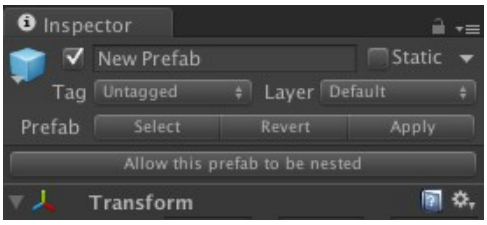
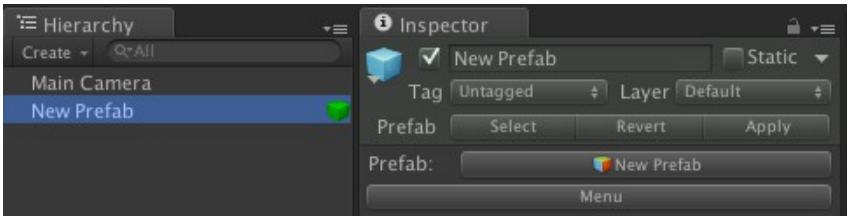
Prefab Evolution.

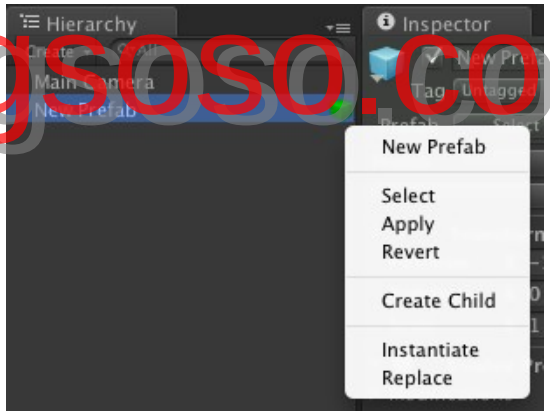
To allow any prefab to be nested or inherited, just select it in scene, and click «Allow this prefab to be nested» in Inspector window.



Now this prefab can be nested in another prefab. Green cube icon after prefab in Hierarch window shows that all changes applied to this prefab will be propagated to all nested instances of this prefab, when you click Apply button in Inspector window.



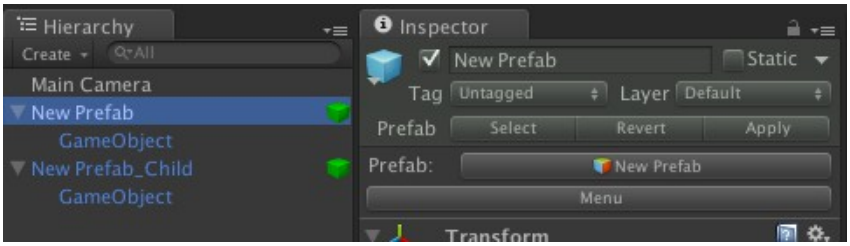
You can click on the green cube in Hierarchy or click Menu button in inspector window to show menu.



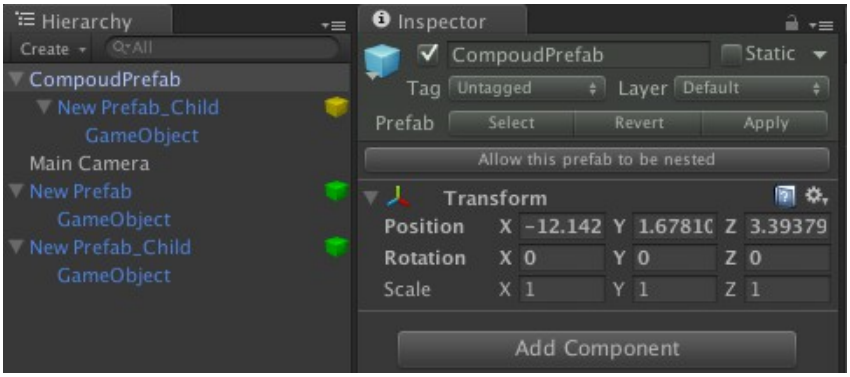
Click «Create Child» menu item to create prefab that will be inherited from current prefab. This prefab will apear in your project with «_Child» prefix and also will be instantiated in the scene.

«Child» prefabs are just like nested prefabs, and they will receive all changed applied to theirs parent. Also «Child» prefabs have button with name of its parent, click on it parent prefab will be highlighted in Project window.

Now lets place in «New Prefab» new GameObject and click «Apply» button in inspector, or in prefab menu. This changes will be applied to the child prefab.

