

Introduction à la SGBD

Pape Abdoulaye BARRO

Enseignant – chercheur

UIDT – EPT – Réseaux des E-LAB

PLAN

- Généralités
- Le modèle conceptuel
- Le modèle relationnel
- Le langage SQL
- Etude et réalisation d'une base de données
- La sécurité des données

PLAN

- Généralités
- Le modèle conceptuel
- Le modèle relationnel
- Le langage SQL
- Etude et réalisation d'une base de données
- La sécurité des données

Généralités

objectif

L'objectif de ce chapitre est :

- de définir la notion de base de données ainsi que les principaux concepts qui s'y rattachent.
- de présenter:
 - la méthodologie qui permet de les concevoir,
 - les applications informatiques associées à leur mise en œuvre (SGBD)
 - ainsi que les différents métiers possibles.

Généralités

définitions – base de données

● C'est quoi une base de données?

- Il est très difficile de donner une définition exacte d'une base de données. Mais dès que des données sont stockées de manière organisée dans un ou plusieurs fichiers, on peut donc parler de base de données. Dès que l'on dispose d'un ensemble structuré, son exploitation (**ajout, suppression, mise à jour, etc.**) devient alors possible.

Une base de données est un ensemble de fichiers – informatiques ou non – structurés et organisés afin de stocker et de gérer de l'information.

Généralités

définitions – base de données

- Une **information** est un critère de choix parmi les éléments d'un ensemble de données qui permet de restreindre la taille de l'ensemble pour donner une réponse à une question.
 - Un *exemple d'ensemble de données* pourrait être la liste des élèves d'une classe. On pouvait se poser la question suivante: *qui est le meilleur élève de la classe ?*
 - On doit recherche parmi les meilleurs élèves (ceux qui ont obtenu une moyenne ≥ 12), Recherche : le meilleur d'entre eux (celui qui a la plus grande moyenne)
- L'**informatique** est le traitement automatique de l'information.

Généralités

définitions – base de données

● Quelques remarques:

- Vous avez peut-être une idée intuitive des bases de données. Attention toutefois, le mot est souvent utilisé pour désigner n'importe quel ensemble de données. Il s'agit là d'un abus de langage qu'il convient d'éviter.
- Pour mériter le terme de base de données, un ensemble de données non indépendantes doit être interrogable par le contenu.
 - Les données doivent pouvoir être retrouvées selon n'importe quel critère :
 - Il devrait être possible de trouver une liste d'articles qui coûtent moins de 500 francs.
 - Il doit également être possible de retrouver leur structure :
 - un produit a un nom, un prix, une quantité, ...

Généralités

définitions – base de données

- Les bases de données se situent au cœur de l'activité des entreprises, de l'administrations, de la recherche et de bon nombre d'activités humaines désormais liées à l'informatique.
- Une base de données est une représentation partielle et très simplifiée du monde réel, que l'on a obtenu par un processus de **modélisation**.
- Pour les manipuler, on utilise généralement un logiciel spécialisé appelé **SGBD** (**S**ystème de **G**estion de **B**ases de **D**onnées).
- Dans le domaine purement informatique, elles interviennent dorénavant à tous les niveaux.
 - Les développeurs d'applications s'appuient sur des bases de données externes pour gérer leurs données alors qu'auparavant elles étaient intégrées dans le programme.
- Les bases de données reposent sur des théories solides et sont à l'origine d'une des plus importantes disciplines de l'informatique : **l'ingénierie des systèmes d'information**.

Généralités

définitions - SGBD

- Un SGBD peut être perçu comme un ensemble de *logiciels systèmes* permettant aux utilisateurs d'insérer, de modifier et de rechercher efficacement des *données spécifiques* dans une grande masse d'informations (pouvant atteindre quelques milliards d'octets) partagée par de multiples utilisateurs.
 - Le SGBD rend transparent le partage, à savoir donne l'illusion à chaque utilisateur qu'il est seul à travailler avec les données.

Généralités

définitions - SGBD

- Les **SGBD** se distinguent clairement des **systèmes de fichiers**.
 - Les **SGBD** permettent la description des données (*définition des types par des noms, des formats, des caractéristiques et parfois des opérations*) de manière séparée de leur utilisation (*mise à jour et recherche*).
 - Ils permettent aussi de retrouver les caractéristiques d'un type de données à partir de son nom (*par exemple, comment est décrit un produit*).
 - Le **système de fichiers** est un composant de plus bas niveau ne prenant pas en compte la structure des données.
- La tendance aujourd'hui est d'intégrer le système de fichiers dans le SGBD.

Généralités

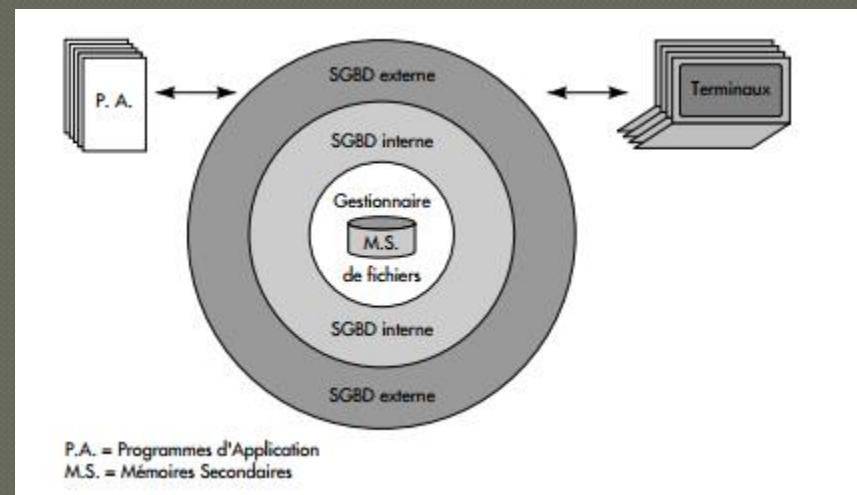
définitions - SGBD

- Un *SGBD* se compose en première approximation de trois couches emboîtées de fonctions (de la mémoire à utilisateur).
 - Le système de gestion de fichiers (ou gestionnaire de fichiers) fournit aux *couches supérieures* des mémoires secondaires idéales adressables par objets et capables de recherches par le contenu des objets (mécanismes d'indexation notamment).
 - le système d'accès aux données (ou SGBD interne) (*deuxième couche*) qui assure la gestion des données stockées dans les fichiers, l'assemblage de ces données en objets, le placement de ces objets dans les fichiers, la gestion des liens entre objets et des structures permettant d'accélérer les accès aux objets.
 - SGBD externe (*la troisième couche*) dont le rôle consiste à mettre en forme et à présenter des données aux programmes d'applications et aux utilisateurs interactifs. Il a pour rôle, d'assurer:
 - d'une part, l'analyse et l'interprétation des requêtes utilisateurs en primitives internes,
 - d'autre part, la transformation des *données extraites de la base* en *données échangeables avec le monde extérieur*.

Généralités

définitions - SGBD

- Au delà des SGBD, les systèmes d'informations intègrent plus souvent **des ateliers de génie logiciel** permettant de modéliser les données d'une base de données et de représenter les traitements associés à l'aide de graphiques et de langages de spécifications.
- **Ces outils d'aide à la conception**, bien que non intégrés dans le SGBD, permettent de spécifier les descriptions des données (supportés par les SGBD)



composition d'un SGBD

Généralités

Historique des SGBD

Les toutes premières bases de données ont vues le jour vers les années 60 lorsqu'on utilisait les fichiers reliés par des pointeurs.

- C'est vers 1965, avec l'apparition des systèmes IDS.I de Honeywell et IMS.I de IBM (pour la mission APOLLO), que les SGBD modernes ont commencé à voir le jour. Cela a permis de constituer des chaînes d'articles entre fichiers et de parcourir ces chaînes.
 - La première génération de SGBD est marquée par la séparation de la description des données et de la manipulation par les programmes d'application. Elle coïncide avec l'avènement des langages d'accès navigationnels.
 - Elle est basée sur les modèles réseau ou hiérarchique et a été dominée par les SGBD TOTAL, IDMS, IDS 2 et IMS 2. Elle traite encore aujourd'hui une partie importante du volume de données gérées par des SGBD.

Généralités

Historique des SGBD

- La **deuxième génération de SGBD** a fait son apparition vers les années 1970 dans les laboratoires (début du modèle relationnel) et vise à enrichir et à simplifier le SGBD externe afin de faciliter l'accès aux données pour les utilisateurs. Les données sont représentées sous forme de tables et les recherches et mises à jour sont effectuées à l'aide d'un langage non procédural standardisé appelé *SQL(Structured Query language)*.
- Les systèmes de la deuxième génération sont commercialisés depuis 1980 et aujourd'hui, ils représentent l'essentiel du marché des bases de données. Les principaux systèmes sont ORACLE, INGRES, SYBASE, INFORMIX, DB2 et SQL SERVER.

Généralités

Historique des SGBD

- La troisième génération de SGBD est apparue vers les années 1980 dans les laboratoires et est fortement adoptés dans le monde industriel. Elle supporte des modèles de données extensibles associant relationnel et objet, ainsi que des architectures mieux réparties, permettant une meilleure collaboration entre des utilisateurs concurrents.
- Parmi les systèmes objet-relationnels nous pouvons citer Oracle 8, IBM DB2 Universal Database, Informix Universal Server et ObjectStore (plus novatrice).
- Tous ces systèmes tentent de répondre aux besoins des nouvelles applications (multimédia, Web, CAO, bureautique, environnement, télécommunications, etc.).

Généralités

Historique des SGBD

- La **quatrième génération** quant à elle, est mise en place pour mieux supporter Internet(le Web), **les informations non structurées**, les objets multimédias, mais aussi faciliter l'extraction de connaissances(data mining) ainsi que l'aide à la prise de décisions. Il devient alors de plus en plus difficile de développer un nouvel SGBD. Cette nouvelle génération est juste une évolution des SGBD de 3e génération plutôt qu'à une nouvelle révolution.

Finalement, l'évolution des SGBD peut être perçue comme celle d'un arbre, des branches nouvelles naissant mais se faisant généralement absorber par le tronc, qui grossit toujours d'avantage.

Généralités

les étapes de la conception

Le processus de conception d'une base de données peut être décomposé en plusieurs étapes allant de l'analyse du monde réel jusqu'à la création et l'utilisation.

- l'**analyse du monde réel** est l'étape de collecte des informations et des besoins des futurs utilisateurs. C'est dans cette étape que l'on détermine les objectifs du système d'information à mettre en place et identifie tous les éléments à prendre en compte.
 - Il permet d'identifier les objets et de modéliser les liens existant entre eux . Par exemple:
 - une personne achète un terrain à 5 000 000 FCFA.
 - Dans cette phrase, nous avons deux objets liés à savoir *personne* et *terrain* et le *prix* est un composant du lien.
 - Il permet également d'exprimer des règles de validité de données. Par exemple:
 - Les terrains dont nous disposons coutent au *minimum* 1 000 000 FCFA et au *maximum* 10 000 000 FCFA.
 - L'exploitation des données ou des liens existant entre les entités est réalisée à l'aide du modèle conceptuel «*entité-association*» ou, plus couramment aujourd'hui, exprimé avec le langage **UML** (Unified Modeling Language).

Généralités

les étapes de la conception

- Le **passage au SGBD** est l'étape de transformation de la représentation précédente afin de la rendre acceptable par le SGBD choisi. Lors de cette étape, il est question de vérifier la qualité de la base de données en éliminant les redondances et autres. Le modèle relationnel, avec ses outils, permet d'arriver à cette fin.
 - On obtient alors un schéma des données qui fournira aux utilisateurs les informations nécessaires pour effectuer leurs **requêtes**, par exemple, la description des *noms de tables*, de *champs* et leurs *types*.
 - C'est dans cette phase que l'on définit les « *vues* » du système d'information qui sont adaptées à chaque *catégorie d'utilisateurs*.

