

---

Subject: Westwood Scripts [M02]

Posted by [Troll King](#) on Thu, 28 Jun 2012 10:54:26 GMT

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

---

Greetings all people and happy modders of Renegade.

(If you wish to see the link right now, because you can't wait to see the scripts. Go down to the bottom of the message.)

A few weeks ago I had a thought in my mind to Decompile the original Westwood scripts.

HOWEVER: I do not claim this is an EXACT copy of the scripts. I've tried to get them as close as I possibly could. I'm still a newb at doing this. Also, I couldn't identify some Script Events (like Created/Custom etc.) and some local class variables. I left their names as Unknown\*\*. I left some comments in the scripts, also at the Unknown events for hints to discover what event it could be.

Commands->Find\_Object is often used and I rarely check what object it is looking for so I just named those GameObject \*SearchObj\*\*.

DO NOT TRUST THE ActionParamsStructs COMPLETELY BECAUSE IT IS VERY HARD FOR ME AT THIS TIME TO KNOW EXACTLY HOW IT WAS USED, IT'S ONLY A CLOSE REFLECTION HOW IT MUST HAVE LOOKED LIKE.

It is true that all the names of variables are made up by me. All their names are lost FOREVER and are most likely NOT the same as the original. Although all strings are EXACTLY the same as the original.

This release contains the following scripts:

- M02\_Encyclopedia\_Reveal
- M02\_Data\_Disk
- M02\_Commando\_Start
- M02\_GDI\_Helicopter
- M02\_Nod\_Jet\_Waypath
- M02\_Nod\_Jet
- M02\_Mendoza
- M02\_Destroy\_Vehicle
- M02\_Approach\_Vehicle
- M02\_Player\_Vehicle
- M02\_Nod\_Vehicle
- M02\_Nod\_Apache
- M02\_Stationary\_Vehicle
- M02\_GDI\_Soldier
- M02\_Nod\_Convoy\_Truck
- M02\_Destroy\_Objective
- M02\_Helipad
- M02\_Dam\_MCT
- M02\_Power\_Plant
- M02\_Obelisk
- M02\_Nod\_Soldier

M02\_Reset\_Spawn  
M02\_Objective\_Zone  
M02\_Objective\_Controller

I believe these are ALL M02 scripts besides 'M02\_Respawn\_Controller'. I didn't decompile this one just yet because there are too many things in there I don't understand just yet.

I only ask these things from you:

Please do not claim that this release was yours (feel free to decompile them yourself, but do not use my link as yours).

Please do give me feedback about this, these scripts only have cost me alot of time. I'm currently working on the MX0 scripts aswell.

Happy Modding!

Q: So why the M02 scripts you ask?

A: It was a request from zunnie.

Q: Do all the numbers in there resemble the original numbers?

A: Most likely yes, but I cannot guarantee that.

Q: What help text number is this from the Commands->Set\_HUD\_Help\_Text?

A: I have not a clue, nor do I know what all other numbers resemble for example from Create\_Logical\_Sound.

Q: What does Debug\_Message?

A: Dunno.

Q: Can I ask you something about M02\_Respawn\_Controller?

A: Sure you can. My preference goes to IRC. I'm usually in #mpf from MultiPlayerForums and #tt from BlackIntel.

Special thanks to JonWil, for helping me out by indicating what program I should use etc.

IMPORTANT NOTE: NONE OF THE NEGATIVE FLOATS ARE CORRECT, HAVE TO LOOK INTO IT HOW TO FIX IT!

