Subject: [REQUEST]Sound on turret rotate
Posted by Generalcamo on Fri, 21 Dec 2012 04:50:10 GMT
View Forum Message <> Reply to Message

Hey, I am trying to create a turret rotate sound, but the only one I can find, RA2VEN_TurretSound, is giving me heaps of problems. Could the TT team potentially create another, easier to use script please?

Subject: Re: [REQUEST]Sound on turret rotate
Posted by StealthEye on Fri, 21 Dec 2012 06:32:07 GMT
View Forum Message <> Reply to Message

Sorry, that is not a priority right now. If someone writes one we can add it, but that's it.

Subject: Re: [REQUEST]Sound on turret rotate
Posted by danpaul88 on Fri, 21 Dec 2012 16:22:44 GMT
View Forum Message <> Reply to Message

What problems are you having with it? Perhaps you're just using it incorrectly?

Subject: Re: [REQUEST]Sound on turret rotate
Posted by Generalcamo on Tue, 25 Dec 2012 00:30:33 GMT
View Forum Message <> Reply to Message

Mostly that it doesn't sound the sound I put into it. I know I used it right, because I temporarily set it to a visible daves arrow to see if it attaches, and it does.

Subject: Re: [REQUEST]Sound on turret rotate
Posted by danpaul88 on Tue, 25 Dec 2012 19:56:42 GMT
View Forum Message <> Reply to Message

Can you list the parameters you are using with the script? Looking at the code it should work, in theory.

Subject: Re: [REQUEST]Sound on turret rotate
Posted by Generalcamo on Wed, 26 Dec 2012 00:37:45 GMT
View Forum Message <> Reply to Message

test_sam
Type: Mounted AI Defense

Scripts:
ra2ven_TurretSound
Parameter:
TurretSoundObj: SAM_TURRET

SAM_TURRET
Type: Daves Arrow
Scripts:
RA2Ven_TurretSoundObj
Parameters
RotationAngleLimit: 572957
Rotate3dSound:Turret_Sound
Explosion_Preset: Blamo_Killer
Time: 1.00
Timer Number: 0
Wav_Length: 2.00
Timer Number2: 0

Subject: Re: [REQUEST]Sound on turret rotate
Posted by danpaul88 on Wed, 26 Dec 2012 11:32:00 GMT
View Forum Message <> Reply to Message

There's your problem. TimerNumber and TimerNumber2 must be >0 and different from each other.

Oh, and "RotationAngleLimit" doesn't do anything apparently. Good eh?

Subject: Re: [REQUEST]Sound on turret rotate
Posted by Generalcamo on Thu, 27 Dec 2012 01:44:40 GMT
View Forum Message <> Reply to Message

danpaul88 wrote on Wed, 26 December 2012 06:32There's your problem. TimerNumber and TimerNumber2 must be >0 and different from each other.

Oh, and "RotationAngleLimit" doesn't do anything apparently. Good eh?

Tried that, it didn't work. Plus, the documentation specifically states not to set them unless you have other timers in your map, and you know exactly what you are doing.

Subject: Re: [REQUEST]Sound on turret rotate
Posted by danpaul88 on Thu, 27 Dec 2012 14:41:46 GMT
View Forum Message <> Reply to Message

having read the code, the documentation is wrong

Subject: Re: [REQUEST]Sound on turret rotate
Posted by Omar007 on Thu, 27 Dec 2012 14:48:06 GMT
View Forum Message <> Reply to Message

TBH the timer system in Renegade sucks. I vote for improvement (if it is possible to change it)
Doing that would break existing scripts though :/

Subject: Re: [REQUEST]Sound on turret rotate
Posted by danpaul88 on Thu, 27 Dec 2012 22:13:35 GMT
View Forum Message <> Reply to Message

Actually... I'm not even sure timers NEED unique numbers... custom messages do because all scripts receive them, but timers pass a pointer to the calling script into the timer and I believe only that script gets notified when the timer expires.

As for saying they suck... why exactly do they suck? I find them very flexible and haven't found anything I can't do with either a timer or a custom.

Subject: Re: [REQUEST]Sound on turret rotate
Posted by Omar007 on Fri, 28 Dec 2012 01:35:12 GMT
View Forum Message <> Reply to Message

It does allow you to do whatever you want and it's more of a personal preference but I'd like to be able to give a function pointer instead of a number so that that function is executed when the timer ends instead of the Timer_Expired function where you have to check the number.

