

<http://www-adele.imag.fr/users/Didier.Donsez/cours>

Le Modèle Entité-Association (et UML) et les Bases de Données Relationnelles

Didier DONSEZ

Université Joseph Fourier
PolyTech'Grenoble - LIG/ADELE

Didier.Donsez@imag.fr

Didier.Donsez@ieee.org

Motivations

■ Modèle Entité/Association (*Entity/Relationship*)

- point de départ aux MCD de nombreuses méthodes
 - Merise, OMT, UML, ...
- extension objet (héritage)

■ Modèle Relationnel

- modèle de base pour les SGBDs relationnels
 - >70 % des applications SI
- théorie de la normalisation
 - suppression des redondances
 - mais il faut déterminer toutes les dépendances fonctionnelles

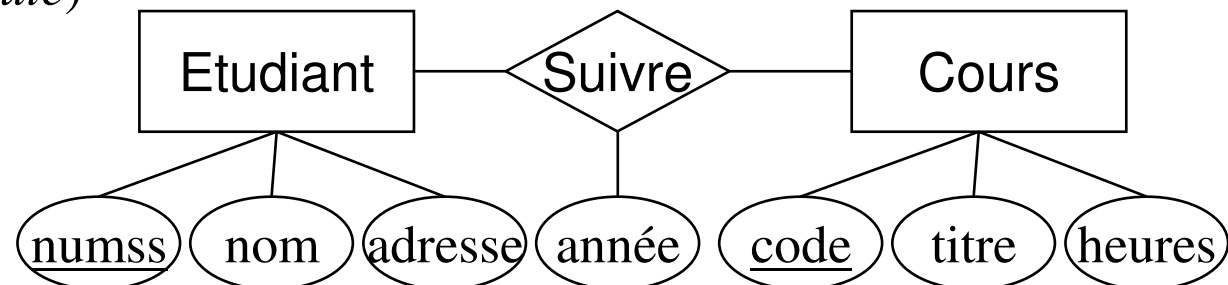
■ Transformation E/A vers le Relationnel

Modèle Entité-Association E/A [Chen76]

Entity-Relationship Model E/R

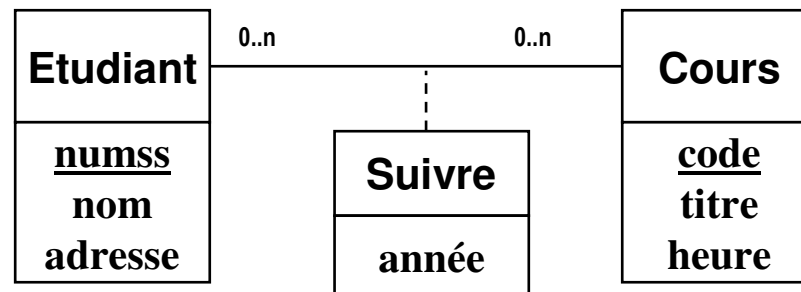
■ Concepts de Base

- Type d'Entité (*Entity Set*)
- Type d'Association (*Relationship Set*)
- Attribut (*Attribute*)
- Clé (*Key*)



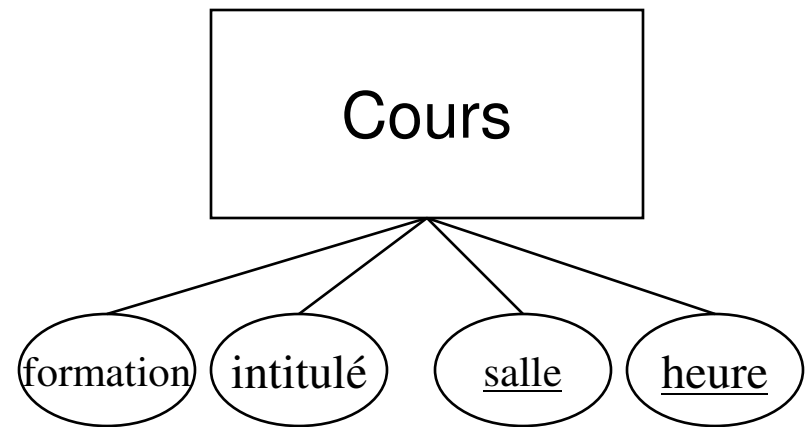
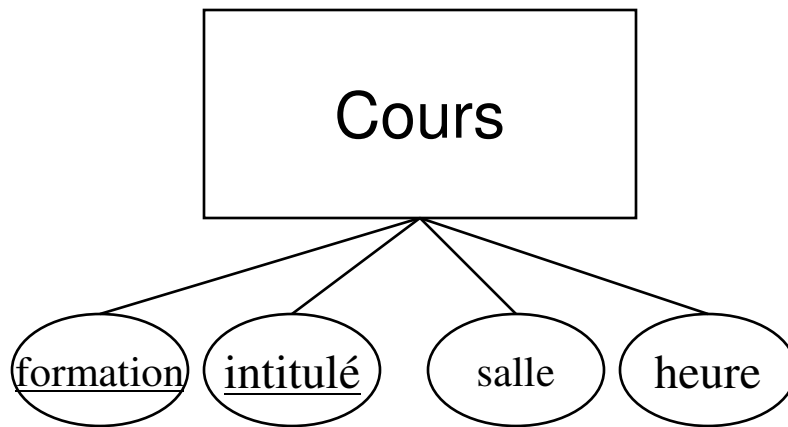
■ Base pour de nombreux autres méthodologies

- Merise, OMT, Booch, ..., **Diagramme de classes UML (1 & 2)**



Les Clés

- Une entité a une et une seule clé
- même si plusieurs clés sont candidates
 - les attributs clé sont soulignés



Choisir une Clé

■ Naturelle (*Natural*)

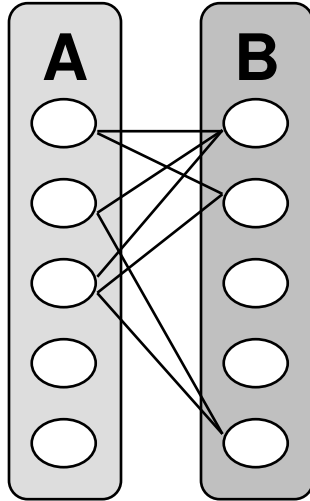
- Attribut seul ou composition d 'attributs
 - doit toujours être renseigné (NOT NULL)
- Remarque : Attribut clé représentant une composition
 - ISBN : numéro intl d 'éditeur + numéro d 'ouvrage
 - GENCOD, EAN, SKU : numéro intl de fabricant + numéro de produit
- clé d 'une entité faible

■ Artificielle (*Surrogate*)

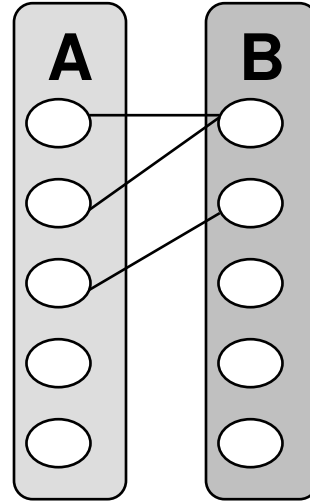
- introduction d 'un attribut artificiel n 'appartenant pas au système décrit.
 - performance en comparaison
 - compacité du stockage
 - souvent employé pour les bases décisionnelles

Cardinalités des Associations

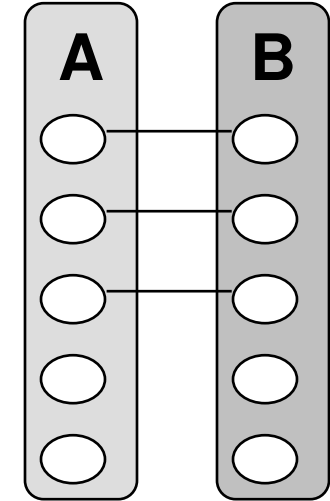
Multiplicity of Relationships



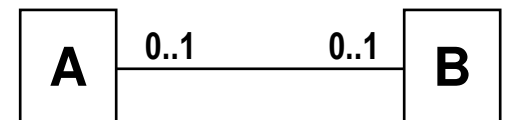
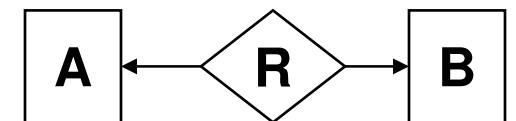
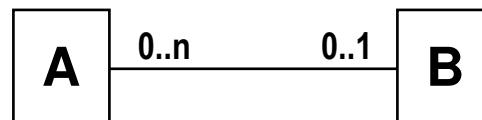
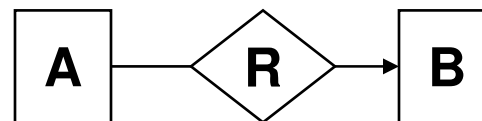
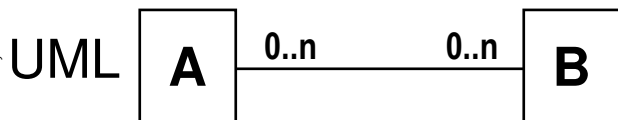
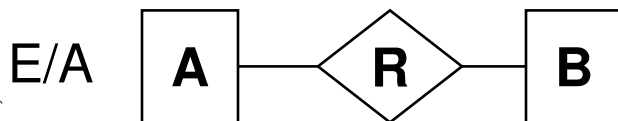
Many-to-many
N-M
0,n - 0,n



