

Pierre Chevalier Géologue EI
Mesté Duran
32100 Condom
Tél+fax: 09 75 27 45 62
05 62 28 06 83
06 22 08 23 99
<pierrechevaliergeol@free.fr>

Auteur: P. Chevalier
Date: 05/12/2014

Documentation de la base de données d'exploration minière BDEXPLO

Table des matières

1. Historique.....	2
2. Structure de la base BDEXPLO.....	3
2.1. Schémas.....	3
2.2. Table operations.....	4
2.3. Ouvrages: sondages et tranchées.....	5
2.3.1. Table dh_collars.....	5
2.3.2. Table shift_reports.....	5
2.3.3. Table dh_followup.....	6
2.3.4. Table dh_quicklog.....	6
2.3.5. Table dh_core_boxes.....	6
2.3.6. Table dh_tech.....	6
2.3.7. Table dh_devia.....	7
2.3.8. Table dh_litho.....	7
2.3.9. Table dh_struct_measures.....	7
2.3.10. Table dh_density.....	8
2.3.11. Table dh_sampling_grades.....	8
2.3.12. Table dh_mineralised_intervals.....	9
2.3.13. Table dh_sampling_bottle_roll.....	9
2.3.14. Table dh_thinsections.....	9
2.4. Résultats analytiques, suivi des échantillons.....	9
2.4.1. Table lab_ana_batches_expedition.....	9
2.4.2. Table lab_ana_batches_reception.....	10
2.4.3. Table lab_ana_results.....	10
2.4.4. Table lab_ana_columns_definition.....	10
2.5. Contrôle analytique, contrôle qualité.....	11
2.5.1. Table qc_sampling.....	11
2.5.2. Tables qc_standards et lex_standard.....	11
2.5.3. Table lab_ana_qaqc_results.....	11
2.6. Traçabilité des données.....	12
2.6.1. Table lex_datasource.....	12
2.7. Observations de terrain.....	12
2.7.1. Table field_observations.....	12
2.7.2. Table field_observations_struct_measures.....	12
2.7.3. Table field_photos.....	13
2.8. Pré-exploitation.....	13
2.8.1. Table grade_ctrl.....	13
2.9. Géophysique au sol.....	13
2.9.1. Table gpy_mag_ground.....	13
2.10. Prélèvements de surface: prospection, géochimie ruisseau, sol.....	13
2.10.1. Table surface_samples_grades.....	13
2.10.2. Table geoch_sampling.....	14
2.10.3. Table geoch_ana.....	14
2.10.4. Table geoch_sampling_grades.....	15

2.11. Permis, prospects, indices, données générales.....	15
2.11.1. Table licences.....	15
2.11.2. Table occurrences.....	15
2.11.3. Table ancient_workings.....	16
2.11.4. Table mag_declination.....	16
2.11.5. Table index_geo_documentation.....	16
2.11.6. Table lex_codes.....	16
2.11.7. Table formations_group_lithos.....	16
2.11.8. Table baselines.....	17
2.11.9. Table topo_points.....	17
2.12. Tables de PostGIS.....	17
2.12.1. Table geometry_columns.....	17
2.12.2. Table spatial_ref_sys.....	17
3. Procédures principales pour l'entretien de la base de données.....	18
3.1. Connexion à la base de données.....	18
3.1.1. Se connecter au serveur via le middleware ODBC.....	18
3.1.2. Se connecter au serveur pour travail en ligne de commandes.....	20
3.2. Traçabilité des données.....	21
3.2.1. Création d'une nouvelle source de données (datasource).....	22
3.2.2. Suppression de tous les enregistrements correspondant à un datasource.....	22
3.3. Import de données analytiques.....	23
3.3.1. Import d'un fichier.....	23
3.3.2. Import d'une série de fichiers.....	24
3.4. Import d'un fichier .csv quelconque dans une table temporaire.....	25
3.5. Calculs de passes minéralisées.....	25
3.6. Vérifications de cohérence des données.....	27
3.7. Création des vues standard.....	28
3.8. Création d'un compte utilisateur sur le système et sur la base.....	28
3.9. Traitement d'un ensemble de commandes SQL par lot (batch).....	28
3.10. Documentation sur le moteur de base.....	28
3.11. Sauvegarde intégrale de la base de données.....	28
3.12. Sauvegarde des tables principales de la base en .csv.....	29
3.13. Vues cartographiques.....	29
3.14. Utilitaires, trucs et astuces.....	30
3.15. Récupérer les dernières versions des programmes.....	32

Ceci est une documentation succincte de la base BDEXPLO; la documentation de référence sera maintenue en ligne prochainement.

