

ANF « Comment concevoir une base de données en archéométrie »  
Réseau CAI-RN & rBDD - 05/06/2014 au 06/06/2014



# Du monde réel à SQL la modélisation des données

Marie-Claude Quidoz (CEFE/CNRS)



**Ce document est mis à disposition selon les termes de la licence Creative Commons 3.0  
« Attribution - Pas d'Utilisation Commerciale - Partage dans les Mêmes Conditions »**

**Vous êtes libre de :**

- reproduire, distribuer et communiquer l'œuvre
- adapter l'oeuvre

**Selon les conditions suivantes :**



**Attribution** — Vous devez mentionner le nom des auteurs de la manière suivante :  
« Marie-Claude Quidoz, CEFE-CNRS, 2014 »



**Pas d'Utilisation Commerciale** — Vous n'avez pas le droit d'utiliser cette œuvre à des fins commerciales.



**Partage dans les Mêmes Conditions** — Si vous modifiez, transformez ou adaptez cette œuvre, vous n'avez le droit de distribuer votre création que sous une licence identique ou similaire à celle-ci.

Voir la version intégrale de la licence : <http://creativecommons.org/licenses/by-nc-sa/3.0/fr/legalcode>

# Pourquoi modéliser ?

- Représenter les différents éléments constitutifs du système d'information
- Description des entités et leur dépendances
- La modélisation est un processus important et indispensable car elle conditionne la structure de la base de données

# Utilisation d'un formalisme

- Pourquoi suivre un formalisme ?
  - Utiliser un langage connu et commun
- Deux formalismes très utilisés
  - Entité-Association
  - UML
- Formalisme « quasi identique » si on veut modéliser une base de données
- Des nuances cependant
  - Entité pour EA - classe pour UML
  - Cardinalités indiquées de façon différente

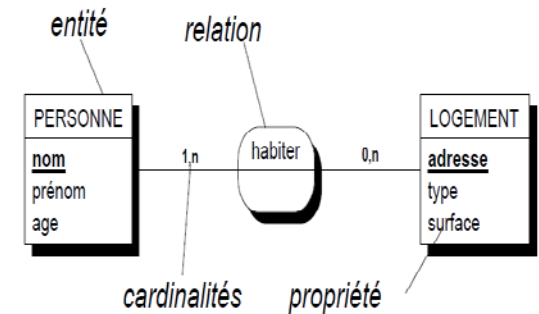
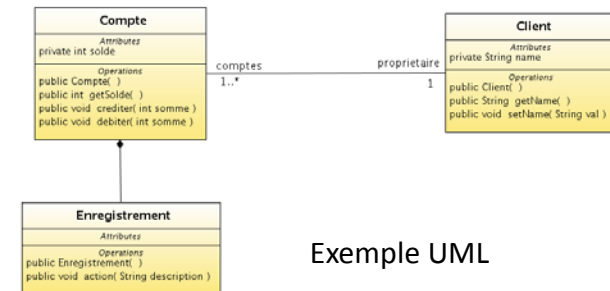


Figure 7.2 : Les concepts du formalisme entité - relation

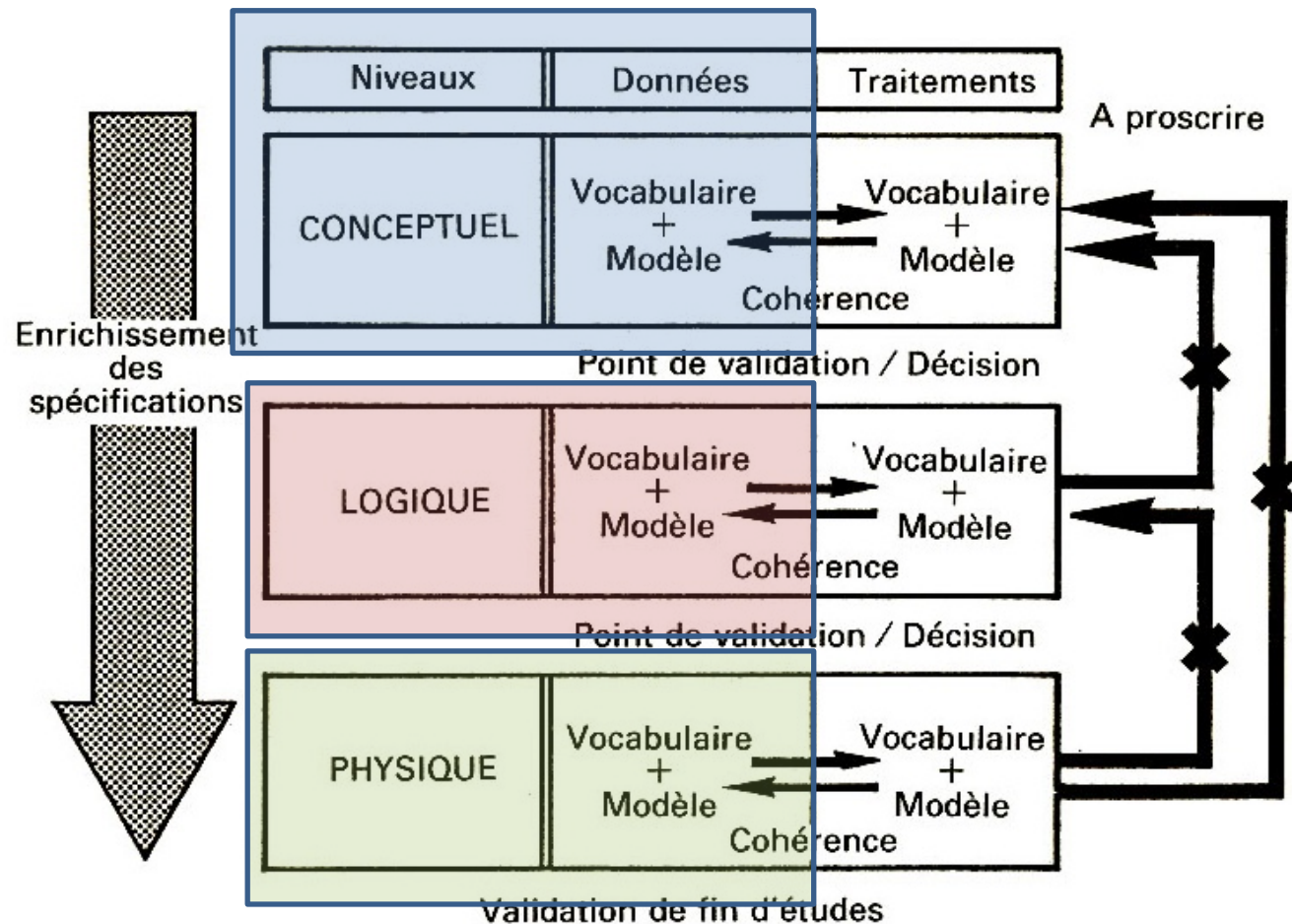


Exemple UML

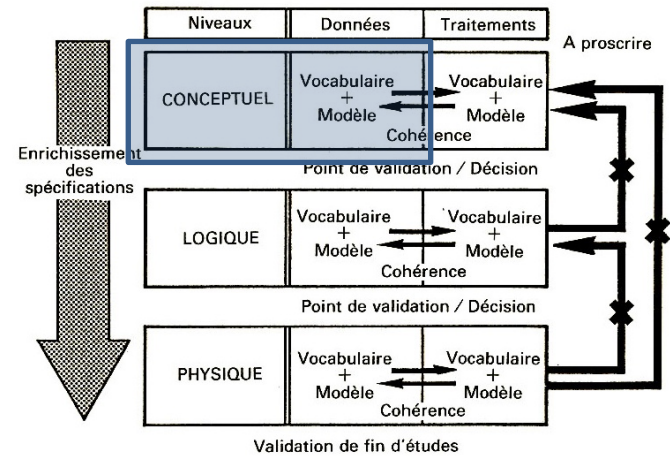
# Utilisation d'une méthode

- Pourquoi utiliser une méthode d'analyse ?
  - Avoir une méthodologie !
- Merise : méthode dans le domaine des bases de données
  - Résultat des travaux menés par Hubert Tardieu dans les années 1970 et qui s'inséraient dans le cadre d'une réflexion internationale, autour notamment du modèle relationnel d'Edgar Frank Codd
- Méthode française complète
  - Qui couvre de l'analyse à la réalisation
  - Qui traite séparément des données et des traitements

# Merise



*La méthode MERISE préconise d'analyser séparément données et traitements, à chaque niveau.*

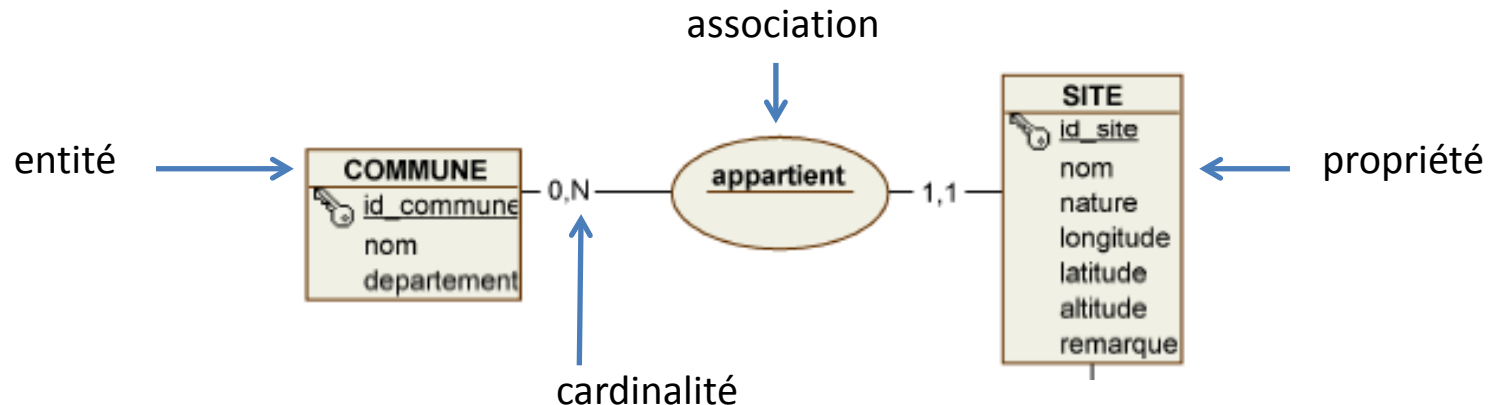


*La méthode MERISE préconise d'analyser séparément données et traitements, à chaque niveau.*

# Modélisation au niveau conceptuel

# Modèle conceptuel de données (MCD)

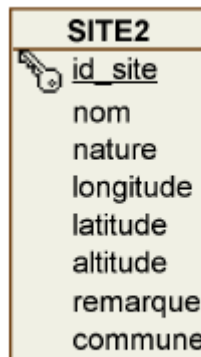
- Représentation graphique et structurée des informations mémorisées dans un système d'information
- 4 notions principales :