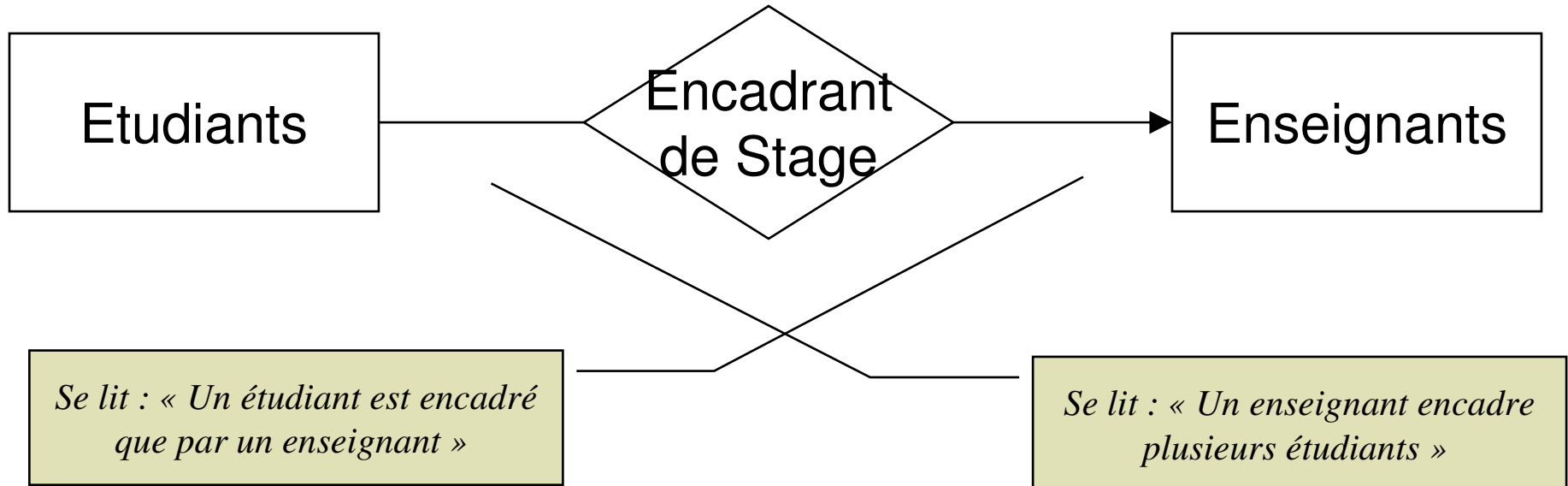
Many-to-one
N-1
0,n - 0,1



One-to-one
1-1
0,1 - 0,1

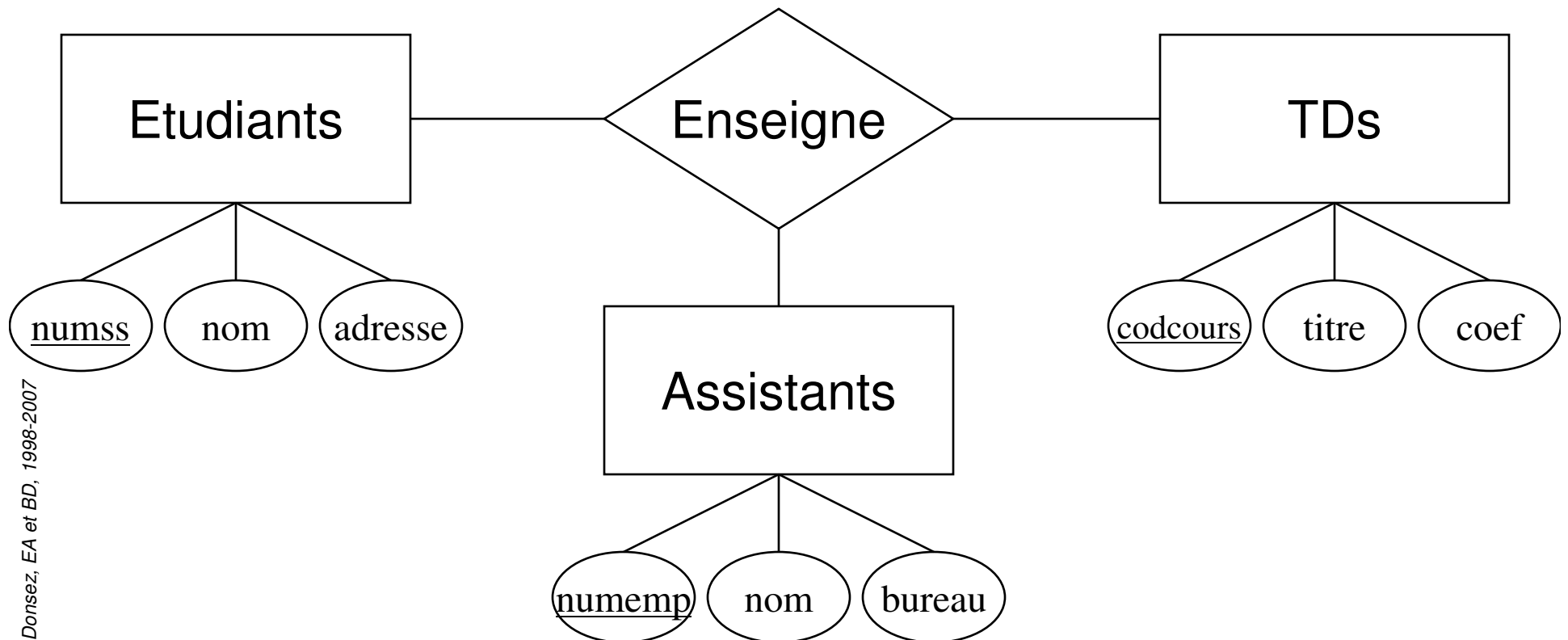


Associativité Many-to-One



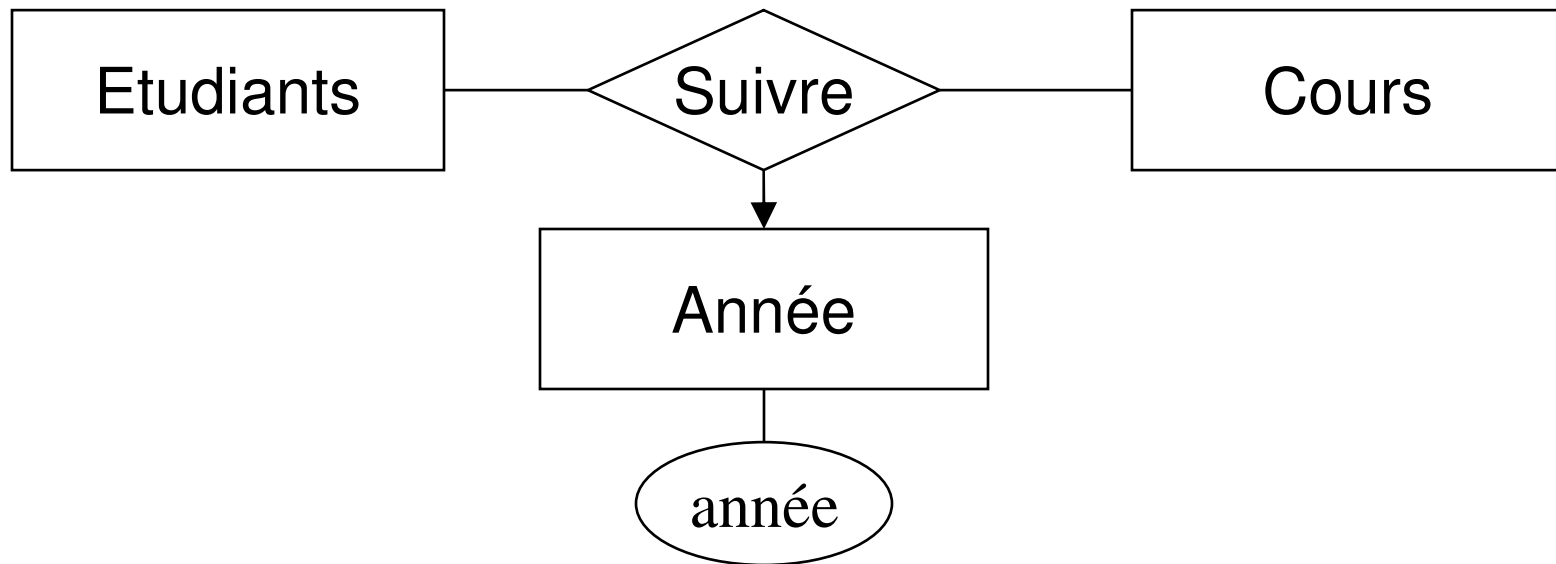
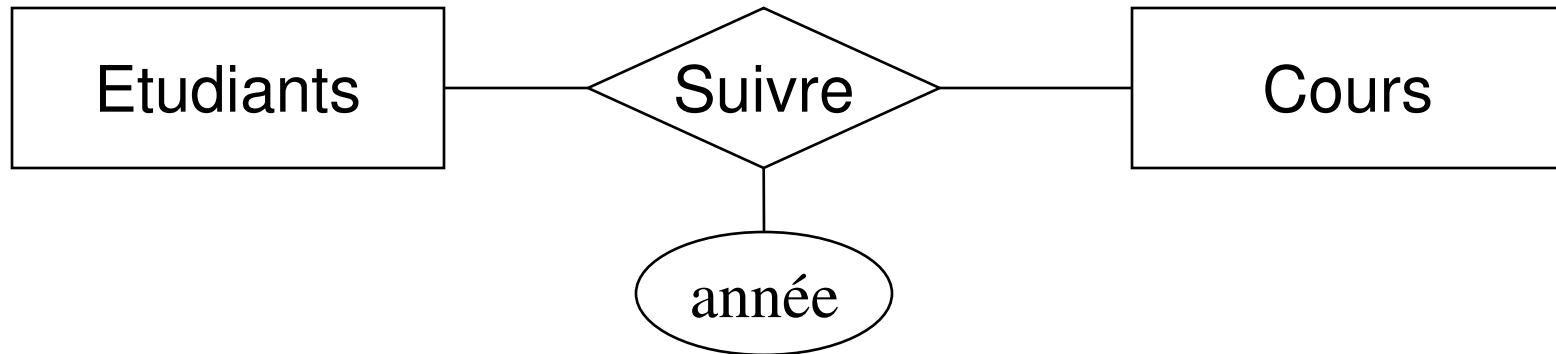
Association n-Aire

- Plusieurs entités peuvent participer à la même associations
- Exemple d 'un association ternaire



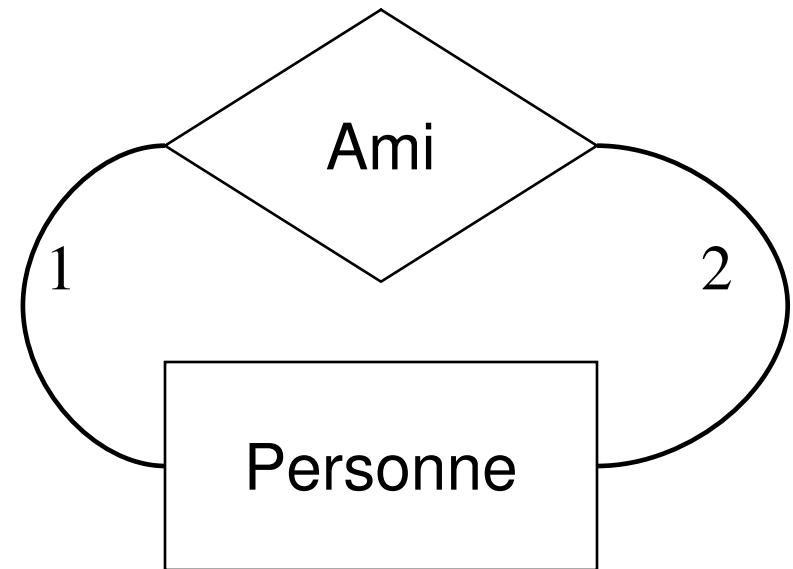
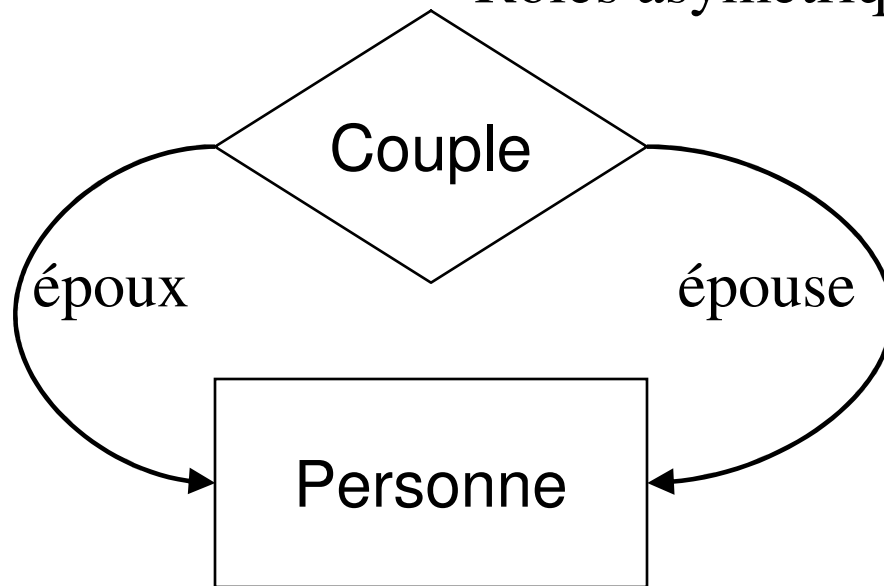
Modèle Entité-Association

