

Université Sorbonne Paris Cité	UNIVERSITE PARIS 13
Institut Galilée	Année Universitaire
Laboratoire d'Informatique de Paris Nord	2019 / 2020
<i>La Data e(s)t le monde de demain</i>	👤 👤 👤 F. Boufarès
La Data/La Donnée	boufares@lipn.univ-paris13.fr
	👤 👤 👤
Bases de Données Avancées- Entrepôts de Données	GCI → Smart Data



DATA MANGEMENT & MACHINE LEARNING, the future!
(DATA BASE, DATA WAREHOUSE, MASTER DATA, DATA LAKE, BIG DATA !)

Think DIFFERENTLY, BIGGER and SMARTER!
Votre mission, si vous l'acceptez, est l'excellence
« EID : The Excellence in Data Use ! »
Si vous échouez, nous nierons avoir eu connaissance de vos agissements !
EID : L'Excellence dans l'Investigation des Données → EDI : Excellence in Data Investigation



Etude de Cas (Application)

Une Gestion Commerciale Intelligente (GCI)
 Interrogations des données ; Analyse du Chiffre d'Affaires ;
 Suivi des variations des prix des articles ; ... ; Systèmes de Recommandation ; ...

La Data : Bases de Données ; Entrepôts de Données ; Data Lake ; ... Big Data !
Conception & méta-modélisation des données ; Intégration de Données ; Qualité des Données ;
Enrichissement des données, Extractions de connaissances !
 Maîtrise de SQL (SQL2, SQL3...) ; PL/SQL ; Partage de données, Optimisation de Requêtes...
 Construction & Manipulation de bases & d'entrepôts de données...



Les dictons du jour !


→ **Aujourd'hui, j'arrête de fumer ; Le TABAC t'ABAT ; M T Dents !**
 Fumer nuit gravement à ta santé et à celle de ton entourage !

→ **SMILE and the World SMILES with you!**

→ **MANGER + MANGER = GROSSIR**
 (??? M, A, N, G, E, R, O, S, S, I)
 Manger & Bouger ;
 Pour votre santé mangez 5 fruits et légumes par jour ;
 Pour votre santé ne mangez pas trop gras, trop salé, trop sucré ;
 Pour votre santé faites une activité physique régulière ;
 Pour votre santé faites 30 mn de marche par jour !

→ **Pour votre santé faites des BD !!!** Dr. M. Faouzi Boufarès
 


Table des matières

1. Schémas de Base de Données : Structures de données & Méta-Données.....	3
1.1. Le schéma conceptuel de données (EER / UML)	3
1.2. Le schéma logique de données	4
1.3. Le dictionnaire de données & les méta-données	5
2. Exemple de contenu de la base de données (instance de BD)	7
3. Conception de BD & Rappels SQL	10
3.1. Règles de conception de Bases de Données	10
3.2. Définition avancée-complète d'une table.....	11
3.3. Insertion des données dans une table et qualité des données	13
4. Maîtrisez SQL & PL/SQL : SQL avancé, SQL pour le multidimensionnel !	14
4.1. Rappels syntaxiques de SQL	14
4.2. Interrogations & Manipulations en SQL	16
4.3. Optimisation des requêtes	22
4.4. SQL avancé ; SQL pour le multidimensionnel ; PL/SQL	27
4.5. Gestion de Gros volumes de Données - Export / Import) en SQL	29
5. Détection et Correction des anomalies dans les données, Mesures de la qualité des Données, Profilage des données	30
5.1. Qualité des Données ! Enrichissement des données !	30
5.2. Détection des dépendances fonctionnelles entre les colonnes d'une table	31
5.3. Détection des lignes similaires ou en double d'une table	31
5.4. Qualité des Données : iDQMS-An intelligent Data Quality Management System tool	32
6. Intégration de données ; Exploration Intelligente des Données et Décisionnel !	33
6.1. Historique des données (Plusieurs versions sont archivées)	36
6.2. Introduction aux Systèmes de Recommandation (SysReco)	38
6.3. Intégration de Données (Des bases aux entrepôts des données : Utilisation intelligente des données- Intelligent use of data ; Aide à la décision)	39
7. Entrepôts de données : Les schémas en étoile : à la belle étoile ! Where is the Data ? Is it in the House or in the Lake?...41	
7.1. Modélisation des Données (Des bases aux entrepôts des données)	41
7.2. Optimisation des requêtes	43

1. Schémas de Base de Données Structures de données & MétaDonnées

1. Schémas de Base de Données : Structures de données & Méta-Données

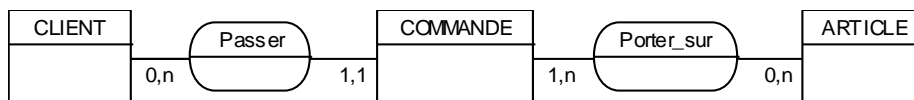
Le système d'information de l'entreprise internationale BB BOBO est articulé autour de la base de données de nom GCI (Gestion Commerciale Intelligente) décrite par le schéma conceptuel ainsi que le schéma relationnel (les tables) ci-dessous. L'entreprise BB BOBO commercialise des articles pour les bébés et les enfants dans plusieurs pays dans le monde essentiellement européens et africains. Elle dispose de plusieurs magasins dans chacun des pays où elle se trouve.

Excellez dans l'imagination ; faites du conceptuel (Extended Entity-Relationship / UML)!

1.1. Le schéma conceptuel de données (EER / UML)

Le **schéma conceptuel de données** (SCD) ou **le modèle conceptuel de données** (MCD) est décrit ci-dessous selon plusieurs formalismes :

→ **Formalisme « Entity-Relationship »** : La description des données comporte plusieurs **entités** (ou type d'entité = les rectangles) reliées entre elle par des **associations** (ou type d'association = les ovales), les arcs sont étiquetés par des contraintes de **cardinalité** (1,1 ; 1,n et 0,n) :



Les attributs de l'entité *CLIENT* sont : **CodeCli**, CiviCli, NomCli, PrenCli, CateCli, AdreCli, CPCLi, VillCli, PaysCli, MailCli, TelCli, ...

Les attributs de l'entité *COMMANDE* sont : **NumeCom**, DateCom

Les attributs de l'entité *ARTICLE* sont : **RefeArt**, NomArt, PVArt, QstkArt, PAArt

Les attributs de l'association *PASSER* sont : Aucun attribut supplémentaire !

Les attributs de l'association *PORTER-SUR (DETAILSCOM)* sont : QteCom, PUArt, Remise

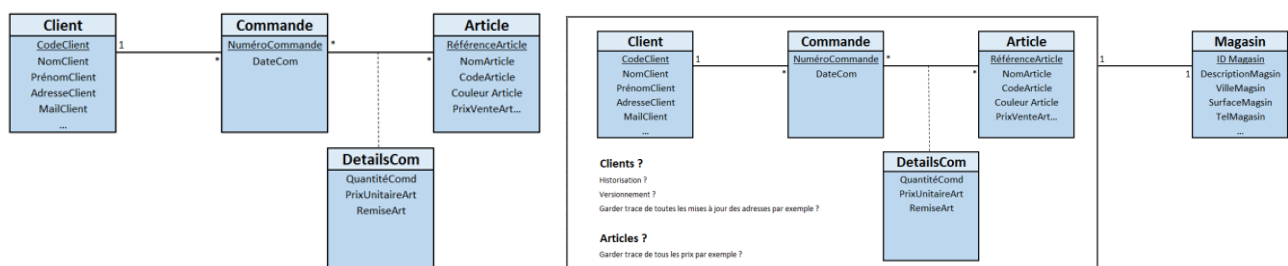
Ce schéma conceptuel peut être complété par d'autres données telles que les factures ou les magasins.

L'entité *FACTURE* peut être dans l'entité *COMMANDE* si à une commande correspond une et une seule facture.

L'entité *FACTURE* peut aussi être une entité indépendante reliée à l'entité *COMMANDE* par une association « Correspondre » avec des cardinalités du type 1,1 – 1,n.

Toutes les informations de cette base de données sont relatives à un magasin et un seul.

→ **Formalisme « UML »** : La description des données comporte plusieurs **classes** reliées entre elles par des arcs étiquetés par des contraintes de **multiplicité** (1 ; 1,* et *) :



1.2. Le schéma logique de données

Le **Schéma relationnel de la Base de Données** (Modèle Logique de Données MLD) GCI pour chacun de ses magasins est composé de :

Table des articles :

ARTICLES (RefeArt , NomArt, PVArt, QstkArt, PAArt)

Table des clients :

CLIENTS (CodeCli , CiviCli, NomCli, PrenCli, CateCli, AdreCli, CPCLi, VillCli, PaysCli, MailCli, TelCli)

Table des commandes :

COMMANDES (NumeCom , CodeCli ↑, DateCom)

Table des détails des commandes :

DETAILSCOM (NumeCom ↑, RefeArt ↑, QteCom, PUArt, Remise)

Evidemment d'autres tables peuvent faire partie de la Base de Données telles que :

Table des détails des magasins :

MAGASINS (NumMag , NomMag, TelMag, MailMag...)

Table des détails des fournisseurs :

FOURNISSEURS (NumFour , NomFour, TelFour, MailFour...)

Table des détails des factures :

FACTURES (NumeFac , DateFac, NumeCom ↑)

Pour la gestion commerciale (la gestion de ses commandes, de ses factures, de ses magasins...) :

Les articles sont identifiés de manière unique par la référence (RefeArt).

Les clients sont identifiés par le code (CodeCli).

Les commandes sont retrouvées grâce au numéro (NumeCom). Une commande n'est faite que par un seul client. Elle comporte au moins un article. Un même article ne peut exister qu'une seule fois dans une même commande. Il peut figurer dans plusieurs commandes différentes.

Une facture, référencée par son numéro (NumeFac), ne peut correspondre qu'à une seule commande. La date de la facture (DateFac) est évidemment supérieure ou égale à la date de commande (DateCom). Le montant de la facture est celui de la commande.

Un **dictionnaire de données** contient les détails de toutes les descriptions des différentes colonnes de toutes les tables de la BD.

Ces descriptifs constituent une partie des **métadonnées** de la BD.

Le dictionnaire de données de la BD GCI est détaillé ci-dessous :

1.3. Le dictionnaire de données & les méta-données

Le **dictionnaire de données** de la BD GCI est le suivant (ces descriptifs constituent un sous ensemble des **métadonnées** de la BD) :

Attribut Colonne	Signification - Description	Domaine -Type de données Contrainte
AdreCli	Adresse du client (N° + Rue/Avenue ...)	Caractères/Texte, 100
AdrNumVMag	Adresse du magasin (Numéro dans la voie)	Caractères/Texte, 10
AdrTypVoiMag	Adresse du magasin (Type de la voie)	Caractères/Texte, 20, en MAJUSCULE
AdrNomVMag	Adresse du magasin (Nom de la voie)	Caractères/Texte, 100
AdrCPMag	Adresse du magasin (Code Postal)	Caractères/Texte, 10
AdrVilleMag	Ville où se trouve le magasin	Caractères/Texte, 20, en MAJUSCULE
AdrPaysMag	Pays où se trouve le magasin	Caractères/Texte, 20, en MAJUSCULE
CateCli	Catégorie du client	Entier, compris entre 1 et 7
CiviCli	Civilité du client	Caractères/Texte, 12 ; Mademoiselle, Madame, Monsieur
CodeCli	Code du client (Unique par client)	Caractères/Texte, 10
CPCli	Code postal du client	Caractères/Texte, 10
DateCom	Date de la commande	Date de format : jj/mm/aaaa
DateFac	Date de la facture	Date de format : jj/mm/aaaa ; <= date de la commande
MailCli	Le mail du client	Caractères/Texte, 30 ; vérifie une Expression régulière
MailMag	Mail du magasin	Caractères/Texte, 30 ; vérifie une Expression régulière
NomArt	Nom (désignation) de l'article	Caractères/Texte, 30
NomCli	Nom du Client	Caractères/Texte, 20, en MAJUSCULE
NomMag	Nom du magasin	Caractères/Texte, 30
NumeCom	Numéro de la commande (Unique par commande)	Caractères/Texte, 14
NumeFac	Numéro de la facture (Unique par facture)	Caractères/Texte, 14
NumMag	Numéro du magasin (Unique par magasin)	Caractères/Texte, 10
PAArt	Prix d'achat de l'article	Réel, 10 dont 2 décimales
PaysCli	Pays du client	Caractères/Texte, 20, en MAJUSCULE
PrenCli	Prénom du client	Caractères/Texte, 20, Seule la première lettre est en majuscule
PVArt	Prix de vente de l'article (dans le catalogue)	Réel, 10 dont 2 décimales
PUArt	Prix de vente de l'article à la date de commande	Réel, 10 dont 2 décimales
QstkArt	Quantité en stock de l'article	Entier, 3, >0
QteCom	Quantité commandée	Entier, 3, >0
RefeArt	Référence de l'article (Unique par article)	Caractères/Texte, 8
Remise	Taux de remise appliqué à l'article	Réel, 4 dont 2 décimales
TelCli	Téléphone du client	Caractères/Texte, 30 ; vérifie une Expression régulière
TelMag	Téléphone du magasin	Caractères/Texte, 30 ; vérifie une Expression régulière
VillCli	Ville où habite le client	Caractères/Texte, 20, en MAJUSCULE

Remarques : La définition des données est ambiguë (**Homogénéisations**, Expressions régulières...) !

Les définitions des adresses (**Clients & Magasins**) sont **incompatibles** !

Ceci est une source d'anomalies ! IL FAUT **UNIFIER** LES DESCRIPTIONS DES DONNEES !

AdreCli	Adresse du client (N° + Rue/Avenue ...)	Caractères/Texte, 100
CPCli	Code postal du client	Caractères/Texte, 10
VillCli	Ville où habite le client	Caractères/Texte, 20, en MAJUSCULE
PaysCli	Pays du client	Caractères/Texte, 20, en MAJUSCULE

AdrNumVMag	Adresse du magasin (Numéro dans la voie)	Caractères/Texte, 10
AdrTypVoiMag	Adresse du magasin (Type de la voie)	Caractères/Texte, 20, en MAJUSCULE
AdrNomVMag	Adresse du magasin (Nom de la voie)	Caractères/Texte, 100
AdrCPMag	Adresse du magasin (Code Postal)	Caractères/Texte, 10
AdrVilleMag	Ville où se trouve le magasin	Caractères/Texte, 20, en MAJUSCULE
AdrPaysMag	Pays où se trouve le magasin	Caractères/Texte, 20, en MAJUSCULE

Expression de quelques contraintes sur les données :

La **civilité d'un(e) client(e)** prend une valeur dans la liste suivante : {Mademoiselle, Madame, Monsieur} ou {Mme., M}, les valeurs sont du type **alphabétique**.

