

Weapons Tutorial 1.0
By
Chavez

Warning: *This is not the final product it will change with the progress of the community.*

For this tutorial I will show you the way I successfully imported A MK23 pistol.

Preparing the Scripts

We will only talk about the basics because everything else is pretty much the same.

- ClassRegistry
- Pickup
- Weapon script (recycled script for this example)

Lets take a look at the [ClassRegistry.lua](#).

First let's go ahead and add our pickup Class

```
EntityClassRegistry =
{
-- entity_type, name, ID, script_file (this is pretty self explanatory)

-- weaponpickups

    Type,                Name, ID, Script File
    {"Pickup",           "PickupFalcon",27,"Pickups/PickupFalcon.lua"},
    -- 27
    {"Pickup",           "PickupAG36",28,"Pickups/PickupAG36.lua"},
    -- 28
    {"Pickup",           "PickupMP5",29,"Pickups/PickupMP5.lua"},
    -- 29
    {"Pickup",           "PickupMachete",30,"Pickups/PickupMachete.lua"},
    -- 30
    {"Pickup",           "PickupShotgun",31,"Pickups/PickupShotgun.lua"},
    -- 31
    {"Pickup",           "PickupSniperRifle",32,"Pickups/PickupSniperRifle.lua"},
    -- 32
    {"Pickup",           "PickupOICW",33,"Pickups/PickupOICW.lua"},
    -- 33
    {"Pickup",           "PickupRL",34,"Pickups/PickupRL.lua"},
    -- 34
    {"Pickup",           "PickupP90",35,"Pickups/PickupP90.lua"},
    -- 35
    {"Pickup",           "PickupM4",36,"Pickups/PickupM4.lua"},
    -- 36
    {"Pickup",           "PickupShocker",37,"Pickups/PickupShocker.lua"},
    -- 37
    {"Pickup",           "PickupWrench",38,"Pickups/PickupWrench.lua"},
    -- 38
    {"Pickup",           "PickupM249",39,"Pickups/PickupM249.lua"},
    -- 39
    {"Pickup",           "PickupMK23",207, "Pickups/PickupMK23.lua"},

```

Note: It's always a good idea to add your comments so you can keep track of what was done when!!!!

--207 Added 8/12/04 Mk23

PickupClass

We could just make our script from scratch but what's the point. In this example we already have something in common with an existing weapon.

Here is the original

```
Script:LoadScript("scripts/default/entities/pickups/basepickup.lua");

local params={
    weapon="Falcon",
    ammotype="Pistol",
    model="Objects/Weapons/DE/DE_bind.cgf",
    default_amount=32,
    sound="sounds/weapons/DE/deweapact.wav",
}

PickupFalcon=CreateWeaponPickup(params);
```

Here is our new one

```
Script:LoadScript("scripts/default/entities/pickups/basepickup.lua");

local params={
    weapon="MK23",
    ammotype="Pistol",
    model="Objects/Weapons/MK23/MK23_bind.cgf",
    default_amount=32,
    sound="sounds/weapons/DE/deweapact.wav",
}

PickupMK23=CreateWeaponPickup(params);
```

Weapon Script

*Note: (--) represents a comment and in order to show you the changes made I simply commented out the original line. The comments that are **red** are the original comments blue comments are made for reminder and explanation.*

```
MK23 = {
  -- DESCRIPTION
  -- Default weapon, loud and does not travel far but does the job
  name = "MK23",
  object = "Objects/Weapons/MK23/MK23_bind.cgf", --third person
--object = "Objects/Weapons/de/de_bind.cgf",
  character = "Objects/Weapons/MK23/MK23.cgf", --first person
--character = "Objects/Weapons/de/de.cgf",

  PlayerSlowDown = 1.0,
  -- factor to slow down the player when he holds that weapon
  -----
  ActivateSound = Sound:LoadSound("Sounds/Weapons/DE/deweapact.wav",0,120),
  -- sound to play when this weapon is selected
  -----

  MaxZoomSteps = 1,
  ZoomSteps = { 1.2 }, --1.2
  ZoomActive = 0,
  AimMode=1,

  ZoomOverlayFunc=AimModeZoomHUD.DrawHUD,
  ZoomFixedFactor=1,
  ZoomNoSway=1,

  FireParams = { -- describes all supported
firemodes
  {
  pistol
    type = 2, -- used for choosing animation - is
    min_recoil=2,
    max_recoil=2.5,
    HasCrosshair=1,
    AmmoType="Pistol",
    reload_time= 1.62,
    fire_rate= 0.2,
    distance= 500,
    damage= 11,
    damage_drop_per_meter= 0.008,
    bullet_per_shot= 1,
    bullets_per_clip=8,
    fire_activation=bor(FireActivation_OnPress,FireActivation_OnHold),
    FModeActivationTime = 2.0,
    iImpactForceMul = 20,
    iImpactForceMulFinal = 65.02,
```

```
BulletRejectType=BULLET_REJECT_TYPE_SINGLE,
```

```
-- make sure that the last parameter in each sound (max-distance) is equal to  
"whizz_sound_radius"
```

```
whizz_sound_radius=8,  
whizz_probability=350,    -- 0-1000  
whizz_sound={  
    Sound:Load3DSound  
("Sounds/weapons/bullets/whiz1.wav",SOUND_UNSCALABLE,100,1,8),  
    Sound:Load3DSound  
("Sounds/weapons/bullets/whiz2.wav",SOUND_UNSCALABLE,100,1,8),  
    Sound:Load3DSound  
("Sounds/weapons/bullets/whiz3.wav",SOUND_UNSCALABLE,100,1,8),  
    Sound:Load3DSound  
("Sounds/weapons/bullets/whiz4.wav",SOUND_UNSCALABLE,100,1,8),  
},
```

```
FireSounds = {  
    "Sounds/Weapons/DE/FINAL_MONO_DEFIRE1.wav",  
    "Sounds/Weapons/DE/FINAL_MONO_DEFIRE2.wav",  
    "Sounds/Weapons/DE/FINAL_MONO_DEFIRE3.wav",  
},
```

```
FireSoundsStereo = {  
    "Sounds/Weapons/DE/FINAL_STEREO_DEFIRE1.wav",  
    "Sounds/Weapons/DE/FINAL_STEREO_DEFIRE2.wav",  
    "Sounds/Weapons/DE/FINAL_STEREO_DEFIRE3.wav",  
},
```

```
DrySound = "Sounds/Weapons/DE/dryfire.wav",
```

```
ShellCases = {  
    geometry=System:LoadObject("Objects/Weapons/shells/smgshell.cgf"),  
    focus = 1.5,  
    color = { 1, 1, 1},  
    speed = 0.1,  
    count = 1,  
    size = 4.0,  
    size_speed = 0.0,  
    gravity = { x = 0.0, y = 0.0, z = -9.81 },  
    lifetime = 5.0,  
    frames = 0,  
    color_based_blending = 0,  
    particle_type = 0,  
},
```

```
SmokeEffect = {  
    size = {0.15,0.07,0.035,0.01},  
    size_speed = 1.3,  
    speed = 9.0,  
    focus = 3,  
    lifetime = 0.5,
```

```

        sprite = System:LoadTexture("textures\\cloud1.dds"),
        stepsoffset = 0.3,
        steps = 4,
        gravity = 1.2,
        AirResistance = 3,
        rotation = 3,
        randomfactor = 50,
    },

    MuzzleEffect = {
        size = {0.1},--0.15,0.25,0.35,0.3,0.2},
        size_speed = 4.3,
        speed = 0.0,
        focus = 20,
        lifetime = 0.03,
        sprite = System:LoadTexture
("Textures\\WeaponMuzzleFlash\\muzzle1.dds"),
        stepsoffset = 0.15,
        steps = 1,
        gravity = 0.0,
        AirResistance = 0,
        rotation = 30,
        randomfactor = 10,
    },

    -- remove this if not nedded for current weapon

    MuzzleFlash = {
        geometry_name = "Objects/Weapons/Muzzle_flash/mf_DE_fpv.cgf",
        bone_name = "spitfire",
        lifetime = 0.07,
    },

    MuzzleFlashTPV = {
        geometry_name = "Objects/Weapons/Muzzle_flash/mf_DE_tpv.cgf",
        bone_name = "weapon_bone",
        lifetime = 0.05,
    },

    -- trace "moving bullet"
    -- remove this if not nedded for current weapon
    Trace = {
        geometry=System:LoadObject("Objects/Weapons/trail.cgf"),
        focus = 5000,
        color = { 1, 1, 1},
        speed = 110.0,
        count = 1,
        size = 1.0,
        size_speed = 0.0,
        gravity = { x = 0.0, y = 0.0, z = 0.0 },
        lifetime = 0.04,
        frames = 0,
        color_based_blending = 3,
        particle_type = 0,
        bouncyness = 0,
    },