Généralités

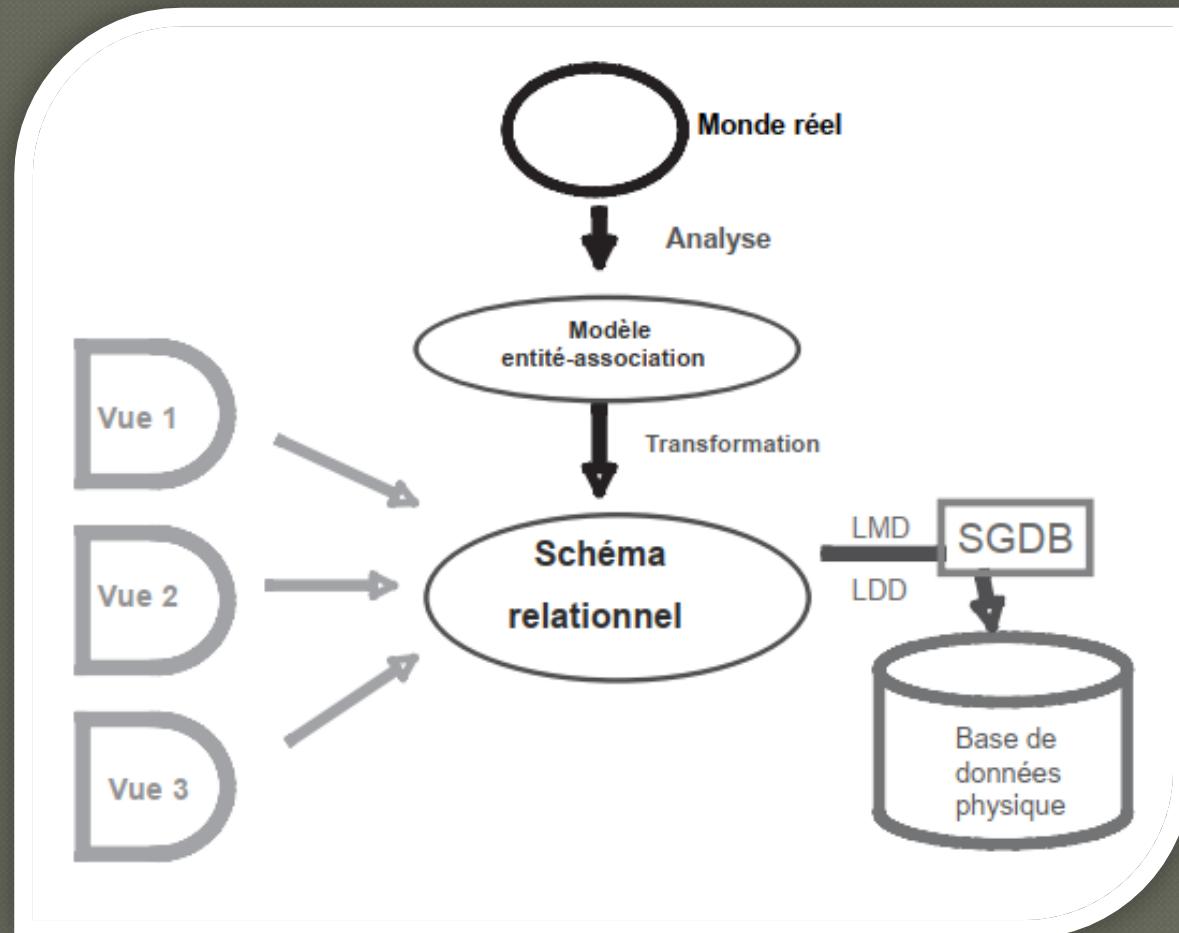
les étapes de la conception

- Enfin, nous passons à l'étape de création et d'utilisation de la base de données en utilisant le schéma précédemment défini. Il s'agit de la création des tables qui constituent la base de données, et puis, d'insérer les valeurs dans les tables.
 - La création des tables ainsi que l'utilisation de la base de données nécessitent le *langage SQL*.

Généralités

les étapes de la conception

Étapes de la conception d'une base de données.



Généralités

les différents métiers possibles

Dans le processus d'étude et de mise en place d'une base de données, diverses acteurs peuvent y intervenir donnant ainsi naissance à d'intéressants métiers d'**ingénierie des systèmes d'information**. Parmi lesquels, nous avons:

- le métier de *Consultants/Analystes* qui prennent en charge la première étape de conception c'est-à-dire l'étape consistant à analyser les activités et les flux d'information du système (activités d'une entreprise par exemple) à modéliser. Cette phase nécessite beaucoup de dialogue et de psychologie pour bien parvenir à exprimer les besoins des futurs utilisateurs.
 - Il n'est pas toujours nécessaire que l'acteur soit purement technique, il suffit juste qu'il soit capable de correctement exprimer les besoins afin de proposer un modèle conceptuel de donnée (le plus juste possible).

Généralités

les différents métiers possibles

- le métier de *Concepteurs de base de données* qui s'occupent de la transformation du modèle conceptuel en un modèle logique pouvant être exploité par le SGBD. C'est le spécialiste qui doit mettre en place les tables, les vues, les schémas d'accès à la base en collaboration avec l'administrateur.
 - Parfois, il peut être analyste et concepteur en même temps, dans ce cas, le cahier des charges aura tendance à être trop technique. Il peut aussi être administrateur, ce qui ne pose aucun souci.
- Le métier d'*Administrateurs de base de données* qui a la responsabilité du fonctionnement général du SGBD.
 - Il crée les ressources(base, comptes) a la demande et donne les droits d'accès aux usagers.
 - Il vérifie la disponibilité des ressources, veille aux défaillance sécuritaire et est en étroite collaboration avec l'administrateur système et réseau de la structure.
- Le métier de *Programmeurs d'application* qui est un utilisateur standard du système d'information. Ils y ont accès grâce aux vues définies par le concepteur de la base. Pour mener à bien leur taches, les développeurs doivent avoir une très bonne connaissance de la SGBD mise en place.

Généralités

résumé

- Il est très difficile de donner une définition exacte d'une base de données. Une base de données peut être considérée comme "*un ensemble de fichiers – informatiques ou non – structurés et organisés afin de stocker et de gérer de l'information*".
- Un SGBD peut être perçu comme un ensemble de logiciels systèmes permettant aux utilisateurs de pouvoir efficacement manipuler les données de la base.
- Un SGBD se compose, en première approximation, de trois couches à savoir la couche gestionnaires de fichiers, la couche SGBD interne et la couche SGBD externe.
- Les toutes premières bases de données ont vu le jour vers les années 60 et leurs évolutions ont vu naître diverses générations qui ont suivies le même trajet, en prenant en compte, à chaque étape, les besoins ou exigences des utilisateurs.
- Pour mettre en place une base de données de manière efficace, il est nécessaire de passer par quelques étapes à savoir, l'étude du monde réel, la proposition d'un modèle conceptuel, la traduction du modèle conceptuel en modèle relationnel avant donc de pouvoir mettre en place la base de données.
- Ces différents étapes impliquent à plusieurs acteurs faisant naître des métiers connexes comme le métier de consultant/analiste, le métier de concepteur de base, le métier d'administrateur de base de données et puis le métier de programmeur d'application.

Généralités

Discussions

- ◉ Q1: Quelles sont les différences majeures entre un fichier informatique et une base de données gérée par un SGBD ?

Q1: Quelles sont les différences majeures entre un fichier informatique et une base de données gérée par un SGBD ?

- Il n'est pas nécessaire de connaître la méthode de stockage des informations sur le disque pour manipuler les données avec une base de données.
- Un fichier informatique simple n'est pas conçu pour effectuer une recherche d'information par le contenu : pour retrouver les enregistrements, on est obligé de parcourir tout le fichier.
- Les modifications de structure (ajout/suppression d'un champ ou modification de sa taille...) nécessitent de recréer un autre fichier et d'y recopier les données.
- Une base de données contient en général plusieurs fichiers dont les enregistrements sont reliés entre eux.
- Etc.

Généralités

Discussions

- ◉ Q2: Est-il possible de faire réaliser toutes les étapes de la conception d'une base de données par une même personne ? Si oui, quelles sont alors ses compétences minimales ?

Généralités

Discussions

● Q2: Est-il possible de faire réaliser toutes les étapes de la conception d'une base de données par une même personne ? Si oui, quelles sont alors ses compétences minimales ?

- Dans une petite structure, c'est souvent la même personne qui réalise l'ensemble du processus de construction d'une base de données. Ce n'est évidemment pas la bonne méthode, car la vision d'un système d'information élaboré par un administrateur de base de données est très orientée par le SGBD qu'il emploiera. On peut facilement faire le parallèle avec le développement de logiciels où un programmeur va avoir une approche déformée par les préoccupations liées au langage plutôt que d'adopter un point de vue sur la structure générale de l'application. Au minimum, la personne devra disposer des compétences en conception de base de données et en administration du SGBD qui sera utilisé

Généralités

Discussions

- Q3: On décide de recopier régulièrement une base de données complète sur chacun des six sites de l'entreprise. Quel est l'intérêt de cette solution ?

Généralités

Discussions

- Q3: On décide de recopier régulièrement une base de données complète sur chacun des six sites de l'entreprise. Quel est l'intérêt de cette solution ?

C'est une solution coûteuse en ressources, en particulier pour la synchronisation de toutes les mises à jour, mais qui peut remplir deux fonctions :

- De toute évidence, l'idée est que les différents sites de l'entreprise accèdent à leur copie locale des données. Cela permet d'accélérer les accès et de répartir la charge sur les serveurs locaux de chaque site. Accessoirement, la circulation des requêtes d'interrogation et de mise à jour peut être limitée au réseau local, ce qui renforce la sécurité.
- Il existe six copies des données sur des sites géographiquement séparés. En cas de sinistre, on peut repartir sans problèmes avec une des copies de la base.

Avant de mettre en place un tel dispositif, on doit se poser la question de la mise à jour des données. Est-ce que les modifications se font uniquement sur la base "maître", ce qui semble plus raisonnable, ou peut-on les effectuer sur toutes les bases et "consolider" ensuite ?

PLAN

- Généralités
- Le modèle conceptuel
- Le modèle relationnel
- Le langage SQL
- Etude et réalisation d'une base de données
- La sécurité des données

Le modèle conceptuel

objectif

L'objectif de ce chapitre est de présenter :

- la première étape du processus de modélisation:
 - Recueil d'information;
 - Transcription pour le passage au modèle relationnel;
- les concepts et la mise en œuvre du modèle entité-association

Le modèle conceptuel

Démarche d'analyse > approche du monde réel

- La question à se poser à cette étape est: *comment appréhender et simplifier le monde réel, afin de pouvoir réaliser la modélisation?* Une question que le consultant doit chercher à répondre à travers ses diverses interventions. Il a la responsabilité d'identifier les besoins des utilisateurs ainsi que les objectifs et les processus d'alimentation en données des systèmes d'information à concevoir.
- Ces différentes étapes se déroulent souvent en même temps que le processus d'**analyse du problème**.
- C'est un processus itératif (des allers-retours entre ces différentes étapes de la conception) permettant de raffiner le modèle conceptuel.
 - L'**entretiens avec les utilisateurs** est la première phase de l'analyse du problème. Il permet effectuer une analyse du discours afin d'en extraire l'**information utile** .

Le modèle conceptuel

Démarche d'analyse > approche du monde réel

- L'**expression des besoins** repose sur la formulation du problème à l'aide de phrases simples qui décrivent la réalité à modéliser. Ces phrases se présentent sous la forme « sujet-verbe-complément », avec une tournure active quand cela est possible. Le but est d'obtenir deux types de phrases :
 - Celles qui décrivent les liens entre les objets du monde réel, généralement une action ou une propriété.
 - **Exemple** : Un lecteur emprunte un livre. Un livre a un auteur
 - Celles qui caractérisent la manière dont sont reliés ces objets (regroupements logiques de données).
 - **Exemple** : Un lecteur est considéré comme lecteur s'il a au moins déjà emprunté un livre. Un livre peut être emprunté par plusieurs lecteurs. Il n'y a pas de livres anonymes, un livre est écrit par au moins un auteur.
- Par la suite, il faut préciser les données qui constituent les objets ainsi que celles qui caractérisent les liens entre les objets.

Le modèle conceptuel

Démarche d'analyse > mise en œuvre

- On est intéressé sur "Comment procéder de manière intuitive pour l'obtention de ces phrases ?".

- Pour mieux répondre à cette question, il faut d'abord faire l'inventaire des objets tangibles du monde réel. Une fois ces objets identifiés, on cherche à exprimer les liens permettant de les associer.
- **Exemple:** si l'on doit modéliser une activité de location de DVD, les objets que l'on peut appréhender immédiatement sont les DVD et les clients. En ce qui concerne les liens entre ces objets, on note qu'un client réserve un DVD ou qu'un client loue un DVD, etc.
- Ensuite, il faut identifier les objets moins faciles à percevoir directement : les fournisseurs, les acteurs, les réalisateurs...
- Enfin, une fois les objets identifiés, on cherche à qualifier les liens trouvés. Il faut tenir compte du fait que le lien est toujours à double sens.
- **Par exemple**, un client emprunte plusieurs DVD. Un DVD est emprunté plusieurs fois ou n'est jamais emprunté par un client.

