



PANTHEA VS

Last edition 30.11.2017

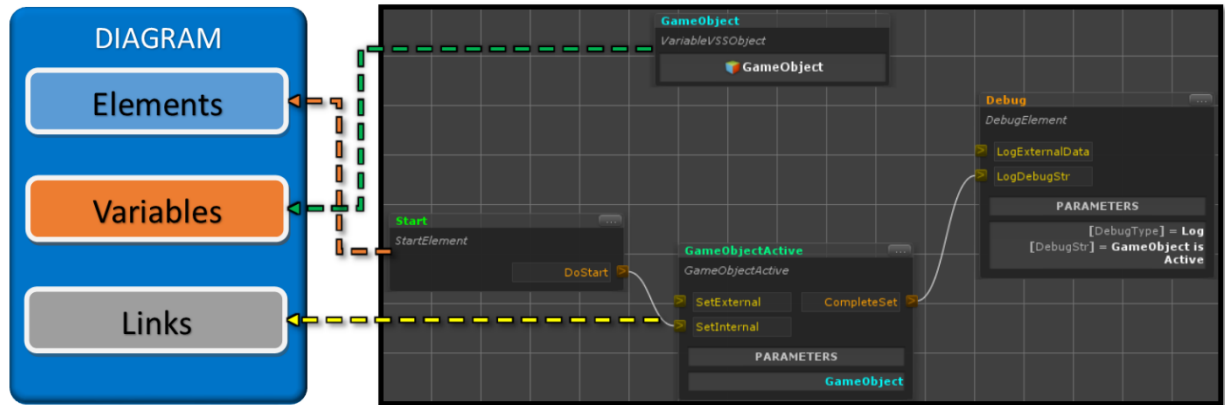
User manual

Introduction	2
Start working with Panthea VS.....	2
The editor	3
Working with diagrams	3
Working with elements	5
Adding elements to a diagram	5
Moving elements.....	6
Interacting with elements	6
Finding elements in a diagram	7
The catalog of elements	8
Element inspector	9
Element groups	10
Diagram variables.....	11
Working with Unity objects.....	11
Links between elements	12
Creating a link.....	12
Working with a link	13
Setting the links order	14
Diagram recovery	15
Debugging a diagram	15
Settings.....	18
Localization.....	20
Global messages.....	23
Sound	24

Introduction

Welcome to the Panthea VS user guide. This document will describe the basic principles of working with the diagram editor: creating and editing diagrams, setting up, working with elements, the catalog and the property inspector. Also, the specifics of working with sound, localization and global messages will be considered.

Before proceeding with the description of the editor, let's summarize what the visual programming system of Panthea VS is in general.



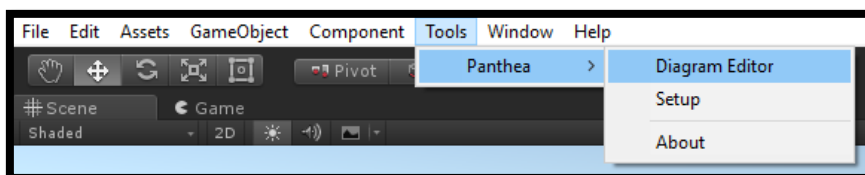
The figure shows a generalized scheme that describes the main parts that determine the system of visual programming. Here:

- **Elements** are logic that implements a certain functional. In general, this is the part that receives, processes, and transmits data.
- **Variables** are data that is shared between elements in one diagram.
- **Links** - this is the part that determines the sequence of execution of the elements logic.
- **Diagram** is a set of elements, variables, and links that implements some functional (s) of the application.

Let's now describe in more detail how to create diagrams and breathe life into them.

Start working with Panthea VS

To start working with the visual programming system, open the scene and select the menu item, as shown in the figure below.



After starting the diagram editor, to initialize the scene, you should press the button as shown in the figure (the same button recovers the scene diagrams, if any).

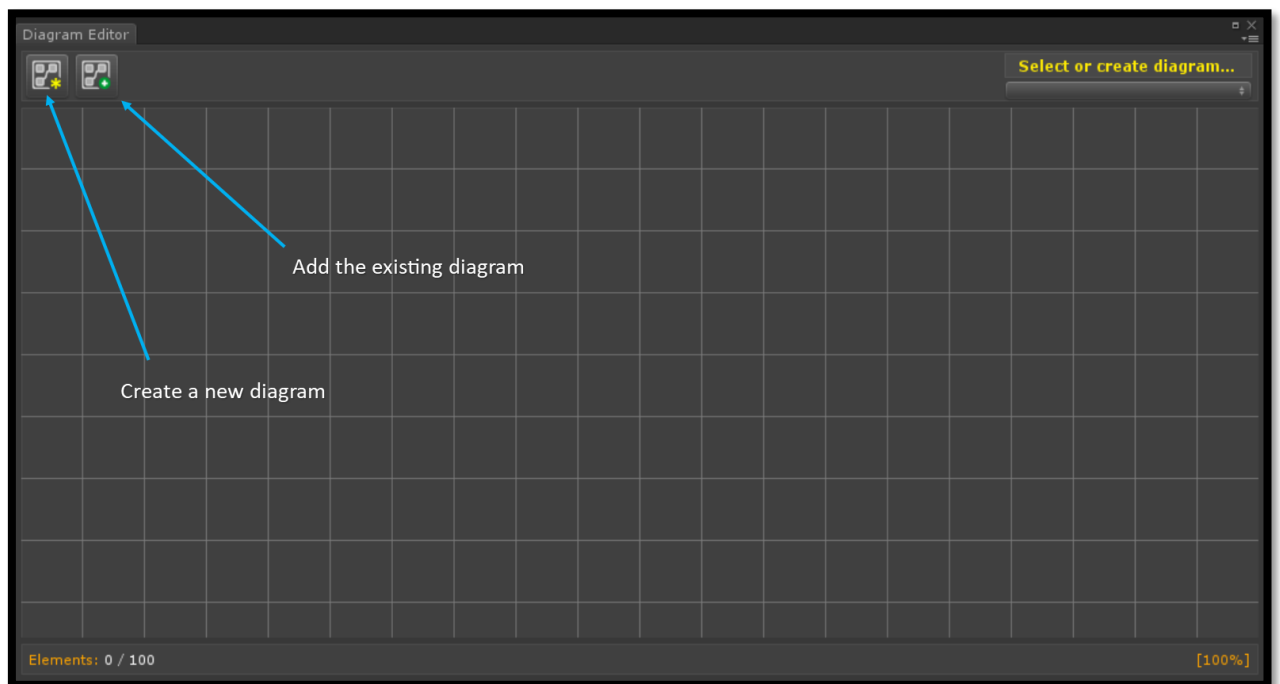


Now the scene is ready for work.

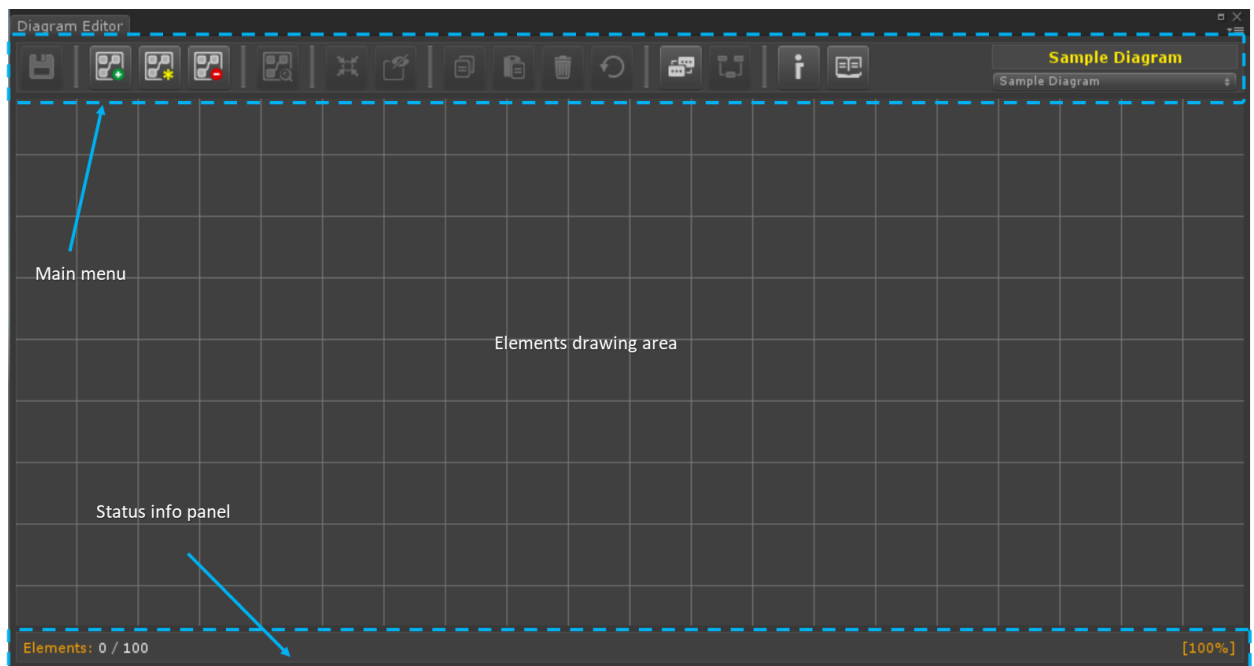
The editor

Working with diagrams

After the scene is initialized, to start working with the diagram, you need to create it, or add an existing one (for example, from templates), use the buttons as shown in the figure.



After creating a new diagram (or adding an existing one), the editor is ready to work with it. Here are its components in more details.



