

Map Analyst

Version 2.5

Wenn man eigene Materials/Textures, Models oder Sounds benutzt, müssen diese Dateien zusammen mit der Level-Datei (BSP/VBSP) verschickt werden, damit die Materials/Textures, Models oder Sounds beim Spieler auch sichtbar bzw. hörbar werden.

Man kann aber auch mit speziellen Programmen die benötigten Dateien in die Level-Datei hineinpacken lassen. Der Map Analyst ('MAN') ist eines dieser Programme.

Er analysiert HL2 Map-Dateien und bestimmt alle zu diesem Level gehörenden Recourcen, d.h. Material-, Texture-, Sound-, Model-Dateien usw.

Diese Dateien können dann automatisch in die Level-Datei (BSP/VBSP) gepackt werden.

Besonderen Dank schulde ich Gosuke, der mich auf die Idee für diese Programm brachte. Nach den ersten Schritten haben mich viel Mapper unterstützt. Neben Gosuke haben sich Thomas 'Flausch' Abts, DJ Flyer alias Markus, Spice und Thorn besondere Verdienste erworben.

Ohne ihre Hilfe hätte der Map Analyst seinen Beta-Status längst noch nicht verlassen und viele Features würden fehlen.

DANKE!

1. Installieren des Map Analyst



Die ZIP-Datei kann an beliebiger Stelle ausgepackt werden. Üblich ist aber den Ordner 'Programme' in Hauptverzeichnis 'C:' zu wählen und dort einen Ordner 'man' zu erstellen.

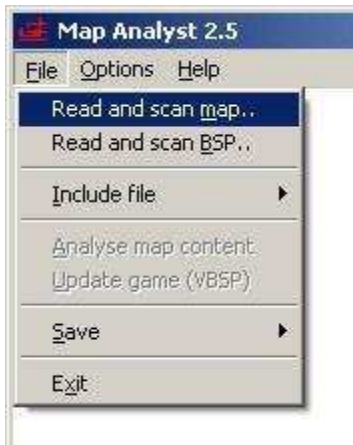
Nach dem Auspacken sollten dort die Dateien 'man.htm' (diese Datei) und das Programm 'man.exe', sowie der Unterordner 'pics' zu finden sein. Mit einem Doppelklick auf 'man.exe' wird das Programm gestartet.

Bevor mit der Analyse begonnen werden kann, muss der Game-Path gesetzt werden. Nur damit ist es möglich, auch die Dateien zu erfassen, die nur indirekt ermittelt werden können.

Im Options-Menü ist das Untermenü 'Game path' auszuwählen und dort 'Set game path'. Für das Spiel 'Counter Strike' könnte der Pfad 'C:\Valve\SteamApps\Gordon Freeman\counter-strike source\cstrike' lauten. Darunter befinden sich die Ordner wie 'materials', 'maps' usw.

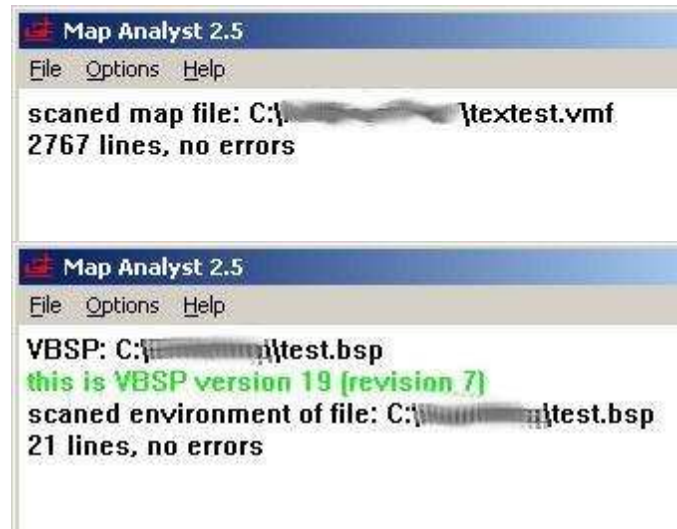
Für jedes Mod lautet der Pfad natürlich anders. Nachdem der Pfad gesetzt wurde, kann mit der Analyse der Map begonnen werden. Mit dem Menü-Eintrag 'View game path' kann der Pfad jeder Zeit kontrolliert werden.

