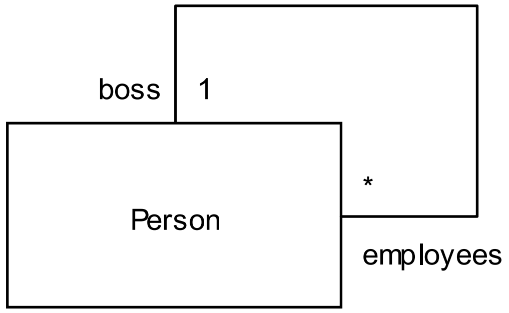
UML Puzzle – Part I

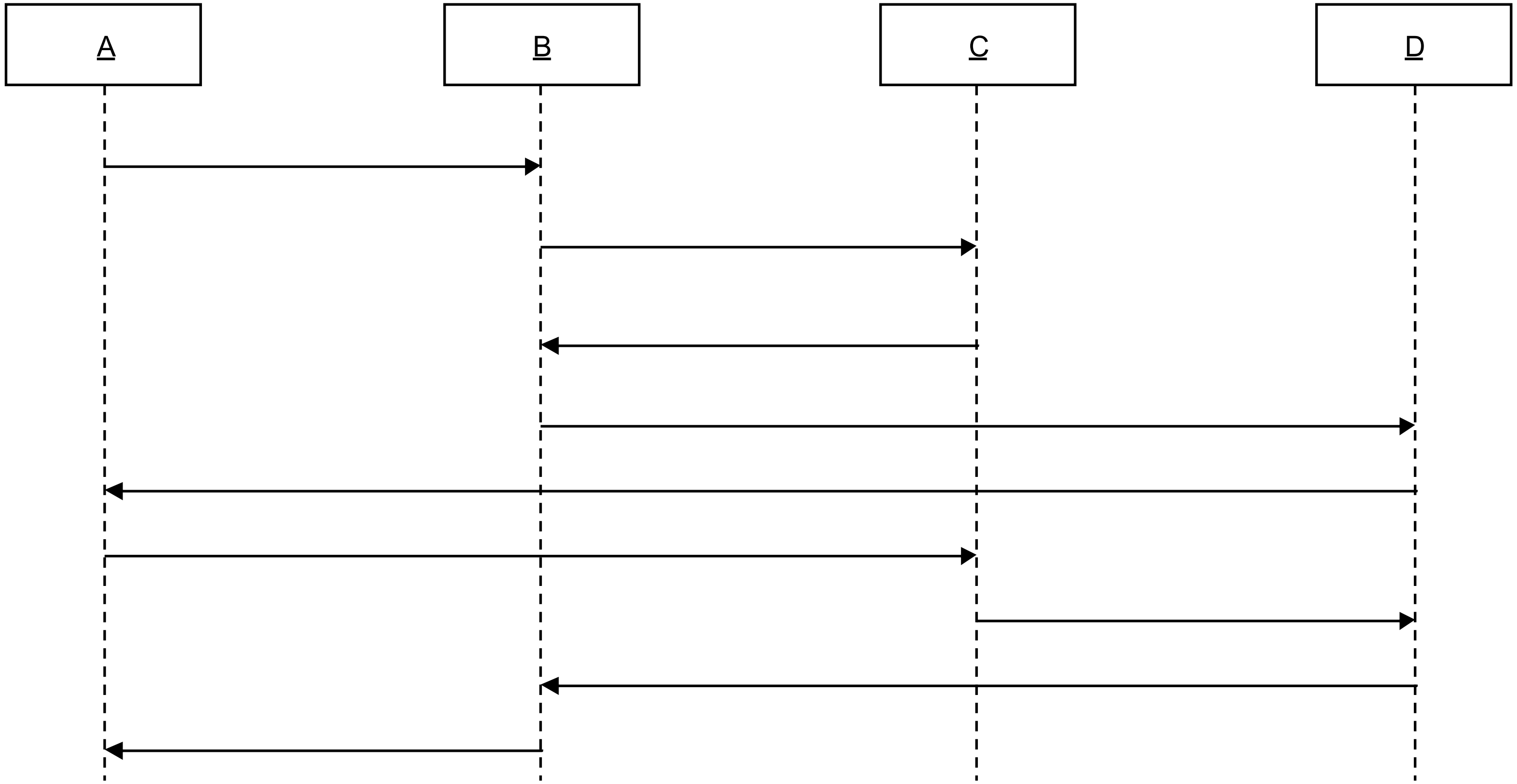
1



Which of the following sentences can apply to the class model?