```

```

    },
    SoundMinMaxVol = { 200, 4, 2600 },
    LightFlash = {
        fRadius = 3.0,
        vDiffRGBA = { r = 1.0, g = 1.0, b = 0.7, a = 1.0, },
        vSpecRGBA = { r = 1.0, g = 1.0, b = 0.7, a = 1.0, },
        fLifeTime = 0.1,
    },
},
},
},

SoundEvents={
    --      animname,      frame,      soundfile      ---
    {      "reload1",      10,      Sound:LoadSound
("Sounds/Weapons/DE/DEclipout_10.wav",0,100)},
    {      "reload1",      25,      Sound:LoadSound
("Sounds/Weapons/DE/DEclipin_25.wav",0,100)},
    {      "reload1",      38,      Sound:LoadSound
("Sounds/Weapons/DE/DEclipslap_38.wav",0,100)},
    --      {      "swim",      1,      Sound:LoadSound
("Sounds/player/water/underwaterswim2.wav",0,255)},
    },
}

```

```
CreateBasicWeapon(MK23);
```

```
-----
--ANIMTABLE
-----
```

```

MK23.anim_table={}
--SINGLE SHOT
MK23.anim_table[1]={
    idle={
        "Idle11",
        "Idle21",
    },
    reload={
        "Reload1",
    },
    fidget={
        "fidget11",
        "fidget21",
    },
    fire={
        "Fire11",
        "Fire12",
    },
    melee={
        "Melee"
    }
}

```

```
    },
    swim={
        "swim"
    },
    activate={
        "Activate1"
    },
}
```

Basically all that was done was changing some of the naming as well as pointing to our own model location.

Next I added a little bit to the `WeaponSystems.lua`

```
--Script:LoadScript("SCRIPTS/Default/Entities/Weapons/WeaponsParams.lua");
```

```
WeaponClassesEx = {
```

```
MY NEW ONE WITH OUR OWN ID
```



```

MK23 = {
    id = 17, --ID
    script = "Weapons/MK23.lua",
},

```

ORIGINAL

```

Falcon = {
    id = 10,
    script = "Weapons/Falcon.lua",
}

```

All of this just points to our *WeaponsParameters.lua* which is where we change the functionality of our weapon. *“which I did not bother doing simply copied and pasted the original and renamed it.”*

```

WeaponsParams={

```

```

-----
--MK23 (Pistol/Silenced Weapon)
-----

```

```

    MK23={
        Std=
        {
            -----SINGLE FIRE-----
            {
                fire_mode_type = FireMode_Instant,
                damage_type = "normal",
                aim_improvement = 0.03,
                aim_recoil_modifier = 0.5,
                accuracy_decay_on_run = 0.8,
                min_accuracy = 0.75,
                max_accuracy = 0.90,
                reload_time = 1.62,
                fire_rate = 0.5,
                tap_fire_rate=0.2,
                distance = 60,
                damage = 70,
                damage_drop_per_meter = .1,
                bullet_per_shot = 1,
                min_recoil = 1.0,
                max_recoil = 2.0,
                iImpactForceMul = 100,
                iImpactForceMulFinal = 40,
                iImpactForceMulFinalTorso = 130,
                hud_icon = "single",
                accuracy_modifier_prone = 0.5,
                accuracy_modifier_crouch = 0.7,
                recoil_modifier_standing = 1.0,
            }
        }
    }
}

```

```

    },
  },
  --In multiplayer, if this table exist will be merged with the table above for
  the firemodes in common.
  Mp=
  {
    {
      accuracy_modifier_standing = 0.75,
    },
  },
},

```

Preparing our models

- First things first we need a weapon

This weapon is from the kick-ass Far Cry mod Navy SEALs Preemptive Ops look them up they are a talented Team.