HERE COMES THE ALMIGHTY LINK TO THEM!! @ @#%%^&

<http://pastebin.com/iC0FtbdD>

File Attachments

1) [Neijwiert\\_M02\\_Scripts.h](#), downloaded 356 times

---

---

Subject: Re: Westwood Scripts [M02]  
Posted by [iRANian](#) on Thu, 28 Jun 2012 19:04:11 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

did you check what 'Commands->' function is being called by comparing the offset being called with the offsets that are loaded up in Get\_Script\_Commands()? what tricks/programs did you use

---

Subject: Re: Westwood Scripts [M02]  
Posted by [iRANian](#) on Thu, 28 Jun 2012 19:15:51 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

that's pretty impressive btw

---

Subject: Re: Westwood Scripts [M02]  
Posted by [Troll King](#) on Fri, 29 Jun 2012 09:09:07 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Yes I did that. The actual Definition of Script\_Commands is in TT.dll and Scripts2.dll is calling from that aswell. so it most likely has scripts.h too or something like that and it would call upon a link up to TT.dll. An example how Commands->Get\_Position(); would look like: `*(int (__cdecl **)(char *, int))(dword_101D3700 + 64))(&v63, a2);`

where dword\_101D3700 would be the memory position of the beginning of Script\_Command\_Class and + 64 the offset. I'm using IDA Pro V6.1 with Hex-Rays V1.5.

And thanks

---

Subject: Re: Westwood Scripts [M02]  
Posted by [iRANian](#) on Fri, 29 Jun 2012 12:22:11 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

ah yeah that's how i'd do it, was kinda hoping you figured out how to automate it so you don't have to look up the offsets every time a ScriptsCommands function get called.

---

Subject: Re: Westwood Scripts [M02]  
Posted by [Troll King](#) on Fri, 29 Jun 2012 15:05:03 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

I made this list for it, if anybody needs it.

```
dword_101D3700 + 8 = void Debug_Message(char *format, ...);  
dword_101D3700 + 12 = void Action_Reset(GameObject *obj, float priority);  
dword_101D3700 + 16 = void Action_Goto(GameObject *obj, const ActionParamsStruct
```

```

&params);
dword_101D3700 + 20 = void Action_Attack(GameObject *obj, const ActionParamsStruct
&params);
dword_101D3700 + 24 = void Action_Play_Animation(GameObject *obj, const
ActionParamsStruct &params);
dword_101D3700 + 28 = void Action_Enter_Exit(GameObject (GameObject *obj, const
ActionParamsStruct &params);
dword_101D3700 + 32 = void Action_Face_Location(GameObject *obj, const ActionParamsStruct
&params)
dword_101D3700 + 36 = void Action_Dock(GameObject *obj, const ActionParamsStruct
&params);
dword_101D3700 + 40 = void Action_Follow_Input(GameObject *obj, const ActionParamsStruct
&params);
dword_101D3700 + 44 = void Modify_Action(GameObject *obj, int ActionID, const
ActionParamsStruct &params, bool modify_move, bool modify_attack);
dword_101D3700 + 48 = int Get_ActionID(GameObject *obj);
dword_101D3700 + 52 = bool Get_Action_Params(GameObject *obj, ActionParamsStruct
&params);
dword_101D3700 + 56 = bool Is_Performing_Pathfind_Action(GameObject *obj);
dword_101D3700 + 60 = void Set_Position(GameObject *obj, const Vector3 &position);
dword_101D3700 + 64 = Vector3 Get_Position(GameObject *obj);
dword_101D3700 + 68 = Vector3 Get_Bone_Position(GameObject *obj, const char *bone_name);
dword_101D3700 + 72 = float Get_Facing(GameObject *obj);
dword_101D3700 + 76 = void Set_Facing(GameObject *obj, float degrees);
dword_101D3700 + 80 = void Disable_All_Collisions(GameObject *obj);
dword_101D3700 + 84 = void Disable_Physical_Collisions(GameObject *obj);
dword_101D3700 + 88 = void Enable_Collisions(GameObject *obj);
dword_101D3700 + 92 = void Destroy_Object(GameObject *obj);
dword_101D3700 + 96 = GameObject *Find_Object(int obj_id);
dword_101D3700 + 100 = GameObject *Create_Object(const char *type_name, const Vector3
&position);
dword_101D3700 + 104 = GameObject *Create_Object_At_Bone(GameObject *host_obj, const
char *new_obj_type_name, const char *bone_name);
dword_101D3700 + 108 = int Get_ID(GameObject *obj);
dword_101D3700 + 112 = int Get_Preset_ID(GameObject *obj);
dword_101D3700 + 116 = const char *Get_Preset_Name(GameObject *obj);
dword_101D3700 + 120 = void Attach_Script(GameObject *object, const char *scriptName, const
char *scriptParams);
dword_101D3700 + 124 = void Add_To_Dirty_Cull_List(GameObject *object);
dword_101D3700 + 128 = void Start_Timer(GameObject *obj, ScriptClass *script, float duration,
int timer_id);
dword_101D3700 + 132 = void Trigger_Weapon(GameObject *obj, bool trigger, const Vector3
&target, bool primary);
dword_101D3700 + 136 = void Select_Weapon(GameObject *obj, const char *weapon_name);
dword_101D3700 + 140 = void Send_Custom_Event(GameObject *from, GameObject *to, int
type, int param, float delay);
dword_101D3700 + 144 = void Send_Damaged_Event(GameObject *obj, GameObject
*damager);

