

BULLETIN D'INFORMATIQUE APPROFONDIE ET APPLICATIONS

COMPUTATION - INFORMATION

N° 78 – DECEMBRE 2007

DIRECTEUR

Jean - Michel Knippel

FONDATEUR

Edmond Bianco

SERVEUR DE PUBLICATION

Christian Blanvillain

SECRETARIAT

Kalassoumi Adjilani

Université de Provence
Equipe Hermès. Case 33
3 place Victor Hugo
F - 13331 Marseille Cedex 3
Téléphone: (0)4 91 10 62 30
Télécopie : (0)4 91 50 91 10

DEPOSITAIRE

Université de Provence
Bibliothèque Universitaire
1 place Victor Hugo
F - 13331 Marseille Cedex 3
Téléphone : (0)4 91 10 85 29
Télécopie : (0)4 91 95 75 57

IMPRIMEUR

Université de Provence
Service Reprographie
3 place Victor Hugo
F - 13331 Marseille Cedex 3
Téléphone : (0)4 91 10 60 48

1 EDITORIAL
Motion du CNU 27

par les membres du CNU 27 (informatique)

**3 Conception et normalisation des bases
de données en Sciences Humaines et Sociales**

par Jean – Michel Knippel

**5 Introduction aux bases de données
Conception et normalisation**

par Marc Grange

27 VOZZAVEDIBISAR
Turlubec

par Edmond Bianco

<http://www.univ-provence.fr/biaa>

Publication trimestrielle, gratuite, de l'Université de Provence

Impression : avril 2008

ISSN 0291 - 5413



EDITORIAL

Motion du CNU 27

Les membres du CNU 27 (informatique)

Motion LRU (19/11/2007)

Les membres du CNU 27 (informatique) réunis ce jour pour l'élection du bureau, expriment leur désaccord avec de nombreux points de la loi LRU, en particulier :

- * le remplacement des commissions de spécialistes par de comités nommés par l'université sans garantie de représentativité, ni de légitimité scientifique,

- * l'attribution des PEDR au niveau local,

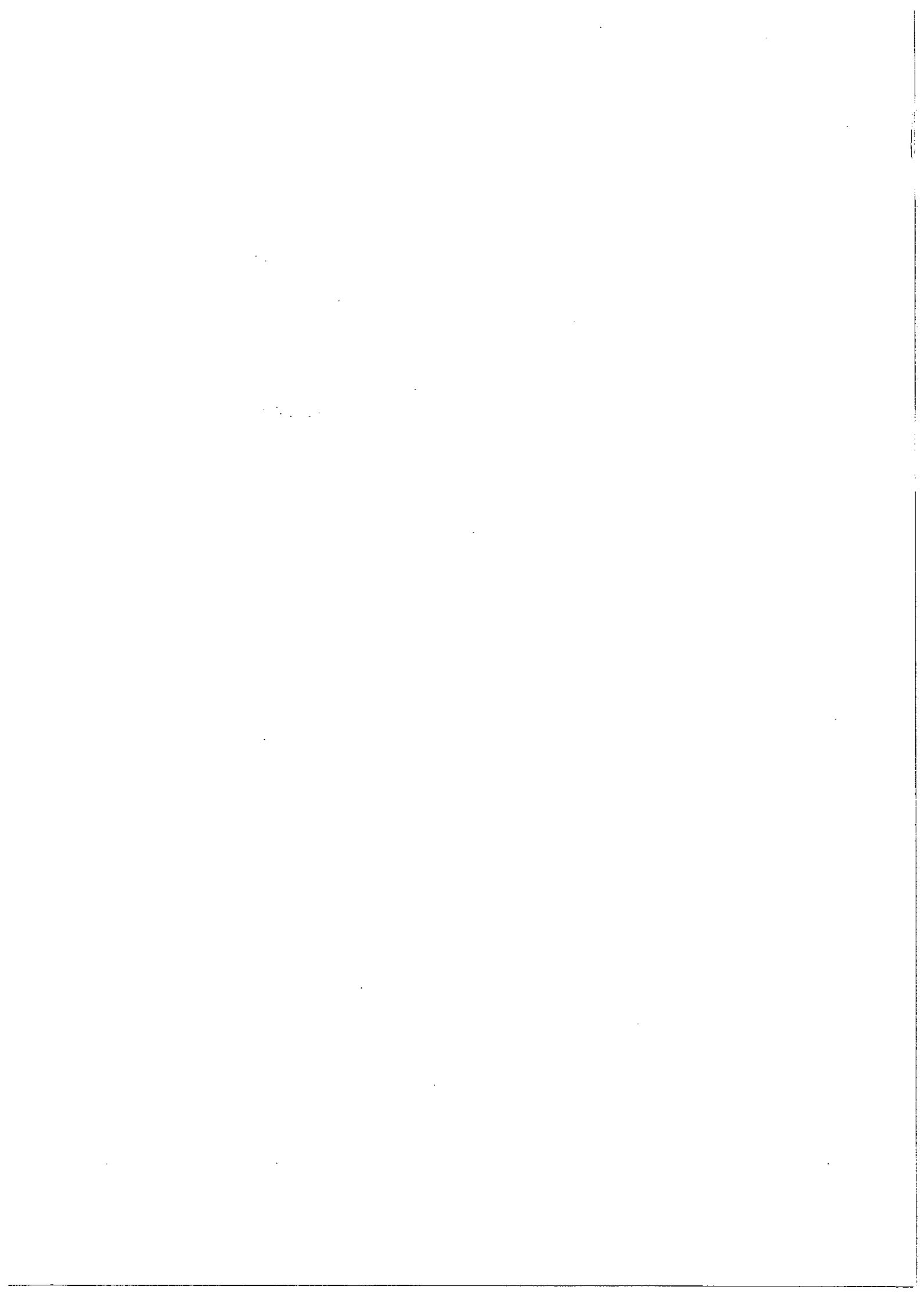
- * la remise en cause du statut des enseignants-chercheurs, notamment,

- o le recrutement de contractuels non nécessairement qualifiés,

- o le mode de gestion des carrières essentiellement local (promotions, modulation de services),

et demandent l'ouverture de négociations associant l'ensemble de la communauté universitaire pour aboutir à une loi renforçant le service public d'enseignement supérieur et de recherche.