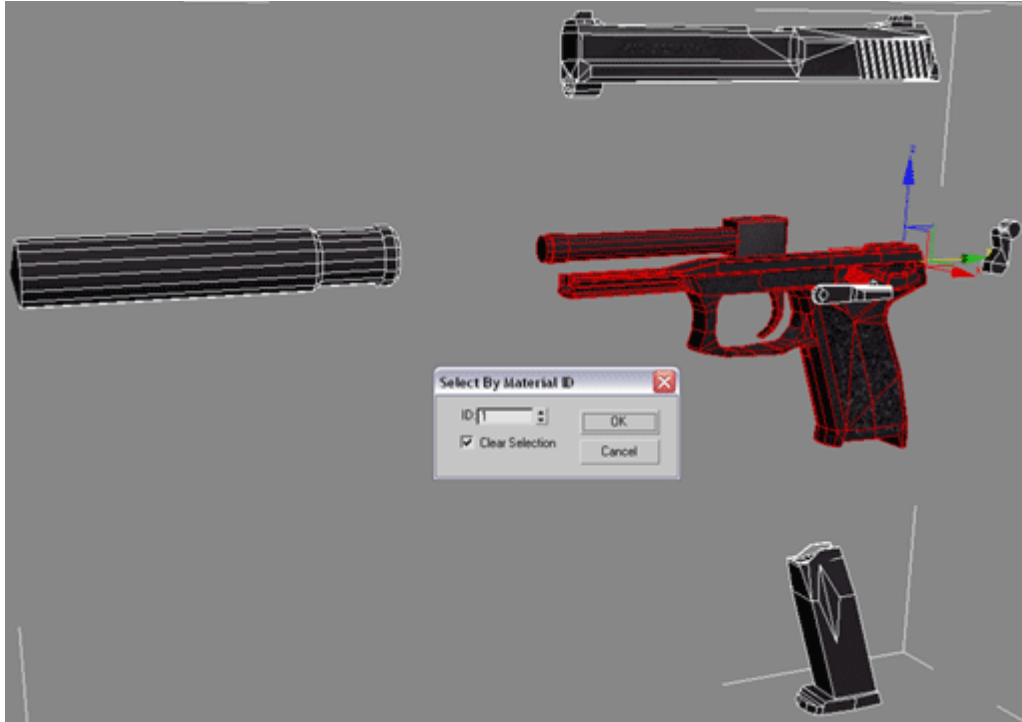


- Next we will also go to the SDK example and steal the arms ;)
- After that we will have to put it all together so that we can import to the game.

Starting with the weapon

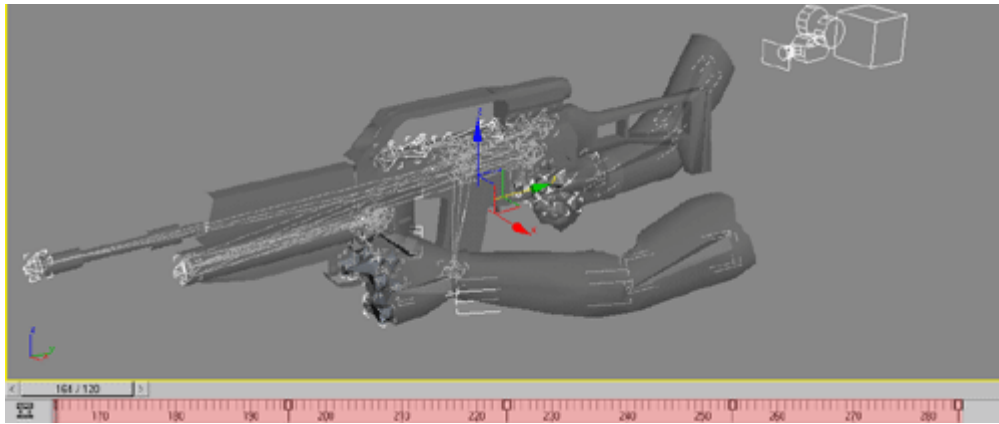
Open up your model and begin separating it into parts namely the ones that will receive animation. Such as the image below.

Tip: an easy way to do this is when modeling the weapon take the animation into consideration and assign a material ID to each individual piece that will receive animation. By doing this you can select each piece without a whole lot of trouble.



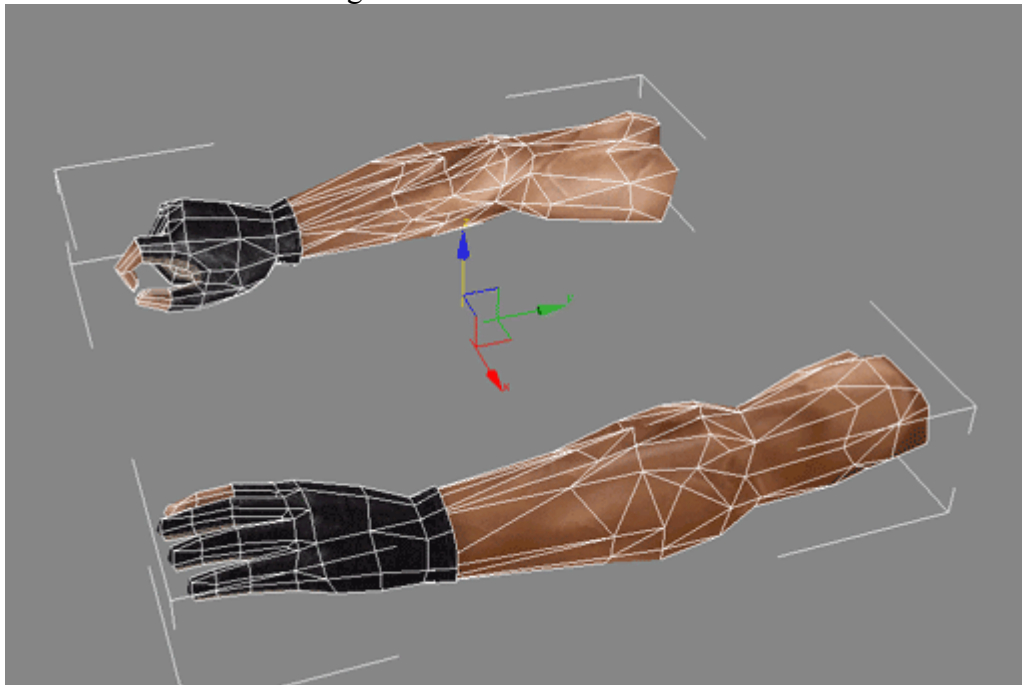
Once all of your pieces are separated save your weapon as a separate file like (ex. weapMerege.max). Next export your weapon as a .3ds file. Ok lets join it up with the arms so lets open up the SDK example.

Merging the arms

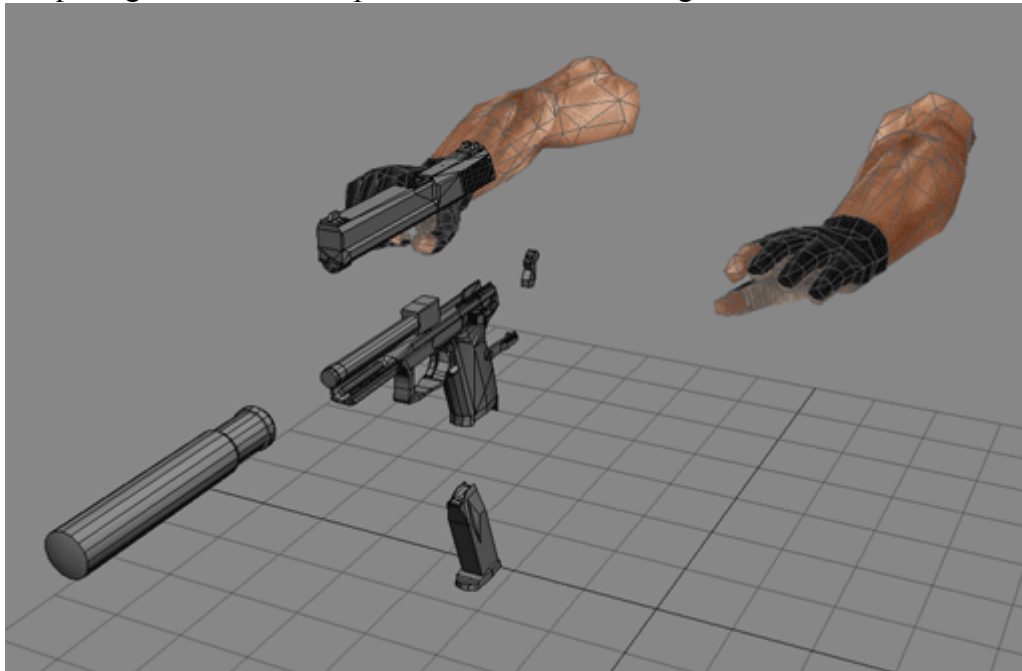


First select everything in the scene, then select all the animation key frames and delete them. (The animation key frames are the red highlighted area)

Next select the arms and delete the Physique modifier, and in poly subobject mode delete the weapon. Deselect the arms and select everything in the scene **EXCEPT** the arms and delete. You should have something that looks like this.



