# Colored Petrinets Simulator Gui Documentation

Release alpha 2

**Christoph Kuhr** 

### CONTENTS

1	gui	3
2	model	11
3	inout	23
4	Indices and tables	25
Pτ	thon Module Index	27

Contents:

CONTENTS 1

2 CONTENTS

### **CHAPTER**

### ONE

### **GUI**

### class gui.MainWindow.MainWindow(workingDir)

Bases: PyQt4.QtGui.QMainWindow

The MainWindow is loaded by the application main loop.

Member workingDir Working directory.

**Member editors** List of all editors: gui.DiagramEditor.

Member editorwidgets List of all editor widgets.

Member logWidget Log widget.

**Member tabWidget** Widget containing *editors[0]*, and the *logWidget*.

Member simulator model. CPNS imulator instance.

Member timer Simulation timer.

#### addSubnetEditor (subnet)

Create a subnet editor.

@param subnet: The string name of the substitution transition, name of the subnet

### closeEvent (event)

Close window event.

**Parameters** event – *QtGui.closeEvent*.

createItemsAssignToEditor(editor, subnet, tmpSubnets, importSubnet=False)

Assign places, transition and connections to subnet editors

### **Parameters**

- editor gui.DiagramEditor subnet.
- **subnet** Subnet string name.
- tmpSubnets List of subnet names and corresponding items ["subnet",[items]].
- **importSubnet** Flag determining, whether new net is loaded or a subnet is imported.

**Return subnetConnections** List of connection in subnet for later inter subnet processing.

**createSubnetItemLists** (*tmpSubnets*, *serPlaces*, *serTransitions*, *serConnections*)
Assign subnet to items.

### **Parameters**

- tmpSubnets List of subnet names
- **serPlaces** List with parameters for the Place generation: [[uniqueName, name, portClone, port, [pos], [initMarking]],...].
- **serTransitions** List with parameters for the Transition generation: [[unique-Name, name, [pos], guardExpression, subnet],...].

• **serConnections** – List with parameters for the Connection generation: [[unique-NameSRC, uniqueNameDST, name, sourceConnector, destinationConnector],...].

**Return newTmpSubnets** List of subnet names and corresponding items ["subnet",[items]].

### export\_Step\_as\_SVG()

Export Petrinet and subnets to seperate SVG files

### importSubnet (transition)

Import subnet into editor.

**Parameters** transition – Substitution transition *model.TransitionItem*.

#### initNetFromFile (editor, subnetList)

Create all items for a subnet.

#### **Parameters**

- editor gui.DiagramEditor subnet.
- **subnetList** "subnet",[items].

**Return places, transitions, connectionsInSubnet** List of visual places, List of visual Transitions, list of visualk connections.

loadNet (editor=None, filename=None, subnet=None, importSubnet=False)

Load a net with subnets or import subnets from file.

#### **Parameters**

- editor gui.DiagramEditor to fill with the subnet. editors[0] if editor=none
- **filename** The filename that is used when a new net is loaded.
- **subnet** Subnet string name.
- **importSubnet** Flag determining, whether new net is loaded or a subnet is imported.

### lookupSubstitutionConnectors (subnetConnections)

Lookup and substitute connectors.

**Parameters subnetConnections** – List of connection in subnet for inter subnet processing.

### newNet (init=False)

Create a new Petrinet.

This method resets the main editor widget. It deletes the previously loaded Petrinet, as well as any hierarchical subnet.

@param init: Load on startup?

### openSubnet (transition)

Open subnet editor.

 $\textbf{Parameters transition} - Substitution \ transition \ \textit{model.} Transition Item.$ 

### saveNet()

Save Petrinet with respect to subnets to XML file.

setArc (editor, srcConnector, dstConnector, itemName)

Create arcs based on new uniqueNames.

### **Parameters**

- editor gui.DiagramEditor subnet.
- srcConnector Source model. AbstractItem. Connector.
- $\bullet \ \, \textbf{dstConnector} Destination \ \textit{model.} AbstractItem. Connector.$
- itemName Annotation text.

4 Chapter 1. gui

**Return newArc** Newly created *model.ArcItem*.

```
gui.MainWindow.bin_(QTextStream) \rightarrow QTextStream gui.MainWindow.hex_(QTextStream) \rightarrow QTextStream gui.MainWindow.oct_(QTextStream) \rightarrow QTextStream
```

Bases: PyQt4.QtGui.QWidget

Editor widget, containing the libraryModelView, the colorSetListView and the gui.DiagramScene.

Member mainWindow gui.MainWindow. Main application window.

Member parent Parent widget.

Member workingDir Working directory.

Member subnet Name of visualized subnet, also window title.

Member colorListItems Defined colors.

Member libItems Visual library items, CPN elements.

**Member portConnections** List of connections between portClone port places and substitution transitions.

Member visualPlaces List of all model.PlaceItem's shown in this 'gui.DiagramScene.

**Member visualTransitions** List of all model.TransitionItem's shown in this 'gui.DiagramScene.

Member visualConnectionList List of all model. ArcItem's shown in this 'gui. DiagramScene.

Member libraryModel Library model for visual CPN elements.

Member mouseScreenPos Recorded mouse position.

Member diagramScene Contained gui.DiagramScene.

Member diagramView gui.DiagramView controlling gui.DiagramScene.

Member startedArc Status, determining whether the creation of a connection has started.

**Member showCanvasInfos** Status variable for activation of the tooltip *model.AbstractItem.DescriptionCanvas* s.

### createIcon (nodeType)

Create Icons for libraryModel.

**Parameters** nodeType – Type of icon to create.

### defineNewToken()

Define a new color set.

deleteArc (editor, connection)

Delete arc.

### **Parameters**

- editor gui. Diagram Editor containing the connection.
- connection model.ArcItem for deletion.

### deleteItems (editor, items)

Delete items.

### **Parameters**

- **editor** *gui.DiagramEditor* containing the *items*.
- items List with items (selection) to delete.

### deletePlace (editor, place, portClone=False)

Delete places, port places and port clone places.

#### **Parameters**

- editor gui.DiagramEditor containing the place.
- place model.PlaceItem for deletion.
- portClone Flag determining whether a port clone place shall be deleted.