Il s'agit d'une base de données relationnelle très classique, servie par un serveur PostgreSQL 9.0, lui-même tournant sur un serveur GNU/Linux Debian. Son implémentation est placée sous licence GPL. La version de référence est sur github.

1. Historique

La structure de cette base de données a été conçue de manière très pragmatique, vers le début des années 1990, dans le cadre de projets d'exploration minière.

Elle a d'abord été implémentée sous forme de fichiers ascii simples s'interfaçant par le biais de programmes maison avec des logiciels tiers (sermine, techbase), puis sous la forme d'une base de données réelle sur dBaseIII+ puis dBaseIV. Elle était utilisée en production sur les projets de Yamfo, Subenso, Kenyase, avec des interfaces externes en dBase, Q+E, MS-Access, et GDM comme logiciel métier. Elle était développée et utilisée en même temps que GBis.

Elle a ensuite été utilisée sur le projet Tasiast, où l'interface et la base ont été migrées en MS-Access, avec gestion des accès concurrents pour une utilisation réseau multi-utilisateurs.

Enfin, elle a été reprise à Hassaï, au Soudan, avec un moteur PostgreSQL cette fois, après une tentative non convaincante en MySQL¹. Elle a ensuite été utilisée dans le cadre d'autres projets d'exploration à des stades stratégiques à tactiques.

Depuis lors, elle est développée de manière libre, placée sous licence GPL, octroyant ainsi toute liberté à

¹ Les tests effectués ont montré une vitesse un millier de fois plus grande, pour exactement les mêmes données et les mêmes requêtes, entre MySQL (plus lent) et PostgreSQL (plus rapide).

l'utilisateur.

En 2010, elle a été utilisée dans le cadre d'opérations faites pour la SMI, afin de regrouper et vérifier de nombreuses données dispersées dans des tableaux divers, des .mdb, des .hed-.ind-.bsd, des .b2d, des .xls, .xlw, etc.

Elle a été utilisée en production pour sortir tous les documents techniques du rapport de la campagne d'exploration SMI 2007-2008-2009, en liaison directe avec GDM. Elle a ensuite été utilisée en 2011 pour la révision du modèle géologique de Mont-Itzy, puis pour mettre au point le programme initial de délimitation de Mont-Itzy. Ensuite, dès le début de la campagne d'exploration en cours, fin 2011 - début 2012, elle a été utilisée en production, permettant le suivi quotidien de la campagne, avec production de cartes, coupes et logs, tous ces documents étant maintenus à jour en temps réel.

Courant 2012, la base a été cédée à la SMI, avec le matériel préconfiguré: il s'agissait de la machine qui a fait tous les travaux depuis 2011. Les données des autres opérations que la SMI ont été effacées à ce moment, et la base compactée, par souci de confidentialité.

C'est une base multi-opérations (multi-projets), qui permet aujourd'hui de gérer en parallèle 19 opérations totalement indépendantes au quotidien, dans la version de référence de la base.

La base était initialement implémentée en anglais, puis elle fut traduite en Français. Les développements se font maintenant en suivant les règles de bonnes pratiques de développement logiciel, ce qui implique l'emploi de l'anglais dans les sources. L'implémentation de schémas externes permettant un support multi-lingue transparent est prévue.

