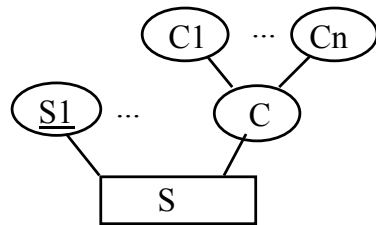


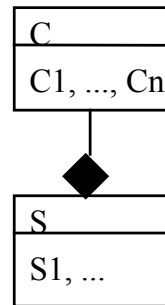
# E/A vers relationnel

MODELE EA



Ex :  $S = \text{Personne}$ ;  
 $C = \text{Adresse}(\text{Numéro}, \text{Rue}, \text{Code})$

UML



MODELE RELATIONNEL

## Solution 1

$S(\underline{S1}, S2, \dots, C1, \dots, Cn)$

## Solution 2

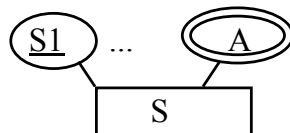
$S(\underline{S1}, S2, \dots)$

$C(\underline{S1}, C1, \dots, Cn)$

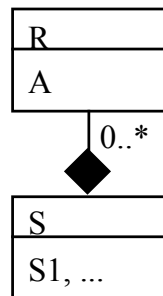
$C.S1 \text{ ref } S.S1$

on delete cascade  
 on update cascade

**1- Attribut composé :** Les attributs simples d'un attribut composé peuvent être placés dans la table S du type S (la composition est alors perdue) ou dans une table séparée C liée à S par une contrainte de référence.



Ex:  $S = \text{Personne}$ ;  $A = \text{Prénom-Enfant}$



$S(\underline{S1}, S2, \dots)$

$R(\underline{S1}, \underline{A})$

$R.S1 \text{ ref } S.S1$

on delete cascade  
 on update cascade

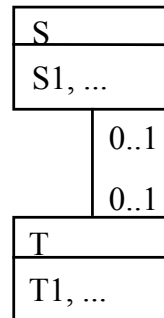
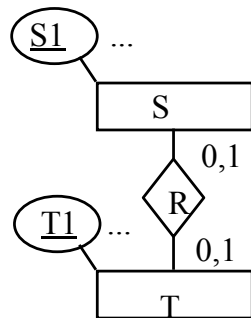
**2- Attribut multivalué :** Un attribut multivalué A est obligatoirement placé dans une autre table R liée à la table S du type d'entité (la clé primaire de R est la juxtaposition de A et d'une clé étrangère faisant référence à la clé primaire de S).

# E/A vers relationnel

MODELE EA

UML

MODELE RELATIONNEL



$S(\underline{S1}, \dots, T1)$   
 $T(\underline{T1}, \dots)$   
 $S.T1$  unique ref  $T.T1$   
 on delete set null  
 on update cascade

Autre solution : solution symétrique

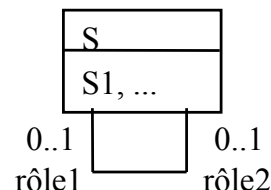
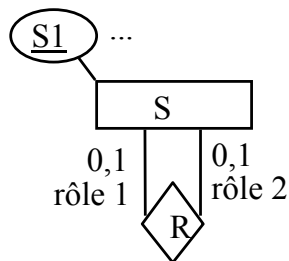
*Ex:  $S=Appartement$ ;  $T=Garage$*

## 3- Association binaire 1-1

### Solution spécifique

Si le type S qui référence a la cardinalité min de 1 il faut spécifier :  
 on delete { set default|cascade}.

Il n'est pas possible d'exprimer simplement une card min du type référencé T égale à 1. Le mieux est de passer par des contraintes dynamiques (triggers).



$S(\underline{S1} \dots, S1\text{-rôle2})$   
 $S1\text{-rôle2}$  unique ref  $S.S1$   
 on delete set null  
 on update cascade

Autre solution : solution symétrique

*Ex :  $S=Personne$ ; rôle1=est-tuteur;  
 rôle2=a-pour-tuteur*

Cas réflexif : On place dans la table S une clé étrangère qui représente le deuxième rôle que joue l'entité S dans l'association R.