### deleteSubnet (editor)

Recursivly delete subnet editor and their content.

**Parameters** editor – Root gui.DiagramEditor for deletion.

### deleteToken (editor, token)

Delete a token.

### **Parameters**

- **editor** *gui.DiagramEditor* containing the *place*.
- token model. Token I tem for deletion.

### deleteTransition (editor, transition)

Delete transitions, substitution transitions and their subnets.

#### **Parameters**

- **editor** *gui.DiagramEditor* containing the *transition*.
- transition *model.TransitionItem* for deletion.

### keyPressEvent (event)

Callback method, when a key is pressed on the keyboard.

**Parameters** event – QtGui.keyPressEvent.

### newToken (item)

Callback method, when add Token is clicked in colorListView.

**Parameters** item – *colorListView* item.

### sceneMouseMoveEvent (event)

Catch sceneMouseMoveEvent during arc creation.

 $\textbf{Parameters} \ \textbf{event} - \textit{QtGui.sceneMouseMoveEvent}.$ 

### sceneMouseReleaseEvent (event)

Catch QtGui.sceneMouseReleaseEvent on arc completion.

 $\textbf{Parameters} \ \textbf{event} - \textit{QtGui.sceneMouseReleaseEvent}.$ 

### setPortConnection (portPlaceClone, transition)

Set a connection between a port place clone and a substitution transition.

### **Parameters**

- portPlaceClone Port place clone *model.PlaceItem*.
- $\bullet \ \ \textbf{transition} Substituta ion \ transition \ \textit{model.TransitionItem}.$

### setTokenForPlace(place, actualMarking)

Show the tokens contained in place.

### **Parameters**

- place *model.PlaceItem* to show the *model.TokenItem* for.
- actualMarking SNAKES Marking to determine the tokens for *place*.

### setTokens (actualMarking)

Set tokens for actualMarking.

6 Chapter 1. gui

**Parameters** actualMarking – SNAKES Marking to determine the tokens for *place*.

#### shortcutCreateNode()

Create CPN elements with keyboard shortcuts.

### startArc (nodeItem)

Start an arc conneciton.

**Parameters** nodeItem – model.AbstractItem source element.

#### validConnection()

Callback method, when a valid arc connection was created.

#### wheelEvent(event)

Callback method, when the mouse wheel is used.

Parameters event - QtGui.wheelEvent.

### class gui.DiagramEditor.EditorGraphicsView(parent=None)

Bases: PyQt4.QtGui.QGraphicsView

Viewport for the gui.DiagramScene s.

It controls the zooming feature and drag and drop operations from the item library.

Member parent Parent editor widget.

Member scaleFactor Zooming factor.

### dragEnterEvent (event)

Callback method, when an icon is dragged from the libraryModelView to gui.diagramScene.

**Parameters** event – *QtGui.dragEnterEvent*.

#### dragMoveEvent (event)

Callback method, when an icon is moved over the gui.diagramScene.

**Parameters** event – *QtGui.dragMoveEvent*.

### dropEvent (event)

Callback method, when an icon is dropped on the gui.diagramScene.

**Parameters** event – *QtGui.dropEvent*.

### validDrop()

Callback method, when the drop on gui.DiagramScene was valid.

### wheelEvent(event)

Callback method, when the mouse wheel is used.

Parameters event - QtGui.wheelEvent.

```
gui.DiagramEditor.bin_{(QTextStream)} \rightarrow QTextStream
```

```
gui.DiagramEditor.hex_{Q}(QTextStream) \rightarrow QTextStream
```

```
\texttt{gui.DiagramEditor.oct}_(QTextStream) \rightarrow QTextStream
```

class gui.DiagramScene.DiagramScene (parent=None)

Bases: PyQt4.QtGui.QGraphicsScene

Drawing Area.

Member editor Parent gui.DiagramEditor.

**Member hovering** Flag determining, whether hovering is happening. (Workaround, since hovering is not forwarded).

### mouseMoveEvent (event)

Forward mouseMoveEvent during arc creation and reimplement hovering of model. AbstractItem. Connector.

 $\textbf{Parameters} \ \textbf{event} - \textit{QtGui.mouseMoveEvent}.$ 

```
mouseReleaseEvent (event)
          Forward mouseReleaseEvent during arc creation.
              \textbf{Parameters} \ \textbf{event} - QtGui.mouseReleaseEvent.
gui.DiagramScene.bin_(QTextStream) \rightarrow QTextStream
gui.DiagramScene.hex_(QTextStream) \rightarrow QTextStream
qui.DiagramScene.oct (QTextStream) → QTextStream
class gui.LibraryModel.LibraryModel(parent=None)
     Bases: PyQt4.QtGui.QStandardItemModel
     Abstract class for drag and drop support
     mimeData(idxs)
     mimeTypes()
\texttt{gui.LibraryModel.bin}_{\_}(\textit{QTextStream}) \rightarrow \texttt{QTextStream}
qui.LibraryModel.hex_(QTextStream) → QTextStream
\texttt{gui.LibraryModel.oct}\_(\textit{QTextStream}) \rightarrow \texttt{QTextStream}
class gui.NameDialog.NameDialog(parent=None, item=None, itele='Unnamed', default='a')
     Bases: PyQt4.QtGui.QDialog
     cancel()
     getItem()
              Return self.item
     getName()
              Return self.lineEdit.text()
     ok ()
qui.NameDialog.bin_(QTextStream) \rightarrow QTextStream
gui.NameDialog.hex_(QTextStream) \rightarrow QTextStream
qui.NameDialog.oct_(QTextStream) \rightarrow QTextStream
class qui.ParameterDialog.ParameterDialog(node, parent=None)
     Bases: PyQt4.QtGui.QDialog
     OK()
gui.ParameterDialog.bin_(QTextStream) \rightarrow QTextStream
qui.ParameterDialog.hex (QTextStream) \rightarrow QTextStream
gui.ParameterDialog.oct_(QTextStream) \rightarrow QTextStream
class gui.SubnetDialog.SubnetDialog(mainWindow, parent=None, superNet=None, sub-
                                            net=None)
     Bases: PyQt4.QtGui.QDialog
     Dialog for subnet creation
     cancel()
     closeEvent(event)
              Parameters event -
qui.SubnetDialog.bin (QTextStream) \rightarrow QTextStream
gui.SubnetDialog.hex_(QTextStream) \rightarrow QTextStream
gui.SubnetDialog.oct_(QTextStream) \rightarrow QTextStream
```

