Sprites

Programming – Cross Platform Development

Last modified 11/02/16 by Sam Cartwright

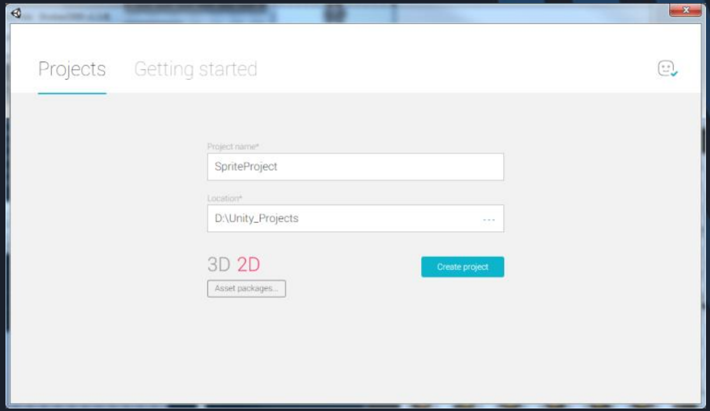
aie
SPECIALIST EDUCATORS IN
GAMES, ANIMATION & FILM VFX

Contents

- Using Unity’s 2D features
- Importing Sprite sheets
  - Packing sprites
  - Animating sprites
- 2D Physics

Unity 2D

- Unity started out as a 3D game engine
  - The tools for making 2D games was fairly limited
- Many small teams used Unity for their 2D projects, often building their own toolsets on top
- Because of this, Unity added a number of tools to make 2D game development much easier
  - Sprite based animation editing
  - Texture packing
  - 2D Physics support through Box2D
- When creating a project, you can specify whether you would like it to default to a 2D or 3D project
  - There is nothing special about what kind of project you make. Selecting 2D or 3D only changes some default settings.



Project name*: SpriteProject

Location*: D:\Unity_Projects

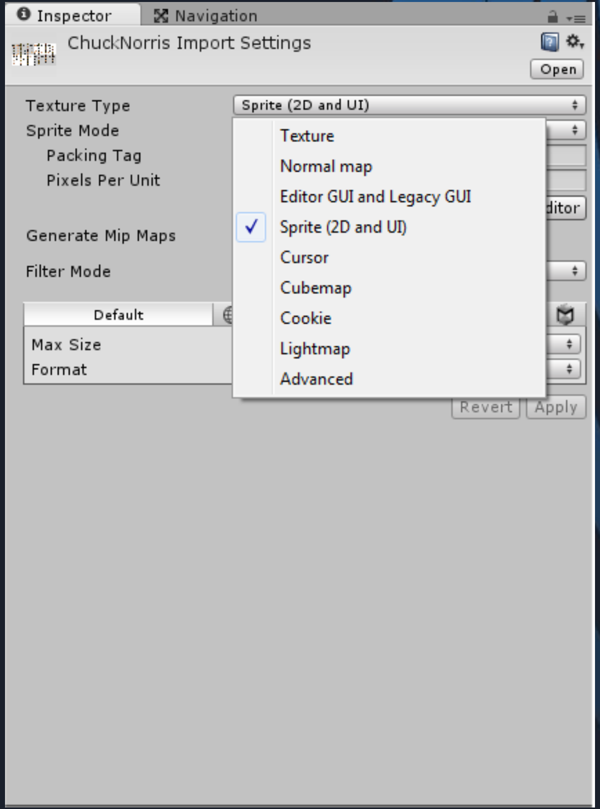
3D 2D

Unity Sprites

- Sprites in Unity are catch-all objects for 2D assets in your game. This includes:
  - Packed textures
  - Animated sprite sheets
- Sprites have few advantages over regular textures
  - The texture can be cut up into valid regions, allowing you to group multiple images into a single texture
  - The pivot point of a sprite can be changed. This is the point it translates and rotates around.
- If you are making a 2D game, most graphical assets in your project will be sprites.

Importing Sprites

- Sprites are built from imported textures
  - In a 2D project, any image you add to the project will be set to sprite by default.
  - If your texture is not already a sprite, you can change it to one in the inspector for the asset



ChuckNorris Import Settings

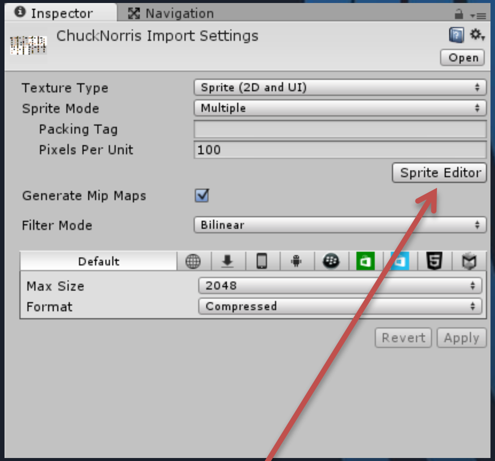
Texture Type Sprite (2D and UI)

The Sprite Editor

- A sprite texture can contain a single graphic element but it is often more convenient to combine several related graphics together into a single image.