Motion adoptée : 34 pour, 2 abstentions et 1 nul.



Conception et normalisation des bases de données en Sciences Humaines et Sociales

Jean - Michel Knippel

Lorsque la réforme L.M.D. arriva dans les universités d'Aix-Marseille en 2004, je me trouvais à monter en Licence un enseignement avancé de bases de données avec un volume d'une trentaine d'heures réparties en 12 heures de cours et 18 de T.D. utilisant le logiciel ACCESS. Je ne parlerai pas ici de l'applicatif ACCESS, mais de la partie de conception méthodologique qui se trouve en amont.

Je recommandais trois ouvrages à mes étudiant(e)s :

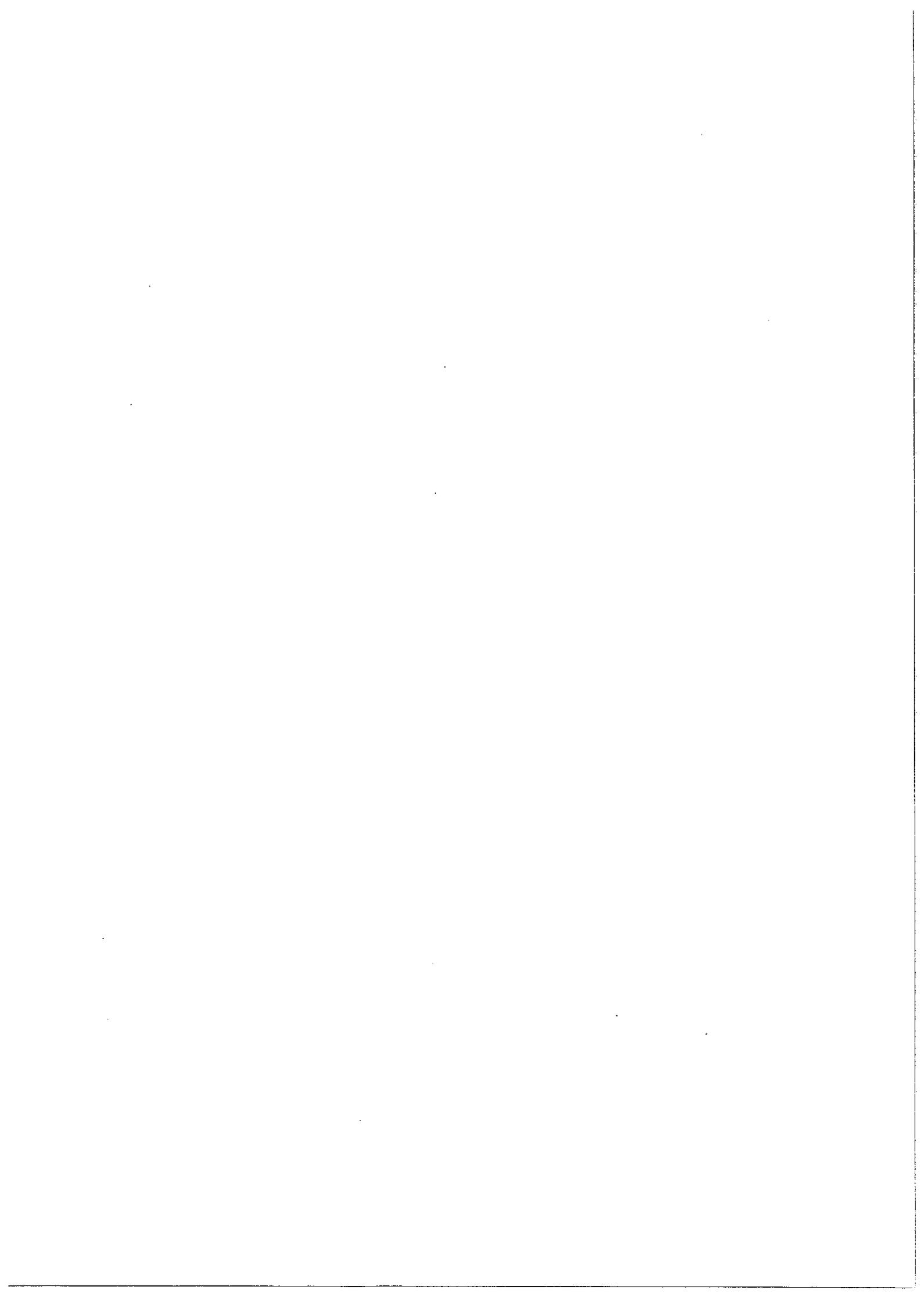
- | | |
|--------------------------|--|
| <i>A. Aho, J. Ullman</i> | Concepts fondamentaux de l'informatique Editions DUNOD |
| <i>G. Gardarin</i> | Bases de données Editions Eyrolles |
| <i>A. Meier</i> | Introduction pratique aux bases de données relationnelles Collection IRIS Editions Springer. |

Parmi mes recherches de supports de cours en ligne sur la toile, je conseillais le document de mon collègue Marc Grange de l'université de Lyon 2. Pourquoi ? Pour sa clarté et sa pédagogie et la mise en amont des concepts du modèle relationnel indépendamment des domaines d'applications, ici les sciences humaines et sociales. Ensuite, suit l'apprentissage, dans le cas du relationnel, du langage SQL ou de l'utilisation du logiciel BIEasy+ développé par Marc Grange qui permet la mise en pratique des concepts développés dans l'article qui suit.

Le lecteur, plus intéressé par les aspects formels peut parcourir les pages de l'article consacrées à la normalisation.

Enfin, vous trouverez des compléments en ligne sur le serveur de Marc Grange, à l'adresse <http://www.marc-grange.net> :

- exemples de bases des domaines de la sociologie, démographie, psychologie et géographie,
- présentation de SQL,
- possibilité de télécharger son logiciel BIEasy,
- adresses de serveurs sur les bases de données,
- présentation de la partie conception et normalisation des bases de données, qu'il nous a autorisé à reproduire dans notre bulletin.