# Entité

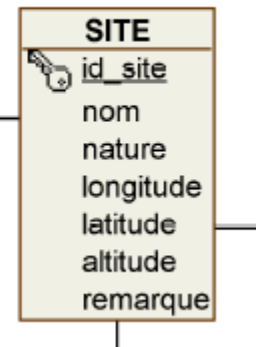
- Ensemble d'objets de même nature, concrets ou abstraits perçus d'intérêt dans le discours
- Choix du concepteur en fonction de l'intérêt que présente cette entité dans son système d'information



on est juste intéressé  
par le nom de la  
commune

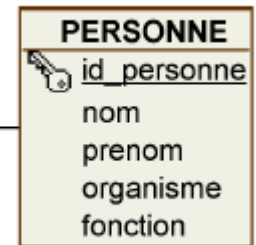


on est intéressé par en  
savoir plus sur la  
commune



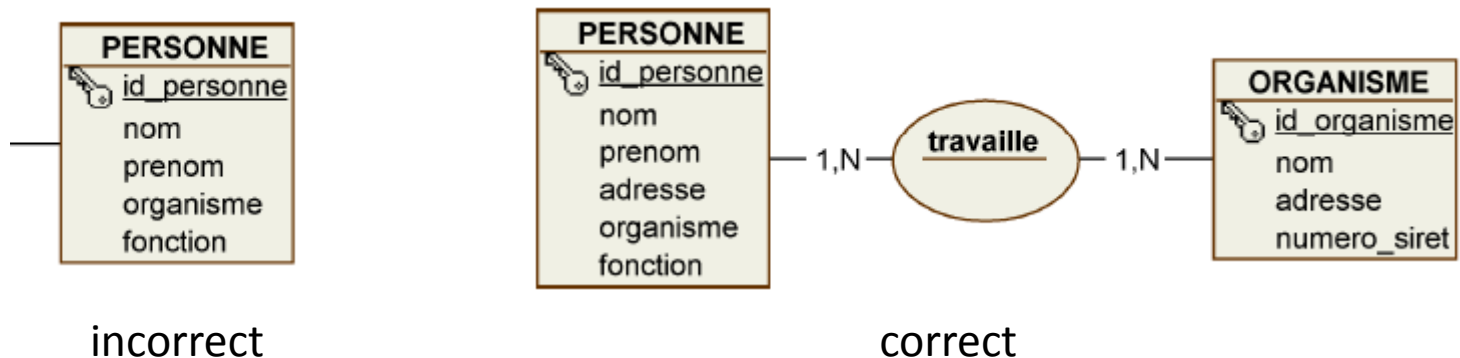
# Choix de l'identifiant

- Chaque entité doit être doté d'un identifiant
  - Propriété simple : nom
  - Propriété composée : nom + prénom
  - Propriété artificielle : id\_personne
  - Possibilité d'avoir plusieurs clefs candidates
- Avantages / inconvénients
  - Une propriété artificielle est toujours unique vu du modèle mais elle ne garantit pas que la personne soit unique
  - Une propriété composée nécessite que chaque composant soit défini (c'est-à dire non nulle)
- Remarque : un identifiant doit être stable



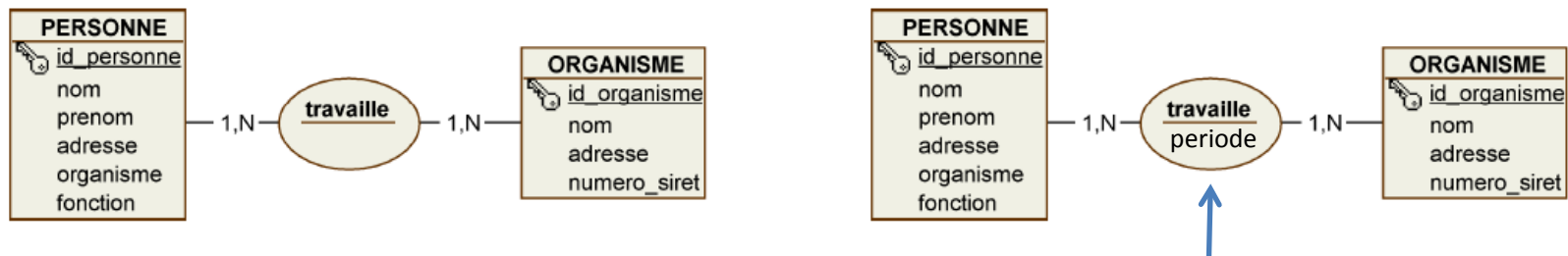
# Propriété

- Une entité est décrite par une liste de propriétés
- A toute occurrence de l'entité type, il ne peut y avoir, dans la mémoire du système d'information, au plus qu'une valeur de la propriété
- Ex : une personne travaille dans deux organismes



# Association

- Liaison entre deux entités qui a une signification propre au système d'information
- Traduit une partie des règles de gestion qui n'ont pas été satisfaites par la simple définition des entités
- Certaines associations peuvent être porteuses de propriété



Ajout de la propriété « période » pour conserver l'historique des employeurs

# Choix du nom de l'association

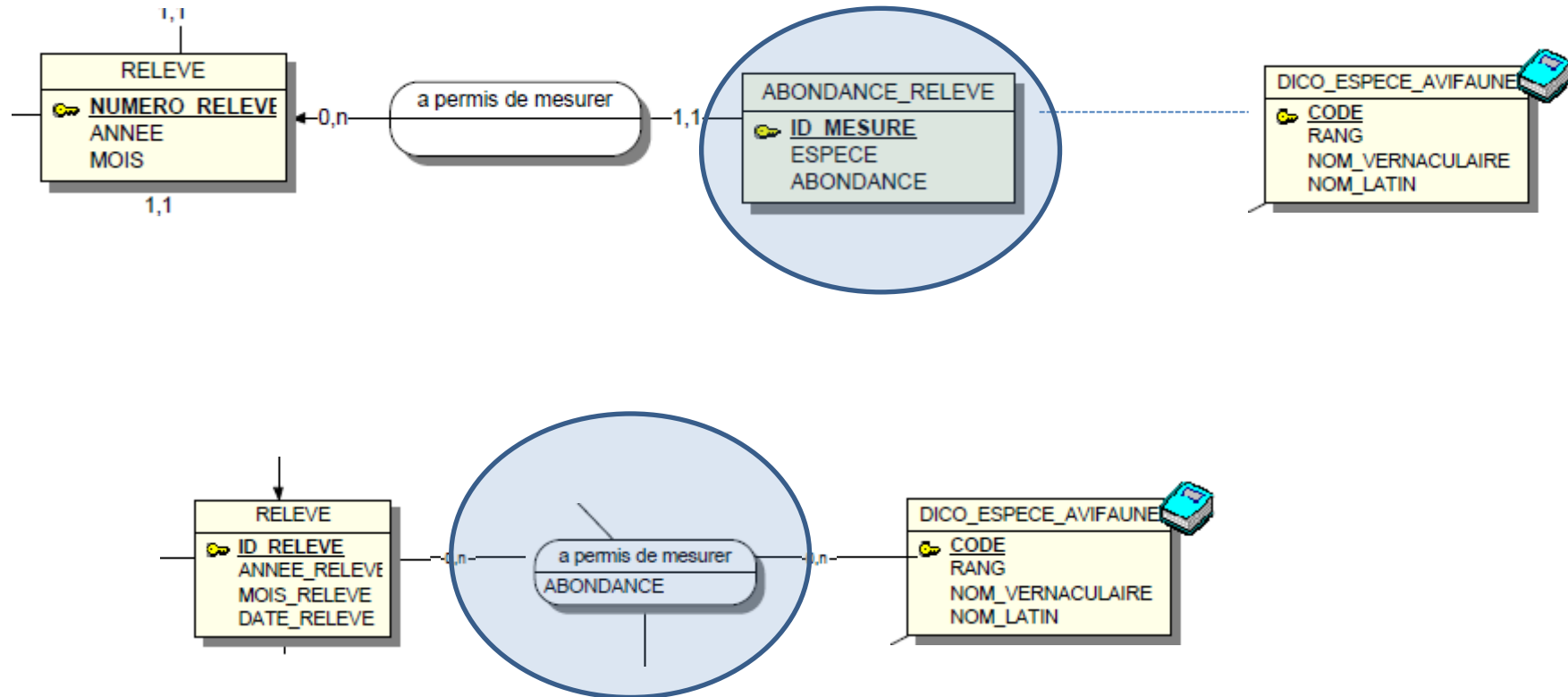
- Phase difficile qui ne doit pas apporter d'ambiguïtés
- Plein de conseils (style utiliser des noms de verbes plutôt statique, à l'infinitif, à la limite sous forme active ou passive pour orienter le sens de lecture) mais c'est plutôt au feeling



1 opération est sous la responsabilité d'1 personne  
1 personne a la responsabilité d'1 opération

- Remarque : de nombreuses associations expriment une notion d'appartenance. Il est recommandé de les qualifier en évoquant l'une des entités

# Association ? Entité ?



# Cardinalité

- Participation des occurrences d'une entité type aux occurrences d'une relation type
- Cardinalité minimum et cardinalité maximum
- Cardinalités les plus répandus : 0,n ; 1,n ; 0,1 ; 1,1
  - 0 exprime la participation optionnelle
  - 1 exprime la participation obligatoire
  - n exprime la multiplicité de participation



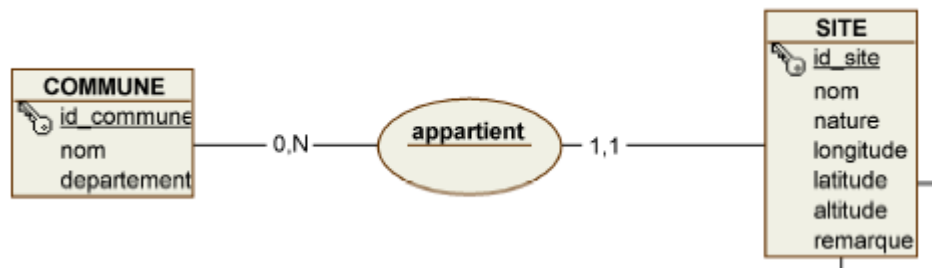
1

1 opération est sous la responsabilité d'1 ou plusieurs personnes

2

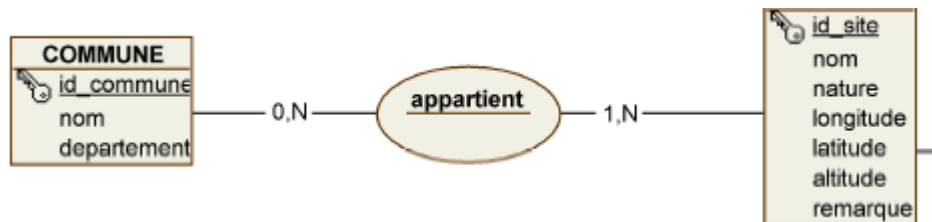
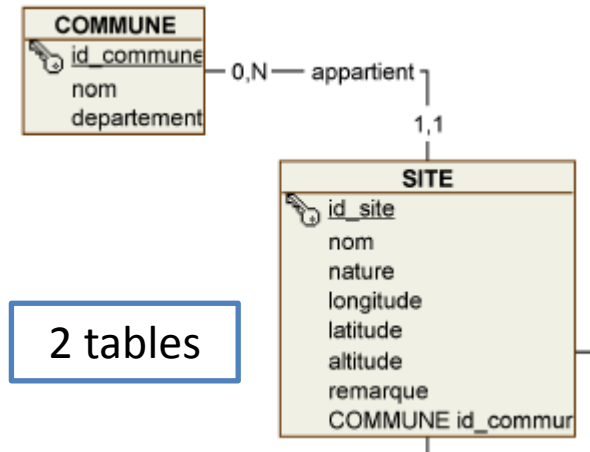
1 personne a la responsabilité d'1 ou plusieurs opérations

