

Données, base de données et SGBD

Modèle conceptuel

Modèle logique

Modèle physique

Importation/exportation de données

Requêtes

Logiciel R et bases de données

Utilisation du package RODBC



Sandrine Auzoux

UR AIDA

Jean-Christophe Soulié

UR Recyclage et Risque

Conception de bases
de données
expérimentales à des
fins de modélisation

Interfaçage avec R



Modèle hiérarchique des connaissances

Donnée :

résultat direct d'une mesure (faits, observations, éléments bruts)

40°C

Information :

donnée interprétée (qui, quoi, quand, où)

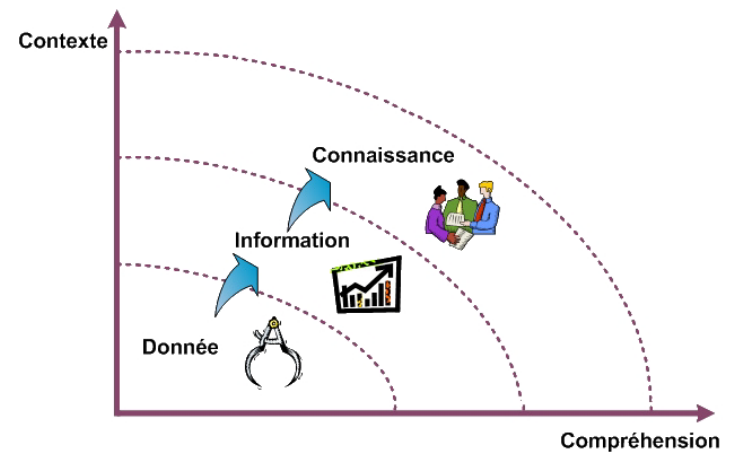
Température de l'air en degré Celsius à 14h
à Bobo-Dioulasso

Connaissance :

information comprise (comment, pourquoi)

Il fait chaud à 14h à Bobo-Dioulasso

➡ J'allume la climatisation.



Chaîne linéaire du modèle hiérarchique
des connaissances (Françoise Rossion, 2008)

Les métadonnées

Les métadonnées sont des "données qui décrivent des données".

- **Information structurée** associée à une donnée (ou un objet, un document...)
- **Documentation** qui permet à l'utilisateur de comprendre la donnée

Les métadonnées sont un maillon essentiel pour le **partage des données** et l'**interopérabilité des bases de données** lorsqu'elles sont structurées dans des **dictionnaires de données**.



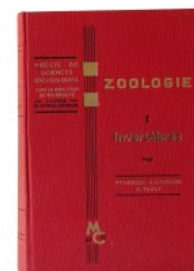
Il est conseillé de produire les métadonnées au **moment de la collecte** ou de la **création** des données plutôt qu'à posteriori.

Des objets discernés par tous



Un objet sans étiquette n'est connu que de son auteur

Métadonnées



Base de données relationnelle



Une base de données est un **ensemble de données**, stockées de façon :

- ✓ **Persistante** : de stockage permanent
- ✓ **De redondance minimale** : la même information n'est présente qu'une fois (unicité)
- ✓ **Exhaustive** : la base de données contient toutes les informations requises pour le service attendu
- ✓ **Structuré** : la structure est définie dans un **schéma** (« modèle »)
- ✓ Elle est gérée par un **système de gestion de bases de données**.

Une base de données peut-être stockée de manière :

- ✓ **Locale** → utilisable **par une personne** sur un ordinateur
- ✓ **Distante** → les données sont stockées sur un ou plusieurs serveurs et sont accessibles simultanément **par plusieurs utilisateurs**

Système de gestion de bases de données

Le **Système de Gestion de Bases de Données** (SGBD) est un **logiciel** qui permet de manipuler les informations stockées dans une base de données, en jouant le **rôle d'interface** entre les utilisateurs et la base de données.

Les fonctions principales d'un SGBD :

1. La **description** de la structure de la base de données
2. La **manipulation** des données :
 - Opérations de création, insertion, modification, suppression et recherche
3. Le **contrôle** des données : **intégrité** (qualité, cohérence), **fiabilité** (perte), **confidentialité** (droits d'accès), **partage** et **sécurité** (accès concurrents)
4. Une interface qui permet à l'utilisateur de réaliser les 3 fonctions précédentes

Quelques SGBD relationnels connus

SGBD propriétaire, client/serveur et multiplateformes :

- ✓ Oracle (Oracle Corporation)
- ✓ SQL Server (Microsoft)
- ✓ Sybase (SAP Company)
- ✓ DB2 (IBM)
- ✓ HyperFileSQL (PcSoft)

ORACLE®



SGBD libre et client/serveur :

MySQL (MySQL AB/Oracle) et PostgreSQL (Michael Stonebraker)

SGBD faisant partie d'une suite bureautique:

- ✓ Access (Microsoft - propriétaire)
- ✓ OOo Base (Fondation Apache - libre)



Pourquoi une base de données ?

A	B	C	D	E	F	I
CTFT id	Family	Species	Specific gravity	Country	Sub-continent	Collector's name
1	LAURACEAE	Cinnamomum ilicioides A.Chev.	0.51	Vietnam	Asie continentale	Jard.Col.
2	MAGNOLIACEAE	Magnolia tsiampacca (L.) Figlar & Noot.	0.6	Chine	Asie continentale	SBC
3	MAGNOLIACEAE	Liriodendron chinense Sargent		Amerique Nord	Amerique du Nord	SBC
4	MAGNOLIACEAE	Liriodendron tulipifera L.		France	Europe	SBC
5	MAGNOLIACEAE	Magnolia obovata Thunb.	0.45	Japon	Asie continentale	SBC
6	MAGNOLIACEAE	Magnolia kobus DC.		Japon	Asie continentale	SBC
7	MAGNOLIACEAE	Magnolia sp		Vietnam	Asie continentale	Ag.Indoch.
8	MAGNOLIACEAE	Magnolia fordiana (Oliv.) Hu	0.45	Vietnam	Asie continentale	Bertin
9	MAGNOLIACEAE	Magnolia sp		Etats-Unis n.p.	Amerique du Nord	SBC
10	MAGNOLIACEAE	Magnolia blumei Prantl	0.46	Vietnam	Asie continentale	Bertin
11	MAGNOLIACEAE	Magnolia sp		Vietnam	Asie continentale	Ag.Indoch.
12	MAGNOLIACEAE	Magnolia champaca (L.) Baill. ex Pierre		Reunion	Ocean Indien	Ag.Col.
13	MAGNOLIACEAE	Magnolia hypolampra (Dandy) Figlar	0.59	Vietnam	Asie continentale	Bertin
14	MAGNOLIACEAE	Magnolia hypolampra (Dandy) Figlar		Vietnam	Asie continentale	SBC
15	MAGNOLIACEAE	Magnolia hypolampra (Dandy) Figlar		Vietnam	Asie continentale	SBC
16	MAGNOLIACEAE	Magnolia dodecapetala (Lam.) Govaerts	0.62	Antilles françaises	Amerique Centrale et Antilles	SBC
17	CANELLACEAE	Cinnamosma madagascariensis Danguy	0.99	Madagascar	Ocean Indien	SBC
18	CERCIDIPHYLLACEAE	Cercidiphyllum japonicum Siebold & Zucc.		Japon	Asie continentale	SBC
19	ANNONACEAE	Guatteria caribaea Urb.		Antilles françaises	Amerique Centrale et Antilles	SBC
20	ANNONACEAE	Annona reticulata L.	0.52	Antilles françaises	Amerique Centrale et Antilles	Ag.Col.
21	ANNONACEAE	Anonidium mannii Engl. & Diels	0.33	Cameroun	Afrique	Hedin
22	ANNONACEAE	Anonidium mannii Engl. & Diels	0.37	Cameroun	Afrique	Foury 2
23	ANNONACEAE	Fenerivia oligosperma (Danguy) R.M.K.Saunders		Madagascar	Ocean Indien	SBC
24	ANNONACEAE	Fenerivia oligosperma (Danguy) R.M.K.Saunders	0.52	Madagascar	Ocean Indien	Thouvenot
25	ANNONACEAE	Fissistigma latifolium (Dunal) Merr,	0.33	Vietnam	Asie continentale	SBC
26	ANNONACEAE	Cleistopholis glauca Pierre	0.42	Cameroun	Afrique	Foury 2
27	ANNONACEAE	Cleistopholis patens (Benth.) Engl. & Diels	0.3	Gabon	Afrique	Jard.Col.
28	ANNONACEAE	Cleistopholis patens (Benth.) Engl. & Diels	0.28	Gabon	Afrique	Jard.Col.
29	ANNONACEAE	Cleistopholis patens (Benth.) Engl. & Diels	0.29	Congo	Afrique	Sargos
30	ANNONACEAE	Cleistopholis staudtii Engl. & Diels	0.35	Congo	Afrique	Sargos
31	ANNONACEAE	Cleistopholis staudtii Engl. & Diels	0.31	Cameroun	Afrique	Foury 2
32	ANNONACEAE	Cleistopholis staudtii Engl. & Diels	0.37	Gabon	Afrique	Rabourdin
33	ANNONACEAE	Annickia chlorantha (Oliv.) Setten & Maas	0.53	Gabon	Afrique	Rabourdin
34	ANNONACEAE	Annickia chlorantha (Oliv.) Setten & Maas	0.57	Cameroun	Afrique	Foury 2
35	ANNONACEAE	Annickia chlorantha (Oliv.) Setten & Maas	0.56	Cameroun	Afrique	Hedin
36	ANNONACEAE	Annickia chlorantha (Oliv.) Setten & Maas	0.44	Cameroun	Afrique	SBC
37	ANNONACEAE	Guatteria caribaea Urb.	0.52	Antilles françaises	Amerique Centrale et Antilles	Ag.Col.

Cirad wood collection index



Pourquoi une base de données ?

A	B	C	D	E	F	I
CTFT id	Family	Species	Specific gravity	Country	Sub-continent	Collector's name
1	LAURACEAE	Cinnamomum ilicioides A.Chev.	0.51	Vietnam	Asie continentale	Jard.Col.
2	MAGNOLIACEAE	Magnolia tsiampacca (L.) Figlar & Noot.	0.6	Chine	Asie continentale	SBC
3	MAGNOLIACEAE	Liriodendron chinense Sargent		Amerique Nord	Amerique du Nord	SBC
4	MAGNOLIACEAE	Liriodendron tulipifera L.		France	Europe	SBC
5	MAGNOLIACEAE	Magnolia obovata Thunb.	0.45	Japon	Asie continentale	SBC
6	MAGNOLIACEAE	Magnolia kobus DC.		Japon	Asie continentale	SBC
7	MAGNOLIACEAE	Magnolia sp		Vietnam	Asie continentale	Ag.Indoch.
8	MAGNOLIACEAE	Magnolia fordiana (Oliv.) Hu	0.45	Vietnam	Asie continentale	Bertin
9	MAGNOLIACEAE	Magnolia sp		Etats-Unis n.p.	Amerique du Nord	SBC
10	MAGNOLIACEAE	Magnolia blumei Prantl	0.46	Vietnam	Asie continentale	Bertin
11	MAGNOLIACEAE	Magnolia sp		Vietnam	Asie continentale	Ag.Indoch.
12	MAGNOLIACEAE	Magnolia champaca (L.) Baill. ex Pierre		Reunion	Ocean Indien	Ag.Col.
13	MAGNOLIACEAE	Magnolia hypolampra (Dandy) Figlar	0.59	Vietnam	Asie continentale	Bertin
14	MAGNOLIACEAE	Magnolia hypolampra (Dandy) Figlar		Vietnam	Asie continentale	SBC
15	MAGNOLIACEAE	Magnolia hypolampra (Dandy) Figlar		Vietnam	Asie continentale	SBC
16	MAGNOLIACEAE	Magnolia dodecapetala (Lam.) Govaerts	0.62	Antilles françaises	Amerique Centrale et Antilles	SBC
17	CANELLACEAE	Cinnamosma madagascariensis Danguy	0.99	Madagascar	Ocean Indien	SBC
18	CERCIDIPHYLLACEAE	Cercidiphyllum japonicum Siebold & Zucc.		Japon	Asie continentale	SBC
19	ANNONACEAE	Guatteria caribaea Urb.		Antilles françaises	Amerique Centrale et Antilles	SBC
20	ANNONACEAE	Annona reticulata L.	0.52	Antilles françaises	Amerique Centrale et Antilles	Ag.Col.
21	ANNONACEAE	Anonidium mannii Engl. & Diels	0.33	Cameroun	Afrique	Hedin
22	ANNONACEAE	Anonidium mannii Engl. & Diels	0.37	Cameroun	Afrique	Foury 2
23	ANNONACEAE	Fenerivia oligosperma (Danguy) R.M.K.Saunders		Madagascar	Ocean Indien	SBC
24	ANNONACEAE	Fenerivia oligosperma (Danguy) R.M.K.Saunders	0.52	Madagascar	Ocean Indien	Thouvenot
25	ANNONACEAE	Fissistigma latifolium (Dunal) Merr.	0.33	Vietnam	Asie continentale	SBC
26	ANNONACEAE	Cleistopholis glauca Pierre	0.42	Cameroun	Afrique	Foury 2
27	ANNONACEAE	Cleistopholis patens (Benth.) Engl. & Diels	0.3	Gabon	Afrique	Jard.Col.
28	ANNONACEAE	Cleistopholis patens (Benth.) Engl. & Diels	0.28	Gabon	Afrique	Jard.Col.
29	ANNONACEAE	Cleistopholis patens (Benth.) Engl. & Diels	0.29	Congo	Afrique	Sargos
30	ANNONACEAE	Cleistopholis staudtii Engl. & Diels	0.35	Congo	Afrique	Sargos
31	ANNONACEAE	Cleistopholis staudtii Engl. & Diels	0.31	Cameroun	Afrique	Foury 2
32	ANNONACEAE	Cleistopholis staudtii Engl. & Diels	0.37	Gabon	Afrique	Rabourdin
33	ANNONACEAE	Annickia chlorantha (Oliv.) Setten & Maas	0.53	Gabon	Afrique	Rabourdin
34	ANNONACEAE	Annickia chlorantha (Oliv.) Setten & Maas	0.57	Cameroun	Afrique	Foury 2
35	ANNONACEAE	Annickia chlorantha (Oliv.) Setten & Maas	0.56	Cameroun	Afrique	Hedin
36	ANNONACEAE	Annickia chlorantha (Oliv.) Setten & Maas	0.44	Cameroun	Afrique	SBC
37	ANNONACEAE	Guatteria caribaea Urb.	0.52	Antilles françaises	Amerique Centrale et Antilles	Ag.Col.

Cirad wood collection index



Tableur Vs SGBD ? 1/2

"Je gère ma base de données sur Excel..." : non-sens

Le **tableur** est un logiciel qui permet de traiter efficacement des données numériques et de présenter les résultats sous forme de graphiques.

Avantages du tableur :

- ✓ Facilité et rapidité d'utilisation
- ✓ Exploitation et mise en forme rapide des résultats
- ✓ Fichiers autonomes et portables

Problèmes de l'approche fichiers :

- ✓ Pas de sémantique et vision limitée des données
- ✓ Redondance et risque d'incohérence des données
- ✓ Pas d'indépendance entre les données et les traitements (moyennes, graphiques...)
- ✓ ~~Pas d'accès multi-utilisateurs~~

Remarque : Tout ensemble de données ne justifie pas forcément d'une base de données.

Tableur Vs SGBD ? 2/2

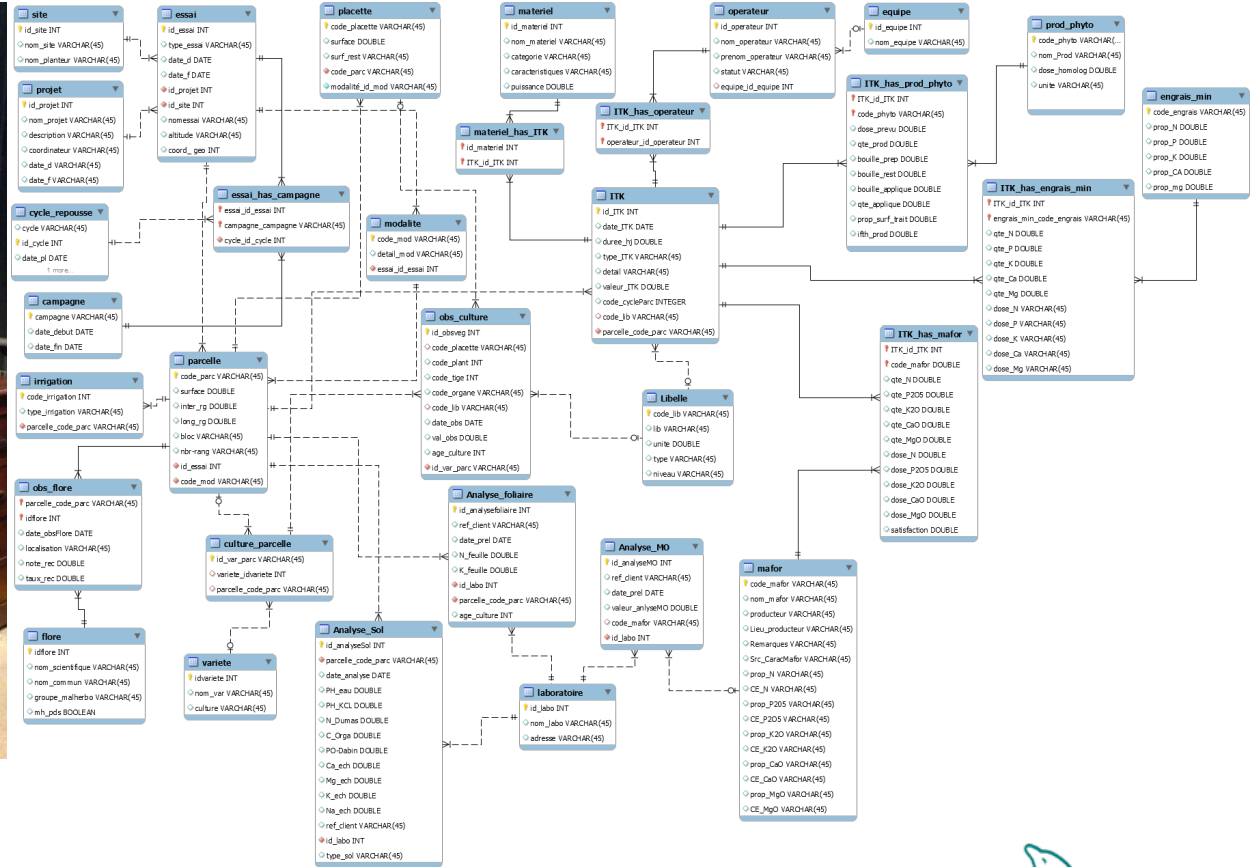
Intérêts d'une base de données :

- ✓ Manipulation facile de gros volumes de données
- ✓ Stockage persistant et interopérabilité entre les bases de données
- ✓ Unicité de l'information
- ✓ Langage normalisé de requêtes et de programmation
- ✓ Contexte multi-utilisateurs

Inconvénients :

- ✓ Nécessite de bien définir les besoins et d'être rigoureux lors de la réalisation de la base de données
- ✓ Nécessite un réel apprentissage

Les bases de données à travers le temps



Ancêtre de la BD



Modèle BD sous MySQL®



Données, base de données et SGBD

Modèle conceptuel

Modèle logique

Modèle physique

Importation/exportation de données

Requêtes

Logiciel R et bases de données

Utilisation du package RODBC



Sandrine Auzoux

UR AIDA

Jean-Christophe Soulié

UR Recyclage et Risque

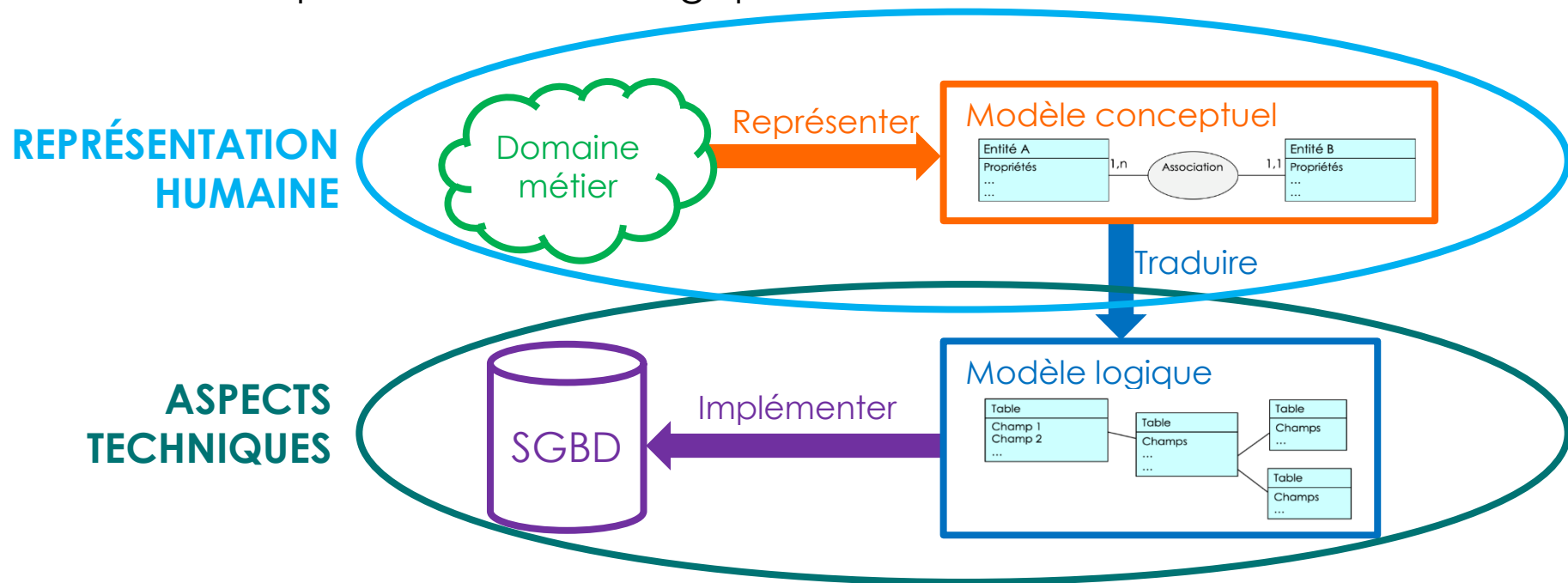
Conception de bases
de données
expérimentales à des
fins de modélisation

Interfaçage avec R



Etapes de conception d'une base de données

1. Analyse de l'existant et des besoins : **spécifications générales**
2. Création d'un **modèle conceptuel** (MCD) qui permet de représenter tous les aspects importants du problème posé → schéma conceptuel
3. Traduction du modèle conceptuel en **modèle logique** (MLD)
4. Implémentation du **modèle physique** (MPD) de la base de données dans un SGBD, à partir du modèle logique



Analyse de l'existant et des besoins

Phase **essentielle** et complexe

1. Ecrire les **objectifs de la base de données** : 1 phrase + mots clés qui répondent aux questions Qui ? Où ? Quand ? Quoi ? Pourquoi ? Comment ? Combien ?
2. **Rassembler tous les documents** que vous avez en votre possession (formulaires papier, fichiers Excel, Word et PowerPoint, photos, vidéo, croquis...)
3. **Collecter et organiser toutes les informations** que vous souhaitez stocker dans la base de données
4. Description en langage naturel des données et des traitements à effectuer sur ces données ⇔ **spécifications générales**



La conception réussie d'une base de données repose sur la précision et l'exhaustivité avec lesquelles les besoins des utilisateurs ont été exprimés.