```

dword\_101D3700 + 148 = float Get\_Random(float min, float max);  
dword\_101D3700 + 152 = int Get\_Random\_Int(int min, int max);  
dword\_101D3700 + 156 = GameObject \*Find\_Random\_Simple\_Object(const char \*preset\_name);  
dword\_101D3700 + 160 = void Set\_Model(GameObject \*obj, const char \*model\_name);  
dword\_101D3700 + 164 = void Set\_Animation(GameObject \*obj, const char \*anim\_name, bool looping, const char \*sub\_obj\_name, float start\_frame, float end\_frame, bool is\_blended);  
dword\_101D3700 + 168 = void Set\_Animation\_Frame(GameObject \*obj, const char \*anim\_name, int frame);  
dword\_101D3700 + 172 = int Create\_Sound(const char \*sound\_preset\_name, const Vector3 &position, GameObject \*creator);  
dword\_101D3700 + 176 = int Create\_2D\_Sound(const char \*sound\_preset\_name);  
dword\_101D3700 + 180 = int Create\_2D\_WAV\_Sound(const char \*wav\_filename);  
dword\_101D3700 + 184 = int Create\_3D\_WAV\_Sound\_At\_Bone(const char \*wav\_filename, GameObject \*obj, const char \*bone\_name);  
dword\_101D3700 + 188 = int Create\_3D\_Sound\_At\_Bone(const char \*sound\_preset\_name, GameObject \*obj, const char \*bone\_name);  
dword\_101D3700 + 192 = int Create\_Logical\_Sound(GameObject \*creator, int type, const Vector3 &position, float radius);  
dword\_101D3700 + 196 = void Start\_Sound(int sound\_id);  
dword\_101D3700 + 200 = void Stop\_Sound(int sound\_id, bool destroy\_sound);  
dword\_101D3700 + 204 = void Monitor\_Sound(GameObject \*game\_obj, int sound\_id);  
dword\_101D3700 + 208 = void Set\_Background\_Music(const char \*wav\_filename);  
dword\_101D3700 + 212 = void Fade\_Background\_Music(const char \*wav\_filename, int fade\_out\_time, int fade\_in\_time);  
dword\_101D3700 + 216 = void Stop\_Background\_Music();  
dword\_101D3700 + 220 = float Get\_Health(GameObject \*obj);  
dword\_101D3700 + 224 = float Get\_Max\_Health(GameObject \*obj);  
dword\_101D3700 + 228 = void Set\_Health(GameObject \*obj, float health);  
dword\_101D3700 + 232 = float Get\_Shield\_Strength(GameObject \*obj);  
dword\_101D3700 + 236 = float Get\_Max\_Shield\_Strength(GameObject \*obj);  
dword\_101D3700 + 240 = void Set\_Shield\_Strength(GameObject \*obj, float strength);  
dword\_101D3700 + 244 = void Set\_Shield\_Type(GameObject \*obj, const char \*name);  
dword\_101D3700 + 248 = int Get\_Player\_Type(GameObject \*obj);  
dword\_101D3700 + 252 = void Set\_Player\_Type(GameObject \*obj, int type);  
dword\_101D3700 + 256 = float Get\_Distance(const Vector3 &p1, const Vector3 &p2);  
dword\_101D3700 + 260 = void Set\_Camera\_Host(GameObject \*obj);  
dword\_101D3700 + 264 = void Force\_Camera\_Look(const Vector3 \*target);  
dword\_101D3700 + 268 = GameObject \*Get\_The\_Star();  
dword\_101D3700 + 272 = GameObject \*Get\_A\_Star(const Vector3 &pos);  
dword\_101D3700 + 276 = GameObject \*Find\_Closest\_Soldier(const Vector3 &pos, float min\_dist, float max\_dist, bool only\_human);  
dword\_101D3700 + 280 = bool Is\_A\_Star(GameObject \*obj);  
dword\_101D3700 + 284 = void Control\_Enable(GameObject \*obj, bool enable);  
dword\_101D3700 + 288 = const char \*Get\_Damage\_Bone\_Name();  
dword\_101D3700 + 292 = bool Get\_Damage\_Bone\_Direction();  
dword\_101D3700 + 296 = bool Is\_Object\_Visible(GameObject \*looker, GameObject \*obj);  
dword\_101D3700 + 300 = void Enable\_Enemy\_Seen(GameObject \*obj, bool enable);

