
Subject: scripts.dll 2.7 WIP update

Posted by [jonwil](#) on Sat, 15 Apr 2006 14:01:32 GMT

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

Here is a list of all the features currently added to 2.7:

Various small improvements to engine calls in engine.cpp (e.g. some engine calls now use the new Disarm_Beacon engine call to disarm beacons)

An improvement to Console_Output to enable it to be used like printf (in that you can use %s, %f etc and pass variables to it)

new engine calls:

void Send_Message_With_Obj_Color(GameObject *obj,const char *Msg); //send a messages in a given objects color

void Send_Message_With_Team_Color(int Team,const char *Msg); //Send a message in a given teams color

void Damage_Occupants(GameObject *obj,float Damage,const char *Warhead); //Damage all the occupants of a vehicle

int Get_Current_Bullets(GameObject *obj); //Get loaded bullets for an objects current gun

int Get_Current_Clip_Bullets(GameObject *obj); //Get clip/backpack bullets for an objects current gun

int Get_Current_Total_Bullets(GameObject *obj); //Get total bullets for an objects current gun

int Get_Total_Bullets(GameObject *obj,const char *weapon); //Get total bullets for a specific gun (if the object doesnt have the gun, return is zero)

int Get_Clip_Bullets(GameObject *obj,const char *weapon); //Get clip/backpack bullets bullets for a specific gun (if the object doesnt have the gun, return is zero)

int Get_Bullets(GameObject *obj,const char *weapon); //Get loaded bullets bullets for a specific gun (if the object doesnt have the gun, return is zero)

int Get_Current_Max_Bullets(GameObject *obj); //Get max loaded bullets for an objects current gun

int Get_Current_Clip_Max_Bullets(GameObject *obj); //Get max clip/backpack bullets for an objects current gun

int Get_Current_Total_Max_Bullets(GameObject *obj); //Get total bullets for an objects current gun

int Get_Max_Total_Bullets(GameObject *obj,const char *weapon); //Get max total bullets for a specific gun (if the object doesnt have the gun, return is zero)

int Get_Max_Clip_Bullets(GameObject *obj,const char *weapon); //Get max clip/backpack bullets bullets for a specific gun (if the object doesnt have the gun, return is zero)

int Get_Max_Bullets(GameObject *obj,const char *weapon); //Get max loaded bullets bullets for a specific gun (if the object doesnt have the gun, return is zero)

int Get_Position_Total_Bullets(GameObject *obj, int position); //Get total bullets for an objects gun at a specific position

int Get_Position_Bullets(GameObject *obj,int position); //Get loaded bullets for an objects gun at a specific position

int Get_Position_Clip_Bullets(GameObject *obj,int position); //Get clip/backpack bullets for an objects gun at a specific position

int Get_Position_Total_Max_Bullets(GameObject *obj, int position); //Get total bullets for an objects gun at a specific position

