



MAGELLAN

a Multimodal Authoring and
Gaming Environment for
Location-based collaborative
Adventures



Handbook for Creating Location-based games using the Beta Version of the MAGELLAN Authoring Tool

Version 1.0

Authors:

Petros Lameris, Despina Anastasiadou
Serious Games Institute, Coventry University

Artwork:

King Bagasina
Serious Games Institute, Coventry University

© Copyright by the **MAGELLAN** consortium, 2013-2017 **MAGELLAN** is a project that has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 611526. Please see www.magellanproject.eu for more information. **DISCLAIMER:** This document contains material, which is the copyright of the MAGELLAN consortium members and the European Commission, and may not be reproduced or copied without permission, except as mandated by the European Commission Grant Agreement no 611526 for reviewing and dissemination purposes. The information contained in this document is provided by the copyright holders "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the members of the MAGELLAN collaboration, including the copyright holders, or the European Commission be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of the information contained in this document, even if advised of the possibility of such damage.

Contents

Contents	2
Introduction	3
Chapter 1: Scenario level.....	4
Magellan Application User Interface	4
Edition tool.....	5
Asset import.....	6
Entities, activities, actions, conditions and groups	7
Scenario level toolbar overview	7
Basic Entities	7
Activities	8
Question	8
Select Role	9
Media player.....	10
Message.....	11
Weather.....	11
3D activity.....	12
Augmented reality.....	13
HTML activity.....	13
QML activity.....	13
Actions	14
Conditions	14
Group activities	15
Chapter 2: Behavioural level	16
Entity browser.....	18
The entity editor	20
Creating a 3D scenario	22
Importing files to the 3D scene	24
Chapter 3: Visual scripting.....	26
Lua tasks	26
Single Attributes	27
Calling entity methods.....	27
Synchronisation tasks	27
Compound tasks	28
Task library.....	28
GUI overlays.....	29
Conclusions	31



Introduction

The MAGELLAN project is a 4 year project aiming at designing, developing and implementing an authoring gaming platform which introduces innovative processes and methods for designing location-based games. Visual authoring principles, natural user interfaces and the latest interactive mobile and geo-location technologies, for single player and multi-player games, have been adopted as means to immerse players into a fun and educative gaming experience.

MAGELLAN¹ delivers a gaming authoring environment based on visual and authoring principles in order to help novice and experienced users to cost-effectively, swiftly and seamlessly to author and publish individual and collaborative authoring experiences. The project delivers also a scalable web-platform featuring social networking means and supporting search, retrieve, share and execute location-based games. Furthermore, Magellan produces a series of guides for authors of designing and implementing their own location-based experiences that will help them familiarise with the features of the Magellan Authoring Platform (MAT) thus serve as a reference for creative people to use the MAT in their own settings and context.

Aim of this 2nd handbook version is to facilitate end user's efforts to understand the overarching principles and practical features of the (Pre)- Beta MAT release. Specifically, it aims to present activities, actions, conditions, groups and entities used in conjunction for creating a location-driven experience based on the 3 levels of authoring: **Scenario, Behavioural and Visual Scripting**. It also aims to offer support and guide in terms of how these tools and features can be combined into practice while using the tool for each different level.

The 2nd version of the handbook describes the different levels of authoring in associated chapters. Chapter 1 describes the scenario level along with all basic activities (i.e. activities that were already included in the Alpha release along with the activities that have just been introduced in the beta MAT). A brief introduction on the Magellan Application User Interface is also provided particularly focused on the editing / controlling and scenario level features. Chapter 2 describes the behavioural level features, explaining in depth the entity browser, for modifying and manipulating 3D objects, creating paths, zones, and placing cameras. Chapter 3 presents the scripting level instantiated with the synapse editor. We further describe script tasks, synchronisation tasks, compound tasks, system tasks and task cloning. We also explain how the task library is accessed, searched and initiated as well as adding customised tasks. Lua and QML coding is then described for creating complex behaviours.

This is the second version of the Magellan Handbook. Iterative updated versions of training content in the form of a handbook will be produced omnipresent to the continuous development of the MAT for helping end users and practitioners alike to use future versions of the MAT (beta and final) during workshops and pilots across Europe.

¹ <http://magellan.diginext.fr/>



Chapter 1: Scenario level

Chapter 1 provides a discussion on, and analysis of, the scenario authoring level for helping authors to plan, design and implement certain aspects or features of their games such as scenario logic, the geo-location of the activities, the creation of simple entities and the configuration of basic parameters.

Magellan Application User Interface

Before getting familiarised with the scenario level features, it is important to understand the Magellan Application User Interface. The user interface consists of layouts and views that can be edited accordingly as to have the optimal environment for accessing the different editors, browsers and cameras necessary to develop the location-driven experience.

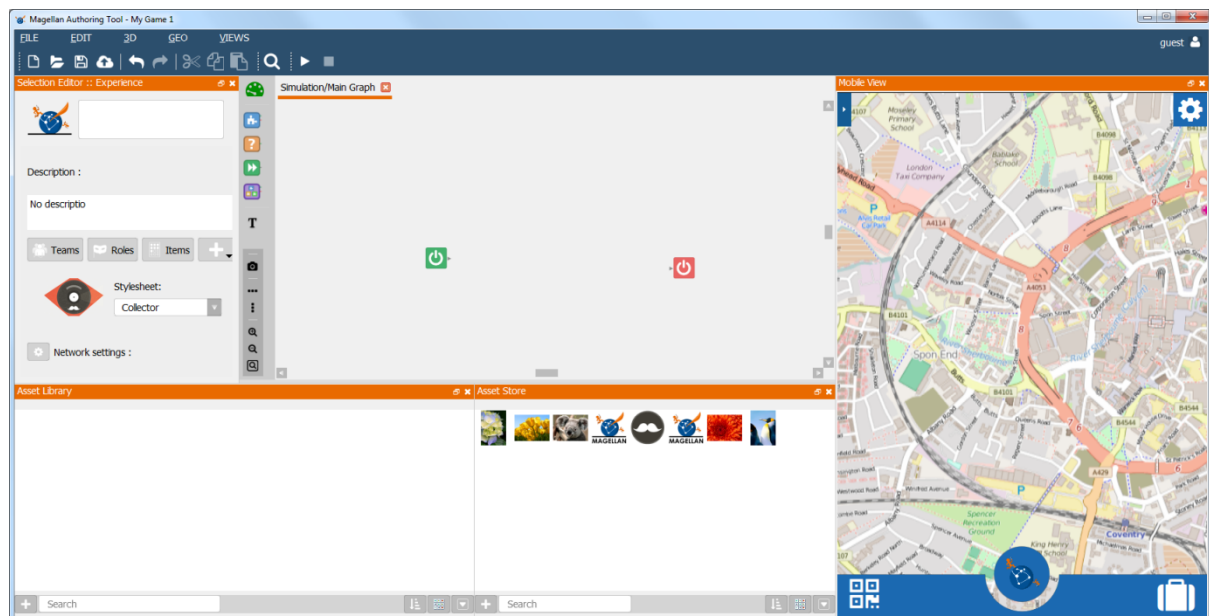


Figure 1: The Magellan Application User Interface

On the blue-coloured ribbon at the top, there are five menus: FILE, EDIT, 3D, GEO and VIEWS. Clicking on the **FILE** menu, you can create a new project (NEW), open an existing project (OPEN), save (SAVE) and save us (SAVE US) publish your game on the Magellan Portal Servers (PUBLISH), close the project (CLOSE), view a summary of your project (i.e. project name, file name, folder, number of entities, and total project size). Clicking on the **EDIT** menu, you can undo or redo the style sheet of the simulation, you can cut, copy and paste, delete or find. Clicking on the **3D** menu, you can select, manipulate and edit the pivot of a 3D object. You can also detach, pan, zoom, walk, orbit and zoom. The 3D menu is particularly useful for the 3D object editing and manipulation that it is going to be discussed in Chapter 2. Clicking on the **GEO** menu, you are able to detach a geo layer and a geo element. Clicking on the **VIEWS** menu, you are able to edit the layout, set it to default or save your own. You can also lock the layout, debug tools and select which of the views and editors you like to appear on your screen.

Edition tool



Figure 6: Edition tools

- | | |
|-----------------|-----------------------------|
| 1. New project | 6. Redo |
| 2. Open project | 7. Cut the selection |
| 3. Save project | 8. Copy the selection |
| 4. Publish game | 9. Paste the selection |
| 5. Undo | 10. Show the Synapse Editor |

Figure 2: Edition tools

Use the “play” button (1) to start simulation. Use the “stop” button (2) to stop the simulation. The ‘play’ button becomes the ‘stop’ button when the simulation starts. **NB:** To continue designing your game experience you need to press the ‘stop’ button.



Figure 3: Playback control tools

At the **selection editor**, you can edit the name and the description of your game, choose the teams, roles, items and add (+) an NPC, a profile, events and QR codes. You can also edit network settings.

In the **scenario level toolbar**, you can switch from the scenario to the behavioural level, by clicking the green icon; you can initiate a new activity by clicking the blue icon and then drag and drop; you can initiate a new condition by clicking the orange button and then drag and drop; you can initiate a new action by clicking the green button and then drag and drop, you can initiate a new group activity by clicking and then drag and dropping the purple icon. Also you can click on the **Text (T)** icon and you can insert some textual feedback that will help you to do certain tasks while you are designing your game. The text annotation feature is not executable and it is only for annotating design ideas, concepts, and sequences of features you wish to use for designing your game. It is also a collaborative tool for other game authors to provide ideas, thoughts and comments on your designs. You can also **capture a view**, enter a scale and then select an image from your file menu. By default, it opens in **Program Files>Diginext>Magellan>Bin**. You can zoom in or out by clicking the icon at the bottom of the scenario level toolbar.

The middle screen, **the scenario editor**, is similar to the one presented with the Alpha version, where the games authoring process for all levels (scenario, behavioural and scripting) is instantiated.

The mobile view shows the designated map that has been selected, the zones that have been created and starting/ finishing points of a location. In the mobile view emulator you can also see a preview of your game as it would appear on the mobile device. This could serve for testing and debugging purposes.