Entity-Relationship Model



Rôles dans une association

Rôles asymétrique / symétrique

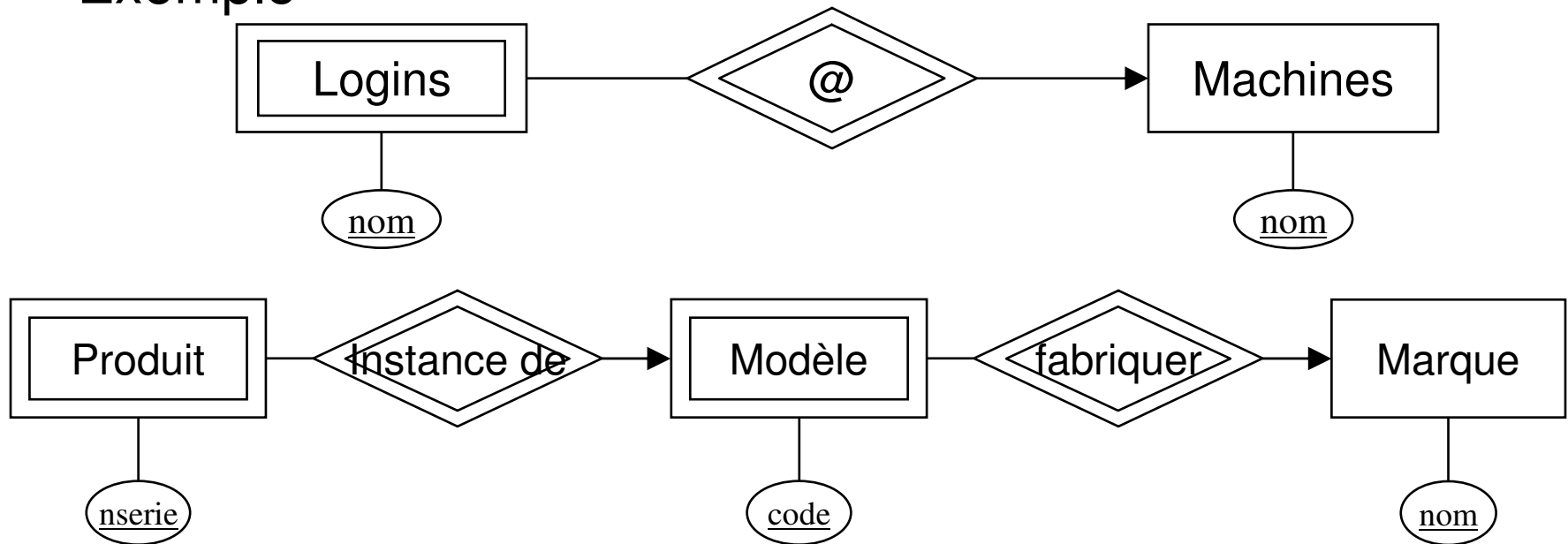


époux	épouse
Joseph Sanson	Marie Dalida

1	2
Pierre Paul	Paul Jean

Les Entités Faibles (*Weak Entity*)

- Parfois, un attribut de la clé d'une entité (*dite faible*) provient d'une autre entité
- Exemple



- Remarque

- Certains attributs clés sont composés et forme la clé d'une entité faible
 - ISBN, GENCOD, EAN, SKU, IBAN, ...

