MODDING KIT GUIDE

Introduction

Welcome to the Modding Guide of Might and Magic X - Legacy. This document provides you with an overview of several content creation tools and data formats. With this information and the resources which are shared with the game, you will be able to create your own adventures.

First we want to start with some general explanations. This Modding Kit is NOT a simple-to-use level editor. It is an enhancement of the Unity 3D engine editor. Nearly all of the full features of the engine and tools can be used. In this way the Modding Kit allows all modders to create content to the same level of quality as the dev team itself. We are sharing the same tools Limbic Entertainment is using to create the world of Might and Magic X - Legacy.

To use the Modding Kit, modders will be required to work with the Unity 3D engine. If you are new to Unity 3D, we strongly recommend that you check the tutorials and documentation that accompany the engine. This document will not explain how the engine works. It will only explain how to use the tools from Limbic Entertainment to create worlds/dungeons/cities for the game Might and Magic X - Legacy.

Modding Kit Feature List

General overview of the modding kit feature list.

- Creating modding packs for own levels/adventures.
- Creating levels (e.g. dungeons, villages) for MMXL.
  - Flexible level height configuration
  - Setting for special area markers (free, blocking, hazard, ...)
- Modder can include their own models/textures in the levels.
- Access to the Castle, Cave and Shantiri tile set.
- Monsters and Interactive Objects from the full game can be used in the Modding Kit.
  - All monsters which are part of the main game are available.
  - Door, barrels, levers and other interactive elements from the main game can be used.
- Modders can create their own quests and link them in the levels.
- Each mod has its own game database which can be modified freely
  - Monster values and assigned 3D models
  - Item stats and enchantments
  - Loot of monsters and chests
  - Stats of all 12 of the character classes of the main game
  - Merchant offers and pricing
  - Many more.
- Multi language support: Each mod can configure its localization independently from the main game.
- Minimap / region map background can be configured for each level.
- For each mod an image can be configured to be used for the start screen.
- Full support of the current Unity 3D professional version (most of the Unity features can be used for creating levels for MMXL)

Unity 3D Editor

Unity is available in different license packages. In this section we are focusing on the Unity free and professional editions.

Unity professional contains many tools for lightmap and occlusion calculation. Some of these tools are required in order to create 3D worlds with high quality and good performance. These features don't come with the Modding Kit. But to get the best game experience it is recommended that modders use the full feature set of the engine.

The Modding Kit itself will not fully run with the Unity free version. The main reason is that the core feature which is used for the export of the 3D environment is a Unity premium-only feature. We are sorry for this, but at the moment we have no workaround available which would replace the need for this premium feature. Because of this Unity free is not officially supported by this Modding Kit. Although we are aware that it won't be enough for passionate modders who really want to go in deep with the Modding Kit, it should be noted that the professional version of Unity is available as a free 30-day trial.

Aside from the 3D environment export issue, the Modding Kit should work fine with Unity free. This means that you can create, edit, and save levels in the normal scene structure of Unity. These can then be exchanged with friends. So if you have a contact who has access to Unity professional, he can make the exports for you if you share your source files. The exported files are required to load your Mod in the game.

UI Modding

The Modding Kit does not contain any functionality for UI modding yet, as the currently used pipeline does not support templates. Therefore it is not possible to change the look and feel of Might and Magic X - Legacy UI. For the moment we cannot say if and when we will add functionalities for UI modding.

Installation Guide for Unity 3D

The current Unity 3D version can be downloaded here: http://unity3d.com/unity/download

All components available through the Unity installer should be installed. The demo which is included with Unity can help you get a grasp of the general technology.

The Unity 3D documentation is available here: http://unity3d.com/learn/documentation

Running Unity and loading the Modding Kit project

On Unity startup you will be asked which project you want to open. Here you should select the ModdingKit folder in your game installation target folder. You can select a new path by clicking on the "Open Other ..." button.
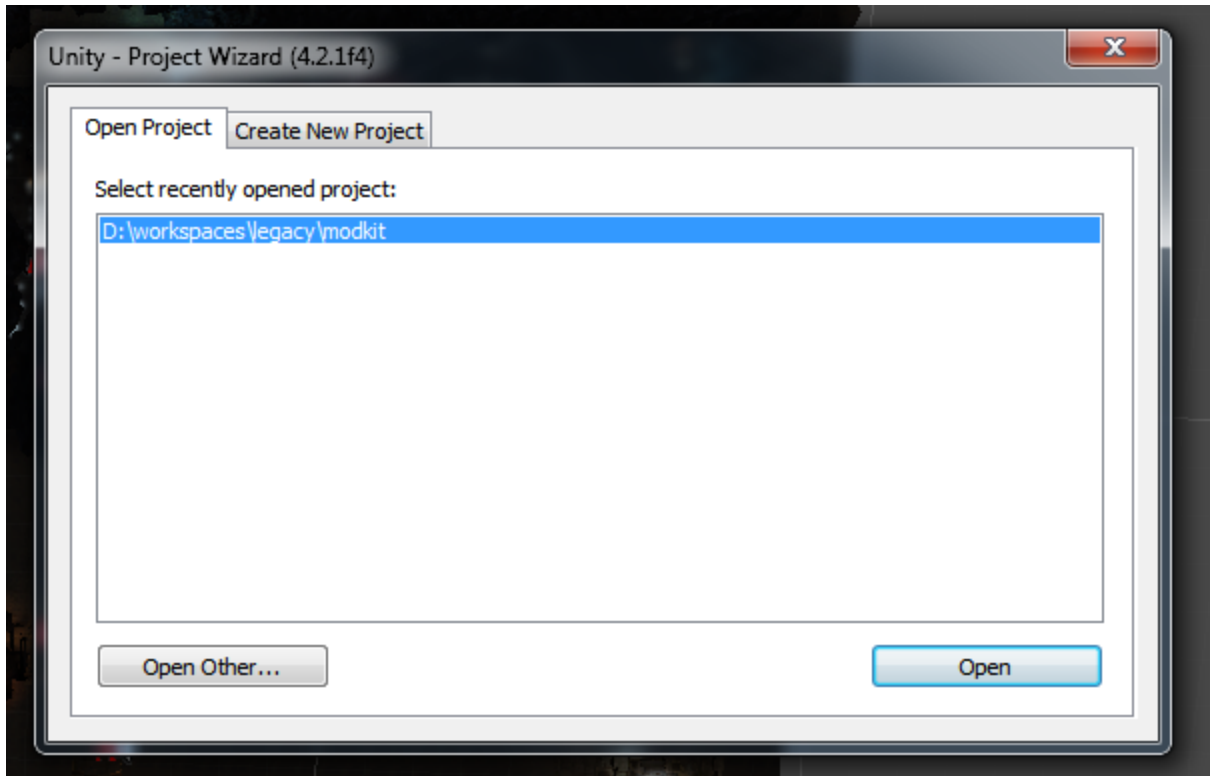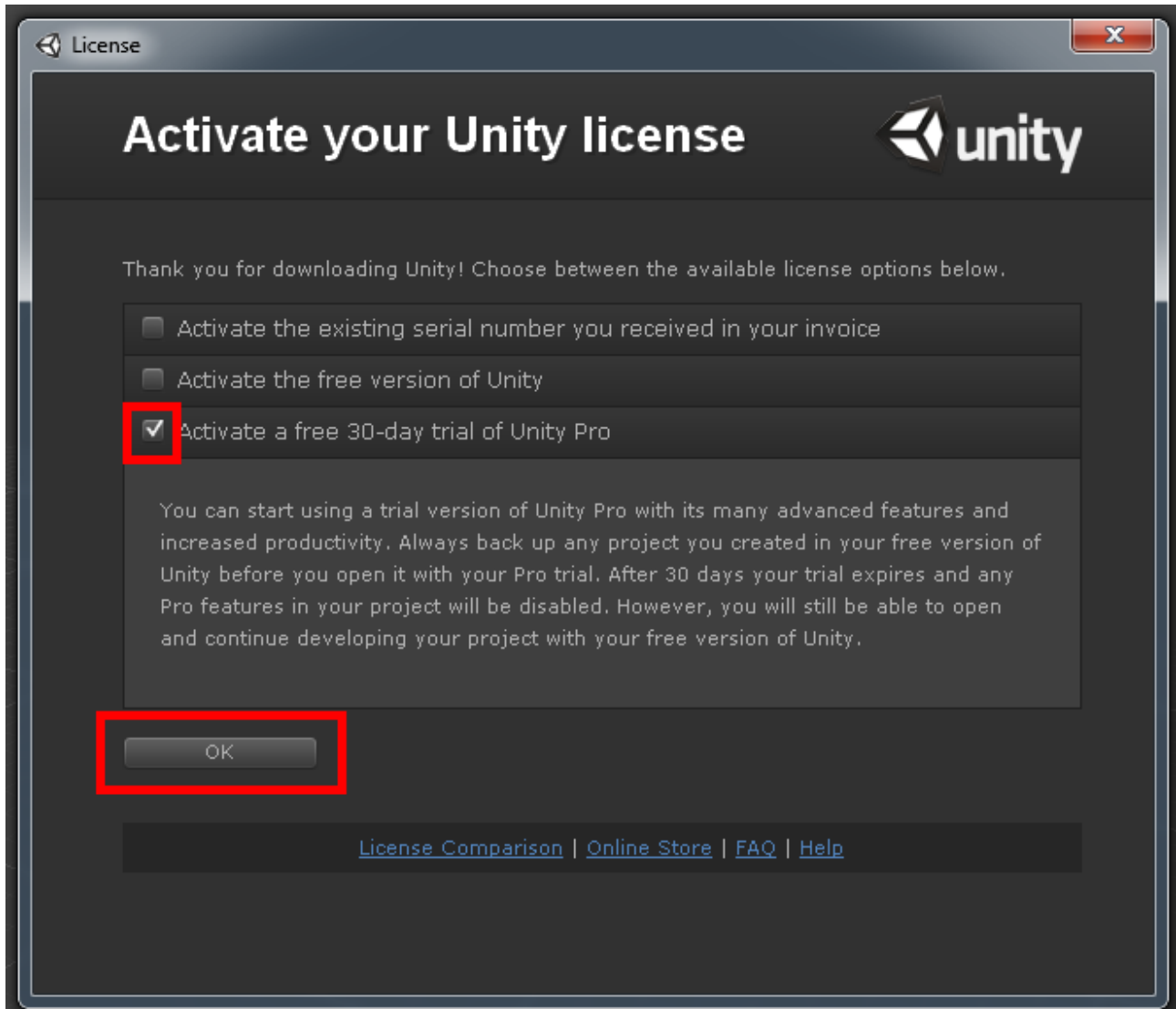

