Subject: [CODE]m00_BuildingStateSoundSpeaker sound fix
Posted by iRANian on Tue, 30 Jul 2013 22:15:44 GMT

View Forum Message <> Reply to Message

This script and its controller script handle ambient (and announcment?) sounds added to buildings. It doesn't work properly when you're not a host in online games because the scripts loop ambient sounds starting from the start of the map. If you join or rejoin after map start the sounds stop working and they will also randomly not work even if you were on a map at map start.

This code is for a replacement scripts which registers itself as the original m00_ scripts. I did some quick testing with Aircraftkiller more than a year ago and it seems to work, no guarantees though.

```
void Iran_BuildingStateSoundSpeaker::Created(GameObject *obj)
{

// Console_Output("Custom BuildingStateSoundSpeaker created\n"); // DEBUG CRAP
 destroyed = false;
 explode = true;
 GameObject *object = Commands->Find_Object(Get_Int_Parameter("BuildingController_ID"));
 if (object)
 {
  char s[18];
  sprintf(s,"%d",Commands->Get_ID(obj));
  Commands->Attach_Script(object, "JFW_BuildingStateSoundController",s);
 }
 if (Get_Int_Parameter("Frequency_Min") == -1)
 {
  Timer_Expired(obj,0);
 }
 else
 {
  float random = 
Commands->Get_Random(Get_Float_Parameter("Frequency_Min"),Get_Float_Parameter("Frequency_Max"));
  Commands->Start_Timer(obj,this,random,0);
 }
 Commands->Start_Timer(obj,this, 60.f,1);
}

void Iran_BuildingStateSoundSpeaker::Custom(GameObject *obj,int type,int param,GameObject 
*sender)
{
 switch (type)
 {
 case CUSTOM_EVENT_SOUND_ENDED:
  if (destroyed)
  {
   if (Get_Int_Parameter("Frequency_Min_Destroyed") != -1)
   {
    float min = Get_Float_Parameter("Frequency_Min_Destroyed");
    float max = Get_Float_Parameter("Frequency_Max_Destroyed");
    float frequency = Commands->Get_Random(min,max);
    Commands->Start_Timer(obj,this,frequency,0);
   }
  }
  else
  {
   if (Get_Int_Parameter("Frequency_Min") != -1)
   {
    float min = Get_Float_Parameter("Frequency_Min");
    float max = Get_Float_Parameter("Frequency_Max");
    float frequency = Commands->Get_Random(min,max);
    Commands->Start_Timer(obj,this,frequency,0);
   }
  }
  break;
 case 9026:
  if (param == 1)
  {
   Commands->Stop_Sound(sound,1);
   destroyed = true;
   Timer_Expired(obj,0);
  }
  break;
 case 9027:
  if (explode)
  {
   Vector3 v = Commands->Get_Position(obj);
   v.X += Commands->Get_Random(2.0,5.0);
   v.Y += Commands->Get_Random(2.0,5.0);
   v.Z += Commands->Get_Random(2.0,5.0);
   Commands->Create_Explosion(Get_Parameter("Explosion_Name"),v,0);
   Commands->Send_Custom_Event(obj,obj,9027,1,Commands->Get_Random(3.0,6.0));
  }
  break;
 case 9028:
  explode = false;
  break;
 case 9029:
  explode = true;
  break;
 case 9030:
  Commands->Stop_Sound(sound,1);
  Timer_Expired(obj,0);
  break;
 }
}

void Iran_BuildingStateSoundSpeaker::Timer_Expired(GameObject *obj,int number)
{
 if ( (number == 1) && (Get_Int_Parameter("Frequency_Min") == -1) )
 {
  Commands->Start_Timer(obj,this, 60.f,1);
  Commands->Stop_Sound(sound,1);
 }
 if (destroyed)
 {
  bool is3d = Get_Int_Parameter("Is_3D_Destroyed");
  Vector3 pos = Commands->Get_Position(obj);
  Vector3 offset = Get_Vector3_Parameter("Offset_Destroyed");
  pos.X += offset.X;
  pos.Y += offset.Y;