Le modèle conceptuel

Démarche d'analyse > mise en œuvre

- Pour parvenir à ce résultat, il faut savoir "quelles questions faut-il se poser et quelles questions doit-on poser aux acteurs de l'organisation ?".
 - Comment allez-vous décrire l'activité globalement, en termes simples, sans entrer dans les détails, pour identifier les objets et leurs liens éventuels?
 - Demandez quelles sont les « procédures » utilisées dans l'activité afin de pouvoir caractériser les liens existants entre les objets. Les procédures permettent d'énoncer les contraintes qui seront intégrées ensuite dans la base de données.
- L'objectif est de parvenir à modélisé l'action (le lien entre les objets) qui représente une activité (rarement des éléments statiques).
 - **Exemple d'action:** une personne *emprunte* un DVD, une voiture *est achetée* par un client,
- Le **temps** est une **notion** importante puis qu'une base de données modélise des actions qui ont lieu durant une période de temps. Pour éviter des erreurs de conception, il va falloir toujours avoir à l'esprit cet aspect.
 - Une erreur classique est de confondre l'aspect simultané d'une action avec la possibilité de la réitérer durant la période concernée.
 - Lorsque l'on spécifie qu'« un livre peut être emprunté plusieurs fois », il est évident qu'un livre ne peut être emprunté par deux personnes simultanément, mais plutôt qu'il pourra être emprunté à plusieurs reprises durant la période modélisée du fonctionnement de la bibliothèque.

Le modèle conceptuel

Démarche d'analyse > Cas pratique

- Pour des besoins d'illustration, on peut essayer de modéliser schématiquement le fonctionnement d'un hôtel.
- 1. Quelles phrases simples pour décrire l'activité de l'hôtel ?
 - En première approche, on peut dire: *Un hôtel loue des chambres à des clients qui effectuent des réservations.*
 - Après analyse, on peut la réécrire de la manière suivante: *Un client loue une chambre ; un client réserve une chambre.*
 - ❖ On a pu identifier deux objets du monde réel à savoir "*la chambre*" et "*le client*" qui sont liés doublement (cas assez fréquent).
- 2. Comment procéder à la caractérisation des liens ?
 - Une chambre peut n'avoir jamais été louée ni réservée.
 - Un client intègre le système à partir du moment où il a effectué soit une réservation, soit une location.
 - Un client peut réserver ou louer plusieurs chambres.
 - Une chambre peut être réservée ou louée plusieurs fois, mais pas pendant la même période de temps.
 - ❖ On considère toujours une modélisation associée à une période de temps donnée. Au début du processus, une chambre peut ne pas encore avoir été louée.
- 3. On procède à la description des données des objets et des liens.
 - Un client est caractérisé par son nom, son adresse et son numéro de téléphone.
 - Une chambre est caractérisée par son numéro, un nombre de places, son tarif journalier et la présence ou non d'un cabinet de toilettes.
 - Une location est caractérisée par une date de début, un nombre de jours et les consommations annexes (petits déjeuners et autres...).
 - Une réservation est caractérisée par une date de début, un nombre de jours et le versement d'une avance éventuelle.

Le modèle conceptuel

Modélisation > Modèle entité-association

- Le **modèle conceptuel** est l'équivalent du schéma technique d'un appareil ou du plan d'un bâtiment. Il permet de donner une vue d'ensemble des données et des liens qui les caractérisent. Nous allons utiliser le **modèle entité-association** (*ou entity-relationship en anglais*) pour schématiser le besoins .
- Le formalisme "entité-association" utilise une représentation graphique sous forme de diagrammes.
 - Les entités sont les objets concrets ou abstraits du monde réel.
 - Les associations représentent le lien entre ces entités.
- Comme on l'a vu précédemment, on peut identifier les entités et les associations en effectuant une analyse du discours, c'est-à-dire des phrases de type "sujet-verbe-complément". Les sujets et les compléments sont les entités, et le verbe modélise l'association.

Le modèle conceptuel

Modélisation > Entités

- Une **entité** a un nom et est composée de champs de données appelés attributs. Un des attributs (ou un ensemble d'attributs), doit être choisi comme identifiant (notion de clé) de l'entité, afin d'identifier de manière unique une occurrence (ou représentant) de cette entité.
- Une entité est représentée par un rectangle qui contient son nom et ses attributs. L'identifiant est soit souligné ou soit précédé d'un caractère '#'. Voir exemple ci-dessous:

Entité	Occurrence ou représentant	Occurrence ou représentant
Chambre	Chambre	Chambre
# IDChambre	221	75
NombreDePlaces	2	1
Tarif	25 000	15 000
...

Le modèle conceptuel

Modélisation > Entités

- Le choix de l'identifiant n'est pas toujours évident. Ce qui pousse des fois à introduire un attribut supplémentaire arbitrairement afin de pouvoir disposer d'un identifiant.
- Si on prenait le cas pratique précédent, pour le cas d'un client, comme aucun des attributs issus de l'analyse ne permet de l'identifier de manière unique, il va falloir lui intégrer un attribut servant d'identifiant. Il peut être nommé IDClient.

Client
IDClient
Nom
Adresse
NumTéléphone
...

Le modèle conceptuel

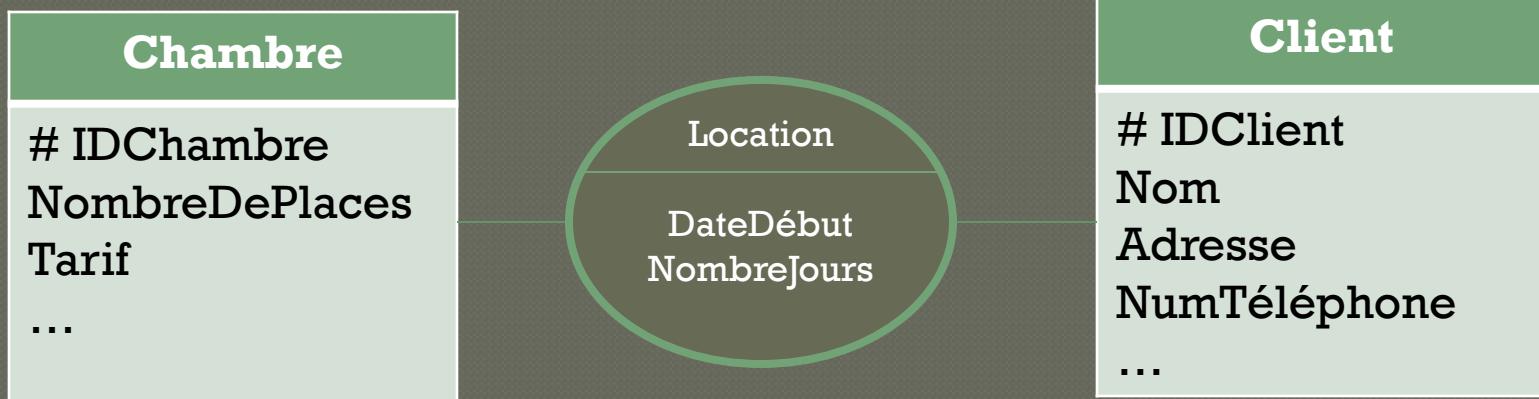
Modélisation > Entités

- En réalité, un identifiant peut être une juxtaposition de différents attributs. Par exemple, pour un client, on peut avoir "nom+prénom+date_naissance+ville_naissance" comme identifiant puisqu'il n'est pas évident de trouver deux clients de même nom et prenom, qui sont nées le même jour et dans la même ville.
- Dans la pratique, il est recommandé de choisir un seul attribut comme identifiant plutôt que de constituer un identifiant composite pour éviter des problèmes d'invalidité lorsque les données évoluent. Par exemple, pour une personne, on préfère l'identifier par un numéro de sécurité sociale plutôt que par un numéro de passeport qui a une durée de validité limitée.

Le modèle conceptuel

Modélisation > Associations

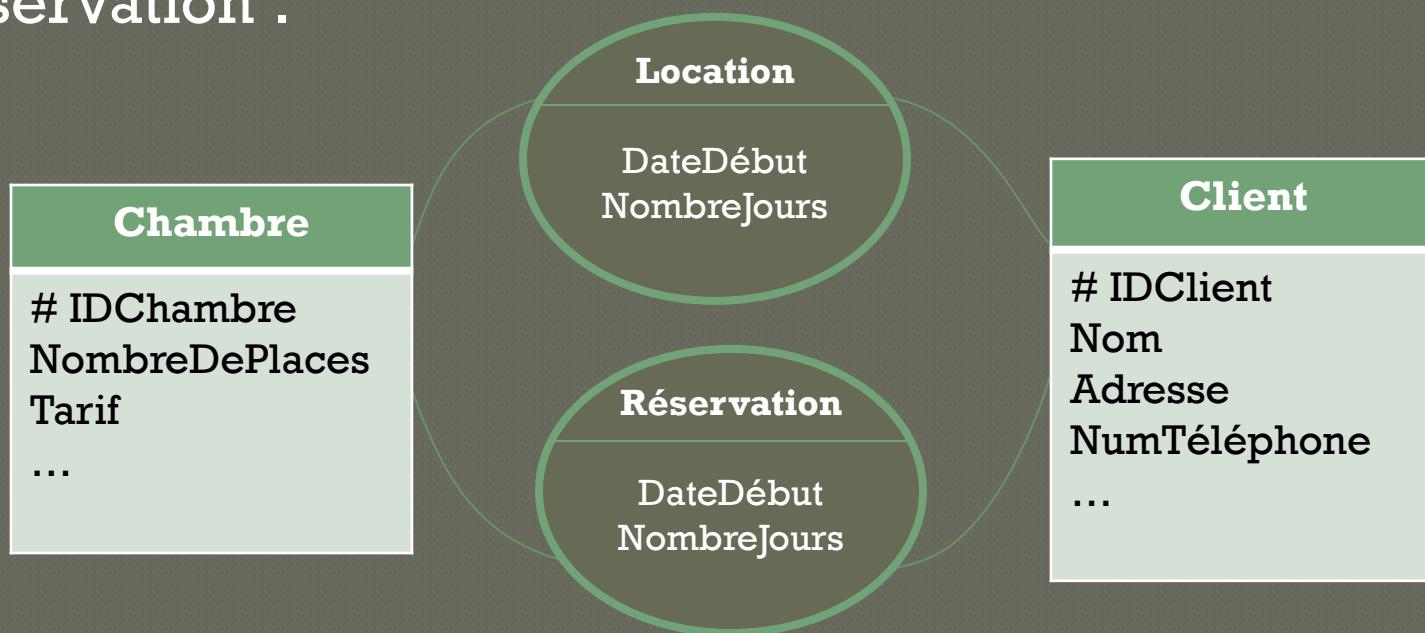
- Une **association** permet de représenter les liens existant entre les entités. Elle peut éventuellement contenir des attributs. Ce qui veut dire que l'identifiant n'est pas nécessaire.
- Les entités sont dites **binaires** lorsqu'elles sont associées par deux. Si c'est plus de deux, on parle de **n-aires**.
- Une association est représentée par un ovale contenant son nom et éventuellement ses attributs.



Le modèle conceptuel

Modélisation > Associations

- Deux entités peuvent être liées par plusieurs associations. C'est le cas entre 'client' et 'chambre' qui sont reliées par deux associations, à savoir 'location' et 'réservation'.



Le modèle conceptuel

Modélisation > Associations

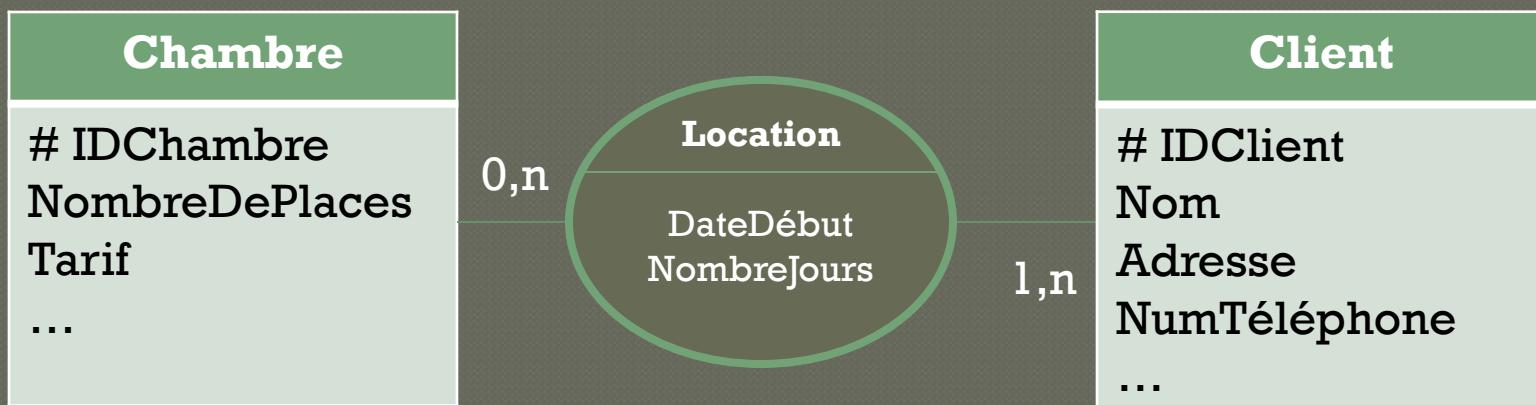
- Une association est dite **réflexive**, si elle relie une même entité. Par exemple, si je prends le cas d'un mariage, "Massamba est marié à Mafatou" , l'entité 'personne' est associée à elle-même par l'association 'est_marié_à'.



