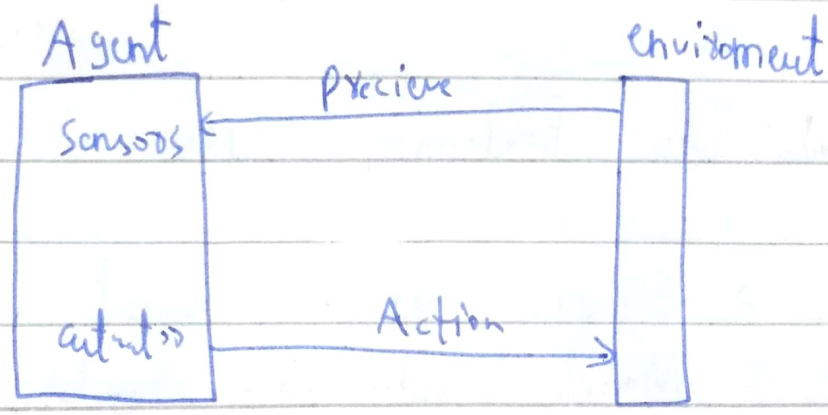
Agent :-

An agent precieves something from environment throught sensor and perform actions on environment throught actuators

-> The actions performed by an agent are calculated by a agent Fuction
It maps percept sequence to actions



-> An agent program implements the agent Fuction.

-> This program runs on some sort computing with physical sensors and actuators.

→ The key difference b/w agent program and agent function is that ~~agent~~

" agent program takes current percept as input but agent function takes entire entire percept history "

:: AGENT TYPES ::

Agents have Following types.

-> Simple reflex agents
-> Model based reflex agents / reflex agent with state
-> Goal based agent
-> Utility based agent.

Simple Reflex Agents :-

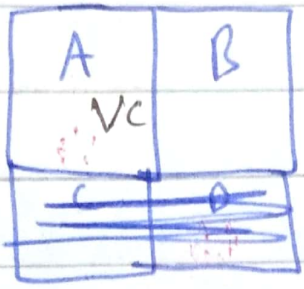
-> These agents perform actions on the basis of current percept.

-> These agent uses ~~cond~~ Condition Action Rules that maps a state to and action

if condition/state then Action.

Example

Consider the vaccum cleaner example where VC is a reflex agent.



It action will completly depends on the current state and ~~its~~ ~~actions can~~ it does nt think about the future consequences. Its actin can be mapped by following agent Funtion.

location & status

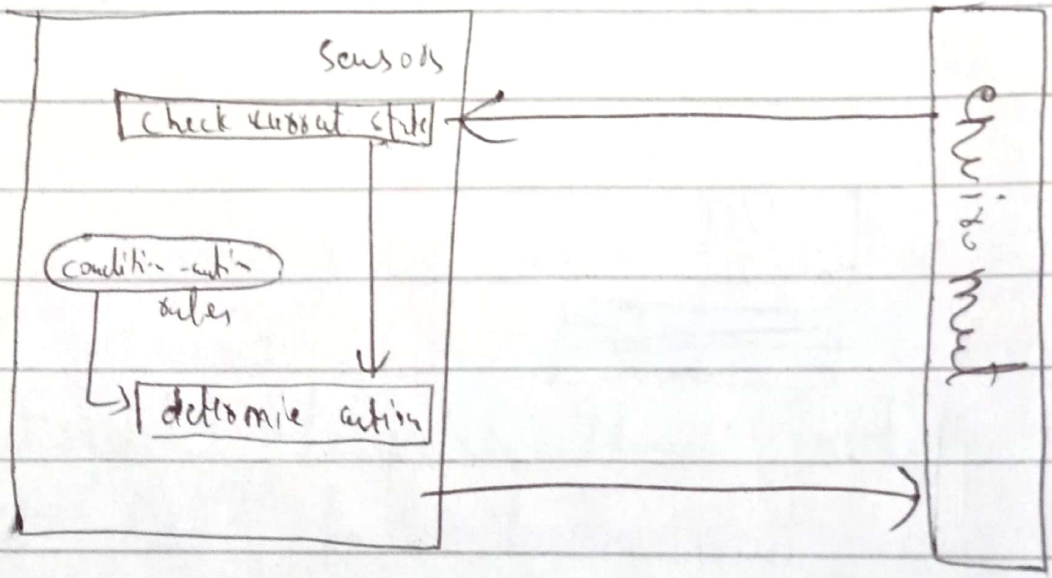
Function reflex-vacuum-agent (l, s) return action

Condition Action rules.

① if status = dirty then return suck
② else if location = A then return right
③ else if location = B then return left

①②③ = checking current percept / state

Schematic diagram for simple reflex agent



General action program :

A generalized action program for reflex based agents is as follows :-

```
Function Simple-reflex-agent (percept) return an action
    persistent rules (a list of condition action rule)
        state ← INTERPRET-INPUT (percep)
        rule ← RULE-MATCHING (rules, state)
        action ← rule . Action
    Return action.
```

Explanation :

-> The agent Funtion will take the current percept as input
-> rules is a global varible i.e persistant.
-> Interpret-Input Funtion will find the state of system.
-> Rest is self explanatory.

