

Achievements

This page describes how to use the achievements system included in the TopDown Engine and customize it to your game.

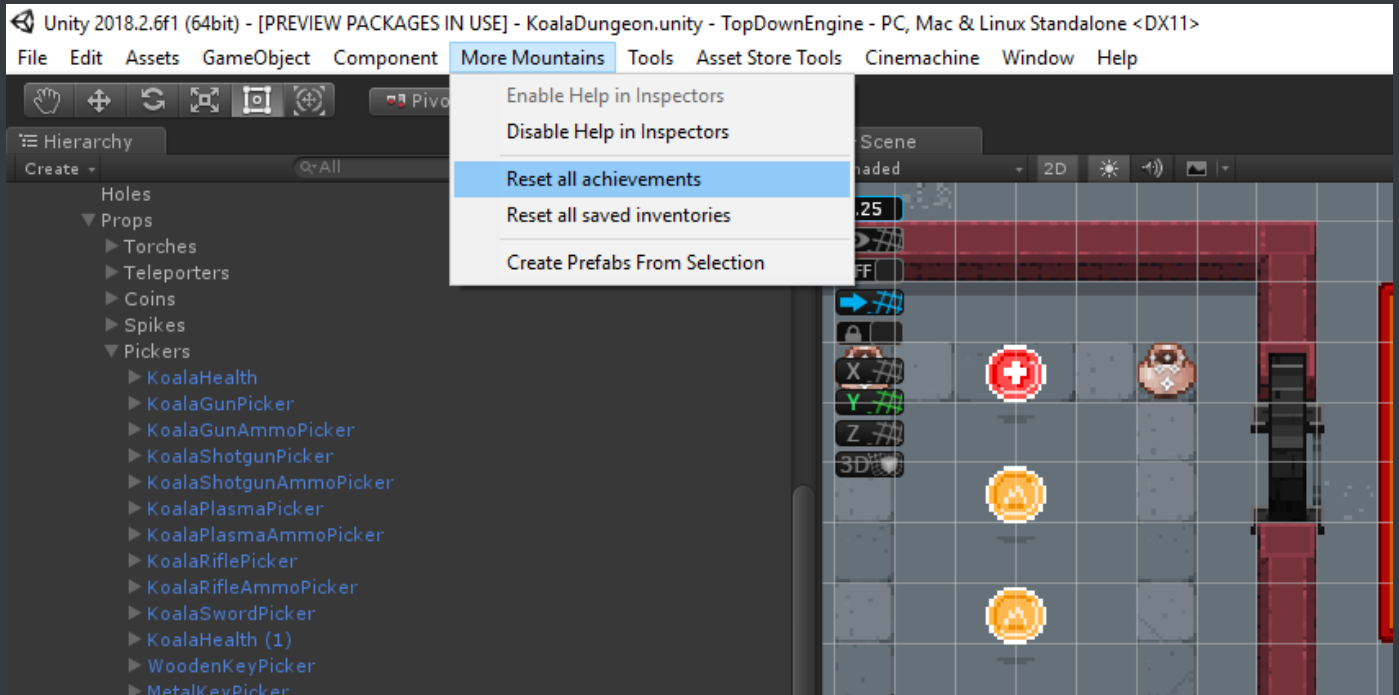
- [Introduction](#)
- [Resetting Achievements](#)
- [The Achievement List](#)
- [Unlocking achievements](#)
- [MMAchievementRules](#)
- [Achievements Display](#)
- [Interfacing with Steam, iOS, etc](#)

Introduction

The TopDown Engine includes a simple to use yet powerful achievements system. It'll allow you to define, trigger and save achievements. Plus it's **very modular** and easy to extend, if you want to plug it to a specific platform's social API for example.