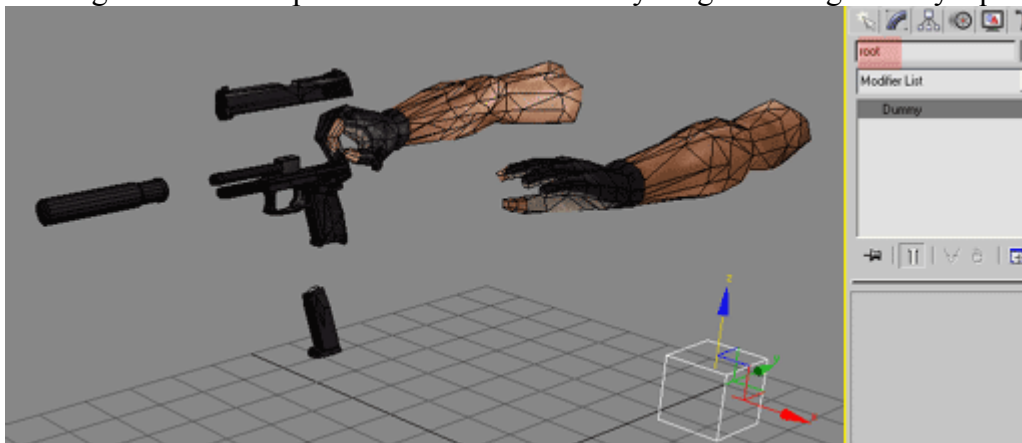
Now it is time to merge the weapon with the arms so got to File>Import and find your weapMerge.3ds file and import. Position it something like this.



And attach the arms to the weapon. Now comes the fun part reassign all your material so that the textures are coming from the folder in which you will export your model. Example `c:\MasterCD\Mods\Mod Name\ Objects\Weapons\MK23`. In other words place your textures that you compressed into .dds files in your weapons folder of your mod and then reassign to the model.

Setting up the rig


Ok take in to account I am not an actual “professional” animator so everything I do may be the wrong way to do it. However it worked and I guess that’s the main thing. Let’s start by creating our “root” helper in which to attach everything including our Physique modifier.

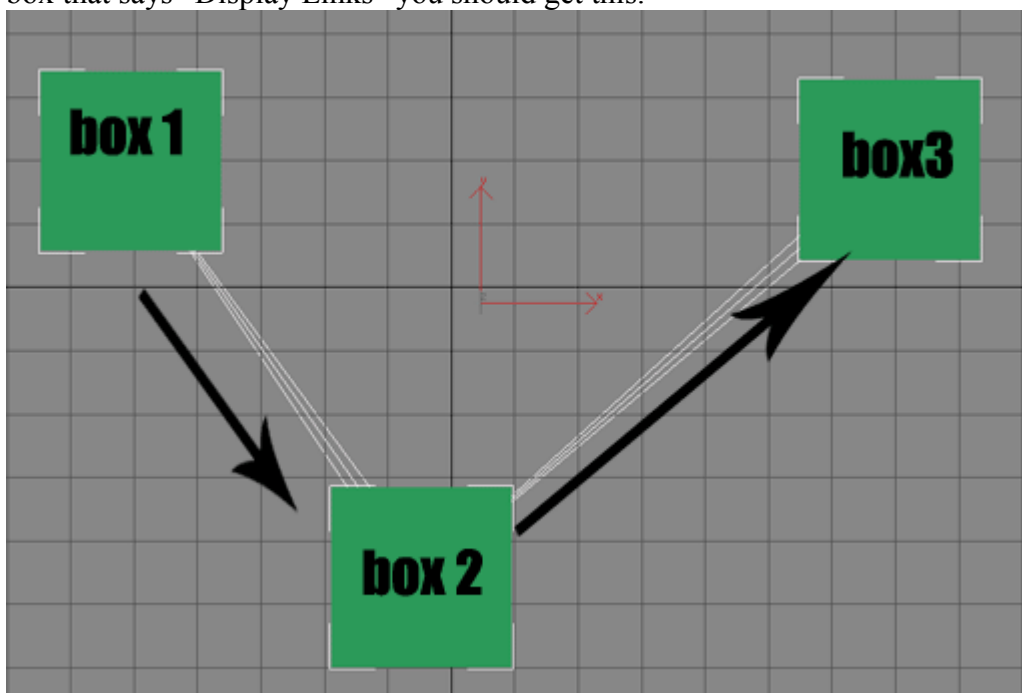


Next we will make the bones for the weapon. By creating a small box (or cube).