La catégorie **d'un(e) client(e)** est du type **entier, compris entre 1 et 7**

Le **prénom d'un(e) client(e)** ne peut être composé que de lettres alphabétiques

Le **nom d'un(e) client(e)** ne peut être composé que de lettres alphabétiques

Le **numéro dans la voie dans une adresse** est du type alphanumérique.

Le **type de la voie dans une adresse** prend une valeur dans la liste suivante : {RUE, BOULEVARD, AVENUE, QUAI, IMPASSE, PONT, PLACE, SQUARE, ALLÉE, ALLÉES, VOIE, MONTÉE, ESPLANADE, ROUTE, VOIRIE, CITÉ, CHEMIN, PARVIS...}, les valeurs sont du type alphabétique.

La **quantité en stock d'un article** est une valeur du type **entier positif**.

Il faudrait utiliser des **expressions régulières** pour définir correctement le **mail**, le **téléphone** etc...

MétaDonnées	NOM <i>de la donnée, de la colonne le plus souvent insignifiant</i>	Colonne 1 <i>P ou C</i>	Colonne 2 <i>Mail ou MeI</i>	Colonne 3 <i>T</i>	...	Colonne N <i>DN</i>
	TYPES de données <i>SYNTAXE</i>	String Char, Varchar Text	String Char, Varchar Text	Number Decimal, Real Integer	Boolean	Date
	CONTRAINTES <i>le plus souvent absentes et donc à découvrir</i> <i>SEMANTIQUE</i>	Check : A set of values, A list of values, A range of values, Positive, Negative, ... Uniqueness of each value, Mandatory values, ..., Default values, Relationships, links, Dependencies, orders... Regular expressions, Uppercase, Lowercase, Vowel, Consonant, ... Categorization and sub categorization of data Data deduplication... Data cleaning				
	COMMENTAIRES	/ * a text, a comment */		/ * a text, a comment */	/ * a text, a comment */	/ * a text, a comment */
Données	Source de données ... Table					
	Fichier CSV, Tableau Table SGBD Relationnel Objet-Relationnel SQL, NoSQL XML...	Espagne France RU France Italie Allemagne United-Kingdom China	inès.rama@gmail.com adam.traifor@gmail.com miame.nissan@gmail.com aline.reno@gmail.com edite.fiate@gmail.com Aimé.BEYEMEV@lipn.fr harissa.ford@yahoo.fr ala.ferrari@yahoo.fr	171 cm 1.89 m 165 cm 188 1.92 m 1.89 m 1.89 m 1234567890		02/01/2019 03/01/2019 04/01/2019 05/01/2019 06/01/2019 08 févr 2019 09 févr 2019 11 févr 2019
		Source de données ... CSV				
		Espagne;inès.rama@gmail.com;171 cm;;02/01/2019 France;adam.traifor@gmail.com;1.89 m;;03/01/2019 RU;miame.nissan@gmail.com;165 cm;;04/01/2019 France;aline.reno@gmail.com;188;;05/01/2019 Italie;edite.fiate@gmail.com;1.92m;;06/01/2019 Allemagne;Aimé.BEYEMEV@lipn.fr;1.89 m;True;08 févr 2019 United-Kingdom;harissa.ford@yahoo.fr;1.89 m;;09 févr 2019 China;ala.ferrari@yahoo.fr;1234567890;False;11 févr 2019				
MétaDonnées	? What	Country	Email	Size	BOOLEAN	DATE
	? What are Date Sources ??	DSi & DSj i,j	DSi & DSk i,k		data traceability	
	? What are Languages, Measures... ??	the language of the data		units of measurement used		
	Donnée sensible	No	Yes	No	No	Yes
MétaDonnées	Donnée obligatoire	Yes	Yes	No	No	Yes

2. Exemple de contenu de la base de données (instance de BD)

2. Exemple de contenu de la base de données (instance de BD)



Tables

Table des articles : ARTICLES (RefArt, NomArt, PVArt, QstkArt, PAArt)

RefeArt	NomArt	PVArt	QstkArt	PAArt
00.37291	SIEGE DE TABLE	55,00 €	10	26,43 €
00.37325	CASQUE CYCLISTE PROTECTION	27,00 €	20	20,00 €
00.37341	BONNET PARE-CHOC	10,71 €	100	5,71 €
00.37366	STORE DE PROTECTION	7,43 €	5	4,29 €
00.37408	COINS DE PROTECTION	4,00 €	100	1,71 €
00.37424	CACHE PRISE DE COURANT	2,29 €	500	0,71 €
00.37432	BLOQUE PORTE	5,00 €	4	2,29 €
00.37457	BANDE DE FERMETURE	2,29 €	4	1,71 €
00.37465	PROTECTION DE FOUR	37,86 €	10	26,43 €
00.37473	PROTECTION DE CUISINIERE	30,71 €	10	22,86 €
00.37515	BABY PHONE	80,71 €	3	45,71 €
00.37630	BRETELLE DE SECURITE REMOND	15,57 €	8	11,43 €
00.39321	HAUSSE-CHAISE	42,71 €	1	25,43 €
00.39495	COUVRE ROBINETTERIE	7,86 €	12	4,29 €
11.36613	BARRIERE DE PORTE, EN BOIS	32,71 €	2	25,71 €
11.36639	BARRIERE DE PORTE, EN METAL	79,14 €	3	28,43 €
11.36860	BARRIERE DE LIT	23,71 €	10	15,71 €
11.37108	PORTE-BEBE	44,86 €	6	34,29 €
11.37173	SIEGE-AUTO COSMOS	126,86 €	2	71,43 €
11.37280	SIEGE-AUTO EUROSEAT	121,29 €	2	71,43 €
WD.001	K7 VIDEO - TOY STORY	21,29 €	10	9,29 €
WD.002	DVD - TOY STORY 2	34,29 €	10	12,14 €
WD.003	K7 VIDEO - WINNIE L'OURSON	32,86 €	5	9,29 €
MV.001	Jeux Vidéo PS4 SPIDER-MAN	59,00 €	250	19,00 €

Table des clients : CLIENTS (CodeCli, CiviCli, NomCli, PrenCli, CateCli, AdreCli, CPCli, VillCli, PaysCli, MailCli, TelCli)

CodeCli	CiviCli	NomCli	PrenCli	CateCli	AdreCli	CPCli	VillCli	PaysCli	MailCli
2994570	Madame	RAHMA	CLEMENCE	3	8, BOULEVARD FOCH	93800	EPINAY-SUR-SEINE	FRANCE	clemence.rahma@up13.fr
2996100	Monsieur	CLEMENCE	ALEX@NDRE	1	5, RUE DE BELLEVILLE	75019	PARIS	FRANCE	alexandre.clemence@up13.fr
2997007	Monsieur	TRAIFOR	ADAM	2	13, AVENUE J. B. CLEMENT	93430	VILLETANEUSE	FRANCE	adam.traifor@up13.fr
2997777	Mademoiselle	LE BON	CLEMENTINE	1	18, BOULEVARD FOCH	93800	EPINAY-SUR-SEINE		clementine.le bon@up13.fr
2998500	Monsieur	CHEV@LIER	INES	1	13, RUE DE LA PAIX	93800	EPINAY SUR SEINE	FRANCE	ines.chevalier@up13.fr
3001777	Monsieur	LE BON	Adam	1	186, AVENUE D'ITALIE	75013	PARIS	FRANCE	adam.le bon@up13.fr
2998505	Mademoiselle	TRAIFOR	ALICE	2	6, RUE DE LA ROSIERE	75015	PARIS		alice.traifor@up13.fr
3000106	Monsieur	HARISSA	FORD	1	16, AVENUE FOCH	75016	PARIS	FRANCE	ford.harissa@up13.fr
3000107	MO NSIEUR	ONRI	PANDA	2	77, RUE DE LA LIBERTE	13001	MARSEILLE	FRANCE	panda.onri@up13.fr
3000108	Madame	EDITE	FIAT	1	21, AVENUE D'ITALIE	75013	PARIS		fiat.edite@up13.fr
3000109	Madame	TOYOTA	JACKSON	3	55, RUE DU JAPON	94310	ORLY VILLE	FRANCE	jackson.toyota@up13.fr
3000110	MADAME	ONRI	HONDA	2	77, RUE DE LA LIBERTE	13001	MARSEILLE	FRANCE	honda.onri@up13.fr
3000111	Madame	GENEREUX	EVE	1	18, BOULEVARD FOCH	91000	EPINAY SUR ORGE	FR	eve.genereux@up13.fr
2999999	Mlle	BON	CLEMENTINE	1	18, BOULEVARD FOCH	93800	EPINAY-SUR-SEINE	FRANCE	clementine.bon@up13.fr
3001	M.	LE BON	Adam	1	186, AVENUE D'ITALIE	75013	Paris		adam.le bon@up13.fr
3001777	Mr	LE BON	Adem	1	186, AVENUE D'ITALIE	75013	PARIS	france	adem.le bon@up13.fr

Table des commandes : COMMANDES (NumeCom, CodeCli, DateCom)

NumeCom	CodeCli	DateCom
20150917BB0001	2997007	17/09/2015
20150917BB0002	2998500	17/09/2015
20151016BB0001	2996100	16/10/2015
20151024BB0001	2996100	24/10/2015
20120156BB0001	2996100	16/01/2012
20120157BB0007	2996100	17/01/2012
20180917BB0007	3001777	17/09/2018

Table des détails des commandes : DETAILSCOM (NumeCom, RefeArt, QteCom, PUArt, Remise)

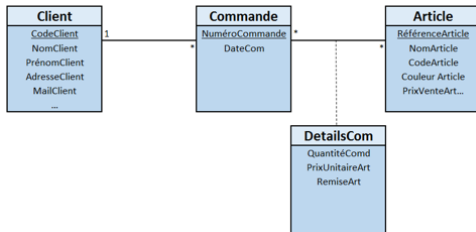
NumeCom	RefeArt	QteCom	PUArt	Remise
20150917BB0001	00.37291	2	35,00 €	10,00%
20150917BB0001	00.37341	4	10,71 €	10,00%
20150917BB0001	11.36639	2	49,14 €	0,00%
20150917BB0001	WD.001	1	21,29 €	0,00%
20150917BB0002	00.37291	2	35,00 €	0,00%
20150917BB0002	WD.001	1	21,29 €	0,00%
20150917BB0002	WD.002	1	34,29 €	0,00%
20150917BB0002	WD.003	1	22,86 €	0,00%
20151016BB0001	WD.001	1	21,29 €	20,00%
20151016BB0001	WD.002	1	34,29 €	0,00%
20151016BB0001	WD.003	1	22,86 €	0,00%
20151024BB0001	00.37457	2	3,29 €	0,00%
20120156BB0001	WD.003	2	25,00 €	0,00%
20120157BB0007	WD.001	5	22,00 €	50,00%
20120157BB0007	WD.002	10	34,29 €	50,00%
20180917BB0007	MV.001	2	59,00 €	10,00%

Table des factures : FACTURES (NumeFac, DateFac, NumeCom)

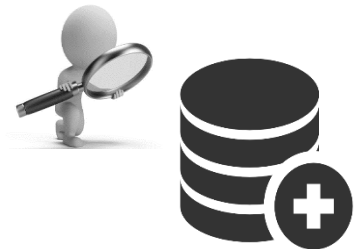
NumeFca	DateFac	NumeCom
20150917FB0001	17/09/2015	20150917BB0001
20150917FB0002	17/09/2015	20150917BB0002
20151016FB0001	16/10/2015	20151016BB0001
20151024FB0001	24/10/2015	20151024BB0001
20120156FB0001	26/01/2012	20120156BB0001
20120157FB0007	27/01/2012	20120157BB0007
20180917FB0007	17/09/2018	20180917BB0007

Une Gestion Commerciale Intelligente (GCI)

Version 1



G1



Les dictons du jour !

→ **Aujourd'hui, j'arrête de fumer ; Le TABAC t'ABAT ; M T Dents !**
Fumer nuit gravement à ta santé et à celle de ton entourage !

→ **SMILE and the World SMILES with you!**

→ **MANGER + MANGER = GROSSIR**

(??? M, A, N, G, E, R, O, S, S, I)

Manger & Bouger ;

Pour votre santé mangez 5 fruits et légumes par jour ;

Pour votre santé ne mangez pas trop gras, trop salé, trop sucré ;

Pour votre santé faites une activité physique régulière ;

Pour votre santé faites 30 mn de marche par jour !

→ **Pour votre santé faites des BD !!!** Dr. M. Faouzi Boufarès



Le problème dont souffrent beaucoup de gens
The problem that many people are suffering



3. Quelques Questions Simples pour Commencer ! Conception de BD & Rappels SQL

3. Conception de BD & Rappels SQL

3.1. Règles de conception de Bases de Données

3.1.1. Conception de Bases de Données : La table ci-dessous qui contient les détails des commandes est **mal conçue** (mal construite) ! Commentez « le pourquoi ? » !

NumeCom	RefeArt	Nomart	QteCom	PUArt	Remise
20150917BB0001	00.37291	SIEGE DE TABLE	2	35,00	10,00%
20150917BB0001	00.37341	BONNET PARE-CHOC	4	10,71	10,00%
20150917BB0001	11.36639	BARRIERE DE PORTE, EN METAL	2	49,14	0,00%
20150917BB0001	WD.001	K7 VIDEO - TOY STORY	1	21,29	0,00%
20150917BB0002	00.37291	SIEGE DE TABLE	2	35,00	0,00%
20150917BB0002	WD.001	K7 VIDEO - TOY STORY	1	21,29	0,00%
20150917BB0002	WD.002	DVD - TOY STORY 2	1	34,29	0,00%
20150917BB0002	WD.003	K7 VIDEO - WINNIE L'OURSON	1	22,86	0,00%
20151016BB0001	WD.001	K7 VIDEO - TOY STORY	1	21,29	20,00%
20151016BB0001	WD.002	DVD - TOY STORY 2	1	34,29	0,00%
20151016BB0001	WD.003	K7 VIDEO - WINNIE L'OURSON	1	22,86	0,00%
20151024BB0001	00.37457	BANDE DE FERMETURE	2	3,29	0,00%
20120156BB0001	WD.003	K7 VIDEO - WINNIE L'OURSON	2	25,00	0,00%
20120157BB0007	WD.001	K7 VIDEO - TOY STORY	5	22,00	50,00%
20120157BB0007	WD.002	DVD - TOY STORY 2	10	34,29	50,00%

La table est mal conçue à cause de la colonne Nomart. Celle-ci n'est pas nécessaire, elle est redondante. Elle peut générer des anomalies de mises à jour.

3.1.2. Conception de Bases de Données : La table suivante est-elle correcte (est-elle **bien conçue**) ? Commentez !

ComCli				
NumeCom	CodeCli	DateCom	NomCli	CateCli
20151016BB0001	2996100	16/10/2015	CLEMENCE	1
20151024BB0001	2996100	24/10/2015	CLEMENCE	1
20120156BB0001	2996100	16/01/2012	CLEMENCE	1
20120157BB0007	2996100	17/01/2012	CLEMENCE	1
20150917BB0001	2997007	17/09/2015	TRAIFOR	2
20150917BB0002	2998500	17/09/2015	CHEVALIER	1

La table n'est pas bien conçue, elle contient des redondances (NomCli et CateCli). Elle peut générer des anomalies de mises à jour.