8 Chapter 1. qui

```
class gui. TokenDialog. TokenDialog (parent=None, title='Choose Colour and Amount of the To-
                                            ken')
     Bases: PyQt4.QtGui.QDialog
     Dialog to choose a color for new token
      cancel()
     getCountToken()
               Return self.countToken
     getInitMarking()
               Return self.initMarking
     getListEntry()
               Return self.listEntry
     ok()
      setCountToken (value)
           Set token count.
               Parameters value -
      setInitMarking(value)
           Set initial Marking.
               Parameters value -
      setListEntry()
           Add new color to list.
\texttt{gui.TokenDialog.bin}\_(\textit{QTextStream}) \rightarrow \texttt{QTextStream}
gui.TokenDialog.hex_(QTextStream) \rightarrow QTextStream
\texttt{gui.TokenDialog.oct}\_(\textit{QTextStream}) \rightarrow \texttt{QTextStream}
```

10 Chapter 1. gui

### **TWO**

### **MODEL**

class model.CPNSimulator.CPNSimulator(mainWindow)

Bases: object

CPN Simulator.

Member mainWindow gui.MainWindow. Main application window.

Member net SNAKES Colored Petrinet.

Member markingHistory SNAKES Marking stored for every 'simulationStep'.

Member simulationStep Steps calculated for the Petrinet net

Member displayStep Step displayed in editors.

Member simulatorSpeed The speed with which the simulator progresses.

Member enabledTransitions Number of enabled transitions.

Member uniqueNameBase Unique integer for the creation of new CPN elements.

Member initialMarking SNAKES Marking at step 0.

Member colourSets Set of defined string colors.

**Member connectionList** List of all *model.ArcItem* s, except port substitution transition connections.

Member transitions All model. Transition Item except substitution transitions.

**Member places** All *model.PlaceItem* except port places.

Member subnets List of all subnets.

```
___init___(mainWindow)
```

Create CPN Simulator.

**Parameters mainWindow** – *gui.MainWindow*. Main application window.

### back2beginning()

Return to step 0.

### backStep()

Go one step back in history.

### checkTranistionActivation (transition, mode, currentStep)

Check whether a SNAKES transition is activated.

### **Parameters**

- transition SNAKES transition to check.
- mode SNAKES mode to check for activation.
- **currentStep** Step for which to calculate the activation

**Return activated** Activated True or False.

**checkTransitionEnabled** (*transition*, *mode*, *currentStep*, *activated*, *transitions2Fire*) Check whether a SNAKES transition is enabled.

#### **Parameters**

- transition SNAKES transition to check.
- mode SNAKES mode to check for activation.
- **currentStep** Step for which to calculate the enabling.
- activated Flag that determines, whether *transition* is activated.
- transitions2Fire List of SNAKES transitions enabled to fire.

**Return enabled** Enabled True or False.

#### defineNewColour(colName)

Define a new Color.

Parameters colName - String color.

### fireEnabledTransitions(currentStep)

Fire transitions in list *transitions2Fire*.

**Parameters** currentStep – Step for which to calculate the firing.

### forward2lastStep()

Go forward to last step in history, given at 'simulationStep'.

### forwardStep()

Go step forward in history or claculate new step.

### getActualMarking(actualMarking)

Process status of visual net, depending on actualMarking.

Parameters actualMarking - SNAKES Marking.

### resetSimulator(init=False)

Reset simulator history to step 0.

Parameters init – Initialization on startup/load

### setNetName()

Set the name of the *net*.

### startSim()

Start simulator with the speed chosen with the radio edit.

### stopSim()

Stop simulator.

### tokenAdded()

Called when a *model.TokenItem* was added to the *net*.

```
model.CPNSimulator.bin_(QTextStream) \rightarrow QTextStream
```

model.CPNSimulator.hex (QTextStream)  $\rightarrow$  QTextStream

 $model.CPNSimulator.oct_(QTextStream) \rightarrow QTextStream$ 

### class model.AbstractItem.AbstractItem(parent=None)

Bases: PyQt4.QtGui.QGraphicsItem

Base class of the CPN elements transition and place.

Member parent gui.DiagramEditor. Editor to show in.

Member tokens List of model. TokenItem s, if inheriting class is model. PlaceItem else None

Member name Name of the CPN element.

**Member posCallbacks** List of callback functions, to calculate position changes.

Member connectorList List of 20 Connector s

**Member planeMap** Dictionary->Set(), mapping the relative orientation of an any other *model.AbstractItem* in the containing editor.

**Member connectorMap** Dictionary->List(), mapping the relative orientation of the 20 *model.AbstractItem.Connector* s.

**Member nodeType** Determining the type of the item at creation time.

**Member superNet** The super net is used to determine the editor for port place clones.

**Member label** *QtGui.QGraphicsTextItem*, visual representation of the *model.AbstractItem* s name.

Member descCanvas gui. AbstractItem. Description Canvas, showing detailed information.

```
___init___(parent=None)
```

Create abstract item.

**Parameters** parent – gui.DiagramEditor. Editor to show in.

### checkItem(item, orientation)

Check item is of propper type.

#### **Parameters**

- item model. AbstractItem to lookup.
- **orientation** Orientation to assign, if type check is passed.

createItem(editor, name='Untitled', nodeType='undefined')

Create the typed item.

#### **Parameters**

- editor gui.DiagramEditor. Editor to show in.
- name Name of the CPN element.
- **nodeType** Determining the type of the item at creation time.

**Return w, h** Width and height of CPN element, depending on the label length.

### deleteItemLocal()

Capture delete event and call editor delete function.

### findItemsInPlanes()

Finds the orientation of any other item in the gui.DiagramScene.

### mouseDoubleClickEvent (event)

Edit visual name on QtGui.mouseDoubleClickEvent.