Formalismes de modélisation et méthodes

2 formalismes très utilisés dans le domaine des bases de données :



Méthode d'Etudes et de Réalisation Informatique pour les Systèmes d'Entreprise (1978), basée sur le schéma Entités-Associations (E.F. Codd 1969)



Unified Modeling Language (Booch, Rumbaugh, Jacobson, 1996). C'est la référence en termes de modélisation objet, qui consiste à créer une représentation informatique des éléments du monde réel auxquels on s'intéresse, sans se préoccuper de l'implémentation.

La méthode MERISE et le modèle E-A

1. Définition des spécifications ➡ **Dictionnaire de données**

2. Modèle conceptuel de données (**MCD**)

*modélisation formelle et standardisée de la structure d'une base de données sans
Faire référence à une solution technique*

Modèle entité-association
(modèle conceptuel des données)

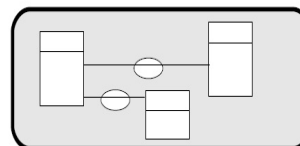


Schéma conceptuel

3. Modèle logique de données (**MLD**)

*modélisation de la structure suivant laquelle les
données seront stockées dans la future
base de données*

Modèle logique des données

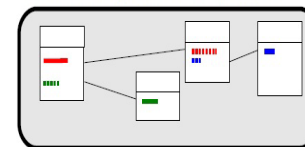


Schéma logique

Modèle physique des données

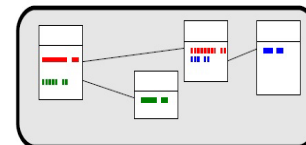


Schéma physique

4. Modèle physique de données (**MPD**)

représentation exacte de la BD en fonction du SGBD

Le dictionnaire des données

Le dictionnaire des données est un document qui regroupe toutes les données à conserver dans la base (MCD)

Code mnémorique	Désignation	Type	Taille	Remarque
Nom_agriculteur	Nom de l'agriculteur	A	30	
ID_placette	Identifiant de la placette	AN		Code de la placette où se situe la modalité
Surface	Surface de la parcelle	N		
Date_plantation	Date de plantation	date		Au format jj/mm/aaaa
Recolte	Récolté ou pas?	booléen		

Inventaire épuré : données élémentaires

- ✓ Pas de données calculées
- ✓ Pas de données composées
- ✓ Pas de synonymes (même propriété avec plusieurs noms)
- ✓ Pas de polysèmes (plusieurs propriétés de même nom)
- ✓ Pas de redondances (répétitions des mêmes propriétés)

Modèle conceptuel de données (MCD)

- Représentation formelle et standardisée de l'ensemble des données que l'on doit traiter pour répondre au problème posé
- **Avantage : être à la portée des utilisateurs.** Ceux-ci expriment en langage naturel leur connaissance des données et participent activement à l'élaboration du schéma conceptuel.
- **Indépendant de toute technique informatique** (physique et logique)
- **Basé sur le modèle Entité-Relation ou Entité-Association (EA)** choisi comme Standard en 1988.
 - concepts de base du modèle : **propriété, entité, relation**
 - règles de construction pour aboutir à une représentation graphique standard qui élimine les redondances et les ambiguïtés.



Ne pas négliger l'importance de la phase de modélisation conceptuelle

Définitions MCD – Entité (1/5)

Une **entité** est un ensemble d'éléments homogènes

Ex : exploitant, parcelle

Une entité possède un ensemble de **propriétés**

Ex : NoExploitant, nomExploitant, prenomExploitant

Une **occurrence** est une valeur prise par la propriété

Ex : Daniel

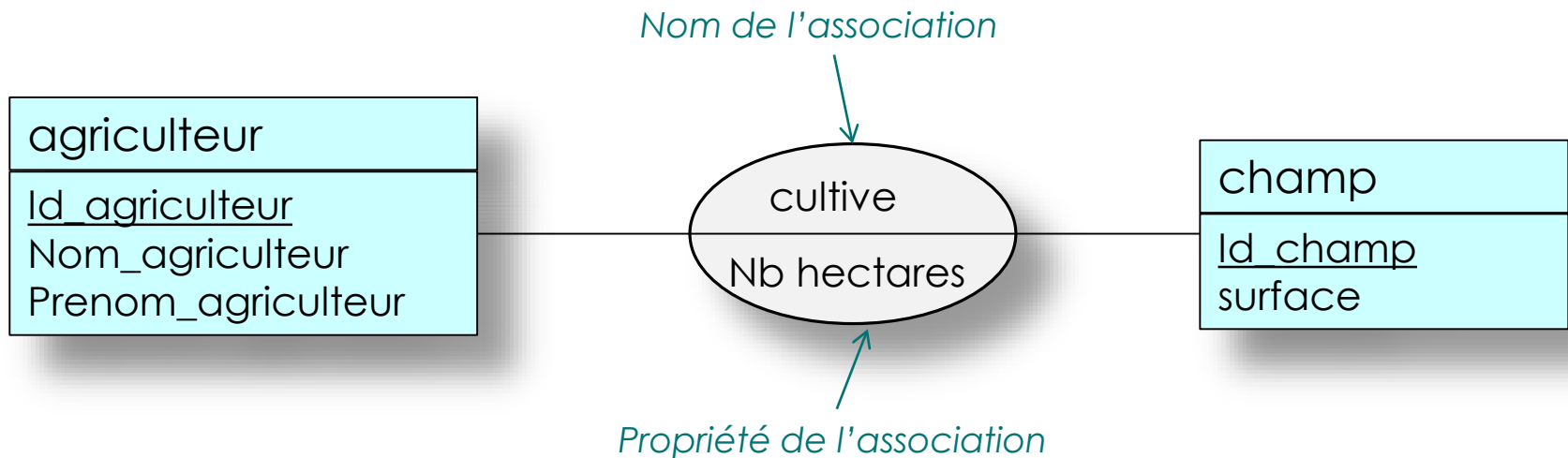
Un **identifiant** correspond à une/plusieurs propriétés d'une entité qui ont une valeur unique pour chaque occurrence

Ex : NoExploitant

Libellé entité
Liste des propriétés
•
•
•

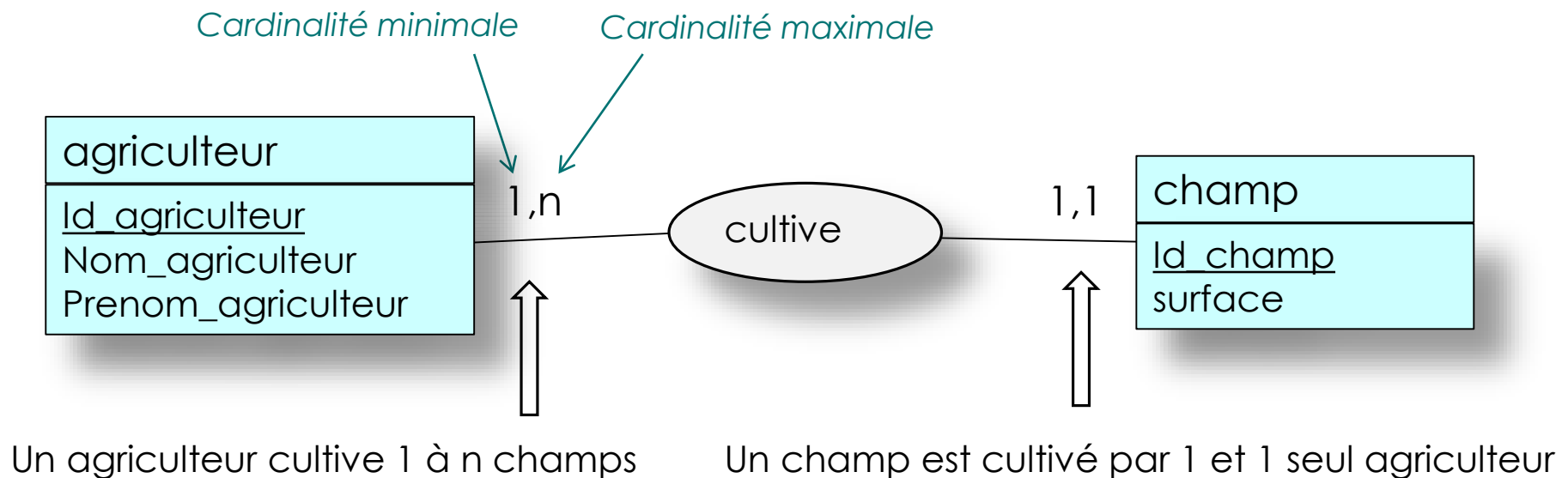
Définitions MCD – Association (2/5)

Une **association** (relation) est un lien sémantique entre plusieurs entités.



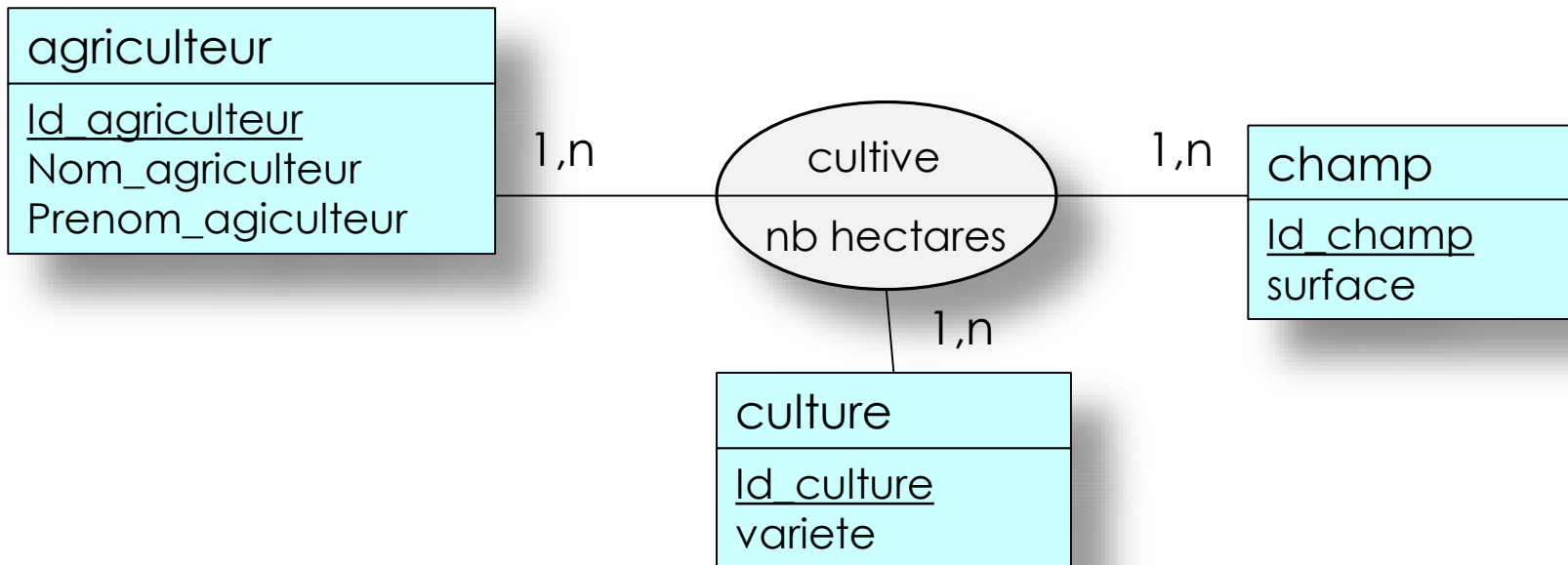
Définitions MCD – Cardinalité (3/5)

Les **cardinalités** représentent le nombre d'occurrences minimum et maximum d'une association par rapport à une entité.



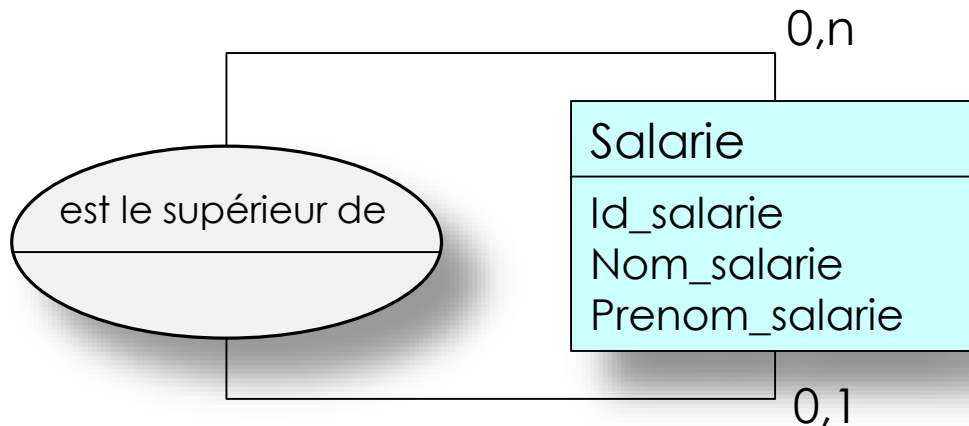
Définitions MCD – Dimension (4/5)

La **dimension** d'une association est le nombre d'entités concernées par celle-ci (nombre de pattes de l'association). Elle peut être binaire, ternaire ou n-aire.

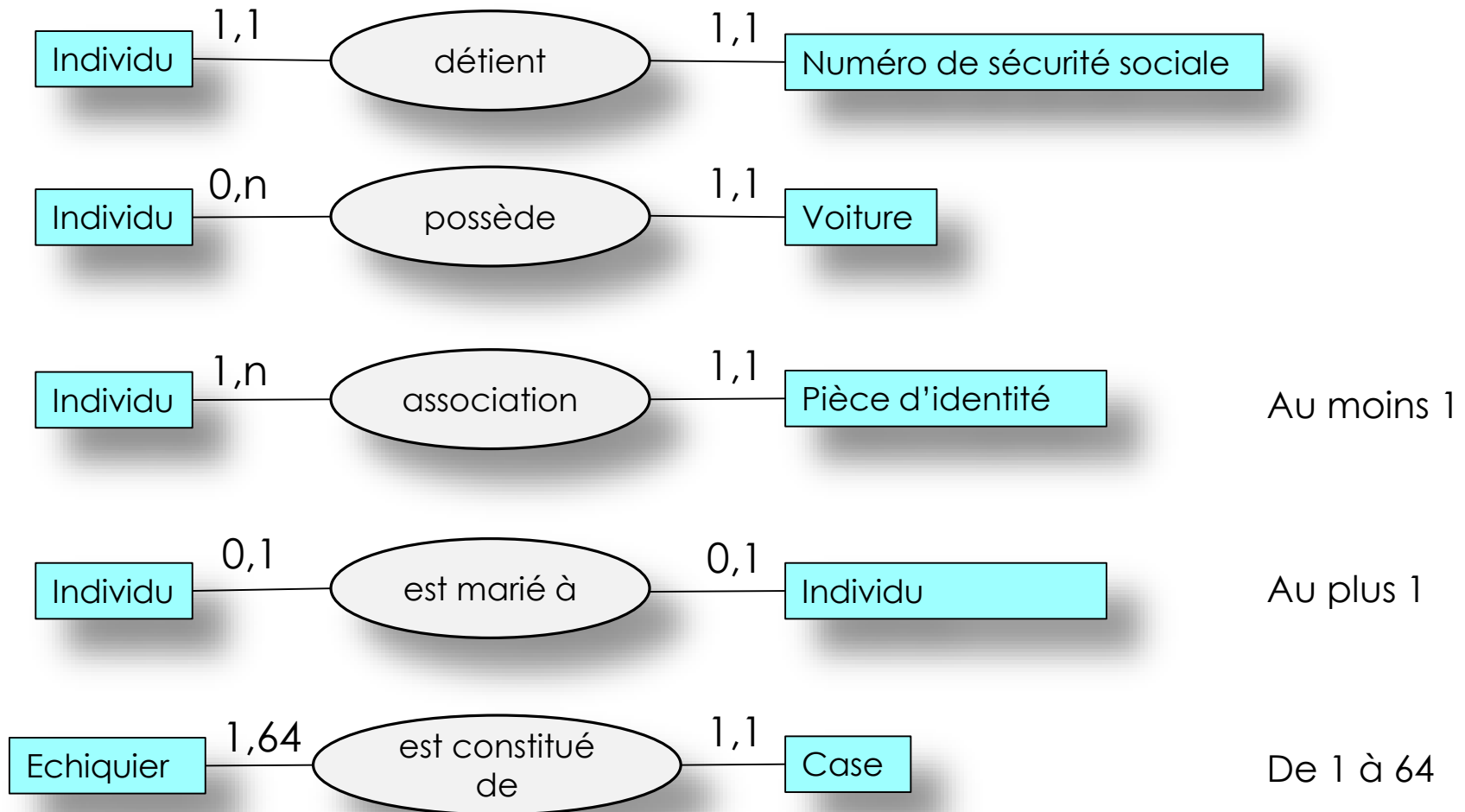


Définitions MCD – Réflexivité (5/5)

L'**association** réflexive est une association dont plusieurs « pattes » lient la même entité. Dans ce cas, plusieurs occurrences de la même entité seront associées.



MCD – Valeurs des cardinalités

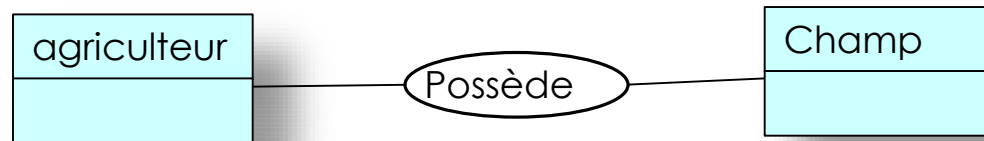


Construction d'un MCD

Solution 1: modélisation directe

Identification des entités et des associations à partir des noms et des verbes

*Exemple, un agriculteur possède un champ : 2 entités **agriculteur** et **champ**, et 1 association **possède***



Solution 2 : modélisation par analyse des dépendances fonctionnelles

- 1) Construction et épuration du *dictionnaire de données*
- 2) Création et simplification de la *matrice des dépendances fonctionnelles simples*
- 3) Recherche des *dépendances fonctionnelles composées*
- 4) Élaboration du modèle conceptuel de données

Exercice Kilouplus

Création du MCD

1. Création du dictionnaire de données
2. Construction de la matrice des dépendances fonctionnelles (tableau reprenant la liste des identifiants et des propriétés)
3. Réalisation de la matrice des dépendances fonctionnelles composées (même tableau que précédemment avec les propriétés non définies)
4. Ecriture des dépendances fonctionnelles
5. Modèle conceptuel de données

Inventaire exhaustif

Propriétés	Provenance	Remarque
Code Client	Liste clients	Constitué des 3 lettres nom + 3 lettres prénom + département + Ordre A, B,....
Civilité	Liste clients	
Prenom	Liste clients	
Nom	Liste clients	
Adresse	Liste clients	
Code Postal	Liste clients	
Ville	Liste clients	
Code	Communes + code post	
Code Postal	Communes + code post	
Ville	Communes + code post	
Code	Liste des agences	
Nom	Liste des agences	
Référence	Catalogue	
Désignation	Catalogue	
Tarif / Semaine	Catalogue	
Caution	Catalogue	
Code article	Stocks	
Nom	Stocks	
Quantité dispo / agence	Stocks	
Date	Contrat	
Nom de l'Agence	Contrat	
Numero contrat	Contrat	
Civilité	Contrat	
Prenom	Contrat	
Nom	Contrat	
Adresse	Contrat	
Code postal	Contrat	
Ville	Contrat	
Durée	Contrat	
Reference	Contrat	
Materiel	Contrat	
Montant Unitaire	Contrat	
Quantité	Contrat	
Montant total (ligne)	Contrat	
Total facture	Contrat	
Caution	Contrat	

Détection des synonymes, des calculs, des polysèmes

1. Renommer les propriétés
2. Supprimer les propriétés calculées

Propriétés	Synonymes, calculs, polysèmes
Code Client	Code Client
Civilité	Civilité
Prenom →	Prenom Client
Nom →	Nom Client
Adresse →	Adresse Client
Code Postal	Code Postal
Ville	Ville
Code →	Code CP
Code Postal	Code Postal
Ville	Ville
Code →	Code Agence
Nom →	Nom Agence
Référence →	Référence Matériel
Désignation	Désignation
Tarif / Semaine	Tarif / semaine
Cautiion	Cautiion
Code article →	Référence Matériel
Nom →	Désignation
Quantité dispo / agence	Quantité dispo / agence
Date →	Date Commande
Nom de l'Agence	Nom Agence
Numero contrat	Numero Contrat
Civilité	Civilité
Prenom →	Prenom Client
Nom →	Nom Client
Adresse →	Adresse Client
Code postal	Code Postal
Ville	Ville
Durée	Durée
Reference →	Référence Matériel
Matériel →	Désignation
Montant Unitaire →	Tarif / semaine
Quantité →	Quantité Louée
Montant total (ligne)	Calculé
Total facture	Calculé
Cautiion	Calculé

Détection des redondances

3. Supprimer les propriétés en double

Propriétés	Inventaire épuré
Code Client	Code Client
Civilité	Civilité
Prenom Client	Prenom Client
Nom Client	Nom Client
Adresse Client	Adresse Client
Code Postal	Code Postal
Ville	Ville
Code CP	Code CP
Code Postal	
Ville	
Code Agence	Code Agence
Nom Agence	Nom Agence
Référence Matériel	Référence Matériel
Désignation	Désignation
Tarif / semaine	Tarif / semaine
Caution	Caution
Référence Matériel	
Désignation	
Quantité dispo / agence	Quantité dispo / agence
Date Commande	Date Commande
Nom Agence	
Numero Contrat	Numero Contrat
Civilité	
Prenom Client	
Nom Client	
Adresse Client	
Code Postal	
Ville	
Durée	Durée
Référence Matériel	
Désignation	
Tarif / semaine	
Quantité Louée	Quantité Louée
Calculé	
Calculé	
Calculé	

La matrice des dépendances fonctionnelles

La MDF permet de mettre en évidence les liens sémantiques et logiques entre les données.