Si le rôle2 a une card min de 1 il faut spécifier :

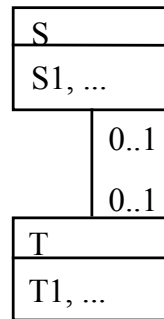
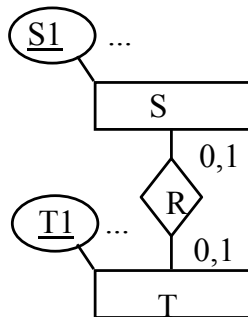
on delete { set default|cascade}.

# E/A vers relationnel

MODELE EA

UML

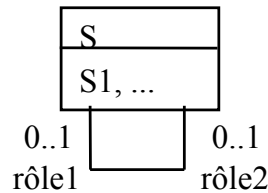
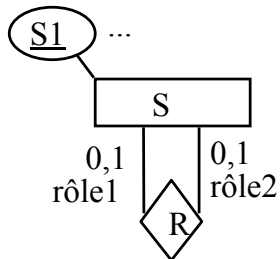
MODELE RELATIONNEL



$S(\underline{S1}, \dots)$   
 $T(\underline{T1}, \dots)$   
 $R(\underline{S1}, T1)$   
 $R.S1$  Primary Key ref  $S.S1$   
 on delete cascade  
 on update cascade  
 $R.T1$  not null unique ref  $T.T1$   
 on delete cascade  
 on update cascade

Ex:  $S=Appartement$ ;  $T=Garage$

Autre solution : Solution symétrique



$S(\underline{S1}, \dots)$   
 $R(\underline{S1-rôle1}, S1-rôle2)$   
 $S1-rôle1$  Primary Key ref  $S.S1$   
 on delete cascade  
 on update cascade  
 $S1-rôle2$  not null unique ref  $S.S1$   
 on delete cascade  
 on update cascade

Ex :  $S=Personne$ ;  $rôle1=est-tuteur$ ;  
 $rôle2=a-pour-tuteur$

## 3- Association binaire 1-1

### Solution universelle

Une seule des clés étrangères est clé primaire dans R (prendre celle qui référence la table dont la cardinalité minimale est 1 dans l'association). L'autre est clé candidate. Sa présence est indispensable d'où l'option not null qui lui est associée.

Si S a une card min =1 on peut choisir l'option on delete set default pour R.T1.

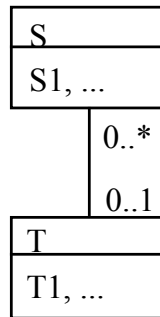
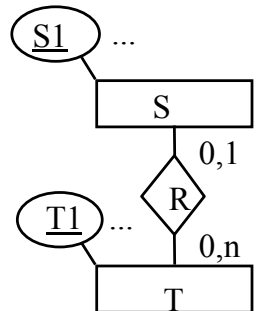
Cas réflexif : La table R qui représente l'association contient deux clés étrangères correspondant à chacun des rôles que peut jouer une occurrence de l'entité S dans R.

# E/A vers relationnel

MODELE EA

UML

MODELE RELATIONNEL



$S(\underline{S1}, \dots, T1)$   
 $T(\underline{T1}, \dots)$   
 $S.T1 \text{ ref } T.T1$   
 on delete set null  
 on update cascade

*Ex : S=Employé; T=Service*

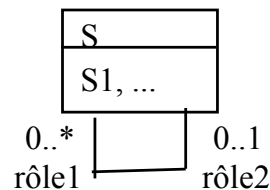
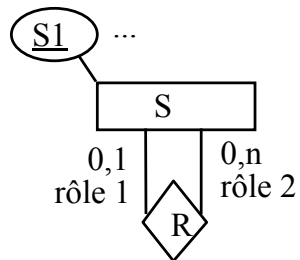
## 4- Association binaire 1-n

### Solution spécifique

S étant le type dont la cardinalité maximale est un, on installe dans la table S une clé étrangère vers la table T. Si la card min de S est 1, il faut utiliser l'option :

on delete { set default|cascade }.