The diagram editor menu items are:



- Save the current diagram (saving is performed automatically if the application is launched in editor mode). You can also configure periodic automatic saving (see section Settings).



- Add an existing diagram or a ready template to the scene.



- create a new diagram.



- delete the current diagram from the scene (diagram file will remain, to totally delete the diagram you must delete the diagram file).



- undo all unsaved changes in the diagram.



- view the links between diagrams



- select a scene diagram and change its name.

Note: to change the name you need to click on it.

Zooming and scrolling

You can change the view in the editor window by scrolling to the different parts of a diagram, and by changing the view scale.

To scroll the view:

(PC) - drag the mouse with 3-d button pressed, or hold the Alt key and drag the mouse with the left button pressed.

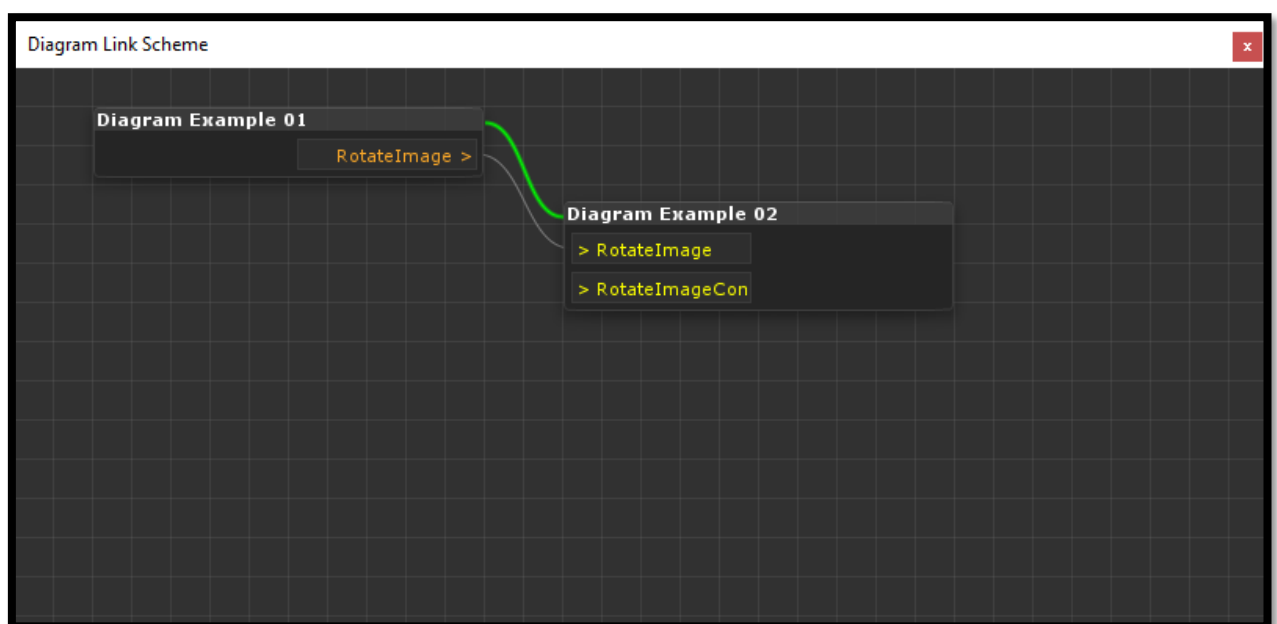
(Mac) – Hold the Alt key and drag the mouse with left button pressed (or touchpad corner).

To zoom the view in/out:

(PC) – use the mouse wheel.

(Mac) – use your apple mouse surface, or move two fingers over the trackpad.

Here are some details about the links between diagrams. During the work, you may often need to transfer data from one diagram to another. This can be done with the help of elements (direct transfer), or via sending a global message. In either of these cases, a link is established. Such links can be quite a lot, and it will be difficult to track them without a special tool. To address this, Panthea VS system has the ability to view them in a separate window.



Here, green marks the direct links between the diagrams (links that ensure that only certain diagrams will process the data). White shows the links through global messages (read more about them in a separate section).

After getting familiar with diagrams, now let's learn how to fill them with elements.

Working with elements

Adding elements to a diagram

In order to add an element to the scene, there are two ways:

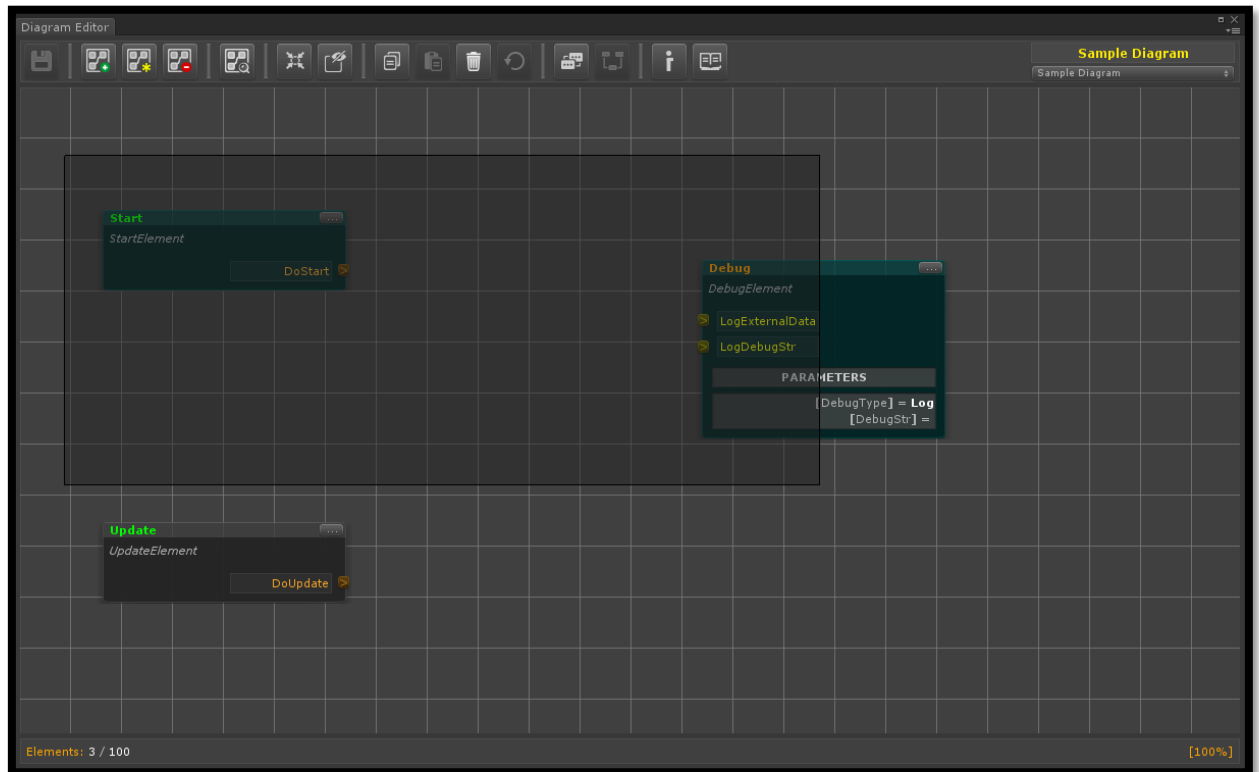
1. Drag an element from the catalog to the drawing area of the elements.
2. Add an element through the context menu, called by clicking on the right mouse button.

This way you can fill the diagram with elements, after which you can perform various manipulations on them.

Moving elements

To move an element, you need to select it, just click on it with the left mouse button. You can also select several elements. To do this, you need to perform the above manipulations, but with the Ctrl key pressed. Selecting several elements in the area is done by moving the mouse with the left mouse button pressed.

Once the elements are selected, you can move them by holding the same mouse button on the element title and moving it to the desired position. Below is a picture for explanation.



Interacting with elements

With the help of the editor menu buttons you can also perform a number of manipulations with the elements:



- minimize or maximize all elements of the diagram.



- hide or show elements parameters.



- copy selected elements to the clipboard.



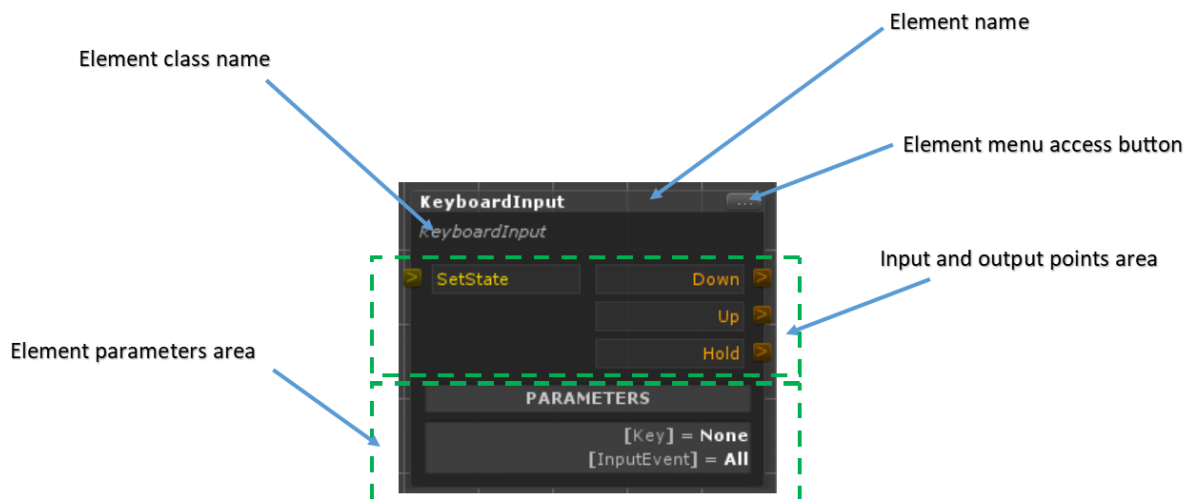
- paste the copied elements from the clipboard.