 $\textbf{Parameters} \ \textbf{event} - \textit{QtGui.mouseDoubleClickEvent}.$ 

### renameElement()

Capture rename event and call virtual function of model. PlaceItem or model. TrasnitionItem.

```
class model.AbstractItem.Connector(parent, idx)
```

Bases: PyQt4.QtGui.QGraphicsRectItem

Connector for visual arcs.

A connector is a socket, to or from which a visual arc may be connected.

Member parent Parent port place.

**Member orientation** Orientation relative to its parent: "N","NE","E","SE","S","SW","W","NW".

Member idx Index assinged from parent.

Member connectionArc Reference to the arc connected to this Connector.

**Member posCallbacks** List of callback functions, to calculate position changes.

```
Member position Position relative to parent.
```

```
__init__ (parent, idx)
Create a connector.
```

### **Parameters**

- parent Parent AbstractItem.
- idx Absolut number in the parent list of connectors.

### hoverEnterEvent (event)

Make connector visible on hoverEnterEvent.

Parameters event - hoverEnterEvent

#### hoverLeaveEvent (event)

Make connector invisible on hoverLeaveEvent.

Parameters event - hoverLeaveEvent

### itemChange (change, value)

Item position has changed, calculate new position.

#### **Parameters**

- change Change event.
- value QtCore.QPointF().

**Return value** QtCore.QPointF(x, y) or super(Connector, self).itemChange(change, value).

### mousePressEvent (event)

Capture QtGui.mousePressEvent with Shift-Key modifier.

Parameters event – QtGui.mousePressEvent

```
class model.AbstractItem.DescriptionCanvas (parent=None)
```

```
Bases: PyQt4.QtGui.QGraphicsRectItem
```

Description area for tokens, exceptions and modes. Toggle with Ctrl\*\*+\*\*M.

In this canvas label, detailed information about the CPN element is shown. Firstly the unique name used by the simulator is shown, followed by the visual name, that is assigned by the user. The latter does not have to be unique.

**Member parent** Parent CPN element.

Member label Visual representation of the information about the CPN element.

**Member text** Unique name of parent CPN element.

**Member visibility** Switch for the visibility of the description canvas.

```
__init__(parent=None)
```

Create description canvas.

Parameters parent – Abstract CPN element.

### setCanvasString(infoString)

Set info string, that appended to the first line.

**Parameters** infoString – Info string, may contain line breaks.

### setVisibility(visible)

Set visibility of DescriptionCanvas.

Parameters visible - True if visible, False if invisible.

```
model.AbstractItem.bin_(QTextStream) \rightarrow QTextStream model.AbstractItem.hex_(QTextStream) \rightarrow QTextStream model.AbstractItem.oct_(QTextStream) \rightarrow QTextStream
```

class model.PlaceItem.PlaceItem (editor, name, position, initMarking=[], uniqueName='p0', port=None, loadFromFile=False, portDirection=None, port-Clone=None, superNet=None, subnet=None)

Bases: PyQt4.QtGui.QGraphicsEllipseItem, model.AbstractItem.AbstractItem

CPN Transition element.

Member editor gui. Diagram Editor. Editor to show in.

Member name Name of the CPN element assigned by the user.

**Member subnet** Name of the subnet this transition is contained in.

**Member position** Position of this item in the gui.DiagramScene.

**Member port** *model.PortItem* if this is a port place, else None.

**Member portClone** A port place in the super net or None.

**Member portDirection** Direction of the port, if this is a port place.

Member uniqueName Unique name of this transition used by the simulator.

**Member place** SNAKES place node with id *uniqueName*, except portClones.

Member initMarking List of initial tokens present in this place.

**Member tokens** List (deque) of *model.TokenItem* s in this place.

Member toolTipString Information string containing the tokens present in this place.

\_\_init\_\_ (editor, name, position, initMarking=[], uniqueName='p0', port=None, loadFrom-File=False, portDirection=None, portClone=None, superNet=None, subnet=None) Create and initialize place item.

#### **Parameters**

- editor gui.DiagramEditor. Editor to show in.
- $\bullet\,$   $\,$  name Name of the CPN element assigned by the user.
- **position** Position of this item in the *gui.DiagramScene*.
- initMarking List of initial tokens present in this place.
- uniqueName Unique name of this transition used by the simulator.
- port model.PortItem if this is a port place, else None.
- loadFromFile Flag determining, whether the place is loaded from file.
- portDirection Direction of the port, if this is a port place.
- portClone A port place in the super net or None.
- **superNe** The super net is used to determine the editor for port place clones.
- **subnet** Name of the subnet this transition is contained in.

### addToken()

Method to add a model. Token I tem to this place

### contextMenuEvent (event)

Generate Context menu on context menu event.

Parameters event – QContextMenuEvent.

### deleteItemLocal()

Capture delete event and call editor delete function.

### mouseMoveEvent (event)

Prevents the movement of this item, when connections are drawn.

**Parameters** event – QtGui.mouseMoveEvent

## mousePressEvent (event) Prevents the movement of this item, when connections are drawn. **Parameters** event – *QtGui.mousePressEvent* mouseReleaseEvent (event) Forward QtGui.mouseReleaseEvent. **Parameters** event – *QtGui.mouseReleaseEvent* newTokenValue() Accept new token. renameModifications (name) Make neccessary modification and renaming. **Parameters** name – Name of the CPN element assigned by the user. setPort (loadFromFile=False) Create a new *model.PortItem* or edit the existing port. **Parameters** loadFromFile – Flag determining, whether the place is loaded from file. setPortDiag() Callback function to modify the port. stackTokens() Method to order the visual stacking of different tokens. $model.PlaceItem.bin_(QTextStream) \rightarrow QTextStream$ $model.PlaceItem.hex_(QTextStream) \rightarrow QTextStream$ $model.PlaceItem.oct_(QTextStream) \rightarrow QTextStream$ class model.TransitionItem.TransitionItem (editor, name, position, guardExpression=None, uniqueName='t0', loadFromFile=False, sub*stitutionTransition=False*, *subnet=None*) Bases: PyQt4.QtGui.QGraphicsRectItem, model.AbstractItem.AbstractItem CPN Transition element. **Member editor** gui.DiagramEditor. Editor to show in. Member name Name of the CPN element assigned by the user. Member substitutionTransition Flag indication a substitutaion transition. Member uniqueName Unique name of this transition used by the simulator. Member transition SNAKES transition node with id uniqueName, only if substitutionTransition is False else None. Member subnet Name of the subnet this transition is contained in. **Member position** Position of this item in the *gui.DiagramScene*. **Member enabled** Flag determining if this transition is enabled. **Member guardExpression** The guard expression for this transition.