# Conséquences des cardinalités

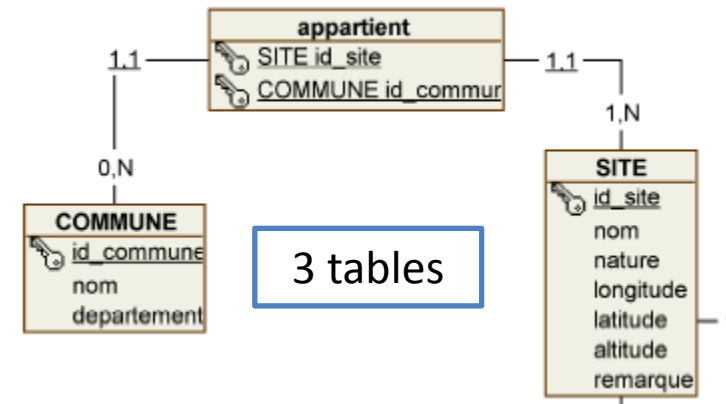


1 site est situé sur 1 seule commune  
1 commune comprend 0 ou n sites

2 tables



1 site est situé sur 1 ou + de communes  
1 commune comprend 0 ou n sites



3 tables

← analyse →

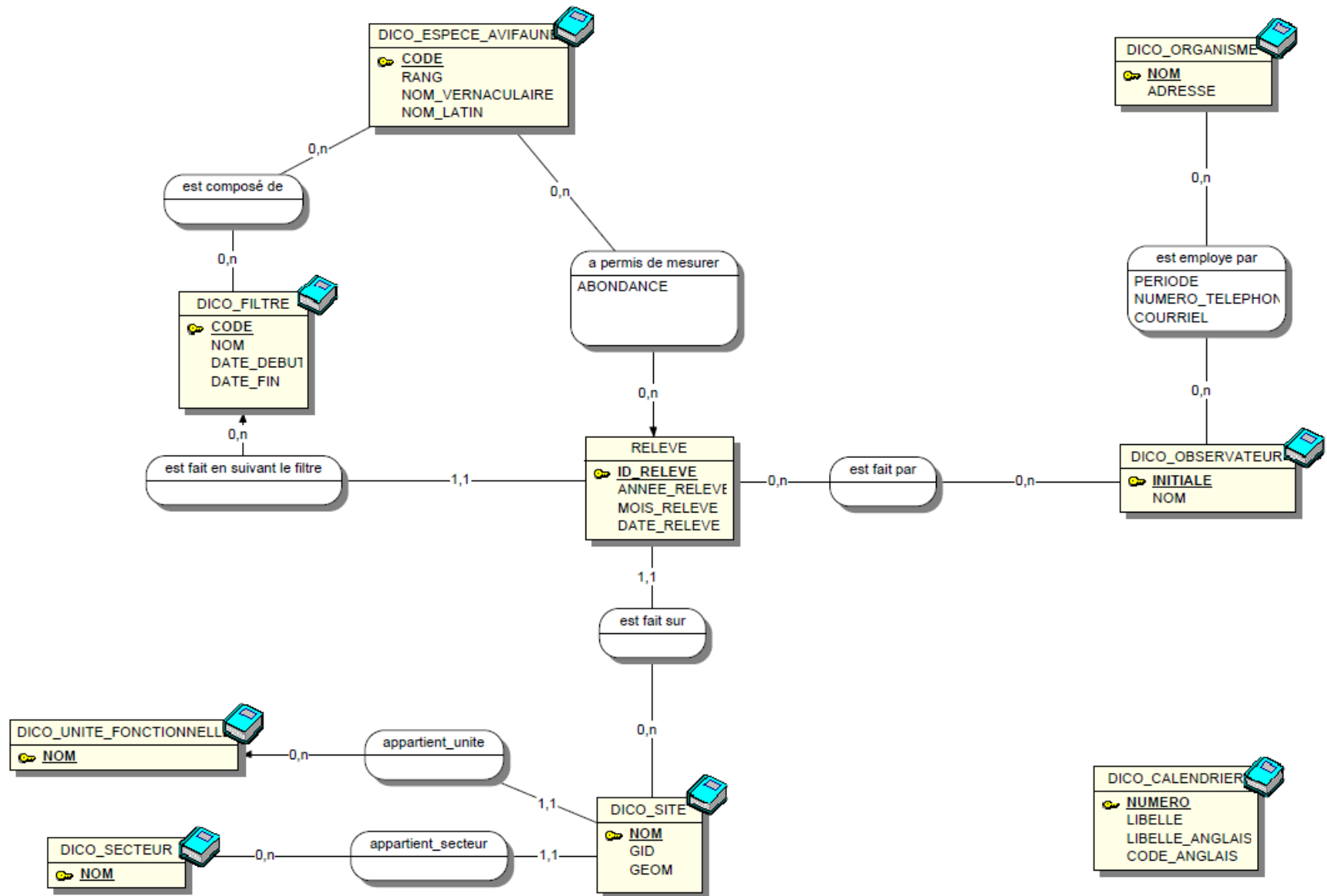
← implementation →



# En conclusion (de cette partie)

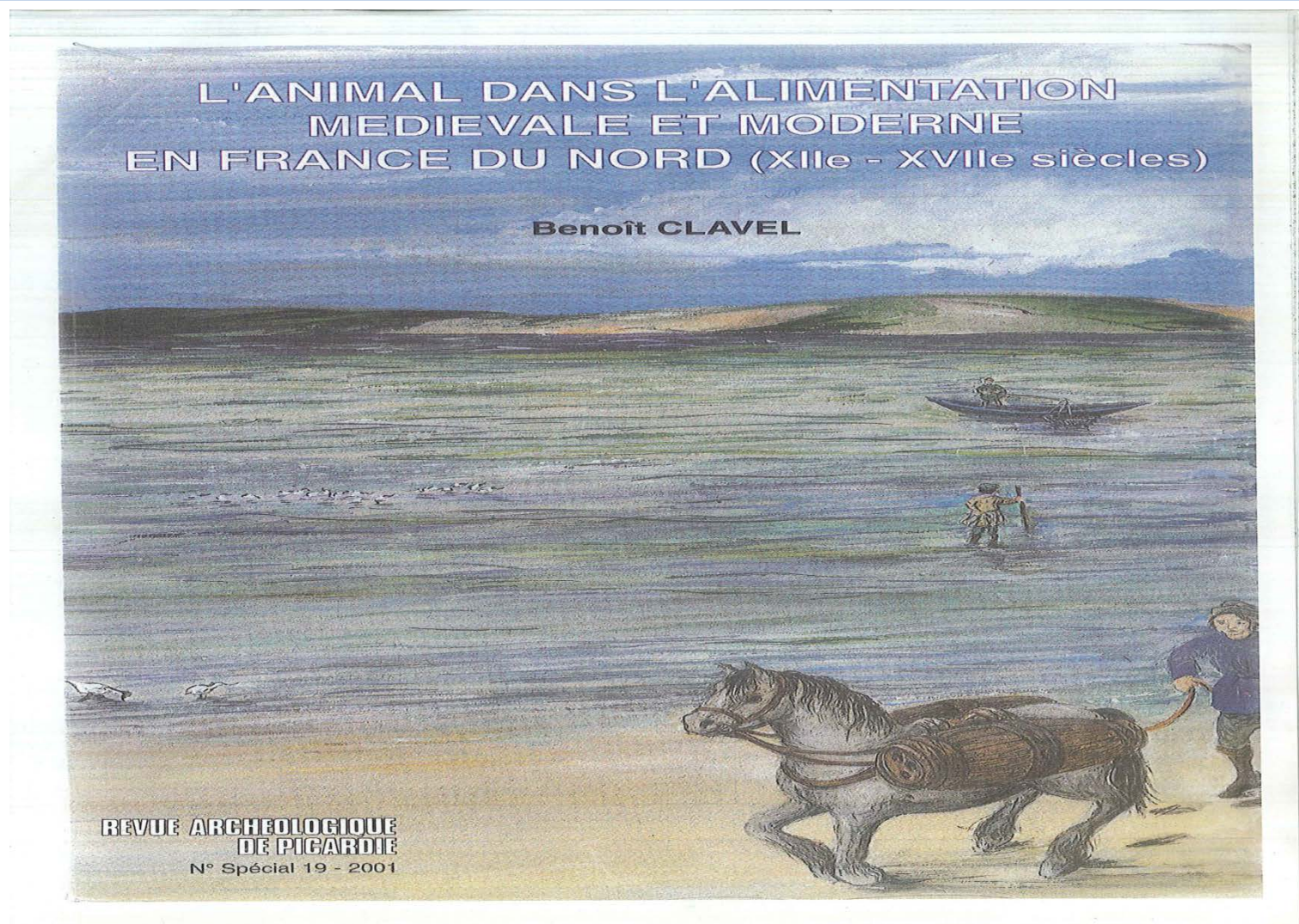
- 4 notions mais nombreuses combinaisons possibles
- Indépendant du modèle de données (relationnel, objet, ...) et du SGBD choisis
- Pour le valider, le présenter à quelqu'un extérieur
- Élément pour assurer la pérennisation des données
- Remarque : rester modeste