Introduction aux bases de bases de données

Conception et normalisation

Marc Grange

Conception de Bases de données

Cette présentation n'est pas un cours complet sur la conception de bases de données, mais une première approche destinée à préciser la démarche et les concepts nécessaires pour construire des bases de données cohérentes.

L'approche et le vocabulaire utilisé sont volontairement destinés à des lecteurs non spécialistes sans pour autant éluder les notions essentielles indispensables au processus de conception : "Ce qui se conçoit bien s'énonce clairement, et les mots pour le dire arrivent aisément"... (Boileau)

La démarche et l'outil (voir ci-dessous) présentés dans ces pages ne sont pas des cas d'école et permettent de construire et d'utiliser des bases de données "réelles" à usage aussi bien individuel que professionnel .

La méthode de conception présentée ci-après, est une méthode simple (accessible tous) et opérationnelle. Mais simple ne veut pas dire simpliste. Cette méthode, comme toutes les méthodes de conception de bases de données relationnelles, repose sur les travaux de Codd (1970) de Date (1975) et suit les recommandations du groupe de travail ANSI-SPARK (1975). Seul l'aspect inutilement verbeux des méthodes "traditionnelles" a été délibérément supprimé.

Le logiciel IBEasy+, développé par l'auteur, permet la mise en pratique des concepts développés dans cette présentation :

1. Utilisation pratique de la démarche de conception de bases de données
2. Conception et utilisation de bases de données relationnelles.

Le lecteur, plus intéressé par les aspects formels que les aspects applicatifs, peut se reporter dans ces pages à la rubrique "Normalisation".

Notion de Base de Données

Une base de données est un ensemble *structuré* d'informations *non redondantes* dont l'organisation est régie par un *modèle de données*.

Les deux mots-clefs qui interviennent dans la définition d'une base de données sont :

structuration (l'aide du modèle de données) et *non répétition* (non redondance ou redondance minimale) des données.

Construire une base de données consiste regrouper les données en paquets "homogènes", les (**entités, tables**), chaque entité (table) étant composée d'un nombre fini de données élémentaires, les **Attributs** ou **Champs**, la répétition (redondance) des attributs devant être minimale. Le modèle de données utilisé dans cette présentation est le modèle "entités-associations".

Système de Gestion de Base de Données (SGBD)

Un SGBD est un logiciel qui joue le rôle d'interface entre les utilisateurs et la Base de Données.

Un SGBD permet de décrire, manipuler et interroger les données d'une Base de Données. Il est chargé de tous les problèmes liés aux accès concurrents, la sauvegarde et la restauration des données. Il doit de plus veiller au contrôle, l'intégrité et la sécurité des données.

Base de données et tableur

Une erreur, fréquemment commise par les utilisateurs débutants, consiste confondre une base de données avec un tableur. Les tables d'une base de données se présentent effectivement en lignes et colonnes, comme dans un tableur, mais la comparaison s'arrête là et les différences entre les deux

approches sont importantes et nombreuses. Le tableau suivant, résume les principales différences entre une base de données et un tableur.

| Différences sur... | Tableur | Base de données |
|-----------------------------------|--|--|
| Utilisation principale | Calculs | Gestion et traitement des données |
| Structuration des données | Aucune | Structuration et cohérence forte |
| Contrôles d'intégrité des données | Aucuns | Vérification stricte des valeurs possibles de chaque donnée |
| Accès aux données | Mono utilisateur | Multi-Utilisateurs |
| Confidentialité des données | Aucun contrôle | Vérification des droits d'accès de chaque utilisateur |
| Taille des données | - Une table - Quelques dizaines de lignes | - Plusieurs tables - Plusieurs milliers de lignes par table |
| Traitement sur les données | Quantitatifs | Qualitatifs et quantitatifs |
| Interrogations des données | Réalise par des procédures spécifiques | Langage "universel" SQL |

La conception de Bases de Données

La conception d'une Base de Données, implique 3 étapes fondamentales :

1. Analyse des documents : construction du dictionnaire des données.
2. Structuration du dictionnaire des données : détermination des entités et associations.
3. Mise en relation des entités : Schéma des données.

Conception de Bases de données : étape 1

Analyse des documents

La première étape, dans la construction d'une base de données, consiste à réunir tous les documents représentatifs des données que l'on souhaite modéliser.

Le terme « documents » doit être ici pris dans un sens large. Il peut s'agir en effet soit de documents papier, soit de documents magnétiques (enregistrement d'entretiens, disque ou disquette informatique)... soit de tout autre support utilisable pour conserver des informations.

Documents de travail

Pour illustrer les différentes étapes de conception d'une base de données nous utiliserons, comme exemple type, la gestion d'étudiants qui suivent les différents enseignements d'un diplôme. Les documents de travail utilisés sont les suivants :

| Liste des étudiants | | | |
|---------------------|-------------------|-------------------|-----------|
| N° étudiant | Nom de l'étudiant | Date de naissance | Sexe |
| 1 | Dupont, Charles | 18-03-1981 | H (Homme) |
| 2 | Dubois, Jules | 02-11-1982 | H (Homme) |
| 3 | Favier, Isabelle | 02-02-1979 | F (Femme) |
| ... | ... | ... | ... |

| Relevé de notes | | | |
|---------------------------------|---------------|-------------|-----------|
| N° étudiant : 1 Dupont, Charles | | | |
| N° matière | Nom | Coefficient | Note / 20 |
| 1 | Mathématiques | 3 | 10 |
| 2 | Informatique | 2 | 9 |
| 3 | Sociologie | 2 | 12,5 |
| 4 | Histoire | 1 | 13 |
| 5 | Géographie | 1 | 7 |
| Moyenne Générale | | | 10,3 |

| Liste des enseignants | | | | |
|-----------------------|---------------------|-------|------------|------------------------|
| N° enseignant | Nom | Grade | Ancienneté | Matière enseignée |
| 10 | Bertrand, Pierre | ASS | 2 | Sociologie |
| 11 | Dupont, Auguste | MCF | 3 | Mathématiques |
| ... | ... | ... | ... | ... |
| 15 | Simon, Etienne | ASS | 5 | Histoire Géographie |

Règles de gestion :

- Une matière n'est enseignée que par un seul enseignant.
- Un étudiant n'a qu'une note par matière.

