

Node Editor framework user manual

Functions (last update 17/11/2020)

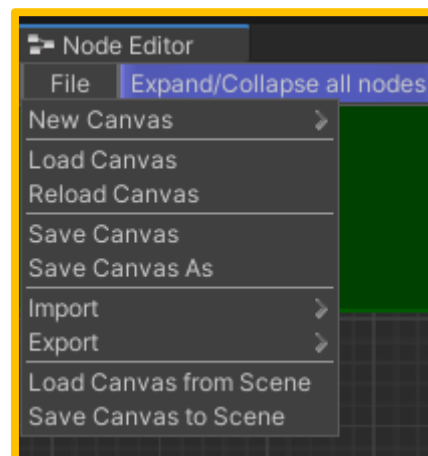
Open the editing window: *Window/Node Editor* will open the editing window.



Create new graph: With *File/Create new graph* of the editing window, the possible types of canvas/graphs available will be shown. The one we are interested in is StoryGraphCanvas.

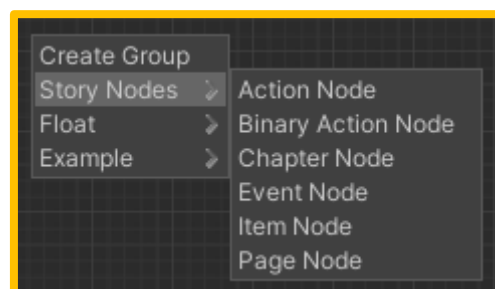
Save and load a graph:

- With *File/Save canvas as* you can save a graph that has not been previously saved.
- With *File/Save canvas* you can overwrite the graph that is open and saved somewhere in the project.
- With *File/Load canvas* you can load a saved graph.
- With *File/Reload canvas* you can reload the open graph, undoing unsaved changes.



Create a new node: By *right-clicking* on any part of the empty canvas, a context menu will appear where you can select a type of node to create. The ones we are interested in are in the *Story Nodes* submenu.

Connections between nodes: Nodes will have one or more input and output ports. By *dragging with the mouse* from an output port it can be connected to an input port of another node, or vice versa.



Connections allow a certain number of connections. See the annex on the different types of nodes.

Connections can be edited depending on the number of connections allowed. To facilitate the removal of all connections from a port *right click* on a port, a context menu will appear with the only option of *Clear connections*.

Restrictions on connections between nodes: When working with StoryNodes, only those ports that share color can be connected, although there are exceptions:

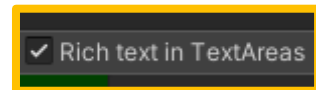
- A *Page In Other Chapter* output connection from an ActionNode (dark blue) can (and should) be connected to the main PageNode input (green).
- A ChapterNode's Lost Pages output connection (grayish) can be connected to the main PageNodes inputs.

Node deletion and duplication: *Right-clicking* on a node displays a contextual menu with the option to delete that node (*Delete*) or duplicate it (*Duplicate*).

The selected nodes can also be deleted by pressing *Delete*, and duplicated by pressing *Ctrl+D*.

RichText: The text of some node fields may be formatted. The visibility of this will depend on the *Checkbox* in the top bar of the editor that indicates *Rich text in TextAreas*.

Formatted text will be entered manually by the user, having to surround the text to be formatted with tags like ``.



Editing nodes from the Unity Inspector: In addition to various editable parameters such as color, when selecting a node, it is possible to edit in in the Inspector since a representation of the node will be drawn in this window.

Node icons: All nodes that inherit from StoryNode (our nodes) have the ability to display icons, which can be added, modified and removed from the Inspector with a selected node.

In addition, some nodes will also display default icons depending on certain conditions. See the annex on the different types of nodes.



Node selection: To select any node individually, *left-click* on it. To deselect any node or nodes, *left-click* on any empty part of the canvas.

To select multiple nodes, a multiple selection can be made by *dragging the mouse*, which will draw a connection box indicating which nodes are to be selected. This type of selection is

cumulative, i.e., multiple selections in a row will add their selections together, rather than one multiple selection undoing what was selected by the previous one.