3.1.3. Conception de Bases de Données : La table suivante est-elle correcte (est-elle **bien conçue**) ? Commentez !

CLI			
CodeCli	CiviCli	NomCli	PrenCli
2994570	Madame	RAHMA	CLEMENCE
2996100	Monsieur	CLEMENCE	ALEXANDRE
2994570	Monsieur	TRAIFOR	ADAM
2997777	Mademoiselle	LE BON	CLEMENTINE

La table est bien conçue car elle ne contient pas de redondance.

3.1.4. Conception de Bases de Données et déclarations des contraintes :

Quelles sont les clés **primaire et étrangère** de la table DETAILSCOM.

Commentez les contraintes d'intégrité relatives aux contraintes de multiplicité (Qualité des données !)

Clé primaire	La clé primaire est composée de deux attributs (NumeCom, RefeArt)
Clé étrangère	Deux clés étrangères : NumeCom et RefeArt

Contraintes : Elles doivent être vérifiées dans la déclaration des tables avec CONSTRAINT ou en utilisant des déclencheurs (TRIGGERS) !

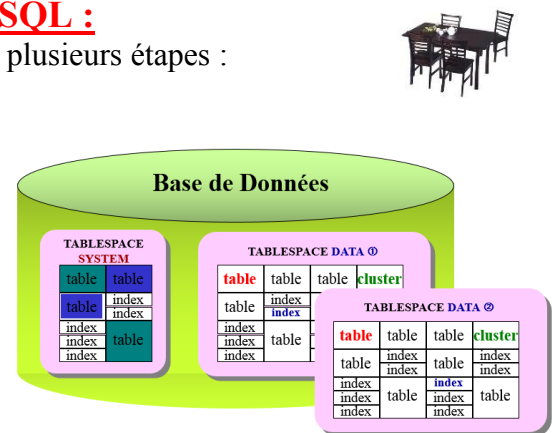
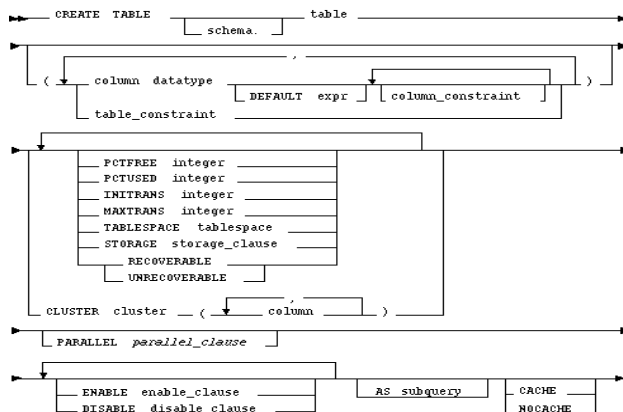
Un même article ne peut exister qu'une seule fois dans une même commande !

Une commande contient au moins un article (ne peut pas être vide) !

3.2. Définition avancée-complète d'une table

→ → → Définir une table - Créer une table en SQL :

La « bonne et complète » définition d'une table se fait en plusieurs étapes :



- La définition des attributs (des colonnes)

Attributs → types de données de base (prédéfinis par le système)

Numérique, Alphanumérique (Caractères, Texte), Date, ...

Attributs → types de données définis par l'utilisateur

- La définition des contraintes

Contraintes → contrôles de validités des données

Clés primaire et étrangère, Vérifications (appartenance à une liste ou à un intervalle, comparaison à une valeur, non vide ou nullité ...)

PROBLEME DE COHERENCE GLOBALE & DE QUALITE DE DONNEES

- La définition des caractéristiques physiques (des paramètres de stockage)

Taux de remplissage des blocs, évolution ...

Lieu de stockage (disque, Mémoire vive -cache-), parallélisme...

Optimisations : Partitionnement, découpage

→ → → Définir une table - Créer une table en SQL :

CREATE TABLE NomTable

(
Attributs,
Contraintes
)

TABLESPACE nomtablespace

PCTUSED x1

PCTFREE x2

INITTRANS x3

MAXTRANS x4

STORAGE (

INITIAL n1 NEXT n2

MINEXTENTS m1 MAXEXTENTS m2

PCTINCREASE p)

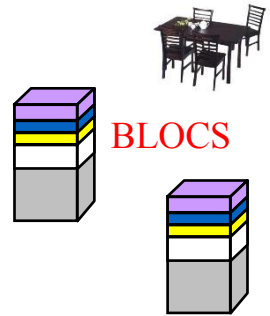
PARALLEL p / NOPARALLEL (appliquer le parallélisme)

LOGGING / NOLOGGING (pour ne pas copier les mises à jour dans le **Rollback**)

CACHE / NOCACHE (permet de garder la table en mémoire : à utiliser avec précaution)

PARTITIONnement (une sorte de découpage de la table)

;



- La création des commentaires sur les tables et leurs colonnes

Commenter les tables et leurs colonnes est une étape très importante pour créer les **métadonnées** nécessaires lors de la phase **d'intégration des données**.

- La définition des chemins d'accès rapides aux données (des index)

Indexer : Accélérer les traitements, Reconstruire les blocs...

- La construction de vues et de vues matérialisées (des calculs préétablis, ...)

Créer des **vues**, Créer des **vues matérialisées**

- Le partage des données entre plusieurs utilisateurs

Créer des **droits** d'accès, des **rôles**, des **profiles**...

- Le contrôle de la validité des données avec des procédures et des fonctions

Créer des **mécanismes** de contrôles (**Procedure**, **Function**, **Package**)

- Le déclenchement d'actions en cascade à la suite des mises-à-jour

Créer des **déclencheurs** (**Trigger**)

3.3. Insertion des données dans une table et qualité des données

→ → → Créer les données dans une table / Insérer les données dans une table en SQL :

La « **bonne et complète** » insertion des données dans une table devrait se faire avec une **procédure PL/SQL** qui permet de **contrôler** le contenu de chaque colonne et non seulement avec la commande SQL « **INSERT INTO NomTable (...) VALUES (...)** » :

```
CREATE OR REPLACE PROCEDURE InsertDon_NomTable (Val1, Val2, ...) IS
...
BEGIN
  Anomalie:=FALSE;
  IF ... THEN ...
  IF Anomalie = FALSE THEN -- Si TOUT va Bien, aucune anomalie selon les contraintes !
    INSERT INTO NomTable (Col1, Col2, ...) VALUES (Val1, Val2, ...) ;
    COMMIT ;
  END IF ;
END;
/
```

Voir **SCRIPT SQL** pour la **Version 1** de la BD GCI : **Bxy-G1-CreatInsertDon.sql**
→ La création des tables (création de la structure des données, structures vide !)
→ La création des données (insertions des données dans la BD créée)



Le fichier **Bxy-G1-CreatInsertDon.sql** est un script SQL2 sous le SGBD ORACLE.

NB : La **Version 1** devra être enrichie dans la **Version 2** ! → → → **Bxy-G2-CreatInsertDon.sql**

4. Maîtrisez SQL & PL/SQL

SQL avancé, SQL pour le multidimensionnel... !

4. Maîtrisez SQL & PL/SQL : SQL avancé, SQL pour le multidimensionnel !

4.1. Rappels syntaxiques de SQL

4.1.1. Manipulation de Bases de Données et SQL :

Quelques mots clés en **SQL** ?

Le Concept (le mot)	désigne quel(s) Mot(s) clé dans SQL
Attribut/Champ/Colonne	SELECT
Table	FROM
Critères de sélection ou de jointure	WHERE
Tri (<i>Croissant</i>)	ORDER BY ASC par défaut.
Tri (<i>Décroissant</i>)	ORDER BY DESC.
Opération de <i>Regroupement</i>	GROUP BY
Opération de <i>Regroupement avec sélection du résultat</i>	GROUP BY ... HAVING
Union, Intersection, Différence	UNION ... INTERSECT... MINUS ...

4.1.2. Manipulation de Bases de Données et SQL :

Parmi les requêtes ci-dessous, quelles sont celles qui sont correctement écrites en **SQL** ?

Requête	Code SQL	Oui	Non
A	SELECT CateCli, COUNT (CodeCli) FROM Clients ;		X
B	SELECT CateCli AS Catégorie, COUNT (CodeCli) AS NombreDeClients FROM Clients WHERE VillCli = "Paris" ORDER BY CateCli;		X
C	SELECT CodeCli, NomCli, DateCom FROM Clients, Commandes WHERE Clients.CodeCli = Commandes.CodeCli;		X
D	SELECT CateCli AS Catégorie, COUNT (CodeCli) AS NombreDeClients FROM Clients WHERE VillCli = "Paris" AND COUNT (CodeCli) > 2 ;		X
E	SELECT Catégorie, NombreDeClients FROM Clients, Commandes WHERE VillCli = "Paris" AND DateCom < #01/01/2014#;		X
F	SELECT CateCli AS Catégorie, COUNT (*) AS NombreDeClients FROM Clients GROUP BY CateCli;	X	
G	SELECT CateCli AS Catégorie, COUNT (*) AS NombreDeClients FROM Clients WHERE VillCli = "Paris" GROUP BY CateCli;	X	
H	SELECT NumeCom, SUM (QteCom*PUArt*(1-Remise)) AS Montant FROM DetailCom GROUP BY NumeCom;	X	
I	SELECT CateCli AS Catégorie, COUNT (*) AS NombreDeClients FROM Clients WHERE VillCli = "Paris" GROUP BY CateCli HAVING COUNT (*) > 2;	X	
J	SELECT Clients.CodeCli, NomCli, DateCom FROM Clients, Commandes WHERE Clients.CodeCli = Commandes.CodeCli;	X	
K	SELECT CateCli AS Catégorie, COUNT (*) AS NombreCli FROM Clients, Commandes WHERE Clients.CodeCli = Commandes.CodeCli GROUP BY CateCli ;	Y	

4.1.3. Manipulation de Bases de Données et SQL :

Remplissez le tableau suivant (Cochez le Oui si la phrase est correcte, le Non si la phrase est incorrecte) :

N°	Phrase-Question	Oui	Non
A	Une table peut-elle avoir des colonnes en double ?		X
B	La requête SQL suivante est-elle bien écrite ? SELECT SUM(QteCom*PUArt) AS Montant FROM DetailCom GROUP BY NumeCom;		X
C	La requête SQL suivante est-elle bien écrite ? SELECT DESC NomCli FROM Clients;		X
D	Est-ce qu'un tableau croisé dynamique (Excel) correspond à une requête SQL sans GROUP BY ?		X
E	Le Système d'Information d'une entreprise a-t-il comme données de base une base de données ?	X	
F	Une clé primaire d'une table peut-elle être composée de plusieurs colonnes ?	X	
G	Une clé étrangère d'une table peut-elle être composée de plusieurs colonnes ?	X	
H	Une table peut-elle avoir plusieurs clés étrangères ?	X	
I	Une table ne peut-elle pas avoir des colonnes en double ?	X	
J	Les requêtes SQL suivantes sont-elles équivalentes ? "SELECT COUNT(NomCli) FROM Clients;" et "SELECT COUNT(*) FROM Clients;"	X	
K	Une colonne clé primaire peut-elle contenir des cases vides ?		X
L	Une table peut-elle contenir deux colonnes de mêmes noms ?		X
M	Une colonne clé primaire peut-elle contenir des doublons ?		X
N	Dans une table qui contient 100 lignes, le type de la clé primaire peut-il être un texte qui appartient à la liste "Madame", "Mademoiselle", "Monsieur" ?		X
O	Une requête SQL peut-elle contenir uniquement les mots clés SELECT et WHERE ?		X
P	La requête suivante est-elle correcte ? SELECT CodeCli FROM CLIENT ; WHERE VillCli = "Paris";		X
Q	La jointure de la table T1 avec la table T2 est-elle équivalente à la jointure de la table T2 avec la table T1 ? (Optimisation des requêtes)	X	
R	Est-il possible de joindre une table avec elle-même ?	X	
S	Une colonne clé étrangère peut-elle contenir des cases vides ?	X	
T	Une requête SQL peut-elle contenir uniquement les mots clés SELECT et FROM ?	X	



Requêtes

(Interrogations & Manipulations de Bases de Données, Extractions de connaissances)

4.2. Interrogations & Manipulations en SQL

4.2.1. Exemples de Manipulation et d'interrogation de Bases de Données (GCI)

Formulez les requêtes ci-dessous en **SQL** ! (Testez et complétez les scripts SQL 2 & SQL 3)

Plusieurs types de requêtes par ordre de difficultés croissantes :

Requêtes A_i ; Requêtes B_i ; Requêtes C_i ; Requêtes D_i ;
Requêtes E_i ; Requêtes F_i ; Requêtes G_i ; Requêtes H_i ;
Requêtes I_i ; Requêtes J_i ; Requêtes K_i ; Requêtes H_i ;

Voir **SCRIPT SQL pour la Version 1 de la BD GCI : Bxy-G1-ManipDon.sql**
→ Manipulations des données (Interrogations et Mises à jour)

Le fichier **Bxy-G1-ManipDon.sql** est un script SQL2 sous le SGBD ORACLE)



Requêtes A_i (Sélections + Projections)

SELECT ... FROM ... UneTable... WHERE ... ;

A01. Les Noms des clients (Avec éventuellement des doublons)

A02. Les Noms des clients (Sans les doublons)

A03. Les articles dont le prix de vente est supérieur ou égal au double du prix d'achat

A04. Les articles dont le prix de vente est soit 4 soit 27 soit 35

A05. Les articles dont le prix de vente est compris entre 30 et 50

A06. Les commandes du mois de septembre 2015

A07. Les détails des commandes d'une année donnée (2012)

A08. Les clients de « Paris » (Civilité Nom Prénom, Ville), le nom de la ville **s'écrit comme** « Paris »

A09. Les clients dont le nom commence par « C »

A10. Les articles dont le nom commence par « Barrières de ... »

A11. Les articles du fournisseur « WD »

A12. Les clients pour lesquels **on n'a pas de téléphone**

A13. Les clients dont le nom de la ville **se prononce comme** « pari » ou « bariz » ou « pary »

A14. Les clients dont le nom **est similaire à** « pari » ou « bariz » ou « pary »

A15. **Décodification** de la catégorie des clients : **Transformez** la catégorie comme suit : 1 ? Grossiste, 2 ? Demi-Gros, ? Détaillant

Qualité des données : Y-a-t-il des anomalies dans les données ; Quelles sont ces anomalies ?