Dictionnaire des données

Partant des documents, la phase d'analyse consiste en extraire les informations élémentaires (non décomposables) qui vont constituer la future base de données.

La réunion de l'ensemble des données élémentaires, que l'on appelle des **attributs** ou des **champs**, constitue le dictionnaire des données. Chaque attribut (champ) du dictionnaire de données peut être caractérisé par les propriétés suivantes :

| Propriété | Signification |
|-------------------------|---|
| Mnémonique | Abréviation du nom de l'attribut. |
| Libellé | Libellé contenant la signification précise et le règle de l'attribut. |
| Type de donnée | Type de l'attribut : entier, réel, chaîne de caractères, date... |
| Contraintes d'intégrité | Liste des contraintes sur les valeurs possibles de l'attribut |
| Règle de calcul | Règle de calcul (d'obtention) de l'attribut correspondant. |

Le dictionnaire des données relatifs aux documents précédents est le suivant :

| Mnémonique | Libellé | Type | Contraintes | Règle de calcul |
|------------|---------------------------|------------|-----------------------|--------------------------------------|
| Ancien | Ancienneté dans le grade | Entier | ≥ 0 | |
| Cdsexe | Code sexe | Chaîne(1) | H ou F | |
| Coeff | Coefficient de la matière | Entier | > 0 et < 6 | |
| Dtnaiss | Date de naissance | Date | | |
| Grade | Grade de l'enseignant | Chaîne(3) | ASS ou MCF ou PR | |
| Lbsexe | Libellé du sexe | Chaîne(7) | Homme ou Femme | |
| Moyenne | Moyenne au diplôme | Réel | ≥ 0 et ≤ 20 | $\sum (Note * Coeff) / \sum (Coeff)$ |
| Nomat | Nom de la matière | Chaîne(15) | | |
| Nomens | Nom de l'enseignant | Chaîne(20) | | |
| Nometu | Nom de l'étudiant | Chaîne(20) | | |
| Note | Note obtenue à la matière | Réel | ≥ 0 et ≤ 20 | |
| Numat | N° de la matière | Entier | > 0 | |
| Numens | N° de l'enseignant | Entier | > 0 | |
| Numetu | N° de l'étudiant | Entier | > 0 | |

Conception de Bases de Données : étape 2

Structuration des données

Les informations contenues dans le dictionnaire de données doivent être structurées en "paquets" homogènes (entités, tables) dans lesquels la répétition (redondance) d'informations doit être minimale.

Les attributs qui sont calculés (qui ont une règle de calcul) *ne doivent pas* être pris en compte dans la phase de structuration des données. Ils sont en effet obtenus partir d'autres attributs, et les inclure dans la phase de structuration introduirait une forme de redondance et donc des possibilités d'incohérence. Dans notre exemple, l'attribut "Moyenne" appartient cette catégorie.

On dira qu'un ensemble d'attributs est homogène si tous les attributs qui le composent ont un lien "direct".

La redondance à l'intérieur d'un ensemble doit être minimale car la redondance est source d'incohérence. En effet, si la même information est répétée plusieurs fois, cela signifie :

- Qu'elle devra être saisie plusieurs fois, ce qui multiplie les risques d'erreurs de frappe.
- Que lorsqu'elle devra être mise à jour, les modifications devront intervenir à plusieurs à endroits d'où des risques de mise à jour partielle.

Pour regrouper les attributs du dictionnaire de données on utilise un élément structurant qui s'appelle la dépendance fonctionnelle (DF).

Pour une justification plus formelle du processus de structuration des données présent ci-après, se reporter, dans ces pages à la rubrique Normalisation.

Notion de dépendance fonctionnelle (DF)

On dit qu'il existe une dépendance fonctionnelle entre un attribut A1 et un attribut A2, on note $A1 \rightarrow A2$, si connaissant une valeur de A1 on ne peut lui associer qu'une seule valeur de A2. On dit aussi que A1 détermine A2. A1 est la source de la dépendance fonctionnelle et A2 le but.

Exemple de DF

Dans notre base de données exemple, connaissant un numéro d'étudiant, on connaît de manière unique le nom de cet étudiant et, entre autres, sa date de naissance. On a donc :

Numetu \rightarrow Nometu, Dtnaiss.

En revanche, l'expression : Numetu \rightarrow Note est fausse.

En effet, connaissant un numéro d'étudiant on peut connaître les notes qu'il a obtenues dans chaque matière. La connaissance du numéro d'étudiant ne permet donc pas de connaître une note particulière et il n'y a donc pas de DF.

DF simple :

On dira qu'une DF est simple si sa source n'est composée que d'un seul attribut.

Une DF simple caractérise une *entité (table)* dont la source est la clé et dont les propriétés sont constituées par le but de la DF.

DF composée :

On dira qu'une DF est composée si sa source est composée par la réunion de plusieurs attributs.

Une DF composée caractérise une *association entre entités (tables)* dont la source est la clé et dont les propriétés sont constituées par le but de la DF. Il ne doit pas y avoir d'attributs superflus dans la source d'une DF composée. Par exemple si nous avons les 2 DF composées :

- 1- A1, A2, A3 \rightarrow A4
- 2- A1, A2 \rightarrow A4

La DF numéro 1 est incorrecte puisque A3 est superflu. La DF composée "correcte" est : A1, A2 -> A4

Pour représenter les entités et associations nous utiliserons la notation suivante :

NOM_DE_L'ENTITE (Source, But ...). La clé est soulignée.

Soit par exemple :

| DF | Entités |
|-------------|--------------------------------|
| A1>A2,A3,A4 | ENTITE1 (<u>A1</u> ,A2,A3,A4) |
| A3->A4 | ENTITE2 (<u>A3</u> ,A4) |
| A1,A5->A6 | ASSOC1 (<u>A1</u> ,A5,A6) |

Suppression de la redondance d'informations

L'utilisation des DF permet de structurer les informations contenues dans le dictionnaire des données mais n'élimine pas, directement, la redondance éventuelle de ces informations.

