Toons on Film

Making Movies in
city of heroes

Presented By Michelle Travis

<u>Acknowledgements</u>

Thanks to:

Ex Libris and Lighthouse, for allowing me to present at HeroCon 2008

The *City of Heroes/City of Villains* NCNorCal development team, for being fans of my work

The *City of Heroes/City of Villains* videographer community, for graciously sharing their expertise, advice, good humor, and support

The Sinclair Syndicate, for helping me put stuff together

My husband John, for everything else

And oh, yeah… Castle, please note that you're on the cover. I don't want to hear another complaint that you never appear in my stuff.

For more information, please visit the

Samuraiko Productions website at

http://www.samuraikoproductions.com/

Introduction to City of Heroes/City of Villains Machinima

Hi. You're probably reading me while Michelle is busy talking, which is fine. Because she'll mention stuff that's not in me, and I'll mention stuff that she'll forget to talk about because she's nervous.

Anyway, I'm the Handout for Michelle's workshop "Making Movies in City of Heroes" – within these pages, you'll find how to use the /demorecord command, what a demofile looks like, how to edit the demofile to make your characters do stuff they normally can't, and even a few things about how to use Sony Vegas Movie Studio to make your machinima. But first, let's address a primary question – what *is* machinima? According to Wikipedia, machinima is defined as:

> *"A collection of associated production techniques whereby computer-generated imagery (CGI) is rendered using real-time, interactive 3-D engines instead of professional 3D animation software. Engines from first-person shooter and role-playing simulation video games are typically used. Consequently, the rendering can be done in real-time using PCs (either using the computer of the creator or the viewer), rather than with complex 3D engines using huge render farms. Usually, machinima productions are produced using the tools (demo recording, camera angle, level editor, script editor, etc.) and resources (backgrounds, levels, characters, skins, etc.) available in a game."*

A rather lengthy definition, huh? Short answer – machinima is making videos using a game instead of traditional animation software.

Machinima can take a lot of different forms. Some videographers use it to make game trailers, touting their favorite MMORPG or first-person-shooter game by showing off how cool it is. Others make music videos, recording or creating in-game action to a certain song or instrumental piece of music. Still others use machinima in a slightly looser sense to record in-game footage to show how the game is played (say, how to play a certain archetype).

Quite a few CITY OF HEROES/CITY OF VILLAIN machinima have already been created – if you visit the Multimedia section of the Forums, you'll find a stickied thread called "Fan Videos," which is a constantly-updated list of all sorts of videos, created by dozens of different players. Some are hosted on YouTube, others on personally owned and maintained sites, but all are available for your enjoyment.

What a lot of people don't know is that CITY OF HEROES/CITY OF VILLAINS (or CoX for short) has the aforementioned tools for making videos. The CoX software and game engine have the capacity to record what goes on in the game, modify camera angles, swap out character models, and other things to allow a remarkable amount of customization, without affecting the actual in-game experience or the game engine itself.

These tools are how videographers make the nifty videos they do, allowing the videographers to make Lord Recluse mosh, make Carnies dance, and even make the Kraken hold up protest signs. It's all there.

But how do you use these tools?

The Art of Demo Recording

The most important tool in the CoX videographer's arsenal is a single, in-game slash command, called /demorecord. What this command does is tells the game to create a text file, filling it with everything that is going on in the zone while the /demorecord command is active. Every player, every NPC, every clickable item, every power, every in-game zone event, it all gets recorded, exactly as it happens while you're recording. This command will then continue until you a) change zones, b) disconnect, or c) tell the recording to /demostop.

To create a demofile, you enter the command:

> **/demorecord [filename]** (replacing [filename] with what you want to call the file)

So let's say the Rikti are attacking Kings Row, and I want to record the infamous videographer character Johanna Sinclair in action against the Rikti. I'd probably name the file:

> **/demorecord JoVsRiktiKR**

That tells me what's going on, which character it is, and which zone I'm in. So then Jo proceeds to whale on the Rikti for several minutes, while the game busily creates the demofile of her every attack, every blast, every injury, every teleporting Rikti, you name it. Once the invasion is over, and I'm done recording, I type:

> **/demostop**

Now, assuming that you're running Windows, and you installed CITY OF HEROES to your C: drive, if you went to C:\Program Files\City of Heroes, you'd now see a folder called "client_demos". Inside that folder would be a file called "JoVsRiktiKR.cohdemo" – this is your demo file, which contains everything about Jo's battle versus the Rikti.