Another option for multiple selection is by *left-clicking with Shift* on the nodes to be selected. And also, by *left-clicking with Ctrl* on the header of a group, you can select the nodes within that group.

Finally, with *Ctrl+A* all nodes will be deselected if any are selected and all will be selected if none are selected.

Autogeneration of nodes: If, in a StoryGraphCanvas, you *hold down the Ctrl key* just before releasing a connection arising from a StoryNode, the node corresponding to that type of connection and to that port of that StoryNode will be generated. See the appendix on node types to find out which node is generated from which port of which StoryNode.

Export and import: *File/Export* will display the available export formats (*StoryXML* is the one we are interested in). Once one of them is selected, a Windows Explorer window will open to select the path and the name under which the XML file will be exported.

File/Import will display the available import formats (the same as export). A window for selecting the file to be imported will also open.

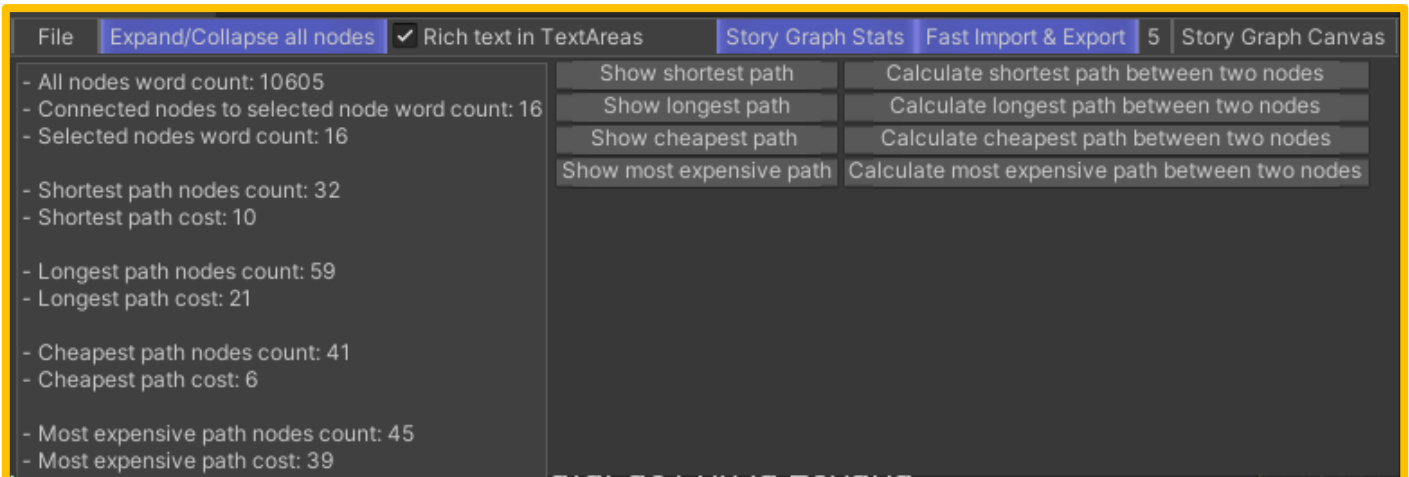
Exporting a file does not mean saving it. The extension of saved graphs will be *.asset* and that of exported ones *.xml*. Also, given the purpose of using the editor, it may be more interesting to work only with import and export.

Recommendation: in case of modifying a script and observing that the changes are not executed correctly on an open graph, it is advisable to reimport it.

Graph stats: The window accessible from *Window/Graph Stats* contains the following information:

- Word count: Counts the words in the text field of all nodes, of the selected nodes and of all nodes connected to the output of the selected node (if only one), whether directly or indirectly connected.
- Information about the paths over the graph: 4 paths over the graph are automatically calculated – the shortest, the longest, the cheapest and the most expensive – between

the source BookNode and the last ActionNode connected to the graph; and of these paths, the number of nodes that compose it and the cost of traversing it are shown.



In addition to this information, there are 8 buttons, 4 of them to display each of the automatically calculated paths, and the other 4 to select two nodes of the graph, calculate the path between them and display it.

