

## Средства для реализации спрайтовой анимации в Unity3d

*Азаров Андрей Евгеньевич*

*Приамурский государственный университет имени Шолом-Алейхема*

*Студент*

*Кочитов Михаил Евгеньевич*

*Приамурский государственный университет имени Шолом-Алейхема*

*Студент*

### **Аннотация**

В данной статье будет рассмотрено, каким образом создаётся двухмерная анимация в Unity и какие для этого существуют специальные инструменты, как с ними работать. В результате, данная статья предоставляет инструкции для создания собственной анимации.

**Ключевые слова:** unity, анимация, управление.

## **Tools for implementing sprite animation in Unity3d**

*Azarov Andrey Evgenevich*

*Sholom-Aleichem Priamursky State University*

*Student*

*Kochitov Mikhail Evgenevich*

*Sholom-Aleichem Priamursky State University*

*Student*

### **Abstract**

This article will discuss how to create a two-dimensional animation in Unity and what special tools are there for it, how to work with them. As a result, this article provides instructions for creating your own animation.

**Keywords:** Photoshop, animation, management.

Ни одна двухмерная игра, созданная с помощью Unity3D не может обойтись без анимации. Анимация в двухмерных играх является набором множество заранее отрисованных изображений которые очень быстро сменяют друг друга, тем самым оживляя любую игру.

Данное исследование не уникально, в интернете существует множество статей с темой исследования, о том, как нужно работать с анимацией в Unity [1], однако Unity обновляется, появляются новые возможности и меняются способы работы со старыми, данная статья написана лишь для создания более полной картины о возможностях Unity и работе с анимацией, в версии программы за январь 2018 года 5.6.3p2. Также

многие участки кода могли устареть и не работать, если использовать версию Unity всего годичной давности. Самую актуальную информацию о скриптинге анимаций можно найти на официальном сайте документации Unity [2].

Для создания и управления анимацией в Unity3d существует несколько специальных инструментов, которые находятся в окнах animation и animator, также существует специальный объект который называется animation controller.

Для создания анимации необходим набор рисунков с анимацией персонажа, но не каждый человек сможет нарисовать набор изображений для анимации персонажа, поэтому в данной статье используется бесплатный набор спрайтов «The knight», рисунок 1. Таких бесплатных наборов существует большое количество в интернете [3-5].



Рисунок 1 – Набор спрайтов “The knight”

После скачивания данного набора и импортирования его в наш проект, наши спрайты будут выглядеть как на рисунке 2. На данном рисунке выделены все спрайты для будущей анимации атаки.

Чтобы создать анимацию необходимо выделить спрайты как на рисунке 2 и перенести с помощью левой кнопкой мыши в окно игры, далее unity предложит сохранить анимацию в формате animation controller и далее этот файл можно использовать в окнах animator и animation. Далее разберем основные инструменты для управления анимации в программе.



Рисунок 2 – Спрайты анимации атаки

Окно Animator – данное окно отображает схему всех доступных анимаций, также имеется возможность настраивать самостоятельно связи между этими анимациями и условия перехода. На рисунке 3 изображена такая схема. На данном рисунке, анимация по умолчанию выделена оранжевым цветом, эта анимация воспроизводится персонажем во время отсутствия движения или любого другого действия и также здесь присутствует анимация бега, данные анимации связаны между собой и имеют определенные условия перехода. В окне Animator, нужно объявить переменную типа float - Speed, которая отвечает за скорость персонажа и если скорость персонажа больше 0.1 то персонаж переходит в анимацию бега (рисунок 5), если скорость персонажа менее 0.1, то персонаж переходит в анимацию ожидания (рисунок 4). Также на рисунках 4 и 5 изображен вид персонажа во время анимации, ниже окна Animator.

