The Tikz BPMN package

Sander J.J. Leemans

January 7, 2025

This manual describes the BPMN Tikz package, used to draw BPMN models.

1 Usage

Place the file `tikzlibrarybpmn.code.tex` in the same folder as your .tex file, and include `\usetikzlibrary{bpmn}` in the preamble of your .text file.

1.1 Gateways

Symbol	code
	`\node [exclusive gateway] {};` `\node [xor gateway] {};`
	`\node [parallel gateway] {};` `\node [and gateway] {};`
	`\node [inclusive gateway] {};` `\node [or gateway] {};`
	`\node [eventbased gateway] {};`

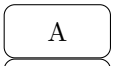
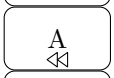
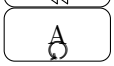
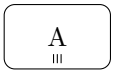
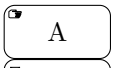
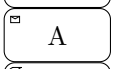
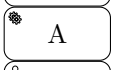
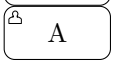
1.2 Events

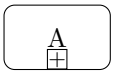
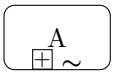
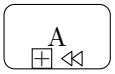
Symbol	code
	`\node [start event] {};`
	`\node [end event] {};`
	`\node [message start event] {};`
	`\node [catching message intermediate event] {};`
	`\node [throwing message intermediate event] {};`
	`\node [message end event] {};`
	`\node [timer start event] {};`
	`\node [timer event] {};` `\node [intermediate timer event] {};`
	`\node [throwing compensation intermediate event] {};`
	`\node [throwing compensation end event] {};`
	`\node [error event] {};` `\node [throwing error end event] {};`
	`\node [signal start event] {};`
	`\node [catching signal intermediate event] {};` `\node [signal intermediate event] {};`
	`\node [throwing signal intermediate event] {};`
	`\node [signal end event] {};`

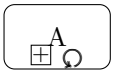
1.2.1 Background colour

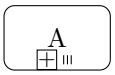
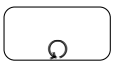
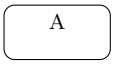
The background colour is an optional argument for several event styles, and required to be set if the background is not white for correct rendering. The default can be set using `\colorlet{defaultbackgroundcolour}{new default colour}` after loading the package.

1.3 Tasks & sub-processes

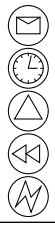
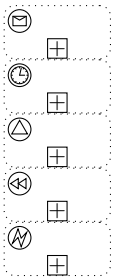
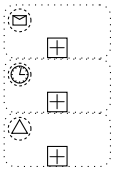
Symbol	code
	`\node [task] {A};`
	`\node [compensation task] {A};`
	`\node [loop task] {A};` requires the `amssymb` and `amsmath` packages
	`\node [multiinstance task] {A};`
	`\node [manual task] {A};`
	`\node [receive task] {A};`
	`\node [script task] {A};`
	`\node [send task] {A};`
	`\node [service task] {A};`
	`\node [user task] {A};`

Symbol	code
	`\node [subprocess] {A};` `\node [collapsed subprocess] {A};`
	`\node [adhoc subprocess] {A};` `\node [collapsed adhoc subprocess] {A};`
	`\node [compensation subprocess] {A};` `\node [collapsed compensation subprocess] {A};`

	`\node [loop subprocess] {A};`
	`\node [collapsed loop subprocess] {A};`
	requires the `amssymb` and `amsmath` packages

	`\node [multiinstance subprocess] {A};` `\node [collapsed multiinstance subprocess] {A};`
	`\node [expanded subprocess] {};` use in combination with the `fit` key
	`\node [expanded adhoc subprocess] {};` use in combination with the `fit` key
	`\node [expanded compensation subprocess] {};` use in combination with the `fit` key
	`\node [expanded loop subprocess] {};` use in combination with the `fit` key
	`\node [expanded multiinstance subprocess] {};` use in combination with the `fit` key
	`\node [subprocess label=A] {};` use in combination with the `fit` key

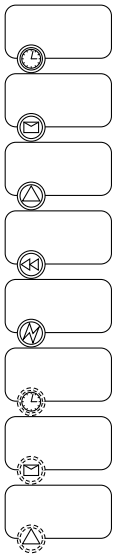
1.4 Event sub-processes

Symbol	code
	`\node [message subprocess event] {};`
	`\node [timer subprocess event] {};`
	`\node [signal subprocess event] {};`
	`\node [compensation subprocess event] {};`
	`\node [error subprocess event] {};`
	`\node [message noninterrupting subprocess event] {};`
	`\node [timer noninterrupting subprocess event] {};`
	`\node [signal noninterrupting subprocess event] {};`
	`\node [message event subprocess] {};`
	`\node [timer event subprocess] {};`
	`\node [signal event subprocess] {};`
	`\node [compensation event subprocess] {};`
	`\node [error event subprocess] {};`
	`\node [message noninterrupting event subprocess] {};`
	`\node [timer noninterrupting event subprocess] {};`
	`\node [signal noninterrupting event subprocess] {};`

1.5 Boundary events

In the following, the number 225 controls the position of the boundary event (in degrees). Naturally, a label can be used on any node. To connect associations to a label, use the style `name=`.

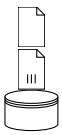
Symbol	code

	`\node [task, label={[timer boundary event]225:}] {};`
	`\node [task, label={[message boundary event]225:}] {};`
	`\node [task, label={[signal boundary event]225:}] {};`
	`\node [task, label={[compensation boundary event]225:}] {};`
	`\node [task, label={[error boundary event]225:}] {};`
	`\node [task, label={[timer noninterrupting boundary event]225:}] {};`
	`\node [task, label={[message noninterrupting boundary event]225:}] {};`
	`\node [task, label={[signal noninterrupting boundary event]225:}] {};`

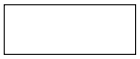
All boundary events use fill to draw over the border of the parent node, and accept an argument for the fill colour. The default fill colour can be changed using `\colorlet{defaultbackgroundcolour}{new default colour}` after loading the package.

1.6 Data

Symbol	code

	`\node [data object] {};`
	`\node [data collection] {};`
	`\node [data store] {};`

1.7 Resources

Symbol	code
	`\node [pool] {};` use in combination with the `fit` key
	`\node [pool label=A] {};` use in combination with the `fit` key