

UNIVERSITÉ de Picardie Jules Verne

Systèmes d'information et bases de données

Licence Professionnelle ATC-MTS

Année Universitaire 2009-2010

UNIVERSITÉ de Picardie Jules Verne

Notions de Systèmes d'information

- Un système d'informations est chargé de véhiculer l'information
- Trois fonctions
 - Collecter les informations et mémoriser les données manipulées
 - Traiter ces données
 - Transmettre des informations à l'intérieur et à l'extérieur du système

☒ Les bases de données sont généralement l'élément central d'un système d'informations

UNIVERSITÉ de Picardie Jules Verne

Base de données ?

- Définition : ensemble de données modélisant les objets du mode réel et servant de support à une application informatique
- Critères :
 - Données accessibles par une communauté d'utilisateurs
 - Interrogeables par le contenu
 - Extraction des données que satisfont un critères (exemple : extraire tous les produits dont le prix est supérieur ou égal à 10€)
 - Possibilité de retrouver la structure des données
 - Par exemple, on doit pouvoir savoir qu'un produit est décrit par son nom, son prix, sa référence

3

UNIVERSITÉ de Picardie Jules Verne

SGBD ?

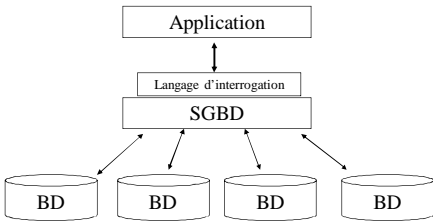
- Système de Gestion de Bases de Données
 - Outil informatique permettant la sauvegarde, l'interrogation, la recherche et la mise en forme de données stockées sur mémoire secondaire
 - ensemble de logiciels systèmes permettant aux utilisateurs d'INSERER MODIFIER RECHERCHER efficacement des données spécifiques dans une grande masse d'informations (pouvant atteindre quelques milliard d'octets) partagées par de multiples utilisateurs
- Peut gérer plusieurs bases de données

4

UNIVERSITÉ de Picardie Jules Verne

SGBD? (1/2)

- Un environnement base de données sera donc construit autour du SGBD de la manière suivante :



```

graph TD
    Application[Application] <--> LI[Langage d'interrogation]
    LI <--> SGBD[SGBD]
    SGBD --> BD1[(BD)]
    SGBD --> BD2[(BD)]
    SGBD --> BD3[(BD)]
    SGBD --> BD4[(BD)]
  
```

5

UNIVERSITÉ de Picardie Jules Verne

SGBD? (2/2)

- L'application est codée dans un langage de programmation particulier
- Elle va utiliser le langage d'interrogation proposé dans le SGBD avec lequel elle interagit
- Le SGBD va extraire les données requises sur la mémoire secondaire et les renvoyer dans la mémoire principale
- L'application récupère ces données dans des variables locales pour les manipuler
- Lorsqu'une application souhaite effectuer une série de modifications dans la base de données, elle le fait par le biais de transactions : sessions au cours desquelles différentes opérations d'E/S sont pratiquées sur la base de données

6

Objectifs d'un SGBD (1/2)

- Indépendance des programmes aux données
 - c'est en fait réaliser l'indépendance des structures de stockage par rapport aux structures des données du monde réel
 - Par exemple on doit pouvoir regrouper deux fichiers du disque sans avoir à modifier le schéma de la base
- Manipulation des données par des langages d'interrogation
 - ce sont généralement des langages basés sur des assertions logiques qui permettent de définir les objets désirés au moyen de relations entre les objets et de propriétés sur ces objets

7

Objectifs d'un SGBD (2/2)

- Administration facilitée des données
 - le SGBD doit fournir des outils permettant la description des données
- Efficacité des accès aux données
- Cohérence des données
- Partage des données
- Sécurité des données

8

Quelques S.G.B.D. répandus...

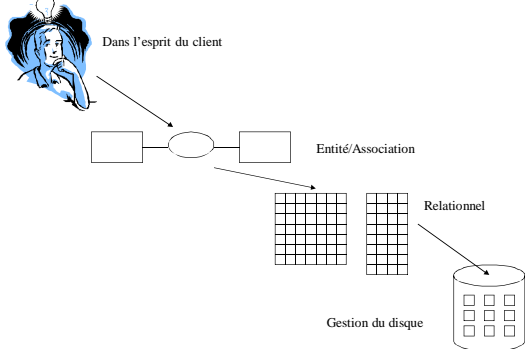
- Oracle (1981) : multi-plateforme (systèmes propriétaires et ouverts)
- DB2 (1983) : systèmes propriétaires IBM, UNIX, WINDOWS
- Sybase : UNIX
- Ingres, Informix, Unify : UNIX
- Universe (Pick (Siemens) / Unix)
- SQL/Server : WINDOWS
- PostgreSQL, MySQL, InstantDB : LINUX
- Access, Dbase, Paradox, FoxPro, Windev : WINDOWS

Quelques SGBD

- Les SGBD les plus répandus à l'heure actuelle sont les SGBD relationnels
 - Exemple : ACCESS, ORACLE, MySQL...
- Ils sont basés sur le modèle relationnel
 - Développé par Codd dans les années 70

10

Modèles de données



Qu'est-ce que le modèle relationnel?