-> ~~They are only effecient in a~~
-> For reflex agents the environment is not Fully observable.
-> It works perfect in fully observable

→: Model - Based - Reflex - Agents :-

-> If the enviroment is partially observable then the simple reflex based agents can cause some problems such as it may go into an infinite loop. e.g if we take vacuum cleaner example it will always go b/w A and B if both are clean. We must have some sort of Knowledge stored in the agent about history that it already clean A and B.

-> In such kind of problem where the world is partially observable we require an agent that must "keep track of world it can't see right now" that is it must maintain some sort of Internal state that depends on the percept history

that can be used for taking decisions of current state.

Updating the internal state:

For updating the internal state of such agents we require two kinds of Knowledge:

-> How the world evolves independently of the agent:

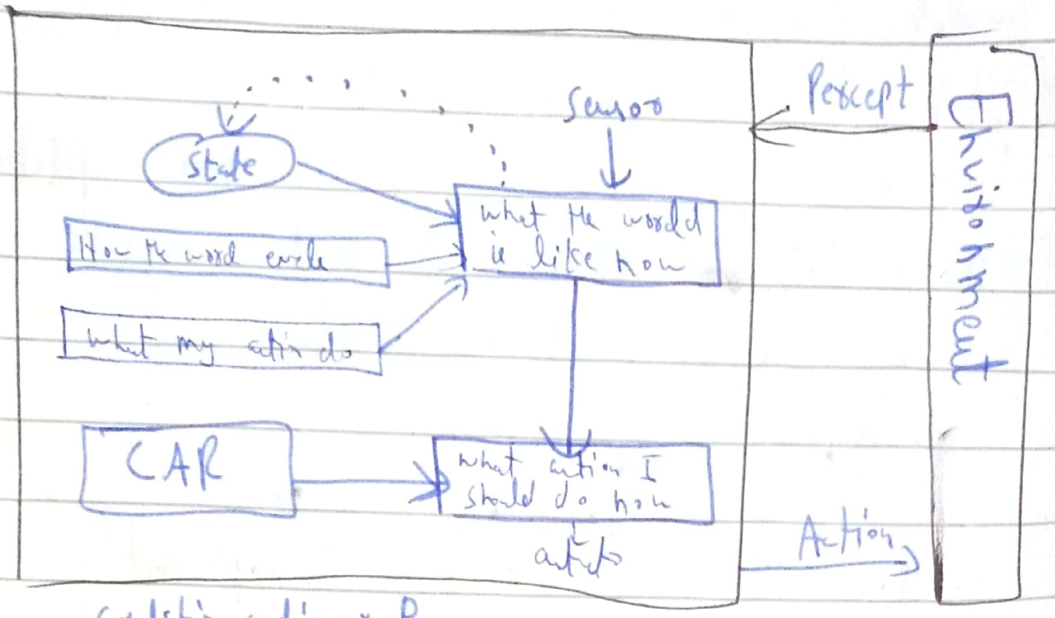
It means if agent do nothing then what happens in world.

-> How agent's own action affect the world:-

It means if agent do some action how it affect the world.

The both knowledge will be combined and stored as internal state of the agent.

-> This know of "How the world works" is called Model.



condition action rule

Explanation of diagram:

-> The sensor will get the percept of of the environment

-> It see how the partially observable world is like now.

-> It ~~→~~ اب اس کو پتہ چلے گا کہ اسکی کی history: ایک state اس کی perception میں ہے۔

-> Now it combine the knowledge of stored state, how the word evolves in such state and how agent own action affect the world, in such state these knowledge

Explanation of code :

→ The state in persistent represents the internal state of agent.

