



Base de Données 'COLZA'

Présentation
et
Mises à jour

Sonia Gharbi et Katarzyna Adamczyk

Juillet - Septembre 2002

Table des matières

I	Présentation de la Base de Données 'COLZA'	4
1	Introduction	5
2	Base de données 'COLZA': Rôle	6
3	Base de données 'COLZA': Connexion	7
3.1	Interrogation de la base (contenu des tables, requêtes prédéfinies): Via internet	7
3.2	Modification de la base (tables et requêtes): Via banian	12
3.2.1	Interfaces PSQL et PgAccess	12
4	Base de données 'COLZA': Structure	21
4.1	Enquêtes	25
4.1.1	Exploitants	25
4.1.2	Champs	25
4.1.3	Compo_champs	26
4.1.4	Parcelles	26
4.1.5	Trajets	27
4.1.6	Compo_trajets	27
4.1.7	(Arcs)	28
4.1.8	Segments	28
4.1.9	Silos	28
4.1.10	Transports	29
4.1.11	Traitements	29
4.1.12	Varietes	30
4.1.13	Mel_esp	31
4.1.14	Mel_var	31
4.2	Echantillonnage	32
4.2.1	Populations	32
4.2.2	Plantes	34
4.2.3	Isozymes	34

4.2.4	Acides	35
4.2.5	Glucosinolates	35
4.2.6	Trajets_ech	35
4.2.7	Routes	35
4.2.8	Segments	36
 II Mises à jour de la partie enquête		37
1	Avant de commencer	38
2	Mises à jour de la sous partie Culture	40
2.1	Table Exploitants	40
2.2	Tables Parcelles / Compo_champs / Champs (Mel_esp / Mel_var / Varietes)	40
2.2.1	Lecture de l'enquête	40
2.2.2	Tables Parcelles / Compo_champs	41
2.2.3	Tables Champs / Mel_esp / Mel_var / Varietes	43
2.3	Table Traitements	45
3	Mises à jour de la sous partie Transport	46
3.1	Table Transports	46
3.2	Table Silos	46
3.3	Tables Trajets / Compo_trajets / Segments	47
3.3.1	Table Trajets	47
3.3.2	Table Compo_trajets	48
3.3.3	Table arcs (en cours de remplacement par la table segments)	49
 III Mises à jour de la partie échantillonnage		50
1	Table Populations	51
1.1	Présentation de la nouvelle table Populations	51
1.2	Mise à jour	55
1.3	Intégration dans la base 'COLZA'	60
2	Table Plantes, acides et gls : résultats d'analyses	61
2.1	Traitement des fichiers d'analyses : tables "acides" et "gls"	61
2.2	Traitement des fichiers populations récoltées : table "plantes"	61
2.3	Confrontation des fichiers populations et analyses	62

3	Table de correspondance populations cultivées/parcelles cadastrales	64
3.1	Populations avec numéro cadastral renseigné	64
3.2	Visualisation sous Arcview	64
3.3	Perspectives	65
3.3.1	Procédure postgeo	65
3.3.2	Procédure de traitement des effectifs	65
3.3.3	Saisie des données de terrain	66
3.3.4	Résultats d'analyses	66
IV	Annexes	67
	Installation de PgAccess	68
	Structure des tables:	
	SQL Commandes	70
	Bibliographie	80

fichier source: /home/gharbi/modes_emploi/mode_emploi.tex

Première partie

Présentation de la Base de Données 'COLZA'

Chapitre 1

Introduction

A la veille d'une orientation de plus en plus concrète vers l'exploitation réelle des OGM (Organismes Génétiquement Modifiés), divers organismes tentent d'établir les conséquences possibles d'une telle évolution.

Dans ce contexte, un sujet de thèse, débutée en 1996, a débouché sur une Action Incitative Programmée (AIP), proposée par l'Institut National de la Recherche Agronomique (INRA) impliquant différents domaines d'activité :

- Une considération environnementale développée par le laboratoire d'Ecologie, Systématique et Evolution (ESE) de l'Université ParisXI (Orsay)
- Une intégration informatique, statistique et spatiale dirigée par le laboratoire de Biométrie et Intelligence Artificielle (BIA) de l'Institut National de la Recherche Agronomique (INRA) de Jouy-en-Josas
- Un support analytique (enquêtes auprès d'agriculteurs et analyses biochimiques) assuré par le laboratoire d'analyses d'Orléans du Centre Technique Interprofessionnel des Oléagineux Métropolitains (CETIOM).

L'objectif de cette AIP est d'établir la dissémination de plants de colza, à partir de variétés type, dans une zone agricole déterminée.

Chapitre 2

Base de données ‘COLZA’ : Rôle

Le projet AIP ”Analyse rétrospective sur le devenir des variétés de colza ’0’ et ’00’ dans l’environnement” a nécessité¹ une organisation de l’ensemble des informations recueillies. La base de données ‘COLZA’ assure ainsi la cohérence et l’accès aux informations stockées dans un modèle physique de données particulier (ici un réseau).

Le schéma relationnel adopté (liens entre les différentes tables² déterminant l’organisation de la base) est présenté ci-après, accompagné de la composition globale de chaque table (colonnes³ et clefs⁴ (en italique), primaires (en gras) ou étrangères⁵).

Depuis 1999, l’équipe de terrain bénéficie, outre les cartes IGN (au 1/25000) d’un système GPS, permettant une acquisition des données géographiques avec une précision avoisinant le mètre. Ces données sont utilisées, d’une part numériquement, dans le cadre de la base pour identifier la position d’une population ou d’une parcelle, d’autre part visuellement, grâce au Système d’Information Géographique (SIG) Arcview.

1. arrivée à terme en juillet dernier

2. Ensembles de données relatives à un même concept, aussi appelées *entités*

3. Eléments verticaux d’une table représentant un ensemble de valeurs la définissant, aussi appelées *attributs*

4. Identifiants d’une table, composées d’un ou de plusieurs attributs. Les valeurs d’une clef doivent être uniques au sein d’une table

5. liens avec d’autres tables

Chapitre 3

Base de données 'COLZA' : Connexion

Deux accès à la base de données COLZA sont possibles suivant les circonstances (droits d'accès) et l'utilisation prévue des données de la base.

3.1 Interrogation de la base (contenu des tables, requêtes prédéfinies) : Via internet

- <http://www.inra.fr/bia/J/AB/COLZA/welcome.php>

Le nom de la base:	<input type="text" value="colza"/>
Le nom de la machine:	<input type="text" value="banian"/>
Le nom de l'utilisateur:	<input type="text" value="w3"/>
Le mot de passe:	<input type="password" value=""/>

- Mot de passe géré par les administrateurs base de données
ab@banian.jouy.inra.fr
ka@banian.jouy.inra.fr
- Choix possibles

[voir la structure d'une ou plusieurs tables](#)

[exécuter une fonction pré-définie](#)

[exécuter une requête contenue dans un fichier](#)

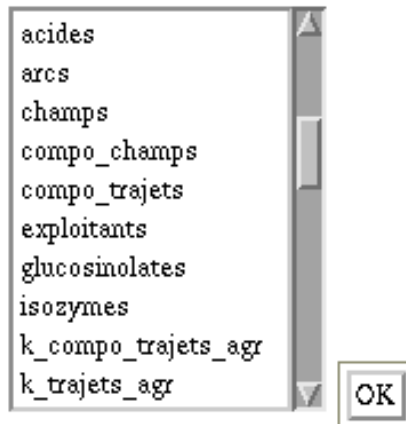
[exécuter une requête pré-définie](#)

[exécuter une requête en langage PostgreSQL](#)

[sélectionner l'intégralité d'une table](#)

voir la structure d'une ou plusieurs tables:

Cliquez sur une ou plusieurs des 47 lignes:



Exemple pour la table Acides

nom	caractéristiques	type	possibilités de clé étrangère	contrôles
id_plante		int2	id_plante=>plantes	
gls		float4		(gls >= 0)

exécuter une requête pré-définie :

[liste détaillée](#)

Attention, requête de sélection uniquement : pas de requête de mise-à-jour ni de création de données.

Cliquez sur une des 47 lignes :

(1.2) surfaces_champs: donne la surface totale emblavée, le nombre de champs ainsi que la surface moyen
(1.3) total_champs_parcelles:
(2.1) nb_exp/annee: nombre d'agriculteurs enquêtes par année
(2.10.a) nb_agric_nb_trait:
(2.10.b) nb_agric_type_trait: nombre d'agric qui applique les différents type de traitement
(2.2) nb_champs/annee: Répartition annuelle des superficies des champs par type variétal
(2.3) varietes_cultivees: donne le nombre de culture d'une variété donnée et la proportion par rapport à 1
(2.4) nb_exp,surf_champs/silo:
(2.6) longueur_route: donne la longueur (en metre) des routes parcourues une année donnée
(2.7) repartition_des_pop: nb de pops hors champs, repousses et jachères

Puis, choisissez le format de sortie : HTML pour exporter

Pour le format 'pour exporter', précisez :

le délimiteur entre valeurs: tabulate par défaut la représentation des valeurs nulles: \N par défaut

Pour stocker les résultats dans un fichier, sauveez la page qui va apparaître en utilisant le menu 'File -> Save as' de votre navigateur.

Exemple pour la requêtes “details_exploitation_nets”

No	Nom	Commentaires	Source sql
108147280	details_exploitation_nets	Donne les différentes cultures d'un exploitant, toutes années confondues. Sert à la réalisation des enquêtes futures : présentation d'une exploitation.	select annee, espece, ville, section, no_cad, champs.surf from compo_champs natural join champs natural join parcelles where id_exp = [parameter " Les cultures de quel exploitant désirez-vous?"] order by ville, annee, espece, section, no_cad

exécuter une fonction pré-définie:

[liste détaillée](#)

Attention, sélection uniquement: pas de mise-à-jour ni de création de données.

Cliquez sur une des 10 lignes:

```
_arcmin: (plpgsql)
_arcmin1: (plpgsql)
_arcsilo: (sql)
_compare_champs: (plpgsql)
_routemin: (plpgsql)
compare_champs: (sql)
dist_plseg: (sql)
meme_ch: (sql)
path2point: (C) Convertir le point d'indice donné en 2ième argument du chemin donné en 1ier argu
path2text: (C) Convertir le point d'indice donné en 2ième argument du chemin donné en 1ier argu
```

Puis, choisissez le format de sortie: HTML pour exporter

Pour le format 'pour exporter', précisez:

le délimiteur entre valeurs: tabulate par défaut la représentation des valeurs nulles: 'N' par défaut

Pour stocker les résultats dans un fichier, sauvez la page qui va apparaître en utilisant le menu 'File -> Save as' de votre butineur.

Exemple pour la fonction "path2point"

No	Nom	Langage	Commentaires	Source
967639	path2point	C	Convertir le point d'indice donné en 2ième argument du chemin donné en 1ier argument en type 'point'.	path2point

exécuter une requête en langage PostgreSQL:

Attention, requête de sélection uniquement: pas de requête de mise-à-jour ni de création de données. N'oubliez pas le point-virgule en fin de commande.

Exemple: sélectionner l'intégralité d'une table.

```
SELECT * FROM plantes ORDER BY id_plante;
```

pour sélectionner l'intégralité de la table plantes, trié par identificateur de plante

Tapez la requête:



Puis, choisissez le format de sortie: HTML pour exporter

Pour le format 'pour exporter', précisez:

le délimiteur entre valeurs: tabulate par défaut la représentation des valeurs nulles: \N par défaut

Pour stocker les résultats dans un fichier, sauvez la page qui va apparaître en utilisant le menu 'File -> Save as' de votre navigateur.

exécuter une requête de sélection en langage PostgreSQL contenue dans un fichier:

Tapez le pathname complet d'un fichier contenant une requête de sélection:



Puis, choisissez le format de sortie: HTML pour exporter

Pour le format 'pour exporter', précisez:

le délimiteur entre valeurs: tabulate par défaut la représentation des valeurs nulles: \N par défaut

Pour stocker les résultats dans un fichier, sauvez la page qui va apparaître en utilisant le menu 'File -> Save as' de votre navigateur.

3.2 Modification de la base (tables et requêtes) : Via banian

- telnet banian.jouy.inra.fr
login :
password:
- Accès (et donc mot de passe) géré par les administrateurs réseaux
monvert@banian.jouy.inra.fr
vr@banian.jouy.inra.fr
- Utilisation des interfaces **PSQL** (brute) ou **PgAccess** (conviviale mais nécessitant l'installation du langage TCL/TK et du programme PgAccess¹ :

3.2.1 Interfaces PSQL et PgAccess

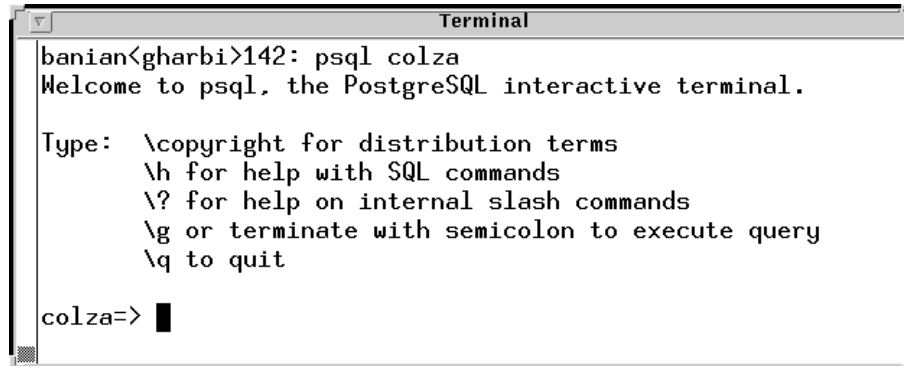
Une description des caractéristiques principales des deux interfaces permet de se rendre compte de leur champ d'application respectifs :

Caractéristiques	PSQL	PgAccess
Convivialité		
Visualisation du résultat des requêtes	Linéaire	Interactive (tables provisoires)
Paramétrage (exécution plus rapide : réponses utiles)	Non	Oui
Limitation de la taille des requêtes	260 caractères	Aucune observée (> 1200 car.)
Facilitation de l'écriture des requêtes	Non	Visual Designer
Structuration		
Définition de clefs (liens nécessaires entre les tables)	Primaires et étrangères	Primaires

1. cf. Annexe 1 'Installation de PgAccess sous Windows'

Interface PSQL

Lancée par la commande : `psql nom_de_la_base`



```
Terminal
banian<gharbi>142: psql colza
Welcome to psql, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms
      \h for help with SQL commands
      \? for help on internal slash commands
      \g or terminate with semicolon to execute query
      \q to quit

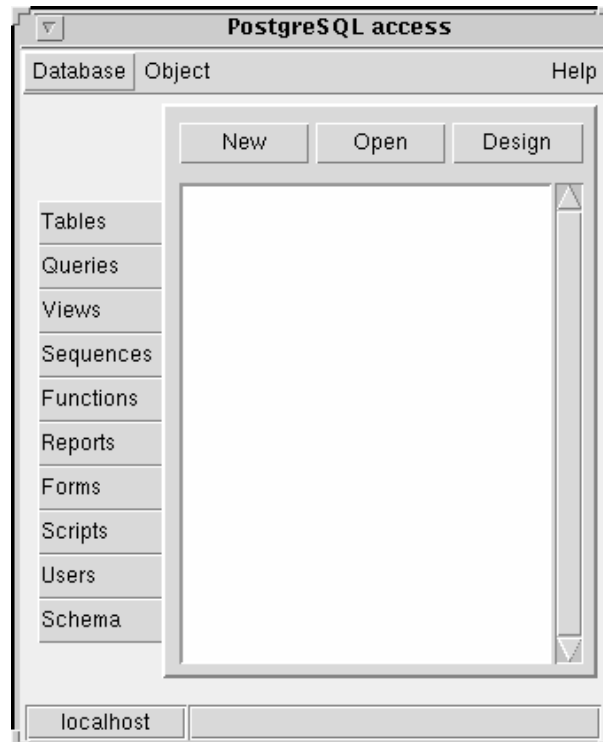
colza=> █
```

Toutes les commandes utilisées sont en langage SQL.