Il s'agit d'un tableau à 2 entrées : en lignes et en colonnes , on inscrit les données issues du dictionnaire de données.

Pour remplir ce tableau, on pose une question pour chaque colonne :

Pour une valeur de cette donnée, existe-t-il une seule valeur de la donnée située en ligne ?

Dans l'affirmative, on inscrit le chiffre 1 à l'intersection

Cible	Source	ID_Placette	IDagriculteur
ID_placette		*	
Surface		1	
IDagriculteur			*
Nomagriculteur			1

Matrice des dépendances fonctionnelles simples

Propriétés	Code client	Civilité	Prenom Client	Nom Client	Adresse Client	Code Postal	Ville	Code CP	Code Agence	Nom Agence	Reference Materiel	Designation	Tarif / semaine	Cautious	Quantite dispo / agence	Date commande	Numero Contrat	Duree	Quantite louee
Code client	*																		
Civilite	1 *																1		
Prenom Client	1		*														1		
Nom Client	1			*													1		
Adresse Client	1				*												1		
Code Postal	1					*		1									1		
Ville	1						*	1									1		
Code CP	1							*											
Code Agence									*										
Nom Agence									1 *								1		
Reference Materiel											*								
Designation											1 *								
Tarif / semaine											1		*						
Cautious											1			*					
Quantite dispo/agence															*				
Date commande																*	1		
Numero Contrat																	*		
Duree																	1 *		
Quantite louee																			*

Pour une valeur de chaque colonne, existe-t-il une seule valeur de la donnée située en ligne ?

Simplification de la MDF

Propriétés	Code client	Civilité	Prenom Client	Nom Client	Adresse Client	Code Postal	Ville	Code CP	Code Agence	Nom Agence	Reference Materiel	Designation	Tarif / semaine	Cauton	Quantite dispo / agence	Date commande	Numero Contrat	Duree	Quantite louee
Code client	*																		
Civilite	1 *																1		
Prenom Client	1		*														1		
Nom Client	1			*													1		
Adresse Client	1				*												1		
Code Postal	1					*		1									1		
Ville	1						*	1									1		
Code CP	1							*											
Code Agence									*										
Nom Agence									1 *								1		
Reference Materiel											*								
Designation											1 *								
Tarif / semaine											1		*						
Cauton											1			*					
Quantite dispo / agence															*				
Date commande																*	1		
Numero Contrat																	*		
Duree																	1 *		
Quantite louee																			*

Pour une valeur de chaque colonne, existe-t-il une seule valeur de la donnée située en ligne ?

Résultat de la simplification

Propriétés	Code client	Code CP	Code Agence	Reference materiel	Numero contrat
Civillite	1				1
Prenom client	1				1
Nom client	1				1
Adresse client	1				1
Code postal	1	1			1
Ville	1	1			1
Nom agence			1		1
Designation				1	
Tarif/semaine				1	
Cauton				1	
Quantite dispo/agence					
Date commande					1
Duree					1
Quantite louee					

Pour une valeur de chaque colonne, existe-t-il une seule valeur de la donnée située en ligne ?

Elimination des dépendances transitives

Propriétés	Code client	Code CP	Code Agence	Reference materiel	Numero contrat
Civilite	1				1
Prenom client	1				1
Nom client	1				1
Adresse client	1				1
Code postal	1	1			1
Ville	1	1			1
Nom agence			1		1
Designation				1	
Tarif/semaine				1	
Cauton				1	
Quantite dispo/agence					
Date commande					1
Duree					1
Quantite louee					

Plusieurs 1 sur la ligne = transitivité, choisir 1 et 1 seul identifiant par colonne

Mise en évidence et labellisation des entités

- La matrice des dépendances **fonctionnelles simple** met en évidence les entités.

	Client	CodePostal	Agence	Materiel	Contrat
Propriétés	Code client	Code CP	Code Agence	Reference materiel	Numero contrat
Civilité	1				
Prenom client	1				
Nom client	1				
Adresse client	1				
Code postal		1			
Ville		1			
Nom agence			1		
Designation				1	
Tarif/semaine				1	
Cauton				1	
Date commande					1
Duree					1

Matrice des dépendances fonctionnelles composées

Certaines données ne contiennent pas de 1 ni dans la ligne, ni dans la colonne

Propriétés	Code client	Civilité	Prenom Client	Nom Client	Adresse Client	Code Postal	Ville	Code CP	Code Agence	Nom Agence	Reference Materiel	Designation	Tarif / semaine	Caution	Quantite dispo / agence	Date commande	Numero Contrat	Duree	Quantite louee
Code client	*																		
Civilite	1 *																1		
Prenom Client	1		*														1		
Nom Client	1			*													1		
Adresse Client	1				*												1		
Code Postal	1					*		1									1		
Ville	1						*	1									1		
Code CP	1							*											
Code Agence									*										
Nom Agence									1 *								1		
Reference Materiel											*								
Designation											1 *								
Tarif / semaine											1			*					
Caution											1			*					
Quantite dispo/agence															*				
Date commande																*	1		
Numero Contrat																	*		
Duree																	1 *		
Quantite louee																			*

Il s'agit donc de dépendances fonctionnelles composées: ces données dépendent de 2 ou de plusieurs colonnes.

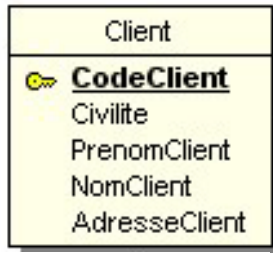
Matrice des dépendances fonctionnelles composées

- La matrice des dépendances fonctionnelles composées met en évidence les associations porteuses de propriétés dans le MCD.

Propriétés	Code client	Code CP	Code Agence	Reference materiel	Numero contrat
Quantite dispo/agence			1	1	
Quantite louee				1	1

Est-ce qu'une propriété correspond 1 et 1 seule fois à l'identifiant 1 et à l'identifiant 2?

MCD Kilouplus



Données, base de données et SGBD

Modèle conceptuel

Modèle logique

Modèle physique

Importation/exportation de données

Requêtes

Logiciel R et bases de données

Utilisation du package RODBC



Sandrine Auzoux

UR AIDA

Jean-Christophe Soulié

UR Recyclage et Risque

Conception de bases
de données
expérimentales à des
fins de modélisation

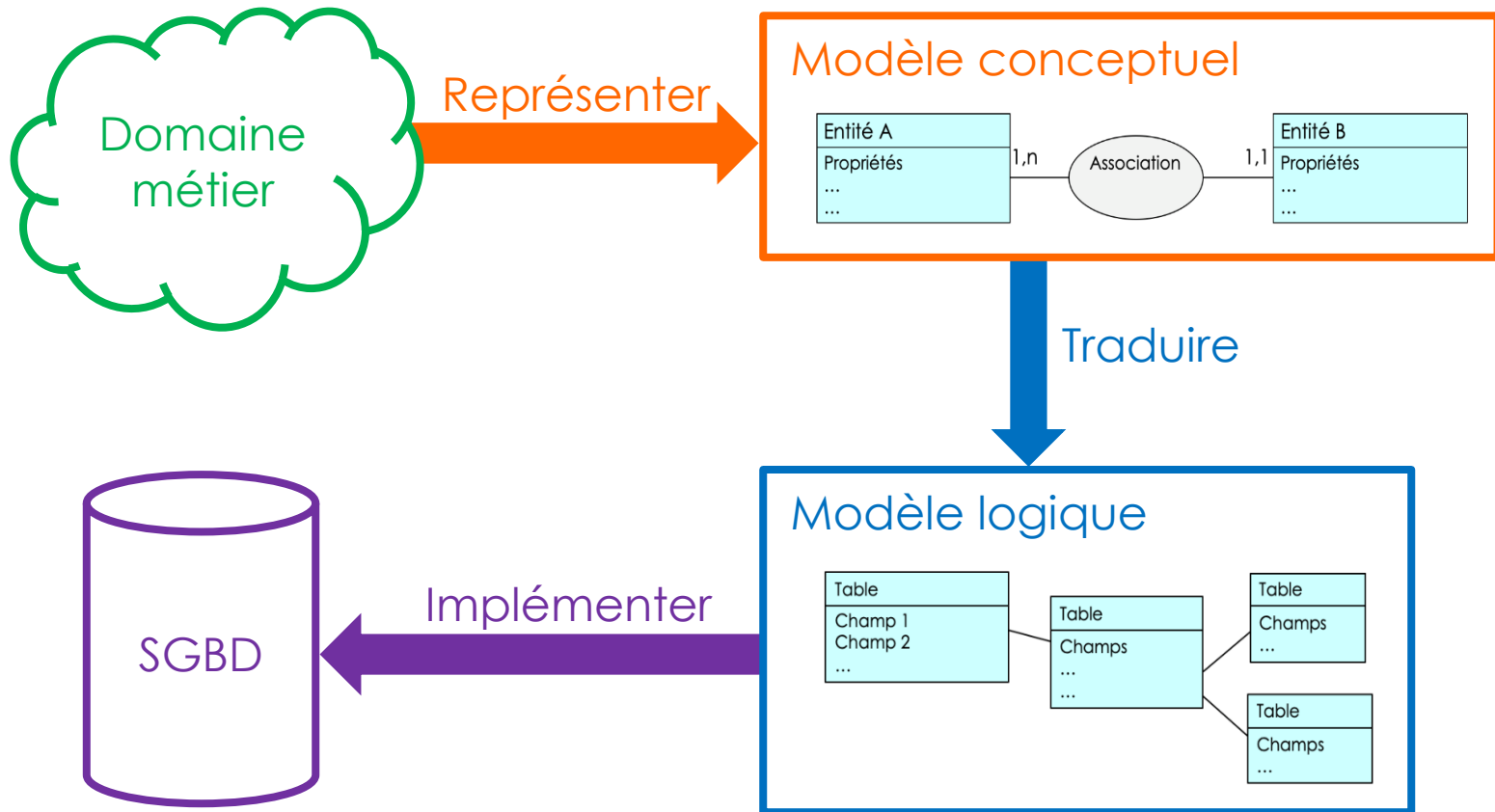
Interfaçage avec R



AGRICULTURAL RESEARCH
FOR DEVELOPMENT

Modèle Logique de Données

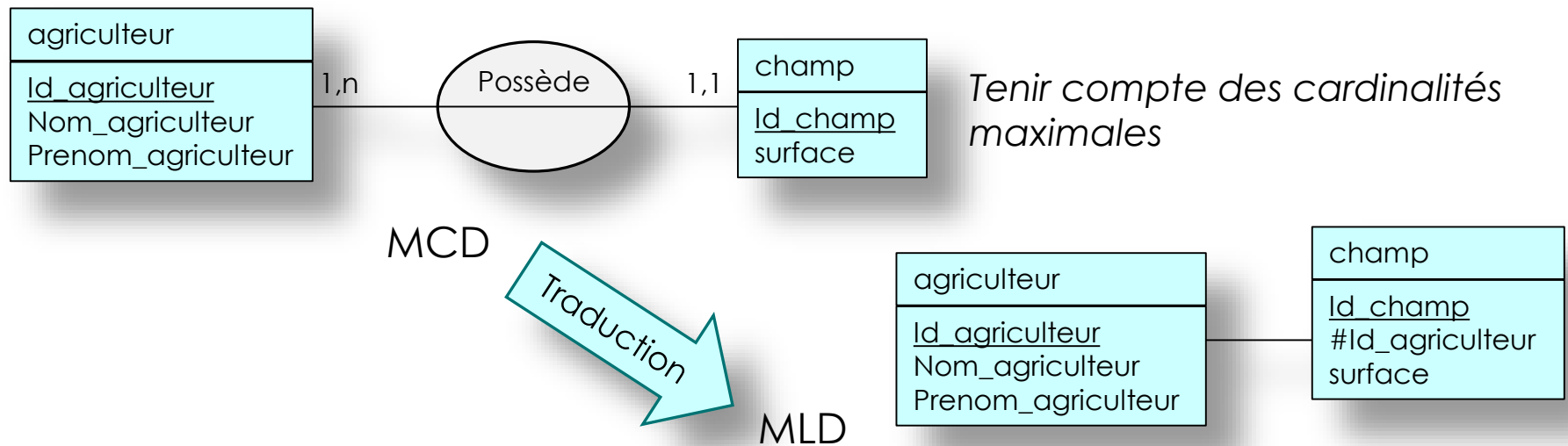
Basé sur le formalisme graphique Merise, le MLD est une étape intermédiaire pour passer du MCD, qui est un modèle sémantique, vers une représentation physique des données : SGBD.



Traduction d'un MCD en MLD (1/4)

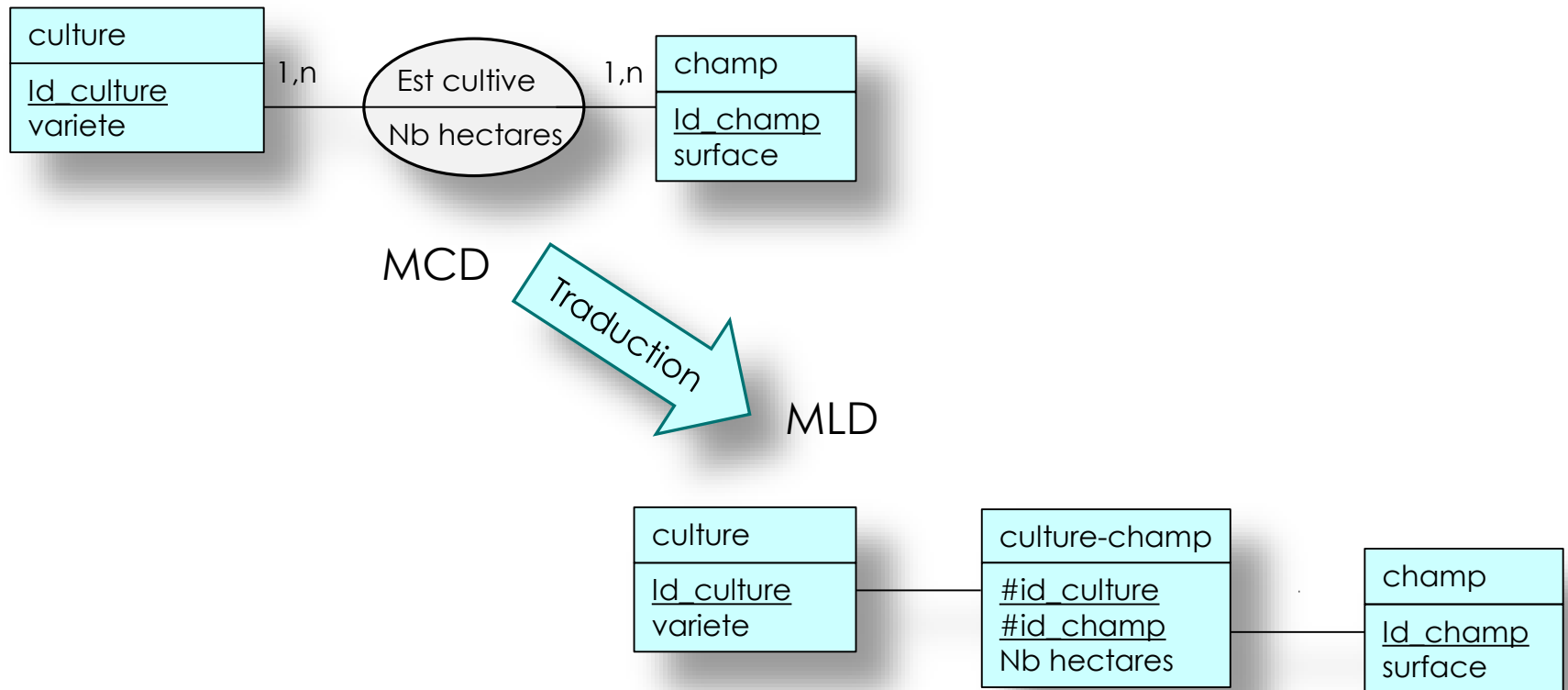
Règle 1 : toute entité devient une table dans laquelle les attributs deviennent des colonnes (champs). L'identifiant de l'entité devient la clé primaire de la table.

Règle 2 : toute association de type (1- n) disparaît. L'identifiant de la table côté n migre dans la table du côté 1 et devient une clé étrangère.



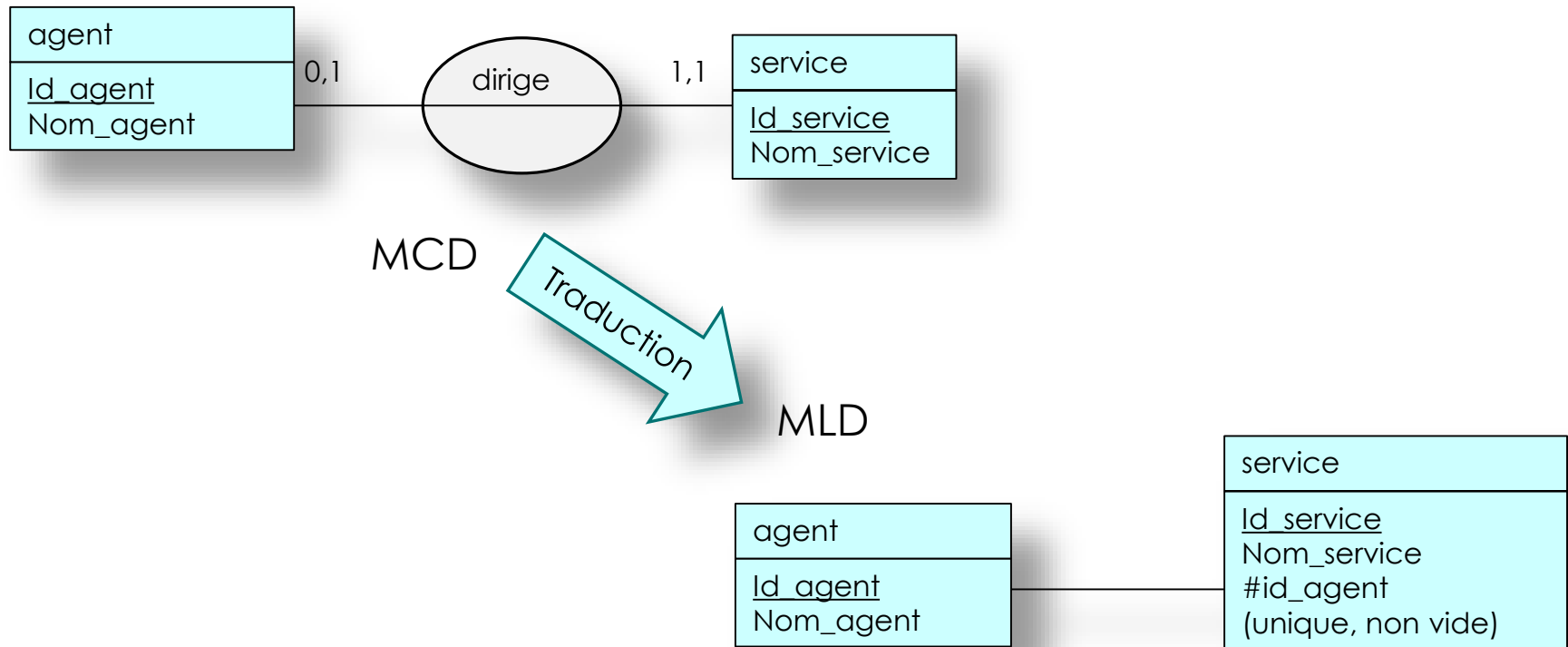
Traduction d'un MCD en MLD (2/4)

Règle 3 : toute association de type (n-n) devient une table dont la clé primaire est la concaténation des clés primaires des tables associées. Si l'association est porteuse, les propriétés deviennent des attributs de la table.