- Modèle ensembliste simple
- Supporte des ensembles d'enregistrements au niveau de :
 - La description des données
 - On va avoir un ensemble de données représenté sous forme de « tables »
 - La manipulation des données
 - Lorsque l'on va extraire un sous-ensemble de ces données, on obtient à nouveau une représentation sous forme de « tables »
 - On a donc une grande cohérence dans la représentations des données.

12

Structures de base du modèle relationnel

- La **relation** : table de données caractérisée par un nom
 - VINS(NV, cru, millésime, degré, couleur)
- L'**attribut** : la colonne d'une table (i.e. la colonne d'une relation)
 - Ici les attributs sont NV, CRU, MILL, DEG et couleur
- Le **domaine** : le domaine de valeur que peut prendre une donnée d'une relation
 - le domaine de COULEUR = {Rosé, Blanc, Rouge }

13

Structures de base du modèle relationnel

- Syntaxe d'une relation selon le modèle relationnel :
Nom_relation (attribut1, attribut 2, ... , attribut n)
- Un enregistrement dans une relation est nommé : **t-uple**

14

Exemple

La relation CLIENT(NumCli, NomCli, PrenCli, Ville, Tel) peut comporter la table suivante :

CLIENT				
NumCli	NomCli	PrenCli	Ville	Tel
001	Dupont	Pierre	Amiens	0322222222
002	Durand	Paul	Paris	0140404040
003	Martin	Jacques	Lille	0320202020

15

Règles de base du modèle relationnel

- Unicité des clés :
 - Il ne peut exister de t-uples en double dans une relation. Pour identifier les t-uples sans donner toutes les valeurs : introduction de la notion de clé.
 - Définition d'une clé : Ensemble minimal d'attributs dont la connaissance des valeurs permet d'identifier un t-uple de la relation considérée de façon unique
 - Exemple : *Personne(NumeroSécu, Prénom, Nom, DateNaissance)*
 - Toute relation possède au moins une clé car la connaissance de tous les attributs permet d'identifier un tuple unique.
 - Cependant, si on reprend l'exemple de la relation Vins, étant donné que chaque vin se voit attribuer un numéro, il est clair que l'ensemble minimal d'attributs qui permet d'identifier un t-uple de façon unique est l'attribut NV (numéro du vin)

16

Règles de base du modèle relationnel

- Contraintes de référence :
 - Des relations peuvent être associées entre-elles
 - La clé va permettre d'être référencées dans d'autres t-uples en tant que clé étrangère pour faire les liens de références entre relations
 - Exemple : *on peut associer des Vins à un producteur*
Vins(NV, CRU, MILL, DEG, REG, #NomProd)
Producteurs(NomProd, ADR, TEL, CA..)
 - Exemple : *on peut associer certains Vins à certaines caves*
Vins(NV, CRU, MILL, DEG, REG, #NomProd)
Caves (NomCave, Adresse, #Propriétaire)
VinsCaves(#NomCave, #NV, Quantité)

17

Exemple de CONTRAINTE DE REFERENCE

- Cave(numcave, surface, température)
 - On a alors une association avec le vin qui est stocké dans une cave
 - Donc la relation vins devient : *Vins(nv, cru, mill, deg, reg, #numcave)*
- Vins(nv, cru, mill, deg, reg)
 - Buveur(nb, nom, pren, age)
 - On peut créer une association entre ces deux entités régissant les consommations de vins des buveurs.
 - Etant donné qu'un buveur peut consommer plusieurs vins et qu'à son tour un vin peut être bu par plusieurs buveurs, alors on ajoute la table qui régit cette association
 - Cette relation serait alors du type : *Consommation(#nv, #nb, date, quantité)*
 - La clé de cette table est la concaténation des deux clés étrangères ainsi que de la date (car ils peuvent boire) plusieurs fois du même vin à des dates différentes

18

Règles de base du modèle relationnel

■ Les contraintes de domaine :

- Les domaines de valeur d'une colonne de relation peuvent vérifier une règle
- *Exemple : le salaire d'un employé à temps plein est forcément supérieur ou égal au SMIC et inférieur à 150 000 €*

19

Conception d'une BD ?