If you have already loaded a different project (for example the sample project from Unity) you can switch the project by opening a new one:
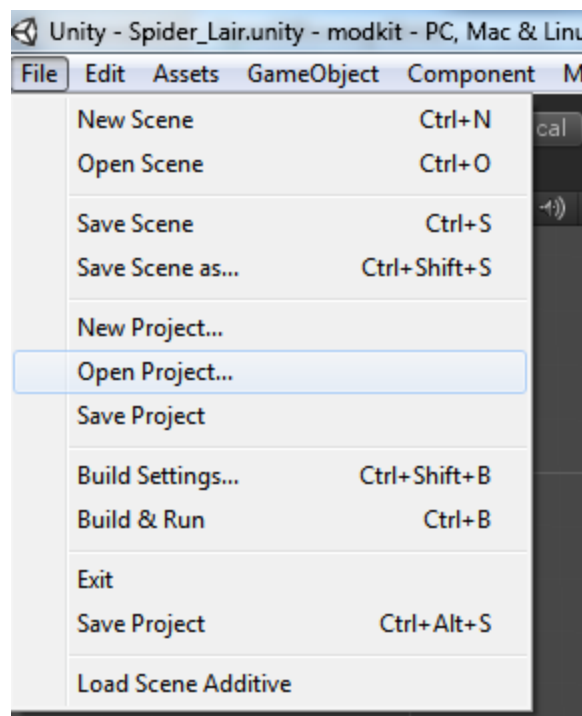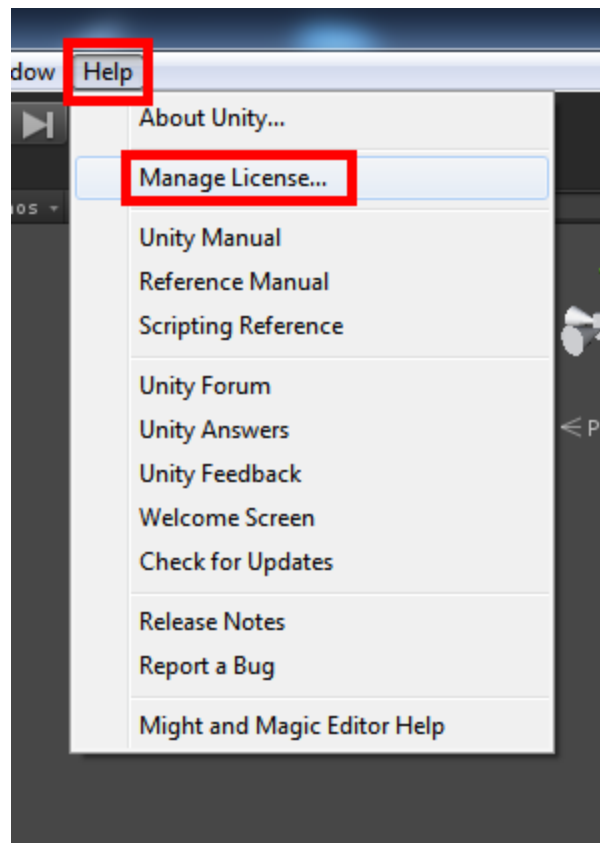
Warning: The asset import on the first load will take a while.
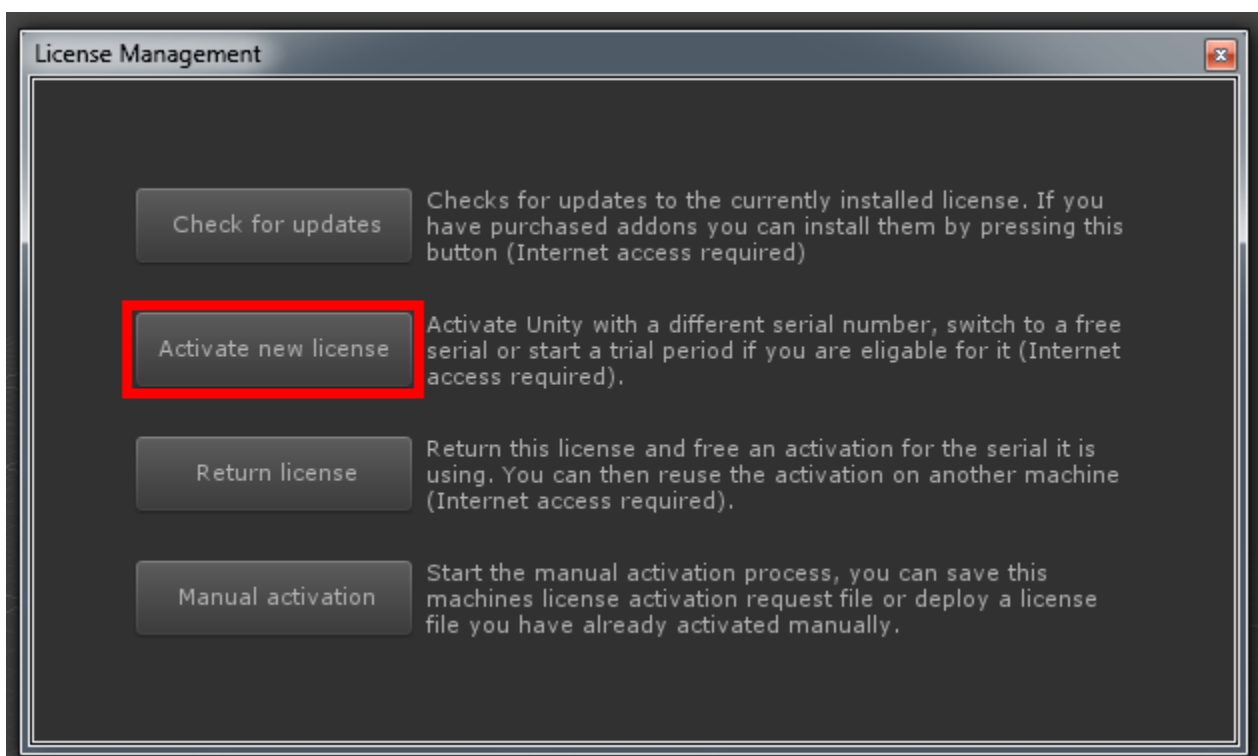
Unlocking Unity Pro trial

On Unity startup you will get a request to activate your version (free, pro or pro trial).
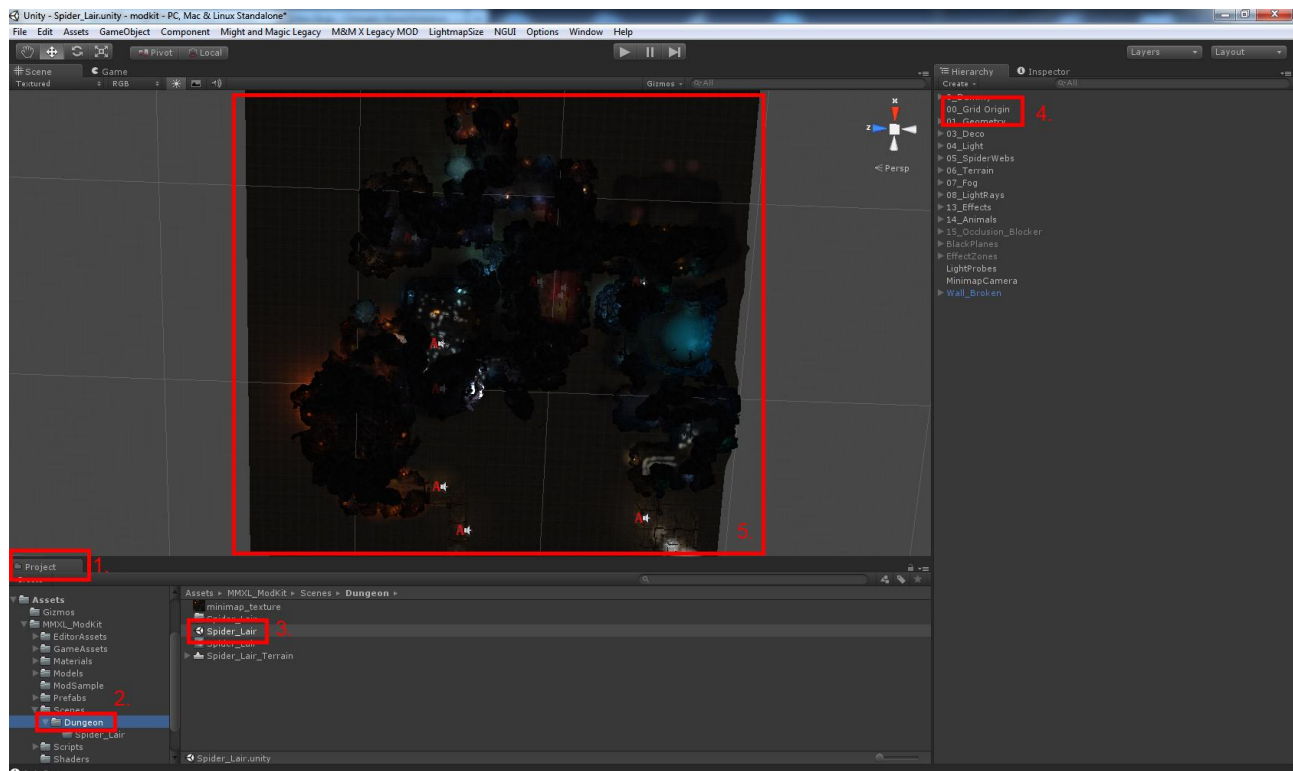

Select the 30-day trial and press "OK". If you have already a free version of Unity running you can also upgrade it by going to the "Manage License" window:

And select "Activate new license".

Loading sample level

To open the sample level we added to the Modding Kit please follow these steps:
1. Switch to the project tab.
2. Open the Assets / MMXL_ModKit / Scenes / Dungeon folder.
3. Open the Spider_Lair scene file (the file with the Unity logo) with a double click.
4. If nothing is displayed after following these steps, check the camera position. To do that, open the grid origin in the Hierarchy tab by double clicking it.
5. In the scene view you will see the spider lair demo level.

