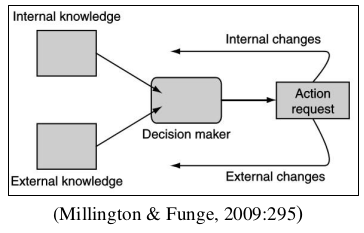
**DECISION MAKING**

* The input to the decision making system it the knowledge that the character processes, and the output is the action an action request.
* Knowledge is further broken down into two:
  + 1. Eternal knowledge

e.g. position of other characters, the layout of the level et c.

* + 1. internal knowledge

e.g. its health, its ultimate goal etc.

* External knowledge the information that the character knows about its surrounding in the game environment.
* Internal knowledge is information about the character’s internal states or thought process.
* Actions, correspondingly, can have two components:
  + 1. They can request an action that will Change the external state of the character (such as throwing a switch, firing a weapon, moving into a room) or
    2. Actions that only affect the internal state (see the diagram above). Changes to the internal state.

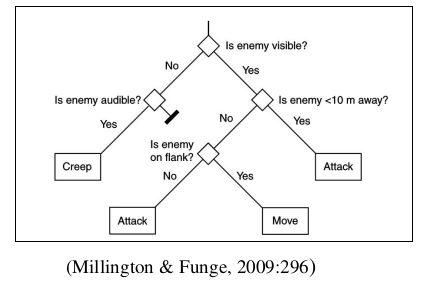
**Decision making tree**

* This is the simplest decision making technique.
* Extensions to algorithms can make them sophisticated, i.e. they can become complex.
* They are used to control game character and other in game decision making, such as animation.
* They are very modular and easy to create

**The problem**

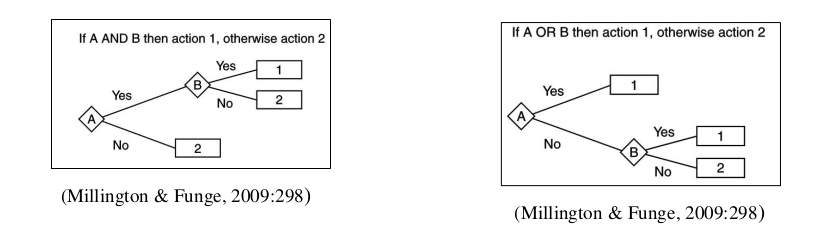
* We need a method that can easily group lots of actions inputs together under one action, while allowing the inputs that are significant to control the output.

**The algorithm**

* A decision making tree is made up of connected decision points. Has a starting decision(the root).For each decision, starting from the root, one of a set of ongoing options is chosen.
* The character makes decision based on its knowledge .
* An action is attached to each leaf of the decision making tree.

**Combinations of decisions**

* The decision tree is efficient because the decisions are typically very simple. Each decision makes only one test
* When Boolean combinations of tests are required, the tree structure represents this. To AND two decisions together, they are placed in series in the tree
* To OR two decisions together, we also use the decisions in series, but with the two actions swapped over from the AND example.



**Decision making complexity**

* Because decisions are made in trees, the number of decisions that need to be made is usually smaller than the number of decisions in a tree.

**Branching**

* The decision with two options is called the binary decision tree.
* Decision tree can more than two options.
* c