# Base de données AVIMED (CEFE/CNRS)



# **TP 1 : REALISER LE MODELE CONCEPTUEL DES DONNÉES**

# Présentation du cas

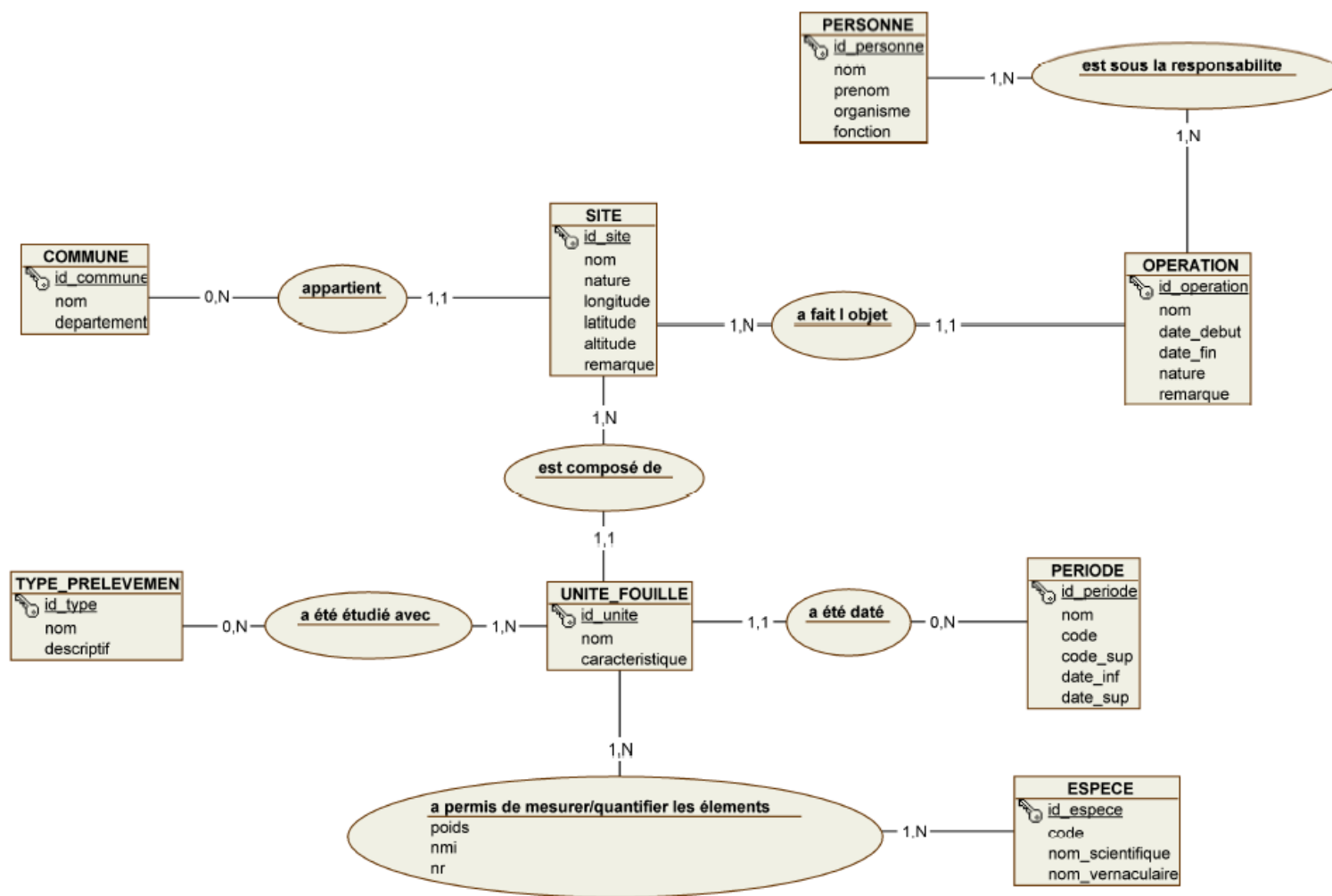


# Présentation du logiciel

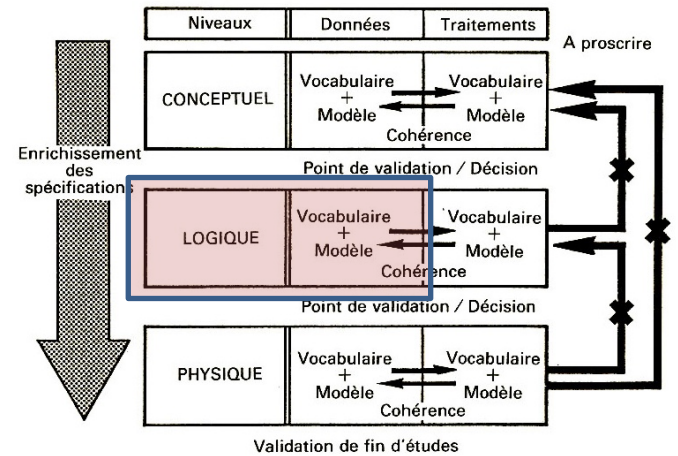
- OpenModelSphere
  - Formalisme : Entité-Association
  - Modèle : conception à la réalisation
  - Retro conception
  - Libre (GPL)
  - Windows
- Un outil à conseiller du point de vue pédagogique
  - Malgré un abord un peu difficile



# MCD proposé







*La méthode MERISE préconise d'analyser séparément données et traitements, à chaque niveau.*

# Modélisation au niveau logique

# Terme : bases de données

- Ensemble structuré et organisé permettant le stockage de grandes quantités d'informations afin d'en faciliter l'exploitation (ajout, mise à jour, recherche de données)
- Organisée selon un modèle de description de données
  - Hiérarchique, réseau, relationnel, objet, relationnel objet
- Relationnel
  - En 1970, Edgar F. Codd, chercheur chez IBM, propose dans une thèse mathématique d'utiliser les informations présentes dans les enregistrements pour assurer les liens entre les informations et de regrouper les enregistrements dans des tables
  - Modèle le plus courant
    - Présentation simple des données sous forme de tables
    - Puissance et simplicité du langage de requête



# Modèle logique de données (MLD)

- Représentation (graphique ou non) de la structure de la base de données selon le modèle de description des données retenu (relationnel dans notre cas)
- Traduction du MCD en MLD (ou MRD ou MLR) en appliquant des règles de transformation
- Remarque : MLD est indépendant du SGBD retenu

# Concepts du modèle relationnel (1)

Identificateur

Type

Colonne  
=  
Attribut

Contrainte de table  
[PK] : Primary Key

Ligne =  
Enregistrement  
Tuple ou n-  
uplet

	numero_releve [PK] integer	point_geographique character(50)	date_releve date	heure_debut time without time zone	nuage smallint	vent smallint	pluie smallint
1	1	PicSaintLoup-1	2003-06-06	07:38:00	1	0	0
2	2	PicSaintLoup-10	2003-04-30	08:01:00	4	70	1
3	3	PicSaintLoup-100	2003-05-07	10:04:00	1	20	0
4	4	PicSaintLoup-101	2003-05-07	06:55:00	3	50	1
5	5	PicSaintLoup-102	2003-05-07	07:28:00	2	30	0
6	6	PicSaintLoup-103	2003-05-07	11:34:00	3	30	0
7	7	PicSaintLoup-104	2003-06-05	10:11:00	1	20	0
8	8	PicSaintLoup-105	2003-05-01	09:18:00	0	10	0
9	9	PicSaintLoup-106	2003-05-01	08:01:00	0	0	0
10	10	PicSaintLoup-107	2003-05-01	07:28:00	0	30	0

1502 lignes.

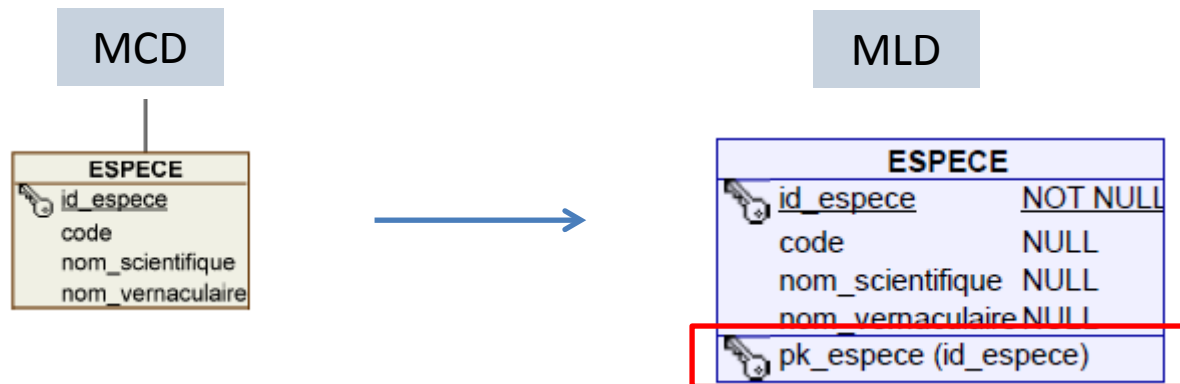
## Concepts du modèle relationnel (2)

- Clef primaire sert à identifier une ligne de manière unique
- Clef étrangère permet
  - de gérer les relations entre plusieurs tables
  - de garantir la cohérence des données (intégrité référentielle)

# Règle 1 : Transformation des entités



Toute entité est transformée en table. Les propriétés de l'entité deviennent les attributs de la table. L'identifiant de l'entité devient la clé primaire de la table.







La clef primaire s'appelle pk\_espece



Elle fait référence à l'attribut « id\_espece » de la table « ESPECE »



# Première étape de la transformation



- 8 tables qui correspondent aux 8 entités



COMMUNE	
 <u>id_commune</u> NOT NULL	
nom	NULL
departement	NULL
 pk_commune(id_commune)	



SITE	
 <u>id_site</u> NOT NULL	
nom	NULL
nature	NULL
longitude	NULL
latitude	NULL
altitude	NULL
remarque	NULL
 pk_site(id_site)	



PERSONNE	
 <u>id_personne</u> NOT NULL	
nom	NULL
prenom	NULL
organisme	NULL
fonction	NULL
 pk_personne(id_personne)	

OPERATION	
 <u>id_operation</u> NOT NULL	
nom	NULL
date_debut	NULL
date_fin	NULL
nature	NULL
remarque	NULL
 pk_operation(id_operation)	

TYPE_PRELEVEMENT	
 <u>id_type</u> NOT NULL	
nom	NULL
descriptif	NULL
 pk_type(id_type)	

UNITE_FOUILLE	
 <u>id_unite</u> NOT NULL	
nom	NULL
caracteristique	NULL
 pk_unite(id_unite)	

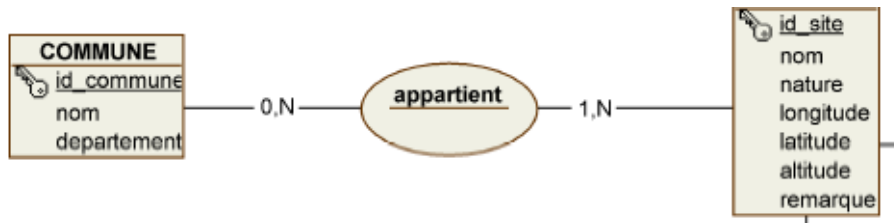
PERIODE	
 <u>id_période</u> NOT NULL	
nom	NULL
code	NULL
code_sup	NULL
date_inf	NULL
date_sup	NULL
 pk_période(id_période)	

ESPECE	
 <u>id_espece</u> NOT NULL	
code	NULL
nom_scientifique	NULL
nom_vernaculaire	NULL
 pk_espece(id_espece)	

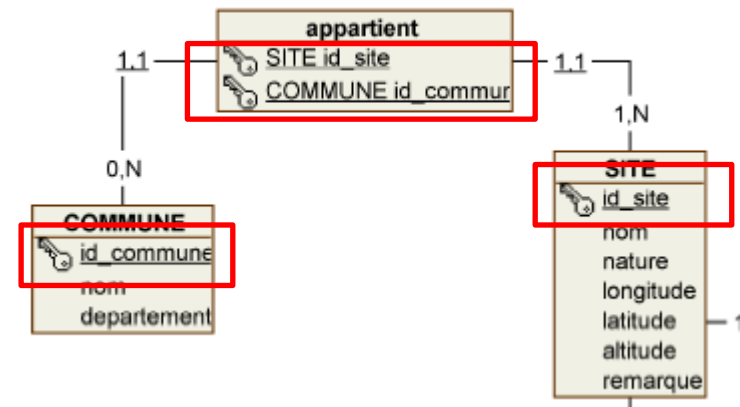
# Règle 2 : Transformation des relations binaires du type $(x,n) - (x,n)$ avec $x = 0$ ou $1$

⚠ On crée une table supplémentaire ayant comme clé primaire une clé composée des clés primaires des 2 tables. Lorsque la relation contient elle-même des propriétés, celles-ci deviennent attributs de la table supplémentaire. Une propriété de la relation qui est soulignée devra appartenir à la clé primaire composée de la table supplémentaire.