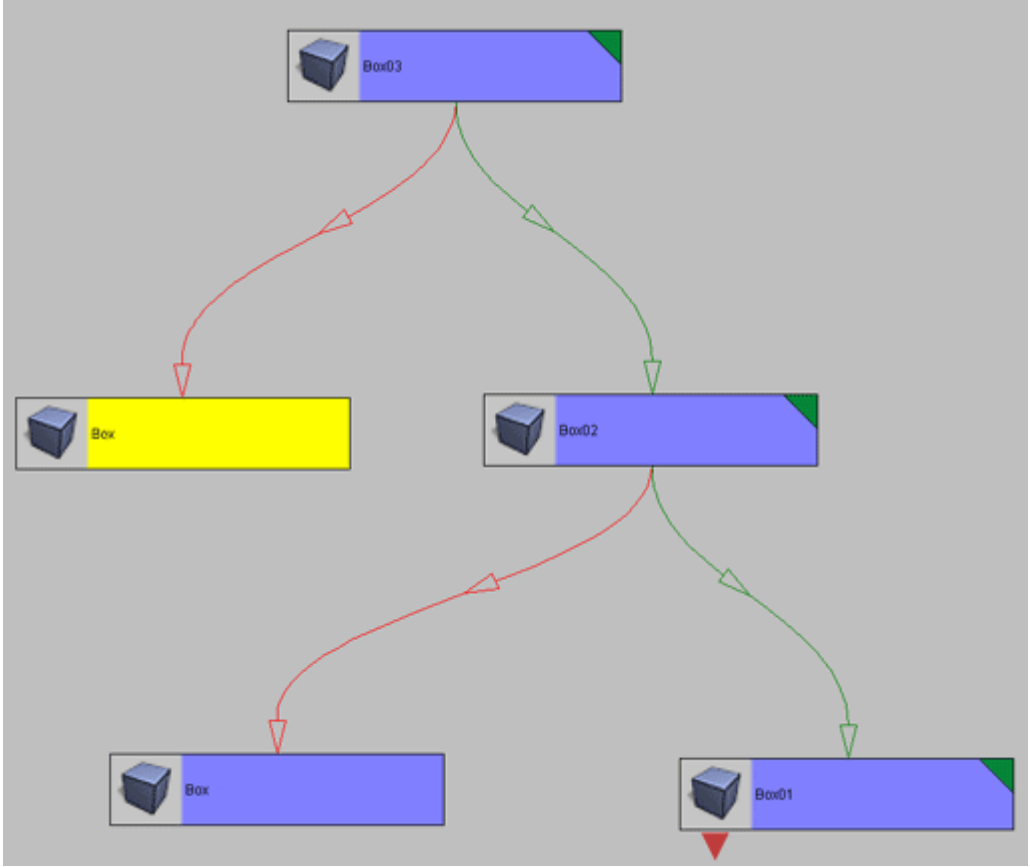


You should have something that looks like this so far each cube is a pivot point for our bones. I have the “weapon” pivot selected this will be the root for the gun. By using the “Select and Link” tool start from the outer most boxes and link them to the root in which we name “weapon”. The reason for this is because we want the whole weapon to move by selecting a single point in our hierarchy i.e. our root “weapon”. The way this works is when you select an object it is the child and what you drag and link it to is its parent to see what I mean create a

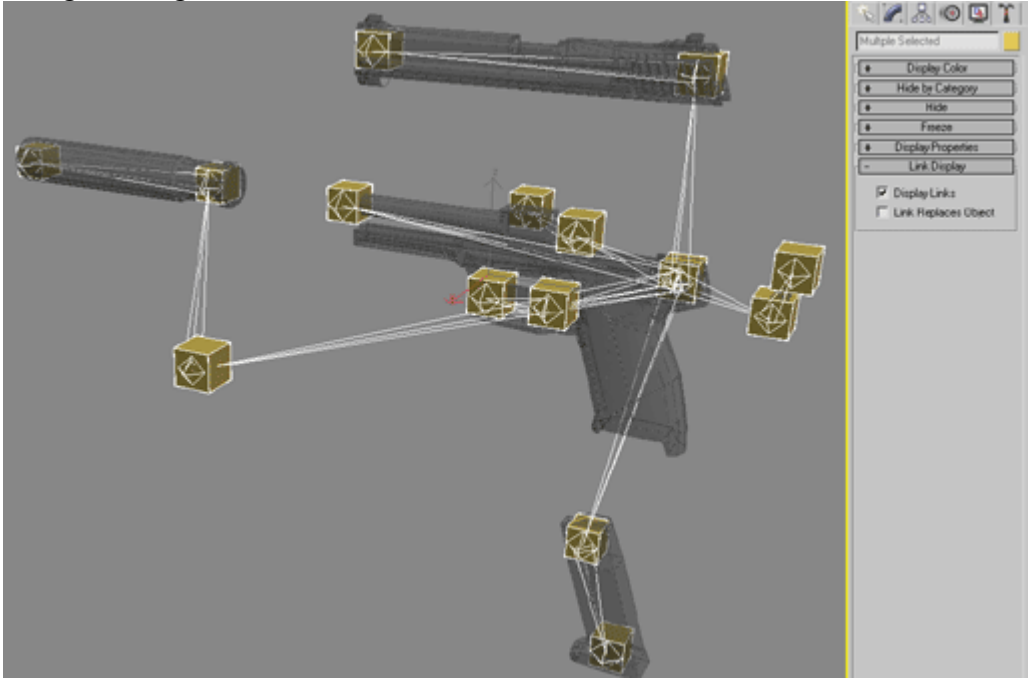
three cubes off to the side and grab the “Select and Link” tool . Click on the first cube and hold down the mouse button you will get a little dotted line when you drag your mouse around. Select the next cube and release the mouse to link the two. Go ahead and link the second cube to the third. Select all the cubes and go to Display>Link Display and check the box that says “Display Links” you should get this.



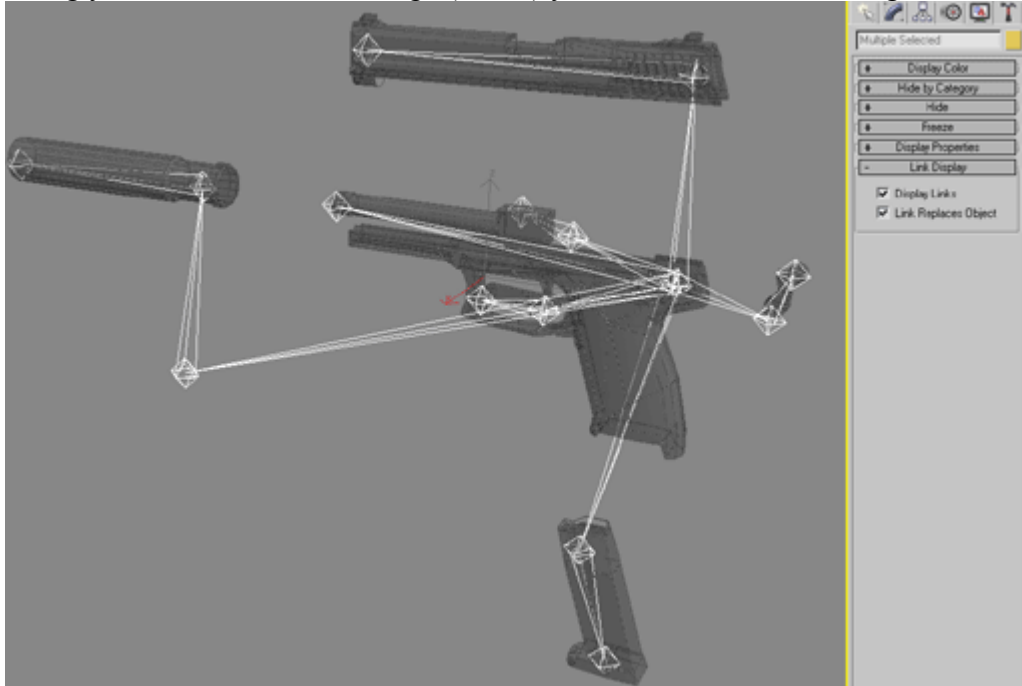
Box1 is the child of Box2 and Box3. Box 2 is the child of Box3 and the parent of Box1 and in this example Box3 is the parent over all. For further demo purposes go ahead and click “Link Replaces Objects” from the same menu. Now grab any link and move it around to understand the relationship each object has with each other. Here is a schematic view of the parent child relationship.