The asset library stores the content you would like to include in your game. You can store all of your images, audios, videos and 3D models by drag and dropping them from your file manager to the asset library. You can also use assets directly from the **asset store** panel. Once you are logged in to the asset store all assets stored in your Magellan portal profile, will be visualised and available to use from the asset store locally. To log in or sign up to the asset store, click the icon on the icon at the right-hand corner above the map, and you will be prompted to log in or sign up.

Asset import

The asset library lists all the assets that the author wishes to include in the game scenario. The asset store mirrors the online content included on the Magellan portal and available for download by the author.

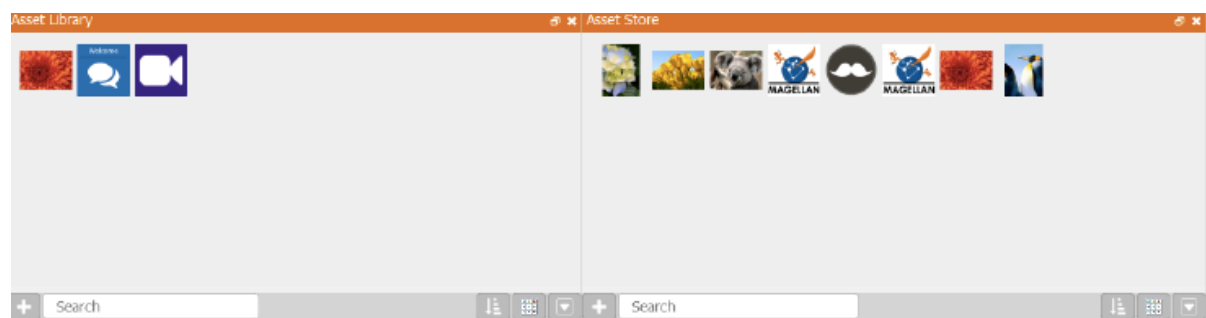


Figure 4: Asset library and asset store

To import an asset from the file system, just drag and drop the corresponding file onto the asset library. Multiple files can be selected simultaneously by pressing the alt key.

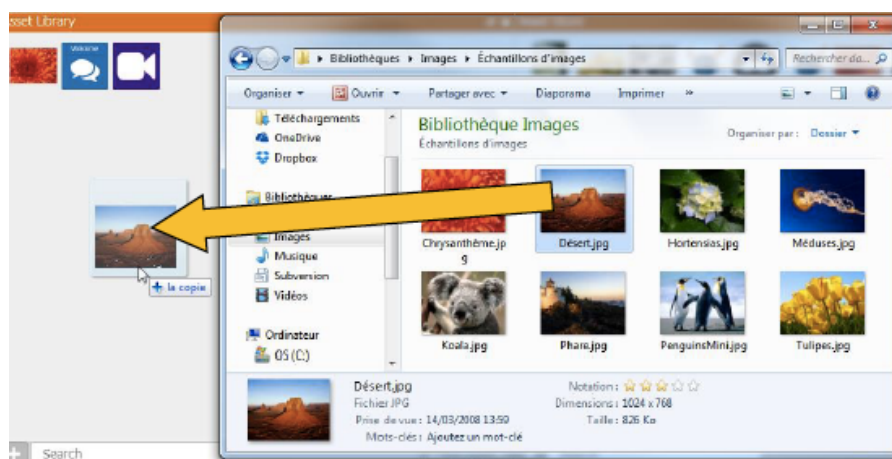




Figure 5: Importing a file to the asset library





Entities, activities, actions, conditions and groups



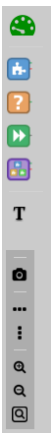
Activities, actions, conditions and groups are the overarching features for creating a location-based experience. They can be selected by clicking on the different icons as visualised in the **scenario level toolbar**. An **activity**  is the overarching part of the game design process and it engages the players into certain situations. An example might be to solve a puzzle, or to send a message, to select a question or to show a video. Activities are consisted of two outputs: “Success” and “Failure” triggered according to the activity result. Activities can be linked together to create an activity flow describing the game scenario.

A **condition**  performs different actions depending on whether a specified condition evaluates to true or false. Conditions can be related to graph, time, geo-localisation, event and participant.

An **action**  is somewhat an invisible process that runs into the background and cannot be visualised in the game as opposed to an activity which can be distinguished in the game play. At their simplest form, actions can be used to set an avatar, to drop an item, to play a sound or to raise an event.

A **group of activities**  consists of individual activities that can be triggered collectively. It is useful for game scenarios that run parallel activities to which the author wishes to disable some of those activities. Exit from the group activity can be performed by placing a condition or exiting automatically when the activity ends.

Scenario level toolbar overview



1. Authoring level scenario button (scenario, behaviour, script) switch button
2. Activity button to be clicked, dragged and dropped
3. Condition button
4. Action node
5. Group activity node
6. Graph annotation
7. Horizontal node alignment
8. Zoom the graph in
9. Zoom the graph out
10. Align the graph view

Basic Entities

An entity is an identifiable and separate object. An entity may be concrete like a person or participant it can be a male or female or it can be abstract like an inventory or a role. The basic entities on Magellan are:

- **Participant:** It is created when someone joins the game. The scenario is created for a number of participants that will join the game and interact with the features, characters, roles and teams.
- **Character:** It resembles a character in the game. It can be a **Non-Player Character (NPC)** which is controlled by the simulation or a **Player Character (PC)** which is controlled by the player. The behaviour of the characters can be edited from the behavioural level.



- **Team:** A team is a group of participants having common objectives and goals and are collaborating together in order to achieve those goals.
- **Item:** An item is a virtual object that has a value to a participant and character. Items are collectable and can be shared between and within teams.
- **Inventory:** An inventory is a collection of resources by a participant or character. Inventories can be selected, shared and assigned in **conditions**.
- **Role:** A role is assigned to a participant who belongs to a team associated with specific objectives. Roles can be assigned, edited and tested during the scenario design.
- **Attribute:** Is a sub-element of an entity. For example, it may be the location or the gender of a character, the colour of an object etc.
- **Events:** are the actions occurring when an entity is triggered. Typical events are **OnActivate** (when simulation starts) **OnClick** (when the participant clicks the mouse on the entity).
- **Methods:** represent the object's behaviour. It can be something as simple as "get location" or "Play" or "GetItem".
- **Graphs:** represent ways of representing certain designs. In a graph, events are graphically connected to **tasks** in order to trigger certain actions.
- **A task:** is a process that describes the purpose of the entity. For example, if the task specifies that you need to get an object, then the method should be 'GetanObject'.
- Entities can be accessed using the selection editor 🗑️

Activities

Activities can be accessed in two ways: (i) by clicking and then drag and dropping the blue icon on the scenario level toolbar or (ii) by right clicking your mouse in the scenario editor and then choose activities. There are 5 activities categories: (1) debug, (2) generic, (3) game, (4) 3D, and (5) basics.

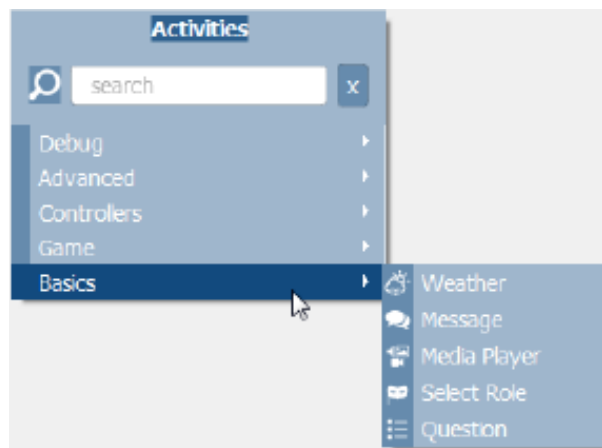


Figure 6: Activity categories

The basics activity includes activities that are considered to form the basic characteristics of a game such as: **a question, a QR code, select role, media player, message and weather.**

Question

A question activity is selected by following the path: Activities>basics>question. By posing a question, the player has to select the correct answer to a question that is specified as to proceed to the next activity. Question and associated answers, are provided via the selection editor once the

question activity is clicked. Authors can describe the question in the description text box, can pose the question at the question text box and can specify as many answers as they wish. To trigger the correct answer, you just connect the correct answer with the next activity / output and the wrong answers by other activities / outputs or by finishing the game.

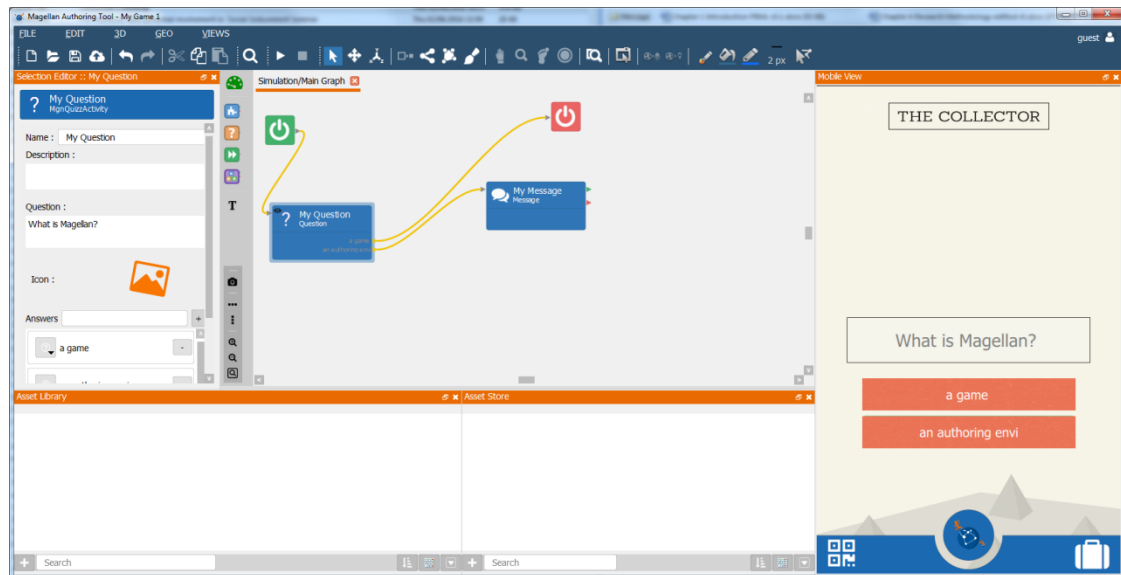


Figure 7: forming a question and linking it to outputs

Select Role

This activity is performed by a specific role. It is usually combined with an action “Give a Role” and/or with a role condition under controllers>conditions>participants>role. The teams and roles should be already formed from the scenario editor experience.