MCD







MLD







# Deuxième étape de la transformation



- 3 tables en plus qui correspondent aux 3 relations binaires du type (x,n) – (x,n)



a été étudié avec	
	UNITE_FOUILLE id_unite
	TYPE_PRELEVEMENT id_type



a permis de mesurer/quantifier les éléments	
	poids
	nmi
	nr
	ESPECE id_espece
	UNITE_FOUILLE id_unite



est sous la responsabilité	
	OPERATION id_operation
	PERSONNE id_personne



COMMUNE	
	id_commune NOT NULL
	nom NULL
	departement NULL
	pk_commune(id_commune)



SITE	
	id_site NOT NULL
	nom NULL
	nature NULL
	longitude NULL
	latitude NULL
	altitude NULL
	remarque NULL
	pk_site(id_site)



PERSONNE	
	id_personne NOT NULL
	nom NULL
	prenom NULL
	organisme NULL
	fonction NULL
	pk_personne(id_personne)

OPERATION	
	id_operation NOT NULL
	nom NULL
	date_debut NULL
	date_fin NULL
	nature NULL
	remarque NULL
	pk_operation(id_operation)

TYPE_PRELEVEMENT	
	id_type NOT NULL
	nom NULL
	descriptif NULL
	pk_type(id_type)

UNITE_FOUILLE	
	id_unite NOT NULL
	nom NULL
	caracteristique NULL
	pk_unite(id_unite)

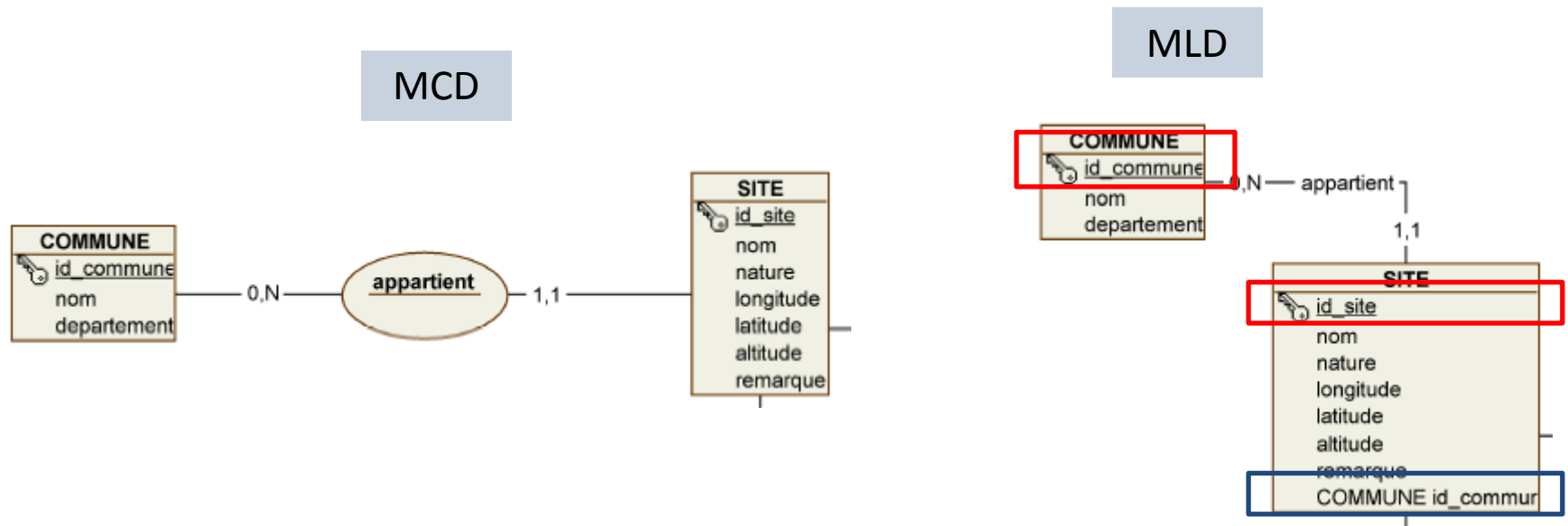
PERIODE	
	id_pperiode NOT NULL
	nom NULL
	code NULL
	code_sup NULL
	date_inf NULL
	date_sup NULL
	pk_pperiode(id_pperiode)

ESPECE	
	id_espece NOT NULL
	code NULL
	nom_scientifique NULL
	nom_vermaculaire NULL
	pk_espece(id_espece)

E  
t  
a  
p  
e  
  
1

# Règle 3 : Transformation des relations binaires du type $(x,n) - (x,1)$ avec $x = 0$ ou $1$

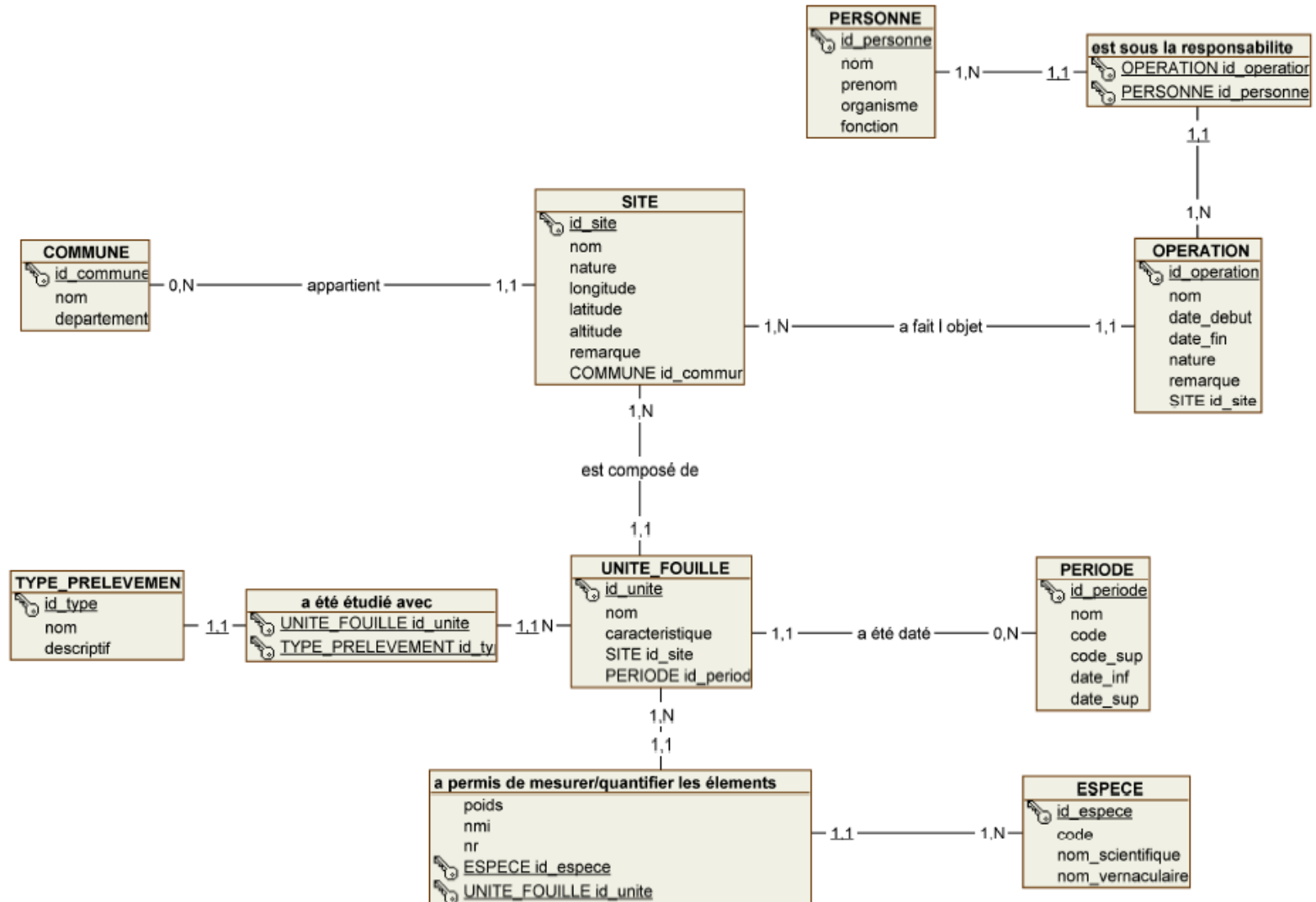
**!** Afin de représenter la relation, on duplique la clé primaire de la table basée sur l'entité à cardinalité  $(x,n)$  dans la table basée sur l'entité à cardinalité  $(x,1)$ . Cet attribut est appelé clé étrangère. Les deux tables sont liées par une flèche nommée selon la relation, qui pointe de la table à clé étrangère vers la table qui contient la clé primaire correspondante.



L'attribut « id\_commune » qui est clef primaire de la table COMMUNE devient clef étrangère dans la table SITE






# Troisième étape de la transformation






# Remarque importante

- ✓ Présentation simplifiée des règles de transformation
- ✓ Il faut ajouter la notion de clef étrangère dans la règle 2 (relations binaires du type  $(x,n) - (x,n)$ ) pour représenter la relation entre les deux tables.