On ne peut pas exprimer directement une card min de T égale à 1. Le mieux est de passer par des contraintes dynamiques (triggers).



$S(\underline{S1}, \dots, S1\text{-rôle2})$   
 $S1\text{-rôle2 ref } S.S1$   
 on delete set null  
 on update cascade

*Ex : R=direction; rôle1=subalterne; rôle2=directeur*

Cas réflexif : La clé étrangère à placer dans la table S correspond au rôle dont la cardinalité maximale est n (ici rôle2). Si le rôle1 a une card min =1, il faut utiliser l'option :

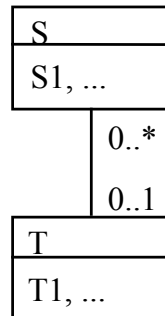
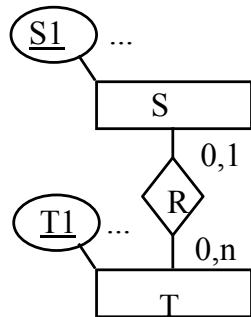
on delete { set default|cascade }.

# E/A vers relationnel

MODELE EA

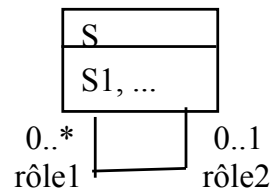
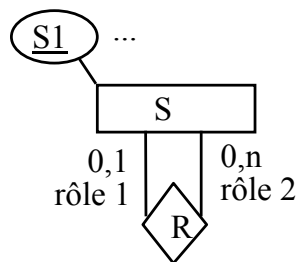
UML

MODELE RELATIONNEL



S(S1, ...)  
 T(T1, ...)  
 R(S1, T1)  
 R.S1 Primary Key ref S.S1  
 on delete cascade  
 on update cascade  
 R.T1 not null ref T.T1  
 on delete cascade  
 on update cascade

Ex:  $S=Pièce$ ;  $T=Commande$



S(S1, ...)  
 R(S1-rôle1, S1-rôle2)  
 S1-rôle1 Primary Key ref S.S1  
 on delete cascade  
 on update cascade  
 S1-rôle2 not null ref S.S1  
 on delete cascade  
 on update cascade

Ex :  $R=Direction$ ;  $rôle1=subalterne$ ;  $rôle2=directeur$

## 4- Association binaire 1-n

### Solution universelle

La clé primaire de R est la clé étrangère pointant vers la table associée à l'entité dont la card max=1.

Dans le cas où la card min de S est 1 (resp. T) on peut aussi utiliser pour R.T1 (resp. R.S1) l'option on delete set default.

On ne peut pas exprimer directement le cas où les deux card min sont égales à 1.

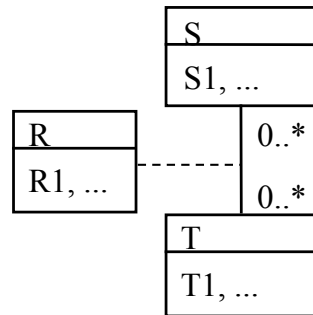
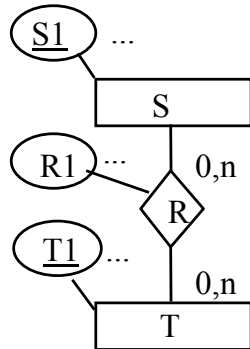
Cas réflexif : La table R qui représente l'association R contient deux clés étrangères correspondant à chacun des rôles. La clé primaire est la clé étrangère qui est associée au rôle dont la card max=1.