- delete selected elements (also can be done by pressing the **Del** key)

The same actions can be performed on an element through the context menu called by clicking the right mouse button on the element.

In addition to the menu buttons described above, you can directly interact with a specific element. The element consists of several components:



Here:

- The element menu button opens a context menu for interacting with an element, which can contain the following items:
 - **Minimize/Maximize** – button for minimization or maximization of the element (switch the view of the element into a compact form). The button may be absent if the element has no more than one entry point or an exit point. Also you can switch visual state of the element by double-clicking on its header.
 - **Inversion** – this button inverts the position of the input and output points.
 - **Open Script** – opens a file with the code that implements the logic of the element.
- **Parameters** – button to hide or show the parameter area.
- The element parameters area is the zone where the key parameters of the element and their values are displayed.

Finding elements in a diagram

Depending on the needs of the developer, the diagram can contain a large number of elements, which in turn can complicate the navigation on it. To solve this problem in the editor, you can search the diagram with the possibility of focusing on the found element. To open the search window, press the



menu button

Below is a picture of this window.



In order to find a specific element by name, it must be entered in the input line. Search among the elements is carried out contextually. To focus on the element, you must click on the corresponding button in the search results area.

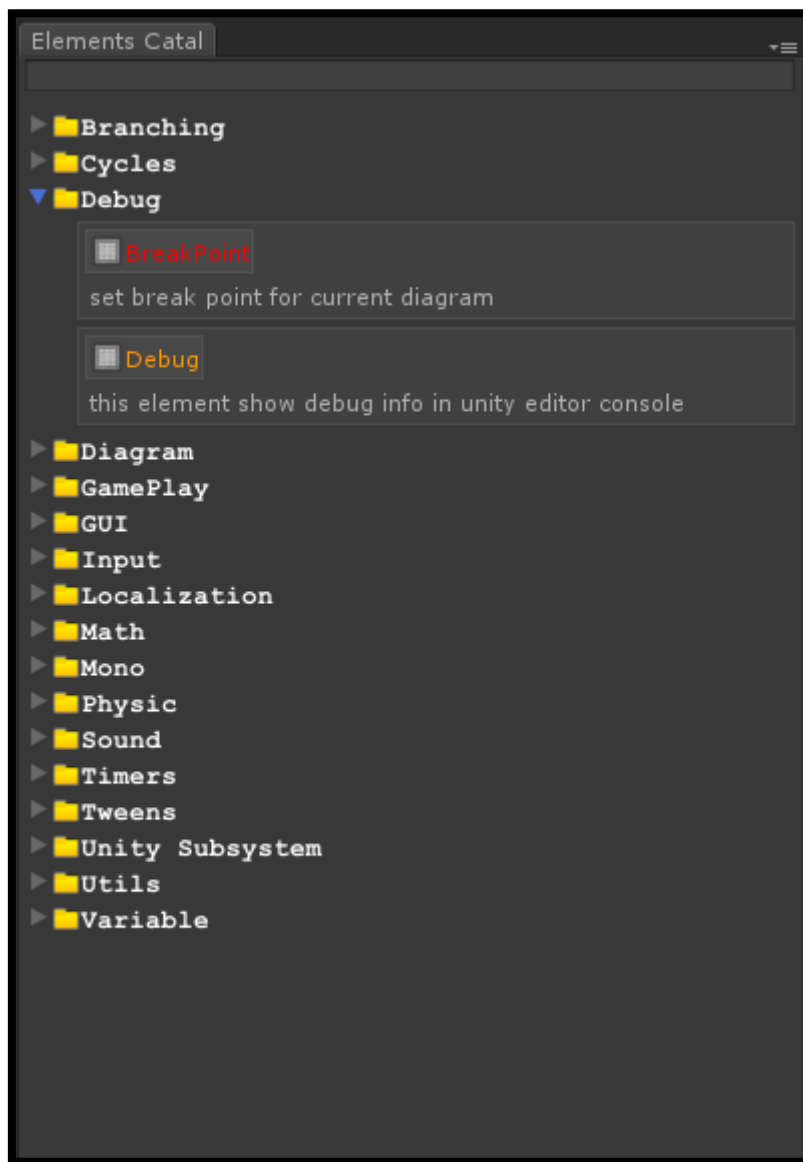
The catalog of elements

To add an element to the diagram it can be dragged from the catalog to the drawing area.



To open the catalog window, click the button in the main editor menu.

Below is a picture showing the window of the elements catalog:



The catalog is a structure of folders and elements in them. Each element is presented in the form of a name and a brief description of what it does. Since a large number of elements can be present in the catalog, for convenience of navigation on them, it is possible to search through the input line at the top of the window. Search among the elements is done contextually as you type.

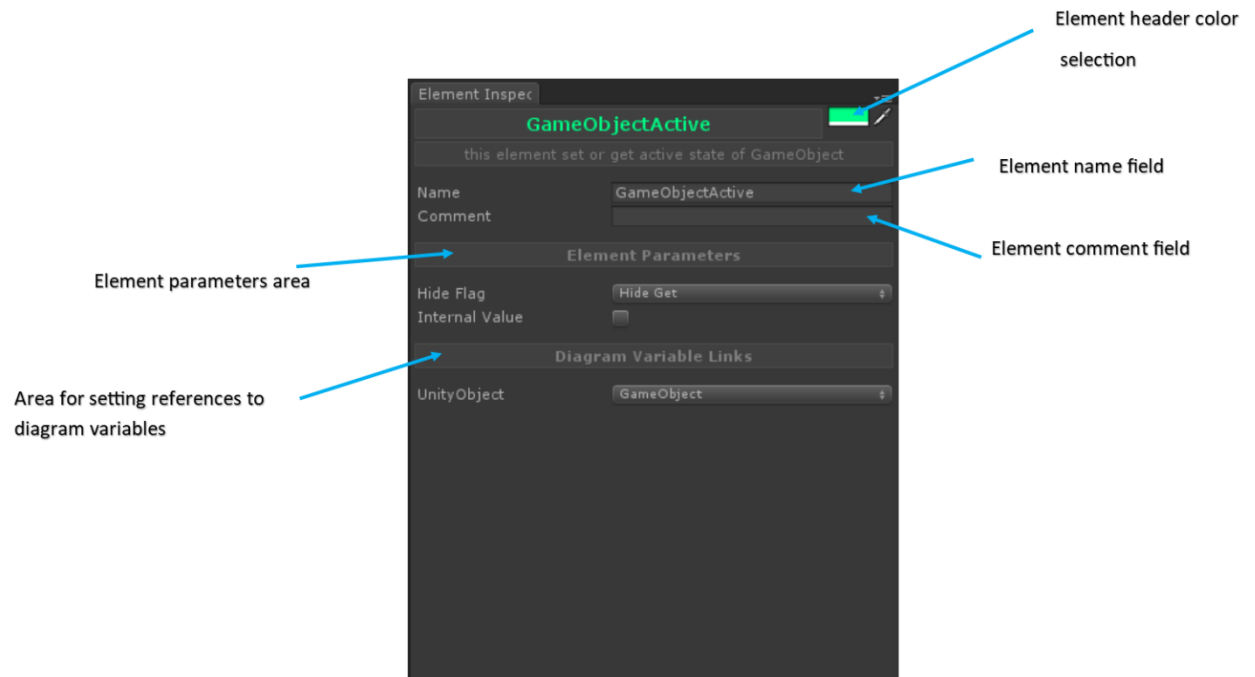
Element inspector

Above we have considered almost all aspects of interaction with the element, the last important part is setting up element parameters. For this purpose, there is a specialized inspector in the editor

whose window can be called up with this button



from the main menu. Look at it in the picture below.



Here:

- In the upper right corner, there is a button for setting the color of the element header.
- **Name** – field for changing the name of an element
- **Comment** – field for setting a comment for the element (the comment is displayed when you hover the mouse over an element).
- **Element Parameters** – element parameters area.
- **Diagram Variable Links** – an area for setting references to diagram variables, if any. Read more about the variables below.

Element groups

While working with the diagram, you may need to increase its readability, so that other developers can quickly understand what is happening there. In the basic version, the logic of the game is divided among the diagrams, so that each of them is responsible for its specific part. However, even in this case, the number of elements can be quite large, and it can be difficult to understand diagram logic. To improve the perception of the diagram in the Panthea VS system, the visual element groups can be created. Below is an image showing a group of variables.



To create a group, select the elements and chose **Group** in the right-click menu. After creating a group, you can move it, just like a single item (holding the mouse cursor on the group header).

To configure the name of the group and its color, double-click on the header. The figure below shows the editing mode.



For the newly created element to be immediately placed in a group, you must either drag it from the catalog to the group area, or right-click it in the group.

To exclude elements from the group, select them and select **Ungroup** from the right-click menu.

Note: Group boundaries are automatically adjusted depending on the position of the contained elements.

