Thank you for your purchase! The most recent documentation can be found online. If you have any questions feel free to post on the forums or email support@opsive.com.

Move Towards

The Move Towards task will move the agent towards the target (without pathfinding).

speed
The speed of the agent

arriveDistance
The agent has arrived when the square magnitude is less than this value

lookAtTarget
Should the agent be looking at the target position?

maxLookAtRotationDelta
Max rotation delta if lookAtTarget is enabled

targetTransform
The transform that the agent is moving towards

targetPosition
If target is null then use the target position

Rotate Towards

The Rotate Towards task will rotate the agent towards the target.

rotationEpsilon
The agent is done rotating when the square magnitude is less than this value

maxLookAtRotationDelta
Max rotation delta

targetTransform
The transform that the agent is rotating towards

targetRotation
If target is null then use the target rotation

Seek

The Seek task will move the agent towards the target with pathfinding.

speed
The speed of the agent

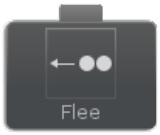
angularSpeed
Angular speed of the agent

arriveDistance
The agent has arrived when the square magnitude is less than this value

targetTransform
The transform that the agent is moving towards

targetPosition
If target is null then use the target position

Flee



The Flee task will move the agent away from the target with pathfinding.

speed
The speed of the agent

angularSpeed
Angular speed of the agent

fleedDistance
The agent has fleed when the square magnitude is greater than this value

lookAheadDistance
The distance to look ahead when fleeing

targetTransform
The transform that the agent is fleeing from

targetPosition
If target is null then use the target position

Pursue

The Pursue task is similar to the Seek task except the Pursue task predicts where the target is going to be in the future. This allows the agent to arrive at the target earlier than it would have with the Seek task.

speed
The speed of the agent

angularSpeed
Angular speed of the agent

arriveDistance
The agent has arrived when the square magnitude is less than this value

targetDistPrediction
How far to predict the distance ahead of the target. Lower values indicate less distance should be predicated

targetDistPredictionMult
Multiplier for predicting the look ahead distance

targetTransform
The transform that the agent is pursuing

Evade

The Evade task is similar to the Flee task except the Evade task predicts where the target is going to be in the future. This allows the agent to flee from the target earlier than it would have with the Flee task.

speed
The speed of the agent

angularSpeed
Angular speed of the agent

fleedDistance
The agent has fleed when the square magnitude is greater than this value

lookAheadDistance
The distance to look ahead when fleeing

targetDistPrediction
How far to predict the distance ahead of the target. Lower values indicate less distance should be predicated

targetDistPredictionMult
Multiplier for predicting the look ahead distance

targetTransform
The transform that the agent is evading

Patrol

The Patrol task moves from waypoint to waypint.

speed
The speed of the agent

angularSpeed
Angular speed of the agent

arriveDistance
The agent has arrived when the square magnitude is less than this value

randomPatrol
Should the agent patrol the waypoints randomly?

waypointPauseDuration
The length of time that the agent should pause when arriving at a waypoint

waypoints
The waypoints to move to

Cover

The Cover task will move the agent into cover from its current position.

speed
The speed of the agent

angularSpeed
Angular speed of the agent

arriveDistance
The agent has arrived when the square magnitude is less than this value

maxCoverDistance
The distance to search for cover

maxRaycasts

The maximum number of raycasts that should be fired before the agent gives up looking for an agent to find cover behind

rayStep
How large the step should be between raycasts

coverOffset
Once a cover point has been found

lookAtCoverPoint
multiply this offset by the normal to prevent the agent from hugging the wall

rotationEpsilon
Should the agent look at the cover point after it has arrived?

maxLookAtRotationDelta
The agent is done rotating to the cover point when the square magnitude is less than this value

---

Wander

The Wander task moves the agent randomly throughout the map with pathfinding.

speed
The speed of the agent

angularSpeed
Angular speed of the agent

arriveDistance
The agent has arrived when the square magnitude is less than this value

wanderDistance
How far ahead of the current position to look ahead for a wander

wanderRate
The amount that the agent rotates direction

Search

The Search task will search the map by wandering until it finds the target. It can find the target by seeing or hearing the target.

speed
The speed of the agent

angularSpeed
Angular speed of the agent

arriveDistance
The agent has arrived when the square magnitude is less than this value

wanderDistance
How far ahead of the current position to look ahead for a wander

wanderRate
The amount that the agent rotates direction

fieldOfViewAngle
The field of view angle of the agent (in degrees)

viewDistance
The distance that the agent can see

senseAudio
Should the search end if audio was heard?