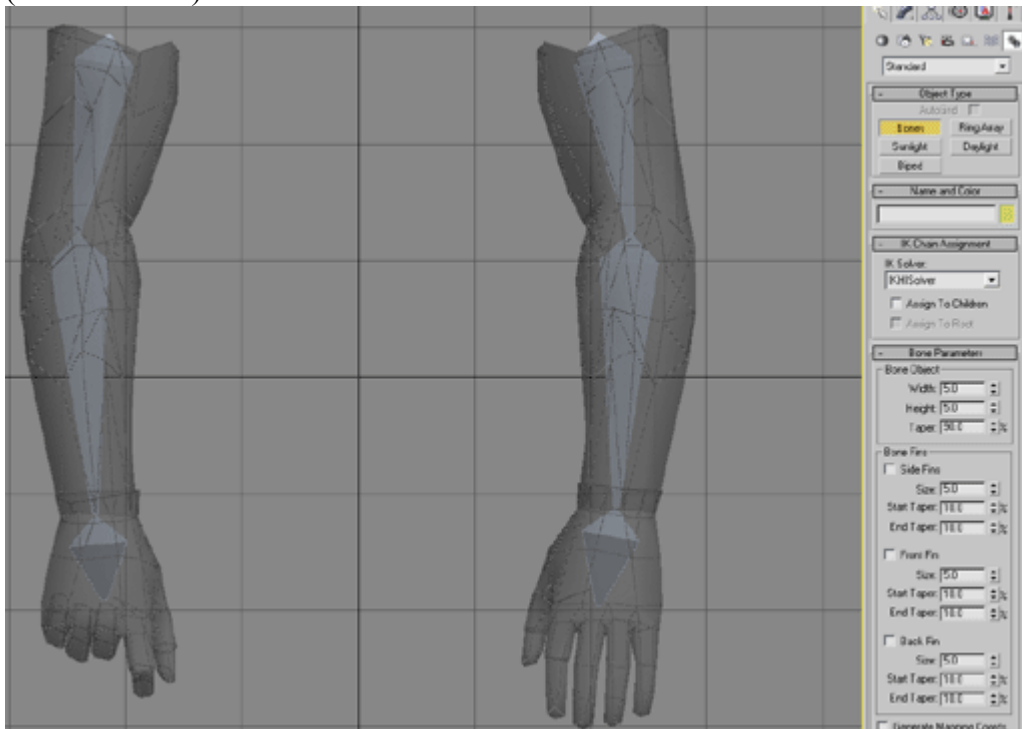
Alright enough of that back to the task at hand.



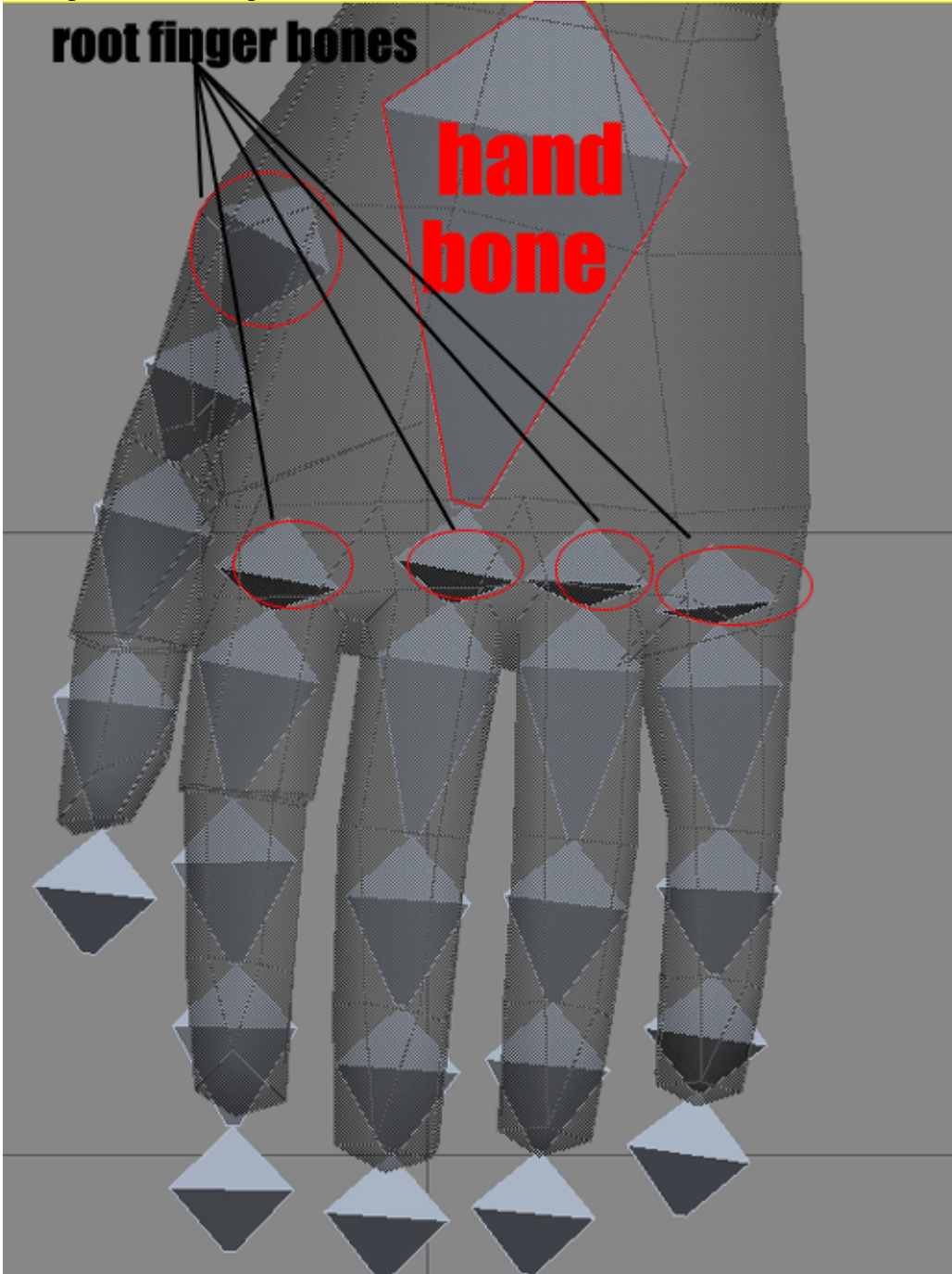
Using your new found knowledge (or not) you should have something like the above.