Transformation E/A vers Relationnel

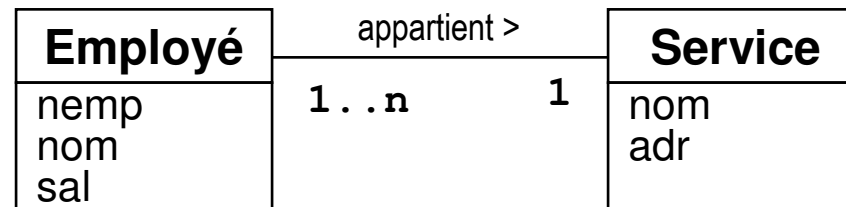
■ But

- Implantation du modèle E/A sur un SGBD réel

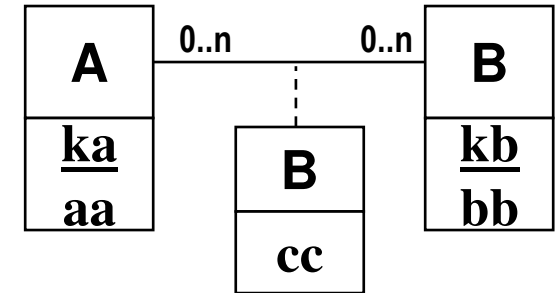
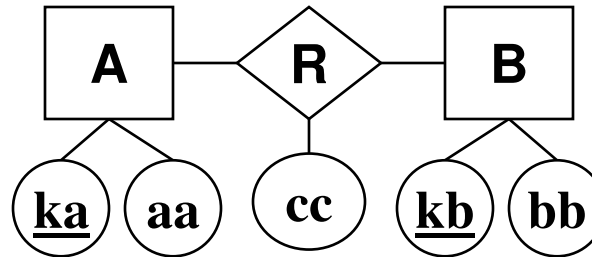
■ Méthode

- transformation
des associations n-aires en associations binaires
- 1 entité = 1 relation
- 1 association = dépendant de sa cardinalité

■ Cas de navigation restreinte



Cas général : Association Many-to-Many



Many-to-Many
N-M

A	<u>ka</u>	aa

R	<u>ka</u>	<u>kb</u>	cc

B	<u>kb</u>	bb

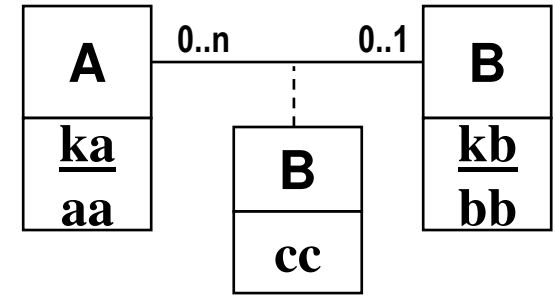
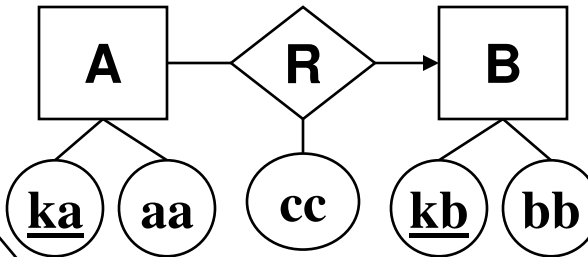

```
create table A (
  ka ..., aa ...,
  primary key(ka)
)
```

```
create table R (
  ka ..., kb ..., cc ...,
  primary key(ka,kb),
  foreign key (ka) references A,
  foreign key (kb) references B
)
```

```
create table B (
  kb ..., bb ...,
  primary key(kb)
)
```

Cas d'une Association Many-to-One

Coût des jointures !



A	<u>ka</u>	aa
...

R	<u>ka</u>	kb	cc
...

B	<u>kb</u>	bb
...

Many-to-One
N-1

A	<u>ka</u>	aa	kb	cc
...

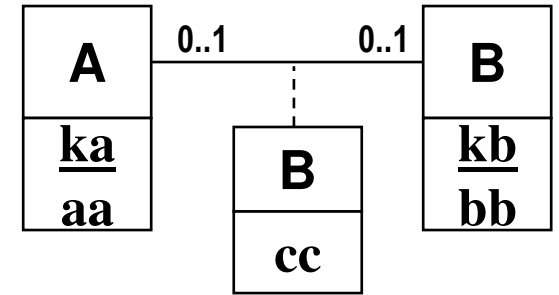
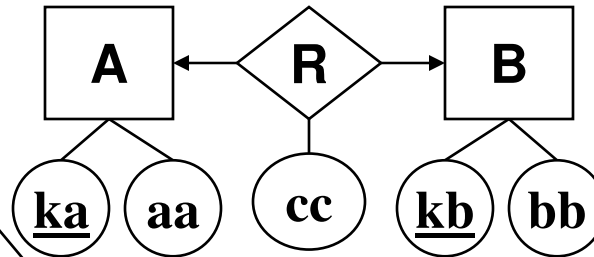
B	<u>kb</u>	bb
...

```
create table A (
  ka ..., kb ..., aa..., cc ...,
  primary key(ka),
  foreign key (kb) references B
  check(cc IS NULL
        OR kb IS NOT NULL)
)
```

```
create table B (
  kb ..., bb ...,
  primary key(kb)
)
```

Cas d'une Association One-to-One

Coût des jointures !



A	<u>ka</u>	aa
...

R	<u>ka</u>	<u>kb</u>	cc
...

B	<u>kb</u>	bb
...

One-to-One
1-1

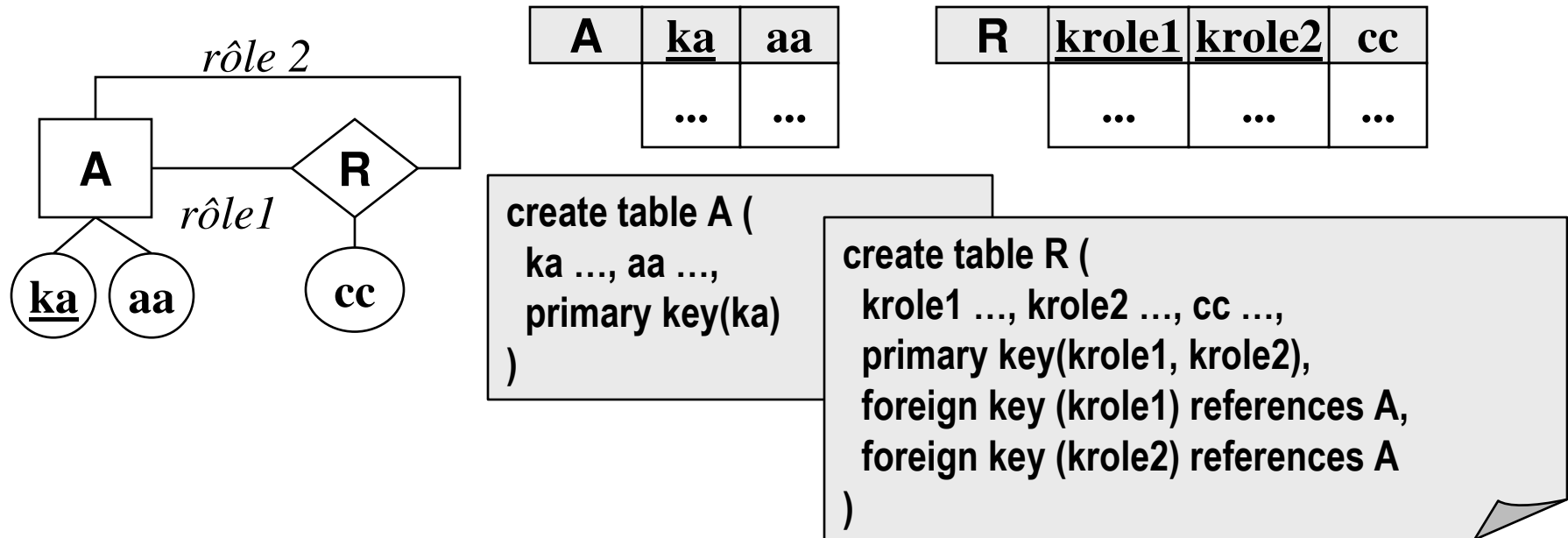
A	<u>ka</u>	aa	kb	cc
...