2. Einlesen einer Map



Der Map Analyst wird mit einem **Doppelklick** gestartet.

Über das Filemenü kann eine Map Datei (VMF oder BSP) eingelesen werden. Dabei wird der Inhalt auch vorverarbeitet. Das Ergebnis dieser Analyse wird im Arbeitsfenster angezeigt. Alternativ kann die Datei mit Drag and Drop auch auf das Arbeitsfenster gezogen werden.



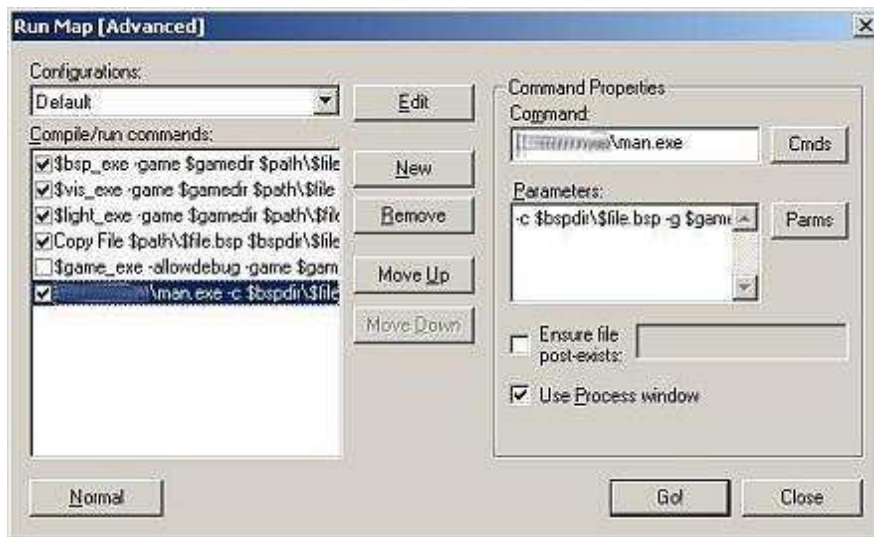
Man kann aber auch einfach die Map Datei (VMF oder BSP) **auf das Programm ziehen**. Hierbei ist aber zu beachten, dass die Konfigurationsdatei in diesem Fall nicht im selber Ordner wie das Programm selber steht. Daher muss auch für diese Art der Benutzung der 'Game path' gesetzt werden. Siehe hierzu auch 7. Optionen.

2.a Starten das Map Analyst von Hammer aus

Man kann den Map Analyst auch von Hammer aus starten lassen:

1. Compilieren der Map im 'Expert Mode'.
2. Ein neues Kommando einfügen.
3. Das 'Cmds' Feld muss auf den installierten MAN verweisen.
4. In der 'Parms' Zeile ist **-c \$bspdir\$file.bsp -g \$gamedir** einzutragen.
5. Dies sollte das letzte Kommando sein.

Jedes mal, wenn die Map im 'expert mode' compiliert wird, werden automatisch alle eigenen Dateien durch MAN in die BSP geschrieben.