Modding UI

Modding Main Menu
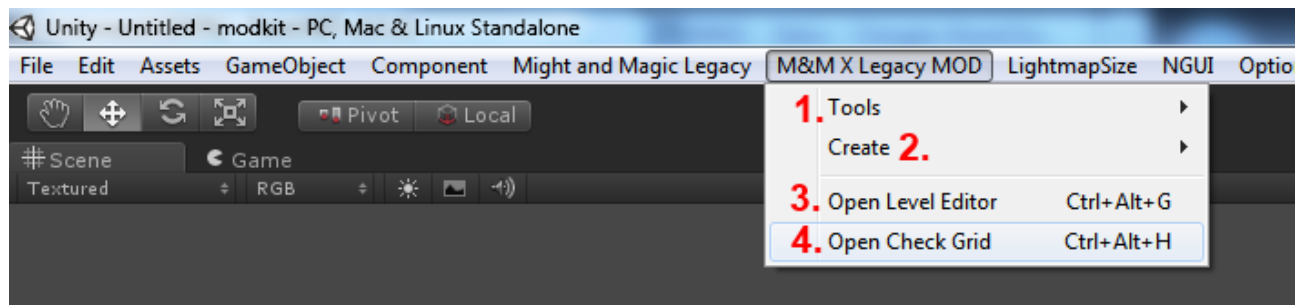
1. All tools which help create and set up new scenes (needed as first step for new levels), including creating background textures for minimaps are located here.
2. This is a collection of creation tools, which you can use to create databases (for localization and dialogues), new grid data, and grid origin.
3. This opens the level editor tab, which contains all the functionality which is required to add game logic to a level.
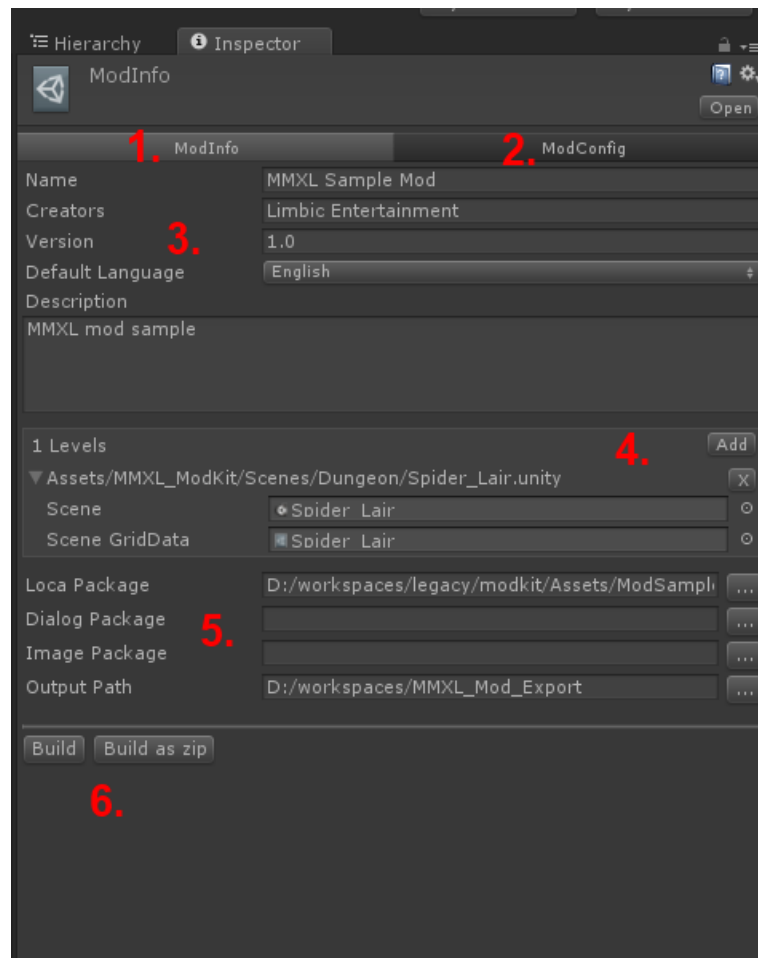4. This tool is for checking the condition of the grid and reports issues.

Mod Info Element
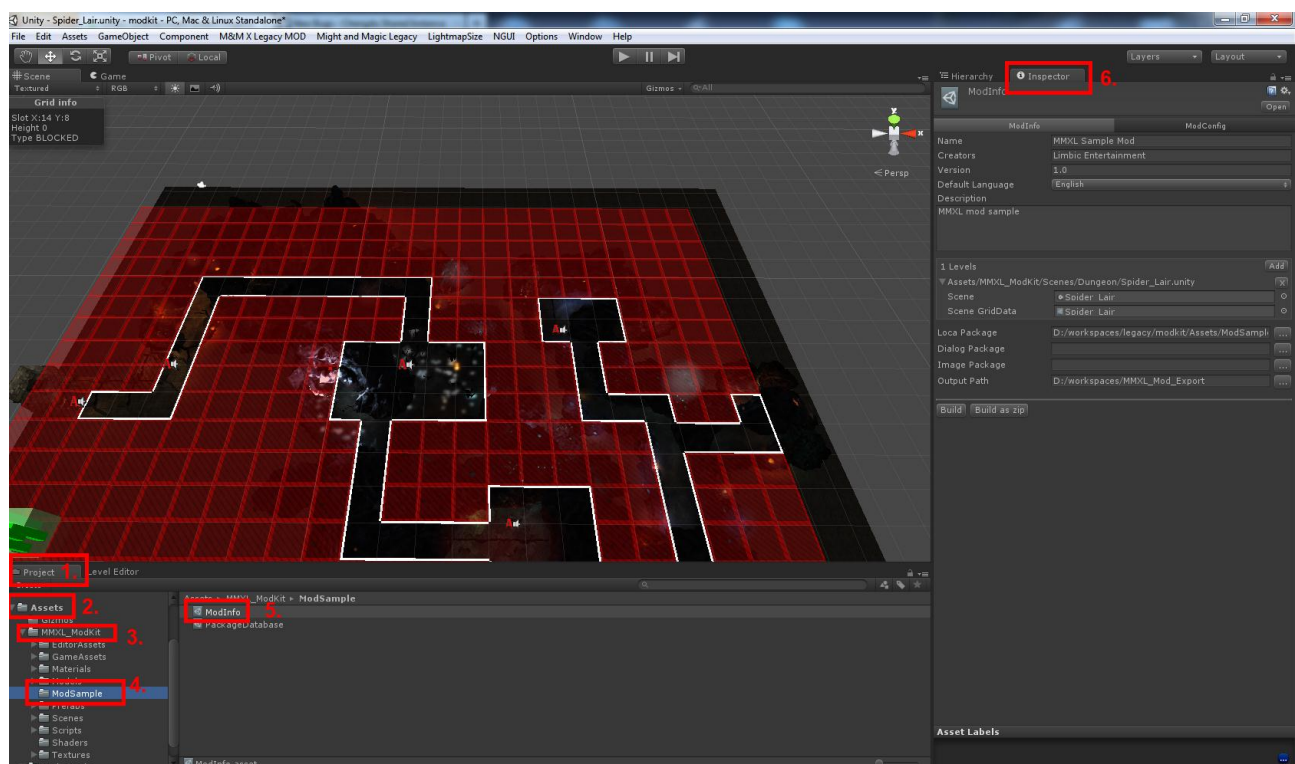
To open the mod info element of the sample you need to follow these steps:

1. Open the "Project" tab
2. Open the "Assets" folder
3. Open the "MMXL_ModKit" folder

4. Open the "ModSample" folder
5. Open the "ModInfo" file by double clicking
6. Open the "Inspector" tab

The ModInfo window contains these elements:

1. The "ModInfo" tab contains the general mod configuration.
2. The "ModConfig" tab contains the configuration for the game. This configuration replaces the game configuration after the mod is loaded.
3. In this area you can enter header data for your mod. The name and the creators are displayed in the game client, when the mod is loaded. The version number is also used to check if a save game is still valid. It will not be possible to load a save game with an older version number than the mod which is installed.
4. In this area you can add the levels which are part of the mod. Here you need to add (just drag and drop from the project tab) the scene and the grid data file. Only levels assigned to the ModInfo tab can be exported into the modding package.
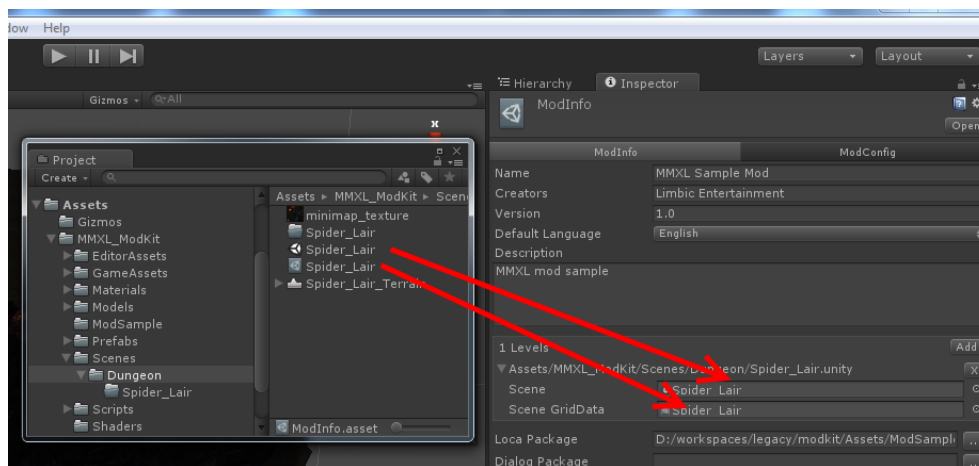
5. Here you can configure custom language and dialogue files. Also the background image of the main menu of your mod can be configured. The last input field is for the path where the mod should be exported. If these files are not configured the mod will use the files from the main game as backdrop.

6. These two buttons are used for exporting the mod. It will build the asset bundles of the assigned scenes and also copy all data which is required to run the mod. You can decide if you want to export raw files or a zip archive (reduced size makes it easier to exchange the mod). To use a mod in the game it needs to be unzipped.
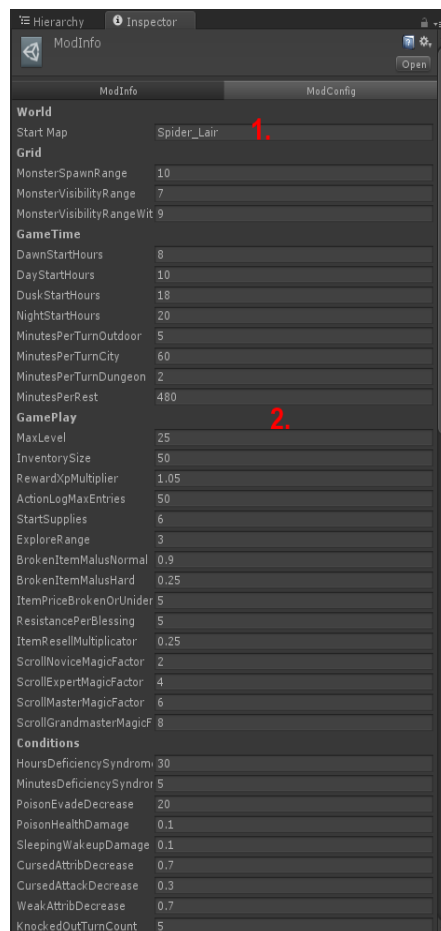
The second tab of the ModInfo window contains the general game configuration for the mod.

