

Behaviour Tree Docs

Ivar Nuij

Links:

Github: [Link](#)

UML: [BehaviourTreeUML.svg](#)

StateDiagram: [BehaviourTreeStateDiagram.svg](#)

Ik heb ervoor “gekozen” om een behaviour tree te maken.

De tree bestaat uit een blackboard en lijst van ITickables. Deze ITickables zijn 1 van de volgende 4 nodes: Composite Node, Leaf Node, Decorator Node of een Condition Node.

Nodes:

Composite Node:

Deze node implementeert het ISequenceable interface. Dit betekent dat deze node een lijst aan ITickables opslaat. De behaviour trees root node is ook een composite node.

Leaf Node:

Je kan deze node ticken en dan geeft hij je een state terug.

Decorator node:

Is een leaf node maar bevat ook een child die de node ticked.

Condition Node:

Houd 2 lijsten van ITickables (Nodes) bij en ticked 1 van de 2 lijsten op basis van een argument.

Sequences:

In het sequences script kan je de sequences vinden die de behaviour trees kunnen uitvoeren. Dit zijn gewoon lange lijsten van ITickables die de behaviour tree uitvoert.

Blackboard:

De blackboard slaat een dictionary van blackboardData op. BlackboardData slaat een dictionary van T op. Hierdoor kan je elk type in het blackboard opslaan.

Reflectie / PMI:

Ik heb geprobeerd een systeem te maken dat modulair is maar door tijds tekort heb ik het een beetje moeten afraffelen. Hierdoor zijn de sequences niet echt leesbaar in code. Ik heb wel het gevoel dat ik er veel van heb geleerd en zou graag een keer een verbeterde implementatie schrijven.

Plus:

Je kan je AI veel complexer maken dan met de meeste andere oplossingen.

Min:

De sequences worden in code al snel onleesbaar.

Interesting:

Dat je uit zulke simpele nodes zo complex gedrag kan maken.

En nu heb ik nog 2 min om de documentatie in te leveren.