

GAME AI FOUNDATIONS - DAY 0

Introduction to Game AI

WORKSHOP GUIDE BY
REDEFINE GAMEDEV - 2022

| | |
|---|-----------|
| Summary | 3 |
| Introduction to Game AI | 4 |
| AI in Games | 5 |
| AI as Enemies | 5 |
| AI as Companions | 5 |
| AI as Your own units | 5 |
| AI as Your own character | 5 |
| AI as Invisible helpers | 5 |
| The AI Agent | 6 |
| Most Used AI Systems | 7 |
| Simple AI | 7 |
| State Machines | 7 |
| Behavior Trees | 7 |
| Planning | 7 |
| Improve Your Player's Experience | 8 |
| Give your player SUPERPOWERS | 8 |
| Have Your AI Communicate more | 8 |
| Slow down your AI | 8 |
| AI System - Pick the right one for you | 9 |
| Ending Words & Contact | 10 |

Summary

Welcome to the world of Artificial Intelligence for games!

We will be covering in this free workshop a lot of topics and answer a lot of questions. If you have more, you can also email them to me later at this [email address](#).

- In the **Introduction to Game AI** we will look at some challenges that modern games have and how by understanding AI we can make better games
- Next, we will look into **different types of AIs in games**, or in what shape or form we can find the AI systems applied
- In the next part we will look into **tips & tricks for how to improve your Game AI** for an enhanced player experience
- Lastly, we will tackle the topic of **which AI to use for your game**

Introduction to Game AI

YOUR MULTIPLAYER-ONLY GAME MIGHT FAIL

There are a lot of challenges that multiplayer-only games need to face. The lure is there - you, the game developer, see the huge success of Among Us, Valheim, Fortnite and think “I should probably make a multiplayer-only” game as well.

And it’s a valid assumption - these multiplayer games earned a lot of revenue. Who wouldn’t want to jump into that boat?

Here are a couple of technical challenges a multiplayer game faces: latency compensation / prediction, managing servers, multiplayer-specific game implementation, infrastructure management and scale-up not even mentioning the cheating problem which most big games are plagued with.

Of course, probably the **biggest challenge** is not a technical one: **releasing a Dead on Arrival Game**. Indie games don’t have the marketing budget to attract a lot of players to fill their worlds, especially from the release day. Even Among Us had a ‘failed’ release back in 2018. It took work from big streamers to put the game on the map. **Not even being AAA guarantees** multiplayer success. Notable examples here are Titanfall 2 and Evolve, games that were dead on arrival.

LEARNING AI PROPERLY WILL IMPROVE YOUR GAME

How can learning AI help your game?

Fake it till you make it

If you are decided to go for a multiplayer game, why not have hidden AI Agents that act as players? **Faking multiplayer** is a technique used by most .io games (the ones in the browser) and they get away with it!

What would your player prefer - an empty world? Or a world filled with AI Agents ready to be swapped by real players, if they join?

Make an awesome single player experience

There are a lot of single player games where the AI is either too easy or too hard. Having a custom implementation will improve your game's chances to attract players.

AI in Games

AI as Enemies

Example here: all the games that have people running toward you and shooting at you -> most games will make use of AIs like this. They need to be tweaked in several ways from how many they spawn, to where, to how difficult it is to kill them.

AI as Companions

In games where you have a NPC following you. These games are also single player experiences where you are not alone. Either a solo companion will join you like Half-Life 2, Alyx. Or a squad like gears of war or Mass Effect

AI as Your own units

In all RTS games. Be it your own armies, or the enemies' one, you will have to implement all the unit movement in the game.

AI as Your own character

In some top-down point & click games Diablo, Turn based, etc. The AI is not a decisional one here, but a pathfinder. If you control your character with point & click, a path needs to be created so that the character could perform the move.