Interface Pgaccess

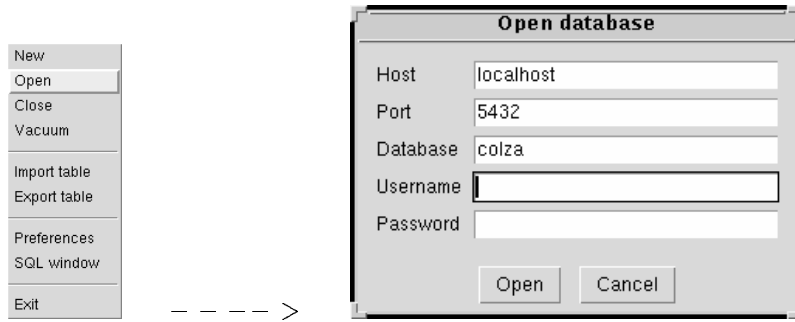
Lancée par la commande : `pgaccess`

Nous traitons ici les cas les plus complets, mais il existe la commande : `pgaccess nom_de_la_base` qui permet une connexion plus rapide à une base donnée, avec username et password par défaut.



La connexion à la base débute par la sélection de l'icône **D**atabase.

1. Connexion

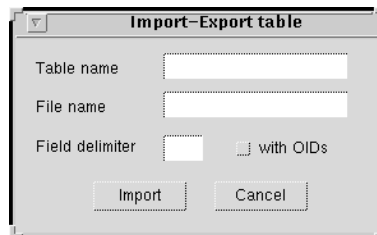


Une fois l'option 'Open' sélectionnée :

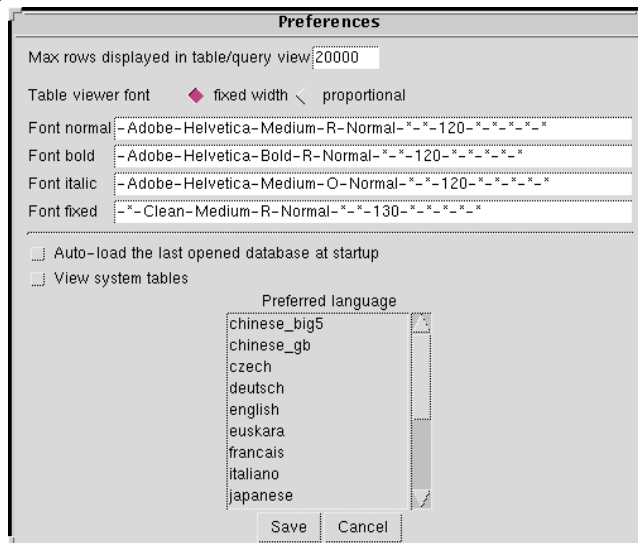
- introduire l'adresse de la machine sur laquelle se trouve la base (banian.jouy.inra.fr)
- préciser le nom de base (colza par exemple)
- donner le nom d'utilisateur et le mot de passe permettant d'accéder à la base.

Après connexion à une base, cet index déroulant offre différents menus :

- Import d'une table dans un fichier ou d'exporter les données d'un fichier dans une table.

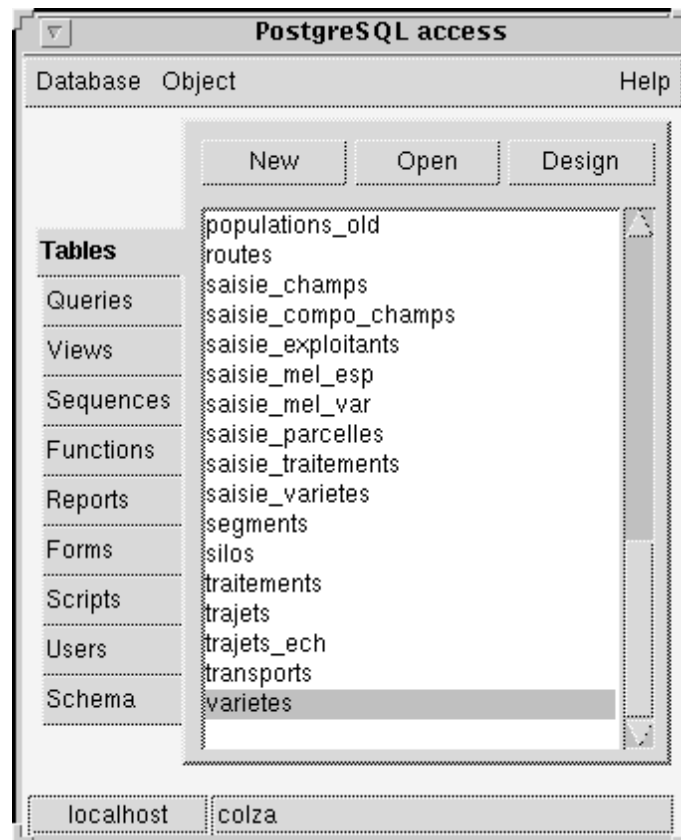


- Preferences du nombre d'enregistrements des tables aux besoins (le temps de chargement des tables étant proportionnel à leur taille: faible pour de petites tables, élevé pour des tables plus grandes).



2. Tables

Présentation générale (Ecran obtenu lors de la connexion directe à la base colza²)



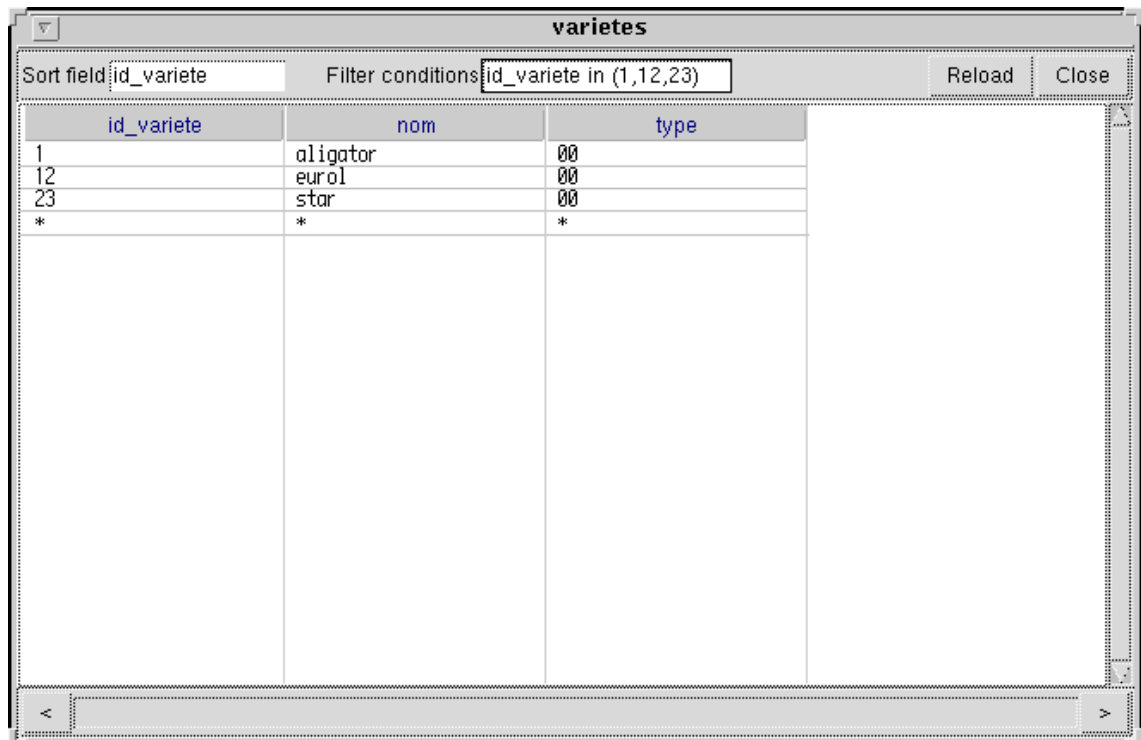
Cette interface permet d'accéder à différents menus : **Tables**, **Queries**, **Functions**...

L'onglet **Tables** permet d'effectuer différentes opérations sur les tables :

- **New** pour créer une nouvelle table (voir plus loin)
- **Open** pour ouvrir la table sélectionné par un simple click (manœuvre remplaçable par un double click sur la table voulue)
- **Design** pour obtenir toutes les informations concernant la table sélectionnée (voir plus loin)

2. par la commande `pgaccess colza`

Pour visualiser le contenu d'une table, il suffit de double-cliquer sur le nom de la table voulue, ici la table **varietes**³:



id_variete	nom	type
1	aligator	00
12	eurol	00
23	star	00
*	*	*

Le contenu de la table peut-être trié suivant les valeurs d'un attribut donné par l'option **Sort field** : ici par id_variete (tri croissant par défaut, sinon ajout de l'option desc à la suite du critère de tri : *'id_variete desc'*).

Certains enregistrements de la table peuvent être sélectionnés selon des critères définis par l'option **Filter conditions** : ici sélection des lignes ayant un id_variete égal à 1, 12 ou 23.

Les tris et sélection peuvent être multiples, séparés par une virgule pour les tris, et par les conjonctions sql classiques (*and, or...*) pour les sélections.

En cas de droits adéquats sur la table (voir plus loin), les données sont modifiables en cliquant sur la case voulue.

De nouvelles conditions sont prises en compte par l'option **Reload**.

La table est fermée classiquement par l'option **Close**.

3. avec quelques options

Création de nouvelles tables

Create new table

Table name: Inherits:

Constraint: check:

field name: type: size:

Default value: field cannot be null primary key

K	field name	type	options
*	id_variete	int2	NOT NULL
	nom	text	NOT NULL
	type	text	NOT NULL

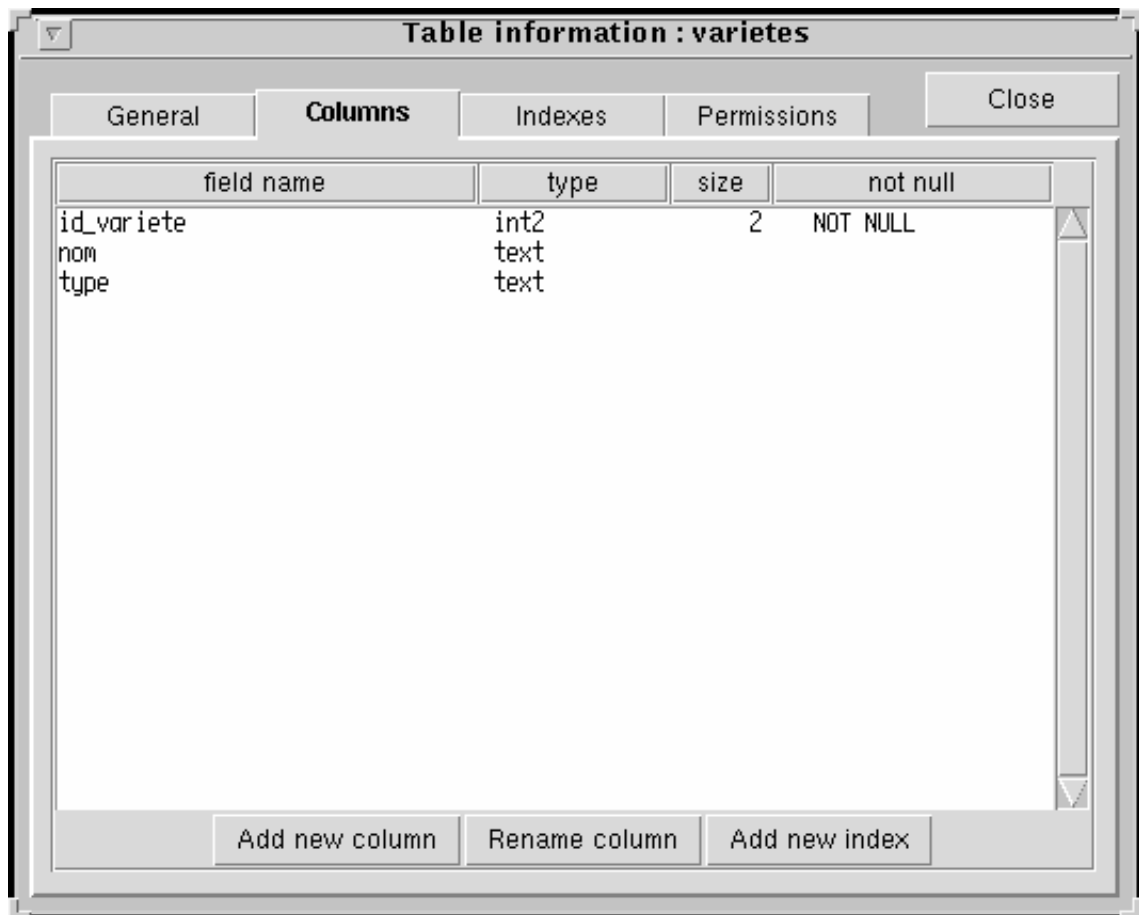
Les différents champs proposés suivent les commandes PSQL : **Table name**, **Constraint...**

Il est important de noter le comportement de la commande **Inherits** : elle permet de récupérer toutes les caractéristiques d'une table préexistante (à condition d'avoir les droits adéquats), mais en gardant un lien avec elle. Ainsi, toute modification dans la table fille se répercute sur la table mère et inversement...Il est donc préférable de recréer une table par les commandes SQL (voir MISES À JOUR DE LA PARTIE ÉCHANTILLONNAGE).