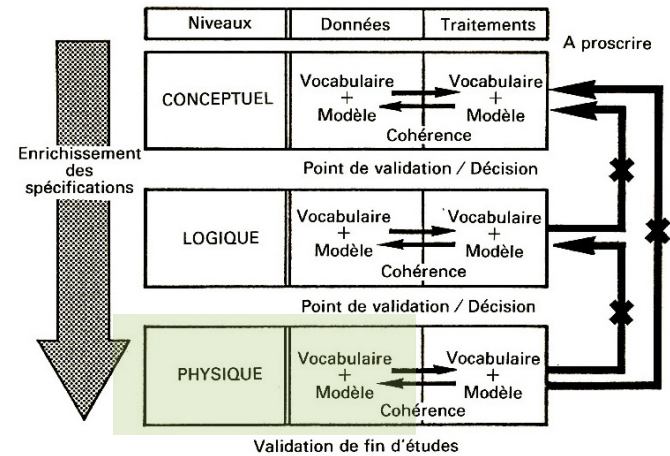
 On crée une table supplémentaire ayant comme clé primaire une clé composée des clés primaires des 2 tables. Lorsque la relation contient elle-même des propriétés, celles-ci deviennent attributs de la table supplémentaire. Une propriété de la relation qui est soulignée devra appartenir à la clé primaire composée de la table supplémentaire.

est sous la responsabilité	
	<u>OPERATION id_operation</u>
	<u>PERSONNE id_personne</u>

Règle2

est sous la responsabilité	
	FK1 <u>OPERATION id_operation</u> NOT NULL
	FK2 <u>PERSONNE id_personne</u> NOT NULL
	Clé primaire / unique(OPERATION id_operation, PERSONNE id_personne)
	FK1 Clé étrangère (OPERATION id_operation)
	FK2 Clé étrangère (PERSONNE id_personne)

La table complète



*La méthode MERISE préconise d'analyser séparément données et traitements, à chaque niveau.*

# Modélisation au niveau physique

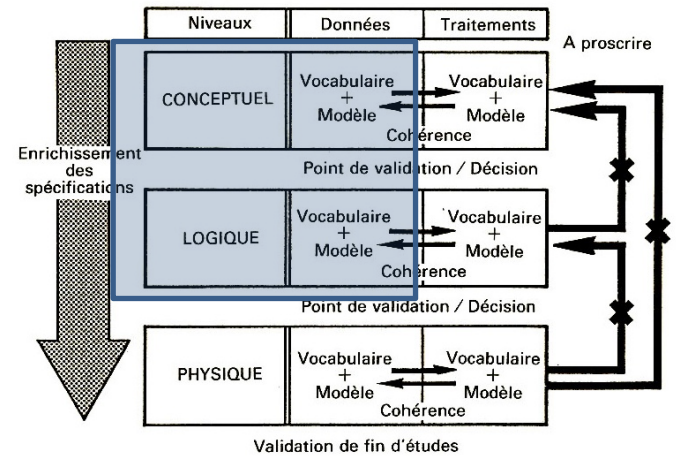
# Modèle physique de données (MPD)

- Génération du script SQL adapté aux spécificités du SGBD retenu
  - Commandes
  - Types de données
- Cette phase est intéressante uniquement si :
  - Vous utilisez un logiciel de modélisation
  - Toutes les propriétés des entités ont été définies dans la réalisation du modèle conceptuel de données

# Quelques références

- Règles de transformation du MCD en MLD  
<http://steveostine.free.fr/pdf/Rgles%20de%20transformation%20du%20MCD%20au%20MLD.pdf>
- Initiation à la conception de base de données relationnelles avec MERISE  
<http://ineumann.developpez.com/tutoriels/merise/initiation-merise/>
- Conception d'une base de données  
<http://cyril-gruau.developpez.com/uml/tutoriel/ConceptionBD/>
- Logiciel OpenModelSphere  
[http://www.modelsphere.org/fr/open\\_modelsphere.html](http://www.modelsphere.org/fr/open_modelsphere.html)



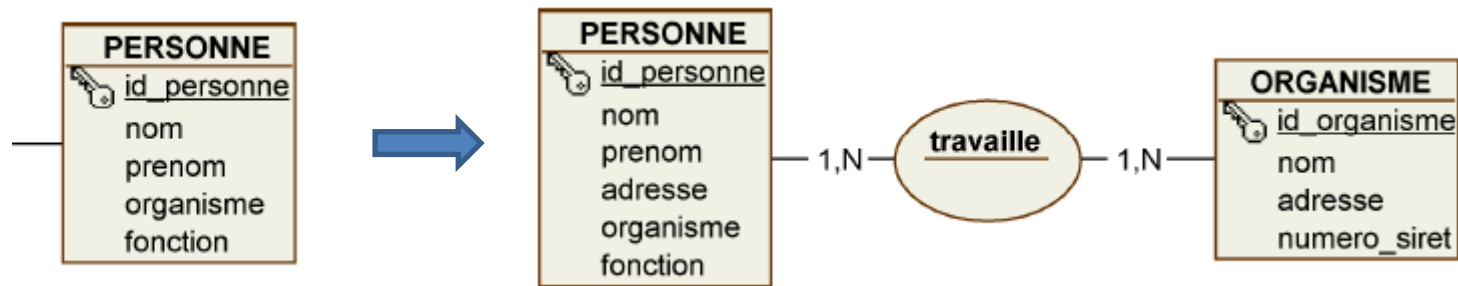


*La méthode MERISE préconise d'analyser séparément données et traitements, à chaque niveau.*

# Règles de normalisation

# Première forme normale

- A un instant donné dans une entité, pour un individu, un attribut ne peut prendre qu'une valeur et non pas, un ensemble ou une liste de valeurs. Si un attribut prend plusieurs valeurs, alors ces valeurs doivent faire l'objet d'une entité supplémentaire, en association avec la première.

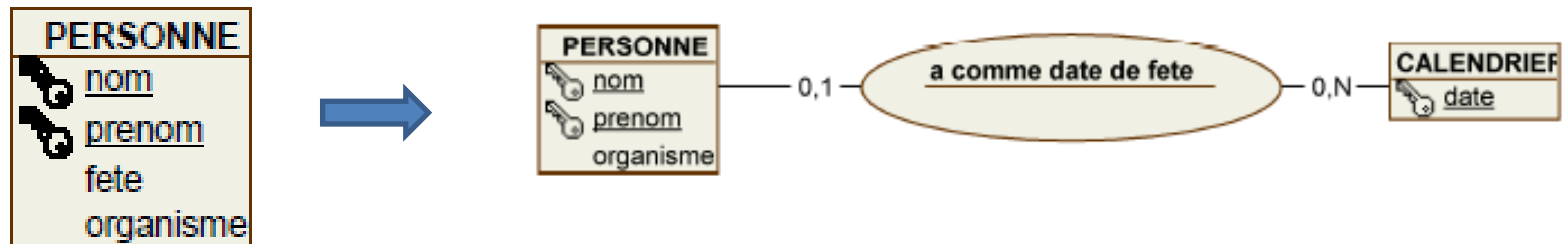


Ex : une personne travaille dans deux organismes



# Deuxième forme normale

- L'identifiant peut être composé de plusieurs attributs mais les autres attributs de l'entité doivent dépendre de l'identifiant en entier (et non pas une partie de cet identifiant).

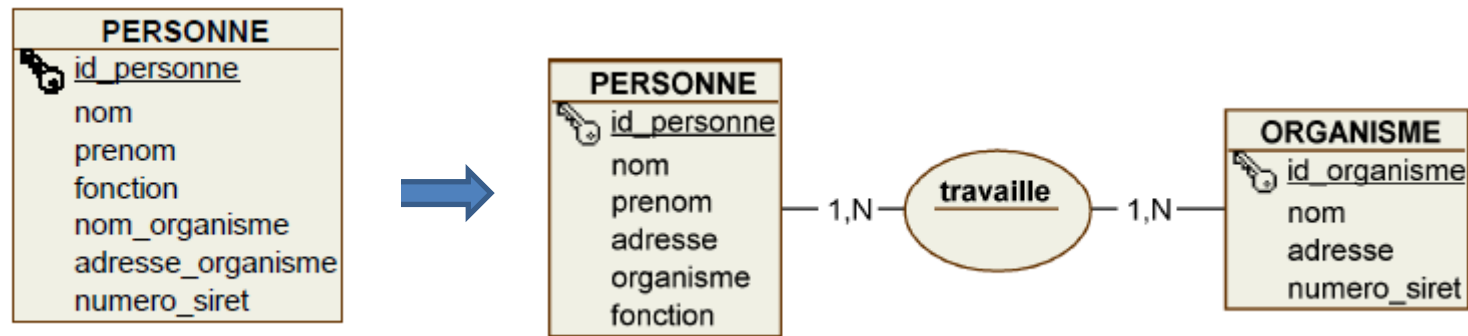


fete ne dépend que d'une partie de la clef primaire composée (prenom)

- Remarque : cette forme normale peut être oubliée si on n'utilise que des identifiants non composés et de type entier

# Troisième forme normale

- Tous les attributs d'une entité doivent dépendre directement de son identifiant et d'aucun autre attribut. Si ce n'est pas le cas, il faut placer l'attribut pathologique dans une entité séparée, mais en association avec la première.



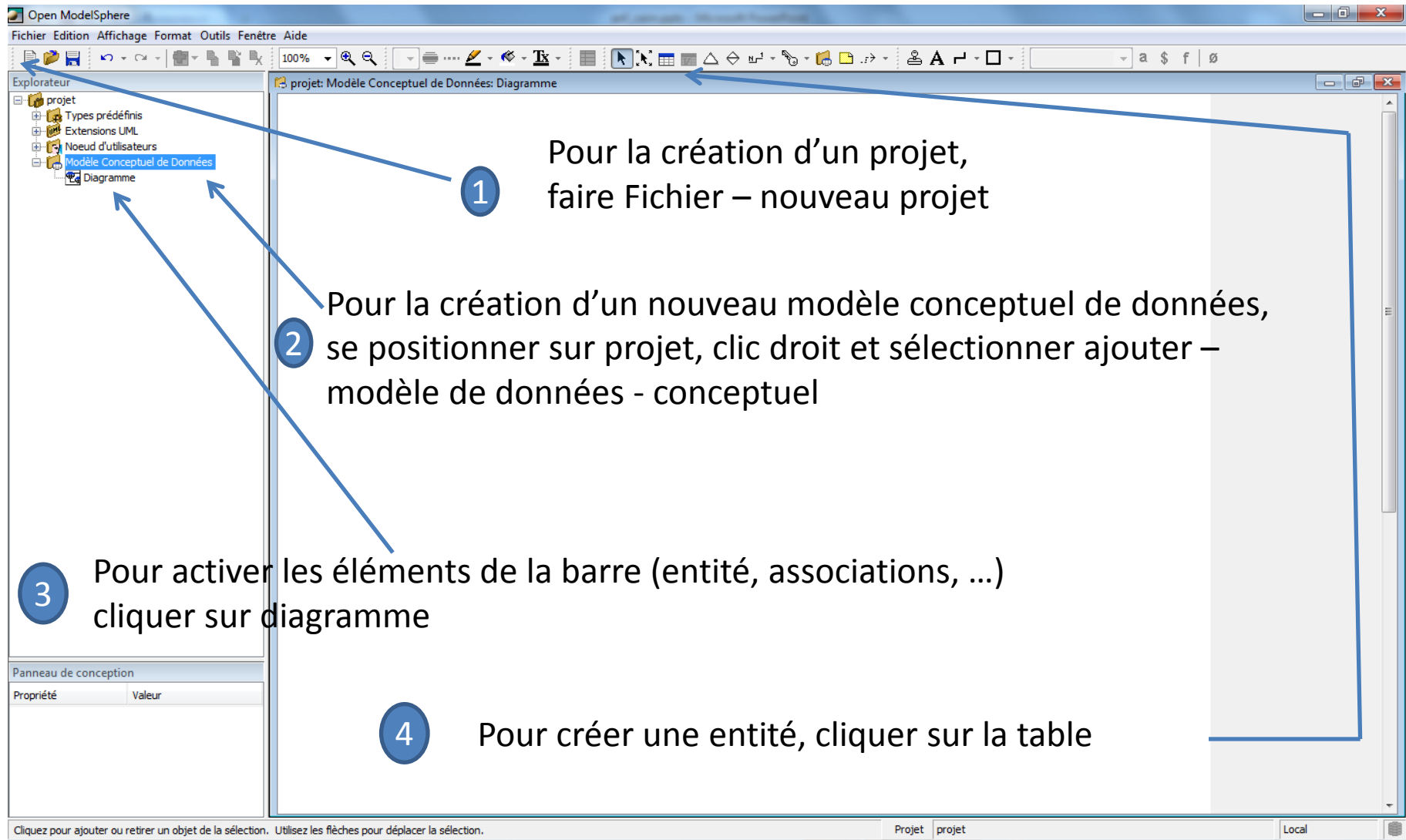
adresse\_organisme et numero\_siret ne dépendent pas directement de id\_personne