Group creation and editing: *Right-clicking* on any empty part of the canvas displays the same context menu for manually creating nodes, which also includes the *Create Group* option. A group will be created with a default title and size and a random color.

To edit the dimensions of a group, there are handles on the edges and corners that, if dragged, will modify the size of that group. To modify the name and color, select the group (*left-click* on the header) and the Inspector will display, among other information, the fields to edit these values.

A group can include nodes inside it only if the upper left corner and the lower right corner of the node are contained by the group. The same is true for containing other groups.

To move a group, *drag the group header with the left mouse button*, but this movement will also include the contents of the group. To move only the group, you must accompany by dragging with the *Shift key*.

Finally, a group can also be created around the selected nodes by pressing *Ctrl+G*.

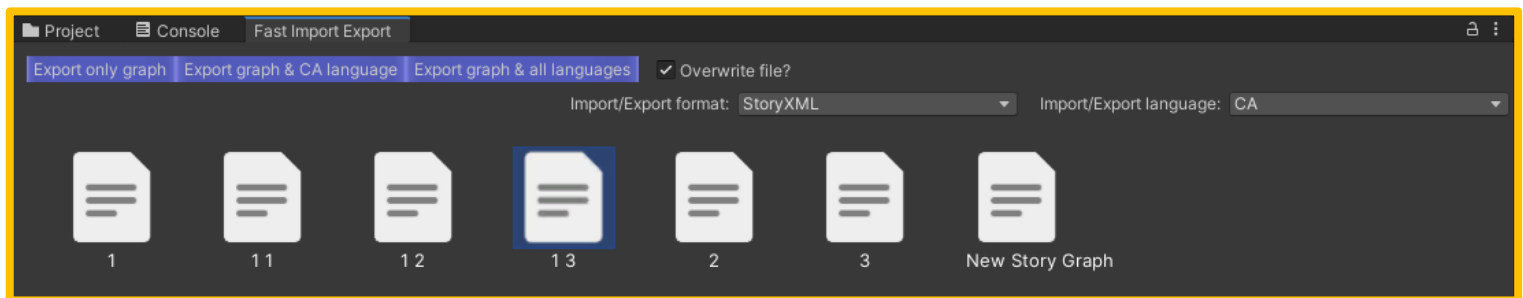
Node and group collapse: A node can be collapsed (it shows only important and cannot be modified from the canvas) by *clicking on it with the middle mouse button*. It is uncollapsed with the same button.

You can collapse all nodes of the graph using the *Expand/Collapse all nodes* button in the top bar of the editor.

Finally, groups can also be collapsed by clicking the middle button on the node's header, which makes all of its content invisible except for one node, the main node.

Fast Export and Import: The window accessible from *Window/Fast Import Export* displays icons and names corresponding to the books present in the Resources/Books folder. This window has different functions and options:

- Import: To import any of the books represented by the icons, just *click* on any of them (they are buttons).
- Export: There are three buttons to export, the first one exports only the graph, the second one exports the graph together with the selected language and the third one exports the graph and all available languages of this graph.
- *Overwrite file?*: If checked, when exporting, it will overwrite the file currently open. Otherwise, it will export the book on a new file whose name will be entered in a text field that appears next to this option.
- *Import/Export format*: Dropdown showing the available export and import formats. Depending on the one selected, only the books with associated XML that comply with that format will be displayed. Books will be exported following the selected format.
- *Import/Export language*: Dropdown containing the language codes available in the currently open book. Changing it will change the language in which the texts are displayed in the nodes.



Translation Assistance: Through the window accessible from *Window/Translation Assistance* it is possible to modify the texts of the nodes in a given language. You can choose the reference language, which also the language in which the nodes of the editor window are displayed and also the *Import/Export language* of the *Fast Import Export* window; and you can choose the language of the translation, whose texts are the ones that can be modified.

On the other hand, there is the possibility of adding languages to the current graph and removing them if there is more than one.

Finally, there are several visibility and filtering options to display only the desired node types.

Navigation: With the window accessible from *Window/Navigation Assistance* you can navigate the graph more easily by clicking on the nodes shown in its list. These nodes are ordered according to their hierarchy and some of them can be opened and closed to show or hide those below them in the hierarchy.