1. The most important value you have to set is the starting level. You need to configure it, so that the game knows which level to load first.
2. The other values are default values from the main game, so don't need to be changed. If you want to have different settings for the day/night change you can just configure it. Not every value will have an impact though. For example if you are changing the inventory size, the game logic knows how to handle that, but the UI will stay unchanged.
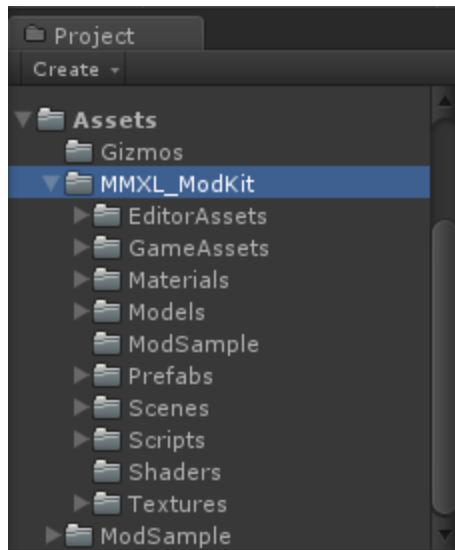
Modding Kit Assets

With this version of the Modding Kit the following tile sets are shared:

1. Castle
2. Cave
3. Shantiri
4. Village (Sorpigal buildings)

The different components of these assets are available in the following folders:

- Assets/MMXL_ModKit/Materials
- Assets/MMXL_ModKit/Models
- Assets/MMXL_ModKit/Prefabs
- Assets/MMXL_ModKit/Textures

Monsters and interactive Objects are not available as raw assets. However, they can be placed using spawners and will be loaded automatically in the game, so are available to use.
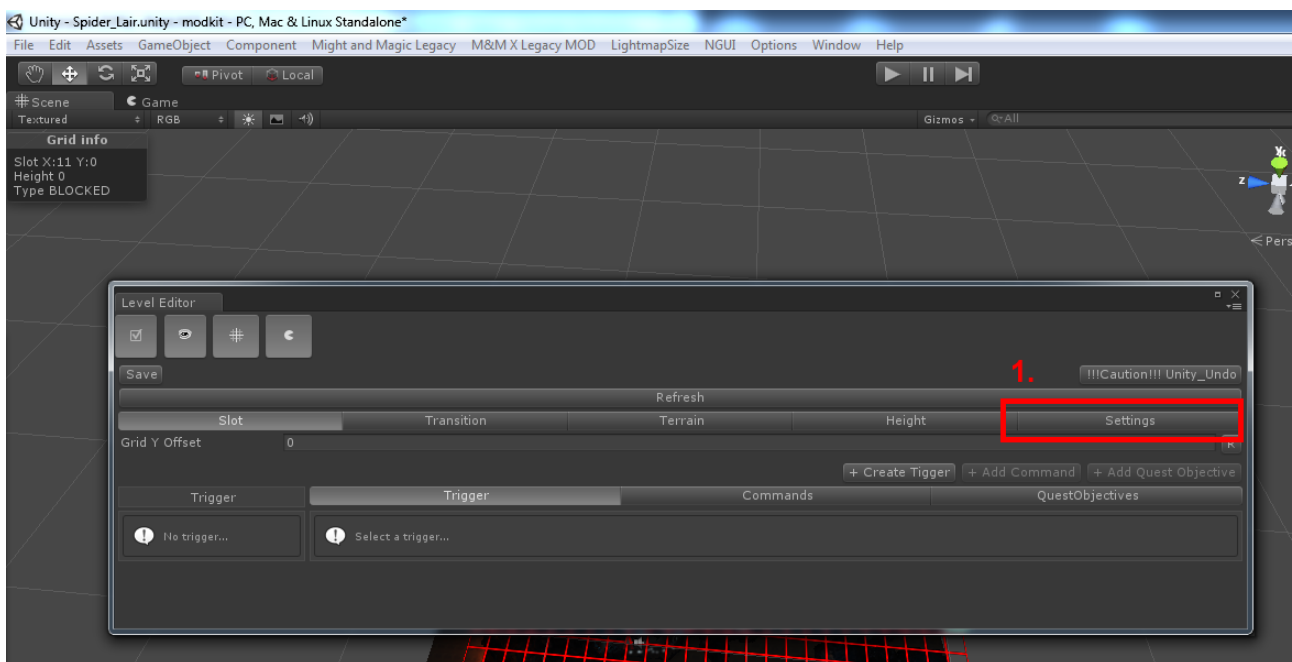
Level Editor Tab

First setting up

This is the central tool used to configure the game logic of a level.
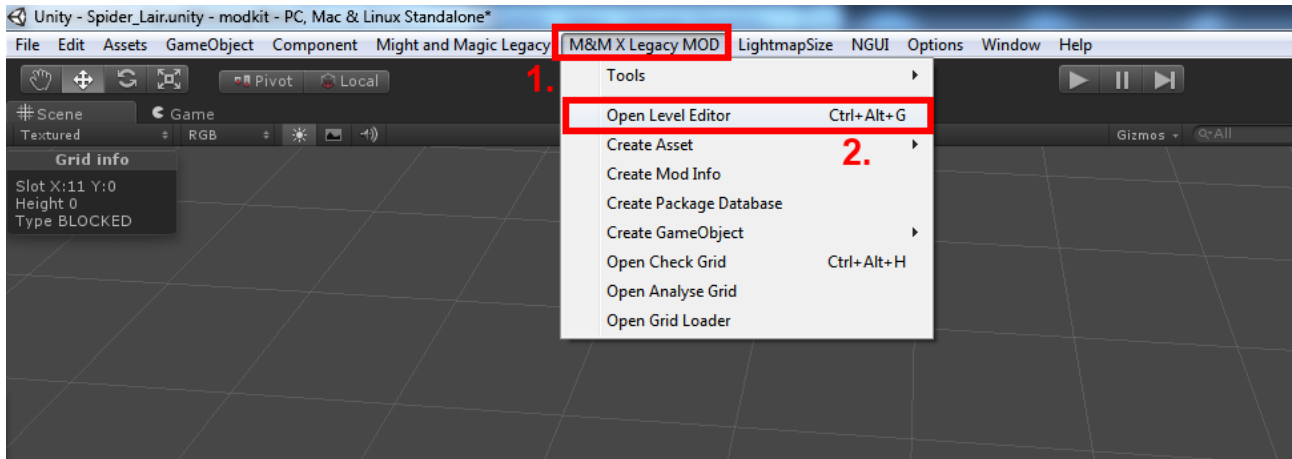

To open the Level Editor tab you need to:
1. Open "M&M X Legacy MOD" main menu entry
2. Select "Open Level Editor"

The first thing you should check is if the Level Editor is using the correct static game data files. To do this, open the "Settings" tab of the Level Editor. Directly after installation of the Modding Kit this entry is most likely empty and should be updated with the path of your Modding Kit folder. If this is not correctly configured there is a high risk that the modding UI will not work properly.
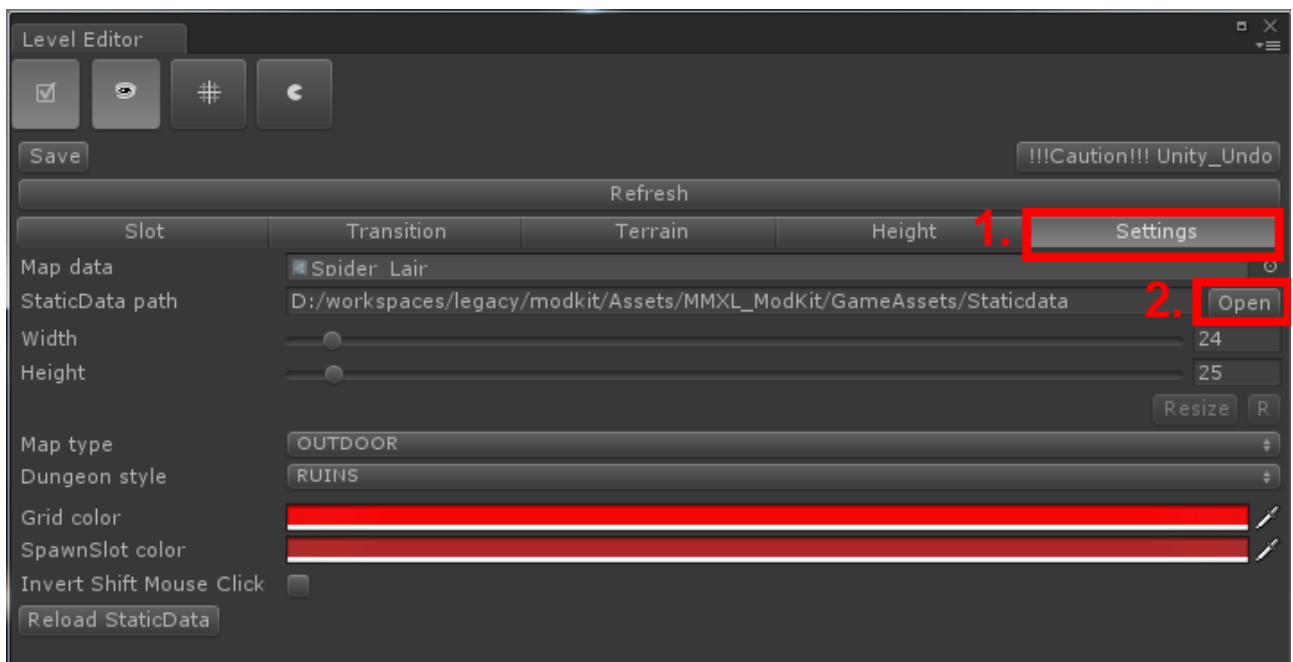
To select the correct path to the static data, you need to open the "Settings" tab and click on "Open". Browse to the installation path of your game. There, select the Modding Kit folder and browse to "Assets/MMXL_ModKit/GameAssets/Staticdata". Like in the image above. The static data will be loaded automatically and the Level Editor will be ready to use.

Slot tab

The "Slot" tab of the Level Editor is used to configure most of the gameplay related logic. In this area you are able to define triggers, commands and quest objective changes.
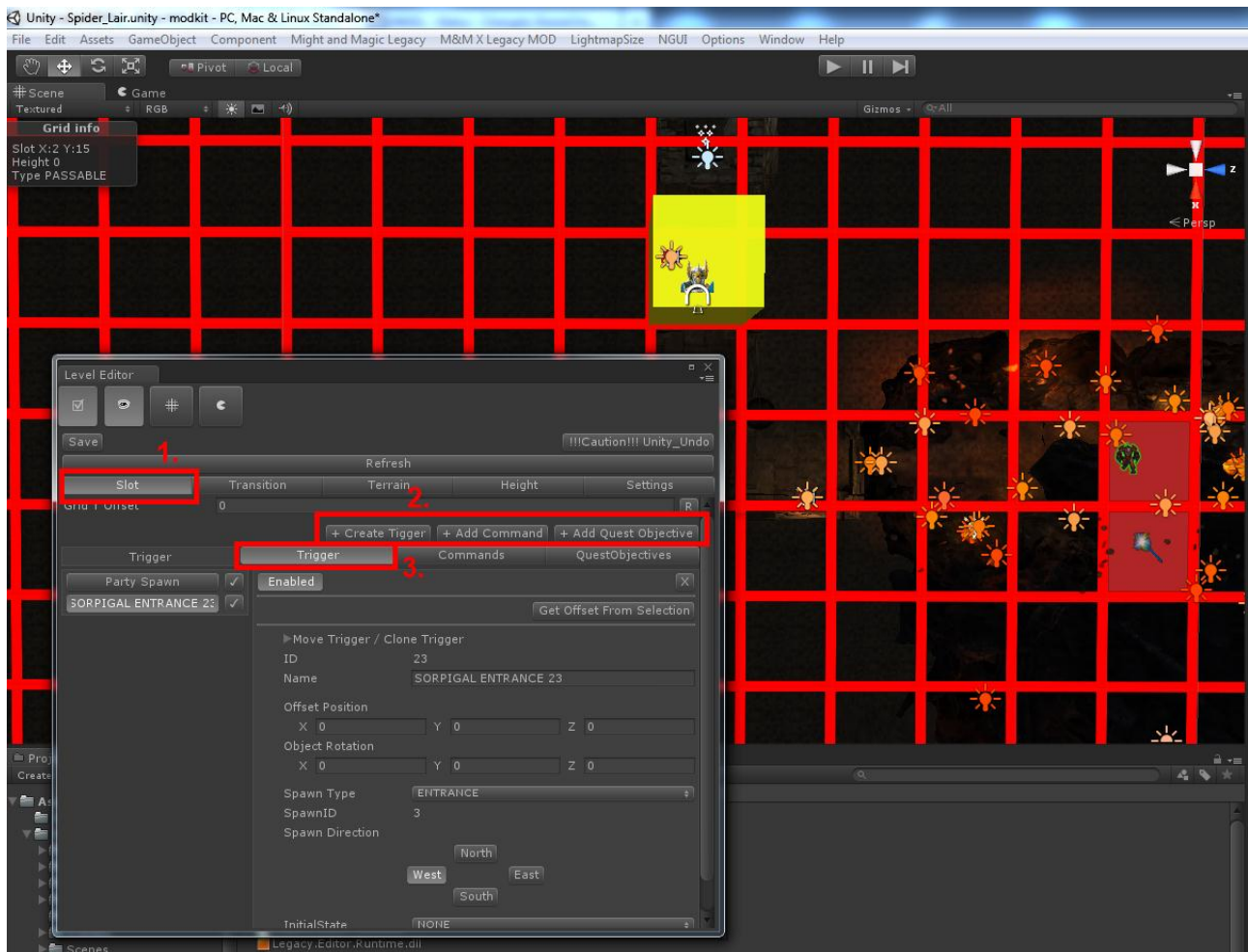
1. Open the "Slot" tab in the Level Editor.
2. Here you can create new Triggers, Commands and Quest Objective operations.
3. The "Trigger" tab is used to place interactive elements.

