

“The Battle of Fei Tuvinien”

Making of

By Frédéric Louguet

Hi fellow readers,

This document relates the Making Of “The Battle of Fei Tuvinien”, a 3D project created with Vue 5 Infinite (e-on software) and Poser 6 (e-frontier), with the help of some great utilities (P3dO Explorer, ACDSee, SkinVue).

This particular scene is rather extreme in terms of processing power and memory usage. That was the goal : to push Vue to the limits and see how complex things could go. I hope that this document will be useful to anyone using the combo Vue / Poser. The creation of this scene has been a small adventure that I am happy to share with you.

Most of the characters, creatures, clothing and props used in this scene are commercial products available at Daz 3D.

The theme

I wanted to recreate a scene from a battle between the Forces of Light and the Forces of Darkness, in the pure heroic fantasy tradition. Obviously, to have a battle, you need characters, and more than one or two. First problem : handling more than a few realistic Poser characters in the same scene is quite difficult. But as you can see, this one is really crowded.



A simple example : in the first version of the scene, I had forgotten to move a scabbard, which was falling through the elf warrior's robe !



Before : oops !



After : better !

If I had not kept the PZ3, only image editing could have cured the problem, and it would not have been very satisfying. When you work with complex scenes and numerous characters, it is very easy to overlook small details, so better safe than sorry : keep the PZ3 somewhere ☺

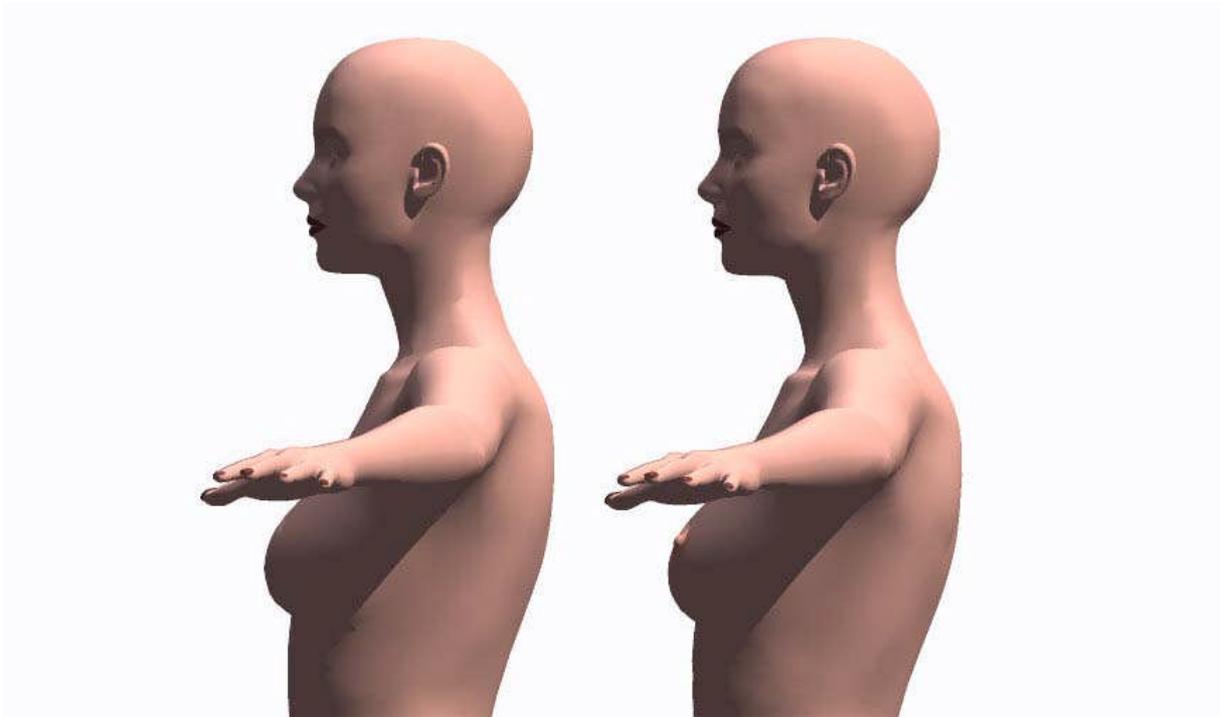
Also, materials don't always import well in Vue. The highlights must often be corrected, and some materials are too reflective : you don't want a shiny superchrome material on what is supposed to be a rusted helmet... So there is a bit of tweaking here too.

Reducing the memory footprint

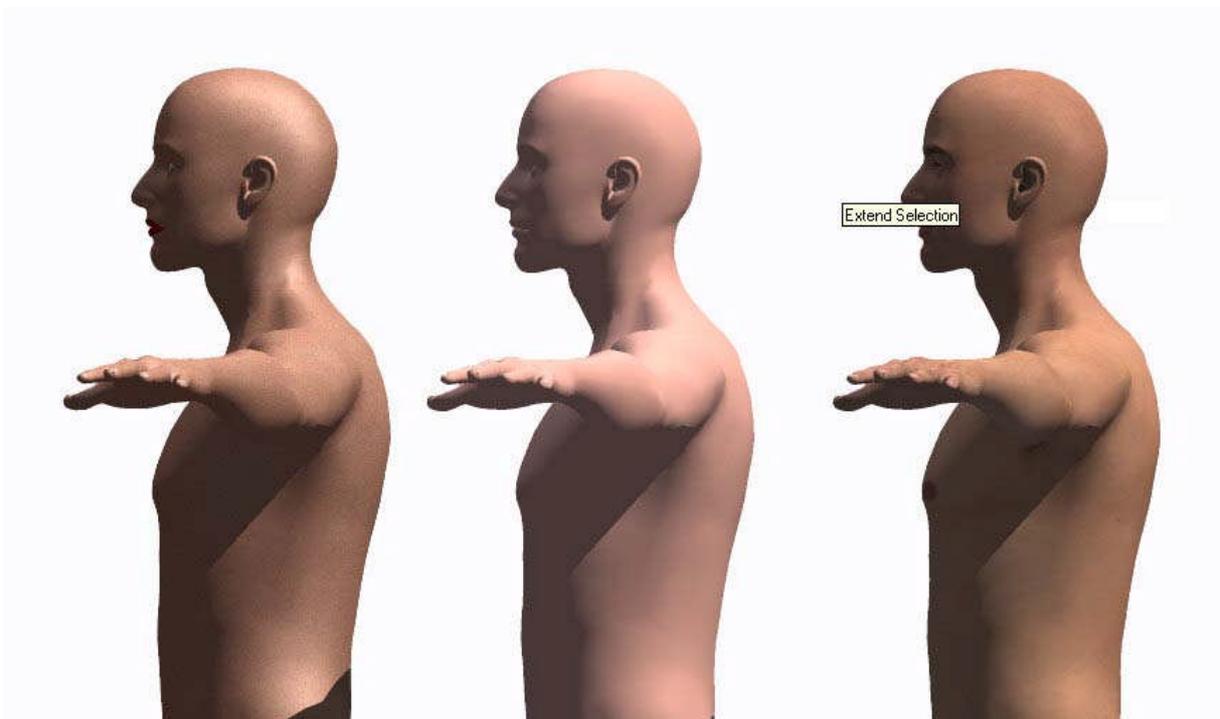
Daz M3 and V3 have a very high polycount (about 143000 polygons), and high resolution textures eat up memory at lightspeed. For the 5 human / elf characters in the scene, I did essentially two things :