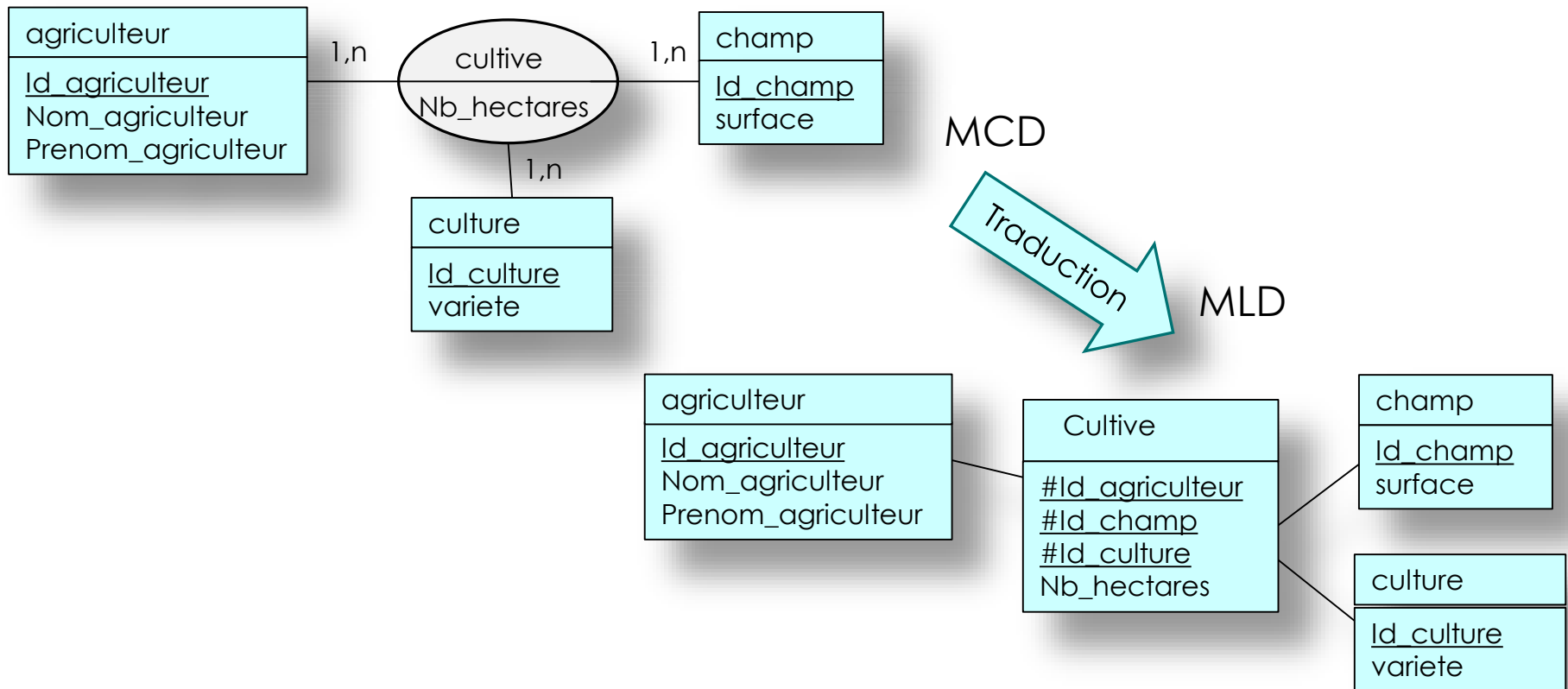
Traduction d'un MCD en MLD (3/4)

Règle 4 : toute association binaire de type (1-1) est traduite comme une association de type (n-n) à l'exception que la clé étrangère peut contenir ni doublon, ni valeur nulle.

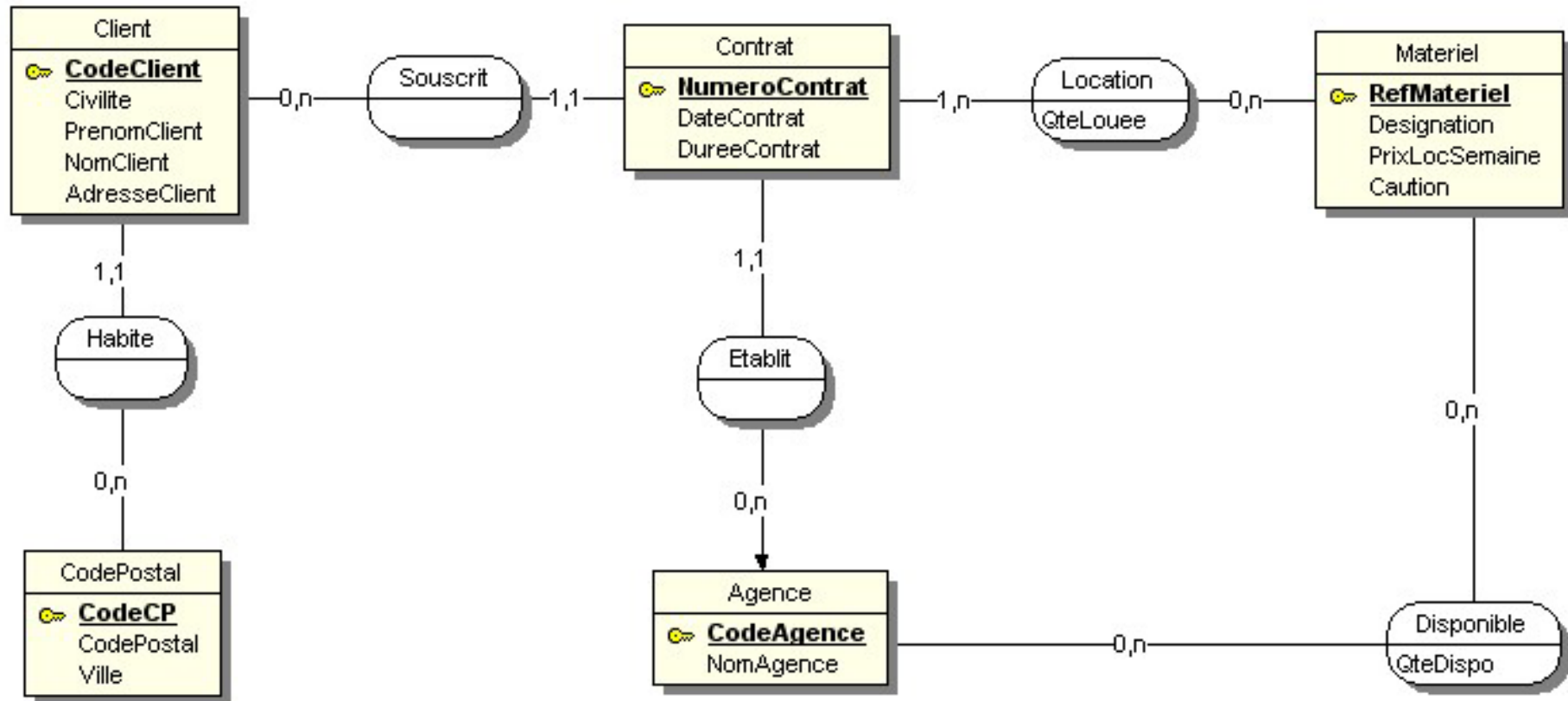


Traduction d'un MCD en MLD (4/4)

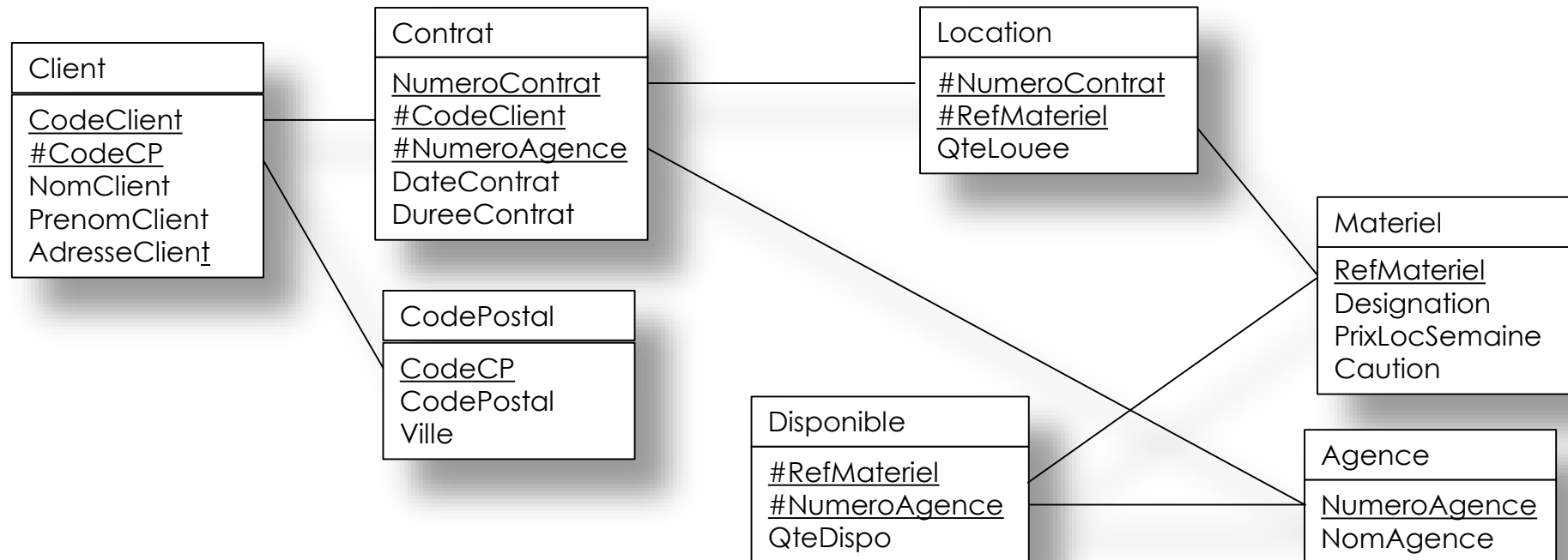
Règle 5 : Toute **association non binaire** est traduite par une **table supplémentaire** dont la clé primaire est composée d'autant de clés étrangères que d'entités en association. Les attributs de l'association deviennent des colonnes de la table.



MCD Kilouplus



MLD kilouplus



Modèle Physique de Données

Le MPD consiste à l'implémentation physique d'une base de données dans un SGBD.

Le MPD étant lié au SGBD, 2 méthodes possibles pour construire la structure de la base de données :

- ✓ Soit en utilisant les outils proposés par le SGBD
- ✓ Soit en utilisant le langage de définition des données du SQL (Structured Query Language).

Le MPD est basé sur la structure du MLD : **tables**, **champs**(attributs) et **relations** (associations). La traduction du MLD en MPD consiste à **typer les données** de chaque champ (chaîne de caractères, numérique...).

Le SGBDR Microsoft Access



C'est un **Système de Gestion de Bases de Données Relationnelles**

C'est un Programme de la suite bureautique  **Office** créée par **Microsoft**
(Licence propriétaire, environnement Windows)

La base de données est représentée par 1 fichier dont le nom se termine par l'extension ".mdb" (Access 2000 à 2003) ou « .accdb » (depuis Access 2007)

Une base de données Access est composée de 6 catégories d'objets :

- ✓ **Tables** : données stockées sous forme de tableaux
- ✓ **Requêtes** : filtres, tris, calculs sur 1 ou plusieurs tables/requêtes suppression, ajout, modification de données en masse
- ✓ **Formulaires** : masques de saisie basés sur des tables/requêtes
- ✓ **Etats** : aperçus avant impression personnalisés
- ✓ **Macros** : mini-programmes qui exécutent une tâche répétitive
- ✓ **Modules** : blocs de code Visual Basic for Application (VBA)

Les tables sous Access

Le **champ** (colonne) est l'élément fondamental

L'ensemble des champs constitue un **enregistrement** (ligne)

La **clé primaire** identifie chaque enregistrement de façon unique

(Access peut la créer lui-même et gérer automatiquement les données contenues)

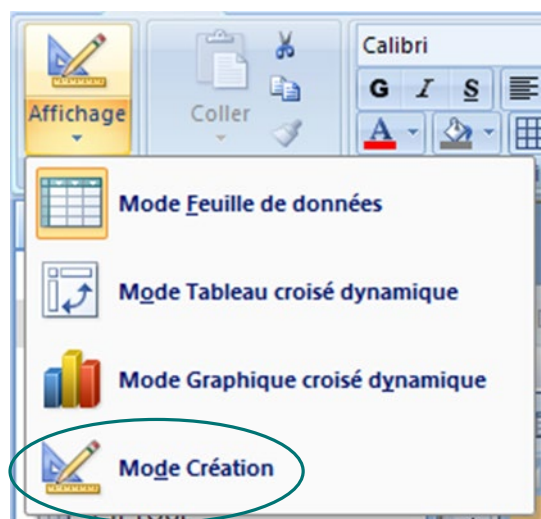
Clé primaire

champ

enregistrement

numdatamet ▼	numstatmet ▼	datemetec ▼	tmin ▼	tmax ▼	tmoy ▼	hrmin ▼	hrmax ▼	hrmoy ▼	ventmax ▼	ventmoy ▼
1	2	01/01/2013	17,27	30,65	23,58031	13,45	28,85	21,70	2,908	1,27
2	2	02/01/2013	17,49	31,94	24,3676	14,08	32,38	23,22	2,324	1,08
3	2	03/01/2013	17,58	32,29	24,57521	13,62	31,67	22,22	2,507	1,04
4	2	04/01/2013	17,9	32,94	24,92312	10,73	29,07	19,50	2,098	1,12
5	2	05/01/2013	17,58	31,77	24,58	10,34	27,58	17,90	2,538	1,13
6	2	06/01/2013	16,99	32,38	23,99187	10,29	27,07	18,45	2,811	0,83
7	2	07/01/2013	18,04	33	24,85854	13,6	30,62	22,70	2,012	0,94
8	2	08/01/2013	18,51	33,05	25,44167	15,86	33,13	23,91	1,943	0,94
9	2	09/01/2013	19,22	33,63	25,9976	15,77	28,93	22,73	2,18	1,11
10	2	10/01/2013	19,64	33,12	25,87104	10,69	26,67	18,57	2,069	1,13
11	2	11/01/2013	18,57	32,97	24,40073	11,96	36,56	22,27	2,16	0,78
12	2	12/01/2013	18,19	32,83	24,74562	12,11	28,56	20,57	2,153	1,09
13	2	13/01/2013	18,22	34,76	25,63521	9,45	23,61	16,75	1,992	1,03
14	2	14/01/2013	18,93	34,31	25,58865	11,02	31,8	18,90	1,868	0,73
15	2	15/01/2013	19,79	34,56	26,35302	10,67	32,76	20,09	2,031	0,59

Structure d'une table sous Access



Le mode *création* permet de définir la structure de la table :

- ✓ Nom des champs
- ✓ Type de données
- ✓ Description de la donnée contenue dans le champs (métadonnée)

Nom du champ	Type de données	Description
numdatamet	NuméroAuto	
numstatmet	Numérique	
datemeteo	Date/Heure	
tmin	Numérique	température minimale de l'air °C
tmax	Numérique	température maximale de l'air en °C
tmoy	Numérique	température de l'air moyenne en °C
hrmin	Numérique	humidité relative minimale de l'air en %
hrmax	Numérique	humidité relative maximale de l'air en %
hrmoy	Numérique	humidité relative moyenne de l'air en %
ventmax	Numérique	vitesse du vent maximale en m/s
ventmoy	Numérique	vitesse du vent moyenne en m/s
rayonglobal	Numérique	rayonnement global en MJ/m ²
eto	Numérique	évapotranspiration mm
pluie	Numérique	pluviométrie en mm (-9999,9 signifie que l'appareil était en panne)
pluiecumule	Texte	pluie cumulée en mm
patm	Numérique	pression atmosphérique en hPa

Liste des
champs

Les types de données sous Access

Type	Description	Taille
Texte Court	Caractères alphanumériques	<= 255 caractères
Texte Long	Caractères alphanumériques	<= 65535 caractères
Numérique	Nombres avec ou sans décimales	Entier : -32 768 à 32 767 Entier long : - 2 147 438 648 à + 2 147 438 648 Réel simple : -3,402823.10 ³⁸ à +3,402823.10 ³⁸ Réel double :-1,79769313486232.10 ³⁰⁸ à + 1,79769313486232.10 ³⁰⁸
Date/Heure	Date et/ou heure	
Monétaire	Evite l'arrondissement pendant les calculs Précision : 15 chiffres + 4 chiffres	-922 337 203 685 477,5808 à + 922 337 203 685 477,5808
Numéro auto	Nombre entier incrémentée automatiquement	Comme entier long
Liste de choix	Permet de choisir une valeur dans une autre table ou liste de valeurs	
Oui/Non	Combinaison logique (vrai/faux, 0 ou 1)	
Objet Ole	Objets provenant d'autres applications (Image, son, vidéo...)	<= 1Go
Lien hypertexte	Adresse internet, mail	<= 65535 caractères

Les propriétés des champs

Taille du champ : longueur pour les textes, longueur et type pour les nombres

Format : comme Excel

Masque de saisie : *date, tel, code postal...*

Valeur par défaut : remplit automatiquement le champ

Valide si : indique des conditions (ex: *note >0 et <20*)

Null interdit : peut-on ne rien mettre ?

Chaine vide autorisée : texte vide

Indexé : définit les clés et accélère le processus de recherche

Liste de choix : évite de taper l'intégralité du texte lorsque les choix sont limités

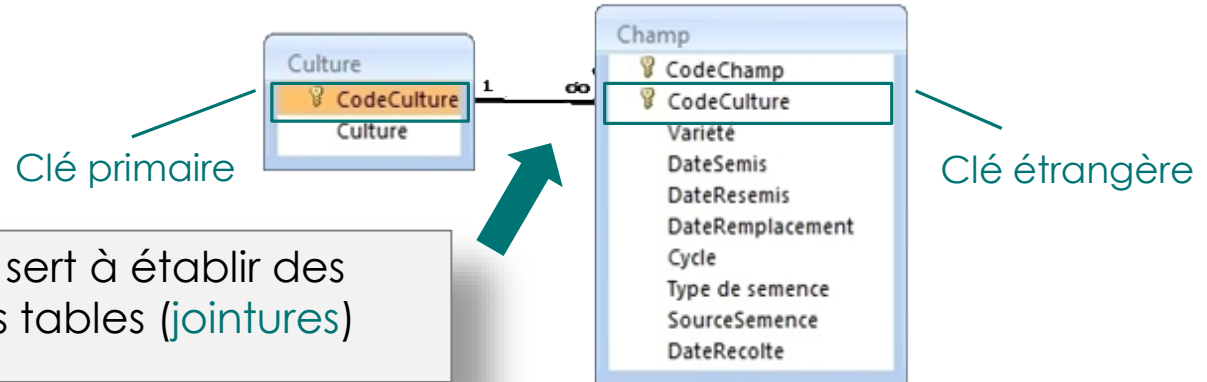
Général	Liste de choix
Taille du champ	Réel double
Format	Standard
Décimales	2
Masque de saisie	
Légende	
Valeur par défaut	
Valide si	
Message si erreur	
Null interdit	Non
Indexé	Non
Balises actives	
Aligner le texte	Général



Pour de meilleurs résultats, attribuez toujours la *valeur la plus petite possible* à la propriété *Taille du champ*.

Les clés étrangères - relations

Une **clé étrangère** est une clé qui représente la **valeur de la clé primaire** d'une table connexe.



codeculture	culture
Ab	Arbuste
An	Anacardier
Ar	Arachide
Au	Autres
Ba	Banancier
Cc	Crin-crin
Co	Coton
Eu	Eucalyptus
Go	Gombo
He	Herbe
Ic	Imperata cylind
Ig	Igname
Ma	Manguier
Mn	Manioc
Ms	Maïs
Ni	Niébé
Pa	Palmier

	codechamp	codeculture	variété	datesemis	cycle	sourcesemer	daterecolte
	MsN3	Ms		03/06/2013	3	achat au march	23/08/2013
	MsRilgN1	Ms		23/05/2013	3	produit par lui-r	01/09/2013
	MsSoN1	Sp		25/06/2013	6	produit par lui-r	30/11/2013
	NiSE2	Ni		15/04/2014	2	produit par lui-r	15/06/2014
	RiB72	Ri	BL19	01/07/2013	3	achat au magas	15/11/2013
*							

table Champ

Contrairement aux **clés primaires** qui sont **uniques**, les **clés étrangères** peuvent prendre **plusieurs fois la même valeur**.

table culture

L'intégrité référentielle

Il s'agit de règles simples suivies par le SGBDR pour maintenir la cohérence des relations entre les tables et garantir l'intégrité des données stockées.

Pour que la relation soit toujours valide, toutes les valeurs de la clé étrangère de la table liée doivent exister dans le champ clé primaire de la table de référence.

Sous Access, 2 actions étendent ce principe :

- ✓ « mettre à jour en cascade » : lorsque l'utilisateur change la valeur de la clé primaire, automatiquement, la valeur du champ correspondant de la table liée est corrigée.
- ✓ « suppression en cascade » : lorsque l'utilisateur efface un enregistrement, automatiquement, tous les enregistrements des tables liées qui ont un champ de même valeur sont supprimés.

Création du schéma relationnel (MPD)

Diagram illustrating the creation of a relational schema (MPD) using a software interface.

Top Menu Bar: Fichier, Accueil, Créer, Données externes, Outils de base de données, Aide, Dites-nous.

Top Toolbar: Compacter et réparer une base de données (Outils), Visual Basic (Macro), Exécuter une macro (Macro), Relations (Relations), Dépendances d'objet (Relations), Documentation de base de données (Analyser), Analyse des performances (Analyser), Analyser la table (Analyser).

Diagram (MPD): Shows tables and their relationships.

- Civilite:** CodeCivilite (PK), Civilite.
- CodePostal:** CODECP (PK), CodePostal, ville, Commentaire.
- Client:** CodeClient (PK), Civilite, NomClient, PrenomClient, Adresse, CODECP.
- Contrat:** NumContrat (PK), DateContrat, DureeContrat, CodeClient.
- ListeMatContrat:** NumContrat (PK), RefMateriel (PK), Qtelouee.
- Materiel:** RefMateriel (PK), Designation, PrixLocSemaine, Cautious.

Relationships:

- Civilite (1) to Client (∞).
- CodePostal (1) to Client (∞).
- Client (1) to Contrat (1).
- Client (1) to ListeMatContrat (∞).
- ListeMatContrat (∞) to Materiel (1).

Modifier des relations dialog box:

- Table/Requête : CODEPOSTAL
- Table/Requête liée : CLIENT
- CodeCP (PK) to CodeCP (FK)
- ☒ Appliquer l'intégrité référentielle
- ☒ Mettre à jour en cascade les champs correspondants
- ☐ Effacer en cascade les enregistrements correspondants
- Type de relation : Un-à-plusieurs
- Buttons: OK, Annuler, Type jointure..., Nouvelle relation...