hearingRadius
How far away the unit can hear

objectLayerMask
The LayerMask of the objects that we are searching for

linearAudibilityThreshold
The furtuer away a sound source is the less likely the agent will be able to hear it. Set a threshold for the the minimum audibility level that the agent can hear

objectFound
The object that is within sight

Within Distance

Check to see if the any object specified by the object list or tag is within the distance specified of the current agent.

magnitude
The distance that the object needs to be within

lineOfSight
If enabled the object must be within line of sight to be within distance. If this option is enabled then an object behind a wall will not be within distance even though it may be physically close to the other object

objects
An array of objects to check to see if they are within distance

objectTag
If the object list is null then find the potential objects based off of the tag

foundObject
The object variable that will be set when a object is found what the object is

---

Can See Object

The Can See Object task is a conditional task that returns success when it sees an object in front of the current agent.

fieldOfViewAngle
The field of view angle of the agent (in degrees)

viewDistance
The distance that the agent can see

objectLayerMask
The LayerMask of the objects that we are searching for

objectInSight
The object that is within sight

Can Hear Object

The Can Hear Object task is a conditional task that returns success when it hears another object.

hearingRadius
How far away the unit can hear

objectLayerMask
The LayerMask of the objects that we are searching for

audibilityThreshold
The furtuer away a sound source is the less likely the agent will be able to hear it. Set a threshold for the the minimum audibility level that the agent can hear

objectFound
The object that is within sight

Flock

The Flock task moves a group of objects together in a pattern.

agents
All of the agents that should be flocking

speed
The speed of the agent

angularSpeed
Angular speed of the agent

neighborDistance
Agents less than this distance apart are neighbors

lookAheadDistance
How far the agent should look ahead when determine its pathfinding destination

alignmentWeight
The greater the alignmentWeight is the more likely it is that the agents will be facing the same direction

cohesionWeight
The greater the cohesionWeight is the more likely it is that the agents will be moving towards a common position

separationWeight
The greater the separationWeight is the more likely it is that the agents will be separated

---

Leader Follow

The Leader Follow task moves a group of objects behind a leader object.

agents
All of the agents that should be following

speed
The speed of the agent

angularSpeed
Angular speed of the agent

neighborDistance
Agents less than this distance apart are neighbors

leaderBehindDistance
How far behind the leader the agents should follow the leader

separationDistance
The distance that the agents should be separated

aheadDistance
The agent is getting too close to the front of the leader if they are within the aheadDistance

leader
The leader to follow

Queue

The Queue task will move a group of objects through a small space in an organized way.

agents
All of the agents that should be queuing

speed
The speed of the agent

angularSpeed
Angular speed of the agent

neighborDistance
Agents less than this distance apart are neighbors

separationDistance
The distance that the agents should be separated

maxQueueAheadDistance
The distance the the agent should look ahead to see if another agent is in the way

maxQueueRadius
The radius that the agent should check to see if another agent is in the way

slowDownSpeed
The multiplier to slow down if an agent is in front of the current agent