- I replaced the M3 and V3 figures by their reduced-resolution (RR) versions. These versions have roughly one third the vertices/polygons count of M3 and V3 (about 46500 polygons), and for non-closeup scenes the difference in visual quality is minimal.
- I used the SkinVue 2 procedural shader instead of high resolution texture maps. There are no skin texture maps on any human character in the scene.

Next page you can see in the first image a comparison between V3 and V3 RR, and in the second image the procedural shader SkinVue 2 compared to no texture and to a highres map.



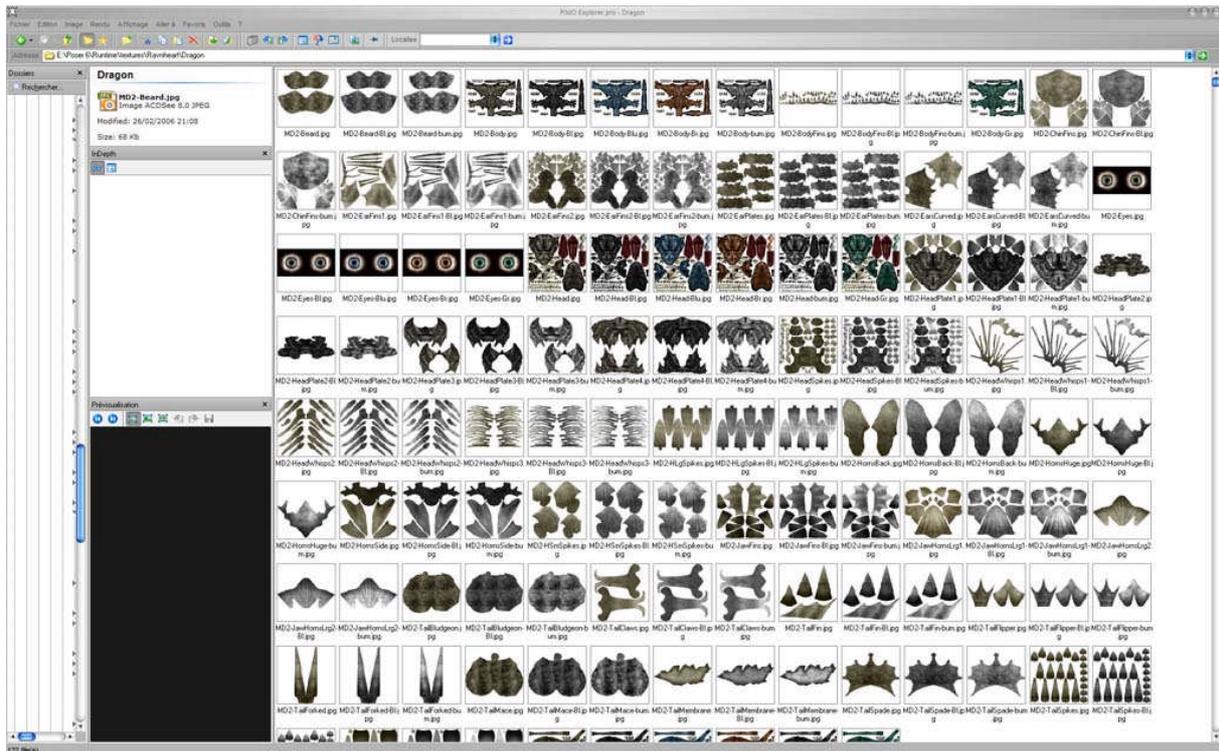
On the left, V3 Reduced Resolution (46 K poly). On the right, V3 SAE (143 K poly).



On the left, M3 RR with SkinVue 2 procedural shader at default settings. In the middle, imported with no textures. On the right, imported with a texture map from Poser.

For the monsters and the various clothes and props, I resized the textures. They generally come in 3000 x 2000 resolution, sometimes even more. This is good for close-ups, but if you don't need to get close you can safely reduce their size in most cases without any visible loss in quality.

So the first thing to do is to find the textures path. If I had to search manually for all the textures used in this scene and resize them all one by one, I would have gone mad ! So I used two fantastic utilities : P3dO Explorer and ACDSee 8, opened simultaneously. First, P3dO allows me to find instantly which textures are used by which character / prop, and displays the directory where they are stored.

