

## **Flocking in 2D/3D**

This simple package will help you create flocking behavior in your game and more importantly, balance it to your specific needs. The core purpose of this project is to save you some time implementing flocking simulation behavior.

The example scene that is included has a transform that is used to set the area in which the gameobjects(boids) live. This is a 2D plane but can be extended to create a 3D cube (other types of objects are not supported). If they exit this area, they are removed from the simulation.

The package includes an exampleScene script, which spawns gameobjects and makes sure that the boids do not live outside of the simulation area.

To alter the different components of the flocking simulation, use the sliders.

Once you are happy with the results, you can implement the flocking behavior in your own game. To implement the simulation in your game, simply copy the manager into your own project. If you want to use the exact flocking area from the scene, you should also copy the exampleScene script, which makes sure the boids are kept in the area and are spawned.

If you do not want to use the flocking area and simply want to use the sliders for your own flocking behavior, you can import the package Flocking > Prefabs > Manager\_For\_Own\_Flocking\_Area, which has the uiManager, gameManager and sliders set up for you. Drag this into your scene and copy the Resources folder into your own project (or simply copy the two files into your own resources folder, either in Assets > Flocking > Resources or change the path in the XmlReadWrite.cs script).

Also, copy the Boid.cs script into your project and make sure that any gameObject that wishes to flock inherits from this class instead of MonoBehaviour. Finally make sure that these boids are added in the \_boids list of the gameManager monoBehavior and removed when they no longer attend the flock; only the velocity of boids that are added in the boids list will be updated.