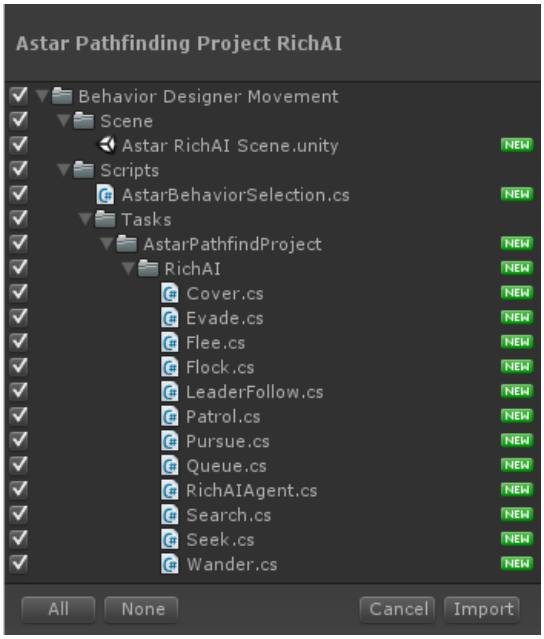
seekPosition
The target to see towards

A* Pathfinding Project Integration

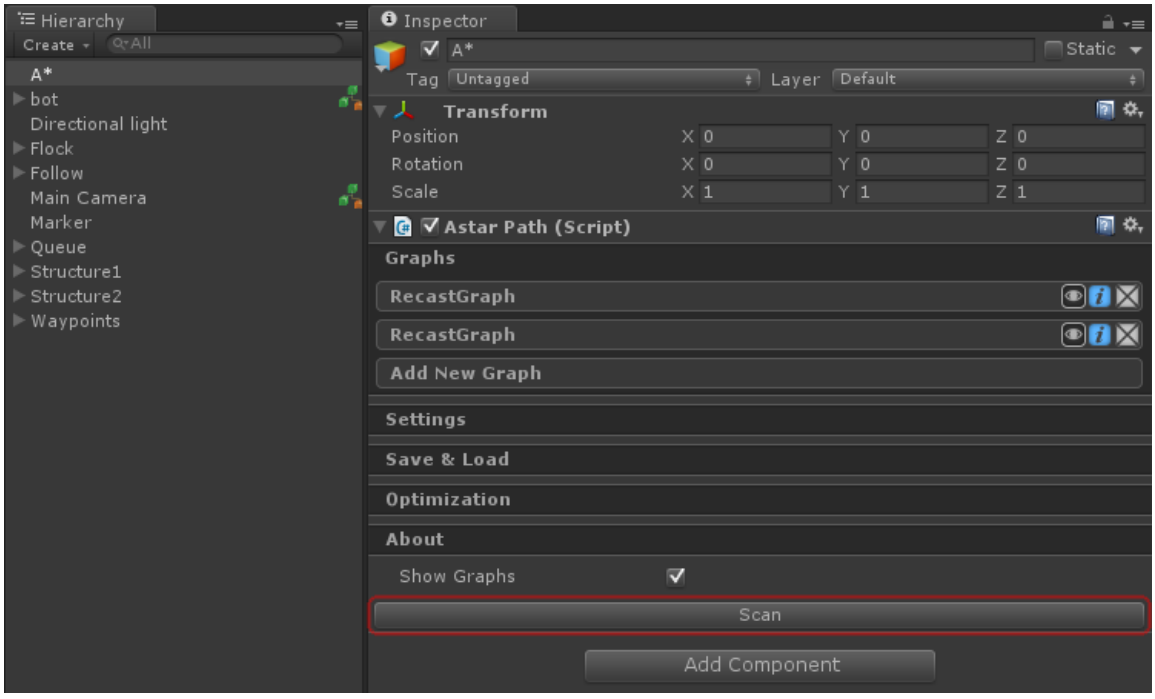
Any Movement task that involves pathfinding is integrated with the A* Pathfinding Project. The A* Pathfinding Tasks are located on the integrations page because the Movement Pack does not require the A* Pathfinding Project to work. Furthermore, there are two versions of the A* Pathfinding Project: a free version and a paid (Pro) version. Among other features, one of the differences is that the Pro version includes RichAI support whereas the free version only supports AIPath. The Movement Pack supports both of these implementations and they are located at /Behavior Designer Movement Pack/Third Party after importing the A* package from the integrations page.

Instead of adding the RichAI or AIPath component to your agent, add the RichAIAgent or AIPathAgent component to your agent. This will allow your A* agents to communicate with the tasks.

The RichAI scene requires one extra set to setup so we will be using that package within this example. The first step is to make sure Behavior Designer and the A* Pathfinding Project have already been imported. Following that, import Astar Pathfinding Project RichAI.unitypackage:



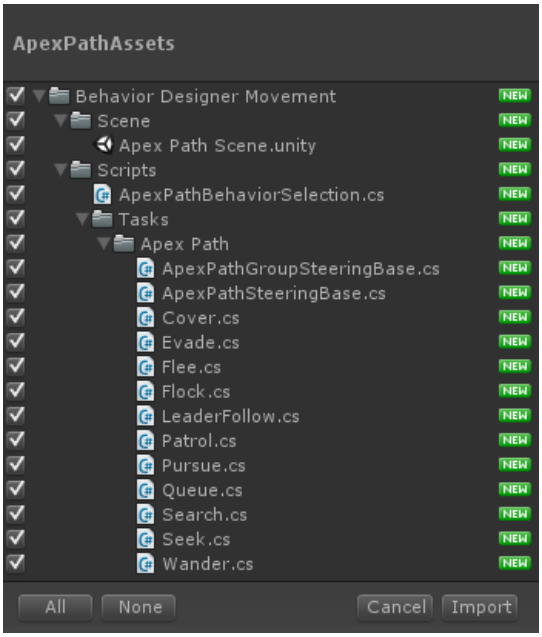
To run the RichAI demo scene, one extra steps is necessary. In our tests we have found that the navigation mesh doesn't save correctly when you export the scene to a Unity Package. As a result, after you open the scene you'll need to click the "Scan" button within the Astar Path component:



This is the only extra step required - after the navigation mesh has been generated all of the tasks will work correctly. As a reminder, the AIPath package does *not* need to go through this process.