■ Pour concevoir des BD, on doit passer par la Modélisation Conceptuelle des Données

- Plusieurs étapes à respecter pour arriver à partir d'un besoin, à une spécification complète du modèle relationnel de la base de données.
- Définition précises des relations et des liens qui relient les relations les unes aux autres

20

Conception d'une BD

■ Exemple :

- Représentation d'un fichier d'entreprises clients et on souhaite appeler le siège social de tous les clients qui ont un CA > 500000 €
- Configuration 1 : 1 seule relation
Client(Raison-sociale, Adresse, N°Tel, CA, MoyFact)
Si un client a plusieurs numéros de tel et adresses possibles, en plus de son siège social :
Duplication d'un même client dans la table, avec des adresses différentes
==> risque d'appeler plusieurs fois le même client

21

Conception d'une BD

- Configuration 2 : 2 relations différentes liées
Client(RaisonSociale, AdresseSS, N°TelSS, CA, MoyFact)
AutresCoord(Adresse, Tel, #RaisonSociale)
- Ici on associe à un Client, autant d'AutresCoord qu'on le souhaite
- Mais dans le fichier client, n'apparaît que l'adresse de contact principale
- D'où la nécessité de faire une modélisation conceptuelle des données...

22

La Modélisation Conceptuelle des Données

Licence Professionnelle Métiers des Télé-Services

Année Universitaire 2008-2009

Les étapes de la conception

1. Elaborer le dictionnaire des données
2. Etudier les Dépendances Fonctionnelles entre ces données
3. Elaborer le Modèle Entité-Association de ces données
4. En déduire le Modèle Relationnel



Cas pratique de l'entreprise REPCHAUF

24

Les étapes de la conception

1. **Elaborer le dictionnaire des données**
2. Etudier les Dépendances Fonctionnelles entre ces données
3. Elaborer le Modèle Entité-Association de ces données
4. En déduire le Modèle Relationnel



Cas pratique de l'entreprise REPCHAUF

25

Elaboration du dictionnaire des données

- Définition
 - Liste exhaustive de toutes les données nécessaires à la modélisation du système d'information et donc de la base de données
- Chaque donnée est caractérisée par un nom et un domaine de définition
 - Pour l'élaboration de ce dictionnaire, le premier travail consiste, à la lecture de l'énoncé, à établir une liste de toutes les données nécessaires à la modélisation de la base de données. Avant d'en déduire le dictionnaire de données, il y a lieu de faire une analyse précise qui a pour but de s'assurer
 - Que toutes les données sont élémentaires, c'est à dire non décomposables
 - Qu'à chaque donnée correspond un seul libellé et réciproquement

26

Elaboration du dictionnaire des données

- Etape 1 : lecture de l'énoncé et détermination des données brutes
- Etape 2 : traiter les données de façon à ne récupérer que les données élémentaires
 - Une donnée est élémentaire lorsque les caractères qui la composent sont toujours utilisés systématiquement ensemble

27

Elaboration du dictionnaire des données

- Exemple d'une donnée « Référence »
 - Valeur possible = 1025
 - Identifie de manière certaine un produit dans la base.
- Mais l'information qu'elle porte n'est peut-être pas simple
 - Si le 1 signifie que le produit appartient à une catégorie particulière ==> deux données élémentaires
 - Le numéro de la catégorie (entre 1 et 9)
 - Le numéro du produit proprement dit dans sa catégorie (entre 0 et 999)
- Dans le cas de REPCHAUF : Ref-produit est élémentaire car pas de catégorie de produits

28

Elaboration du dictionnaire des données

- Exemple de la donnée « Adresse »
 - Une adresse comprend habituellement
 - Un numéro
 - Un nom de rue, d'avenue etc.
 - Un code postal
 - Un nom de ville
 - Un nom de pays éventuellement
 - Si adresse est utilisée globalement pour adresser du courrier alors ==> donnée élémentaire
 - En revanche, si l'on a besoin de lister les clients d'une ville ou d'un département ==> à décomposer en : Rue / Code postal / Ville
 - Remarque : repchauf ne travaillant pas à l'étranger, on ne considère pas le pays comme une donnée

29

Elaboration du dictionnaire des données -> Liste des données élémentaires

1	Adresse Client	Adresse du client
2	Coût horaire	Coût fixe d'une heure de main d'œuvre
3	Date	Date de l'intervention
4	Date	Date de la facture
5	Distance	Distance entre la société et le client visité
6	Distance totale	Distance totale pour toutes les interventions d'une facture
7	Durée	Durée de l'intervention
8	Montant	Montant total HT par type de pièce par intervention
9	N°Client	Numéro séquentiel de client
10	N°Facture	Numéro séquentiel de facture
11	N°Fiche	Numéro séquentiel de fiche d'intervention
12	Nom client	Nom du client
13	Nom intervenant	Nom de l'intervenant
14	Nom pièce	Intitulé d'une pièce
15	Nom responsable	Nom du responsable de l'intervention

30

Elaboration du dictionnaire des données

16	Nombre d'heures	Nombre d'heures totales (somme des durées des interventions)
17	Prénom client	Prénom du client
18	Prénom intervenant	Prénom de l'intervenant
19	Prénom responsable	Prénom du responsable de l'intervention
20	Prise en charge	Forfait de prise en charge par intervention
21	PU HT	Prix unitaire hors taxe d'une pièce
22	Quantité	Quantité d'une pièce utilisée dans une intervention
23	Référence	Numéro séquentiel pour une pièce
24	Référence facture	Numéro séquentiel d'une facture
25	Taux/km	Taux kilométrique fixe pour le calcul des frais de déplacement
26	Total HT	Montant total HT d'une facture
27	Total pièces	Montant total HT de toutes les pièces d'une intervention
28	Total TTC	Montant total TTC d'une facture
29	TVA	Taux fixe de TVA