int Get_Position_Max_Bullets(GameObject *obj,int position); //Get loaded bullets for an objects gun at a specific position

```
int Get_Position_Clip_Max_Bullets(GameObject *obj,int position); //Get clip/backpack bullets for
an objects gun at a specific position
void Set_Current_Bullets(GameObject *obj,int bullets); //Set current loaded bullets for an object
void Set_Current_Clip_Bullets(GameObject *obj,int bullets); //Set current clip/backpack bullets for
an object
void Set_Position_Bullets(GameObject *obj,int position,int bullets); //Set position loaded bullets for
an object
void Set_Position_Clip_Bullets(GameObject *obj,int position,int bullets); //Set position
clip/backpack bullets for an object
void Set_Bullets(GameObject *obj,const char *weapon,int bullets); //Set loaded bullets for an
object
void Set_Clip_Bullets(GameObject *obj,const char *weapon,int bullets); //Set clip/backpack bullets
for an object
const char *Get_Team_Name(int Team); //Get a teams name
const char *Get_Vehicle_Name(GameObject *obj); //Get the name of a vehicle (as used for the
HUD display)
INIClass *Get_INI(char const *filename); //Open an INI file and read stuff from it (you can access
various things through the iniclass that is returned to read bool, float, int, string)
void Release_INI(INIClass *ini); //Close an INI file
void Disarm_Beacon(GameObject *obj); //Disarm a beacon
void Disarm_Beacons(int ID); //Disarm all beacons for a player
GameObject *Find_Harvester(int team); //Find this teams current harvester, if any
int GetMaxPlayerID(); //Get the maximum currently used player ID
new script JFW_Damage_Occupants_Death which damages all occupants of a vehicle when the
vehicle is killed
Versions of JFW_Character_Buy_Poke, JFW_Refill_Buy_Poke, JFW_Vehicle_Buy_Poke,
JFW_Powerup_Buy_Poke, JFW_Preset_Buy_Poke, JFW_Group_Purchase_Poke,
JFW_Powerup_Buy_Poke_Timer, JFW_Weapon_Buy_Poke, JFW_Preset_Buy_Poke_Timer &
JFW_Preset_Buy_Poke_Custom that play a sound if the poker has not got enough cash for the
purchase.
Versions of JFW_Switch_Door, JFW_Switch_Lock_Door, JFW_Toggle_Door,
JFW_Toggle_Lock_Door, JFW_Switch_Door_Team & JFW_Toggle_Door_Team that use the
object they are attached to as the object to act on (instead of needing to pass the object ID in)
new script, JFW_Vehicle_Extra which basicly creates an object at a particular bone of whatever
JFW_Vehicle_Extra is on and attaches the object to the bone. Then, when the object with
JFW_Vehicle_Extra on it is killed, the other object is destroyed too.
new console command to disarm all C4 of a player
new console command to disarm all proximity C4 of a player
new console command to disarm all beacons of a player
Bug fixes to a few places to use a more accurate player count (instead of using the number of
players on the server, it uses another value that returns the highest player ID currently in use).
Among other things, this should fix the reported PINFO bug.
A new hook that gets called anytime the "The version of bhs.dll for player x is y" message gets
printed. The hook gets passed the player ID and the version number.
A new engine call to set wireframe or not wireframe. Like the mine limit and vehicle limit, this stays
set until you set it to something else, even across maps.
Code to send the Disable_All_Collisions, Disable_Physical_Collisions and Enable_Collisions
script commands over the network.
```

A new engine call to load an ini file at runtime that will replace certain stuff read from hud.ini. Like the engine call to replace the HUD_MAIN texture, this stays in force until changed again. Note that this will still work even if there is stuff in the ini file that the command doesn't read, the other stuff will be ignored. Obviously, the client needs the ini file and all the textures (just like with hud.ini)

New tags for hud.ini, as follows: (details will be forthcoming once I actually figure out how to document all this stuff in a way that someone other than myself can understand)

Firstly, you can define entries for colors (with a red, green and blue) which can be referenced elsewhere in the file.

Secondly, you can completely customize the number that shows your current health and the one that shows your current shield

Also, you can completely customize the numbers that show your current bullets (both of them)

Also, you can customize both the name and image for the current weapon (which includes things like the steering wheel/gun/seat icons for inside a vehicle)

You can set a setting to make the names (and only the names) of enemy soldiers invisible. Also, there is an exception list so that certain presets will remain visible (e.g. for spies)

Also, you can have an "unlimited" (limited only by system resources/the engine) number of "textures". Each has a texture filename and a number of rectangles associated with it. Each rectangle has a color, a location and a UV setting associated with it (the UV setting defines a rectangle on the texture, if the texture is 128x128 and the UV is 0,0,64,64 then that selects the top left corner. The UV is also used to specify the size of the rectangle.

You can customize the health bar and shield bar including textures, UV offsets, colors and many other things (you can do both renegade health bar style bars and renegade shield strength style bars)

The health/shield bars and health/shield numbers take settings to say "if player has x%, use this color"

Also, all the new hud elements can be disabled completely if you don't want them (e.g. shield bar if your mod has no shield strength settings anymore)

Plus, you can disable the radar, compass, weapon box (i.e. the circle thing that goes behind the weapon image, weapon name and bullet counts) and info box (i.e. the thing that covers every part of the main hud except the radar and the text items)

Note that some of this stuff (all the HUD stuff for example, also the wireframe mode and other things) will need bhs.dll on the client.

