



TopDown Engine Support

Copyright © Pixel Crushers. All rights reserved.

TopDown Engine © More Mountains.

# Contents

- Chapter 1: Introduction..... 4
- Chapter 2: Demo Scene..... 4
- Chapter 3: Setup..... 5
  - How To Set Up Scenes..... 5
  - Demo Scene Setup..... 5
- Chapter 6: Save System..... 6

# Chapter 1: Introduction

This manual describes how to use More Mountains' TopDown Engine with Quest Machine.

Quest Machine's TopDown Engine integration provides these features:

- Inventory Engine “has items” quest condition.
- Inventory Engine “add/remove items” quest action.
- Message listener that sends Message System messages when inventories change.
- Inventory Engine reward system for procedurally-generated quests.
- Save inventories using the Pixel Crushers Save System.

Note that the TopDown Engine integration also works well with the Inventory Engine integration.

## Chapter 2: Demo Scene



The TopDown Engine integration package's demo scene contains a quest to defeat a ninja. Talk to the green Guide NPC. Note: TopDown Engine also comes with Inventory Engine. Quest Machine has its own integration for Inventory Engine, including a separate demo scene that shows how to run quests that involve inventory items.

The example scene is in 2D. Remember to add the Scripting Define Symbol **"USE\_PHYSICS2D"** as described in the Quest Machine manual.

If the camera doesn't follow the player in this scene or TopDown's original Koala2D scene, make sure the Scripting Define Symbol CINEMACHINE\_INSTALLED is set.

## Chapter 3: Setup

To enable Inventory Engine integration, import these two packages:

- **Plugins ► Pixel Crushers ► Common ► Third Party Support ► TopDown Engine Support**
- **Plugins ► Pixel Crushers ► Quest Machine ► Third Party Support ► TopDown Engine Support**

If you want to use Inventory Engine, also import the corresponding Inventory Engine Support packages.

### How To Set Up Scenes

Use these steps to set up a TopDown Engine scene for Quest Machine:

- Add the Quest Machine prefab.
- Add a PauseTopDown component to the Quest Machine GameObject.
- Create a quest database, assign it to the Quest Machine GameObject, and add quests to it.
- Add a Quest Journal component to your player prefab.
- Add Quest Giver components to your quest giver NPCs. You can also add a TDE Button Activated component to call QuestGiver.StartDialogueWithPlayer.
- To perform actions when a character such as an NPC dies or is revived, add an On Death Event component. You can also add a Quest Control component if you want the events to do something quest-related such as send a message system message.

### Demo Scene Setup

The demo scene was set up using the steps above. In addition, it also has:

- A UI Button to toggle the quest journal open/closed.
- An On Death Event component on the quest's kill target. Its OnDeath() event calls QuestControl.SendMessageToMessageSystem to send "Killed" + "Ninja", which is the message that the quest listens for.

If you're using More Mountains' Inventory Engine, see the guide included in the Inventory Engine integration. In particular, please keep in mind that Inventory Engine disables UI navigation while the inventory is closed. If you want to be able to navigate other UIs with a joystick/keyboard such menus, see the setup section.

## Chapter 6: Save System

For use with Quest Machine's save system, the script `TDECharacterSaver.cs` is a starter script for saving TDE data. You can add it to your player prefab to save the player's health, or duplicate it as a new script and customize your new script if you want to save additional player data.

To tie the Inventory Engine to Quest Machine's Save System, add an **Inventory Engine Saver** component to any inventory `GameObjects`.

If you're using `MMSceneLoadingManager` to handle scene transitions, use **PixelCrushersMMSceneLoadManager** instead. This subclass of `MMSceneLoadingManager` ties in the Pixel Crushers save system to save the state of the outgoing scene and then apply saved state to the newly-loaded scene.

### Final Advice

Pick apart the demo scene to see how it works. If you have any questions, we're here to help! Please visit the [forum](#) or email [support@pixelcrushers.com](mailto:support@pixelcrushers.com) any time!