Several nodes can be selected with *Control* and a whole set of nodes can be selected from an initial and a final node with *Shift*.

Finally, you can filter the nodes to be displayed by entering text that must contain the nodes to be displayed in a text field at the top of the window.

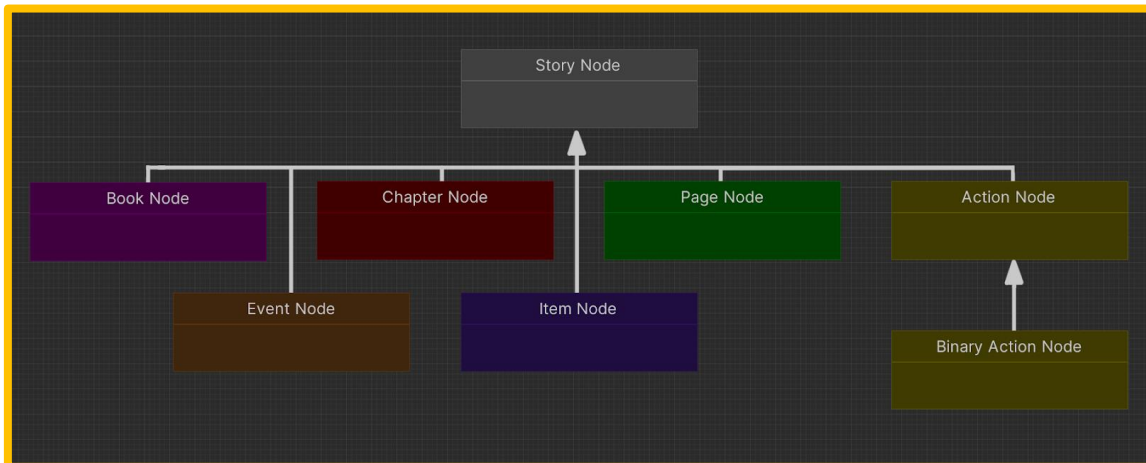
The screenshot shows the 'Translation Assistance' window with the following sections:

- Visibility settings:** Checkboxes for 'Show book', 'Show chapters', 'Show pages', 'Show actions', and 'Show items' are all checked. Below are buttons for 'Open/Close all nodes', 'Add language', and 'Delete ES language'.
- Reference text / Translated text:** Two dropdown menus are set to 'EN' and 'ES' respectively.
- Book node:** Book title: 'El alzamiento3'. Translated text: 'El alzamiento2'.
- Chapter node:** Chapter title: 'El camino de Samasé'. Translated text: 'El camino de Samasé'.
- Death message:** Reference text: 'Las decisiones que has tomado durante tu camino te han condenado a un destino terrible. Samasé te espera.' Translated text: 'Las decisiones que has tomado durante tu camino te han condenado a un destino terrible. Samasé te espera.'
- Page node:** Texts: Reference text: 'Aferras con tanta fuerza tu espada que tus nudillos palidecen, la lluvia cala tu cuerpo y dificulta la visión. Los árboles se agitan, golpeados violentamente por el viento. Intentas secarte el agua de la cara con la mano, pero sigues aferrando la espada con fuerza.' Translated text: 'Aferras con tanta fuerza tu espada que tus nudillos palidecen, la lluvia cala tu cuerpo y dificulta la visión. Los árboles se agitan, golpeados violentamente por el viento. Intentas secarte el agua de la cara con la mano, pero sigues aferrando la espada con fuerza.'

