acts@keyword~
550   \evt@acts@tmp~
551 }
552 % Really do it now
553 \begin{array}{l}
554   \Bevt{\evt@name}\widehat{=}\sim
555   \pretty@sts
556   \pretty@absepts
557   \pretty@pars
558   \pretty@grds
559   \pretty@wits
560   \pretty@acts
561   \Bend
562 \end{array}
563 }
564
565 %%%% INITIALISATION label
566 \newBevt{init}
567
568 %%%% Pretty print the initialisation: no ‘refines’ clause. no parameters, no
569 %%%% guards
570 %%%% Arguments:
571 %%%% 1. (Newline(\))-separated) list of assignments.
572 %%%%
573 %%%% Usage: \init{S1(v,x,y)\S2(w,x,y)}
574 %%%% will produce the following
575 %%%%
576 %%%% init
577 %%%% begin
578 %%%% S1( v , x , y )
579 %%%% S2( w , x , y )
580 %%%% end
581 %%%%
582 \newcommand{\initialisation}[1]{
583   \event{\init}{-}{-}{-}{-}{#1}
584 }

```

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		
\@empty 416, 426, 427, 496, 502, 508, 514, 524, 533, 541, 542	\B@tab 56, 62, 68, 75, 298, 307, 312, 372, 378, 384, 391, 397, 409, 410, 421, 425, 428, 446	C \carriersets 112 \colorlet 84, 86, 88, 90 \constants 115 \csname 155, 157
\ 54, 60, 66, 73, 306, 330–332, 334, 372, 378, 384, 393, 397, 409, 412, 423, 425, 428, 439, 457– 459, 461, 571, 573	\Bact . 150, 171, 223, 251 \Bany 130, 384, 505 \Baxm 142, 163, 190 \Bbap 152 \Bbegin . . . 131, 428, 543 \Bcst 141, 162 \Bctx 139, 160 \begin 111, 298, 305, 307, 312, 391, 410, 421, 438, 553 \Bend 132, 446, 561 \Bevt 148, 168, 201, 212, 223, 240, 251, 262, 271, 439, 554 \bfseries 152 \Bgrd 151, 170, 262 \Bhspace . 55, 61, 67, 74 \Bidentifier . 80, 95, 140, 141, 146, 149 \Binv 147, 167, 201 \Blabel 81, 98, 142, 143, 147, 148, 150, 151 \Bmch 145, 165 \Bpar 149, 169 \Bpo 82, 101, 180, 190, 201, 212, 223, 231, 240, 251, 262, 271 \Brefines . 133, 378, 499 \Bset 140, 161 \Bstatus 134, 372 \Bthen 135, 425, 540 \Bthm . 143, 164, 180, 212 \Bvrb 146, 166 \Bvspace . 54, 60, 66, 73 \Bwhen 136, 397, 516 \Bwhere . . . 137, 384, 505 \Bwith 138, 409, 528	D \DeclareOption . 39, 58, 64, 71, 83, 105 \def . . 155, 157, 417, 420, 425, 428, 433, 486–492, 494, 497, 499, 503, 505, 509, 511, 515, 520, 525, 527, 534, 537, 540, 543, 548
A \axioms 118 \axmwdpo 189, 190		E \else 418, 429, 431, 498, 504, 510, 519, 526, 535, 544, 546 \end . . 111, 300, 309, 310, 314, 393, 412, 423, 447, 562 \endcsname . . . 155, 157 \ensuremath . . . 10, 21, 32, 79–82, 93, 96, 99, 102, 139, 145, 289, 318 \event 6, 40, 583 \eventB 110 \eventinline 285 \evt@absevt 363, 375, 378, 488, 496, 499 \evt@acts 367, 416, 417, 422, 492, 533, 534, 538 \evt@acts@keyword 425, 428, 434, 540, 543, 549 \evt@acts@tmp 420, 435, 537, 550 \evt@grds 365, 387, 392, 427, 490, 508, 512, 542
B \B@declaration 17, 44, 113, 116, 122, 128 \B@declarationbase 22, 45, 297 \B@event 11, 41, 334, 359, 461 \B@eventinline 290, 485 \B@fontsize 53, 59, 65, 72, 111 \B@identifier 80, 96, 274 \B@ifstrequal 358, 369, 375, 381, 387, 395, 405 \B@keyword . . 79, 92, 130–138, 299, 306 \B@keywordbase 79, 93, 273 \B@label . . . 81, 99, 275 \B@newmacro 153, 160–171 \B@oldfboxsep 5, 7, 14, 18, 25, 29, 36, 286, 293 \B@po 82, 102, 318 \B@section 28, 48, 119, 125 \B@sectionbase 33, 49, 303		

<code>\evt@grds@tmp</code>		
. 511, 517, 520		
<code>\evt@name</code>		
. 362, 439, 487, 554		
<code>\evt@pars</code>		
. 364, 381, 384,		
395, 426, 489,		
502, 505, 514, 541		
<code>\evt@sts</code>		
. 361, 369, 372, 486		
<code>\evt@wits</code>		
. 366, 405,		
411, 491, 524, 529		
<code>\ExecuteOptions</code> . . .		