2. Structure de la base BDEXPLO

2.1. Schémas

Liste des schémas:

```
--duran bdxplo=>
\dn+
```

Nom	Description
public	standard public schema
checks	Views selecting inconsistent, incoherent, unprobable data
input	Tables with same structure as in public schema, for data input before validation and dump into final tables (apparently unused on 2013_08_03_11_40_18)
stats_reports	Views with statistics and reports, for daily/weekly/monthly statistics
tmp_imports	Schema used by external tools to import automatically .csv files in temporary tables
pierre	} (Schémas utilisateur)
tanguy	
kalvin	
zz_poubelle	Junk tables; just a safe recycle bin
pg_catalog	system catalog schema
information_schema	
pg_toast	reserved schema for TOAST tables
pg_toast_temp_1	

Dans le schéma public² se trouvent toutes les tables contenant les données:

Liste des relations		
Schéma	Nom	Description
public	operations	operations, projects, operator or client name
public	dh_collars	Drill holes collars or trenches starting points
public	dh_shift_reports	Daily reports from rigs/sites: one report per shift/machine/tool
public	dh_followup	Simple table for daily drill holes followup
public	dh_quicklog	Quick geological log, typically done on hole finish, for an A4 log plot
public	dh_core_boxes	Core drill holes boxes
public	dh_tech	Technical drilling data, and geotechnical parameters
public	dh_devia	Drill holes or trenches deviations measurements
public	dh_litho	Drill holes or trenches geological descriptions
public	dh_struct_measures	Structural measurements done on core, or in trenches
public	dh_density	Density measurements along drill holes or trenches
public	dh_sampling_grades	Samples along drill holes and trenches, with grades
public	dh_mineralised_intervals	Drill holes mineralised intercepts: stretch values over mineralised intervals, along drill holes or trenches
public	dh_sampling_bottle_roll	Mineralurgical samples, bottle-roll tests results
public	dh_thinsections	Thin sections for petrological studies
public	lab_ana_batches_expedition	Batches of samples sent for analysis
public	lab_ana_batches_reception	Batches of samples results received from laboratory
public	lab_ana_results	Laboratory results table, after laboratory instructions, related to LIMS system
public	lab_ana_columns_definition	Definition of columns; obsolete (unused: to be removed)
public	qc_sampling	Quality control samples, duplicates, blanks, standards
public	qc_standards	Quality control standard samples, most of them are CRM (Certified Reference Materials)
public	lex_standard	table contenant les valeurs des standards or et multi elements
public	lab_ana_qac_results	Quality control assay results, internal to analytical laboratory
public	lex_datasource	Lexicon of data sources, keeping track of imported file, for reference
public	field_observations	Field observations: geological observations, on outcrops, floats, or any other observations; coherent with GeolPDA
public	field_observations_struct_measures	Structural measurements, related to an observation; coherent with GeolPDA
public	field_photos	Photographs taken in field, related to an observation

² Notion différente de la notion de public ou privé; c'est le schéma par défaut dans postgresql

public	grade_ctrl	Grade-control samples during mining exploitation
public	gpy_mag_ground	Geophysics: ground mag
public	surface_samples_grades	Ponctual samples taken from surface: stream sediments, alluvial sediments, till, soils, termite mounds, rock outcrops, floats, etc. with grades
public	geoch_sampling	Geochemistry samples, from soil or stream sediments, location and description
public	geoch_ana	Assay results from geochemistry samples
public	geoch_sampling_grades	Geochemistry samples with grades; table inherits from geoch_sampling
public	licences	Licences, tenements
public	occurrences	Occurrences table: targets, mines, showings, deposits, mines. Compiled from various tables, and updated.
public	ancient_workings	Ancient workings, either historic or recent
public	mag_declination	Magnetic declination
public	index_geo_documentation	Index for any documentation, with lat-lon rectangles, so that any documentation may be accessed geographically
public	lex_codes	General look-up table with codes for various tables and coded fields
public	formations_group_lithos	Groups of lithologies, for simplification, typically for drill holes sections
public	baselines	Baselines, for each prospect, defined as a theoretical line between two points
public	topo_points	topographical data
public	geometry_columns	
public	spatial_ref_sys	

Note: les deux dernières tables listées, geometry_columns et spatial_ref_sys, appartiennent à l'extension spatiale PostGIS, qui est installée dans la base.

