By Ngaba Franck

MetaScript Software Architecture

Vol 1.

Introduction

How a program in Metascripts Works.

- We code the Program Logic into The Core
- We choose the Architecture where we want to Render, Simple Web App, OpenGL Software , Web2D, Web3D, High Ressources Computer (Unreal Engine), Mid Ressources Computer (CryEngine), Low Ressources Computer (Godot) or any other Setup
- There are plugins already configured which will manage how the scenes will been displayed.

The Core

There we write instructions based on Metascript Algebras properties for the Worlds we want to simulate, An MS Algebra Lemme is a set of phenomenon between meta-components

The Metascript Syntax is the C++ syntax to which we added the implementation of each algebra property in a custom grammar

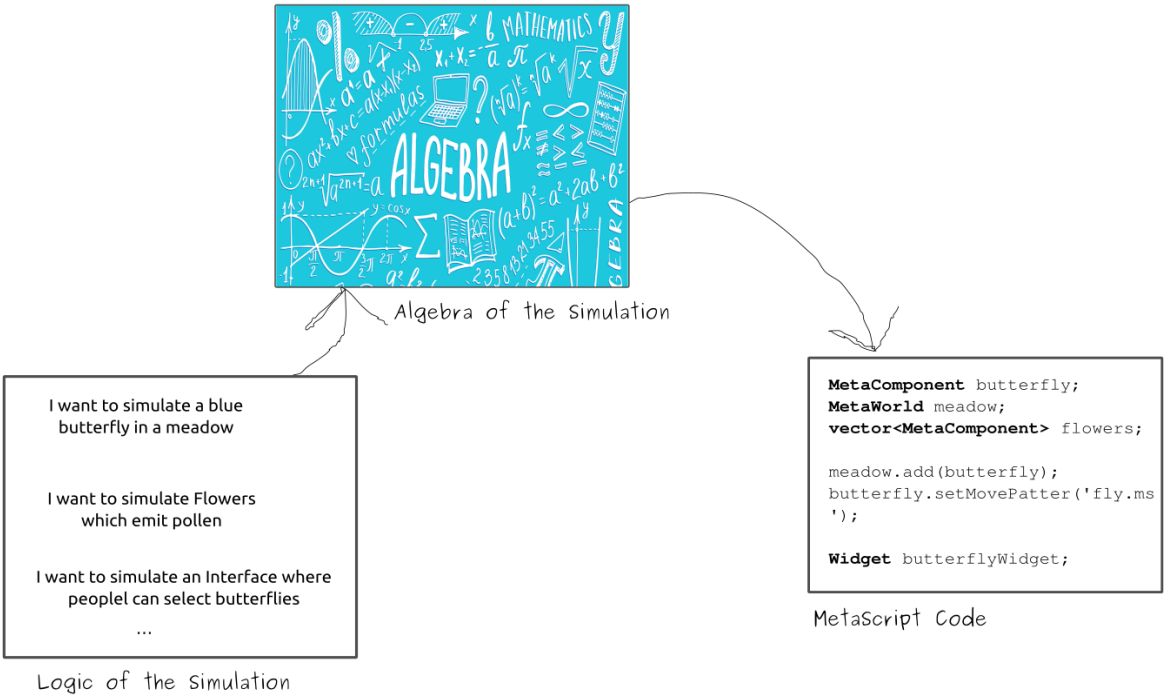

Figure 1: The Core Program of Metascript

The Connectors

Each of the MS Algebra notions has already been configured to render its equivalent in some engines and some technologies

The differents connectors are part of the Metascript Compiler.

It's planned to add many other game engine and software renderer as possible.

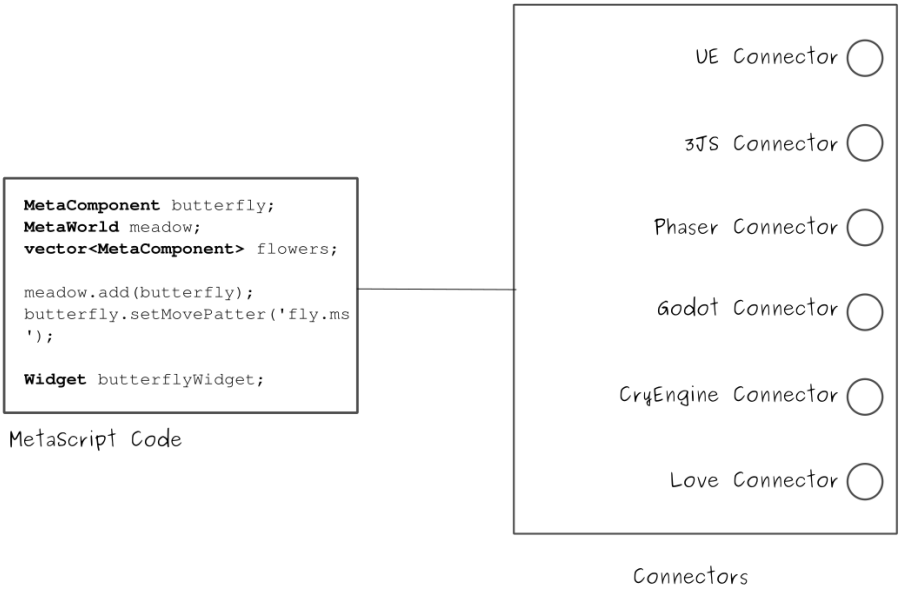

Figure 2: The Connectors of Metascript

The Scene

It is where we render the result of our Metascript Core, the scene is handled by the Engine whe choose to use

A Program made in Metascript can be used in many Engines, the user can easily decide to switch from Godot to UE for example (or either to a 3D reprsentation to a 2D)

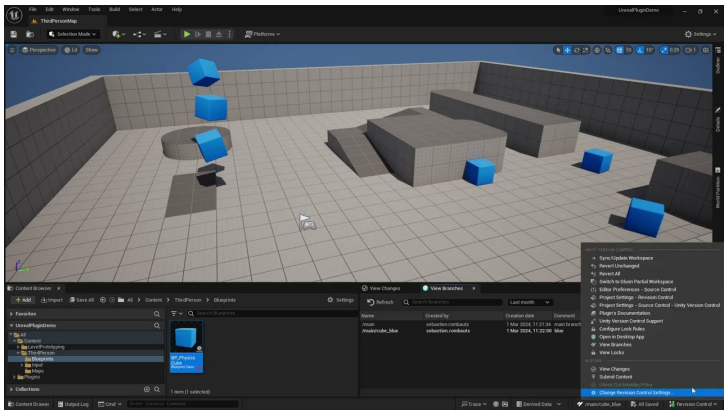

Figure 3: The default 3rd Person UE Scene rendere by Metasctipt