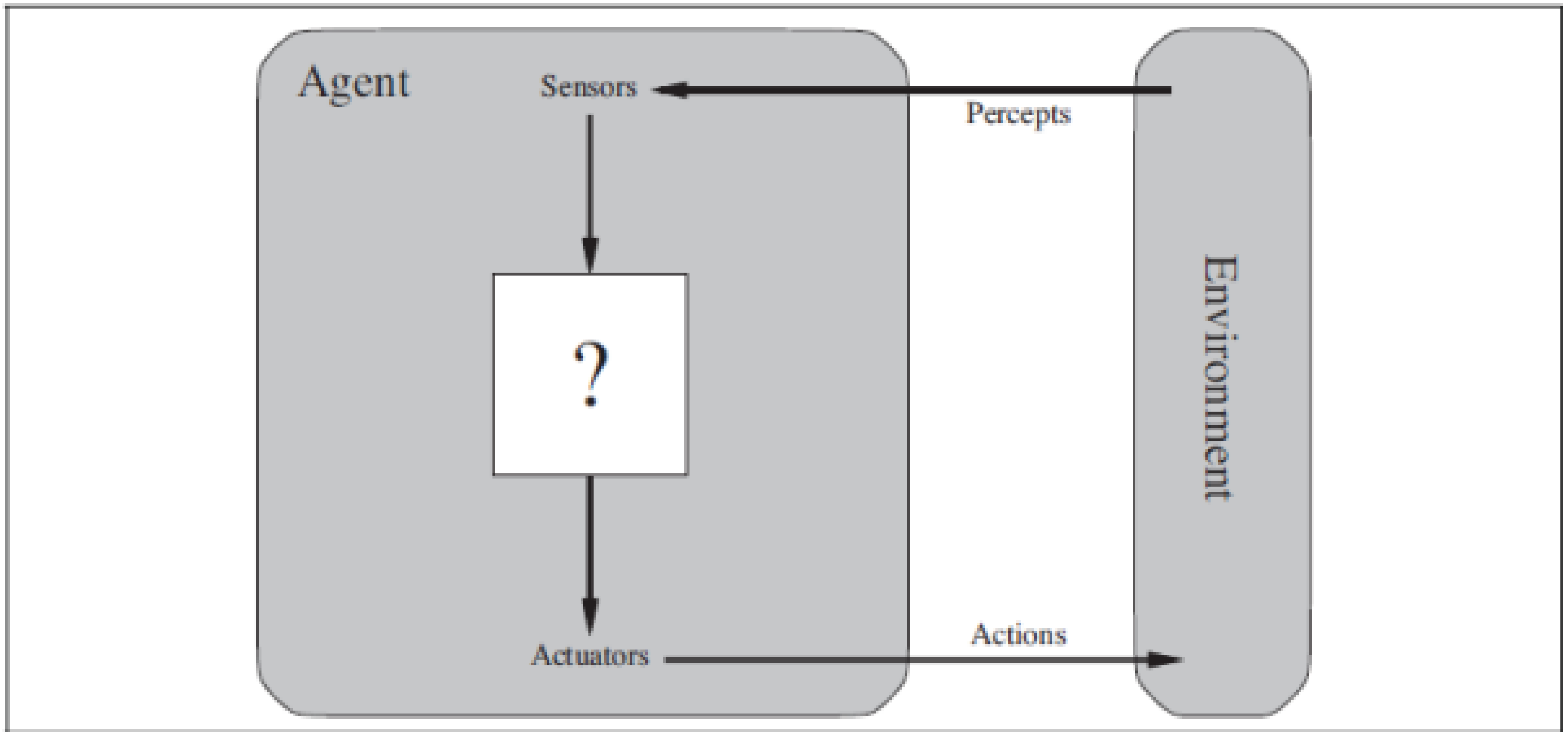
Reflex vacuum world

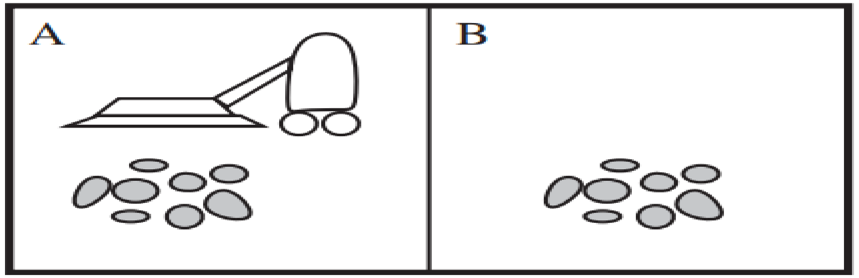
- An **agent** is anything that can be viewed as perceiving its **environment** through **sensors** and acting upon that environment through **actuators.**