Member subnetBorder Second border indicating a substitution transition.

Member exceptionString Exception string containing simulator errors.

Member modeString Mode string containing the enabled modes for this transition

\_\_init\_\_(editor, name, position, guardExpression=None, uniqueName='t0', loadFrom-File=False, substitutionTransition=False, subnet=None)
Create transition item.

### **Parameters**

• editor – gui.DiagramEditor. Editor to show in.

- name Name of the CPN element assigned by the user.
- position Position of this item in the *gui.DiagramScene*.
- **guardExpression** The guard expression for this transition.
- uniqueName Unique name of this transition used by the simulator.
- loadFromFile Flag determining, whether the transition is loaded from file.
- **substitutionTransition** Flag indication a substitutaion transition.
- **subnet** Name of the subnet this transition is contained in.

### acceptEditGuard()

Apply modified guard expression, after gui.NameDialog.

### contextMenuEvent (event)

Generate Context menu on context menu event.

Parameters event – QContextMenuEvent.

### deleteItemLocal()

Capture delete event and call editor delete function.

### editGuard()

Callback function to modify the guard expression.

#### importSubnet()

Forward menu action Import Subnet to gui.DiagramEditor.

### initTransition (loadFromFile)

Initialize transition.

**Parameters loadFromFile** – Flag determining, whether the transition is loaded from file

### mouseMoveEvent (event)

Prevents the movement of this item, when connections are drawn.

Parameters event - QtGui.mouseMoveEvent

### mousePressEvent (event)

Prevents the movement of this item, when connections are drawn.

**Parameters** event – QtGui.mousePressEvent

### mouseReleaseEvent (event)

Forward QtGui.mouseReleaseEvent.

Parameters event - QtGui.mouseReleaseEvent

### openSubnet (

Forward menu action Open Subnet to gui.DiagramEditor.

### renameModifications (name)

Make neccessary modification and renaming.

**Parameters** name – Name of the CPN element assigned by the user.

### setInfoString(stringInfo)

Set the string for the gui. AbstractItem. Descrition Canvas

**Parameters stringInfo** – String appended to the first line of its *gui.AbstractItem.DescritionCanvas*.

```
model.TransitionItem.bin_(QTextStream) \rightarrow QTextStream model.TransitionItem.hex_(QTextStream) \rightarrow QTextStream model.TransitionItem.oct_(QTextStream) \rightarrow QTextStream
```

Bases: PyQt4.QtGui.QGraphicsItem

CPN arc connection element.

Member editor gui. Diagram Editor. Editor to show in.

**Member srcConnector** Source model. AbstractItem. Connector.

**Member dstConnector** Destination model. AbstractItem. Connector.

**Member isPortConnection** Flag that determines, whether the arc represents the connection between a substitution transition and a port place..

**Member name** Expression of the CPN arc assigned by the user.

Member arcDefined Flag that determines, whether an arc creation was successful (internal).

Member variable SNAKES Variable.

Member expression SNAKES Expression.

Member pos1 Point of origin of the arc.

**Member pos2** Point of destination of the arc.

**Member arrowPolygon** *QtGui.QPolygonF* showing the direction of the arc.

\_\_init\_\_ (editor, srcConnector, dstConnector, name='undefined', isPortConnection=False)

#### **Parameters**

- editor gui.DiagramEditor. Editor to show in.
- srcConnector Source model. AbstractItem. Connector.
- **dstConnector** Destination *model.AbstractItem.Connector*.
- name Expression of the CPN arc assigned by the user.
- **isPortConnection** Flag that determines, whether the arc represents the connection between a substitution transition and a port place..

### checkForExpression (text)

Check whether the annotation is intended to be an expression or a variable.

**Parameters** text – Annotation text.

Return ret True/False.

### deleteItemLocal()

Capture delete event and call editor delete function.

### getName()

Return the annotation of this arc.

### multiInput (multiArcAnnotations)

Register a multi input arc in the simulator.

**Parameters multiArcAnnotations** – SNAKES MultiArc annotation.

### rename()

Rename arc and apply changes to simulator and visual representation.

### **setArcAnnotation** (annotationText=None)

Finalize arc creation and set annotation.

Parameters annotationText - String arc annotation.

**Return self.arcDefined** Flag that determines, whether an arc creation was successful (internal).

```
Callback method to keep pos1 up to date.
              Parameters pos1 – QtCore.QPointF().
      setDestination(dstConnector)
          Sets the destination model. AbstractItem. Connector.
               Parameters dstConnector – Destination model. AbstractItem. Connector.
      setEndPos (endpos)
          Callback method to keep pos2 up to date.
              Parameters endpos – QtCore.QPointF().
      setName (name)
          Set the annotation of this arc.
              Parameters name – New annotation.
      setPolygon()
          Calculate position and rotation of the arc arrow head.
      singleInput (variableExpression)
          Register a single input arc in the simulator.
              Parameters variable Expression – SNAKES Variable or Expression.
      singleOutput (variableExpression)
          Register a multi input arc in the simulator.
              Parameters variable Expression – SNAKES Variable or Expression.
class model.ArcItem.LineItem(parent)
     Bases: PyQt4.QtGui.QGraphicsLineItem
     Visual representation of the line itself.
          Member parent Parent model. ArcItem.
      __init___(parent)
          Create line.
              Parameters parent - Parent model. ArcItem.
     mouseDoubleClickEvent (event)
          Edit arc annotation on QtGui.mouseDoubleClickEvent.
              \textbf{Parameters} \ \textbf{event} - \textit{QtGui.mouseDoubleClickEvent}.
model.ArcItem.bin_(QTextStream) \rightarrow QTextStream
model.ArcItem.hex (QTextStream) \rightarrow QTextStream
model.ArcItem.oct_(QTextStream) \rightarrow QTextStream
class model.TokenItem.TokenItem(editor, token, count, qpos, parent=None)
     Bases: PyQt4.QtGui.QGraphicsEllipseItem
     A String Token.
          Member editor Referenced editor.
          Member countToken Number of tokens.
          Member countTokenLabel Visible representation of the number of tokens.
          Member token String value of token, shown in tooltip
       _init__ (editor, token, count, qpos, parent=None)
          Create a token.
```

setBeginPos(pos1)