Exercice Kilouplus suite

Création du MPD sous ACCESS :

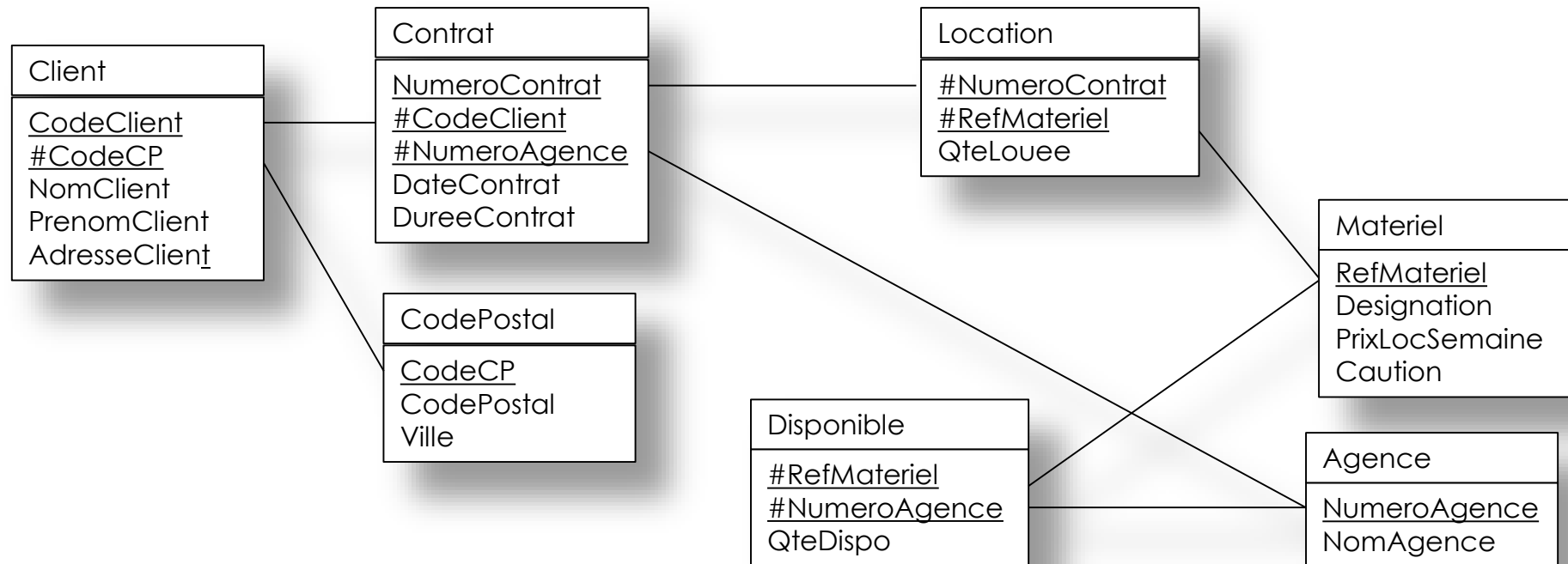
1. **Créer les tables**

- ✓ Définir les champs (type et taille)
- ✓ Définir les restrictions
- ✓ Identifier les clés primaires

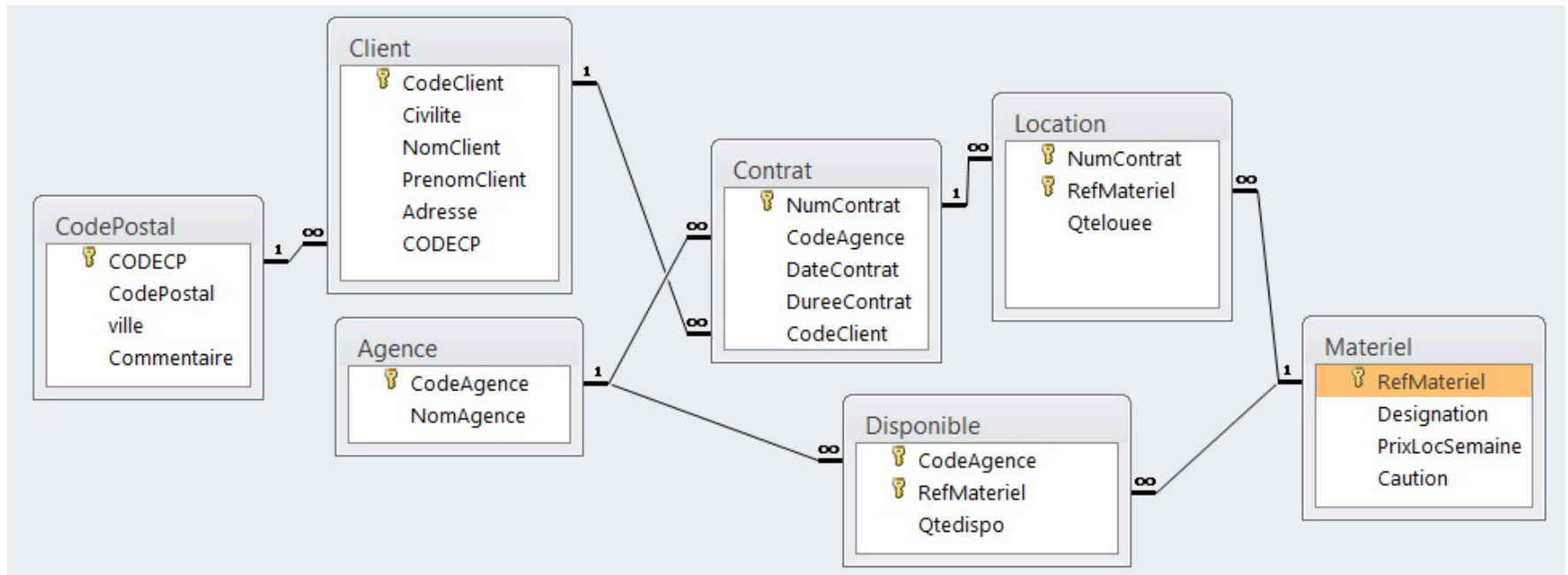
2. **Construire le schéma relationnel**

- ✓ Identifier les contraintes d'intégrité référentielles
- ✓ Construire les relations entre les tables

MLD kilouplus



MPD Kilouplus sous Microsoft Access



Données, base de données et SGBD

Modèle conceptuel

Modèle logique

Modèle physique

Importation/exportation de données

Requêtes

Logiciel R et bases de données

Utilisation du package RODBC



Sandrine Auzoux

UR AIDA

Jean-Christophe Soulié

UR Recyclage et Risque

Conception de bases
de données
expérimentales à des
fins de modélisation

Interfaçage avec R



AGRICULTURAL RESEARCH
FOR DEVELOPMENT

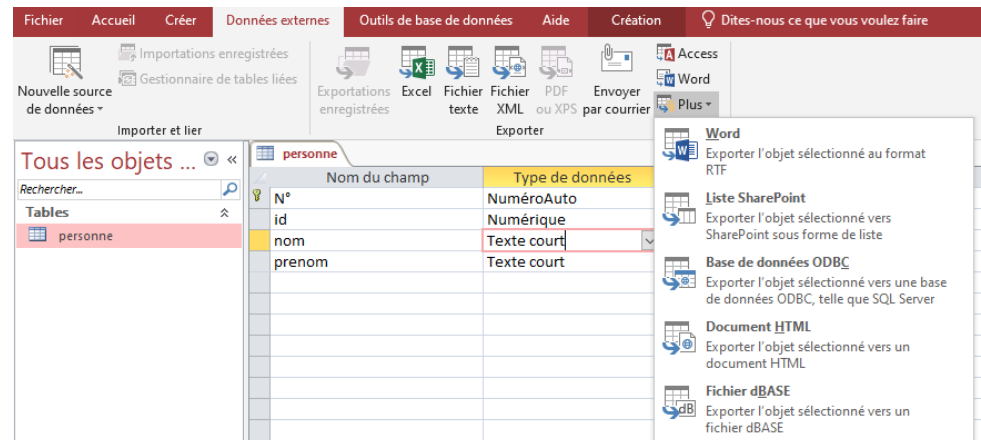
Alimentation d'une base de données Access

Access propose plusieurs méthodes pour entrer des données dans une base de données :

- ✓ Saisie directement dans une table : risque d'erreurs et fastidieux
- ✓ Saisie à l'aide d'un formulaire : plus précise, plus rapide et plus facile
- ✓ Importation de données dans Access : plus rapide et simple

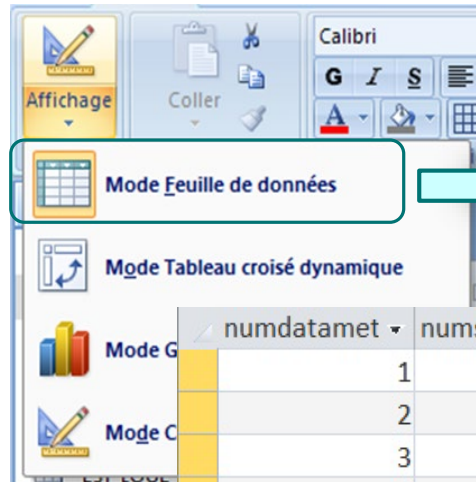
L'une des fonctionnalités les plus utiles dans Access est la possibilité d'utiliser des données provenant d'autres programmes.

Les commandes sous l'onglet "données externes" permettent d'exporter, d'importer et de lier des données.



Saisie des données dans une table

Le mode "feuille de données" permet d'afficher et de saisir les données dans une table. La saisie s'effectue comme dans un tableau Excel. Toute donnée saisie est enregistrée automatiquement.



champ

enregistrement

numdatamet	numstatmet	datemetec	tmin	tmax	tmoy	hrmin	hrmax	hrmoy	ventmax	ventmoy
1	2	01/01/2013	17,27	30,65	23,58031	13,45	28,85	21,70	2,908	1,27
2	2	02/01/2013	17,49	31,94	24,3676	14,08	32,38	23,22	2,324	1,08
3	2	03/01/2013	17,58	32,29	24,57521	13,62	31,67	22,22	2,507	1,04
4	2	04/01/2013	17,9	32,94	24,92312	10,73	29,07	19,50	2,098	1,12
5	2	05/01/2013	17,58	31,77	24,58	10,34	27,58	17,90	2,538	1,13
6	2	06/01/2013	16,99	32,38	23,99187	10,29	27,07	18,45	2,811	0,83
7	2	07/01/2013	18,04	33	24,85854	13,6	30,62	22,70	2,012	0,94
8	2	08/01/2013	18,51	33,05	25,44167	15,86	33,13	23,91	1,943	0,94
9	2	09/01/2013	19,22	33,63	25,9976	15,77	28,93	22,73	2,18	1,11
10	2	10/01/2013	19,64	33,12	25,87104	10,69	26,67	18,57	2,069	1,13
11	2	11/01/2013	18,57	32,97	24,40073	11,96	36,56	22,27	2,16	0,78
12	2	12/01/2013	18,19	32,83	24,74562	12,11	28,56	20,57	2,153	1,09
13	2	13/01/2013	18,22	34,76	25,63521	9,45	23,61	16,75	1,992	1,03
14	2	14/01/2013	18,93	34,31	25,58865	11,02	31,8	18,90	1,868	0,73
15	2	15/01/2013	19,79	34,56	26,35302	10,67	32,76	20,09	2,031	0,59

Saisie à partir d'un formulaire

Le formulaire est utilisé pour créer l'interface utilisateur d'une application de base de données.

Le formulaire "lié" est directement connecté à une source de données : une (des) table(s) ou une (des) requête(s).

Il facilite la saisie des données : ajout, modification, suppression.

+ agréable à utiliser que la feuille de données (affichage en ligne et en colonne)

The screenshot shows a web-based questionnaire form titled 'Questionnaire'. It contains several sections for data entry:

- Header Section:** Includes fields for 'N° questionnaire:' (1), 'Date de l'enquête:' (13/10/2013), 'Nom de l'enquêteur:' (Awil Ibrahim et Sama Djibril), 'Nom du paysan:' (Ambarka), 'Prenom Paysan:' (Adiza), 'Age du paysan:' (52), 'Genre:' (F), and 'Code du champ:' (RiS7). A 'Valider' button is present with a red warning message 'avant la saisie de la culture'.
- Navigation Tabs:** 'Culture', 'Intervention', and 'Farmers' Knowledge'. 'Culture' is the active tab.
- Culture Section:** Contains fields for 'Culture:' (Riz), 'Motif choix culture:' (adaptation), 'Stade culture à la date de l'enquête:', 'Variété (nom):', 'Cycle (nb mois):' (3), 'Type de semence:' (locale non traitée), 'Source de semence:' (achat au marché), 'Date de semis:' (12/07/2013), 'Date de resemis:', 'Date de remplacement:', and 'Date de récolte (envisagée):' (13/11/2013).
- Antécédents cultureaux Section:** A table-like structure showing historical crops over 4 years.

1 an	2 ans	3 ans	4 ans
riz	gombo	riz	gombo

Below this is a field for 'Cultures prévues dans le futur:' with the value 'gombo'.
- Production and Storage Section:** Includes fields for 'Production estimée (nb sacs):', 'poids du sac (kg):', 'Stockage après récolte:' (checked), 'Structure de stockage:' (chambre), 'Capacité de stockage (kg):', 'Durée de stockage (nb mois):' (2), 'Prix de vente au marché (cfa/unité):' (500), 'unité (kg, mesure):' (mesure), 'Marchés d'écoulement:' (Bougou), 'Moyen de transport de la production:' (camion), and 'Commentaire:'.
- Distance Section:** Fields for 'Distance max (km):' with sub-fields 'du champ au stockage:' (1) and 'du stockage au marché:' (5).
- Footer:** Navigation controls including 'Enr: 1 sur 1', 'Aucun filtre', and a 'Rechercher' button.

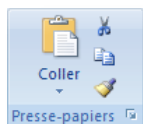
Données qu'Access peut importer, lier ou exporter

Format	Importation autorisée?	Liaison autorisée?	Exportation autorisée?
Fichier Excel	Oui	Oui	Oui
Fichier Access	Oui	Oui	Oui
Fichier texte	Oui	Oui	Oui
Fichier Word	Non, enregistrer fichier texte et importer	Non, enregistrer fichier texte et lier	Oui (exporter en tant que fusion word ou texte enrichi)
Fichier XML	Oui	Non	Oui
Fichier PDF	Non	Non	Oui
Document HTML	Oui	Oui	Oui
Fichier DBase	Oui	Oui	Oui
BD ODBC (SQL server par exemple)	Oui	Oui	Oui

Importer des données Excel vers Access

Plusieurs façons de stocker des données d'un classeur Excel dans une base de données Access :

1. Copier/Coller d'1 feuille de calcul ouverte dans 1 feuille de données Access.



2. Importer des données en écrivant du code.



3. Avec l'assistant d'importation d'Access, on peut importer 1 feuille de calcul dans 1 nouvelle table ou 1 table existante.

4. Lier une feuille de calcul à une base de données Access.

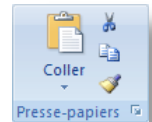


Il est impossible d'enregistrer un classeur Excel en tant que base de données Access. Excel n'intègre aucune fonctionnalité pour créer une base de données Access.

Exporter des données Access vers Excel

Plusieurs techniques pour transférer des données d'Access vers Excel:

1. Copier/Coller des données d'Access dans une feuille de calcul Excel.



2. Avec l'Assistant Exportation d'Access,

- ✓ On peut exporter une table, le résultat d'une requête, un formulaire ou un état vers une feuille de calcul Excel.
- ✓ 1 objet peut être exporté à la fois
- ✓ Possibilité de fusionner les données dans plusieurs feuilles de calcul Excel

3. Exporter des données à l'aide de code.



4. Lier une feuille de calcul à une base de données Access.



Données, base de données et SGBD
Modèle conceptuel
Modèle logique
Modèle physique
Importation/exportation de données
Requêtes

Logiciel R et bases de données
Utilisation du package RODBC



Sandrine Auzoux

UR AIDA

Jean-Christophe Soulié

UR Recyclage et Risque

Conception de bases
de données
expérimentales à des
fins de modélisation

Interfaçage avec R



SQL Généralités

SQL signifie "**Structured Query Language**"
c'est-à-dire "langage d'interrogation structuré"

C'est le **langage de requête standard** des SGBDR.

C'est à la fois :

- ✓ Un langage de définition de données (**LDD**) : création, modification, suppression des bases de données et des tables
- ✓ Un langage d'interrogation de la base (**LID**) : interrogation de la base en vue d'avoir une liste de résultats
- ✓ Un langage de manipulation de données (**LMD**) : sélection, insertion, modification ou suppression des données dans une table
- ✓ Un langage de contrôle de l'accès aux données (**LCD**) : définition des permissions au niveau des utilisateurs d'une base de données

SQL - Les commandes de base

Manipulation des tables

CREATE : création de table

DROP : suppression de table

ALTER : modification de la structure de table

Manipulation des données

INSERT : insertion de données dans 1 table

DELETE : suppression de données d'1 table

UPDATE : mise à jour de données d'1 table

SELECT : interrogation de données de 1 à n tables

Gestion des privilèges utilisateurs (droits d'accès)

GRANT : ajout de privilèges

REVOKE : suppression de privilèges

La commande Select

SELECT est la **commande la plus courante de SQL**. Elle consiste à lire les données contenues dans une base de données ou à calculer de nouvelles données à partir de données existantes.

SELECT [DISTINCT ou ALL]	Précise les colonnes qui vont apparaître dans la réponse
FROM	Précise la (ou les) table(s) intervenant dans l'interrogation
WHERE	Précise les conditions à appliquer sur les lignes. On peut trouver : <ul style="list-style-type: none">• Des comparateurs : =, >, <, >=, <=, <>• Des opérateurs logiques : AND, OR, NOT• Les prédicats : IN, LIKE, NULL, ALL, SOME, ANY, EXISTS...
GROUP BY	Précise la (ou les) colonne de regroupement
HAVING	Précise la (ou les) conditions associées à un regroupement
ORDER BY	Précise l'ordre dans lequel vont apparaître les lignes de la réponse : <ul style="list-style-type: none">- ASC : En ordre ascendant (par défaut)- DESC: En ordre descendant

Exemple de commande SELECT

Je sélectionne tous les agriculteurs qui cultivent du blé.



SELECT * **FROM** parcelle, agriculteur **WHERE** culture='blé'

parcelle		
parc_id	culture	surface
3106	blé	25500
2455	verger	4680



culture = blé



agriculteur
Nomagriculteur
John
Céline
Robert



parc_id	culture	surface	nom
3106	blé	25500	John
3106	blé	25500	Robert
3106	blé	25500	Céline

résultat de la requête

SQL – Les fonctions agrégats

Les fonctions **agrégats** sont utilisées dans une clause **SELECT** ou **HAVING** :

COUNT(<nom d'une colonne>) : comptabilise le nombre de lignes de la colonne spécifiée.

SUM(<nom d'une colonne>) : retourne la somme des valeurs d'une colonne spécifiée de type numérique.

MIN(<nom d'une colonne>) : retourne la valeur minimale d'une colonne spécifiée de type caractère ou numérique.

MAX(<nom d'une colonne>) : retourne la valeur maximale d'une colonne spécifiée de type caractère ou numérique caractère ou numérique.

AVG(<nom d'une colonne>) : calcule la moyenne arithmétique d'une colonne spécifiée de type numérique.

Exemple fonctions agrégats

Je sélectionne toutes les cultures dont la surface totale est supérieure à 30000



SELECT culture, **sum**(surface) **as** stot **FROM** parcelles **GROUP BY** culture
HAVING **sum**(surface) > 30000

parcelles		
parc_id	culture	surface
3106	blé	25500
3108	blé	54300
3004	maïs	42200
2311	vigne	8740
3016	tournesol	21500
3156	blé	26400
2308	vigne	12350
2455	verger	4680



résultat de la requête

culture	stot
blé	106200
maïs	42200

SQL – Les jointures

« Les jointures en SQL permettent *d'associer plusieurs tables* dans une même requête. Cela permet d'exploiter la puissance des bases de données relationnelles pour obtenir des résultats qui combinent les données de plusieurs tables de manière efficace. »

2 types de jointures peuvent être utilisés dans des requêtes en code SQL :

- ✓ Les *jointures internes* permettent de sélectionner les enregistrements ayant des *correspondances* entre les deux tables liées.
- ✓ Les *jointures externes* permettent de sélectionner à la fois :
 - les enregistrements ayant des correspondances entre les deux tables liées.
 - les enregistrements n'ayant *pas de correspondance* dans la première, dans la seconde ou dans toutes les tables.