Le modèle conceptuel

Modélisation > Cardinalités

- Les **cardinalités** décrivent les caractéristiques de l'association entre les entités. C'est un couple de valeurs minimales et maximales permettant de caractériser l'association. Ces nombres modélisent le nombre d'occurrences minimales et maximales des entités impliquées dans l'association. Chaque entité participe de manière différente à l'association.



- Une chambre peut être louée plusieurs fois (n) et elle peut ne pas être occupée (0).
- Un client loue au minimum une (1) chambre et il peut en louer plusieurs (n).

Le modèle conceptuel

Modélisation > Cardinalités

Les cardinalités peuvent prendre les valeurs suivantes:

- De un à un, notée 1,1. Exemple: une brosse à dents possède en théorie un (1) et un (1) seul propriétaire.
- De un à plusieurs, notée 1,n. Exemple: un livre a au moins un (1) auteur ; il peut en posséder plusieurs (n).
- Optionnel, notée 0,1. Exemple: une personne (chrétienne) est célibataire (0) ou mariée (légalement...) à une (1) autre personne au plus.
- De zéro à plusieurs, notée 0,n. Exemple: un appartement peut être libre (0) ou habité éventuellement par plusieurs habitants (n).

Le modèle conceptuel

Modélisation > Cas pratique

On envisage de modéliser la vente d'une voiture.

- On peut facilement identifier deux entités du monde réel que sont client et voiture.
- Une voiture est caractérisée par sa marque, son type, sa couleur.
- Un client (ou une personne) est caractérisé par son nom, son âge, sa ville, son sexe.
- Un client peut acheter plusieurs voitures ou aucune.
- Un client peut acheter aucune ou plusieurs voitures.
- Une voiture peut être vendue ou non.

Le modèle conceptuel

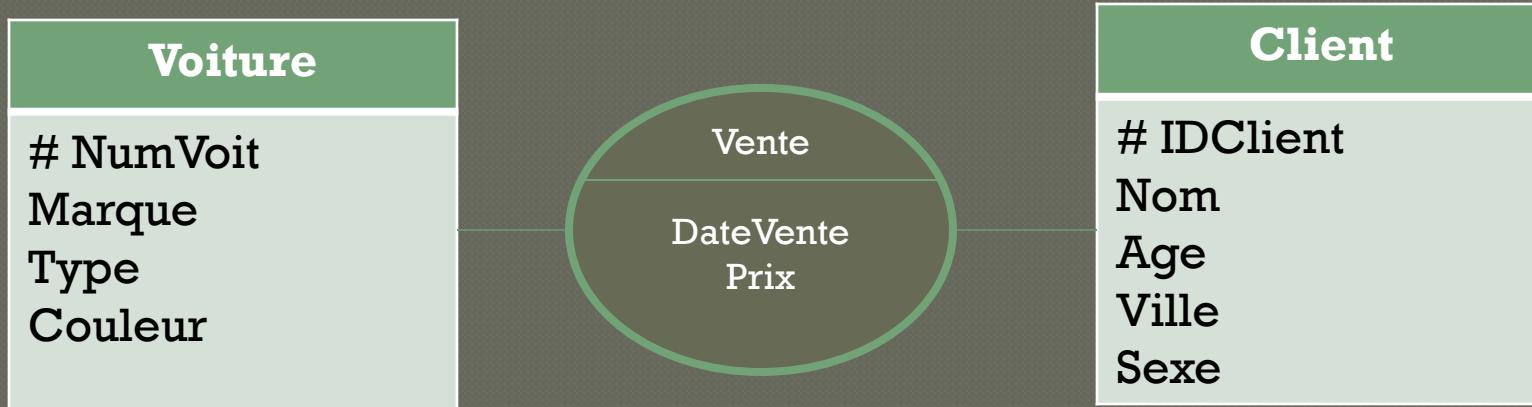
Modélisation > Cas pratique

- Aucun attribut ni ensemble d'attributs ne permet de définir sans ambiguïté un identifiant. Les identifiants doivent alors être introduits arbitrairement.
- Les cardinalités:
 - Un client peut acheter plusieurs (n) voitures ou aucune (0): 0,n
 - Une voiture peut être vendue une seule fois (1) ou jamais (0): 0,1

Le modèle conceptuel

Modélisation > Cas pratique

- Ci-dessous, le modèle entité-association:



Le modèle conceptuel

Raffinage > Qualité des attributs

Quelques règles simples pour permettre de guider le choix des attributs.

- Il faut stocker que les attributs strictement nécessaires. Ce choix s'effectue en tenant compte des fonctionnalités attendues du système d'information par élimination systématique des données inutiles. Les personnes ont tendance de prévoir le plus d'attributs possible juste au cas où. Ce qui n'est pas nécessaire ici.
- Il ne faut jamais retenir les attributs qui peuvent être déduits soit par calcul, soit par un lien sémantique d'autres attributs.

Exemple:

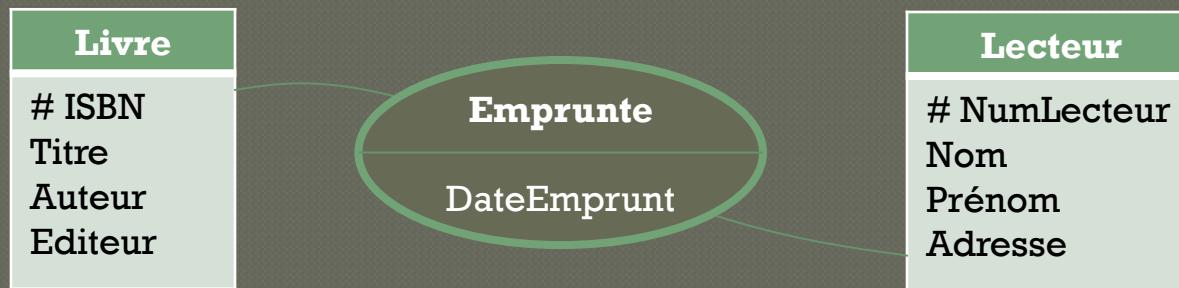
- Un chiffre d'affaire peut être calculé à partir du prix de vente et de la quantité.
 - Une référence peut être construite à partir du nom du produit et du fournisseur.
- Le nom des attributs doit être parlant. Dans ce cas, il ne faut pas hésiter d'utiliser des noms longs si cela facilite la compréhension.
Par exemple, on peut préférer écrire 'Numero_Serie' plutôt d'écrire 'ns'.

Le modèle conceptuel

Raffinage > Réorganisation des entités

Plutôt de découper en entités par une première approche, mieux vaut procéder par une approche itérative. Prenons l'exemple du "*lecteur qui empruntait un livre à la bibliothèque*". Les deux entités déduites sont '*lecteur*' et '*livre*' qui sont liées par l'association '*emprunte*'.

- Un lecteur est caractérisé par son nom, son prénom et son adresse. Comme il n'y a pas d'attribut possédant les caractéristiques d'un identifiant pour l'entité 'lecteur', on ajoute un attribut identifiant (le numéro du lecteur).
- Un livre est caractérisé par son titre, son auteur, son numéro ISBN, son éditeur. Le numéro ISBN est ici un identifiant pour l'entité puisqu'il est unique.
- L'association 'emprunte' a pour attribut la date d'emprunt.



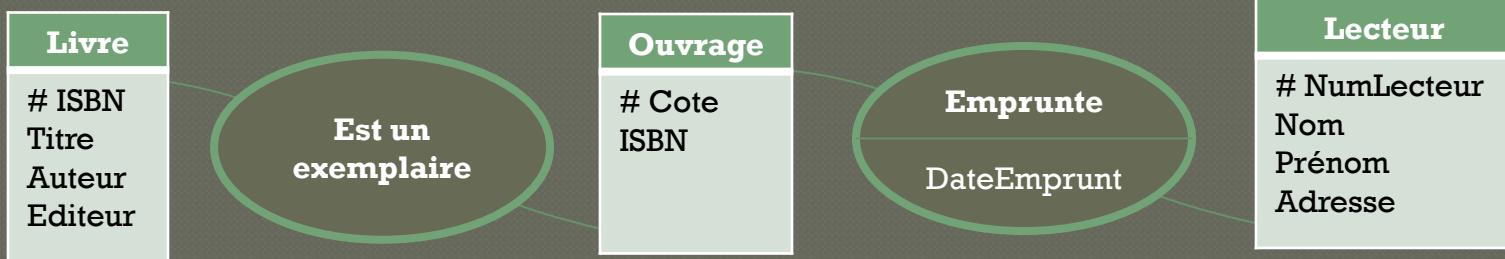
Cette illustration simple permet de se poser quelques questions qui vont conduire à une réorganisation du modèle.

Le modèle conceptuel

Raffinage > Réorganisation des entités

Que se passe-t-il si l'on possède plusieurs exemplaires du même ouvrage ?

- Le numéro ISBN n'est plus identifiant puisqu'il est le même pour chacun des exemplaires. Une solution consiste à ajouter un numéro supplémentaire unique pour chaque livre, qui correspond à la notion de cote dans une bibliothèque. Ainsi, même si l'on a dix exemplaires d'un ouvrage, il est possible de les différencier.
 - L'inconvénient de cette solution est que l'on répète les informations communes aux différents ouvrages (titre, auteur...) à chaque exemplaire.
 - L'un des risques est de répéter incorrectement ces informations et d'aboutir ainsi à des incohérences.
- La solution correcte dans ce cas est de séparer l'entité 'livre' en deux entités 'livre' et 'ouvrage'. L'activité de la bibliothèque est alors décrite par deux phrases :
 - Un lecteur emprunte un exemplaire.
 - Un livre représente un exemplaire d'un ouvrage.



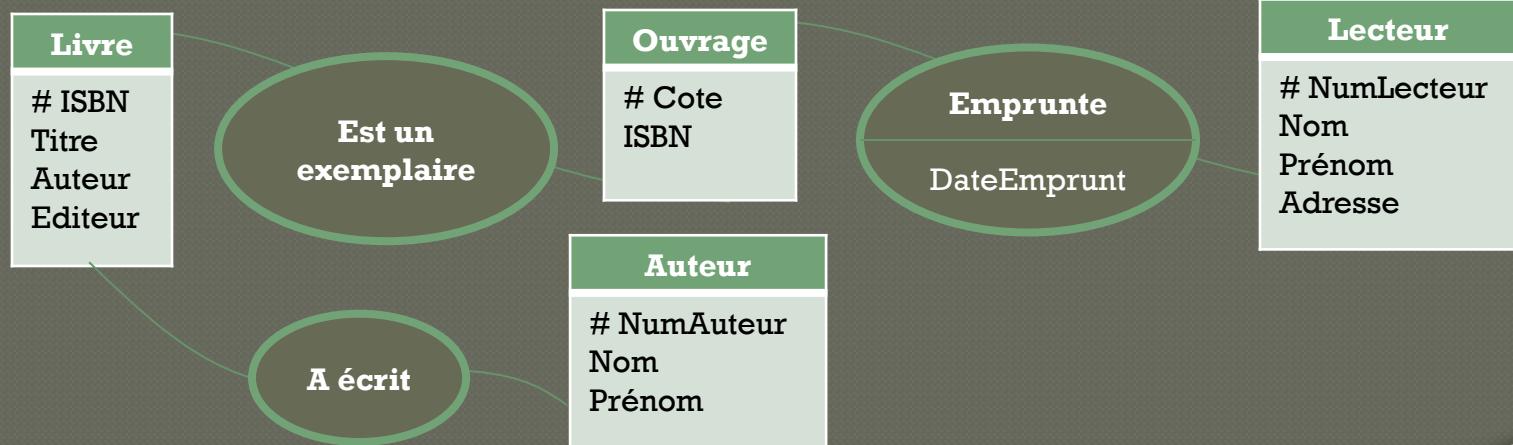
Un ouvrage est caractérisé par un numéro (cote) identifiant et un ISBN. Un livre est caractérisé par un ISBN, qui est bien dans ce cas un identifiant, un titre, un auteur et un éditeur. L'ouvrage est un regroupement d'attributs qui n'a pas d'existence dans le monde réel.

Le modèle conceptuel

Raffinage > Réorganisation des entités

Que se passe-t-il si l'ouvrage possède plusieurs auteurs ?

- Une solution simpliste est de prévoir un champ par auteur supplémentaire, c'est-à-dire d'ajouter des champs 'auteur2', 'auteur3', 'auteur4', etc. Cette solution pose de nombreux problèmes :
 - Si seulement dix livres sur un million possèdent plusieurs auteurs, on réserve la place pour les champs auteurs supplémentaires qui sera inutilisée.
 - Si un livre possède un nombre d'auteurs supérieur au nombre de champs prévus, on ne résout pas le problème.
 - Si l'on considère qu'un auteur peut avoir écrit plusieurs ouvrages, on répète dans ce cas les informations le concernant pour chacun de ses ouvrages. Cela constitue un cas typique de redondance qui risque de provoquer des incohérences.
- La solution correcte dans ce cas est de créer une entité supplémentaire pour ces attributs qui sont sémantiquement de même type. On créera ici une entité auteur qui contiendra un numéro d'auteur (identifiant), son nom et son prénom. Cette entité est reliée à l'entité ouvrage par l'association 'a_écrit'.