On va maintenant reprendre l'ensemble de ces tables, en commençant par les données de sondages (et tranchées), et en suivant un ordre logique.

Les données sont contenues essentiellement dans les tables suivantes:

2.2. Table operations

La **table maîtresse** des données est la table des opérations; une base peut contenir plusieurs opérations, ou projets. La **clé unique** de la table est **opid**, à rappeler dans **toute** jointure:

Table « public.operations »	Type	Modificateurs	Description
opid	integer	non NULL Par défaut, nextval...	Operation identifier
operation	character varying(4)		Operation code
full_name	character varying(50)		Complete operation name
operator	character varying(50)		Operator: mining operator, exploration company, client name
year	integer		Year of operation activity
confidentiality	boolean	Par défaut, true	Confidentiality flag, true or false; default is true
lat_min	numeric(10,5)	non NULL	South latitude, decimal degrees, WGS84
lon_min	numeric(10,5)	non NULL	West longitude, decimal degrees, WGS84
lat_max	numeric(10,5)		North latitude, decimal degrees, WGS84
lon_max	numeric(10,5)		East longitude, decimal degrees, WGS84
comments	character varying(500)		
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
numauto	integer	non NULL Par défaut, nextval...	Automatic integer

Index :

"opid" PRIMARY KEY, btree (opid)

Référence par :

TABLE "public.dh_collars" CONSTRAINT "dh_collars_opid_fkey" FOREIGN KEY (opid) REFERENCES operations(opid) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE

INITIALLY DEFERRED

TABLE "public.mag_declination" CONSTRAINT "mag_declination_opid_fkey" FOREIGN KEY (opid) REFERENCES operations(opid) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE

INITIALLY DEFERRED

TABLE "public.occurrences" CONSTRAINT "occurrences_opid_fkey" FOREIGN KEY (opid) REFERENCES operations(opid) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE

INITIALLY DEFERRED

Contient des OID: non

Deux opérations sont présentes à la SMI, les opid 11 et 18:

```
--duran bdxplo=>
SELECT * FROM public.operations;
-[ RECORD 1 ]-----
opid          | 11
operation     | SMI
full_name     | Ity
operator      | Société des Mines d'Ity
year          | 2010
confidentiality | t
lat_min       | 6.70000
lon_min       | -8.20000
lat_max       | 7.00000
lon_max       | -7.90000
comments      | Exploration pour la SMI
-[ RECORD 2 ]-----
opid          | 18
operation     | SMI
full_name     | Ity
operator      | Société des Mines d'Ity
year          | 2012
confidentiality | t
lat_min       | 6.70000
lon_min       | -8.20000
lat_max       | 7.00000
lon_max       | -7.90000
comments      | Exploration pour la SMI, à partir de 2010, quand les analyses ont été faites à Bureau Veritas
```

Historiquement, le passage de l'un à l'autre correspond au changement de laboratoire d'analyses fait en 2010, avec le recours à Bureau Veritas pour l'opid 18.

Nota: une table operation_active peut se trouver dans le schéma utilisateur: elle permet de ne travailler que sur une ou des opération(s) sélectionnée(s), par le biais de vues dans le schéma utilisateur, vues qui sont homonymes des tables du schéma public. Ainsi, on peut utiliser toutes les requêtes³ sans aucun changement, en n'interagissant qu'avec la ou les opérations choisies.

2.3. Ouvrages: sondages et tranchées

Toutes les tables sont préfixées en dh_ pour Drill Hole.

2.3.1. Table dh_collars

La table dh_collars est la table maîtresse des ouvrages: les collets, ou têtes de sondages, avec la plupart des informations concernant les sondages qui y sont rapportées. Les tranchées sont considérées comme des sondages, leur "collet" est leur point de démarrage.