Resetting Achievements

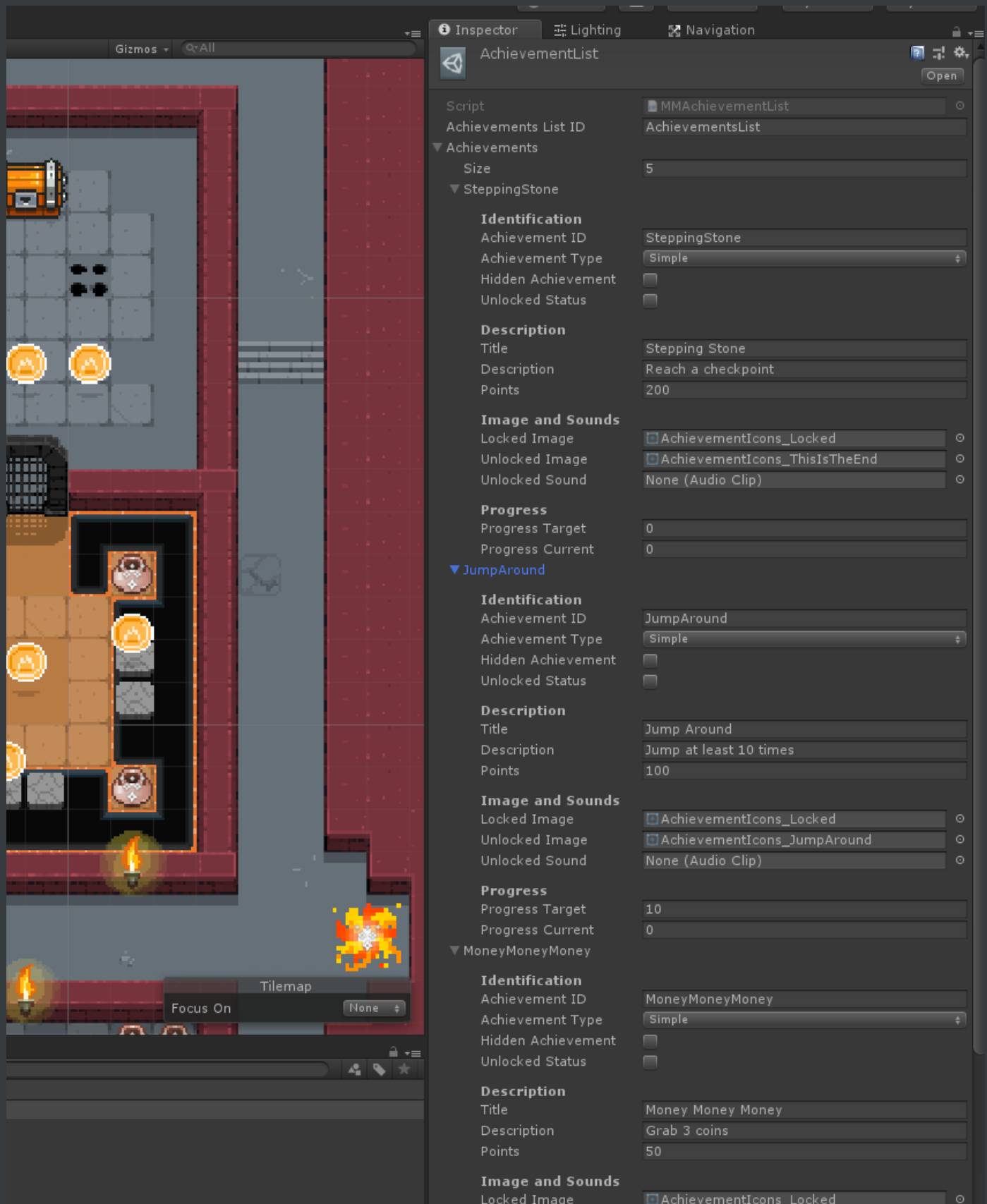
The first thing you need to know about the achievement system is **how to reset it** (that's the most asked question about it anyway). This is done either at the bottom of the Achievement's list inspector, or via the top menu, by clicking More Mountains > Reset all achievements. :



The More Mountains top menu.

The Achievement List

The second important thing to know about achievements is **how to define them**. This is done via a Scriptable Object, the AchievementsList. You can create your own, but there's already one in the asset that you can modify. You can find it under Common/Resources/Achievements. You can just select it and modify its content via its inspector. Note that the list has to be within a Resources folder, otherwise Unity won't be able to load it.



The Achievements List.

As you can see on the picture above, an achievement is made of an ID (used to unlock it via your scripts), a type, title, description, associated images and sounds, and potential progress. There are 2 types of achievements : simple (finish the game, find the treasure, unique stuff like that), and progress based ones (collect 50 coins, die 20 times, etc). In the case of progress based ones, you'll need to specify the target progress needed to unlock the achievement. All achievements can only be unlocked once, unless you reset them of course.

Unlocking achievements

Once you've added an achievement to the list, it's time to switch to code to make it unlockable. This can be done from any class, and of course where exactly you'll do it depends on your game and your achievements. There are a few examples in the engine so you can get a better understanding of how it's done. Note that this achievements system uses **Events**, you may want to [read this page](#) if you want to get a better understanding of how it all works.

So unlocking achievements is very simple and can be done in a single line of code. Where you put that line of code depends on what you want to do. I'd recommend extending the AchievementRules class so you centralize them all in the same place, and have that class listen to events and act on them (check out this class' code if you want to know more about how to do it). But you can also do it from anywhere in your code if you prefer.

Here's how you unlock a simple achievement called "SteppingStone" (of course you'll want to replace that with your achievement's ID) :

```
MMAchievementManager.UnlockAchievement("SteppingStone");
```

And here's how you add progress to a progress based achievement :

```
MMAchievementManager.AddProgress ("JumpAround", 1);
```

Of course you can replace that "1" with any value of your choice to get progress faster.

MMAchievementRules

As mentioned earlier, right now achievements are triggered by the AchievementRules class. It's a MonoBehaviour, so you need to add it to an empty GameObject in each of the scenes where you want achievements to be unlocked from. Achievements are global though, so if you unlock one in Level A, you won't be able to unlock it again in Level B. If you don't want to use the AchievementRules class though, make sure you add these lines somewhere in a manager at the start of your game :

```
// we load the list of achievements, stored in a ScriptableObject in our
Resources folder.
MMAchievementManager.LoadAchievementList ();
// we load our saved file, to update that list with the saved values.
MMAchievementManager.LoadSavedAchievements ();
```

Achievements Display



Unlocking an achievement displays a small popup in the screen's corner.

The asset includes an achievement displayer. To use it, you'll need to have a container for them. I've found that a Vertical Layout Group works quite well, but you can use anything you want. That container will need to have a `MMAchievementDisplayer` component added to it. From its inspector you'll be able to specify what prefab to instantiate when an achievement is unlocked, for how long it should remain on screen, and how fast the transition should be. You can of course modify or create a new prefab to display the achievements exactly the way you want. It'll need an `MMAchievementDisplay` component added to it so you can bind GUI elements.

Interfacing with Steam, iOS, etc

To avoid creating **dependencies** to external code, this asset doesn't include ready-to-use interfaces for Steam, Google Play, iOS or other platforms. But this achievement system has been built with that kind of evolution in mind, so you can easily extend it and call the desired methods, as they all share more or less similar structures for achievements. The

best way to do so is to create a new class that listens for `MMAchievementUnlockedEvents` (these are of course triggered everytime an achievement is unlocked) and pass that achievement to the target social platform's API.