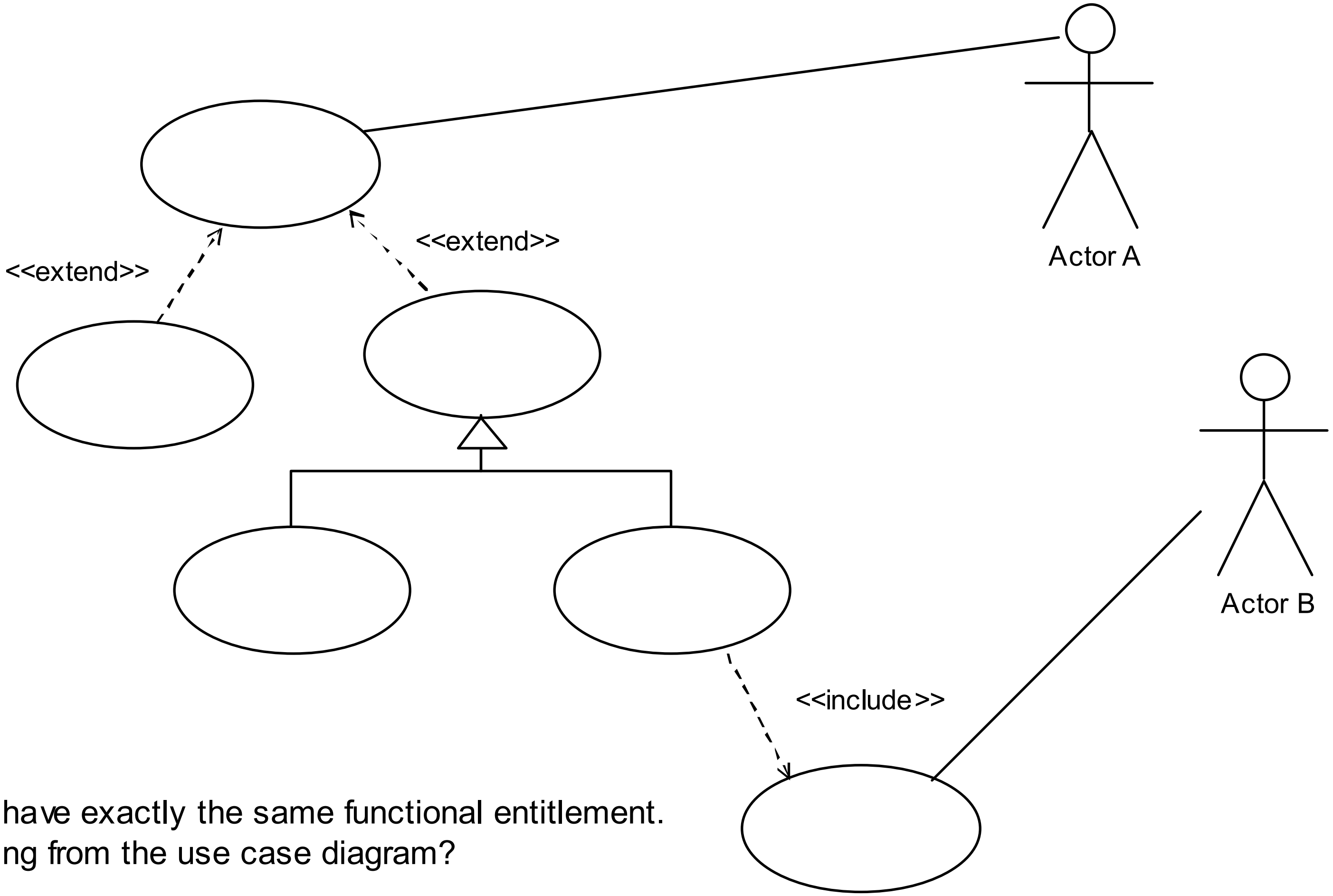
a) “Be your own boss”
b) “Neither a leader nor a follower be”
c) “Sitting at the top of the tree”
d) “Too many chiefs, not enough Indians”

