Motion Control

Pawns have motions - like climbing down, or rappeling down 20 feet.

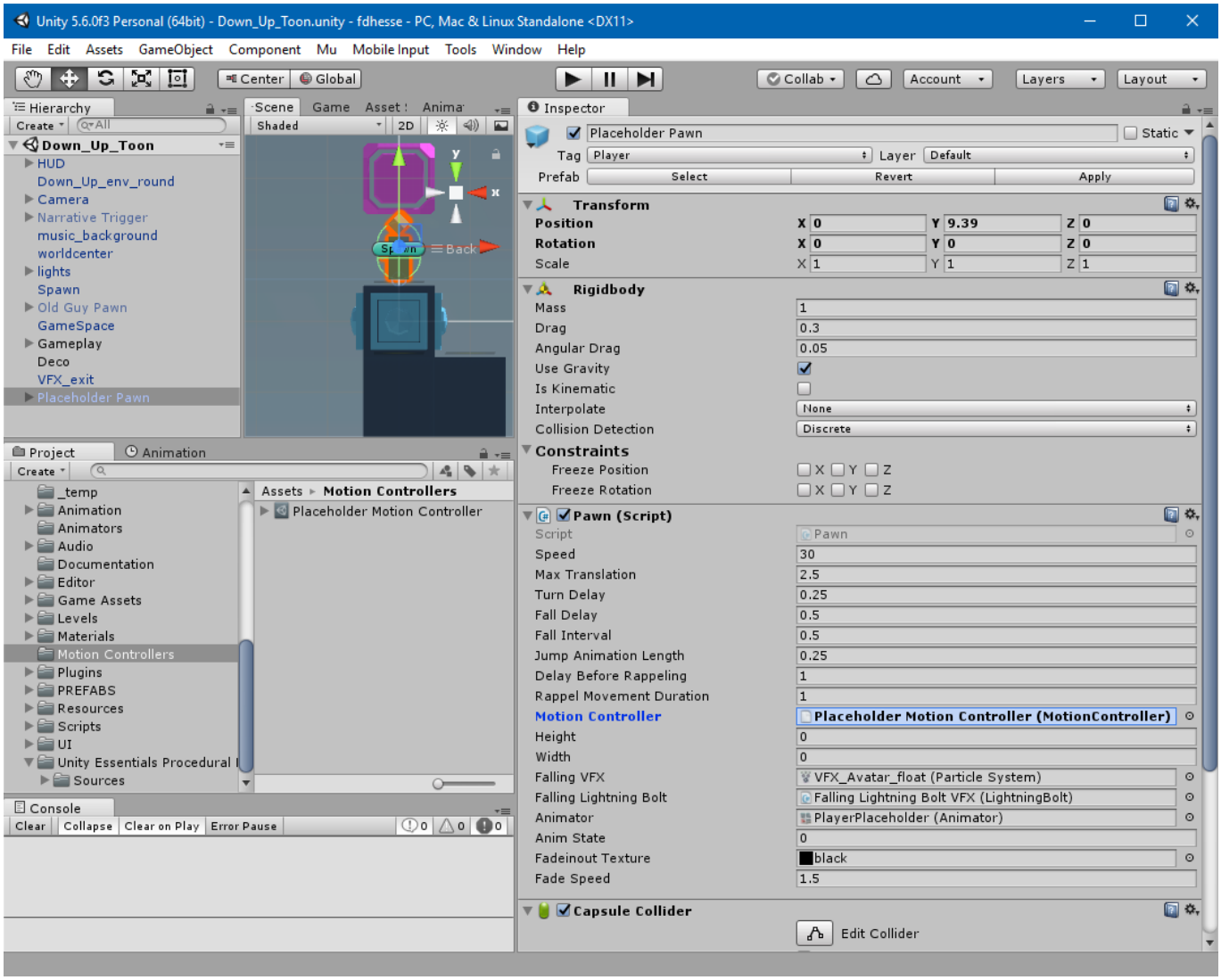
In some situations the motion corresponds to one animation, in other situations a motion might have multiple animations or no animation at all.

Motions aren't tweaked on the Pawn itself but on a special kind of custom asset - a 'Motion Controller'.

Motion controllers can be found in 'Assets/Motion Controllers'.