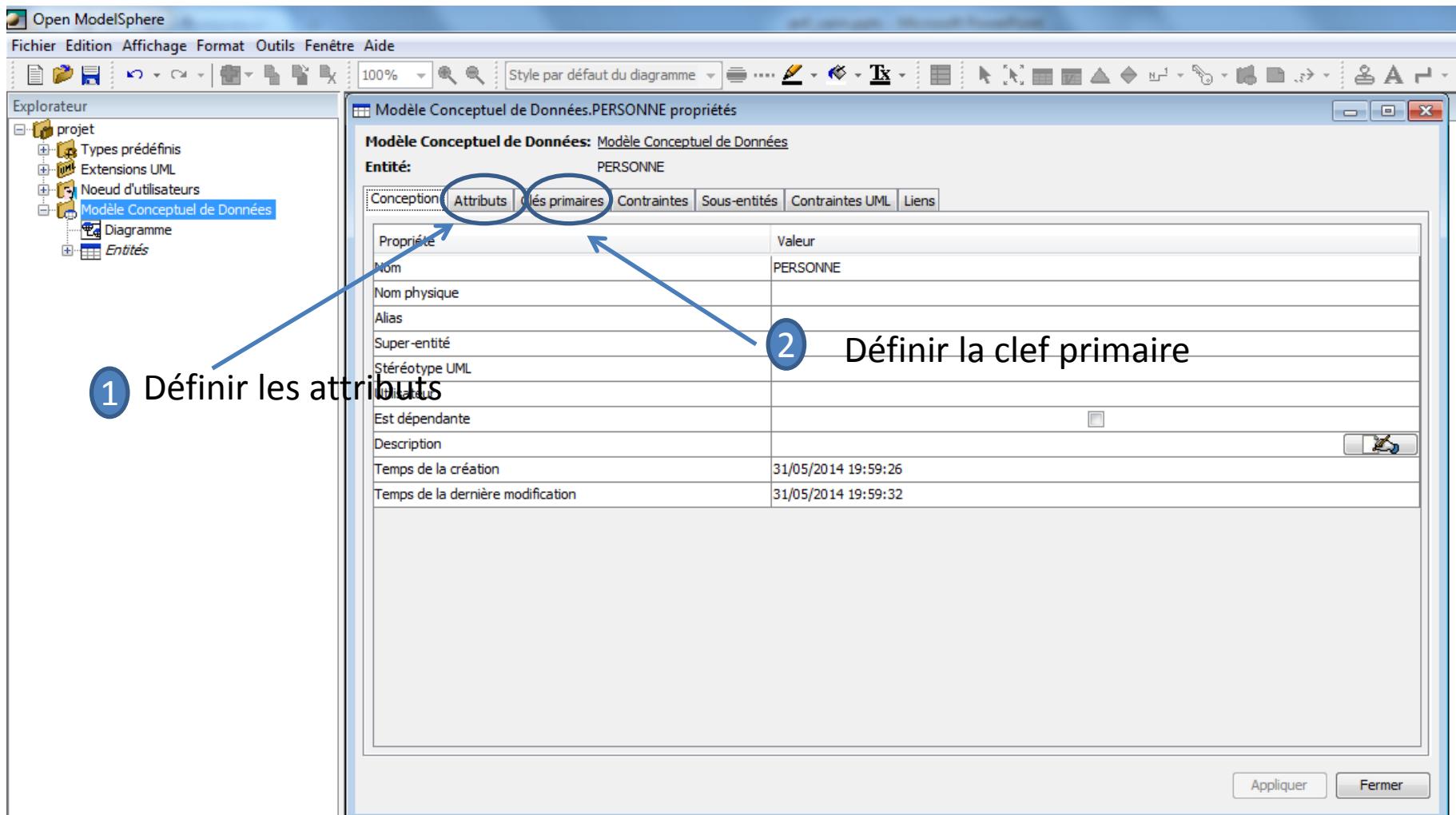
# Les autres formes normales

- Les quatrièmes, cinquièmes et sixièmes formes normales évitent principalement la redondance d'information, elles sont plus précises
- En pratique la forme normale 3 est suffisante
- Les projections et les jointures sont coûteuses pour le système, ainsi une trop forte normalisation diminue fortement les performances



# OpenModelSphere





On obtient cet écran en sélectionnant *la table*, clic droit et propriétés

Open ModelSphere

Fichier Edition Affichage Format Outils Fenêtre Aide

Explorateur

- projet
- Types prédéfinis
- Extensions UML
- Noeud d'utilisateurs
- Modèle Conceptuel de Données
- Diagramme
- Entités
- DONNEES

Modèle Conceptuel de Données.DONNEES propriétés

Entité: DONNEES

Conception Attributs Clés primaires Contraintes Sous-entités Contraintes UML Liens

Nom	Nom physique	Alias	Super-clé	Stéréotype UML	Prim...	Descrip...	Temps de la créa...	Temps de la dernière modifica...
pk_donnees					<input checked="" type="checkbox"/>		27/05/2014 15:11:...	27/05/2014 15:11:58

1 Définir le nom de la clef

2 Lier le nom à un attribut

Modèle Conceptuel de Données.DONNEES.pk\_donnees propriétés

Entité: DONNEES

Clé primaire: pk\_donnees

Conception Dépendances Attribut Sous-clé Contraintes UML Liens

Nom

Lier Délier Propriétés Monter Descendre

Ajouter Appliquer Fermer

Open ModelSphere

Fichier Edition Affichage Format Outils Fenêtre Aide

100% Style par défaut du diagramme

Explorateur

- projet
  - Types prédéfinis
  - Extensions UML
  - Noeud d'utilisateurs
  - Modèle Conceptuel de Données
    - Diagramme
    - Entités
    - Associations
    - Arcs

anf-regle3.sms: Modèle Conceptuel de Données: Diagramme

1 Pour créer association, cliquer sur le symbole 1-n

3 Pour changer les cardinalités, cliquer sur 0,N

2 Donner un nom à l'association

Modèle Conceptuel de Données: Association propriétés

Modèle Conceptuel de Données: Modèle Conceptuel de Données

Association:

Conception Attributs Contraintes Sous-association Contraintes UML Liens

Propriété	Valeur
Nom	Association
Nom physique	
Alias	
Super-association	
Stéréotype UML	
Utilisateur	
Est dépendant	
Description	
Temps de la création	31/05/2014 21:05:20
Temps de la dernière modification	31/05/2014 21:05:20