# E/A vers relationnel

MODELE EA

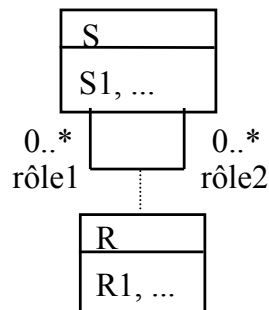
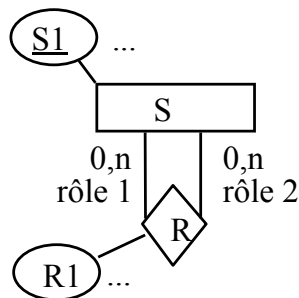
UML

MODELE RELATIONNEL



$S(\underline{S1}, \dots)$   
 $T(\underline{T1}, \dots)$   
 $R(\underline{S1}, \underline{T1}, R1, \dots)$   
 $R.S1 \text{ ref } S.S1$   
 on delete cascade  
 on update cascade  
 $R.T1 \text{ ref } T.T1$   
 on delete cascade  
 on update cascade

Ex:  $S=Pi\grave{e}ce$ ;  $T=Commande$



$S(\underline{S1}, \dots)$   
 $R(\underline{S1-r\^ole1}, \underline{S1-r\^ole2}, R1, \dots)$   
 $S1-r\^ole1 \text{ ref } S.S1$   
 on delete cascade  
 on update cascade  
 $S1-r\^ole2 \text{ ref } S.S1$   
 on delete cascade  
 on update cascade

Ex :  $R=Nomenclature$ ;  $r\^ole1=compose$ ;  $r\^ole2=assemble$

## 5- Association n-n

### Solution universelle

La clé primaire de la table R est la concaténation des clés primaires de S et T.

Les attributs éventuels de R sont insérés dans la table R.

Le cas où la card min de S (resp. T) est égale à 1 peut être traité avec l'option on delete set default pour R.T1 (resp. R.S1). On ne peut pas traiter directement le cas où les deux card min sont égales à 1. Le mieux est de passer par des contraintes dynamiques (triggers).

Cas réflexif : La table R contient deux clés étrangères correspondant à chacun des rôles que peut jouer une occurrence de l'entité S dans R. La clé primaire est la concaténation de ces deux clés étrangères.

# E/A vers relationnel

---

## 6 – Type d'association n-aire

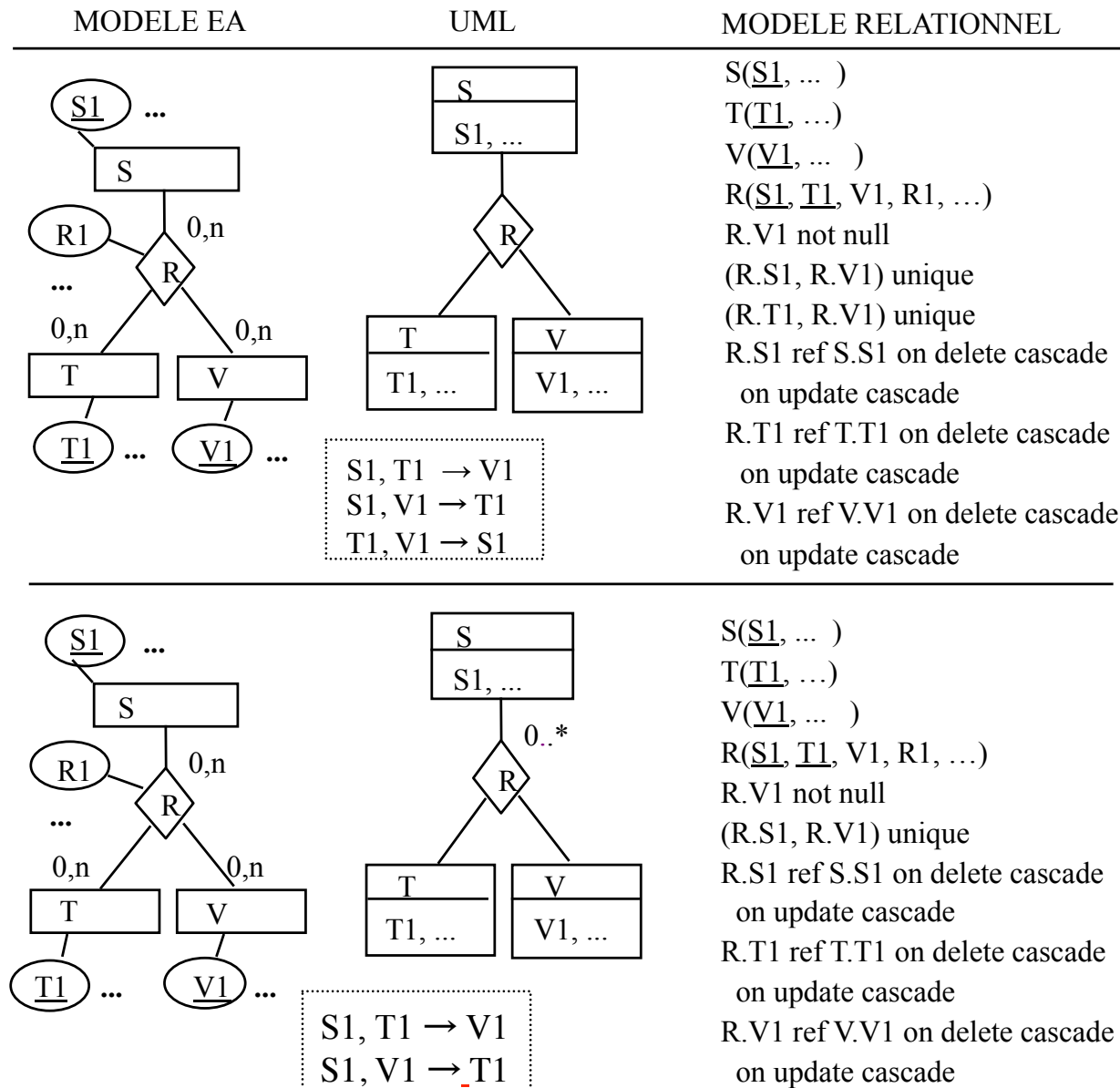
Pour chaque type d'association de degré  $n > 2$ , créer une nouvelle relation R.