dword\_101D3700 + 304 = void Set\_Display\_Color(unsigned char red, unsigned char green, unsigned char blue);  
dword\_101D3700 + 308 = void Display\_Text(int string\_id);  
dword\_101D3700 + 312 = void Display\_Float(float value, const char \*format);  
dword\_101D3700 + 316 = void Display\_Int(int value, const char \*format);  
dword\_101D3700 + 320 = void Save\_Data(ScriptSave &saver, int id, int size, void \*data);  
dword\_101D3700 + 324 = void Save\_Pointer(ScriptSave &saver, int id, void \*pointer);  
dword\_101D3700 + 328 = bool Load\_Begin(ScriptLoader &loader, int \*id);  
dword\_101D3700 + 332 = void Load\_Data(ScriptLoader &loader, int size, void \*data);  
dword\_101D3700 + 336 = void Load\_Pointer(ScriptLoader &loader, void \*\*pointer);  
dword\_101D3700 + 340 = void Load\_End(ScriptLoader &loader);  
dword\_101D3700 + 344 = void Begin\_Chunk(ScriptSaver &saver, unsigned int chunkID);  
dword\_101D3700 + 348 = void End\_Chunk(ScriptSaver &saver);  
dword\_101D3700 + 352 = bool Open\_Chunk(ScriptLoader &loader, unsigned int \*chunkID);  
dword\_101D3700 + 356 = void Close\_Chunk(ScriptLoader &loader);  
dword\_101D3700 + 360 = void Clear\_Radar\_Markers();  
dword\_101D3700 + 364 = void Clear\_Radar\_Marker(int id);  
dword\_101D3700 + 368 = void Add\_Radar\_Marker(int id, const Vector3 &position, int shape\_type, int color\_type);  
dword\_101D3700 + 372 = void Set\_Obj\_Radar\_Blip\_Shape(GameObject \*obj, int shape\_type);  
dword\_101D3700 + 376 = void Set\_Obj\_Radar\_Blip\_Color(GameObject \*obj, int color\_type);  
dword\_101D3700 + 380 = void Enable\_Radar(bool enable);  
dword\_101D3700 + 384 = void Clear\_Map\_Cell(int cell\_x, int cell\_y);  
dword\_101D3700 + 388 = void Clear\_Map\_Cell\_By\_Pos(const Vector3 &world\_space\_pos);  
dword\_101D3700 + 392 = void Clear\_Map\_Cell\_By\_Pixel\_Pos(int pixel\_pos\_x, int pixel\_pos\_y);  
dword\_101D3700 + 396 = void Clear\_Map\_Region\_By\_Pos(const Vector3 &world\_space\_pos, int pixel\_radius);  
dword\_101D3700 + 400 = void Reveal\_Map();  
dword\_101D3700 + 404 = void Shroud\_Map();  
dword\_101D3700 + 408 = void Show\_Player\_Map\_Marker(bool onoff);  
dword\_101D3700 + 412 = float Get\_Safe\_Flight\_Height(float x\_pos, float y\_pos);  
dword\_101D3700 + 416 = void Create\_Explosion(const char \*explosion\_def\_name, const Vector3 &pos, GameObject \*creator);  
dword\_101D3700 + 420 = void Create\_Explosion\_At\_Bone(const char \*explosion\_def\_name, Gameobject \*object, const char \*bone, GameObject \*creator);  
dword\_101D3700 + 424 = void Enable\_HUD(bool enable);  
dword\_101D3700 + 428 = void Mission\_Complete(bool success);  
dword\_101D3700 + 432 = void Give\_PowerUp(GameObject \*obj, const char \*preset\_name, bool display\_on\_hud);  
dword\_101D3700 + 436 = void Innate\_Disable(GameObject \*object);  
dword\_101D3700 + 440 = void Innate\_Enable(GameObject \*object);  
dword\_101D3700 + 444 = bool Innate\_Soldier\_Enable\_Enemy\_Seen(GameObject \*obj, bool state);  
dword\_101D3700 + 448 = bool Innate\_Soldier\_Enable\_Gunshot\_Heard(GameObject \*obj, bool state);  