A16. Les clients pour lesquels le nom et/ou le prénom **sont invalides** (Code, Nom, et Prénom)

A17. Les clients pour lesquels les mails **sont invalides** (Code, Nom, et Mail)

A18. Les clients pour lesquels les téléphones **sont invalides** (Code, Nom, et Mail)

☺ ☞ Quelques Réponses aux Requêtes A:

A01. Les Noms des clients (Avec éventuellement des doublons)

NomCli	SQL
RAHMA	SELECT NomCli FROM Clients ;
CLEMENCE	
TRAIFOR	
LE BON	
CHEVALIER	
TRAIFOR	
HARISSA	
HENRI	
EDITE	
TOYOTA	
HENRI	
GENEREUX	

A02. Les Noms des clients (Sans les doublons)

NomCli	SQL
CHEVALIER	SELECT DISTINCT NomCli FROM Clients ;
CLEMENCE	
EDITE	
GENEREUX	
HARISSA	
HENRI	
LE BON	
RAHMA	
TRAIFOR	
TOYOTA	
TRAIFOR	

A03. Articles dont le prix de vente est supérieur ou égal au double du prix d'achat

RefeArt	NomArt	PVArt	PAArt	SQL
00.37291	SIEGE DE TABLE	55,00	26,43	SELECT RefeArt, NomArt, PVArt, PAArt FROM Articles WHERE PVArt >= 2 * PAArt ;
00.37408	COINS DE PROTECTION	4,00	1,71	
00.37424	CACHE PRISE DE COURANT	2,29	0,71	
00.37432	BLOQUE PORTE	5,00	2,29	
11.36639	BARRIERE DE PORTE, EN METAL	79,14	28,43	
WD.001	K7 VIDEO - TOY STORY	21,29	9,29	
WD.002	DVD - TOY STORY 2	34,29	12,14	
WD.003	K7 VIDEO - WINNIE L'OURSON	32,86	9,29	

A04. Les articles dont le prix de vente est soit 4 soit 27 soit 35

RefeArt	NomArt	PVArt	QStkArt	SQL
00.37325	CASQUE CYCLISTE PROTECTION	27,00	20	SELECT RefeArt, NomArt, PVArt, QStkArt FROM Articles WHERE PVArt IN (4, 27, 35);
00.37408	COINS DE PROTECTION	4,00	100	

A05. Les articles dont le prix de vente est compris entre 30 et 50

RefeArt	NomArt	PVArt	QStkArt	SQL
00.37465	PROTECTION DE FOUR	37,86	10	SELECT RefeArt, NomArt, PVArt, QStkArt FROM Articles WHERE PVArt BETWEEN 30 AND 50;
00.37473	PROTECTION DE CUISINIERE	30,71	10	
00.39321	HAUSSE-CHAISE	42,71	1	
11.36613	BARRIERE DE PORTE, EN BOIS	32,71	2	
11.37108	PORTE-BEBE	44,86	6	
WD.002	DVD - TOY STORY 2	34,29	10	
WD.003	K7 VIDEO - WINNIE L'OURSON	32,86	5	

A06. Les commandes du mois de septembre 2015

NumeCom	CodeCli	DateCom	SQL
20150917BB0001	2997007	17/09/2015	SELECT NumeCom, CodeCli, DateCom FROM Commandes WHERE DateCom >= '09/01/2015' AND DateCom <= '09/30/2015';
20150917BB0002	2998500	17/09/2015	

A08. Les clients de « Paris » (Civilité Nom Prénom, Ville)

Client	Ville	SQL
Monsieur CLEMENCE ALEXANDRE	PARIS	SELECT CivCli ' ' NomCli ' ' PrenCli AS Client, VillCli AS Ville FROM Clients WHERE UPPER(VillCli) = 'PARIS';
Mademoiselle TRAFOR ALICE	PARIS	
Monsieur HARISSA FORD	PARIS	
Madame EDITE FIAT	PARIS	

A09. Les clients dont le nom commence par « C »

CodeCli	CiviCli	NomCli	PrenCli	SQL
2996100	Monsieur	CLEMENCE	ALEXANDRE	SELECT CodeCli, CiviCli, NomCli, PrenCli FROM Clients WHERE NomCli LIKE 'C%';
2998500	Monsieur	CHEVALIER	INES	
3000111	Madame	GENEREUX	EVE	

A10. Les articles dont le nom commence par « Barrières de ... »

RefeArt	NomArt	PVArt	QStkArt	SQL
11.36613	BARRIERE DE PORTE, EN BOIS	32,71	2	SELECT RefeArt, NomArt, PVArt, QStkArt FROM Articles WHERE NomArt LIKE 'BARRIERE DE%';
11.36639	BARRIERE DE PORTE, EN METAL	79,14	3	
11.36860	BARRIERE DE LIT	23,71	10	

A13. Décodification de la catégorie des clients : Transformez la catégorie comme suit : 1 → Grossiste, 2 → Demi-Gros, → Détaillant

SELECT catecli, DECODE(catecli, 1, 'Grossiste', 2, 'Demi-Gros', 'Détaillant') AS Categorie FROM Clients;

Requêtes B_i (Tris + Sélections + Projections + Calculs)

SELECT ... function(...) ... FROM ... WHERE ... ORDER BY ...;

B01. Classez les clients par ville, par ordre croissant

B02. Classez les articles de PV < 20, dans l'ordre décroissant du stock

B03. Calculez la marge pour chaque article (marge = PV – PA !), présentez le résultat par ordre décroissant de la marge

B04. Calculez la marge pour tous les sièges, présentez le résultat par ordre croissant de la marge

B05. Calculez le nombre de valeurs nulles

Quelques Réponses aux Requêtes B_i

B01. Classez les clients par ville, par ordre croissant

CodeCli	CiviCli	NomCli	PrenCli	CateCli	AdreCli	CPCli	VillCli	SQL
3000111	Madame	GENEREUX	EVE	1	18, BOULEVARD FOCH	91000	EPINAY SUR ORGE	SELECT * FROM Clients ORDER BY VillCli ;
2998500	Monsieur	CHEVALIER	INES	1	13, RUE DE LA PAIX	93800	EPINAY SUR SEINE	
2997777	Mademoiselle	LE BON	CLEMENTINE	1	18, BOULEVARD FOCH	93800	EPINAY SUR SEINE	
2994570	Madame	RAHMA	CLEMENCE	3	8, BOULEVARD FOCH	93800	EPINAY SUR SEINE	
3000110	Madame	HENRI	HONDA	2	77, RUE DE LA LIBERTE	13001	MARSEILLE	
3000107	Monsieur	HENRI	PANDA	2	77, RUE DE LA LIBERTE	13001	MARSEILLE	
3000109	Madame	TOYOTA	JACKSON	3	55, RUE DU JAPON	94310	ORLY VILLE	
3000108	Madame	EDITE	FIAT	1	21, AVENUE D'ITALIE	75013	PARIS	
3000106	Monsieur	HARISSA	FORD	1	16, AVENUE FOCH	75016	PARIS	
2998505	Mademoiselle	TRAFOR	ALICE	2	6, RUE DE LA ROSIERE	75015	PARIS	
2996100	Monsieur	CLEMENCE	ALEXANDRE	1	5, RUE DE BELLEVILLE	75019	PARIS	
2997007	Monsieur	TRAFOR	ADAM	2	13, AVENUE J. B.CLEMENT	93430	VILLETANEUSE	

B02. Classez les articles de PV < 20, dans l'ordre décroissant du stock

RefeArt	NomArt	PVArt	QStkArt	SQL
00.37424	CACHE PRISE DE COURANT	2,29	500	SELECT RefeArt, NomArt, PVArt, QStkArt FROM Articles WHERE PVArt < 20 ORDER BY QStkArt DESC ;
00.37408	COINS DE PROTECTION	4,00	100	
00.37341	BONNET PARE-CHOC	10,71	100	
00.39495	COUVRE ROBINETTERIE	7,86	12	
00.37630	BRETELLE DE SECURITE REMOND	15,57	8	
00.37366	STORE DE PROTECTION	7,43	5	
00.37457	BANDE DE FERMETURE	2,29	4	
00.37432	BLOQUE PORTE	5,00	4	

Requêtes C_i (Calculs + Sélections+ Projections)

SELECT ...function(...)... FROM ... WHERE ...;

C01. Nombre de clients (Femme + Homme)

C02. Nombre de clientes (Femme)

C03. Nombre de clients (Homme)

C04. Le prix de vente le plus élevé

C05. Moyenne des prix de vente des articles

C06. Le chiffre d'affaires global

C07. Les articles dont le prix de vente est supérieur à la moyenne des prix de vente

😊 ☞ Quelques Réponses aux Requêtes C_i

C01. Nombre de clients (Femme + Homme)

NombreDeClients	SQL
12	SELECT COUNT(*) AS NombreDeClients FROM Clients;

C04. Le prix de vente le plus élevé

PrixVente_le_plus_élevé	SQL
126,86	SELECT MAX(PVArt) AS PrixVente_le_plus_élevé FROM Articles;

C05. Moyenne des prix de vente des articles

MoyPV_Art	SQL
36,789130418197	SELECT AVG(PVArt) AS MoyPV_Art FROM Articles;

Requêtes D_i (Calculs + Sous-Totaux + Sélections + Projections + Tris)

SELECT ... function(...) ... FROM ... WHERE ...

GROUP BY ... ORDER BY ...;

D01. Nombre de clients par catégorie

D02. Nombre de clients parisiens par catégorie

D03. Montant de chaque commande

D04. Nombre de clients parisiens par catégorie, nombre > 2

D05. Total des quantités commandées par article

D06. Total des quantités commandées par catégorie d'article

D07. Total du chiffre d'affaires par catégorie d'article

D08. Total, moyenne, min et max du chiffre d'affaires par catégorie d'article

D09. Le chiffre d'affaires par année

D10. Le nombre de valeurs différentes par colonne

Quelques Réponses aux Requêtes D_i

D01. Nombre de clients par catégorie

Catégorie	NombreDeClients	
1	6	SELECT CateCli AS Catégorie, COUNT(*) AS NombreDeClients
2	4	FROM Clients
3	2	GROUP BY CateCli;

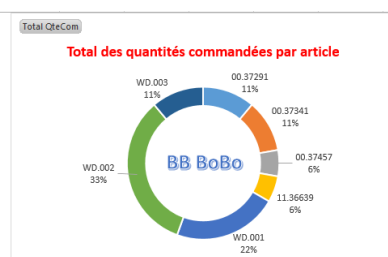
D02. Nombre de clients parisiens par catégorie

Catégorie	NbrCli_Paris	
1	3	SELECT CateCli AS Catégorie, COUNT(*) AS NombreDeClients FROM Clients
2	1	WHERE VilleCli = 'Paris' GROUP BY CateCli;
		SELECT CateCli AS Catégorie, COUNT(*) AS NombreDeClients FROM Clients
		WHERE UPPER(VilleCli) = 'PARIS' GROUP BY CateCli;

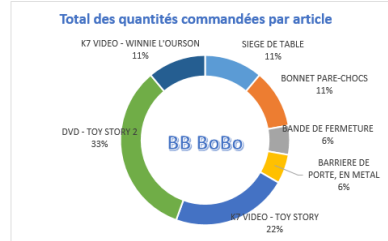
D05. Total des quantités commandées par article

NumeCom	RefeArt	QteCom	PUArt	Remise
20150917BB0001	00.37291	2	35	10.00%
20150917BB0001	00.37341	4	10.71	10.00%
20150917BB0001	11.36639	2	49.14	0.00%
20150917BB0001	WD.001	1	21.29	0.00%
20150917BB0002	00.37291	2	35	0.00%
20150917BB0002	WD.001	1	21.29	0.00%
20150917BB0002	WD.002	1	34.29	0.00%
20150917BB0002	WD.003	1	22.86	0.00%
20151016BB0001	WD.001	1	21.29	20.00%
20151016BB0001	WD.002	1	34.29	0.00%
20151016BB0001	WD.003	1	22.86	0.00%
20151024BB0001	00.37457	2	3.29	0.00%
20120156BB0001	WD.003	2	25	0.00%
20120157BB0007	WD.001	5	22	50.00%
20120157BB0007	WD.002	10	34.29	50.00%

Articles RefeArt	Total QteCom
00.37291	4
00.37341	4
00.37457	2
11.36639	2
WD.001	8
WD.002	12
WD.003	4
Total général	36



Articles RefeArt	Articles NomArt	Total QteCom
00.37291	SIÈGE DE TABLE	4
00.37341	BONNET PARE-CHOCS	4
00.37457	BANDE DE FERMETURE	2
11.36639	BARRIÈRE DE PORTE, EN METAL	2
WD.001	K7 VIDEO - TOY STORY	8
WD.002	DVD - TOY STORY 2	12
WD.003	K7 VIDEO - WINNIE L'OURSON	4
Total général		36



Produits cartésien & jointures entre plusieurs tables

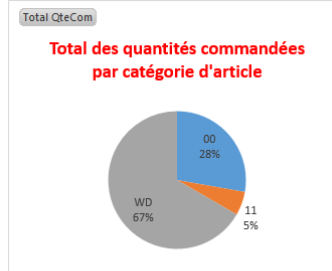
D06. Total des quantités commandées par catégorie d'article

D07. Total du chiffre d'affaires par catégorie d'article

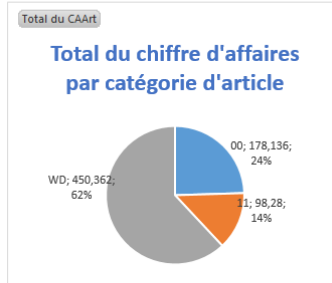
D08. Total, moyenne, min et max du chiffre d'affaires par catégorie d'article

NumeCom	RefeArt	CatArt	QteCom	PUArt	Remise	CAArt
20150917BB0001	00.37291	00	2	35	10,00%	63
20150917BB0001	00.37341	00	4	10,71	10,00%	38,556
20150917BB0001	11.36639	11	2	49,14	0,00%	98,28
20150917BB0001	WD.001	WD	1	21,29	0,00%	21,29
20150917BB0002	00.37291	00	2	35	0,00%	70
20150917BB0002	WD.001	WD	1	21,29	0,00%	21,29
20150917BB0002	WD.002	WD	1	34,29	0,00%	34,29
20150917BB0002	WD.003	WD	1	22,86	0,00%	22,86
20151016BB0001	WD.001	WD	1	21,29	20,00%	17,032
20151016BB0001	WD.002	WD	1	34,29	0,00%	34,29
20151016BB0001	WD.003	WD	1	22,86	0,00%	22,86
20151024BB0001	00.37457	00	2	3,29	0,00%	6,58
20120156BB0001	WD.003	WD	2	25	0,00%	50
20120157BB0007	WD.001	WD	5	22	50,00%	55
20120157BB0007	WD.002	WD	10	34,29	50,00%	171,45

Catégories Articles	Total QteCom
00	10
11	2
WD	24
Total général	36



Catégories Articles	Total du CAArt
00	178,136
11	98,28
WD	450,362
Total général	726,778



Catégories Articles	Total CA	Moyenne du CA	Min du CA	Max du CA
00	178,136	44,534	6,58	70
11	98,28	98,28	98,28	98,28
WD	450,362	45,0362	17,032	171,45
Total général	726,778	48,45186667	6,58	171,45

Optimisation des requêtes - Requêtes sur plusieurs tables

Produits cartésien & jointures

La jointure est un cas particulier du produit cartésien

Plusieurs types de jointures : θ -jointure

(θ appartient à $\{<, \leq, =, >, \geq, \neq\}$ - (θ appartient à $\{<, <=, =, >, >=, <>\}$)

Jointure, équi-jointure, jointure naturelle, full outer join, left join, right join

4.3. Optimisation des requêtes

Requêtes E_i (Produit-Cartésien, Jointures + Sélections + Projections)

SELECT ... FROM ...PlusieursTables... WHERE ... ;

E01. → Différents types de jointures (équi-jointure) : voir pages suivantes

E02. full outer join

E03. left join

E04. right join

E05. Produit Cartésien (Le cas le plus général sans clause WHERE ; la jointure est un cas particulier)

E06. La téta-jointure (téta appartient à $\{<, <=, >, >=, <>\}$)

E07.