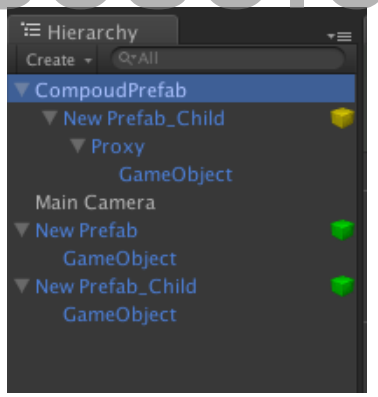


Lets create a normal prefab, put «New Prefab_Child» inside it, and click «Apply» button.

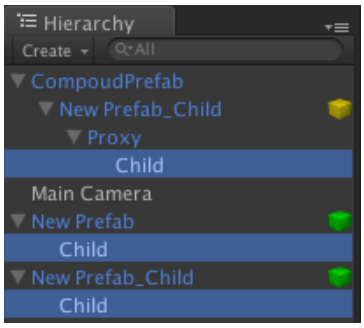


Yellow cube show that this is a nested prefab, and any changes applied to this instance will not affect another instances of «New Prefab_Child», but will be overridden. You can make any changes to nested instance or to the child prefabs. For example you can create or remove new Game Objects or Components, or also you can change parent transforms of any object inside this instance.

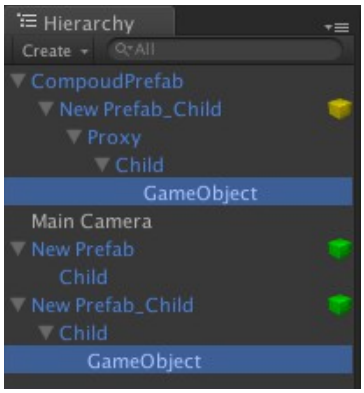
Lets place new «Proxy» Game Object in «New Prefab_Child» instance, and place «GameObject» as child.



Now lets rename «GameObject» inside «New Prefab» to «Child», and apply changes to «New Prefab». All instances of «GameObject» change theirs names to «Child». As you can see that «Proxy» GameObject still inside «New Prefab_Child» instance, and «Child» still child of the «Proxy» GameObject.



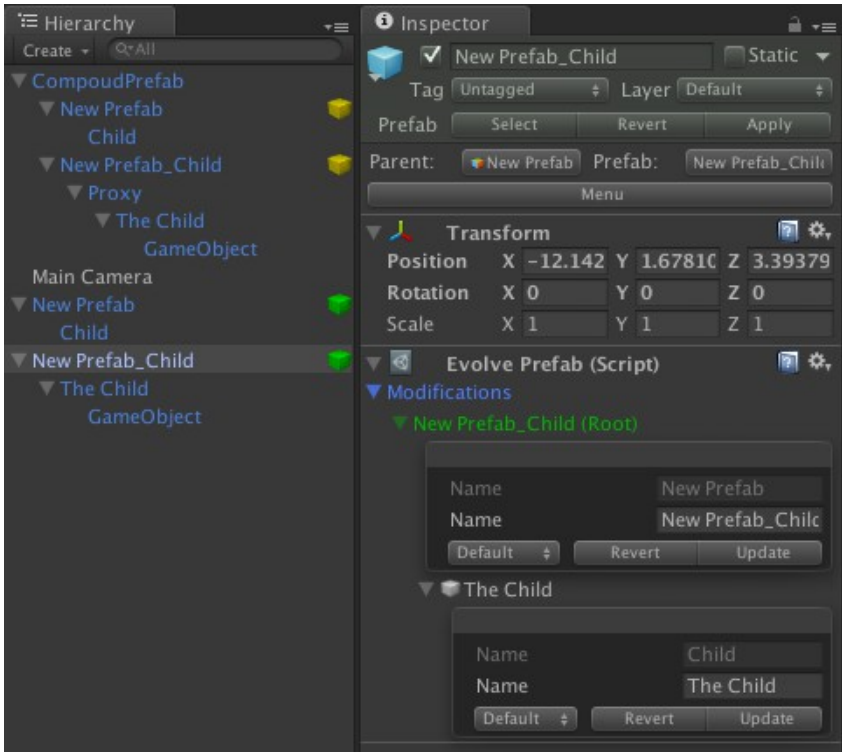
Now put new GameObject to the «Child» GameObject of «New Prefab_Child» and Apply changes to «New Prefab_Child». The GameObject will appear on nested instance of «New Prefab_Child», but, obviously, not on «New Prefab», because «New Prefab_Child» just extend «New Prefab» and can't apply any changes to their parent. Also nested prefab instances can't apply changes to theirs prefabs.



Place instance of «New Prefab» inside «Compoud Prefab» and apply changes to «Compoud Prefab»

Than rename «Child» GameObject of «New Prefab_Child» prefab to «The Child», and Apply changes to «New Prefab_Child» prefab.

