
Subject: scripts.dll 2.8 is out

Posted by [jonwil](#) on Sat, 24 Jun 2006 00:36:43 GMT

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

Get it from <http://www.sourceforge.net/projects/rentools>.

Complete list of changes:

Items with a * need bhs.dll on the client

New logging feature for FDS (win32 and linux) that logs all text that appears on the console as UDP packets to a specified host and port. You can then grab these and parse them (they are sent as raw null terminated strings)

*Changes to wireframe mode to make only the 3D objects wireframe, not the 2D UI etc.

*New feature to allow for a scrolling map texture as a background to the radar.

New hud.ini keyword to disable the 2x cost multiplier that you get when your powerplant goes down

New hud.ini keyword to change the "delay factor" that is applied when your pp goes down (not 100% sure what it does though)

New hud.ini keyword to disable vehicle ownership (i.e. where only you can get in the vehicle you just bought)

*New hud.ini keyword to let you use the extra PT pages (the ones you access with ALT) as regular PT pages.

*New hud.ini keyword to change the way the game checks if the vehicles and characters icons should be enabled which is intended for use by mods that want to use

Set_Can_Generate_Soldiers or Set_Can_Generate_Vehicles.

Some fixes to some possible server exploits

*A patch that will make mods that use the Com Center building or the Enable_Base_Radar to disable the radar work properly

New engine call Get_Build_Time_Multiplier that gets the current build time multiplier (if the pp is down, you get the value set with the above hud.ini keyword or the default of 2, otherwise you get 1)

*New engine call to change the settings for the scrolling map feature (intended to be used on a per-map basis)

New hook to hook beacon purchases so you can block them if you want (or handle them specially, e.g. enforce beacon limit)

New hook to hook vehicle purchases so you can do stuff like block normal weapons factory if your script builds the vehicle some other way)

New hook to hook character purchases, same as the other 2.

New hooks to let you monitor/log vehicle, character or beacon purchases.

You can register as many of these hooks as you like

*New engine call Set_Currently_Building which sets a flag. When this flag is set for a given team, the PT will display "building" underneath the vehicles button regardless of what the normal weapons factory is doing.

*New engine call Is_Currently_Building which returns the value set by Set_Currently_Building

Bug fixes to some engine calls

Bug fixes & improvements to SimpleDynVecClass and SimpleVecClass

Improvements to the definitions of cGameData and friends (including definitions of

BaseControlerClass, NetworkObjectClass and others). With this new work, cGameDataCnC, cGameDataSinglePlayer, cGameDataSkirmish and cGameData now have pretty much 100%

correct definitions.

Bug fixes to the object create hooks to make them crash less.

New engine calls:

```
void Power_Base(int team,bool powered); //Power a base up or down, correctly handles the
doubled build time and costs
void Set_Can_Generate_Soldiers(int team,bool cangenerate); //Sets if soliders are purchasable
void Set_Can_Generate_Vehicles(int team,bool cangenerate); //Sets if vehicles are purchaseable
void Destroy_Base(int team); //Destroys a base completly
void Beacon_Destroyed_Base(int team, bool destroyed); //Sets if a base was destroyed by a
beacon in the beacon zone
void Enable_Base_Radar(int team, bool enable); //Enables radar for a base, same thing as the
communications center does
bool Is_Harvester(GameObject *obj); //Is this object one of the 2 team AI harvesters?
void Disable_Enlisted(unsigned int team, unsigned int position); //Disable an enlisted soldier
void Disable_Preset(unsigned int team, unsigned int type, unsigned int position); //Disable a
preset
GameObject *Get_GameObj_By_Player_Name(const char *name); //get the gameobject of a
player given their name
unsigned long Get_Definition_Class_ID(const char *name); //get the class ID of a definition
(matches the #defines above)
unsigned int Get_Vehicle_Definition_Mode(const char *preset); //get the mode of a vehicle given
its preset name
bool Is_Radar_Enabled(int team); //returns if the radar is enabled
int Building_Type(GameObject *obj); //returns the type of a BuildingGameObj
bool Is_Building_Dead(GameObject *obj); //Does the game consider the building dead
GameObject *Find_Building(int team,int type); //Find a building by team and type
GameObject *Find_Base_Defence(int team); //Find base defence for a team
bool Purchase_Item(GameObject *obj,int cost); //Pass a soldier object & a cost to deduct that
much money from the soldier. Returns true if the transaction suceeded, false otherwise
bool Is_Map_Flying(); //Is the current map a flying map
GameObject *Find_Closest_Zone(Vector3 &Location,unsigned int Type); //Find the closest zone
bool IsInsideZone(GameObject *zone,GameObject *obj); //is <solder/vehicle> inside <zone>
bool Can_Team_Build_Vehicle(int Team); //Can this team build vehicles
GameObject *Find_Closest_Preset_By_Team(int Team,const Vector3 &pos,const char *Preset);
//Find the closest object of this preset to this position
GameObject *Find_Random_Preset_By_Team(int Team,const char *Preset); //Find a random
object on this team with this preset
void Disable_All_Ground_Vehicles(unsigned int team); //Removes all ground vehicles from this
teams PT menu
void Disable_All_Flying_Vehicles(unsigned int team); //Removes all flying vehicles from this
teams PT menu
int Get_Object_Count(int Team,const char *Preset); //Get the number of objects that exist with this
preset
bool Is_Valid_Preset_ID(unsigned long ID); //Is this a valid preset id
bool Is_Valid_Preset(const char *Preset); //Is this a valid preset name
GameObject *Find_Building_With_Script(int Team,int Type,const char *Script,GameObject
*Caller); //Find a building of this type with this script attached to it
unsigned int Get_Vehicle_Definition_Mode_By_ID(unsigned long ID); //Get the mode of a vehicle
```