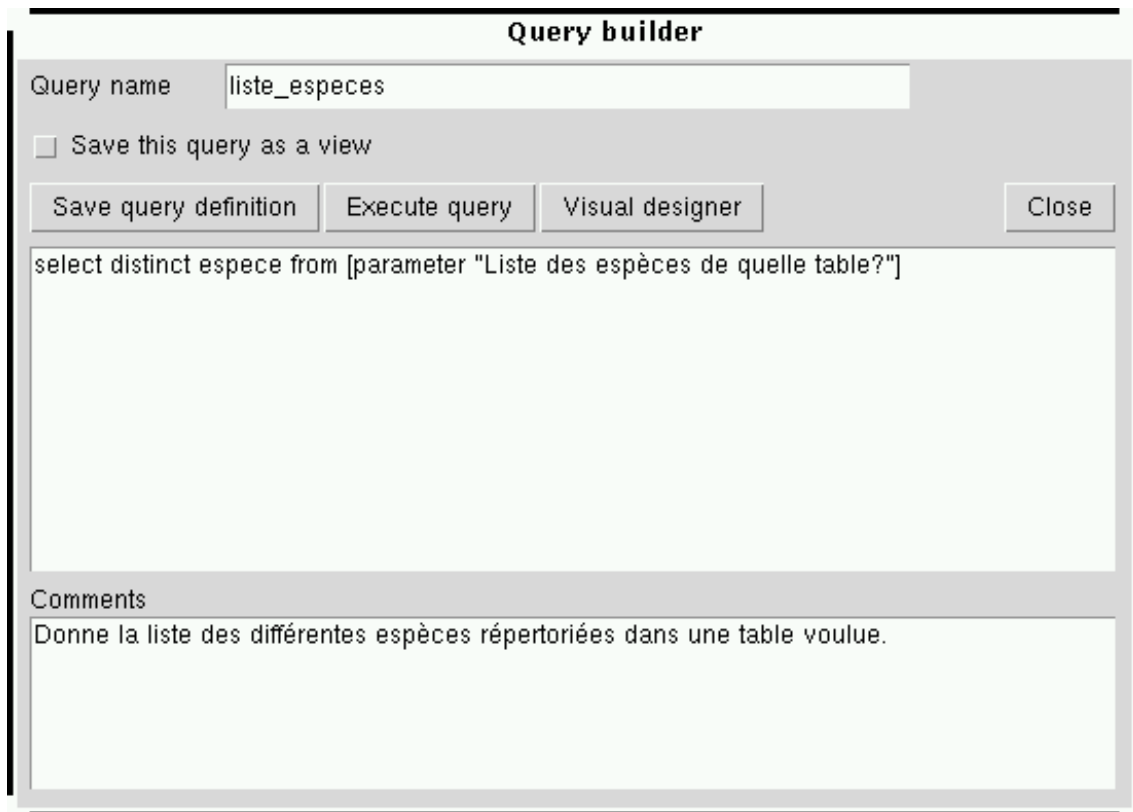
Design : Options

Ce menu donne accès à un descriptif structural de la table sélectionnée :

- **General** : propriétaire, caractéristiques générales...
- **Columns** : attributs, type des attributs, clés...
- **Indexes** : références à d'autres tables.
- **Permissions** : modifiables par le propriétaire de la table (droits de sélection (**select**), de mises à jour (**update**), d'insertion de nouvelles données (**insert**), de changement de statut (**rule**))



3. Requêtes et Fonction



Les requêtes et fonctions possèdent des modes d'utilisation similaires :

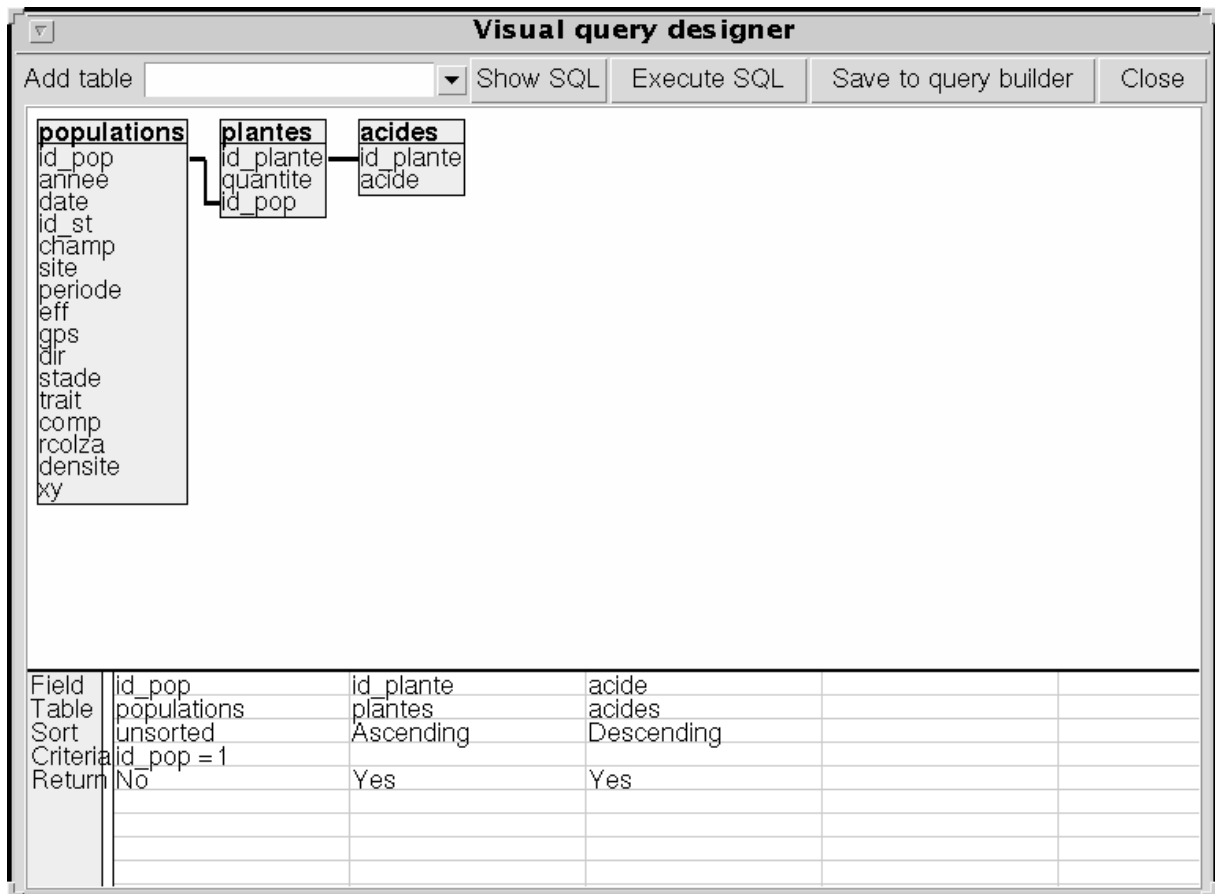
- Attribution d'un nom
- Description (avec paramètres d'entrée et de sortie pour les fonctions)
- Sauvegarde (et exécution pour les requêtes)

Il est conseillé de mettre un maximum de commentaires (action, champs d'action, liens éventuels avec d'autres requêtes ou fonctions, limites...)

L'option **Visual designer** permet de créer des requêtes simples sans utiliser le langage SQL.

L'exemple qui suit présente une requête mettant en relation les tables **acides**, **plantes** et **populations**:

les résultats d'analyse d'acide érucique,
de chaque plante de la population d'id_pop 1,
triés par id_plante croissant et acide décroissant.



Tables : Un menu déroulant permet de choisir les tables concernées par la requête (penser aux tables utilisées uniquement comme lien entre deux autres).

Liens : Les liens entre les tables sont symbolisés par le coulisement des attributs concernés d'une table à une autre.

Sortie : Les attributs à renvoyer sont mentionnés (par coulisement toujours) en bas de la fenêtre, avec leur ordre de tri (**ascendant**, **descendant** ou indifférent **unsorted**).

Conditions : La ligne **criteria** permet de mentionner des conditions sur les attributs.

Exécution et Sauvegarde : La requête ainsi formulée peut être exécutée directement, être visualisée en langage sql, ou encore sauvegardée parmi les autres requêtes.

Cette option ne permet que les requêtes relativement basiques mais est utile, soit dans le cas de début de manipulation du langage sql, soit en cas d'impasse sur une partie d'une requête plus élaborée.

Chapitre 4

Base de données 'COLZA': Structure

La base de données COLZA comprend 20 tables¹, réparties en deux grand ensembles, marquant les deux origines possibles des données qu'elle contient:

- Données d'enquêtes auprès des agriculteurs:
 - * concernant les cultures.
 - * concernant le transport des semences et récoltes.
- Données d'échantillonnage sur le terrain.

Les principales caractéristiques des différentes tables², suivies du schéma relationnel de la base de données présentent la structure des données adoptée, avec mise en italique des clés de chaque table : **primaires** et étrangères:

1. Les noms de tables sont par convention au pluriel, définies dans le fichier */home/ka/2001/psql/colza/SQLCOMM*

2. Evolution du dossier 'Base de données COLZA et Cartographie' par Véronique Emeriau et Katarzyna Adamczyk

Schéma relationnel provisoire de la base

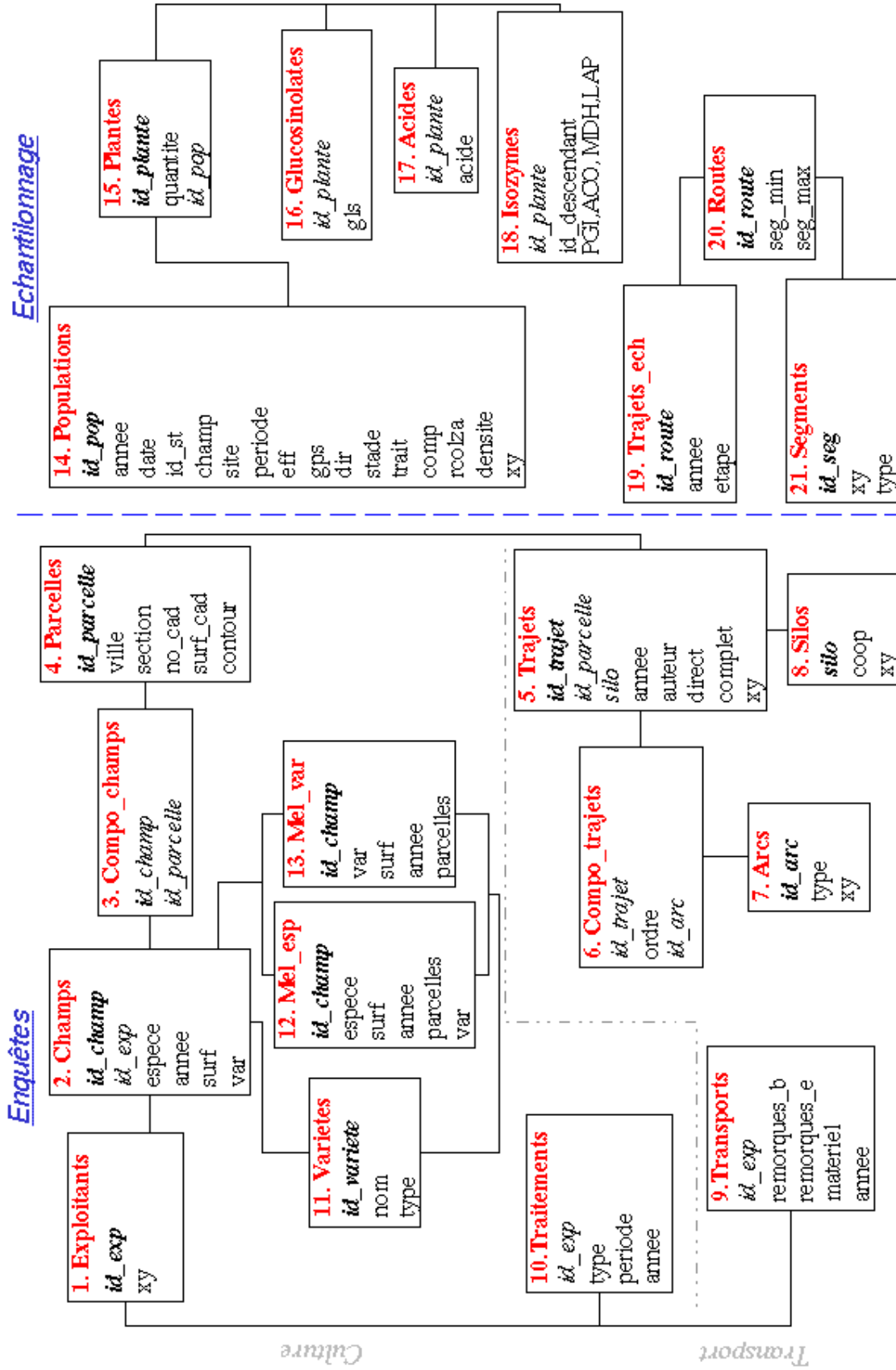
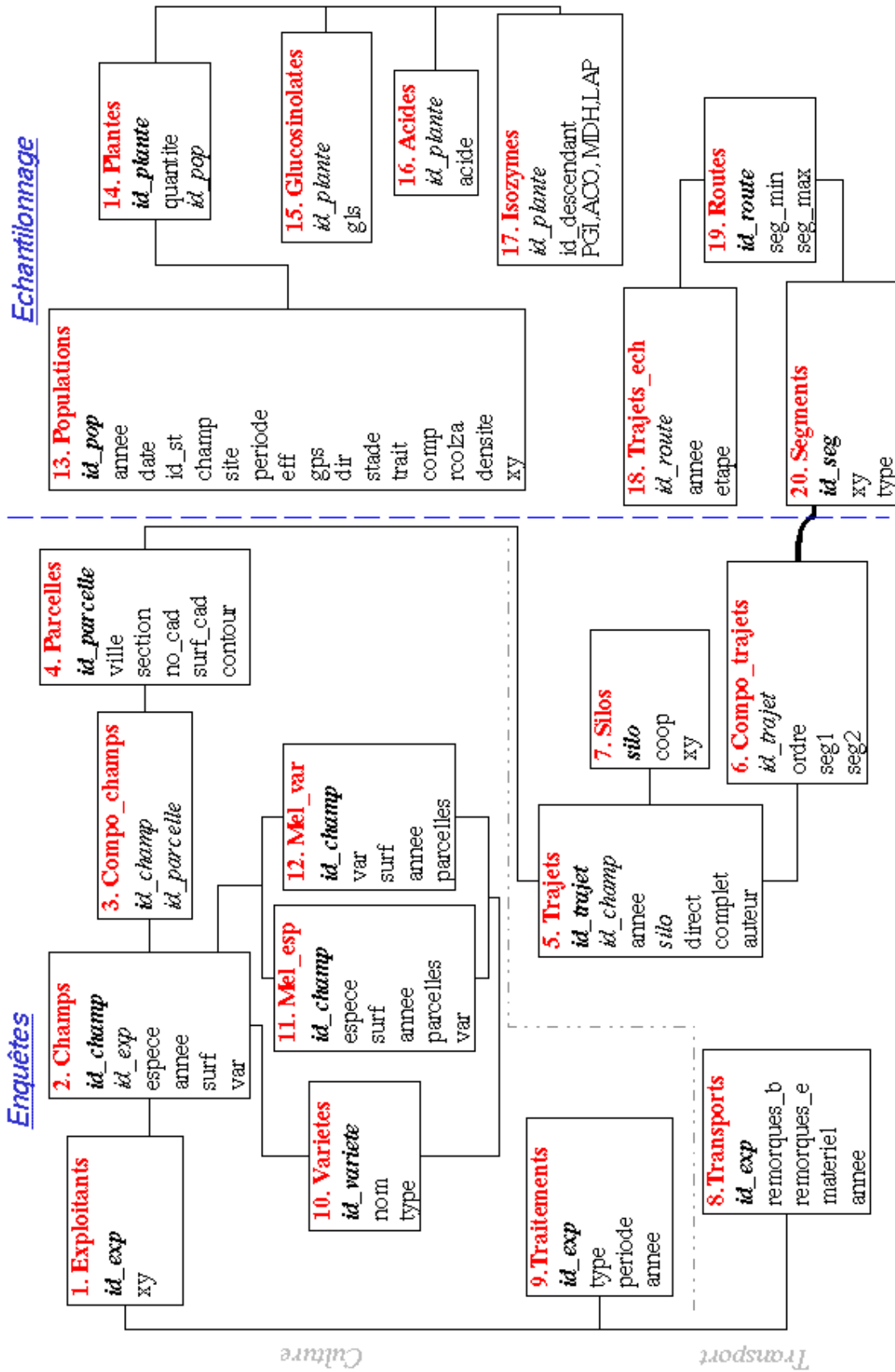


Schéma relationnel de la base (futur)



Tables	Informations délivrées	Clés
<i>Partie enquête</i>		
1.Exploitants	déf. des exploitants enquêtés	id_exp
2.Champs	déf. des cultures	id_champ <i>id_exp</i>
3.Compo_champs	composition en parcelles cadastrales des champs	<i>id_champ</i> <i>id_parcelle</i>
4.Parcelles	déf. des parcelles cadastrales	id_parcelle
5.Trajets	déf. des trajets entre un champ et son silo de récolte	id_trajet <i>id_parcelle</i> <i>id_silo</i>
6.Compo_trajets	composition en arcs de 3m des trajets	<i>id_trajet</i> <i>id_arc</i>
(Arcs)	(déf. d'arcs)	(id_arc)
7.Silos	déf. des silos de stockage	silo
8.Transports	modalités de transports des graines de colza	<i>id_exp</i>
9.Traitements	traitements des bordures appliqués par agriculteur	<i>id_exp</i>
10.Varietes	déf. des variétés de colza semées	id_variete
11.Mel_esp	détail des mélanges d'espèces d'un champ	<i>id_champ</i>
12.Mel_var	détail des mélanges de variétés de colza d'un champ	<i>id_champ</i>
<i>Partie échantillonnage</i>		
13.Populations	déf. des populations repérées (en champ et hors champ)	id_pop
14.Plantes	déf. des plantes analysées	id_plante <i>id_pop</i>
15.Glucosinolates	teneur en gluco des plantes analysées	<i>id_plante</i>
16.Acides	teneur en érucique des plantes analysées	<i>id_plante</i>
17.Isozymes	teneur en différents isozymes des descendants des plantes analysées	<i>id_plante</i>
18.Trajets_ech	déf. des trajets parcourus	id_route
19.Routes	déf. des routes parcourues	id_route
<i>Table partagée</i>		
20.Segments	déf. des segments de routes de 3 m.	id_seg

4.1 Enquêtes

Toutes les informations contenues dans cette partie ont pour origine les enquêtes réalisées auprès des agriculteurs.

