

## WHAT YOU NEED:

- knowledge of photoshop (color range selection, alpha channels)

## 1. OVERLAYS VS. PROP\_DYNAMIC VS. LIGHTMAPS



dynamic shadows

de\_mediterrano

Maybe some of you know that prop\_dynamics cast a really crisp and detailed shadow on any brush geometry in the source engine. This technique is ment for use on dynamic models(prop\_dynamic or any prop\_physics) or entities ONLY. Using these shadows on „static“ object which neither change their position not change their shape shouldn't become a dynamic model because the rendering speed will slow down too much as soon as you fill your level with these entities. One reason for this ist that dynamic shadows are always rerendered to a texture. This rerendering for every frame costs many frames per second.



lightmaps

de\_static beta

So what about Lightmaps? Lightmaps don't have a that high resolution to reach the same fine level of detail in your shadows. Due to the fact that those lightmap shadows are rendered by „looking on the physics mesh“ of a model, the fine leaf shapes of your textures would get lost too.



overlays

de\_corse

Using a static overlay is maybe the best way. It is not 100% accurate but it has the same level of detail compared to dynamic shadows and it has nearly the same rendering speed compared to lightmaps. The only thing you have to invest is time on creating these shadow textures and placing them in your level but as I think performance bonus and more eyecandy make it worth the work.

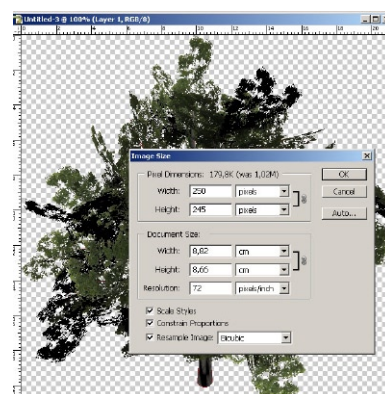
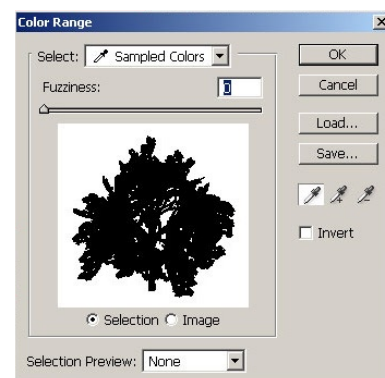


## 2. Creating the shadow texture



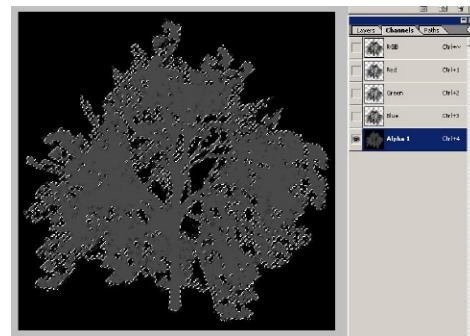
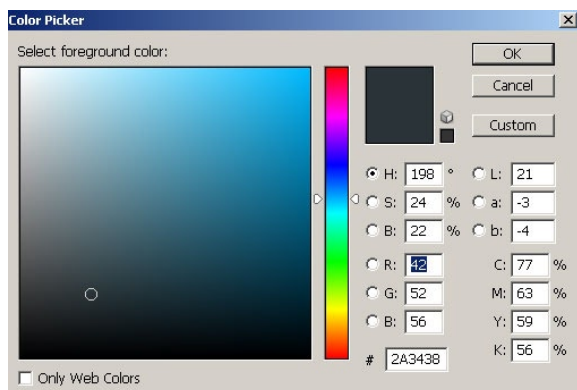
Open your tree model in HLMV and choose a background that has a high color contrast to your tree (red or pink is usually a good choice) and pose your camera in a direction as „if you were the sun“ (looking from above in an 60°-80° pitch angle). Now make a screenshot by pressing the „prnt scrn“ key on your keyboard. Start photoshop and paste the image of your clipboard into a new file. Crop all your unneeded programs off the image (Press c and then select the region you need. Then double click into the selection.).

To get extract the shadow shape of the tree select -> color range and pick the red color. Now press delete on your keyboard to get rid of this red background. To make the texture fit into a power of 2 texture copy the current image (should be the tree on a transparent background; no red left) to a new image (I prefer to use 256x256 maps for such shadows.). Make sure that you leave some empty space at your texture borders to protect you overlay from mipmapping „errors“ (maybe some shadows would appear on the opposite side due to reducing the texture resolution in mipmaps).



To get a mask of the tree hold ctrl and click on your tree layer. Create a new alphachanel and fill it with a bright grey. Switch back to your color layer and fill it with a dark blue (ambient blue).

Just save this image as a 32bit tga file and convert it to a vtf by using vtex.exe.



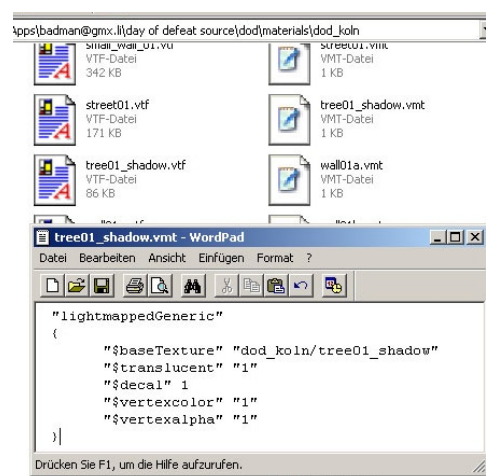


## 3. Implementing the overlay texture

Writing the VMT isn't a that big deal. Just copy this and replace the textures name. (the \$decals are for preventing z-fights of the overlay with the polygons of your world brushes (i forgot to do this in my corse fake shadows ;-)).

```
"lightmappedGeneric"  
{  
    "$baseTexture" "dod_koln/tree01_shadow"  
    "$translucent" "1"  
    "$decals" 1  
    "$vertexcolor" "1"  
    "$vertexalpha" "1"  
}
```

In the hammer editor you should open your map file and make sure that all your trees are prop\_static. In the properties you should disable their shadows because we will place them manually. Pick your new material and place it on the brushes the tree would cast shadows on. Of course a 256<sup>2</sup> texture is much too small so you have to resize the overlay using the default modifier cubes of hammer (those white quadrangles at the borders of your overlay or brushes). You may also use the rotation circles to make the shadows fit to your sunlight angles.







## 4. Proof of the performance bonus

First screenshot shows dynamic shadows. The second one shows fake shadows.

Dynamics



Overlays