  pos.Z += offset.Z;
  offset = Get_Vector3_Parameter("Offset_Randomness_Destroyed");
  pos.X += Commands->Get_Random(offset.X,-offset.X);
  pos.Y += Commands->Get_Random(offset.Y,-offset.Y);
  pos.Z += Commands->Get_Random(offset.Z,-offset.Z);
  if (is3d)
  {
   sound = Commands->Create_Sound(Get_Parameter("Sound_Destroyed"),pos,obj);
  }
  else
  {
   sound = Commands->Create_2D_Sound(Get_Parameter("Sound_Destroyed"));
  }
 // Console_Output("Playing dead: %s\n", Get_Parameter("Sound_Destroyed"));
 }
 else
 {
  bool is3d = Get_Int_Parameter("Is_3D");
  Vector3 pos = Commands->Get_Position(obj);
  Vector3 offset = Get_Vector3_Parameter("Offset");
  pos.X += offset.X;
  pos.Y += offset.Y;
  pos.Z += offset.Z;
  offset = Get_Vector3_Parameter("Offset_Randomness");
  pos.X += Commands->Get_Random(offset.X,-offset.X);
  pos.Y += Commands->Get_Random(offset.Y,-offset.Y);
  pos.Z += Commands->Get_Random(offset.Z,-offset.Z);
  if (is3d)
  {
   sound = Commands->Create_Sound(Get_Parameter("Sound_Normal"),pos,obj);
  // Console_Output("Playing alive 3D: %s\n", Get_Parameter("Sound_Normal"));
  }
  else
  {
   sound = Commands->Create_2D_Sound(Get_Parameter("Sound_Normal"));
  // Console_Output("Playing alive 2D: %s\n", Get_Parameter("Sound_Normal"));
  }

 }
 Commands->Monitor_Sound(obj,sound);
}

void Iran_BuildingStateSoundSpeaker::Register_Auto_Save_Variables()
{
 Auto_Save_Variable(&sound,4,1);
 Auto_Save_Variable(&destroyed,1,3);
}

void Iran_BuildingStateSoundController::Created(GameObject *obj)
{
 speakerid = Get_Int_Parameter("BuildingSpeaker_ID");
// Commands->Start_Timer(obj,this,60.f,1);
}

void Iran_BuildingStateSoundController::Killed(GameObject *obj,GameObject *killer)
{
 GameObject *object = Commands->Find_Object(speakerid);
 if (object)
 {
  Commands->Send_Custom_Event(obj,object,9026,1,0);
  Commands->Send_Custom_Event(obj,object,9027,1,0);
 }
 Destroy_Script();
}


void Iran_BuildingStateSoundController::Timer_Expired(GameObject *obj,int number)
{
 if (number == 1)
 {
  GameObject *object = Commands->Find_Object(speakerid);
  if (object)
  {
   Commands->Send_Custom_Event(obj,object,9030,1,0);
  }
 }
}

void Iran_BuildingStateSoundController::Custom(GameObject *obj,int type,int param,GameObject
*sender)
{
 if (type == 9028)
 {
  GameObject *object = Commands->Find_Object(speakerid);
  if (object)
  {
   Commands->Send_Custom_Event(obj,object,9028,1,0);
  }
 }
 if (type == 9029)
 {
  GameObject *object = Commands->Find_Object(speakerid);
  if (object)
  {
   Commands->Send_Custom_Event(obj,object,9029,1,0);
  }
 }
}

void Iran_BuildingStateSoundController::Register_Auto_Save_Variables()
{
 Auto_Save_Variable(&speakerid,4,1);
}

ScriptRegistrant<Iran_BuildingStateSoundSpeaker>
M00_BuildingStateSoundSpeaker_Registrant("M00_BuildingStateSoundSpeaker","Sound_Normal:string,Sound_Destroyed:string,BuildingController_ID:int,Is_3D=1:int,Offset:vector3,Offset_Randomness:vector3,Frequency_Min=-1:float,Frequency_Max:float,Is_3D_Destroyed=1:int,Offset_Destroyed:vector3,Offset_Randomness_Destroyed:vector3,Frequency_Min_Destroyed=-1:float,Frequency_Max_Destroyed:float,Explosion_Name:string");

ScriptRegistrant<Iran_BuildingStateSoundController>
M00_BuildingStateSoundController_Registrant("M00_BuildingStateSoundController","BuildingSpeaker_ID:int");
```