



NavTiles: Getting Started

This tutorial is intended for people who have just started working with NavTiles. It will show you how to easily integrate NavTiles into your Unity project. This tutorial is also available online, along with the rest of the NavTiles documentation, at [this link](#).

New Project

When starting a new 2D project, it is recommended that you build your tilemap using Unity's features. If you need help setting up tilemaps, you can take a look at the following online resources:

- [Tilemap Manual](#)
- [Creating Tilemaps](#)
- [Tile Palette](#)
- [Painting on Tilemaps](#)

If you prefer a video tutorial, we recommend watching [Brackeys' tilemaps video](#).

Existing Project

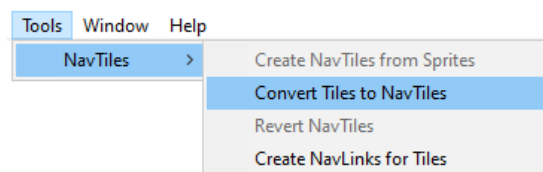
When you have an existing project with a Tilemap setup, the first step is to import the NavTiles asset through the asset store. You may be prompted with a setup dialog; this is necessary to place some script icons in the gizmos folder. If you are using an older version of Unity, you might be prompted to change the scripting API of the project. If you deny, the asset will not work properly.

After importing NavTiles, it is time to create some actual NavTiles. The easiest way to do this is to convert existing tiles to NavTiles. Navigate to the folder where your tiles are saved, and select all tiles that will influence pathfinding.



Selecting Unity tiles.

With the tiles selected, convert them by clicking on **Tools > NavTiles > Convert Tiles to NavTiles**, using the toolbar at the top of the screen. Properties like textures will be preserved, so don't worry about losing data. You can convert NavTiles back into Unity tiles by selecting them and clicking on **Tools > NavTiles > Revert NavTiles**.

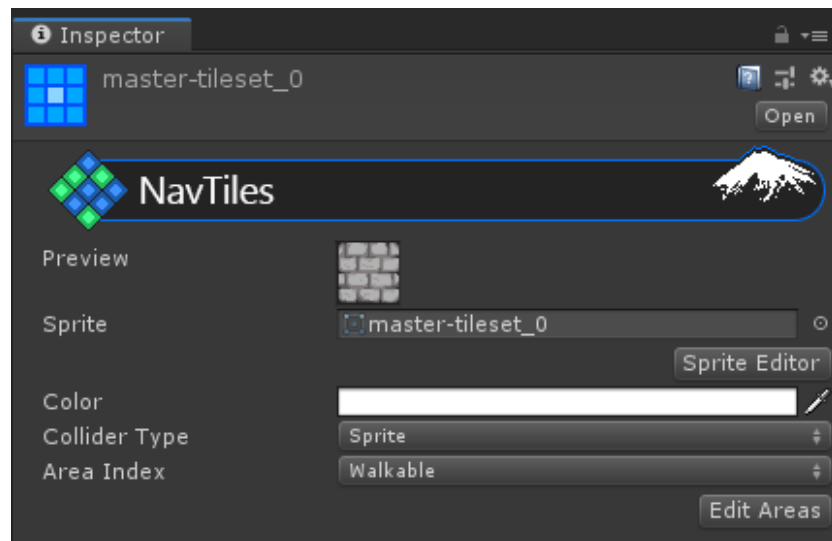


Converting tiles to NavTiles in the tools menu.



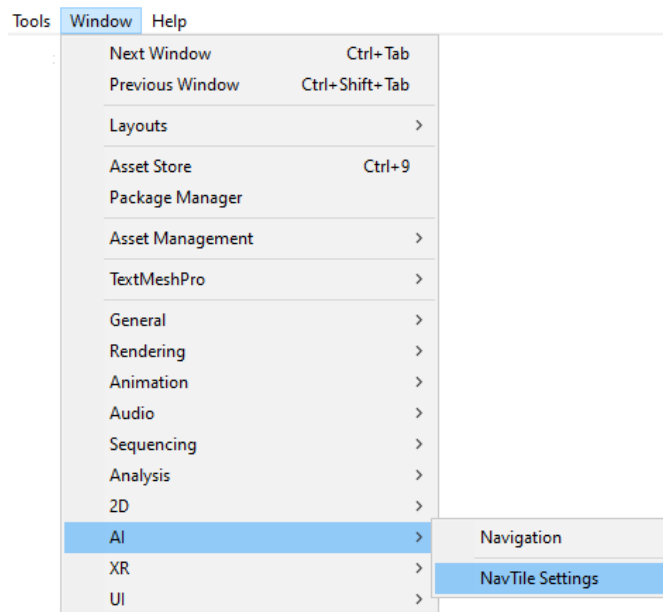
If the convert button is greyed out, your selection likely contains something other than a Unity tile. If you are trying to convert a different type of tile, please take a look at the online page for [setting up NavTiles](#).

