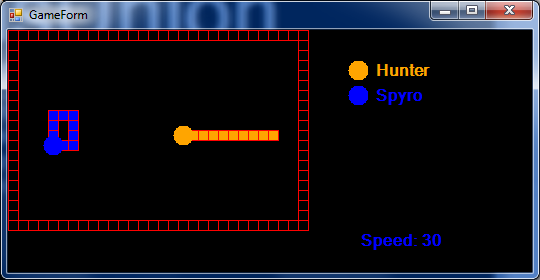
# BotWars2 – TronsUnion!

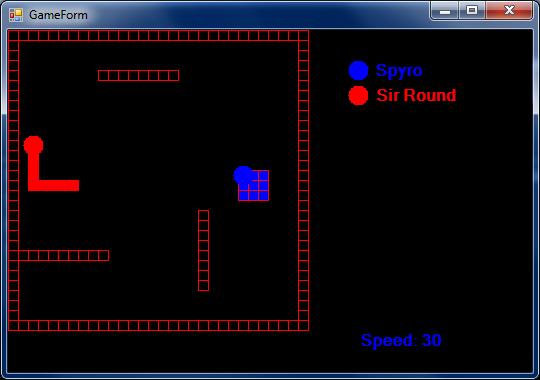
## The Game

TronsUnion is played on a grid of varying sizes, players, and obstacles. You will move your bot around the grid avoiding the walls, tails, and borders of the arena. Colliding with any of these will cause you to lose. The winner is the bot who survives the longest.



The two players navigate around the arena. The bot will be deactivated if it collides with one of the borders, a wall, an opponents’ tail, or its own tail. If two or more bots collide then each will be deactivated.

The arenas will grow more complex as the game continues. Walls will be added as the afternoon progresses.



Later in the afternoon these walls will begin to move!

### Round 1

Two bots with no walls.

### Round 2

Two bots with two internal walls.

### Round 3

Two bots with three walls, some of which will move!

### Round 4

All the bots playing a single game in one large arena.

## Getting Started

Teams should create their bot by modifying the MyBot class within the client solution. You can download the framework code from here:

<https://github.com/ardliath/BotWars2Client>

### Server

Your App.config has a key (BotWarsServer) - your game host will provide the value you need to use to connect to the correct game.

### Name

Your bot has the property:

public string Name => "Player Bot";

By default. This property is the name gives to the server when it registers and therefore the property you’ll want to customise before you play the first round.

### SecretCommandCode

BotWars2 has state of the art cryptography and encryption on it’s backlog. Instead each bot uses a simple secret property which it sends to the server

public string SecretCommandCode => "123";

It is highly recommended that you change this value before the first game (otherwise other teams may well hijack your bot). Please do not change your secret code after the first round has been played as this may cause unexpected problems.

### GetMove

The GetMove method is where you write your main logic.

public Direction GetMove(GetMove getMoveData)

When the game starts, and each subsequent turn the server will ask your bot which direction it wishes to travel in. The four options are Up, Down, Left, or Right. You cannot opt not to move.

You will use the information within the getMoveData object data to decide where to move.

The GetMove contains a property called RadarScan, each time the server requests your move it provides you with a radar scan of the current game. This is made up of a range for Up, Down, Left, Right, your current position, and the position of other plays in the arena with you.

In the scan below you (as the red bot) would get the following values:

|  |  |
| --- | --- |
| Up | 4 |
| Down | 0 |
| Left | 7 |
| Right | 11 |
| EnemyPositions[0] | 13, 5 |
| MyPosition | 8, 8 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 9 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 11 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 12 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 13 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 14 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 15 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

### Other Methods

There are several other methods which can be used to capture information;

public void StartGame(StartGame data)

{

}

public void EndGame(EndGame endGameData)

{

}

public void EndRound(EndRound endGameData)

{

}

These are called before each game your bot is playing begins, after each game finishes, and when all games in a round have completed. Each contain varies pieces of useful information you may wish to capture to successfully play the game.