

PAK Files and Directory Folders

By Marc 'Delete_Me' Henry

Originally appeared in VERC Collective in Mar 4, 2002 and archived in TWHL since August 1, 2008.

What is it?

The .pak files are places that store sounds, models, maps, etc. in Half-Life and her mods. You can make your own using [PakScape](#) or even Wally (Both can be downloaded in the Utilities Section). Sometimes, people wish to change certain sounds or models in Half-Life and they modify the pak0.pak, which isn't recommended. In this tutorial, I will show you the several ways to insert a custom sound that replaces, for example, the scientist's 'yes.wav' sound.

Pak Files - When You Are Distributing A Map Pack

The .pak files are file holders that can overlap each other. They are coded so that if there are any files that have the same name and location in two .pak files, the file in the .pak with the highest number will override the one in the lower number.

```
| | pak0.pak → — pak1.pak → — pak2.pak
```

In this example, pak1.pak will override any duplicate files in pak0.pak, and pak2.pak can override any duplicate files in either pak0.pak or pak1.pak. This pattern continues on and on. There is a catch, though. Pak files have to be in numerical order, or else they won't be loaded. So if I have the .pak set-up like so:

```
| | pak0.pak → — pak1.pak —X— pak5.pak —X— pak436.pak
```

pak0.pak and pak1.pak would be loaded and modify the game. However, pak5.pak and pak436.pak would not be counted, as the game will be looking for a pak3.pak file, and once it doesn't see it, it will stop looking for any more. If I would simply rename the above .paks to look like this:

```
| | pak0.pak → — pak1.pak → — pak2.pak → — pak3.pak
```

All the .paks will load and override duplicate files. I can use our custom 'yes.wav' as an example now. Notice how the default HL .pak file is called pak0.pak. This means that it is the lowest .pak there is. To make my sound play in HL, I would use my Pak editor to make a new file called pak1.pak in the /valve/ folder, as 1 is a greater number than zero, for my numerically challenged friends. Then I would

construct the same folder structure that the file I am replacing. Since the 'yes.wav' file is located in the sound/scientist/ folder in the pak0.pak file, I am going to make the same folders in my pak1.pak. So that my pak1.pak has a basic setup such as this:

```
-- pak1.pak
|
|
+sound/
|
|
+scientist/
```

Now, I would place my 'yes.wav' in the new scientist folder and then save the .pak (Make sure that your new .pak file is located in your HL/valve/ directory!). Now, whenever I run my game, the scientist will say my sound instead of the normal HL one!

The Folders - When You're Just Playing Around

If you look inside the /valve/ directory, or any other mod's directory for that matter, you can see that there are many folders that match the ones that are contained inside the pak0.pak file. This is because of the last step of the overriding process of the .pak files. If there are any files that have the same name and location in any of the .pak files and folders, the one in the folder will override the one in the .pak file, no matter what the number of the .pak is. So if I put my custom 'yes.wav' file in the valve/sound/scientist/ folder, it will override the one in the pak0.pak, producing the same result as the first technique.

The Custom Mod Folder - When You Distribute A Mod

Many people, when they take all the time to make a lot of new sounds, maps, etc. they just go all the way and make a new mod. Their mods are selectable from the Custom Games menu in Half-Life and have their own folder inside the /half-life/ directory. They put all their files into that custom folder, still adhering to the .pak and folder techniques described above. What's different is that they name their .pak file pak0.pak, as there is no other .pak file in the directory. Now, you may be asking if they did this, if they would have to insert all the media they used from the original pak0.pak in the /valve/ folder. The answer is no. If a mod folder doesn't have something that Half-Life requires, it will automatically, pardon my expression, run home to mommy and use the needed materials from Valve's pak0.pak, or any other .pak's in the /valve/ directory. As you can see, every way has its own uses. Whichever you use all has to do with what you are planning on doing with your materials.

From:

<http://xash3d.ru/> - **Xash3D**

Permanent link:

<http://xash3d.ru/doku.php?id=xashcookbook:en:articles:archived:pakfiles&rev=1402231661>

Last update: **2014/06/08 13:47**