E08. Jointure & Optimisation des requêtes

E09. Jointure avec une requête dans la clause FROM

E10. Jointure & Ordre d'exécution des tables



Quelques Réponses aux Requêtes E_i

Optimisation des requêtes

Produits Cartésien & Jointures

Le Produit Cartésien est le cas le plus général pour croiser toutes les lignes des tables concernées sans la clause WHERE : **SELECT ... FROM Table1, Table2 ;**

E01. Le Produit Cartésien : Croisement de chacune des lignes de la table des clients avec toutes les lignes de la table des commandes

SELECT * FROM Clients, Commandes ;

La jointure et ses variantes,
constituent des cas particuliers du produit cartésien avec la clause WHERE :
SELECT ... FROM Table1, Table2 WHERE Table1.Colonne 0 Table2.Colonne ;

E02. Les clients ayant commandé et leurs commandes (équi-jointure)

TTITLE 'Jointure : Les clients ayant commandé et leurs commandes '

SELECT * FROM Clients, Commandes WHERE Clients.codecli= Commandes.codecli;

E03. Les clients ayant commandé et leurs commandes

TTITLE 'Full outer join : Les clients ayant commandé et leurs commandes'

SELECT * FROM Clients full outer join Commandes on Clients.codecli = Commandes.codecli;

E04. Les clients ayant commandé et leurs commandes

TTITLE 'Left join : Les clients ayant commandé et leurs commandes'

SELECT * FROM Clients left outer join Commandes on Clients.codecli = Commandes.codecli;

E05. Les clients ayant commandé et leurs commandes

TTITLE 'Right join : Les clients ayant commandé et leurs commandes'

SELECT * FROM Clients right outer join Commandes on Clients.codecli = Commandes.codecli;

E06. Les dates des commandes des clients de PARIS

TTITLE 'Jointure : Les dates des commandes des clients de PARIS '

SELECT Clients.codecli, Commandes.datecom FROM Clients, Commandes
 WHERE Clients.codecli= Commandes.codecli AND Clients.villcli=LOWER('paris');

SELECT Clients.codecli, Commandes.datecom FROM Clients, Commandes
 WHERE Clients.codecli= Commandes.codecli AND Clients.villcli=UPPER('PARIS');

Les requêtes ci-dessous ne donnent pas le même résultat à cause des majuscules-minuscules !

SELECT Clients.codecli, Commandes.datecom FROM Clients, Commandes
 WHERE Clients.codecli= Commandes.codecli AND Clients.villcli='PARIS';

SELECT Clients.codecli, Commandes.datecom FROM Clients, Commandes
 WHERE Clients.codecli= Commandes.codecli AND Clients.villcli='paris';

SELECT Clients.codecli, Commandes.datecom FROM Clients, Commandes
 WHERE Clients.codecli= Commandes.codecli AND Clients.villcli='Paris';

Jointure & Optimisation des requêtes

Les différentes formulations, ci-dessous, ne donnent pas les mêmes performances

E07. Les clients (Codes & Noms des clients) de Paris ayant commandé

TTITLE '1. Jointure première écriture : Les clients de paris ayant commandé'

```
SELECT Clients.Codecli, Clients.Nomcli FROM Clients, Commandes
WHERE Clients.codecli= Commandes.codecli AND Clients.villcli=LOWER('paris');
```

TTITLE '2. Jointure deuxième écriture : Les clients de paris ayant commandé'

```
SELECT Clients.Codecli, Clients.Nomcli FROM Clients,
WHERE Clients.vilcli=LOWER('paris')
AND Clients.codecli IN (SELECT Commandes.codecli FROM Commandes);
```

TTITLE '3. Jointure troisième écriture : Les clients de paris ayant commandé'

```
SELECT Clients.Codecli, Clients.Nomcli FROM Clients,
WHERE Clients.vilcli=LOWER('paris')
AND Clients.codecli = ANY (SELECT Commandes.codecli FROM Commandes);
```

TTITLE '4. Jointure quatrième écriture : Les clients de paris ayant commandé'

```
SELECT Clients.Codecli, Clients.Nomcli FROM Clients,
WHERE Clients.vilcli=LOWER('paris')
AND EXISTS (SELECT * FROM Commandes WHERE Commandes.codecli = Clients.codecli);
```

TTITLE '5. Jointure cinquième écriture : Les clients de paris ayant commandé'

```
SELECT Clients.Codecli, Clients.Nomcli FROM Clients,
WHERE Clients.vilcli=LOWER('paris')
AND 0 < (SELECT COUNT(*) FROM Commandes WHERE Commandes.codecli = Clients.codecli);
```

E08. Les clients (Code des clients et Dates des commandes) de Paris ayant commandé

TTITLE '6. Jointure : Requête SELECT dans le FROM'

```
SELECT t.codecli, Commandes.datecom
FROM (SELECT * FROM Clients WHERE UPPER(vilcli)='PARIS') t, Commandes
WHERE t.codecli= Commandes.codecli;
```

Jointure & choix de l'ordre d'exécution des tables pour des raisons de performances !
SELECT /* + ordered */ ← ceci s'appelle le hint !

E09. Les clients (Code des clients et Dates des commandes) de Paris ayant commandé

TTITLE '7. Jointure : **Choix de l'ordre d'exécution** des tables **Commandes** et **Clients** '

```
SELECT /* + ordered */ Clients.codecli, Commandes.datecom FROM Commandes, Clients
WHERE Clients.codecli= Commandes.codecli AND UPPER(Clients.vilcli)='PARIS';
```

TTITLE '8. Jointure : **Choix de l'ordre d'exécution** des tables **Clients** et **Commandes** '

```
SELECT /* + ordered */ Clients.codecli, Commandes.datecom FROM Clients, Commandes
WHERE Clients.codecli= Commandes.codecli AND UPPER(Clients.vilcli)='PARIS';
```

Requêtes F_i (Calculs + Sous-Totaux + Unions + Intersections + Différences + Jointures)

SELECT ... FROM ... Plusieurs Tables ... WHERE ... ;
SELECT ... FROM ... WHERE ... UNION S F W ;
SELECT ... FROM ... WHERE ... MINUS S F W ;
SELECT ... FROM ... WHERE ... INTERSECT S F W ;

F01. Clients ayant commandé en septembre 2011

F02. Montant total des commandes de septembre 2011

F03. Commandes ayant des articles dont le prix vente est supérieur à 20 (Commande, Article, PV)

F04. Commandes ayant des articles dont le prix vente est supérieur à 20 (Commande, Nombre)

F05. Commandes ayant 4 articles dont le prix vente est supérieur à 20

F06. Les clients de paris qui n'ont pas commandé en octobre 2011

F07. Les clients de paris ou ceux ayant commandé en octobre 2011

☺ ☞ Quelques Réponses aux Requêtes F_i

F06. Les clients de Paris qui n'ont pas commandé en octobre 2011

SELECT * FROM Clients WHERE UPPER(VillCli) = 'PARIS'
AND CodeCli NOT IN
(SELECT CodeCli FROM Commandes WHERE DateCom LIKE '%10/2011');

F07. Les clients de Paris qui n'ont pas commandé en octobre 2011

SELECT * FROM Clients WHERE UPPER(VillCli) = 'PARIS'
MINUS
SELECT CodeCli FROM Commandes WHERE DateCom LIKE '%10/2011';

F08. Les clients de Paris ou ceux ayant commandé en octobre 2011

SELECT CodeCli FROM Clients WHERE UPPER(VillCli) = 'PARIS'
UNION
SELECT CodeCli FROM Commandes WHERE DateCom LIKE '%10/2011';

Requêtes G_i (Divisions)

G01. Les articles qui figurent sur **toutes** les commandes !

G02. Articles commandés par **tous** les parisiens

G03. Les articles qui figurent sur **toutes** les commandes d'une période donnée !

😊 ☞ Quelques Réponses aux Requêtes G_i

G02. Articles commandés par **tous** les parisiens

```
SELECT      REFEART, NOMART      FROM ARTICLES
WHERE       NOT EXISTS
            (SELECT * FROM CLIENTS WHERE UPPER(VILCLI) = 'PARIS'
            AND NOT EXISTS
            (SELECT * FROM COMMANDES, DETAILCOM
            WHERE COMMANDES.NUMECOM = DETAILCOM.NUMECOM
            AND DETAILCOM.REFEART = ARTICLES.REFEART
            AND COMMANDES.CODECLI = CLIENTS.CODECLI ));
```

Algèbre relationnelle de la division !

4.4. SQL avancé ; SQL pour le multidimensionnel ; PL/SQL

Requêtes H_i (SQL Avancé, SQL pour le multidimensionnel) CUBE ; ROLLUP ; RANK() OVER

H01. Nombre de clients
H02. Nombre de pays
H03. Nombre de clients par catégorie
H04. Nombre de clients par catégorie et par ville
H05. Nombre de clients par ville et par catégorie
H06. Nombre de clients par catégorie et par ville → fonction **CUBE**
H07. Nombre de clients par catégorie et par ville → fonction **ROLLUP**

H08. Classez les clients par ordre décroissant du chiffre d'affaires CA
H09. Classez les clients par ordre décroissant du CA, donnez le rang
H10. Classez les clients par ordre décroissant du CA, donnez le rang : **RANK() OVER...**
H11. Clients par ordre décroissant du CA et de la catégorie, donnez le rang
H12. Clients par ordre décroissant du CA et de la catégorie, donnez le rang

😊 📌 Quelques Réponses aux Requêtes H_i

H01. Nombre de clients

ttitle 'Nombre de clients'
select count(codecli) as nbclients from Clients;

H02. Nombre de pays

ttitle 'Nombre de pays'
select count(distinct paycli) as nbpays from Clients;

H03. Nombre de clients par catégorie

ttitle 'Nombre de clients par catégorie'
select catecli as categorie, count(*) as nbc from Clients
group by catecli;

H04. Nombre de clients par catégorie et par ville

ttitle 'Nombre de clients par catégorie et par ville'
select catecli as categorie, villcli as ville, count(*) as nbcv from Clients
group by catecli, villcli;

H05. Nombre de clients par ville et par catégorie

ttitle 'Nombre de clients par catégorie et par ville'
select villcli as ville, catecli as categorie, count(*) as nbcv from Clients
group by villcli, catecli;

H06. Nombre de clients par catégorie et par ville → fonction CUBE

ttitle 'Nombre de clients par catégorie et par ville CUBE'
select catecli as categorie, villcli as ville, count(*) as nbcv from Clients
group by CUBE(catecli, villcli);

H07. Nombre de clients par catégorie et par ville → fonction ROLLUP

ttitle 'Nombre de clients par catégorie et par ville ROLLUP'
select catecli as categorie, villcli as ville, count(*) as nbcv from Clients
group by ROLLUP(catecli, villcli);

H08. Classez les clients par ordre décroissant du chiffre d'affaires CA

tttitle 'Les clients par ordre décroissant du CA'
select * from cli??? Est-une Vue order by cacli desc;

H09. Classez les clients par ordre décroissant du ca, donnez le rang

tttitle 'Les clients par ordre décroissant du CA, donnez le rang'
select (select count(*)+1 from cli C2 where C1.cacli<C2.cacli) as rang,
cacli, codecli, nomcli, catcli, vilcli, paycli
from cli C1
order by rang;

H10. Classez les clients par ordre décroissant du ca, donnez le rang : **RANK() OVER...**

tttitle 'Classez les clients par ordre décroissant du ca, donnez le rang'
select **rank() over**(order by cacli desc) as rang,
cacli, codecli, nomcli, catcli, vilcli, paycli
from cli;

H11. Clients par ordre décroissant du ca et de la catégorie, donnez le rang

tttitle 'Clients par ordre décroissant du ca et de la catégorie, donnez le rang'
select **rank() over** (order by cacli desc, catcli desc) as rang,
cacli, codecli, nomcli, catcli, vilcli, paycli
from cli;

H12. Clients par ordre décroissant du ca et de la catégorie, donnez le rang

tttitle 'Clients par ordre décroissant du ca et de la catégorie, donnez le rang'
select count(C2.cacli) as rang,
C1.cacli, C1.codecli, C1.nomcli, C1.catcli, C1.vilcli, C1.paycli
from cli C1, cli C2
where C1.cacli <= C2.cacli or (C1.cacli=C2.cacli and C1.catcli=C2.catcli)
group by C1.catcli, C1.cacli, C1.codecli, C1.nomcli, C1.vilcli, C1.paycli
order by C1.cacli desc, C1.catcli desc;

Remarques :

La première partie de la clause **WHERE C1.cacli <= C2.cacli** permet de ne pas compter le nombre d'occurrences où la valeur de la colonne cacli est <= à elle-même.

Si la colonne cacli ne contient pas de doublons, cette partie de la clause where serait en elle-même suffisante pour générer le rang correct.

La deuxième partie de la clause **WHERE or (C1.cacli=C2.cacli and C1.catcli=C2.catcli)**, garantit qu'en présence de doublons dans la colonne cacli, chaque valeur obtiendrait le correct.

A VERIFIER pour les clients ex æquo

4.5. Gestion de Gros volumes de Données - Export / Import) en SQL

Requêtes I_i (SQL Gestion de Gros volumes de Données - Export / Import)

I01. **Export** depuis une BD gérée par un SGBD vers des fichiers (Extraction/Export/Déchargement)

Utilisation de l'utilitaire **SPOOLER** pour extraire/exporter les tables d'une BD vers des fichiers textes

(sql> spool nomdefichier.txt	sql>select...	sql> spool off)
(sql> spool nomdefichier.lst	sql>select...	sql> spool off)

Déchargez toutes les tables de la BD GCI dans des fichiers **CSV** en utilisant le « ; » comme séparateur de colonnes.

I02. **Import** Depuis des fichiers vers une BD gérée par un SGBD (Import/Chargement)

Importer/Chargez toutes les tables de la BD GCI à partir de fichiers de différents types tels que CSV, Excel et autres.