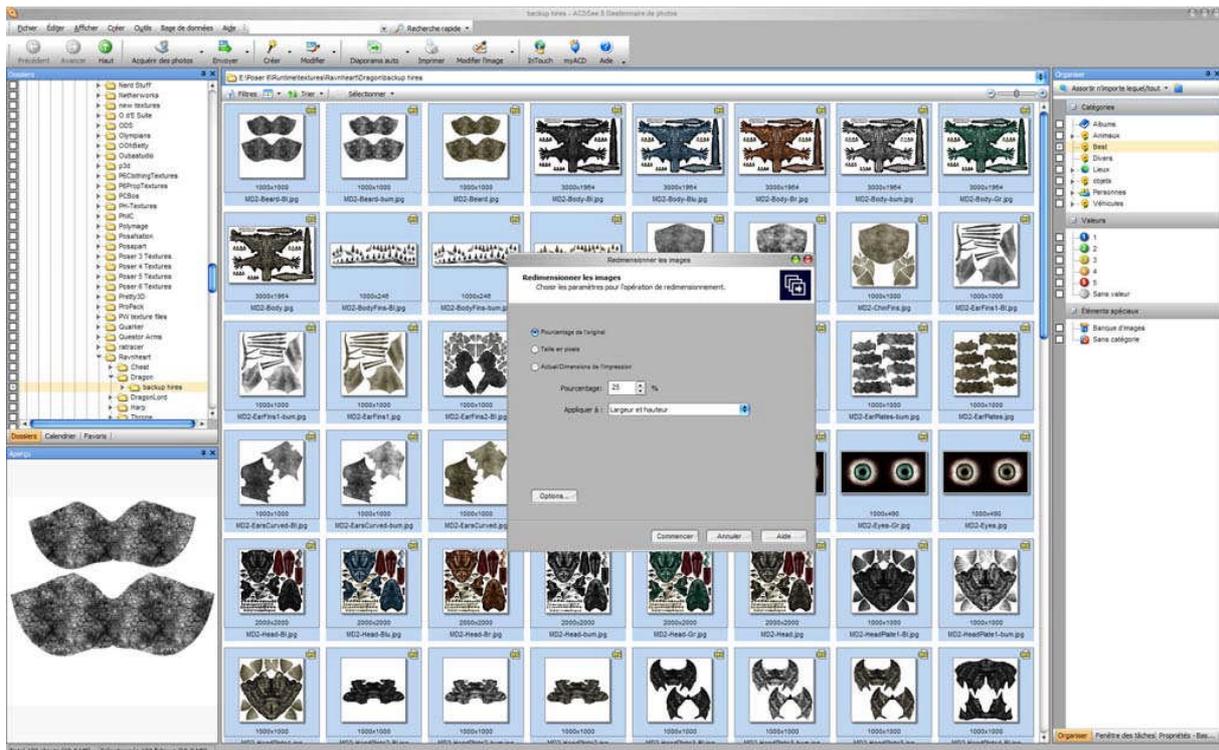


P3dO Explorer instantly finds the textures directory.

Then I switch to ACDSee 8.0. I first create a 'backup hires' sub-directory and save the original hires textures in it. Then, I resize a copy of all the original textures in their original directory. ACDSee lets you resize as many images as you want in a single command !

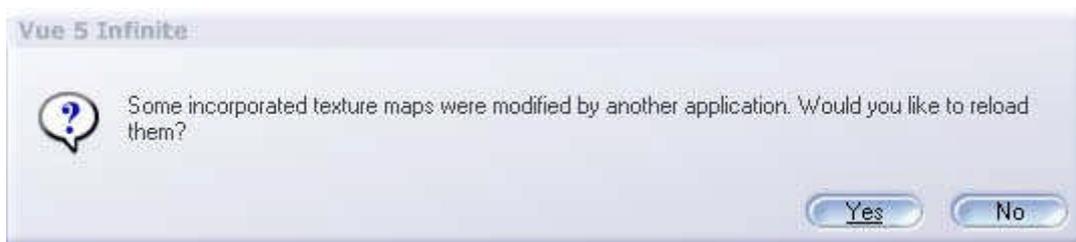
With this method, Poser will automatically use the reduced resolution textures when applying the MATS, no need to point to another directory. If I need the hires textures later, I will simply copy them back in their original directory from the 'backup hires' where they are safely stored.

In this particular scene, I resized the textures at one fourth of their original size (25%) in X and Y directions, except one or two where I needed a better resolution (for the black dragon for example I resized at 40%). I did this for every character, piece of clothing and prop which came with high-resolution textures (but thanks to ACDSee it was relatively fast). Of course I created a different 'backup hires' directory for every item concerned.



ACDSee 8 resizes all the textures in one stroke.

One interesting thing : when you resize some textures in ACDSee while Vue Infinite is loaded with the scene, Vue informs you that textures used in the scene have changed, and ask if you want to reload them. A good way to interactively reduce the texture resolution : you can verify if the loss of detail is negligible by doing quick renders in Vue after each reduction.



Vue is aware when a texture map used by a VOB is modified by another application.

Note that you can also try to decimate imported meshes to reduce their geometric complexity and memory footprint. Unfortunately, decimation doesn't work well with complex models : artefacts (like missing polygons) are very visible, especially on clothing details. This is the reason why no model has been decimated in 'The Battle of Fei Tuvinién' scene; their full geometric complexity has been preserved.



On the left, the original model with 113000 polygons. On the right, a slightly (15%) decimated version (96000 polygons). Still, the missing polygons are clearly visible on the bracers (the white patches).

Even with the help of P3dO and ACDSsee, this preparation work was quite long, patience was certainly required. But hey, this was an extreme scene, it had to be done.

Measuring memory usage

To evaluate each character / creature memory usage, all you need to do is to open the Windows task manager just after loading Vue and see how much physical memory is available (let's call this value APM1 for Available Physical Memory 1). Then load the PZ3 or VOB character, wait for the CPU to drop at 0% activity, and look again at the available physical memory (APM2). The memory consumption (MC) of the character is $APM2 - APM1$. Believe me, looking at this number is an eye-opener 😊

Mémoire physique (Ko)	
Total	2882176
Disponible	1939040
Cache système	1833052

Mémoire physique (Ko)	
Total	2882176
Disponible	1879088
Cache système	1833704

Example : this model uses almost 60 MB of memory (look on the second line, the screenshot was made with the french version of Windows XP).