Les comptes-rendus d'enquêtes sont ensuite traités soit par la personne les ayant réalisées, soit par un tiers personne.

Dans le second cas, certaines données peuvent ne pas être suffisamment claires ou précises pour être rentrées dans la base de données (tout particulièrement dans le cadre de la mise à jour réalisée en 2002, reprenant toutes les données d'enquêtes depuis 1998).

La base de données contient donc un maximum d'informations mais n'est pas exhaustive.

4.1.1 Exploitants

id_exp	Identifiant dun exploitant
xy	Coordonnées géographiques du site d'exploitation

Représente la liste des exploitants enquêtés, accompagnés des coordonnées de leur site d'exploitation.

Les informations de l'ancienne table **exploitants** est maintenant répartie sur trois tables : **exploitants**, **transports** et **champs**.

4.1.2 Champs

id_champ	Identifiant d'un champ
<i>id_exp</i>	Identifiant de l'exploitant auquel appartient ce champ
espece	Nom symbolique de l'espèce cultivée sur ce champ (dont mélange)
annee	Année de culture de ce champ
surf	Surface du champ emblavée (totale pour un mélange)
var	Identifiant d'une variété de colza (0 pour un mélange)

Regroupe toutes les cultures (colza et autres³) déclarées par les agriculteurs enquêtés, et vérifiées (cohérentes avec le parcellaire à disposition).

Un champ correspond à un agencement de parcelles voisines utilisé par un agriculteur, une année, présentant un minimum d'espèces différentes (minimalisation des mélanges).

Complétée par les tables **varietes**, **mel_esp** et **mel_var** pour définir les variétés de colza semées et les mélanges éventuels d'espèces et/ou de variétés de colza.

3. voir le résultat de la requête 'liste des especes'. A titre d'exemple, on obtiendrait actuellement: ble, ble_d, ble_hybride, ble_o, ble_t, colza, escourgeon, fetuque, feverolles, jach, jach_colza, jach_moutarde, jach_n, jach_seigle, jach_trefle, luzerne, mache, mais, melange, millet, orge, orge_h, orge_p, pois, prairie, seigle, tournesol, trefle, triticales, vigne.

La modalité 'melange' de l'attribut *espece* permet de ne définir chaque champ d'un agriculteur qu'une fois par année.

4.1.3 Compo_champs

<i>id_champ</i>	Indentifiant du champ décomposé
<i>id_parcelle</i>	Identifiant de la parcelle faisant partie du champ

Fournit la relation (univoque) entre un champ et les parcelles qui le composent.

4.1.4 Parcelles

id_parcelle	Identifiant d'une parcelle
<i>ville</i>	Commune à laquelle appartient la parcelle
<i>section</i>	Section cadastrale de la parcelle
<i>no_cad</i>	Numéro cadastral de la parcelle
<i>surf_cad</i>	Surface cadastrale de la parcelle
<i>contour</i>	Cordonnées géographiques des contours de la parcelle

Définit les parcelles rencontrées lors des enquêtes (quelques parcelles ne sont pas attribuées du fait de la dernière mise à jour), telles quelles sont distinguables sur les cartes et cadastres disponibles⁴.

L'ensemble (*ville*⁵, *section*, *numéro cadastral*) définit une parcelle de manière unique.

4. Certaines parcelles hors zone ou non visibles sur les cadastres sont considérées aussi, selon la fiabilité des données

5. Communes actuellement répertoriées: Baigneaux, Boisseaux, Champigny_en_beauce, Conan, Coulommiers_la_tour, Crucheray, Faye, Perigny, Rhodon, Selommes, Ste_Gemmes, Tourailles, Villefrancoeur, Villemardy, Villeromain, Villetrun.

4.1.5 Trajets

id_trajet	Identifiant d'un trajet
<i>id_champ</i>	Identifiant du champ à l'origine du trajet
<i>silo</i>	Silo où arrive le trajet (voir table silo)
auteur	Personne à l'origine du tracé du trajet (agriculteur ou supposé par l'enquêteur)
direct	Fractionnement du trajet (booléen: direct au silo ou passage par la ferme)
complet	Exhaustivité du trajet (booléen: complet ou partiellement hors zone)
xy	Coordonnées géographiques du trajet

Regroupe les trajets utilisés par un agriculteur pour transporter la récolte d'un champ au silo de récolte.

Avant 2000, un trajet est supposé identique pour toutes les années renseignées (une carte pour l'ensemble de la rétrospective de 1999).

Les trajets parcourus par les agriculteurs sont, dans la majeure partie des cas, les mêmes d'une année sur l'autre. Les cas exceptionnels peuvent cependant présenter deux trajets similaires pour un même champ, selon l'année d'enquête.

Un trajet peut être dessiné sur une carte par l'agriculteur ou simplement supposé par l'enquêteur (information donnée par l'attribut *auteur*: agr ou sup). Dans le cas où celui-ci est incomplet, seules les parties de trajet exactes sont considérées.

L'ensemble des trajets est déclaré comme direct au silo ou indirect (passage par la ferme par exemple).

Les coordonnées d'un trajet sont récupérées à partir de celles des segments qui le composent.

On retrouve le parallèle de cette table dans la partie échantillonnage: **Trajets_ech**.

4.1.6 Compo_trajets

<i>id_trajet</i>	Identifiant du trajet décomposé
ordre	Numéro d'ordre des intervalles de segments composant le trajet
<i>seg1</i>	Identifiant du premier segment de l'intervalle
<i>seg2</i>	Identifiant du dernier segment de l'intervalle

Chaque trajet est formé de segments d'une longueur de 3m maximum.

De nombreux trajets empruntent les mêmes routes (routes principales notamment): une portion de route fréquemment utilisée sera donc, en général, caractérisée par les mêmes intervalles des segments.

On retrouve le parallèle de cette table dans la partie échantillonnage: **Routes**.

4.1.7 (Arcs)

id_arc	Identifiant d'un arc
type	Nature de la voie à laquelle appartient l'arc (chemin de terre, route, lien avec la parcelle)

Définit les arcs: coordonnées géographiques et nature de la voie à laquelle il appartient (R: route, T: chemin de terre, lien_parcc: lien entre le trajet et la parcelle).

Cette table n'est que provisoire: la table **arcs** sera supprimée dès que tous les trajets seront définis par la table **segments** (mise à jour en cours).

4.1.8 Segments

id_seg	Identifiant d'un segment de 3m
xy	Coordonnées géographiques du segment
type	Type de la voie

La nature des voies (goudron, voie simple, voie double...) donne une indication sur les possibilités de propagation des graines (matériau, épaisseur, traitement possible etc...).

Table commune aux deux parties, enquêtes et échantillonnage.

4.1.9 Silos

silos	Commune à laquelle appartient le silo de récolte
coop	Coopérative propriétaire du silo
xy	Coordonnées géographiques du silo (centre)

Décrit les différents silos⁶ de récolte mentionnés dans les enquêtes, identifiés par le nom de la ville à laquelle ils appartiennent.

Ancienne table stockage (simplification par retrait de l'attribut *id_lieu*, redondant avec *lieu*, maintenant nommé *silos*).

6. Silos actuellement répertoriés: Averdon, Champigny (avec coord.), Coulommiers (avec coord.), Ferme (avec coord.), Herbault, Oucques, Selommes (avec coord.), Tourailles, Tours, Villejoin.

4.1.10 Transports

<i>id_exp</i>	Identifiant de l'exploitant
remorques_b	Bâchage des remorques
remorques_e	Etat d'étanchéité des remorques
materiel	Appartenance du matériel (personnel ou partagé)
annee	Année de référence

Indique, pour chaque année, les modalités de transports déclarées lors des enquêtes: bâchage (B: remorques bâchées, NB: remorques non bâchées, BNB: remorques bâchées ou non bâchées) et étanchéité (C: correcte, CTB: correcte et très bonne, TB: très bonne) des remorques, provenance du matériel (P: personnel, E: partagé au sein d'une entreprise, mixte: PE).

Ces informations ne sont disponibles que jusqu'en 1999 (compris).

4.1.11 Traitements

<i>id_exp</i>	Identifiant de l'exploitant
type	Type d'entretien (mécanique, chimique, par fauche, et combinaisons)
periode	Période(s) mensuelle(s) ou mi mensuelle(s) du type de traitement
annee	Année de référence du type de traitement

Regroupe les différents traitements appliqués par les agriculteurs sur les bordures de leurs champs.

Un agriculteur peut utiliser un type de traitement (broyage - B -, chimie - C -, fauche - F -, combinaisons - MB ou MF pour le moment -, sans traitement - rien) à différentes périodes de l'année. La précision éventuelle de ces périodes, listées pour un type de traitement donné, ne dépasse pas la quinzaine de jours (mois - fevrier, mars, avril, mai, juin, aout, juillet - ou mi mois - deb, mi, fin). Les périodes non précisées sont marquées par un point d'interrogation.

L'indication 'rien' signifie qu'aucun traitement n'est appliqué (à distinguer de l'absence de données sur le traitement effectué par un agriculteur une année en particulier).

4.1.12 Varietes

id_variete	Identifiant de la variété
nom	Nom commercial de la variété
type	Type de variété

Cette table permet de visualiser les différentes variétés de colza rencontrées lors des enquêtes, accompagnées de leur type variétal, symbolisé par une association binaire correspondant aux teneurs relatives⁷ des graines de colza en glucosinolates et acide érucique (0 pour faible et 1 pour élevée):

Teneur en glucosinolates	Teneur en acide érucique	Type variétal	Exemple
faible	faible	00	synergy ("double zéro")
faible	forte	01	bienvenue ("simple zéro")
forte	faible	10	zéroca ("érucique")
forte	forte	11	gaspard ("érucique")

Une requête permet le parallèle dans la partie échantillonnage: les types variétaux des plantes analysées (biochimiquement) suivent les mêmes conventions (avec quelques modalités supplémentaires pour les mesures intermédiaires de dosages).

Seuls les noms sous lesquels les semences sont commercialisées sont considérés (pas de colza "printemps" ou de numéro de série).

Dans les autres tables, les noms de variétés sont manipulés au moyen de leur identifiant, dépendant uniquement de leur ordre d'apparition dans les enquêtes.

7. Appréciations des teneurs:

- en glucosinolates:

faible $\leq 40 \mu\text{mol/g}$

forte $\geq 60 \mu\text{mol/g}$

- en acide érucique:

faible $\leq 5\%$ des acides gras totaux

forte $\geq 30\%$ des acides gras totaux

4.1.13 Mel_esp

<i>id_champ</i>	Identifiant du champ dont le mélange est détaillé
<i>espece</i>	Nom de l'espèce cultivée sur le champ
<i>surf</i>	Surface emblavée de la culture
<i>annee</i>	Année concernée par le mélange
<i>parcelles</i>	Parcelles emblavées au moins en partie avec cette culture (si détaillées)
<i>var</i>	Variété de colza composant le mélange (0 si mélange de variétés de colza)

Les mélanges⁸ de la table **champs** (cas où l'attribut *espece* est indiqué 'melange') impliquant différentes espèces sont ici explicités, par espèce.

Cette table permet de considérer les surfaces respectives de chaque culture (surtout si la surface totale n'est pas disponible), ainsi que leur répartition au niveau parcellaire, lorsque celle-ci est détaillée.

Un mélange d'espèces impliquant un mélange de variétés de colza sera détaillé dans la table spécifique **mel_var**.

4.1.14 Mel_var

<i>id_champ</i>	Identifiant du champ dont le mélange en colza est détaillé
<i>var</i>	Variété de colza composant le mélange
<i>surf</i>	Surface emblavée de la culture
<i>annee</i>	Année concernée par le mélange
<i>parcelles</i>	Parcelles emblavées au moins en partie avec cette variété (si détaillées)

Détaille les mélanges de variétés de colza indiqués, soit dans la table **champs**, soit dans la table **mel_esp** (cas où les attributs *espece* et *variete* sont indiqués 'colza' et '0' respectivement), par variété.

Sœur de la table **mel_esp** (avec un attribut *espece* = 'colza' par défaut), elle présente les mêmes caractéristiques.

8. Un mélange est défini dès que plus d'une espèce est présente sur l'une des parcelles d'un champ. Voir MISE À JOUR DE LA PARTIE ENQUÊTE pour plus de détails et exemples.

4.2 Echantillonnage

Cette partie contient toutes les données issues des échantillonnages sur le terrain.

Selon les contraintes des tables définies, certaines données incomplètes sont gardées dans des fichiers spécifiques mais non considérées dans la base de données (données non exploitables, populations analysées mais sans coordonnée géographique par exemple).

4.2.1 Populations

id_pop	Identifiant d'une population
annee	Année de repérage de la population
date	Date de repérage de la population (JJ-MM-AA)
id_station	Identifiant de l'emplacement repéré
champ	Nature de la population (booléen: champ ou hors champ)
site	Localisation des populations hors champ
periode	Période de repérage (floraison ou récolte)
eff	Effectif de la population (≤ 50 exact, $50 \leq \text{estimé} \leq 10.000$) (en récolte: $100.000 \rightarrow \geq 10$ plantes mais sans précision supplémentaire)
gps	Numéro de gps référant à l'enregistrement des coordonnées de la population
dir	Direction du véhicule lors de l'enregistrement des coordonnées de la population
stade	Stade de développement général de la population
trait	Traitement apparent de l'environnement de la population
comp	Existence (booléen) d'une compétition apparente ou possible des repousses avec des mauvaises herbes
rcolza	Colza visiblement distinct (booléen) du champ qu'il borde (pour les populations en bordure de champ)
densite	Densité évaluée des plantes de la population (isolées, groupées, en touffes, denses)
xy	Coordonnées géographiques de la population

Définit toute population de colza repérée, accompagnée de ses coordonnées géographiques.