Le « ; » est utilisé comme séparateur de colonnes.



Quelques Réponses aux Requêtes I_i

5. Détection et Correction des anomalies dans les données

Mesures de la qualité des Données

Profilage des données

5. Détection et Correction des anomalies dans les données, Mesures de la qualité des Données, Profilage des données

5.1. Qualité des Données ! Enrichissement des données !

Etant donné les requêtes ci-dessous qui permettent de voir aisément qu'il y a des anomalies dans les données !
La question qui se pose est « Comment détecter et corriger les anomalies dans les données ? ».

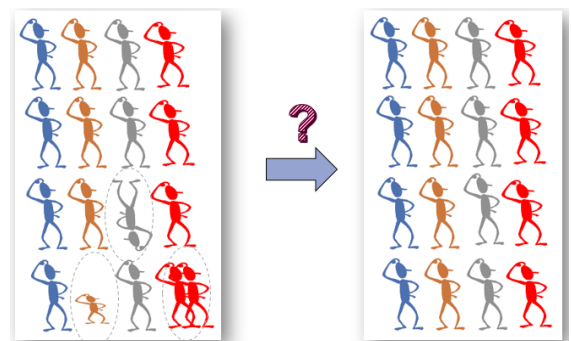
```

REM Liste des pays
TTITLE 'Liste des pays'
SELECT DISTINCT paycli AS Pays FROM Clients;

REM Nombre de pays
TTITLE 'Nombre de pays'
SELECT COUNT(DISTINCT paycli) AS NbPays FROM Clients;

REM Liste des villes
TTITLE 'Liste des villes'
SELECT DISTINCT vilcli AS Villes FROM Clients;

REM Nombre de villes
TTITLE 'Nombre de villes'
SELECT COUNT(DISTINCT vilcli) AS NbVilles FROM Clients;
    
```



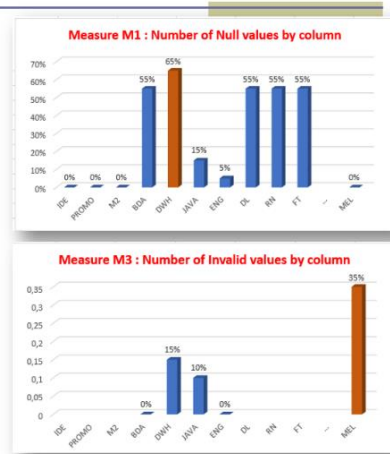
Exemple d'Indicateurs-Mesures de la Qualité

Profilage des données

Column Identifier	Column Name	Measure M11 Null Values	Measure M12 Different Values	Measure M13 Invalid Values	Measure M... M...	Measure M1p Min	Measure M1n Max
Col01	ID	0%	100%				
Col02	PROMO	0%	10%			2017	2018
Col03	M2	0%	10%				
Col04	BEA	55%	55%	0%		9	18
Col05	DWH	65%	...	15%	...	11	16
Col06	JAVA	15%	...	10%	...	5	16
Col07	ENG	5%	...	0%	...	10	16
Col08	DL	55%				7	13
Col09	RM	55%				10	16
Col10	FT	55%				5	18
Col k	MEL	0%		35%			

Objectifs des mesures (M1i, M2j, ...)
Détection de contraintes :

- Afin de détecter des anomalies :
- INTRA-COLONNE
 - INTER-COLONNES
 - INTER-LIGNES



- 5.1.1. Quelles sont les anomalies dans la table CLIENTS ?
- 5.1.2. Comment les détecter ?
- 5.1.3. Comment les corriger ?
- 5.1.4. Quelles sont les dépendances sémantiques entre les colonnes ?
- 5.1.5. Comment compléter les valeurs manquantes (valeurs nulles) ?
- 5.1.6. Comment détecter les lignes en double et/ou similaires

Requêtes J_i (SQL & PL/SQL)

Déclencheurs-Triggers ; Procédures-Fonctions ;
Cohérence globale des données ; Qualité des données

J01. Développez une procédure PL/SQL qui permet de vérifier la cohérence de la BD : Vérifiez si toutes les commandes portent au moins sur un article et affichez celles qui sont donc vides !

J02. Développez une procédure PL/SQL qui permet de vérifier certaines dépendances sémantiques telles que les dépendances fonctionnelles dans la table des clients !

J03. Développez une procédure PL/SQL qui permet de repérer (d'éliminer) les doubles et/ou les similaires dans la table des clients !



Quelques indications de Réponses aux Requêtes J_i

J01. Développez une procédure PL/SQL qui permet de vérifier la cohérence de la BD :
Vérifiez si toutes les commandes portent au moins sur un article et affichez celles qui sont donc vides !

```
SET SERVEROUTPUT ON;
CREATE OR REPLACE PROCEDURE COHERENCE_BD IS
    CURSOR curseur IS
        SELECT NUMCOM FROM COMMANDES2
        MINUS (SELECT NUMCOM FROM DETAILCOM2);
BEGIN
    DBMS_OUTPUT.PUT_LINE('Le(s) numéro(s) de commande vide(s) sont : ');
    FOR i IN curseur LOOP
        DBMS_OUTPUT.PUT_LINE(' ' || i.NUMCOM);
    END LOOP;
END;
/
```

5.2. Détection des dépendances fonctionnelles entre les colonnes d'une table

J02. Développez une procédure PL/SQL qui permet de vérifier certaines dépendances fonctionnelles (dépendances sémantiques) dans la table des clients !

5.3. Détection des lignes similaires ou en double d'une table

J03. Développez une procédure PL/SQL qui permet d'éliminer les doubles dans la table des clients !



5.4. Qualité des Données : iDQMS-An intelligent Data Quality Management System tool

Développez un noyau d'outil ETL [Extract-Transform-Load] (iDQMS tool : An intelligent Data Quality Management System tool → >>>>> SmartDATA) qui permet de traiter (gérer la qualité) d'une source de données CSV.

5.4.1. Quelles sont les anomalies dans une source de données pour laquelle **on ne connaît pas la structure** (Fichier CSV / Table) ?

5.4.2. Comment les détecter ?

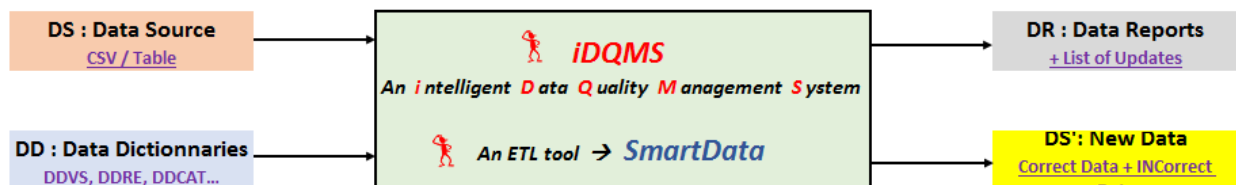
5.4.3. Comment les corriger ?

5.4.4. Quelles sont les dépendances sémantiques entre les colonnes ?

5.4.6. Comment détecter les lignes en double et/ou similaires

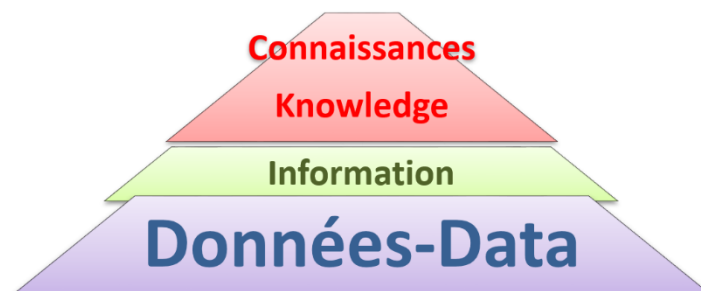
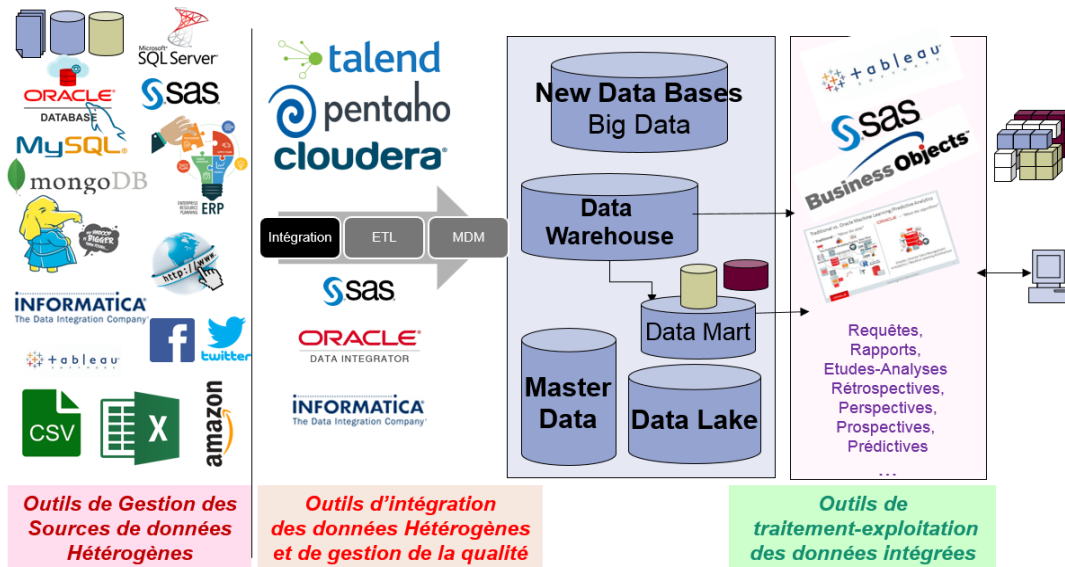
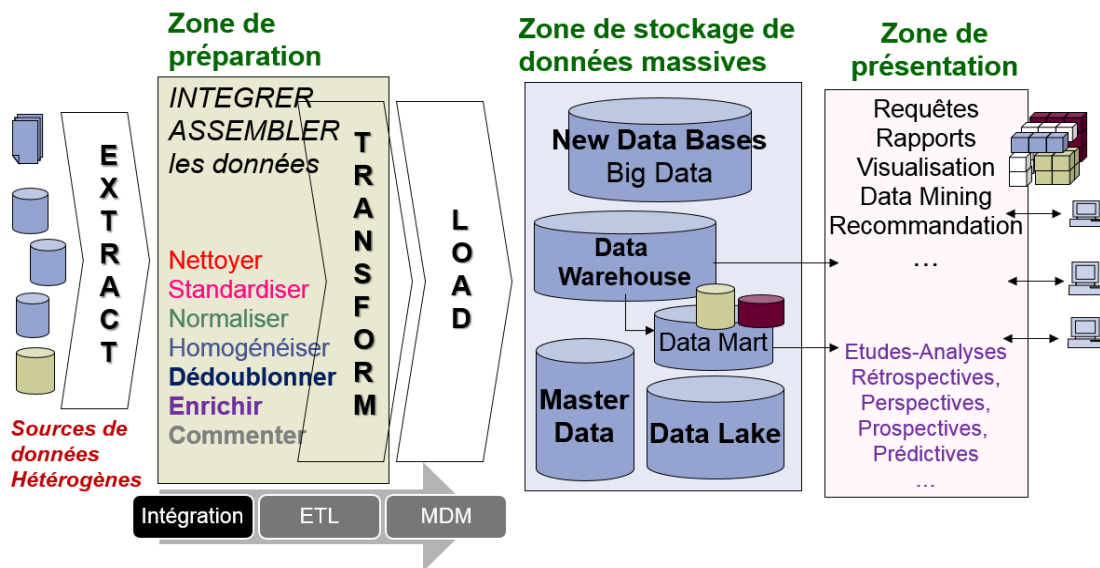
Voir le document du Projet CSV2TAB

Projet à développer avec les SGBD Oracle, MySQL et MongoDB



	A	B	C	D	E	F	G	H	I	J	K
1											
2											
3											
4											
5											
6											
7											
8											
9											
10											
11											
12											
13											

6. Intégration de données ; Exploration Intelligente des Données et Décisionnel !



**From DATA BASES to
DATA WAREHOUSE, DATA MART, MASTER DATA, DATA LAKE**

**Bases de Données - Entrepôts de Données – Magasins de Données
Référentiel de Données, Lacs de données**

Think DIFFERENTLY, BIGGER and SMARTER !

**Votre mission, si vous l'acceptez, est : The Excellence in Data Use !
Si vous échouez, nous nierons avoir eu accès à vos agissements !**

C Koi alors Big Data ? → c 1telligent ? → C Koi IA ?



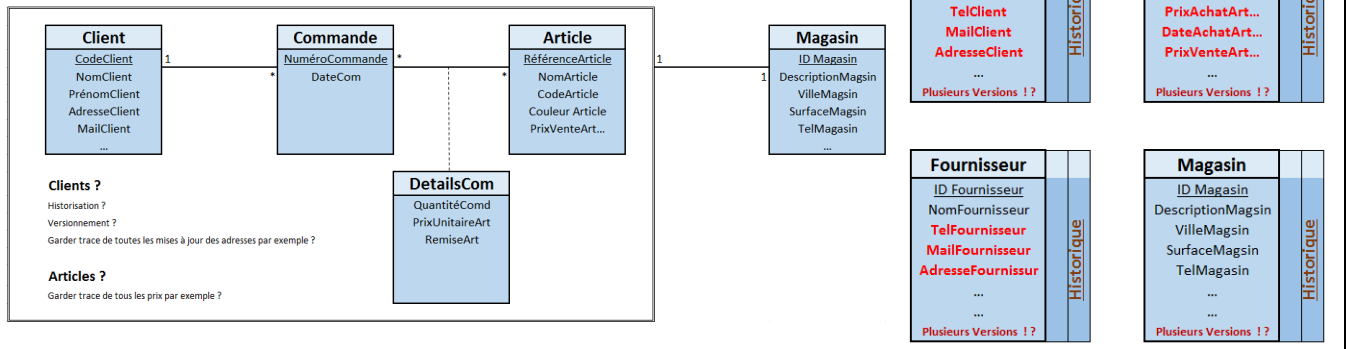
DATA MANGEMENT & MACHINE LEARNING, the future!
(DATA BASE, DATA WAREHOUSE, MASTER DATA, DATA LAKE, ... BIG DATA!)



Une Gestion Commerciale Intelligente (GCI)

Version 2

G2



Les dictons du jour !

→ Aujourd'hui, j'arrête de fumer ; Le TABAC t'ABAT ; M T Dents !

Fumer nuit gravement à ta santé et à celle de ton entourage !

→ SMILE and the World SMILES with you!

→ MANGER + MANGER = GROSSIR

(??? M, A, N, G, E, R, O, S, S, I)

Manger & Bouger ;

Pour votre santé mangez 5 fruits et légumes par jour ;

Pour votre santé ne mangez pas trop gras, trop salé, trop sucré ;

Pour votre santé faites une activité physique régulière ;

Pour votre santé faites 30 mn de marche par jour !