Note that you cannot have a good idea of the memory consumption by simply looking at the size of the PZ3 or VOB files on the hard disk, because textures are compressed in a file and do not take a lot of space. However, they are not compressed when loaded in memory : a 3000 x 2000 compressed JPG file is maybe be a 2,5 MB file, but it eats 17 MB of memory ! When you have ten of these textures applied to a character and his clothing, things get tense 😊

By using the various optimization and monitoring techniques described above, I managed to reduce the memory usage of some fully clothed characters by a factor of 3 or 4. Without them, I would not have been able to put more than 4 or 5 realistic characters in the scene.



On the left, M3 with high-res texture maps, hi-res clothing textures, PZ3 format : 197 MB in memory. On the right, M3 RR with SkinVue 2 procedural shader only, low-res texmaps for the clothes, VOB format : 52 MB in memory.

A closer look reveals the differences in quality, but if you do not get too close to the character they should not be too much of a problem.

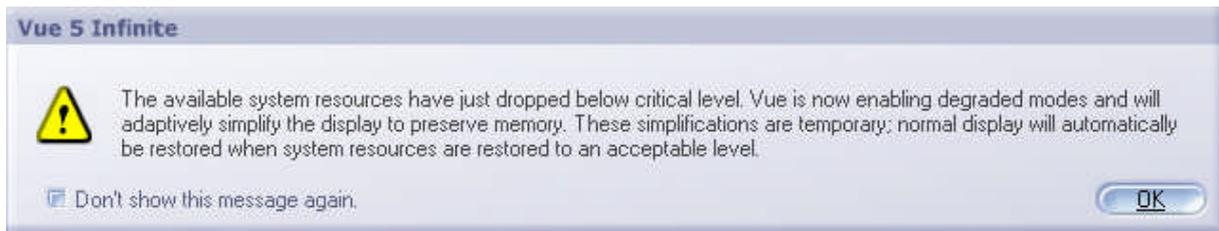
Environment and skeletons

The environment of the scene is simple compared to the characters : the sunlight, a small mountain in the distance, a stone wall on the right of the scene, and an EcoSystem for the field vegetation. It doesn't seem so, but this vegetation takes a long time to render.

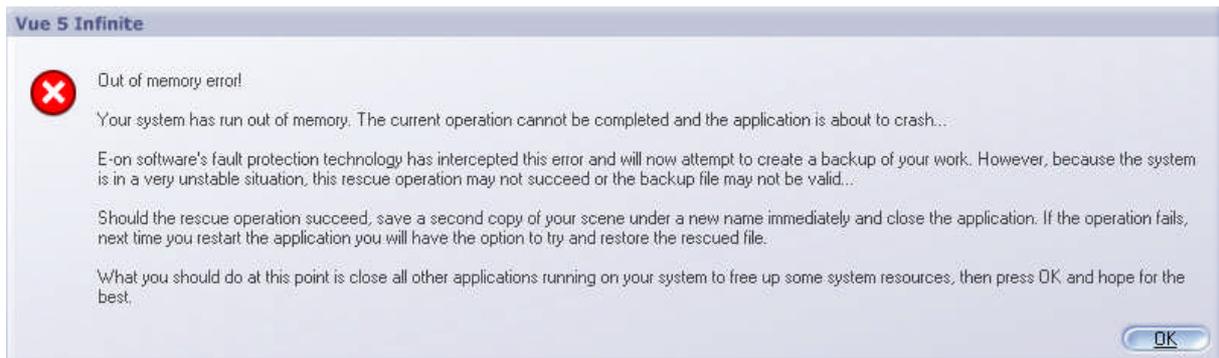
The skeleton army is also an EcoSystem, based on the original Poser skeleton (not the hires Daz one, to save a few more megabytes). To avoid the same exact look for every skeleton and get a more natural feeling, the EcoSystem is based on three different poses for the skeleton. If you add slight variations in size, orientation and repartition, the bony crowd looks reasonably good. You don't have the impression that you are looking at Nth times the same skeleton.

Fighting slowness

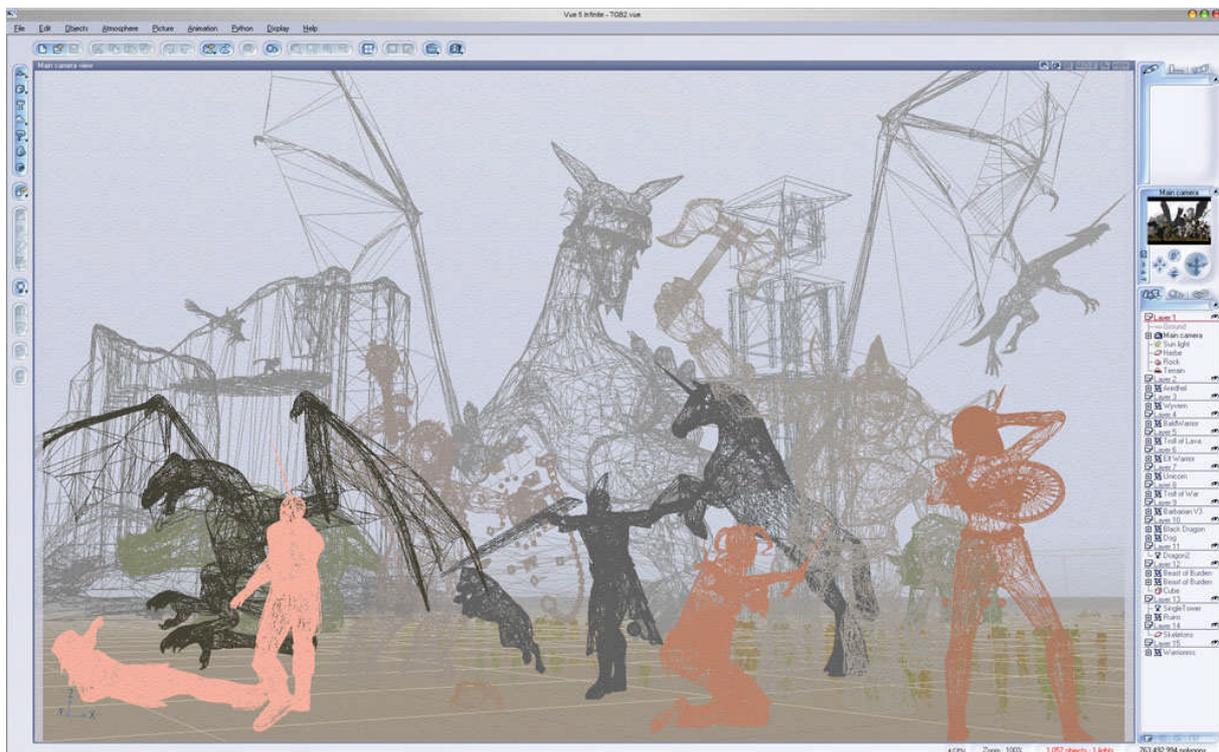
To be honest, even on a powerful computer, manipulating such an extreme scene is really a pain ! You have to be extra-careful each time you move the mouse, because your actions are executed several seconds after the click. That is, if you get impatient, you can easily duplicate a figure instead of putting it in a new layer, or move one figure instead of another. The horror. Using the undo feature just *once* brings you the dreaded 'limited resources' dialog box.