31

Elaboration du dictionnaire des données

- Etape 3 : Epuration des données = analyse du rapport données-libellé des données
- Deux types de défauts : Synonymie et polysémie
- Synonyme : propriétés de noms différents mais ayant la même signification
 - Exemple : N° facture et Référence facture renvoient tous les deux à la référence que les clients portent sur leur commande lorsqu'ils l'adressent à Repchauf ==> Un seul libellé N°Facture

32

Elaboration du dictionnaire des données

- Polysème : propriétés de même nom ayant des significations différentes
 - Exemple de polysème : deux données libellées par le mot « Date » représentant d'une part la date d'intervention et d'autre part une date de facturation
 - Un tel défaut n'est pas acceptable dans une base de données. A un libelle ne doit correspondre qu'une donnée.
 - ➔ Les deux libellés deviennent Dateintervention et Datefacture

33

Elaboration du dictionnaire des données

- Etape 4 : élimination des données non pertinentes
- Le dictionnaire ne doit contenir que les données qui seront utiles
 - Exemple de donnée non pertinente : la signature du client sur un bon de commande ou bien l'adresse de la société que l'on est en train d'étudier

34

Elaboration du dictionnaire des données

- Etape 5 : Elimination des données calculées déterminables à tout moment
- La base de données ne doit pas contenir de données calculées à partir de données élémentaires - sauf exceptions

35

Elaboration du dictionnaire des données

- Données qui peuvent être déterminées à tout moment grâce aux données élémentaires :
 - Montant
 - Total pièces
 - Total HT
 - Total TTC
 ==> A éliminer

36

Elaboration du dictionnaire des données

- Données pour lesquelles les données présentes dans la base peuvent avoir évolué :
 - Montant de la commande ==> calculé à partir des PU HT
 - Si les prix unitaires changent un mois après la commande, le montant de la commande doit lui rester le même
==> Conserver le Montant de la commande
==> Proposer un PU HT courant et un PU HT facturé

37

Elaboration du dictionnaire des données

- Etape 6 : Mise à part des paramètres
- Paramètre = donnée qui peut évoluer dans le temps
- Exemple dans l'énoncé
 - TVA
 - Taux kilométrique
 - Prise en charge

38

Elaboration du dictionnaire des données

- Ensuite le dictionnaire des données est prêt à être rédigé sous la forme suivante :

Numéro	Nom donnée	Domaine de définition	Type
...

Numérotation des données

Nom des données
Libellé complet
Déjà dans les
Données
élémentaires

Normales
Calculées
Paramètres

39

Elaboration du dictionnaire des données

- Règles d'écriture du domaine de définition
 - Nature des données
 - N donnée numérique
 - A donnée alphabétique
 - AN donnée alphanumérique
 - DATE
 - BOOLEEN
 - COMPTEUR
 - FINANCIER
 - Forme et longueur des données
 - 6N : donnée de 6 caractères numériques
 - 6,2N donnée de 8 caractères dont deux décimaux
 - 2N+2N+2N+2N+2N : pour un numéro de téléphone
 - Il est aussi possible de donner un ensemble de valeurs aux données
 - « Bleu,rouge,vert,jaune » pour une donnée qui ne peut prendre que 4 couleurs différentes

40

Le dictionnaire des données

1	Adresse Client	50AN	
2	Date facture	DATE	
3	Date intervention	DATE	
4	Distance	3N en km	
5	Durée	2N+2N en heures et mn	
6	N°Client	3N	
7	N°Facture	4N	
8	N°Fiche	5N	
9	Nom client	20A	
10	Nom intervenant	20A	
11	Nom pièce	20A	
12	Nom responsable	20A	
13	Prénom client	20A	

41

Le dictionnaire des données

14	Prénom intervenant	20A	
15	Prénom responsable	20A	
16	PU HT courant	FINANCIER	
17	PU HT facturé	FINANCIER	
18	Quantité	3N	
19	Référence	4N	
20	Cout horaire	FINANCIER	Paramètre
21	Prise en charge	FINANCIER	Paramètre
22	Taux/km	FINANCIER	Paramètre
23	TVA	2,2N	Paramètre

42

Les étapes de la conception

1. Elaborer le dictionnaire des données
2. **Etudier les Dépendances Fonctionnelles entre ces données**
3. Elaborer le Modèle Entité-Association de ces données
4. En déduire le Modèle Relationnel



Cas pratique de l'entreprise REPCHAUF

43

Etude des dépendances fonctionnelles

- **Définition** : une DF met en jeux deux données. L'une est source et l'autre est but de la dépendance
 - Source ----> But
- Le lien entre ces deux données : à une valeur de la donnée source ne correspond qu'une valeur de la donnée but
 - **Exemple** : N°Client ----> Nom client