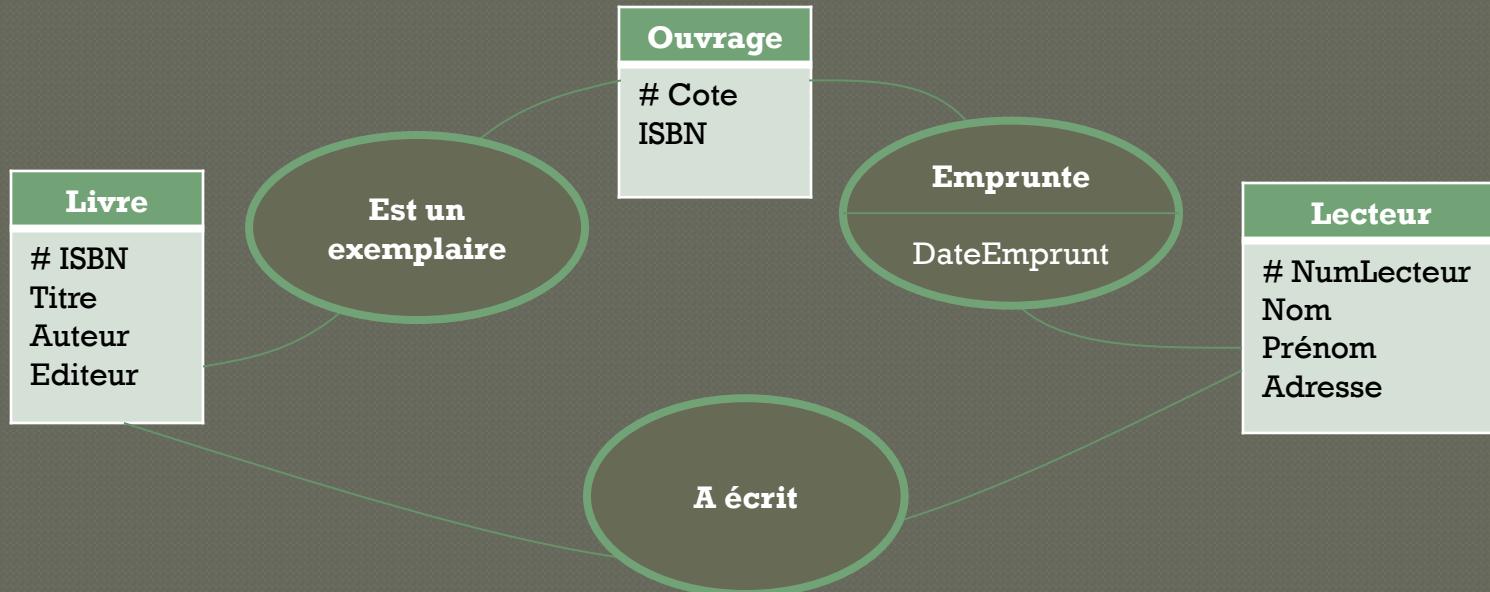


Le modèle conceptuel

Raffinage > Réorganisation des entités

Que se passe-t-il si un auteur emprunte un livre ?

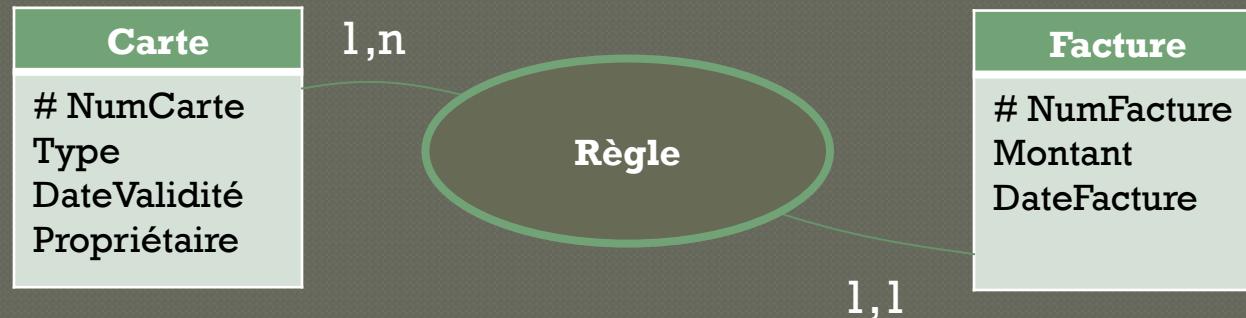
- Un auteur peut également emprunter un livre et revêtir par conséquent le rôle de l'emprunteur. Dans ce cas, on répète les informations le concernant dans l'entité 'lecteur'. Cela provoque de la redondance et peut générer des incohérences.
- Comme les entités 'lecteur' et 'auteur' ont la même structure, la solution consiste à les fusionner en une entité unique 'personne'.



Le modèle conceptuel

Raffinage > Elimination d'associations

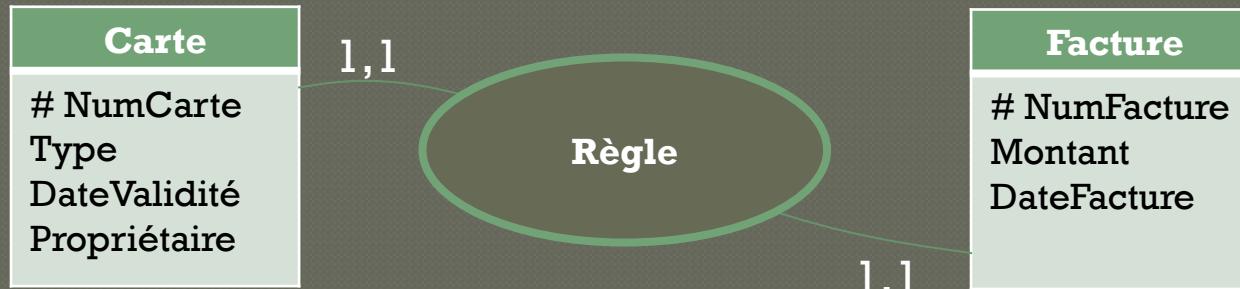
- Si on considère le cas d'un acte d'achat effectué sur Internet avec une carte bancaire. "*Une facture est réglée par une carte*". On peut distinguer les deux entités '*facture*' et '*carte*' .
 - Une facture est identifiée par un numéro de facture, et est constituée d'un montant et d'une date.
 - Une carte bancaire est identifiée par son numéro, son type (Visa, MasterCard), sa date de validité et son propriétaire.
 - Une facture est payée par une et une seule carte (1,1).
 - Une carte peut servir à régler plusieurs factures (1,n).



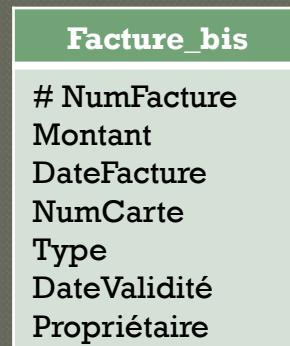
Le modèle conceptuel

Raffinage > Elimination d'associations

- On peut utiliser une E-carte qui permet d'améliorer la sécurité de ces transactions. Une E-carte est une carte "virtuelle" associée à une véritable carte bancaire, valable pour une seule transaction.
 - Une E-carte ne permet de régler qu'une et une seule facture (1,1).



- Dans ce cas, 'règle' n'a plus lieu d'être puisqu'il s'agit d'une pure **bijection**.
 - À une facture correspond une E-carte et une seule, et à une E-carte correspond une facture et une seule.
- On peut donc fusionner les deux entités 'carte' et 'facture' et éliminer l'association

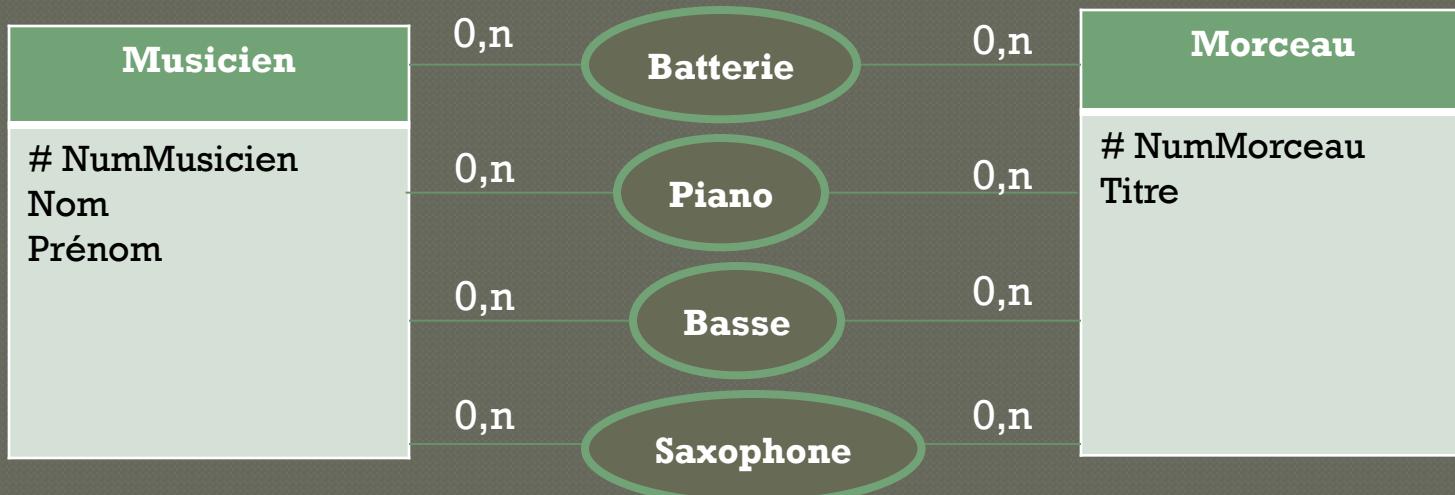


Le modèle conceptuel

Raffinage > Fusion d'associations

Suivant le même principe, il est possible de fusionner plusieurs associations ayant le même rôle sémantique.

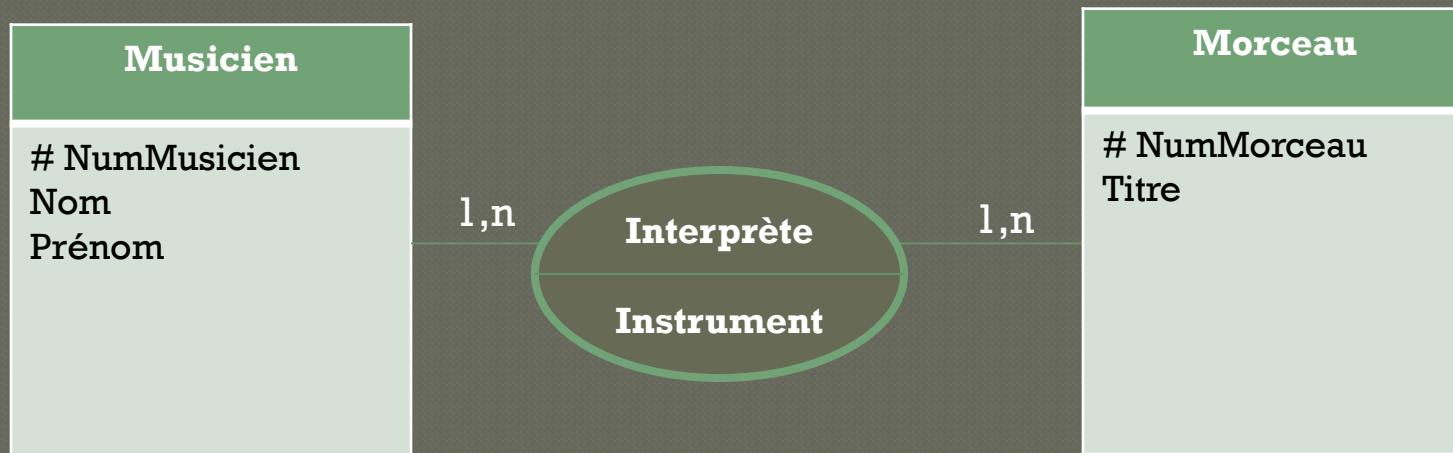
- Si l'on considère la description de l'activité suivante, liée à l'exécution de morceaux de jazz en quartet. Pour un morceau donné, le premier musicien joue la partie de basse, le deuxième celle de batterie, le troisième celle de piano et le quatrième celle de saxophone. Cela donne un schéma du genre:



Le modèle conceptuel

Raffinage > Fusion d'associations

- Comme il s'agit d'une même activité, "un musicien interprète un morceau", on peut remplacer toutes les associations par une seule 'interprète' contenant l'attribut 'instrument'.