Cette table reprend le maximum d'informations récupérées sur le terrain au cours des années (évolution de la structure de la table, chaque année, selon les fichiers de données).

Une population est désignée dans la table **populations** par son **id_pop**. Cependant, cet identifiant unique pour une population, est obtenu après plusieurs manipulations des fichiers source.

Ainsi, depuis 2000, l'attribut `gps` est venu s'ajouter à l'attribut `id_station` pour définir une population, selon le cahier de terrain tenu lors de l'échantillonnage.

L'identifiant de station correspond à une numérotation journalière (voir semi journalière) des endroits (ou stations) repérés. Cette donnée est pour le moment facultative (l'ensemble (*annee*, *periode*, *gps*) définit actuellement une population), mais s'avèrera utile si l'ensemble des attributs (*annee*, *periode*, *gps*) ne suffit plus pour définir une population (cest-à-dire reprise de la numérotation `gps` lors de la floraison ou de la récolte d'une année).

Les populations cultivées et hors champs sont désormais distinguées par l'attribut `champ` (vrai ou faux), avec un détail des populations anciennement dites spontanées grâce à l'attribut `site`⁹.

La période de floraison (`fl`) correspond suivant les années aux mois mars et/ou avril, celle de récolte (`rec`) aux mois juin et/ou juillet.

En période de récolte, les échantillonnages destinés aux analyses (biochimiques et autres) comptent 10 plantes.

Leur effectif, s'il n'est davantage précisé dans les fichiers source, est symbolisé par un nombre remarquable de 100.000 plantes. Cette valeur représente donc un effectif de 10 plantes au minimum:

Effectifs observés	Prélèvement	Effectifs dans les fichiers	Effectifs dans la base
≤10	Toutes les plantes	Effectif exact	Effectif observé
≥10	10 plantes	Effectif exact	Effectif observé
≥10	10 plantes minimum	Aucune donnée	Effectif symbolique: 100.000

Les relevés `gps` des populations cultivées peuvent être réalisés, selon les années et les conditions (climatiques et de densité), au sein ou en bordure interne des champs.

L'attribut `dir`, disponible depuis 2000, permet de discriminer le côté de la route auquel appartient une population en bordure de champ (et donc le champ qu'elle borde ou dont elle est issue), à partir de la direction de la voiture (N, S, E, O). Cette information sera bientôt remplacée ou complétée par la saisie du `no_cad` d'une des parcelles du champ en question (en cours).

Une compétition interspécifique supposée constitue une composante du développement des plantes d'une population, de même que la densité des populations (*i*: isolées, *g*: groupées, *t*: en touffes, *d*: denses).

9. Sites actuellement catalogués: bordure, jach, et repousses en champ : `c`, `c_bettes`, `c_ble`, `c_crucif`, `c_feverolle`, `c_labour`, `c_mais`, `c_orge`, `c_pois`, `c_tournesol`, `c_trefle`.

Des aspects visiblement distincts d'un colza bordant un champ, fourni par l'attribut 'rcolza' et des colzas de ce champ peuvent apporter une information sur l'origine de ce colza (dans le contexte d'analyses génétiques notamment).

4.2.2 Plantes

id_plante	Identifiant d'une plante d'une population
quantite	Quantité de plantes par analyse (1 par défaut, plus si une plante représente une quantité trop faible pour être analysée)
<i>id_pop</i>	Identifiant de la population à laquelle appartient la plante

Seules les plantes possédant un résultat d'analyse en teneur d'acide et/ou de glucosinolates sont répertoriées. La numérotation des id_plante est réalisée après obtention des résultats d'analyse (retrait des plantes 'virtuelles' sans aucun résultat)

Une population de 10 plantes peut alors trouver des correspondances pour 10 plantes au maximum (non analysées, problèmes d'analyses...)

4.2.3 Isozymes

<i>id_plante</i>	Identifiant d'une plante d'une population
id_descendant	Identifiant du descendant de la plante
PGI	Profil du marqueur isoenzymatique PGI
Aco	Profil du marqueur isoenzymatique Aco
MDH	Profil du marqueur isoenzymatique MDH
LAP	Profil du marqueur isoenzymatique LAP

Les analyses isoenzymatiques¹⁰ sont réalisées sur 3 à 5 descendants issus de germination en serre des graines échantillonnées (allèles éventuellement chevauchant à mettre en évidence).

Le profil donné par l'ensemble de ces quatre loci isoenzymatiques permettrait de descendre dans de nombreux cas au niveau variétal. Ces données pourront être approfondies plus tard par les profils génétiques des graines échantillonnées (coûteux).

10. Protocole de Chèvre et *al.*, cf thèse de Fabrice Pessel

4.2.4 Acides

<i>id_plante</i>	Identifiant d'une plante d'une population
acide	Teneur en acide érucique en % d'acides gras totaux

Les teneurs en acide érucique sont données en pourcentage d'acides gras totaux¹¹.
Seules les populations possédant un résultat d'analyse en acide érucique sont répertoriées.

4.2.5 Glucosinolates

<i>id_plante</i>	Identifiant d'une plante d'une population
gls	Teneur en glucosinolates totaux en $\mu\text{mol/g}$ de graine

De même que pour les teneurs en acide érucique, les teneurs en glucosinolates totaux sont données, en $\mu\text{mol/g}$ de graine¹².

Seules les populations possédant un résultat d'analyse en glucosinolates sont répertoriées.

4.2.6 Trajets_ech

id_route	Identifiant d'une route parcourue
annee	Année de parcours de la route
etape	Periode de parcours de la route (floraison ou récolte)

Liste des routes parcourues lors de l'échantillonnage, en floraison et/ou en récolte.
Parallèle de la table **Trajets**.

4.2.7 Routes

id_route	Identifiant d'une route parcourue lors de l'échantillonnage
seg_min	Premier segment de 3m constituant la route parcourue
seg_max	Dernier segment de 3m constituant la route parcourue

Les routes sont fragmentées automatiquement en segments de 3m.
Parallèle de la table **Compo_trajets** de la partie enquête.

11. selon une méthode d'analyse non mentionnée dans la base de données (efficacité des différentes méthodes considérée équivalente du point de vue du rendement substantiel)

12. selon une méthode HPLC (High Pressure Liquid Chromatography)

4.2.8 Segments

id_seg	Identifiant d'un segment de 3m
xy	Coordonnées géographiques du segment
type	Type de la voie

La nature des voies (goudron, voie simple, voie double...) donne une indication sur les possibilités de propagation des graines (matériau, épaisseur, traitement possible etc...).

Table commune aux deux parties, enquêtes et échantillonnage.

Deuxième partie

Mises à jour de la partie enquête

Chapitre 1

Avant de commencer

Au vu des diverses erreurs possibles lors des saisies, les mises à jour sont réalisées dans des copies¹ des tables concernées (avec conservation des contraintes), qui seront ajoutées aux originales après vérifications.

Pour une saisie optimale, il est nécessaire d'avoir à disposition différents éléments :

- ★ 1 table **champs** à mettre à jour, triée par `id_champ` croissant (par défaut).
- ★ 2 tables **compo_champs** :
 - une à mettre à jour, triée par `id_champ`
 - une pour la recherche d'appartenance d'une parcelle à d'anciens champs, en utilisant 'Filter conditions' : `id_parcelle = ?`.
- ★ 2 tables **parcelles** :
 - une à mettre à jour, triée par `id_parcelle`.
 - une pour la recherche de l'identifiant d'une parcelle donnée, en utilisant 'Filter conditions' : `ville = '?' and section = '?' and no_cad = '?'`.
- ★ 1 table **varietes** réduite car, une année donnée, les mêmes varietes reviennent rapidement, en utilisant 'Sort field' : `id_variete` pour la saisie d'une nouvelle variété, ou 'Sort field' : `nom` pour la recherche de l'`id_variete` correspondant à une variété donnée.
- ★ 1 table **mel_esp** et 1 table **mel_var** réduites car utilisées uniquement en cas de mélanges d'espèces ou de variétés.
- ★ 1 table **traitements** à mettre à jour.
- ★ 1 requête adéquate qui fournisse les cultures antérieures d'un agriculteur (vérifications plus simples et plus rapides).

Les tables **mel_esp**, **mel_var** et **traitements** ne nécessitent pas de tri particulier avant ajout car toutes les nouvelles données de ces tables comportent une année nouvelle, et ne peuvent donc pas être identiques aux anciennes données.

1. La commande la plus simple pour créer des copies de tables modifiables et avec contraintes est:
Create table `nom_copie_table` as (select * from `nom_original_table` where `cle_table` isnull)

La marche à suivre restera la même **pour tout nouvel enregistrement** dans une table : les données **doivent être triées par l'attribut clé** de la table et les nouvelles données seront mises à la suite. Ceci permet d'éviter l'apparition de doublons.

Une série de requêtes de contrôles est par ailleurs disponible, donnant le détail des données redondantes ou manquantes.

La saisie devrait être simplifiée par la possession, lors des enquêtes, du bilan de l'exploitation de chaque agriculteur au cours des années précédentes (utilisation de la requête *details_exploiation_nets* qui liste les cultures antérieures d'un agriculteur, encore en cours de modifications).

Chapitre 2

Mises à jour de la sous partie Culture

2.1 Table Exploitants

La table exploitants est à compléter¹ si de nouveaux exploitants sont enquêtés (avec si possible les coordonnées de leur exploitation, acquises par un logiciel adapté).

2.2 Tables Parcelles / Compo_champs / Champs (Mel_esp / Mel_var / Varietes)

2.2.1 Lecture de l'enquête

Depuis 1999, le relevé des cultures d'un agriculteur est accompagné d'une carte (la plus précise possible) qui permet la localisation de ses différentes cultures.

La première étape est de comparer le compte-rendu d'enquête et la carte annexe afin de :

- déterminer la cohérence des deux types de données (particulièrement si l'enquête est réalisée par une tierce personne).
- mettre en évidence les regroupements éventuels de parcelles (si non réalisés par l'agriculteur).
- mettre en évidence d'éventuels mélanges d'espèces sur un même regroupement de parcelles (à considérer ensembles dans la table **champs**).

Les vrais mélanges présentent plus d'une espèce sur une même parcelle :



1. Rappel: **Tout nouvel enregistrement** doit posséder un **identifiant nouveau** et consécutif aux précédents; toute mise à jour doit donc être précédée par un tri par l'identifiant clé de la table (voir BASE DE DONNÉES 'COLZA': STRUCTURE).

Dans le cas de parcelles à la limite de deux communes, il faut bien vérifier les communes auxquelles elles sont censées appartenir (possibilités de confusion).

Le parcellaire est en priorité respecté, mais il n'est actuellement pas complet; au niveau de ces zones imprécises², la fiabilité de l'ensemble de l'enquête détermine l'information à suivre (la carte ou l'enquête).

Un élément pris en compte dernièrement est le remembrement des parcelles cadastrales (une parcelle peut avoir un nom une année donnée, et un autre nom une année suivant le remembrement).

Des champs partagés par deux exploitants sont mentionnés deux fois dans la table **champs** mais idéalement une seule fois dans les autres tables (ils déclarent en général les mêmes répartitions de cultures...).

2.2.2 Tables Parcelles / Compo_champs

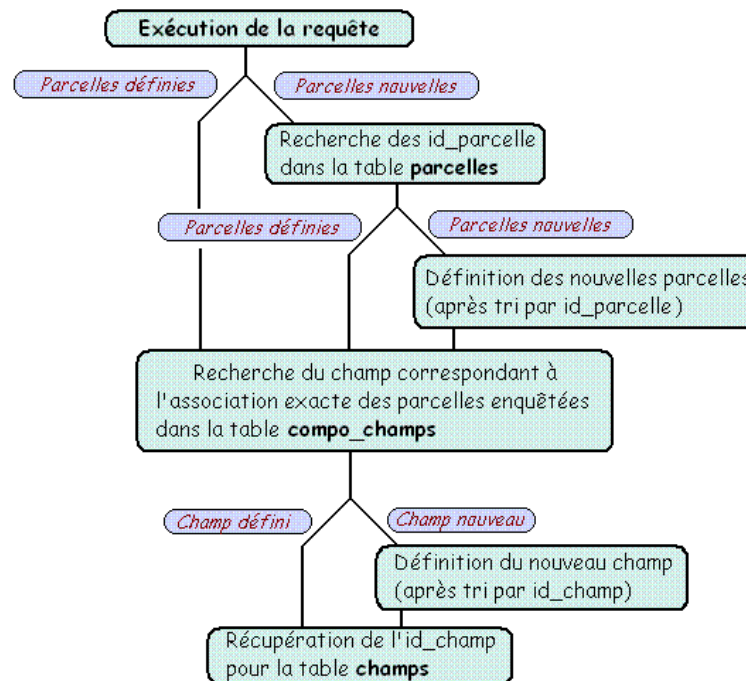
La requête '*details_exploitation_bruts*' donne le relevé des exploitations antérieures d'un agriculteur, afin de faciliter les mises à jour annuelles (les exploitations étant relativement similaires d'une année à l'autre):

Exemple partiel pour l'agriculteur 1:

id_champ	annee	espece	ville	section	no_cad	id_parcelle
25	2000	orge	Coulommiers_la_tour	ZE	53	243
25	2000	orge	Coulommiers_la_tour	ZE	54	244
25	2001	jach	Coulommiers_la_tour	ZE	53	243
25	2001	jach	Coulommiers_la_tour	ZE	54	244
30	2000	jach_n	Perigny	ZB	18	263
30	2000	jach_n	Perigny	ZB	19	264

2. Ces parcelles sont répertoriées dans un fichier à part dans une optique de vérification en bloc : */home/gharbi/misc_a_jour/reprise_enquetes/parcelles_a_verifier.xls*

A partir du résultat de la requête, différents choix sont possibles:



En résumé: un ensemble de parcelles (définies dans la table **parcelles**), doit composer un champ (défini dans **compo_champs**). Les cultures de ce champ peuvent alors être précisées dans la table **champs**.

La vérification des compositions des champs proposés est facilitée par la requête *details_champ* qui fournit la composition en parcelles correspondant à un id_champ donné.

Dans cette partie de la mise à jour, la démarche reste toujours la même: rechercher une donnée dans les tables, la créer³, selon les modalités préexistantes, si elle n'y existe pas encore.

Pour vérification, la requête *champ_db* recherche les compositions de champ identiques⁴. Les champs redondants seront alors remplacés dans la table **champs** puis retirés de la table **compo_champs**.

3. Rappel: **Tout nouvel enregistrement** doit posséder un **identifiant nouveau** et consécutif aux précédents; toute mise à jour doit donc être précédée par un tri par l'identifiant clé de la table (voir BASE DE DONNÉES 'COLZA': STRUCTURE).

