
Subject: About Westwood's Scripts

Posted by [TNaismith](#) on Fri, 27 May 2011 13:16:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

I'll lay out the background story first. If you want to skip to my question, feel free to jump to the bottom of the post.

Background Story:

My work in Level Edit primarily revolves around co-op mapping/mission/level making, and the extensive use of pre-written, public-released scripts are necessary for setting up bot behavior, trigger zones, checkpoints, mission objectives, mission messages, and so forth. I don't have the knowledge to code, script, or write my own. Neither do I think will I ever have the invested time/interest to do so... never adequately proper ones at least.

The custom scripts released by jonwil, and subsequently contributed to by various scripters/coders over the past years in the Renegade Community, have formed an impressive collection of usable scripts that, if you do your own little online searching, can yield helpful documentation in form of (this).

If you do enough self-research, you might stumble across (this) site, a SourceForge download page from jonwil which allows you to download and access some of the following files in the image below. Take note of the highlighted .txt readme(s). Each one corresponds to any of the scripts included in jonwil's 3.4.4 scripts by other contributors in the community. In addition, each of them (as of the 3.4.4 scripts) are lightly to moderately helpful documentation and explanation(s) of what the scripts do and how they function. The effort the folks spent typing this up and including it with their work is almost candidly appreciated by grunt mappers/modders like myself whom work in co-op missions where humans oppose the AI, and whom don't have the skills or invested talent to code or write our own scripts.

The Deeper Story:

For those wanting to get that extra bit of background information, here is the full scenario of what was going on prior to writing this post.

The question was posed earlier yesterday during a co-op map testing about the kind of script available for bots that would make them follow the player (escort-style) throughout the mission once being deployed from a GDI transport chinook. Doing what felt like some light and quick research at the time, I opened up Level Editor and skimmed through the list of "M01_" scripts -- Someone had mentioned that we should check to see if a script from M01 where GDI friendly bots

drop near the Church area and escort you via poke was actually available for use. After locating the area in the Level Editor

I checked things out and tried to determine how Westwood had setup those GDI bots to follow and escort the player upon being poked. After some probing and research, I found the following scripts to be the most likely related.

M01_CHURCH_Chinook_Spawned_Soldier01_GDI
M01_GDI_Escort_Conversation_Controller_GDI

I confess this might not be everything there is to how Westwood did it, there could very well be other scripts that are being used which I missed/didn't find. However, going with what what feeble results I had, I created a new map to reproduce the scripting setup in Level Editor that Westwood had done for M01. Placed a GDI bot on the map, attached "M01_CHURCH_Chinook_Spawned_Soldier01_GDI", and created a Daves Arrow with the script "M01_GDI_Escort_Conversation_Controller_GDI" attached to it.

At this point, it might be important to note that the way Westwood had set things up for this scenario was via a cinematic.txt file. Using XCC Mixer, I was able to open "Always.dat" (the existing one after installing core patches and scripts 3.4.4, aka, this wasn't "Always2.dat") and locate the cinematics file "x1i_gdi_church_troopdrop.txt", which seemed most reasonable to assume as being used for the Church area GDI bot drop-off. Peering inside, this was where I found "M01_CHURCH_Chinook_Spawned_Soldier01_GDI" and tried to then proceed with reproducing the scripted situation on a new map.

I assumed that the cinematics procedure wasn't relevant or necessary to executing the scripted situation Westwood had done, and went ahead to test my reproduced version of a GDI bot being poked and escorting the player. Unfortunately, it didn't work. I'm fairly sure (as far as I understand) that the cinematics system isn't necessary for making this script work, but I'm welcome to hearing alternative theories or evidence to prove other-wise.

So after this attempt didn't work, one thing I decided to find out for myself was that I would need to learn more about the scripts I was attempting to use. Where for jonwil's "JFW_" scripts, or any other of the custom scripts, I could simply open up some of the readme files and get adequate explanations or expanded details around the scripts like in (this) manner, Westwood's scripts, as I may have mentioned before, and will mention later, have no such official or easy-to-read documentation as the custom scripts in 3.4.4 do.

As you'll read in a few paragraphs below, the only hints or explanations of what some of Westwood's original scripts do are in very obscure manners or do not fully cover the full library of Westwood's original scripts included with C&C Renegade.

