

# Creating AI-Bots in **Multiplayer Games**

by

Dr. Edwin F. Armstrong

<http://ed-karen.com/~efa/papers/>

or

<http://tinyRealm.com/~efa/papers/>

# What are A.I. Bots

- An A.I. Bot is any character in a game that has the appearance of intelligence, interacts with the players in some way and is autonomous.
- In my definition, they are more than simple sprites that only have a sequence of animated moves. - like a bird flying across the sky or doors opening and closing..
- Their intelligent actions are generally implemented as a state machine, although a subsumptive design is often employed – more on this later in the talk.

# Here is a simple door sprite



# Why have Bots

- An A.I. Bot can be a player's only adversary or can be utilized in a single player mode. At their simplest they are simply test cases to work out bugs in the game – during development, we can't always find someone to help with game testing.
- A.I. Bots can also be used in a multiplayer mode as a common adversary to all other players.
- Their actions are generally implemented as a state machine, although a subsumptive design is implied or sometimes employed – more on this later in the talk.

Here is an Space Orc as an A.I. Bot -  
he was designed to miss on occasion,  
my students insisted.



- In a multi-team game like capture the flag they can be team-mates – the red-team bots help red-team players and the blue-team bots help the blue-team players. They can defend their team-mates or attack the other team or just act as guards.
- There are many uses of A.I. Bots and the list that I have created is not exhaustive. In fact, it is mostly focused on adversarial games.
- Bots can also be used as companions or interesting characters that are encountered along a quest or in other puzzle like games. Their perceived intelligence can also vary – from simple minded bots to bots whose actions are very close to that of a human player.

# Multiplayer game types

- There is the generally two types:
- 1) A Non-persistent game where players create a server and a master server connects other players to the created server. These end when the Server shuts down – you've got to sleep sometime.
- 2) A persistent game which is up all the time 24/7 where people join and leave as they wish. Something like the “Ever Quest” or “World Of War Craft” games. If you'd like a non-violet example “Puzzle Pirates” is a great game of this type. These games create a world where players enter, interact and leave as they wish. An interactive virtual world.

# **O.K. Now to the details of creating aiBots using the Torque game engine.**

- Most of the work in Torque is done in scripting files and in the map files for specific maps. The scripting files can be either on the server or client side. Everything local happens on the client side and every global, seen by other players is done on the server side.
- How do we place aiBots in a game? We place bots in the game by creating an AISpawn point in a map. All the rest of the work is done by aiBots.cs, a script file in the server directory.

**This is from map StoneWoodCanyon.mis**

```
New SimGroup(AITeamDropPoints) {  
    canSaveDynamicFields = "1";  
    new SpawnSphere() {  
        canSaveDynamicFields = "1";  
        position = "-99.6857 -54.96 115.755";  
        rotation = "1 0 0 0";  
        scale = "1 1 1";  
        dataBlock = "SpawnSphereMarker";  
        Radius = "1";  
    };  
};
```

**This is from the aiBot.cs script file.**

```
function pickAISpawnPoint()
{
    %groupName = "MissionGroup/AISpawnPoints";
    %group = nameToID(%groupName);
```