Although the files have the extension .cohdemo, these are text files, and can be opened and edited in Notepad (my program of preference).

But demofiles can be complicated things, so let's take a step-by-step look at one.

```
1   0   Version 2
0   0   Map maps/City_Zones/City_03_02/City_03_02.txt
0   0   Time 4.950943
```

The first part of any demofile is to define version, place, and time. The version will never change, so don't worry about that. Next, you'll see that there's a map associated with the file. In this case, the map indicated by City_03_02.txt is Independence Port. Lastly, the game engine uses a 24-hour system to determine the day/night cycle. This code indicates it's about five in the morning, in-game time.

```
0   CAM POS -814.441223 5.402523 -5648.874512
0   CAM PYR 0.034 -2.178715 -0
```

The next part of the demofile tells you where the camera starts off, indicated by the camera's Position, or POS for short. The game engine uses 3-d coordinates (x/y/z) to determine a location in-game. For those of you familiar with BadgeHunter or ParagonWiki, you've probably seen coordinates listed for badges and plaques and such. You can find your current location in-game by typing **/loc** at any time.

After that comes the camera's Pitch, Yaw, and Roll, or PYR. This indicates if the camera is tilted in any way, looking up, down, left, right, or slanted. In this code, the camera is tilted very slightly downward, and somewhat to one side.

```
0   66  NEW Statesman
0   66  NPC Model_Statesman
0   66  POS -806 0.6875 -5643
0   66  PYR 0 -2.086214 -0
0   66  MOV READY 0
```

Aha – now we find stuff! Or in this case, a person. Specifically, Statesman standing tall and proud on his aircraft carrier. For NPCs (be they people or items), they'll always have these five lines of code – who or what it is, how it's represented, where it is, how it's positioned, and what it's doing.

The first part gives his name (if it's more than one word, you'll see it in quotes), while the second tells you the in-game model used to represent it. No, you can't modify Statesman to get him out of his uniform, sorry.

Then, just like we saw with our camera, we've got a set of POS coordinates and PYR coordinates. Lastly, we can see what he's doing – which is pretty much nothing.

And like I said, you can't modify Statesman. But as for your own characters… that's a different story!

```
0   2870 Player
0   2870 NEW "Johanna Sinclair"
0   2870 COSTUME 1 c9dfff 4.119521 -0.175800 0.000000 0.413585 0.353637
0.510715 0.120002 -0.080880 0.000000 -0.420000 0.720000 0.000000 -0.470000
0.040000 -0.620000 -0.440000 -0.500000 0.440000 -1.000000 0.410000 -
0.120000 -0.300000 0.700000 0.020000 -0.730000 -1.000000 1.000000 -0.490000
0.470000 0.380000
0   2870 PARTSNAME shorts skin_tights bikini_3 000000 737373
0   2870 PARTSNAME Jackets !Chest_V_Fem_Skin_Wedding_01
!Chest_V_Fem_Skin_Wedding_01_Mask 1f0b00 737373
0   2870 PARTSNAME V_fem_Head.GEO/GEO_Head_V_Asym_Standard
!v_sf_face_skin_head_10 !V_Sf_Face_Makeup_3 220066 737373
0   2870 PARTSNAME Smooth skin_fingerless_01a skin_fingerless_01b 000000
737373
0   2870 PARTSNAME Hi_Heels smooth_matte none 000000 000000
0   2870 PARTSNAME none none none 000000 000000
0   2870 PARTSNAME Long_04 Long_01a Long_01b 001a26 1e1e61
0   2870 PARTSNAME none none none 00000000 00000000
0   2870 PARTSNAME Glasses_05 Glasses_01a Glasses_01b 000000 000000
0   2870 PARTSNAME none none none 000000 e3e3e3
0   2870 PARTSNAME none none none 1f1f1f 737373
```

```
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 PARTSNAME none !Cape_Trenchcoat_Leather_01 none 000000 000000
000000 000000 capes/CapeTrenchFem_02.fx
O   2870 PARTSNAME none none none 614131 000000 000000 000000
Auras/Female/GlowingAura/GlowAuraHead.fx
O   2870 PARTSNAME Pencil_01 !Skirt_Wedding_01 none 1f0b00 000000
O   2870 PARTSNAME V_FEM_TOP.GEO/GEO_Top_Trenchcoat_01 V_Motorcycle_03
V_Motorcycle_02_Mask 000000 000000
O   2870 PARTSNAME V_FEM_SLEEVES.GEO/GEO_SLEEVES_Motorcycle_03
V_Motorcycle_03 none 000000 575757
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 PARTSNAME none none none 00000000 00000000
O   2870 MOV READY 0
O   2870 HP 1444.20
O   2870 HPMAX 1444.20
O   2870 POS -814.441223 0.675949 -5648.874512
O   2870 PYR -0 0.962877 0
```