Here is what I still have on the 2.7 to-do list, not all of it will make it into 2.7 though:

Definitely in 2.7:

Fix the bug in Set_Death_Points/Set_Damage_Points so that they actually work

Further changes to the WOL stuff that blazer needs for LFDS WOL (expect the WOLSEND console command to go away in 2.7)

Fix the bug where the obelisk glow doesn't seem to work right in some cases

A script you can put on a map to either set wireframe mode or not set wireframe mode

A script to play one sound if the player has key x and another sound if they don't.

Engine calls to get the vehicle owner (i.e. the purchaser) and the "lock time" (i.e. how long before it will become unowned)

Note that entering a vehicle resets the owner as does the expiration of the lock time (defaults to 26 seconds when you buy the vehicle)

An engine call to give a specific weapon (not a powerup, the weapon preset)
An engine call to get the weapon preset that a particular PowerupGameObj will grant.
Check if the code to change bullets, add weapon or remove weapon has to go over the network in order to work 100% and if so, make it happen.
Disable options for other HUD elements (e.g. team/player info, credits/time remaining info)
Custom code to add a little icon for the health and one for the shield (like the little plus next to the health number in normal renegade). This will support the same color changeability as the other health/shield options plus it will optionally allow you to set a flashing icon (like renegade does when you have low health)

Probably going to be in 2.7:

Full customizability for the compass
Customizability for the radar (how much is unknown at this time but I want to implement the ability to use a texture of the current map like the ones in SP as a radar background and then it will scroll depending on where you are on the map. It would always stay pointing "north" though (i.e. top of texture = top of screen)
Changing the time remaining and credits
Investigate/fix the reported blurry screenshot bug when you take a PNG screenshot in windowed mode.
Support for the side buttons on my Microsoft USB Optical Intellimouse.
Code to read various PT related items (specifically, being able to read data from the presets listed under "Purchase Settings" and "Team Purchase Settings" in the preset tree in leveledit)

Might make 2.7, unclear at this point:

Various scripts that use the weapon related engine calls.
Ability to change the team/player info at the top of the screen. (i.e. move it and stuff)
Look into the reported turret lag issue in RenAlert (which is where the location that the graphics engine draws the turret and any bullets etc at and the actual location are different)
Find out if Action_Complete is or isn't being called properly and if its not, try and make it work.
Engine calls to set the owner and lock time of a vehicle (this will need bhs.dll on the client)
Code to change the PT data at runtime (same stuff as the read code). This will need bhs.dll on the client to work.

Naturally, all of the stuff in this post is subject to change (e.g. if testing reveals something isn't going to work, I will remove it or change it)

Subject: Re: scripts.dll 2.7 WIP update
Posted by [jonwil](#) on Mon, 17 Apr 2006 14:36:46 GMT
[View Forum Message](#) <> [Reply to Message](#)

Stuff completed:

Fix the bug in Set_Death_Points/Set_Damage_Points so that they actually work
A script you can put on a map to either set wireframe mode or not set wireframe mode
A script to play one sound if the player has key x and another sound if they dont.
Engine call to get the vehicle owner (i.e. the purchaser)
An engine call to get the weapon preset that a particular PowerupGameObj will grant.

Check if the code to change bullets, add weapon or remove weapon has to go over the network in order to work 100% and if so, make it happen. (Change Bullets works 100% no network send, Remove Weapon needs network send for 100% working, Add Weapon bumped to later version and will be tested if its ever implemented)

Investigate/fix the reported blurry screenshot bug when you take a PNG screenshot in windowed mode. (I am unable to reproduce this, if anyone can show me a screenshot taken with the TGA code that is non-blurry and the same (or almost the same) taken with the PNG code that is blurry, I will investigate further.

Stuff still on the to-do list that is definatly in 2.7:

Further changes to the WOL stuff that blazer needs for LFDS WOL (expect the WOLSEND console command to go away in 2.7)

Fix the bug where the obelisk glow doesnt seem to work right in some cases

Disable options for other HUD elements (e.g. team/player info, credits/time remaining info)

Custom code to add a little icon for the health and one for the shield (like the little plus next to the health number in normal renegade). This will support the same color changeability as the other health/shield options plus it will optionally allow you to set a flashing icon (like renegade does when you have low health)

Code to read various PT related items (specifically, being able to read data from the presets listed under "Purchase Settings" and "Team Purchase Settings" in the preset tree in leveledit). This is almost finished and is definatly going to be finished soon.