4. Cette requête tient compte des tables originales et des copies utilisées pour la saisie

2.2.3 Tables **Champs** / **Mel_esp** / **Mel_var** / **Varietes**

- Cas général : 1 espèce sur le champ prédéfini
 - * Attribut **id_champ** (**Champs** / **Mel_esp** / **Mel_var**)
Il s'agit de l'**id_champ** défini précédemment.
Le tri n'est pas obligatoire ici, ce champ appartenant à une nouvelle année.
 - * Attribut **id_exp** (**Champs**)
 - * Attribut **espece** (**Champs** / **Mel_esp**)
Les différentes modalités de l'attribut **espece** sont données par le résultat de la requête *liste_especes* (évolutives).
Les nomenclatures adoptées suivent l'économie de place tout en restant significatives:
 - colza → colza
 - blé → ble
 - tournesol → tournesol
 - maïs → mais
 - pommes de terre → pdt etc...Les sous-classes d'espèces précisées suivent le nom d'espèce, après un underscore, symbolisées par leur première lettre:
 - orge d'hiver → orge_h
 - orge de printemps → orge_p

 - blé tendre → ble_t
 - blé dur → ble_dAfin de conserver un maximum d'information, toute précision peut être ajoutée après un underscore (facilement escamotable):
 - jachère nue → jach_n
 - jachère tournesol → jach_tournesol
 - * Attribut **Annee** (**Champs** / **Mel_esp** / **Mel_var**)
 - * Attribut **surface** (**Champs** / **Mel_esp** / **Mel_var**)
Cette valeur correspond à la surface emblavée déclarée par l'agriculteur.
 - * Attribut **var** (**Champs** / **Mel_esp** / **Mel_var** / **Varietes**)
Les variétés de colza sont mentionnées dans les tables **Champs**, **Mel_esp** et **Mel_var** selon un code numérique simple (**id_variete**), dont la correspondance est donnée par la table **Varietes**.
Toute nouvelle variété est ajoutée à la suite des autres, avec un **id_variete** nouveau et consécutif, son nom commercial et son type variétal⁵.

5. voir BASE DE DONNÉES COLZA: STRUCTURE

- Cas des mélanges⁶ : Plus d'une espèce sur une des parcelles d'un champ.

Tout mélange d'espèces ou de variétés de colza utilise une symbolique qui renvoie à la table Mel_esp et/ou Mel_var (selon le cas), qui préciseront ce mélange.

Symboliques utilisées:

- mélange d'espèces : attribut espece = melange.
- mélange de variétés de colza : attribut espece = colza et attribut var = 0.

Les tables Mel_esp et Mel_var suivent les mêmes principes de mise à jour que précédemment, avec la seule particularité de fournir la répartition parcellaire des différentes espèces (entourée par des accolades).

Exemple:



Cet exemple présente le cas de figure le plus complexe, avec:

- cohabitation sur un même champ de plusieurs espèces
- dont plusieurs variétés de colza
- des surfaces manquantes.

Le traitement de cette répartition sera:

◇ Table **Champs**

id_champ	id_exp	espece	surf	annee	var
1	25	melange		2002	

◇ Table **Mel_esp**

id_champ	espece	surf	annee	parcelles	var
1	colza	7.5	2002		0
1	ble		2002	{1,2}	
1	orge	6	2002	{3}	

◇ Table **Mel_var**

id_champ	var	surf	annee	parcelles
1	30	4	2002	{1,2}
1	5	3.5	2002	{1,2,3}

6. voir définition d'un mélange dans la section LECTURE DE L'ENQUÊTE

Tous les cas litigieux sont listés dans un cahier prévu à cet effet; toute bizarrerie devrait alors trouver une explication.

Divers requêtes⁷ permettent de vérifier quelques erreurs ou oublis:

- Champs non définis dans **compo_champs**: *champs_compo_champs*.
- Champs avec un mélange d'espèces non défini dans la table **mel_esp**: *champs_mel_esp*.
- Champs avec un mélange de colzas non défini dans la table **mel_var**: *champs_mel_var*.
- Composition de champs non mentionnées dans la table **champs**: *compo_champs_champs*.
- Mélanges d'espèces non mentionnés dans la table **champs**: *mel_esp_champs*.
- Mélanges de colzas non mentionnés dans la table **champs**: *mel_var_champs*.

2.3 Table Traitements

La mise à jour de cette table ne présente pas de difficultés, outre les modalités et contraintes.

Un type de traitement n'est mentionné qu'une fois par exploitant et par année.

Si l'agriculteur utilise le même type de traitement plusieurs fois par année, les différentes périodes seront listées dans la colonne *periode*.

Les types de traitements rencontrés sont:

- Traitement mécanique → M
- Traitement chimique → C
- Traitement par fauche → F
- Traitements combinés → CF par exemple

Les périodes d'entretien, au maximum semi mensuelles, sont encadrées par des accolades (et facultativement de guillemets) et séparées par des virgules.

Exemple: {mi juin, deb juil}.

7. tenant compte des tables originales et des copies utilisées pour la saisie

Chapitre 3

Mises à jour de la sous partie **Transport**

3.1 Table Transports

Les renseignements contenus dans cette table ne sont plus spécifiés dans les enquêtes depuis l'année 2000.

Les éléments fournis par les agriculteurs sont textuellement repris:

- Bâchage des remorques → t (true) ou f (false)
- Etanchéité des remorques → B (bon), TB (très bon)
- Matériel agricole utilisé → P (personnel) ou E (entreprise)

3.2 Table Silos

La limitation géographique de la zone d'étude laisse supposer que cette table n'a pas à subir de mise à jour pour le moment.

Les différents silos sont désignés par le nom de la commune qui les héberge, selon une écriture spécifique: majuscule, et underscore en séparation pour les noms composés (*Exemple: Coulommiers_la_tour*).

Les coordonnées géographiques sont acquises par visualisation du parcellaire sous le logiciel spécifique Arcview (les silos hors zone restent avec des coordonnées géographiques non renseignées).

3.3 Tables Trajets / Compo_trajets / Segments

Les trajets parcourus par les exploitants, de leurs champs de colza à leur(s) silo(s) de récolte, sont décrits dans cette sous partie 'Transport'.

Chaque trajet défini dans la table **trajets** part d'un champ de colza et arrive à un silo de récolte (passage éventuel par la ferme).

La décomposition de chaque trajet en segments de routes est fournie par la table **compo_trajets**, elle-même explicitée par la table **segments**, contenant les coordonnées géographiques correspondant à chaque segment de routes. L'ordre de la saisie est alors tout tracé.

Cette saisie nécessite l'utilisation en parallèle du logiciel Arcview (visualisation des routes et chemins numérisés).

3.3.1 Table Trajets

* Attribut 'id_trajet'

Les nouveaux trajets sont rentrés à la suite des trajets prédéfinis après tri par id_trajet (clé de la table).

* Attribut 'id_champ'

La sous partie 'Culture' étant mise à jour, la requête *bilans_exploitation_bruts* permet de retrouver directement les champs (et donc leur identifiant) concernés par la mise à jour.

* Attribut 'annee'

Correspond à l'année de parcours du trajet.

* Attribut 'silo'

Les différents silos sont désignés par le nom de la commune qui les héberge, selon une écriture spécifique: majuscule, et underscore en séparation pour les noms composés.

* Attribut 'direct'

L'exploitant déclare en général en tête de l'enquête sil fractionne sa récolte (passage par la ferme) ou s'il effectue un trajet direct vers le silo.

* Attribut 'complet'¹

Les trajets peuvent ne pas être complètement décrits sur la carte, car en partie hors zone. Dans ces cas, les trajets considérés sont les parties de trajets comprises dans la zone.

* Attribut 'auteur'

Selon les années, une signalétique particulière indique l'origine des trajets: trajet avéré (agr) ou trajet supposé (sup), décrits par lagriculteur².

1. Une évolution possible de cet attribut serait la caractérisation des trajets incomplets même compris dans la zone d'étude.

2. Une évolution possible de cet attribut serait: trajet fourni par l'agriculteur, par l'enquêteur, ou simplement supposé.

3.3.2 Table Compo_trajets

La visualisation des routes et chemins déjà numérisés sous le logiciel Arcview permet de déterminer les routes suivies lors des trajets.

Les différentes routes sont signalées par des couleurs différentes mais la palette de couleurs étant limitée, les routes empruntées doivent être observées en zoom. En effet, deux couleurs proches ou même identiques ne sont pas faciles à distinguer...

Il faut donc vérifier que l'identifiant de la route sur laquelle on se trouve reste le même jusqu'à la prochaine intersection repérée.

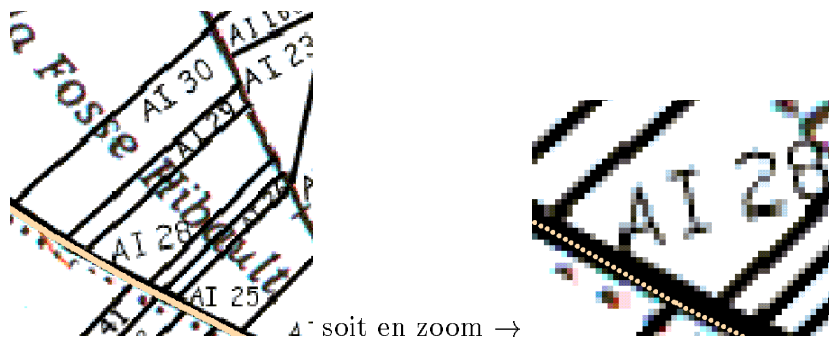
Le projet Arcview utilisé possède deux niveaux de précision; ainsi, si l'arc le plus proche semble trop éloigné, le fractionnement supplémentaire des routes est disponible afin de déterminer l'arc réellement le plus adéquat.

Cette précision demande cependant une gestion plus longue, il est donc conseillé de désélectionner le thème correspondant pour la suite de la saisie.

Exemple pour un trajet partant de la parcelle AI18 (segments définis en orange):



Le premier segment à choisir n'est pas clairement défini, il faut alors sélectionner la table complète (arcs de taille inférieure à 3m):



Le segment le plus approprié peut alors facilement être déterminé.

Chaque trajet sera, à partir de maintenant, décomposé en portions de route ordonnées, correspondant à des intervalles de segments de 3m (identifiants du premier et du dernier segments concernés).

Cette structure permettra une jonction directe avec la partie échantillonnage.

Cette saisie peut être réalisée sous Excel dans un premier temps, les routes principales étant très fréquemment utilisées, et donc les suites d'arcs correspondantes. Excel est davantage adapté à ce type de répétitions³(moins d'erreurs à priori que de manipuler des séries de chiffres similaires). Cette particularité explique que certains identifiants d'arcs soient souvent mentionnés.

Les trajets non complets seront démarrés au plus proche du champ d'origine (hors zone).

L'utilisation de routes non utilisées jusqu'à présent sera précédée d'une étape de découpage en segments et d'acquisition de leurs coordonnées.

3.3.3 Table arcs (en cours de remplacement par la table segments)

Les arcs sont ici définis: leur coordonnées géographiques issues du traitement de tracés réalisés sous Arcview, le type de voie auquel ils appartiennent.

Ces arcs de taille variable sont progressivement remplacés par les segments de 3m de la partie échantillonnage.

3. possibilité de *Copier-Coller*

Troisième partie

Mises à jour de la partie échantillonnage

Chapitre 1

Table Populations

La table populations a été supplée de quelques attributs, et complétée par quelques modalités. Les nouveaux attributs ont été représentés en rouge, alors que les attributs simplement renommés (attention à la mise à jour des requêtes) sont indiqués en bleu.

Populations
id_pop
annee
date
id_st
champ
site
periode
eff
gps
dir
stade
trait
comp
rcolza
densite
xy

1.1 Présentation de la nouvelle table Populations

Commençons par une brève description des nouveaux attributs et des attributs renommés:

- **champ** :

Booléen

Equivalent de l'ancien attribut "type"

"t" (true) ou "f" (false)

Indique si la population appartient à un champs (t) ou non (f)

- **site :**

Texte

Précision pour les repousses (attribut champs = "false")

Plantes cultivées (champs=t) -- > \ N

Repousses en bordure (féral) -- > bordure

Repousses dans une jachère -- > jach

Repousses dans un champs

culture non précisée -- > c

culture de blé -- > c_ble

culture d'orge -- > c_orge

culture de seigle -- > c_seigle

culture de trèfle -- > c_trefle

culture de tournesol -- > c_tournesol

culture de maïs -- > c_maïs

culture de betteraves -- > c_bettes

culture de crucifères -- > c_crucif

culture de féverolles -- > c_feverolles

culture de pois -- > c_pois

champs labouré -- > c_labour

- **eff**

Numérique (int4)

Ancien attribut effectif

Effectifs précis jusqu'à 50 environ, puis effectifs approximatifs

Effectif de 100.000 plantes attribué aux populations récoltées (periode = 'rec) ayant un effectif non précisé mais composé de 10 sacs ou plus; ce nombre de sacs représente statistiquement une population (remarque: dans ces cas, une population de 11 sacs peut apparaître du fait de l'ajout d'une plante intéressante après coup). Ce chiffre de 100.000 a été choisi pour dénoter, il sera à changer (attention au changement de type éventuel) si les effectifs observés augmentent de trop (maximum 10.000 plantes estimées dans une population pour le moment).

- **periode**

Texte

Différencie les populations relevées en floraison "fl" ou en récolte "rec".

- **gps**

Numérique (int2)

Numéro de gps dune population: unique pour une date donnée, au contraire d'id_station dont la numérotation peut se répéter dans une même journée.

Les populations sans numéro de gps sont laissées dans un fichier à part mais ne sont pas rentrées dans la base colza; en effet, ces populations étant en très faible nombre, leur éventuelle utilisation ne justifie pas la modification des contraintes de la table populations (présence de coordonnées).

- **dir**

Texte

Indique la direction de la droite de la personne lors du démarrage du gps (N, S, O, E).

- **stade**

Texte

Indique le stade de développement global des populations (rosette, puis de C1 à G7).

Selon la période d'échantillonnage on peut retrouver les stades D1 ou BF, D1 étant plus précis que BF, Bouton Floral.

- **trait**

Texte

indique si le(s) traitement(s) appliqué(s) aux chemins et bords de routes sur lesquels les populations ont été repérées.

traitement chimique -- > C

fauche -- > F

broyage -- > B

et leurs combinaisons -- > CF, CBF, BF

sans traitement -- > rien

- **comp**

Booléen

Indique si une compétition interspécifique est visible (t pour true, f pour false).

- **rcolza???**

Booléen

Indique la présence (t pour true, f pour false) en bordure d'un champ de colza, de plants de colza d'un type distinct (susceptibles de ne pas provenir de ce champ).

- **densite**

Texte

Précise la répartition d'une population :

plantes de la population isolées -- > i

plantes groupées -- > g

plantes en touffes -- > t

populations denses -- > d

La combinaison des attributs (*annee, date, gps*) définit de façon unique une population (anciennement (*annee, periode, gps*) cf. **gps**).

1.2 Mise à jour

1. Consultation des fichiers source

La première étape consiste à détailler le contenu des fichiers source : formats (excel ou texte), données (une colonne des fichiers représente au moins un attribut potentiel dans la table, voire une autre table), différentes modalités pour chaque colonne, remarques importantes...

Il est conseillé de garder une copie des fichiers originaux, au cas où :

Exemple: */home/gharbi/mise_a_jour/saisie_analyses_annee/fichiers_originaux*

Selon les cas, des données peuvent être réparties sur plusieurs fichiers (avril et mars pour floraison; champs, féral et jachère pour une période donnée...). Il est conseillé de les uniformiser (vérifier que, pour chaque fichier, le type de population soit précisé - champs, féral, jachères) afin de pouvoir les regrouper (si tous les fichiers sont disponibles) et ainsi les traiter en une seule fois...

