

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

Part 1 – Understand the relationship of flc files

What is and what is not

First, this tutorial is not about how to use Flicster. Moeniir has explained what Flicster can do in his thread very well. This is also not for explaining the format of flc. Again Moeniir has done an excellent job in detailing the technical details on flc format. This tutorial is about what you should understand and do to create a new animation. It does not cover changing palette information (i.e. changing the color of a unit). This tutorial is mainly aimed at those using basic drawing or photo for animation.

Part 1 – Understand the relationship of flc files

To create a smooth flowing animation for a new unit, you must first understand how the flc link with each other for Civ3 units. Basically there are quite a number of flc files for any single units but not all the flc are necessary for every unit. Below is a list of the flc files used:

- Default
- Run
- Fortify
- AttackA
- AttackB
- AttackC
- Defense
- Victory
- Fidget
- Death
- Drop
- Irrigate
- Road
- Build
- Plant
- Clear Forest
- Clear Jungle
- Settle
- Capture

As can be seen from the list above, not every flc are necessary for any given unit. Only those units that have worker ability will need Irrigate/Road/Build/Plant/Clear Jungle/Clear Forest and only those that are settlers need settle animation. Of course only paratroopers need Drop animation and unit which can be captured such as workers and settlers need Capture animations.

Default is the most important flc for any given unit. This is the animation you will see at all time for every unit except when they are fortified or when they are fidgeting (fidget last for a second or less such as rifleman sneezing and pikeman coughing). In general this is one animation that has the least movement (well, at least not drastic movements).

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

To make sure each animation run smoothly, you need to understand how each animation link to the other. Since Default is the most common flc, naturally almost all other animation is a continuation from Default. Below is a list of flc relationship:

Default --> Fidget --> Default
Default --> Run --> Default
Default --> Capture --> Default
Default --> Defend --> Death
Default --> Fortify --> AttackA --> AttackB --> AttackC
AttackA/AttackB/AttackC --> Death
AttackA/AttackB/AttackC --> Victory
Drop --> Default
Default --> Defend --> Death
Default --> Settle

Example of some frames from Leader Modern:



Last frame of **Default**-----→First and last frame of **Run**

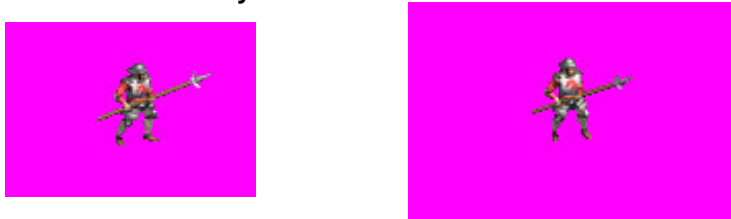
Examples of some frames from pikeman:



Last frame of **Default** -----→ First frame of **Fortify**



Last frame of **Fortify** ---→ First frame of **Attack**



Last frame of **Attack** ---→ First frame of **Death**

The relationship above means that the last frame of the animation file in front is the same as the first frame of the next one (or at least flow to the next one)

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

As you can notice, not all flc flows. All the worker's action is independent as each task uses different tool while the default worker carries a shovel. Some never flow back to the starting flc while some flc can animate smoothly in a loop by itself. Those that can animate smoothly by themselves are:

Default
Run
Attack
All worker actions

Those that never flow back to the flc before them and also played only once:

Death
Settle

Now, all the above is not that hard to understand right? Most of them are pretty obvious; especially the reason why some can loop by themselves while some played only once.

Alright, lets move to some special arrangement of flc files which can be used by all but at the moment only used by a few. These are AttackA, AttackB, AttackC & Defend. If you open up the ini file for a unit, you will see a number of lines that let you tell the program which flc file to use for a particular action. If you look under Attack you will see three lines, Attack1, Attack2 & Attack3. In general only 1 is used, namely Attack1. However, if you choose to use more than 1, then only the last one is looped while the earlier ones played only once in the sequence.

As far as I can see, Defend was not used for the units at the moment. I believe it is more or less the same as attack sequence.

Fortify flc is also another unique flc where you never see a unit fortify facing other direction except for SouthEast. It took me a while to figure this one out. I finally realize that the fortify flc is used to link the default flc with attack flc.