The screenshot shows the 'Translation Assistance' window with a search filter applied. The search bar contains the text 'El alzamiento3'. The list of nodes is filtered to show only those containing this text:

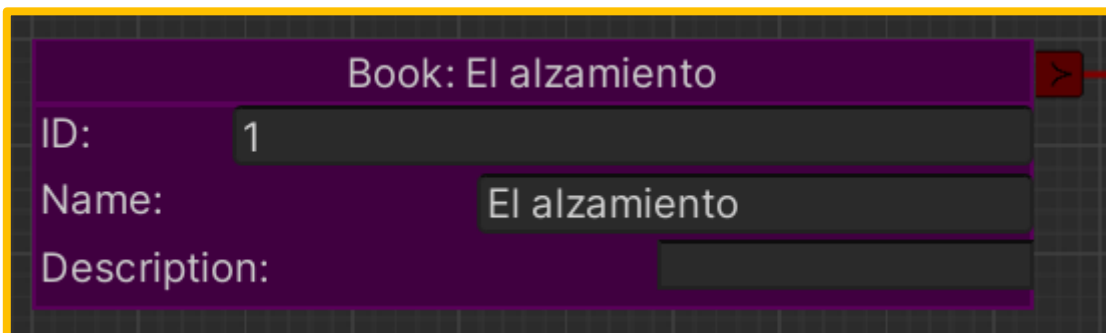
- BookNode:** El alzamiento3
- ChapterNode:** El camino de Samasé
- PageNode:** Aferras con tanta fuerza tu espada que tus nudillos palidecen,...
- ActionNode:** Atacas con la espada al enemigo de la derecha, parece...
- ActionNode:** Lanzas un conjuro Luz Sagrada para deslumbrarlos.
- ActionNode:** Adoptas una posición defensiva y esperas a que ataquen ellos....
- EventNode:** EquipItem | GathanGreine
- EventNode:** EquipItem | FrayedCloak
- EventNode:** EquipItem | LeatherBoots
- EventNode:** EquipItem | PatchedPants
- PageNode:** «¿A cuánto lo tengo? ¿Tres metros?» Aprietas los dientes, tus...
- ActionNode:** Te giras y atacas con la espada.
- ActionNode:** Lanzas un conjuro Escudo Divino, para aumentar tu resistencia
- EventNode:** TakeDamage | 7
- PageNode:** Pronuncias las palabras aprendidas y notas la fuerza de Sama...
- PageNode:** Clavas tus pies en el barro y sositenes firmemente tu...
- PageNode:** El asaltante retira su espada y se prepara para un...
- PageNode:** El asaltante retira su espada y se prepara para un...
- PageNode:** La lluvia arrecia, notas como la ropa pesa más. «¿Es...
- PageNode:** —¡Vas a pagar lo que le has hecho a nuestros...
- PageNode:** Murmuras las oraciones estudiadas. —¡Vas a pagar lo que le...
- PageNode:** Dedicas una oración a las almas de los asaltantes caídos....
- PageNode:** Recorres el claro con la mirada en busca de tu...
- PageNode:** La lluvia arrecia, notas como la ropa pesa más. «¿Es...
- PageNode:** Atardece. Tras un largo día de camino, Ceffyl Gwyn y...
- PageNode:** Aunque ya ha amanecido, el denso manto de nubes de...
- PageNode:** Dedicas una oración a las almas de los asaltantes caídos....
- PageNode:** —¿Qué pasa Ceffyl Gwyn? ¿Qué has visto?. Observas con ate...
- PageNode:** Ceffyl Gwyn finalmente te hace caso y se dirige al...
- PageNode:** En el centro de la estancia, algo arde dentro de...
- PageNode:** —Espérame aquí Ceffyl Gwyn. Lentamente rodeas al gigante
- PageNode:** Tus labios susurran las palabras y sientes en tus manos...
- PageNode:** —Espérame aquí Ceffyl Gwyn —murmuras a tu montura—. Si c...
- PageNode:** Ves como el Carch-celva prepara sus músculos para atacarte
- PageNode:** Sin tomarte un instante para pensar, alzas la mano y...
- PageNode:**

Annex: Types of nodes (last update 17/11/2020)



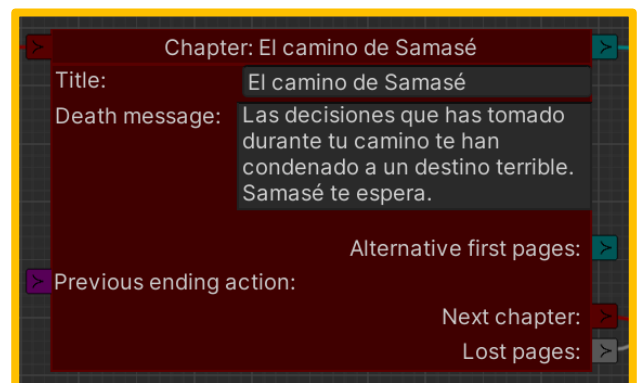
StoryNode: It is an abstract node useful to generate a common base from which the rest of the nodes will inherit. All StoryNode can be collapsed and have icons. It has two ports, StoryIn and StoryOut, but they will only be visible and used only visually when it is the main node of a closed group.