La clé primaire de R est la juxtaposition de clés étrangères relatives à certaines des clés primaires des relations issues des types d'entités participants. Cette juxtaposition est déterminée en analysant les dépendances fonctionnelles qui peuvent exister entre ces clés.

Par exemple dans le cas où  $d=3$  il y a quatre situations possibles.

Nous donnons ci-après un exemple illustrant chacun de ces cas.

# E/A vers relationnel



**6a-** S: Technicien, T: Projet, V: Cahier. Les notes d'un technicien pour un projet sont consignées dans le même cahier. Les notes d'un technicien dans un cahier sont toujours relatives au même projet. Les notes d'un projet dans un cahier sont effectuées par le même technicien.

**6b-** S: Employé, T: Projet, V: Lieu. Pour un projet, un employé ne travaille que dans un seul lieu. Dans un lieu, un employé ne travaille que pour un seul projet.

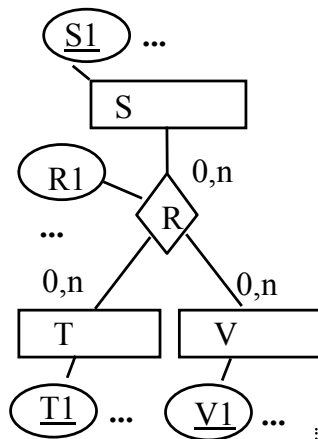


# E/A vers relationnel

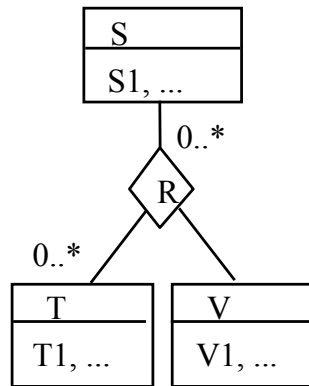
MODELE EA

UML

MODELE RELATIONNEL



$S1, T1 \rightarrow V1$



$S(\underline{S1}, \dots)$

$T(\underline{T1}, \dots)$

$V(\underline{V1}, \dots)$

$R(\underline{S1}, \underline{T1}, V1, R1, \dots)$

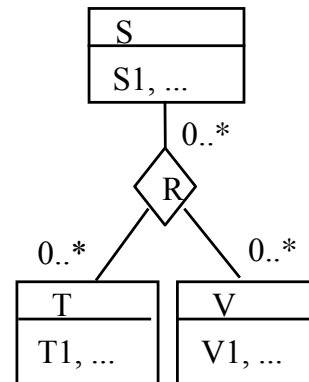
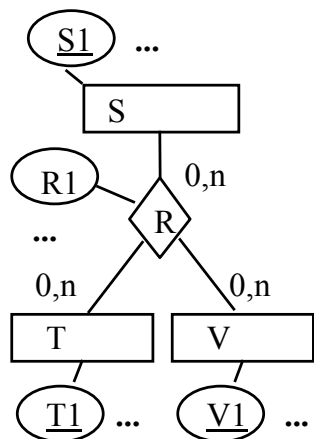
$R.V1$  not null

$R.S1$  ref  $S.S1$  on delete cascade  
on update cascade

$R.T1$  ref  $T.T1$  on delete cascade  
on update cascade

$R.V1$  ref  $V.V1$  on delete cascade  
on update cascade