Le modèle conceptuel

Résumé

- Le but de ce chapitre consiste d'abord à extraire les informations du monde réel puis à en proposer une représentation formelle appelée modèle conceptuel.
 - Pour parvenir à cela, il faut d'abord identifier les données qui interviennent dans le domaine considéré et de les regrouper dans des objets.
 - Ensuite, caractériser les liens qui les unissent.
 - Enfin, on décrit le système à modéliser par des phrases simples de type "sujet-verbe-complément", que l'on peut classer en deux grandes catégories:
 - celles qui décrivent les objets du monde réel et les liens qui les unissent : le sujet et le complément sont représentés par des entités et le verbe est représenté par une association ;
 - celles qui décrivent la manière dont sont reliés ces objets : on en déduira les cardinalités des associations.
- Les entités sont constituées de champs de données que l'on nomme "attributs".
- Pour identifier de manière unique les représentants d'une entité, appelés également "instance", un attribut (ou un ensemble d'attributs) est choisi: on l'appelle "identifiant".

Le modèle conceptuel

Applications

● Enoncé 1:

On veut modéliser l'activité de vente de billets pour un théâtre. Quelles phrases vont nous permettre d'identifier les entités et la manière dont elles sont associées ?

- Proposez les attributs que vous utiliseriez pour décrire ces entités et leurs associations ainsi que les identifiants de chaque entité.
- Que se passe-t-il si le prix du billet varie pour chaque séance et en fonction de la place ?

Le modèle conceptuel

Applications

● Enoncé 2:

On veut représenter les liens de nourriture entre des humains, des animaux et des végétaux. L'idée, à partir des schémas d'alimentation modélisés, est de pouvoir déduire des chaînes alimentaires de ce type : "un homme mange un lapin qui mange des carottes".

Le modèle conceptuel

Applications

● Enoncé 3:

À partir de la base de données exemple de vente de voitures, on souhaite ajouter les informations concernant le vendeur qui a réalisé la vente. Proposez une (ou plusieurs) modification(s) du modèle entité-association élaboré précédemment. Ajoutez les nouvelles cardinalités introduites par cette modification.

PLAN

- Généralités
- Le modèle conceptuel
- Le modèle relationnel
- Le langage SQL
- Etude et réalisation d'une base de données
- La sécurité des données

Objectif

- Dans ce chapitre, nous présentons le concept de relation fondamental du modèle relationnel, ainsi que les opérations qui lui sont associées.
- Ensuite, nous aborderons les méthodes qui permettent de passer du modèle conceptuel à un ensemble de relations.
- Enfin, nous présenterons quelques méthodes de résolution les incohérences ainsi que la forme normale de Boyce-Codd.

Concept de base

- Le modèle relationnel tire son nom de la notion de relation (mathématique) entre des éléments. Chacun de ces éléments peut prendre des valeurs dans un ensemble défini.
 - Prenons le cas d'une cuisine constituée d'**appareils** (réfrigérateur, cuisinière, hotte, robot, lave-vaisselle) de diverses **couleurs** (rouge, bleu, vert, jaune, blanc, noir, rose, jaune).
 - On pourra donc avoir des couples de valeurs suivantes:
 - (réfrigérateur, rouge)
 - (robot,mauve)
 - (cuisinière,jaune)
 - (lave-vaisselle,rouge)
 - Cet ensemble de couples de valeurs liées entre elles, nommé **tuples**, ou «nuplets» ou «enregistrements» dans le modèle relationnel, représente la relation entre les éléments 'appareil' et 'couleur'.
 - On désigne également les éléments constitutifs de ces couples par les termes «attributs» ou «champs».

Concept de base

- La relation peut être écrite formellement de la manière suivante : *ma_cuisine(appareil, couleur)*.
 - Cette écriture représente le schéma relationnel de la relation 'ma_cuisine'.
 - Les valeurs énoncées précédemment pour les champs représentent leurs domaines, c'est-à-dire les ensembles de toutes les valeurs possibles pour un champ.
- Une relation est décrite par:
 - le schéma relationnel ;
 - les domaines des différents champs ;
 - les tuples qui la constituent.
- Le nombre de champs de la relation s'appelle son degré de la relation. La relation 'ma_cuisine' est de degré 2.
- Le nombre de tuples se nomme la cardinalité de la relation. La relation 'ma_cuisine' est de cardinalité 4.

Concept de base

- On représente une **relation** par une **table**, correspondant à la notion de tableau.
 - Les **tuples** correspondent aux **lignes** et les **colonnes** aux **champs** de la relation.

Appareil	Couleur
Réfrigérateur	Rouge
Robot	Mauve
Cuisinière	Jaune
Lave-vaisselle	Rouge

- Dans le **modèle relationnel**, la **relation** est l'élément fondamental.
 - Toutes les opérations sur une ou plusieurs relations retourneront une relation.
 - Un ensemble de relations reliées entre elles par des liens sémantiques constitue une **base de données**.

Concept de base

NOTION DE CLÉ D'UNE RELATION ET DÉPENDANCE FONCTIONNELLE

- ◎ Dans une **base de données**, pour accéder à un enregistrement par le contenu d'un ou de plusieurs champs il faut une **clé**.
 - On nomme **clé d'une relation**, un champ, ou un ensemble de champs, d'une relation qui permet d'identifier de manière *unique un enregistrement*.
 - Cette clé doit être minimale (doit contenir le minimum de champs) et est appelée **la clé primaire** de la relation.
 - Une relation peut comprendre plusieurs clés possibles. Dans ce cas, sont **les clés candidates**.

Concept de base

NOTION DE CLÉ D'UNE RELATION ET DÉPENDANCE FONCTIONNELLE

- Une clé ne peut être déduite simplement à partir du contenu de la relation; on ne peut préjuger du contenu futur des enregistrements.
 - **Par exemple**, si l'on prenait la relation 'ma_cuisine', le champ 'Appareil' semble être une clé puisqu'il contient une valeur unique pour chacun des enregistrements. **Cependant**, il est tout à fait possible que la cuisine considérée comprenne un autre réfrigérateur de couleur bleue, auquel cas la valeur ne serait plus unique et ne permettrait pas de retrouver l'enregistrement. **Dans ce cas de figure**, la combinaison 'Appareil' 'Couleur' pourrait sembler être une clé, mais on ne peut en être certain, compte tenu de l'**évolution des données**.

Concept de base

NOTION DE CLÉ D'UNE RELATION ET DÉPENDANCE FONCTIONNELLE

- Pour désigner une **clé primaire**, il faut donc également prendre en compte le «sens» des données dans la vie réelle. *Les relations qui existent entre les différents champs d'une relation vont être importantes:* on exprime ces relations à l'aide de **dépendances fonctionnelles**.
 - Une *dépendance fonctionnelle* existe entre deux ensembles de champs si les valeurs contenues dans l'un des ensembles de champs permettent de déterminer les valeurs contenues dans l'autre ensemble.

Concept de base

NOTION DE CLÉ D'UNE RELATION ET DÉPENDANCE FONCTIONNELLE

- Si on considère l'exemple suivant, correspondant à la relation Lecteur(Numero_carte, Nom, Age, Ville, Etablissement), modélisant les lecteurs d'une bibliothèque.

Numero_carte	Nom	Age	Ville	Etablissement
1	Henri	10	Paris	Université Sorbonne
2	Stanislas	34	Paris	Université Jussieu
3	Henriette	44	Lyon	CHU Bron
4	Dominique	19	Nancy	Université Poincaré
5	Isabelle	56	Nancy	INPL
6	Olivier	51	Marseille	Université Saint Charles
7	Henri	98	Paris	Université Sorbonne
8	Jerome	23	Nancy	INPL
9	Laurence	34	Bordeaux	Université Victor Segalen
10	Christian	41	Paris	Ecole Normale Supérieure
11	Antoine	16	Marseille	Université Saint Charles
12	Laurence	34	Paris	Université Jussieu

- En **examinant les données**, on constate qu'il ne peut y avoir dépendances fonctionnelles puisque qu'il existe un enregistrement pour lequel les valeurs des champs (Nom, Age) correspondent à deux valeurs différentes de (Ville, Etablissement).

Concept de base

NOTION DE CLÉ D'UNE RELATION ET DÉPENDANCE FONCTIONNELLE

- Si on considère que dans une ville, il ne peut y avoir qu'un établissement et un seul, cela signifiera alors qu'il existe une relation de dépendance entre les champs '**Etablissement**' et '**Ville**'. A une valeur donnée de '**Etablissement**' correspond bien une valeur unique de '**Ville**'.
- On constate bien que la valeur du champ '**Numero_carte**' est unique pour chacune des personnes et ses valeurs sont identifiantes pour tous les autres champs de la relation c'est-à-dire, chaque champ dépend fonctionnellement du champ '**Numero_carte**'.
 - Ses valeurs sont uniques et jamais vides : c'est une clé candidate.
 - C'est d'ailleurs la seule clé possible car les autres champs n'ont jamais de valeur unique.
 - Il est, dans ce cas, choisi comme **clé primaire** de la relation.

Concept de base

COHÉRENCE DES DONNÉES ET CONTRAINTES D'INTÉGRITÉ

Pour garantir la cohérence des données sur l'ensemble des relations constitutives de la base de données, on applique des restrictions sur le contenu des données que l'on nomme **contrainte d'intégrité**. La vérification de la cohérence se situe à plusieurs niveaux :

- Adapter le contenu des données:

On parle de la cohérence par rapport au sens des données qui est liée à la notion de domaine de valeurs du champs (on parle de **contrainte de domaine** ou de **cohérence sémantique**). *Par exemple, l'âge d'une personne ne peut être négatif ou excéder 120. Ces contraintes sont définies lors de l'étape d'analyse du monde réel et leur mise en œuvre effective interviendra au moment de la création de la relation et sera réalisée par le SGBD.*

Concept de base

COHÉRENCE DES DONNÉES ET CONTRAINTES D'INTÉGRITÉ

□ Préserver la cohérence des données:

Comme une base de données est constituée par un ensemble de relations reliées entre elles, les contenus des champs capables de lier ces relations doivent être cohérents entre eux pour pouvoir effectuer l'opération de jointure.

Si l'on considère l'exemple de la base de données 'casse', on ne doit pas permettre la saisie d'une valeur identifiant une voiture dans la relation 'vente' qui n'existe pas dans la relation 'voiture'.

- On fait dans ce cas référence au contenu d'une colonne d'une autre relation (le champ 'NumVoit' de la relation 'voiture' afin de contrôler le contenu du champ 'NumVoit' de la relation 'vente').
- Le champ 'NumVoit' est alors une **clé étrangère** qui permet de réaliser la notion **d'intégrité référentielle**.

ces références sont aussi déterminées lors de la phase d'analyse du monde réel et le SGBD permet de les réaliser.

ces contraintes sont généralement appliquées non pas lors de la création des relations, mais plutôt lorsque les données ont déjà été insérées, en particulier dans les relations de références afin d'éviter des problèmes de références impossibles à résoudre entre les relations, ce qui provoque parfois l'incapacité à insérer des données.

Concept de base

COHÉRENCE DES DONNÉES ET CONTRAINTES D'INTÉGRITÉ

□ Éliminer la redondance des données:

Les incohérences provoquées par la redondance d'information représentent le principal souci du concepteur d'une base de données. En effet, lorsque les données sont dupliquées, aucun mécanisme ne peut garantir que le changement de la valeur d'une donnée est répercuté correctement sur les autres données. Dans *l'exemple de la relation des lecteurs de bibliothèque*, on remarque **la redondance d'information** qui existe entre les champs 'Ville' et 'Etablissement'. Ici, si le nom de l'établissement 'INPL' change, il faut mettre à jour toutes les lignes qui contiennent son nom.

- La redondance est mise en évidence par la dépendance fonctionnelle qui existe entre ces deux champs.
- Si dans cet exemple, la détection de la redondance est évident, il ne sera pas toujours le cas lorsque le nombre de relations est élevé ou encore si le sujet modélisé par la base de données n'est pas familier.
- Les incohérences de ce type seront résolues par la réorganisation des relations lors de la phase de **normalisation**.

Les opérations

ENSEMBLISTES

Pour pouvoir manipuler les données dans le modèle relationnel, on fait appel aux opérations formelles qui sont pour la plupart issues de la théorie des ensembles. Parmi lesquelles, nous avons:

- Les opérations ensemblistes

Ces opérations sont utilisés que pour des relations ayant la même structure ou pour des structures compatibles, à l'exception du produit cartésien. Elles sont de type binaire(appliquant à deux relations seulement).

- L'union

L'opération d'union consiste en la mise en commun des enregistrements de chaque relation en intégrant en une seule fois ceux qui sont identiques. Elle est représentée par le caractère « \cup » et sert typiquement à la « consolidation » de données de même type provenant de différentes sources.