SQL – Sans jointure

Si aucune condition de jointure est précisée, la requête correspondante réalisera le *produit cartésien* des attributs des tables impliquées

SELECT id_champ, culture, nomagriculteur **FROM** agriculteur, champ

champ		
Id_champ	culture	Idagriculteur
3106	blé	2
2455	verger	1

agriculteur	
Id.agriculteur	nomagriculteur
1	John
2	Céline



Id_champ	culture	nomagriculteur
2455	verger	John
2455	verger	Céline
3106	blé	John
3106	blé	Céline

SQL – jointure interne

Je sélectionne les champs qui sont cultivés par un agriculteur



SELECT C.id_champ, C.culture, A.nomagriculteur **FROM** champ C
INNER JOIN agriculteur A **ON** A.idagriculteur = C.idagriculteur

champ			agriculteur	
Id_champ	culture	Idagriculteur	Idagriculteur	nomagriculteur
3106	blé	2	1	John
2455	verger	1	2	Céline
			3	Robert



Id_champ	culture	nomagriculteur
2455	verger	John
3106	blé	Céline

SQL – jointure externe

Je sélectionne tous les agriculteurs, y compris ceux qui ne cultivent pas de champ



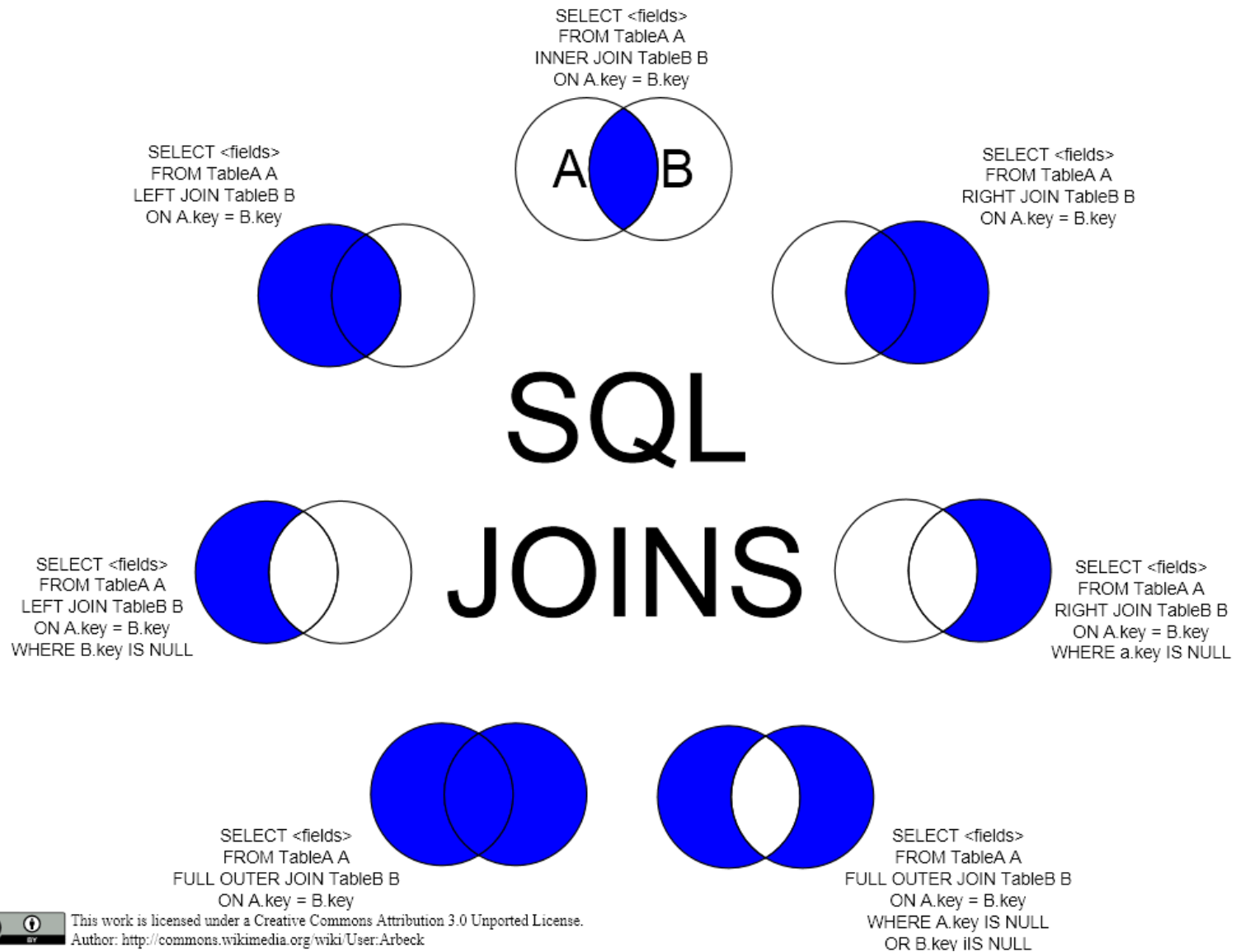
SELECT C.id_champ, C.culture, A.nomagriculteur **FROM** agriculteur A
RIGHT JOIN champ C **ON** A.idagriculteur = C.idagriculteur

champ			agriculteur	
Id_champ	culture	Idagriculteur	Idagriculteur	nomagriculteur
3106	blé	2	1	John
2455	verger	1	2	Céline
			3	Robert



Id_champ	culture	nomagriculteur
2455	verger	John
3106	blé	Céline
		Robert

SQL – Les jointures en un coup d'œil

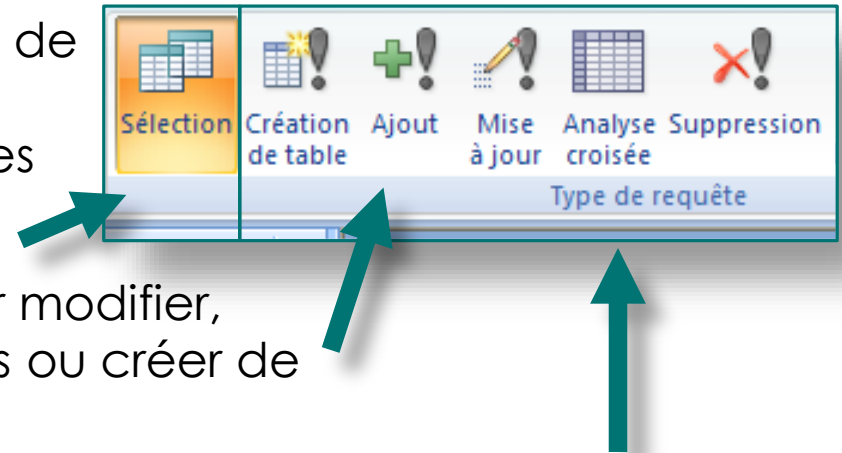


This work is licensed under a Creative Commons Attribution 3.0 Unported License.
Author: <http://commons.wikimedia.org/wiki/User:Arbeck>

Les requêtes dans Access

Il existe 3 grands types de requêtes :

- ❑ Les requêtes **sélections** qui permettent de sélectionner des enregistrements, d'effectuer des tris ou des calculs sur les données d'une ou plusieurs tables.
- ❑ Les requêtes **actions** sont utilisées pour modifier, supprimer, ajouter des enregistrements ou créer de nouvelles tables.
- ❑ Les requêtes d'**analyse croisée** présentent les résultats sous forme de tableaux type Excel.



Access offre 3 manières de créer une requête:

- ✓ en utilisant **les assistants** (requête simple, analyse croisée, trouver les doublons, de non correspondance)
- ✓ en la bâtissant du début à la fin en **mode création graphique**
- ✓ en l'écrivant en **langage SQL** en mode SQL.

Les étapes pour la création d'une requête

1. Sélectionner la/les tables et la/les requêtes
2. Choisir le type de requête sélection
3. Sélectionner le/les champs à afficher
4. Déterminer si les champs ont besoin d'être triés
5. Définir les critères de sélection
6. Options spécialisées : regroupement ou générateur d'expression
7. Cacher les champs au besoin
8. Exécuter la requête et valider le résultat
9. Convertir la requête sélection en requête action (Ajout, Suppression, Mise à jour, création de table)
10. Exécuter à nouveau la requête et confirmer l'opération

Requêtes action, attention !!! Impossible de revenir en arrière lorsque la requête a été exécutée. Créer et exécuter toujours une requête sélection avant de créer une requête action.

Les critères des requêtes dans Access

Un critère de requête est une **condition** appliquée à un **champ**, qui est définie en fonction des valeurs de ce champ.

C'est une chaîne qui comprend des **noms de champs**, des **opérateurs**, des **constantes**, des **valeurs** et des **fonctions** spécifiques.

Dans ACCESS, on parle aussi d'**expressions**

Exemple : concaténation du prénom et du nom = `[Prénom] &" " & [nom]`

Les formats de saisie standard d'un critère de requête sont :

- ✓ les **#** entourent les dates
- ✓ les **"** entourent les textes
- ✓ les **[]** entourent les noms d'objets. Obligatoire si le nom comporte un espace ou un signe de ponctuation.

Le générateur d'expression

Il permet de saisir des expressions plus rapidement et avec plus de précision

Exemple d'expression pour créer un champ **désignation** correspondant au **code du produit** suivi de **nom du produit**

Générateur d'expression

Entrez une expression pour définir le champ de requête calculé :
(Exemples d'expressions : [champ1] + [champ2] et [champ1] < 5)

Désignation: [Produits].[Code du produit] & "" & [Produits].[Nom du produit]

OK
Annuler
Aide
<< Moins

Éléments d'expression

- Factures
- Fournisseurs
- Opérations d'inven
- Privilèges
- Privilèges employé
- Produits
- TVA des commande
- Types d'opérations

Catégories d'expressions

- ID
- N° fournisseurs
- Code du produit
- Nom du produit
- Description
- Coût standard
- Prix unitaire
- Niveau réapprovisionnement
- Niveau ciblé
- Quantité par unité
- Rupture de stock

Valeurs d'expression

<Valeur>

Expression = Formule



Identificateurs :
champs de tables, de
formulaire et de requêtes



Opérateurs :
< , <= , > , >= , = , <>



Fonctions : somme,
moyenne, date...



Constantes : chaînes de
caractères, valeur numérique

Les opérateurs

Symbole	Description	Exemples
Opérateurs de comparaison		
< , <= > , >= = , <>	Est inférieur à; Est inférieur ou égal à, Est supérieur; Est supérieur ou égal à, Est égal à; Est différent de	Surface de la parcelle < 300 ha
Opérateur de concaténation		
&	Concatène des chaînes de caractères	[Prénom] & " " & [nom]
Opérateurs divers		
Entre Dans Est Null Est pas Null Pas Comme	Sélectionne les enregistrements dont les valeurs sont comprises dans l'intervalle Sélectionne les enregistrements dont la valeur est dans la liste Sélectionne les enregistrements qui n'ont pas ou seulement des valeurs Sélectionne les enregistrements qui ne correspondent pas au critère Effectue une sélection sur une donnée approximative	Entre 10 et 20 Entre 07/10/2012 et 10/12/2012 Dans ("Saint Denis", "Saint Pierre") Pas dans (« Saint Pierre ») Comme "R*" (commence par R et retourne Réunion...) Comme "1#3" (trouve 103, 113, 123...) Comme "*un*" (contient la chaîne de caractère un et retourne Réunion)

Opérations de regroupement

Opération	Description
Regroupement	Définir les groupes de données sur lesquelles effectuer les calculs
Somme	Calcule le total des valeurs du champ
Moyenne	Calcule la moyenne des valeurs du champ
Min	Trouve la valeur la plus petite du champ
Max	Trouve la valeur la plus grande du champ
Compte	Calcule le nombre de valeurs dans un champs, sans compter les valeurs nulles
EcartType	Calcule l'écart type des valeurs d'un champ
Var	Calcule la variance des valeurs d'un champ
Premier	Trouve la première valeur du champ
Dernier	Trouve la dernière valeur du champs
Expression	Pour créer un champ calculé incluant une fonction de regroupement
Où	Pour spécifier les critères d'un champ qui n'est pas utilisé pour définir des regroupement.

Exemple de requête

Requête_Diptera

Echantillon

- * numechantillon
- description
- codechamp
- pays
- plante
- identification
- guilde
- ordre
- famille
- genre
- espece

Champ : Echantillon.* Ordre

Table : Echantillon Echantillon

Tri :

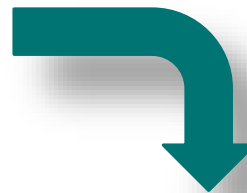
Afficher : ☒ ☐ ☐

Critères : "Diptera" ☐ ☐

Ou :

Création de la requête

Je sélectionne dans la table échantillon tous les échantillons d'insectes collectés dans le champ dont l'ordre est "Diptera"



Résultat de la requête

num	codechamp	pay	plante	identification	guilde	ordre	famille
19	RiS3	B	Ri	Sepedon sp.		Diptera	Sciomyzidae
20	RiS3	B	Ri			Diptera	
41	RiS3	B	Ri	Diopsis thoracica Westw.	phytophage	Diptera	Diopsidae
42	RiS3	B	Ri	Diopsis apicalis Dalman.	phytophage	Diptera	Diopsidae
58		B	He	Diopsis apicalis Dalman.	phytophage	Diptera	Diopsidae



Sandrine Auzoux

UR AIDA

Jean-Christophe Soulié

UR Recyclage et Risque

Données, base de données et SGBD
Modèle conceptuel
Modèle logique
Modèle physique
Importation/exportation de données
Requêtes

Logiciel R et bases de données

Utilisation du package RODBC

Conception de bases
de données
expérimentales à des
fins de modélisation

Interfaçage avec R

 **est, avant tout, un logiciel libre destiné aux statistiques et à la science des données.**

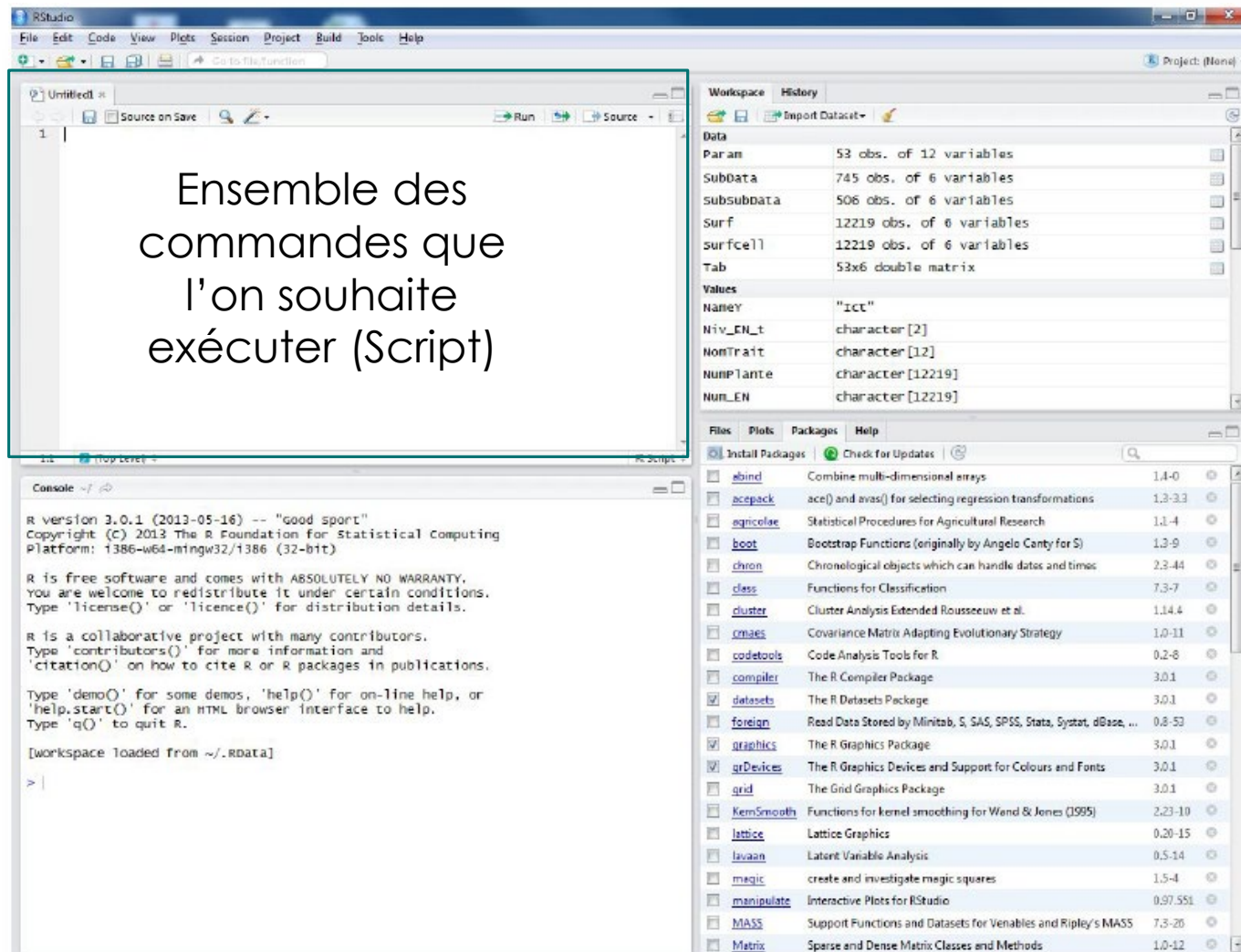
Il dispose d'une bibliothèque très large de **fonctions statistiques**, d'autant plus large qu'il est possible d'en intégrer de nouvelles par le système des "**packages**", modules externes, que l'on peut télécharger gratuitement sur internet.

 propose également une gamme étendue de **fonctionnalités graphiques**.

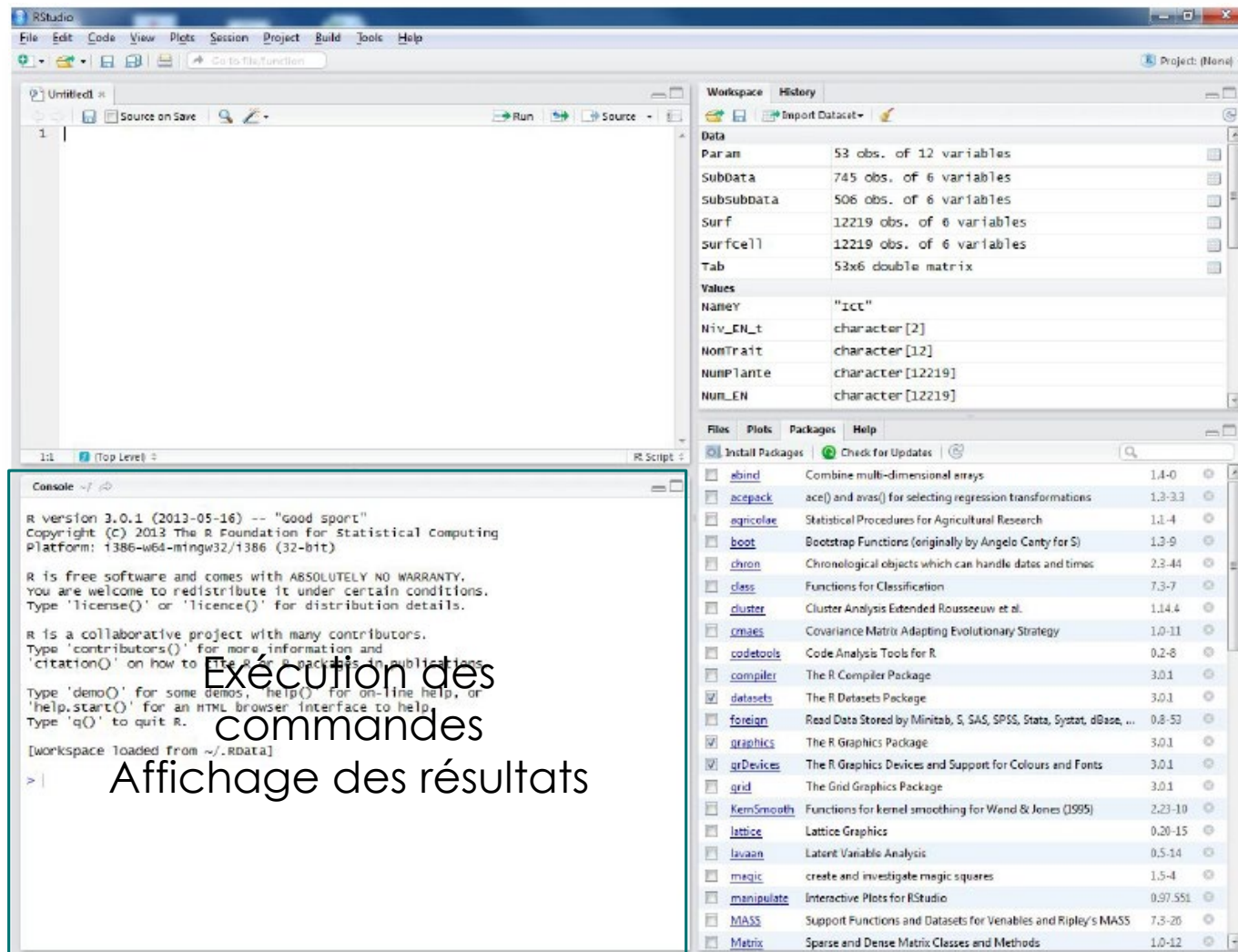
 est, aussi, un langage de programmation

Il est ainsi possible de créer de **nouvelles fonctions** de traitement de données.