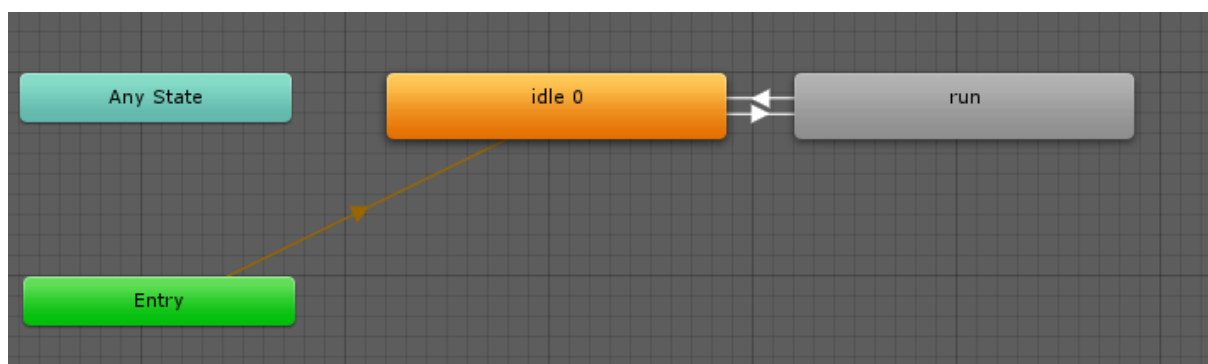


Рисунок 3 – Вид окна Animator со связями

Чтобы установить связи между анимациями необходимо нажать правой кнопкой мыши на прямоугольник анимации и выбрать пункт Make

Transition (создать переход) и нажать левой кнопкой мыши на прямоугольники анимации к которой вы хотите присоединить связь.

Далее, чтобы анимация изменялась во время игры, необходимо параметр speed внести в игровой скрипт отвечающий за управления персонажем.

Ниже приведены основные необходимые строки скрипта для управления и переключения анимации персонажа.

```
private Animator anim - Создаем переменную типа Animator.
```

```
anim = GetComponent<Animator>() - устанавливаем, что переменная anim это компонент animator который уже находится на персонаже.
```

```
float move = Input.GetAxis("Horizontal") - данная переменная принимает значение нажатия клавиши, если нажата клавиша управления, то устанавливается 1, если клавиша движения в противоположную сторону, то -1.
```

```
anim.SetFloat("Speed", Mathf.Abs(move)) - данная строка устанавливает, что если персонаж движется, то переменная скорости равна переменной движения по модулю. А так как выше было описано, если скорость больше 0.1, то персонаж приходит в движения и его анимация переключается на анимацию бега.
```