2



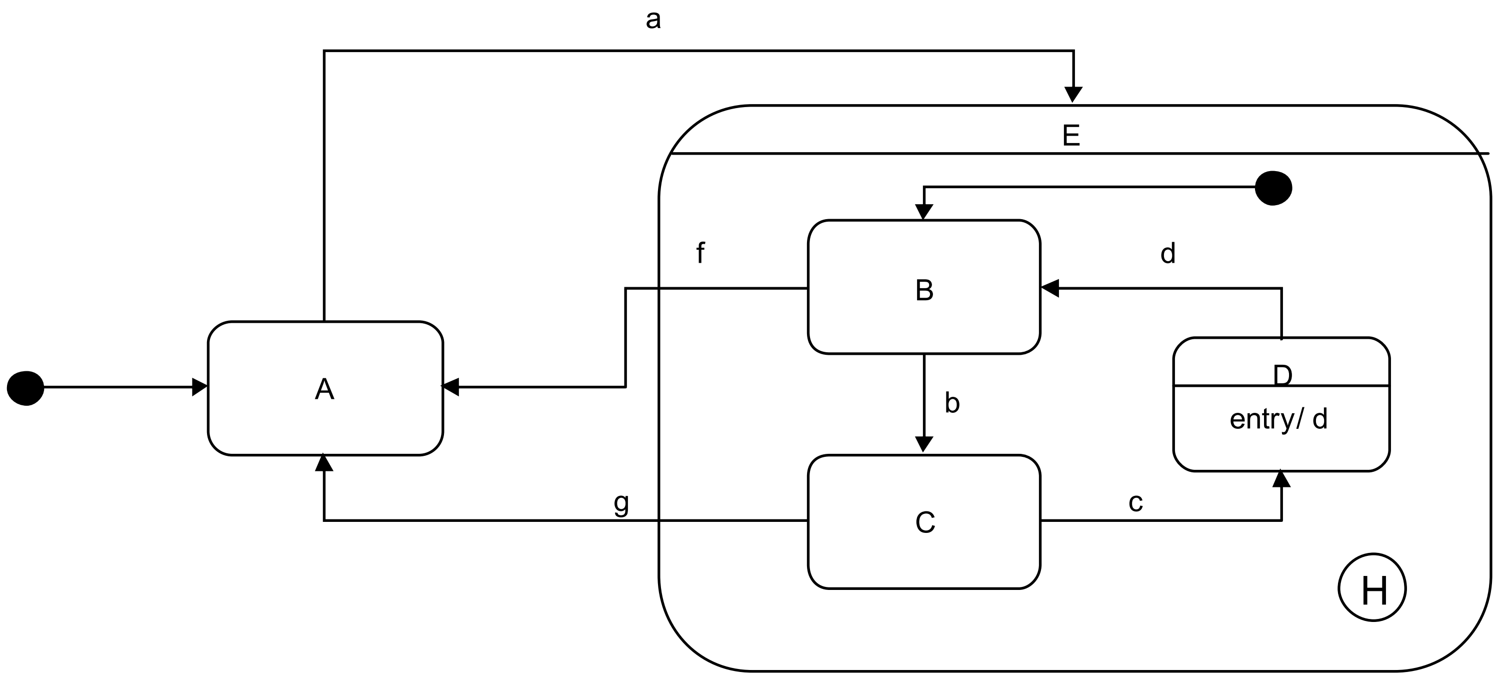
How many *bidirectional* relationships exist in this model?

3



Actor A and Actor B both have exactly the same functional entitlement. What relationship is missing from the use case diagram?

4

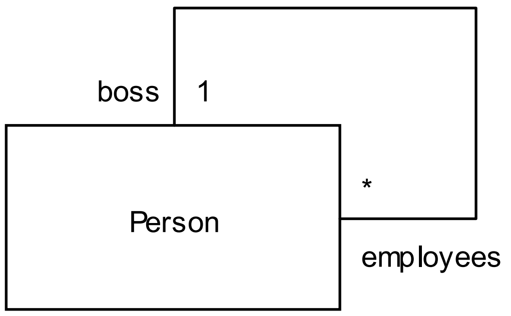


What state(s) is the object in after this sequence of events?

