Unity Isometric Map

This is a 2d isometric map plugin.

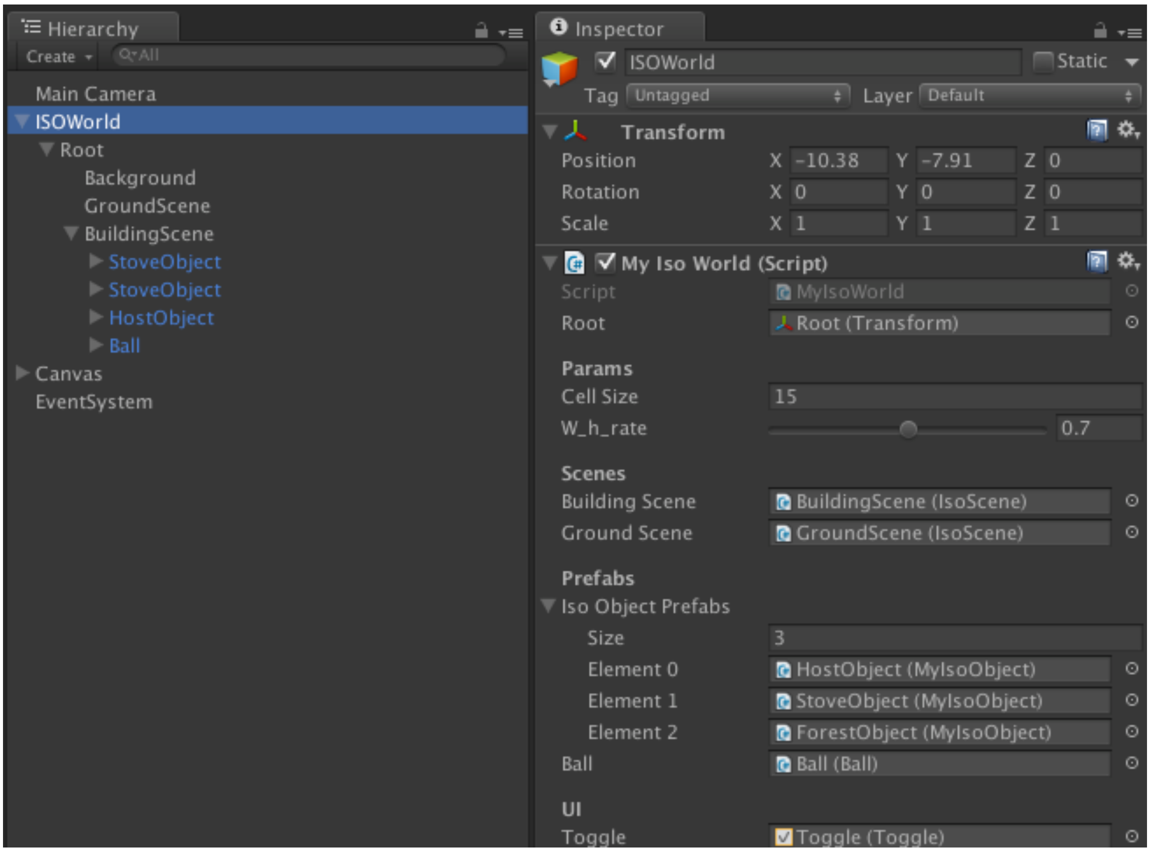
- Custom Map Angles.
- Z Order Sort.
- Contain Astar(Binary Search Path).
- Easy To Extend.
- Full Script.

Contact :173556135@qq.com

Guide

• Create isometric world map

- Add a game object on scene.
- Add `MyISOWorld` script. (The MyISOWorld is inherit from IsoWorld)
- Add a child game object name by `Root`. (The ‘Root’ can scale)



Main Camera
ISOWorld
  Root
    Background
    GroundScene
    BuildingScene
      StoveObject
      StoveObject
      HostObject
      Ball
Canvas
EventSystem