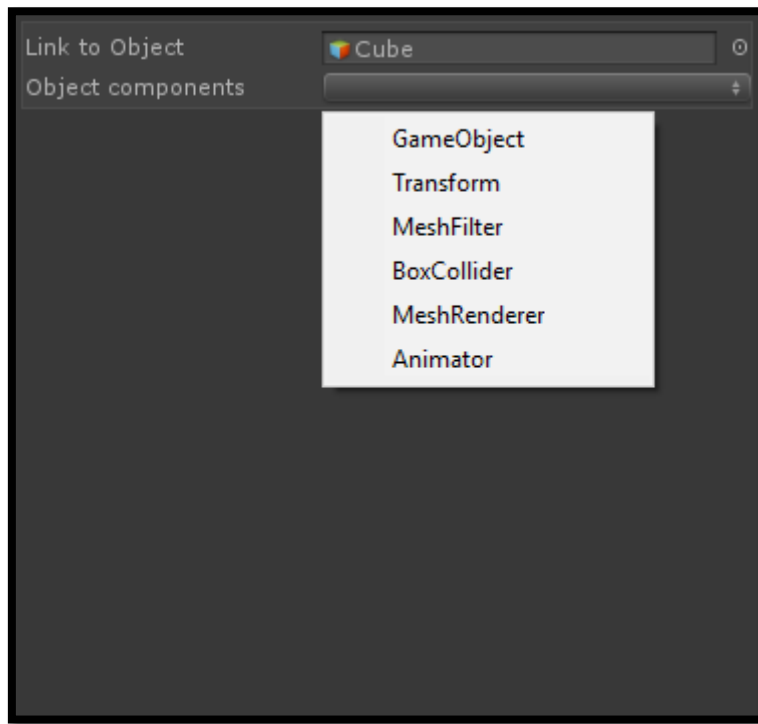
Diagram variables

Variables are the same elements that do not have input and output points, and have only one parameter that is responsible for the value of the variable. The main meaning of this entity is to be able to separate data between the elements of the diagram so that you do not have to perform the same manipulations with each new element. Thus, if two elements refer to the same variable, then if you change the value of the variable, you do not need to do anything with the elements, they will automatically use the new value.

Note: when setting references to variables, their list in the inspector is automatically generated for the type of data that is required for the operation of the elements.

Working with Unity objects

Due to the specifics of the Panthea VS system operation, direct links to Unity objects cannot be used in it, since they cannot be restored when the diagram is loaded. To solve this problem, a specialized VSSObject shell was created, which allows you to create such references, and can save and load them. Among other things, this shell has a special property editor that allows you to get components from any scene object (see the figure below) if you want to refer to them. With VSSObject, you can store references not only to scene objects and their components, but also to prefabs and resource files, such as textures, sounds, and so on.



Important!:

If you use an existing diagram in another scene, object references will be lost, including links to prefabs, because the scene acts as their store. This also needs to be considered, if you plan to use the diagram as a template, in this case, the best option would be to transfer the necessary links to it from the outside (for example, from a diagram that is tied to a scene).

Links between elements

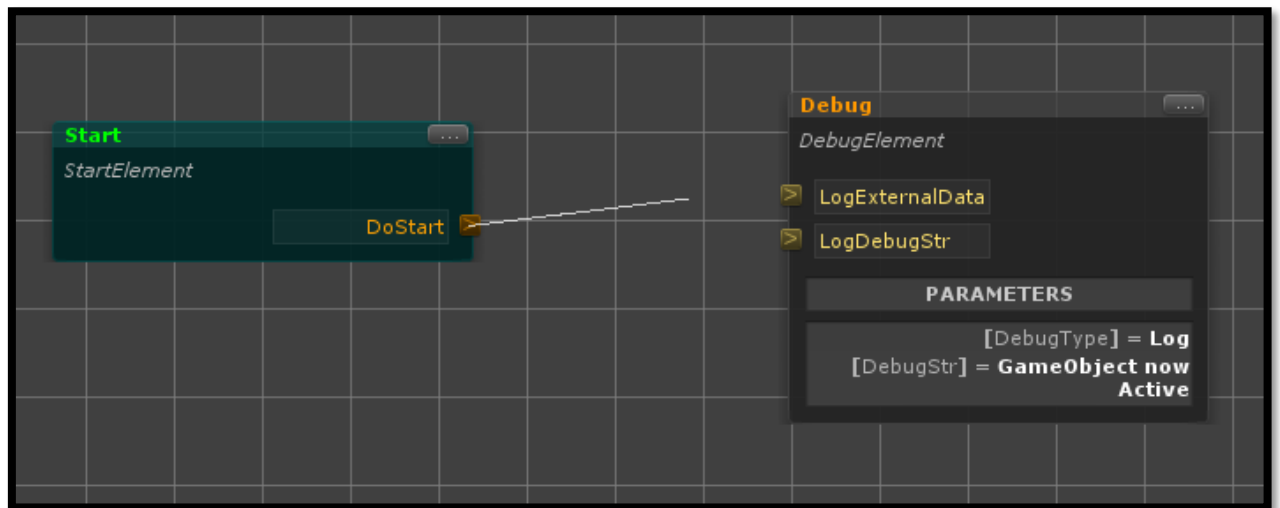
Creating a link

Previously, we determined how to create diagram and how to fill it with elements and variables. The last step is to create links between the elements to determine the sequence of their work. The process of creating links is simple enough, but it has several rules:

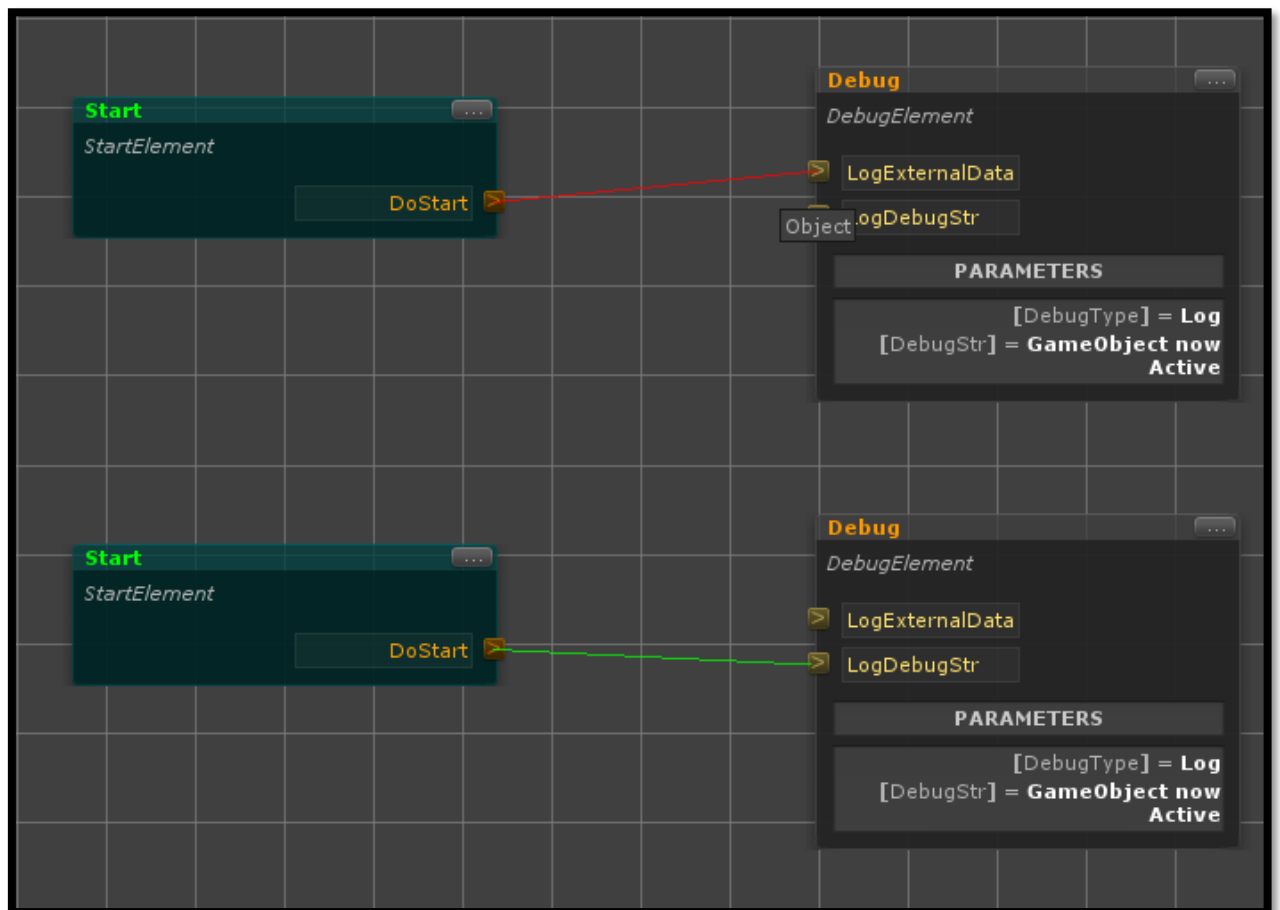
1. The link is always established between the output point of one element and the input point of another element (in a particular case, you can establish a connection between the points of the same element).
2. The link cannot be established between points transmitting and receiving data of different types.

Note: You can establish a link if the input point is of type object, or does not accept data at all.

To create a link, click on the output point with the left mouse button and without releasing it, start moving to the input point of the element (see the figure below).