Une fois les fichiers regroupés, les colonnes inutiles pour la table populations sont supprimées. Ces fichiers constituent la base des données à saisir.

En l'état actuel de la table populations, l'idéal est la constitution de deux fichiers (ou deux feuillets Excel) :

Exemple:

/home/gharbi/mise_a_jour/saisie_analyses_annee/fichiers_arranges/periode ou mois_annee

- Premier fichier ou feuillet : attributs

Date	id_station	Type	Eff.	N ° Gps	Dir.	Stade	Trait.	comp.inter	Bord.colza	densité
------	------------	------	------	---------	------	-------	--------	------------	------------	---------

- Deuxième feuillet : positions

Différents formats de fichiers de positions ont pu être rencontrés jusqu'à présent :

- format classique

gps1	
x1	y1
x2	y2
x3	y3
gps2	
x1	y1
x2	y2
x3	y3

- format espace

gps1	x1	y1
	x2	y2
	x3	y3
gps2	x1	y1
	x2	y2
	x3	y3

- format bloc

gps1	x1	y1
gps1	x2	y2
gps1	x3	y3
gps2	x1	y1
gps2	x2	y2
gps2	x3	y3

OU

gps1	x1	y1
gps1	x2	y2
gps1	x3	y3
gps2	x1	y1
gps2	x2	y2
gps2	x3	y3

Ces deux formats sont équivalents après tri par numéro de gps.

2. Mise au format 'base des fichiers

- Positions

Les valeurs comprises par la procédure postgeo comporte des points pour les décimales et non des virgules. Il faut donc s'assurer de cette concordance avant toute manipulation.

Selon le format des coordonnées à disposition, différentes actions sont utiles :

- format classique : il suffit de rajouter des 'END' dans les espaces vides entre les pop, et utiliser la procédure **geopost_classique**
- format espace : la procédure **geopost_espace** a été dernièrement écrite pour gérer ce format, il faut cependant pour le moment rajouter une ligne vide (deux tabulations) à la fin du fichier pour que le dernier jeu de coordonnées soit pris en compte
- format bloc : la procédure **geopost_bloc** est adaptée, avec de même une ligne vide en fin de fichier

Après action de la procédure appropriée, le fichier subira des modifications sous Excel ou exclusivement en Texte sous Unix (Soffice ayant une capacité limitée en terme de longueur des données dans chaque case doù troncature des coordonnées).

Il arrive que des caractères particuliers soient incorporés dans les fichiers de données; ceux-ci sont mis en évidence lors de l'intégration dans la base et seront corrigés au cas par cas à l'aide du fichier d'origine (disparition d'un crochet, ajout d'un caractère etc...).

- Attributs

Les différents attributs de la table populations ont des modalités définies, qu'il faut respecter. Pour cela, après avoir visualisé les différentes possibilités par un filtre (*Données-Filtrer-Filtre automatique*) des différentes colonnes, l'option "remplacer" (*Edition-Remplacer*, symbolisée par -- > par la suite) permet de mettre les attributs tels qu'ils puissent être rentrés dans la base.

A titre d'exemple, pour la table populations :

- **annee** :
recopie de l'année donnée.
- **champ** :
copier-coller puis insertion des cellules copiées pour créer la colonne "site".
"champ" -- > t (pour true)
autres -- > f (pour false)

- **site:**

cf. “DESCRIPTION DE LA NOUVELLE TABLE POPULATIONS” pour les différentes modalités dont le détail des repousses , sans accent).

copie de la colonne donnant les types de repousses ainsi modifiée puis collage spécial sur la colonne “site” avec l’option “ignorer les blancs”:

Champs	Champs_bis	Site		Champs	Champs_bis	Site		Champs	Site
f	Repchamps	Blé	→	f	Repchamps	c_ble	→	f	c_ble
f	Repchamps	mais		f	Repchamps	c_mais		f	c_mais
f	jach			f	jach			f	jach
t				t				t	

- **periode :**

recopie de la période (fl pour mars/avril et rec pour juin/juillet).

- **eff:**

vérifier les effectifs de récolte par rapport aux numéros de sacs renseignés (le logiciel de repérage peut attribuer une valeur par défaut, donc erronée)

pour les populations cultivées et récoltées(champs =t, periode = 'rec et n° sacs non null), un effectif symbolique de 100.000 plantes remplace l’absence de précision (populations composées de 10 sacs ou plus). Ceci reste à vérifier pour l’année 2002.

- **gps :**

vérifier qu’un n° gps soit unique pour une année et une date données.

- **dir :**

Lettre initiale et majuscule de la direction.

vérifier avec le filtre des différentes valeurs (possibilité d’introduction d’espaces: 'Nord et 'Nord ' par exemple).

- **stade :**

nouveaux stades éventuels à intégrer aux contraintes de la table.

- **trait :**

penser à supprimer le “+” utilisé lors de combinaison de traitements (le remplacer par un vide mais pas un espace).

Les autres colonnes ne posent pas de problème. Les cases vides seront remplacées par “\N”, symbole de la valeur vide pour PSQL.

Excel utilisant des formats automatiques pour les dates, celles-ci seront à reprendre sous un éditeur de texte ou à protéger sous un format texte (10-04-2000 par exemple).

Il est préférable de comparer les fichiers attributs et positions par un tri de chacun par n° gps et vérifier que toutes les populations aient un n° gps (sinon mise à part et retrait pour la base) et

que tous les numéros de gps aient une population associée (sinon, suppression du jeu de coordonnées inutile).

Une attention toute particulière doit être accordée aux fichiers de récolte car les numéros de sacs des populations récoltées peuvent révéler des regroupements de populations après coup et donc la matérialisation plus tard dans la base par une seule population (regroupement des coordonnées après vérification sous Arcview de leur cohérence).

Des repérages complémentaires peuvent venir compléter le fichier de base (par exemple mars 2000 complété par des données d'avril 2000).

Seules les populations nouvelles, non repérées en mars doivent être prises en compte. Une visualisation sous Arcview (à partir des coordonnées brutes) permet de déterminer les populations utiles (non présentes en mars ou plus étendues en avril qu'en mars).

Exemple: */home/gharbi/mise_a_jour/fichiers_alex_arranges/floraison_2000/arcview*

Ces projets (un projet par type de populations, féral, jachère ou champs) mettent en évidence les populations d'avril (en vert), celles de mars (en rouge) et celles de juillet (en bleu). Il est alors aisé de comparer les positions des populations en fonction des mois (différentes épaisseurs permettent de bien les distinguer en cas de juxtaposition).

Les positions de juillet permettent de conclure dans le cas de populations d'avril plus étendues que celles de mars: si la population de juillet est de même étendue que celle d'avril, les données d'avril vont remplacer celles de mars car plus complètes; sinon, les données de mars sont conservées. Un nouveau fichier regroupe alors les données de mars et les données utiles d'avril (possibilité de fichiers intermédiaires avec les données d'avril et de mars utiles: retrait des populations d'avril redondantes et des populations de mars moins complètes que celles d'avril).

Une fois toutes les vérifications réalisées, les id_pop peuvent être attribuées: du dernier id_pop de la base jusqu'à la dernière population du fichier, en commençant pour chaque année par les populations en floraison.

1.3 Intégration dans la base 'COLZA'

Les données propres (sous format texte) peuvent être intégrées ensemble dans la base (rassemblement des fichiers attributs et positions) ou séparément (création de deux tables réunies par une jointure en utilisant l'attribut gps).

La seconde méthode est d'autant plus longue qu'il faudra faire la jointure séparée des fichiers flo-raison ou récolte pour avoir un n ° gps unique.

Dans le premier cas, il suffit de créer une table identique à la table populations:

sous psql colza

1. Création d'une table populations temporaire

- Création indirecte :

- create table populations_année as (select * from populations)

- truncate populations_année

- Création directe :

- create table populations_année as (select * from populations where id_pop isnull)

2. Insertion des nouvelles données dans la table populations temporaire :

- \copy populations_année from *nom_fichier_donnees*

3. Vérification des données de la table temporaire (possibilité de erreurs de transfert)

4. Insertion des données de la table temporaire dans la table populations réelle :

- insert into populations (select * from populations_année)

Chapitre 2

Table Plantes, acides et gls : résultats d'analyses

2.1 Traitement des fichiers d'analyses : tables "acides" et "gls"

Les fichiers d'analyses du CETIOM peuvent se présenter sous format texte ou sous format Excel selon les années. Dans la mesure du possible, il est préférable de travailler sur les fichiers Excel car la précision des teneurs y est meilleure.

Selon les formats des fichiers et leur nombre, il sera préférable de les rassembler dès le départ ou de les traiter séparément. Avant tout regroupement, il est important de vérifier le contenu de chaque fichier : nombre et intitulés des colonnes renseignées, types d'informations données dans chaque colonne (valeurs ou calculs de colonnes), commentaires ou légendes (unités de mesure, cas particuliers)...

Les seules informations à conserver seront :

- Pour les données d'**acides** : l'identifiant unique pour chaque échantillon (le numéro de labo en général) et la teneur en érucique (C22:1), le symbole '-' correspondant à une teneur en acide nulle et non à une absence d'analyse (absence caractérisée par l'absence du fichier d'un identifiant donné)
- Pour les données de **glucosinolates** : l'identifiant unique pour chaque échantillon (le numéro de descendant en général) et la teneur totale en gls.

Le format décimal correct pour l'intégration dans la base utilise un point et non une virgule.

Les tables "acides" et "gls" sont obtenues directement.

2.2 Traitement des fichiers populations récoltées : table "plantes"

Il est préférable d'avoir des fichiers populations (attributs pour période =rec) propres avant de procéder à cette mise à jour, afin de limiter les remaniements de fichiers.

Les informations utiles ici dans les fichiers attributs de populations sont les colonnes : id_pop, n° sacs, et éventuellement remarques si elle sont pertinentes pour cette saisie (populations regroupées, pas de sac ou NR pour Non Récoltées...)

Les effectifs récoltés des populations peuvent être obtenus par une option d'ouverture du fichier : séparation des colonnes du fichier texte par, selon les cas, 'à', ou '-' puis différence (pensez au '+ 1' de l'opération de différence pour obtenir l'effectif correct) entre les deux colonnes obtenues (ou plus dans le cas de rassemblement de sacs).

L'étape suivante pourra être automatisée par la création d'une procédure appropriée: création d'autant de lignes que de plantes dans la population (effectif calculé) associées à chaque id_pop.

Exemple :

id_pop	eff_calculé	n ° sacs	remarques
1	4	1	
1		2	
1		3	
1		4	

2.3 Confrontation des fichiers populations et analyses

Tous les id_plante de la base 'COLZA' correspondent à une plante analysée, il faut donc faire le tri et retirer les plantes "virtuelles".

Exemple :

- Au départ

id_pop	n ° sacs	remarques	n ° labo = n ° sacs	erucique	desc. = n ° sacs	gls
1	1		1	0.2	1	3.6
1	2		2	0.3	4	5.7
1	3		4	0.4		
1	4					

- A l'arrivée

id_pop	n ° sacs	remarques	n ° labo = n ° sacs	erucique	desc. = n ° sacs	gls
1	1		1	0.2	1	3.6
1	2		2	0.3		
1	4		4	0.4	4	5.7

La plante correspondant potentiellement au n ° sac 3 est retirée car non analysée (ni en acide, ni en gls).

Une fois ce ménage effectué, une simple attribution d'un id_plante (à partir du dernier id_plante de la base) suffit à compléter la table plantes.

La valeur de la colonne "quantite" est de 1 par défaut (voir les remarques sinon).

Exemple :

id_pop	quantite	id_plante
1	1	1
1	1	2
1	1	3

id_plante	acide
1	0.2
2	0.3
3	0.4

id_plante	gls
1	3.6
3	5.7

Chapitre 3

Table de correspondance populations cultivées/parcelles cadastrales

Les opérateurs géométriques disponibles en PSQL sont d'une utilisation assez contraignante (détermination d'une distance permettant de discriminer deux populations... Ainsi, depuis juillet 2002, le champ auquel appartient une population cultivée est repéré par son numéro de parcelle cadastrale. Cette information sera intégrée dans la base sous forme d'une table supplémentaire dont la structure définitive reste à définir.

Le fichier de base de cette correspondance entre la partie échantillonnage et la partie enquête de la base est cependant disponible.

Le but d'un fichier de correspondance est de pouvoir attribuer un `id_parcelle` aux numéros de parcelles cadastrales repérées lors de l'échantillonnage.

3.1 Populations avec numéro cadastral renseigné

Un simple tri par numéro cadastral sur le fichier populations suffit à les isoler.

3.2 Visualisation sous Arcview

L'étape suivante consiste à utiliser Arcview pour situer les populations et ainsi vérifier et compléter les informations concernant les n° cad.

La situation optimale pour visualiser une population sous Arcview est de partir d'un fichier sous un format de type :

gps	x	y
1	x1	y1
1	x2	y2
1	x3	y3

Il faut donc pouvoir obtenir ce type de fichier : la procédure postgeo pourra être revue afin d'assurer ce type de conversion de façon automatique.

Sous Arcview, ajouter la table correspondant à ce fichier et visualiser le thème correspondant (penser à laisser la ligne d'en tête).

Un thème préexistant de parcelles sera utilisé jusqu'à la mise à jour de la table parcelles (en cours sous le nom 'k_parcelles').

L'utilisation d'une requête pour faire cette comparaison risque de laisser échapper des cas particuliers que seule la visualisation permet de mettre en évidence (incertitudes de la localisation selon le parcellaire...

Sur un fond de carte du parcellaire de l'année, le fichier peut-être complété (il est conseillé de suivre un ordre prédéfini, par n ° gps croissant par exemple):

Exemple :

id_pop	section	n ° cad	ville	enquêtés?	confrontation	remarques
--------	---------	---------	-------	-----------	---------------	-----------

La colonne "enquêtés?" est obtenue grâce au parcellaire de l'année où les champs enquêtés et non enquêtés sont distingués.

La colonne confrontation indique si la visualisation donne un résultat cohérent avec les données de terrain (imprécisions de localisation), et la colonne remarques permet de préciser cette dernière en cas de non concordance.

3.3 Perspectives

3.3.1 Procédure postgeo

Une modification de la procédure postgeo permettrait de passer plus directement des données de la table populations à leur visualisation sous Arcview.

3.3.2 Procédure de traitement des effectifs

La mise à jour des tables **plantes** et **glucosinolates** passent par une manipulation actuellement fastidieuse: il faut créer, pour chaque population, autant de lignes que de plantes la composant. Ce nombre de plantes étant donné par la colonne effectif, il est envisageable de concevoir une procédure qui créerait le nombre de lignes voulu, pour chaque population.

3.3.3 Saisie des données de terrain

Une saisie optimale consisterait à reprendre le format de la base pour les parcelles cadastrales, à savoir : section, n° cad et ville séparés. Ceci permettrait une comparaison directe avec les données de la base, il ne resterait plus alors qu'à compléter le fichier avec toutes les parcelles jouxtant une population donnée.

3.3.4 Résultats d'analyses

Les résultats d'analyses sous format Excel sont plus précises que sous format texte; Dans la mesure du possible, la récupération de tels fichiers pourrait affiner les données d'analyses de la base.