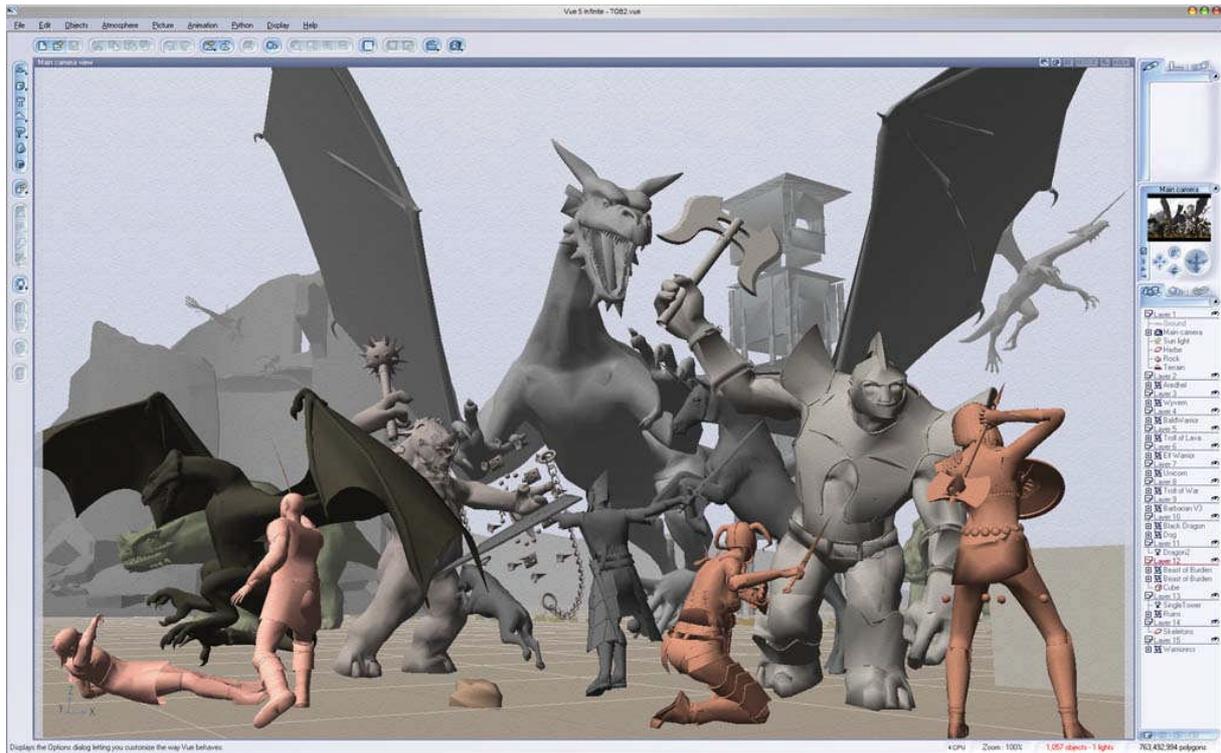


At this point, you'd better save your 240 MB scene immediately. If not, the 'Out of memory box' will soon pay your computer a visit and teach you how to behave properly 😊



So the big secret is : be patient. Use layers to hide unneeded characters. Wait until a command is executed before clicking everywhere. Save often. You can have a slightly better response time when you choose a wireframe display instead of the OpenGL one, which consumes a lot of memory. Personally I don't do this, I much prefer working with OGL.





Using the wireframe mode (see previous page) instead of OpenGL (above) can save you 200 to 300 MB of memory if you work in high definition. But it's kind of frustrating...

Network rendering ? Nope.

Network rendering in Vue works quite well (at least for me). However, this scene simply doesn't want to be rendered on a network. It wants you to suffer ☺

Every time I tried a HyperVue render, Vue crashed. The extreme memory consumption is probably responsible. I used every little trick and then some (the description of those would be utterly technical and unbelievably boring), but to no avail.

Of course, that doesn't mean I could not use the renderfarm efficiently. While the final image was being rendered on a lonesome - but fearless - PC, I was able to begin a few animation sequences with the others (see 'The animation'). But before that, let's talk about the render settings.

The rendering parameters

The first thing I did is reduce the Shadow Density from 100 to 80%, so the shadows would not be too dark.