- EG - The image could contain component parts of a single character, as with a car whose wheels move independently of the body.

- We can use Unity’s sprite editor to extract elements from a composite image.

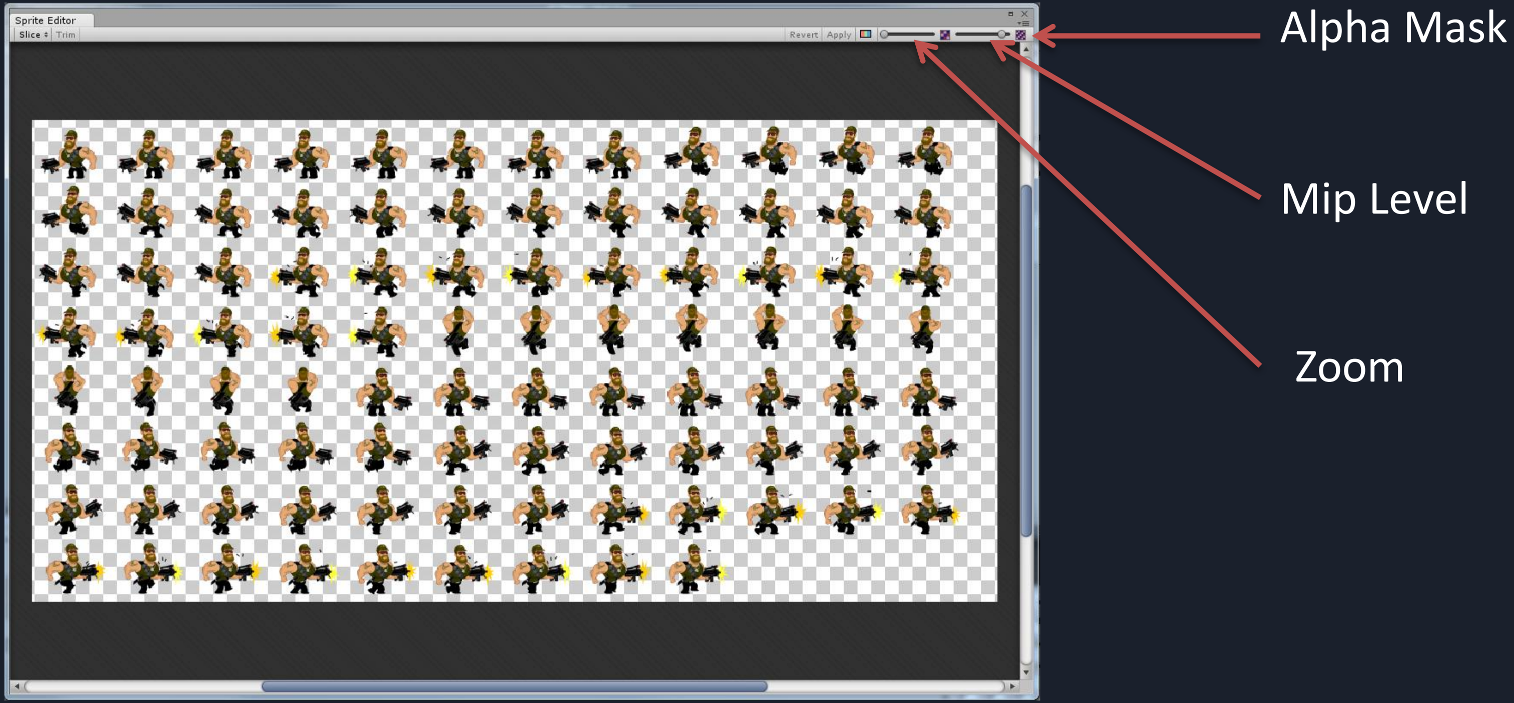


ChuckNorris Import Settings

Texture Type: Sprite (2D and UI)
Sprite Mode: Multiple
Packing Tag:
Pixels Per Unit: 100
Generate Mip Maps: ☑
Filter Mode: Bilinear
Max Size: 2048
Format: Compressed

Opening the sprite editor

The Sprite Editor



Alpha Mask

Mip Level

Zoom

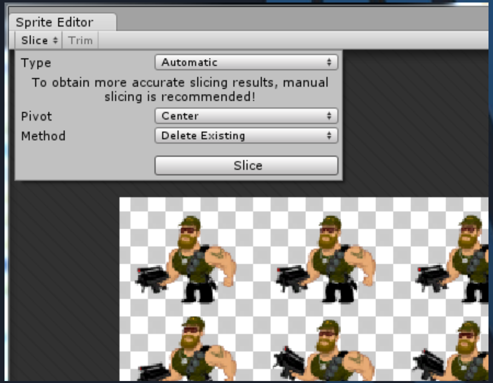
The Slice Menu

- The most important part of the Sprite Editor is the Slice menu.