Now let's rig the arms. Go to Create>Systems>Bones and set the width and height to 5 and the taper to 90%. And starting from the upper arm create a chain that stops at the hand, making sure to delete the end bone that occurs when you right click to end creation like this. (On both sides)

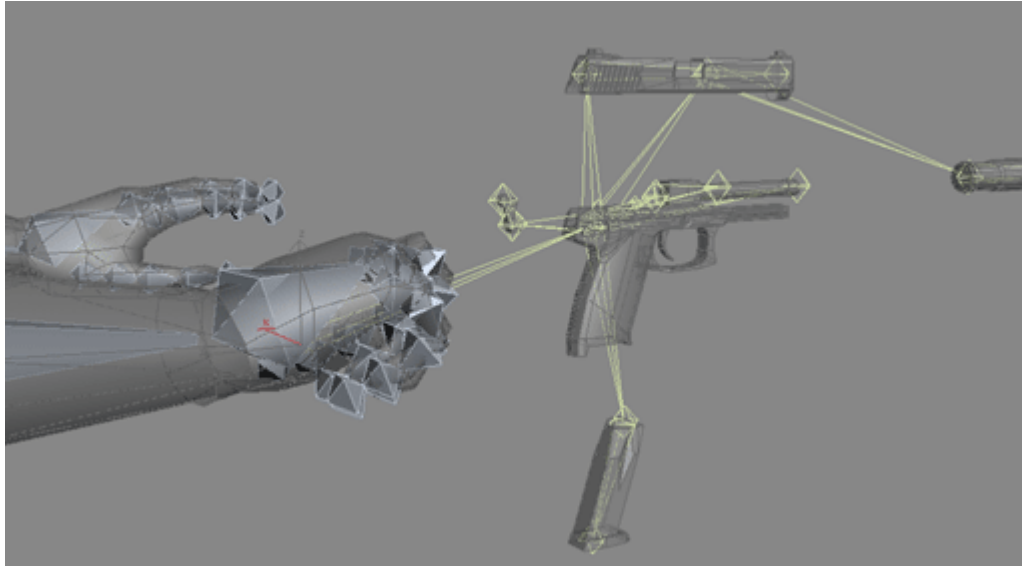


Change the bone height and width to 2 and continue on like this.

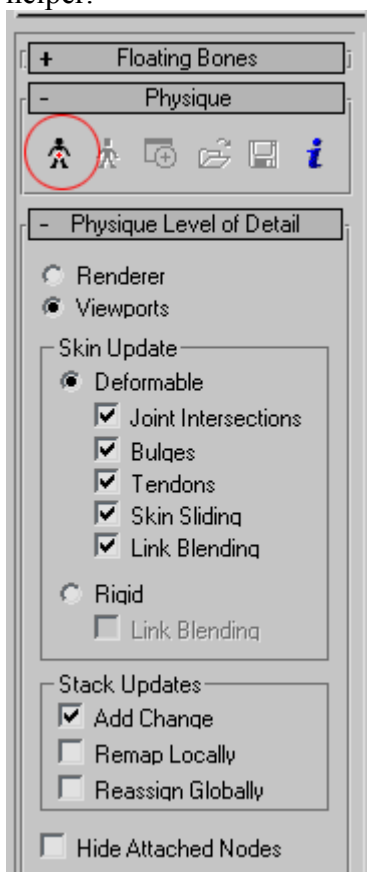


Now in the sdk manual it says that the fewest bones the better but I want more animation in the hand hehe so I am going to do it this way. Once you have edited the bones so they fit inside the arms and hands and have linked the root finger bones to its' corresponding hand bone you can then link the "weapon" bone to the right hand or left depending on your preference using the same method we used for the weapon like this.

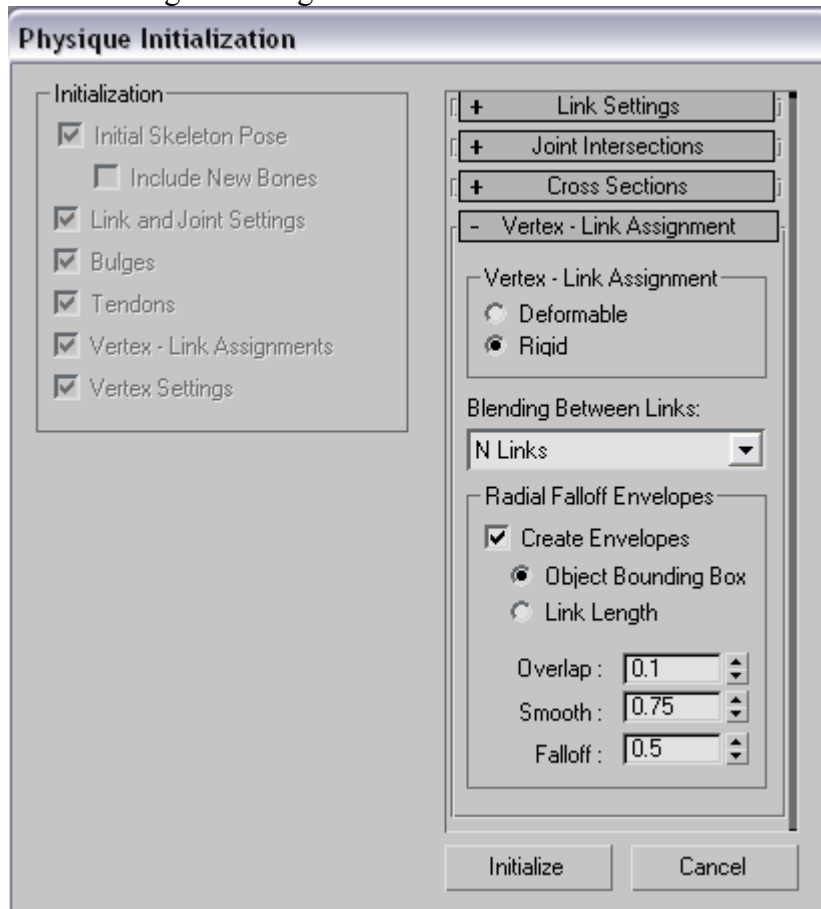
Note: important thing to remember is to make sure all your bones have there own names Such as Bone01, Bone02n, Bone03etc...



Now that everything is in place select and link the upper arm bones (one at a time) i.e. the parent bones and link them to the "root" helper. Now we can apply our Physique modifier. So select your mesh and apply your Physique modifier. Click this button and select your "root" helper.



You should get a dialog like this



Just click rigid and initialize.

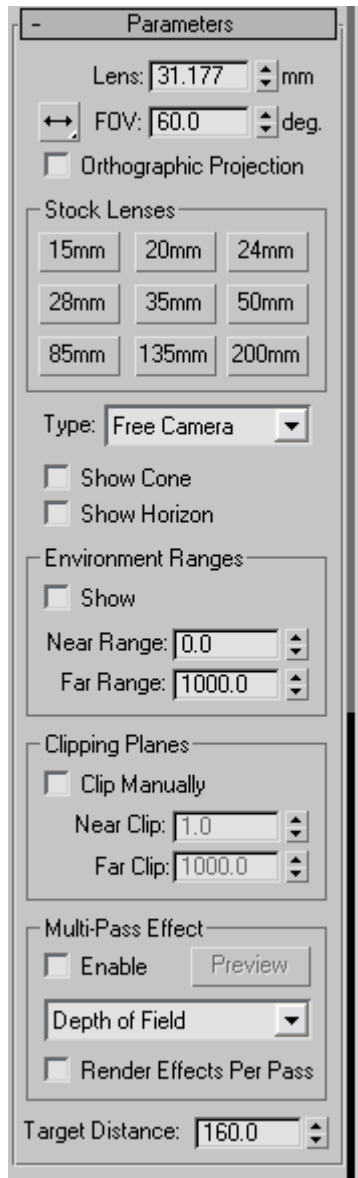
You should now be able to adjust your envelopes, position your arms, weapon, etc to get it ready for animation and export. I am not going to show that because you can find a trillion tutorials on how to do that such as the ones that come with your copy of Character Studio.