Trigger section

The Spider Lair sample level contains several samples for triggers. For example, if you click on a tile with a little monster icon you can see the settings which the game actually uses.
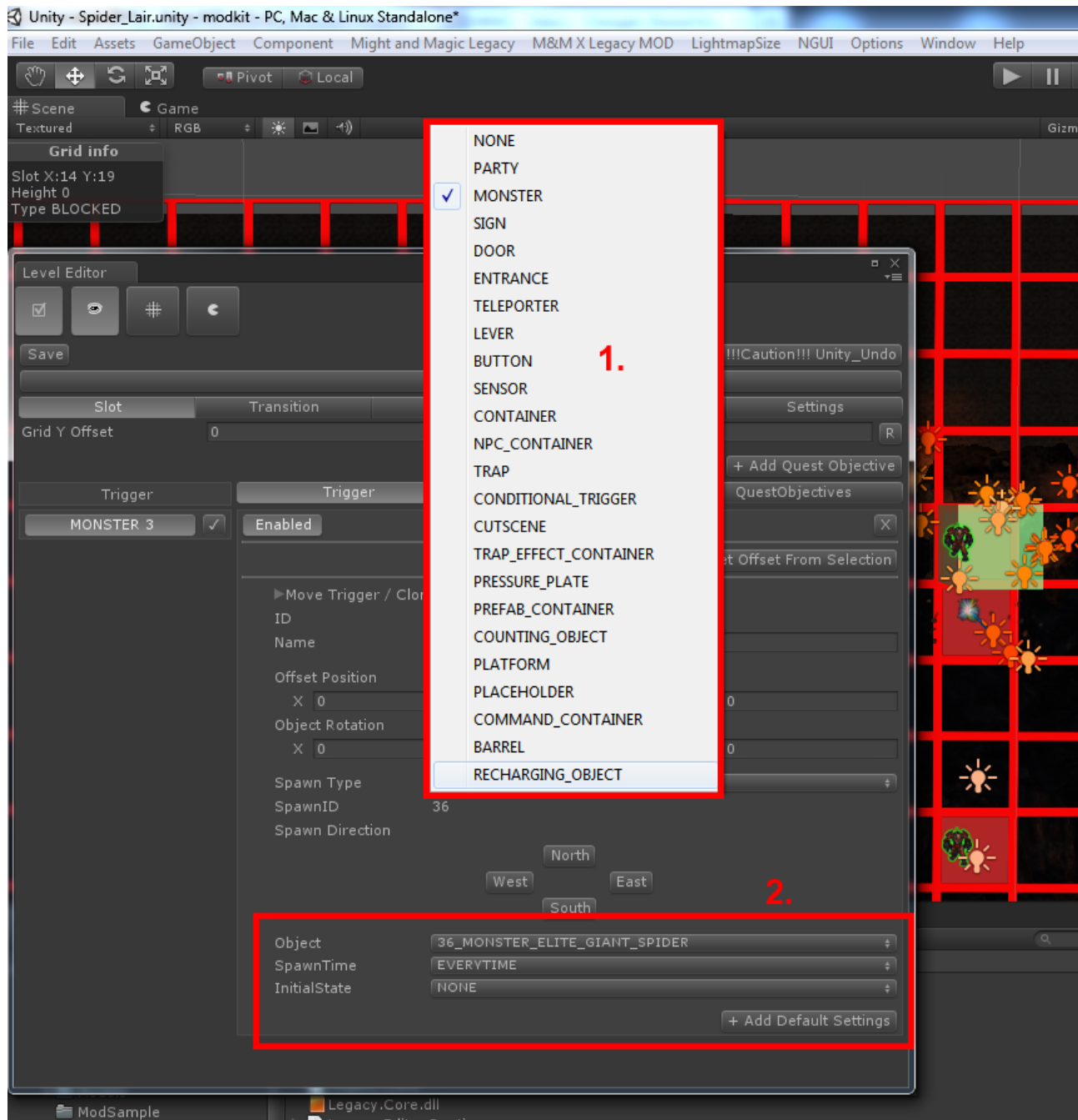
This trigger is reacting to movement and will spawn a monster when the player is in spawn range (this is a setting in the global game config which you can modify in the "ModInfo" window). You can select the type of object you want to spawn by clicking on the "Spawn Type" (1). Depending on the object's type you choose, you will see different settings (2). In the screenshot above you can see the monster configuration, since the selected spawn type is the monster. Here you can configure which monster is spawned and how often it is triggered.

The following list is a general overview of the most common triggers and what they are used for:

- **Party**
  This trigger is used to define party spawn points.

- **Monster**
  This trigger spawns an instance of the configured monster. For normal spawn behaviour no extra commands are needed.

- **Sign**
  This trigger can be used for signs. The data which is shown on a sign is defined by a corresponding view sign interaction, which is added to the sign trigger.
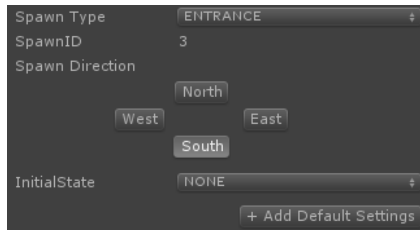

- **Door**
  The door trigger is usually used together with the toggle door command. With this combination doors can be opened and closed by the player.

- **Entrance**
  With this trigger you can start level changes. The level change itself is defined by entrance

commands.

- **Teleporter**
  This trigger is used with the teleport command to move the party from one tile to another.
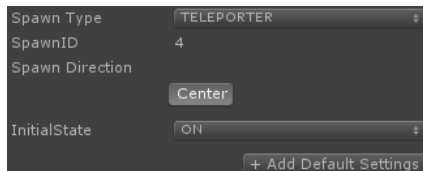
  Teleportation to a different level is also possible.

- **Lever**
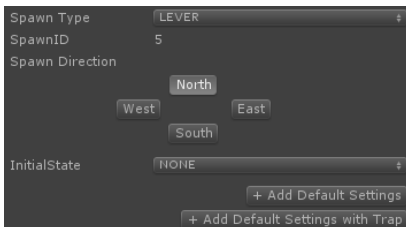
  This trigger is usually used together with the lever toggle and the door toggle command.

- **Sensor**
  This trigger will be executed when a specific tile is entered.
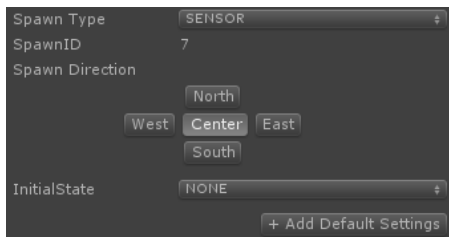
- **Container**
  With this trigger you can interact with lootable objects. It is used together with the open

  container command. You can also add traps directly to a container.

- **NPC container**
  With this trigger you can start an interaction with NPCs (which are handled by an NPC container). It is used together with the start dialogue command.
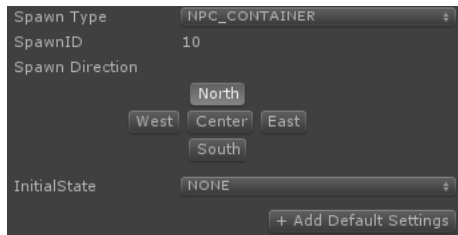
- **Trap effect container**
  Traps are split into two triggers. One to mark the tile on which a trap is triggered and one for the effect. This split was made to have better control over the location that a trap effects (the trigger and the effect don't need to be placed on the same tile). The trap trigger marks the tile on which a trap is triggered, while the trap effect contains the visual and game logic related information.
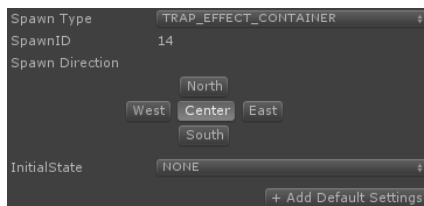
  Some metadata needs to be added to the trap effect container with the set data command to make the trap work:
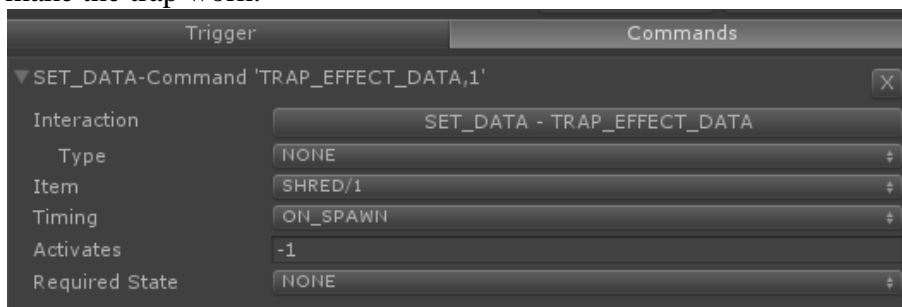
- **Pressure plate**
  The pressure plate trigger can be used very flexibly to trigger different types of commands.
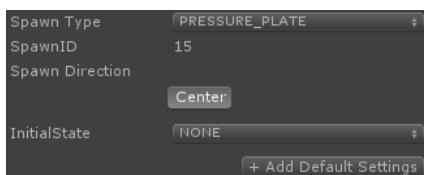
  For example it can be used to control buttons, doors or complex riddles with several prefabs.