Par exemple, dans le tableau ci-dessus, les attributs A3 et A4 sont redondants puisqu'ils se trouvent dans deux entités : ENTITE1 et ENTITE2. Il est donc nécessaire d'éliminer cette redondance **sans toutefois perdre des informations** : c'est dire supprimer des entités

Dans cet exemple, on ne peut supprimer A3 de l'entité2 car cela impliquerait la suppression de l'entité et donc une perte d'information. A4 ne peut pas non plus être supprimé de l'entité2 puisque c'est la seule propriété de cette entité. En revanche A4 peut être supprimé de l'entité1, sans perte d'information, puisqu'il peut être obtenu (dérivé) par le biais de A3, en utilisant la propriété de transitivité des DF.

La structure "optimale" (cohérence maximale et redondance minimale) du schéma de DF précédent est donc :

A1 -> A2, A3

A3 -> A4

A1, A5 -> A6

D'où la règle :

Pour supprimer les redondances sur un ensemble de DF il faut éliminer les attributs qui peuvent être obtenu par transitivité

Démarche de structuration des données

La structuration des données du dictionnaire des données s'effectue en 5 étapes :

1. Détermination de la liste des DF simples
2. Prise en compte des attributs non classés dans l'étape 1 et détermination des DF composées
3. Elimination des transitivités du schéma des DF
4. Construction, à partir des DF simples, des entités de la base de données
5. Construction, à partir des DF composées, des associations de la base de données

L'application de cette démarche à notre exemple nous donne les résultats suivants :

1-DF simples de la base de données ETUDIANTS

Numetu -> Nometu, Dtnaiss, Cdsexe, Lbsexe

Cdsexe -> Lbsexe

Numens -> Nomens, Grade, Ancien

Numat -> Nomat, Coeff, Numens, Nomens, Grade, Ancien

2-Attributs non classés et DF composées de la base de données ETUDIANTS

L'attribut "note" n'a pas été classé dans l'étape précédente, il intervient donc dans une DF composée. Cette DF est la suivante :

Numetu, Numat -> Note

3-Elimination des transitivités de l'ensemble des DF

Numetu -> Nometu, Dtnaiss, Cdsexe

Cdsexe -> Lbsexe

Numens -> Nomens, Grade, Ancien

Numat -> Nomat, Coeff, Numens

4-Construction des entités

ETUDIANT (Numetu, Nometu, Dtnaiss, Cdsexe)

SEXE (Cdsexe, Lbsexe)

ENSEIGNANT (Numens, Nomens, Grade, Ancien)

MATIERE (Numat, Nomat, Coeff, Numens)

5-Construction des associations

NOTES (Numetu, Numat, Note)

Conception de Bases de données : étape 3

Construction du schéma des données

La dernière étape consiste à mettre en relation les entités et associations trouvées dans l'étape précédente afin de construire la structure générale des données : **le schéma des données**.

Le schéma obtenu par application de la méthode est le schéma conceptuel des données.

Mise en relation des entités

Les entités sont mise en relation par l'intermédiaire des attributs qu'elles possèdent en commun. Les types de relations autorisées pour relier deux entités (associations) appartiennent aux deux catégories suivantes :

- .Relation de type 1-1** : à un élément de l'ensemble départ on ne peut faire correspondre qu'un seul élément de l'ensemble d'arrivée et réciproquement.
 - . Relation de type 1-n** : à un élément de l'ensemble de départ on peut faire correspondre plusieurs éléments de l'ensemble d'arrivée.
-

Par application de la méthode, les seules relations possibles entre entités et associations sont des relations de type n-1, soient des dépendances fonctionnelles. Cette étape vise donc déterminer les dépendances fonctionnelles entre entités et associations : construction du graphe des index.

Pour déterminer les relations entre entités (associations), il faut donc examiner

le cardinal de chacune des entités qui interviennent dans la relation.

Dans notre exemple, nous avons des relations suivantes :

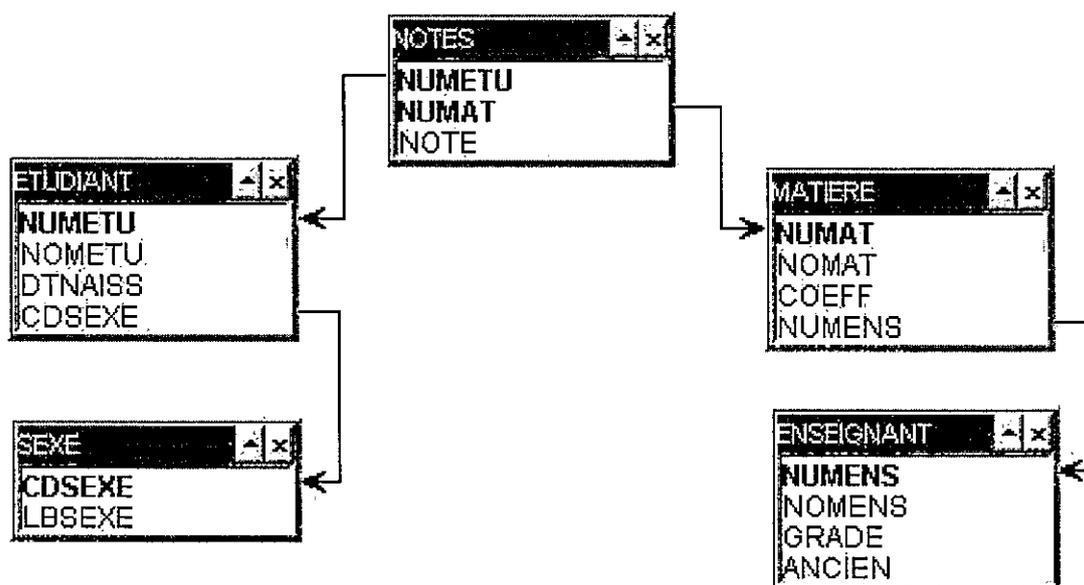
| Entité source | Entité but | Relation sur l'attribut | Type de relation |
|---------------|------------|-------------------------|------------------|
| SEXE | ETUDIANT | Cdsexe | 1-n |
| ETUDIANT | NOTES | Numetu | 1-n |
| MATIERE | NOTES | Numat | 1-n |
| ENSEIGNANT | MATIERE | Numens | 1-n |

Si l'on examine la première ligne du tableau, nous avons précisé que la relation entre SEXE et ETUDIANT était de type 1-n.