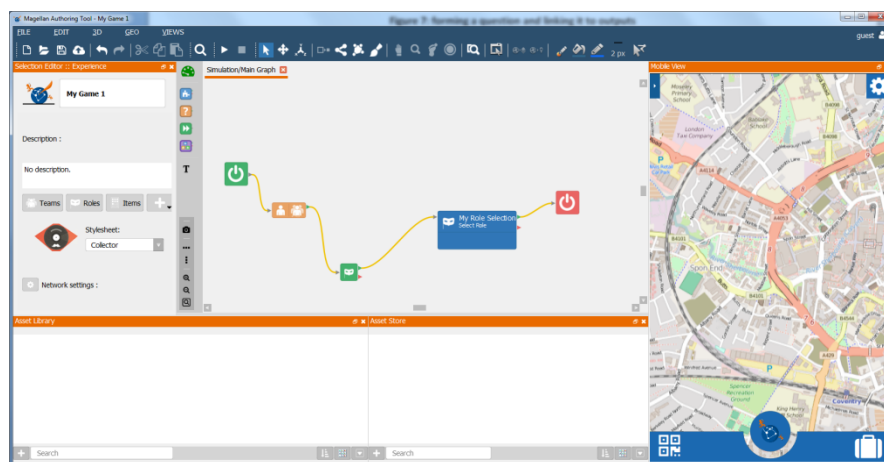


Figure 8: Selecting a role

NB: To create a team: From the selection editor, you click the team icon>enter team>click + button to store. You can create as many teams as you like. To create a **role**, click on the Roles>create new>click + button to store. To assign roles to teams go to teams>select team>list of role>+ button>select the role to assign to a team.



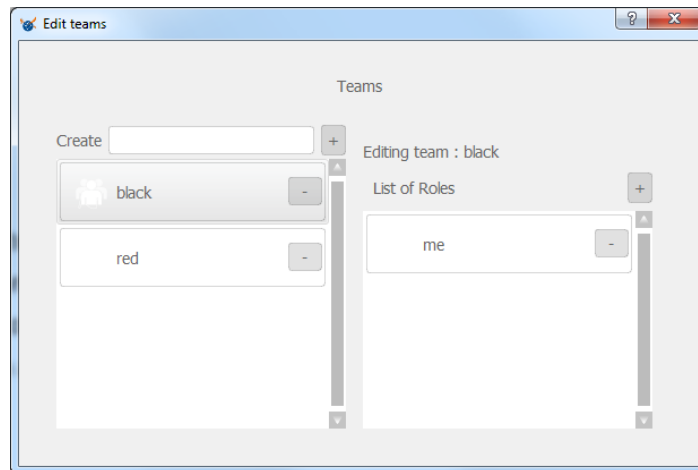


Figure 9: Creating a team

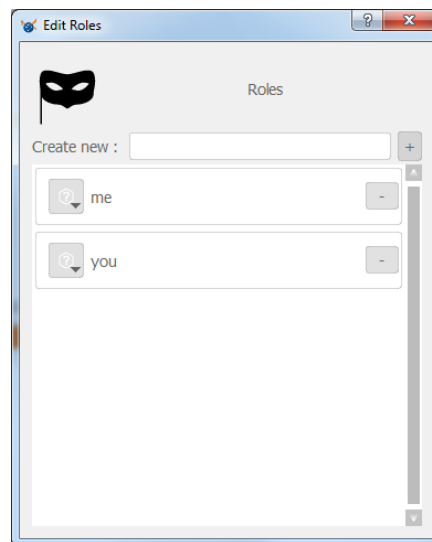


Figure 10: Creating a role

Media player

To create a media player activity, go to activities>basics>media player. With the media player activity, you can play content such as video with sound or only video. Simply drag and drop your media content to the asset input field from the selection editor or from the selection editor>asset>search>type the name of your content file. To delete a file simply delete the media player activity or click on the 'X' button in the asset field>search.



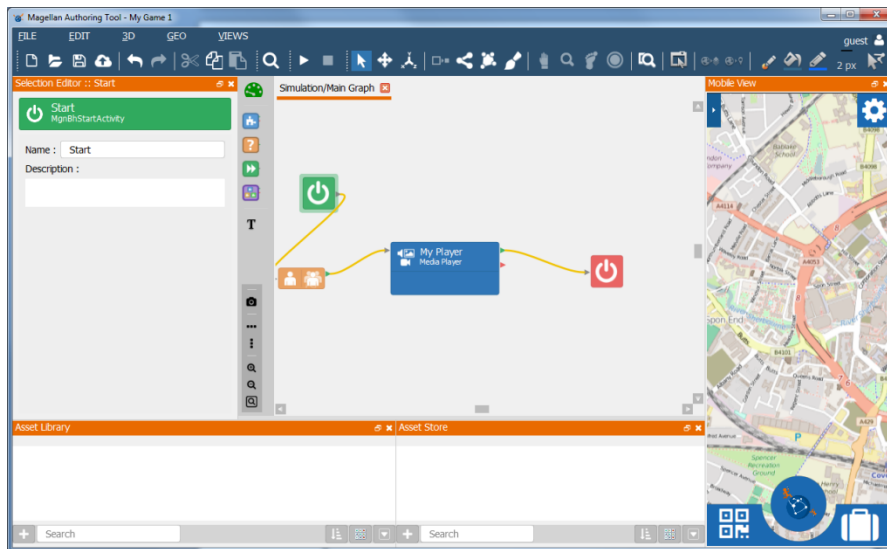


Figure 11: Creating a media player activity

Message

To create a message activity, go to `activities>basics>message`. It is an informative activity that can be placed when a specific goal has been achieved or for notifying the player for something like for giving feedback, receiving an icon or object or for completing a level. You can display a message with the form of an image or a text. Two optional buttons that denote success or failure can be included as well. The text and the image are added from the selection editor by inserting text in the description and message text boxes and by dragging and dropping the image to the associated image icons.

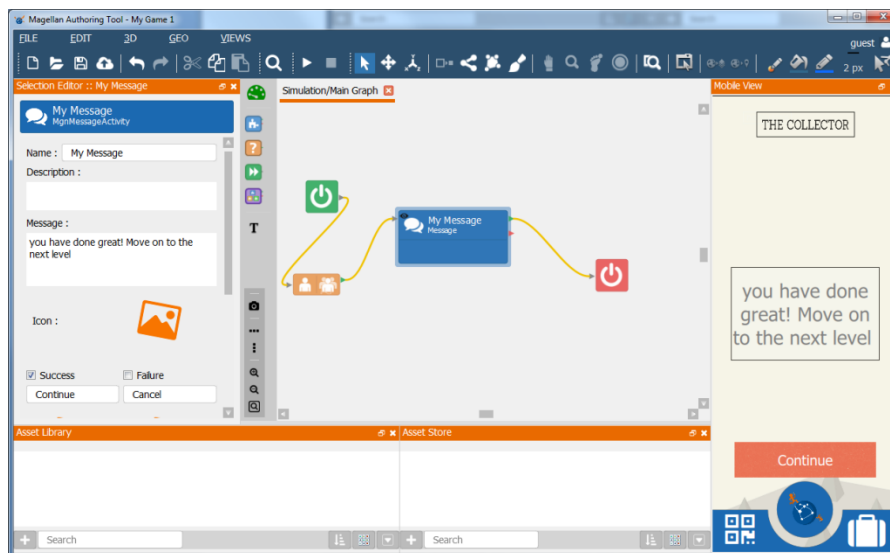


Figure 12: Creating a message activity

Weather

To create a weather activity, go to `activities>basics>weather`. It specifies the weather at a given location. The activity is connected to a weather server which fetches current weather conditions. You can enter a Weather's city from the city attributes.



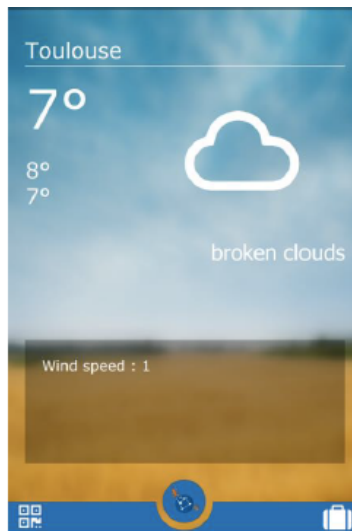


Figure 13: Creating a weather activity

3D activity

3D activities are key for displaying interactive content such as characters, environments, objects etc. 3D objects and their attributes are edited in the behavioural level. From the scenario level, you can simply initiate a 3D activity by going to activities>3D>3D activity. By clicking the 3D activity you can see the 3D scene at the mobile view. To import a 3D model, you simply drag and drop it from the asset library to the scene editor. You then select the 3D option and then appears on your 3D scene. You can edit its behaviour from the behavioural level which we are going to explain further in Chapter 2.