- **Barrel**
  Barrels can be used with this trigger. You can link different commands to it like the barrel interaction for buffs or loot.
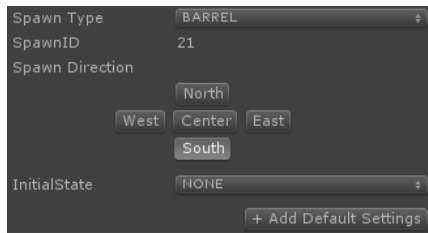
- **Recharging object**
  With this trigger you can activate effects which are regenerated after a certain amount of
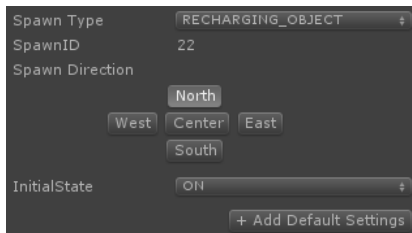
  time. It is used together with the recharging interaction command.

To see working examples of each of these spawn types we recommend looking at the triggers already added to the sample level. Also, you can take a look at the levels from the game. The XML files can be accessed in the "Might and Magic X Legacy_Data/StreamingAssets/Maps" folder of your game installation. Additional documentation for the triggers and spawn types will follow shortly.
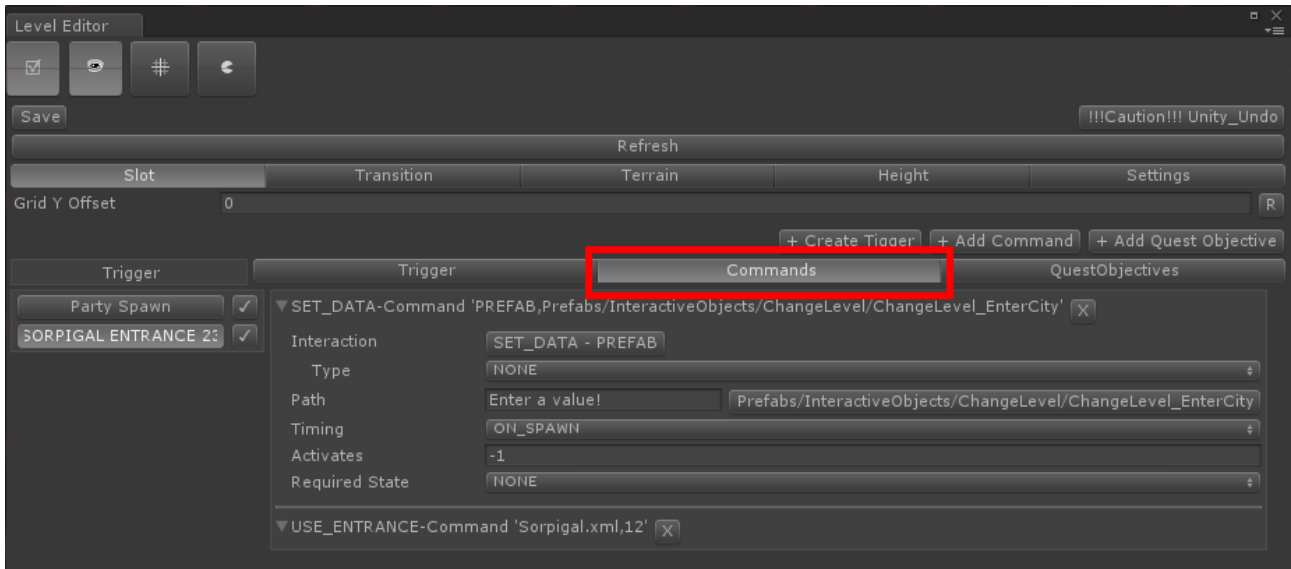
Command section

With the "Commands" tab you can define interactions and their configuration. When you create a command it is automatically assigned to the current active trigger or is assigned by the id which is configured in the Target Spawn ID field. A command is basically an action which is performed after its trigger is activated (depending on the configured timing). An interaction is, for example, a level change, opening a door or pulling a lever. Some interactions are assigned to prefabs for visualization purposes.

The exact timing of commands can be configured using the following states. Depending on these settings a command is executed after its trigger is executed.

- Never
- On spawn
- On execute
- On success
- On fail
- On after spawn
- On level loaded

In the following subsections you will find several test cases and examples on how to use commands together with triggers. These are production samples from Might & Magic X - Legacy and are used in the course of the game.

Door with key

Trigger	Commands
▼TOGGLE_DOOR-Command ''	

▼TOGGLE_DOOR-Command ''		X
Interaction	TOGGLE_DOOR -	
Type	PARTY_CHECK	↕
Decision	NONE	↕
Main text		
Success text		
Fail text		
Required token	DEN_OF_THIEVES/KEY_GAMMA_112	↕
Without token	NONE_-1	↕
Blessing	-1	
Required Hireling I	-1	
Without Hireling ID	-1	
Inactive quest step	-1	
Active quest step I	-1	
Solved quest step	-1	
Target SpawnID	46	-DOOR 46 - Key Gamma SetMyID
Timing	ON_EXECUTE	↕
Activates	-1	
Required State	NONE	↕

▼REMOVE_TOKEN-Command '112'		X
Interaction	REMOVE_TOKEN -	
Type	NONE	↕
Target SpawnID	46	-DOOR 46 - Key Gamma SetMyID
Token	DEN_OF_THIEVES/KEY_GAMMA_112	↕
Timing	ON_SUCCESS	↕
Activates	-1	
Required State	NONE	↕

For this you need to place a normal door trigger and a toggle door and remove token command (shown above). The toogle door command checks if the door can be toogled or not, depending if the configured token is available or not. This will be done on the execution phase of the trigger. After a successful execution of the trigger the command to remove the token will be triggered. It removes the configured token from the party.

Tokens are configured in the static data tables of the game/mod. You can re-use the tokens from the main game or you can define your own if you create a separate static data database for your mod. After they are configured in the tables you can access them in the editor and place them--for example as loot from monsters or in chests. Tokens are a very important mechanic for flow control. You can even define which dialogue is shown depending on a token. In the data of the main game you will find many examples on how to use tokens.

Secret Doors

The secret door uses a normal door trigger with two set data commands and an open door command. The first set data defines the used prefab of the secret door. The 2nd set data command flags it as secret. The open door command contains a "who will challenge" (based on the magic attribute) and the text used for the popup (the text needs to be contained in the localization database). Players are only able to interact with secrets if they have the spot secret ability/spell active.

Teleporter
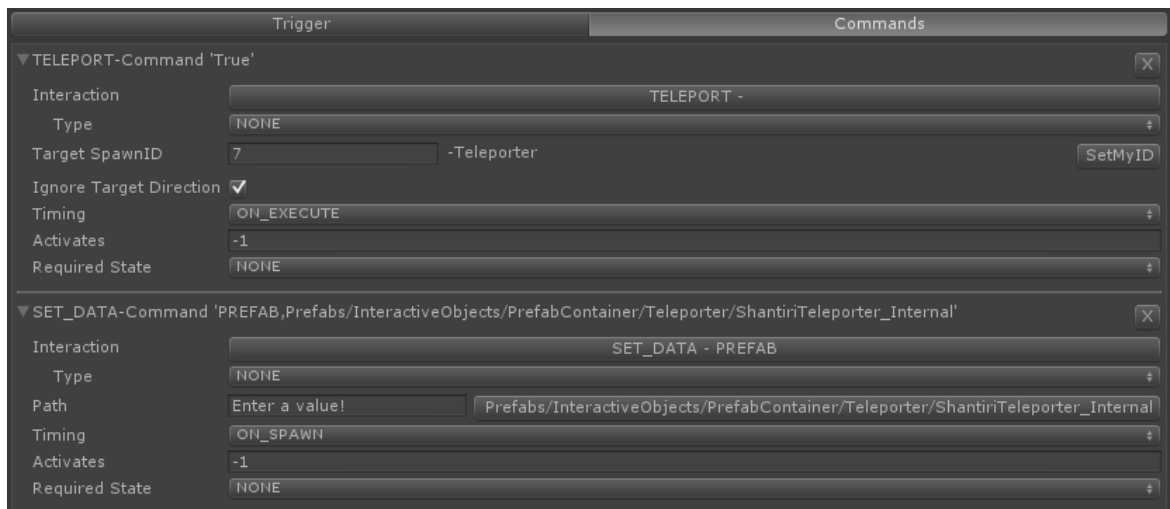
To implement a teleporter you need two teleport triggers which have each been assigned a set data command. The set data is used to define which prefab is used for the teleporter. Both triggers are used as start and endpoints of the teleport. The starting trigger also needs a teleport command. In this command the id of the endpoint trigger needs to be entered in the Target Spawn ID field. Players will be teleported after interacting with the first trigger to the location of the 2nd trigger.

Level switch
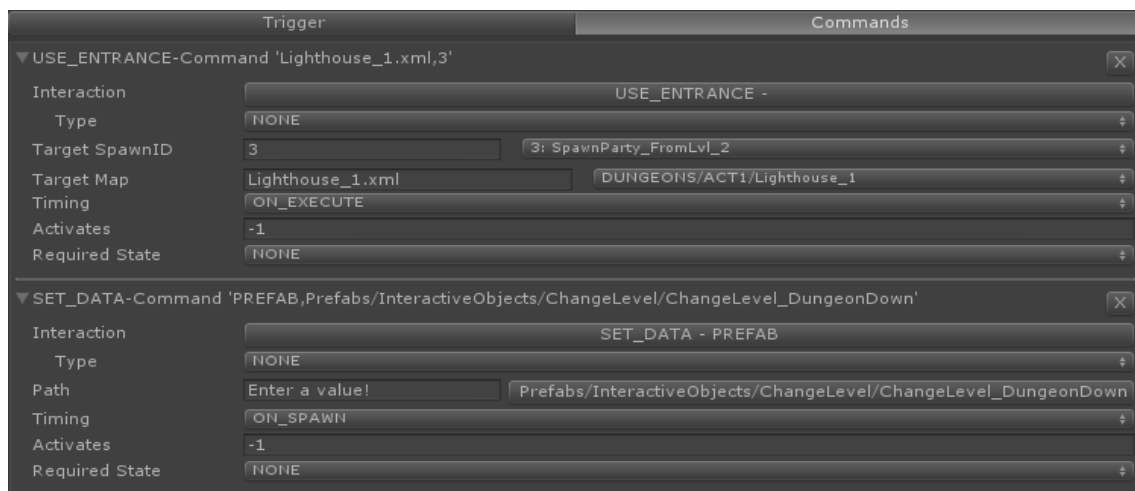
For the transition between levels you need an entrance trigger. The prefab used for this trigger can be configured with a corresponding set data command. Here the assigned prefab is not a 3D model. You are assigning the animated 2D image which is used to show a level transition (for example, see the picture below used for entering the spider lair). This is called a transition icon. For the level change itself a use entrance command is required. There you can configure the target map with a spawn point of the party.

[Space]: enter Spider Lair

Loot Containers

For any lootable object (for example a chest or barrel) you need a container trigger. The visuals of the container are set with the set data command. A series of prefabs are available in the menu. The loot is also defined with set data commands. In the static game data you can define pre-configured loot. This pre-configured loot is referenced in the "set data - container generated" commands. The game will create the loot when the container is spawned out of the values which are configured in the tables and in the editor. To open the chest you need to add an open container command which is linked to the container trigger.

