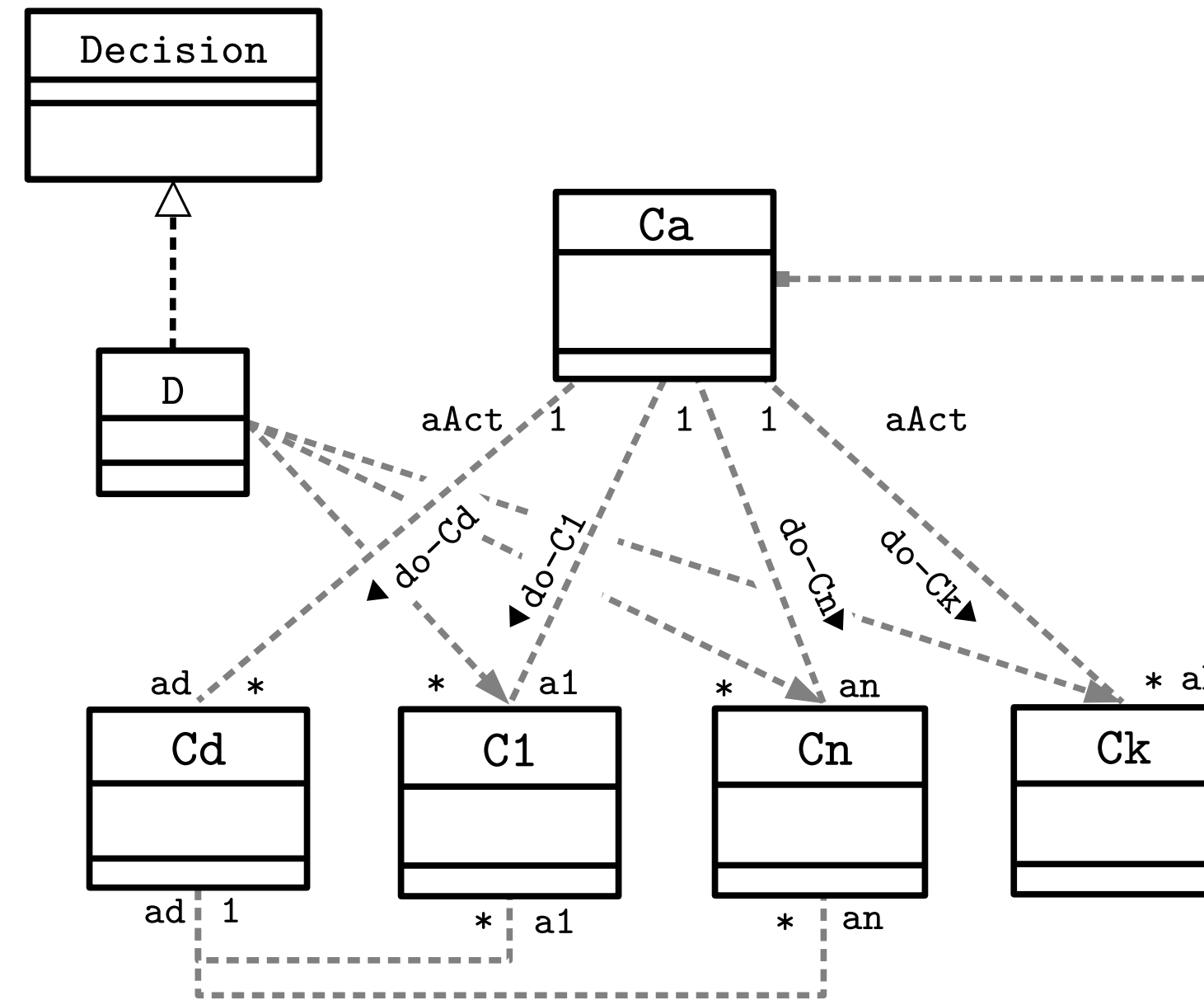
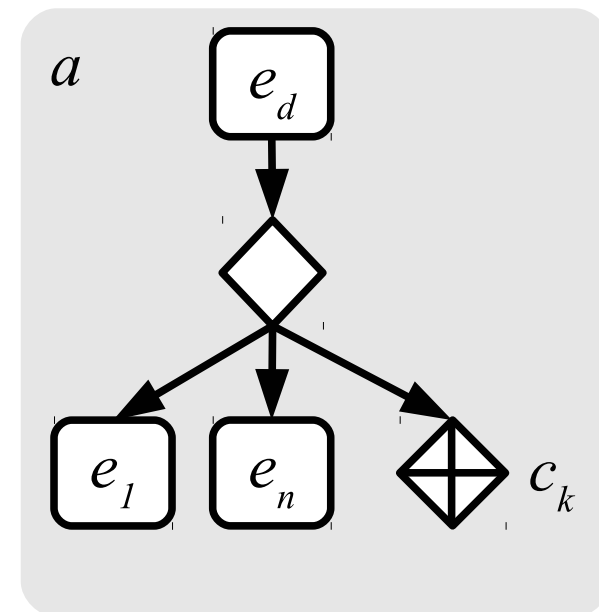


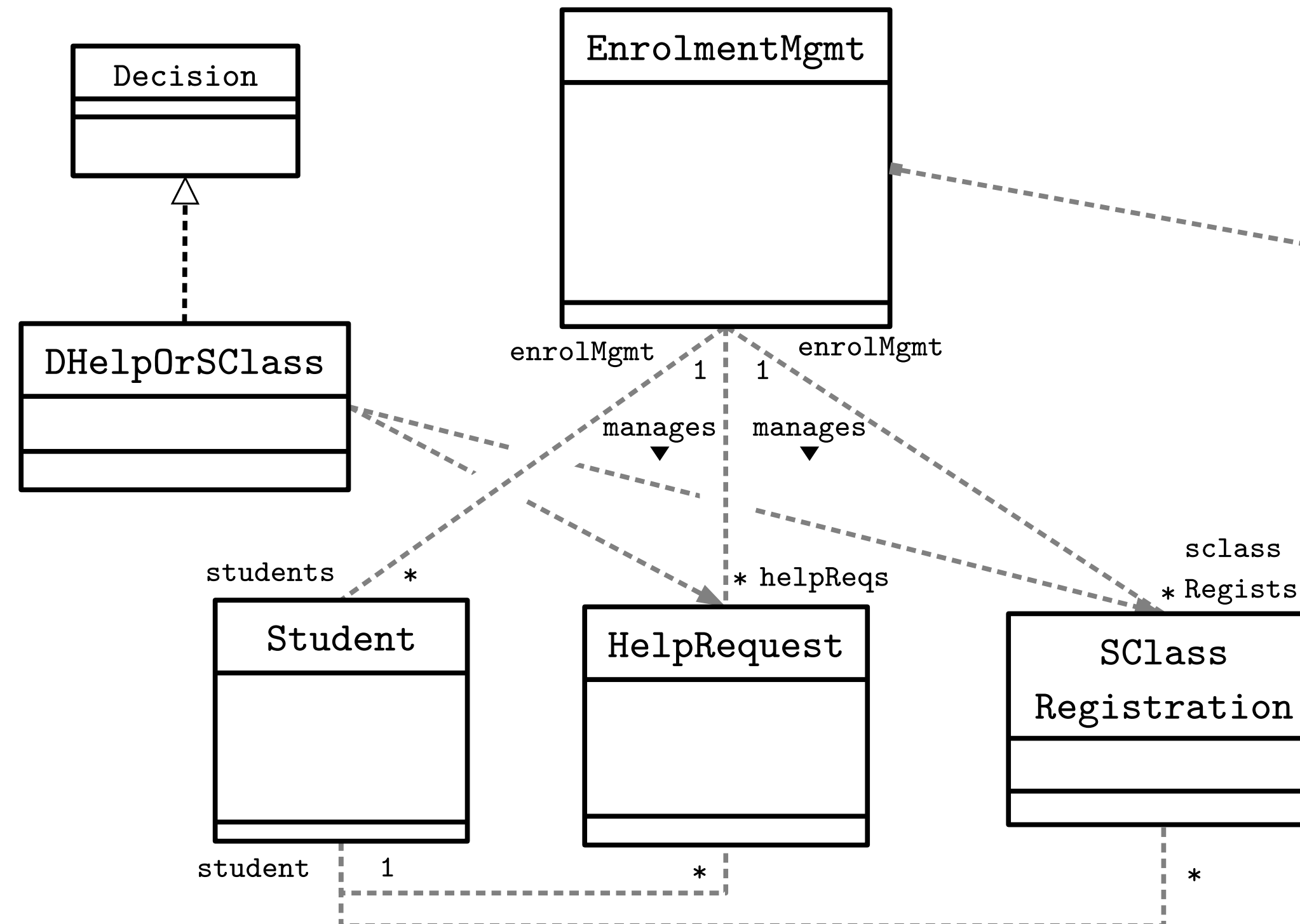
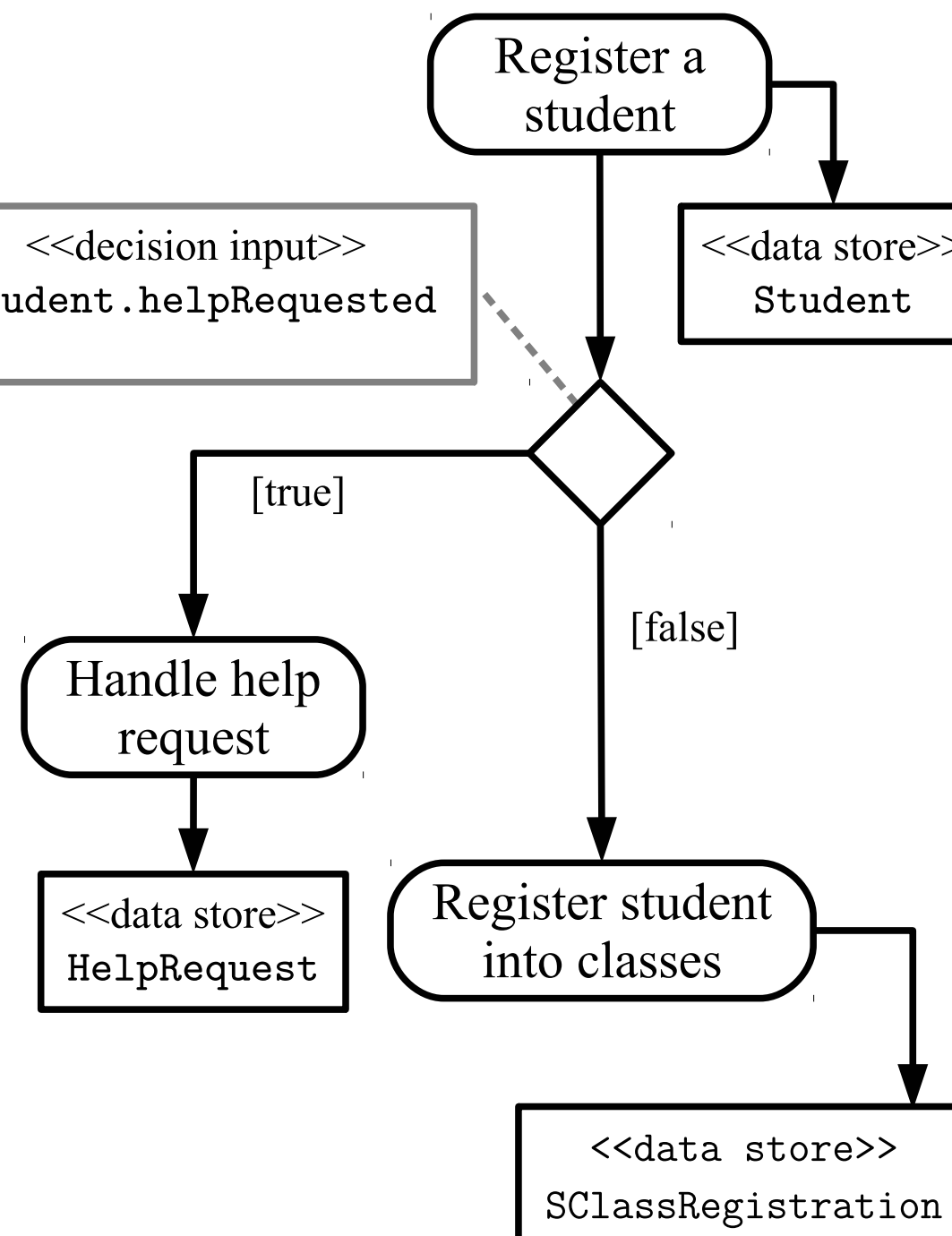
decisional pattern  
(if  $c_k$  is present then  $n \geq 1$   
else  $n \geq 2$ )



```

AGraph { nodes=[
  ANode { refCls=Cd,serviceCls=DataController
    outClses=[D]
    actSeq=[
      MAct{actName=newObject,pstStates=[Created]} ]
    init=true}
  ANode { refCls=D, nodeType=Decision
    outClses=[C1,Cn,Ck]}
  ANode { refCls=C1,serviceCls=DataController
    actSeq=[
      MAct{actName=newObject,pstStates=[NewObject]},