A script (or more likely scripts) similar to JFW_Vehicle_Extra but with extra functionality so that you build both the vehicle and the "extra" with the same health, shield strength, armour, skin etc (i.e. make them identical as far as health related settings goes) then anytime damage is done to the vehicle, the extra is set to the same health & shield settings and vice versa. Not 100% perfect (in that max health changes for example wont be transfered) but good enough for the use to which HTMLGOD wants it for in SWMOD.

```
const char *Get_Definition_Name(unsigned long id); //Get the name of a definition/preset given its ID
```

```
unsigned long Get_Definition_ID(const char *name); //Get the ID of a definition/preset given its name
```

Stuff that is still planned for 2.7 at this point but might not make it:

Support for the side buttons on my Microsoft USB Optical Intellimouse. (I found most of the pieces of the renegade mouse handling code, still need to find out how the configuration dialog works though . This feature (if/when I implement it) would probably enable use of any buttons that map to directx mouse values DIMOFS_BUTTON3 and up (e.g. the aforementioned side buttons on the aforementioned optical mouse)

Various scripts that use the weapon related engine calls. (need to decide which ones are worth implementing)

Ability to change the team/player info at the top of the screen. (i.e. move it and stuff)

Look into the reported turret lag issue in RenAlert (which is where the location that the graphics engine draws the turret and any bullets etc at and the actual location are different)

Find out if Action_Complete is or isnt being called properly and if its not, try and make it work. (if its not working, that would explain a lot of things that have been happening in tests I have done recently)

Code to change the PT data at runtime (same stuff as the read code). This will need bhs.dll on the client to work. (shouldnt be that hard, most important thing is to make sure that when a new player

joins, they are sent the right data so their PT is up-to-date with what the server has.

Full customizability for the compass

Customizability for the radar (how much is unknown at this time but I want to implement the ability to use a texture of the current map like the ones in SP as a radar background and then it will scroll depending on where you are on the map. It would always stay pointing "north" though (i.e. top of texture = top of screen)

Changing the time remaining and credits display

Stuff removed from the 2.7 todo list and bumped to later version (for various reasons):

An engine call to give a specific weapon (not a powerup, the weapon preset). This one removed because it would need to send data over the network plus I haven't figured out how to add a weapon given the weapon definition yet.

Engine call to get the "lock time" of a vehicle (i.e. how long before it will become unowned).

Engine calls to set the owner and lock time of a vehicle (this will need bhs.dll on the client)

This one removed because I haven't figured out how to work with changing the object referenced by a ReferencerClass (I can get the object back but not change it at this point)

Subject: Re: scripts.dll 2.7 WIP update

Posted by [jonwil](#) on Wed, 19 Apr 2006 11:29:15 GMT

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

Further progress update:

Thanks to WD, some new scripts:

MDB_Base_Defense_Popup_No_VTOL

MDB_Base_Defense_Popup_VTOL_Only

MDB_Base_Defense_Popup

MDB_Send_Custom_Enemy_Seen

These are nice, you can use them to do things like the popup sam sites in Tiberian Dawn (IIRC there is actually a .gmax file in the westwood buildings.zip file that would work for this) and other things. They have an animation that plays when they pop up and also an (optional) sound that can be played. (e.g. grinding gears for example)

The code to read the PT is complete now, you can read costs, texture names, string IDs and preset IDs for all the PT buttons now. Plus, I got these 2 engine calls:

```
unsigned int Get_Team_Cost(const char *preset, unsigned int team); //Get the cost of a preset for a given team. Returns zero if the preset is not found in any of the purchase terminal data or if it is one of the free units.
```

```
unsigned int Get_Cost(const char *preset); //Get the cost of a preset. Returns zero if the preset is not found in any of the purchase terminal data or if it is one of the free units.
```

Both of these are complete also:

```
const char *Get_Definition_Name(unsigned long id); //Convert a definition/preset ID into a name
```

```
unsigned long Get_Definition_ID(const char *name); //Convert a definition/preset name into an ID
```

