

Behavior Designer - Movement Pack

Table of Contents

Behavior Designer - Movement Pack	2
--	----------

Behavior Designer - Movement Pack

The Behavior Designer - Movement Pack includes 17 tasks focused on movement. The default set of Movement Pack tasks use Unity's [navigation mesh](#) to traverse the world. The Movement Pack doesn't do the actual movement - it instead sets the destination for the underlying pathfinding implementation (Unity's NavMesh, A* Pathfinding Project, etc).

Integrations

You can download the Movement Pack integrations from this page. After you have imported an integration you can start to use the integration tasks by adding them to your behavior tree. For example, if you want to use the A* Pathfinding Project version of the seek task you would add the task located under Actions -> Movement -> A* Pathfinding Project -> Seek.

Common Questions

Why does the agent wobble at the destination point instead of stopping?

If the agent doesn't stop moving at the destination point the *Stopping Distance* parameter on the [NavMeshAgent](#) is too small. Try increasing this value. We have had good results with a value of 0.1.

Why doesn't Can See Object detect the target object?

When Can See Object tries to detect the target object it fires a raycast from the agent's pivot position to the target's pivot position. If the pivot position is located beneath the ground collider then Can See Object will not be able to see the object and return a status of failure. This can be solved by increasing the y value of the *Offset* and *Target Offset* field of the Can See Object Task.

Does the Movement Pack work in 3D space for a space or underwater game?

Pathfinding in 3D space is computationally expensive so there are no pathfinding implementations built into Unity or the Asset Store which can do it. While no true 3D pathfinding implementation exists there are a few workarounds:

- Fake it by placing an invisible plane that the pathfinding implementation uses. The relative y position can then be adjusted with a separate height adjusting component and the agent will appear to be moving in 3D space.
- In certain pathfinding implementations (such as Unity's NavMesh) the height offset can be adjusted. This is similar to the above workaround except it doesn't require the position to be manually updated.
- Do not use any pathfinding implementation at all and instead use basic steering behaviors with local avoidance. This would have to be done using custom tasks as the Movement Pack doesn't support steering behaviors as an implementation.