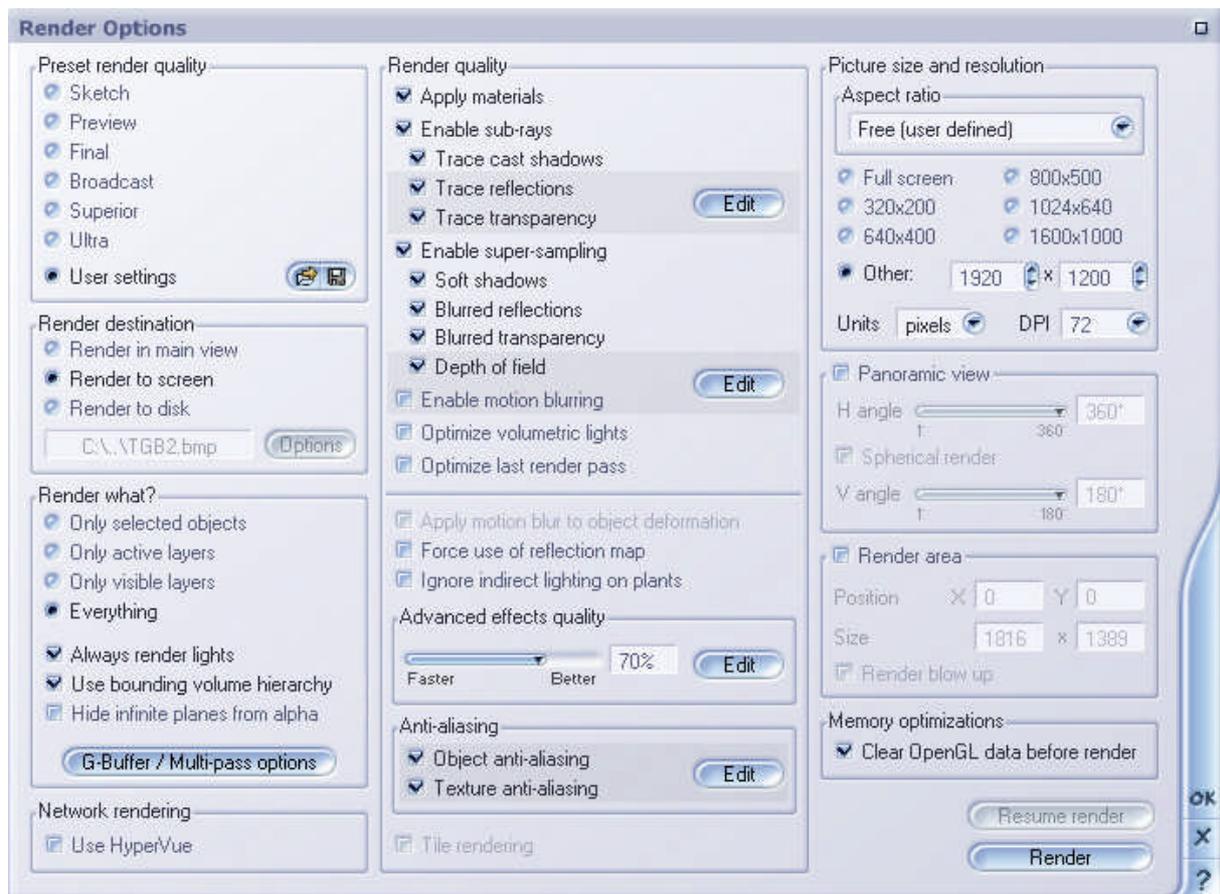


Shadows at 100%

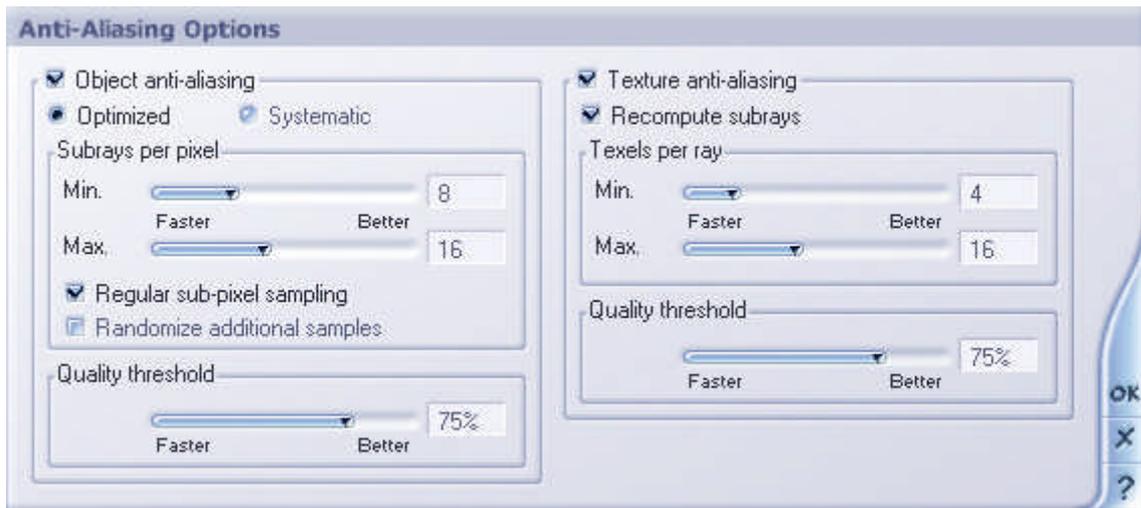


Shadows at 80%

I rendered many small images and found that GR (*Global Radiosity*) gives a small but noticeably better quality on this scene than GI (*Global Illumination*). To avoid unbelievably long render times (courtesy of Ultra mode, Inc.), I choose instead to define specific user settings. Here they are :



And here are the antialiasing settings :



Using less antialiasing can lead to grainy images in shadows. If the Sun light softness is set to 10°, you get the image on the left below (final mode). By reducing the softness to 5° there is much less grain (on the right) because the surface covered by the traced rays is smaller.