dword\_101D3700 + 452 = bool Innate\_Soldier\_Enable\_Footsteps\_Heard(GameObject \*obj, bool state);  
dword\_101D3700 + 456 = bool Innate\_Soldier\_Enable\_Bullet\_Heard(GameObject \*obj, bool

```

state);
dword_101D3700 + 460 = bool Innate_Soldier_Enable_Actions(GameObject *obj, bool state);
dword_101D3700 + 464 = void Set_Innate_Soldier_Home_Location(GameObject *obj, const
Vector3 &home_pos, float home_radius);
dword_101D3700 + 468 = void Set_Innate_Aggressiveness(GameObject *obj, float
aggressiveness);
dword_101D3700 + 472 = void Set_Innate_Take_Cover_Probability(GameObject *obj, float
probability);
dword_101D3700 + 476 = void Set_Innate_Is_Stationary(GameObject *obj, bool stationary);
dword_101D3700 + 480 = void Innate_Force_State_Bullet_Heard(GameObject *obj, const
Vector3 &pos);
dword_101D3700 + 484 = void Innate_Force_State_Footsteps_Heard(GameObject *obj, const
Vector3 &pos);
dword_101D3700 + 488 = void Innate_Force_State_Gunshots_Heard(GameObject *obj, const
Vector3 &pos);
dword_101D3700 + 492 = void Innate_Force_State_Enemy_Seen(GameObject *obj, GameObject
*enemy);
dword_101D3700 + 496 = void Static_Anim_Phys_Goto_Frame(int obj_id, float frame, const char
*anim_name);
dword_101D3700 + 500 = void Static_Anim_Phys_Goto_Last_Frame(int obj_id, const char
*anim_name);
dword_101D3700 + 504 = unsigned int Get_Sync_Time();
dword_101D3700 + 508 = void Add_Objective(int id, int type, int status, int short_description_id,
char *description_sound_filename, int long_description_id);
dword_101D3700 + 512 = void Remove_Objective(int id);
dword_101D3700 + 516 = void Set_Objective_Status(int id, int status);
dword_101D3700 + 520 = void Change_Objective_Type(int id, int type);
dword_101D3700 + 524 = void Set_Objective_Radar_Blip(int id, const Vector3 &position);
dword_101D3700 + 528 = void Set_Objective_Radar_Blip_Object(int id, ScriptableGameObj
*unit);
dword_101D3700 + 532 = void Set_Objective_HUD_Info(int id, float priority, const char
*texture_name, int message_id);
dword_101D3700 + 536 = void Set_Objective_HUD_Info_Position(int id, float priority, const char
*texture_name, int message_id);
dword_101D3700 + 540 = void Shake_Camera(const Vector3 &pos, float radius, float intensity,
float duration);
dword_101D3700 + 544 = void Enable_Spawner(int id, bool enable);
dword_101D3700 + 548 = GameObject *Trigger_Spawner(int id);
dword_101D3700 + 552 = void Enable_Engine(GameObject *object, bool onoff);
dword_101D3700 + 556 = int Get_Difficulty_Level();
dword_101D3700 + 560 = void Grant_Key(GameObject *object, int key, bool grant);
dword_101D3700 + 564 = bool Has_Key(GameObject *object, int key, bool grant);
dword_101D3700 + 568 = void Enable_Hibernation(GameObject *object, bool enable);
dword_101D3700 + 572 = void Attach_To_Object_Bone(GameObject *object, GameObject
*host_object, const char *bone_name);
dword_101D3700 + 576 = int Create_Conversation(const char *conversation_name, int priority,
float max_dist, bool is_interruptable);
dword_101D3700 + 580 = void Join_Conversation(GameObject *object, int