The static game data for loot containers can be configured in the static data of the main game or in your own database. This database can be customized by you to match the needs of your mod.

NPC containers

NPCs are placed in so-called NPC containers. Even if you want to use only one NPC you need to place him into an NPC container. The sample here shows the NPC container of the shrine in Sorpigal. It contains many different configurations.

```
▼ SET_DATA-Command 'NPC_IDS,5'                                              X
  Interaction              SET_DATA - NPC_IDS
     Type          NONE                                                     ⬍
  NPC              5_NPC_NAME_LISANDRE                                       ⬍
  Timing           ON_SPAWN                                                 ⬍
  Activates        -1
  Required State   NONE                                                     ⬍
```

```
▼ SET_DATA-Command 'NPC_IDS,107'                                            X
  Interaction              SET_DATA - NPC_IDS
     Type          NONE                                                     ⬍
  NPC              107_NPC_NAME_ROSALIE                                     ⬍
  Timing           ON_SPAWN                                                 ⬍
  Activates        -1
  Required State   NONE                                                     ⬍
```

To add an NPC to a container you use the set data – npc ids interaction. The NPCs you can select are configured in the static game data. The dialogues are configured in XML files, which you can find in the "Dialog" folder of the static game data.

```
▼ START_DIALOGUE-Command ''                                                 X
  Interaction              START_DIALOGUE -
     Type          NONE                                                     ⬍
  Target SpawnID   34                       -SHRINE 34               SetMyID
  Timing           ON_EXECUTE                                               ⬍
  Activates        -1
  Required State   NONE                                                     ⬍
```

The start dialogue command is used to open the conversation screen and load the corresponding dialogues. For the target you need to select the NPC container which should be used.

```
▼ SET_DATA-Command 'INDOOR_SCENE,Shrine'                                    X
  Interaction              SET_DATA - INDOOR_SCENE
     Type          NONE                                                     ⬍
  IndoorScene      Enter a value!         IndoorScene      Shrine          ⬍
  Timing           ON_SPAWN                                                 ⬍
  Activates        -1
  Required State   NONE                                                     ⬍
```

With the set data – indoor scene you can define which 3D scene should be used for the conversation.

```
▼ SET_DATA-Command 'PREFAB,Prefabs/InteractiveObjects/ChangeLevel/ChangeLevel_IndoorScene'                    X
Interaction                              SET_DATA - PREFAB
   Type          NONE                                                                                      ▲▼
Path             Enter a value!          Prefabs/InteractiveObjects/ChangeLevel/ChangeLevel_IndoorScene
Timing           ON_SPAWN                                                                                  ▲▼
Activates        -1
Required State   NONE                                                                                      ▲▼
```

```
▼ SET_DATA-Command 'MINIMAP_TOOLTIP,LOCATION_SORPIGAL_CHAPEL_OF_ELRATH'                                       X
Interaction                          SET_DATA - MINIMAP_TOOLTIP
   Type          NONE                                                                                      ▲▼
Value            LOCATION_SORPIGAL_CHAPEL_OF_ELRATH
Timing           ON_SPAWN                                                                                  ▲▼
Activates        -1
Required State   NONE                                                                                      ▲▼
```

With these set data commands you can define which transition icon is used when the party stands in front of the NPC container and also which text should be displayed. Below you can see the sample from the shrine in Sorpigal. The text needs to be defined in the localization database.
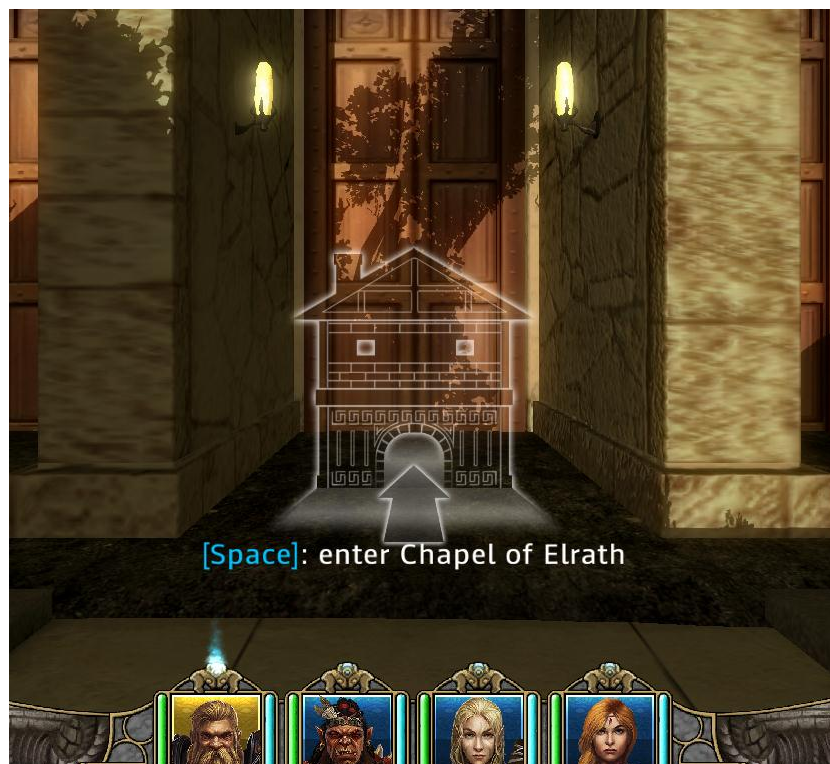
Spiders out of a barrel

This feature requires the use of several triggers and commands. This sample is good example of how different features can be combined to create new game behaviour.
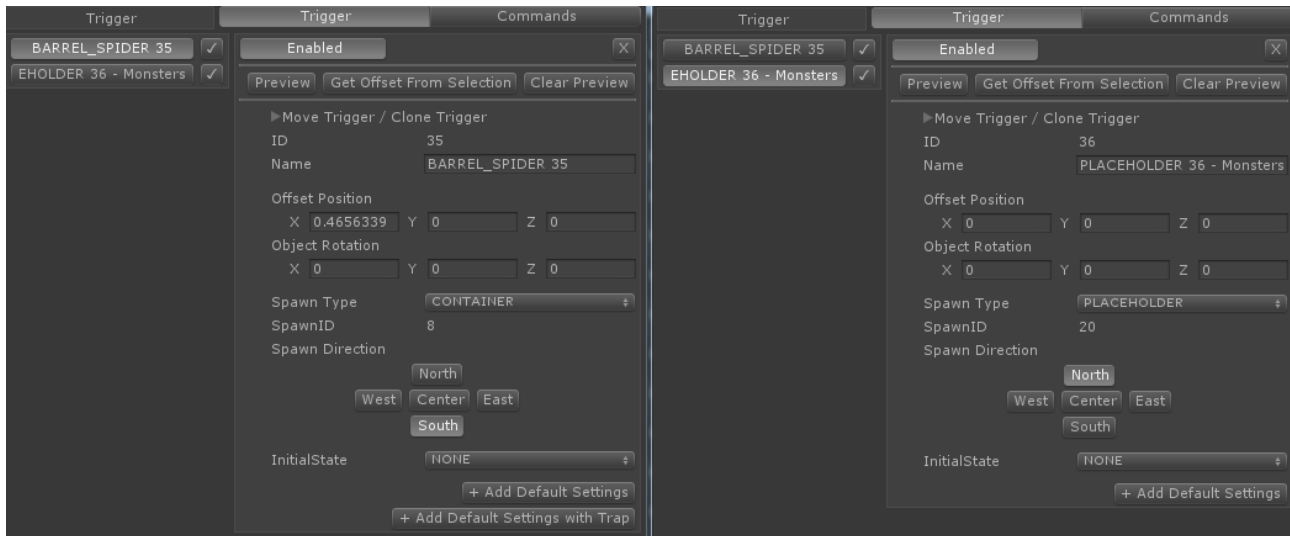
For this sample you need a placeholder trigger for the monster (the position where it should be spawned) and a container trigger for the barrel. In this sample the party is pushed back by one tile to make enough space for the spider. For this we also need a teleporter trigger which is placed on the tile on which the party should be teleported. The entire logic will be triggered when the party interacts with the barrel. So the barrel needs a series of interactions / commands.
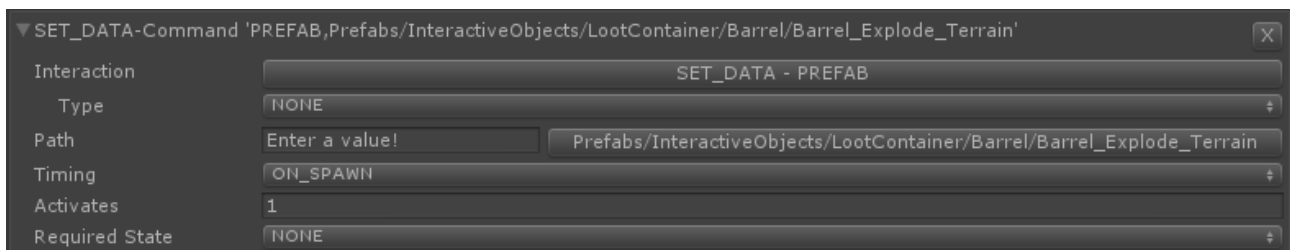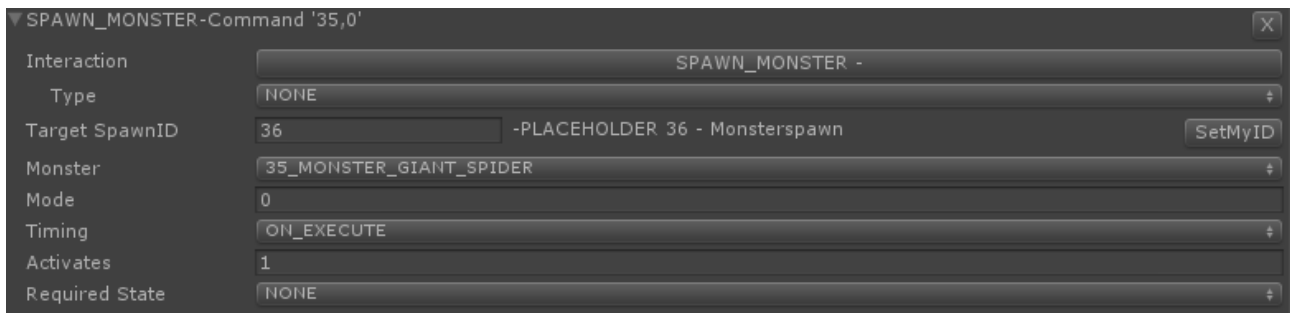

The first set data command assigns the barrel prefab to the container.
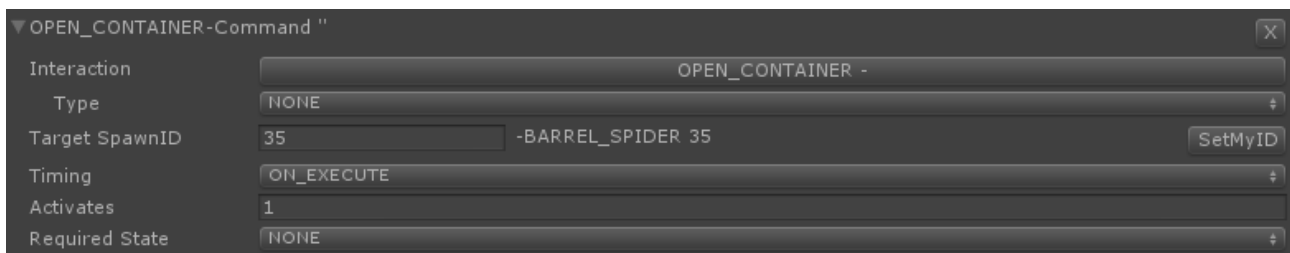
The second command is to open the container (to trigger the open animation of the container).

