#### Advanced NavMesh techniques to improve Unity Navigation

Objective

The main objective of this blog post is to give you a complete and advanced idea of what NavMesh can do inside Unity and how to speedup your complex navigation in Unity.

Step 1 Before starting NavMesh techniques to improve Unity Navigation

I hope you’ve gone through my previous blog & have a basic idea on Unity’s NavMesh. If you haven’t read it, I strongly recommend you to get a [Basic Idea About NavMesh](http://www.theappguruz.com/blog/learn-unity-navigation-using-navmesh).

Let's move on to those techniques now.

Step 2 Areas

An Area in a NavMesh is simply a part of the mesh with a cost. You can find Area tab in the Navigation window.



As you can see there are 3 built-in areas in Unity. You can change the cost values of all areas except Not Walkable *(because it’s Not Walkable, you can never go into that area regardless of the cost)*.

You can also define your own area if you need too. Some examples can be mud, water, sand, etc.

The cost of an area plays a decisive role in navigation as our agent (not 007, NavMesh agent ;)) will try to follow the path that has the least possible cost.

Let’s see how to use them now:

1. Select the object you want a different area on it.
2. Select area in object tab of navigation window.



You’ll see the effect after you bake the NavMesh.

Here’s a link you can refer:

* [NavMesh Baking - Unity Official Tutorials](https://www.youtube.com/watch?v=VcNly-cMZV4&amp;feature=youtu.be&amp;t=2m17s)

Step 3 Off-Mesh links

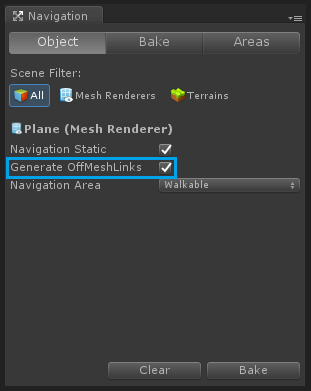
Off-Mesh links are links that creates a connection between two different meshes. In other words you can jump between multiple meshes using these links.

They can be generated automatically or you can also generate them by off-mesh link component if you want a better control over the navigation behaviour.

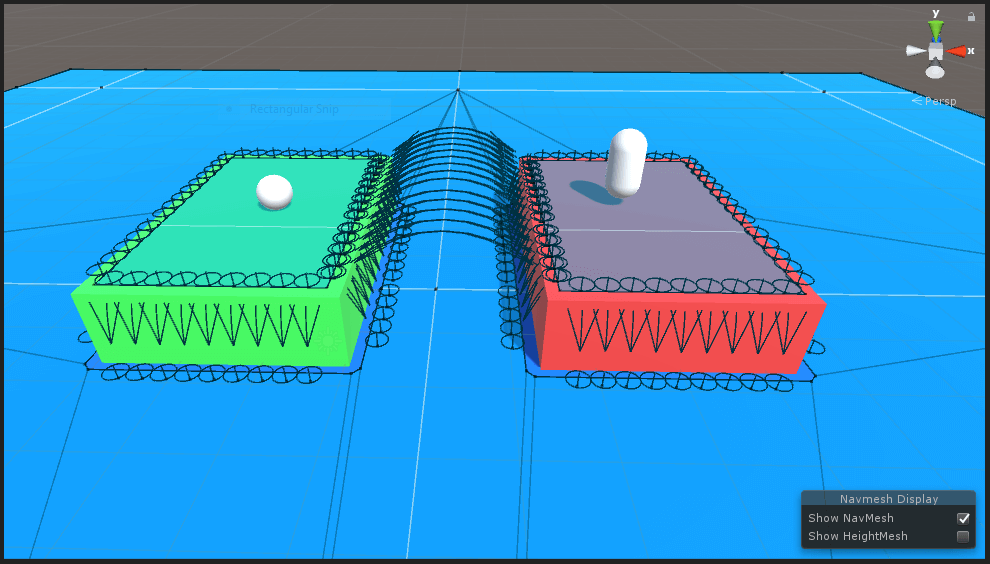
Let me show you what difference it can make while learning how to create them.

To create off-mesh links automatically:

1. Select the object from hierarchy you want links on in.
2. Check the Generate Off-mesh Links check box in Navigation window.

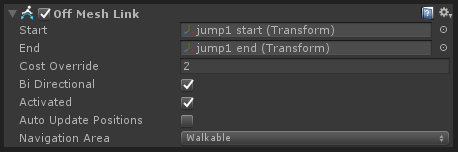


This image displays automatically generated Off-Mesh links.