Quatrième partie

Annexes

Annexe 1

Installation de PgAccess

Source : </home/ab/COLZA/trucs.html>

Installation de PgAccess sous Windows

- Installer PgAccess sur Unix: il est incorporé à PostgreSQL mais si on veut l'installer en autonome, voir: <http://ns.flex.ro/pgaccess/>

- Installer PgAccess sur micro

Comment Kasia a installé pgaccess version 0.98.7, (pgaccess est une interface à PostgreSQL) sur des micros Windows en mars 2002

Remarques préalables

Ici, n'est décrit que l'installation sur windows.

En effet, sur Solaris, PgAccess étant incorporé à PostgreSQL, il n'y a pas d'installation spécifique à faire: installer PostgreSQL installe automatiquement PgAccess (voir postgresql.html)

Il faut que Tcl/Tk soit installé préalablement

L'installation a été faite sur 2 micros travaillant sous plate-forme Windows NT, v.4. Kasia a aussi fait la procédure d'installation sur sa machine perso, sous Windows95.

Étapes d'installation:

1. Installation préalable de Tcl/Tk , version 8.2.3 (compatible avec PgAccess 0.98.7)
 - téléchargement du fichier d'installation <http://prdownloads.sourceforge.net/tcl/tcl823.exe>
 - Attention : téléchargement parfois très lent !
 - installation sous Windows : lancer tcl823.exe, suivre les étapes standards proposées dans la procédure d'installation (on peut placer TCL/TK sous 'C:/ProgramFiles')
2. Installation de PgAccess:
 - téléchargement du fichier <http://ns.flex.ro/pgaccess/download/pgaccess-0.98.7.zip>
 - unzip du fichier téléchargé (on peut le placer également sous 'C:/Program Files')
 - téléchargement des librairies nécessaires pour la version Tcl/Tk: <http://ns.flex.ro/pgaccess/download/6.5> et <http://ns.flex.ro/pgaccess/download/libpq.dll-6.5.1>
 - déplacement de ces deux fichiers dans le répertoire C:/WINNT/system et changement des noms en libpgtcl.dll et libpq.dll

Exécution de PgAccess:

Dans le répertoire d'installation de PgAccess (ici 'C:/Program Files/pgaccess') retrouver le fichier 'main.tcl', l'exécuter : la fenêtre de PgAccess doit apparaître

Connexion à la base:

Supposons que l'on veuille se connecter à partir de la machine 'fourier', avec PgAccess installé, à la base de données 'ogmtest' existant sous PostgreSQL sur la machine 'banian'.

Pour cela, il faut d'abord modifier le fichier 'pg_hba.conf' de PostgreSQL de 'banian' en ajoutant la ligne suivante : host ogmtest 138.102.16.41 trust

'host' - connexion autorisée à la machine extérieure

'ogmtest' - le nom de la base

'138.102.16.41' - le numéro IP de la machine extérieure

'trust' - mode d'accès sans le mot de passe

Après avoir fait ces modifications, on a l'autorisation de consulter la base 'ogmtest' à partir de PgAccess de 'fourier'. Pour cela il suffit d'activer l'option 'Database-Open' et dans la fenêtre de dialogue de préciser host (ici le numéro IP de banian), le nom de la base de données (ici 'ogmtest') et, si nécessaire, le nom d'utilisateur et son mot de passe. (inutile si le mode d'accès autorisé = 'trust').

Annexe 2
Structure des tables : Commandes SQL

```

create table populations
(
id_pop int2 NOT NULL primary key,
annee int2,
date date,
id_st int2,
champ text,
site text,
periode text,
eff int4,
gps int2,
dir text,
stade text,
trait text,
comp text,
rcolza text,
densite text,
xy path,
CONSTRAINT annee CHECK (annee BETWEEN 1996 and EXTRACT(YEAR FROM now())),
CONSTRAINT champ CHECK (champ IN ('t','f')),
CONSTRAINT site CHECK (site IN ('bordure','jach','bordure+c','c','c_ble','c_orge','c_seigle','c_trefle','c_tournesol','c_crucif','c_feverolle','c_pois','c_labour') OR site IS NULL),
CONSTRAINT periode CHECK (periode IN ('fl','rec')),
CONSTRAINT eff CHECK (eff >0 or eff IS NULL),
CONSTRAINT gps CHECK (gps >0),
CONSTRAINT dir CHECK (dir IN ('O','N','E','S')),
CONSTRAINT stade CHECK (stade IN ('C1','C2','F0','F1','F2','G1','G2','G3','G4','G5','BF','D1','D2','E','G6','G7') OR stade IS NULL),
CONSTRAINT trait CHECK (trait IN ('rien','C','F','B','BF','CBF','CF') OR trait IS NULL),
CONSTRAINT comp CHECK (comp IN ('t','f') OR comp IS NULL),
CONSTRAINT rcolza CHECK (rcolza IN ('t','f') OR rcolza IS NULL),
CONSTRAINT densite CHECK (densite IN ('i','g','t','d') OR densite IS NULL),
UNIQUE (date,periode,gps)
);

```

```

create table plantes
(
id_plante int2 NOT NULL PRIMARY KEY,
quantite int2,
id_pop int2,
FOREIGN KEY (id_pop) REFERENCES populations (id_pop),
CONSTRAINT quantite CHECK (quantite >0 OR quantite IS NULL)
);

```

```

create table acide
(
id_plante int2 NOT NULL,
acide float4,
FOREIGN KEY (id_plante) REFERENCES plantes (id_plante),
CONSTRAINT acide CHECK (acide >= 0)
);

```

```

create table glucosinolates
(
id_plante int2 NOT NULL,
gls float4,
FOREIGN KEY (id_plante) REFERENCES plantes (id_plante),
CONSTRAINT gls CHECK (gls >= 0)
);

```

```

create table isozymes
(
id_plante int2 NOT NULL,
id_descendant int2 NOT NULL,
pgi int2,
aco int2,
mdh int2,
lap int2,
PRIMARY KEY (id_plante, id_descendant),
FOREIGN KEY (id_plante) REFERENCES plantes(id_plante),
CONSTRAINT id_descendant CHECK (id_descendant BETWEEN 1 AND 10),
CONSTRAINT pgi CHECK ((pgi BETWEEN 0 AND 3) OR (pgi=990) OR (pgi IS NULL)),
CONSTRAINT aco CHECK ((aco BETWEEN 0 AND 22) OR (aco=990) OR (aco IS NULL)),

```

```

CONSTRAINT mdh CHECK ((mdh BETWEEN 0 AND 14) OR (mdh=990)OR (mdh IS NULL)),
CONSTRAINT lap CHECK ((lap BETWEEN 0 AND 2) OR (lap=990) OR (lap IS NULL))
);

```


```

create table silos
(
silo text NOT NULL,
coop text,
xy point,
PRIMARY KEY(silo)
);

```

```

create table parcelles
(
id_parcelle int2 NOT NULL,
ville text NOT NULL,
section char(2) NOT NULL,
no_cad int2 NOT NULL,
surf_cad float4,
xy polygon,
PRIMARY KEY (id_parcelle),
CONSTRAINT ville CHECK (ville IN ('Baigneaux','Boisseaux', 'Champigny_en_beauce','Conan', 'Coulommiers_la_tour', 'Crucheray', 'Faye', 'Perigny', 'Rhodon', 'Selommes', 'Ste.Gemmes', 'Tourailles', 'Villefrancoeur', 'V
UNIQUE (ville,section,no_cad)
);

```

```

create table arcs
(
id_arc int4 NOT NULL,
type text,
xy path,
PRIMARY KEY(id_arc),
CONSTRAINT type CHECK (type IN ('lien_parc','R','T'))
);

```

```

create table trajets
(
id_trajet int2 NOT NULL,
id_parcelle int2 NOT NULL,
silo text,
annee int2,
auteur text,
direct boolean,
complet boolean,
xy path,
PRIMARY KEY (id_trajet),
FOREIGN KEY (id_parcelle) REFERENCES parcelles(id_parcelle),
FOREIGN KEY (silo) REFERENCES silos(silo),
CONSTRAINT annee CHECK (annee BETWEEN 1988 and EXTRACT(YEAR FROM now())),
CONSTRAINT auteur CHECK (auteur in ('agr','sup'))
);

```

```

create table compo_trajets
(
id_trajet int2 NOT NULL,
ordre int2 NOT NULL,
id_arc int4 NOT NULL,
FOREIGN KEY (id_trajet) REFERENCES trajets(id_trajet),
FOREIGN KEY (id_arc) REFERENCES arcs(id_arc)
);

```

```

create table transports
(
id_exp int2 NOT NULL,
remorques_b text,
remorques_e text,
materiel text,
annee int2,
FOREIGN KEY (id_exp) REFERENCES exploitants(id_exp) ,
CONSTRAINT remorques_b CHECK (remorques_b IN ('B','NB','BNB')),
CONSTRAINT remorques_e CHECK (remorques_e IN ('C','TB','CTB')),
CONSTRAINT materiel CHECK (materiel IN ('P','E','PE')),
CONSTRAINT annee CHECK (annee BETWEEN 1999 and EXTRACT(YEAR FROM now())));

```

```

create table traitements
(
id_exp int2 NOT NULL,
type text NOT NULL,
periode text[],
annee int2 NOT NULL,
FOREIGN KEY (id_exp) REFERENCES exploitants(id_exp),
CONSTRAINT type CHECK (type IN ('C','M','MF','MB','MC','rien')),
CONSTRAINT annee CHECK (annee BETWEEN 2000 and EXTRACT(YEAR FROM now())),
UNIQUE (id_exp, type, annee)
);

```

% triggers à ajouter

```

create table compo_champs
(
id_champ int2 NOT NULL,
id_parcelle int2 NOT NULL,
UNIQUE(id_champ,id_parcelle)
);

```

% triggers à ajouter

```

create table champs
(
id_champ int2 NOT NULL,
id_exp int2 NOT NULL,
espece text NOT NULL,
annee int4 NOT NULL,
surf float8,
var int4,
FOREIGN KEY (id_exp) REFERENCES exploitants(id_exp),
FOREIGN KEY (var) REFERENCES varietes (id_variete),
UNIQUE (id_champ, annee, espece,id_exp),
CONSTRAINT annee CHECK (annee BETWEEN 1998 and EXTRACT(YEAR FROM now())),
CONSTRAINT var CHECK (var > -1)
);

```

```

% triggers à ajouter

create table mel_esp
(
id_champ int2 NOT NULL,
espece text NOT NULL,
surf float8,
annee int2 NOT NULL,
parcelles _int2,
var int2,
CONSTRAINT annee CHECK (annee BETWEEN 1998 and EXTRACT(YEAR FROM now())),
CONSTRAINT var CHECK (var <= 1),
UNIQUE (id_champ,annee,espece)
);

% triggers à ajouter

create table mel_var
(
id_champ int2 NOT NULL,
var int2 NOT NULL,
surf float8,
annee int2 NOT NULL,
parcelles _int2,
CONSTRAINT annee CHECK (annee BETWEEN 1998 and EXTRACT(YEAR FROM now())),
CONSTRAINT var CHECK (var <= 1),
UNIQUE (id_champ,annee,var)
);

create table routes_annee
(
id_route int2,
annee int2,
FOREIGN KEY (id_route) REFERENCES routes(id_route),
UNIQUE (id_route,annee),
CONSTRAINT annee CHECK (annee BETWEEN 1996 and EXTRACT(YEAR FROM now()))
);

```

```

CREATE TABLE segments
(
id_seg int4 NOT NULL
, xy lseg,
type text,
PRIMARY KEY (id_seg),
CONSTRAINT type CHECK type IN ('g1v','g2v','ch','?')
);

```

```

CREATE TABLE routes
(
id_route int4 NOT NULL,
seg_min int4,
seg_max int4,
PRIMARY KEY (id_route),
FOREIGN KEY (seg_min) REFERENCES segments(id_seg),
FOREIGN KEY (seg_max) REFERENCES segments(id_seg),
UNIQUE (seg_min, seg_max)
);

```

```

CREATE TABLE trajets_ech
(
id_route int4 NOT NULL,
annee int2,
periode text,
FOREIGN KEY(id_route) REFERENCES routes(id_route),
CONSTRAINT annee CHECK (annee BETWEEN 1996 and EXTRACT(YEAR FROM now())),
CONSTRAINT periode CHECK (periode IN ('fl','rec')),
UNIQUE(id_route, annee, periode)
);

```

```

CREATE TABLE cultures
(
id_culture int4 NOT NULL,
id_champ int4 NOT NULL,
surf float8,
id_variete int4,

```

```

annee int4,
semence text,
PRIMARY KEY (id_culture),
FOREIGN KEY (id_champ) REFERENCES champs(id_champ),
CONSTRAINT annee CHECK (annee BETWEEN 1988 and EXTRACT(YEAR FROM now())),
CONSTRAINT semence CHECK (semence IN ('C','F','CF'))
);

```

```

create table exploitants
(
id_exp int2 NOT NULL,
xy point,
PRIMARY KEY (id_exp)
);

```

```

CREATE TABLE varietes
(
id_variete int2,
nom text,
type text,
PRIMARY KEY (id_variete),
CONSTRAINT type CHECK (type IN ('00','01','10','11')),
CONSTRAINT id_variete CHECK (id_variete >= 1),
UNIQUE (nom)
);

```

```

CREATE TABLE compo_trajets_agr
(
id_trajet int4 NOT NULL,
ordre int2 NOT NULL,
seg1 int4 NOT NULL,
seg2 int4 NOT NULL,
FOREIGN KEY (seg1) REFERENCES segments(id_seg)
);

```

```
CREATE TABLE trajets_agr
(
id_trajet int4 NOT NULL,
id_champ int4 NOT NULL,
annee int2 NOT NULL,
silo text NOT NULL,
direct text,
complet text,
auteur text,
CONSTRAINT annee CHECK (annee BETWEEN 1998 and EXTRACT(YEAR FROM now())),
FOREIGN KEY (silo) REFERENCES silos(silo),
CONSTRAINT auteur CHECK (auteur in ('agr','sup')),
CONSTRAINT direct CHECK (direct in ('oui','non')),
CONSTRAINT complet CHECK (complet in ('oui','non'))
);
```

Annexe 4
Bibliographie

- Sources logistiques
 - ★ babafou. *Apprends L^AT_EX!*. Ecole nationale supérieure de techniques avancées
 - ★ Marc Baudouin. *Apprends L^AT_EX*: http://tex.loria.fr/apprends_latex/apprends_latex.html
 - ★ Leslie Lamport. *L^AT_EX: A Document Preparation System*. Addison-Wesley, 5ème édition, 1986
 - ★ Marie-Paul Luth. *FAQ L^AT_EX française pour débutants et confirmés*. 1997
 - ★ Equipe Grappa. *FAQ L^AT_EX*: <http://www.grappa.univ-lille3.fr/FAQ-LaTeX/index.php>
 - ★ Unité de biométrie et intelligence artificielle de Jouy-en-Josas. *Accueil à l'interrogation d'une base de données*: <http://www.inra.fr/bia/J/AB/COLZA/welcome.php>

- Sources analytiques
 - ★ Fabrice Pessel. *De l'origine à la dynamique des populations spontanées de colza (Brassica napus): Une contribution à l'estimation et à la gestion des risques associés à l'introduction de variétés transgéniques*. thèse soutenue en juillet 2000.
 - ★ Véronique Emeriau et Katarzyna Adamczyk. *Base de données "Colza" et cartographie" (mise à jour 2001)*, dans le cadre du projet AIP Analyse rétrospective sur les devenir des variétés de colza 0 et 00 dans l'environnement, 2001