Apex Path Integration

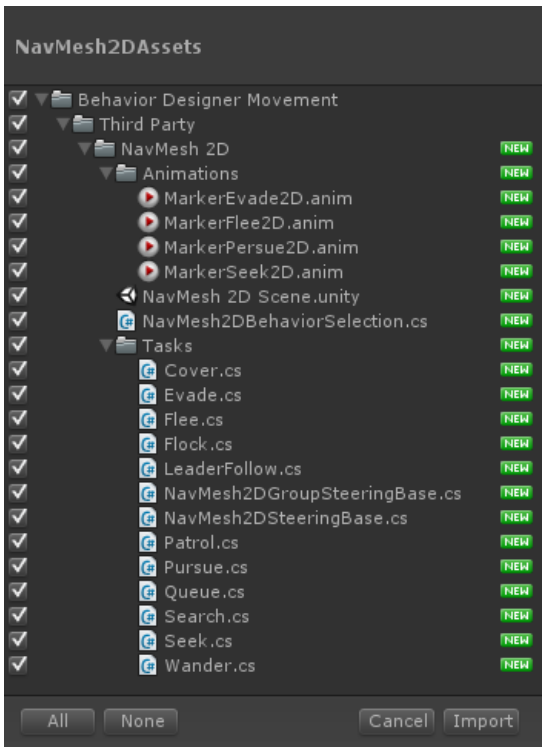
Any Movement task that involves pathfinding is integrated with Apex Path. The Apex Path files are located on the integrations page because the Movement Pack does not require Apex Path to work. The following tasks are imported for the Apex Path integration:



To use the Apex Path tasks, you must first add all of the Apex Path components to your agent. This can be added via the Components -> Apex -> QuickStarts -> Navigating Unit menu option.

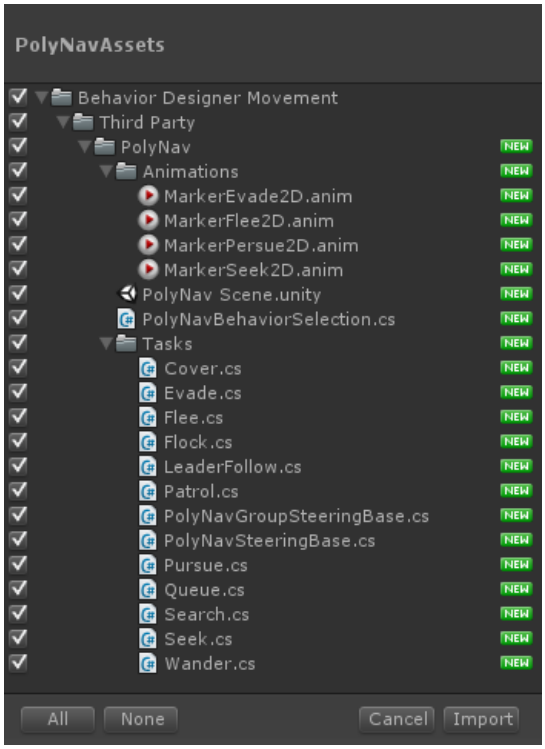
NavMesh 2D Integration

Any Movement task that involves pathfinding is integrated with NavMesh 2D. The NavMesh 2D files are located on the integrations page because the Movement Pack does not require NavMesh 2D to work. The following tasks are imported for the NavMesh 2D integration:



Poly|Nav Integration

Any Movement task that involves pathfinding is integrated with Poly|Nav. The Poly|Nav files are located on the integrations page because the Movement Pack does not require Poly|Nav to work. The following tasks are imported for the Poly|Nav integration:



Third Person Controller Integration

The Movement Pack is integrated with the Third Person Controller to allow characters to move with pathfinding while still using root motion. Unit's NavMesh, AIPath, and Apex Path are supported with this integration. The integration files can be downloaded from the integrations page

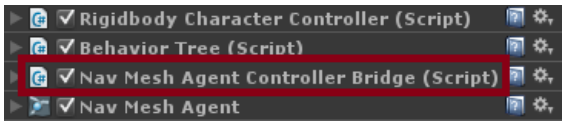
A bridge component needs to be added in order to synchronize the pathfinding velocity with the Third Person Controller. This bridge component is independent of the behavior tree so it will work with any of the Movement Pack tasks. The following bridge components are included with this integration:

NavMeshAgent Controller Bridge
Used with the Unity NavMesh

AIPathAgent Controller Bridge
Used with A* Pathfinding Project

MovableUnit Controller Bridge
Used with Apex Path

Assign one of these components to your GameObject which contains the behavior tree and the Third Person Controller will automatically synchronize with the pathfinding direction.



Support

We are here to help! If you have any questions/problems/suggestions please don't hesitate to ask. You can email us at support@opsive.com or post on the forum.