After converting, the tile assets should have a blue icon, rather than the default purple icon. The inspector should look different as well. NavTiles have an extra property called an "Area Type", which defaults to "Walkable". You can change it to "Non Walkable" if this tile should block traversing agents, like a wall or similar obstacle would do.



An example of the NavTile inspector.

After successfully converting your tiles, it is good practice to open the [NavTile Settings Window](#). This can be done by clicking on **Window > AI > NavTile Settings** with the toolbar at the top of the screen.

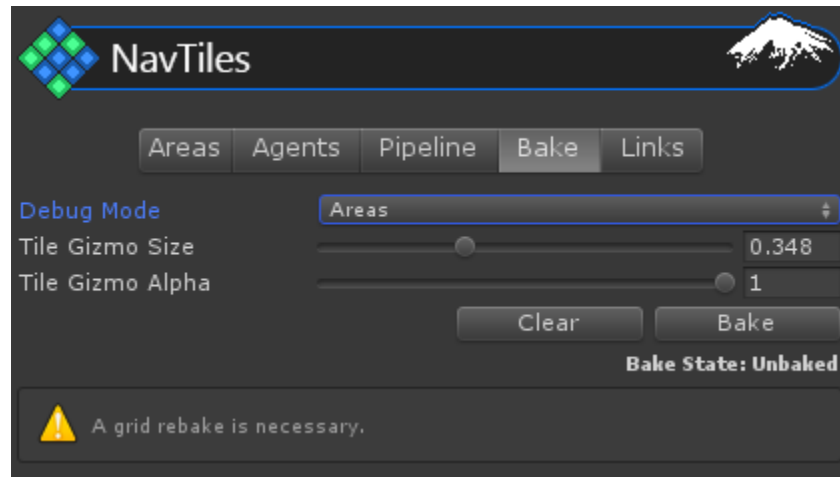


Opening the NavTile Settings Window.



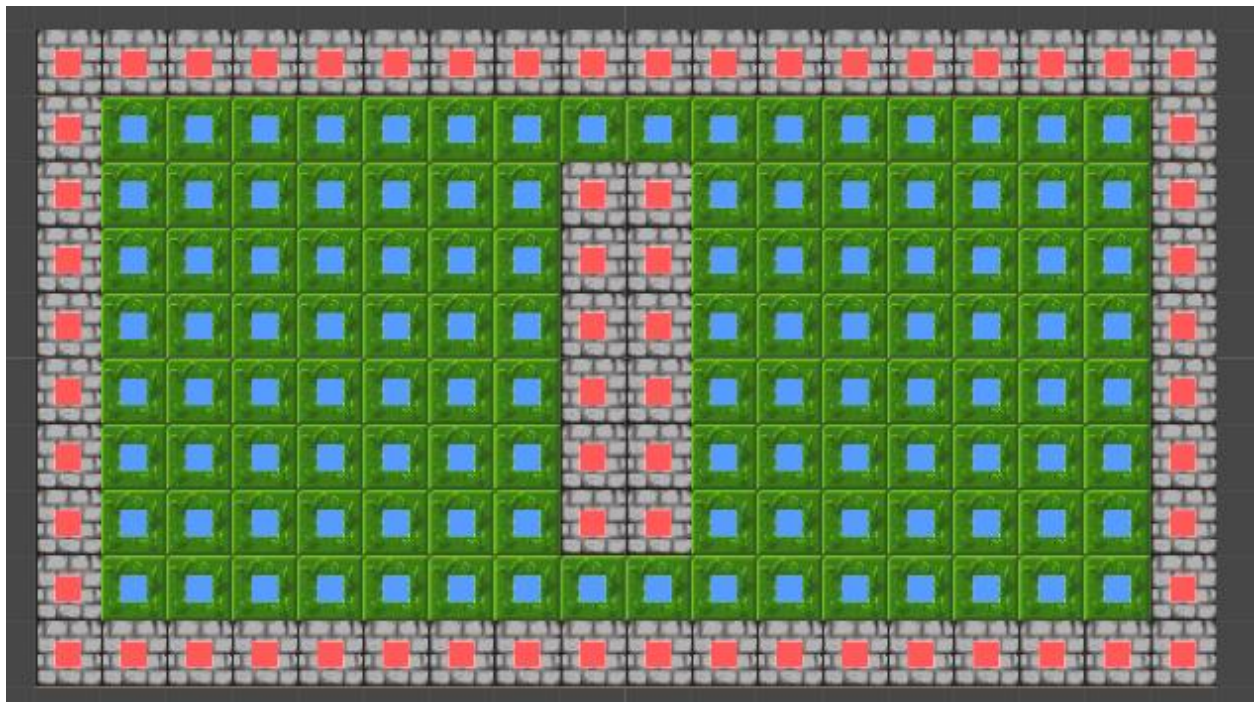
This will open the NavTile Settings window on the [areas tab](#) with two areas already filled in. This window controls most of the functionality of NavTiles.