B	<u>kb</u>	bb
...

A	<u>ka</u>	aa
...

B	<u>kb</u>	bb	ka	cc
...

Cas d'une Association avec des Rôles



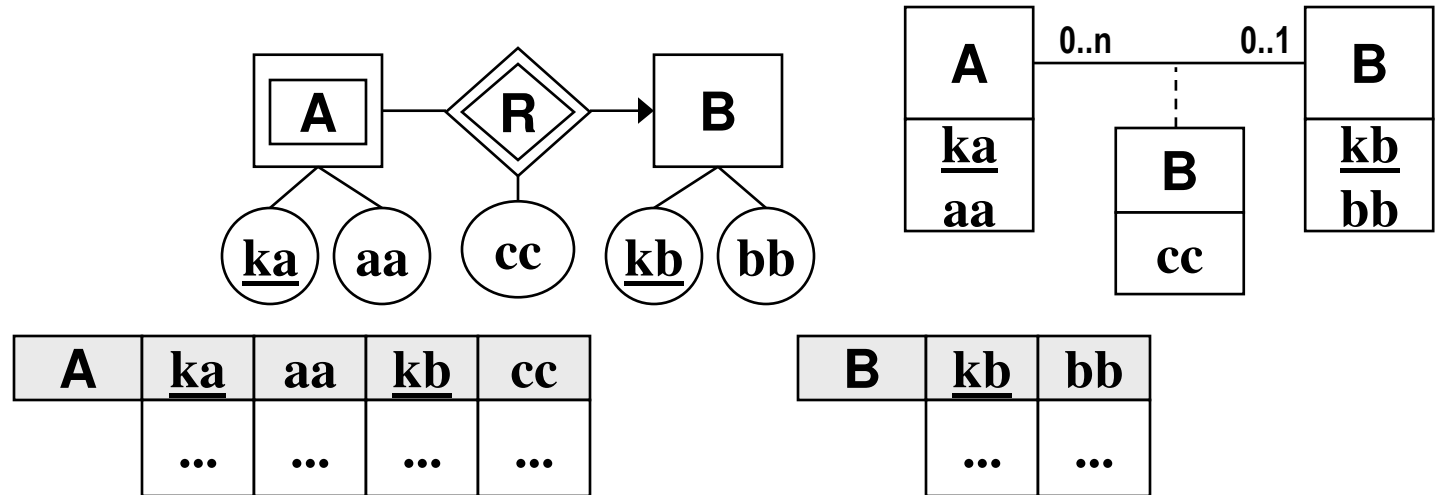
■ Rôle symétrique et One-To-One

- quel est le danger de cette modélisation ?

A	<u>ka</u>	aa	krole2	cc

*Rôle Symétrique
et One-to-One*

Cas de l'entité faible (Weak Entity)



```

create table A (
  ka ..., kb ...,
  aa..., cc ...,
  primary key(ka,kb),
  - - différent du cas N-to-1
  foreign key (kb) references B
)
  
```

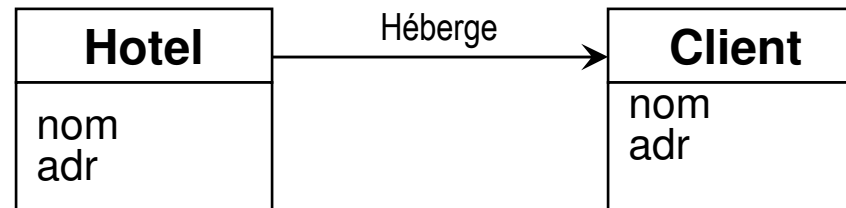
```

create table B (
  kb ..., bb ...,
  primary key(kb)
)
  
```

- Cas de l'agrégation en UML

Cas de navigation restreinte

- La navigation est unidirectionnelle
- Exemple



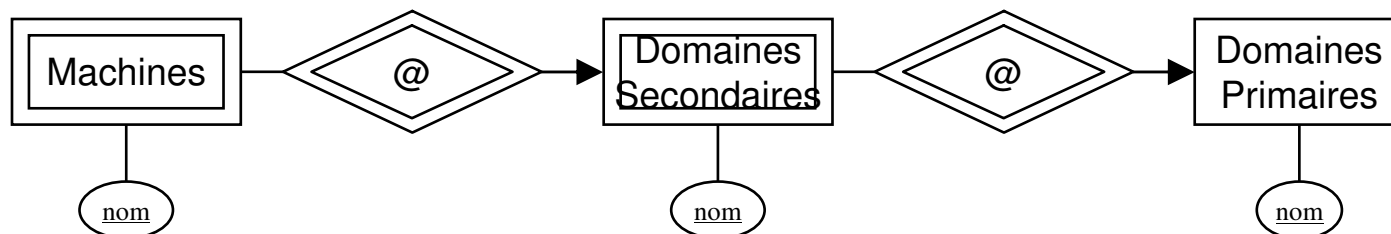
Exercices de transformation E/A vers Relationnel

■ Exercice 1

- *transformez les cas de l'entité faible avec 3 tables A, B, R*

■ Exercice 2

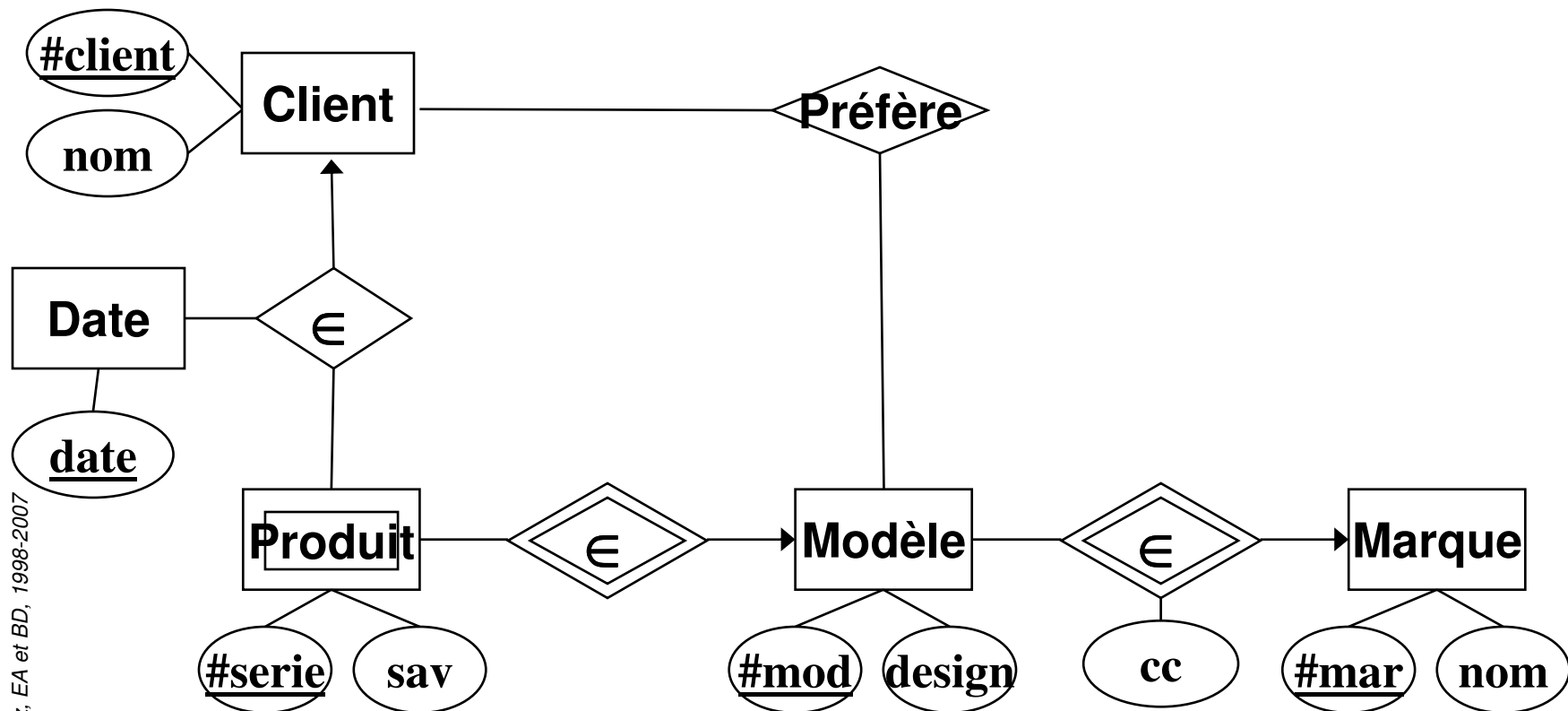
- *transformez le schéma suivant*
 - attention à l'ordre des noms



