
Subject: !rtc c++

Posted by [SODPaddy](#) on Mon, 08 Aug 2011 21:56:22 GMT

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

Hey, im trying to create !rtc (request teamchange)

i found this source code in SSGM2.0.2XWFDS /

http://www.renegadeforums.com/index.php?t=msg&goto=437221&rid=19922&srch=SSGM2.0.2XWFDS#msg_437221

SSGM2.0.2XWFDS doesnt support Veteran.dll so i need to add !rtc myself

```
Quote:class RTCChatCommand : public ChatCommandClass {
void Triggered(int ID,const TokenClass &Text,int ChatType) {
    if (SwapCheck(ID) == false) {
        RequestTeamChange(Get_Player_Name_By_ID(ID));
    }
    else {
        Console_Input(StrFormat("ppage %d You have already swapped once this map.",ID).c_str());
    }
}
};
ChatCommandRegistrant<RTCChatCommand>
RTCChatCommandReg("!swap;!rtc;!requestteamchange",CHATTYPE_ALL,0,GAMEMODE_ALL);
```

my problem:

```
1>.\gmmain.cpp(1321) : error C3861: "SwapCheck": Bezeichner wurde nicht gefunden.
```

```
1>.\gmmain.cpp(1322) : error C3861: "RequestTeamChange": Bezeichner wurde nicht gefunden.
```

I dont found SwapCheck or RequestTeamChange in header files

sorry, my english is very bad - my last English school lesson is many years ago

Subject: Re: !rtc c++

Posted by [Xpert](#) on Mon, 08 Aug 2011 22:19:42 GMT

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

engine_player.cpp

```
int Get_Team_Player_Count(int Team)
{
    int Total = 0;
    GenericSLNode *x = BaseGameObjList->HeadNode;
```

```

while (x)
{
    GameObject *o = (GameObject *)x->NodeData;
    if (o && Commands->Is_A_Star(o))
    {
        if (Get_Team(Get_Player_ID(o)) == Team)
        {
            Total++;
        }
    }
    x = x->NodeNext;
}
return Total;
}

```

gmmain.h

```

class RTC_timer : public ScriptImpClass {
    void Created(GameObject *obj);
    void Timer_Expired(GameObject *obj,int number);
    int ID;
};

```

gmmain.cpp

```

bool IsRTC = false;
int idrtc = 0;

```

```

struct swapped {
    std::string SwappedPlayerName;
};

```

```

std::vector<swapped> SwappedPlayers;

```

```

bool SwapCheck(int ID) {
    for (int i = 0; i < SwappedPlayers.size(); i++) {
        if (SwappedPlayers[i].SwappedPlayerName == Get_Player_Name_By_ID(ID)) {
            return true;
        }
    }
    return false;
}

```

```

void SwapAddPlayer(int ID) {
    if (SwapCheck(ID) == false) {
        swapped temp;
    }
}

```

```

temp.SwappedPlayerName = Get_Player_Name_By_ID(ID);
SwappedPlayers.push_back(temp);
}
}

void SwapClearPlayers() {
SwappedPlayers.erase(SwappedPlayers.begin(), SwappedPlayers.end());
}

void RequestTeamChange(const char *Name) {
GameObject *Player = Get_GameObj_By_Player_Name(Name);
int ID = Get_Player_ID(Player);
int GDIPlayers = Get_Team_Player_Count(0);
int NodPlayers = Get_Team_Player_Count(1);

if (!IsRTC) {
if (GDIPlayers > 0 && NodPlayers > 0) {
GameObject *rtccontroller =
Commands->Create_Object("Invisible_Object",Vector3(0.0f,0.0f,20.0f));
idrtc = ID;
Commands->Attach_Script(rtccontroller,"RTC_timer","");
Console_Input(StrFormat("msg %s has requested to change teams. Type !rtc if you would like
to change teams.",Name).c_str());
IsRTC = true;
}
else {
Console_Input(StrFormat("page %d There are not enough players for you to request a team
change.",ID).c_str());
}
}
else if (IsRTC) {
GameObject *obj = Get_GameObj(idrtc);
if (!obj) {
GameObject *rtccontroller =
Commands->Create_Object("Invisible_Object",Vector3(0.0f,0.0f,20.0f));
idrtc = ID;
Commands->Attach_Script(rtccontroller,"RTC_timer","");
Console_Input(StrFormat("msg %s has requested to change teams. Type !rtc if you would like
to change teams.",Name).c_str());
}
else {
if (Get_Team(ID) != Get_Team(idrtc)) {
Console_Input(StrFormat("team2 %d %i",ID,Get_Team(idrtc)).c_str());
SwapAddPlayer(ID);
Console_Input(StrFormat("team2 %d %i",idrtc,Commands->Get_Player_Type(Player)).c_str());
SwapAddPlayer(idrtc);
Console_Input(StrFormat("msg %s and %s have changed teams! The !rtc command is up for
new request.",Name,Get_Player_Name_By_ID(idrtc)).c_str());
}
}
}
}
}

```

```

    Console_Input("player_info");
    GameObject *timerthing = Find_Object_With_Script("RTC_timer");
    Remove_Script(timerthing,"RTC_timer");
    Commands->Destroy_Object(timerthing);
    IsRTC = false;
    idrtc = 0;
}
else if (ID == idrtc) {
    Console_Input(StrFormat("msg %s has revoked their !rtc request. The !rtc command is up for
new request.",Get_Player_Name_By_ID(idrtc)).c_str());
    GameObject *timerthing = Find_Object_With_Script("RTC_timer");
    Remove_Script(timerthing,"RTC_timer");
    Commands->Destroy_Object(timerthing);
    IsRTC = false;
    idrtc = 0;
}
else if (Get_Team(ID) == Get_Team(idrtc)) {
    Console_Input(StrFormat("ppage %d You're on the same team, you cannot swap with
%s.",ID,Get_Player_Name_By_ID(idrtc)).c_str());
}
}
}
}
}

```

```

void RTC_timer::Created(GameObject *obj) {
    Commands->Start_Timer(obj, this,60.0f, 1);
}

```

```

void RTC_timer::Timer_Expired(GameObject *obj, int number) {
    if (number == 1) {
        IsRTC = false;
        GameObject *obj2 = Get_GameObj(idrtc);
        if (!obj2) {
            idrtc = 0;
            Commands->Destroy_Object(obj);
            Destroy_Script();
        }
        else {
            Console_Input(StrFormat("msg %s's request to change teams has
expired.",Get_Player_Name_By_ID(idrtc)).c_str());
            Console_Input(StrFormat("ppage %d Your request to change teams has timed
out.",idrtc).c_str());
            idrtc = 0;
            Commands->Destroy_Object(obj);
            Destroy_Script();
        }
    }
}
}
}

```

```
ScriptRegistrant<RTC_timer> RTC_timer_Registrant("RTC_timer", "");
```

```
class RTCChatCommand : public ChatCommandClass {  
void Triggered(int ID,const TokenClass &Text,int ChatType) {  
    if (SwapCheck(ID) == false) {  
        RequestTeamChange(Get_Player_Name_By_ID(ID));  
    }  
    else {  
        Console_Input(StrFormat("ppage %d You have already swapped once this map.",ID).c_str());  
    }  
}  
};  
ChatCommandRegistrant<RTCChatCommand>  
RTCChatCommandReg("!swap;!rtc;!requestteamchange",CHATTYYPE_ALL,0,GAMEMODE_ALL);
```

Copying just the command class doesn't make the code work. You need everything else that makes up the command.