Les opérations

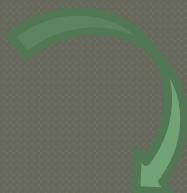
ENSEMBLISTES

- L'exemple ci-dessous, illustre cette opération:

Lecteur_1(Numero_carte, Nom, Age, Ville, Etablissement)				
Numero_carte	Nom	Age	Ville	Etablissement
1	Henri	10	Paris	Université Sorbonne
2	Stanislas	34	Paris	Université Jussieu
3	Henriette	44	Lyon	CHU Bron

Lecteur_2(Numero_carte, Nom, Age, Ville, Etablissement)

Numero_carte	Nom	Age	Ville	Etablissement
3	Henriette	44	Lyon	CHU Bron
4	Dominique	19	Nancy	Université Poincaré
5	Isabelle	50	Nancy	INPL



Lecteur_1 ∪ Lecteur_2

Numero_carte	Nom	Age	Ville	Etablissement
1	Henri	10	Paris	Université Sorbonne
2	Stanislas	34	Paris	Université Jussieu
3	Henriette	44	Lyon	CHU Bron
4	Dominique	19	Nancy	Université Poincaré
5	Isabelle	50	Nancy	INPL

Numero_carte	Nom	Age	Ville	Etablissement
2	Henri	10	Paris	Université Sorbonne
4	Stanislas	34	Paris	Université Jussieu

Les opérations

ENSEMBLISTES

➤ La différence

L'opération différence consiste à désigner les enregistrements qui appartiennent à une relation sans appartenir à l'autre. Elle est représentée par le caractère « - », sert d'éliminer des enregistrements d'une relation par rapport à une liste et n'est pas symétrique.

Lecteur_1 – Lecteur_2

Numero_carte	Nom	Age	Ville	Etablissement
1	Henri	10	Paris	Université Sorbonne
2	Stanislas	34	Paris	Université Jussieu

Les opérations

ENSEMBLISTES

➤ L'intersection

L'opération intersection désigne les enregistrements qui sont communs aux deux relations. Elle est représentée par le caractère « \cap » et sert à trouver les éléments communs à deux relations.

Lecteur_1 \cap Lecteur_2

Numero_carte	Nom	Age	Ville	Etablissement
3	Henriette	44	Lyon	CHU Bron

Les opérations

ENSEMBLISTES

➤ Le produit cartésien

Le produit cartésien permet la combinaison des enregistrements de deux relations sans tenir aucun compte du contenu des données. Les relations n'ont donc pas besoin d'avoir la même structure. Le caractère représentant le produit cartésien est « X ».

ma_cuisine(Appareil, Couleur)	
Appareil	Couleur
Réfrigérateur	rouge
Robot	mauve
Cuisinière	jaune

musicien(Nom, Instrument)

Nom	Instrument
Jaco Pastorius	Basse électrique
Bill Evans	Piano



Le_resultat(Appareil, Couleur, Nom, Instrument) = ma_cuisine(Appareil, Couleur) X
musicien(Nom, Instrument)

Appareil	Couleur	Nom	Instrument
réfrigérateur	rouge	Jaco Pastorius	Basse électrique
réfrigérateur	rouge	Bill Evans	Piano
robot	mauve	Jaco Pastorius	Basse électrique
robot	mauve	Bill Evans	Piano
cuisinière	jaune	Jaco Pastorius	Basse électrique
cuisinière	jaune	Bill Evans	Piano

Les opérations

RELATIONNELLES

- Les opérations relationnelles
 - La projection

La projection consiste à extraire certains champs de la relation, ce qui donne à cette dernière un degré inférieur à la relation de départ. Par exemple, sur la relation ‘Lecteur’, on peut projeter les champs ‘Nom’ et ‘Ville’ et on obtient la relation suivante:

Lecteur_proj(Nom, Ville)

Nom	Ville
Henri	Paris
Stanislas	Paris
...	...

Les opérations

RELATIONNELLES

- Les opérations relationnelles
 - La sélection ou restriction

La sélection consiste à extraire les enregistrements de la relation en utilisant des critères pour les caractériser (les enregistrements sélectionnés). La structure de la relation résultat est la même que celle de la relation de départ. Par exemple, sur la relation ‘Lecteur’, on peut effectuer une sélection des enregistrements dont le contenu du champs ‘Ville’ vaut ‘Marseille’.

Lecteur_sel(Numero_carte, Nom, Age, Ville, Etablissement)

Numero_carte	Nom	Age	Ville	Etablissement
6	Olivier	51	Marseille	Université Saint Charles
11	Antoine	16	Marseille	Université Saint Charles

Les opérations

RELATIONNELLES

➤ La Jointure

La jointure permet d'exprimer le sens du lien existant entre les relations dans le monde réel. La liaison s'effectue par le contenu commun d'un champ. Par exemple, si on considère les deux relations Lecteur_bis(Numéro_carte, Nom, Num_Etablissement) et Etablissement(Num_Etablissement, Ville, Nom_Etablissement):

Lecteur_bis(Numéro_carte, Nom, Num_Etablissement)		
Numéro_carte	Nom	Num_Etablissement
1	Henri	1
2	Stanislas	2
3	Henriette	1



Etablissement(Num_Etablissement, Ville, Nom_Etablissement)		
Num_Etablissement	Ville	Nom_Etablissement
1	Paris	Université Jussieu
2	Lyon	CHU Bron
3	Nancy	Université Poincaré
4	Paris	Université Sorbonne

Lecteur_joint_Etablissement(Numéro_carte, Nom, Num_Etablissement_1, Num_Etablissement_2, Ville, Nom_Etablissement)

Numéro_carte	Nom	Num_Etablissement_1	Num_Etablissement_2	Ville	Nom_Etablissement
1	Henri	1	1	Paris	Université Jussieu
2	Stanislas	2	2	Lyon	CHU Bron
3	Henriette	1	1	Paris	Université Jussieu

jointure naturelle

Lecteur_jointnat_Etablissement(Numéro_carte, Nom, Num_Etablissement, Ville, Nom_Etablissement)

Numéro_carte	Nom	Num_Etablissement	Ville	Nom_Etablissement
1	Henri	1	Paris	Université Jussieu
2	Stanislas	2	Lyon	CHU Bron
3	Henriette	1	Paris	Université Jussieu

Les opérations

RELATIONNELLES

➤ La Jointure

L'opération de jointure peut être vue comme une sélection des enregistrements obtenus par le produit cartésien des relations, dont les contenus du champ sur lequel on effectue la jointure sont égaux. On l'appelle dans ce cas une équijointure. Les champs dont on compare les contenus sont nommés **champs de jointure**.

Produit_cartesien_Etablissement(Numero_carte, Nom, Num_Etablissement_1, Num_Etablissement_2, Ville, Nom_Etablissement)					
Numero_carte	Nom	Num_Etablissement_1	Num_Etablissement	Ville	Num_Etablissement_2
1	Henri	1	1	Paris	Université Jussieu
1	Henri	1	2	Lyon	CHU Bron
1	Henri	1	3	Nancy	Université Poincaré
1	Henri	1	4	Paris	Université Sorbonne
2	Stanislas	2	1	Paris	Université Jussieu
2	Stanislas	2	2	Lyon	CHU Bron
2	Stanislas	2	3	Nancy	Université Poincaré
2	Stanislas	2	4	Paris	Université Sorbonne
3	Henriette	1	1	Paris	Université Jussieu
3	Henriette	1	2	Lyon	CHU Bron
3	Henriette	1	3	Nancy	Université Poincaré
3	Henriette	1	4	Paris	Université Sorbonne

Les opérations

RELATIONNELLES

➤ La Jointure externe

Il s'agit d'une opération de jointure étendue qui inclut dans le résultat les lignes n'ayant pas de correspondance sur le contenu du champ de jointure. Elle permet de répondre plus facilement à des questions du type « Quels sont les établissements qui n'ont pas de lecteurs ? ». Ici, on met en correspondance les valeurs du champ 'Num_Etablissement' de toutes les lignes de la relation 'Etablissement' avec celles de la relation 'Lecteur'.

Etablissement_jointext_Lecteur(Num_Etablissement_1, Ville, Nom_Etablissement,
Numero_carte, Nom, Num_Etablissement_2)

Num_Etablissement_1	Ville	Nom_Etablissement	Numero_carte	Nom	Num_Etablissement_2
1	Paris	Université Jussieu	1	Henri	1
2	Lyon	CHU Bron	2	Stanislas	2
3	Nancy	Université Poincaré	NULL	NULL	NULL
4	Paris	Université Sorbonne	NULL	NULL	NULL

Les opérations

Calculs et agrégats

○ Les Calculs et agrégats

En base de données, tout ce qui peut se calculer ne doit pas être stocké afin d'éviter des informations redondantes pouvant engendrer des incohérences et une perte de place. Des **fonctions de calculs** ont été définies afin de répondre à ce besoin. Par exemple, en considérant la relation La_boutique(Num_facture, Article, Prix, Quantite) suivante:

La_boutique(Num_Facture, Article, Prix, Quantite)

Num_Facture	Article	Prix	Quantite
101	Bananes	12	45
1034	Choux	5	129
2345	Riz	4	60
0987	Gazelle	15	48

Il est alors possible d'ajouter un champ 'Total', par exemple, dont le contenu sera calculé par l'expression 'Prix' * 'Quantité'.

La_boutique_total(Num_Facture, Article, Prix, Quantite, Total)

Num_Facture	Article	Prix	Quantite	Total
101	Bananes	12	45	540
1034	Choux	5	129	645
2345	Riz	4	60	240
0987	Gazelle	15	48	720

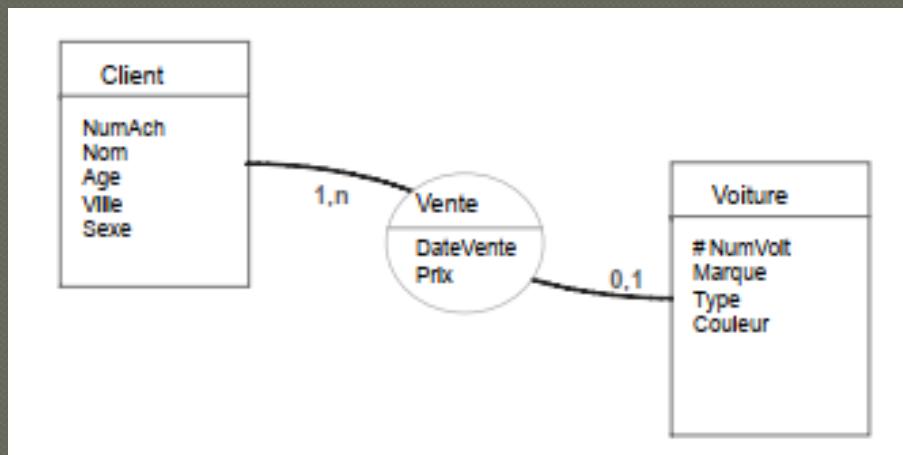
Passage du modèle conceptuel au relationnel

○ Règles générales

En général, nous avons que deux règles avec quelques exceptions que nous utilisons pour simplement d'obtenir un schéma plus compact.

- Une **entité** devient une relation composée des champs de l'entité. La clé de cette relation est la même que celle de l'entité.
- Une **association** devient une relation composée des deux clés des entités associées. S'il possède des champs, ils deviendront les champs de cette nouvelle relation.

Si on prenait l'exemple de la ‘casse’ (vente de voiture):



L'entité client devient:

`client(NumAch, Nom, Age, Ville, Sexe);`

Celle de voiture:

`voiture(NumVoit, Marque, Type, Couleur);`

Et l'association se transforme en relation aussi:

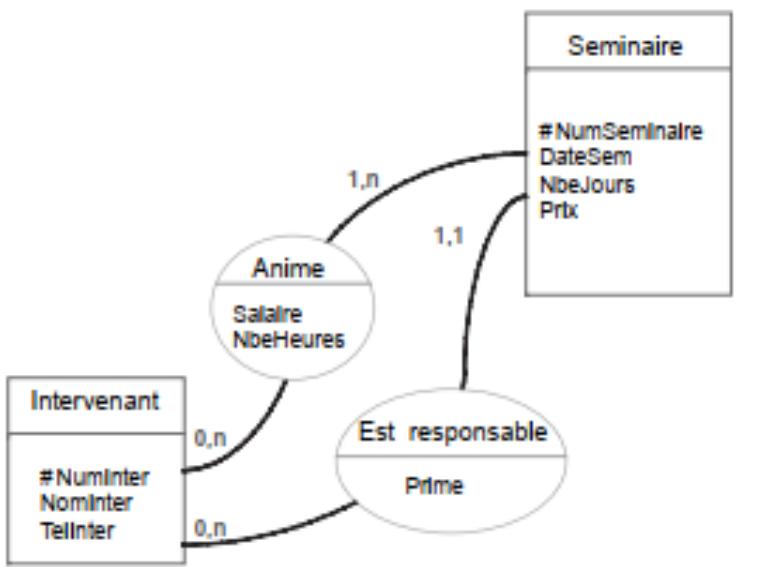
`vente(DateVent, Prix, NumAch, NumVoit).`

Passage du modèle conceptuel au relationnel

○ Cas particulier des associations de type ‘1-1’

Considérons le cas du modèle entité-association qui représente l’activité d’organisation de séminaires décrite très succinctement de la manière suivante :

- Un séminaire est animé par un ou plusieurs intervenants.
- Un séminaire ne possède qu’un seul responsable.