Now, what if you did not create an flc file for a particular action and a user modify the unit's ability so that it can suddenly take on that particular one you didn't make?? Worry not, Civ3 is quite forgiving in this matter. If there is no flc for a particular action, the action is still carried out. Only no animation is shown for that particular action.

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

Part 2 – The view

The view in Civ3 is what people called an isometric view. It is kinda like viewing an object from the top at a 45-degree angle. Even though there are 8 directions, the differences between each view are not exactly 45-degree per se. If you take say the eastern view of an animation and rotate it 45-degree to the right (or south) you will see that it is not facing SE as you desired. This is due to the 45-degree view angle of the object. Therefore, to achieve a 45-degree rotation you only need to rotate the object about 30-degree.

A lot of the views for an object are mirror image of each other as long as the object symmetrical. An example would be a car or truck, which looks the same from both sides. Of course, to make life easy, when you create an object you might want to consider making it symmetrical if possible. However, if it is not symmetrical, say if you put the machine gun on the tank turret on the right side for example, then you should not make the view a mirror image. Now, assuming that your object of animation is symmetrical, then the views that are mirror images are:

NE & NW

SE & SW

E & W

If you do the view correctly when creating the animation (without a 3d modeling program), you should be able to save your work by converting the different directions into the respective mirror images. In fact, if the view is done correctly, you can even rotate your eastward view by say 30-degree to the right to get your SE view.

Perhaps at this point you might ask, why not rotate the eastward view 30-degree to the left to get NE view? Well, if you view an object at 45-degree angle, you see a major part of the front but not the back. And when an object is facing SE you also see a major part of the front portion. But, if you view it when it is facing NE, you see a major part of the back portion instead.

The North and South view are also significantly different from other angle. For a vehicle, both the views will show the vehicle slightly shorter than the East or West view due to the 45-degree view. Of course the other major factor will be the South view show whole frontal view while the North view show whole back view. And if your object has different level of height, this two view become very significant, as the higher level will block a major portion. Take a tank as an example, the turret will block a major portion of the back or front in the South and North view.

Now lets look at a different object of animation, the human body. After all, most of the ancients to industrial age units are human in motion. You might think that human is also a symmetrical object as each has 2 hands and two legs. However, what make human different is which hand holds the weapon or tools. If the unit hold its weapon or tools in the right hand (as is commonly the case), then all mirror imaging are out. A mirror image will make a right-hander become a lefthander. And rotating a human's view is also not feasible, as things such as the shoulder's angle etc are very much different in each view.

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

Part 3- The motion for vehicles and ships

Finally, we come to the part on actually doing the animation rather than discussing about the boring stuff like frame relations and views. This part only cover vehicles and ships as I feel that the motion for human body deserve a part by itself.

Of course, the main part of animation is motion. Without motion there is no animation. To do a motion, you need to consider a few things to make them realistic. We will go through the motion via each different type of flc.

Default

This is the basic and most displayed flc. Now, you might say, why animated the default view at all? Why not just animate the motion as fidget? Well, if the default view is not animated, then the unit looks a little static when not moving. And most of the times they are in default view (those AI units. Now, before you put your object into motion, think about its size and weight. A tank is very heavy and if it shake at all its only a very small movement (notice the Modern Armor and Tank animation in default, very minimal movement). A Battleship is in the same category, you don't expect a monster like that to sway very much in the normal sea right? Now consider a different class of vehicles and ships. Take the Army Truck for example, you did expect it to vibrate as long as the engine is kicking (especially those WWII varieties). So, in the Army Truck animation, I show the truck shaking in default view. BUT, please remember that not the whole object is animated in every case. In the case of Army Truck, the back portion of the truck is shaking on its suspension while only the hood in the front portion is shaking. Think of a real car and you will get what I mean.

For the case of ship, take the Perry Class Frigate as an example, I expect the ship to sway in the waves. In this case, of course the whole ship sways rather than only certain part.

Ok, enough talk already, how do we make the animation? It's actually very simple for vehicles and ships. Let's take vehicle first. Again, using the Army Truck as example. First duplicate 2 copies of the first frame. We are going to leave the 1st and last frame as it is and work on the 2nd frame. Now, copy the back portion of the graphic from the top until the suspension area (back portion only). Next, paste the selection 1 pixel down. Now copy the top portion of the hood and paste the selection 1 pixel up. Simple, right?