Wow! Now, I can't really go into what all of this means, but suffice it to say, this big batch of code is what tells the game how you made your character. Every costume part you selected is represented here, along with the hexadecimal color code for said part, and numerical values for those sliding scales that customized your character's face and build. Most of these are fairly self-explanatory, such as auras and costume parts.

You'll also see those same familiar POS and PYR coordinates, plus my character's current and maximum 'hit points'. (Demofiles also track gains and losses in health and endurance during combat, so you can see just how badly Lusca slapped you when you AoE'd his tentacles.) And as we can see from that MOV line, Johanna is also just standing there on Statesman's aircraft carrier.

I could spend a week going on and on about what all is involved in a demofile on a much, much deeper level, but that's a bit beyond the scope of the workshop. So, if you want to know more about what all this stuff means, visit the outstanding articles on Demo Recording at the ParagonWiki website. (Virtually all of the forum-active videographers have contributed to the articles in some form or another.)

As a general rule, though, I'd caution you against messing with your character's code too much unless you know what you're doing. Swapping out colors or an aura is easy enough, but anything more than that, and you'd be better off forking out the inf at the Tailor's to make your change and then /demorecord yourself to get it right.

Let's assume, for brevity's sake, that you understand all this. What does all this mean?

Well, for starters… this is what allows you to switch out who happens to appear in a given demofile. For example, as part of the presentation, you'll film Ms. Liberty standing in Atlas Park… and then you'll see how to put Ghost Widow there instead. Or even your own character, standing there ready to train (or ignore) the masses of low-level heroes begging for guidance. You'll also see how to change the MOV command to something more entertaining than just READY. Imagine seeing Lord Recluse yapping on his cell phone on Ms. Liberty's podium.

You get the idea.

To play back your demofile, I highly recommend Zloth's Demo Player (the link to this free tool can be found in the Even More Useful Stuff section). This then allows you to view what you just recorded (and if you edited it, it will show you the edited version). Hooray!

In-Game Cinematography

While it's usually easier to play CoX when you've got the camera at your back and you're looking where you're going, sometimes you want to film other angles and such. (After all, that's why you get to choose a face for your character!)

One thing you'll want to master is the art of camera angles. You're probably already conversant with the fact that if you hold down the Page Up or Page Down keys while moving your mouse around, you can swing the camera around your character, looking up or down. If you have a scrolling wheel on your mouse, you can also zoom in or out on your character.

The best place to practice this is in your starting zones (Atlas Park, Galaxy City, and Fort Darwin), especially once you've gotten a couple of levels under your belt. This way, you can swing the camera around in front of you or to one side, and /demorecord yourself fighting with stuff without having to worry that you're going to be eating the pavement before you're done.

As long as you've got the /demorecord command active, it will store every camera movement exactly as you control it during your gameplay. You can always then go back later and modify it if necessary; however, in all truth, this can be (and usually is) horrendously time-consuming. Hence my urging you to practice swinging the camera around in a smooth, graceful motion, instead of like a drunken hippo.

You'll also want to practice your timing – although it takes a lot of practice, if you know how long you've got on your Sniper Blast before it goes off, you can get really good at swinging that camera around at just the right second to see your blaster send some unsuspecting Clockwork straight to oblivion. Timing is also important when it comes to stacking emotes or attacks – to that end, I urge you to read some of the excellent guides on keybinds, because these will save you scads of time later on down the road.