On constate l’existence de deux associations entre les entités ‘Séminaires’ et ‘Intervenant’ : ‘**Anime**’ et ‘**Est_responsable**’.

Si on applique les règles générales, on obtient quatre relations:

- Intervenant(NumInter, NomInter, TellInter)
- Séminaire(NumSem, DateSem, Prix, NbeJour)
- Anime(NumInter, NumSem, NbeHeure, SalaireHor)
- Est_responsable(NumInter, NumSem, Prime)

Passage du modèle conceptuel au relationnel

L'association 'Est_responsable' est de type '1-1' et donc la clé de la relation n'est pas minimal car l'identifiant du séminaire détermine celui du responsable c'est-à-dire qu'il existe une dépendance fonctionnelle entre les champs 'NumSem' et 'NumInter'.

On choisit le champ 'NumSem' comme clé de la relation 'Est_responsable', nous aurons alors: Est_responsable(NumSem, NumInter, Prime).

Dans ce cas, les relations 'Est_responsable' et 'Seminaire' ont alors la même clé, et donc, il va falloir les regrouper. On obtient la relation Seminaire_Res(NumSem, DateSem, Prix, NbeJour, NumInter, Prime).

De cette démarche, on peut en déduire une règle complémentaire:

- Lorsque l'association entre deux entités comprend une cardinalité de type '1-1', on ne crée pas de relation pour l'association. Les champs de l'association seront intégrés à la relation créée pour l'entité qui est associée avec la cardinalité '1-1'. La clé de cette relation est toujours celle de l'entité.

Normalisation

La normalisation permet de vérifier une certaines nombres de propriétés afin de tester la qualité et la cohérence des relations et d'apporter des modifications au cas échéant.

○ Première forme normale

La première forme normale s'intéresse au contenu des champs. Elle n'accepte pas la présence de plusieurs valeurs (multivaluation) dans un même champ. Celle-ci est, en effet, lié à la notion de dépendance fonctionnelle entre les champs qui ne peut plus être vérifiée s'ils possèdent plusieurs valeurs. Elle s'exprime de la manière suivante :

➤ *Tout champ contient une valeur atomique.*

Considérons l'exemple suivant:

Publication(NumPubli, Titre, Auteurs)

NumPubli	Titre	Auteurs
13490	Le vin et l'avenir	Jean Lasso, Hubert De la Tuque, Stanislas Wilski
21322	Bière et progrès social	Aristide Salem, Jean Lasso, Salome Dupont
45333	Le champagne et la France	Penelope Light, Vanessa Martinez, Salome Dupont

Le champ 'Auteurs' contient plusieurs valeurs:

- La solution est de décomposer cette relation en trois relations :
- Publication(NumPubli, Titre),
 - Auteur(NumAuteur, Nom, Prenom)
 - EstEcrite(NumPubli, NumAuteur).

Ces trois relations sont la représentation de la réalité «une publication est écrite par des auteurs». Elle se modélise par deux entités 'Publication' et 'Auteur' reliées par l'association 'EstEcrite',

Normalisation

Deuxième forme normale

○ Deuxième forme normale

Pour passer en deuxième forme il faut déjà satisfaire la première forme. Cette dernière sera chargée de rechercher la redondance d'information dans une relation. Elle interdit les dépendances fonctionnelles possibles entre les champs qui composent la clé et les autres champs. On peut l'exprimer de la manière suivante :

- *La relation est en première forme normale.*
- *Tout champ qui n'appartient pas à la clé ne dépend pas d'une partie de la clé.*

Considérons l'exemple suivant:

Produit(Article, Fournisseur, Adresse, Prix)

Article	Fournisseur	Adresse	Prix
Marteau	SOGENO	Paris	5
Tournevis	ARTIFACT	Lille	10
Tournevis	SOGENO	Paris	23
Pince	LEMEL	Paris	34
Mètre	ARTIFACT	Lille	24

La clé est constituée des champs ‘Article’ et ‘Fournisseur’. Or, il y a une relation de dépendance entre le champ ‘Fournisseur’, qui est une partie de la clé, et le champ ‘Adresse’.

- La solution consiste à décomposer la relation en deux. La nouvelle relation créée a pour clé la partie de la clé dont dépendent les autres champs qui constituent ses champs: *il s'agit du champ ‘Fournisseur’*. Les autres champs dépendants constituent le reste de la relation. *Il s'agit ici du champ ‘Adresse’.*

Produit(Article, Fournisseur, Prix)

Article	Fournisseur	Prix
Marteau	SOGENO	5
Tournevis	ARTIFACT	10
Tournevis	SOGENO	23
Pince	LEMEL	34
Mètre	ARTIFACT	24

Fournisseur(Fournisseur, Adresse)

Fournisseur	Adresse
SOGENO	Paris
ARTIFACT	Lille
LEMEL	Paris

Normalisation

○ Troisième forme normale

La troisième forme normale s'intéresse également la redondance d'information dans une relation. On recherche l'existence d'une dépendance entre deux champs qui ne font pas partie d'une clé. Elle n'accepte donc pas les dépendances fonctionnelles dites « transitives ». Elle s'exprime de la manière suivante :

- *La relation est en deuxième forme normale.*
- *Tout champ n'appartenant pas à une clé ne dépend pas d'un autre champ non clé.*

Considérons l'exemple suivant:

Baladeur(NumBal, Marque, Type, Couleur)

NumBal	Marque	Type	Couleur
12	Apple	Ipod	Blanc
43	Creative	Zen	Noir
23	Apple	Ipod	Noir
29	Creative	Zen	Gris
34	Sony	MZ-RH910	Rouge

Baladeur(NumBal, Type, Couleur)

NumBal	Type	Couleur
12	Ipod	Blanc
43	Zen	Noir
23	Ipod	Noir
29	Zen	Gris
34	MZ-RH910	Rouge

Il existe une dépendance fonctionnelle entre le champ 'Type' et le champ 'Marque'.

- La solution consiste à décomposer la relation en créant une nouvelle qui a pour clé le champ dont dépendent les autres champs constituant la dépendance transitive: *il s'agit dans ce cas du champ 'Type'*. Les autres champs de la nouvelle relation sont composés des champs qui en dépendent fonctionnellement : *ici, le champ 'Marque'*.

Baladeur_type(Type, Marque)

Type	Marque
Ipod	Apple
MZ-RH910	Sony
Zen	Creative

Normalisation

○ Forme normale de BOYCE-CODD

La forme normale de Boyce-Codd traite un cas un peu différent de ceux de la deuxième et troisième forme normale. Il s'agit du cas où une partie d'une clé (primaire ou secondaire) dépend d'un champ. Une relation en troisième forme normale n'est pas toujours en forme « Boyce-Codd », mais l'inverse est toujours vrai.

➢ *Tout champ appartenant à une clé ne dépend pas d'un autre champ non clé.*

Considérons l'exemple suivant:

La clé de cette relation est constituée par les champs 'Marque' et 'Produit'. Nous avons une relation de dépendance entre 'Marque' et 'Couleur'. La relation est en troisième forme normale, mais elle n'est pas en forme de Boyce-Codd.

□ Plusieurs décompositions sont possibles :

- Dictaphone(Marque, Produit, Prix) et Marque_coul(Couleur, Marque). **Mais cette décomposition génère des tuples non désirés au moment de la jointure.**
- Dictaphone(Produit, Prix, Couleur) et Marque_coul(Couleur, Marque). **Elle permet de reconstituer l'information de départ par une jointure sur le champ 'Couleur'**

Dictaphone(Marque, Produit, Prix, Couleur)

Marque	Produit	Prix	Couleur
Philips	LD 1024	49	Blanc
Olympus	VN 1664	49	Noir
Philips	LD 5047 H	59	Blanc
ImaginR	VN 1664	69	Gris
Olympus	VN 234 PC	79	Rouge

Le modèle relationnel

Applications

● Enoncé 4:

On considère deux relations. La première *Garage* est de degré 7 et de cardinalité 3. La seconde *Film* est de degré 2 et de cardinalité 15.

- Quels sont le degré et la cardinalité du produit cartésien de *Garage* par *Film* ?
- Quels sont le degré et la cardinalité du produit cartésien de *Film* par *Garage* ?

Le modèle relationnel

Applications

● Enoncé 5:

Quelle est la clé de cette relation Film(Prix, Format, Type, Nombre), avec:

Prix	Format	Type	Nombre
12	4 :3	Couleur	3
4	16 :9	Noir/Blanc	1
12	16 :9	Couleur	1664
35	4 :3	Noir/Blanc	890
12	16 :9	Noir/Blanc	1

Le modèle relationnel

Applications

● Enoncé 6:

Trouver le prix et le type de tous les films de la relation ‘Film’ vue précédemment. Peut-on en déduire que ‘Prix’ & ‘Type’ est une clé candidate, c'est-à-dire que toute combinaison des valeurs du prix et du type d'un film permet d'identifier un film ?

Le modèle relationnel

Applications

● Enoncé 7:

Donnez le prix des films de la base films en « Noir/Blanc ». Quels sont le degré et la cardinalité de la relation obtenue ? Est-il possible de calculer ces valeurs à l'avance, comme on le fait pour un produit cartésien ?

Le modèle relationnel

Applications

● Enoncé 8:

On considère ces deux relations:

- Film(Prix, Format, Type, Nombre, Numero_Film)

Prix	Format	Type	Nombre	Numero_Film
12	4/3	Couleur	3	2
4	16/9	Noir/Blanc	1	4
12	16/9	Couleur	1 664	50
35	4/3	Noir/Blanc	890	12
12	16/9	Noir/Blanc	1	12

- Catalogue(Numero_Film, Titre)

Numero_Film	Titre
2	<i>Le train qui passe</i>
4	<i>A toi !</i>
50	<i>Les chats du Sénégal</i>
111	<i>Le temps expliqué</i>
12	<i>Les impôts faciles</i>

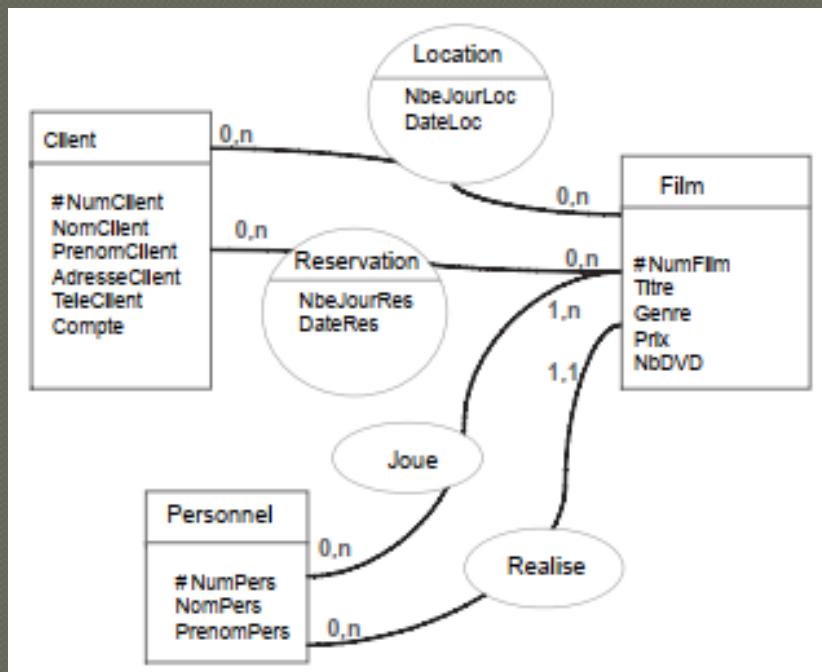
- Trouvez la liste des titres de films et leur format.
- Voyez-vous une incohérence dans le résultat ?
- Est-il possible de faire une jointure entre ces deux relations sur le champ ‘Prix’ de la relation ‘Film’ avec le champ ‘Numero_film’ de la relation ‘Catalogue’ ? Si oui, que signifie le résultat ?

Le modèle relationnel

Applications

Enoncé 9:

À partir du modèle entité-association modélisant une location de DVD, effectuez le passage au modèle relationnel.



Le modèle relationnel

Applications

● Enoncé 10:

Cette relation est-elle en première forme normale ?

- Personne(Nom, Adresse_mail, Poste)

Nom	Adresse_mail	Poste
André Dupont	adup@philips.com, andrc@dupond.fr	Directeur
Stanislas De la Motte	sdlm@versailles.mairie.fr	Employé, assernementé
Elisabeth Macroix	elsa@yago.to, mac@rien.fr	Assistante



Feedback

pape.abdoulaye.barro@gmail.com