Рисунок 4 – Вид окна Animator и вид персонажа

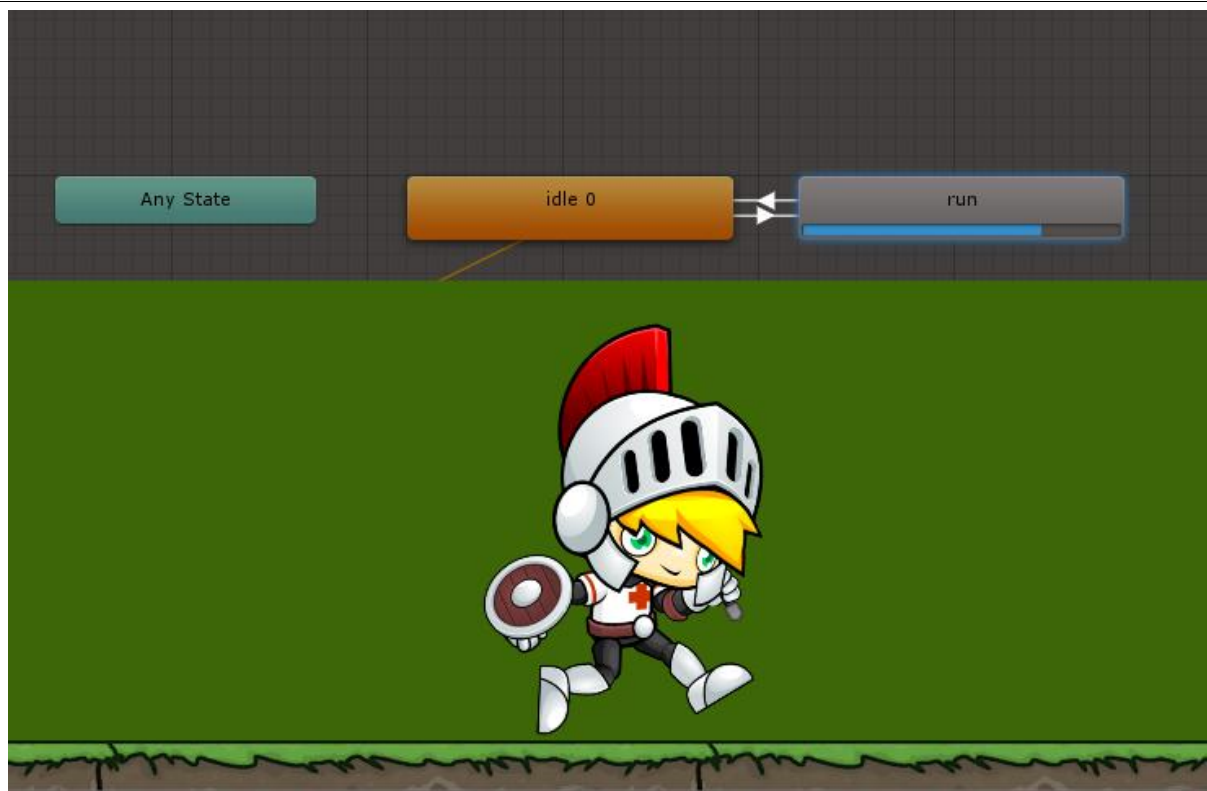


Рисунок 5 – Вид окна Animator и вид персонажа.

Окно Animation (рисунок 6) – это второй необходимый инструмент для создания и управления анимациями в Unity3d. Данное окно содержит инструменты для управления скоростью анимации, её плавностью и установлением количества кадров в секунду.

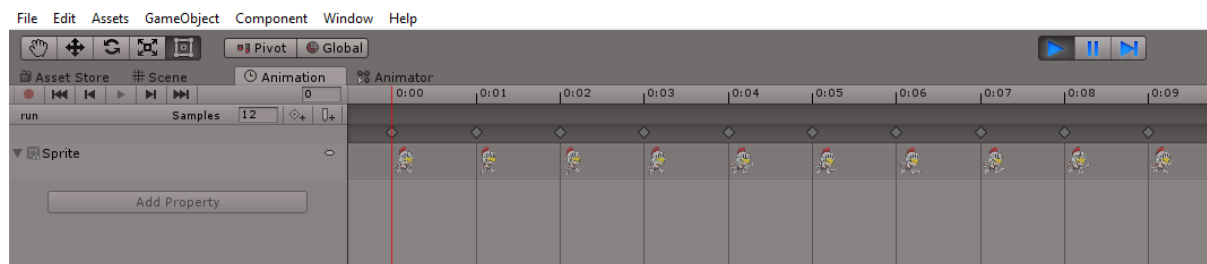


Рисунок 6 – Вид окна Animation

Анимация в двухмерных играх это обязательная часть любого проекта и реализовывать анимацию можно различными способами. Анимирование в Unity 3d с помощью функций animator, animation и средства animation controller является одним из основных средств анимации в unity3d. Анимировать персонажей можно также используя только скрипты, без использования вышеперечисленных инструментов – данный метод может использоваться для создания очень гибкой и нестандартной анимации. Целью данной статьи было рассмотреть и изучить инструменты для управления анимацией в unity, данный материал можно использовать для большинства учебных проектов по учебным дисциплинам связанным с unity3d и расширению знаний о возможностях игровой платформы.

**Библиографический список**

1. Основы создания 2D персонажа в Unity 3D 4.3. // Habrahabr URL: <https://habrahabr.ru/post/211472/> (дата обращения: 22.01.2018).
2. Скриптинг анимации (Legacy) // Uity3d documentation URL: <https://docs.unity3d.com/ru/current/Manual/AnimationScripting.html> (дата обращения: 22.01.2018).
3. Opengameart URL: <https://opengameart.org/> (дата обращения: 22.01.2018).
4. Free Assets For Everyone // gameart2d URL: <https://www.gameart2d.com/freebies.html> (дата обращения: 22.01.2018).
5. 16 Sites Featuring Free Game Graphics for Game Developers // v-play engine URL: [https://v-play.net/game-resources/16-sites-featuring-free-game-graphics#\\_Kenney.nl](https://v-play.net/game-resources/16-sites-featuring-free-game-graphics#_Kenney.nl) (дата обращения: 22.01.2018).