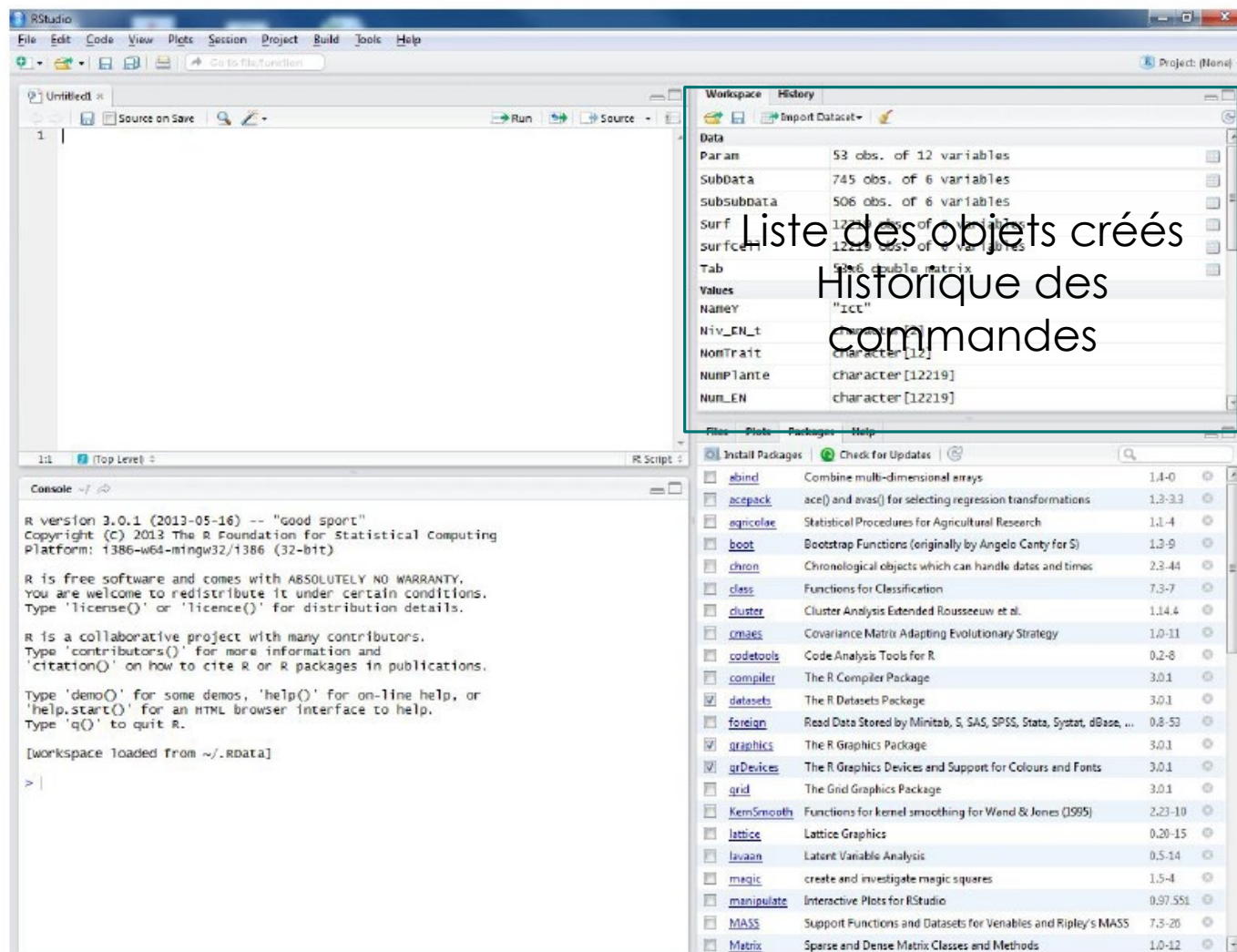
L'interface R Studio



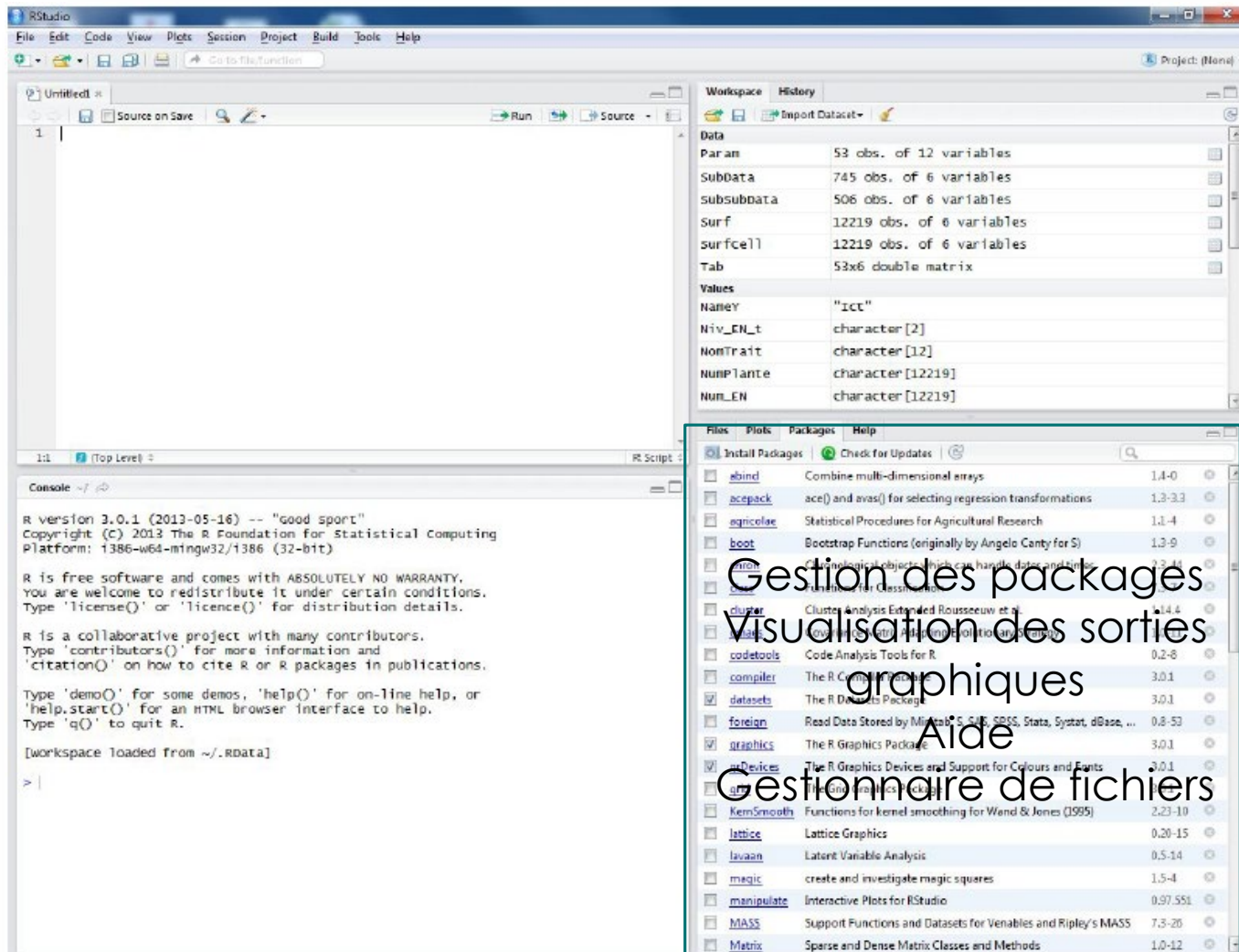
L'interface R Studio



L'interface R Studio



L'interface R Studio



Gestion des packages
Visualisation des sorties
graphiques
Aide
Gestionnaire de fichiers

Que sait gérer ?

Sous  on travaille avec

- des nombres 0, 1, -3.76, π ;
- des chaînes de caractères "genotype", "tree" ;
- des "logiques" TRUE, FALSE.

Pour faire des analyses statistiques ou des graphiques, il est nécessaire de stocker les données en mémoire.

On crée pour cela des objets (vecteur, liste, matrice, tableau de données, ...), auxquels on donne un nom. Cette opération s'appelle l'assignation et se fait à l'aide du symbole <-.

Les objets

 manipule des objets définis par :

- leur **nom**
- leur **contenu**
- des **attributs** qui vont caractériser le type de données représentées par l'objet.

Les objets  ont tous, au moins, **deux attributs** :

- le **mode** indiquant le type des éléments de l'objet (numérique, caractère, logique, fonction, ...) ;
- la **longueur** donnant le nombre d'éléments contenus dans l'objet.

Les objets

Objet	Mode	Plusieurs modes possibles dans le même objet ?
Vecteur	numérique, caractère ou logique	Non
Facteur	numérique ou caractère	Non
Matrice	numérique, caractère ou logique	Non
Tableau de données	numérique, caractère ou logique	Oui
Liste	numérique, caractère, logique, ...	Oui

Les objets

Objet	Mode	Plusieurs modes possibles dans le même objet ?
Requêtes		
Vecteur	numérique, caractère ou logique	Non
Facteur	numérique ou caractère	Non
Matrice	numérique, caractère ou logique	Non
Résultats		
Tableau de données	numérique, caractère ou logique	Oui
Liste	numérique, caractère, logique, ...	Oui

Les vecteurs

Un vecteur est constitué d'une série d'éléments de même mode. La manière la plus courante de créer un vecteur est d'utiliser la fonction `c()` (concaténation).

```
>
> v1 <- c(1,5,24)
> v1
[1] 1 5 24
> mode(v1)
[1] "numeric"
>
> v2 <- c("arbre","feuille","racine")
> v2
[1] "arbre" "feuille" "racine"
> mode(v2)
[1] "character"
>
> v3 <- c(TRUE,FALSE,TRUE)
> v3
[1] TRUE FALSE TRUE
> mode(v3)
[1] "logical"
>
> v4 <- c(1,"leaf")
> v4
[1] "1" "leaf"
> mode(v4)
[1] "character"
>
```

Manipulation des vecteurs de chaînes de caractères

```
>
> Name1 <- c("Dupont")
> FirstName1 <- c("Pierre")
> n1 <- c(FirstName1,Name1)
>
> Name1
[1] "Dupont"
> FirstName1
[1] "Pierre"
> n1
[1] "Pierre" "Dupont"
> length(n1)
[1] 2
> n1bis <- paste(FirstName1,Name1,sep=" ")
> n1bis
[1] "Pierre Dupont"
> length(n1bis)
[1] 1
> |
```

```
> B <- rep("Bloc",5)
> num <- seq(from=1,to=5,by=1)
> paste(B,num,sep="_")
[1] "Bloc_1" "Bloc_2" "Bloc_3" "Bloc_4" "Bloc_5"
```

Les "data.frame"

Les **tableaux de données** (data.frame) sont des objets consacrés spécifiquement au stockage des données destinées à l'analyse.

Ce type de tableau "**individus X variables**" est analogue à une matrice dont les colonnes peuvent être de différents modes.

Chaque **colonne** correspond à une **variable**, et chaque **ligne** à un **individu**.

Un tableau de données se crée soit par **l'import** d'un fichier de données au moyen de la fonction **read.table**, soit directement à l'aide de la fonction **data.frame**.

Il peut être aussi le résultat de l'appel d'une fonction R (**sqlFetch**, **sqlQuery**, ...)

Les "data.frame"

```
> MaConnexion <- odbcConnect("GARP_Formation")
> TabSite <- sqlFetch(MaConnexion, "Site")
> is.data.frame(TabSite)
[1] TRUE
> TabSite
  NumSite      Site StationMeteo Altitude
1      1  Andranomanelatra    cime1    1645
2      2         Ivory    cime1     954
> odbcCloseAll()
> |
```

Les "data.frame"

Si X est un tableau de données, on accèdera à la valeur de la i ème ligne et j ème colonne en exécutant la commande `X[i, j]`.

Pour accéder à toutes les valeurs d'une ligne (resp. **colonne**) donnée, il suffit simplement d'omettre l'indice de la colonne (resp. **ligne**) sans oublier la virgule.

Si le tableau X a des noms de colonnes (**variables**), ceux-ci peuvent être utilisés pour accéder à l'information contenue dans le tableau.

```
> TabSite[2,3]
[1] Cime1
Levels: Cime1
>
> TabSite[1,]
  NumSite      Site StationMeteo Altitude
1        1 Andranomanelatra      Cime1    1645
>
> TabSite[,4]
[1] 1645  954
>
> TabSite$Site
[1] Andranomanelatra Ivory
Levels: Andranomanelatra Ivory
>
> TabSite[TabSite$Site=="Ivory","Altitude"]
[1] 954
```

Le package RODB

Les fonctions du package RODB permettent de gérer des bases de données sous différents **Systèmes de Gestion de Base de Données** : *PostgreSql, MySQL, Microsoft Access ...*, par l'intermédiaire de **R** grâce à un **lien** Open DataBase Connectivity (**ODBC**) préalablement créé avec une base de données déjà existante.

3 fonctions principales :

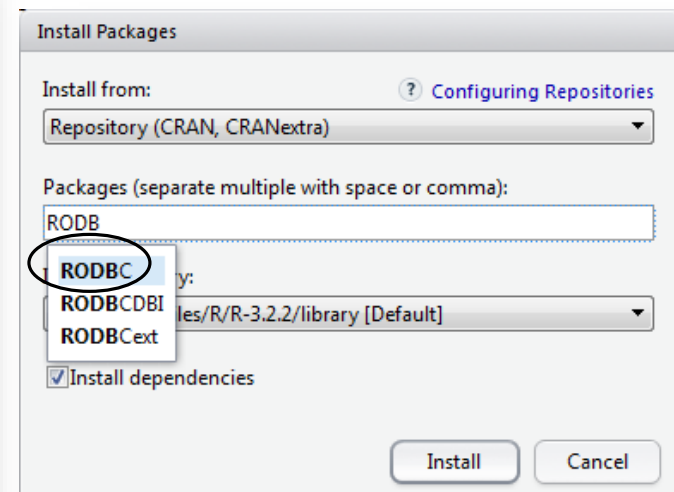
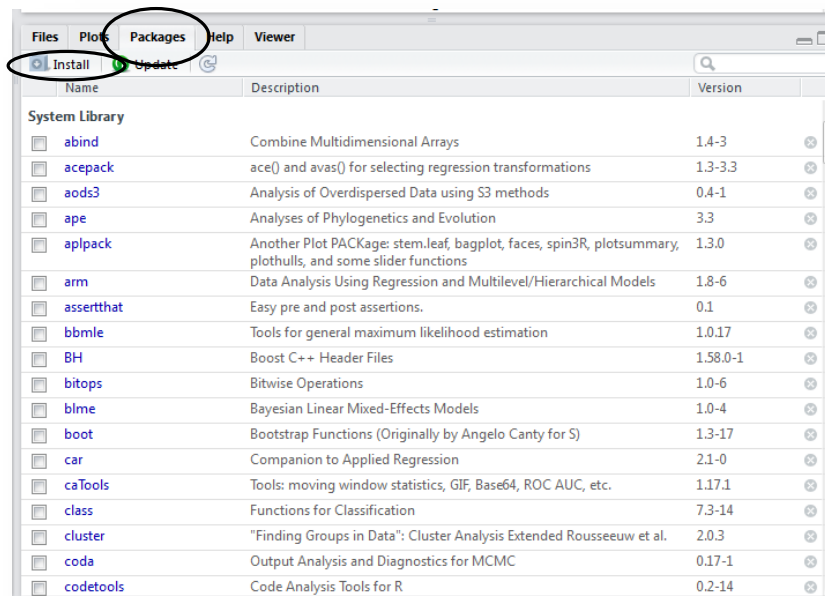
- ✓ **odbcConnect**, permet d'établir une connexion,
- ✓ **sqlQuery**, permet d'exécuter une requête,
- ✓ **odbcClose**, ferme la connexion établie.

Intérêts et inconvénients:

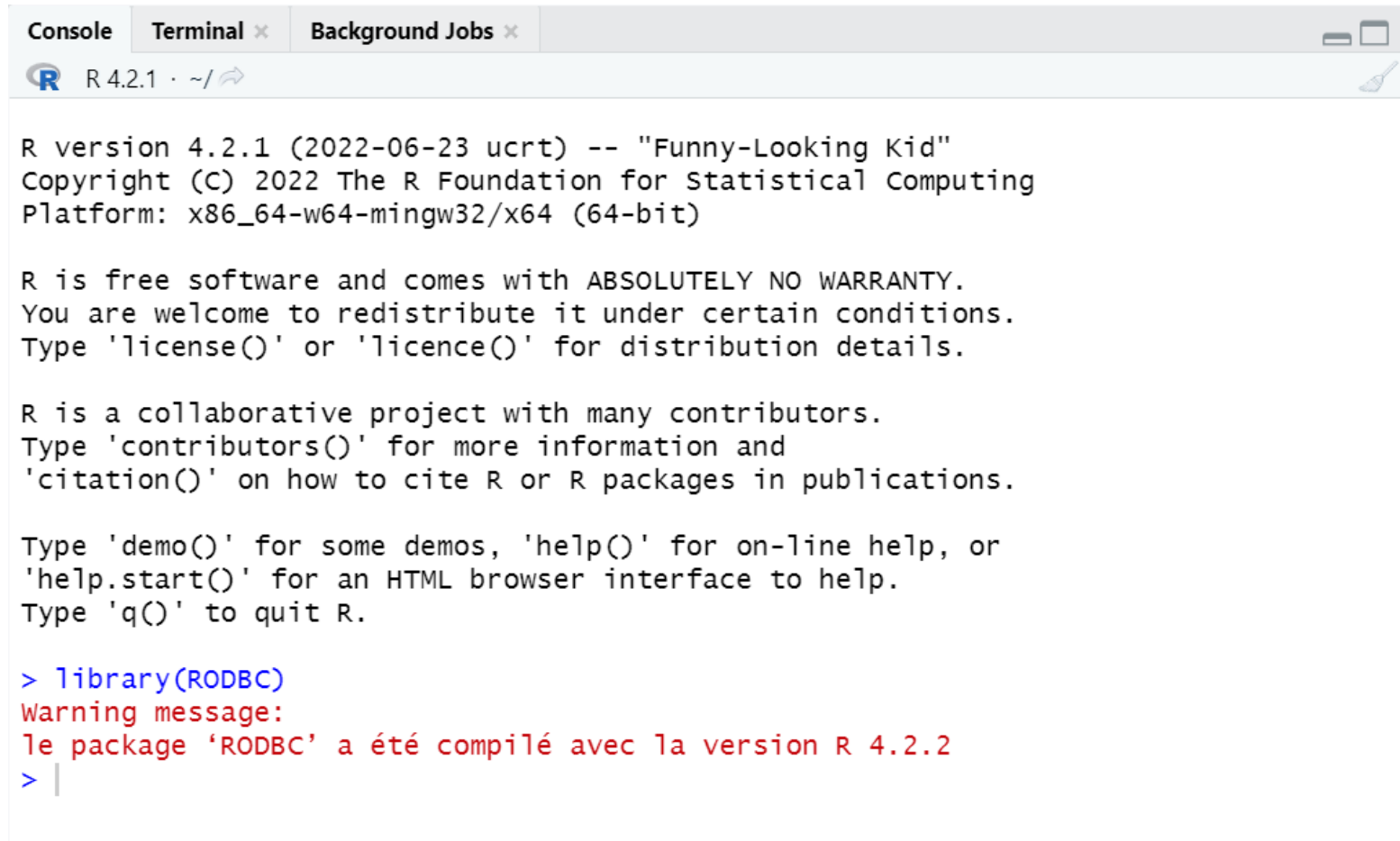
- ✓ Ce package permet d'entrer ou de récupérer des données directement dans une base de données à partir de R ;
- ✓ L'installation de ce seul package permet d'utiliser les SGBD les plus répandus.

Installation du Package R

Depuis Rstudio, aller dans l'onglet **Packages** et cliquer sur **Install**.
Si une connexion internet est disponible, choisir **Install from Repository (CRAN, CRANextra)** et saisir le nom du package **RODBC**.



Chargement du Package R



```
Console Terminal x Background Jobs x
R 4.2.1 · ~/ ↻

R version 4.2.1 (2022-06-23 ucrt) -- "Funny-Looking Kid"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> library(RODBC)
Warning message:
le package 'RODBC' a été compilé avec la version R 4.2.2
> |
```

Etablissement et fermeture d'une connexion

Avant de faire de pouvoir faire des requêtes, il est nécessaire de se connecter à la base de données

```
> library(RODBC)
```

Warning message: le package 'RODBC' a été compilé avec la version R 4.2.2

```
> DRIVERINFO <- "Driver={Microsoft Access Driver (*.mdb, *.accdb)};"
```

```
> MDBPATH <- "D:/Mes Donnees/Enseignement/2023/BD Cirad La Bretagne Mars/GARP.accdb"
```

```
> PATH <- paste0(DRIVERINFO, "DBQ=", MDBPATH)
```

```
>
```

```
> channel <- odbcDriverConnect(PATH)
```

S'il y a un mot de passe restreignant l'accès à la base il faut le spécifier de la manière suivante

```
> channel <- odbcDriverConnect(PATH, uid = "Nom Utilisateur", pwd = "mot_de_passe")
```

Une fois les manipulations faites sur la (les) base(s) de données ouverte(s), il faut fermer la (les) connexion(s)

```
> odbcClose(channel)           Fermeture d'une connexion en particulier
```

```
> odbcCloseAll()              Fermeture de toutes les connexions ouvertes
```

Lecture de la base de données

Pour savoir quelles sont les tables accessibles par la connexion ouverte, on utilise la fonction `sqlTables`

```
> sqlTables(channel)
```

						TABLE_CAT	TABLE_SCHEM	TABLE_NAME	TABLE_TYPE	REMARKS	
1	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysAccessStorage	SYSTEM	TABLE	<NA>
2	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysAccessXML	SYSTEM	TABLE	<NA>
3	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysACES	SYSTEM	TABLE	<NA>
4	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysComplexColumns	SYSTEM	TABLE	<NA>
5	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysIMEXColumns	SYSTEM	TABLE	<NA>
6	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysIMEXSpecs	SYSTEM	TABLE	<NA>
7	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysNameMap	SYSTEM	TABLE	<NA>
8	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysNavPaneGroupCategories	SYSTEM	TABLE	<NA>
9	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysNavPaneGroups	SYSTEM	TABLE	<NA>
10	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysNavPaneGroupToObjects	SYSTEM	TABLE	<NA>
11	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysNavPaneObjectIDs	SYSTEM	TABLE	<NA>
12	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysObjects	SYSTEM	TABLE	<NA>
13	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysQueries	SYSTEM	TABLE	<NA>
14	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysRelationships	SYSTEM	TABLE	<NA>
15	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	MSysResources	SYSTEM	TABLE	<NA>
16	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Bilan_Parcelles_Feuilles		TABLE	<NA>
17	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Bilan_Poquet_Feuilles		TABLE	<NA>
18	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Bilan_Poquet_Pani		TABLE	<NA>
19	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Date_Phenologique		TABLE	<NA>
20	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	DiagnosticFoliaire		TABLE	<NA>
21	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	DonneesMeteo		TABLE	<NA>
22	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Extraction		TABLE	<NA>
23	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	FertiSiteAnnee		TABLE	<NA>
24	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Intervention		TABLE	<NA>
25	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Parcelle		TABLE	<NA>
26	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Parcelle_Temp		TABLE	<NA>
27	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Prelevement		TABLE	<NA>
28	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	PyriFoliaire		TABLE	<NA>
29	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	PyriPaniculaire		TABLE	<NA>
30	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Recolte		TABLE	<NA>
31	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	Site		TABLE	<NA>
32	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	TableDesNoms		TABLE	<NA>
33	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	TableListe		TABLE	<NA>
34	D:/Mes	Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb		<NA>	VarSiteAnnee		TABLE	<NA>

Lecture de la base de données

Pour accéder aux informations sur le contenu d'une table, on utilise la fonction `sqlColumns`

```
> sqlColumns(channel, "Site")
```