En effet, dans l'entité SEXE, une valeur particulière de l'attribut Cdsexe (H ou F) ne sera présente qu'une seule fois (1). En revanche, dans l'entité ETUDIANT, la même valeur de l'attribut Cdsexe pourra être présentée plusieurs fois (n), autant de fois qu'il y a d'étudiants de ce sexe.

Représentation graphique du schéma des données

Pour avoir une vision synthétique de la structure de la base de données (le schéma des données) on utilise une représentation graphique du type suivant :



Cette représentation est le schéma de la base de données "ETUDIANTS". C'est la structure de représentation optimale des données de notre problème : **Cohérence maximale et redondance minimale.**

Remarque importante pour les "puristes"

Le schéma ci-dessus est bien le *schéma conceptuel des données et non le schéma logique des données*. En effet, les relations n-1 représentent les dépendances fonctionnelles qui existent entre les entités et associations. *Le schéma précédent doit donc être vu comme une image du graphe des dépendances fonctionnelles*. Il s'agit en fait de la couverture minimale du graphe.

Avec cette méthode, le passage du schéma conceptuel au schéma logique puis physique est évident :

Schéma logique : Les dépendances fonctionnelles entre entités et associations deviennent des contraintes d'intégrité référentielle.

Schéma physique : Chaque entité/association devient une table

Normalisation

La théorie de la normalisation est liée au modèle de données relationnel créé par Codd en 1970. Ce modèle est basé sur la notion de relation.

Le modèle de données relationnel

Notion de relation

Une relation est un tableau 2 dimensions, dont les colonnes représentent les propriétés de la relation (les attributs ou champs), et dont les lignes correspondent aux différentes valeurs (occurrences) de ces attributs. Toutes les valeurs d'un attribut ne peuvent appartenir qu'à un seul domaine : entier, réel, chaîne de caractères..., et toutes les lignes doivent être différentes, d'où la notion de clé d'une relation.

Clé d'une relation

La clé d'une relation est le sous-ensemble minimum des attributs qui permet d'identifier chaque ligne de manière unique. Toute relation doit obligatoirement avoir une clé.

Schéma d'une relation

Il correspond à la définition en "intention" de la relation et se compose du nom de la relation suivie de la liste des attributs qui composent cette relation, les attributs clés étant soulignés. Soit, par exemple la relation R composée des attributs A1, A2 et A3 dont le schéma est : R(A1,A2,A3).

Le tableau avec les lignes correspond à une "extension" possible de la relation :

| <u>A1</u> | A2 | A3 |
|-----------|-----------|----|
| 1 | Libellé 1 | 50 |
| 2 | Libellé 2 | 60 |
| 3 | Libellé 3 | 50 |
| ... | | |

La notion de relation est donc simple et intuitive, mais conduit de nombreuses anomalies si on l'applique au sens strict. Les anomalies les plus importantes sont les suivantes :

- absence de liaisons sémantiques entre les données.
- redondance des données.
- comportements "anormaux" des relations lors de mises à jour : modifications, suppressions.

Nous utiliserons, pour illustrer les différentes notions présentées, les attributs suivants qui interviennent dans la gestion d'enseignements :

| Mnémonique | Libellé | Domaine |
|-------------|---------------------------|------------|
| Coefficient | Coefficient de la matière | Entier |
| Nomat | Nom de la matière | Chaîne(30) |
| Nomens | Nom de l'enseignant | Chaîne(30) |
| Numat | N° de la matire | Entier |
| Numens | N° de l'enseignant | Entier |

Nous supposons qu'un enseignant n'enseigne que dans une matière, mais qu'une matière est enseignée par plusieurs enseignants. Il y a 5 enseignants par matière et il y a 10 matières dans le cursus. Nous définissons, sur ces attributs, la relation ENSEIGNEMENTS.

ENSEIGNEMENTS(Numens, Nomens, Numat, Nomat, Coefficient)

"Problèmes" de la relation ENSEIGNEMENTS

Nous remarquons tout d'abord que cette relation satisfait complètement la définition de base et peut donc être implantée et utilisée telle quelle dans un SGBD relationnel. En effet, les valeurs des attributs n'appartiennent qu'à un seul domaine et la clé de la relation (Numens) permet de distinguer de manière unique chaque ligne.

Cette relation obtenue, sans structuration des informations, présente toutefois de nombreuses anomalies.

Au niveau sémantique

La clé de cette relation est le numéro de l'enseignant (Numens). Cela veut dire que cet attribut identifie, caractérise, la relation et donc que tous les autres

attributs de cette relation sont des propriétés liées sémantiquement à cet attribut. Cela implique, entre autres, que le coefficient d'une matière, ou son nom, sont des caractéristiques directes d'un enseignant, ce qui est, bien évidemment, inexact.

Ce problème de modélisation provient du fait que la relation ENSEIGNEMENTS ne représente pas un objet, une entité, du monde réel mais un amalgame de plusieurs entités.

Au niveau de la redondance d'informations

Dans la relation ENSEIGNEMENTS, les caractéristiques d'une matière (son nom et son coefficient) devront être répétées autant de fois qu'il y a d'enseignants dans la matière. Dans notre exemple 5 fois. Cette répétition, qui peut être évitée, va entraîner des risques de mise jour partielle des données donc des possibilités d'incohérence.

La taille, en caractères, nécessaire pour stocker cette relation est la suivante :
 $5*10*(2+30+2+30+2)= 3300$

Nous avons considéré qu'un entier était représenté sur 2 octets.

Au niveau des mises jour

-Il est impossible de saisir les caractéristiques d'une matière si elle n'est pas rattachée au moins à un enseignant. En effet, cela voudrait dire que la clé de la relation (Numens) aurait la valeur NULL ce qui est interdit par la contrainte sur l'unicité de la clé.

