World_state vector

WS <isPlayerAlive, isPlayerInRange>

Both are Boolean and respectively check if the player is alive and if he is in range. As it is a simple model both primitives uses the WS vector to check for validity.

WS valid states

<true, true> - Player alive and in range, monster will attack

<true, false> - Player alive but not in range, monster wander around the level

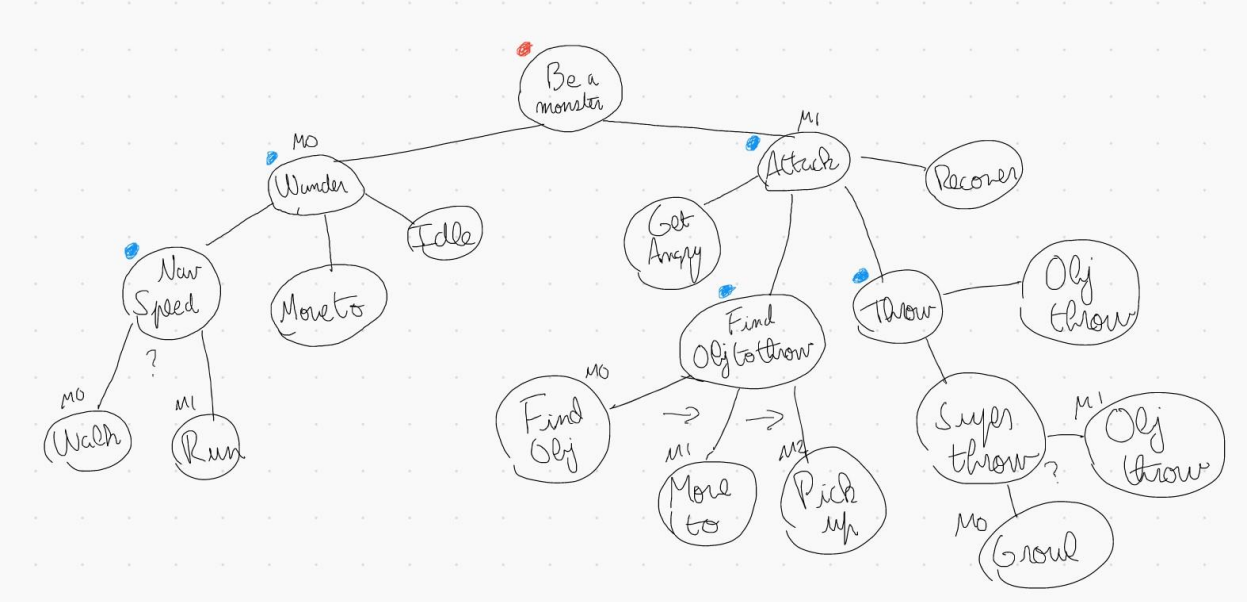
<false, true> - Player was just killed by the monster

WS invalid states

<false, false> - If player is dead, he was killed by the AI therefore he must be in range

The win/loss conditions are not controlled by the HTN as the planner is focused on the monster actions. If player killed the monster stops behaving but if the player wins the game ends all the processes.
<false, true

HTN tree



Made of a root node (red), compound nodes (blue) and primitive nodes (the rest). The root and compound nodes are defined as having a list of children nodes, these can be primitive or compound allowing for trees as complex as desired. Tree was built using polymorphism.