**// It's a bit hard to read, we better use  
the editor...**

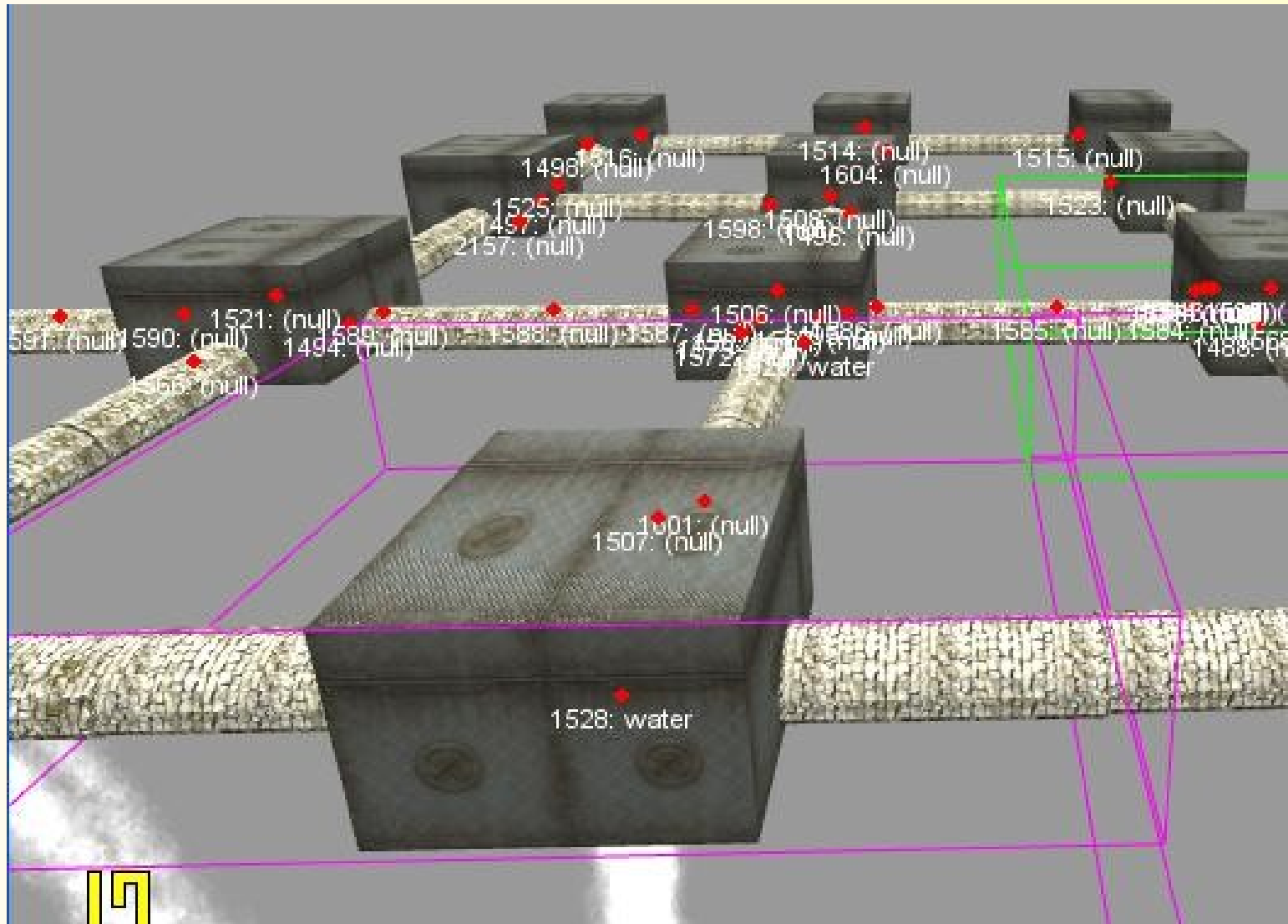
# What do the script and map files do?

- aiBots.cs create the A.I. bots.
- playerOrc.cs is used as our A.I. Player, it's attributes are defined here.
- games.cs starts the game by calling other scripts.
- stonewoodCanyon.mis is a map file to show how A.I. Spawn points are specified.
- aiPlayer.cs is not currently in use. It is an example script from Garage games that shows path following. The aiPlayer is spawning onto a path...

# How does the AIBot get in the game?

- A server script “/core/server/scripts/game.cs” is called when the game program starts.
- This program calls many other scripts among them is aiBots.cs which implements or bots. There are many other files (sounds, textures, dts models, etc. that are used by the program.
- I will walk you through the implementation but you really need the whole program to really get it working. Get it from me or download it from:
- <http://ed-karen.cin/~efa/papers/>

**Here is a map called sewer – it has special problems in that players and bots must spawn inside the pipes...**



# A need for Human like Bots

- These days multiplayer games need to be a social experience. The appeal of Multiplayer games is that you are actually winning or losing against real people. Or in the more social world interacting with real people.
- The problem is how do you initially get those real people to come and play. Once they are there the game play needs to keep them coming back but you have to get them there in the first place.
- The key is having enough players on 24/7 so when someone wants to play there is someone to play with.

- Even if you've got a great game people can't play and wont stay if there is not enough people to play with.
- What you need is Bots so human like that the new players can't tell them from a live human player. They would be deployed from special servers when the number of real players dropped below a certain number.
- Hey! that is cheating. Don't look at it that way. Think of it as a way to keep the game active when people are at work or asleep. In an internet world there is always someone awake and someone else asleep or at work. When I want to play and no ones on – I think what's wrong with this game? Where is everyone?
- Actually it is a real challenge to make real human like bots. If it really bothers you but the info. In your documentation or allow the player to turn the feature on or off.

# Realism in game play

- You always hear about how people want more realism in game play, but this is only true to a point.
- People want only as much realism in a game as is required for them to feel immersed and engaged.
- A player's speed, falling distance and jumping height are some what arbitrary – the player's mostly are here to have fun. Make a game where few if anyone can win and you will have a game that few if anyone will play.
- A game should be easy to learn (play) but hard to master (be hard to solve or master every game aspect). This makes the game fun while still being challenging.

# Action games

- Action games are not puzzle games or educational tools. They offer the player struggle, fear, loss and victory. This invokes a visceral response not present in typical puzzle games. It's easiest to achieve in war games, that have a kill-or-be-killed scenario.
- Racing games give the same thrill of play without the killing. Games that require you to complete a task before the other team or the time runs out can also be interesting but they need to be a sense of loss. This usually requires that other people participate in the win or loss.