ISOWorld ☑ — □ Static
Tag: Untagged — Layer: Default

Transform

	X	Y	Z
Position	-10.38	-7.91	0
Rotation	0	0	0
Scale	1	1	1

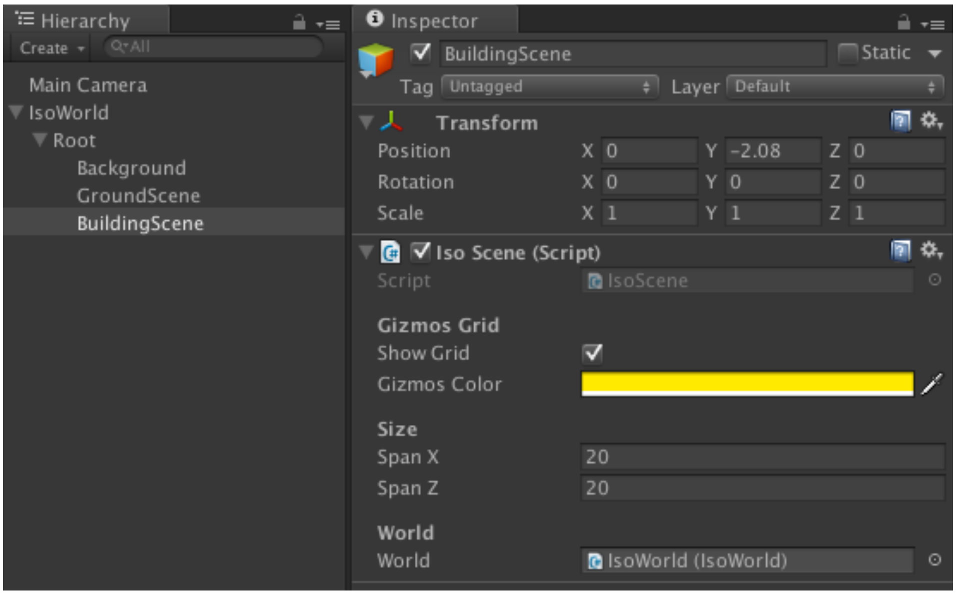
☑ My Iso World (Script)

Script	MyIsoWorld
Root	Root (Transform)
Params	
Cell Size	15
W_h_rate	0.7
Scenes	
Building Scene	BuildingScene (IsoScene)
Ground Scene	GroundScene (IsoScene)
Prefabs	
Iso Object Prefabs	
Size	3
Element 0	HostObject (MyIsoObject)
Element 1	StoveObject (MyIsoObject)
Element 2	ForestObject (MyIsoObject)
Ball	Ball (Ball)
UI	
Toggle	Toggle (Toggle)

Cell Size: The map grid cell size.
W_h_rate: Aspect ratio of diamond lattices.

• Add IsoScene

- Add scenes.
- Add IsoScene script.



Main Camera
IsoWorld
  Root
    Background
    GroundScene
    BuildingScene

BuildingScene ☑ — □ Static
Tag: Untagged — Layer: Default

Transform

	X	Y	Z
Position	0	-2.08	0
Rotation	0	0	0
Scale	1	1	1

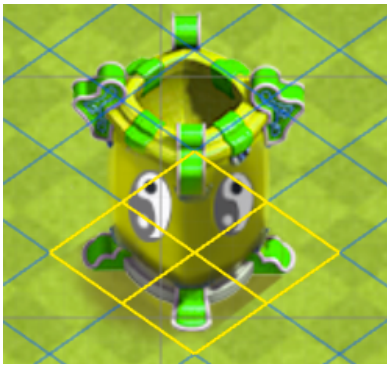
☑ Iso Scene (Script)

Script	IsoScene
Gizmos Grid	
Show Grid	☑
Gizmos Color	🟨
Size	
Span X	20
Span Z	20
World	
World	IsoWorld (IsoWorld)

SpanX & SpanZ : The grid size.
World : Reference to the world.

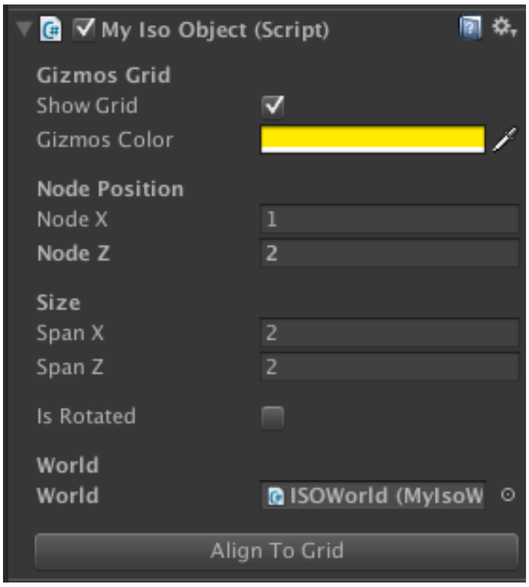
- **Create my isometric prefab**

  - Create game object and add your isometric object script. (You isometric object script inherit from IsoObject)
  - Add sprite below and set to correct position.



  - Set the parameters.

Size : The isometric object's size.
X and Z can be different.
Is Rotated: Rotate the isometric object.
NodeX & NodeZ : The isometric object's grid position when it was on the map.



My Iso Object (Script)

Gizmos Grid
Show Grid ☑
Gizmos Color 🟨

Node Position
Node X 1
Node Z 2

Size
Span X 2
Span Z 2

Is Rotated □

World
World ISOWorld (MyIsoW

  - Save the prefab.