BookNode: It corresponds to the book itself and, if you are working on a StoryGraphCanvas, there will only be one and there must be at least one in the entire graph. Its editable fields are its ID, its name and its description. It has a single output port (red) to connect to a single chapter, the initial chapter, so autogeneration from that port will create a chapter node.



ChapterNode: It corresponds to a chapter of the book. Its editable fields are the chapter title and the death message. It has 6 ports:

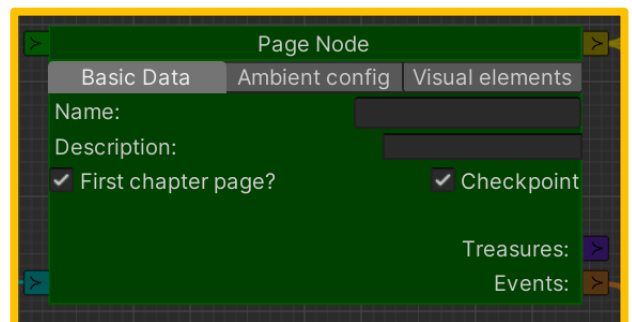
- **Main input port** (simple, red): It can be connected to the output port of a BookNode or to the NextChapter output port of another ChapterNode. Autogeneration from this port creates a ChapterNode.
- **Main output port** (simple, cyan): It connects to the FirstPage input of a single PageNode, that being the first page of the book.



- Autogeneration from this port creates a PageNode marked as the first page.
- **AlternativeFirstPages output port** (multiple, cyan): It connects to *FirstPage* input ports of PageNodes being the alternative first pages of its chapter.
La autogeneración a partir de este puerto crea un PageNode marcado como primera página.
Autogeneration from this port creates a PageNode marked as the first page of its chapter.
 - **PreviousEndingAction input port** (multiple, magenta): It connects to the main ActionNodes outputs when they are in *NextChapter* mode, being these actions the ones that end their chapter and lead to the beginning of the next chapter.
Autogeneration from this port creates an ActionNode in *NextChapter* mode.
 - **NextChapter output port** (simple, red): It connects to the main input of another ChapterNode, the other being the chapter following the current one.
Autogeneration from this port creates a ChapterNode.
 - **LostPages output port** (multiple, white): It connects to the main inputs of PageNodes that are not otherwise connected to the chapter. Connections to this port will be automatically drawn when an XML is imported and a certain PageNode has a *chapterID* corresponding to a ChapterNode but is not connected to either the other ChapterNode output ports or any ActionNode belonging to the chapter.
Autogeneration does not create any nodes from this port.

PageNode: It corresponds to a page of the chapter to which it is connected. Its editable fields are classified into the following categories:

- **BasicData:** Page name, page description, first page checkbox (*First chapter page?*) and checkpoint checkbox.
- **Ambient config:** Status (dropdown of various options) and ambients (combinable checkboxes).
- **Visual elements:** It contains a title checkbox to determine whether the page displays a title or not. Visual elements of two different types can be added, text or image. Each text will have a *TextArea* to modify it and each image, a dropdown with the available images to select.



The connection ports of a PageNode are as follows.

- **Main input port** (multiple, green): The main output of the ActionNode can be connected to this port both in *NextPage* mode (if page and action share chapter) and in *PageInOtherChapter* mode (if page and action are of different chapters), and also the *LostPages* output of a ChapterNode. If you connect to an ActionNode the connections to a *LostPages* port will be cleared and vice versa.
Autogeneration from this port will create an ActionNode in *NextPage* mode.
- **Main output port** (multiple, yellow): The main ActionNodes inputs can be connected to this port.
Autogeneration from this port will create an ActionNode.