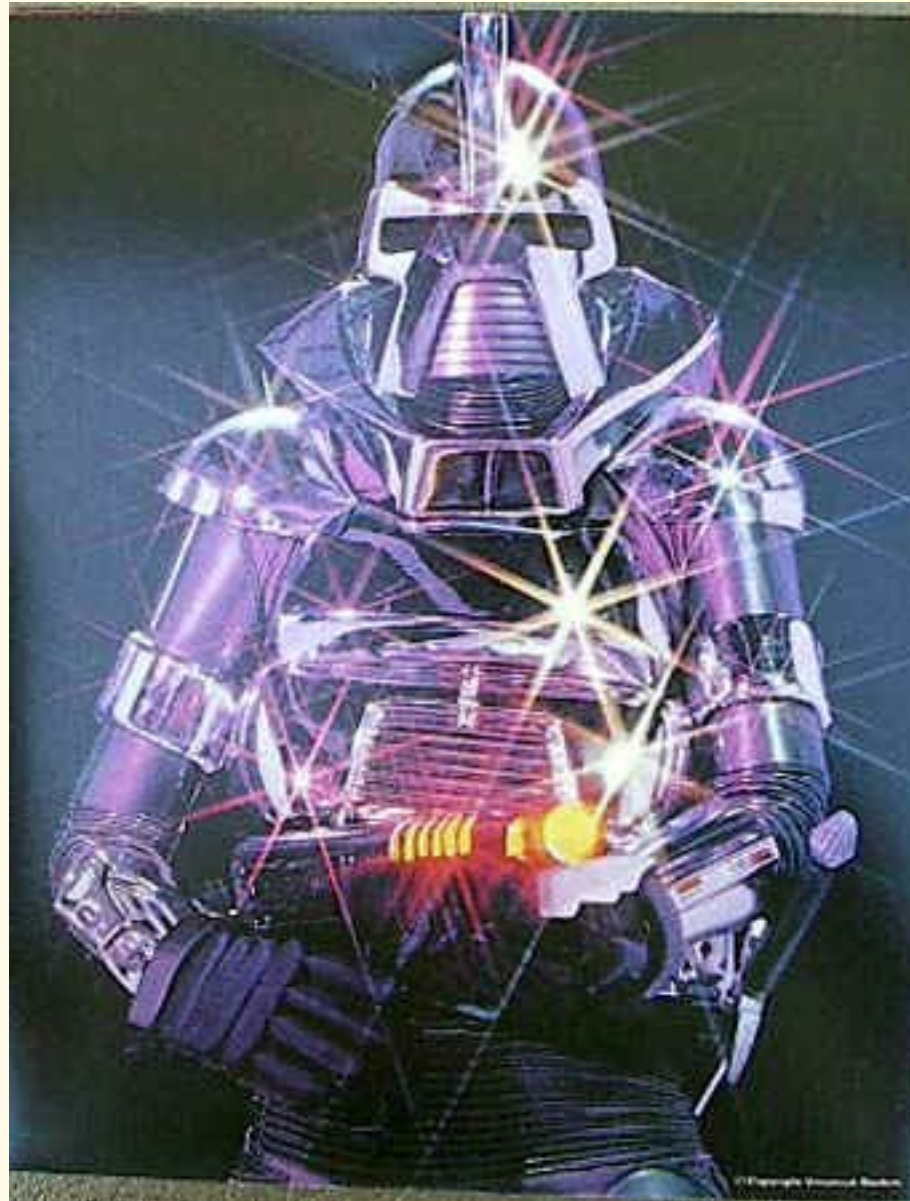
# Action games

- The question is how do you get a visceral reaction during game play without killing people?
- In the racing game it is from your team winning or your team losing – you definitely have a sense of loss or a sense of victory.
- After all, when you played sports in high-school you didn't need to kill the other team to enjoy a win over them.
- Why does anyone have to win or loose? Now we could have a debate; for me if I can't win or lose, the game's not worth playing. I might play once but not twice. Still, I don't need to kill people...

# Don't kill people- How about Space Spiders? There not very cuddly...



**Don't kill people- How about Scary  
Robots? There not human...**



**Don't kill people- How about  
Friendly Robots? They can fix things.  
The player can be the robot or the  
robot can be a helper...**



# No One Lives Forever II

- O.K., I have taught games programming for about 7 years and I don't have the answer. I am still looking for that perfect solution to the question.
- One game that I like is “No One Lives Forever II” – well not all of it but certain features. I like that it's first goal is that you sneak into town to photograph a meeting. There are puzzles to be solved but not too many. You can use a stun gun or knock-out gas, instead of a lethal weapon to achieve your goals. It also has lots of humorous dialogue.
- It's not perfect but I have learned a lot from the game.

# N.O.L.F. 2 (stun-gun video clip)



# Thank you

- I had fun putting together this talk. Can I answer any questions?
- Or does anyone have any comments – really, I'd like to know your thoughts.
- All the source code, movies and slides are available at:
- <http://ed-karen.com/~efa/papers/> or
- <http://tinyrealm.com/~efa/papers/>