Now name property of the child object of prefab «New Prefab_Child» is marked as modified, and you can see this in Inspector window.



Now rename «Child» GameObject of the «New Prefab» prefab to «A Child» and Apply changes to «New Prefab» prefab. As you can see changes was applied only to nested instance of «New Prefab» inside «Compoud Prefab» but not on «New Prefab_Child». Because «New Prefab_Child» is override property.

There is three property override modes: Default, Keep and Ignore. Default mode mean that overridden property can be reseted to prefab value if parent prefab changes this property to the same value. For example if you rename «A Child» GameObject of «New Prefab» to «The Child», than «New Prefab_Child» wont override this property in future and will apply changes to this property if Parent prefab change it. So we will do that. Change «A Prefab» name to «The Prefab» and Apply changes to «New Prefab».

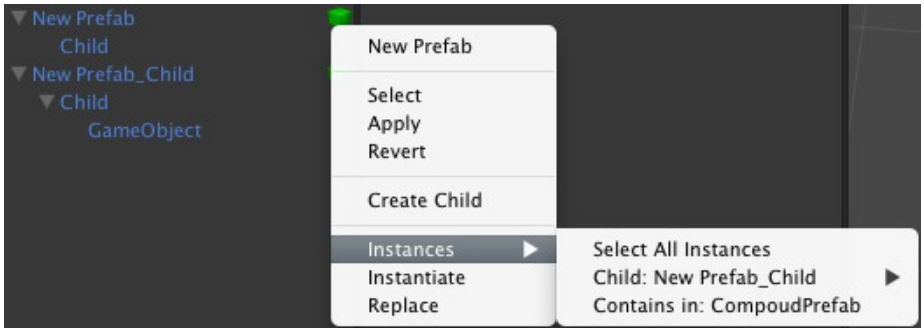
And than change name to «Child» and apply «New Prefab».

«New Prefab_Child» ignore name property override and change name too.

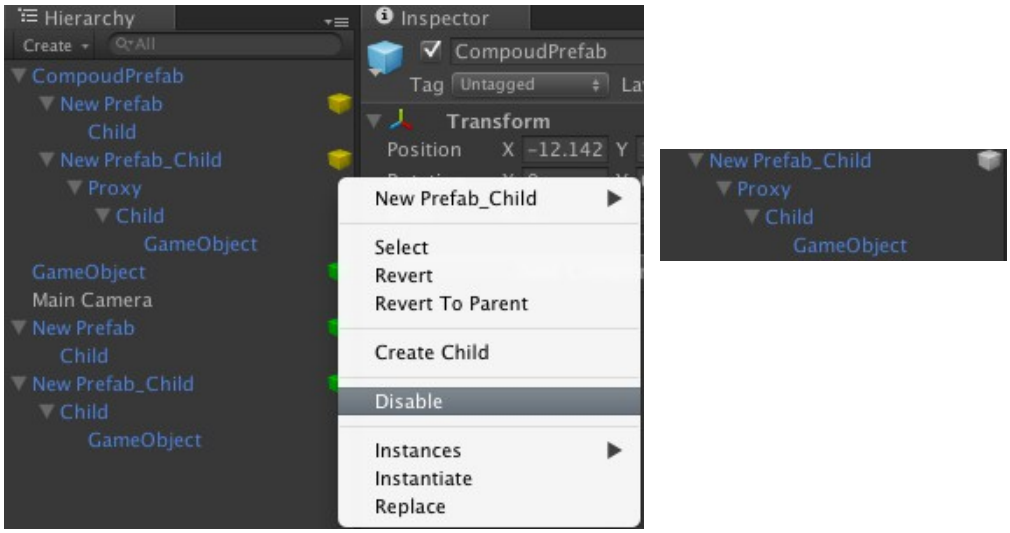
«Keep» mode — mean that the property will never discard override.

But properties with «Ignore» mode will always take value from parent prefab, when you apply changes to parent prefab.

In the prefab menu you can ever see what prefab depends from it.



Any nested prefab instance can be disabled, to prevent changes of the current instance, when the parent prefab changed.



Imported Models

Any imported model will behave like nested prefabs. This mean that you can nest you model, inside another prefab, or inherit from it. After reimport all instances of you model will be updated. This future have one limitation: All the bones of your model must have unique names. This is because it is the only way to determine the hierarchical changes.

Also you can disable this feature in Preferences window(Unity → Preferences → Prefab Evolution).

Migration to source code version:

If you previously use this plugin without included source code, you should convert all you prefabs that use EvolvePrefab script. Open Unity editor Unity → Preferences → Prefab Evolution and click “Migrate”. This operation will replace all references from EvolvePrefab script that imported from dlls to script from EvolvePrefab.cs.