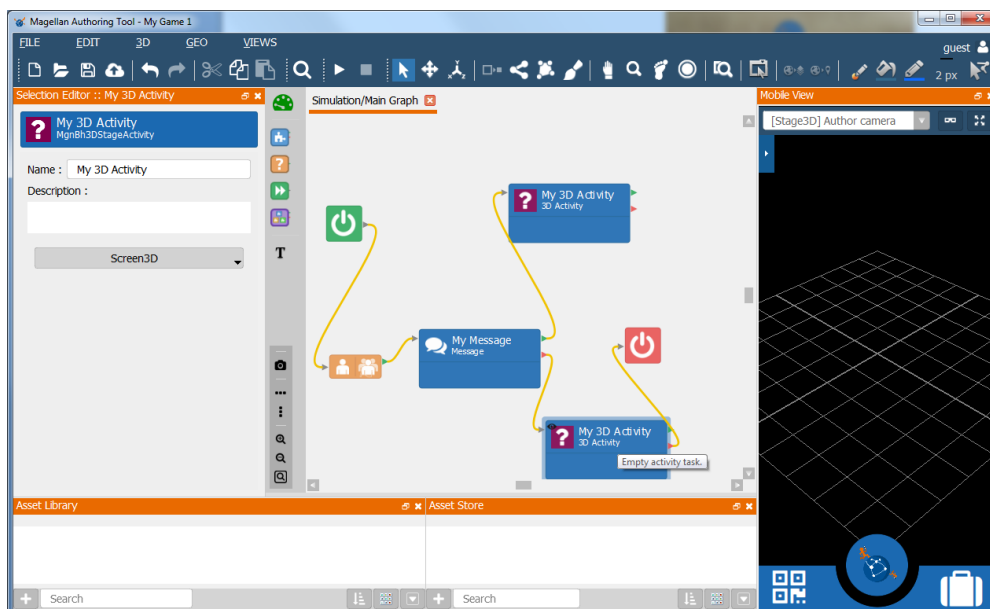


Figure 14: Creating a 3D activity



Augmented reality

To create an augmented reality activity, go to activities>3D>AR activity. AR can be rendered via a 2D trackable image or a 3D trackable model. The virtual camera adjusts accordingly and use it as a trackable reference. The user can add other 3D models for displaying them on the top of the video and animate them in the behavioural level of the MAT as a 3D activity.

HTML activity

To create an HTML activity, go to activities>generic>HTML activity. First you need to package the activity into a single directory. The directory should contain a file named index.html which is the main file of the directory. The folder can create other files by the index.html such as css files, resources, javascripts files etc. Then to trigger success or failure, you need to trigger the activity with the endwithFailure.html and endwithSuccess.html pages. Once the folder is ready is dragged and dropped in the asset attribute located in the selection editor.

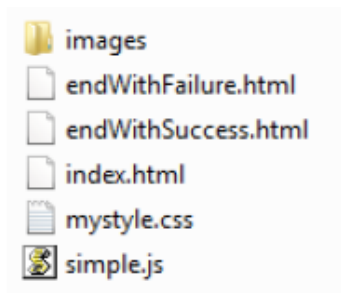


Figure 15: Creating a folder for initiating an HTML activity

QML activity

QML is a user interface mark-up language. It is used for designing user interface-centric applications. To create a QML activity, go to activities>generic>QML activity. A simple text editor where the QML code is dragged and dropped to the asset attribute.

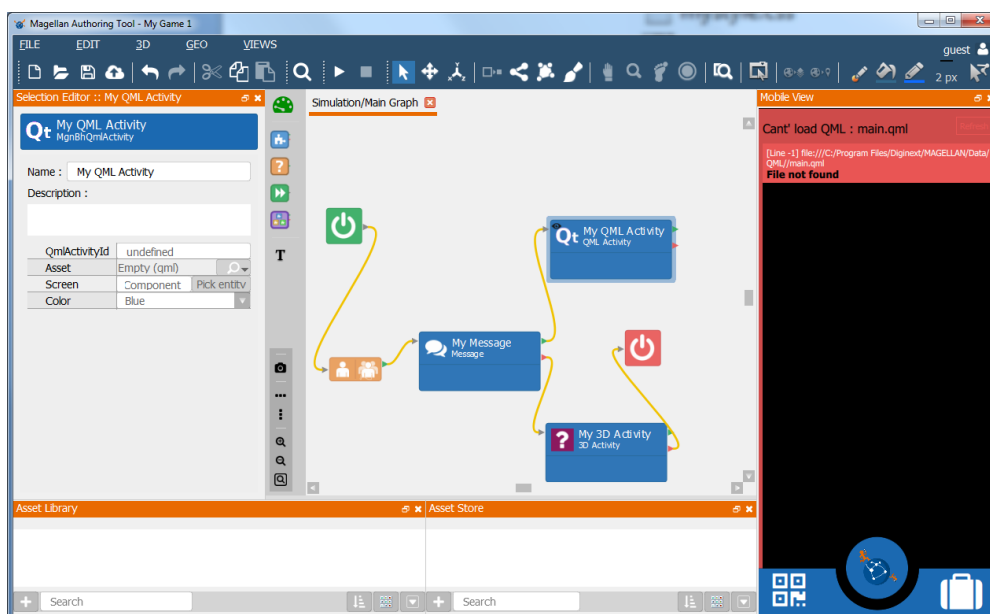



Figure 16: Create a QML activity

Actions

To insert an action you can either drag and drop the green icon  from the scenario level toolbar or you can right click you mouse and select actions from the menu. The actions that can be used are:

Give role: The action assigns a role to a participant

Set profile: The action sets a specific profile linked to an activity or condition

Raise event: An event is raised by the user. The event can be then combined with an event triggered condition in different locations specified in the scenario.

Read text: A text is read using audio. This is particularly useful for participants with special ways of learning or with disabilities.

Give item: The action can be used to pick an item from a location and add it to the participant inventory. If it is a multiplayer game, the item given will not be available to the inventory of the participant who provided it. It would be available in the inventory of the receiving participant.


Send analytics: This allows the author to send an HTTP request to a Google Analytics server in order to track progress during the scenario execution. The author account ID and type of metrics requested is all that is needed for the analytics to be configured. The send analytics action can be inserted at any point in the graph.

Play sound: The action plays a sound selected by the author when it is activated.

Drop item: An item is dropped in a specific map location.

Set Avatar: An avatar is set for executing actions specified by the user.

Conditions

The conditions have not seen any change from the alpha MAT version. To summarise, conditions are inserted from the scenario level toolbar  or from right click and then select controllers>conditions. There are 5 overarching conditions categories: Graph, Time, Geolocation, Event and Participant. Conditions are represented in the graph with orange nodes containing icons. Currently, the following conditions are implemented:

- On the participant properties:
 - The current participant is member of a given team
 - The current participant has an item in his inventory
 - The current participant has the given role
 - The current participant has the given identity
- On location:
 - The current participant is inside a region
 - The current participant is close to a hotspot
 - The whole team is inside a region
- On events:
 - A timeout event occurs
 - An event has been emitted
 - A QR Code has been flashed
- On groups
 - All the contained activities end successfully



- All the contained activities end with failure
- At least one activity fails
- At least one activity succeeded
- A given activity fails
- A given activity succeeded
- On Time
 - A timer condition is added to end the activity successfully
- On graph
 - The condition ends successfully with a graph

To create a condition, right click on the canvas of the logic graph editor and select controllers>conditions>type of condition: graph, time, geo-localization, event, and participant.

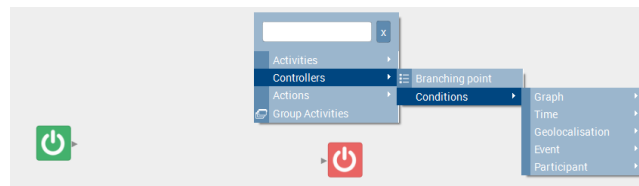


Figure 17: Creating a condition

When multiple conditions are defined in the same node, all the conditions have to be met to continue. It can be seen as an “AND” expression, i.e. “Condition 1 is satisfied **AND** Condition 2 is satisfied”. In order to simulate an “OR” expression, conditions have to be split in separated nodes (see following figure).

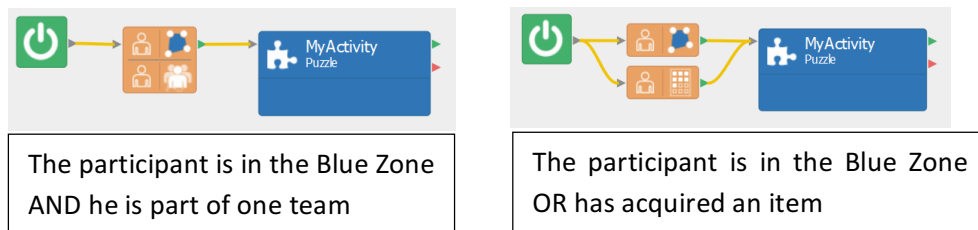



Figure 18: Condition (AND, OR)

Group activities

Group activities are available from the scenario level toolbar  and dragging and dropping the group activity or by right clicking and selecting the group activities from the menu. Group activities are defined as activities that can be executed collaboratively or competitively. Group activities can be used in parallel, in sequential order (i.e. one group activity at a time).

In the following example, on group activation, all unconnected nodes are activated, i.e. all the orange nodes in this example. Then, both conditions can be satisfied in a random order. If the participant first goes to the green zone, the activity01 will be triggered. If he goes to the blue zone, the activity02 will be triggered with no order between them. When the two conditions on the participant inventory are met, the whole group is terminated and the activity is successfully executed.



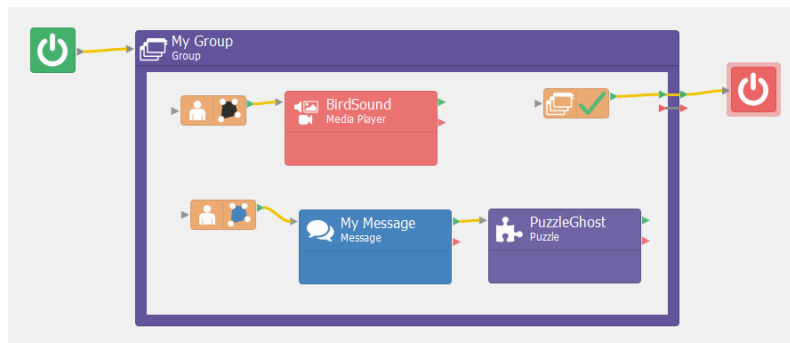


Figure 19: Creating a group activity

Chapter 2: Behavioural level

The behavioural level in Magellan essentially represents the advanced mode of authoring location based games necessary for incorporating more advanced features to the game’s authoring process. The Magellan represents a hierarchical organisation of entities that comprise **assets, entities and behaviours**.