Reflex Vacuum World

Step1:

Build the environment



Step2:

perceiving the environment through sensors to get location1 and status1

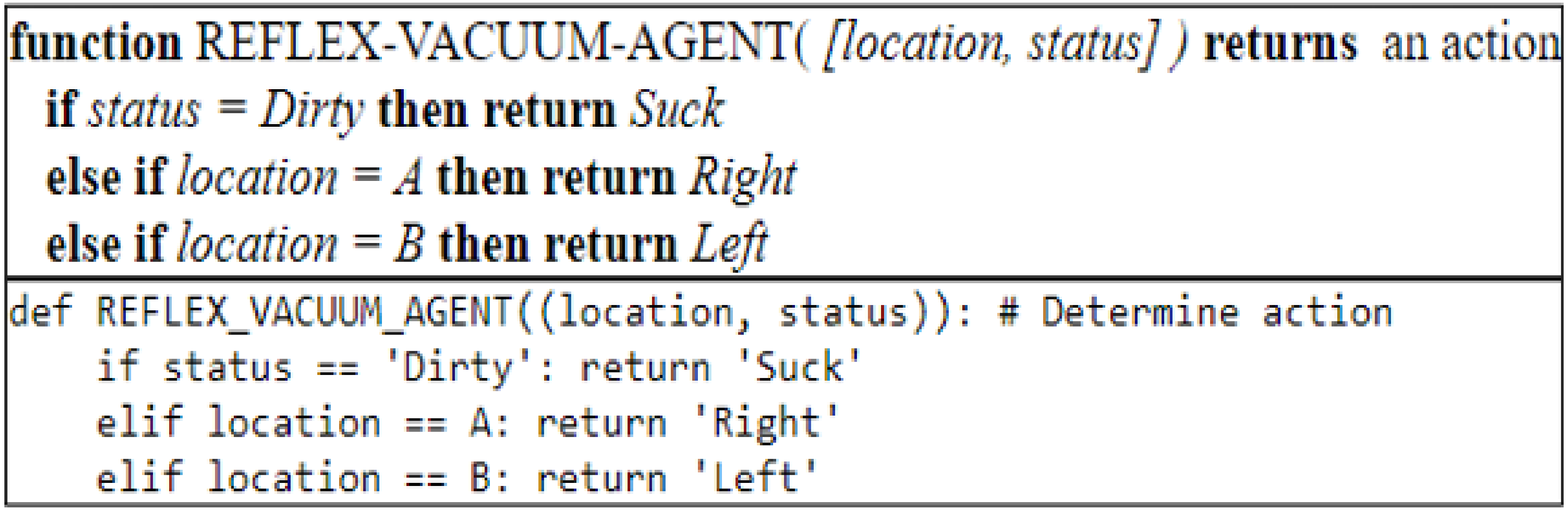
Step3:

Get the proper action for location1 and status1

Step4:

Get location2 and status2 after this action

The algorithm



```
function REFLEX-VACUUM-AGENT( [location, status] ) returns  an action
  if status = Dirty then return Suck
  else if location = A then return Right
  else if location = B then return Left
```

```
def REFLEX_VACUUM_AGENT((location, status)): # Determine action
    if status == 'Dirty': return 'Suck'
    elif location == A: return 'Right'
    elif location == B: return 'Left'
```

The code

```
A='A'
B='B'
enviroment={A:'dirty',B:'dirty','current':A}

def reflex_vacuum_agent(loc,status):
    if status=='dirty':
        return'suck'
    elif loc==A:
        return'right'      # to determine the actions
    elif loc==B:
        return'left'

def sensors():
    loc=enviroment['current']
    status=enviroment[loc]
    return(loc,status)
```

The code

```
def acctuators(action):
    loc =enviroment['current']
    if action=='suck':
        enviroment[loc]='clean'
    elif action=='right':
        enviroment['current']=B
    elif action=='left':
        enviroment['current']=A
def run(n):
    print('( loc1,status1)\taction\t(loc2,status2)')
    for i in range(n):
        (loc1,stat1)=sensors()
        action=reflex_vacuum_agent(loc1,stat1)
        acctuators(action)
        (loc2,stat2)=sensors()
        print((loc1,stat1),'\t',action,'\t',(loc2,stat2))
run(5)
```

The Result

(loc1,status1)	action	(loc2,status2)
('A', 'dirty’)	suck	('A', 'clean')
('A', 'clean’)	right	('B', 'dirty')
('B', 'dirty')	suck	('B', 'clean')
('B', 'clean')	left	('A', 'clean')
('A', 'clean')	right	('B', 'clean')