**6c-** S: Projet, T: Technicien, V: Responsable. Un technicien travaillant pour un projet a un seul responsable. Un projet peut avoir plusieurs responsables, un technicien peut travailler sur plusieurs projets, un technicien peut avoir plusieurs responsables, un responsable peut avoir plusieurs projets.



$S(\underline{S1}, \dots)$

$T(\underline{T1}, \dots)$

$V(\underline{V1}, \dots)$

$R(\underline{S1}, \underline{T1}, \underline{V1}, R1, \dots)$

$R.S1$  ref  $S.S1$  on delete cascade  
on update cascade

$R.T1$  ref  $T.T1$  on delete cascade  
on update cascade

$R.V1$  ref  $V.V1$  on delete cascade  
on update cascade

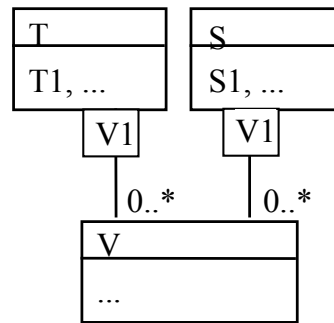
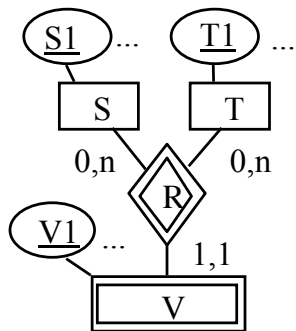
**6d-** S: Employé, T: Métier, V: Projet. Un employé peut exercer différents métiers sur un projet parmi un ensemble de projets, et chaque projet est mené par plusieurs employés avec des métiers variés.

# E/A vers relationnel

MODELE EA

UML

MODELE RELATIONNEL



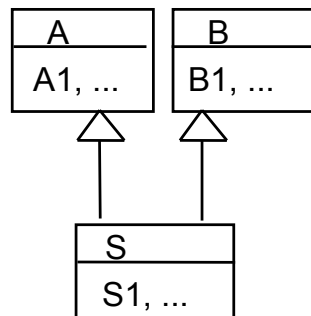
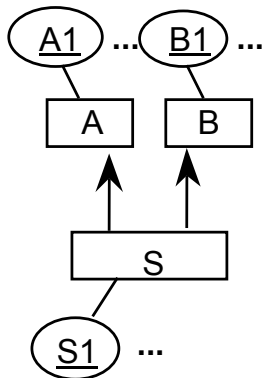
$S(\underline{S1}, \dots)$   
 $T(\underline{T1}, \dots)$   
 $V(\underline{V1}, \underline{S1}, \underline{T1}, \dots)$   
 $V.S1 \text{ ref } S.S1$   
 on delete cascade  
 on update cascade  
 $V.T1 \text{ ref } T.T1$   
 on delete cascade  
 on update cascade

Ex:  $S=\text{Résidence}$ ;  $V=\text{Appartement}$

## 7- Entité faible

Traduire chaque type d'entité faible V en une table V. La clé primaire de la table V est la concaténation de l'identificateur partiel de l'entité V avec les clés primaires des tables correspondant aux entités propriétaires dans l'association identificatrice de V.