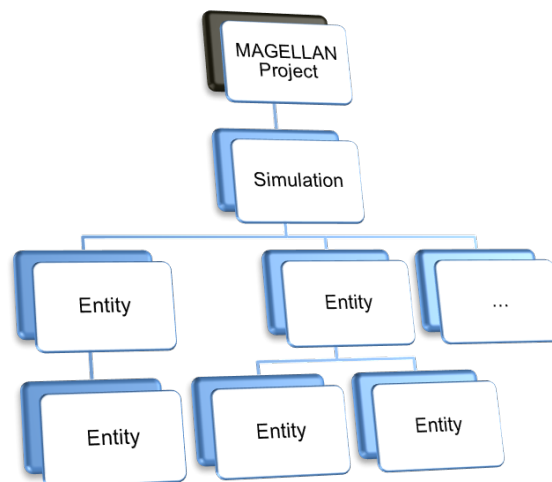


Figure 20: Hierarchical organisation of entities

A structured sequence of task is represented via graphs. Each entity has inner **events** triggered by e.g. OnMouseClicked on a 3D entity. It is possible to create your own events and trigger them dynamically. In the behavioural mode (click the green icon 🟢, at the scenario level toolbar) to enter the behavioural level 🟡. Then click on any activity (e.g. message) to customise it. The following screen appears.



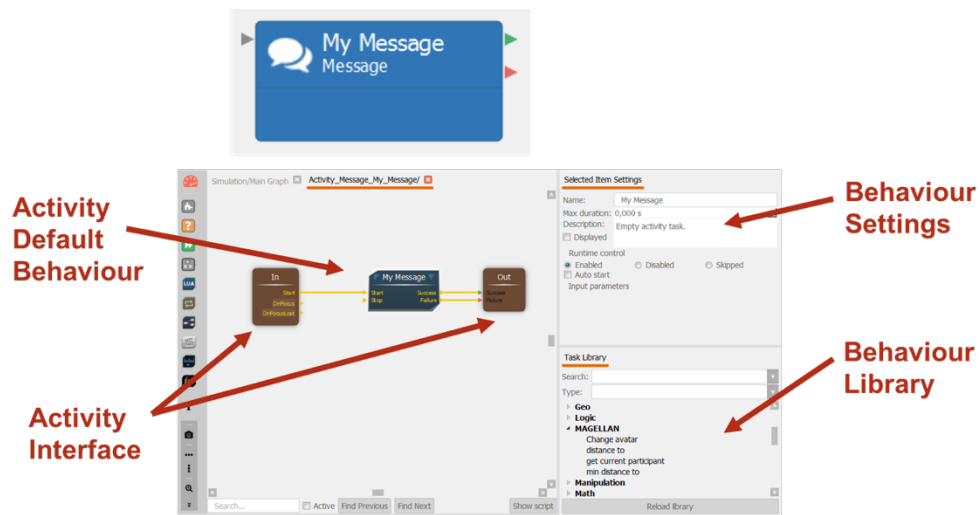


Figure 21: Behavioural level features

The task node is connected by default as it registers the activity to the system for implementing its default behaviour. Then the default behaviour is launched which triggers the success/failure outputs when necessary.

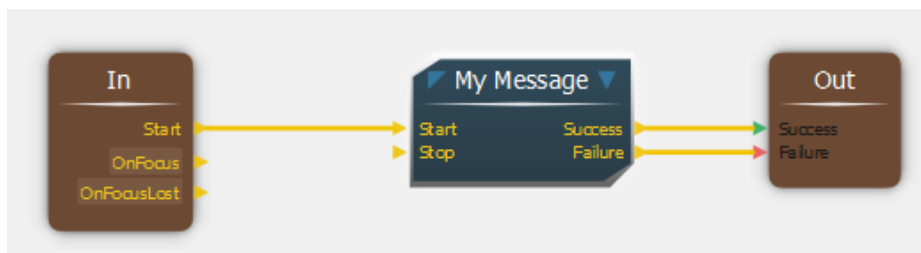


Figure 22: Activity default behaviour

The activities at the behavioural level can be customised by inserting and linking actions, conditions on activity start, success and/or failure. For example, in a message activity, we set a timer at startup to stop the activity after 30sec and play a sound on failure. If success, then give an item before quitting.




Figure 23: Adding simple actions and condition to an activity



Entity browser

The entity browser contains all the **entities** (in the entity library) and the entities of the current game are shown in the **project** tab. The entity browser is used to:

- Import entities from the library to the current project;
- Rename, delete and move entities in the project tree
- Import assets from the Explorer to the current project
- Browse and search entities

The entity browser is integrated into the behavioural level of the Magellan authoring environment. It is located on the left side in the behaviour level  of the Magellan main window. The entity browser is available when a project is opened.

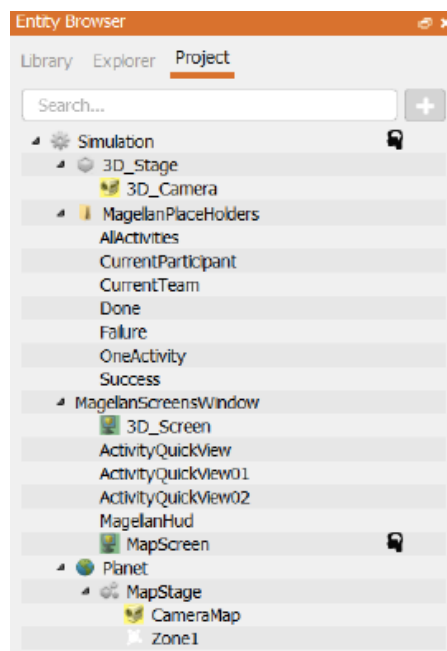


Figure 24: The entity browser

- The **library** tab displays and handles available low level entities that are already stored by default and can be copied in the project to create an activity. You can to add an entity from the simulation to the project library and will be available on all of your projects. To delete an entity just right click and select delete from library.

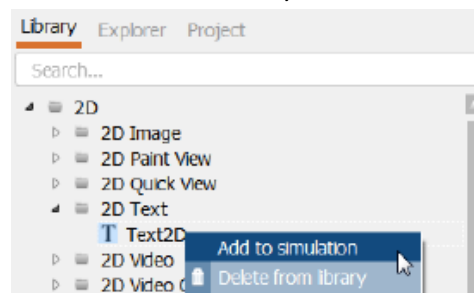


Figure 25: Adding an entity from the library



- The **explorer** tab displays a file explorer used to browse the data directory or content included in different projects folder. You can import models or other assets in your simulation. You can import an asset by dragging and dropping an asset to the explorer browser. The types of files that are supported are: 'v3b', 'png' and 'ui' files in 3D simulations.

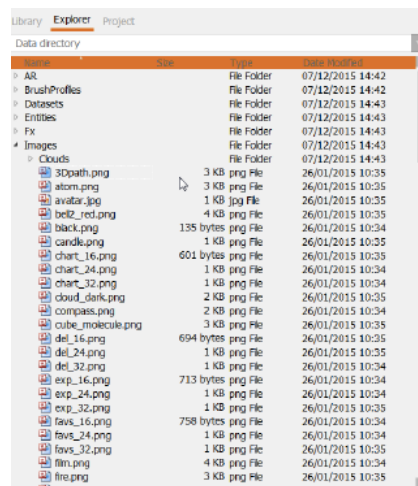


Figure 26: The explorer tab

- The **project** tab includes entity instances used in the current project. These are shown in a tree-like structure. Some entities have children entities such as activities, stages, stage windows or folders. It is not possible to drop assets from the file system to the project tab but you can drop entities from the project tree directly to the 3D scene. Entities can be displayed in the stage window when they belong to a stage. To move an entity to a stage just drag and drop it to the stage. You can directly edit an entity graph from the project tree by double clicking on it. It will ask you to save the name of the graph, if it already exists, it will ask you it will display a pop-up window to choose which graphic to edit.

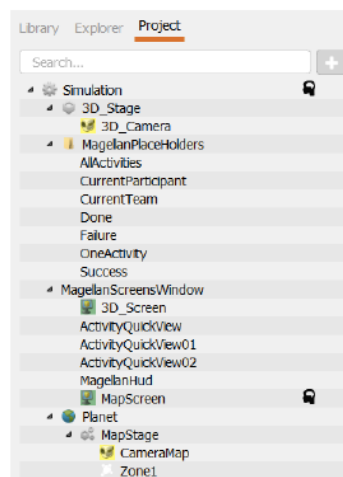





Figure 27: The project tab

- To delete an entity from the simulation just press the 'del' key (fn key with backspace on Windows in a Mac) or right click on the item to show the context menu and select '**delete objects**'.
- Different icons show the current state of the entities:



-  The entity is locked. It cannot be moved or scaled
-  The entity is hidden
-  The entity has at least one behaviour graph
- **The entity context menu:** Actions can be performed on selected entities in the **project tree** by right clicking in the project area for:
 - **Edit behaviour:** create a behaviour graph
 - **Rename:** rename the entity
 - **Delete:** delete entities
 - **Add entity to library:** Add the selected entity to the library
 - **Update entity to library**
 - **Leave advanced edition:** Exit the advanced edition mode
 - **Create folder:** Create a new folder
 - **Generate QR code:** Generate a QR code linked to the entity.

The entity editor

The entity editor is located under the entity browser and allows to edit the **attributes, the events,** and the **graphs** of the selected entities.

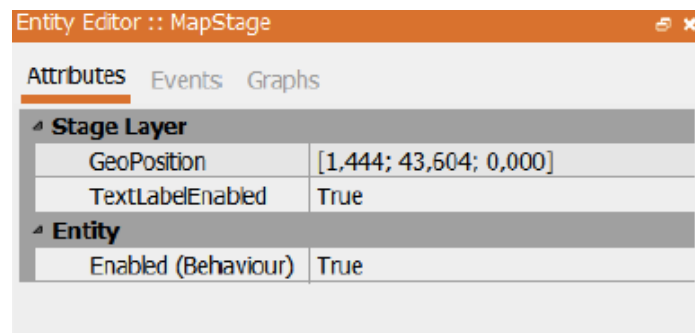





Figure 28: The entity editor

- **The attribute tab** allows to edit all the selected attributes such as the position or the scale of the 3D entities. There are two kinds of attributes: the single attribute which contains only one value and the container attribute which may contain more than one value. By clicking the 'plus' button  you can add an attribute to the entity. A dialog is shown to define new attribute name, description and constraints. To remove an attribute, just click the '-' button . Only custom attributes added by users can be deleted. Built in attributes cannot be removed. Click the 'rename entity'  to rename the selected object.