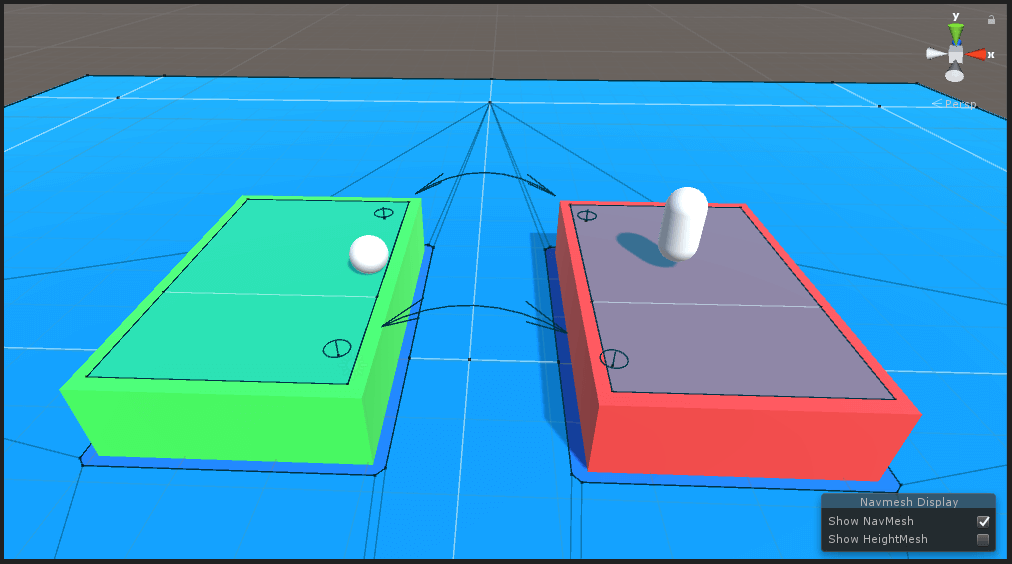
Now to create a link manually, follow these steps:

1. Create two empty GameObjects in hierarchy.
2. Add Off Mesh Link component to any one of them.



As you can see there are 2 variables named Start & End in it. Assign the respective GameObject in it.

This image shows a link that I created.



Note

If the link you created is not a valid link (both ends are placed on a mesh surface), then the color of link will turn into gray rather than black. To see the effect just move any one end object of the link upwards (+y direction) and notice the change in color.

I guess you got the difference by now.

If you opt for automatic link generation, Unity will create every possible link to connect meshes which might not be ideal for your case. If you want a specific route to be followed you can use Off Mesh Link component.

You can choose any of these two methods that suites your requirement.

See how it looks in action here:

* [Unity - Off-mesh Links Tutorial](https://unity3d.com/learn/tutorials/topics/navigation/mesh-links)

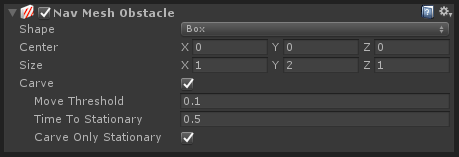
Step 4 NavMesh Obstacles

Until now we were dealing with static objects that don’t move around when the game is running.

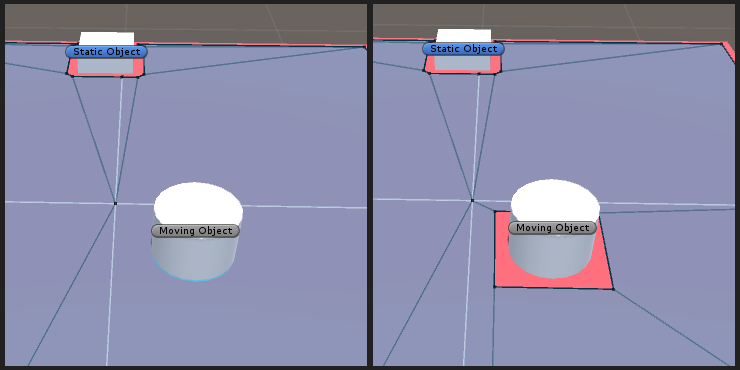
But what if we have a moving object that we need to be a part of our navigation mesh?

*NavMesh Obstacle component is the answer to that. The process is simple.*

*First add NavMesh Obstacle component to the object.*



You won’t see any difference now. Check the carve option and you should be seeing a hole in the mesh.



Character with NavMesh Agent component will avoid this part of navmesh in pathfinding process.

Some examples can be patrolling enemies or moving doors.

See the obstacles working live here:

* [Unity - NavMesh Obstacles Tutorial](https://unity3d.com/learn/tutorials/topics/navigation/navmesh-obstacles)

Step 5 Conclusion

*I hope you got all the things i wanted to pass on about Unity’s NavMesh. This is where I end this post and you start experimenting on navigation meshes.*

*Try it on your game and don’t forget to drop a comment if you have any query or suggestions for this post.*

Got an Idea of Game Development? What are you still waiting for? [Contact us](http://www.theappguruz.com/contact-us) now and see the Idea live soon. Our company has been named as one of the best [Unity 3D Game Development](http://www.theappguruz.com/3d-game-development) Company in India.

- See more at: http://www.theappguruz.com/blog/navmesh-techniques-to-improve-unity-navigation#sthash.nm7Y7I3c.dpuf