So, now you have the original 1st frame, the changed 2nd frame and again the original 3rd frame. Assuming you want to make it a 5 frame animation (3 frame will make the movement very obvious. You did be surprise how major a 1 pixel movement is when the final graphic is so small), you will now insert 1 new frame between 1 & 2 and 1 new frame between 2 & 3. If you are using Animation Shop, it's a piece of cake. Just go to Animation, select Insert Transitional Frame and insert 1 frame. AS will insert a go between frame for you (got to love AS for this). If you are not using AS, I will suggest you do this. For the second frame, copy from somewhere below the canvas top (i.e. not from the very top, somewhere lower) to the suspension area and still paste the selection 1 pixel down. This way, you can show a motion of the truck's back portion moving down slowly in two frames. Same applies for the front portion. Next, use the new 2nd frame as the new 4th frame as well. Now you get a smooth animation, 1st --> 2nd --> 3rd --> 4th (2nd) --> 5th (1st). There, you have created your first simple animation.

Now lets take a look at ships. Ship sway in water. (I like to go for deep sea fishing so I know how it's like on a ship) The process of making ship sway is very much the same like the process of making the truck shake. However, when a ship sway, the ship moves in an arch and not simply side to side So, when you paste the selection you should expect to move the selection 1 pixel to the side and also 1 pixel down. And remember, if there are waves in the sea, the ship usually sway to one side and back. So don't make a ship sway like a yo-yo if you want it to look realistic

Run

This one is easy once you have completed default. For a motor vehicle, the only difference with Default

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

view is the motion of the wheels (or threads if it's a tank). Of course unless your Default view is motionless. To make the wheels in motion is very simple. Take a look at your Default frames. Some parts of the wheels are lighter than the rest of the wheels right? (Unless you make your wheels pitch black but that's not really realistic right?) Now, for each of the subsequent frame after the first one, just move the lighter pixels (1 or the most 2 per wheel) in the right direction. If your vehicle is moving forward, then move the lighter pixel clockwise. Usually move the lighter pixels about 4 times around the wheel (each says a 90-degree move from the one before). Now you got a vehicle that shakes and with a moving wheel! You have just made the first simple moving animation

As for ships, it's slightly more complicated. Ships create wakes at the back of them and also at the stern as they break the water in front of them. Wakes are represented by white color (as in the real wave) in Civ3 so it's a matter of putting some white color both at the back and in front. For a big ship like the Modern Frigate, the ship is pretty stable even though it might still sway as it moves. So the ship remains at the same level but the wake should rise and fall (read: cover more or less of the ship body). As long as you remember that the first frame should show only a little wave and the last frame should also show the wave dies down to about the level of frame 2. Generally due to the arch swaying motion of a ship, the animation frames for default is usually more than 10 frames. This will be perfect for showing the raise and fall of the waves.

Fortify

Since most of the Civ3 units are military units (other than workers, scouts and settlers), fortify is pretty much a quite important animation. However, for vehicles and ships these animations are less important as compared to human units. In most of the cases, you can actually do away with a fortify animation for vehicles and ships. In fact, most of the in game vehicle and ship units do not have a fortify animation (with the exception of bombardment units like artilleries). This is mainly due to the fact that the attack animations of a tank flow smoothly from the default flc. For bombardment units, the fortify flc is more like a pre-attack flc. For canon etc the flc are mainly showing the unit pulling back a little or stabling and ready for attack so not much need to be done for the flc. To illustrate this I will use the Scud Launcher as an example. In order to launch the scud missile, it has to be raised on the launcher. So for the purpose of fortification flc, I just animate the unit to raise the missile. In order to animate part of an object, this is what you should do if you are not using a 3d modeling software.