● -> a -> b -> c -> b -> g -> a

Solutions

1

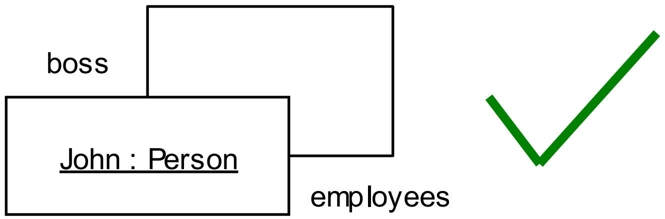


Which of the following sentences can apply to the class model?

a) “Be your own boss”
b) “Neither a leader nor a follower be”
c) “Sitting at the top of the tree”
d) “Too many chiefs, not enough Indians”

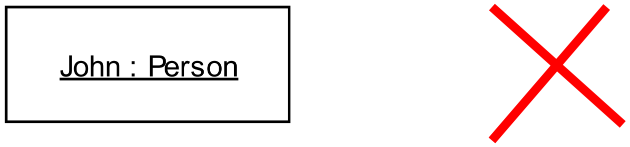
a, c and d can be applied to the model.

a)



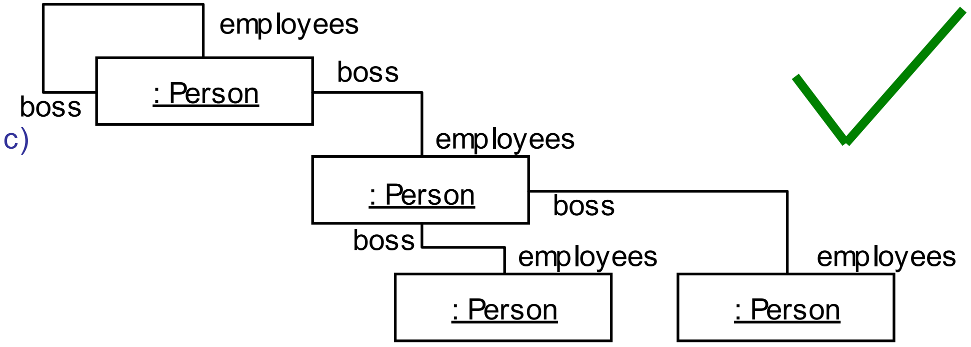
John is his own boss. This is a valid instance of the model.

b)



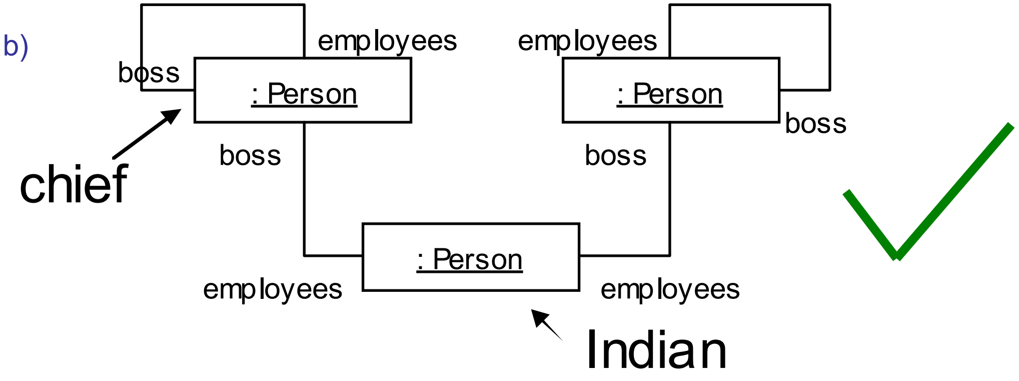
John has neither a boss nor employees, but the model says that every Person must have exactly one boss. This is not a valid instance of the model.

c)