given its preset ID

```
SimpleDynVecClass<GameObject*> Get_All_Objects_By_Preset(int Team,const char *Preset);  
//Returns an array of all objects of this preset
```

```
void Damage_All_Objects_In_Zone(GameObject *Zone,float Damage,const char  
*Warhead,GameObject *Damager); //Damages all objects in this zone
```

New script JFW_Change_Radar_Map which is used to change the scrolling radar map parameters on a per-map basis

New script JFW_Poke_Send_Custom_Toggle which is like JFW_Poke_Send_Custom_2 but sends one custom then another custom then the first one again and so on.

New script JFW_Teleport_Zone_Enable. This is like JFW_Teleport_Zone_Team but is turned on and off by a custom.

Fix to the LFDS script patcher to correctly patch the RH8 binary.

New scripts by Kamuix:

Kamuix_Announce_Preset_Buy

This is like JFW_Preset_Buy Except when bought, It will be announced when purchased. If you do not have enough money

to purchase It you will be paged by the console telling you that you do not have enough credits.

Kamuix_PAMSG_Zone

This is like JFW_PPAGE_Zone but instead of paging the player it just sends an individual AdminMsg to a player.

Kamuix_Send_FDS_MSG_Zone

On zone entry, this sends a specified message to the console input parser.

Kamuix_Team_Change_Zone

Changes a players team of a team specified on zone entry.

Kamuix_Damaged_Send_MsgFds

When the object this is attached to gets damaged, this sends a specified message to the console input parser.

Kamuix_Death_Send_MsgFds

When the object this is attached to gets destroyed, this sends a specified message to the console input parser.

Kamuix_Death_Announce

When the object this is attached to dies, A message gets displayed telling all who destroyed what.

*per-player Set_Fog_Enable, Set_Fog_Range and Enable_Stealth

Improvements to the code for changing PTs (specifically in the way it restores the PT data at the start of the level)

New scripts from WD:

MDB_Water_Zone

MDB_Water_Unit

Scripts to make amphibious APCs, hover MRLSs and other such units that can go over water work.

*MDB_Vehicle_Limit Script for implementing a vehicle limit if you arent using the new vehicle factory logic

MDB_Mine_Limit Script to handle changing the mine limit

*MDB_Unit_Limit Script to implement a unit limit (e.g. for hero units)

*MDB_Send_Custom_On_Key Script to send a custom when a key is pressed

*MDB_Remote_Controlled_Vehicle

*MDB_Remote_Controlled_Vehicle_Bot

Scripts for a remote controled vehicle (could be used for a terror drone or a hunter-seeker I)

- *MDB_ExpVehFac_Helipad
- *MDB_ExpVehFac_Naval
- *MDB_ExpVehFac_Vehicle_Factory
- *MDB_ExpVehFac_Cinematic
- *MDB_ExpVehFac_Cinematic_Vehicle
- *MDB_ExpVehFac_Limit

Scripts that implement a fully working helipad and naval yard setup. Now you can have helipads and/or naval yards and not only that, you can purchase the flying or naval vehicles from any normal PT.