Save one single frame (i.e. the first frame) as a pcx file. Load the pcx file into PaintShopPro or any program that work with pcx format. Erase all the parts that you do not plan to animate. For the example of the scud launcher, I erase the whole vehicle and leave only the missile in the picture. By erase I mean replace the background with transparent and not the standard pinkish color used by Civ3 flc. Save the file using a different name. Now load the original file again but this time erases the part you want to animate. You will need to color those areas beneath the removed part, in my case the body of the launcher that was originally covered by the missile. Again save the file using a different name. Now go back to Animation Shop (or the Storyboard style pcx file) and load the vehicle body without the animated part into the 1st frame (or paste if using storyboard pcx file). Now you have 1 frame of the vehicle body as your base. If you are doing a 5 frames animation, just duplicate the 1st frame 4 times so now you have 5 frames of identical vehicle body. Now, load the missile only frame pcx file back into PaintShopPro and rotate the image by, say 5-degree to the left. Copy this new image and paste it over the first frame's vehicle body. Now rotate the missile another 5-degree and copy then paste it over the 2nd frame's vehicle body. By repeating this process, you will soon get a 5 frames animation that shows the missile raising at a steady speed. Its not necessary always 5-degree per frame as long as the final frame shows the missile at the angle you want the rotation or the raising of the missile can varies for each of the frame as long as the differences per frame is not too much. When you do it this way, do remember that not the whole missile frame is rotated at 5-degree per frame. In my case, the scud missile is supported by some form of a metal platform that rises together with the missile. So for each frame I will go back to the now paste and merged vehicle and raised missile to redrawn the back portion of the metal frame around the tail end of the missile so that the metal frame actually move about 1 pixel per frame. This creates a different layer of animation within an animation and added to the realism of your animation greatly. Remember how the different layer of background scrolls at a different speed when you play those arcade side scrolling game?

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

That is what I am trying to explain here. Of course, all these tiny details will depend on the size of your object of animation. If it is too small to be seen, you really shouldn't bother (in fact, the movement of the frame can barely be seen on my 19" monitor and I doubt that it will be noticed on anything less than 17" but I prefer details to none)

Attack

This is the most interesting animation so far for all the units I created except for the Army Truck which has no attack flc. Remember the discussion in part 1, attack flc flows from the fortify flc. Other than that, most unit recoil from the firing of their guns/cannons (other than air units and radar artillery, which fire rockets). For the case of the scud launcher, the scud missile is launched and therefore there is no recoil motion. Besides, the animation switch to a flying missile and there is no looping as the unit is destroyed after the bombardment like a cruise missile unit so it is not really a good example to explain the Attack flc. I will use the Modern Frigate as an example to illustrate the Attack flc instead.

A ship really recoils when firing its gun so it is a good base for some interesting animation. For the Modern Frigate, it fires a missile from the front so the motion is considered much simpler as compare to those wooden ships or the battleship which fire their guns broadside. To start an attack animation, use the last frame of fortify animation as your first frame. Since there is no fortify flc for a ship, the last frame of default flc will be used instead. Now duplicate the frame and leave the 1st frame unchanged. For the second frame, missile (or gun/canon for other ship type) fire will start to emerge from the missile silo (or gun turret). The flare from the fire is bright and the portion of the ship on the same side as the fire should be illuminated. To do this, you can use a much lighter shade of color to paint over the original color for the whole area that is supposed to be illuminated. For the Modern Frigate, the original color is medium gray so the front portion is now painted with a coat of white color to represent the flare. To effectively do this, you must have a good understanding of where corners are in the area you wish to light up as corners often block the lights. The fire resulted from the shooting/launching of missile will last about 3 frames. However, the lighting effect should last 1 additional frame to represent the delay in the disseminate of light after the missile/shell leave the silo/turret. Once the light is back to the normal level, the ship should now feel the recoil power of the blast/launch. Actually the swaying motion that we created in the default flc can once be used here. However, the different between the swaying in default and attack is that the former is a gentle sway while the later is a result of recoil motion. Thus the later sway should resulted in ripple in the water around the ship. Ripples can be easily presented by white semi curving lines slowly extending 1-2 pixels away from the ship. By the time the ship stop swaying so should the ripples be gone and with the ship once again stationery and no ripple its back to frame 1. There, you now have a ship that burst with a blast that illuminate part of it, sway in recoil causing ripples and back in stationary position which is really a perfect loop. The attack flc must always loop as battle usually last more than 1 round (unless all the unit has only 1hp)

Death

Like the AI in Civ3 likes to say, lets give this deal a decent burial and get on with business. Death flc is the last flc a unit ever need and mind you its actually one of the most beautifully animated one. If there is beauty in dying this is it!