Subject: Re: scripts.dll 2.7 WIP update

Posted by [jonwil](#) on Sun, 23 Apr 2006 11:57:15 GMT

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

I have made quite a bit of progress on 2.7, including further HUD changes, the new stuff blazer wanted for LFDS XWIS plus the new advanced guard tower functionality.
Stuff Still to do for 2.7 (hopefully I will get all of this done, maybe a few things will be dropped):
Document all the new stuff (documenting all the hud.ini stuff is going to be a pain in the ass
Write a complete 2.7 changelog
Test anything that needs to be tested (i.e. anything that hasnt yet been tested)
New script that HTMLGOD wanted for swmod.
Customizable compass (i.e. being able to change its location on the screen, its font and its color).
Customizable credits/time remaining display (i.e. being able to display credits/time remaining anywhere on the screen with optional text before and after it)
Ability to change the PT data at runtime (i.e. being able to change the textures, strings, presets and costs).
A script that, when an object dies, will empty a given PT slot (i.e. make it totally blank) so you cant buy it anymore.
Customizable radar (So far the hard part is getting all the radar blips drawn in the right place and also the scrollable map functionality). I may drop the scrollable map functionality from 2.7.

Subject: Re: scripts.dll 2.7 WIP update

Posted by [Cat998](#) on Sun, 23 Apr 2006 12:04:41 GMT

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

jonwil wrote on Sun, 23 April 2006 13:57 Ability to change the PT data at runtime (i.e. being able to change the textures, strings, presets and costs).

Does this also include refill ?

Subject: Re: scripts.dll 2.7 WIP update

Posted by [jonwil](#) on Sun, 23 Apr 2006 22:37:30 GMT

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

Basicly, everything changeable in leveledit under "team purchase settings" and "purchase settings" will be changeable (except obviously the team and type
There will be a script to remove a PT icon completly. (e.g. helipad dies, cant build helicopters anymore)

Subject: Re: scripts.dll 2.7 WIP update

Posted by [theplague](#) on Mon, 24 Apr 2006 02:30:13 GMT

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

ok, i know leave hook isn't in this version, but can you add something to check whether a person is ingame? when they are alive/dead and loading?

Subject: Re: scripts.dll 2.7 WIP update
Posted by [jonwil](#) on Mon, 24 Apr 2006 08:19:16 GMT
[View Forum Message](#) <> [Reply to Message](#)

Maybe for 2.8, its too late for 2.7.

Subject: Re: scripts.dll 2.7 WIP update
Posted by [jonwil](#) on Mon, 24 Apr 2006 14:41:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

ok, still to do:
Document everything
Write changelog
Test anything still to be tested
2 Scripts HTMLGOD wanted for SWMOD
Write code to correctly send PT data over the network (when its changed for example or when a player first joins so they get the changes too)
Customizable compass
Customizable credits/time remaining
Customizable radar (how customizable it will be is unclear at this point)

If anything is dropped, it would probably be the radar or maybe the compass/time remaining changes.

Subject: Re: scripts.dll 2.7 WIP update
Posted by [danpaul88](#) on Mon, 24 Apr 2006 16:12:26 GMT
[View Forum Message](#) <> [Reply to Message](#)

jonwil wrote on Wed, 19 April 2006 12:29 Thanks to WD, some new scripts:
MDB_Base_Defense_Popup_No_VTOL
MDB_Base_Defense_Popup_VTOL_Only
MDB_Base_Defense_Popup
MDB_Send_Custom_Enemy_Seen

Just out of interest, in what way do these differ from the jfw_base_defense_animated scripts? I tried using those ones before but they didn't work how i needed them too because when the turret

was undeployed it was unable to see enemies, and would therefore never deploy (as the muzzle was underground until deployed).

Subject: Re: scripts.dll 2.7 WIP update
Posted by [jonwil](#) on Tue, 25 Apr 2006 00:12:49 GMT
[View Forum Message](#) <> [Reply to Message](#)

I believe this new script corrects some of those problems I think.