      MAct{actName=setDataFieldValues,fieldNames=["ad"]
        pstStates=[Created]} ]}
  ANode { refCls=Cn,serviceCls=DataController
    actSeq=[
      MAct{actName=newObject,pstStates=[NewObject]},
      MAct{actName=setDataFieldValues,fieldNames=["ad"]
        pstStates=[Created]} ]}
  ANode { refCls=Ck, nodeType=k, outClses=k_out}
]}
  
```

## Enrolment Management



```

AGraph { nodes=[
  ANode { refCls=Student,serviceCls=DataController
    outClses=[DHelpOrSClass]
    actSeq=[
      MAct{actName=newObject,pstStates=[Created]} ]
    init=true}
  ANode { refCls=DHHelpOrSClass, nodeType=Decision
    outClses=[HelpRequest, SClassRegistration]}
  ANode { refCls=HelpRequest,serviceCls=DataController
    actSeq=[
      MAct{actName=newObject,pstStates=[NewObject]},
      MAct{actName=setDataFieldValues
        fieldNames=["student"]
        pstStates=[Created]} ]}
  ANode {
    refCls=SClassRegistration,serviceCls=DataController
    actSeq=[
      MAct{actName=newObject,pstStates=[NewObject]},
      MAct{actName=setDataFieldValues
        fieldNames=["student"]
        pstStates=[Created]} ]} ]}
  
```