Name:

Type: Real List





Description:

Input Constraints

Min: 0 Max: 0

Step: 1 Unit:

Figure 29: The attribute editor

- The events tab: The events tab displays all the events of the selected entity. You can add, edit and delete any of the events. Built in events with the icon , cannot be deleted. However, events created by users with this icon  can be deleted. To edit an event you just double click on it. To add an event you click the plus button  to delete it you click the  button.

Attributes **Events** Graphs

 **OnActivate**
Event triggered when the entity behaviour starts



 **OnDeactivate**
Event triggered when the entity behaviour stops

 **OnUpdate**
Event triggered on each tick of behaviour (many dozens of...


 **OnStageStart**
Event triggered when the stage starts


 **OnStageStop**
Event triggered when the stage stops

Figure 30: The events tab

- **The Graphs tab:** Displays all the graphs of selected entities. You can add, edit or remove a graph. To edit a graph's name, right click to get in the context menu and then edit. To add a graph just click on the '+' button . A dialog is shown to define the new graph name and add description. To delete a graph, click on the '-' button .

Attributes Events **Graphs**

 **Main Graph**
This is the main entity behaviour

 **Start Graph**
This graph will be executed once, at launch time

Rename entity...

Figure 31: The graphs tab



Creating a 3D scenario

We described in Chapter 1, how to create a 3D activity. When the 3D activity is created, we get the 3D view from the activity and all associated entities highlighted in the project tab in the entity browser.

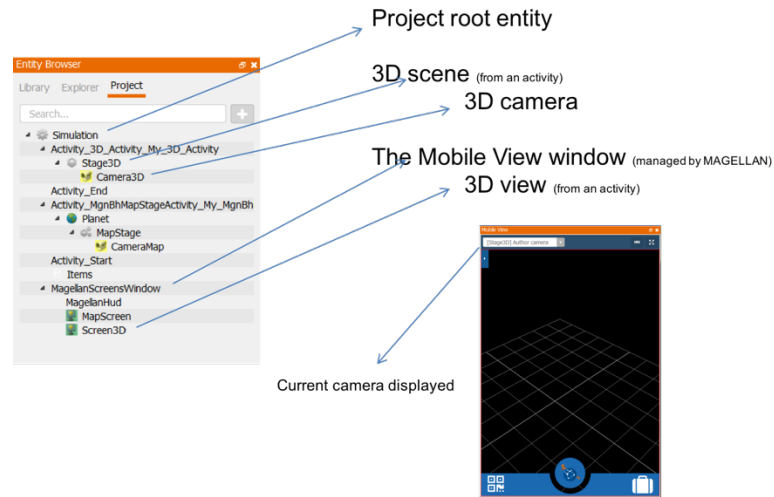


Figure 32: The 3D scene and entities in the project tab

Right click on the blue menu bar and select edition for the edition toolbar to appear.

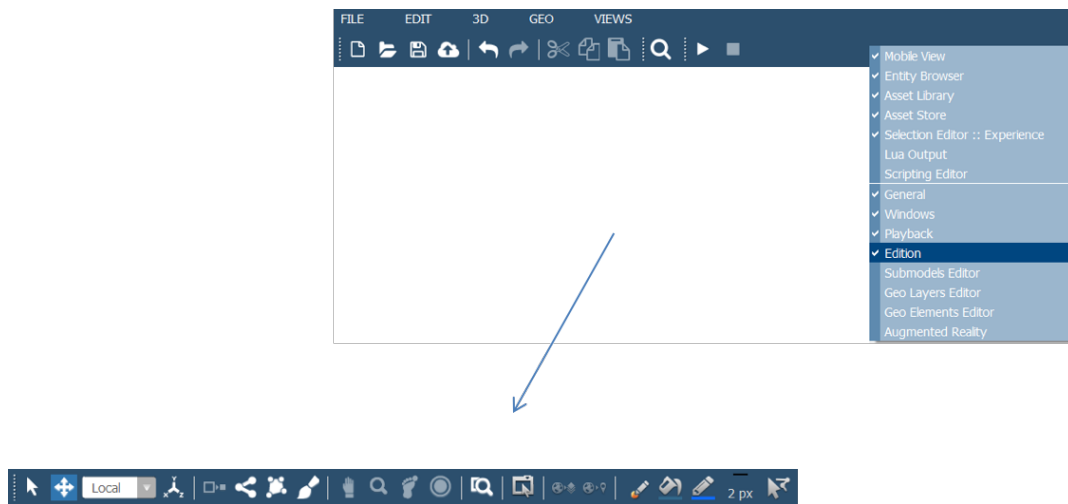


Figure 33: Edition toolbar



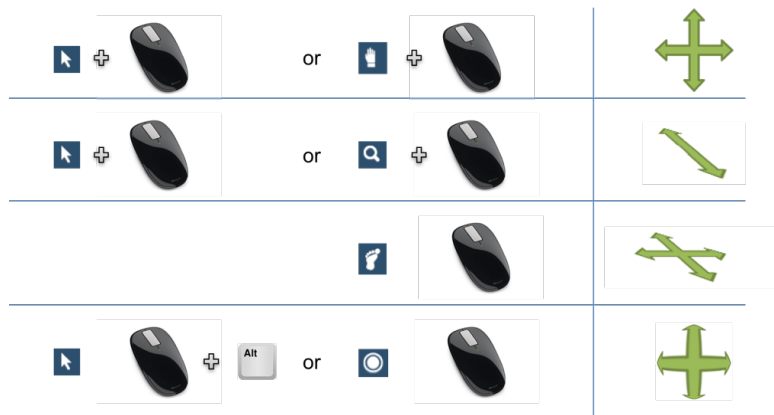


Figure 34: Camera navigation

1. The fastest way to select an object is to click on . It is enabled by default.
2. Multiple selections: Hold down the ctrl key to add or remove objects from the selection
3. Rectangle selection: Hold down the ctrl key to draw a rectangle and select all entities inside.

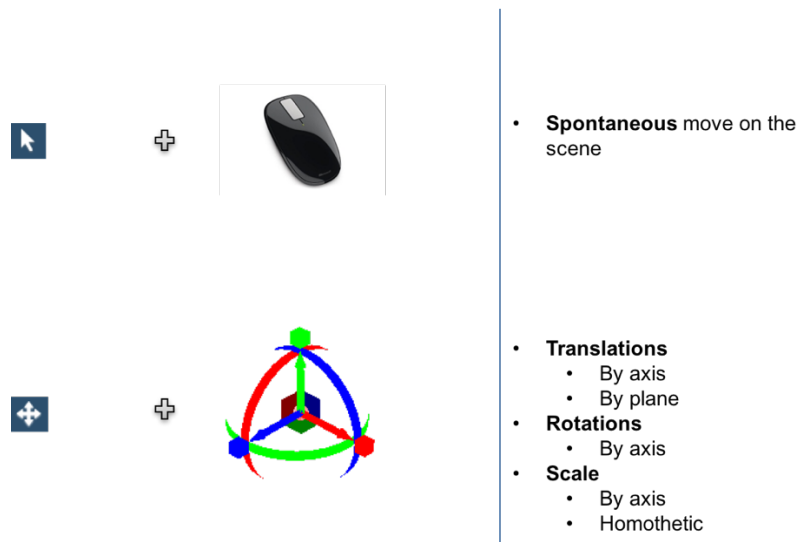


Figure 35: basic navigation features

Other basic navigation features include:

- Orbit : It controls rotation in any direction.
- Pan : It moves the whole view
- Walk : Moves the view to the viewport plane
- Zoom : change the magnification
- Path : Creates a path, click on viewport to create pathpoints; Undo last pathpoint with backspace.



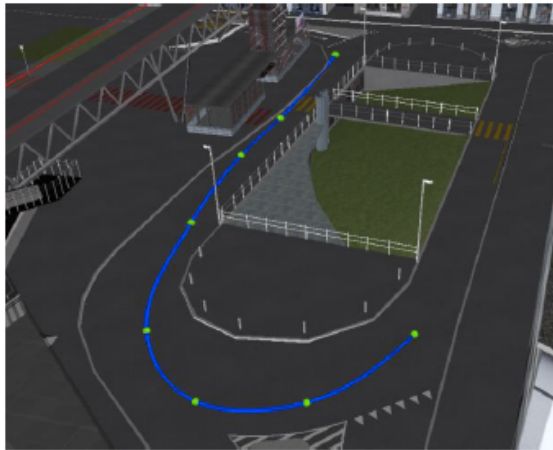


Figure 36: Creating a path

Importing files to the 3D scene

From your system folder you drag and drop any asset to the entity browser at the explorer tab. Then you drag it from the explorer tab to the 3D scene. A pop-up window prompts you to specify whether you wish for it to appear as a 2D/3D image as shown below.

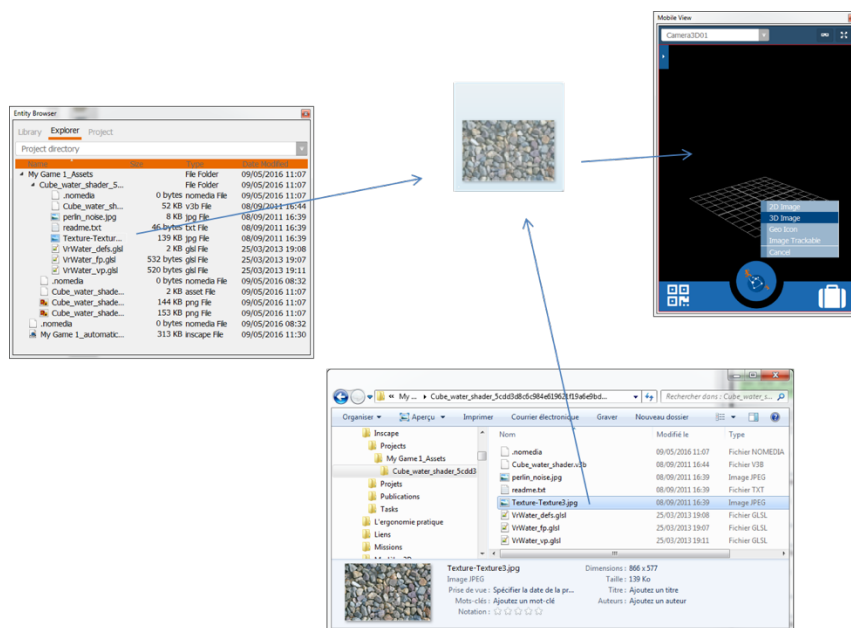


Figure 37: Import file in the entity browser

Then, clicking on the project tab, you will be able to see your asset under the Stage 3D subfolder and at the entity editor>3DQuad>filename.