Perhaps out of desperation, I made an attempt to find out if scripts2.dll could contain some form of documentation for Westwood's scripts, even crude and in the form of rudimentary coding

comments. After realizing I couldn't open the file with a simple program like notepad, I did some research on how to open a .dll file, and came across results like this:

http://wiki.answers.com/Q/How_do_you_open_dll_files_to_see_what_is_written_inside

http://wiki.answers.com/Q/How_do_you_open_dll_files_to_see_what_is_written_inside

There's a lot of different things to consider, it seems; A few possible methods to open .dll files using what they call a 'com+ application', but only if you trust the .dll source, the possibility of opening the file up in Visual Studio only if it was developed/compiled there, and a foreboding warning that .dll files are often written in machine-code and that requires reverse-engineering and steps into the realm of legal liability. One might consider the next thing to try would be opening the scripts2.dll in Visual Studio (as per suggested above). However, I'm unaware of whether this will properly work for Westwood's scripts2.dll file, and more importantly, whether this is actually stepping into any legal grounds with Westwood/EA. My farthest assumption has been that some of jonwil's work with scripts 3.4.4 (and earlier versions), as well as TT's upcoming 4.0 Patch may be based on official permission from Westwood/EA either granted privately/individually or from some existing legal, official statement from EA/Westwood granting public use or access to reverse-engineer the scripts2.dll file.

I am unclear in this area and would appreciate some clarification if possible.

The Problem and Question:

The problem is that while these custom scripts are immensely useful and varied, and will continue to be so for those in the realm of co-op mapping, there is also a huge library of original Westwood scripts that have little to no known documentation released for public use on the internet. The only primary documentation attempt I could ever locate was (this) one started by Canadacdn on June 17th 2007, right here on the forums, which eventually made it onto some of the (old) renhelp sites, (like this one), becoming part of the archived history of tutorials for "Everything C&C Renegade!". There is also some indirect hinting at what some of Westwood's original scripts did among the documentation from jonwil's "JFW_" scripts for 3.4.4 (see here). This is all aside from the Westwood scripts that had somewhat obvious names to make guesses from. (See here again)

But here's what it comes to...

Question: Is there no official documentation (or in a format like this) released on the internet, or embedded in the scripts2.dll, from Westwood, that explains the huge library of original scripts shipped with C&C Renegade, available in places like when using the Level Editor shown (here)?

~ TNaismith

Thank you in advance for any replies or helpful answers, and an extensive nod of gratitude to both

Westwood for creating Renegade, and the many players/contributors across the community that have made custom scripts available for public use or helped build up the helpful knowledge base for using them.

If at any point there are broken links or images in this post, please send me a private message here on the forums and I'll see about fixing it.

Subject: Re: About Westwood's Scripts
Posted by [danpaul88](#) on Fri, 27 May 2011 13:22:53 GMT
[View Forum Message](#) <> [Reply to Message](#)

The M## (except M00) scripts are designed for the level indicated in the number, with M01_ being for level 1, M02_ being for level 2 etc. Many of them were not designed with re-use in mind and rely on hard coded object IDs in the levels they were used in.

M00 scripts are more generic and tend to work better outside of the original mission maps but even they tend to make a lot of assumptions internally and are not as flexible or powerful as many of the custom scripts which replaced them.

You might like to try out;

Quote:JFW_Escort_Poke (script to make an infantry unit follow whoever last poked them)
Shield (when someone pokes this infantry unit, the shield type will be changed to this type)
Shield2 (when the person the unit is following pokes this infantry unit, the shield type will be changed to this type)
If the person who the infantry unit is following pokes the unit, it stops.

Or

Quote:

=====
=====

[Script Name] NHP_Guard_Bot

=====
=====

[Description]

- Makes a bot escort a teammate when shot.

[Parameters]

- Distance (Range to stay within target escort)
- Speed (Speed to arrive at destination, can be more than the character's maximum speed)

[Notes]

- NONE
-

Subject: Re: About Westwood's Scripts
Posted by [saberhawk](#) on Fri, 27 May 2011 13:35:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

danpaul88 wrote on Fri, 27 May 2011 09:22The M## (except M00) scripts are designed for the level indicated in the number, with M01_ being for level 1, M02_ being for level 2 etc. Many of them were not designed with re-use in mind and rely on hard coded object IDs in the levels they were used in.

M00 scripts are more generic and tend to work better outside of the original mission maps but even they tend to make a lot of assumptions internally and are not as flexible or powerful as many of the custom scripts which replaced them.

Not only objects IDs. I've noticed hard-coded coordinates, "conversations", sounds, preset names, and waypoints. A lot of the scripts also assume that they are being run in SP and that there is only one "player".

Subject: Re: About Westwood's Scripts
Posted by [TNaismith](#) on Fri, 27 May 2011 14:49:03 GMT
[View Forum Message](#) <> [Reply to Message](#)

Right, those same observations came across to me as well after browsing and investigating the .mix levels provided (here). Over the last year or so I've opened those up from time to time to see how Westwood did things -- and it certainly makes sense to say most of their scripts, including the M##, are made solely for each level, with almost little to no re-usage outside of that.

