

Package ‘petrinetR’

August 18, 2016

Type Package

Title Building, Visualizing, Exporting and Replaying Petri Nets

Version 0.1.0

Date 2016-08-18

Author Gert Janssenswillen

Maintainer Gert Janssenswillen <gert.janssenswillen@uhasselt.be>

Description Functions for the construction of Petri Nets. Petri Nets can be replayed by firing enabled transitions.
Silent transitions will be hidden by the execution handler. Also includes functionalities for the visualization of Petri Nets and export of Petri Nets to PNML (Petri Net Markup Language) files.

License GPL-3

Depends R(>= 3.0.0)

LazyData true

Imports dplyr, visNetwork

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-08-18 13:38:52

R topics documented:

create_PN	2
enabled	2
enabled_transition	3
execute	3
flows	4
marking	4
parse	4
parsel	5
petrinetR	5

places	5
post_set	6
pre_set	6
transitions	6
tree_to_PN	7
visNetwork_from_PN	7

Index	8
--------------	----------

create_PN	<i>Create Petri Net</i>
-----------	-------------------------

Description

Function to create a petri net by specifying a set of places, transitions, flows and a marking.

Usage

```
create_PN(places, transitions, flows, marking)
```

Arguments

places	A vector of unique places.
transitions	A vector of unique transitions.
flows	A data.frame of flows, with columns named "from" and "to".
marking	The names of the places to be marked.

Examples

```
create_PN("place_1",
"transition_1",
data.frame(from = "place_1",to = "transition_1"),
marking = "place_1")
```

enabled	<i>Enabled transitions</i>
---------	----------------------------

Description

List the enabled transitions in a marked Petri Net. Silent transitions, i.e. starting with "inv_" or "tau" are assumed to be able to fire silently, thereby possible enabling other transitions.

Usage

```
enabled(PN)
```

Arguments

PN A Petri Net

enabled_transition *Enabled Transition*

Description

Check if a transition is currently enabled

Usage

enabled_transition(PN, transition)

Arguments

PN A Petri Net
transition A Transition

execute *Execute*

Description

Executes (fire) an enabled transition and returns the Petri Net with the New marking. If the transition is enabled via the firing of silent transition (i.e. starting with "inv_" or "tau"), it will fire these first. If the transition is not enabled, it will return FALSE.

Usage

execute(PN, transition)

Arguments

PN A Petri Net
transition The transition to be fired

flows	<i>Flows</i>
-------	--------------

Description

Extracts the flows from a Petri Net

Usage

flows(PN)

Arguments

PN	A Petri Net
----	-------------

marking	<i>Marking</i>
---------	----------------

Description

Get the current marking of a Petri Net

Usage

marking(PN)

Arguments

PN	A Petri Net
----	-------------

parse	<i>Parse</i>
-------	--------------

Description

Parses a sequence of transitions. If possible returns the Petri Net with the updated marking. Otherwise returns FALSE

Usage

parse(PN, trace)

Arguments

PN	A Petri Net
trace	A sequence of transitions, stored in a vector.

parse1	<i>Parse (logical)</i>
--------	------------------------

Description

Tests whether a sequence of transitions can be fired by a Petri Net. If so returns TRUE, otherwise FALSE.

Usage

parse1(PN, trace)

Arguments

PN	A Petri Net
trace	A sequence of transitions, stored in a vector.

petrinetR	<i>petrinetR - Building, visualizing, exporting and replaying Petri Nets</i>
-----------	--

Description

Functions for the construction of Petri Nets. Petri Nets can be replayed by firing enabled transitions. Silent transitions will be hidden by the execution handler. Also includes functionalities for the visualization of Petri Nets and export of Petri Nets to PNML-files.

places	<i>Places</i>
--------	---------------

Description

Extracts the places from a Petri Net

Usage

places(PN)

Arguments

PN	A Petri Net
----	-------------

post_set	<i>Postset</i>
----------	----------------

Description

Get the postset of a transition or place in a Petri Net

Usage

```
post_set(PN, node)
```

Arguments

PN	A Petri Net
node	A place or transition in the petri net

pre_set	<i>Preset</i>
---------	---------------

Description

Get the preset of a transition or place in a Petri Net

Usage

```
pre_set(PN, node)
```

Arguments

PN	A Petri Net
node	A place or transition in the petri net

transitions	<i>Transitions</i>
-------------	--------------------

Description

Extracts the transitions from a Petri Net

Usage

```
transitions(PN)
```

Arguments

PN	A Petri Net
----	-------------

tree_to_PN	<i>tree_to_PN</i>
------------	-------------------

Description

Create of petri net from a process tree.

Usage

```
tree_to_PN(tree, name)
```

Arguments

tree	The process tree to be converted
name	A name, which will be used to indicated the start and end transitions of the petri net.

visNetwork_from_PN	<i>VisNetwork from PN</i>
--------------------	---------------------------

Description

Visualize a Petri Net with an interactive network

Usage

```
visNetwork_from_PN(PN)
```

Arguments

PN	Petri Net to visualize
----	------------------------

Index

[create_PN](#), [2](#)

[enabled](#), [2](#)

[enabled_transition](#), [3](#)

[execute](#), [3](#)

[flows](#), [4](#)

[marking](#), [4](#)

[parse](#), [4](#)

[parsel](#), [5](#)

[petrinetR](#), [5](#)

[petrinetR-package \(petrinetR\)](#), [5](#)

[places](#), [5](#)

[post_set](#), [6](#)

[pre_set](#), [6](#)

[transitions](#), [6](#)

[tree_to_PN](#), [7](#)

[visNetwork_from_PN](#), [7](#)