-S'il n'y a qu'un seul enseignant pour une matière, le départ de cet enseignant implique la suppression de toutes les informations de la relation pour cet enseignant et donc, dans ce cas, de celles qui sont relatives la matière.

Toutes ces anomalies proviennent du fait que la relation ENSEIGNEMENTS n'est pas sous une forme "normale".

Normalisation des relations

La mise sous une forme "normale" des relations, ou normalisation, vise à supprimer tous les comportements anormaux décrits ci-dessus, et repose sur la notion de dépendance fonctionnelle dont nous donnons à nouveau la définition

Dépendance fonctionnelle (DF)

On dit qu'il existe une dépendance fonctionnelle entre un attribut A1 et un attribut A2, on note $A1 \rightarrow A2$, si connaissant une valeur de A1 on ne peut lui associer qu'une seule valeur de A2. On dit aussi que A1 détermine A2. A1 est la source de la dépendance fonctionnelle et A2 le but.

Dans notre exemple, nous avons les DF suivantes :

Numens \rightarrow Nomens, Numat, Nomat, Coefficient

Numat \rightarrow Nomat, Coefficient

Dépendance fonctionnelle élémentaire

On dit qu'une dépendance fonctionnelle est élémentaire si la source ne comporte pas d'attributs superflus. La question sur l'élémentarité d'une DF ne doit donc se poser que lorsque la partie gauche de la DF comporte plusieurs attributs.

Dans notre exemple, toutes les DF sont élémentaires. Par contre, si nous avons les DF suivantes :

$A, B \rightarrow C$

$A \rightarrow C$

La dépendance fonctionnelle $A, B \rightarrow C$ n'est pas élémentaire puisque B est superflu.

Dépendance fonctionnelle directe

On dit que la dépendance fonctionnelle $A \rightarrow B$ est directe s'il n'existe aucun attribut C tel que l'on puisse avoir $A \rightarrow C$ et $C \rightarrow B$. En d'autres termes, cela signifie que la dépendance entre A et B ne peut pas être obtenue par transitivité.

Dans notre exemple, la DF : Numat \rightarrow Nomat, Coefficient est directe. Par contre la DF

Numens \rightarrow Nomat n'est pas directe.

Nous avons en effet : Numens \rightarrow Numat et Numat \rightarrow Nomat.

Les formes normales

Première forme normale (1FN)

On dit qu'une relation est en 1FN si tous les attributs qui la compose sont non décomposables.

La relation ENSEIGNEMENTS est en 1FN.

Deuxième forme normale (2FN)

On dit qu'une relation est en 2FN si elle est en 1FN et si toutes les DFs entre la clé et les autres attributs sont élémentaires.

La relation ENSEIGNEMENTS est en 2FN.

Troisième forme normale (3FN)

On dit qu'une relation est en 3FN si elle est en 2FN et si toutes les DFs entre la clé et les autres attributs sont élémentaires et directes.

La relation ENSEIGNEMENTS n'est pas en 3FN.

Forme normale Boyce-Codd (FNBC)

On dit qu'une relation est en FNBC si elle est en 3FN et si les seules DFs qui existent sont celles qui relient la cl aux autres attributs non-cl. La relation ENSEIGNEMENTS n'est pas en FNBC.

Relations en 3FN

Les problèmes de la relation ENSEIGNEMENTS proviennent du fait que cette relation est en 2FN. Pour éliminer les comportements anormaux, il est nécessaire de structurer les attributs de cette relation de manière à la décomposer en plusieurs relations en 3FN.

Pour construire les relations en 3FN relatives notre exemple, nous partons de l'ensemble des DFs :

Numens -> Nomens, Numat, Nomat, Coefficient

Numat -> Nomat, Coefficient

Après éliminations des transitivités, nous obtenons :

Numens -> Nomens, Numat

Numat -> Nomat, Coefficient

Ce qui nous donne les relations suivantes :

ENSEIGNANT(Numens, Nomens, Numat)

MATIERE(Numat, Nomat, Coefficient)

Ces deux relations sont en FNBC.

Pour des informations détaillées sur le processus de conception, se référer, dans ces pages, à la rubrique Conception BD.

Avantages des relations en 3FN (FNBC)

Au niveau sémantique

Les 2 relations précédentes correspondent à de véritables objets, entités, du monde réel. Les attributs propriétés de chaque relation sont bien des caractéristiques directes de la clé de la relation associée.

Au niveau de la redondance d'informations

Les informations caractéristiques d'une matière (son nom et son coefficient) ne seront présentes qu'une seule fois et non 5 fois comme dans la relation ENSEIGNEMENTS.

La redondance d'information est donc réduite au strict minimum, sans perte d'information, ce qui limite les risques d'incohérence.

De plus, la taille des données est la suivante :

La taille de la relation ENSEIGNANT : $5 \cdot 10 \cdot (2+30+2) = 1700$

La taille de la relation MATIERE : $10 \cdot (2+30+2) = 340$

La taille de la base de données est donc de 2040 caractères, soit une diminution de l'ordre de 40% par rapport la base de données ENSEIGNEMENTS (3300 caractères) , sans perte d'informations.