But the single most important thing about in-game cinematography to consider is this...

Do I just film what happens in the game? Or do I create my demofile from scratch?

Both have their advantages and disadvantages, and I know videographers who firmly believe in one over the other. For myself, I use a judicious combination of both. Here's an example of how I use both to film a scene...

My team is about to go beat the snot out of Tyrant, so we can rescue Statesman. I'll start demorecording just as we enter his massive chamber, so I don't forget to start recording later (believe me, that is embarrassing). Then my team trounces all of Tyrant's friends, so the map is clear of distractions.

Then we go whale on Tyrant, pounding him into submission and freeing a grateful (and probably humiliated) Statesman.

So far, I've not done anything special... but once we've beaten Tyrant (and being a good videographer, I might make a note of where we were fighting), I'd probably use the **/loc** command to find a good spot to mark a camera location. Then I'd save a copy of the demofile, remove all of the camera commands from the file, and

enter the new camera POS and PYR that I figured out after the battle. This way, no matter how many times I swung around during the combat, and Tyrant might have moved out of my sight or stepped away from me, the new camera POS and PYR I have set up would be a single static shot.

A bit time-consuming, but sometimes quite effective.

Now comes the fun part… turning your demofile into video footage. Since you can't just import a text .cohdemo file into video editing software, you have to convert it to a video file of some sort (.mpeg, .avi, .wmv, whatever). For this, I recommend the FRAPS program. FRAPS is a video capture application that records whatever's on your screen.

My preference is to use Zloth's Demo Launcher (which removes the chat, numbers, heads-up display, and so on) to play back my demofile, and then use FRAPS to record it. Voila! Now my beating on Tyrant is immortalized as a video file, and ready to be dropped into a video editing program like Windows Movie Maker (boo hiss), Sony Vegas Movie Studio, or Adobe Premiere.

Other Useful Stuff

So now you've got the basics on how to record what happens in your game, how to edit it to make it fit your video vision, and how to turn it into a video file. But what do you do now?

Right about now, in the presentation, you'll be learning how to use Sony Vegas Movie Studio (and maybe even winning your own copy!). Here are a few things you'll want to keep in mind when you're creating your video masterpiece:

- Pacing – not too fast, not too slow. Make sure that like any good story, your video moves quick enough to keep your audience's interest, but not so fast that you lose them.

- Quality – if you shoot some footage and later think, "You know, that really didn't come out that well," if you can, shoot it again. This also goes for your sound quality, too.

- Titles – while not always necessary, text cards can be quite effective in a video, especially for stuff like narration; just don't go overboard or your viewers will spend more time reading than they do watching.

- Effects – almost all video programs come with effects tools, but use these like you use spices in cooking… sparingly. You can have too much of a good thing.

- Size/Length – get your minds out of the gutters, first. Regarding size, the higher-quality you save your movie, the bigger the file. Keep that in mind if you're going to make it available for download. Also, if your video is too long, certain sites will not host it (for example, YouTube now has a ten-minute video length limit). Again, consider how long it will take your audience to cache or download it.

Feel free to experiment! But above all else, every single step of the way (from your earlier /demorecord efforts to your final video edits), SAVE, SAVE, SAVE, SAVE, SAVE. Make sure you keep backups of your stuff because you never know when you might need to go back and use an earlier version of your work.

Conclusion

Hopefully you've found this workshop (and this handout) useful in getting you started on your road to machinima creation. Resources abound for helping you bring your vision to life – be they people, websites, or tools. The videographer community for CITY OF HEROES is one of the most open and helpful you'll find – a large number of the threads in the Multi-media section are questions and answers about how to do things, and many of us are constantly experimenting with new things. As new features are discovered, the info is shared.

Practice, practice, practice. You will constantly be learning about how to improve your work, and don't be afraid to try and fail. Every single videographer has done just that, where we look back at our earlier work and think, "Oy…"

And when you're done, don't forget to share! The "Fan Videos" thread in the Multi-media section is constantly being updated with new creations by the community – don't be afraid to show off your own!

Good luck!