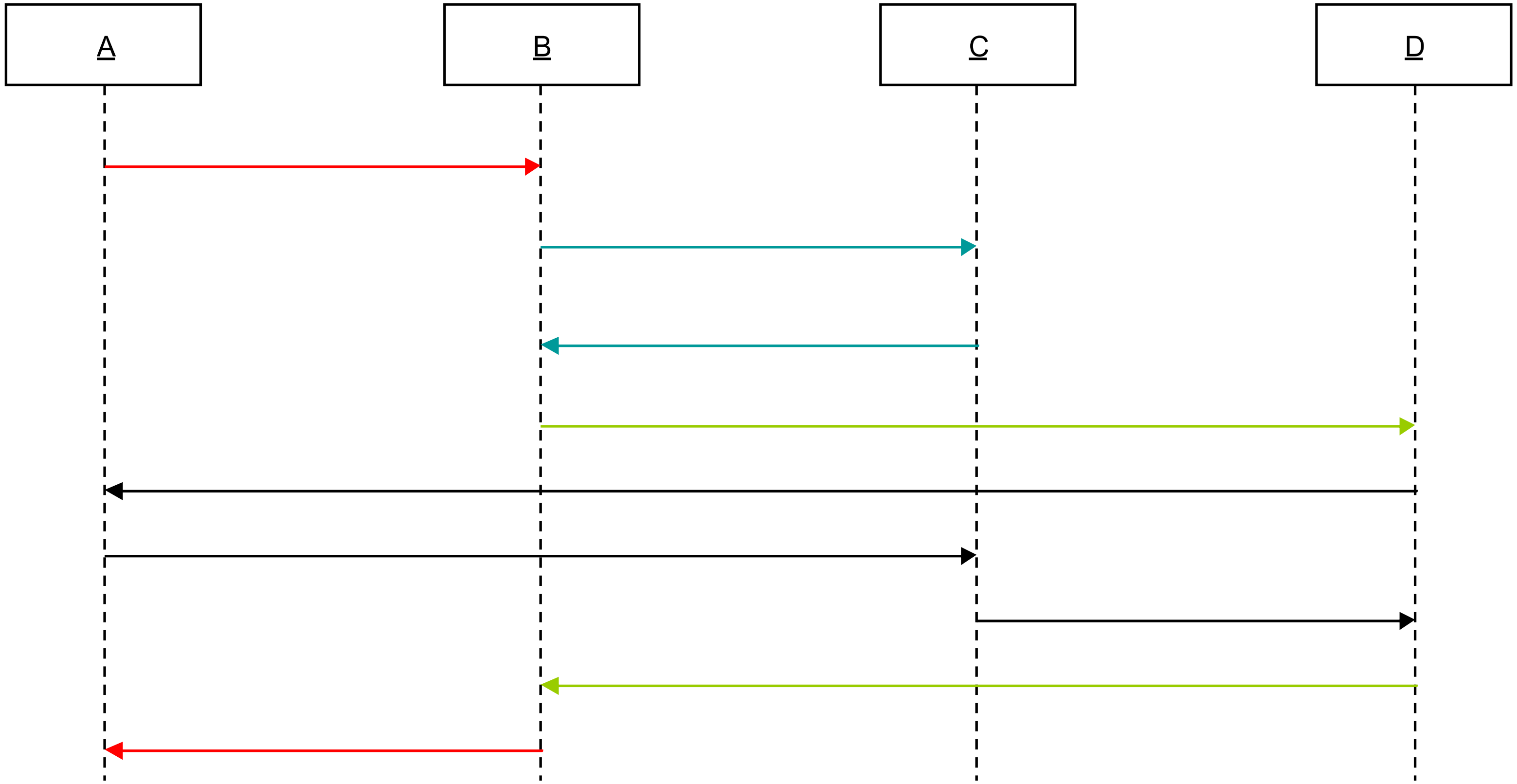
Since a person can be a boss AND an employee (effectively a parent *and* a child), and since a Person need not have employees (i.e., it can be a leaf *or* a branch) we can build trees using this model

b)