```

```

active_conversation_id, bool allow_mode, bool allow_head_turn, bool allow_facing);
dword_101D3700 + 584 = void Join_Conversation_Facing(GameObject *object, int
active_conversation_id, int obj_id_to_face);
dword_101D3700 + 588 = void Start_Conversation(int active_conversation_id, int ActionID);
dword_101D3700 + 592 = void Monitor_Conversation(GameObject *object, int
active_conversation_id);
dword_101D3700 + 596 = void Start_Random_Conversation(GameObject *object);
dword_101D3700 + 600 = void Stop_Conversation(int active_conversation_id);
dword_101D3700 + 604 = void Stop_All_Conversations();
dword_101D3700 + 608 = void Lock_Soldier_Facing(GameObject *object, GameObject
*object_to_face, bool turn_body);
dword_101D3700 + 612 = void Unlock_Soldier_Facing(GameObject *object);
dword_101D3700 + 616 = void Apply_Damage(GameObject *object, float amount, const char
*warhead_name, GameObject *damager);
dword_101D3700 + 620 = void Set_Loiters_Allowed(GameObject *object, bool allowed);
dword_101D3700 + 624 = void Set_Is_Visible(GameObject *object, bool visible);
dword_101D3700 + 628 = void Set_Is_Rendered(GameObject *object, bool rendered);
dword_101D3700 + 632 = float Get_Points(GameObject *object);
dword_101D3700 + 636 = void Give_Points(GameObject *object, float points, bool entire_team)
dword_101D3700 + 640 = float Get_Money(GameObject *object);
dword_101D3700 + 644 = void Give_Money(GameObject *object, float money, bool entire_team);
dword_101D3700 + 648 = bool Get_Building_Power(GameObject *object);
dword_101D3700 + 652 = void Set_Building_Power(GameObject *object, bool onoff);
dword_101D3700 + 656 = void Play_Building_Announcement(GameObject *object, int text_id);
dword_101D3700 + 660 = GameObject *Find_Nearest_Building_To_Pos(const Vector3 &position,
const char *mesh_prefix);
dword_101D3700 + 664 = GameObject *Find_Nearest_Building(GameObject *object, const char
*mesh_prefix);
dword_101D3700 + 668 = int Team_Member_In_Zone(GameObject *object, int player_type);
dword_101D3700 + 672 = void Set_Clouds(float cloudcover, float cloudgloominess, float
ramptime)
dword_101D3700 + 676 = void Set_Lightning(float intensity, float startdistance, float enddistance,
float heading, float distribution, float ramptime);
dword_101D3700 + 680 = void Set_War_Blitz(float intensity, float startdistance, float enddistance,
float heading, float distribution, float ramptime);
dword_101D3700 + 684 = void Set_Wind(float heading, float speed, float variability, float
ramptime);
dword_101D3700 + 688 = void Set_Rain(float density, float ramptime, bool prime);
dword_101D3700 + 692 = void Set_Snow(float density, float ramptime, bool prime);
dword_101D3700 + 696 = void Set_Ash(float density, float ramptime, bool prime);
dword_101D3700 + 700 = void Set_Fog_Enable(bool enabled);
dword_101D3700 + 704 = void Set_Fog_Range(float startdistance, float enddistance, float
ramptime);
dword_101D3700 + 708 = void Enable_Stealth(GameObject *object, bool onoff);
dword_101D3700 + 712 = void Cinematic_Sniper_Control(bool enabled, float zoom);
dword_101D3700 + 716 = int Text_File_Open(const char *filename);
dword_101D3700 + 720 = bool Text_File_Get_String(int handle, char *buffer, int size);
dword_101D3700 + 724 = void Text_File_Close(int handle);