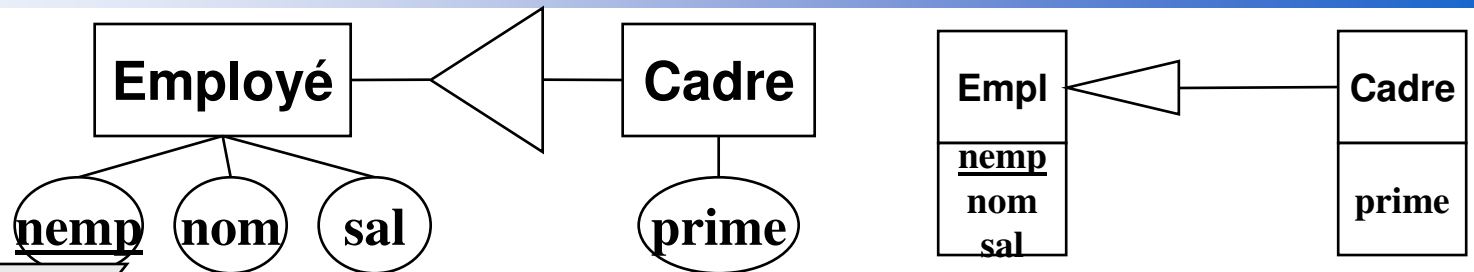
Exercices de transformation E/A vers Relationnel

■ Exercice 3

- *transformez le schéma suivant*



Le Cas des Sous Entités (i)



1 - Separation

Emp	<u>nemp</u>	nom	sal
	100	Dupond	5000
	200	Durant	20000

Cadre	<u>nemp</u>	prime
	200	10000

Cadre.nemp est une clé étrangère sur Emp

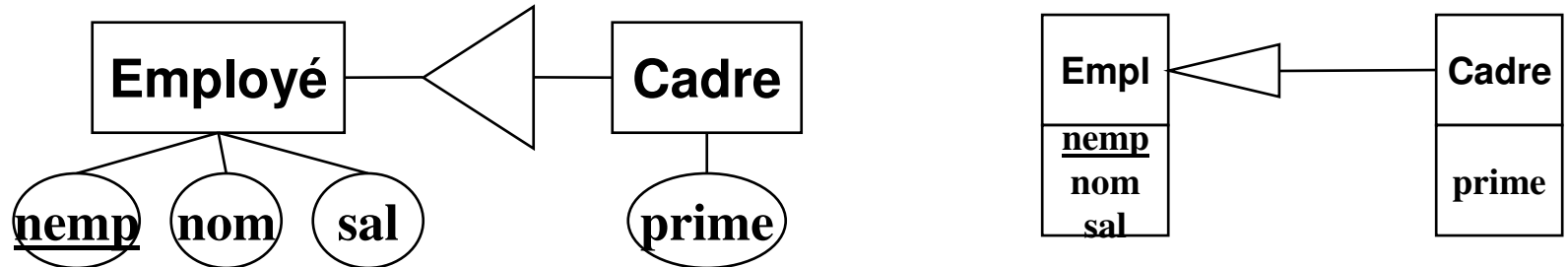
2 - Push down

Emp	<u>nemp</u>	nom	sal
	100	Dupond	5000

Cadre	<u>nemp</u>	nom	sal	prime
	200	Durant	20000	10000

CI : {Emp.nemp} ∩ {Cadre.emp} = ∅

Le Cas des Sous Entités (ii)



3 - Push Up

Emp	<u>nemp</u>	nom	sal	prime
	100	Dupond	5000	NULL
	200	Durant	20000	10000

Le Cas des Sous Entités (iii)

■ Inconvénients

1

- Jointure pour reconstituer Cadre

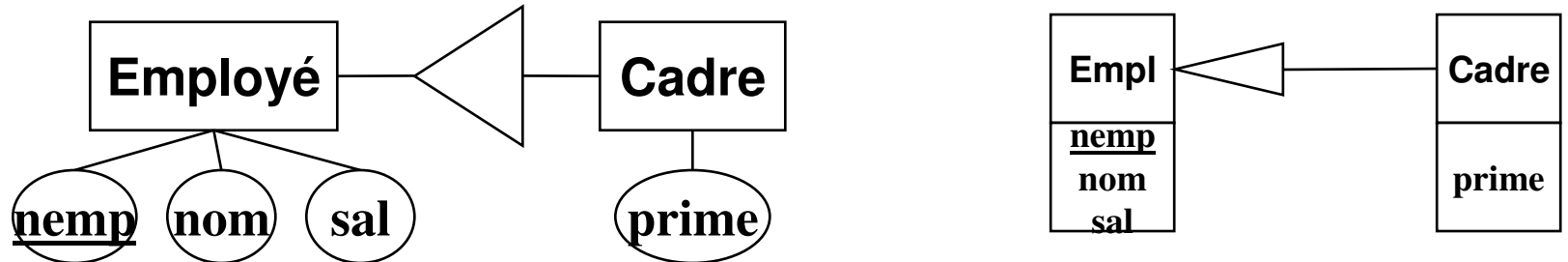
2

- Union pour reconstituer Employé

3

- Pas de distinction entre un Employée et un Cadre avec une prime NULL
- en général, pas d'abstraction dans les Traitements

Amélioration du Cas 3 - Push Up



4

Emp	typemp	<u>nemp</u>	nom	sal	prime
	EMP	100	Dupond	5000	NULL
	CAD	200	Durant	20000	10000

Emp	<u>typemp</u>
	EMP
	CAD

Emp.typemp est une clé étrangère
 CI : Emp.typemp =EMP ⇒ prime=NULL

5

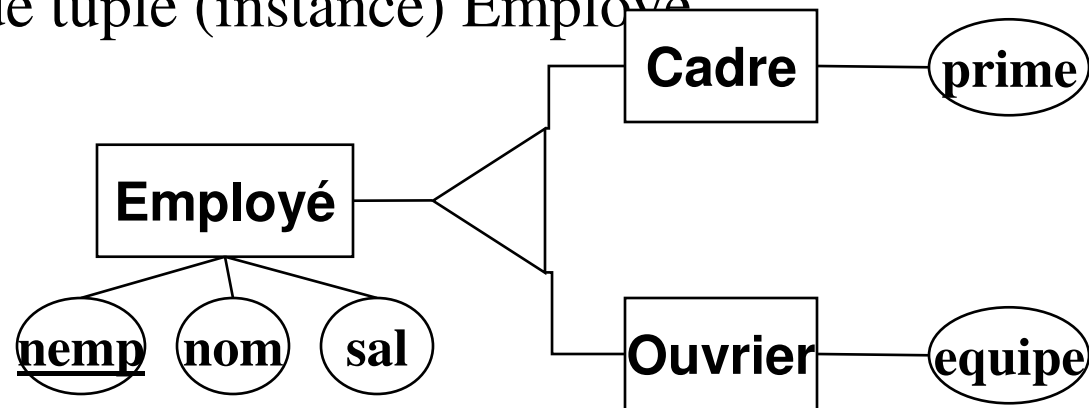
Emp	typemp	<u>nemp</u>	nom	sal	prime
	EMP	100	Dupond	5000	NULL
	CAD	200	Durant	20000	10000

Domaine typemp, CD typeemp IN (EMP,CAD)
 CI : Emp.typemp =EMP ⇒ prime=NULL

Entité Abstraite

■ Employé : entité abstraite (notion dans Java, C#, C++, UML, ...)

