Subject: Re: TT TEASER! - water rendering and widescreen fix
Posted by LR01 on Fri, 17 Apr 2009 15:06:21 GMT
View Forum Message <> Reply to Message

Toggle Spoiler
Saberhawk wrote on Fri, 17 April 2009 15:09LR01 wrote on Fri, 17 April 2009 09:03Saberhawk wrote on Thu, 16 April 2009 23:55LR01 wrote on Thu, 16 April 2009 03:39Nice water there, now I wonder, how does it work?
I mean, how do you create water now? (don't think it's done through renx now is it?)
Ore like the unreal engine.

You first make "stock" water (aka water that'll appear if shaders aren't supported). On that water, you add a few (new) mapper arguments that tell it to be disabled if a certain shader is support. Then you create the new water plane with regular unwrapping and add a few other mapper arguments (like "Shader=<shader name>") to actually set the shader and do a bit of other needed magic to turn it off completely if the shader isn't supported. Then you hook up some other entries in other files and you are good to go.

ah, then something more (say if I to far away from the topic)
when I would make a plane then, I could use the shaders to make "water" of it, but also to make a "mirror" ore "exhaust famues" of it. I assume that happens on the material name assigned to that plane?

For backwards compatibility purposes, it likely could (the shader would need a "UsesMaterialNameAttachment" or "UsesTextureNameAttachement" flag set to true on it.) The proper way of attaching shaders in scripts 4.0, however, is to set the "Shader" mapper argument for the material. The mapper arguments is where you put things like UPerSec, and VPerSec. Attaching shaders this way allows them to bypass the name lookup entirely and put far less stress on the CPU.

cool, makes things easier to do.
now something else, it reflects everything?
(I mean, rockets and explosion and such to?)

(ueh, the meaning of the spoiler was only the hide the big quote, seems like I don't know how to use it properly)