Subject: C++ HUD Time display - Help request
Posted by Raptor RSF on Fri, 15 Jan 2010 20:05:29 GMT
View Forum Message <> Reply to Message

Hello, i created this little feature that shows the time on the hud. But it doesn't support 'Saving Time'.

Does anybody know how to implement it, or does anybody know a good simple script ?

SystemTime.h


```
/* SystemTimeItemClass
 Copyright 2009 Mark Sararu

 This file is part of the Renegade scripts.dll
 The Renegade scripts.dll is free software; you can redistribute it and/or modify it under
 the terms of the GNU General Public License as published by the Free
 Software Foundation; either version 2, or (at your option) any later
 version. See the file COPYING for more details.
 In addition, an exemption is given to allow Run Time Dynamic Linking of this code with any
closed source module
 that does not contain code covered by this licence.
 Only the source code to the module(s) containing the licenced code has to be released.
*/

#ifndef SHADERS_SYSTEMTIME_H_
#define SHADERS_SYSTEMTIME_H_

class SystemTimeItemClass
{
protected:
 bool Enabled;
 bool Enabled2;

 Render2DClass*  Render2D;
 Render2DTextClass* Render2DText;

 bool    FlashActive;
 unsigned int  EndFlashTime;

 unsigned int  SystemTimeFlashDuration; // in msec

 Vector2    TextPosition;
 char *    TextFontFile;

public:
 SystemTimeItemClass();
```

```cpp
  ~SystemTimeItemClass();
  void Load(INIClass* ini);
  void Render();
};

extern SystemTimeItemClass SystemTime;

#endif
```