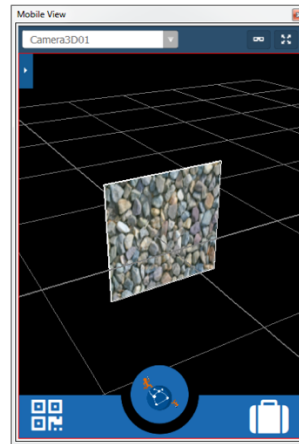
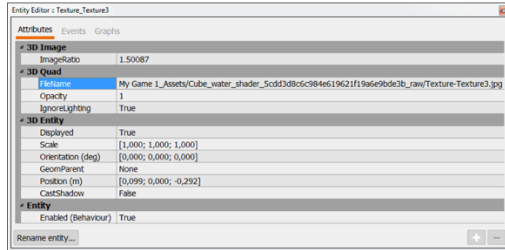
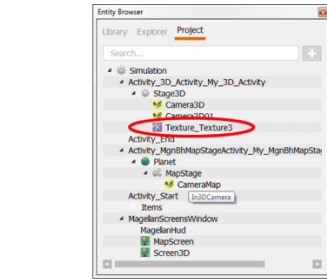


Figure 38: Assets in the stage3D



Chapter 3: Visual scripting

Apart from the scenario and behavioural level, the Magellan Authoring Tool includes more advanced interfaces for creating complex behaviours. Such complex behaviours can be realised via visual scripting. In this chapter we are going to introduce some introductory concepts on visual scripting for users with more advanced programming experience to start taking advantage of such features in Magellan.

In behavioural level, as discussed in chapter 2, the synapse editor capabilities are extended to edit the graphs of each Magellan entity you are creating. The synapse editor appears in a new tab once you click on the activity.

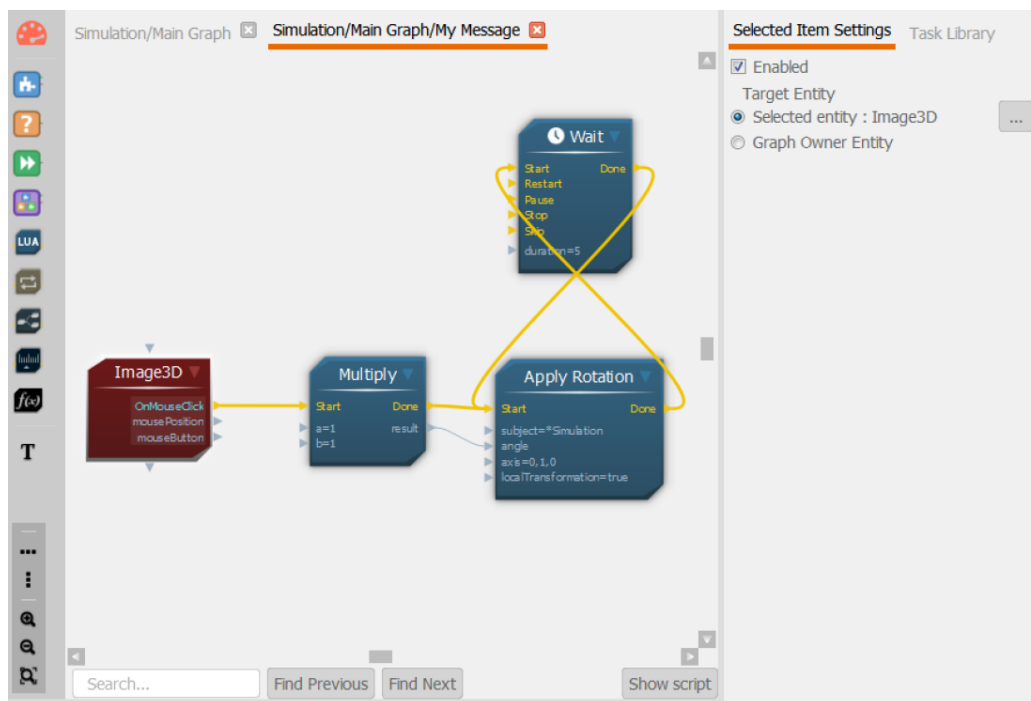





Figure 39: The Synapse editor

You can add a task or an annotation from the toolbar, and then you can drag and drop it in the graph. The Synapse editor on the left has 3 distinguishable tasks:

1. The Lua scripting task 
2. The synchronisation task 
3. The compound task 

Lua tasks

Lua is a scripting language used in Magellan to create complex task behaviours². We have two kinds of scripting tasks: **The classic and the co-routine**. The classic lua types is **bool, int, real, string, Vec2, Vec3, Colour, URL, Object, Matrix, Quat**.

² <http://www.lua.org/docs.html>



The co-routine script tasks run in more than one simulation instances. It calls the Function: (wait) and then waits until the next simulation starts otherwise it ends. Other functions include:

- **WaitAndUpdate():** waits and updates the lua variable
- **waitInput():** It does the same thing as wait() but the coroutine will not be resumed till an input value pin is stimulated.
- **waitInputAndUpdate():** same as waitUpdate but an input is needed to update the Coroutine.

Single Attributes

The value of a single attribute can be retrieved and changed with the following methods:

Value = entity.xxx and entity.xxx = value. Where xxx is an attribute of an entity.

Calling entity methods

To call an entity method use the : operator. For example:

```
Delta = Maria:getPosition() – George:getPosition()
```

```
Distance = delta.length()
```

Synchronisation tasks

Synchronisation tasks may encompass other tasks inside it. When the synchronisation task starts, all the sub tasks that their pin are not connected, they are starting normally. The Done pin is started when all the internal tasks are done. Synchronisation tasks cannot have input or output parameters but output control pins can be added and connect the internal tasks to them. The synchronisation task has a loop option. When enabled, the task loops until its stop pin is triggered.

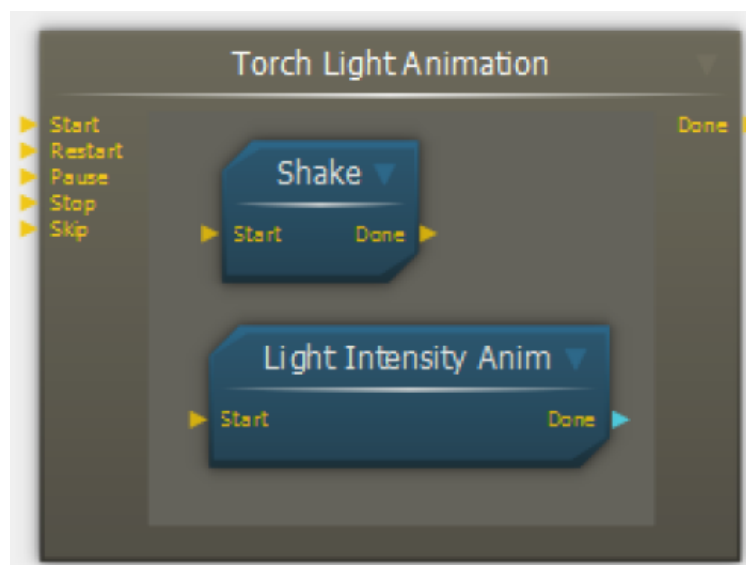


Figure 40: Synchronisation tasks



Compound tasks

The compound task, like the synchronisation task, may include other tasks inside it. The internal graph behaves exactly like a normal graph and it can contain entities.

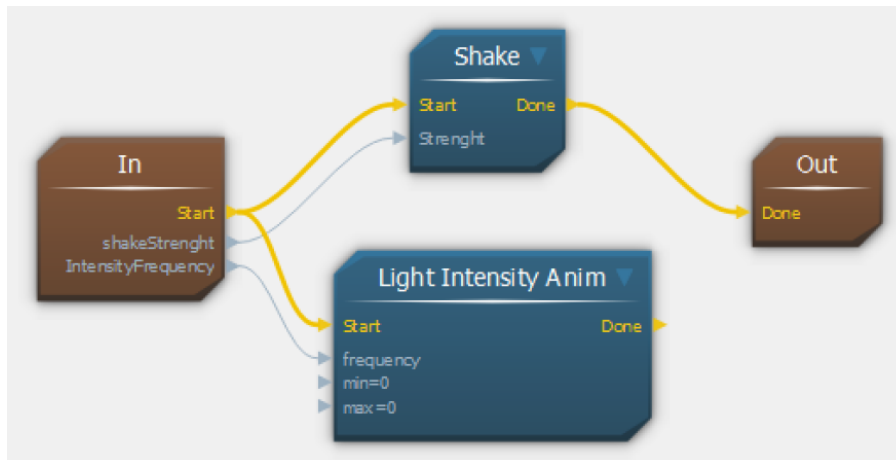


Figure 41: The compound task subgraph

Task library

The task library lists all the tasks available to the user in a tree-like structure. Any tasks are added by dragging and dropping from the library into the graph scene.

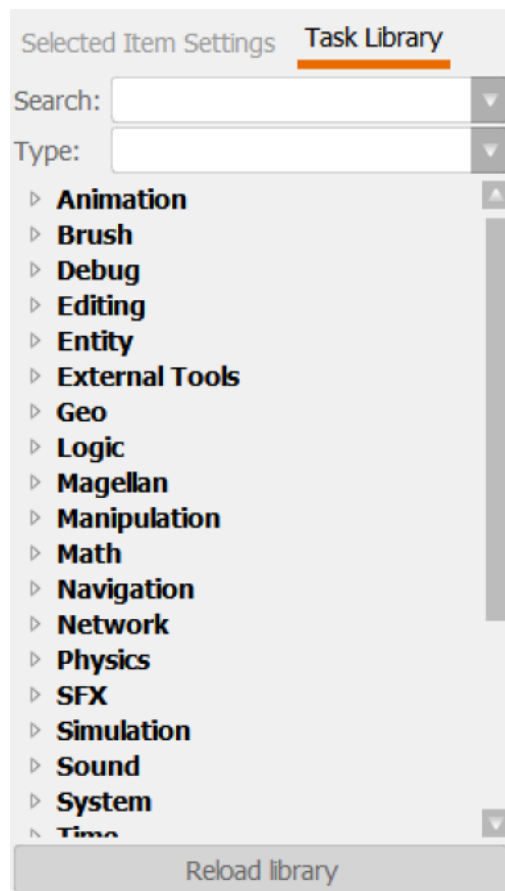


Figure 42: The task library



The subject type is highlighted with bold black and when you click on the subject type then you get a list of sub-tasks that you would like to use. You can also search for a specific task directly from the search menu. The tasks are added to the library by right clicking the context menu (right click) and select 'Add to the library'. Then a dialog pops up for configuring the task and then it is added in the graph.