You might think that dying is pretty easy for vehicles and ships. Unfortunately it is not. For vehicles and ships there are almost always an explosion to accompany its fiery end. This is the flc where you learn how to animate an explosion. Lets go back to the Army Truck flc and again use it as an example to illustrate this. An explosion sequence generally last about 10 frames beginning from the burst until all there is left is smoke in the air. A death explosion is different from the burst of a gun. Even though you would expect both of the burst to illuminate the surrounding area, the death explosion generate a whole lot of thick smoke and thus cover up the surrounding area (which is good news of course as a lump of thick smoke is much easier to render compare to everything else in an animation). In the process of explosion, part of the vehicle will of course be burned and in general the vehicle is expected to be in a mess with broken axles and tires etc. I will use the Army Truck flc once more to explain how this is done. The truck has a canvas top, which I want it to be burned in the explosion. As the explosion comes with

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

lots of smoke, this is easily done. All you need to do is to erase the canvas top when the smoke slowly starts to clear. Of course, with the canvas top burned, the metal frames that support the canvas is now exposed (Details and more details. It is details like these that make an animation convincing). In the case of Army Truck, the explosion sequence lasted 10 frames. The smoke starts to clear slowly from frame 8 to 11 (frame 1 being the same as the default to ensure the flc flows). So the metal frames should start to appear bit by bit, as they are uncovered from the smoke. To achieve this effect, I recommend that the sequence be done frame by frame using the duplication technique. Frame 1 should be the standard default frame and frame 2 should be the beginning of the explosion sequence. When frame 2 is completed, duplicate it to become frame 3 and expand the explosion and smoke a little. Repeat this process until the explosion is done and smoke cleared. This is the same technique used even by the Disney animators. Do not draw your frames 1 by 1. Copy the next frame from the previous one and work from there will ensure that only the part that needs to be animated is changed and the rest of the things in the frame remain constant.

Alright, so we are more or less done with explosion. But explosion is only part of the death sequence. There is still the destruction of the truck to be taken care of! As mentioned in the last paragraph, we want broken axles and flat tires. In this case, we want the breakdown to happen towards the later part of the explosion when the metal frames starts to appear. On the second frame where the metal frames starts to appear, we copy the back portion of the truck from the top until the suspension area and paste them 1 pixel down just like in the default animation. In the next frame, we copy the whole truck from the top until 1 pixel from the bottom at the place where the tires touch the ground. Again we paste it 1 pixel down. Repeat this process for the next frame. Now back to the process of copying only the back portion to suspension area and past 1 pixel. Again repeat the same process for one more frame. In the process of doing this, I have also darkened the color of the truck inside the back portion by 1 shade darker to represent the charred chassis from the explosion. After this process is completed, you now have a death animation that shows explosion, burned canvas top, broken axles which resulted in the back portion to sink, a flat tire which resulted in the whole truck to sink and the continuous motion of the back portion sinking! Now, that isn't so hard right?

Now that you have an idea of how the death sequence for a vehicle is done. Lets look at ships. Since the only animation I have done so far for ship is the Modern Frigate, I will use that as an illustration. Explosion sequence for ships is very much the same as vehicles so I am not going to repeat too much on that. One thing worth mention about explosion sequence is, if your object has some external extruding parts (such as the machine gun on a tank turret, tall mast on a ship etc), you may want to break the part off from the main object when you do the explosion sequence (e.g. the machine gun flown off, the tall mast break in half etc) to make the animation even more realistic (beauty in dying, remember?). A ship does not have a breaking axle or a flat tire, but it sinks! So the animation for ship after the explosion is sinking of course. Since the explosion is shown on board and not below deck (I assume everyone understand these simple ship terms), we can't have the ship slowly sink to the bottom. And since Modern Frigate is a rather large ship, we also can't have it break into half like an old wooden ship. The best way to show this is to make the ship topple to one side and sink as it goes. The toppling motion is very much like the swaying motion; it's in an arch, swaying to one side. The only difference is the ship never sways back to its original position. The sinking process is also very much like the flat tires sequence, copy the whole ship and paste 1 pixel down. Once you learn how 1 animation is done, it can easily be applied to another object's animation. One thing to remember for animating a ship's sinking sequence is, the water tends to rush up and cover some of the top surface before that surface sinks. (think of pushing a bucket down a pond or your bathtub). Once the ship sinks out of sight, what left are only ripples in the water, which should then slowly expand and disappear.