```

dword\_101D3700 + 728 = void Enable\_Vehicle\_Transitions(GameObject \*object, bool enable);  
dword\_101D3700 + 732 = void Display\_GDI\_Player\_Terminal();  
dword\_101D3700 + 736 = void Display\_NOD\_Player\_Terminal();  
dword\_101D3700 + 740 = void Display\_Mutant\_Player\_Terminal();  
dword\_101D3700 + 744 = bool Reveal\_Encyclopedia\_Character(int object\_id);  
dword\_101D3700 + 748 = bool Reveal\_Encyclopedia\_Weapon(int object\_id);  
dword\_101D3700 + 752 = bool Reveal\_Encyclopedia\_Vehicle(int object\_id);  
dword\_101D3700 + 756 = bool Reveal\_Encyclopedia\_Building(int object\_id);  
dword\_101D3700 + 760 = void Display\_Encyclopedia\_Event\_UI();  
dword\_101D3700 + 764 = void Scale\_AI\_Awareness(float sight\_scale, float hearing\_scale);  
dword\_101D3700 + 768 = void Enable\_Cinematic\_Freeze(GameObject \*object, bool enable);  
dword\_101D3700 + 772 = void Expire\_Powerup(GameObject \*object);  
dword\_101D3700 + 776 = void Set\_HUD\_Help\_Text(int string\_id, const Vector3 color);  
dword\_101D3700 + 780 = void Enable\_HUD\_Pokable\_Indicator(GameObject \*object, bool enable);  
dword\_101D3700 + 784 = void Enable\_Innate\_Conversations(GameObject \*object, bool enable);  
dword\_101D3700 + 788 = void Display\_Health\_Bar(GameObject \*object, bool display);  
dword\_101D3700 + 792 = void Enable\_Shadow(GameObject \*object, bool enable);  
dword\_101D3700 + 808 = void Set\_Screen\_Fade\_Color(float r, float g, float b, float seconds);  
dword\_101D3700 + 812 = void Set\_Screen\_Fade\_Opacity(float opacity, float seconds);

---

---

Subject: Re: Westwood Scripts [M02]  
Posted by [iRANian](#) on Fri, 29 Jun 2012 15:05:49 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

If you open the linux fds it'll show you more of the symbols

---

---

Subject: Re: Westwood Scripts [M02]  
Posted by [Troll King](#) on Fri, 06 Jul 2012 12:37:57 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

FILE ADDED

---