**Parameters** 

```
• editor – DiagramEditor. Editor to show in.
                   • token - Token value.
                   • count – Number of Tokens to create.
                   • qpos – Parent top right position.
                   • parent=None - Parent Place Element
     contextMenuEvent (event)
          Generate Context menu on context menu event.
              Parameters event – QContextMenuEvent.
     deleteItemLocal()
          Capture delete event and call editor delete function.
     setCountToken (count)
          Token count, shown in green circle.
              Parameters count – Number to show.
model.TokenItem.bin_(QTextStream) \rightarrow QTextStream
model.TokenItem.hex_(QTextStream) \rightarrow QTextStream
model.TokenItem.oct_(QTextStream) \rightarrow QTextStream
class model.PortItem.PortItem(direction, parent=None)
     Bases: PyQt4.QtGui.QGraphicsEllipseItem
     Port place indicator.
     The label indicates the port direction of the port place.
          Member direction Port direction: (i)nput, (o)utput, (io) bidirectional.
          Member parent Parent port place.
          Member label Visual representation of direction.
       _init__ (direction, parent=None)
          Create a port
              Parameters
                   • direction – Port direction: (i)nput, (o)utput, (io) bidirectional.
                   • parent – Parent port place.
     contextMenuEvent (event)
          Generate Context menu on context menu event.
              Parameters event – QContextMenuEvent.
     editPort()
          Edit port direction.
     getDirection()
          Return direction of the port: "i", "o", "io".
     itemChange (change, value)
```

### **Parameters**

- change Change value.
- value QtCore.QPointF().

Item position has changed, calculate new position.

**Return value** QtCore.QPointF(x, y) or super(PortItem, self).itemChange(change, value).

### **CHAPTER**

### THREE

### **INOUT**

```
class inout.XMLIO.XMLIO(simulator, rootElementName='')
    Bases: object
    XML Input and Output
    loadNet (filename)
```

Parse XML file and prepare data for object creation.

**Return [ netName, subnets, serConnections, serPlaces, serTransitions ]** A list containing lists for object creation.

Parameters filename – Filepath to XML file which shall be loaded.

netToXML (subnet, placesS, transitionsS, connectionsS)
Save Colored Petrinet Subnet to XML tree.

### **Parameters**

- **subnet** Subnet name.
- placesS Place contained in *subnet*.
- transitionsS Transitions contained in *subnet*.
- connectionsS Connections contained in *subnet*.

### saveLog(logList)

Save Log Entries to XML tree.

Parameters logList – List of log entries from the log widget.

### saveNet (filename)

Save XML tree to file.

**Parameters** filename – Filepath where the XML tree is saved.

```
inout.XMLIO.bin_(QTextStream) \rightarrow QTextStream inout.XMLIO.hex_(QTextStream) \rightarrow QTextStream inout.XMLIO.oct_(QTextStream) \rightarrow QTextStream
```

24 Chapter 3. inout

### **CHAPTER**

# **FOUR**

# **INDICES AND TABLES**

- genindex
- modindex
- search

Colored Petrinets Simulator Gui Documentation, Release alpha 2				
,	-			

### PYTHON MODULE INDEX

### g gui.DiagramEditor,5 gui.DiagramScene,7 gui.LibraryModel,8 gui.MainWindow, 3 gui.NameDialog, 8 gui.ParameterDialog,8 gui.SubnetDialog, 8 gui.TokenDialog,8 inout.XMLIO, 23 m model.AbstractItem, 12 model.ArcItem, 17 model.CPNSimulator, 11 model.PlaceItem, 15 model.PortItem, 20 model.TokenItem, 19 model.TransitionItem, 16

init() (model.AbstractItem.AbstractItem method),	checkForExpression() (model.ArcItem.ArcItem method), 18
init() (model.AbstractItem.Connector method), 14	checkItem() (model.AbstractItem.AbstractItem
init() (model.AbstractItem.DescriptionCanvas	method), 13
method), 14	checkTranistionActivation()
init() (model.ArcItem.ArcItem method), 18	(model.CPNSimulator.CPNSimulator
init() (model.ArcItem.LineItem method), 19	method), 11
init() (model.CPNSimulator.CPNSimulator	checkTransitionEnabled()
method), 11	(model.CPNSimulator.CPNSimulator
init() (model.PlaceItem.PlaceItem method), 15	method), 11
init() (model.PortItem.PortItem method), 20	closeEvent() (gui.MainWindow.MainWindow method),
init() (model.TokenItem.TokenItem method), 19 init()	closeEvent() (gui.SubnetDialog.SubnetDialog method),
method), 16	cioseEvent() (gui.SubhetDialog.SubhetDialog method),
method), 10	Connector (class in model.AbstractItem), 13
AbstractItem (class in model.AbstractItem), 12	contextMenuEvent() (model.PlaceItem.PlaceItem
acceptEditGuard() (model.TransitionItem.TransitionItem	method), 15
method), 17	contextMenuEvent() (model.PortItem.PortItem
addSubnetEditor() (gui.MainWindow.MainWindow	method), 20
method), 3	contextMenuEvent() (model.TokenItem.TokenItem
addToken() (model.PlaceItem.PlaceItem method), 15	method), 20
ArcItem (class in model.ArcItem), 17	context Menu Event ()  (model. Transition I tem. Transition I tem
1 101 1 O ( 11 CD)(C) 1 CD)(C)	method), 17
back2beginning() (model.CPNSimulator.CPNSimulator	CPNSimulator (class in model.CPNSimulator), 11
method), 11	createIcon() (gui.DiagramEditor.DiagramEditor
backStep() (model.CPNSimulator.CPNSimulator method), 11	method), 5
bin_() (in module gui.DiagramEditor), 7	createItem() (model.AbstractItem.AbstractItem
bin_() (in module gui.DiagramScene), 8	method), 13
bin_() (in module gui.LibraryModel), 8	createItemsAssignToEditor()
bin_() (in module gui.MainWindow), 5	(gui.MainWindow.MainWindow method), 3
bin_() (in module gui.NameDialog), 8	createSubnetItemLists()
bin_() (in module gui.ParameterDialog), 8	(gui.MainWindow.MainWindow method), 3
bin_() (in module gui.SubnetDialog), 8	defineNewColour() (model.CPNSimulator.CPNSimulator
bin_() (in module gui.TokenDialog), 9	method), 12
bin_() (in module inout.XMLIO), 23	defineNewToken() (gui.DiagramEditor.DiagramEditor
bin_() (in module model.AbstractItem), 14	method), 5
bin_() (in module model.ArcItem), 19	deleteArc() (gui.DiagramEditor.DiagramEditor
bin_() (in module model.CPNSimulator), 12	method), 5
bin_() (in module model.PlaceItem), 16	deleteItemLocal() (model.AbstractItem.AbstractItem
bin_() (in module model.PortItem), 21	method), 13
bin_() (in module model.TokenItem), 20	deleteItemLocal() (model.ArcItem.ArcItem method),
bin_() (in module model.TransitionItem), 17	18
cancel() (gui.NameDialog.NameDialog method), 8	deleteItemLocal() (model.PlaceItem.PlaceItem
cancel() (gui.SubnetDialog.SubnetDialog method), 8	method), 15
cancel() (gui.TokenDialog.TokenDialog method), 9	deleteItemLocal() (model.TokenItem.TokenItem
	method), 20