Even More Useful Stuff

HOW TO ACCESS THE TRAINING ROOM TEST SERVER

The test server is a wonderful thing – you can mess around on it without worrying about jacking up that carefully created character of yours, or losing a mission that you really want to do with your friends. It's also a great resource when you've got ten friends on six different servers who all want to be in your videos!

To access the test server, you need to download and install a second copy of the game. This second copy will patch up the test files to your PC.

> 1. Locate the file 'CohUpdater.exe'. If the game was installed to the default folder then look for: 'C:\Program Files\City of Heroes'. Do not create a shortcut for the desktop shortcut. To do this, you will have to locate the City of Heroes folder.

> 2. After you locate the City of Heroes folder, you will need to create a new shortcut to the cohupdater.exe file in the City of Heroes folder. To create a shortcut, right click on the file you wish to create a shortcut for. You can then select an option **Create Shortcut** that will then create the shortcut.

> 3. Right click on the new shortcut and open the Properties option.

> 4. In the Target window, add the following flag to the end of the line:

> **[space]-test**

Do NOT add the word [space] to the Target. The example shown is intended to visually represent a single keystroke of the space bar, and should not be interpreted literally. An example of the correct use of the Flag is listed below.

The finished selection should look like this after you add the flag:

> **"C:\Program Files\City of Heroes\CohUpdater.exe" -test**

Below are possible incorrect additions of the flag to add the test client. The selection should NOT look like this:

> **"C:\Program Files\City of Heroes\CohUpdater.exe"-test**
> **"C:\Program Files\City of Heroes\CohUpdater-test.exe"**
> **"C:\Program Files\City of Heroes\CohUpdater.exe-test"**
> **"C:\Program Files\City of Heroes\CohUpdater.exe"[SPACE]-test**

Also, this line in the shortcut can take multiple arguments. That is to say, if you have multiple things you want to tell the game when it launches like -test and -compatiblecursors 1 you just tag them all on the end, like:

> **"C:\Program Files\City of Heroes\CohUpdater.exe" -test -compatiblecursors 1**

> 5. Run the shortcut.

The first time you run the shortcut, you will be asked whether you want to install the test client to the default directory. Choose **Yes** if you want to install it there, or choose **No** if you want to choose a different directory. We strongly recommended that you install the test client to a different location than your normal game client.

HOW TO COPY CHARACTERS TO THE TRAINING ROOM TEST SERVER

After the client installs, you will need to copy a character over to the test server.

https://secure.cityofheroes.com/cgi-bin/coh_transfer.pl

Follow the instructions there to copy your character up onto the test server.

VIDEO CONVERSION RESOURCES

- FRAPS - http://www.fraps.com/
- VIRTUALDUB - http://www.virtualdub.org/

DEMO EDITOR RESOURCES

- BALSHOR'S DEMO EDITOR

  http://home.earthlink.net/~balshor/DemoEditorInstaller.exe
- LEANDRO'S COH POSER - http://cohposer.blogspot.com/
- LEANDRO'S COH SPLASHER - http://www.splasher.com.ar/
- ZLOTH'S DEMO EDITOR

  http://home.kc.rr.com/dasloth/CoH/Demos/DemoEdit.htm
- ZLOTH'S DEMO LAUNCHER

  http://home.kc.rr.com/dasloth/CoH/Demos/DemoLaunch.htm

DEMO RESOURCES

- PARAGONWIKI DEMORECORDING ARTICLES

  http://cityofheroes.wikia.com/wiki/Demo_Recording
- SCUZZBOPPER'S CITY OF HEROES CODEX

  http://home.twcny.rr.com/scuzzbopper/coh/

VIDEO STORAGE RESOURCES

- w00t STUDIOS - http://www.w00tstudios.net/
- YOUTUBE - http://www.youtube.com/

FORUM RESOURCES

- THE MULTI-MEDIA~CITY OF HEROES/VILLAINS SECTION

  http://boards.cityofheroes.com/postlist.php?Cat=0&Board=Media
- THE FAN VIDEOS THREAD

  http://boards.cityofheroes.com/showflat.php?Cat=0&Number=10047216&an=0&page=0 - Post10047216
- PLAYER GUIDES TO DEMO EDITING, CUSTOMIZATION, ETC

  http://boards.cityofheroes.com/showflat.php?Number=9943931 - Post9943931