It is indeed possible to have more bosses than non-bosses

2

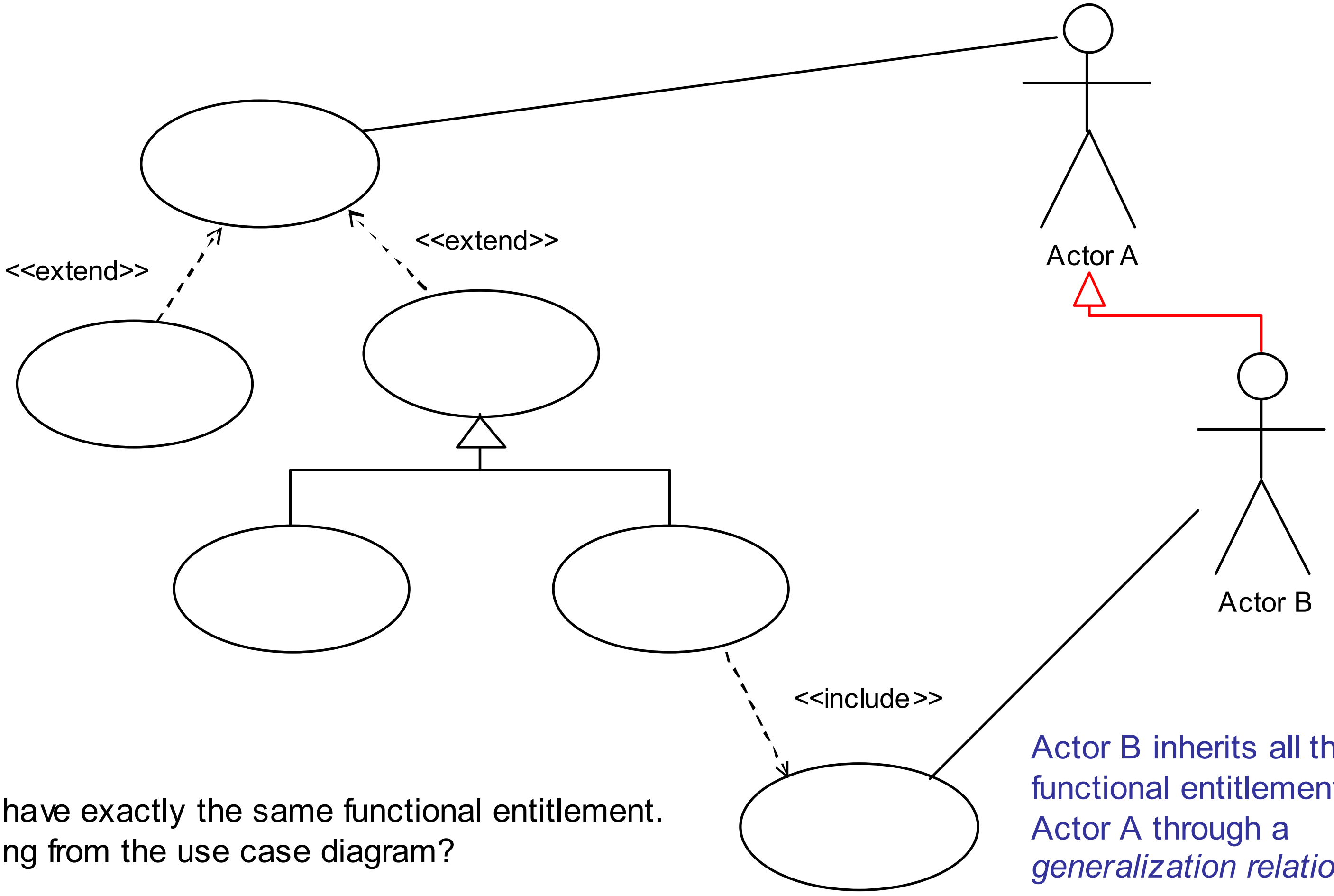


How many *bidirectional* relationships exist in this model?