<ftp://ftp1.discreet.com/web/support/3dstudio/cs4tutorials.exe>

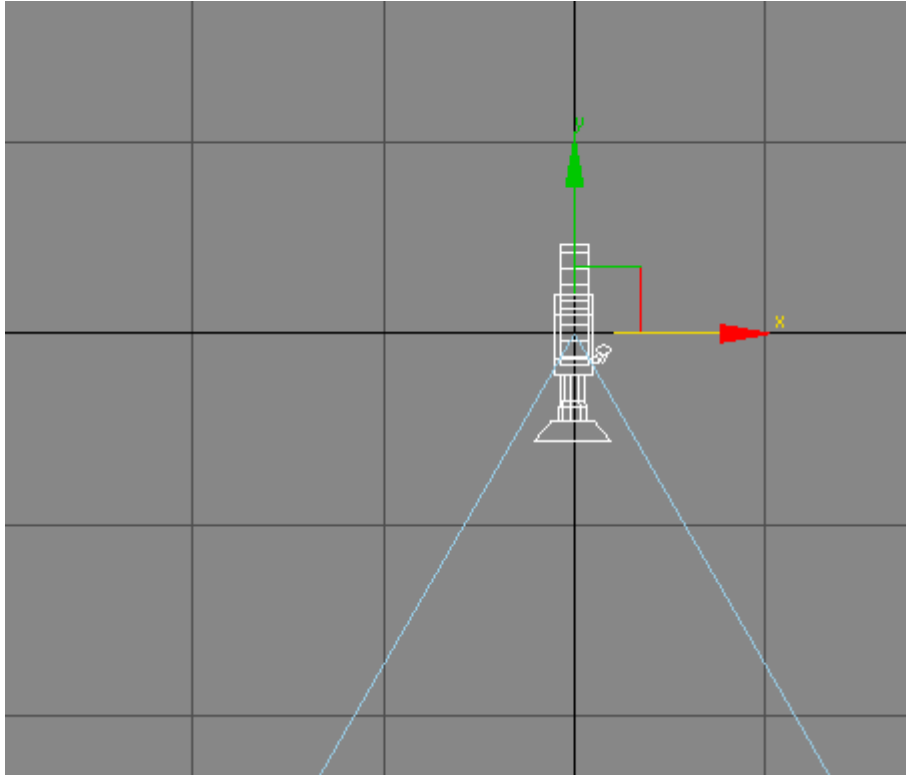
Preparing for Export

OK, now that we have done all that, it is time to see the fruits of our labor.

Let's start by preparing our scene. Create a "free" camera centered to the world at x0.0, y0.0, z0.0.



Like this....



Ok now let's get our in-game pose by selecting our favorite viewport and changing the view to camera01 or whatever your camera happens to be. Now line your model up (using the "root" helper) so that it is pointing center of the screen or whatever suites you. This is how mine looks....



OK assuming you have done your animations (or not) and have applied the shaders we are ready to export. So bring up your handy dandy Cyrtek Exporter and select just your mesh and click add selected and Export.

Select the correct destination in this case it would be (where you put the textures) MasterCD\Mods\Mod Name\Objects\Weapons\MK23\ and save it as the name you specified in the script in this case MK23.cgf.

Next go to the bones Exporter and if it isn't already there select just the "root" helper and add selected node and export this is where its is nice to have your animations in separate max scenes because you must save the bones to an animation such as activate.caf but including the name of your model such as MK23_Activate1.caf.

Here is a shot of what my folder looks like.

Name	Size	Type	Date Modified
MK23.cal	1 KB	CAL File	8/12/2004 3:36 PM
MK23.cgf	525 KB	CGF File	8/12/2004 9:55 PM
MK23.dds	5,462 KB	DDS File	8/12/2004 9:53 PM
MK23_Activate1.caf	5 KB	CAF File	8/12/2004 9:09 PM
MK23_bind.cgf	230 KB	CGF File	8/12/2004 9:58 PM
MK23_Deactivate1.caf	4 KB	CAF File	8/12/2004 5:33 PM
MK23_Fidget11.caf	6 KB	CAF File	8/12/2004 9:47 PM
MK23_Fidget21.caf	6 KB	CAF File	8/12/2004 9:34 PM
MK23_Fire11.caf	6 KB	CAF File	8/12/2004 9:34 PM
MK23_Fire12.caf	6 KB	CAF File	8/12/2004 9:34 PM
MK23_Grenade11.caf	4 KB	CAF File	8/12/2004 5:34 PM
MK23_Grenade12.caf	4 KB	CAF File	8/12/2004 5:34 PM
MK23_Idle11.caf	6 KB	CAF File	8/12/2004 9:35 PM
MK23_Idle21.caf	6 KB	CAF File	8/12/2004 9:35 PM
MK23_Melee.caf	4 KB	CAF File	8/12/2004 5:34 PM
MK23_Reload1.caf	4 KB	CAF File	8/12/2004 5:34 PM
MK23_swim.caf	4 KB	CAF File	8/12/2004 5:34 PM

And here is .cal

```
$AnimDir = .  
  
$AutoUnload = no  
$DelayLoad = no  
  
activate1 = MK23_activate1.caf  
fidget11 = MK23_fidget11.caf  
fidget21 = MK23_fidget21.caf  
fire11 = MK23_fire11.caf  
fire12 = MK23_fire12.caf  
grenade11 = MK23_grenade11.caf  
grenade12 = MK23_grenade12.caf  
idle11 = MK23_idle11.caf  
idle21 = MK23_idle21.caf  
melee = MK23_melee.caf  
reload1 = MK23_reload1.caf  
swim = MK23_swim.caf  
deactivate1 = MK23_Deactivate1.caf  
$AutoUnload = yes  
$DelayLoad = yes
```

Nothing more than a copy of the DE.cal and renamed to the names of my animations.

Simple. All placed inside the same folder you should now be able to open your editor and place it inside your map. Or go to the game in dev mode and have all weapons applied.

Well that is all I did I will update this as the info is accumulated and re-release it to the public. I want to thank Navy SEALs Preemptive Ops for the Kickass weapon model and you guys for reading, learning and helping the community grow if anyone has any questions please come to our (Jungle Combat) forums and ask a question in the “Mod Creations” section. We will get you the answers you need and if anyone would like to join Jungle Combat drop by the forums as well and give us a shout.

Also a special thanks to a fellow modder and a super guy tstrait72 for evaluating this tutorial. Tstrait72 works with another cool mod called...



[Jungle Combat](#)

[Jungle Combat Forums](#)

<http://www.crymod.com/index.php?templateid=showposts&fid=10&tid=508>

Thanx again

~~Chavez