. 69, 76, 106		
<code>\expandafter</code>		
. 155, 157, 358		
	F	
<code>\fbox</code> 9, 20, 31, 288		
<code>\fboxsep</code> 7, 8, 14,		
18, 19, 25, 29,		
30, 36, 286, 287, 293		
<code>\fi</code> 419, 430, 432, 500,		
506, 521, 522,		
531, 536, 545, 547		
<code>\finpo</code> 230, 231		
<code>\fispo</code> 222, 223		
<code>\footnotesize</code> 65		
	G	
<code>\grdpo</code> 239, 261, 262		
<code>\grdthmpo</code> 211, 212		
	H	
<code>\hbox</code> 152		
<code>\hspace</code> 55, 61, 67, 74		
	I	
<code>\ifstrequal</code>		
. 154, 276, 304, 358		
<code>\ifx</code> 416, 426, 427, 496,		
502, 508, 514,		
524, 533, 541, 542		
<code>\init</code> 573, 583		
<code>\initialisation</code> 582		
<code>\invariants</code> 124		
<code>\invpo</code> 200, 201		
	M	
<code>\mathbf</code> 139, 145, 273		
<code>\mathit</code> 274, 279		
<code>\mathsf</code> 277, 318		
	N	
<code>\natpo</code> 270, 271		
<code>\newBact</code> 171		
<code>\newBaxm</code> 163		
<code>\newBcst</code> 162		
<code>\newBctx</code> 160		
<code>\newBevt</code> 168, 566		
<code>\newBgrd</code> 170		
<code>\newBinv</code> 167		
<code>\newBmch</code> 165		
<code>\newBpar</code> 169		
<code>\newBset</code> 161		
<code>\newBthm</code> 164		
<code>\newBvrb</code> 166		
<code>\newcommand</code> 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, 485, 582		
<code>\newenvironment</code> 111		
<code>\newlength</code> 5		
<code>\normalsize</code> 53		
	P	
<code>\pretty@absevt</code> . . .		
. 376, 378,		
441, 497, 499, 556		
<code>\pretty@acts</code>		
. 433, 445, 548, 560		
<code>\pretty@grds</code> 388,		
396, 401, 443,		
509, 515, 520, 558		
<code>\pretty@grds@tmp</code> . .		
. 390, 398, 401		
<code>\pretty@pars</code> 382, 384,		
442, 503, 505, 557		
<code>\pretty@sts</code> 370,		
372, 440, 494, 555		
<code>\pretty@wits</code> 406, 408,		
444, 525, 527, 559		
<code>\ProcessOptions</code> 109		
	Q	
<code>\quad</code> 56		
	R	
<code>\renewcommand</code>		
40, 44, 48, 59–		
62, 65–68, 72–		
75, 92, 95, 98, 101		
<code>\RequirePackage</code> 1–4		
	S	
<code>\scriptsize</code> 72		
<code>\setBIdentifierColour</code>		
. 86, 87		
<code>\setBKeywordColour</code> .		
. 84, 85		
<code>\setBLabelColour</code> 88, 89		
<code>\setBPOColour</code> 90, 91		
<code>\setlength</code> 7, 8, 14,		
18, 19, 25, 29,		
30, 36, 286, 287, 293		
<code>\simpo</code> 250, 251		
<code>\SKIP</code> 321,		
357, 417, 484, 534		
<code>\sl</code> 152		
<code>\small</code> 59		
	T	
<code>\textcolor</code> 93, 96, 99, 102		
<code>\textsc</code> 321		
<code>\thmpo</code> 179, 180		
	V	
<code>\variables</code> 121		
<code>\variant</code> 127		
<code>\varpo</code> 240		
	W	
<code>\widehat</code> 554		
	X	
<code>\xspace</code> 79–82,		
93, 96, 99, 102,		
110, 139, 145, 318		

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