

AsteraX – Components and Classes Diagram

by Jeremy Gibson Bond

Notes:

- A solid black line denotes a parent/child relationship in the Unity Hierarchy pane.
- A solid line with a semi-circle on the end represents one GameObject instantiating another GameObjects as needed.
- A light purple arrow denotes a class pulling info from or utilizing another class.
- Boxes stacked together or touching each other are multiple Components on the same GameObject.
- The AsteraX script has relationships with so many classes that they are denoted with these lines. The open circle is data going from AsteraX, & the closed circle is data going to AsteraX.

Common Components:

These are attached to several different GameObjects

- AODSGS** **ActiveOnlyDuringSomeGameStates.cs**
Sets GameObject.active based on AsteraX.GAME_STATE.
- SPart** **ShipPart.cs**
Script for either a body or turret customizable part
- OSW** **OffScreenWrapper.cs**
Wraps an object around the screen when it exits ScreenBounds

Additional Scripts:

These scripts don't have a place in the diagram. I do not expect you to have these in your diagram.

- Vector3Extensions.cs**
An extension to the Vector3 class that implements a method named ComponentDivide() that divides each component of one Vector3 by its counterpart in another Vector3.
- EnumFlagsAttributePropertyDrawer.cs**
A script posted to the Unity forum by Aqibsadiq that allows any enum with the System.Flags attribute to be edited in the Unity Inspector using MaskField editor (the same pop-up with checkboxes editor that is often shown for Physics Layers).