So kinda like

```
//Ampersand is actually optional but just added it to make it obvious that I'm giving the address of the function to the timer
Commands->Start_Timer(obj, this, 0.5f, &myFunction);

public void myFunction()
{
   //Do stuff
}
```

It'd also make it readable what the job of the timer is by reading the name of the function attached to it. (assuming the programmer named it 'harvesterCheckTimer' and not 'timer123' or something)

But as I said it is more of a personal thing. The system does not suck in that it is limited or anything. It's just that it could be done in a more useable manner.
Scripts wouldn't ever need custom timer numbers to be set in LE.
You would never have the problem of Script A and Script B both running a timer with number 1337 on the same object. (if Timer_Expired is indeed only called on the calling script and not on

the object, you don't have this problem anyway; it has been a while so I'm not sure which one it was)

Subject: Re: [REQUEST]Sound on turret rotate
Posted by danpaul88 on Fri, 28 Dec 2012 19:17:02 GMT
View Forum Message <> Reply to Message

It helps if you used named constants for your timer numbers, instead of magic numbers. For example;

```
void dp88_RemoteControlConsole::Timer_Expired ( GameObject *obj, int number )
{
 if ( number == TIMER_REMOTECONTROL_TIMEOUT && vehicleID == 0 )
 {
  <code>
 }


 else if ( number == TIMER_REMOTECONTROL_DRIVERENTER )
 {
  <code>
 }


 // This can be called for the vehicle being destroyed OR the console becoming disabled
 else if ( number == TIMER_REMOTECONTROL_DRIVEREXIT )
 {
  <code>
 }


 // Count down tick for charge time
 else if ( number == TIMER_REMOTECONTROL_CHARGETICK )
 {
  <code>
 }
}
```

The above is actually very read-able and you could call seperate functions to handle each if/elseif case. You could alternatively use a timed custom and pass the address of a function to be called when the custom is fired if you really wanted to.

This also works for custom messages, in cases where you don't need them to be user-controllable (because the script should't be attached to the same object twice anyway). For instance;

```
void dp88_buildingScripts_baseClass::Custom ( GameObject *obj, int type, int param,
GameObject *sender )
{
  if ( (m_parentId == -1 && obj == sender)
    || (m_parentId != -1 && Commands->Get_ID(sender) == m_parentId ) )
  {
    if ( type == CUSTOM_BUILDINGSCRIPTS_BUILDINGOFFLINE )
      return OnBuildingOffline(obj);
    if ( type == CUSTOM_BUILDINGSCRIPTS_BUILDINGONLINE )
      return OnBuildingOnline(obj);
    if ( type == CUSTOM_BUILDINGSCRIPTS_BUILDINGCAPTURED )
      return OnBuildingCaptured(obj, param);
    if ( type == CUSTOM_BUILDINGSCRIPTS_BUILDINGDESTROYED )
    {
      OnBuildingDestroyed(obj);
      m_parentId = -1;
      return;
    }
  }

  OnCustom(obj,type,param,sender);
}
```

Subject: Re: [REQUEST]Sound on turret rotate
Posted by danpaul88 on Fri, 28 Dec 2012 19:28:38 GMT
View Forum Message <> Reply to Message

UPDATE;

Looking at the code in the script, I think it has a flaw in the way it works. Regardless of which timer fires (Timer1 or Timer2) it restarts Timer1, which means you end up with multiple trigger instances for Timer1 running concurrently... this shouldn't stop the script working, but it will add a lot of unnecessary overhead.

The other bug in the script is that it assumes the sound object has a bone named "turret", despite the fact that the object itself is attached to the parent vehicles turret bone. That will either make it play the sound at the origin position (good) or not at all (bad).

I'm not sure whether it's a good idea to modify the script at this time since another map or mod might have it up and running... then again, since it was written for a now-defunct mod (RA2:Venegence) it's possible it is unused and could be fixed.

Subject: Re: [REQUEST]Sound on turret rotate
Posted by danpaul88 on Fri, 28 Dec 2012 20:30:45 GMT

OK, so I figured the existing script had some fundamental flaws in it's implementation so I wrote a new, much simpler to use, version.

http://doc.tiberiantechnologies.org/scripts.dll/classdp88__turret_sound.html

If you want to edit scripts.dll to start using it now, before the next scripts release, here's the code. Please don't ship any .dll files to anyone with this in, since we don't want people running non-versioned (and therefore, non-debuggable) client side code, but you can use it for testing or in a server side mod.