44

Etude des dépendances fonctionnelles

- Deux étapes :
 - Réalisation de la Matrice des Dépendances Fonctionnelles
 - Réalisation du Graphe des Dépendances Fonctionnelles
- **Matrice des dépendances fonctionnelles**
 - Tableau à deux dimensions
 - Les colonnes contiennent les SOURCES de DF
 - Les lignes contiennent toutes les données du dictionnaire des données (hors paramètres)

45

Matrice des dépendances fonctionnelles

Buts	Sources						
	1	2	3	4	5	...	
1. N° Fiche	*						
2. N° Facture	1	*					
3. Date intervention	1		*				
4. ...				*			

- On met des étoiles sur la diagonale car Source = But
- Chaque 1 placé à l'intersection d'une ligne et d'une colonne représente une DF
 - Ici : N°Fiche---->N°Facture et aussi N°Fiche---->Date intervention

46

Matrice des dépendances fonctionnelles de REPCHAUF

- **Etape 1 : Repérage des données sources et remplissage systématique par colonne**
 - Pour chaque colonne, envisager la donnée comme source de la DF
 - Si la donnée est SOURCE de DF, on note un 1 sur les lignes pour lesquelles les données correspondantes sont BUTS de la DF

====> EXEMPLE AVEC REPCHAUF

47

Matrice des DF de REPCHAUF

Données Buts	Données sources																		
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1 Adresse client	*																		
2 Date facture		*																	
3 Date intervention			*																
4 Distance				*															
5 Durée					*														
6 N° Client						*													
7 N° facture							*												
8 N° fiche								*											
9 Nom client									*										
10 Nom intervenant										*									
11 Nom pièce											*								
12 Nom responsable												*							
13 Prénom client													*						
14 Prénom intervenant														*					
15 Prénom responsable															*				
16 PU HT courant																*			
17 PU HT facturé																	*		
18 Quantité																		*	
19 Référence																			*

48

Matrice des dépendances fonctionnelles de REPCHAUF

- Maintenant on peut éliminer les colonnes relatives aux données qui ne sont pas SOURCES de DF
- On souligne les données qui ont été identifiées comme source de DF
- Pour REPCHAUF :
 - On obtient alors une première version de la matrice des DF suivante

49

Matrice des dépendances fonctionnelles de REPCHAUF

Buts	6	7	8	19
1 - Adresse client	1	1	1	
2 - Date facture		1	1	
3 - Date intervention			1	
4 - Distance			1	
5 - Durée			1	
6 - N° Client	*	1	1	
7 - N° facture		*	1	
8 - N° fiche			*	
9 - Nom client	1	1	1	
10 - Nom intervenant			1	
11 - Nom pièce				1
12 - Nom responsable				1
13 - Prénom client	1	1	1	
14 - Prénom intervenant			1	
15 - Prénom responsable			1	
16 - PU HT courant				1
17 - PU HT facturé				1
18 - Quantité				*
19 - Référence				*

50

Matrice des dépendances fonctionnelles de REPCHAUF

- Deuxième étape sur la matrice des DF : Suppression par transitivité des redondances de DF
- Pour chaque ligne non source de DF, il doit y avoir un maximum 1 seul 1 i.e. une seule source pour chaque cible
 - Exemple : le nom d'un client est associé à plusieurs DF ==> redondance des informations, problèmes de gestion et solution non optimale

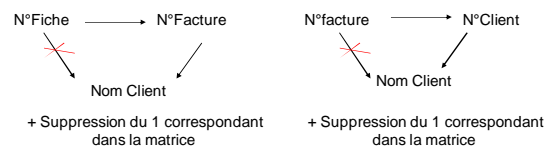
51

Matrice des dépendances fonctionnelles de REPCHAUF

■ Comment faire ?

- On a
 - N°fiche --> Nom client,
 - N°Facture --> Nom Client
 - N°Client --> Nom client

- Il faut supprimer deux de ces DF grâce à la transitivité



52

Matrice des dépendances fonctionnelles de REPCHAUF

Buts	6	7	8	19
1 - Adresse client	1	1	1	
2 - Date facture		1	1	
3 - Date intervention			1	
4 - Distance			1	
5 - Durée			1	
6 - N° Client	*	1	1	
7 - N° facture		*	1	
8 - N° fiche			*	
9 - Nom client	1	1	1	
10 - Nom intervenant			1	
11 - Nom pièce				1
12 - Nom responsable				1
13 - Prénom client	1	1	1	
14 - Prénom intervenant			1	
15 - Prénom responsable			1	
16 - PU HT courant				1
17 - PU HT facturé				1
18 - Quantité				*
19 - Référence				*