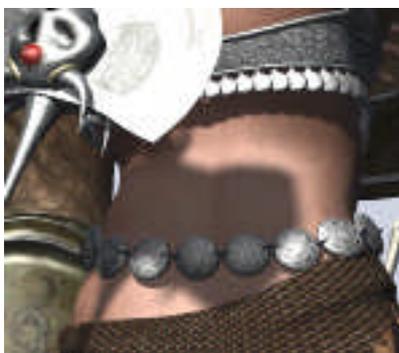


Sun softness 10° (final)



Sun softness 5° (final – 5 seconds)

With the user settings described above, the quality is much better than in final mode, and the render time is not nearly as long as with the Ultra settings :

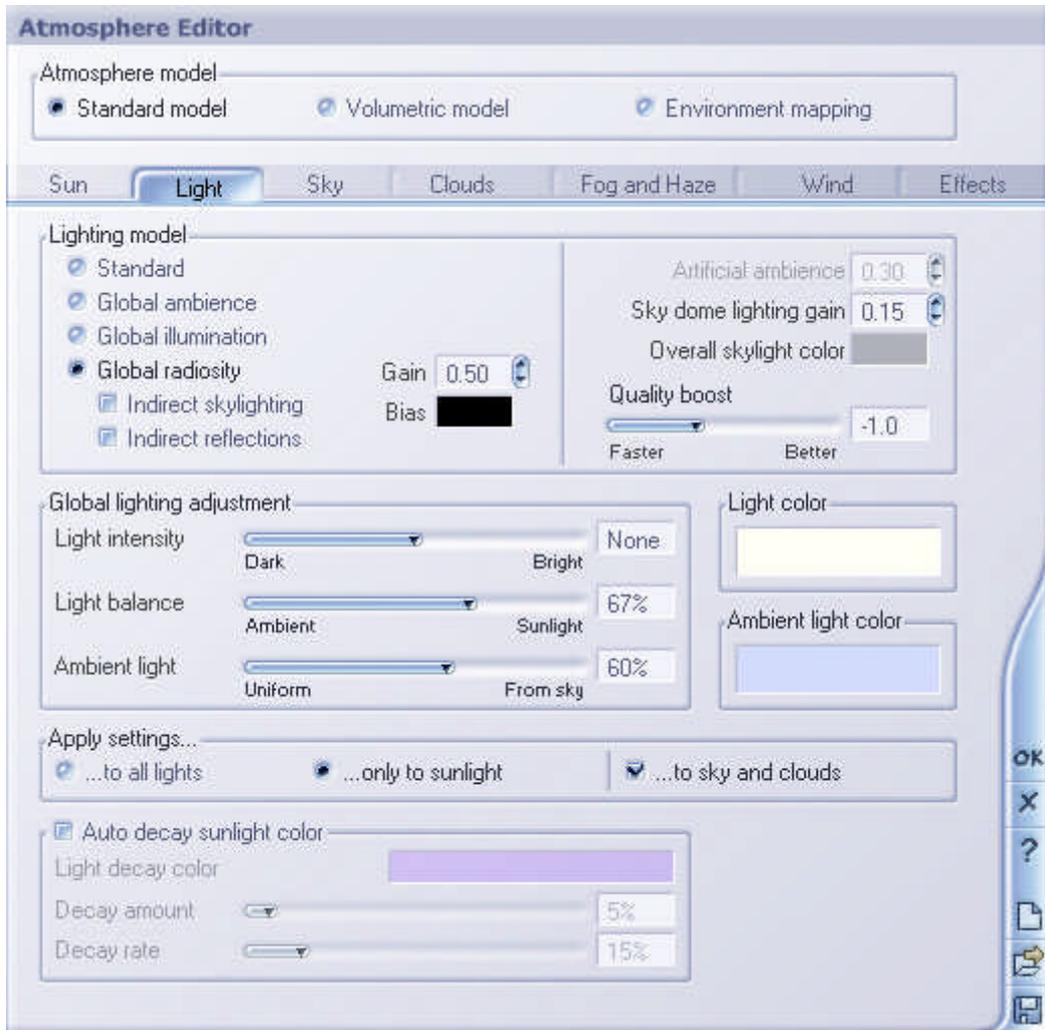


Softness 5° user settings - 29 s



Softness 5° ultra mode – 46 s

As I could not render the main image on the network, I chose to reduce the render time by using the Quality Boost option at a -1.0 value. In exterior scenes, the quality difference between -1.0 and 0.0 is small, but QB-1.0 is easily 2/3 times faster. However, I do not recommend using QB-1.0 for interior scenes, because the loss in quality is much more visible there. Here are the Atmosphere settings :



The 1920 x 1200 final image finally took 7.5 hours to render on the 4 cores (dual dual core) Opteron. That is roughly equivalent to 30 hours on a single Pentium 4 - 3.4 GHz with hyperthreading.

How to evaluate small rendering changes ?

Often, you want to render a series of images in low definition to compare the results for different settings. But this is not easy to do with subtle changes : for example, between Quality Boost at -1.0 and -0.75, what is the difference ? Even looking at the two images opened at the same time in Photoshop is not always clear.

