10.8.1 Render - Workflows - Rendering Animations

Workflows

- Rendering Animations
  - Workflow
  - Frame Sequence Workflow
  - Hints
- Command Line
  - Platforms
  - Examples
- Render Baking
  - Options
  - Workflow
- Multi-View Render
  - Introduction
  - Views Setup
  - Camera
  - Viewport
  - Stereo 3D Display
  - OpenGL Preview
  - Rendering and Image Editor
  - Image Formats
  - Final Considerations
  - Window Stereo 3D Display
  - Stereo 3D Camera
  - Viewport Stereo 3D
  - Multi-View and Stereo 3D Image I/O
  - Image Editor
  - Compositor

Rendering Animations

While rendering stills will allow you to view and save the image from the render buffer when it's complete, animations are a series of images, or frames, and are automatically saved directly out to disk after being

rendered.

After rendering the frames, you may need to edit the clips, or first use the Compositor to do green-screen masking, matting, color correction, DOF, and so on to the images. That result is then fed to the Sequencer where the strips are cut and mixed and a final overlay is done.

Finally you can render out from the Sequencer and compress the frames into a playable movie clip.

Workflow

Generally, you do a lot of intermediate renders of different frames in your animation to check for timing, lighting, placement, materials, and so on. At some point, you are ready to make a final render of the complete animation for publication.

There are two approaches you can use when making a movie, or animation, with or without sound. The approach you should use depends on the amount of CPU time you will need to render the movie. You can render a "typical" frame at the desired resolution, and then multiply by the number of frames that will ultimately go into the movie, to arrive at an total render time.

If the total render time is an hour or more, you want to use the "Frame Sequence" approach. For example, if you are rendering a one-minute video clip for film, there will be (60 seconds per minute) * (24 frames per second) or 1440 frames per minute. If each frame takes 30 seconds to render, then you will be able to render two frames per minute, or need 720 minutes (12 hours) of render time.

Rendering takes all available CPU time; you should render overnight, when the computer is not needed, or set Blender to a low priority while rendering, and work on other things (be careful with the RAM space!).

The **Direct Approach** - highly **not** recommended and not a standard practice - is where you set your output format to an AVI or MOV format, and click ANIM to render your scene directly out to a movie file. Blender creates one file that holds all the frames of your animation. You can then use Blender's VSE to add an audio track to the animation and render out to an MPEG format to complete your movie.

The **Frame Sequence** is a much more stable approach, where you set your output format to a still format (such as JPG, PNG or MultiLayer), and click ANIM to render your scene out to a set of images, where each image is the frame in the sequence.

Blender creates a file for each frame of the animation. You can then use Blender's compositor to perform any frame manipulation (post processing). You can then use Blender's VSE to load that final image sequence, add an audio track to the animation, and render out to an MPEG format to complete your movie. The Frame Sequence approach is a little more complicated and takes more disk space, but gives you more flexibility.

Here are some guidelines to help you choose an approach.

Direct Approach

- short segments with total render time < 1 hour
- stable power supply
- computer not needed for other uses

Frame Sequence Approach

- total render time > 1 hour

- **post-production work needed**
    - Color/lighting adjustment
    - Green screen / matte replacement
    - Layering/compositing
    - Multiple formats and sizes of ultimate product
- intermediate frames/adjustments needed for compression/codec

- precise timing (e.g. lip-sync to audio track) needed in parts

- may need to interrupt rendering to use the computer, and want to be able to resume rendering where you left off.

Frame Sequence Workflow

- First prepare your animation.
- In the *Dimensions* panel, choose the render size, Pixel Aspect Ratio, and the Range of Frames to use, as well as the frame rate, which should already be set.
- In the Output panel set up your animation to be rendered out as images, generally using a format that does not compromise any quality (I prefer PNG or MultiLayer because of their loss-less nature).
- Choose the output path and file type in the Output panel as well, for example `//render/my-anim-.`
- Confirm the range of your animation frame Start and End.
- Save your .blend file.
- Press the big *Animation* button. Do a long task [like sleeping, playing a video game, or cleaning your driveway] while you wait for your computer to finish rendering the frames.
- Once the animation is finished, use your OS file explorer to navigate into the output folder ("`render` in this example). You will see lots of images (.png or .exr, etc... depending on the format you chose to render) that have a sequence number attached to them ranging from 0000 to a max of 9999. These are your single frames.
- In Blender, now go into the *video sequence editor*.
- Choose *Add Image* from the add menu. Select all the frames from your output folder that you want to include in your animation (Press A to Select All easily). They will be added as a strip to the sequence editor.
- Now you can edit the strip and add effects or simply leave it like it is. You can add other strips, like an audio strip.
- Scrub through the animation, checking that you have included all the frames.
- In the Scene Render buttons, in the Post Processing panel, activate *Sequencer*.
- In the Format panel, choose the container and codec you want (e.g. MPEG H.264) and configure it. The video codecs are described on the previous page: *Output Options*.
- Click the ANIMATION render button and Blender will render out the sequence editor output into your movie.

Why go through all this hassle? Well, first of all, if you render out single frames you can stop the render at any time by pressing `Esc` in the render window. You will not lose the frames you have already rendered, since they have been written out to individual files. You can always adjust the range you want to continue from where you left off.

You can edit the frames afterwards and post-process them. You can add neat effects in the sequence editor. You

can render the same sequence into different resolutions (640x480, 320x240, etc) and use different codecs (to get different file sizes and quality) with almost no effort whatsoever.

Options

Post Processing Panel

    Sequencer

        Renders the output of the sequence editor, instead of the view from the 3D scene's active camera. If the sequence contains scene strips, these will also be rendered as part of the pipeline. If Do Composite is also enabled, the Scene strip will be the output of the Compositor.

    Compositing

        Renders the output from the Compositing noodle, and then pumps all images through the Composite node map, displaying the image fed to the Composite Output node.

Hints

You accidentally turned off you're PC right in the middle of rendering my movie!

Unless your animation renders in a few minutes, it's best to render the animation as separate image files. Instead of rendering directly to a compressed movie file, use a loss-less format (PNG for example).

This allows you an easy recovery if there is a problem and you have to re-start the rendering, since the frames you have already rendered will still be in the output directory.

Just disable the *Overwrite* option to start rendering where you left off.

You can then make a movie out of the separate frames with Blender's sequence editor or using 3rd party encoding software.

Animation Preview

It can be useful to render a subset of the animated sequence, since only part of an animation may have an error.

Using an image format for output, you can use the *Frame Step* option to render every *N'th* frame. Then disable *Overwrite* and re-render with *Frame Step* set to 1.