You can create motion controllers by right-clicking on the project panel and chosing 'Create>Motion Controller'

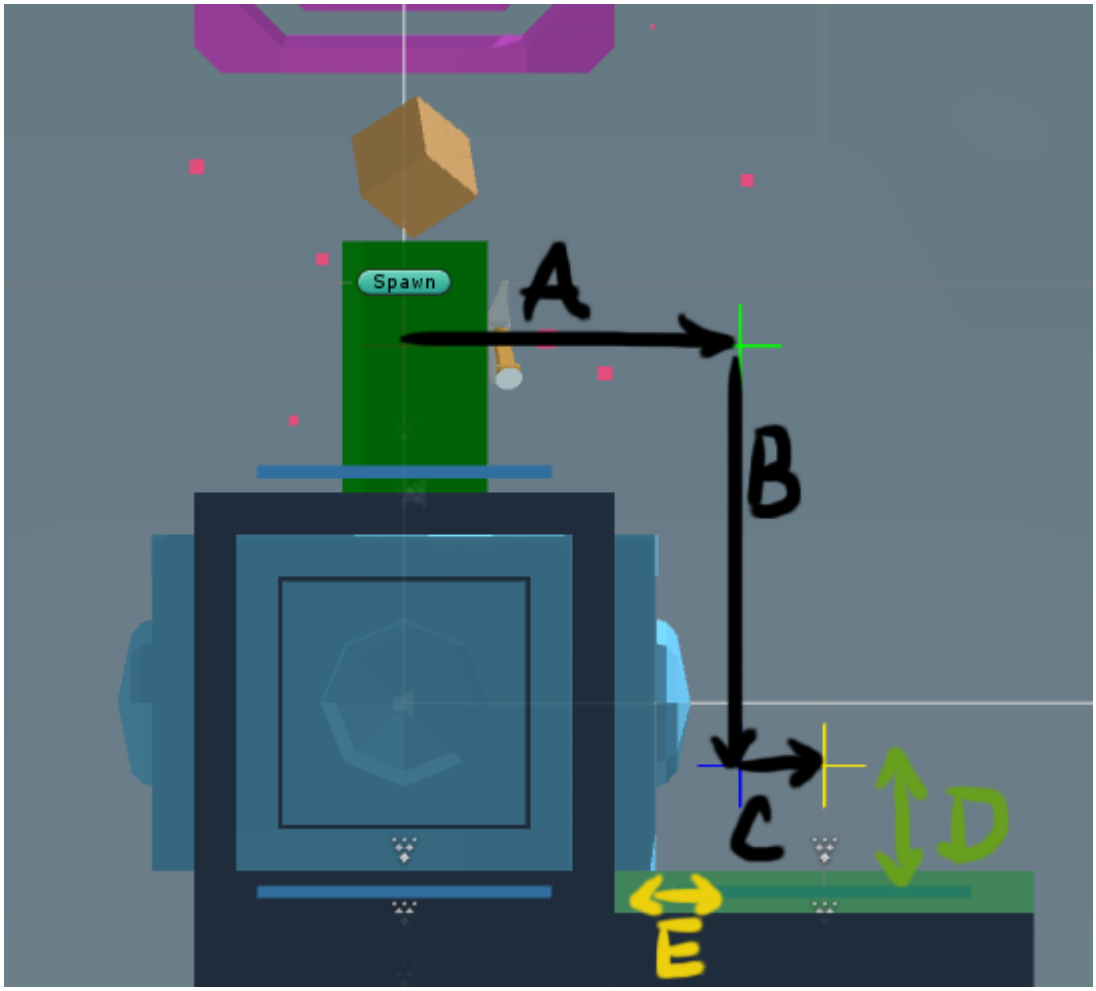
Each pawn has a field for the motion controller, to assign a motion controller to a pawn you need to drag the motion controller into that field.



Motions and Movements

A motion is comprised by one or a series of movements.

Climbing down for example is a sequence of 3 movements:




```
A, B and C: movements that comprise the climbing down motion
D: Height offset for points - calculated via pawn collider height
E: Distance from edge of blocks - half the width of the player's collider width
```

In this example movement speed is linear (the pawn moves at constant speed) for all movements.

The system is put together in such a way that it can be extended to allow for non-linear movements, to have multiple movements with various movement speeds, or to base movement speed off other factors (other than distance).

For example:

```
At the moment the rappel motion is linear. Therefore, for the animation to
be synced with the motion, the animation also needs to be linear.
This makes the fall look unrealistic, because when you rappel down there
is a sort of damping effect at the end, that you don't get with a linear
movement. To achieve a more realistic rappeling, the motion (and the
```