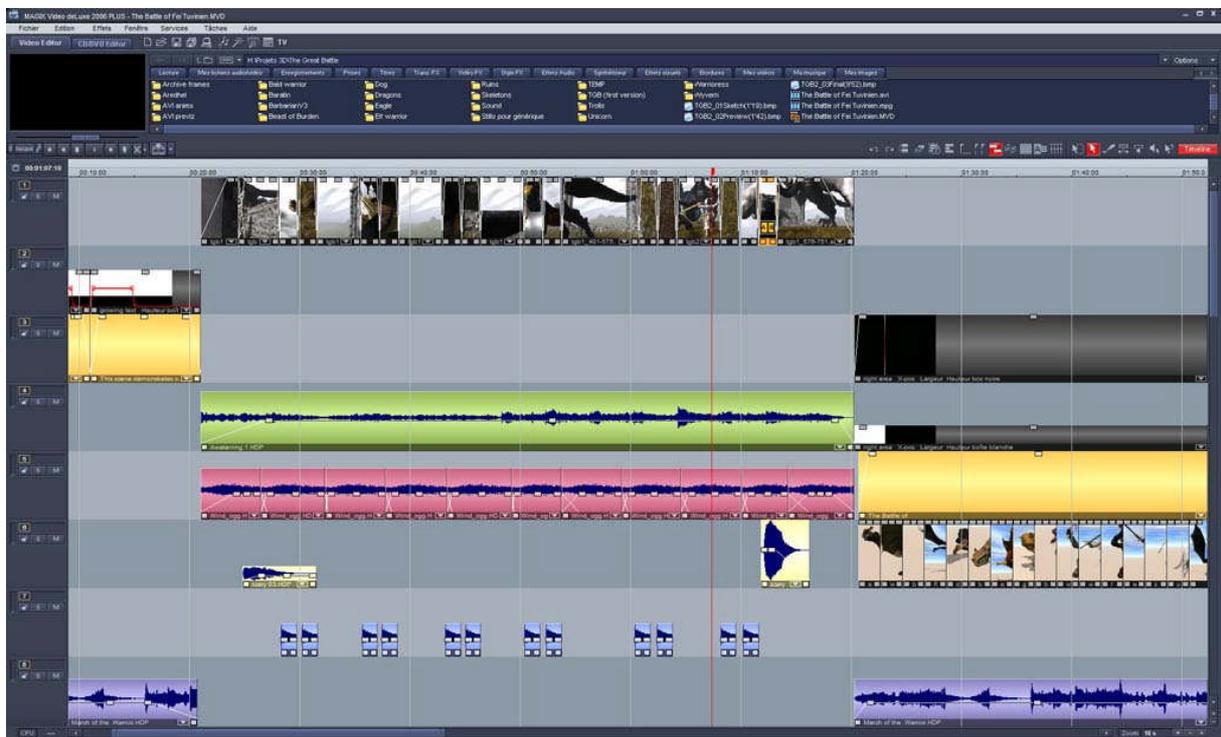
That's where ACDSee 8.0 shines. It allows you to display full screen images in the blink of an eye, so you just have to scroll through your list of images with the mouse wheel. To compare two images, just move the wheel forward and backward to alternate between the two : your

eyes will automatically do the rest by detecting the smallest changes in lighting, aliasing, color, etc. So you will quickly see which areas of the image are affected by your changes.

The animation

The downloadable animation was rendered in broadcast mode (good for fast camera moves and not too costly), in 720 x 405 and 16:9 resolution, but with a standard lighting model, not radiosity like the final still image. It is a sort of 3D flyby in the scene. Sorry, the characters don't move, but animating the whole bunch is simply beyond my patience and the memory limitation of a 32 bit application ☺.

As the characters are static, I used video editing to alternate sequences and give a more dynamic feeling. I also tried to enhanced the mood with an appropriate soundtrack.



I used Magix Video Deluxe 2006 Plus to mix everything and produce an XviD AVI.

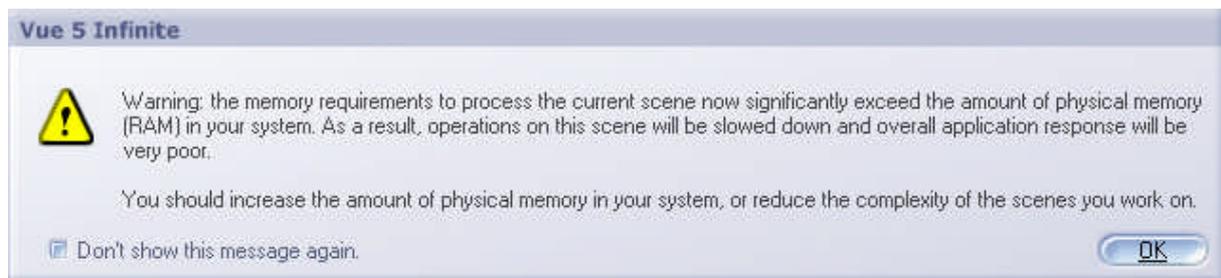
The anim is really a mix of two big sequences (750 and 500 frames) and of thirteen simple 2-seconds, 50-frames shots showing a glimpse of the main characters. I created a new scene for each of these short shots, keeping only the characters which were seen and thus considerably reducing the size of each file. These thirteen scenes were much smaller than the one with all the characters, so I was able to render them on the network, which is a tremendous boost to productivity.

The two big sequences which compose the body of the flyby anim are based on the big scene, so HyperVue network rendering could not work because of crashes. Too bad. Instead I had to improvise some sort of manual network rendering. Not very convenient, but it works.

I calculated the relative speeds of each computer on the network when rendering this particular scene (based on a few sample frames), and manually allocated a corresponding

percentage of the total frames to render to each PC, based on its previously measured performance. To do this I had to launch the main Vue application on each PC. PCs with 1 GB suffered the most, as 2 GB is needed for the whole scene to fit in physical memory. Hard disks were hard at work on these machines as disk swapping went crazy, but in the end everything went well, and no PC crashed. It must be noted that e-on has notably improved memory management in the most recent versions of Vue.

After a rendering, this message appeared on all the PCs equipped with 1 GB RAM :



What about the name ?

If you are wondering about the name 'Fei Tuvinien', let's say it is not a completely random name. It has a nice elvish sound, heh ? But there is something else, and I am sure many of you will figure it out in no time ☺ (it's not difficult).

Conclusion

Well that's it. I hope you enjoyed the image and animation, and the different techniques discussed. The point is : it is possible to create complex scenes with many Poser characters in Vue Infinite, providing you take some time to optimize the imported figures. People who really want to make big scenes will be pleased to know that Vue 6 will be available in a 64 bit version, which will remove many of these limitations. I personally can't wait to have it.

Of course, optimizations will always be useful, even with 64 bit ultra-powerful computers : thanks to them you will be able to pack even more characters into your scenes. The difference is, you will be limited by the amount of RAM installed in your machine, not by the address space of the application. Believe me, it is much less frustrating.

Thanks again for your attention, and be creative !

Frédéric Louguet

<http://renderfred.free.fr/>

March 02, 2006

E-on software : www.e-onsoftware.com

E-frontier : www.e-frontier.com

Daz 3d : www.daz3d.com

ACDSee : www.acdsystems.com

SkinVue : www.skinvue.net

P3dO Explorer : www.senosoft.com