3

A <-> B
B <-> C
B <-> D

3

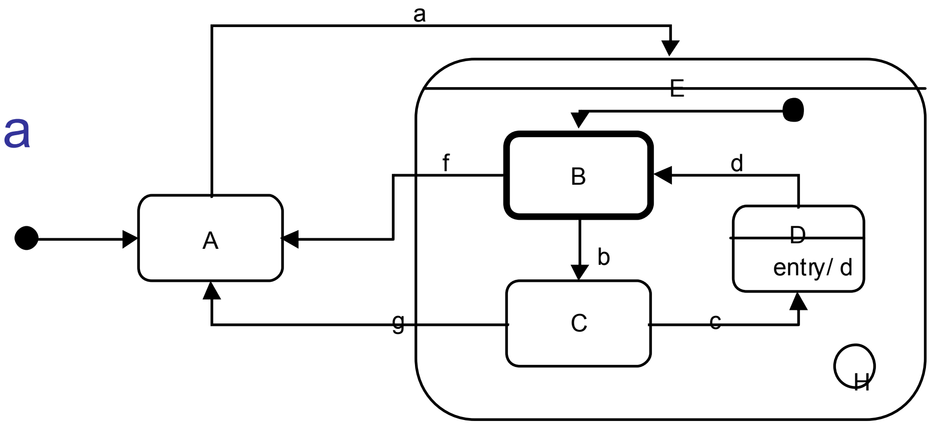


Actor A and Actor B both have exactly the same functional entitlement. What relationship is missing from the use case diagram?

Actor B inherits all the functional entitlements of Actor A through a *generalization relationship*

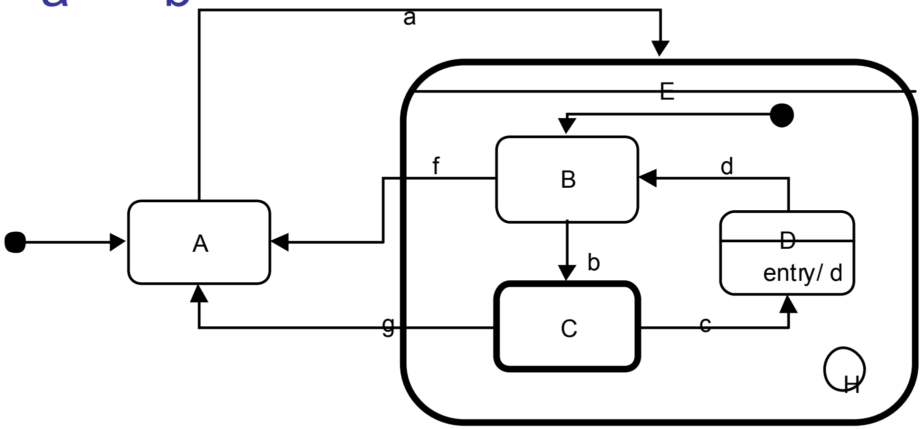
4

● -> a

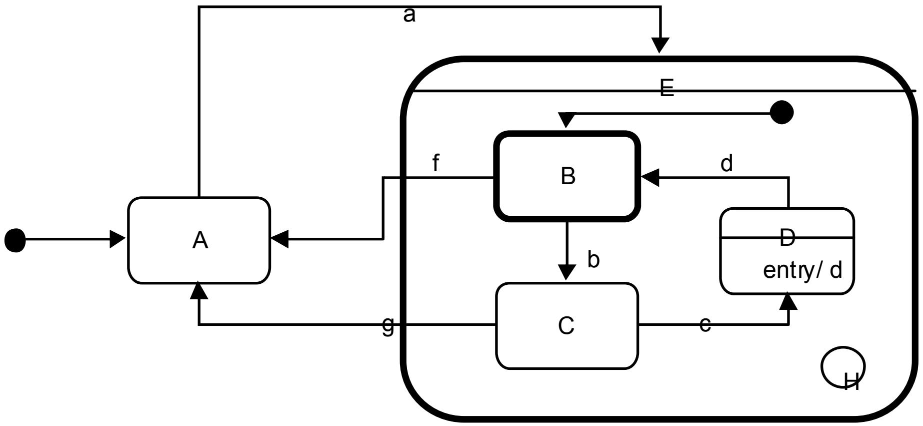


On entering super-state E for the first time, the object goes into sub-state B by default

● -> a -> b



● -> a -> b -> c



On entering sub-state D, the action d is automatically triggered, taking the object back to sub-state B