Toggle Spoiler

dp88_custom_timer_defines.h:

```
#define DP88_TIMER                    0xDB000000
#define MISC                          0x00000000
#define TIMER_TURRETSOUND             (DP88_TIMER|MISC|0x04)          //!< Used by
dp88_turretSound to test for turret rotation
```

dp88_misc.h:

```
/*!
* \brief Turret Rotation Sound Effect
* \author Daniel Paul (danpaul88@yahoo.co.uk)
* \ingroup scripts_sound
*
* This script plays a 3d sound at a vehicles turret bone when that bone is being rotated and stops
* the sound when the rotation stops. The sound will be looped whilst the turret is being rotated.
*
* \note
*   This script uses the difference between the vehicle rotation and the turret bone rotation to
*   determine if the turret is rotating. This means simply aiming in one direction and spinning on
*   the spot will cause the sound to be played, since the turret is rotating relative to the vehicle
*
* \param Sound_Preset
*   The name of a 3D sound preset to be played whilst the turret is rotating
* \param Min_Differential_Rad
*   The minimum difference in the turret rotation, in radians, to be considered as "rotating", this
*   helps to filter out tiny movements caused by driving along uneven terrain.
*/
class dp88_turretSound : public ScriptImpClass
{
protected:
  void Created ( GameObject* pObj );
  void Timer_Expired ( GameObject* pObj, int number );
  void Custom ( GameObject* pObj, int type, int param, GameObject* pSender );

  float Get_Turret_Facing ( class RenderObjClass* pRenderObj );
  void Play_Sound ( GameObject* pObj );
  void Stop_Sound ( GameObject* pObj );

  float m_lastFacing;
  int m_nSoundId;

  /*! \name Cached Script Parameters */
  /*! @{ */
  float m_nMinDifferential;
  /*! @} */
};
```

dp88_misc.cpp:

```
#include "VehicleGameObj.h"
#include "RenderObjClass.h"

void dp88_turretSound::Created ( GameObject* pObj )
{
  if ( VehicleGameObj* vObj = pObj->As_VehicleGameObj() )
  {
    m_nMinDifferential = Get_Float_Parameter("Min_Differential_Rad");
    m_lastFacing = Get_Turret_Facing(vObj->Peek_Model());
    m_nSoundId = -1;
    Commands->Start_Timer(pObj, this, 0.5f, TIMER_TURRETSOUND );
  }
  else
  {
    Console_Output ( "[%d:%s:%s] Critical Error: This script is only compatible with vehicle game
objects. Destroying script...\n", Commands->Get_ID(pObj),
Commands->Get_Preset_Name(pObj), this->Get_Name() );
    Destroy_Script();
  }
}

// -------------------------------------------------------------------------------------------------

void dp88_turretSound::Timer_Expired ( GameObject* pObj, int number )
{
  if ( number == TIMER_TURRETSOUND )
  {
    if ( VehicleGameObj* vObj = pObj->As_VehicleGameObj() )
    {
      float newFacing = Get_Turret_Facing(vObj->Peek_Model());
      // Check if we are rotating - ignore tiny rotation amounts
      bool bRotating = ( abs(newFacing-m_lastFacing) > m_nMinDifferential );

      if ( m_nSoundId == -1 && bRotating)
        Play_Sound(pObj);

      else if ( m_nSoundId != -1 && !bRotating )
        Stop_Sound(pObj);

      m_lastFacing = newFacing;
    }

    // Restart timer - runs even whilst playing sound so we can loop an uninterrupted sound
    // if the turret is still rotating when the sound completes
    Commands->Start_Timer(pObj, this, 0.5f, TIMER_TURRETSOUND );
  }
}

// ---------------------------------------------------------------------------------------------

void dp88_turretSound::Custom ( GameObject* pObj, int type, int param, GameObject* pSender )
{
  if (type == CUSTOM_EVENT_SOUND_ENDED && param == m_nSoundId)
  {
    // We will allow the timer to stop the sound if necessary, since this might trigger
    // on the same engine tick, thus checking our facing against the previous timer
    // facing could produce a false-positive for "stopped rotating"
    Play_Sound(pObj);
  }
}

// ---------------------------------------------------------------------------------------------

float dp88_turretSound::Get_Turret_Facing ( RenderObjClass* pRenderObj )
{
  if ( pRenderObj )
  {
    Matrix3D vehicleTransform = pRenderObj->Get_Transform();
    Matrix3D transform = pRenderObj->Get_Bone_Transform("turret");
    float offset = abs(vehicleTransform.getRotationZ()-transform.getRotationZ());
    return offset;
  }

  return 0.0f;
}

// ---------------------------------------------------------------------------------------------
```

```
void dp88_turretSound::Play_Sound ( GameObject* pObj )
{
  m_nSoundId = Commands->Create_3D_Sound_At_Bone(Get_Parameter("Sound_Preset"),
pObj, "turret");
  Commands->Monitor_Sound(pObj, m_nSoundId);
}

// -------------------------------------------------------------------------------------------------

void dp88_turretSound::Stop_Sound ( GameObject* pObj )
{
  Commands->Stop_Sound(m_nSoundId,true);
  m_nSoundId = -1;
}

// -------------------------------------------------------------------------------------------------

ScriptRegistrant<dp88_turretSound> dp88_turretSound_Registrant(
  "dp88_turretSound",
  "Sound_Preset:string,"
  "Min_Differential_Rad=0.25:float"
);
```