- This is how you split your sprite into multiple sub-sprites

- You select the type
  - Automatic will search for sprites and wrap them in tight bounding boxes
  - Grid will split the sprite evenly
- The pivot for all the generated sprites
- The method decides what to do with existing sprites

- Once you’ve selected your options, click Slice to create your sprites.



Type Automatic

To obtain more accurate slicing results, manual slicing is recommended!

Pivot Center

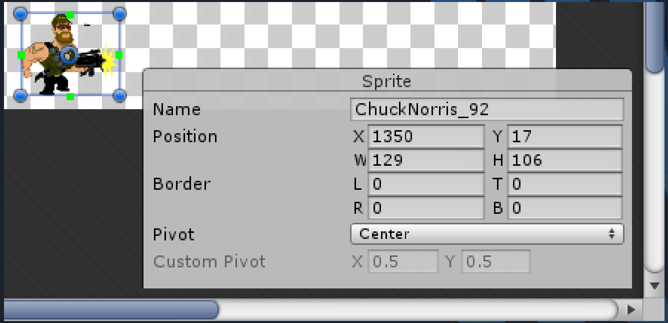
Method Delete Existing

Manual Slicing

- If you want to create your own sprites, or tweak the pre-generated ones, you can manually edit the slices

- Click and drag where there isn't a sprite to create a new one, or click on an existing one to edit it.

- You can edit
  - The position and size
  - The border
  - The pivot



Sprite

Name: ChuckNorris_92

Position: X 1350, Y 17, W 129, H 106

Border: L 0, T 0, R 0, B 0

Pivot: Center

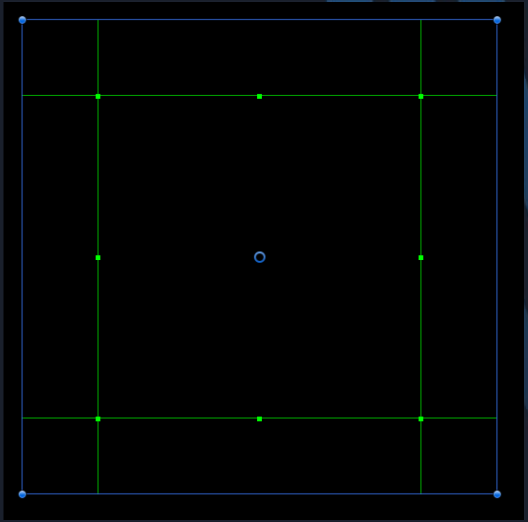
Custom Pivot: X 0.5, Y 0.5

Understanding sprite borders

- Sprite borders are used to help scale sprites, typically for UI elements.

- Borders split sprites into 9 sections
  - One centre region
  - 4 edge regions
  - 4 corner regions

- When the sprite is scaled in the scene:
  - The centre is scaled normally
  - The edges are only scaled on one axis – X for the top and bottom, and Y for the left and right
  - The corners aren’t scaled



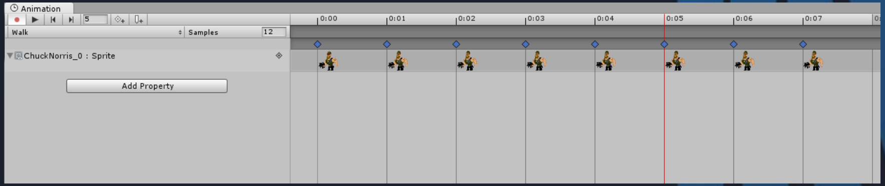
Animating Sprites

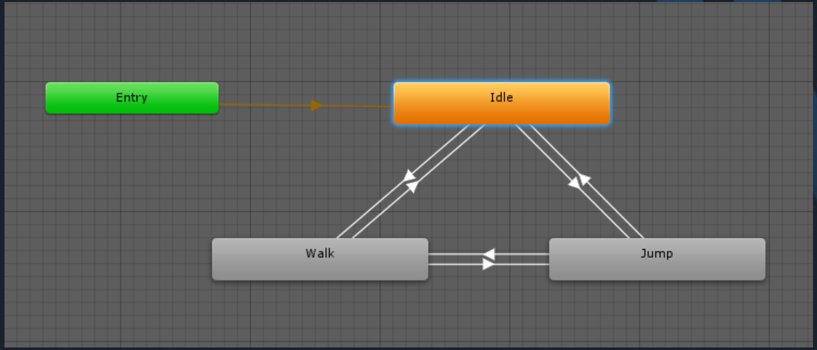
- You can use sprites to build animation clips that can be controlled by mechanim