Next, switch to the [bake tab](#) using the tabs bar at the top of the inspector. This tab is used to bake the NavTile grid and to visualize debugging information when the grid has been baked. Click the "Bake" button to bake the NavTile grid.



The bake tab of the NavTile Settings Window.

Baking the grid will create a data container in the form of a ScriptableObject. This file will be located inside the folder of the currently active scene. If the "Debug Mode" setting is set to "Areas" you should see colored shapes overlaid on your tiles in the scene view.

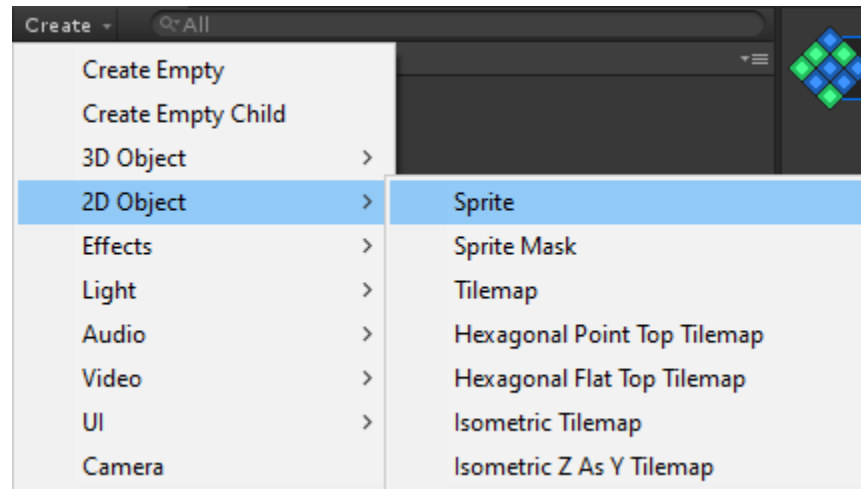


A scene view of correctly baked NavTiles, visualized by the "Areas" debug setting.