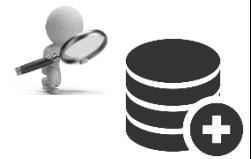
→ Pour votre santé faites des BD !!! Dr. M. Faouzi Boufarès



6.1. Historique des données (Plusieurs versions sont archivées)

Voir SCRIPT SQL pour la Version 2 de la BD GCI : G2CreatInsertDon.sql

→ La création des tables (création de la structure des données, structures vide !)
→ La création des index (création des accélérateurs de traitements !)
→ La création des commentaires !



→ La création des tables historiques afin de garder plusieurs versions des données (création de la structure des données, structures vide !)
→ La création des déclencheurs afin d'enregistrer automatiquement TOUS les mouvements (toutes les mises à jour -Insertion, Modification et Suppression !)

→ La création des données (insertions des données dans la BD créée)



Requêtes J_i (SQL & PL/SQL)

Déclencheurs-Triggers ; Procédures-Fonctions ;

Cohérence globale des données ; Qualité des données

J04. Afin de garder trace (sauvegarde & suivi) des prix unitaires de chaque article, développez un déclencheur qui permet d'enregistrer chaque mise-à-jour de table des articles !

😊 ➡ Quelques indications de Réponses aux Requêtes J_i

J04. Afin de garder trace (sauvegarde & suivi) des prix unitaires de chaque article, développez un déclencheur qui permet d'enregistrer chaque mise-à-jour de tables des articles !

Pour développer le **déclencheur (Trigger)** sur les mouvements des articles, il est nécessaire de **créer une table HISTORIQUE_MVTS_ARTICLES**.

```
DROP TABLE HISTORIQUE_MVTS_ARTICLES;
CREATE TABLE HISTORIQUE_MVTS_ARTICLES
(
    NOMUSER          VARCHAR2(15),
    TYPEMVT           VARCHAR2(15),
    REFART            VARCHAR2(10),
    NOMART             VARCHAR2(50),
    PVART              NUMBER(10, 2),
    PAART              NUMBER(10, 2),
    QSART              NUMBER(3),
    DATMVT             DATE,
    CONSTRAINT NN_HIST_ARTICLES_NOMART CHECK(NOMART IS NOT NULL),
    CONSTRAINT NN_HIST_ARTICLES_NOMUSER CHECK(NOMUSER IS NOT NULL),
    CONSTRAINT NN_HIST_ARTICLES_NOMTYPEMVT CHECK(TYPEMVT IS NOT NULL)
);
```



```

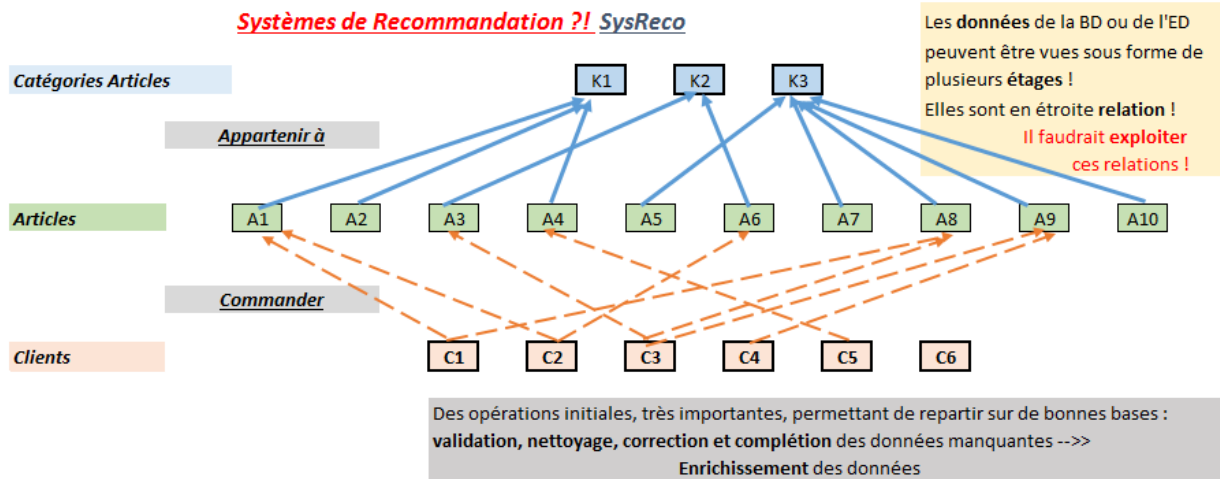
CREATE OR REPLACE TRIGGER MOUVEMENTS_ARTICLES
AFTER UPDATE OR INSERT OR DELETE ON ARTICLES
FOR EACH ROW
DECLARE
    V_REFART      VARCHAR2(10)  := :OLD.REFART;
    V_NOMART      VARCHAR2(50)  := :OLD.NOMART;
    V_PVART       NUMBER(10, 2) := :OLD.PVART;
    V_PAART       NUMBER(10, 2) := :OLD.PAART;
    V_QSART       NUMBER(3)     := :OLD.QSART;
    VI_REFART     VARCHAR2(10)  := :NEW.REFART;
    VI_NOMART     VARCHAR2(50)  := :NEW.NOMART;
    VI_PVART      NUMBER(10, 2) := :NEW.PVART;
    VI_PAART      NUMBER(10, 2) := :NEW.PAART;
    VI_QSART      NUMBER(3)     := :NEW.QSART;

BEGIN
    IF UPDATING THEN
        INSERT INTO HISTORIQUE_MVTS_ARTICLES (NOMUSER, TYPEMVT, REFART, NOMART,
PVART, PAART, QSART, DATMVT)
        VALUES (USER, 'UPDATE', V_REFART, V_NOMART, V_PVART, V_PAART, V_QSART, SYSDATE);
        END IF;
    IF DELETING THEN
        INSERT INTO HISTORIQUE_MVTS_ARTICLES (NOMUSER, TYPEMVT, REFART, NOMART,
PVART, PAART, QSART, DATMVT)
        VALUES (USER, 'DELETE', V_REFART, V_NOMART, V_PVART, V_PAART, V_QSART,
SYSDATE);
        END IF;
    IF INSERTING THEN
        INSERT INTO HISTORIQUE_MVTS_ARTICLES (NOMUSER, TYPEMVT, REFART, NOMART,
PVART, PAART, QSART, DATMVT)
        VALUES (USER, 'INSERT', VI_REFART, VI_NOMART, VI_PVART, VI_PAART, VI_QSART,
SYSDATE);
        END IF;
END;
/

```

6.2. Introduction aux Systèmes de Recommandation (SysReco)

Manipulation intelligente des données : Système de recommandation



6.2.1. Construisez la vue « V_i » qui contient les articles commandés par un client donné « C_i ». Généralisez dans une procédure PL/SQL !

6.2.2. Construisez la vue « V_{ij} » qui contient les articles commandés à la fois par les deux clients « C_i » et « C_j ».

On pourrait considérer que les deux clients « C_i » et « C_j » sont « amis-voisins ou appartiennent au même cluster » si et seulement si la vue « V_{ij} » est non vide (90%)!

On pourrait alors **recommander** ou **proposer** à l'un des deux clients, soit « C_j », les articles de « C_i » - son ami- qu'il n'a pas commandés !

On pourrait aussi **recommander** ou **proposer** à l'un des deux clients, soit « C_j », les articles de « C_i » - son ami- qu'il n'a pas commandés + mais qui doivent appartenir à la même catégorie !

Discussions & Généralisation ! (voir le script sql)

		ARTICLES												
		QrC	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12
CLIENTS	C1	2	3	1	1	1								
	C2	1	1	1										
	C3	5	2	1		1								
	C4		3	3	3	4								
	C5													
	C6							1	1	1				
	C7							1	1					
	C8													
	C9	5									2	2		
	C10	1									2			
	C11	2									1	1		
	C12													
	C13	3	1					4						
	C14	2	1					4	5					
	C15													
	C16					3		1			1			
	C17					1					1			
	C18													
	C19													
	C20	3						1			???	7		
Total commandes		24	11	6	8	6	12	7	1	7	10	0	0	

Cluster 1
Groupe 1
Classe 1

Cluster 2

Cluster 3

Cluster 4

Cluster 5

Nouveau client !

6.3. Intégration de Données (Des bases aux entrepôts des données : Utilisation intelligente des données-Intelligent use of data ; Aide à la décision)

6. Intégration de données Exploration Intelligente des Données et Décisionnel !

Les décideurs de l'entreprise BB BOBO veulent étudier certains phénomènes !

Il s'agit, dans un premier temps par exemple, d'étudier pour tous ses points de ventes le chiffre d'affaires par année ensuite par période ! **Les décideurs constatent alors la baisse continue du chiffre d'affaires !**

Ils tentent de trouver une explication et font appel aux **informaticiens spécialistes**, les « **Data Scientists** » !
Les questions qui se posent sont « **LE POURQUOI ?** » ; « Où sont les difficultés, qu'est ce qui ne marche pas ? »

Quelles sont les données qui vont participer à l'établissement des différents calculs ?

L'entreprise BB BOBO dispose de **toutes les BD en provenance des différents pays** où elle est implantée !

Dans un premier temps, on travaillera sur une BD d'un seul pays !

Dans un deuxième temps, on intégrera toutes les BD de tous les pays concernés !

Intégration de données hétérogènes, construction d'une nouvelle base de données qui assemble toutes les bases de données existantes, construction d'un entrepôt de données

Ils veulent analyser et croiser les données de plusieurs manières sur une longue période.

Par exemple, ils demandent de :

- étudier les liens (**corrélations**) qui existent entre les articles, ou encore
- construire des **rapports**, des **tableaux de synthèses** et des **graphiques** selon les exemples présentés dans les figures 6 ci-dessous.

6.3.1. Intégration des Données !

Construisez une nouvelle **table** de nom **BD-P1-Pays** qui est la **fusion (intégration)** de toutes les tables de la BD GCI **pour un pays P1 donné**.

Nous nous poserons la question « Est-il nécessaire d'intégrer/fusionner les différentes tables de la BD GCI pour un pays donné ? » dans une seule table !

6.3.2. Intégration des Données !

Construisez une nouvelle **BD** de nom **MasseDeDonnees** qui est la fusion (intégration) de toutes les BD GCI **pour tous les pays**.

Comment intégrer/fusionner les différentes BD GCI de tous les pays ?

Ces intégrations de données pourraient être faites en utilisant :

- un système de gestion de données (SGBD) tel qu'Oracle ou
- un des outils d'intégration de données (ETL).

6.3.3. Extraction de connaissances à partir des Données !

6.3.3.1. Quels sont les articles qui sont toujours vendus ensemble pour un pays donné ?

6.3.3.2. Est-ce que les remises appliquées ont eu une certaine influence sur les quantités commandées pour un pays donné ?

6.3.3.3. Etudiez la corrélation entre la variation des prix et celle du chiffre d'affaires pour un pays donné

6.3.3.4. Etudiez la corrélation entre la variation des prix et celle du chiffre d'affaires pour tous les pays

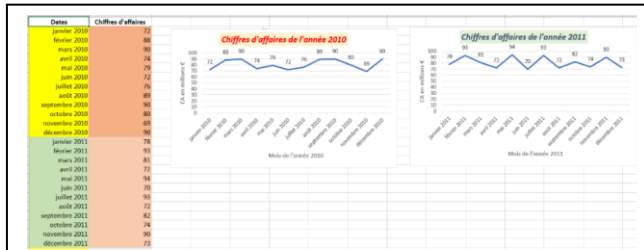


Figure 6.1 : Chiffre d'affaires pour l'année 2010 ; Chiffre d'affaires pour l'année 2011

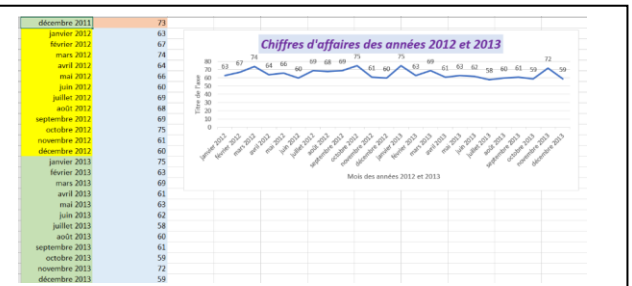


Figure 6.2 : Chiffre d'affaires pour les deux années 2012 et 2013

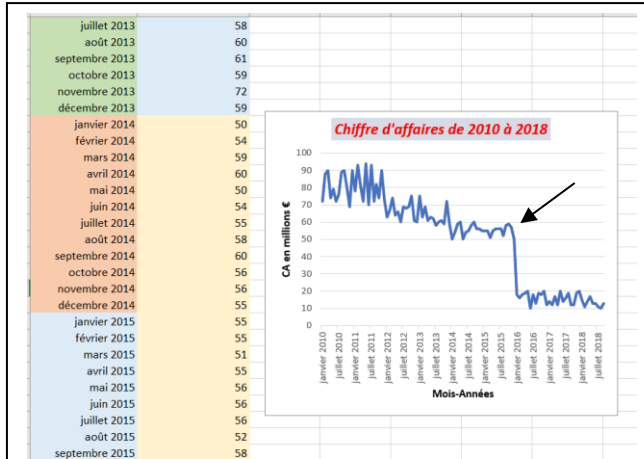


Figure 6.3 : Chiffre d'affaires de 2010 à 2018

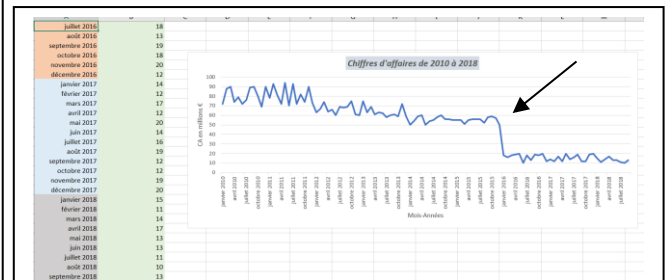


Figure 6.4 : Chiffre d'affaires de 2010 à 2018

Statistiques sur le chiffre d'affaires ; Période de 2010 à 2018					
Année	Total	Moyenne	Minimum	Maximum	Écart type
2010	969	80,75	69	90	8,20
2011	972	81,00	70	94	9,26
2012	796	66,33	60	75	5,02
2013	762	63,50	58	75	5,50
2014	667	55,58	50	60	3,37
2015	660	55,00	50	59	2,73
2016	201	16,75	10	20	3,31
2017	187	15,58	12	20	3,32
2018	117	13,00	10	17	2,18
Global	5331	50,77	10	94	26,21

