## PROCEDURAL GENERATION GRID

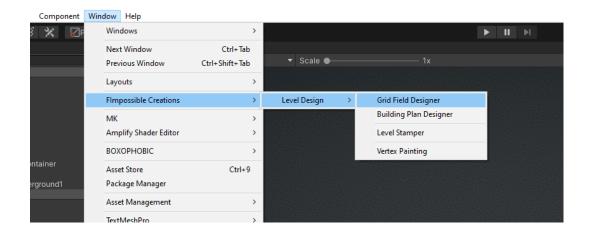
### **QUICK START**

Before using this package you should check the user manual, or watch tutorial videos on Youtube.

If you want example setups and assets import PGG Demos.unitypackage

You can start creating your own procedural generation field setup through:

"Window->FImpossible Creations->Level Design->Grid Field Designer"



Watch this fast overview of working with the packages video here!

# You can find this PGG presets scheme useful, but I recommend checking the full manual and watching tutorial videos;)







Field Setup can have multiple:





#### **GRID**

Simple positions list like (1,0,0) (2,0,0) etc. Just ones considered as inside grid

Cell positions and rotations with additional instructions like "Door Hole" etc.

#### FIELD SETUP

Contains all spawn rules (in Field Modifications)

Defines single cells scale

Runs all spawn rules on provided grid cells

Contains cell commands definitions which can be used with grid's additional instructions

# MODIFICATION PACKS

**Contains multiple Field Modifictions** 

Can be referenced and used by any other field (except field's root pack)

Can provide additional parameters to all modifications inside (in future)



Modification Pack have multiple Field Modificators



Modification can have multiple Field Spawners



Field Spawner generates objects with logics defined by Spawn Rules



#### FIELD MODIFICATION

User adds references to prefababs for spawning

User adds Field Spawners

Contains parameters inherited by all spawners like field tag etc.

#### FIELD SPAWNER

Takes prefab reference to spawn out of Field Modification

Can spawn random prefabs

User adds rules to define spawning logics

Field Modification runs spawners on each grid cell

#### SPAWN RULE

Uses grid cells to define spawn logics

Prevents or allows to spawn objects in cell

User can code custom spawn rules