dalata Itam I agal () (modal Transition Itam Transition Itam	hav () (in modula qui DiagramEditor) 7
deleteItemLocal() (model.TransitionItem.TransitionItem method), 17	hex_() (in module gui.DiagramScene), 8
deleteItems() (gui.DiagramEditor.DiagramEditor	hex_() (in module gui.LibraryModel), 8
method), 5	hex_() (in module gui.MainWindow), 5
deletePlace() (gui.DiagramEditor.DiagramEditor	hex_() (in module gui.NameDialog), 8
method), 5	hex_() (in module guiValueDialog), 8
deleteSubnet() (gui.DiagramEditor.DiagramEditor	hex_() (in module gui. SubnetDialog), 8
method), 6	hex_() (in module gui.TokenDialog), 9
deleteToken() (gui.DiagramEditor.DiagramEditor	hex_() (in module inout.XMLIO), 23
method), 6	hex_() (in module model.AbstractItem), 14
deleteTransition() (gui.DiagramEditor.DiagramEditor	hex_() (in module model.ArcItem), 19
method), 6	hex_() (in module model.CPNSimulator), 12
DescriptionCanvas (class in model.AbstractItem), 14	hex_() (in module model.PlaceItem), 16
DiagramEditor (class in gui.DiagramEditor), 5	hex_() (in module model.PortItem), 21
DiagramScene (class in gui.DiagramScene), 7	hex_() (in module model. TokenItem), 20
dragEnterEvent() (gui.DiagramEditor.EditorGraphicsView	
method), 7	hoverEnterEvent() (model.AbstractItem.Connector
dragMoveEvent() (gui.DiagramEditor.EditorGraphicsVie	
method), 7	hoverLeaveEvent() (model.AbstractItem.Connector
dropEvent() (gui.DiagramEditor.EditorGraphicsView	method), 14
method), 7	method), 14
method), 7	importSubnet() (gui.MainWindow.MainWindow
editGuard() (model.TransitionItem.TransitionItem	method), 4
method), 17	importSubnet() (model.TransitionItem.TransitionItem
EditorGraphicsView (class in gui.DiagramEditor), 7	method), 17
editPort() (model.PortItem.PortItem method), 20	initNetFromFile() (gui.MainWindow.MainWindow
export_Step_as_SVG()	method), 4
(gui.MainWindow.MainWindow method), 4	initTransition() (model.TransitionItem.TransitionItem
(3	method), 17
findItemsInPlanes() (model.AbstractItem.AbstractItem	inout.XMLIO (module), 23
method), 13	itemChange() (model.AbstractItem.Connector
fireEnabledTransitions()	method), 14
(model.CPNSimulator.CPNSimulator	itemChange() (model.PortItem.PortItem method), 20
method), 12	remenange() (modern ortherm orthern method), 20
forward2lastStep() (model.CPNSimulator.CPNSimulator	keyPressEvent() (gui.DiagramEditor.DiagramEditor
method), 12	method), 6
forwardStep() (model.CPNSimulator.CPNSimulator	
method), 12	LibraryModel (class in gui.LibraryModel), 8
	LineItem (class in model.ArcItem), 19
$getActualMarking () \ (model. CPNS imulator. CPNS imulator) \ (model. CPNS imulator) \ (model.$	nloadNet() (gui.MainWindow.MainWindow method), 4
method), 12	loadNet() (inout.XMLIO.XMLIO method), 23
getCountToken() (gui.TokenDialog.TokenDialog	lookupSubstitutionConnectors()
method), 9	(gui.MainWindow.MainWindow method), 4
getDirection() (model.PortItem.PortItem method), 20	
getInitMarking() (gui.TokenDialog.TokenDialog	MainWindow (class in gui.MainWindow), 3
method), 9	mimeData() (gui.LibraryModel.LibraryModel method),
getItem() (gui.NameDialog.NameDialog method), 8	8
getListEntry() (gui.TokenDialog.TokenDialog method),	mimeTypes() (gui.LibraryModel.LibraryModel
9	method), 8
getName() (gui.NameDialog.NameDialog method), 8	model.AbstractItem (module), 12
getName() (model.ArcItem.ArcItem method), 18	model.ArcItem (module), 17
gui.DiagramEditor (module), 5	model.CPNSimulator (module), 11
gui.DiagramScene (module), 7	model.PlaceItem (module), 15
gui.LibraryModel (module), 8	model.PortItem (module), 20
gui.MainWindow (module), 3	model.TokenItem (module), 19
gui.NameDialog (module), 8	model.TransitionItem (module), 16
gui.ParameterDialog (module), 8	mouseDoubleClickEvent()
gui.SubnetDialog (module), 8	(model.AbstractItem.AbstractItem method),
gui TokenDialog (module) 8	13