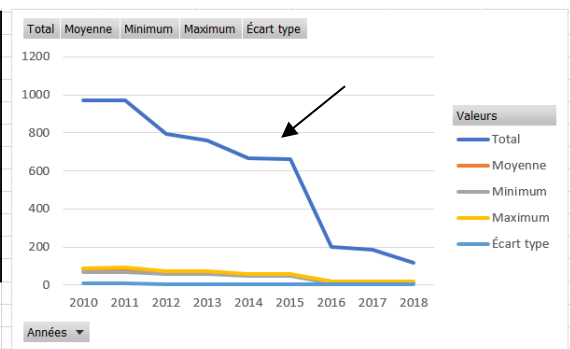


Figure 6.5 : Statistiques sur le chiffre d'affaires – Période de 2010 à 2018

Si l'on dispose des données suivantes cf document sql :

```
CREATE TABLE CACLI (DATECA DATE, MONTANTCA NUMBER);
INSERT INTO CACLI VALUES ('01-01-2010', 72); INSERT INTO CACLI VALUES ('01-02-2010', 88);
INSERT INTO CACLI VALUES ('01-03-2010', 90); INSERT INTO CACLI VALUES ('01-04-2010', 74);
INSERT INTO CACLI VALUES ('01-05-2010', 79); Etc... COMMIT ;
```

6.3.4. Statistiques sur le chiffre d'affaires par année – Période de 2010 à 2018 ! (Construction des vues (matérialisées ou non) pour stocker les résultats des calculs

6.3.4.1. Calculez le chiffre d'affaires par année (construction de vues) pour un pays donné

6.3.4.2. Calculez le chiffre d'affaires par période pour un pays donné

6.3.4.3. Calculez le chiffre d'affaires par catégorie des clients pour un pays donné

6.3.4.4. Calculez le chiffre d'affaires par ville pour un pays donné

6.3.4.5. Calculez le chiffre d'affaires par période, par catégorie et par ville pour un pays donné

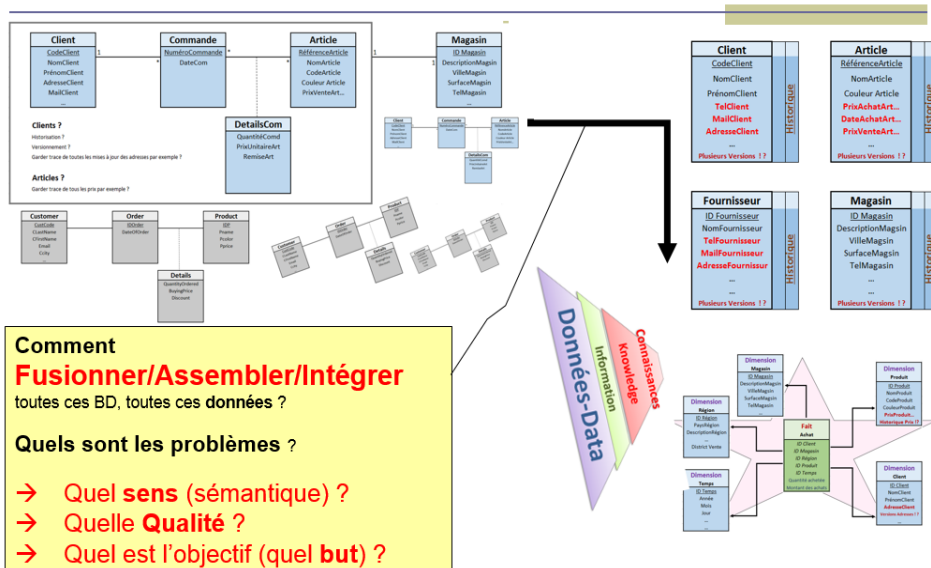
6.3.4.6. Calculez le chiffre d'affaires par ville, par période et par catégorie pour un pays donné

6.3.4.7. Calculez le chiffre d'affaires par pays et par année

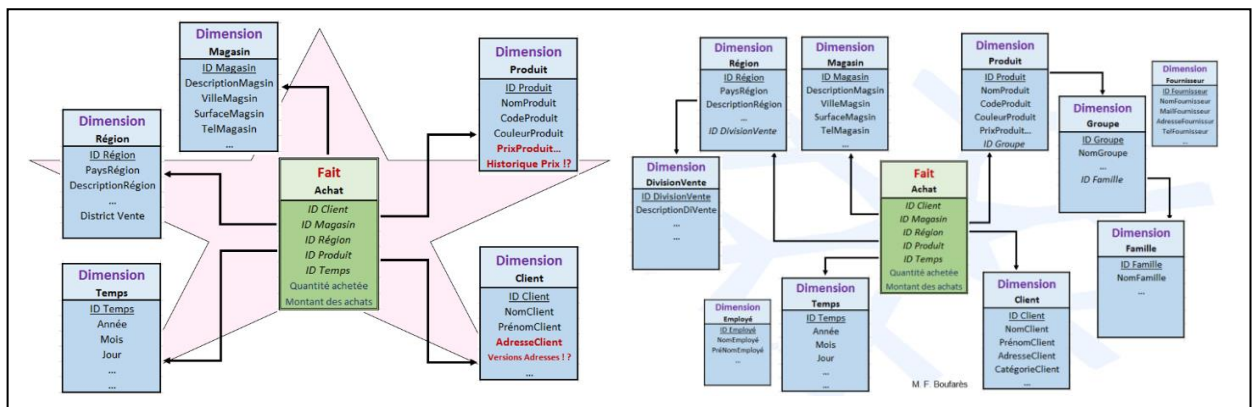
7. Entrepôts de données : Les schémas en étoile : à la belle étoile ! Where is the Data ? Is it in the House or in the Lake?

7. Les schémas en étoile : à la belle étoile ! Where is the Data ? Is it in the House or in the Lake?

Exemple de BD GesCom à intégrer

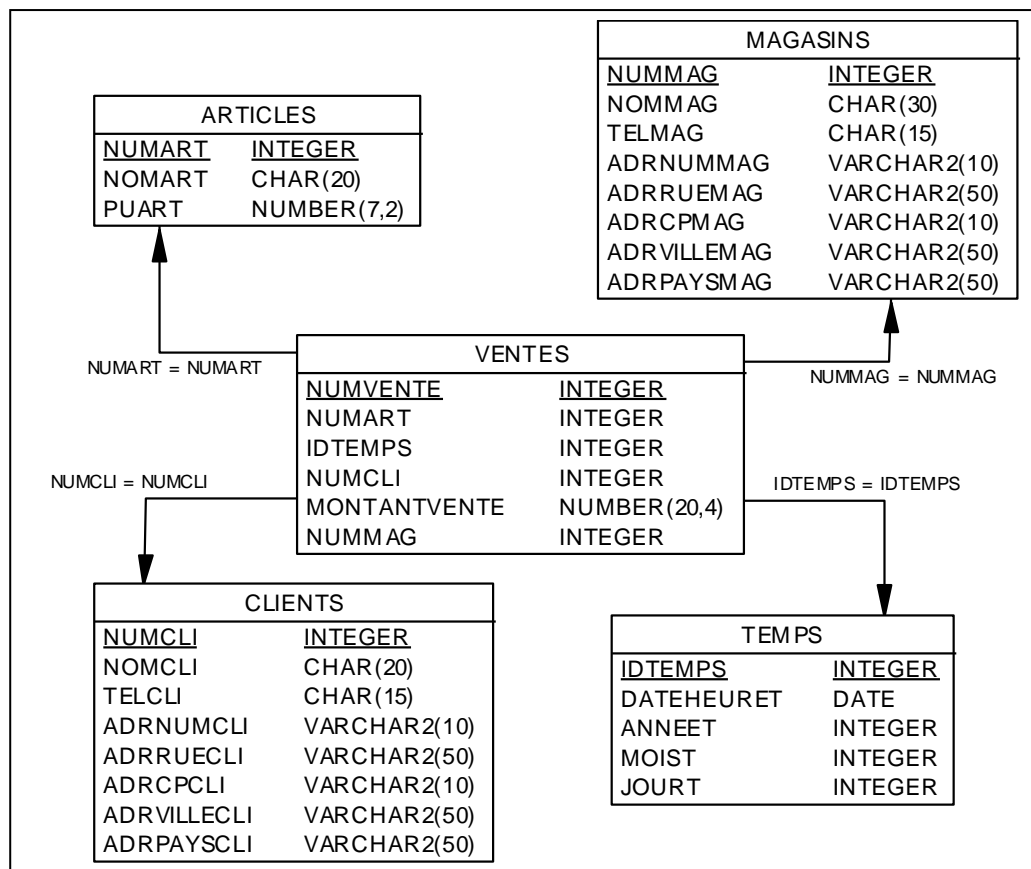
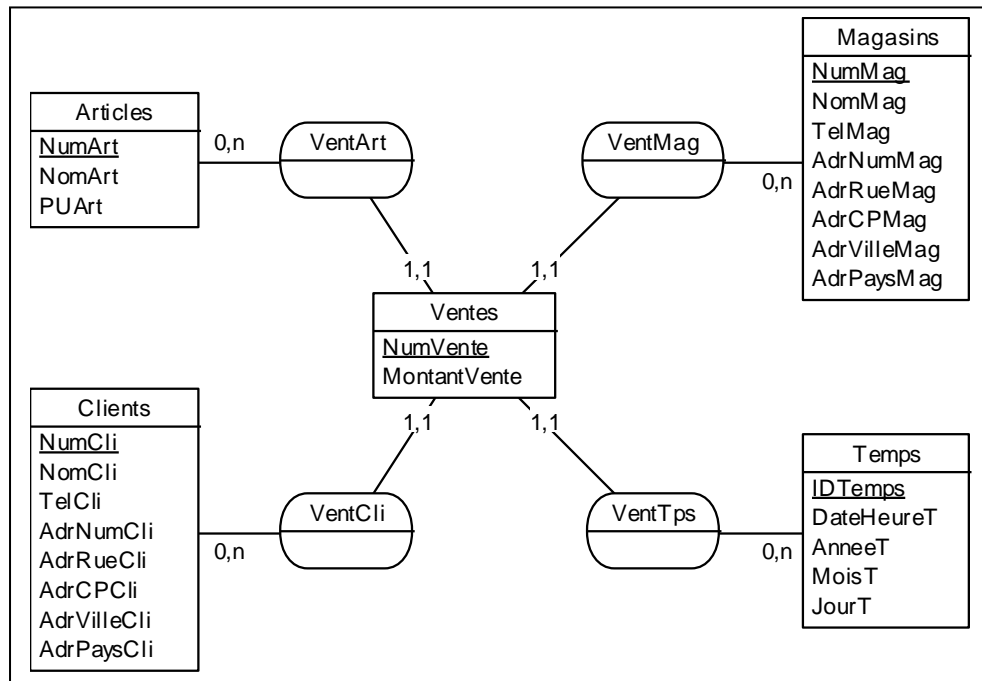


7.1. Modélisation des Données (Des bases aux entrepôts des données)



Etant donné le schéma, en étoile, de BD ci-dessous : BD GesComI***DW.

Construction de Gros Volumes de Données : Développez le « mécanisme » qui permet de passer d'une BD GesComI (relative à un seul magasin et/ou un pays) vers BD GesComI***DW, puis généraliser pour « intégrer » (« fusionner ») toutes les BD GesComI de tous les magasins de tous les pays. **Discussions !**



7.2. Optimisation des requêtes

Etant donné le schéma de BD ci-dessus : BD GesComI***DW.

Considérons le *schéma en étoile* ainsi que les tables (ou vues *matérialisées*) V1, V2 et V3 définies ci-dessous.

Les *vues matérialisées* peuvent servir, afin **d'améliorer les performances** en manipulant les Bases et Entrepôts de données, pour réécrire les requêtes. Etant donné les deux requêtes Q et Q' et les trois vues matérialisées V1, V2 et V3.

Questions :

Montrez que la requête Q donne le même résultat que la requête Q' !

Tester les temps de réponses sur de gros volumes de données !

La requête Q : Total des ventes des magasins en France ou en Belgique -pour la période de 2001 à 2003- par ville et année.

```
→ select ADRVILLEMAG, ANNEET, sum(MONTANTVENTE) as Montant
from VENTES, MAGASINS, TEMPS
where VENTES.NUMMAG=MAGASINS.NUMMAG and VENTES.IDTEMPS=TEMPS.IDTEMPS
and (upper(MAGASINS.ADRPAYSMAG)='FRANCE' or upper(MAGASINS.ADRPAYSMAG)='BELGIQUE')
and TEMPS.ANNEET between 2001 and 2003 group by ADRVILLEMAG, ANNEET ;
```

Les (Tables intermédiaires) vues matérialisées

→ V1 : Total des ventes des magasins à partir de 2002, par ville et année

```
create materialized view v1 (Ville, Annee, Montant1) as
(select ADRVILLEMAG, ANNEET, sum(MONTANTVENTE)
from VENTES, MAGASINS, TEMPS
where VENTES.NUMMAG=MAGASINS.NUMMAG and
VENTES.IDTEMPS=TEMPS.IDTEMPS and TEMPS.ANNEET >= 2002 group by ADRVILLEMAG, ANNEET);
```

→ V2 : Total des ventes des magasins en France par ville, année et mois

```
create or replace view v2 (Ville, Annee, Mois, Montant2) as
(select ADRVILLEMAG, ANNEET, MOIST, sum(MONTANTVENTE)
from VENTES, MAGASINS, TEMPS
where VENTES.NUMMAG=MAGASINS.NUMMAG and VENTES.IDTEMPS=TEMPS.IDTEMPS
and upper(MAGASINS.ADRPAYSMAG)='FRANCE' group by ADRVILLEMAG, ANNEET, MOIST);
```

→ V3 : Total des ventes des magasins en Belgique -avant <= 2001- par ville, année et mois

```
create or replace view v3 (Ville, Annee, Mois, Montant3) as
(select ADRVILLEMAG, ANNEET, MOIST, sum(MONTANTVENTE)
from VENTES, MAGASINS, TEMPS
where VENTES.NUMMAG=MAGASINS.NUMMAG and VENTES.IDTEMPS=TEMPS.IDTEMPS
and upper(MAGASINS.ADRPAYSMAG)='BELGIQUE' and TEMPS.ANNEET <= 2001
group by ADRVILLEMAG, ANNEET, MOIST);
```

La requête Q' utilise des vues matérialisées

Total des ventes des magasins en France ou en Belgique -pour la période de 2001 à 2003- par ville et année

```
select ADRVILLEMAG, annee, Montant1
from v1, (select distinct ADRVILLEMAG from MAGASINS where
upper(MAGASINS.ADRPAYSMAG)='FRANCE' or upper(MAGASINS.ADRPAYSMAG)='BELGIQUE') s1
where v1.ville = s1.ADRVILLEMAG and annee <= 2003
union
select ville, annee, sum(Montant2) from v2 where annee=2001 group by ville, annee
union
select s2.ADRVILLEMAG, Annee, sum(Montant3) from v3, (select distinct ADRVILLEMAG from MAGASINS) s2
where v3.ville = s2.ADRVILLEMAG and annee = 2001 group by s2.ADRVILLEMAG, annee ;
```