3. Analysieren einer Map

Im Filemenü wird **Analyse map content** ausgewählt.

```

check vcd files
check sprites
check game files
add:    1436 maps\textest.txt
add:    240 resource\overviews\textest.txt
check materials
add:    85 materials\SDK\BUMP.vmt
add:    43896 materials\sdk\bump.vtf
add:    43896 materials\sdk\bump_normal.vtf
not added: materials\TOOLS\TOOLSSKYBOX.vmt
add:    93 materials\JDK\JDK_BURN.vmt
add:    2796408 materials\jdk\jdk_burn.vtf
add:    90 materials\JDK\JDK_WALLA.vmt
add:    174968 materials\jdk\jdk_walla.vtf

```

Nacheinander werden alle Model-, Sound-, Material- und Texture-Dateien und andere Game-Dateien (Bot-, Text-Dateien usw.) untersucht. Ist die Datei nicht vorhanden, wird der Ausgabertext blau angezeigt, wird die Datei gefunden normal in schwarz.

Außer der eigentlichen Model-Datei, die auf '.mdl' endet, werden auch die Dateien mit den Endungen '.dx80.vtx', '.dx90.vtx', '.sw.vtx', '.vvd' und '.phy' und dem selben Namen in die Liste eingetragen (wenn sie vorhanden sind).

Alle Material-Dateien (VMT), die in Model-Datei benutzt werden, werden analysiert und die benutzten Textur-Dateien (VTF) ebenfalls mit aufgenommen.

Nacheinander wird die Map auf verwendete Sound-Dateien (WAV), Sprites (SPR) und Scene-Dateien (VCD) untersucht.

Abschließend wird in alle Brushes nach Material-Dateien (VMT) untersucht und diese wie bei den Models nach verwendeten Textur-Dateien analysiert.

Gibt es im 'maps' Ordner eine Text oder Bot Datei (.txt oder .nav), wird auch diese hinzugefügt.

Alle Dateien, die nicht im Level-Verzeichnis gefunden wurden, werden nun in den GCF-Dateien gesucht. Diese Dateien enthalten alle Materials/Textures, Models und Sounds von HL2. Werden die Dateien dort auch nicht gefunden, werden sie in roter Schrift aufgelistet. Diese Dateien existieren also nicht und werden deshalb im Spiel nicht zur Verfügung stehen.

```

sound/ambient/killinghand22.wav
sound/ambient/schoolsout22.wav

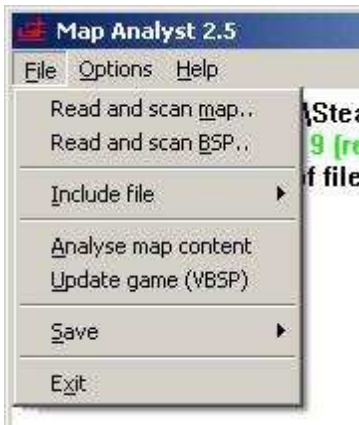
```

4. Einsetzen der Dateien in die Level-Datei (VBSP)

Im Filemenü wird **Update game (VBSP)** ausgewählt.

Alle gefundenen Dateien werden in die Level-Datei (BSP/VBSP) gepackt.

HL2 setzt auch interne Dateien in die BSP Datei ein. Die gefundenen Dateien werden zusammen mit den internen HL2 Dateien in die BSP Datei abgespeichert. Waren bereits gleichlautende, externe Dateien eingetragen, werden sie überschrieben.



Ist die Option 'Overwrite BSP' nicht gesetzt, wird von der BSP eine Kopie gemacht und diese in eine Testdatei 'man_betatest.bsp' im Ordner 'maps' abgespeichert, sonst wird sie überschrieben.

Dateien, die in die Level-Datei eingesetzt werden, werden in schwarzer Schrift ausgegeben. Existierte diese Datei in der Level-Datei bereits wird sie mit dem Wort 'replace' markiert und ersetzt. Ist sie neu, wird sie mit dem Wort 'include' markiert und eingesetzt. Dateien, die nicht durch die Analyse zum Hinzufügen bestimmt wurden, aber schon in der Level-Datei vorhanden waren, bleiben erhalten und werden in blauer Schrift ausgegeben (markiert mit dem Wort 'remain').