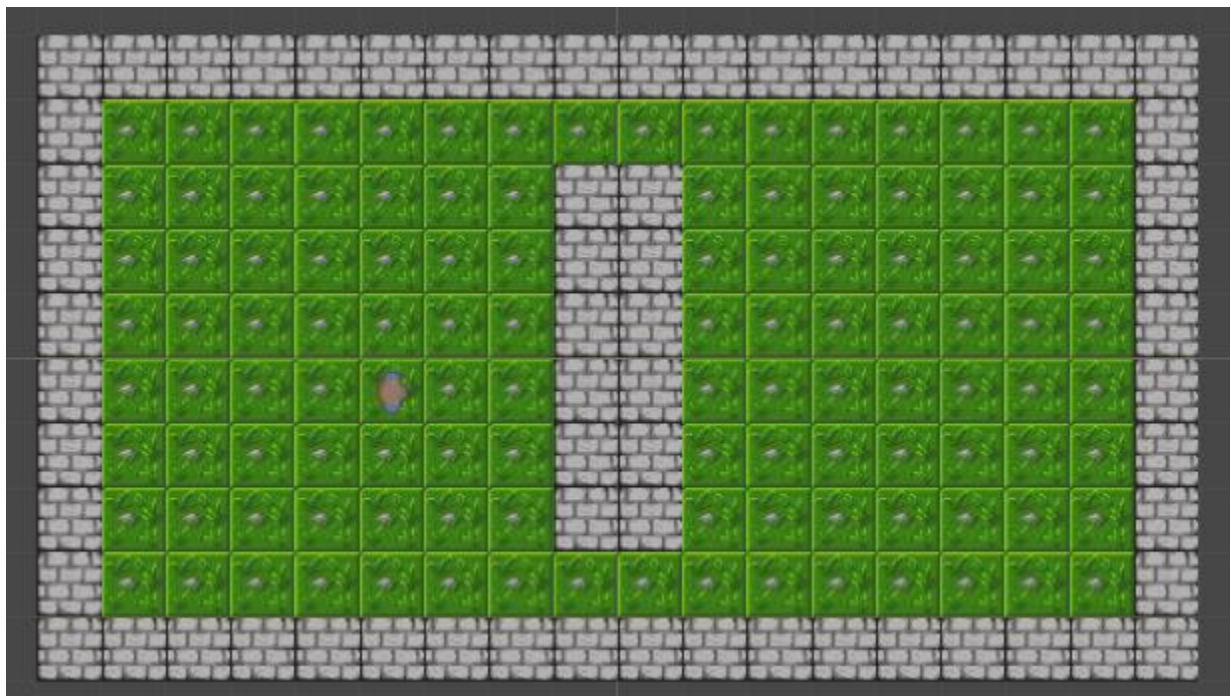
These shapes visualize how the NavTile grid is baked and allow you to easily spot errors. Each color represents an area type, which you can find and edit in the areas tab. By default, these colors are red (Non Walkable) and blue (Walkable).

Great! We now have a baked NavTile grid, ready for pathfinding! Let's create a character and have it walk around on our grid. Start by creating a sprite in the hierarchy through Unity's "Create" menu.



Creating a sprite through the create menu.

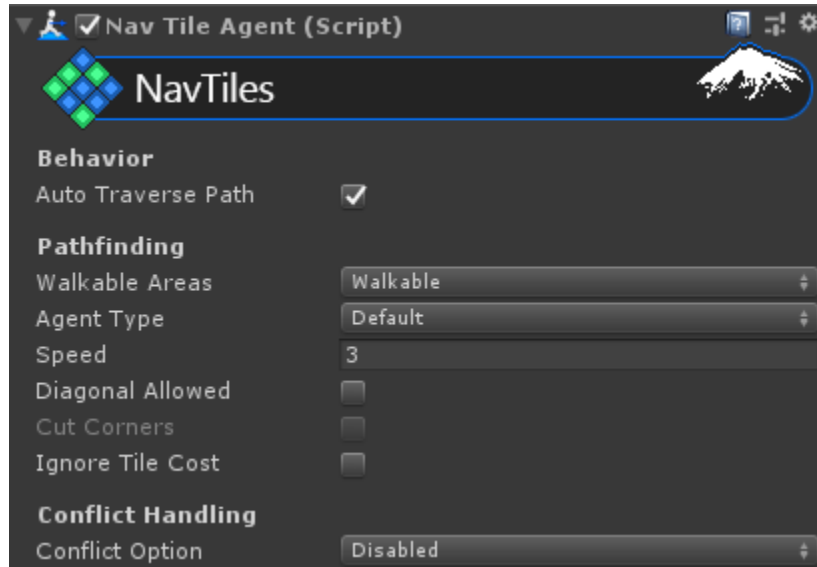
After creating the sprite, reset its transform, assign a sprite, and change its ordering layer to make the sprite visible. Drag the sprite into the scene view to place the character somewhere on a walkable tile.