We have now completed the tutorial on how to create a vehicle and a ship unit. You should be able to create some not too complicated vehicle and ship animation for the moment. In the next part we shall move on to the toughest part of animation, the human body.

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

Part 4- The motion for Human bodies

Ok, now that you know how an animation sequence is done by using duplicate, copy and paste, we should be able to move on to the more complex animation of human body. Before we even start to look at the individual animation sequence such as default, let's take a quick look at a human.

While our body functions together as a whole, individual parts of our body can also function on its own independent of the other. What restrict the movement of a body part are our joints. The joints act like axis for a machine part that allow us to straighten and bend our shoulder, arms, wrist, thighs, legs, ankles, neck, waist and head. Of course there are more to human body than what is mentioned above. However, since this is not a tutorial in biology, and body parts other than those above is too small to be seen in an animation sequence, the coverage above should be good enough for you to know why an animation work in a certain way.

As our body parts like hands and legs move very close to our body, light and shadow often play a very important part when you draw a human body. An elbow that extends to the back can be just black in color instead of the color of the clothes the unit is wearing due to the shadow. So you should have a fairly good idea where your light source in a picture/frame comes from in order to show the correct light and shadow. Due to the small size of human flc (human flc are the smallest of all flc for Civ3 as human is one of the smallest unit), the light and shadow play is very important as some of the animation are merely the shifting of light and dark patches inside a frame

Due to the complexity of human body movement, cut and paste and duplicate from frame to frame type of animation technique is almost out of the question. BUT, as most of the Civ3 original units are human, almost all possible movement is already in the game's current units! This means all you need to do is mix and match those movements with a different set of clothes and outfit. Phew!! Thanks to the animators of the original units from FIRAXIS, creating a new human unit is a lot easier than starting from scratch. In fact, if you have a good perception of how human body work in motion, you can also combine each of the single arm and leg to form a whole new movement that is not in any of the current units.

Alright, let's go through each of the animation sequence with the human body just like we do for vehicles and ships starting once more from Default.

Default

Default for human body is the same as default for vehicles and ships. It is still the most displayed animation. However, most of the human unit don't stay stationary (just like the AI units, human units hardly stay stationary, perhaps there are ants in their pants?) and all default flc has some form of movement. Take any unit flc from the current human units, most of the default flc has one simple movement, moving their upper body from side to side. So if you are going to create a human unit, you should consider what sort of movement you want your unit to have in default. For my terrorist flc, as it was my very first flc, I decide to do something unconventional. I make my terrorist unit puffing nervously away on a cigarette as the default movement. Because it was my first unit animation and I was still learning how to use Animation Shop back then, I actually redraw the arms that hold the cigarette frame by frame. Of course that is all history now and you do not have to go through the trouble that I did now.

To create a new unit, you should first decide which of the current units' outfit suits your new unit and which of the current units' action come closest to your new unit's action. When I did my crossbowman, I decided that the crossbowman appear about the same era as the pikeman. Therefore I use the pikeman as a base for the body so I need to do minimum recoloring for my new unit. Next, since I am doing a crossbowman, the closes attack sequence will be either a rifleman or an infantry. Eventually I selected the infantry for the loading sequence.

For the default, since I use the pikeman flc, I just replace the pike with a crossbow and I have the default flc for my crossbowman. If you are doing another middle age unit and plan to use the pikeman as a base, you can still do like what I do, just replace the pike with the weapon of your choice.

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

Run

The Run sequence is very much the same as the Default. Your main task will be to replace the item that is being held in the hand. Take from the example of my Terrorist unit, I combine the body of the worker and the running motion of the modern leader. On top of that, I added a little briefcase for the terrorist. And to make the motion come alive, I draw the briefcase frame by frame to make it swing as the Terrorist is running. To make an animation look real, bear in mind that we, the human, swing our arms as we run. Therefore, the animation sequence should show both the movements of legs and also arms. In fact, as a result of the movement of the arms, the weapon or item that is being held in the hand should also swing in accordance with the motion. (Just like the briefcase the terrorist is holding or the crossbow held by the crossbowman). Last thing to remember for arm movements, when an arm is swinging, the shoulder will also move. Therefore, for any arm or leg movement, don't just move the lower half of your arm or leg, move those upper halves as well! And move the lower and upper half at a different speed too. (Stretch your arms and you will see what I mean. Movement of your shoulder and elbow and even wrist is not always at the same speed. It depends on what is the motion)