- **FirstPage input port** (simple, cyan): This port can be connected to the main output port or to the *AlternativeFirstPages* port of a ChapterNode. For this port to be available, the first page indicator must be enabled. Autogeneration from this port will create a ChapterNode, connecting it to its main output port.
- **Treasures output port** (multiple, purple): This port can be connected to the main ItemNodes inputs. Autogeneration from this port will create an ItemNode.
- **Events output port** (multiple, orange): This port can be connected to the main EventNodes inputs. Autogeneration from this port will create an EventNode.

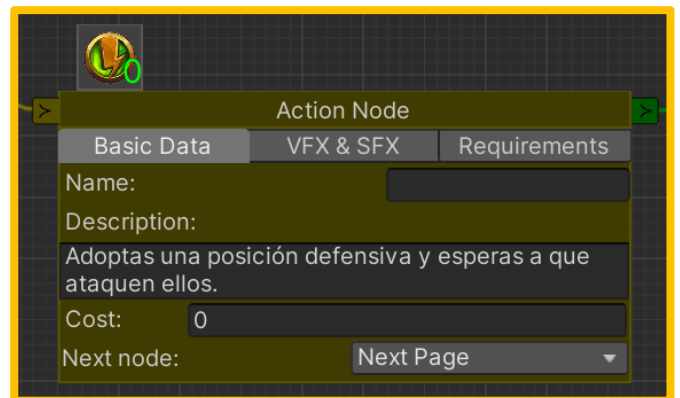
ActionNode: It corresponds to one of the possible actions you can take on a page. Its editable fields are divided into the following categories:

- **Basic Data:** Action name, description, cost and output connection type (*NextPage*, *NextChapter* and *PageInOtherChapter*).
- **VFX & SFX:** Both *VFX* and *SFX* can be added, selectable from a dropdown menu.
- **Requirements:** *ActionRequirement* objects can be added, which have a type, a name (dependent on the type) and a value.

The port of an ActionNode are as follows:

- **Main input port** (simple, yellow): It connects to the main output of a PageNode. Autogeneration from this port creates a PageNode.
- **Main output port** (simple; green, magenta or dark blue): Depending on the type of output connection chosen, it can be connected in different ways:
 - o *NextPage*: It can be connected to the main input of a PageNode of the same chapter.
 - o *NextChapter*: It can be connected to the *PreviousEndingAction* input of a ChapterNode.
 - o *PageInOtherChapter*: It can be connected to the main input of a PageNode of another chapter.

Autogeneration from this port creates, in *NextPage* mode, a PageNode and, in *NextChapter* mode, a ChapterNode.

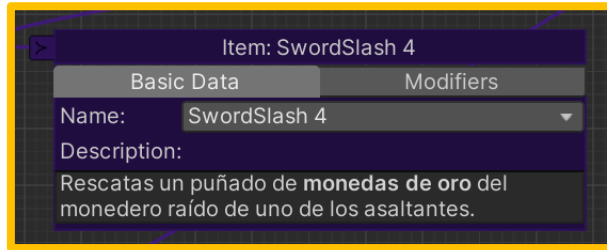


ItemNode: It corresponds to an item that is picked up on a page. Its editable fields are classified into the following categories:

- **Basic Data:** Item name (selectable from those available) and pick-up description.

- **Modifiers:** *ItemModifier* can be added, with a modifying stat and a value.

ItemNodes have only one main input port that connects to the *Treasures* output of a PageNode. Autogeneration from this port creates a PageNode.



EventNode: It corresponds to an event occurring on a page. Its editable fields are the type of the event (selectable from those available) and the value of the event. It has only one main input port that connects to the *Events* output of a PageNode. Autogeneration from this creates a PageNode.



BinaryActionNode: It corresponds to one of the possible actions that can be taken on a page, with two possible outcomes. It is a specialization of the ActionNodes, so it has the same fields, adding the connection type of the second output.

As for the ports, it also has the same ports as an ActionNode, but adds a second output (*Fail*), but it works the same as the main output (*Pass*)