The character placed in the scene view on a walkable tile.

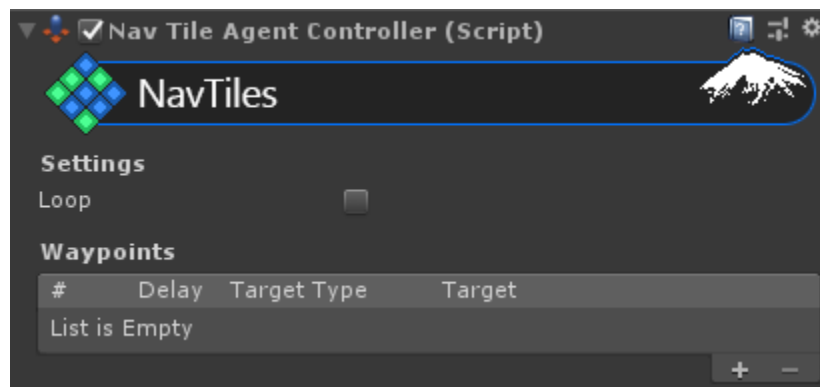


The character is not able to walk around on the grid yet. To enable pathfinding, it needs a [NavTileAgent](#) component. Add this to the character by going to the inspector and clicking "Add Component". You can find the correct component by typing "NavTileAgent" into the search bar, or you can click on **NavTile > Nav Tile Agent** through the toolbar at the top of the editor.



The NavTileAgent inspector with default settings.

The NavTileAgent has a lot of properties that you can toy around with, but for now the default settings will do fine. With the NavTileAgent component added, our character is able to traverse the NavTile grid. However, the NavTileAgent will not do so on its own. It has to be told to move somewhere. This can be done in many ways. The simplest way is to add another component: the NavTileAgentController. This component is a relatively simple script that allows you to give basic move commands without writing any code.

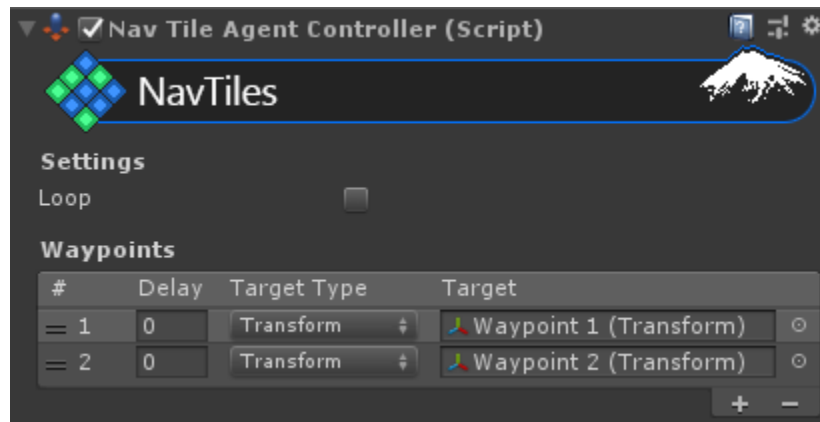


The NavTileAgentController inspector with default settings.

The NavTileAgentController has a list of waypoints that the NavTileAgent will traverse. You can add waypoints by clicking on the plus. Let's start by adding two new waypoints. Every waypoint can be set with one of three target types: a transform, a world position, or a grid coordinate. Depending on the situation, any one of these may have your preference. Let's select the transform option so we have a visual representation of our waypoints in the scene view. Create two empty GameObjects and place



them on walkable tiles in the scene. Next, drag these newly created objects into the transform fields of the waypoints, located under the "Target" column of the NavTileAgentController inspector.



A NavTileAgentController with its waypoints set to transforms.

There you have it! If you press the play button and start the scene, the character should move to each waypoint in order, walking around "Non Walkable" tiles. Feel free to play around with the settings of the NavTileAgent and its controller. For more in-depth information, take a look at the specific documentation pages for each component included in NavTiles.

If you have any issues, questions, or feedback, feel free to send us an email: info@snowcap-games.com.