Subject: Re: scripts.dll 2.7 WIP update
Posted by [Whitedragon](#) on Tue, 25 Apr 2006 00:22:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

Quote: =====
=====

; ***** [Script Name] MDB_Base_Defense_Popup

=====

[Description]

- Base defense script designed to "hide" underground and then popup to attack enemies when they come within range.
Attacks all units.

[Parameters]

- MaxAttackDistance: Maximum range of the weapon this base defense uses.
- ReturnTime: How long the base defense should stay above ground after it last attacked something.
- Animation: The animation of the base defense going from underground to above ground. The first frame(0) should be the base defense completely underground.
The last frame should be the base defense completely above ground, ready to attack.
- LastFrame: The last frame of the animation.
- Sensor_Preset: When your base defense is underground it cannot see units above ground, therefore it needs a "Sensor" object above ground which spots units for it.
This should be an exact copy of your base defense preset, except with it's model set to NULL

and ALL scripts removed from it.

- Sensor_Creation_Bone: Bone to create the Sensor object at. This bone should always be above ground(aka not part of the animation) and should be at about the height of the muzzle bone.

- Sound: Sound preset to play when the base defense goes from underground to above ground, or vice versa.

- Adjust_Aim_For_Infantry: Normally base defense aim at the head of infantry. With some types of weapons, like shells or lasers, this greatly decreases their accuracy. If this is set to 1 the base defense will instead aim at the feet of the infantry.

```
=====
=====
```

```
; ***** [Script Name] MDB_Base_Defense_Popup_No_VTOL
```

```
=====
=====
```

[Description]

- Base defense script designed to "hide" underground and then popup to attack enemies when they come within range.
Attacks all ground units.

[Parameters]

- MaxAttackDistance: Maximum range of the weapon this base defense uses.

- ReturnTime: How long the base defense should stay above ground after it last attacked something.

- Animation: The animation of the base defense going from underground to above ground. The first frame(0) should be the base defense completely underground.
The last frame should be the base defense completely above ground, ready to attack.

- LastFrame: The last frame of the animation.

- Sensor_Preset: When your base defense is underground it cannot see units above ground, therefore it needs a "Sensor" object above ground which spots units for it.

This should be an exact copy of your base defense preset, except with it's model set to NULL and ALL scripts removed from it.

- Sensor_Creation_Bone: Bone to create the Sensor object at. This bone should always be above

ground(aka not part of the animation) and should be at about the height of the muzzle bone.

- Sound: Sound preset to play when the base defense goes from underground to above ground, or vise versa.

- Adjust_Aim_For_Infantry: Normally base defense aim at the head of infantry. With some types of weapons, like shells or lasers, this greatly decreases their accuracy. If this is set to 1 the base defense will instead aim at the feet of the infantry.

```
=====
```

```
; ***** [Script Name] MDB_Base_Defense_Popup_VTOL_Only
```

```
=====
```

[Description]

- Base defense script designed to "hide" underground and then popup to attack enemies when they come within range.
Attacks only VTOL units.

[Parameters]

- MaxAttackDistance: Maximum range of the weapon this base defense uses.
- ReturnTime: How long the base defense should stay above ground after it last attacked something.
- Animation: The animation of the base defense going from underground to above ground. The first frame(0) should be the base defense completely underground.
The last frame should be the base defense completely above ground, ready to attack.
- LastFrame: The last frame of the animation.
- Sensor_Preset: When your base defense is underground it cannot see units above ground, therefore it needs a "Sensor" object above ground which spots units for it.
This should be an exact copy of your base defense preset, except with it's model set to NULL and ALL scripts removed from it.
- Sensor_Creation_Bone: Bone to create the Sensor object at. This bone should always be above ground(aka not part of the animation) and should be at about the height of the muzzle bone.

- Sound: Sound preset to play when the base defense goes from underground to above ground, or vice versa.

Subject: Re: scripts.dll 2.7 WIP update
Posted by [Fifaheld](#) on Tue, 25 Apr 2006 05:14:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

white when comes please a new ssaow version out?

Subject: Re: scripts.dll 2.7 WIP update
Posted by [danpaul88](#) on Tue, 25 Apr 2006 06:40:14 GMT
[View Forum Message](#) <> [Reply to Message](#)

aha, so your scripts use a sensor object... exactly what I needed, great work