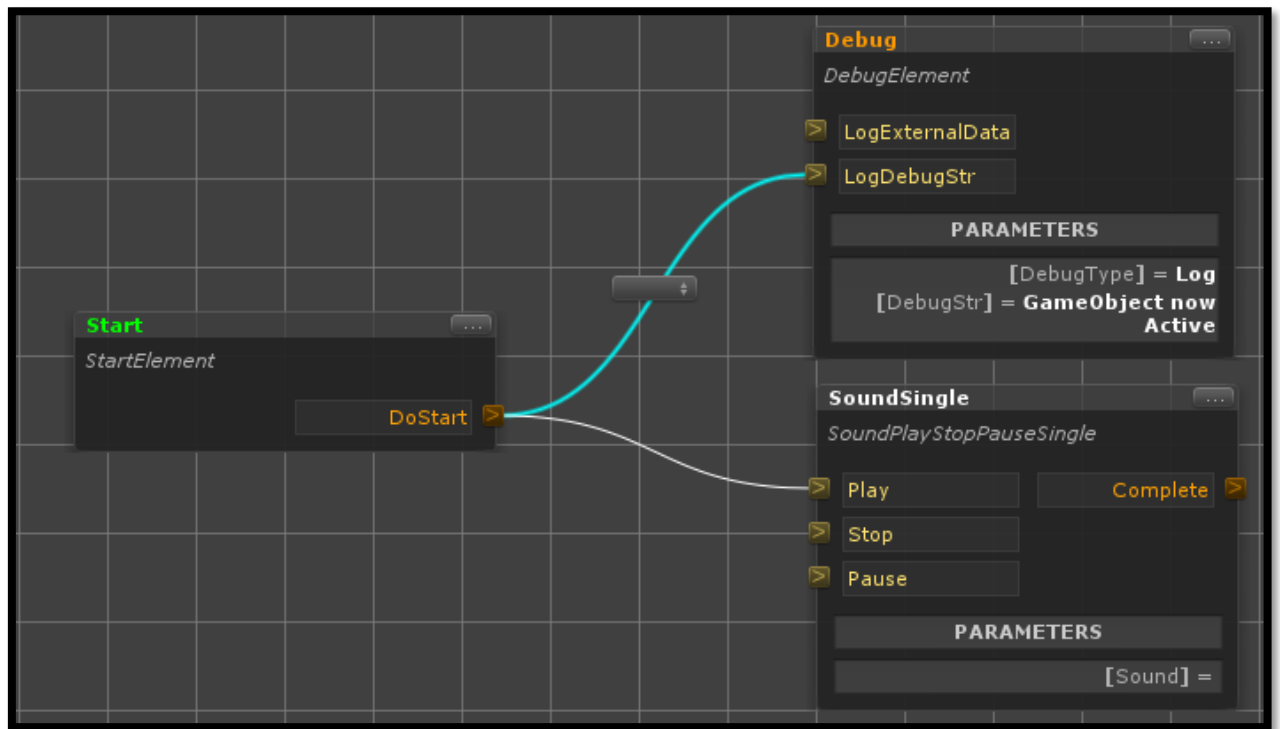
When you move the cursor to the input point, in case the link can be established, it will be highlighted in green, otherwise it will be red (see the figure below).



To create a link, you must release the mouse button when the line is highlighted in green.

Working with a link

Once the link has been established, we can conduct some operations with it. If we need to delete a link, we need to select it, just click the left mouse button on it (see the picture below).



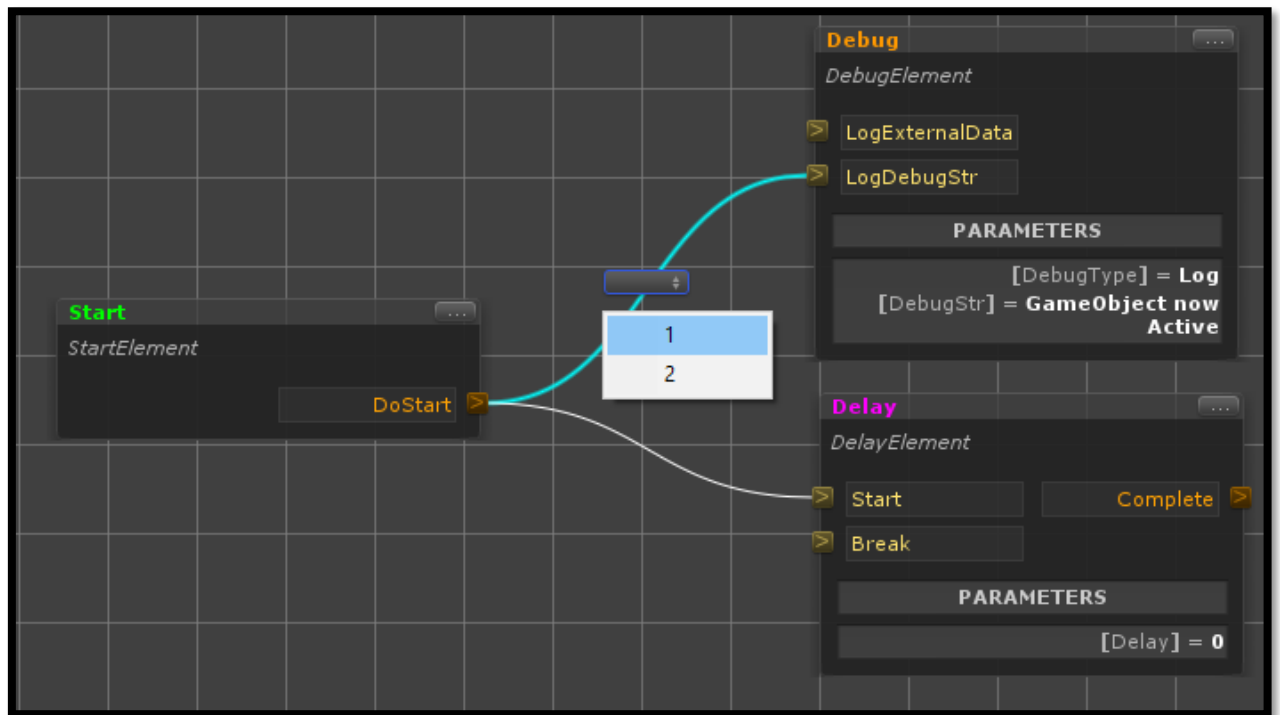
After the link is selected, you can either press the **Del** key to delete it, or use the context menu that is called by the right mouse button.

To select several links, perform the above manipulations, but with the **Ctrl** key pressed.

Note: if you delete an element, all links to it are automatically deleted.

Setting the links order

During the creation of a diagram, often, one output point of an element can be connected to several input points of other elements. In this case, it may be necessary to establish the order of execution of the links to ensure the correct functioning of the logic. To do this, select the sequence number in the drop-down menu that corresponds to the selected link (see the figure below).



Note: by default, all links have a sequence number of execution equal to one.

Diagram recovery

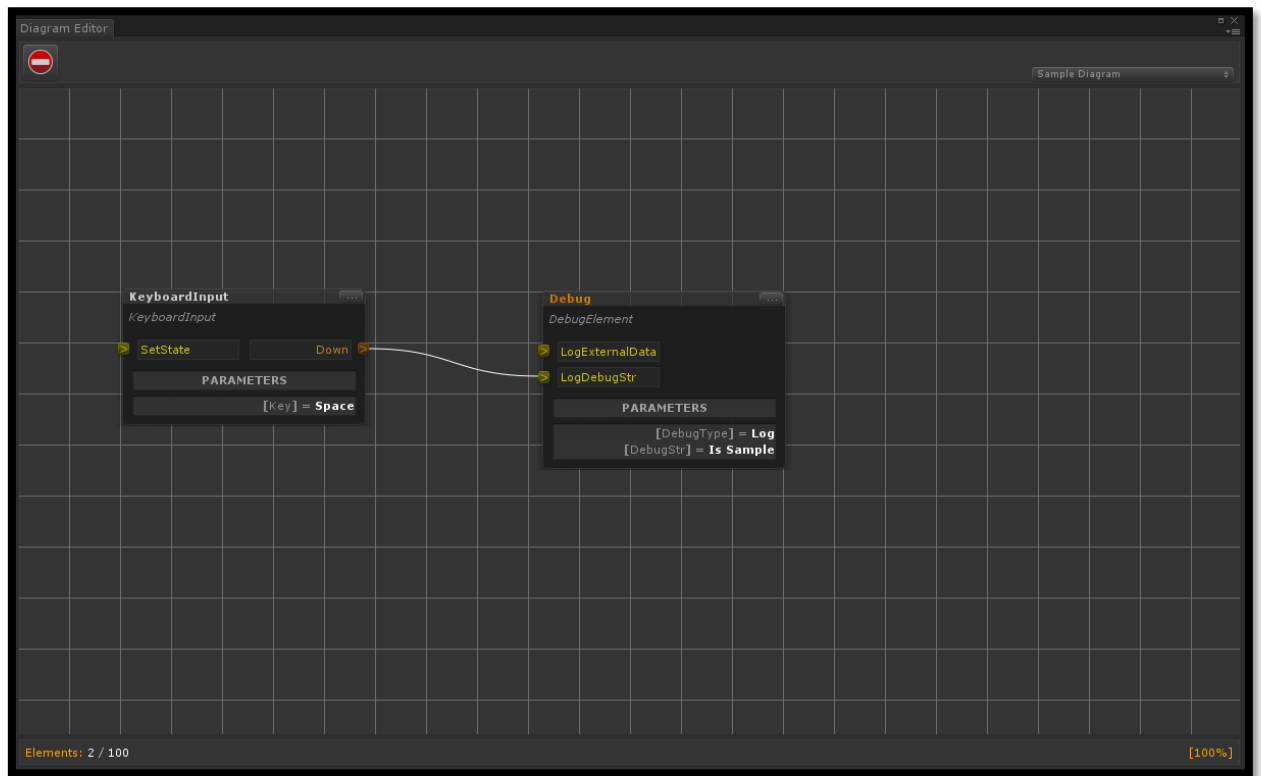
During the work on the application, there are possible force majeure situations of the loss of scene data. Since the diagrams are part of the scene, they will automatically be lost, but they can be recovered. There are two options to restore them:

1. Since the diagrams are files, you can add them to the scene again manually.
2. Use automatic diagram recovery, with the button 

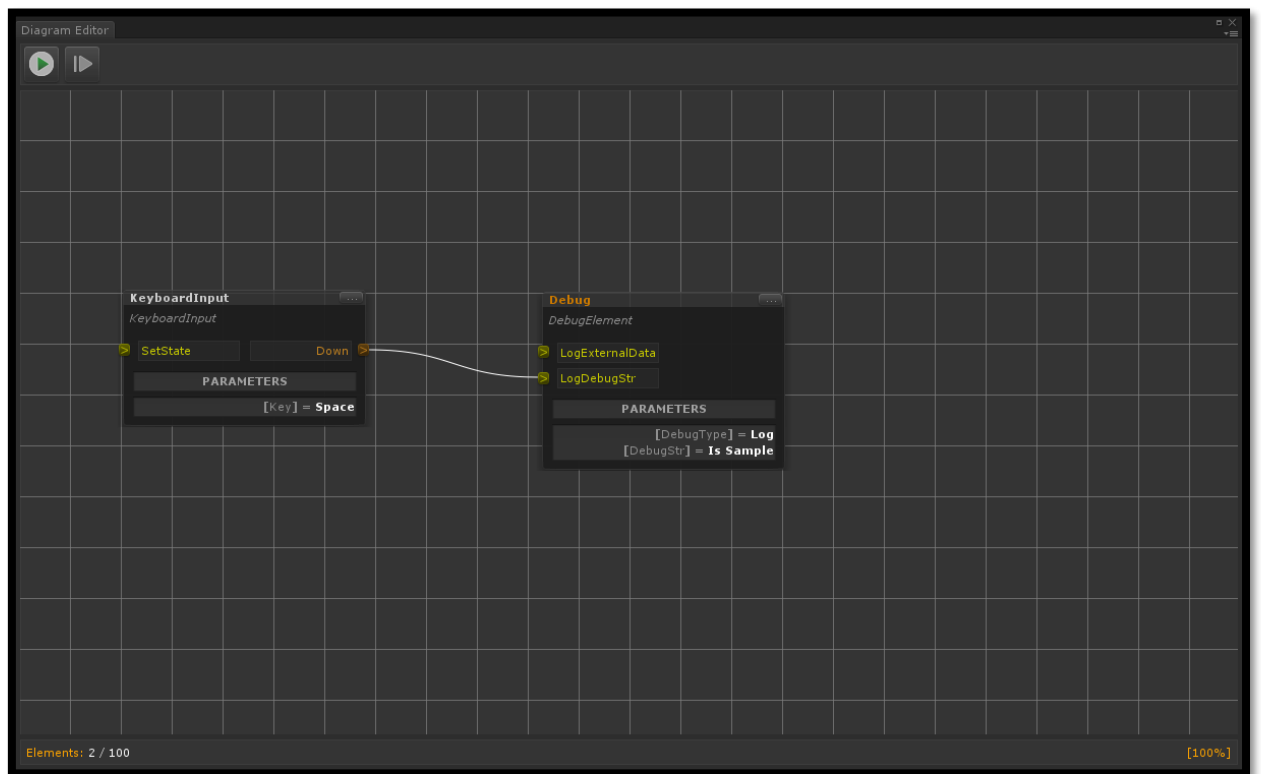
However, while restoring diagrams, there is one nuance that must be taken into account - if the diagram contained variables with links to Unity objects, then all of them will be lost, **so create backups more often!**

Debugging a diagram

To debug a diagram, you need to start the application in editor mode, after which the editor view will change to this (see the figure below).



Here the button puts a stop point in the diagram. After it is clicked, the editor goes into a mode of step-by-step calling of links.



Здесь:

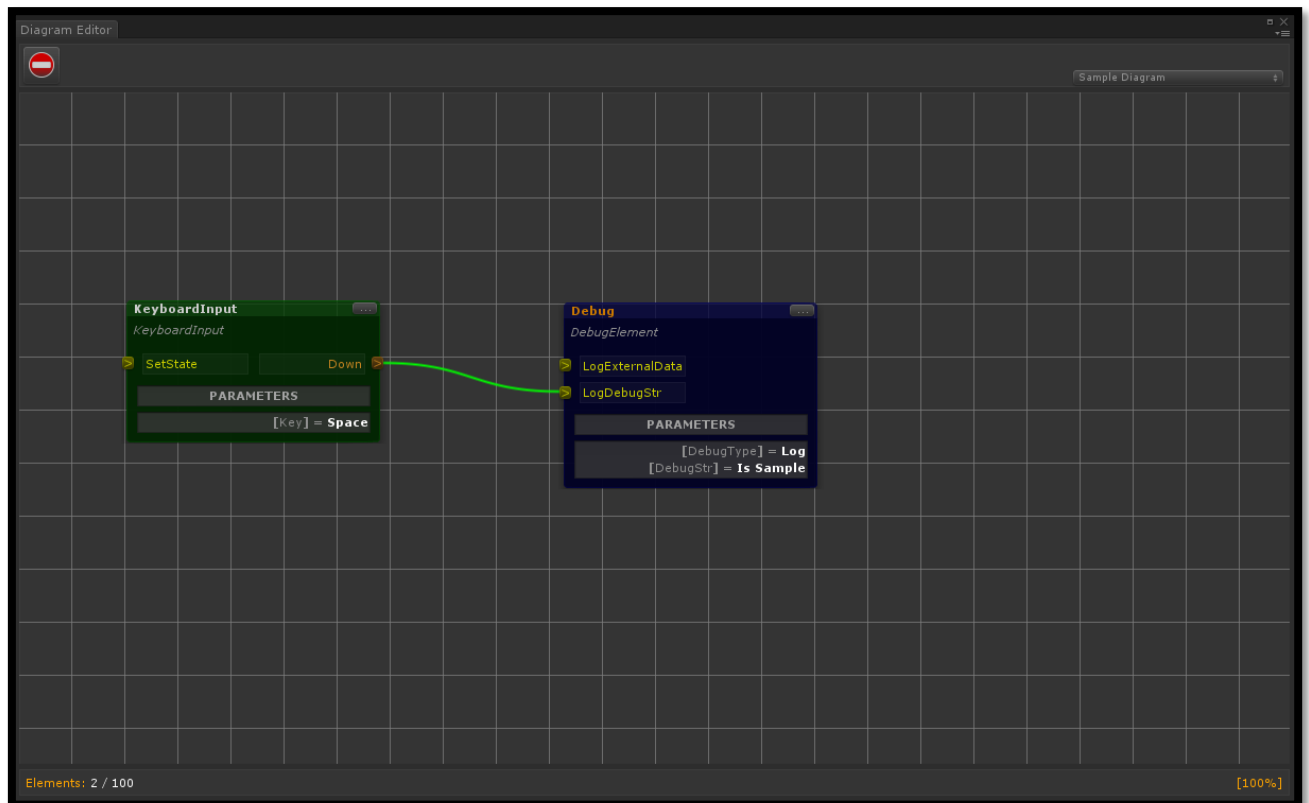


- this button switches execution to the normal mode (without step-by-step debugging)



- this button executes the next step in the chain of link calls.

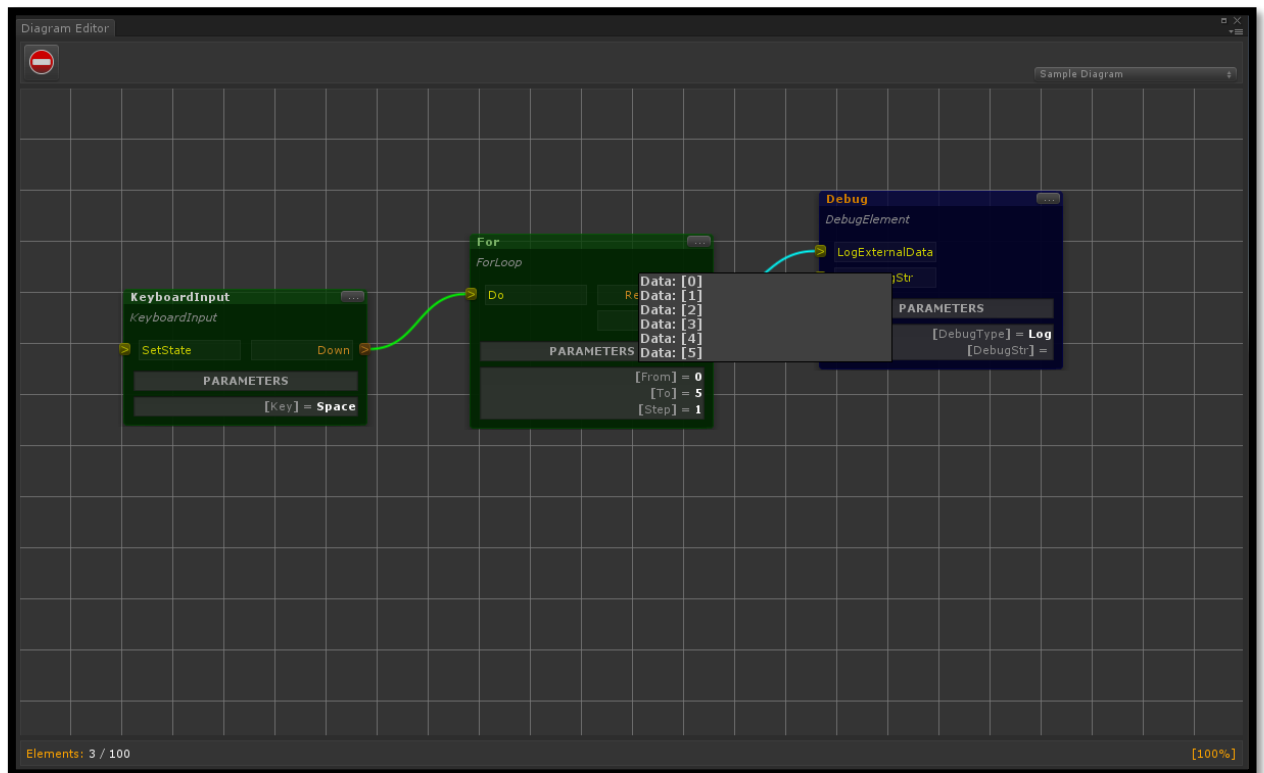
During the execution of the diagram, elements and links are highlighted, depending on the current state.