5. Einfügen weiterer Dateien



Über die Auswahl 'Include' und 'Choose file..' im Filemenü können weitere Dateien angegeben werden, die zusätzlich eingefügt werden sollen. In der Datei-Auswahl-Box können mehrere Dateien selektiert werden.

Will man diese Auswahl an zusätzlichen Dateien auch später noch mal verwenden, können sie mit der Auswahl 'Save file list..' in eine TXL-Datei abgespeichert werden. Dies ist eine reine Textdatei in der die relativen Pfade (relativ zum Level-Pfad) stehen. Man kann diese Liste zwar unter beliebigen Namen abspeichern, es empfiehlt sich aber, als Dateinamen den Namen der Map zu wählen, da diese Liste dann auch automatisch geladen werden kann.

Mit der Auswahl 'Read file list..' kann eine Liste geladen werden.

6. Abspeichern der Ergebnisse



Die Liste der gefundenen Dateien (Content list) kann in eine Textdatei ausgegeben werden.

Der Inhalt des Arbeitsfenster lässt sich mit der Auswahl 'Results' abspeichern.

Mit dem neuen Menüeintrag 'Map Info..' lässt sich eine Zusammenfassung der Map erzeugen und als Text-Datei abspeichern.

7. Optionen

Game path

Map Analyst 2.5

Mit der Option 'Game path' -> 'Set game path..' wird bei der Installation der Pfad des Spiele-Mods gesetzt. Mit der Auswahl 'View game path' kann man sich die aktuelle Einstellung anzeigen lassen.

Sobald eine Option verändert wird, werden die Einstellungen in die Datei 'man.cfg' abgespeichert. Wird 'man' durch einen Doppelklick gestartet, befindet sich diese Datei im selben Verzeichnis wie der Map Analyst.

Zieht man eine Map- oder BSP-Datei auf den Map Analyst wird aber das Verzeichnis 'Eigenen Dateien' genommen. In diesem Fall muss der 'Game path' dort erneut gesetzt werden.



Automatic

Hier kann die Analyse der eingelesenen Datei und das Einsetzen in die Level-Datei auf automatisch gestellt werden. Diese Phasen werden dann automatisch ausgeführt, wenn die vorhergehende erfolgreich abgeschlossen wurde.

Im Optionsmenü 'Automatic' können drei Modes gesetzt werden:

- *Include*: Die Liste der zusätzlichen Dateien wird automatisch geladen.
- *Analyze*: Wurde eine Map/BSP erfolgreich geladen wird automatisch mit der Analyse begonnen.
- *Update*: Wurde die Analyse erfolgreich durchgeführt, wird der Update der BSP automatisch durchgeführt.



MOD set

Alle Dateien, die im Spiel verwendet wurden, aber nicht im Spiele-Verzeichnis gefunden wurden, werden in den GCF-Dateien von HL2 gesucht. Je nach Mod können weitere GCF-Dateien (HL2 Deathmatch oder Counter-strike source) durchsucht werden.

Erstellt man ein Level für Counter-Strike ist hier die Option 'Counter-strike source' zu aktivieren.



Display

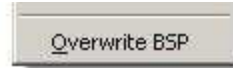
Meist werden sehr viele Dateien verwendet. Bei dieser Fülle kann es leicht geschehen, dass man Fehlermeldungen übersieht. Deshalb kann das Anzeigen der hinzugefügten Dateien bzw. der VBSP Dateien mit



dieser Option ein und aus geschaltet werden.

Overwrite

Durch einschalten der Option 'Overwrite BSP' wird beim 'update' die original BSP überschrieben. **Achtung:** Dies Programm ist noch eine Beta-Version, es ist also nicht ausgeschlossen, dass die BSP-Datei fehlerhaft und damit unspielbar wird.



Copyright 2005 by Rolf Hänisch, Berlin

Download: <http://www.map-lounge.de>

Email: mapbuilder@web.de