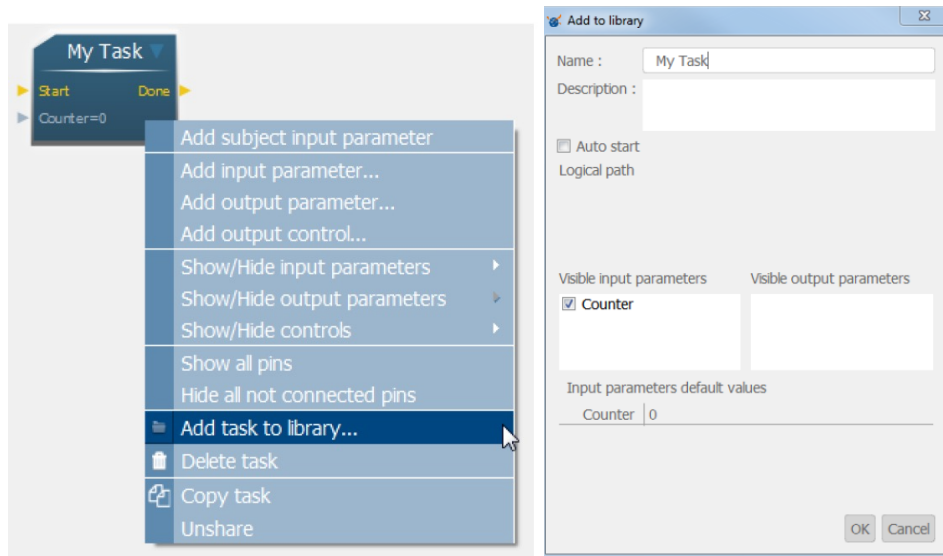


Figure 43: Add task to a library

GUI overlays

GUI overlays can be added from the behavioural level> entity browser>library and search for the 2D quick view and add it to the simulation.

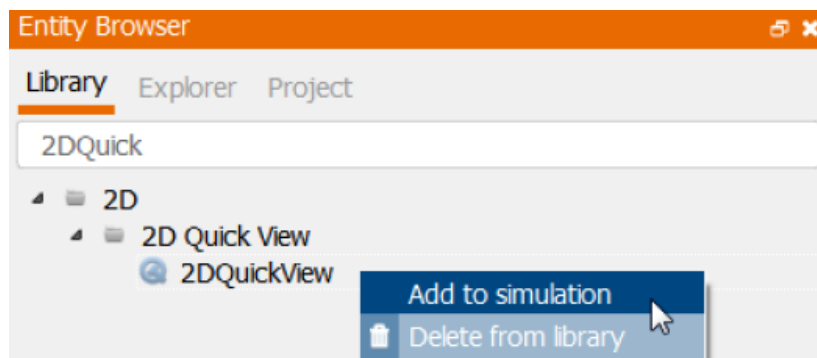


Figure 44: GUI overlays

A text file needs to be created which contains the GUI element. Once the text file is created, it can be imported in the 2D quick view within the election editor. Then drag and drop the 2DQuickView under the screen of the activity you wish to overlay.



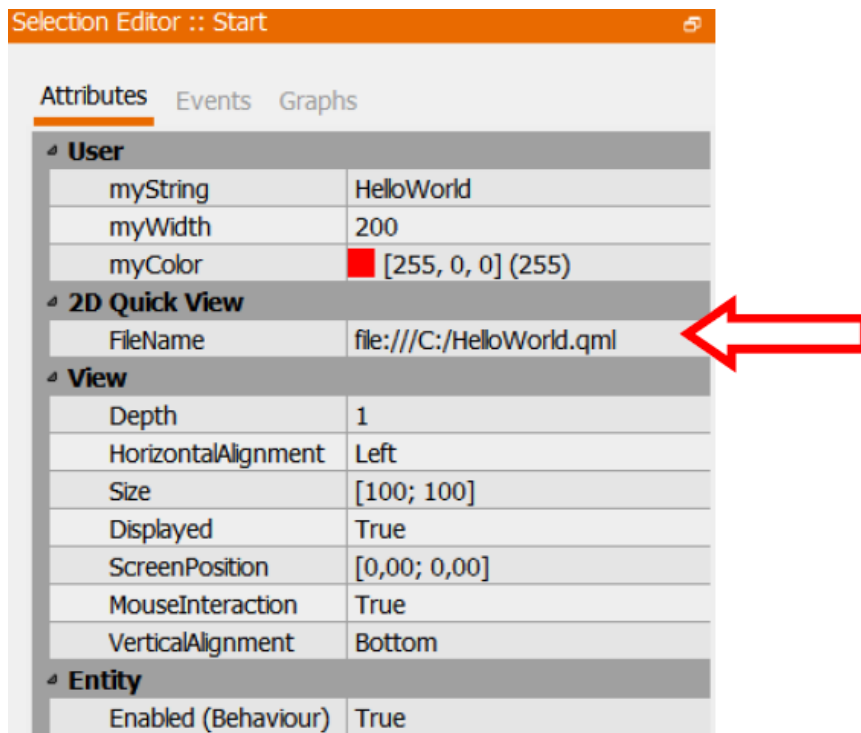


Figure 45: the QML file as it appears in the selection editor

In the events tab of the selection editor, create the event that will be triggered in the QML when clicking the red rectangle. For doing this, select the Event tab, and press the + button. Then double click on the 3d activity you have created to open its sub-graph and drag and drop the 2DQuickView in the sub-graph.

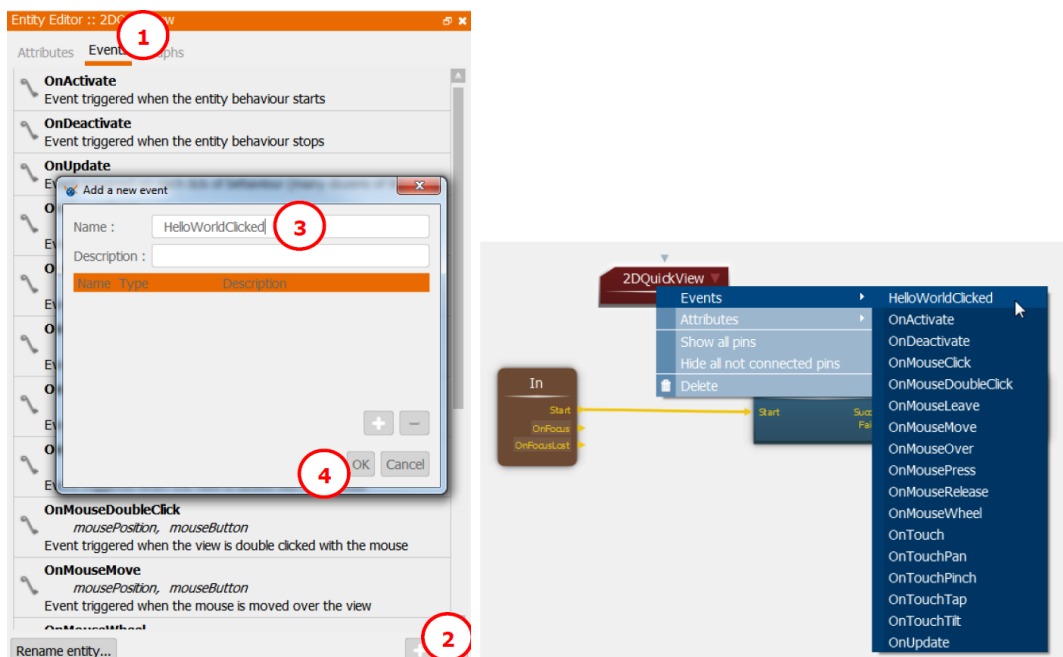


Figure 46: Inserting the 3Dview in the sub-graph



Conclusions

This handbook has been provided to the MAGELLAN end users as a guiding material for helping the process of creating location-based experiences using the scenario, behavioural and scripting levels. The handbook may be perceived as a step-by-step guiding document used in conjunction with other supportive training material produced for the MAGELLAN project such as the video tutorials and mini-activities. In the context of the different face-to-face and online training workshops as well as user pilots carried out in Europe, the trainers and end-users of the MAGELLAN project may use the handbook as means to provide assistance and support to designing individual and collaborative location-based games using the easy-to-use visualised authoring tools. The handbook is not a technical manual that provides a detailed synthesis on the technical complexities of Magellan, rather it presents the logic around creating location-based content in a way that users will understand all the authoring levels in a way that will help them to create a game based on the overarching premise of Magellan- which provides an easy to use tool for users without possessing the technical expertise to create a game. However, it should be noted that a game design should be planned before the actual game implementation for connecting game's logic with the game mechanics inherent to Magellan.

The handbook presents all the features and elements of the pre-beta version of the MAT. It provides the theoretical aspects of how to create location-based experiences and shows how these are being used in practiced. Basic and advanced functionality is presented with special focus on the new Magellan features such as the 3D activity and entity customisations at the behavioural and scripting levels. Since the MAT is deliberately designed in terms of connecting tangible state-of-the-art technologies such as Beacons and QR codes, the handbook places particular emphasis on how the MAT may be twined with such technologies. As such, the handbook provides guidance and support to designing location-based mobile games in field trips, in open spaces, in museums, in science centres or in a building. Individual and collaborative games can be created where people are assigned to teams and take on specific roles.

The handbook guides and support users on features, tools and functionality currently available in the pre-beta MAT version. Updated handbook versions will be developed to describe future releases with additional functionality of the tool.

