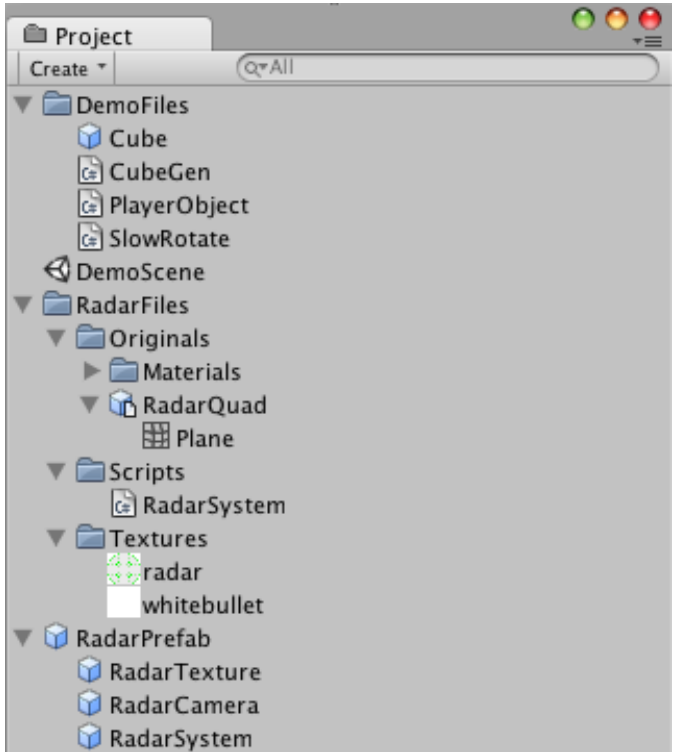
m-space media

3D SPATIAL RADAR SYSTEM

instructions for use

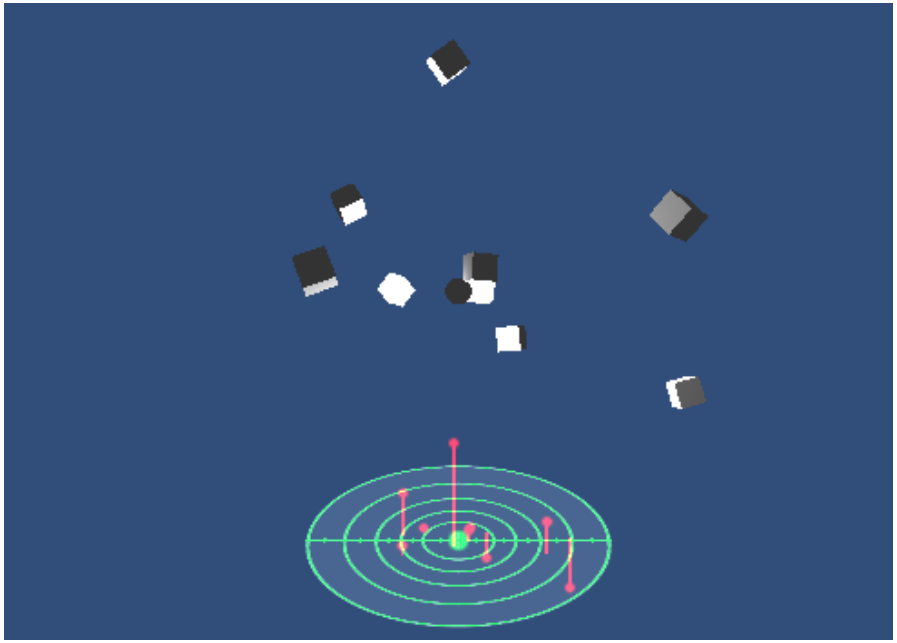
PACKAGE CONTENTS

- Radar Prefab
- Demonstration Scene
- Supporting files



GETTING STARTED

- Open '**Demonstration Scene**'.
- Hit '**Play**' in Editor.
- You will see randomly generated cubes and their corresponding radar blips and height lines.



SETTING UP - RADARSYSTEM

- Create a new layer, call it '**Radar**' - fig. 1
- Add Prefab to hierarchy - fig. 2
- Select '**Radar System**' - fig. 3
- Assign new Blip texture if changing.
- Assign the object to act as the '**Center**'.
- Specify Radar Scale:
  Use this to determine how closely distributed the radar blips are relative to the overall game area.
- Choose whether the Radar blips fade based on distance from center object.

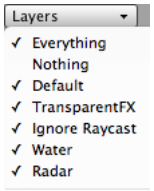


fig. 1

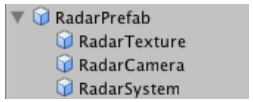


fig. 2

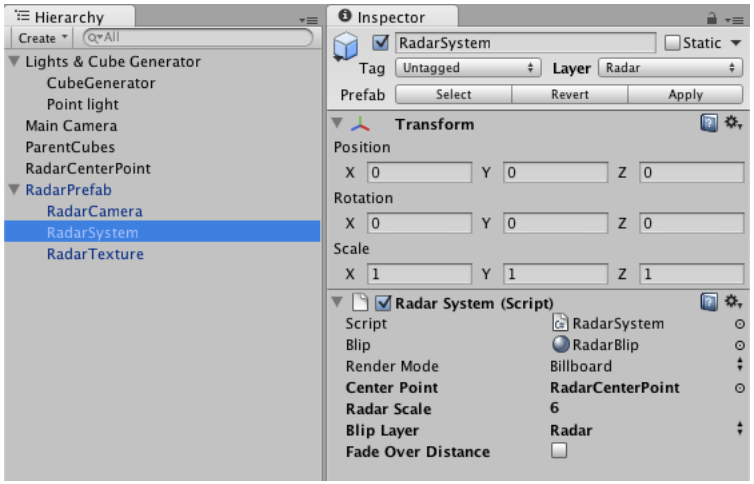


fig. 3

SETTING UP - SCENE CAMERA

- Choose the '**Main Camera**'.
- Remove the '**Radar**' layer from the culling mask - fig. 1