AI as Invisible helpers

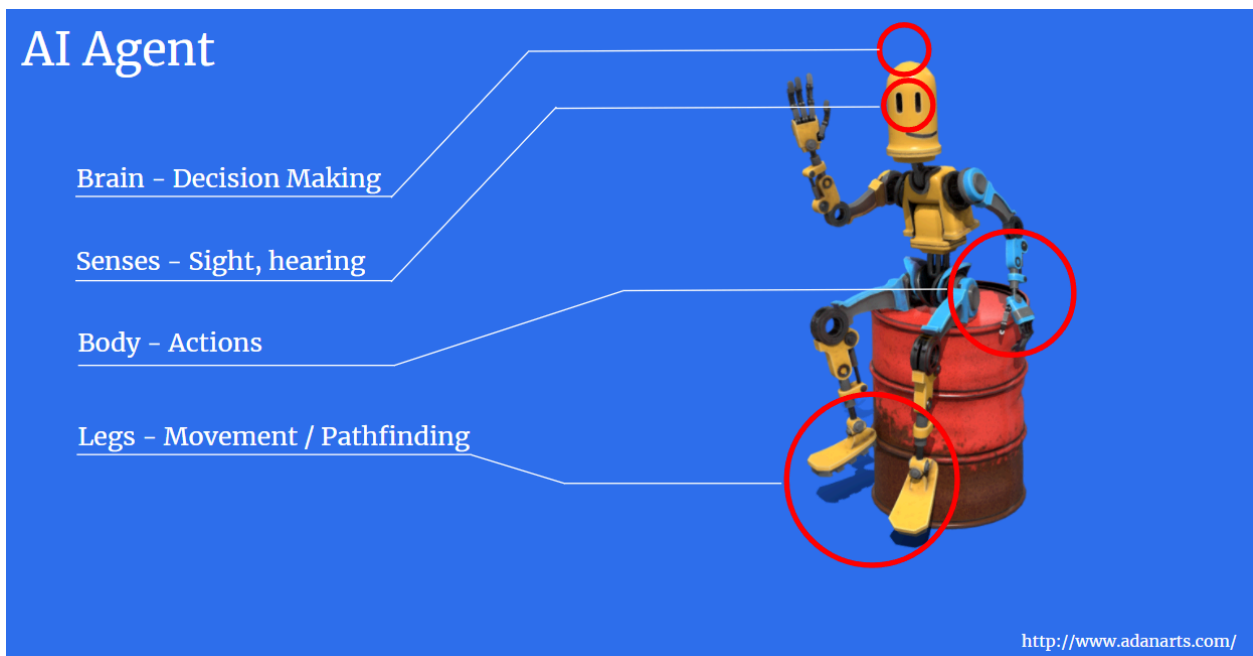
Helper AI systems used in games like the Witness or Apex Legends. This is a behind-the-scenes implementation but it does help game developers a lot since it removes a lot of tedious trial & error level design to get everything right.

The AI Agent

The AI Agent is made of multiple different systems that work in parallel to give the player the feeling that the AI in front of them is “alive”.

This is possible because of the following systems:

- **The Brain:** it's responsible with the decision making
- **The Senses** (or sensors): are the AI's way of detecting enemies, obstacles and more. They are linked with the brain to provide the accurate information in real time
- **The Body:** receives actions from the Brain and executes them
- **The Legs:** are used to move the character. They implement special pathfinding algorithms that compute the route from A to B for the AI to take



Most Used AI Systems

Simple AI

Does not have a particular implementation. Can be implemented fast, but it's very limited. It can support more complex behaviors but continuing to implement them in the code like this will generate spaghetti code.

Games: PAC-MAN

Finite State Machines

This AI is used a lot. Probably if you are familiar with Unity's mecanim system, we have a couple of different states that the AI can be in, with only one active at any given time.

And, transitions between them that are triggered when an event happens.

Games: DOOM

Behavior Trees

BTs are very common in most game engines, having direct integration in UE. Their main selling point is that a designer can create an AI system without having to interact with any code piece.

They are also very flexible to use.

Games: HALO

Planning

This one is very similar to the State Machines, but the main difference is that the states are not connected between them. This gives total freedom for the AI to pick any available option.

Games: FEAR

Utility AI

The Utility AI does not have a particular structure, but rather a set of rules. It's mostly used to handle complex actions like an RPG character that has multiple skills and needs to pick one or another based on the current situation.

Games: Dragon Age: Inquisition

Improve Your Player's Experience

Give your player SUPERPOWERS

Make your player feel special, but in a way that they will experience it as a skill improvement.

- Is the player struggling to shoot the targets? Silently increase the hitboxes a little
- Is the player taking too much damage? Make the AI miss more
- Are there too many AI instances? Maybe spawn fewer in the next encounter

Have Your AI Communicate more

- Did the AI actually detect the player? Sight, hearing -> proper queues will help the player know in which state they are
- Is the AI preparing a special attack on the player? Let the player know a couple of milliseconds, if not seconds before
- Is the AI issuing orders to others? Let the player get the info as well

Slow down your AI

In most cases, a simple implementation will make the AI take fast decisions, but consider adding some lag between them.

- Did the AI see the player? Wait a little until it's confirmed
- Did the AI start to shoot? Make the first projectiles miss every time

AI System - Pick the right one for you

A common question regarding the AI implementation is what system to use and when. Why is this hard to answer?

Well, because we do have multiple systems like

- Simple AI
- Finite State Machines
- Behavior Trees
- And, Planning

They do have pros/cons each and this is probably the most straightforward way to determine what is good for your game.

Let's take a look at this comparison table

| | Implementation Speed | CPU Load | Flexibility |
|----------------------|----------------------|----------|-------------|
| Finite State Machine | ★★★★★ | ★★★★★ | ★ |
| Behavior Tree | ★★★★ | ★★★★ | ★★★★ |
| Planning | ★ | ★ | ★★★★★ |
| Utility AI | ★ | ★★★★ | ★★★★★ |

Right out of the bat - there is no right or wrong answer. Most likely, you will need different types of AI in your game, based on your game's unique implementation.

Ending Words & Contact

AI in video games is as valid now as it was 30 years ago! While the implementation improves continuously, the core concept remains the same - your game's players want a challenging experience.

I encourage you to test and use all the AI systems mentioned and find out which fits your game best. Note that in some cases a mix & match approach might work best.

To fully develop a good AI, keep in mind that the implementation is only a part of it all. Good game design (as the tips previously mentioned) add a lot of value in creating the best possible AI.

The possibilities are unlimited!

Adrian from Redefine Gamedev

- [Youtube](#)
- [Twitter](#)
- [Email](#)