



SINCE 2010

# Sliding Puzzle

Documentation | 21-07-22





# Table of Contents

1. Get started quickly	3
2. Introduction	4
3. Set-Up	5
4. API	7
SlidingPuzzleManager	7
SlideTileController	8
5. Known Limitations	9
6. Support and feedback	10



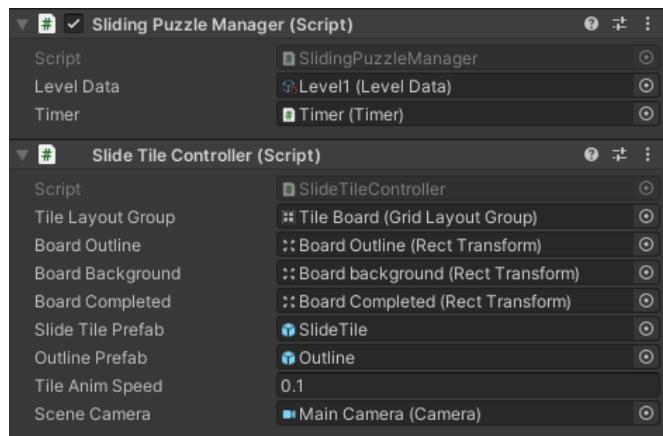
# 1. Get started quickly

To get started quickly, follow these steps.

For a more detailed setup, go to the Set-Up section of this document.

Under the ‘Demo/Scenes’ folder, you can find a demo scene containing an example implementation of the game. The main components needed for the game are on the PuzzleManager object.

These are the **SlidingPuzzleManager** and **SlideTileController** components. They contain all the needed object references and level data.



If you want to add a custom level, right click in the project tab and go to ‘Create/DTT/Mini Game/Sliding Puzzle/LeveData’. This will create a **LeveData** asset, which allows you to specify the puzzle image and dimensions. You can add this level to the **SlidingPuzzleLevelSelectHandler** configs list to be able to choose the level in the level select.



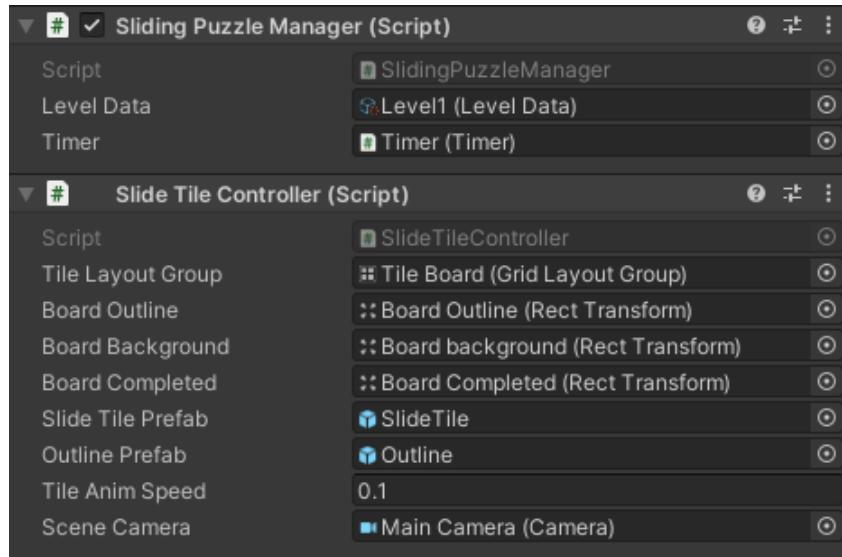
## 2. Introduction

**DTT Sliding Puzzle** is a Unity asset that allows you to easily implement a sliding puzzle game into your project. The asset allows you to customize the playing board and difficulty of the puzzle. Using a Scriptable Object you can change the difficulty of the game, allowing you to have different difficulty settings per level. You can also use any sprites of your project as images for the levels of the puzzle. All this allows for an easy and fast way to implement a simple sliding puzzle into your project.





### 3. Set-Up

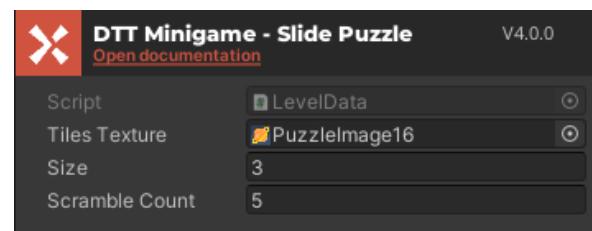


1. To start off, place the **SlidingPuzzleManager** component on an object in your scene. This will also add a **SlideTileController**. The **SlidingPuzzleManager** can be called to start and restart the game. The **TileSlideController** handles the creation, clearing and interaction of the tile board.