Fortify

The fortify sequence for a unit is really pretty standard. The way an archer pulls his bow in full to fortify is the same as those of a bowman and a longbowman. In fact, if you were to create a new unit called Indian bowman (as in Native American Indian. We shouldn't make the same mistake as Columbus), the sequence of pulling a bow in full is still the same. So, once you have a suitable body for the new bowman, you just need to copy and paste the arms and the bow from the archer's sequence. In order to do this, go back to the fortify section when we discuss the fortification of the scud launcher. The method is the same! Just erase everything other than the arms and the bow from the archer, remove the arms and whatever weapon the new unit is holding in his hand, merge the two frames by copy and paste and you get the new Indian bowman. Of course, there is one big difference between the fortification of a vehicle and that of a human unit, the legs move in fortification sequence. If you watch or study those martial art moves, you will notice how the legs are push down and bend in some of the stance. The basic idea is to stabilize your body by lowering the center of gravity. Since the purpose of fortify is to make your enemy harder to dislodge you, the same idea of lowering and bending of legs applies. Even if you think of a modern soldier with an automatic rifle, legs are still lower and bend when firing the rifle. A soldier will still want to stabilize himself before firing a gun to increase the accuracy and to reduce to effect of recoil.

Attack

Attack is an interesting one for human unit. It is still possible to use the existing units' arm movements for a lot of possible attack sequence. For a double-handed sword you can use that of the samurai, for pole arms you can easily use the swinging motion of the samurai but replace the katana with the pole arms (say a halberd). For a short sword, bastard sword or rapier, the motion of Immortals can be used. For my case of crossbowman, I use the firing sequence of an infantry until the part the infantry lower the rifle to eject the spend bullet. From there I manually redraw the right arm, showing it pulling back, insert a bolt, push the bolt forward and pull back the string. Then I reverse the lowering sequence to show the crossbow being rise back to firing position (of course in the new raising sequence I added a bolt to the crossbow as the original lowering sequence has an empty crossbow. Remember, details!) So the attack sequence of my crossbowman shows firing position with loaded crossbow, fire the bolt, lower the empty crossbow, load the crossbow, raise the loaded crossbow back to firing position which is a perfect loop by itself. Just remember, as the fortify flc flows to the attack flc, you should choose the same arm movement that flows for both fortify and attack. Like wise for the leg movement. Crossbowman attack sequence is not that good an example for attack sequence as someone who fire a gun or a crossbow don't move their legs when doing so. If you are swinging a melee weapon, you tend to pull back or raise the weapon before you strike. When a melee weapon is pull back or raise, the legs tend to also pull back and step forward again when doing the strike. To see the effect of this, take a look at the flc for unit such as pikeman or warrior and you will understand what I am trying to say here.

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

Death

As in the case for vehicles and ships, death is once more the most beautifully rendered animation sequence. (Beauty in dying, remember?) There is no explosion involved when a human dies. However, the human body twists and turns before a unit finally drops down and dies in peace. This of course makes it much harder as compared to the explosion sequence. In order to do a realistic death flc for a human unit, you would want to consider a few things. A unit twists and turns not because you just want it to look more complex or beautiful. It's because of the impact of the final blow! Making a good animation is very much like directing a film, a lot of things have to be considered especially for logic and realism.

So, before you even start to do the animation of a death flc, you must consider how you want the unit to die. There is a lot of dying sequences for the current units in the game and most of them are very unique. A unit can twist then drop dead, drop face down, drop side way, drop flat on its back even drop on its knees then die I would suggest you just choose one of the sequences available and copy and paste the movements for your new units. Of course you can create a new sequence, but I feel that what is available now is already rich enough in varieties.

Once a sequence is selected, it is a matter of copy and paste and recoloring of the clothing to finish the job

Fidget

Fidget for a human unit is pretty unique. It makes the animation for unit richer and not so monotonous. While fidget for a vehicle or ship unit is quite simple, it's a totally different story for a human unit. How a unit is going to fidget will really depend on the ingenuity of the creator. For the crossbowman, I took the easy way out by using the fidget flc for pikeman by simply replacing the weapon in hand with a crossbow. As for my terrorist, the default flc is something pretty unique by itself and as a result I did not create any flc for fidget. The more prominent fidget flc in the game are those like the pikeman coughing, rifleman sneezing, swordsman yawning and stretching and infantry breaching the rifle. Every single one of them is very unique and unless you use most of the flc from a current unit (with changed clothing, weapon etc), you will not be able to use what is currently available. So, think of something unique and show the world what you can do on this one!