30 Index

	rename() (model.ArcItem.ArcItem method), 18
method), 19	renameElement() (model.AbstractItem.AbstractItem
mouseMoveEvent() (gui.DiagramScene.DiagramScene	method), 13
method), 7	renameModifications() (model.PlaceItem.PlaceItem
mouseMoveEvent() (model.PlaceItem.PlaceItem method), 15	method), 16 renameModifications()
mouseMoveEvent() (model.TransitionItem.TransitionIter	*
method), 17	method), 17
mousePressEvent() (model.AbstractItem.Connector	resetSimulator() (model.CPNSimulator.CPNSimulator
method), 14	method), 12
mousePressEvent() (model.PlaceItem.PlaceItem	,,
method), 15	saveLog() (inout.XMLIO.XMLIO method), 23
mousePressEvent() (model.TransitionItem.TransitionIten	1 saveNet() (gui.MainWindow.MainWindow method), 4
method), 17	saveNet() (inout.XMLIO.XMLIO method), 23
mouseReleaseEvent() (gui.DiagramScene.DiagramScene	sceneMouseMoveEvent()
method), 8	(gui.DiagramEditor.DiagramEditor method),
mouseReleaseEvent() (model.PlaceItem.PlaceItem	6
method), 16	sceneMouseReleaseEvent()
mouseReleaseEvent() (model.TransitionItem.TransitionIt	em (gui.DiagramEditor.DiagramEditor method),
method), 17	setArc() (gui.MainWindow.MainWindow method), 4
multiInput() (model.ArcItem.ArcItem method), 18	setArcAnnotation() (model.ArcItem.ArcItem method),
NameDialog (class in gui.NameDialog), 8	18
netToXML() (inout.XMLIO.XMLIO method), 23	setBeginPos() (model.ArcItem.ArcItem method), 18
newNet() (gui.MainWindow.MainWindow method), 4	setCanvasString() (model.AbstractItem.DescriptionCanvas
newToken() (gui.DiagramEditor.DiagramEditor	method), 14
method), 6	setCountToken() (gui.TokenDialog.TokenDialog
newTokenValue() (model.PlaceItem.PlaceItem	method), 9
method), 16	setCountToken() (model.TokenItem.TokenItem
and () (in madula and Diagram Edition) 7	method), 20
oct_() (in module gui.DiagramEditor), 7	setDestination() (model.ArcItem.ArcItem method), 19
oct_() (in module gui.DiagramScene), 8	setDirection() (model.PortItem.PortItem method), 20
oct_() (in module gui.LibraryModel), 8 oct_() (in module gui.MainWindow), 5	setEndPos() (model.ArcItem.ArcItem method), 19
oct_() (in module gui.NameDialog), 8	setInfoString() (model.TransitionItem.TransitionItem
oct_() (in module gui.ParameterDialog), 8	method), 17
oct_() (in module gui.SubnetDialog), 8	setInitMarking() (gui.TokenDialog.TokenDialog
oct_() (in module gui.TokenDialog), 9	method), 9
oct_() (in module inout.XMLIO), 23	setListEntry() (gui.TokenDialog.TokenDialog method),
oct_() (in module model.AbstractItem), 14	setName() (model.ArcItem.ArcItem method), 19
oct_() (in module model.ArcItem), 19	setNetName() (model.CPNSimulator.CPNSimulator
oct_() (in module model.CPNSimulator), 12	method), 12
oct_() (in module model.PlaceItem), 16	setPolygon() (model.ArcItem.ArcItem method), 19
oct_() (in module model.PortItem), 21	setPort() (model.PlaceItem.PlaceItem method), 16
oct_() (in module model.TokenItem), 20	setPort() (model.PortItem.PortItem method), 21
oct_() (in module model.TransitionItem), 17	setPortConnection() (gui.DiagramEditor.DiagramEditor
ok() (gui.NameDialog.NameDialog method), 8	method), 6
OK() (gui.ParameterDialog.ParameterDialog method),	setPortDiag() (model.PlaceItem.PlaceItem method), 16
8	setTokenForPlace() (gui.DiagramEditor.DiagramEditor
ok() (gui.TokenDialog.TokenDialog method), 9	method), 6
openSubnet() (gui.MainWindow.MainWindow	setTokens() (gui.DiagramEditor.DiagramEditor
method), 4	method), 6
openSubnet() (model.TransitionItem.TransitionItem	$set Visibility () \ (model. Abstract Item. Description Canvas$
method), 17	method), 14
Paramatar Dialog (class in qui Daramatar Dialog) 9	$shortcut Create Node () \\ (gui. Diagram Editor. Diagram Editor) \\$
ParameterDialog (class in gui.ParameterDialog), 8 PlaceItem (class in model.PlaceItem), 15	method), 7
PortItem (class in model.Placettem), 15 PortItem (class in model.PortItem), 20	singleInput() (model.ArcItem.ArcItem method), 19
i orthem (class in modell orthem), 20	singleOutput() (model ArcItem ArcItem method) 19

Index 31

```
stackTokens() (model.PlaceItem.PlaceItem method), 16
startArc()\ (gui. Diagram Editor. Diagram Editor\ method),
                 (model.CPNSimulator.CPNSimulator
startSim()
         method), 12
                 (model.CPNSimulator.CPNSimulator
stopSim()
         method), 12
SubnetDialog (class in gui.SubnetDialog), 8
tokenAdded()
                 (model. CPN Simulator. CPN Simulator\\
         method), 12
TokenDialog (class in gui.TokenDialog), 8
TokenItem (class in model.TokenItem), 19
TransitionItem (class in model.TransitionItem), 16
valid Connection () \\ \quad (gui. Diagram Editor. Diagram Editor
         method), 7
validDrop()
             (gui.DiagramEditor.EditorGraphicsView
         method), 7
wheelEvent()
                   (gui. Diagram Editor. Diagram Editor\\
         method), 7
wheelEvent() (gui.DiagramEditor.EditorGraphicsView
         method), 7
```

XMLIO (class in inout.XMLIO), 23

32 Index