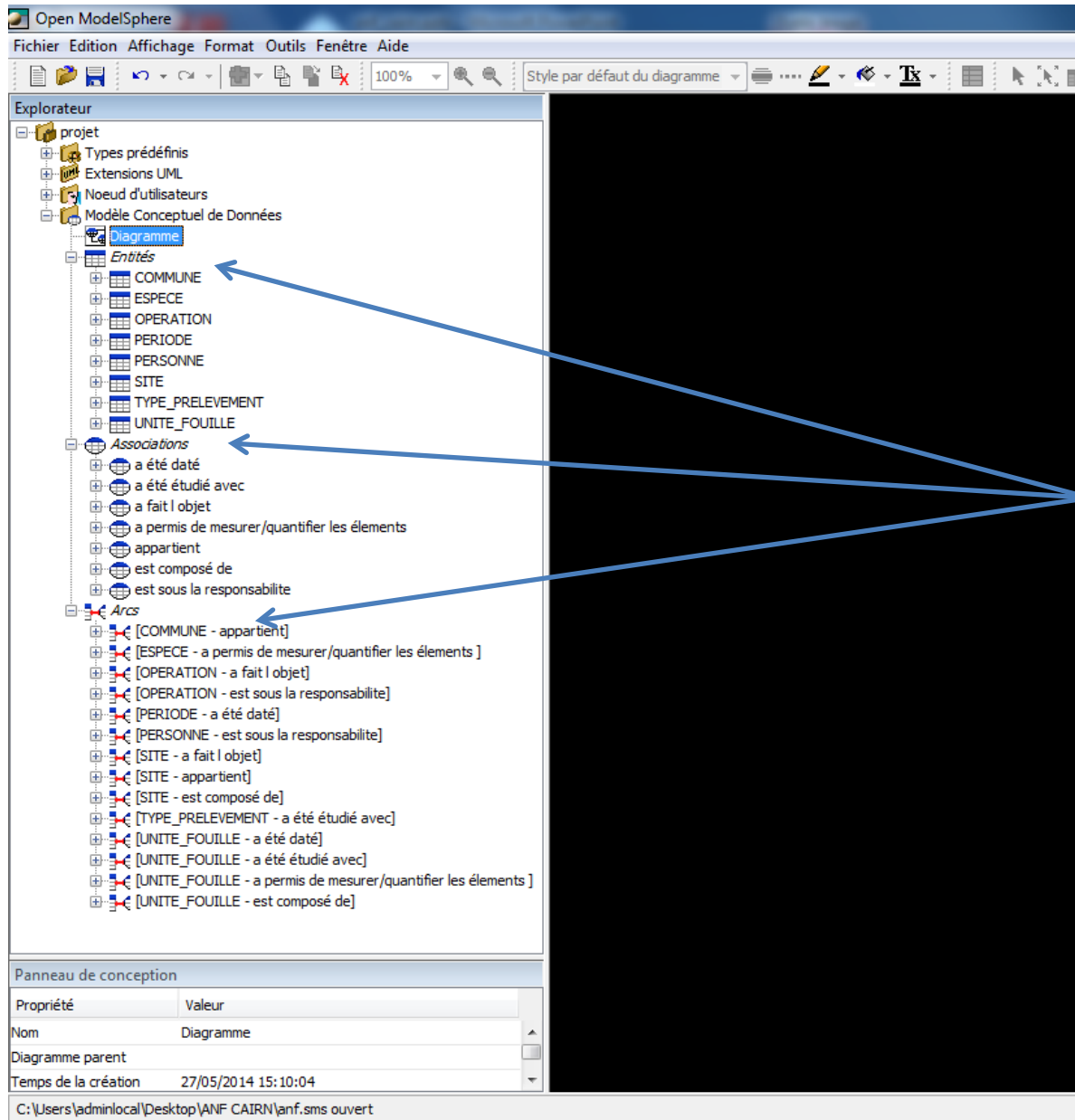
Panneau de conception

Propriété	Valeur

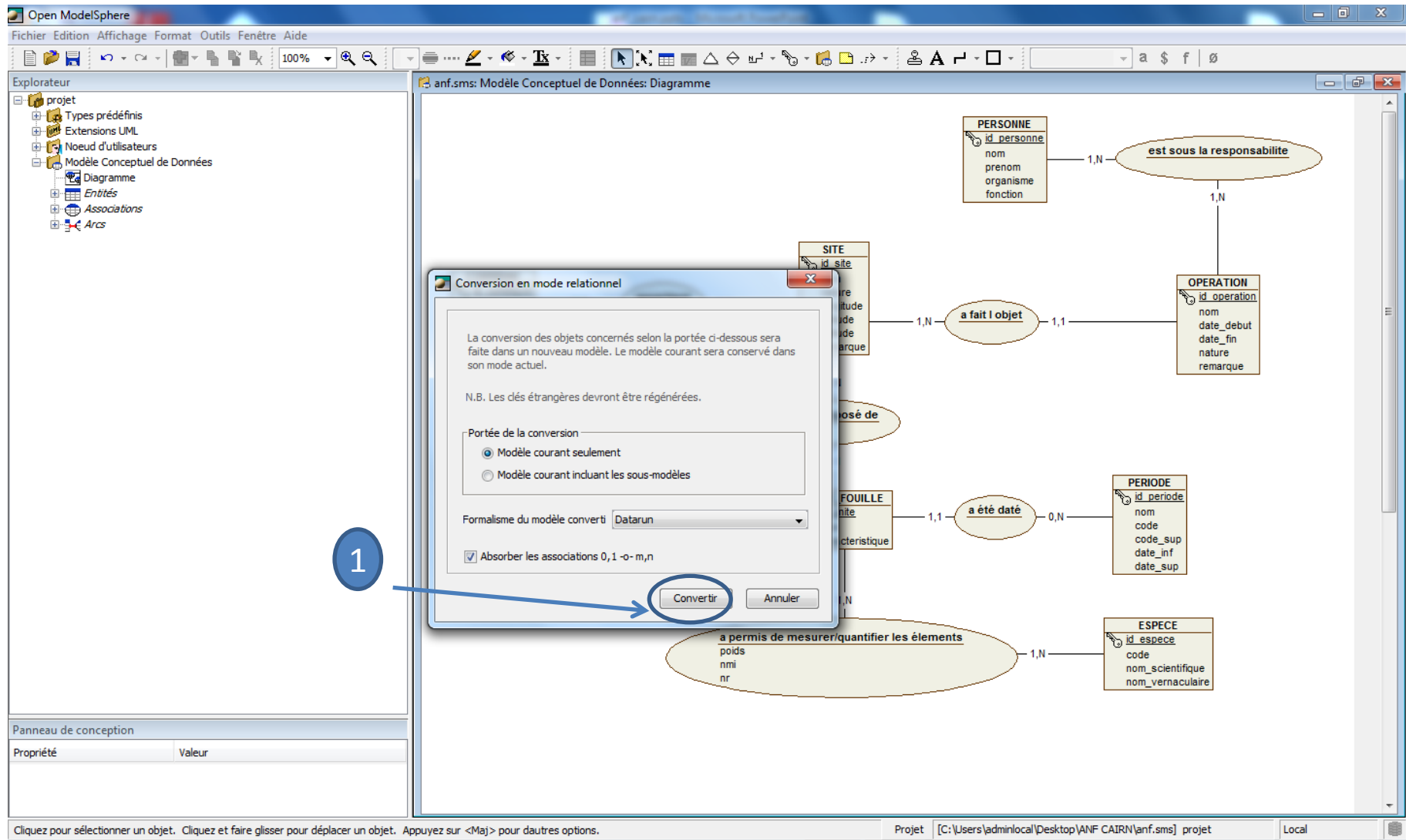
Projet [C:\Users\adminlocal\Desktop\ANF CAIRN\anf-regle3.sms] projet Local

On obtient cet écran en sélectionnant *l'association*, clic droit et propriétés

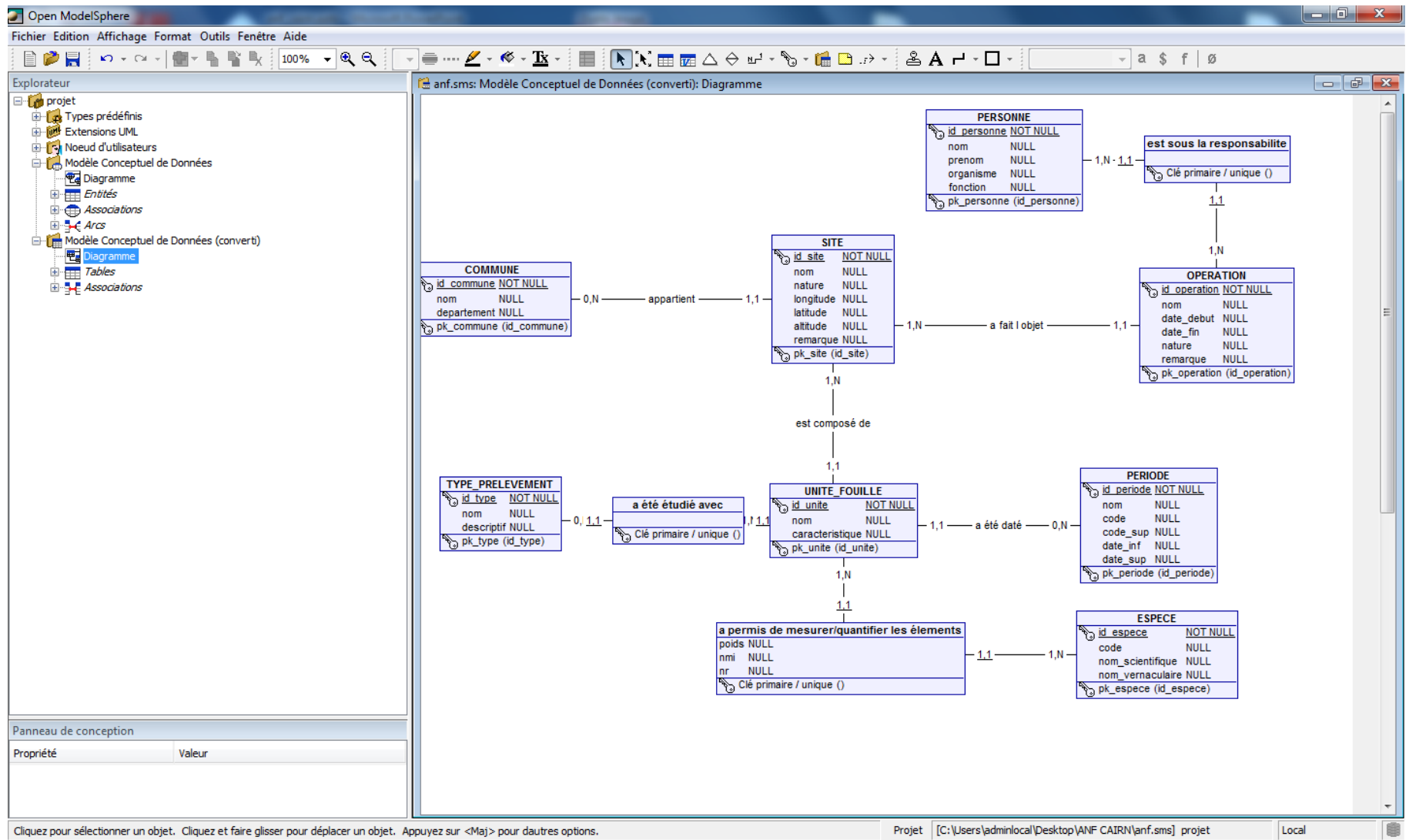




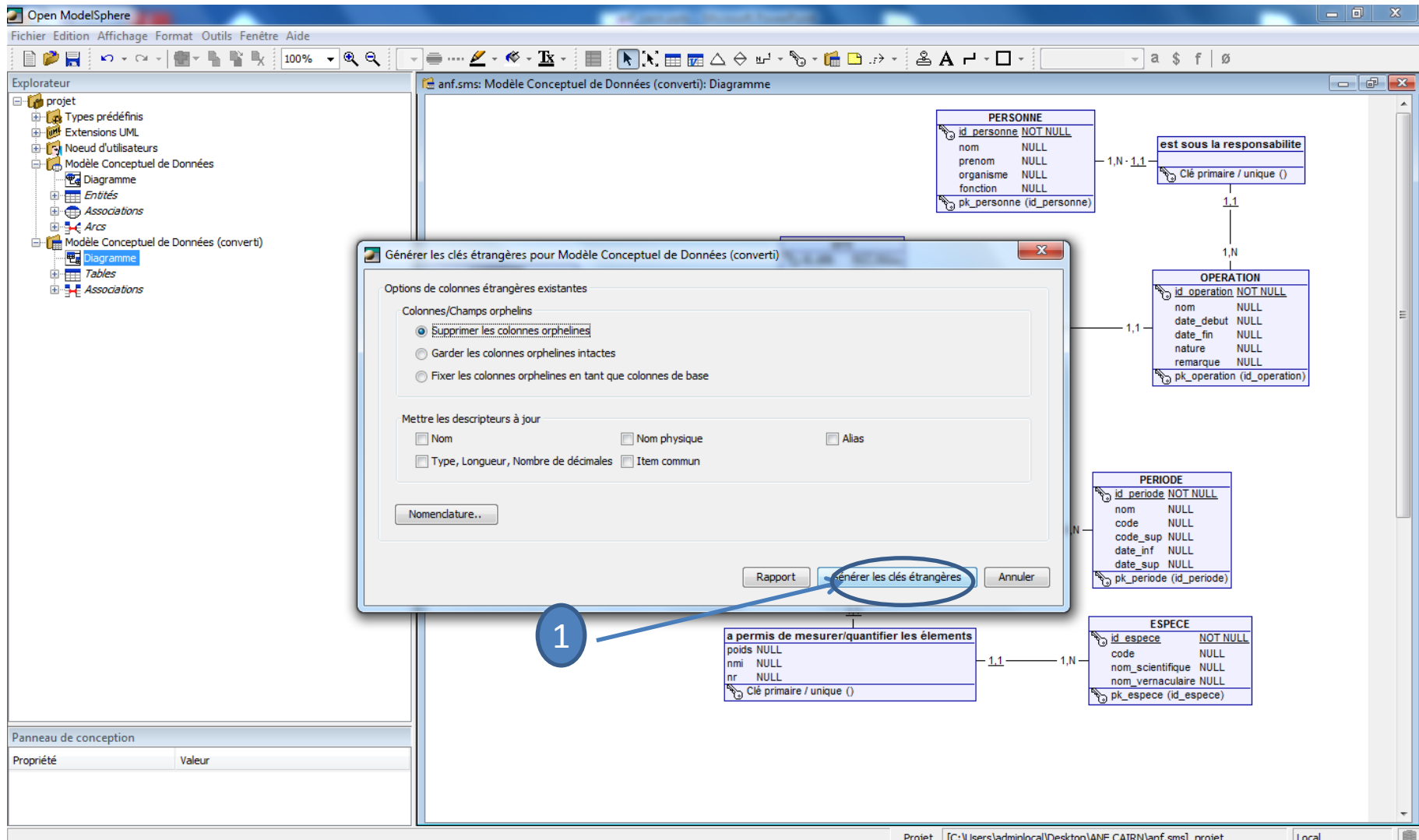
Les éléments  
générés



On obtient cet écran en sélectionnant outil - modèle de données – convertir en mode relationnel



Le modèle logique de données créé ne contient pas de clés étrangères



On obtient cet écran en sélectionnant outil - modèle de données –  
générer les clefs étrangères