Victory

Human is not machine. Therefore, human tends to get excited when they get promoted. This is when the victory flc is played. Most of the victory flc will show the unit shouting or laughing in excitement. Therefore, a victory flc must always reflect the excitement and happiness. An exciting animation can only be presented with the combination of both arms and legs movement. As such, chances of you copying a movement sequence from the current units' animation are pretty slim. For my crossbowman, I show it lowering the crossbow and raising its right fist in the air in excitement and then once again raise the crossbow back to firing position. Do remember that the sequence must flow from attack but back to default. (Victory only results from battle). Victory is also very much like fidget, it depends again on your ingenuity and pretty hard to be copied (though not impossible).

Other animation flc for human units

I do not plan to discuss the animation for other units such as settlers and workers as they are only used for those specific units and nothing much can be changed for those special animation sequence (irrigating IS irrigating.)

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

Mounted units

Civ3 have quite a number of mounted units. When you animate a mounted unit, instead of worrying about the movement of the human legs, you now have to worry about the movement of the horse legs. And 4 of them instead of 2! But seriously, the horse legs are easier to animate than the human ones. The horse will not try to stabilize itself when you fortify or attack so one less worry. But as the horse is independent of the human rider, it tends to have its own movement such as swaying of the head or tail. Also, the horse will have its own little fidget too

Air units

Due to the limited use of air unit at the moment, their animation is also very simple. In fact, the default flc mainly consist of 1 frame 8 direction with no animation whatsoever. And the run flc is mainly with additional air appearing at the plane's tail. The attack sequence is also simple with added machine gun fire on top of the run flc. The bombing flc is just run with added dropping of bombs. Therefore, no further explanation of air unit flc is needed in this tutorial.

How to create a convincing new unit animation

(A tutorial on creating Sid Meier's Civilization III© unit animations for Mod makers by Dark Sheer)

Part 5 - Other things to consider

Now that we have covered all the animation flc, let us look at other things that we have touches on a little along the way such as light and shadow. While light and shadow can be represented by white and black under the normal circumstances, light and shadow on clothing is a different matter. Usually clothing's color in a unit's flc is represented by the so-called unique Civ's color from the first 64 spot in the palette. Therefore, shadow in this case is represented by a darker shade of color. So, do not color your unit's unique color with only one shade! The different shade does not only make your final unit looks better, it also show depth. Remember, anything that has only one shade of color is flat. Only shades of the same color can bring up the depth of a drawing.

Other than the shade of color to show the depth, the shadow at the feet of a unit or the bottom of a vehicle or around the body of a ship is also very important. Without the shadow the unit will look as if it is floating in the air. Furthermore, if an air unit is without shadow, it is very hard for you to tell the height the unit is on.

Another thing worth mention when creating a new flc is, do not try to reinvent the wheel. Study what is available with the current units' flc and learn from the real animation artist from FIRAXIS. They are the professional, we are just amateurs (well, at least I am not a professional artist). If a sequence looks good enough, use it! If a sequence is not available, look at a real life sequence, I try to model my scud launcher's missile launching sequence as close to the launching of a rocket as possible. Think logically and realistically. What I am trying to say here is, do not bend the arms or legs of a unit in an unrealistical angle. Swing a sword like it's a real sword, think of its weight etc.

If you have problem in coming up with the first frame from scratch, look for some drawings or picture. There are lots of good pictures and photograph to starts you off from the Internet or even some other great war games. Try to get the correct statistic such as length, width and displacement (such as those supplied by dannyevilcat when we try to come up with the Perry Class Frigate). And always use a current unit as a base to compare the size such as length and height of your new unit. Do not create a unit in isolation so that the unit will not look too big or too small when it sits next to another unit in the game.

All these might be hard work, but when you get someone who has work on the ship or vehicle that you have created and tell you that it looks exactly like the real thing, I believe any amount of hard work you put into the animation is worth it.