- Animation Clips are built using the Animation window to build sequences of frames

- These clips can then be brought into the Animator editor and used in the mechanim state machine

- We'll go into much more detail on the steps for this in the tutorial

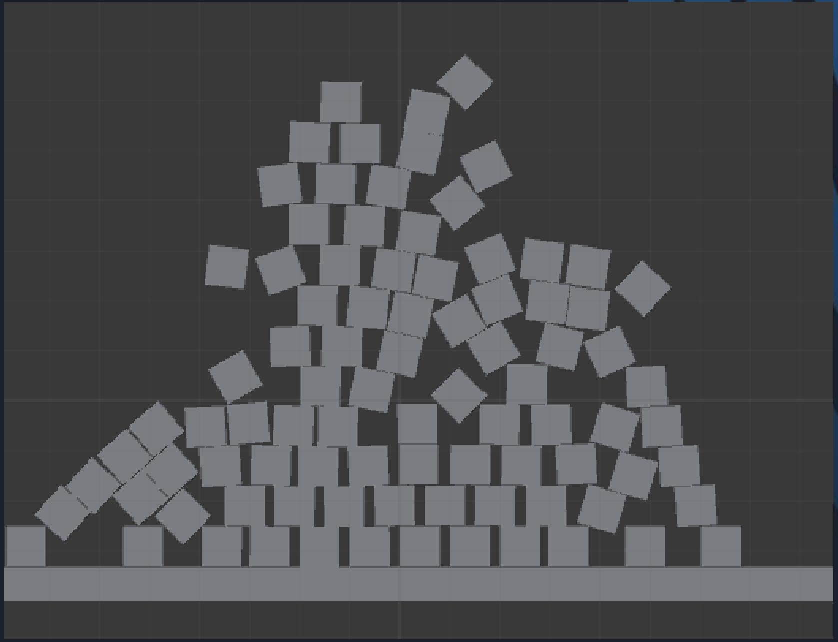




2D Physics

- Along with sprites, Unity also has support for 2D physics through Box2D.

- If 2D physics is all that is needed for a project it has several advantages over 3D
  - Don't have to use constraints to limit to 2D motion
  - Faster to compute



Differences Between 2D and 3D Physics

- The 3D and 2D physics systems do not interact in Unity in any way.
- The APIs for using 2D physics are very similar to their 3D counterparts
- In general there are 3 main changes required to convert from the 3D to the 2D physics systems
  - Replace all 3D colliders with their 2D versions
    - Box Collider -> Box Collider 2D
    - Sphere Collider -> Circle Collider
    - Mesh Collider -> Polygon Collider
  - Replace RigidBody with RigidBody2D
  - Use OnCollisionEnter2D, OnCollisionStay2D, and OnCollisionExit2D instead of their non-2D versions.

Summary

- Sprites let you group and animate 2D textures
- Unity provides a sprite editor to let you modify how a texture is split into sprite frames
- Individual frames can be:
  - Used alone as a single sprite
  - Combined into groups of separately moving components
  - Used as frames in an animation
  - Sprite animations can be controlled through mechanim, like 3D animations
- Unity also provides 2D physics simulation through Box2D
  - Integration is very similar to 3D physics

References

- Unity Technologies. 2016. *Unity - Manual: Sprite Editor*. [ONLINE] Available at:http://docs.unity3d.com/Manual/SpriteEditor. [Accessed 11 February 2016].
- Unity Technologies. 2016. *Unity - Manual: Sprite Packer*. [ONLINE] Available at:http://docs.unity3d.com/Manual/SpritePacker. [Accessed 11 February 2016].
- Unity Technologies. 2016. *Unity - Manual: 2D Physics Reference*. [ONLINE] Available at: http://docs.unity3d.com/Manual/Physics2DReference. [Accessed 11 February 2016].
- Unity Technologies. 2016. *Unity - 2D Physics Overview*. [ONLINE] Available at:https://unity3d.com/learn/tutorials/modules/beginner/2d/physics2d. [Accessed 11 February 2016].