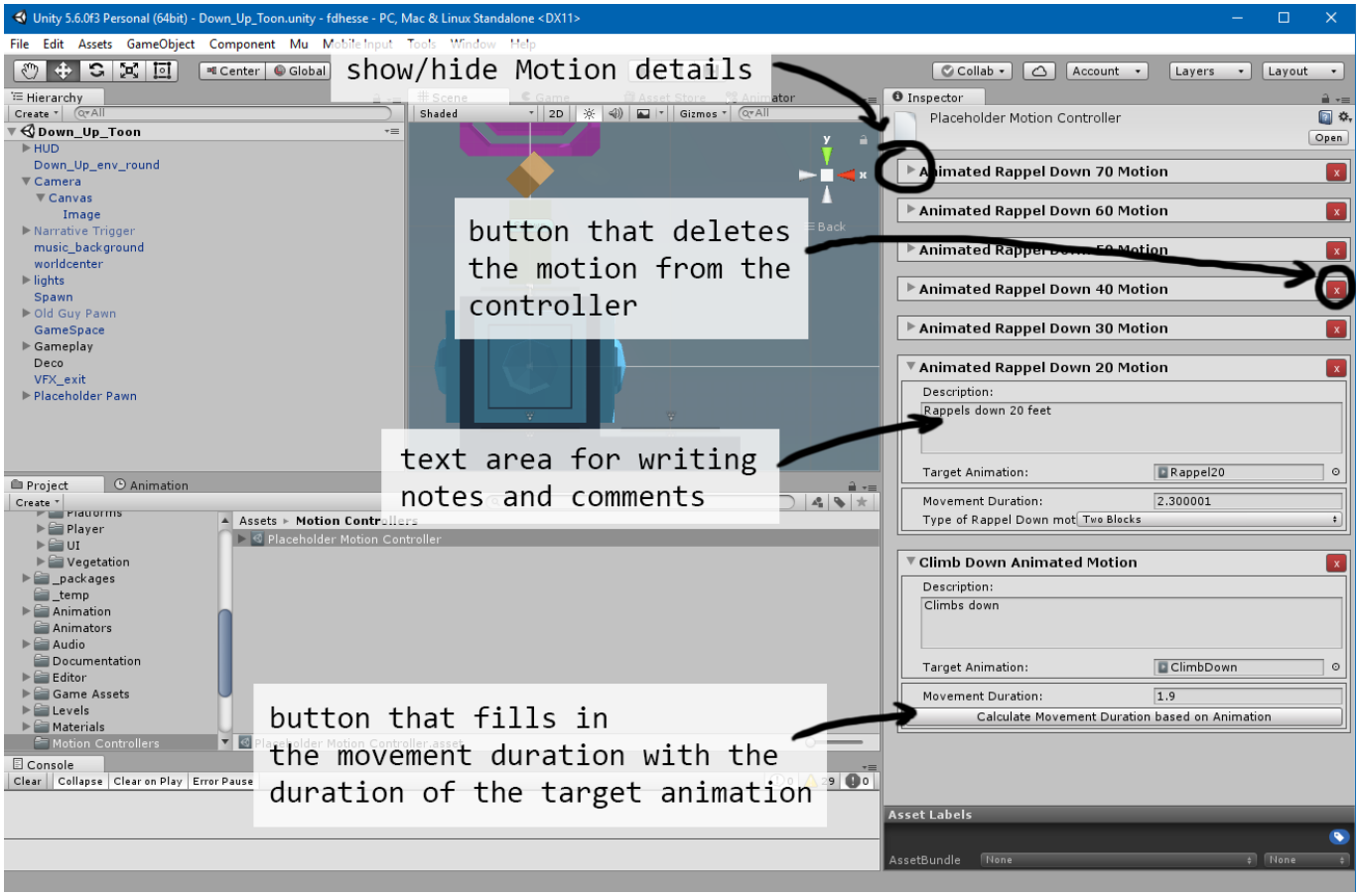
```
animation) must be changed to use non-linear movement.
```

easings.net - This website has preview of non-linear easing functions, I can implement these into the rappel motion or other motions.

Tweaking Motion Values

The motion controller ('Assets/Motion Controllers/Placeholder Motion Controller' for example) makes it very easy to tweak motion durations and assign animations to each motion.

Drag the right animation for your motion into your motion controller and you'll find a button that calculates the appropriate motion duration for your animation.



Rotations, translations, animations and motions

Rotations and translations that are significant in terms of gameplay shouldn't be applied on the main joint of the animation, but should instead be a part of the motion - done via code.

Example:

```
When the pawn climbs down to a certain tile, he turns until he faces
forward. That part of the movement is done via code. Because to climb down
the Pawn needs to turn his back to the target tile, we only know where the
target tile is via code (animation is agnostic to the gameworld).
```

Therefore all the movements that are purelly aesthetic - i.e. have no 'meaning' in terms of gameplay, may be applied to the main joint.

In order for these movements to be displayed correctly ingame, the animator needs to follow some pointers, like facing the pawn forward in his animations and 3d models.

Since some animations rotate the Pawn it is sometimes hard to know which way is 'forward'. On the 'Assets/Levels/3 Down Up/Down_Up_Toon_Pawn_Direction_Tester.scene' there is a Pawn that has a 'nose' that always points to the Pawn's forward vector. This scene can be used as reference to understand what a certain motion is really moving.