Une card min de 1 pour S ou T peut être imposée par des contraintes dynamiques (triggers).



$A(\underline{A1}, \dots, S1)$   
 $B(\underline{B1}, \dots, S1)$   
 $S(\underline{S1}, \dots, A1, B1)$   
 $S.A1 \text{ ref } A.A1$   
 on delete cascade  
 on update cascade  
 $S.B1 \text{ ref } B.B1$   
 on delete cascade  
 on update cascade

Ex:  $A=\text{Etudiant}$ ;  $B=\text{Enseignant}$ ;  
 $S=\text{Moniteur}$

## 8- Sous-classe

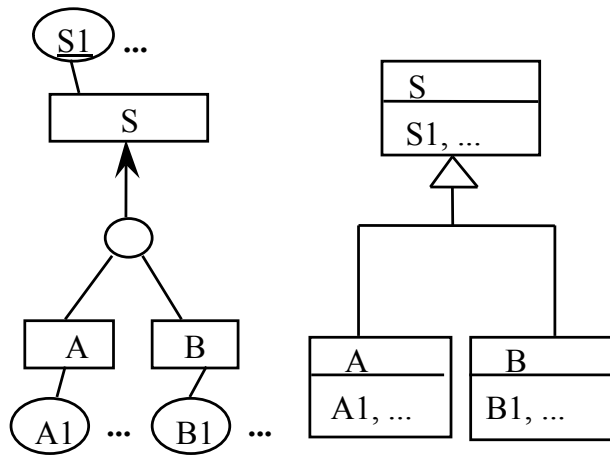
Une sous-classe S possédant deux superclasses A et B est traduite par une relation S qui fait référence à chacune des relations A et B modélisant les classes A et B.

# E/A vers relationnel

MODELE EA

UML

MODELE RELATIONNEL



*Ex: S=Employé; A=Administratif;  
B=Technicien*

## **Solution 1**

$S(\underline{S1}, \dots)$

$A(\underline{S1}, A1, \dots)$

$B(\underline{S1}, B1, \dots)$

A.S1 ref S.S1

on delete cascade

on update cascade

B.S1 ref S.S1

on delete cascade

on update cascade

## **Solution 2**

$S(\underline{S1}, \dots, A1, \dots, B1, \dots, T)$

T not null : spécialisation totale  
disjointe

## **Solution 3**

$S(\underline{S1}, \dots, A1, \dots, B1, \dots, TA, TB)$

TA et TB not null : spécialisation  
non disjointe et totale

## **Solution 4**

$A(\underline{S1}, \dots, A1, \dots)$

$B(\underline{S1}, \dots, B1, \dots)$

convient pour une spécialisation  
totale et disjointe

## **9- Spécialisation**

Une spécialisation  $Z = \{A, B\}$  sur  $S$  peut être traduite de quatre manières.

9a- Créer une table pour chaque type  $S$ ,  $A$ ,  $B$ . Les clés primaires de  $A$  et  $B$  sont des clés étrangères relatives à la clé primaire de  $S$ .

9b- (pour une spécialisation disjointe) Créer une seule table  $S$  rassemblant tous les attributs de  $S$ ,  $A$ ,  $B$ . Un dernier attribut,  $T$ , indique le type de chaque tuple de  $S$  (soit  $A$ , soit  $B$ ). Pour ce tuple seuls les attributs du type  $T$  sont définis.

9c- (généralisation de 11b pour une spécialisation non disjointe). Introduire dans la table  $S$  autant d'attributs de types que de sous-classes dans la spécialisation. Pour un tuple de  $S$ , chacun de ces attributs prend la valeur 1 ou la valeur 0 en fonction du type du tuple.

9d- Solution dérivée de 11a en supprimant la table  $S$  et en incorporant ses attributs dans chacune des tables  $A$ ,  $B$ . Cette solution suppose que toute entité est spécialisée une et une seule fois; pour une spécialisation totale et disjointe.