- **Initial map**

  - Drag the isometric object to the scene.
  - Set the correct position.
  - Click the `Align To Grid` Button . (See the My Iso Object picture)

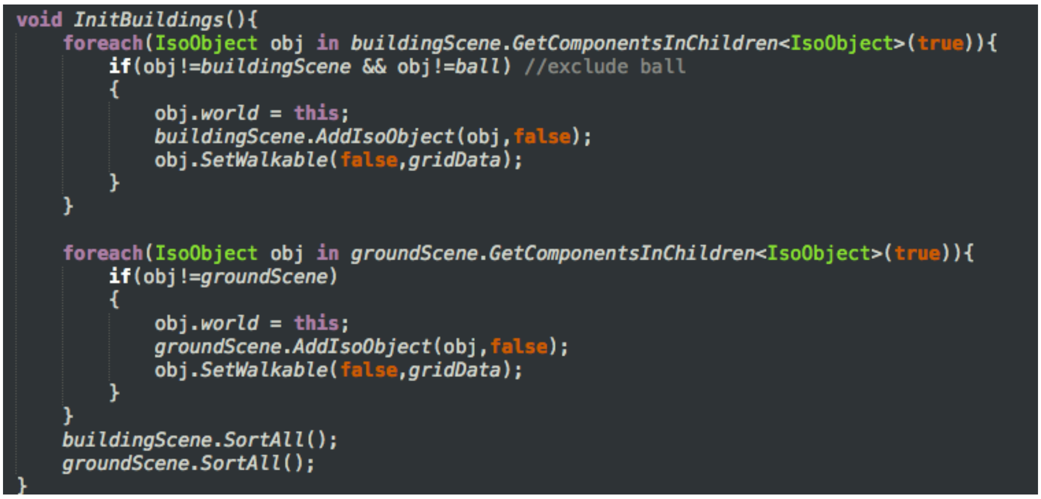
- **Initial Game**

  - Init PathGrid. The path grid data is a grid data. It records the usage information of the grid. You can create multiply path grid data.

```
eg:     gridData = new PathGrid(buildingScene.spanX,buildingScene.spanZ);
        gridData.SetAllWalkable(true);
```

- Init AStar

- Init Buildings



```
void InitBuildings(){
    foreach(IsoObject obj in buildingScene.GetComponentsInChildren<IsoObject>(true)){
        if(obj!=buildingScene && obj!=ball) //exclude ball
        {
            obj.world = this;
            buildingScene.AddIsoObject(obj,false);
            obj.SetWalkable(false,gridData);
        }
    }

    foreach(IsoObject obj in groundScene.GetComponentsInChildren<IsoObject>(true)){
        if(obj!=groundScene)
        {
            obj.world = this;
            groundScene.AddIsoObject(obj,false);
            obj.SetWalkable(false,gridData);
        }
    }
    buildingScene.SortAll();
    groundScene.SortAll();
}
```

API

- **IsoObject**

```
- void Sort();.
- void RotateX( bool value);
- void UpdateFrame();
- void UpdateSpanPos();
- void SetNodePosition(int nodeX,int nodeZ);
- void SetWalkable(bool value,PathGrid grid);
- bool GetWalkable( PathGrid grid);
- bool GetRotatable(PathGrid grid);
- void UpdateScreenPos();
- void Destroy();
```

- **IsoScene**

  - void *AddIsoObject*(IsoObject obj,bool isSort = **true**)
  - void *RemoveIsoObject*(IsoObject obj)
  - void *SortIsoObject*(IsoObject obj)
  - void SortAll()
  - IsoObject GetIsoObjectByNodePos(int nodeX,int nodeZ)
  - List<IsoObject> *GetIsoChildren*()
  - void Clear()
  - void Destroy()

- **IsoWorld**

  - void InitScene()
  - Vector2 LocalPosToGridPos(float px , float py , float offsetX=0 , float offsetY=0)
  - void Clear()
  - void Destroy()

- **PathGrid**

  - void ChangeSize(int gridX,int gridZ)
  - void SetAllWalkable(bool value)
  - List<PathNode> GetNodesByWalkable(bool walkable)
  - bool CheckInGrid(int nodeX, int nodeZ)
  - PathNode *GetNode*(int nodeX,int nodeZ)
  - void SetWalkable(int nodeX, int nodeZ,bool value)
  - void CalculateLinks(int type=0)   //for astar
  - PathGrid *Clone*()
  - void Destroy()

- **AStar**

  - bool *FindPath*(int startNodeX,int startNodeZ,int endNodeX,int endNodeZ)
  - List<PathNode> *path*

    How to use astar:

```
        gridData.CalculateLinks();//when map is changed
        if(astar.FindPath(ball.nodeX,ball.nodeZ,nodeX,nodeZ)){
                ball.MoveByRoads(astar.path);
        }
```