The custom scripts are almost absolutely the main source of scripts that I personally use in co-op maps. Perhaps 90% of all scripts I've ever used in any single map is probably composed of the custom scripts. For those rare instances though, (and they've unfortunately become quite commonplace in my experience of mapping/mission-making), scripts relating to bot behavior(s) or vehicles have had to be specifically pulled from some of the M## scripts. In particular;

"M08_Mobile_Vehicle"
"M08_Nod_Light_Tank"

Are some of few available scripts for vehicle bot behavior. The work of (this) thread and it's contributors have gone a far way in uncovering helpful functions for some of the M## scripts, but the downside is that it's all had to be done from trial and error testing.

I guess it's clear to say then, given what has already been concluded in the replies so far in this thread, (considering the context that the other thread was made in as well), that there is no official documentation of the Westwood scripts, released and/or written by the Westwood developers themselves at this time. And at least none in a format similar to (this).

It's a shame, but understandable and things will just have to proceed on without such resources available. The custom scripts do a pretty good job covering anything a co-op mission maker might need in the Level Editor. I'm also aware that Scripts 4.0/TT Patch will include new custom scripts as well that might cover even more areas a co-op mapper/modder might desire. (Such as an improved vehicle ai script called "CPU_Neo_Vehicle_AI", first mentioned in this tutorial here)

Definitely looking forward to that as being a possible solution and enhancement to those working in LE and creating co-op missions.

Thank you as well for the suggested scripts. I have come across each of those at some point in the past, and I distinctly recall trying them out, but to no avail. However, I'll take a look again.

On another aside, scripts like;

"JFW_Disable_Transition (clone of M00_Disable_Transition)"

Are quite easy to figure out, even if derived from a M## script. However, there are a few, relatively rare, custom scripts that aren't too clear in how to use. For example, the scripts below have only the listed explanations/details for actual use in Level Editor:

Quote:JFW_Engineer_Target (clone of M03_Engineer_Target)

JFW_Engineer_Repair (clone of M03_Engineer_Repair)
Repair_Priority (priority for repairing)

Also, in addition, JFW_Engineer_Target has been modified to check the Shield Strength (i.e. armour) of the object in addition to the health.

The repair object can be any vehicle or infantry that has a repair gun weapon and wheels/legs/tracks/whatever so it can move.

Also, there is a limit to how far away the repair script will detect a damaged target (since I can't find the sound presets it's using, I can't identify how far that is)

In my earlier testing, I attached "JFW_Engineer_Target" to an engineer bot, and "JFW_Engineer_Repair" to an object (infantry/vehicle/inanimate). Testing this in-game, the scripts didn't appear to work. Setting the scripts up vice versa didn't work either. It was a bit confusing to try and guess how the script should be setup and properly used, and there didn't seem to be enough explanation/documentation for it. I think this might derive from the script being

a 'clone' of a M## script, and sounding simple to use, but not quite so simple. Or, perhaps I just haven't figured it out right.

As I said before, scripts like "JFW_Disable_Transition (clone of M00_Disable_Transition)" are quite easy and shouldn't take anyone too long to figure out, but for me, scripts at the next level of seemingly-simple usage like "JFW_Engineer_Target" often stump me. If there was official documentation from Westwood around their original mission scripts, this is an example of where it would come in handy, and eliminate the need to guess and test. As the folks in (this) topic seemingly had to do.

Some things can only be done with hard work I suppose.

Appreciate the swift and helpful response(s). See you around.

Subject: Re: About Westwood's Scripts
Posted by [kamuixmod](#) on Fri, 27 May 2011 14:55:17 GMT
[View Forum Message](#) <> [Reply to Message](#)

actually the object had to have the engineer_target script and the engineer self the engineer_repair script.

btw.They should make scripts in Script 4 for Aircraft behaviors as well because its stupid to have an apache go over you all the time and wait 10 seconds before it shoots. They should make a script where apaches can shoot you from distance as well by being in front of you or etc

Subject: Re: About Westwood's Scripts
Posted by [TNaismith](#) on Fri, 27 May 2011 15:04:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

kamuixmod wrote on Fri, 27 May 2011 07:55

btw.They should make scripts in Script 4 for Aircraft behaviors as well because its stupid to have an apache go over you all the time and wait 10 seconds before it shoots. They should make a script where apaches can shoot you from distance as well by being in front of you or etc

Right, I forgot to mention that as well. "M08_Mobile_Apache" is the only air vehicle script that has been known to work decently, but the flaws are indeed that the air vehicle bot often waits 10 seconds before shooting, and even then, it only fires once it can get a 'lock' onto a player's character for 10 seconds, without the player moving the camera view around, or moving the character around on the terrain. It's quite restrictive in that sense, but it's pretty much the only air vehicle bot script that even works. The script still works even in A Path Beyond, one of the co-op maps utilizes the script to have ai Allied Hinds attack players on the Soviet team.

Unlike scripts for vehicle ai on the ground, I haven't heard of any improved/new scripts for air

vehicle bot behavior that could be considered for scripts 4.0 with the TT patch.