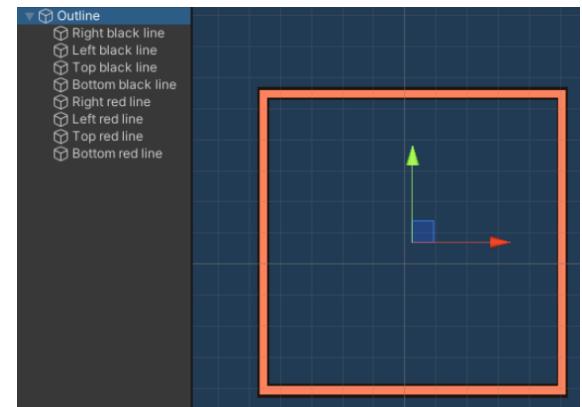
2. Create scriptable objects of

**LevelData** to add the Image, size and the scramble count for each desired level.

3. You need to create a prefab for the tiles used inside the board. For this you all you need to do is create a new game object with rect transform, and add the following components: Image, Button, Canvas Group and Tile Drag (From the Tile Drag Script).



4. You will also need to create a prefab for the outline of the board. Create a new game object and add as child objects new game objects with the Image component. Create at least one for each side of the board. The design for this outline will depend on your design. You can also have multiple Images for each side.





5. Create an object with the rect transform component inside your game canvas. This will hold all the objects required for the board. Then create 4 separate objects inside of it and make sure they have a rect transform component as well. Add them onto the board references of the **TileSlideController**. Tile Board should have a layout group that manages the structure of the board.
6. Create a game object with the **Timer** component. This will handle timing the game. You can add a **TimerUI** component to it if you want the UI to display the time.
7. After these steps you can finally start the game by either calling the **StartGame** method.



## 4. API

### SlidingPuzzleManager

*Handles starting, pausing, continuing and finishing the game.*

Property Name	Type	Description
TileController	TileController	Handles placing and moving the puzzle tiles.
IsPaused	bool	Is true when the game is paused.
IsActive	bool	Is true when the game has started and hasn't finished.
Finish	event	Event invoked when the game has finished.
Started	event	Event invoked when the game has started.
IsPaused	bool	Is true when the game is paused.
IsActive	bool	Is true when the game has started and hasn't finished.
CorrectGuess	event	Event invoked when a correct guess has been made.
Finish	event	Event invoked when the game has finished.
Started	event	Event invoked when the game has started.

Method name	Return Type	Parameters	Description
StartGame	void		Starts the game using the default game settings.
StartGame	void	RhythmConfig config	Starts the game using custom game settings.



Pause	void		Pauses the game.
Continue	void		Continues the game.
ForceFinish	void		Is called when the user wants to end the game.

## SlideTileController

*Handles instantiating and moving tiles.*

Property Name	Type	Description
TileMoveAmount	float	Amount of times the tiles were moved.
IsScrambling	bool	Is true when the scramble animation is playing
onTileMoved	event	Invoked when a tile is moved.
onFirstMove	event	Invoked when the first tile of the game is moved.
onScrambleEnd	event	Invoked when the scrambling animation ended.
TileMoveDir	Vector2Int	Direction the tile is moving in.



## 5. Known Limitations

- None.



## 6. Support and feedback

If you have any questions regarding the use of this asset, we are happy to help you out.

Always feel free to contact us at:

[unity-support@d-tt.nl](mailto:unity-support@d-tt.nl)

(We typically respond within 1-2 business days)

We are actively developing this asset, with many future updates and extensions already planned. We are eager to include feedback from our users in future updates, be they 'quality of life' improvements, new features, bug fixes or anything else that can help you improve your experience with this asset. You can reach us at the email above.

Reviews and ratings are very much appreciated as they help us raise awareness and to improve our assets.

### DTT stands for Doing Things Together

DTT is an app, web and game development agency based in the centre of Amsterdam. Established in 2010, DTT has over a decade of experience in mobile, game, and web based technology.

Our game department primarily works in Unity where we put significant emphasis on the development of internal packages, allowing us to efficiently reuse code between projects. To support the Unity community, we are publishing a selection of our internal packages on the Asset Store, including this one.

More information about DTT (including our clients, projects and vacancies) can be found here:

<https://www.d-tt.nl/en/>