53

Matrice des dépendances fonctionnelles de REPCHAUF

Buts	6	7	8	19
1 - Adresse client	1	1	1	
2 - Date facture		1	1	
3 - Date intervention			1	
4 - Distance			1	
5 - Durée			1	
6 - N° Client	*	1	1	
7 - N° facture		*	1	
8 - N° fiche			*	
9 - Nom client	1	1	1	
10 - Nom intervenant			1	
11 - Nom pièce				1
12 - Nom responsable				1
13 - Prénom client	1	1	1	
14 - Prénom intervenant			1	
15 - Prénom responsable			1	
16 - PU HT courant				1
17 - PU HT facturé				1
18 - Quantité				*
19 - Référence				*

→ redondance
redondance

→ Pas de redondance car donnée source de DF

→ redondance

54

Etude des dépendances fonctionnelles

Buts	6	7	8	19
1 - Adresse client	1			
2 - Date facture		1		
3 - Date intervention			1	
4 - Distance			1	
5 - Durée			1	
6 - N° Client	*	1	1	
7 - N° facture	*		1	
8 - N° fiche		*		
9 - Nom client	1			
10 - Nom intervenant			1	
11 - Nom pièce				1
12 - Nom responsable			1	
13 - Prénom client	1			
14 - Prénom intervenant			1	
15 - Prénom responsable			1	
16 - PU HT courant				1
17 - PU HT facturé				
18 - Quantité				
19 - Référence				*

Résultat après suppression des redondances

55

Matrice des dépendances fonctionnelles de REPCHAUF

- Dernière étape sur la matrice des DF : Recherche des données isolées et des liens non modélisés
 - Recherche des données n'ayant pas de données source
 - En gros, quelles sont les lignes NON SOURCES qui ne possèdent pas de 1
 - Deux lignes dans le cas de REPCHAUF
 - PU HT facturé
 - Quantité

====> création de ce que l'on appelle des **Contraintes d'Intégrités Multiples** i.e. plusieurs données sources conditionnent une seule cible

56

Matrice des dépendances fonctionnelles de REPCHAUF

- Par exemple:
 - Pour la quantité, il s'agit en réel d'une quantité de pièces d'une même référence dans une fiche. On va donc avoir :
 - N°fiche + Référence --> quantité
 - De même, pour le prix unitaire HT facturé, il s'agit d'un prix unitaire d'une pièce d'une même référence, dans une fiche
 - N°fiche + Référence --> PU HT facturé

==> Ajout d'une colonne pour identifier la source N°Fiche+Référence (8+19) dans la matrice

57

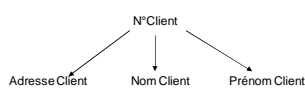
Matrice des dépendances fonctionnelles de REPCHAUF

Buts	6	7	8	19	8+19
1 - Adresse client	1				
2 - Date facture		1			
3 - Date intervention			1		
4 - Distance			1		
5 - Durée			1		
6 - N° Client	*	1	1		
7 - N° facture	*		1		
8 - N° fiche		*			*
9 - Nom client	1				
10 - Nom intervenant			1		
11 - Nom pièce				1	
12 - Nom responsable			1		
13 - Prénom client	1				
14 - Prénom intervenant			1		
15 - Prénom responsable			1		
16 - PU HT courant				1	
17 - PU HT facturé					1
18 - Quantité					1
19 - Référence			*	*	

58

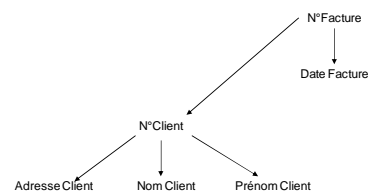
Graphe des dépendances fonctionnelles de REPCHAUF