The teleport command will move the party by one tile. The target teleporter is referenced by Target Spawn ID.


The last command spawns the monster on the position of the placeholder we added in the first step. In the monster field you can select all monsters which are configured in the static game data. These commands are all triggered at the same time i.e. with one execute timing, which allows the party to be pushed back at the same time that the spider is spawned.

In the static game data you can define your own monsters / enemies with their own values. That way it is possible to add your own opponents to the game. At this time it is not possible to add new creature models and animations to the game with your mod. You can place animated models in the scene but the game will not interact with them. However, you can use all monsters which are part of the main game. So if you create a new monster in the data you can assign a monster model from the main game.

Further examples

Examples for interactions can be found in the sample dungeon and also in the game levels. The XML files can be accessed in the "Might and Magic X Legacy_Data/StreamingAssets/Maps" folder of your game installation. Additional documentation for commands, interactions and their values will follow shortly.
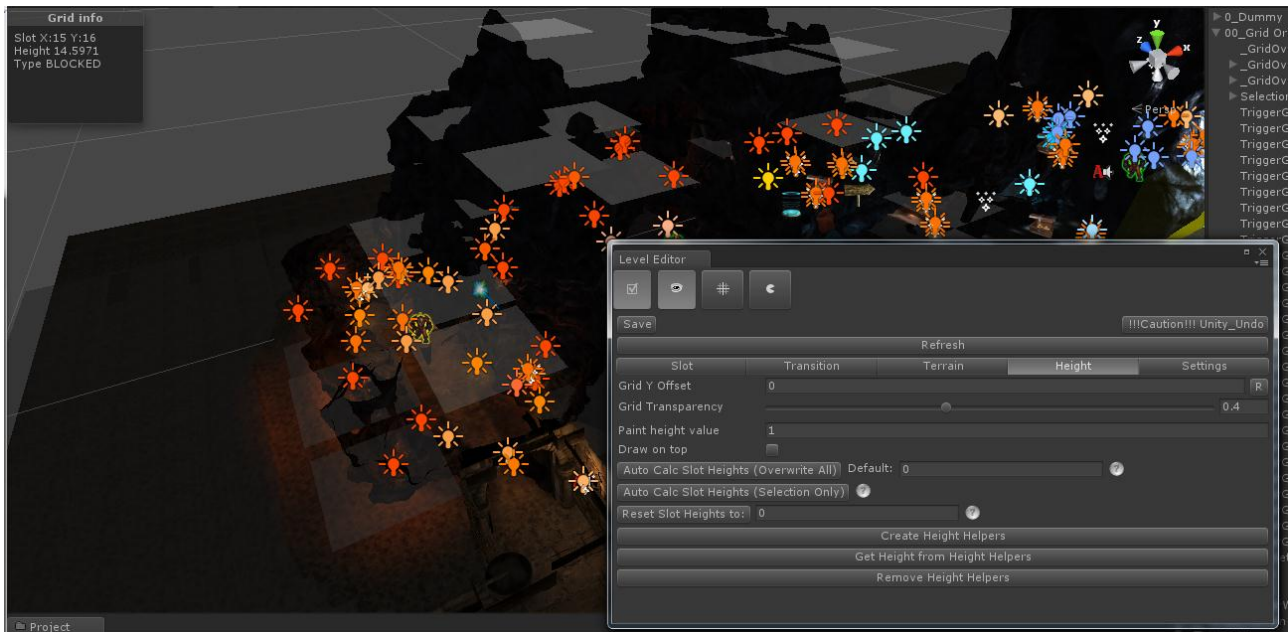
Height tab

With the height tab you can configure the height for each tile in the game. The party camera will then be moved by this defined offset. With this functionality we are implementing stairs, ramps and other vertical elements.

Export mod and load it in MMXL

The sample level can be exported directly. You need to open the ModInfo file in the folder "Assets/MMXL_ModKit/ModSample". In the "Inspector" tab you have the Build options available.
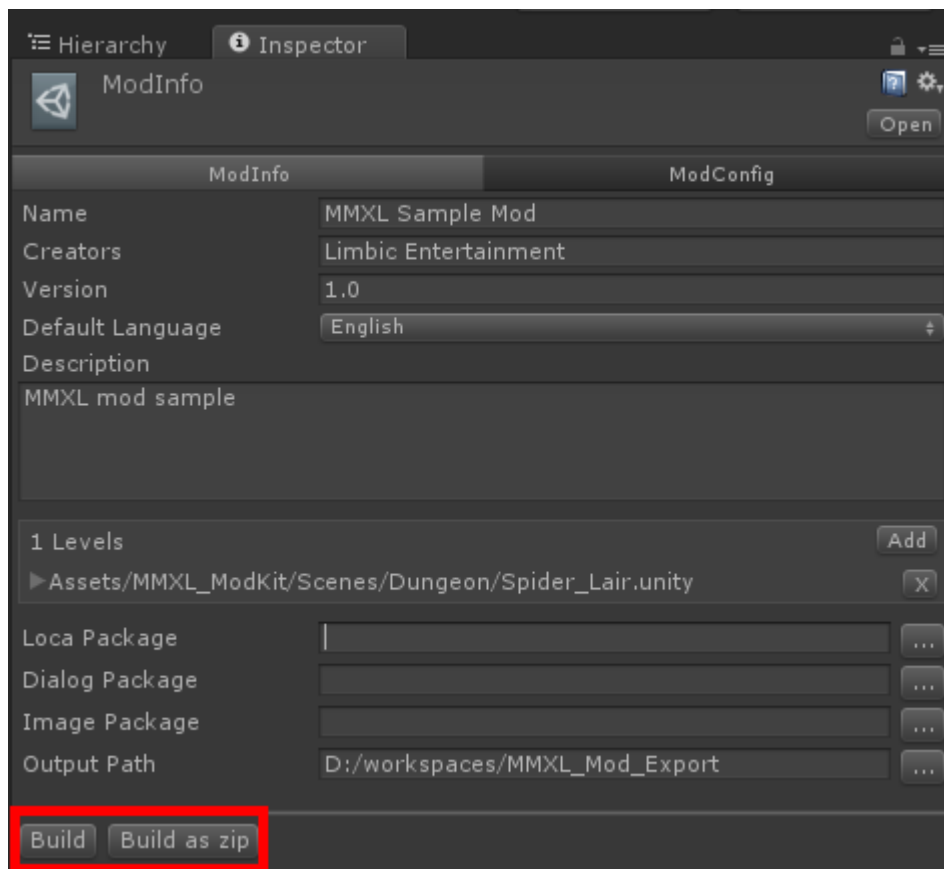
These two buttons are used for exporting the mod. It will build the asset bundles of the assigned scenes and also copy all data which is required to run the mod. You can decide if you want to export raw files or a zip archive (reduced size makes is easier to exchange the mod). To use a mod in the game it needs to be unzipped.

At the export location you will find a folder with the name of the mod. This folder contains all the files which the games needs in order to load the mod. The game requires this folder to be in a specific location, otherwise it will not find it. You need to copy the mod into the mod area of your save game folder (in MyDocuments/MightAndMagicXLegacy/Mods).
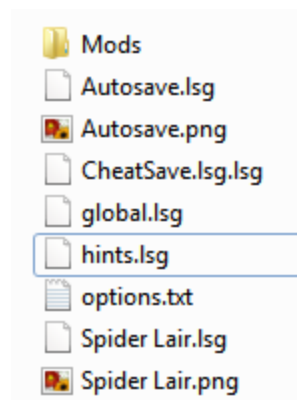
This folder will have been created after you started the game for the first time. Also the save games of your mod will be stored in the Mods folder.

After that you can start the game. In the start menu select "Extra", then "Mods". The following screen will popup showing all of the mods the game was able to find.

Here you need to select the mod you want on the left side and press "Load". After that the normal game is replaced with the mod (you will see the change in the version number of the game and also in the background graphic if configured). Now you can start your mod as usual, just by pressing start and entering the MMXL party creation screen. You can also load save games which are related to your mod.

How to: Creating a new level

Every level is defined by two components. The first component is the scene, which contains the 3D environment. The second component is the grid data, which contains the game logic. Both components will be created by the Scene Creator tool.
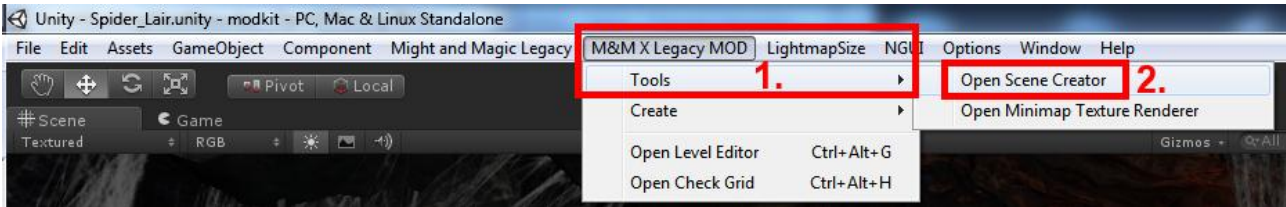
To open the Scene Creator you need to select "M&M X Legacy MOD" in the main menu, then "Tools" and then "Open Scene Creator".

Here you need to enter the name of your scene and also the target folder where you want to store your level files. To make the folder selection work you need to click on your folder on the right side of the "Project" tab (see below).
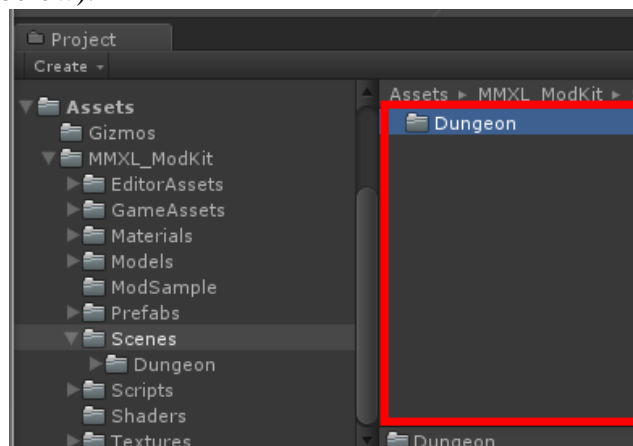
After you click on "Create Scene" your current scene will be closed and the new scene will be opened. From this point you can start to create the 3D environment and configure your grid.