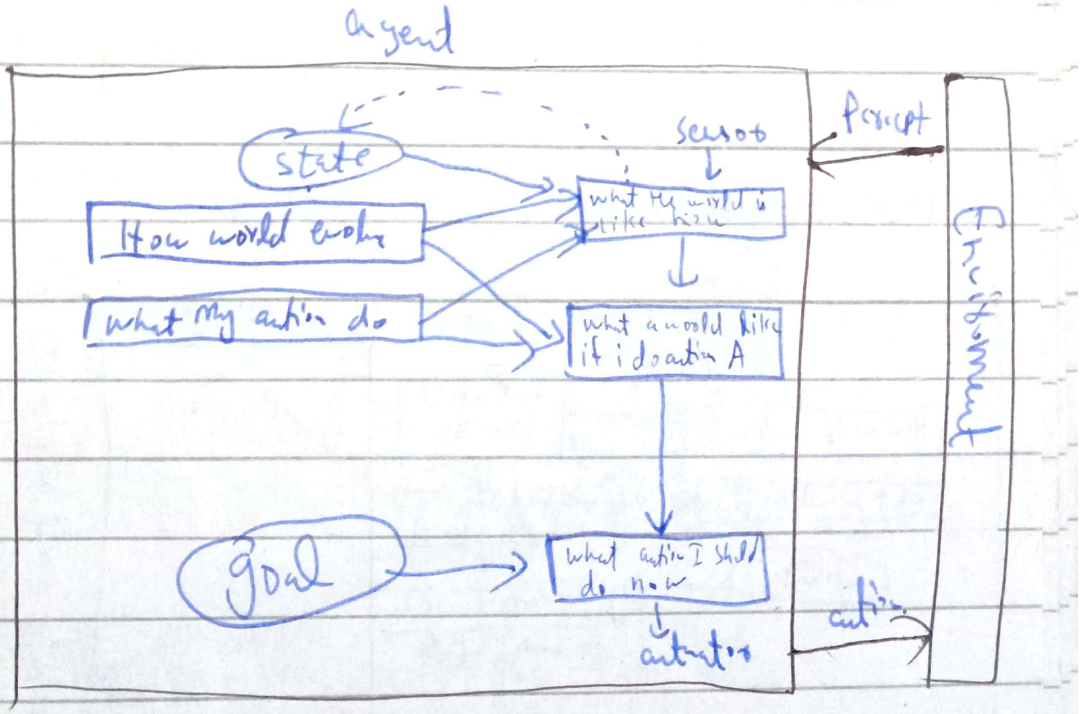
→ Model represent two things
  ↳ how ~~sys~~ world evolve
  ↳ how action affect world
→ rules are condition action rules.
→ action is the recent action of agent.
The rest is self explanatory.

:Goal-Based-Agents:-

-> Goal based agents are extension of model based agents. They also keep an internal state

-> But in these ~~models~~ the agent perform an action that leads towards a Goal.

-> Search and Planning are used to Find the action sequence that leads agent to achieve goal.



:: Utility - based - Agents ::

-> In utility based agents our primarily focus is utility not goal i.e if I go to a state whether it make me happy or not

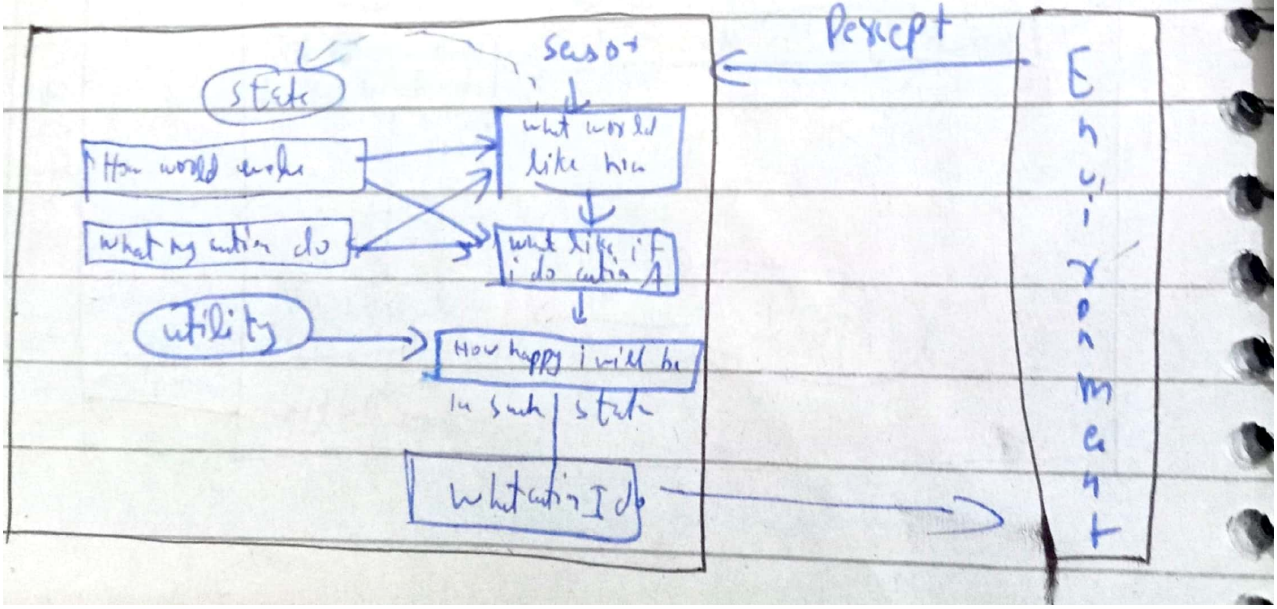
-> Utily based agents maximize the utility.

-> The utily is measured by a utility Fun

Example:

If I want to go VET ~~Here~~ there are many paths that lead me to achieve my goal but the shortest path will maximizes the utily.

-> These agents are more rational.



will be combine we have a new picture of "What the world is like now" and this picture of world also has some info about environment that is not observable.

-> Now this percept is pass to function and it chart condition action rules to determine action.

Action function:

function MODEL-BASED-REFLEX-AGENT (percept) returns action

Persistent state,
model,
rules,
action,

state ← UPDATE-state (model, ~~rules~~, perception, state)

rule ← RULE-MATCH (state, rules)

action ← rule.Action

return action.