- pas de tuple (instance) Employé



Emp	typemp	<u>nemp</u>	nom	sal	prime	equipe
	OUV	100	Dupond	5000	NULL	Nuit
	CAD	200	Durant	20000	10000	NULL

Domaine typemp

CD typeemp IN (OUV,CAD)

CI : Emp.typemp !=CAD ⇒ prime=NULL

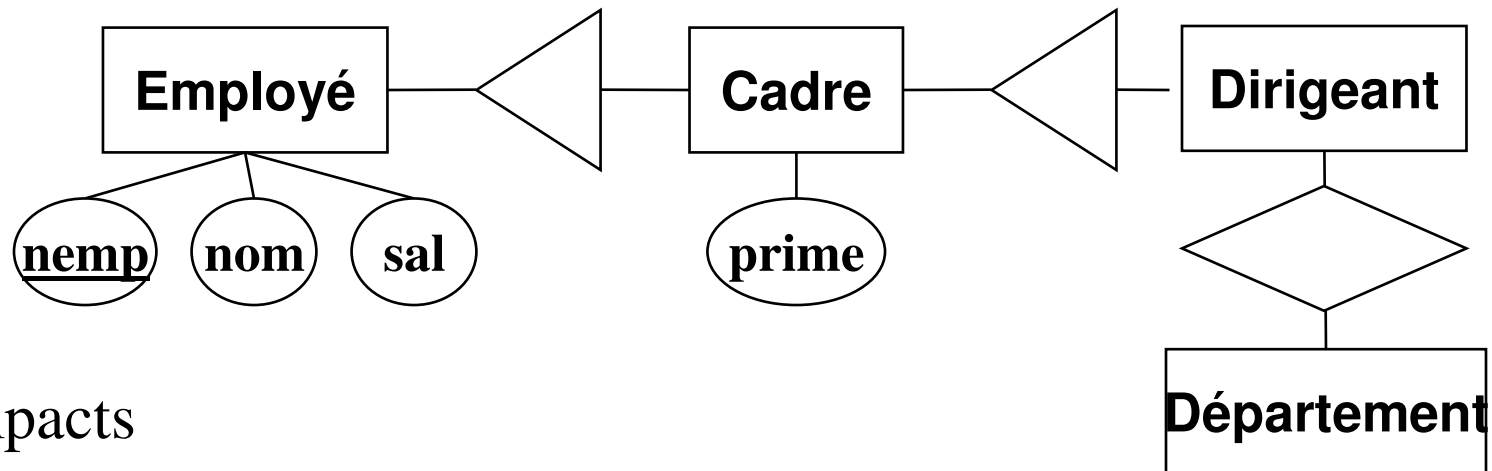
CI : Emp.typemp !=OUV ⇒ equipe=NULL

Transformation E/A vers Relationnel

L'évolution de Schéma

■ Inconvénients

- Ajout d'une sous entité Dirigeant



- Impacts

sur le schéma

sur les contraintes d'intégrité

sur les contraintes de domaines

sur les traitements

■ Solution : l'Objet-Relationnel ou l'Objet Pur

Transformation E/A vers l'Objet-Relationnel

■ But

- Implantation du modèle E/A
sur un SGBD Objet-Relationnel

■ Fonctionnalités disponibles

- pour les Sous-Entités
 - Héritage de Type et Héritage de Table
- pour les Associations
 - table indépendante ou collection imbriquée (*tableau dimensionnable, table imbriquée*)
 - clé étrangère ou référence d'objet
 - nombreuses solutions [Soutou99] pp84,93,103-105

Héritage

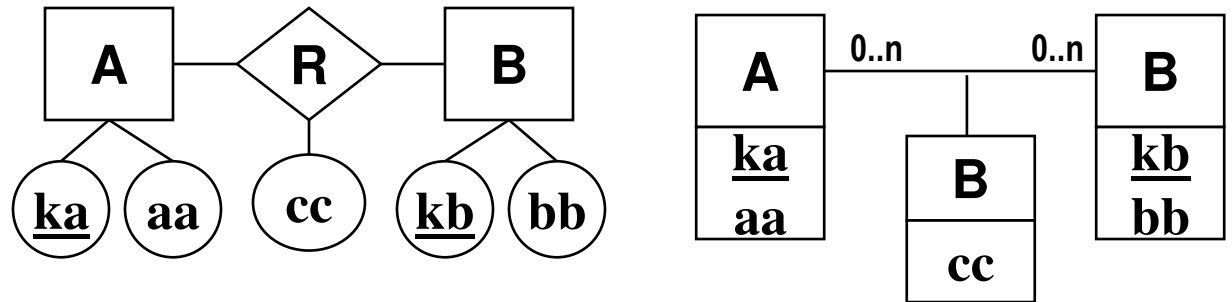
■ Héritage de Table

Mettre en correspondance les Sous-Entités avec des Sous Tables

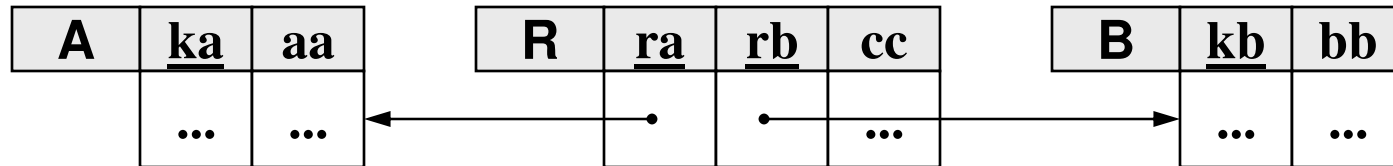
■ Héritage de Type

- Même problème que pour le relationnel pur
- Utilisation des sous types dans le Push-Down

Cas général : Association Many-to-Many (i)



Many-to-Many
N-M



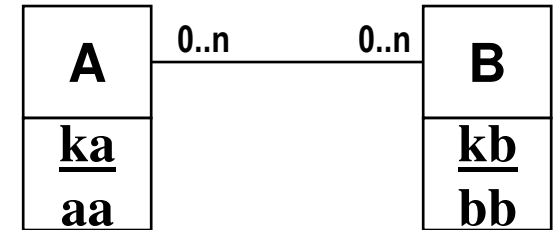
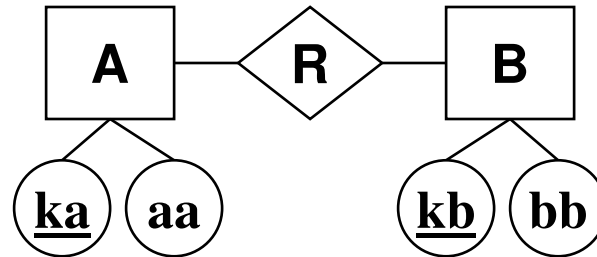
```
create type tA as object (
  ka ..., aa ...,
);
create table A of tA (
  primary key(ka)
);
```

```
create table R (
  ra REF(tA), rb REF(tB),
  cc ...,
  primary key(ra,rb),
  SCOPE FOR ra IS A,
  SCOPE FOR rb IS B
);
```

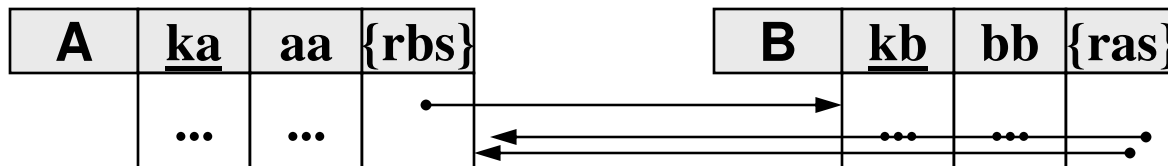
```
create type tB as object (
  kb ..., bb ...,
);
create table B of tB (
  primary key(kb)
);
```

Cas général : Association Many-to-Many (ii)