SystemTime.cpp

```cpp
/* SystemTimeItemClass
 Copyright 2009 Mark Sararu

 This file is part of the Renegade scripts.dll
 The Renegade scripts.dll is free software; you can redistribute it and/or modify it under
 the terms of the GNU General Public License as published by the Free
 Software Foundation; either version 2, or (at your option) any later
 version. See the file COPYING for more details.
 In addition, an exemption is given to allow Run Time Dynamic Linking of this code with any
closed source module
 that does not contain code covered by this licence.
 Only the source code to the module(s) containing the licenced code has to be released.
*/

#include "scripts.h"
#include "shadereng.h"
#include "systemtime.h"


// structure of color data
struct Color
{
 unsigned int ColorValue;
 float Value;
};


SimpleDynVecClass<unsigned int> *Colors8;
unsigned long SystemTimeColor = 0;

SystemTimeItemClass::SystemTimeItemClass():
 Enabled(false),
 Enabled2(false),
```

```cpp
 Render2D(NULL),
 Render2DText(NULL),
 FlashActive(false),
 EndFlashTime(0),
 SystemTimeFlashDuration(5000),
 TextPosition(0, 0),
 TextFontFile(NULL)
{
};

SystemTimeItemClass::~SystemTimeItemClass()
{
 SAFE_DELETE(Render2D);
 SAFE_DELETE(Render2DText);
 SAFE_DELETE(TextFontFile);
};

void SystemTimeItemClass::Load(INIClass *ini)
{
 if (!ini) return; // if you don't have an ini, something is horribly wrong!

 const char* section_name = "SystemTime";

 Enabled = ini->Get_Bool(section_name, "SystemTimeEnabled", false);
 Enabled2 = ini->Get_Bool("General", "SystemTimeEnabled", false);
 if ((!Enabled) && (!Enabled2)) return;



 // Gathers the colors from hud.ini
 Colors8 = new SimpleDynVecClass<unsigned int>;

 unsigned int color = RGB(255,255,255)+0xFF000000;
 Colors8->Add(color);
 unsigned int colors8 = ini->Get_Int("General","ColorCount",0);
 for (unsigned int i = 0;i < colors8;i++)
 {
  char section[10];
  sprintf(section,"Color%d",i+1);
  unsigned int Red = ini->Get_Int(section,"Red",255);
  unsigned int Green = ini->Get_Int(section,"Green",255);
  unsigned int Blue = ini->Get_Int(section,"Blue",255);
  unsigned int Alpha = (ini->Get_Int(section,"Alpha",255) << 24);
  color = RGB(Blue,Green,Red)+Alpha;
  Colors8->Add(color);
 }
 unsigned int SystemTimeCol = ini->Get_Int(section_name,"SystemTimeColor",0);
 SystemTimeColor = (*Colors8)[SystemTimeCol];
```

```cpp
Render2D = CreateRender2DClass();

Vector2 screen_center;
screen_center.X = (ScreenResolution->Right - ScreenResolution->Left) / 2.0f;
screen_center.Y = (ScreenResolution->Bottom - ScreenResolution->Top) / 2.0f;


char temp[512];
ini->Get_String(section_name, "Text.Font.File", "DEFAULT_FONT", temp, 512);
Render2DText = CreateRender2DTextClass(temp);
TextFontFile = newstr(temp);

float average_height = ini->Get_Float(section_name, "Text.Font.AverageCharacterHeight", 16);

bool text_centered = ini->Get_Bool(section_name, "Text.Position.Centered", true);
TextPosition.X = ini->Get_Float(section_name, "Text.Position.X", 0.0f);
TextPosition.Y = ini->Get_Float(section_name, "Text.Position.Y", 0.0f);
if (TextPosition.X < 0)
{
 TextPosition.X += ScreenResolution->Right;
}
if (TextPosition.Y < 0)
{
 TextPosition.Y += ScreenResolution->Bottom;
}

if (text_centered)
{
 TextPosition = TextPosition + screen_center;
 TextPosition.Y -= average_height / 2.0f;
}

SystemTimeFlashDuration = ini->Get_Int(section_name, "SystemTimeFlashDuration", 1000);
};


void SystemTimeItemClass::Render()
{
 if ((!Enabled) && (!Enabled2)) return;

 unsigned int current_time = *SyncTime;

 EndFlashTime = current_time + SystemTimeFlashDuration;
 FlashActive = true;
```

```
  unsigned int color = 0;
  color = SystemTimeColor;

  if (FlashActive && (current_time < EndFlashTime))
  {
   unsigned int alpha = color >> 24;
   alpha *= ((unsigned int)EndFlashTime-*SyncTime);
   alpha /= (unsigned int)SystemTimeFlashDuration;
   color = (color & 0x00FFFFFF)| (alpha<<24);
  }

  Render2DText->Reset();
  RectClass *r = (RectClass *)((char *)Render2DText+0x5B8);
  r->Top = TextPosition.Y;
  r->Left = TextPosition.X;
  r->Bottom = TextPosition.Y;
  r->Right = TextPosition.X;
  char text[64];
  SYSTEMTIME st;
    GetSystemTime(&st);
  sprintf(text,"%02d:%02d:%02d" ,st.wHour,st.wMinute,st.wSecond);
  Render2DText->Draw_Text(text, color);
  Render2DText->Render();
};

//-------------------------------------------------------------------------
// globals
//-------------------------------------------------------------------------
SystemTimeItemClass SystemTime;
```

HUD.ini

```
[General]
SystemTimeEnabled=true

[SystemTime]
Text.Font.File = font12x16.tga
Text.Font.AverageCharacterHeight = 0
Text.Position.Centered = false
Text.Position.X = 250.0
Text.Position.Y = 250.0
SystemTimeColor = 1
```

Subject: Re: C++ HUD Time display - Help request
Posted by Sir Kane on Fri, 15 Jan 2010 22:30:43 GMT
View Forum Message <> Reply to Message

What is 'Saving time'?

Subject: Re: C++ HUD Time display - Help request
Posted by Raptor RSF on Fri, 15 Jan 2010 22:46:57 GMT
View Forum Message <> Reply to Message

Summer / Winter time difference. (1 hour)

So this ingame clock gives me the time of one hour in the past.