Here:

- The elements that have completed their work are marked with a **green** color.
- The **blue** color highlights the elements that are currently working, or were the last ones in the call chain.
- Blinking links show that their call is taking place at a given moment.

To see the data that the element has transmitted over the link to another element, it is necessary to highlight the link. Also, if the data that is displayed in the element parameters changes during the operation of the element, these changes will be reflected in the diagram editor.



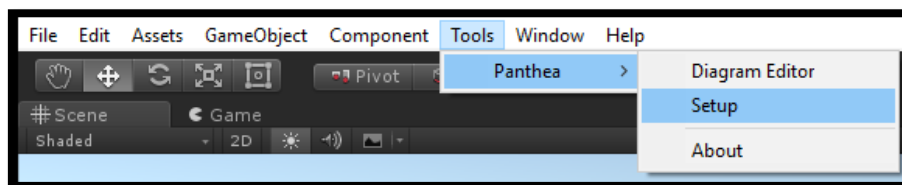
In addition to the above, there are some limitations of the editor in debug mode:

1. It's not possible to add elements during debugging.
2. It's not possible to change the parameters of an element.
3. It's not possible to delete links and elements.
4. Changing elements positions will not be saved when you exit application launch mode.

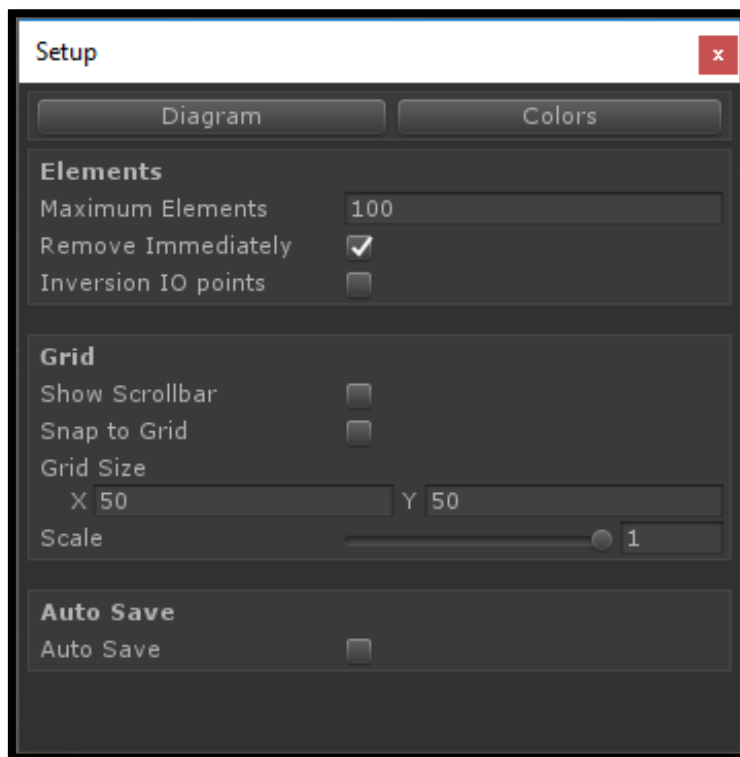
Also, you can use **Breakpoint** element for pausing the execution at specified place. Consult the elements reference for details.

Settings

To open the settings window, use the menu item as shown in the figure.



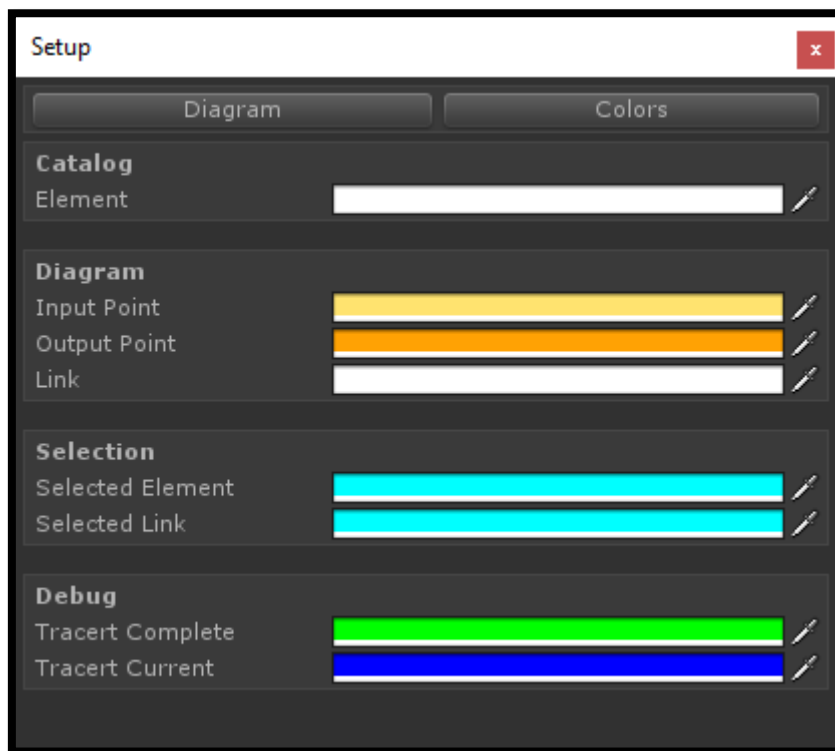
In the settings window, there are two sections. The first is the settings section for the diagram editor, the second is for adjusting the colors. The first one in the picture below.



Here:

- **Maximum Elements** – parameter that limits the number of elements per diagram (to control the size of the diagrams). If set to 0, there are no restrictions.
- **Remove Immediately** – parameter that is responsible for removing elements without confirmation.
- **Inversion IO Points** – parameter responsible for inversion of the input and output points by default (when creating elements).
- **Show Scrollbar** – parameter, responsible for displaying of the scrollbars in the diagram editor window.
- **Snap to Grid** – parameter that is responsible for binding to the grid of elements when they are moved.
- **Grid Size** – grid step.
- **Auto Save** – parameter that is responsible for the automatic saving of the diagram.
- **Auto Save Period** – parameter that defines the period in seconds with which the diagram is automatically saved.

The second section of the settings is devoted to color settings.

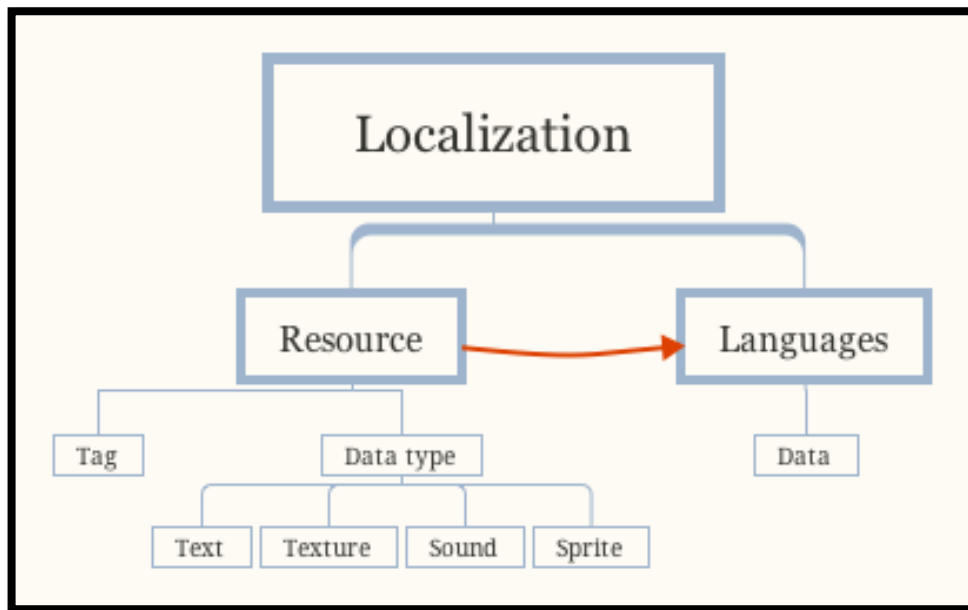


Здесь:

- **Element** – default element header color.
- **Input Point** – input point color.
- **Output Point** – output point color.
- **Link** – color of a link between elements.
- **Selected Element** – selected element color.
- **Selected Link** – selected link color.
- **Tracert Complete** – the color of the element that finished its work.
- **Tracert Current** – the color of the element that is currently executing its work or which was the last in the call chain.