Attention à la mise à jour des listes de refs => risque d'incohérence



Many-to-Many
N-M

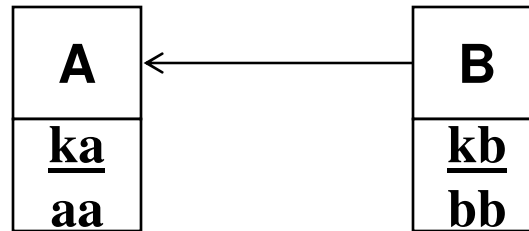


```
create type tB as object;
create type tA as object(
  ka ..., aa ...,
  rbs varray(10) of REF(tB),
);
create table A of tA (
  primary key(ka),
);
```

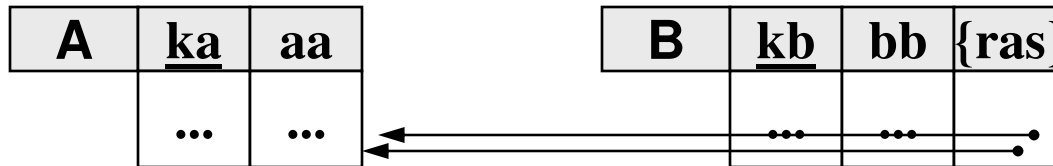
```
create or replace type tB
as object (
  kb ..., bb ...,
  ras varray(10) of REF(tA),
);
create table B of tB (
  primary key(kb)
);
```

Cas général : Association Many-to-Many (ii)

En UML: sens unidirectionnel de navigation (B vers A)



Many-to-Many
N-M



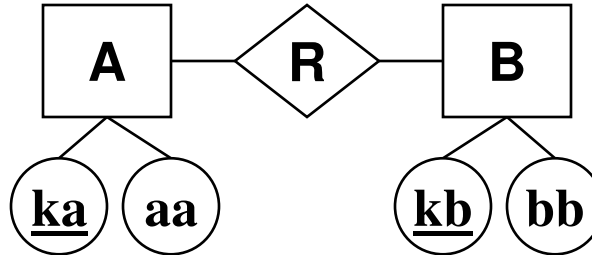
```

create type tA as object(
  ka ..., aa ...
);
create table A of tA (
  primary key(ka),
);
    
```

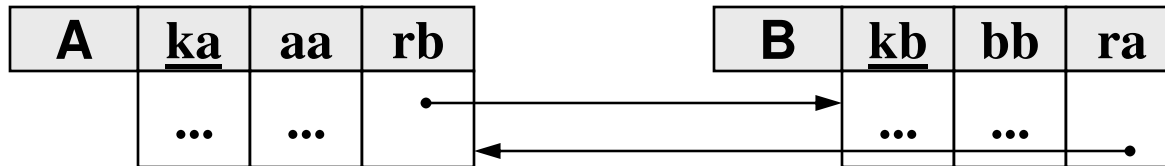
```

create type tB as object (
  kb ..., bb ...,
  ras varray(10) of REF(tA)
);
create table B of tB (
  primary key(kb)
);
    
```

Cas général : Association One-To-One



One-To-One
1,1



```

create type tB as object;
create type tA as object (
  ka ..., aa ..., rb REF(tB)
);
create table A of tA (
  primary key(ka),
  SCOPE FOR rb IS B
);
    
```

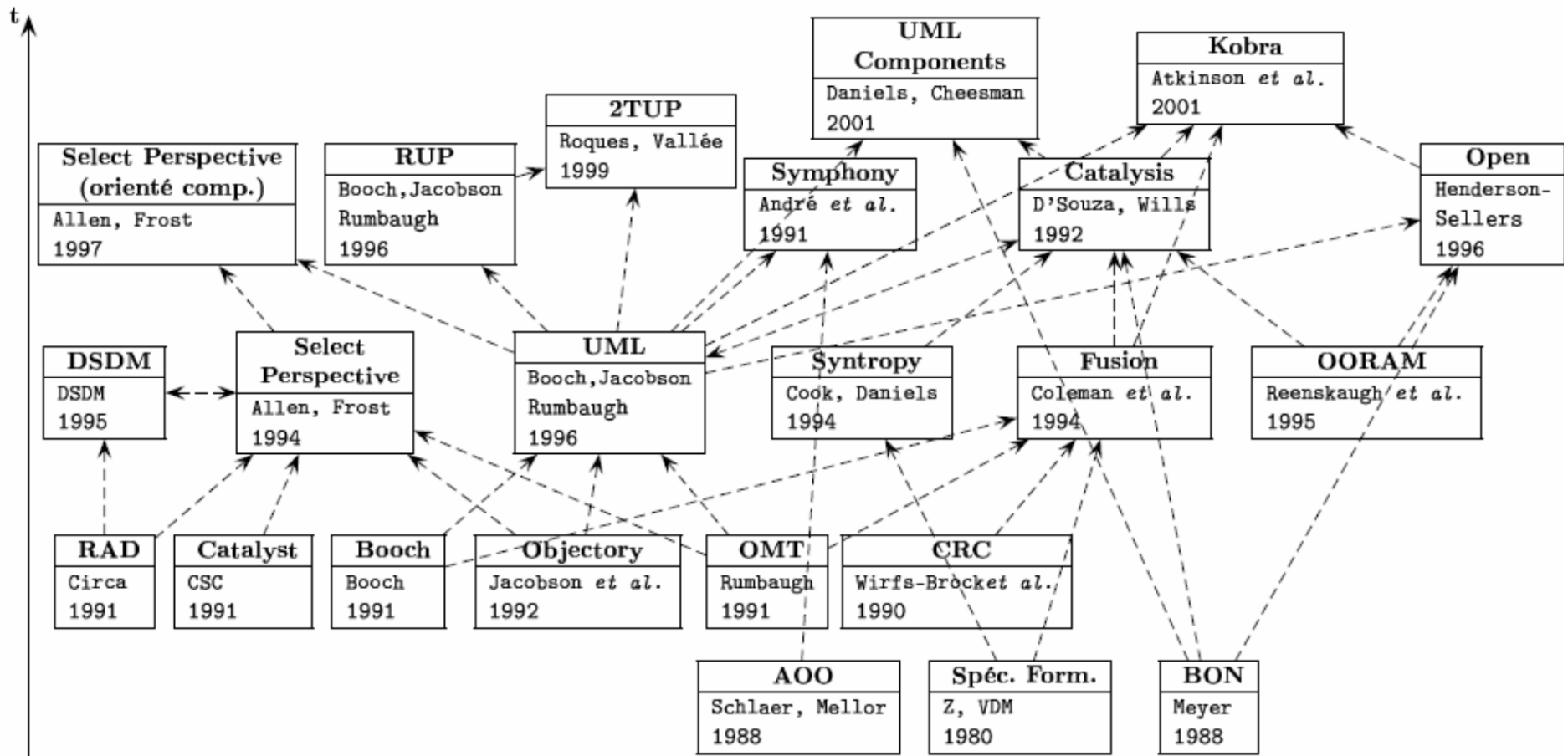
```

create or replace type tB as object (
  kb ..., bb ..., ra REF(tA)
);
create table B of tB (
  primary key(kb),
  SCOPE FOR ra IS A
);
    
```


Bibliographie

- Roger Mounyol, "Merise par l'Exemple", Ed Ellispes, 1991, ISBN 2-7298-9114-5
- Chris Date, "Introduction aux Bases de Données", 6^{ème} édition, Ed Intl Thomson Publ. ISBN 2-84180-964-1, 970 pp
- Jeffrey D. Ullman, Jennifer Widom, "A First Course in Database Systems", 1^{ère} édition, Ed. Prentice Hall Engineering, Science & Math, Avril 1997, ISBN 0-13-861337-0, 470 pp.
- PPS Chen. The Entity-Relationship model toward a unified view of data. ACM Transactions on Database Systems, March 1976.
- Michael Blaha, William Premerlani. Object-Oriented Modeling and Design for Database Applications, Prentice Hall, Englewood Cliffs, New Jersey, 1998.
 - <http://www.omtassociates.com/Pages/papers.html>
- Christian Soutou, "Objet-Relationnel sous Oracle8, Modélisation avec UML", Ed Eyrolles, 1999, ISBN 2-212-09063-3
 - décrit bien les alternatives de conception des associations avec l'objet-relationnel et avec le relationnel
- **Christian Soutou, "De UML à SQL : Conception de bases de données", Ed Eyrolles, 2002, ISBN 2-212-11098-7**
 - La mise à jour du précédent
- Using UML to Design Database Applications
 - <http://www.umlchina.com/Indepth/usinguml.htm>
 - http://www.intelinfo.com/newly_researched_free_training/UML.html

Annexe : Zoo de méthodes



-----> dépendances

d'après HDR Bruel