Au niveau des mises à jour

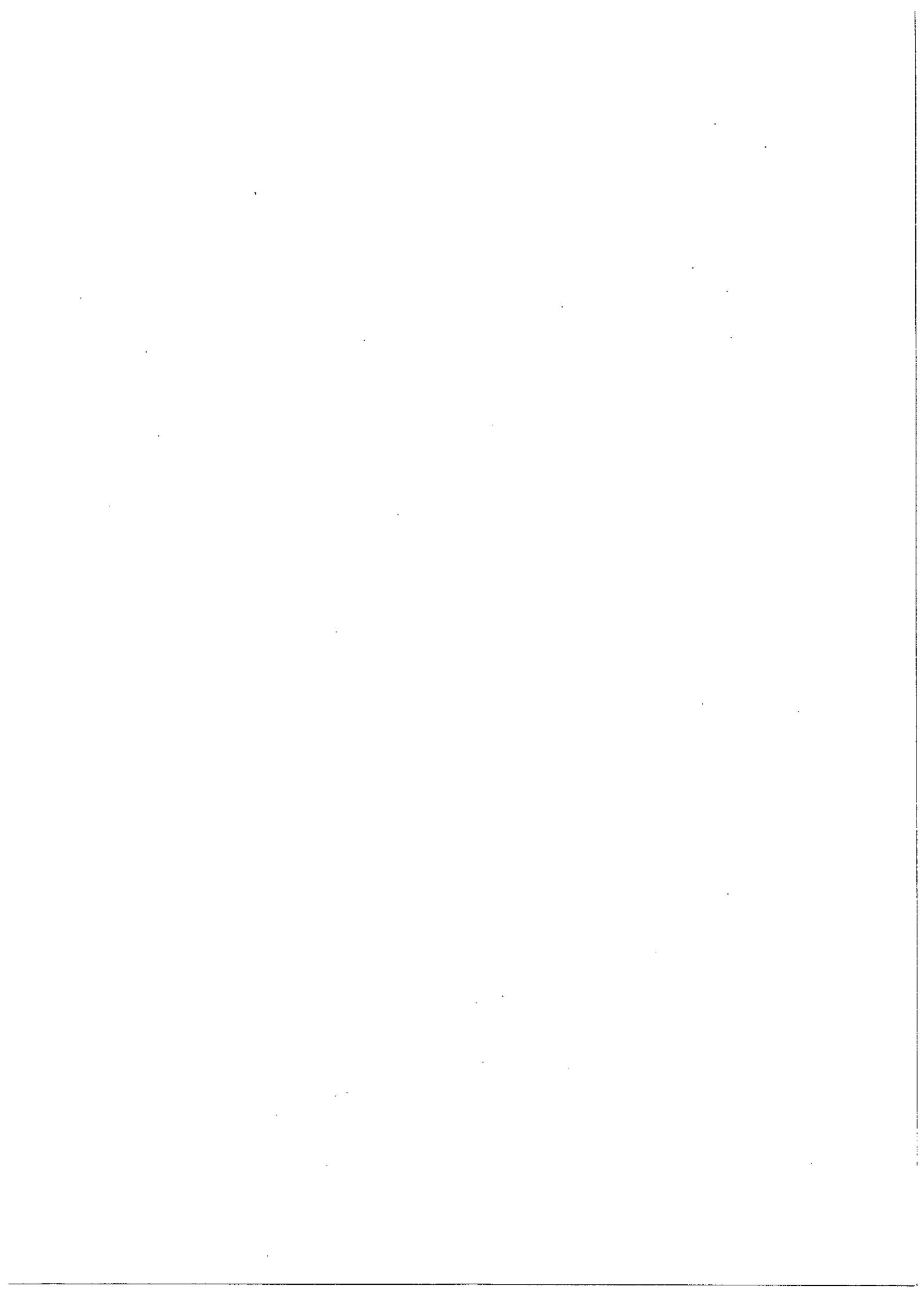
-Il est tout fait possible de saisir les caractéristiques d'une matière même si elle n'est pas rattachée à un enseignant. La saisie est réalisée uniquement dans la relation MATIERE.

-S'il n'y a qu'un seul enseignant pour une matière, le départ de cet enseignant implique la suppression des informations relatives à cet enseignant (une ligne de la relation ENSEIGNANT) et non de celles qui sont relatives à la matière.

Les 2 relations en 3FN, ENSEIGNANT et MATIERE, constituent donc la meilleure représentation possible pour les données de notre exemple.

La construction d'une structure de données "optimale" passe donc par la détermination des relations en 3FN (FNBC), construites à partir de l'ensemble des données de base.

La démarche de conception de bases de données (Conception BD) présentées dans ces pages, permet d'obtenir directement des relations en 3FN (FNBC)



VOUZZAVEDIBISAR

Turlubec

Edmond Bianco

Je m'en vais conter l'histoire étrange de François Turlubec. Non pas que ses aventures soient si différentes de celles que chacun d'entre nous puisse imaginer. Non. Nous avons tous, plus ou moins rêvé à des choses extraordinaires, des choses qui heurtent le sens commun, des choses qui vont à l'encontre des perceptions coutumières. Mais lui, il les a vécues, vraiment, enfin tout au moins nous sommes suffisamment nombreux à être convaincus que les événements se sont ainsi déroulés qu'il me paraît impossible de douter qu'il fut ainsi le héros de nombre de phénomènes surprenants. Qui était François Turlubec ? Je ne suis pas sûr de le savoir vraiment tant il me parut d'abord un individu tout à fait normal, puis tant il parut de plus en plus bizarre au fur et à mesure que j'ai pu le voir vivre.

Je l'ai connu sur une plage. C'était à Marseille, au sud de Marseille, une petite plage non trop envahie, la Marronaise. La côte très rocheuse en cet endroit s'incurve jusqu'à un chenal qui sépare l'île Maire de la terre ferme. La Marronaise est une petite calanque bordée de rocher aigus dont le fond de sable est très agréable. Etalé sur le rocher tiède je me dorais au soleil, avec un grand chapeau dont l'ombre était indispensable. Je laissais errer ma vue au ras du bord du chapeau, ce qui est une manière confortable pour observer sans être ébloui. Et, j'observais entre autres, ce garçon qui choisissait un rocher un peu élevé, au bout de la calanque, prenait son élan et plongeait dans la mer, bien tendu et les bras en croix. Il dessinait dans l'air un arc parfait et en bout de course, les mains en avant, pénétrait dans l'eau avec un minimum d'éclaboussures. C'était un plongeur émérite, voilà tout. Jusque là rien d'extraordinaire. La seule chose qui retenait mon attention était la lenteur apparente avec laquelle il décrivait sa trajectoire. Je le vis plonger plusieurs fois, et à chaque fois, il me semblait décrire la même trajectoire, mais de plus en plus lentement. J'en étais arrivé à compter les secondes et je le vis ralentir de plusieurs secondes lors de ses derniers plongements.