- Faire colonne par colonnes puis une flèche pour chaque DF



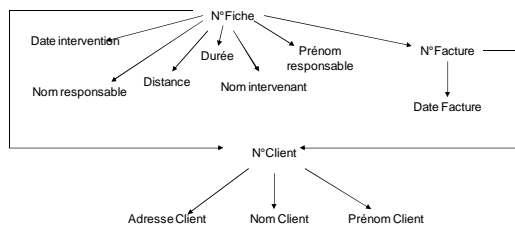
59

Graphe des dépendances fonctionnelles de REPCHAUF



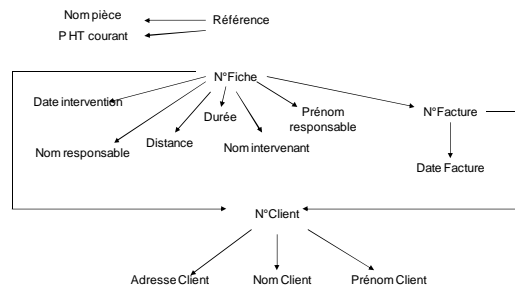
60

Graphe des dépendances fonctionnelles de REPCHAUF



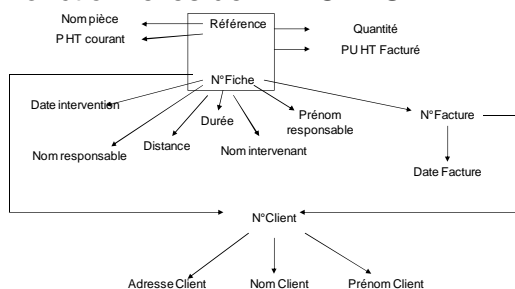
61

Graphe des dépendances fonctionnelles de REPCHAUF



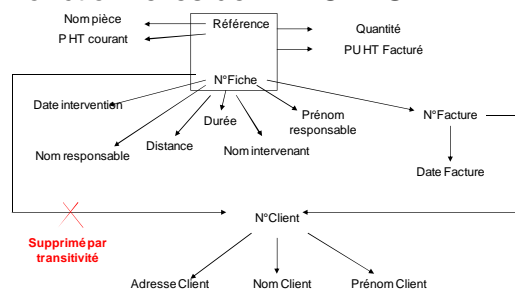
62

Graphe des dépendances fonctionnelles de REPCHAUF



63

Graphe des dépendances fonctionnelles de REPCHAUF



64

Graphe des dépendances fonctionnelles de REPCHAUF

- A partir de ce graphe des Dépendances fonctionnelles, on va pouvoir établir le **MODELE ENTITE ASSOCIATION**

□ ==> représentation graphique du modèle de la base de données

65

Les étapes de la conception

1. Elaborer le dictionnaire des données
2. Etudier les Dépendances Fonctionnelles entre ces données
3. **Elaborer le Modèle Entité-Association de ces données**
4. En déduire le Modèle Relationnel



Cas pratique de l'entreprise REPCHAUF

66

Le modèle Entité-Association

- En 1976, modèle représentant les données du SI => utilisé par plusieurs méthodes comme MERISE
- Modèle entièrement décrit par deux concepts :
 - Le concept d'entité : permet de donner une représentation sémantique des données du dictionnaire
 - Le concept d'association : permet de créer des liens entre ces données

67

Le modèle Entité-Association

- Exemples :
 - Toutes les données concernant les clients vont être regroupées dans une **entité CLIENT**.
 - Toutes celles concernant les factures vont être regroupées sous une **entité FACTURE**
 - Une association permettra par exemple de définir un lien entre un client et une facture particulière → On saura qui est le client à qui est destinée la facture

68

Le modèle Entité-Association

- Avantages du Modèle Entité Association
 - Une approche naturelle et facile à comprendre
 - Une représentation graphique et claire des résultats
- Inconvénients
 - Pas toujours facile à utiliser car les concepteurs hésitent souvent entre plusieurs solutions qui semblent a priori bonnes et équivalentes

69

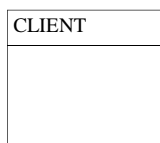
Le modèle Entité-Association

- Définitions :
 - Une entité est un type d'élément (objet, individu) du monde réel défini par
 - Une existence propre et une utilité pour l'organisation étudiée
 - Des occurrences multiples (au moins 2)
 - Des propriétés (au moins une) dont un identifiant
→ Formalisme

70

Le modèle Entité-Association

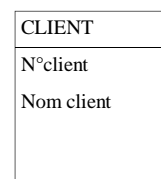
→ Formalisme



71

Le modèle Entité-Association

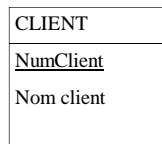
- Une propriété est une donnée du dictionnaire de données qui est sémantiquement liée à une entité ou à une association
→ Formalisme



72

Le modèle Entité-Association

- Un identifiant est une propriété ou un ensemble de propriétés d'une entité qui permet d'identifier de façon unique chaque occurrence de l'entité
→ Formalisme



73

Le modèle Entité-Association

■ Remarques

- Toutes les propriétés d'une entité doivent être en dépendance fonctionnelle avec l'identifiant i.e. que pour une valeur de l'identifiant, il n'y a qu'une valeur possible de chaque propriété de l'entité
- Les propriétés ne peuvent être dupliquées dans plusieurs entités, sinon pb de redondance. Comment dans ce cas s'assurer que les modifications dans une entité sont bien répercutées dans les autres ?

==> CONCEPT D'ASSOCIATION

74

Le modèle Entité-Association

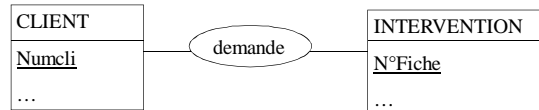
■ Définitions (suite) :

- Une association entre entités est un regroupement nécessaire entre deux ou plusieurs entités pour décrire une réalité de l'organisation.
- Ce regroupement sera nommé et portera le nom de l'association
→ Formalisme

75

Le modèle Entité-Association

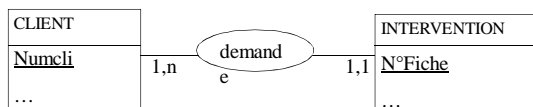
→ Formalisme



76

Le modèle Entité-Association

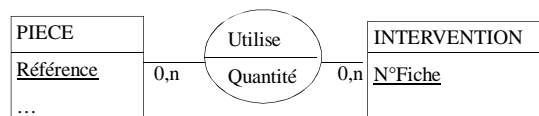
- Les Cardinalités : A toute associations entre deux entités A et B, il faut associer des cardinalités à chaque extrémité
→ Formalisme