					TABLE_CAT	TABLE_SCHEM	TABLE_NAME	COLUMN_NAME	DATA_TYPE	TYPE_NAME	COLUMN_SIZE	BUFFER_LENGTH
1	D:/Mes Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb	<NA>		Site	NumSite	4	COUNTER	10	4
2	D:/Mes Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb	<NA>		Site	Site	12	VARCHAR	50	100
3	D:/Mes Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb	<NA>		Site	StationMeteo	12	VARCHAR	255	510
4	D:/Mes Donnees/Enseignement/2023/BD	Cirad	La Bretagne	Mars/GARP.acddb	<NA>		Site	Altitude	4	INTEGER	10	4
	DECIMAL_DIGITS	NUM_PREC_RADIX	NULLABLE	REMARKS	COLUMN_DEF	SQL_DATA_TYPE	SQL_DATETIME_SUB	CHAR_OCTET_LENGTH	ORDINAL_POSITION	IS_NULLABLE	ORDINAL	
1	0	10	0	<NA>	<NA>	4	NA	NA	1	NO	1	
2	NA	NA	1	<NA>	<NA>	12	NA	100	2	YES	2	
3	NA	NA	1	<NA>	<NA>	12	NA	510	3	YES	3	
4	0	10	1	<NA>	<NA>	4	NA	NA	4	YES	4	

Lecture de la base de données

Pour importer tout ou partie d'une table dans R sous la forme d'un `data.frame`, on utilise la fonction `sqlFetch`

```
> sqlFetch(channel, "FertiSiteAnnee")
```

	numFert	Fertilisation	Type	Caracteristiques	Annee	Site
1	1	F1 fumure de base		sans azote	2009-2010	1
2	2	F2 fumure de base		30N + 23N + 23N	2009-2010	1
3	3	Fv fumure de base		30N + 23N	2009-2010	1
4	4	F1 fumure de base		sans azote	2010-2011	1
5	5	F2 fumure de base		30N	2010-2011	1
6	6	Fv fumure de base		60N + 46N + 46N	2010-2011	1
7	7	F1 fumure de base		sans azote	2010-2011	2
8	8	F2 fumure de base		30N + 15N + 15N	2010-2011	2
9	9	Fv fumure de base		60N + 30N + 30N	2010-2011	2
10	10	F1 fumure de base		sans azote	2011-2012	1
11	11	F2 fumure de base		30N + 23N + 23N	2011-2012	1
12	12	Fv fumure de base		60N + 46N + 46N	2011-2012	1
13	13	F1 fumure de base		sans azote	2011-2012	2
14	14	F2 fumure de base		30N + 15N + 15N	2011-2012	2
15	15	Fv fumure de base		60N + 30N + 30N	2011-2012	2
16	16	F1 fumure de base		sans azote	2012-2013	1
17	17	F2 fumure de base		30N + 23N + 23N	2012-2013	1
18	18	Fv fumure de base		30N + 46N	2012-2013	1
19	19	F1 fumure de base		sans azote	2012-2013	2
20	20	F2 fumure de base		30N + 15N + 15N	2012-2013	2
21	21	Fv fumure de base		30N + 30N	2012-2013	2
22	22	F1 fumure de base		sans azote	2013-2014	1
23	23	F2 fumure de base		30N + 23N + 23N	2013-2014	1
24	24	Fv fumure de base		30N + 46N	2013-2014	1
25	25	Fv fumure de base		30N + 30N	2013-2014	2
26	26	F1 fumure de base		sans azote	2013-2014	2
27	27	F2 fumure de base		30N + 15N + 15N	2013-2014	2

Lecture de la base de données

```
> sqlFetch(channel, "FertiSiteAnnee", max = 10)
```

	numFert	Fertilisation	Type	Caracteristiques	Annee	Site
1	1	F1 fumure de base		sans azote	2009-2010	1
2	2	F2 fumure de base	30N + 23N + 23N		2009-2010	1
3	3	Fv fumure de base		30N + 23N	2009-2010	1
4	4	F1 fumure de base		sans azote	2010-2011	1
5	5	F2 fumure de base		30N	2010-2011	1
6	6	Fv fumure de base	60N + 46N + 46N		2010-2011	1
7	7	F1 fumure de base		sans azote	2010-2011	2
8	8	F2 fumure de base	30N + 15N + 15N		2010-2011	2
9	9	Fv fumure de base	60N + 30N + 30N		2010-2011	2
10	10	F1 fumure de base		sans azote	2011-2012	1

```
> sqlFetchMore(channel, max = 10)
```

	numFert	Fertilisation	Type	Caracteristiques	Annee	Site
1	21	Fv fumure de base		30N + 30N	2012-2013	2
2	22	F1 fumure de base		sans azote	2013-2014	1
3	23	F2 fumure de base	30N + 23N + 23N		2013-2014	1
4	24	Fv fumure de base		30N + 46N	2013-2014	1
5	25	Fv fumure de base		30N + 30N	2013-2014	2
6	26	F1 fumure de base		sans azote	2013-2014	2
7	27	F2 fumure de base	30N + 15N + 15N		2013-2014	2

Exécution d'une requête SQL

Les requêtes SQL s'exécute à l'aide de la fonction `sqlQuery`

```
> request <- "SELECT * FROM Site"
> sqlQuery(channel, request)
```

	NumSite	Site	StationMeteo	Altitude
1	1	Andranomanelatra	Cime1	1645
2	2	Ivory	Cime1	954

« request » est une **chaîne de caractères**.

Le résultat de la requête est renvoyé sous forme d'un `data.frame`

Expérimentation et données (Yates, 1935)

- 4 variétés d'avoine
- 4 doses d'azote
- Rendement

v_3	n_3 156	n_2 118	n_2 109	n_3 99	v_3
	n_1 140	n_0 105	n_0 63	n_1 70	
v_1	n_0 111	n_1 130	n_0 80	n_2 94	v_1
	n_3 174	n_2 157	n_2 126	n_1 82	
v_2	n_0 117	n_1 114	n_1 90	n_2 100	v_2
	n_2 161	n_3 141	n_3 116	n_0 62	
v_3	n_2 104	n_0 70	n_2 96	n_0 60	v_3
	n_1 89	n_3 117	n_1 89	n_1 102	
v_1	n_3 122	n_0 74	n_2 112	n_3 86	v_1
	n_1 89	n_2 81	n_0 68	n_1 64	
v_2	n_2 103	n_0 64	n_2 132	n_3 124	v_2
	n_3 132	n_1 133	n_1 129	n_0 89	
v_3	n_1 108	n_2 126	n_2 118	n_0 53	v_3
	n_3 149	n_0 70	n_3 113	n_1 74	
v_1	n_2 144	n_1 124	n_2 104	n_2 86	v_1
	n_0 121	n_0 96	n_0 89	n_1 82	
v_2	n_0 61	n_3 100	n_0 97	n_1 99	v_2
	n_1 91	n_2 97	n_2 119	n_3 121	

← FOWS →

Area of each plot : 1/80 acre. (28.4 links \times 44 link rows.)

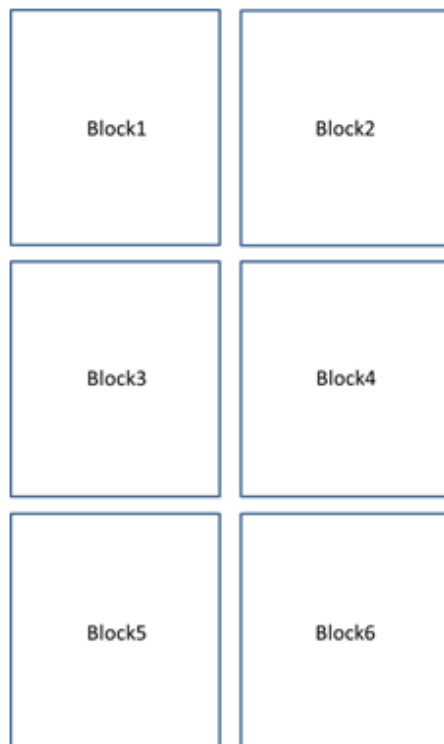
FIG. 2.

Oats Variety and Manuring Experiment.

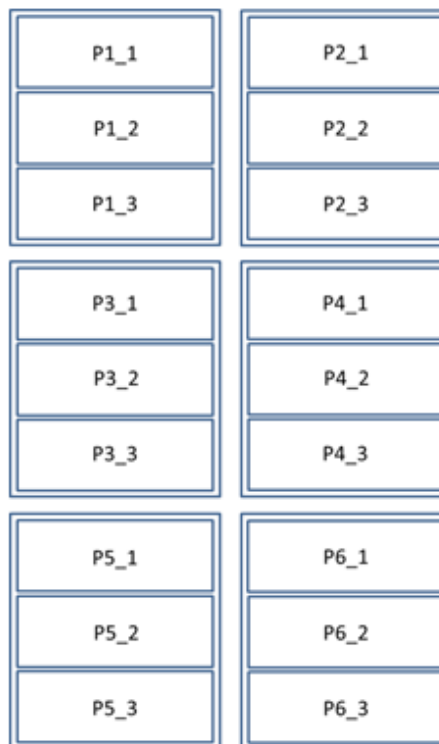
Description de l'expérimentation

Les unités expérimentales

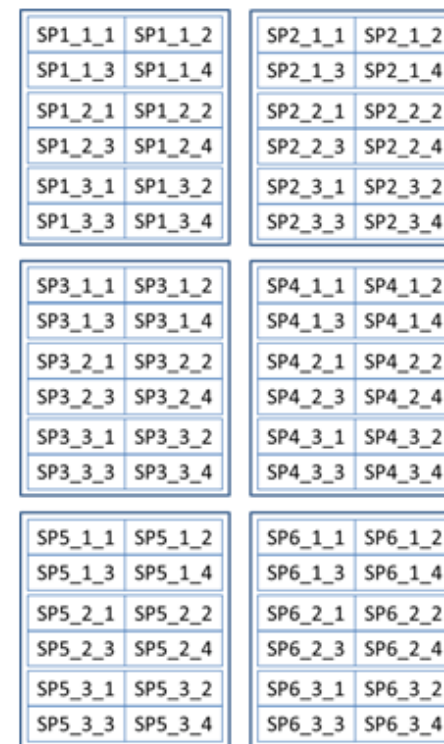
(A)



(B)



(C)



Description de l'expérimentation

Les facteurs

(A)

Variety 3	Variety 3
Variety 1	Variety 2
Variety 2	Variety 1

Variety 3	Variety 2
Variety 1	Variety 1
Variety 2	Variety 3

Variety 2	Variety 1
Variety 3	Variety 2
Variety 1	Variety 3

(B)

n_3	n_2
n_1	n_0
n_0	n_1
n_3	n_2
n_0	n_1
n_2	n_3

n_2	n_3
n_0	n_1
n_0	n_2
n_3	n_1
n_1	n_2
n_3	n_0

n_2	n_0
n_1	n_3
n_3	n_0
n_1	n_2
n_1	n_0
n_2	n_3

n_3	n_0
n_2	n_1
n_2	n_3
n_0	n_1
n_2	n_3
n_1	n_0

n_1	n_2
n_3	n_0
n_3	n_1
n_2	n_0
n_0	n_3
n_1	n_2

n_2	n_0
n_3	n_1
n_3	n_2
n_0	n_1
n_0	n_3
n_2	n_3

(C)

Variety 3, n_3	Variety 3, n_2
Variety 3, n_1	Variety 3, n_0
Variety 1, n_0	Variety 1, n_1
Variety 1, n_3	Variety 1, n_2
Variety 2, n_0	Variety 2, n_1
Variety 2, n_2	Variety 2, n_3

Variety 3, n_2	Variety 3, n_3
Variety 3, n_0	Variety 3, n_1
Variety 2, n_0	Variety 2, n_2
Variety 2, n_3	Variety 2, n_1
Variety 1, n_1	Variety 1, n_2
Variety 1, n_3	Variety 1, n_0

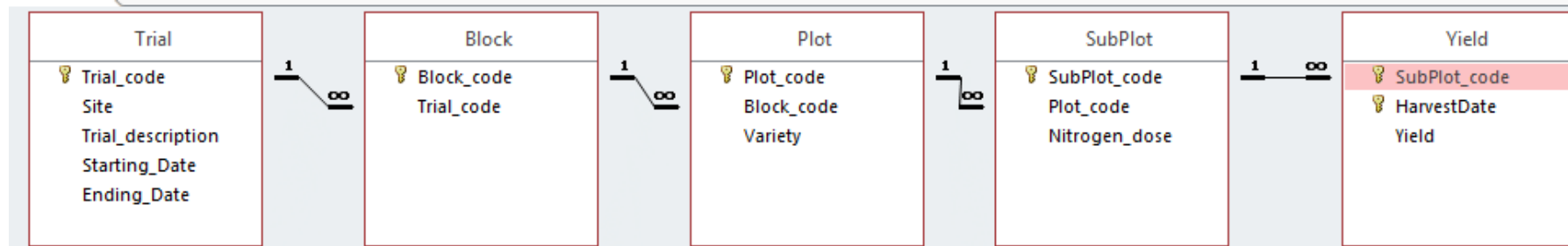
Variety 3, n_2	Variety 3, n_0
Variety 3, n_1	Variety 3, n_3
Variety 1, n_3	Variety 1, n_0
Variety 1, n_1	Variety 1, n_2
Variety 2, n_1	Variety 2, n_0
Variety 2, n_2	Variety 2, n_3

Variety 2, n_3	Variety 2, n_0
Variety 2, n_2	Variety 2, n_1
Variety 1, n_2	Variety 1, n_3
Variety 1, n_0	Variety 1, n_1
Variety 3, n_2	Variety 3, n_3
Variety 3, n_1	Variety 3, n_0

Variety 2, n_1	Variety 2, n_2
Variety 2, n_3	Variety 2, n_0
Variety 3, n_3	Variety 3, n_1
Variety 3, n_2	Variety 3, n_0
Variety 1, n_0	Variety 1, n_3
Variety 1, n_1	Variety 1, n_2

Variety 1, n_2	Variety 1, n_0
Variety 1, n_3	Variety 1, n_1
Variety 2, n_3	Variety 2, n_2
Variety 2, n_0	Variety 2, n_1
Variety 3, n_0	Variety 3, n_1
Variety 3, n_2	Variety 3, n_3

Schéma / Implémentation



```
PATH <- "D:/Mes Donnees/Enseignement/2023/BD Cirad La Bretagne Mars/td_r/"
DATABASENAME <- "MaBDD.accdb"
DRIVERINFO <- "Driver={Microsoft Access Driver (*.mdb, *.accdb)};"
MDBPATH <- paste0(PATH, DATABASENAME)
FULLPATH <- paste0(DRIVERINFO, "DBQ=", MDBPATH)
```

```
setwd(PATH)
```

```
library(RODBC)
```

```
MaConnexion <- odbcDriverConnect(FULLPATH)
```

```
sqlQuery(MaConnexion, "CREATE TABLE Trial
  (Trial_code VARCHAR(10) NOT NULL PRIMARY KEY,
   Site VARCHAR(50),
   Trial_description VARCHAR(250),
   Starting_Date DATE,
   Ending_Date DATE)"
```

```
)
```

Schéma / Implémentation

```
sqlQuery(MaConnexion, "CREATE TABLE Block
    (Block_code VARCHAR(10) NOT NULL PRIMARY KEY,
    Trial_code VARCHAR(10) REFERENCES Trial(Trial_code))"
)

sqlQuery(MaConnexion, "CREATE TABLE Plot
    (Plot_code VARCHAR(10) NOT NULL PRIMARY KEY,
    Block_code VARCHAR(10) REFERENCES Block(Block_code),
    Variety VARCHAR(20)
)"
)

sqlQuery(MaConnexion, "CREATE TABLE SubPlot
    (SubPlot_code VARCHAR(10) NOT NULL PRIMARY KEY,
    Plot_code VARCHAR(10) REFERENCES Plot(Plot_code),
    Nitrogen_dose VARCHAR(2))"
)

sqlQuery(MaConnexion, "CREATE TABLE Yield
    (SubPlot_code VARCHAR(10) REFERENCES SubPlot(SubPlot_code),
    HarvestDate DATE,
    Yield NUMERIC,
    PRIMARY KEY (SubPlot_code, HarvestDate))"
)
```

Alimentation de la base

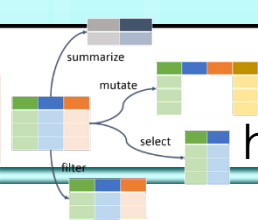
```
Trial_df <- read.table("Trial.txt",header=TRUE,sep='\t',  
                      colClasses=c("character","character","character","Date","Date"))  
Block_df <- read.table("Block.txt",header=TRUE,sep='\t',  
                      colClasses=c("character","character"))  
Plot_df <- read.table("Plot.txt",header=TRUE,sep='\t',  
                    colClasses=c("character","character","character"))  
SubPlot_df <- read.table("SubPlot.txt",header=TRUE,sep='\t',  
                        colClasses=c("character","character","character"))  
Yield_df <- read.table("Yield.txt",header=TRUE,sep='\t',  
                      colClasses=c("character","Date","numeric"))
```

```
sqlSave(MaConnexion, tablename="Trial", dat=Trial_df, append=TRUE, rownames=FALSE)  
sqlSave(MaConnexion, tablename="Block", dat=Block_df, append=TRUE, rownames=FALSE)  
sqlSave(MaConnexion, tablename="Plot", dat=Plot_df, append=TRUE, rownames=FALSE)  
sqlSave(MaConnexion, tablename="SubPlot", dat=SubPlot_df, append=TRUE, rownames=FALSE)  
sqlSave(MaConnexion, tablename="Yield", dat=Yield_df, append=TRUE, rownames=FALSE)
```

```
odbcCloseAll()
```

Quelques requêtes

- Quels sont les plots qui contiennent la variété Victory dans les blocs 1 et 2 ?
- Quels sont les subplots qui contiennent la variété Victory et ont la dose n1 dans les blocs 1 et 2 ?
- Calculer le rendement moyen par Block



Package dplyr

<https://cran.r-project.org/web/packages/dplyr/dplyr.pdf>

dplyr est un package R qui permet de manipuler facilement les données contenues dans une ou plusieurs tables (data frame).

Sa syntaxe est relativement simple et permet de faire des requêtes sans utiliser le langage SQL.

Data Wrangling with dplyr and tidyr

Cheat Sheet

Studio

Syntax - Helpful conventions for wrangling

dplyr::tbl, df (iris)
Converts data to tbl class. tbl's are easier to examine than data frames. R displays only the data that fits on screen.

```
Source: local data frame [150 x 5]
  Sepal.Length Sepal.Width Petal.Length
1           5.1           3.5           1.4
2           4.9           3.0           1.4
3           4.7           3.2           1.3
4           4.6           3.1           1.5
5           5.0           3.6           1.4
```

Variables not shown: Petal.Width (tbl), Species (factor)

dplyr::glimpse(iris)
Information dense summary of tbl data.

utils::View(iris)
View data set in spreadsheet-like display (note capital V).

```
tbl_df [5 x 5]
  Sepal.Length Sepal.Width Petal.Length
1           5.1           3.5           1.4
2           4.9           3.0           1.4
3           4.7           3.2           1.3
4           4.6           3.1           1.5
5           5.0           3.6           1.4
```

dplyr::%>%
Passes object on left hand side as first argument (or argument) of function on right hand side.

x %>% f(y) is the same as *f(x, y)*
y %>% f(x, ..., z) is the same as *f(x, y, z)*

"Piping" with %>% makes code more readable, e.g.

```
iris %>%
  group_by(Species) %>%
  summarise(avg = mean(Sepal.Length)) %>%
  arrange(avg)
```

Tidy Data - A foundation for wrangling in R

In a tidy data set:

- Each variable is saved in its own column
- Each observation is saved in its own row

Tidy data complements R's vectorized operations. R will automatically preserve observations as you manipulate variables. No other format works as intuitively with R.

$M * A \rightarrow M * A$

Reshaping Data - Change the layout of a data set

tidyr::gather(cases, "year", "n", 2:4)
Gather columns into rows.

tidyr::spread(pollution, size, amount)
Spread rows into columns.

tidyr::separate(storms, date, c("y", "m", "d"))
Separate one column into several.

tidyr::unite(data, col, ..., sep)
Unite several columns into one.

dplyr::data_frame(a = 1:3, b = 4:6)
Combine vectors into data frame (optional).

dplyr::arrange(mtcars, mpg)
Order rows by values of a column (low to high).

dplyr::arrange(mtcars, desc(mpg))
Order rows by values of a column (high to low).

dplyr::rename(tbl, y = year)
Rename the columns of a data frame.

Subset Observations (Rows)

dplyr::filter(iris, Sepal.Length > 7)
Extract rows that meet logical criteria.

dplyr::distinct(iris)
Remove duplicate rows.

dplyr::sample_frac(iris, 0.5, replace = TRUE)
Randomly select fraction of rows.

dplyr::sample_n(iris, 10, replace = TRUE)
Randomly select n rows.

dplyr::slice(iris, 10:15)
Select rows by position.

dplyr::top_n(storms, 2, date)
Select and order top n entries (by group if grouped data).

Subset Variables (Columns)

dplyr::select(iris, Sepal.Width, Petal.Length, Species)
Select columns by name or helper function.

Helper functions for select::select

- select(iris, contains("y"))**
Select columns whose name contains a character string.
- select(iris, ends_with("Length"))**
Select columns whose name ends with a character string.
- select(iris, everything())**
Select a very column.
- select(iris, matches("x"))**
Select columns whose name matches a regular expression.
- select(iris, num_range("x", 1:5))**
Select columns named x1, x2, x3, x4, x5.
- select(iris, one_of("Species", "Genus"))**
Select columns whose name is in a group of names.
- select(iris, starts_with("Sepal"))**
Select columns whose name starts with a character string.
- select(iris, Sepal.Length:Petal.Width)**
Select all columns between Sepal.Length and Petal.Width (inclusive).
- select(iris, -Species)**
Select all columns except Species.



Plotly R Open Source Graphing Library

Plotly's R graphing library makes interactive, publication-quality graphs. Examples of how to make line plots, scatter plots, area charts, bar charts, error bars, box plots, histograms, heatmaps, subplots, multiple-axes, and 3D (WebGL based) charts.

Plotly.R is [free and open source](#) and you can [view the source](#), [report issues](#) or [contribute on GitHub](#).