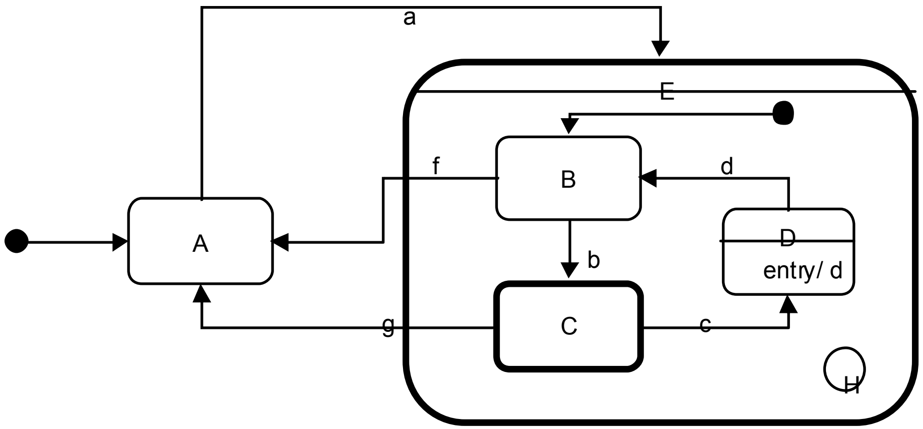
What state(s) is the object in after this sequence of events?

● -> a -> b -> c -> b -> g -> a

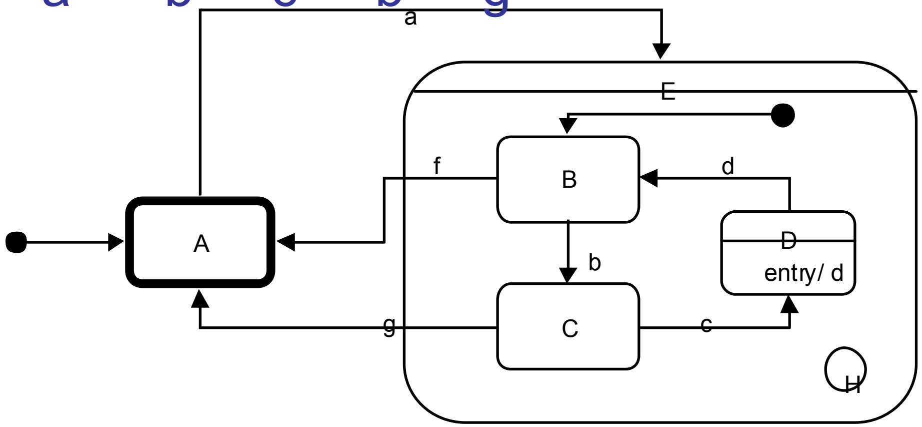
parlez|uml

4

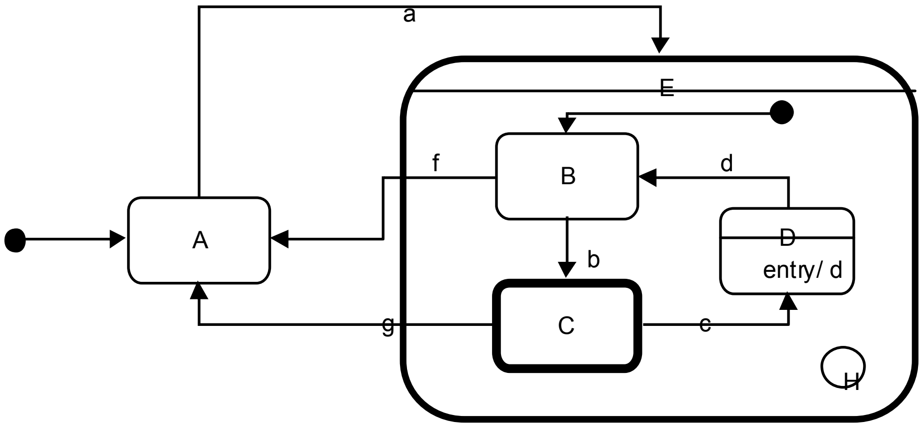
● -> a -> b -> c -> b



● -> a -> b -> c -> b -> g



● -> a -> b -> c -> b -> g -> a



On entering super-state E for the second time, the object is returned to that last sub-state it was in, which is C (because the “H” stands for “history state)

The object ends in the super-state E and the sub-state C