Localization

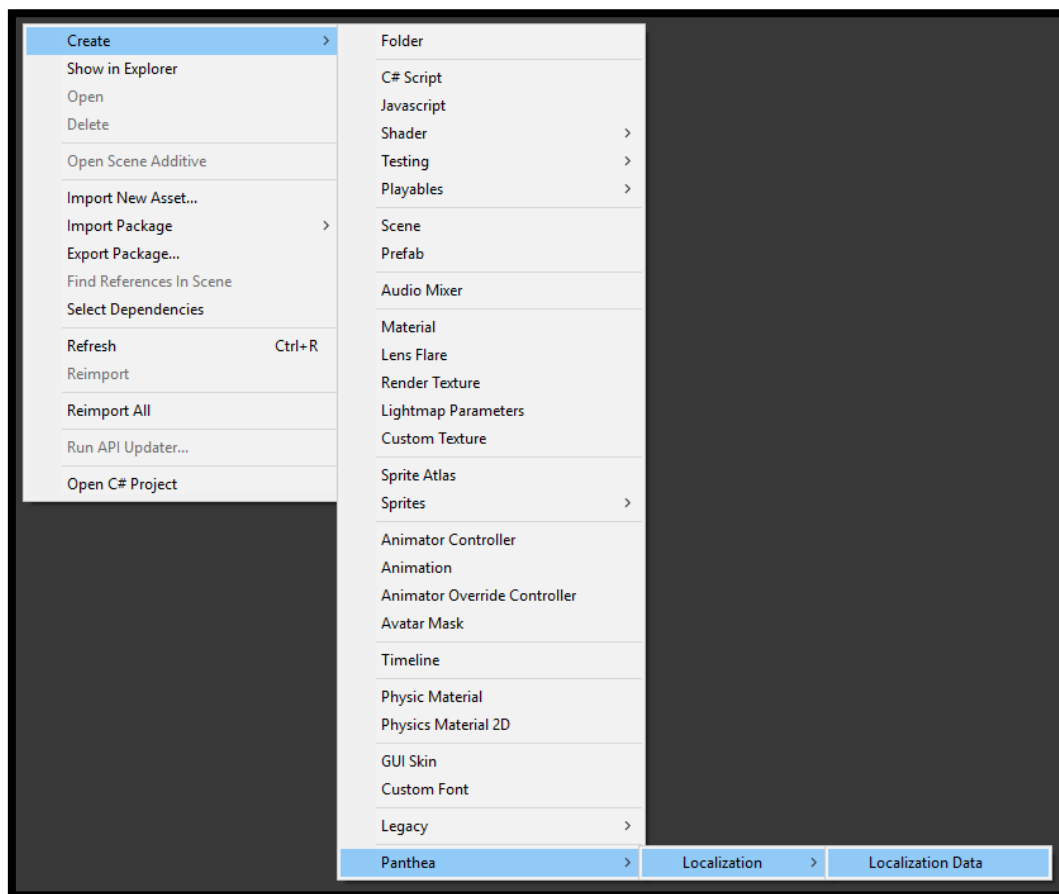
For a detailed description of the elements related to localization, see the corresponding documentation file. In this section, there are the general principles, as well as knowledge about how to create data for these elements. Below is a figure showing the scheme that defines the localization system.



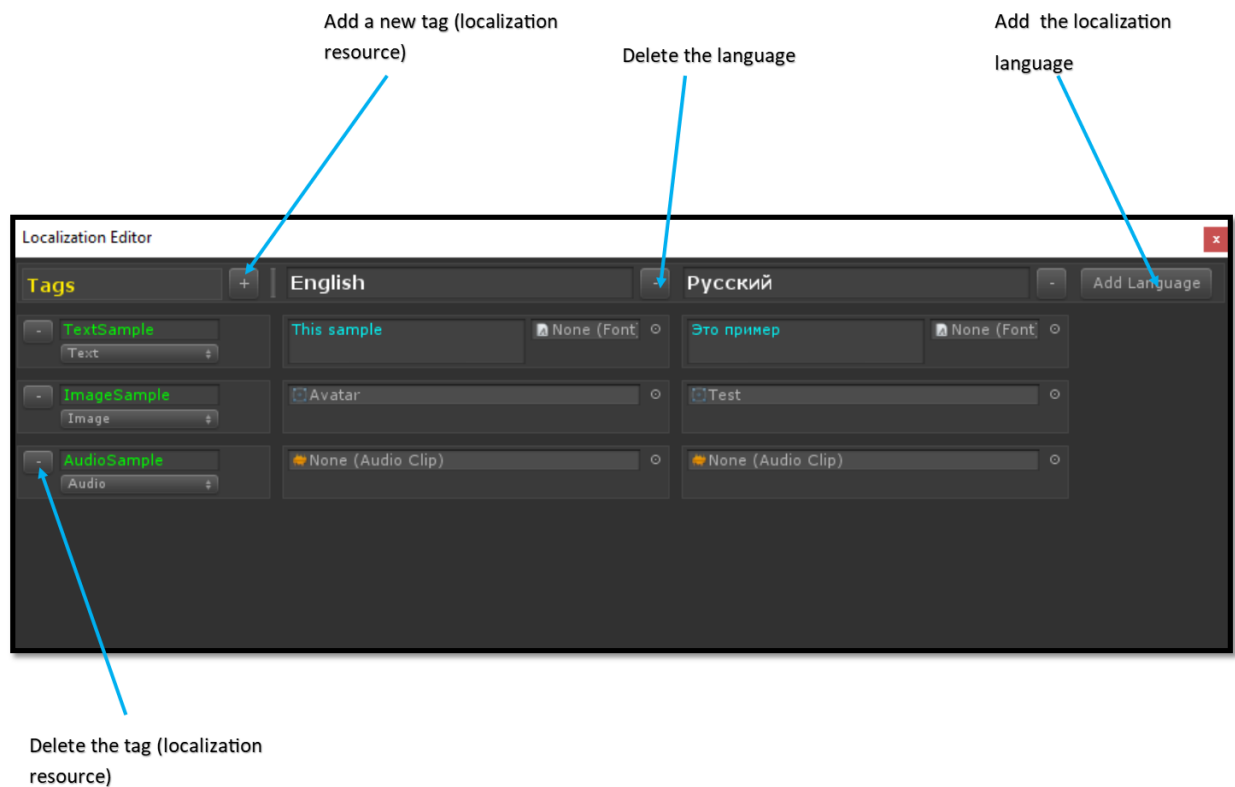
Here:

- A resource is a definition for localization, each resource has a tag (identifier) and a data type that corresponds to it.
- Languages are a set of specific data that are filled in for each of the localization resources.
- The localization system is built with the global events used, so the language is changed in real time.

Here is how the localization data file is created (see the figure below).



Using this menu item, you can create the necessary file, after which you can start filling it, for which you need to open a specialized editor. This editor is accessible by the button, which is displayed in the inspector when selecting a file with localization data. This editor is shown in the figure below.



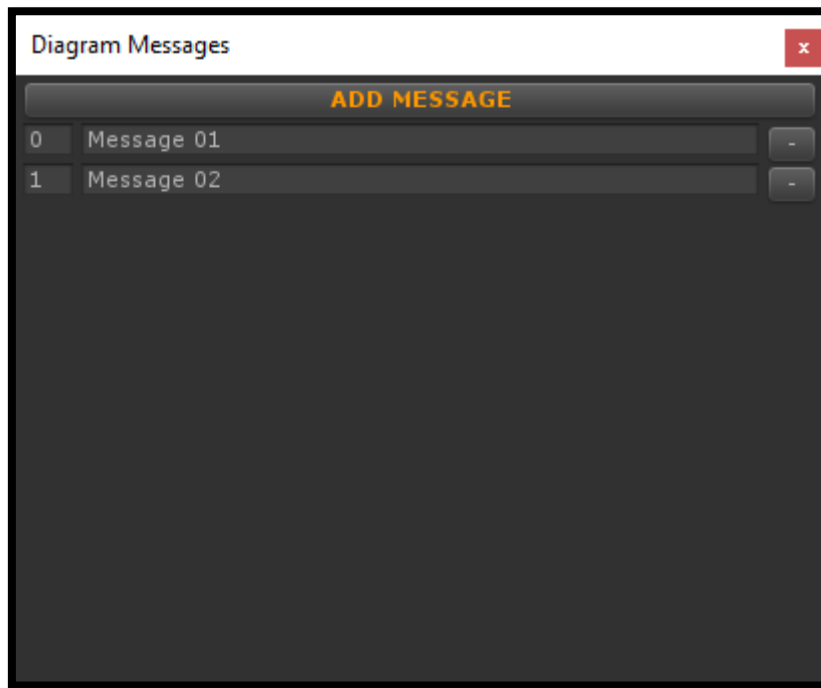
Here:

- If you delete a localization tag (resource), all the data corresponding to it in all languages is deleted.
- When you delete a language, only the data associated with it is deleted
- When adding a new tag, if the languages already exist, you have to specify the new data for each of them.
- The name of the languages is used to generate lists of drop-down menus.

Global messages

Panthea VS uses a system of global events. The complex part in which you can define these events and set the required data set is available for programmers through the API. In the diagram editor, a simplified version is used, which is an add-on for a more general version.

Global messages are events that are sent to all subscribers to them, if certain conditions are met (for example, pressing a key). These messages may transmit data, or may be of a notifying nature. In both cases, to use this system in the diagram editor, the set of messages should be created. This can be done in the messages window, that can be opened by pressing  button in the main menu. The window is shown on the figure below.



Note: all messages in the diagram editor are identified by an integer value, but in the elements, that work with them, they are represented in a string form. This is done to be able to delete messages without consequences.

Sound

In the Panthea VS system, there are several elements responsible for working with sound. These elements can play, stop and pause various sound sources. If you look at the elements, you will notice that they all have a Channel parameter. A channel is a group of sounds that are played with certain conditions, rules and parameters. In this system, the definition of the channel is done on the base of AudioManagerGroup. If this parameter is not set in the sound source, the sound is automatically placed in the Default channel.