77

Le modèle Entité-Association

- Les attributs dans une association ?
→ Formalisme



78

CIM et CIF

■ Deux types d'association :

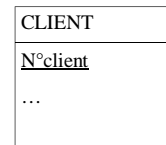
- Contraintes d'Intégrité Fonctionnelle (CIF) = Associations dont l'une des extrémités est de cardinalité « simple »
- Contraintes d'Intégrité Multiples (CIM) = Associations dont les deux extrémités sont « multiples »

79

Passage du graphe des DF au modèle E-A

- Règle 1 : pour chaque DF dont la source est élémentaire, création d'une entité dont la donnée source est l'identifiant

- N°Client donne

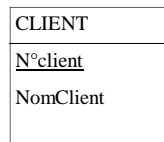


80

Passage du graphe des DF au modèle E-A

- Règle 2 : Pour chaque DF élémentaire, entre identifiant et des données élémentaires ---> propriété

- N°Client--->NomClient

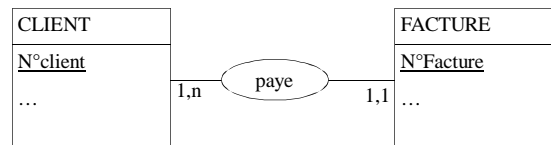


81

Passage du graphe des DF au modèle E-A

- Pour chaque DF élémentaire entre identifiants, CIF

- N°Facture--->N°Client

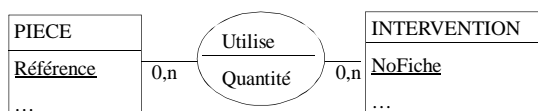


82

Passage du graphe des DF au modèle E-A

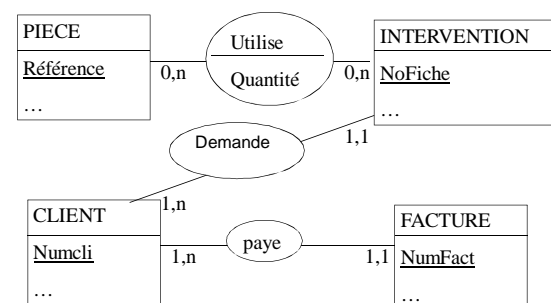
- Pour chaque DF non élémentaire CIM

- Référence + N°Fiche ---> Quantité



83

Le modèle E-A de l'entreprise REPCHAUF



84

Les étapes de la conception

1. Elaborer le dictionnaire des données
2. Etudier les Dépendances Fonctionnelles entre ces données
3. Elaborer le Modèle Entité-Association de ces données
4. **En déduire le Modèle Relationnel**



Cas pratique de l'entreprise REPCHAUF

85

Du modèle Entité-Association au modèle relationnel

■ Règle 1 : toute entité devient une relation

- PIECE(Référence, Nom pièce, ...)
- INTERVENTION(N°fiche, dateinterv,...)
- FACTURE(N°facture, Datefacture,...)
- CLIENT(N°Client, Nomclient,...)

86

Du modèle Entité-Association au modèle relationnel

■ Règle 2 : toute association de type x,1 / y,n entre les entités A et B engendre l'importation de la clé de B dans A en tant que clé étrangère

- INTERVENTION(N°fiche, dateinterv,..., #N°Client)
- FACTURE(N°facture, Datefacture,..., #N°Client)

87

Du modèle Entité-Association au modèle relationnel

■ Règle 3: toute association de type x,n / y,n entre les entités A et B donne lieu à la création d'une relation avec comme clé la concaténation des clés de A et de B (précédée de #) et portant les données éventuelles de l'association

- UTILISE(#Référence, #N°Fiche, Quantité)

88

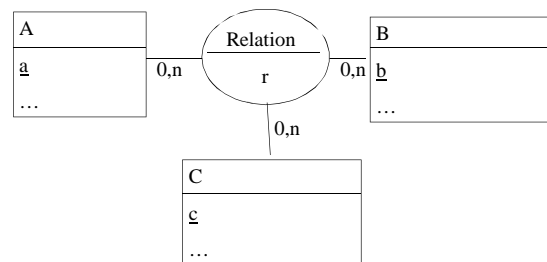
Du modèle Entité-Association au modèle relationnel

■ Le modèle relationnel final est donc :

PIECE(Référence, Nom pièce, ...)
 CLIENT(N°Client, Nomclient,...)
 INTERVENTION(N°fiche, dateinterv,..., #N°Client)
 FACTURE(N°facture, Datefacture,..., #N°Client)
 UTILISE(#Référence, #N°Fiche, Quantité)

89

Relation ternaire



→ Relation(#a, #b, #c, r)

90