

# Grewh2D

Genetically Refined Wheels in 2D  
version 0.1.0

Technical documentation  
by Vilém Zouhar

## Specification breakdown

The purpose of Genetically Refined Wheels in 2D (abbrev. Grewh2D) is to demonstrate the beauty of genetic algorithms. Genomes in this program represent polygons with attached circles, also known as cars. Their goal is to reach the other end of the map, although the last part is extendable.

Functional requirements are: user input on initial population generation, physical simulation and statistics screen. For original specification see *original specification*.

## Implementation

As the choice of 2D physics simulation frameworks is limited, Grewh2D is written in C# and uses Unity3D. This offered the author to focus more on larger scale algorithmic background than on graphics and physics pipeline.

## Architecture & design

### High-level stages

The program is divided into three parts:

1. Population showcase
2. Simulation
3. Statistics

The most common workflow is: 1. → 2. → 3. → 1. → .... Each parts corresponds to one of the three functional requirements.

### High-level objects, High-level structures, Global variables

See UML diagram of Grewh2D (fig. 1)

## Algorithms

### Math

The program uses some specially tailored math functions:

- CreatePrettyPolygon: Orders points in polygon counterclockwise. This is very important as Unity3D meshes and Triangulator can't process polygons with holes. Counterclockwise sort is not very intuitive and sometimes creates weird looking shapes, but many manhours were spent until the author realized that minimal length triangulation is a NP complete problem and thus not appropriate here.
- Normalize, NormalizeByMax: Normalizes values in list so that they can be processed in statistics screen.
- RandGood: Returns a random value based on linear distribution (used in gene picking)

## **CreatureBreeder**

Takes care of creating an ideal population from previous results.

## **Other**

The program contains about 2000 lines of code, but most of the functions are used just as services, not very algorithmic.

## **Data structures**

See UML diagram of Grewh2D (fig. 1)

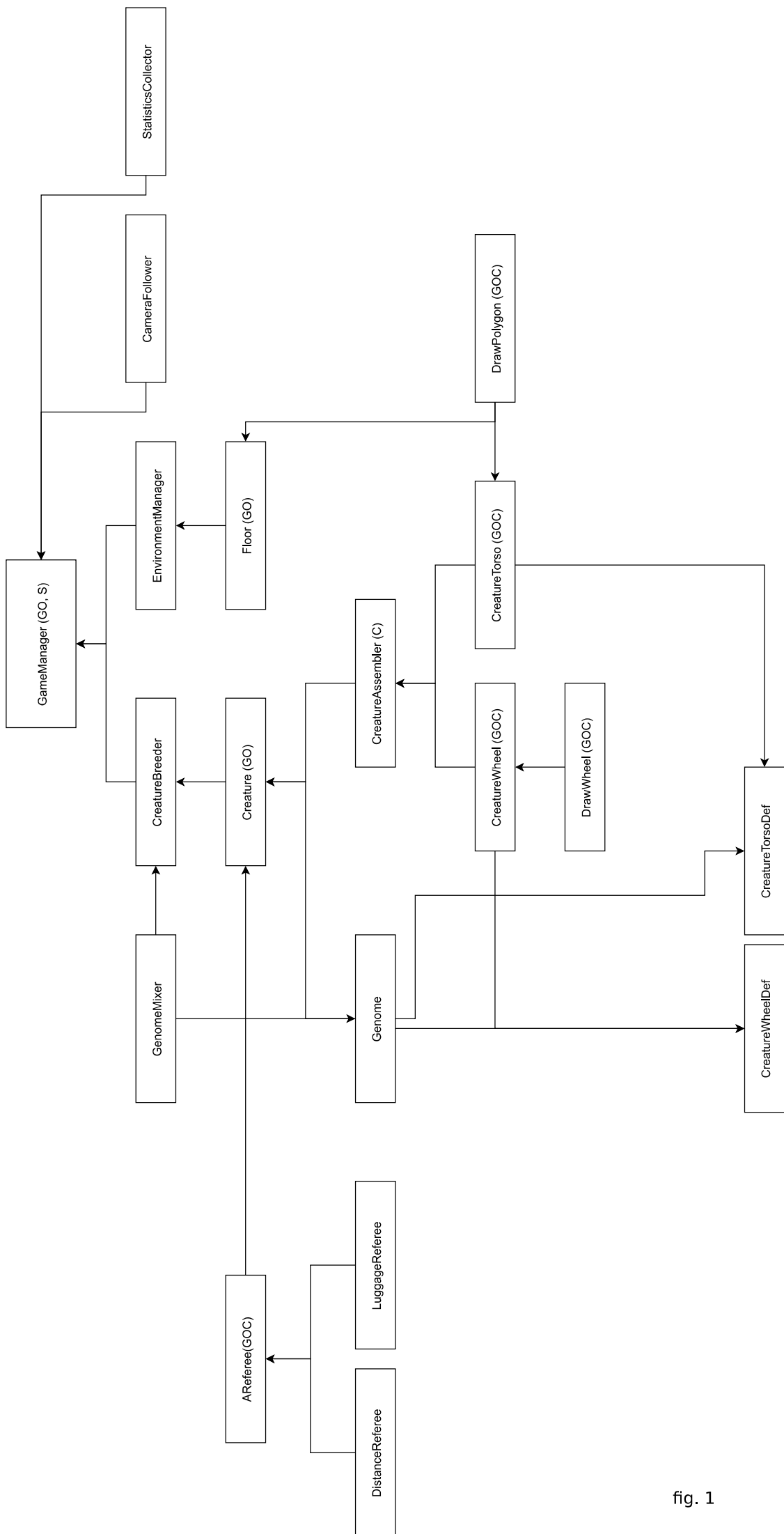


fig. 1