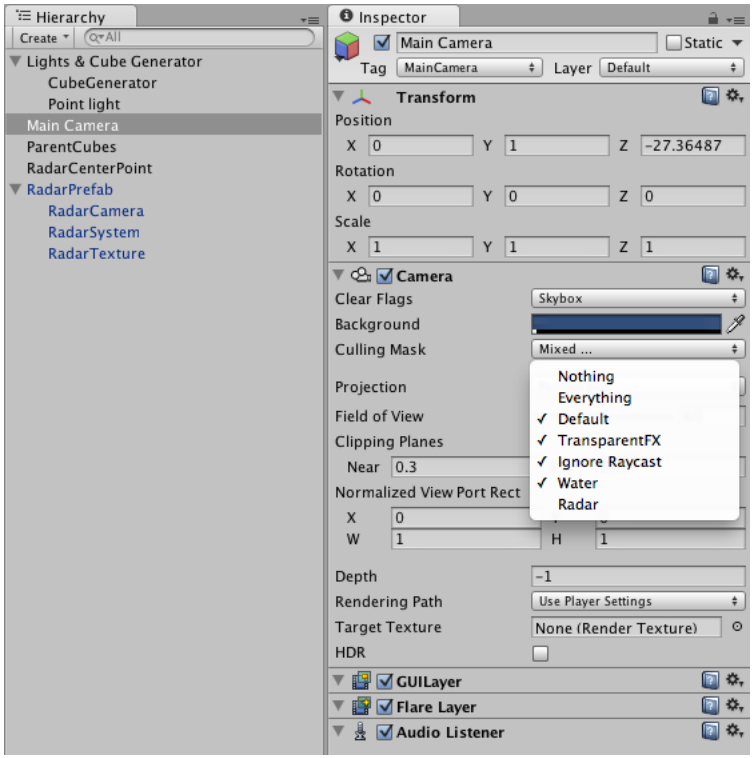


fig. 1

SETTING UP - RADAR CAMERA

- Choose the '**Radar Camera**'.
- Set '**Clear Flags**' to '**Depth Only**' - fig. 1
- Set '**Culling Mask**' to '**Radar**' only.

- You can alter any variables to suit your games needs.

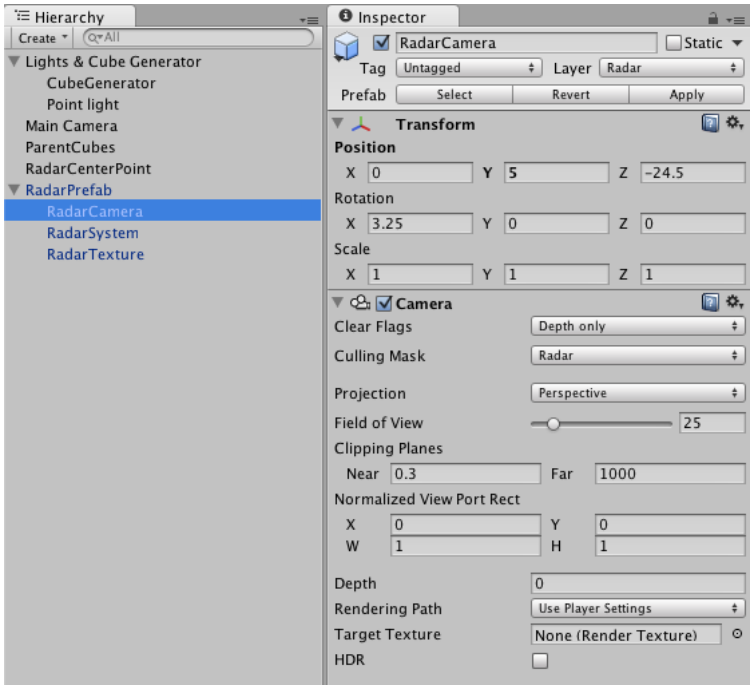


fig. 1

SETTING UP - CREATING BLIPS

- Add a script to the object that will generate a Radar blip.
- Add the RadarSystem script to create an Inspector variable - fig. 1
- Drag and drop the RadarSystem Gameobject into the inspector variable - fig. 2
- In the objects script, use the AddRadarBlip method to create a new RadarBlip - fig. 3

  Specify the object, its color and its size.

- Other methods are available see the RadarSystem script for details.

```
8
9       // Used to provide Inspector Variable in Editor
10      public RadarSystem _RadarSystem;
11
```

fig. 1

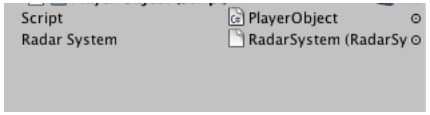


fig. 2

```
29      // Attach this object to the Radar list - Specifies Object, Color and Size
30      _RadarSystem.AddRadarBlip (cuboid, Color.red, 0.5f);
```

fig. 3