Table « public.dh_collars » colonne	Type	Modificateurs	Description
opid	integer	non NULL	Operation identifier
id	character varying(20)	non NULL	Full identifier for borehole or trench, may include zone code, type and sequential number
shid	character varying(10)		Short identifier: type - sequential number
location	character varying		Investigated area code, refers to occurrences table
profile	character varying(10)		Profile number
srid	integer		Spatial Reference Identifier, or coordinate reference system: see spatial_ref_sys from postgis extension
x	numeric(12,3)		X coordinate (Eastings), in coordinate system srid
y	numeric(12,3)		Y coordinate (Northings), in coordinate system srid
z	numeric(12,3)		Z coordinate
topo_survey_type	character varying		Topographic collar survey type: GPS, GPSD, geometry, theodolite, relative, computed from local coordinate system, etc.
azim_ng	numeric(10,2)		Hole or trench azimuth (°) relative to geographic North
azim_nm	numeric(10,2)		Hole or trench azimuth (°) relative to Magnetic North
dip_hz	numeric(10,2)		Drill hole or trench dip relative to horizontal (°)
dh_type	character varying(10)		Type: D for Diamond drill hole, R for RC drill hole, T for Trench, A for Auger drill hole
date_start	date		Work start date
date_completed	date		Work finish date
completed	boolean	Par défaut, false	True: completed; False: planned
contractor	character varying(20)		Drilling contractor
geologist	character varying		Geologist name
length	numeric(10,2)		Total length (m)
nb_samples	integer		Number of samples
comments	character varying		Comments
purpose	character varying	Par défaut, 'EXPLO':character v	Purpose of hole: exploration, delineation, estimation, grade control, etc
x_local	numeric(12,3)		Local x coordinate
y_local	numeric(12,3)		Local y coordinate
z_local	numeric(12,3)		Local z coordinate
accumsum	numeric(10,2)		Accumulation sum over various mineralised intervals intersected by drill hole or trench (purpose: quick visualisation on maps (at wide scale ONLY), quick ranking of interesting holes)
id_pject	character varying		PJ for Project identifier: provisional identifier; aka peg number
x_pject	numeric(10,3)		Planned x coordinate
y_pject	numeric(10,3)		Planned y coordinate
z_pject	numeric(10,3)		Planned z coordinate
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
dh_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"opid_id" PRIMARY KEY, btree (opid, id)			
"dh_collars_numauto_key" UNIQUE, btree (numauto)			
"dh_collars_id" btree (id)			
Contraintes de clés étrangères :			
"dh_collars_opid_fkey" FOREIGN KEY (opid) REFERENCES operations(opid) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
Référéncé par :			
TABLE "public.dh_devia" CONSTRAINT "dh_devia_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
TABLE "public.dh_litho" CONSTRAINT "dh_litho_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
TABLE "public.dh_sampling_grades" CONSTRAINT "dh_sampling_grades_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
TABLE "public.shift_reports" CONSTRAINT "shift_reports_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
Contient des OID: non			

Les données de cette table sont nécessaires et suffisantes pour faire des tracés cartographiques: les vues dh_collars_points et dh_traces_3d permettent de cartographier dynamiquement les points de sondages et leurs tracés en projection verticale, respectivement.

2.3.2. Table shift reports

Cette table contient les rapports journaliers. Ces informations servent aux requêtes de suivis quotidien, hebdomadaire, mensuel, et aux requêtes de contrôles d'attachements de sondeurs.

Table « public.shift_reports » colonne	Type	Modificateurs	Description
opid	integer	non NULL	Operation identifier
date	date		Date of drilling
shift	character varying(1)		Day or night shift
no_fichette	integer	non NULL	Number of fichette = field form filled on a shift and borehole basis
rig	character varying(50)		Name/id of drilling (or digging) machine
geologist	character varying(50)		Geologist(s) following the drill hole on the rig site, doing the logging.
time_start	time without time zone		Drilling starting time
time_end	time without time zone		Drilling ending time
id	character varying(20)		Drill hole identifier, must match collars.id field, e.g.UMA_R086
peg_number	character varying(5)		Peg number: provisional identifier/number; aka PJ for Project identifier
planned_length	numeric(10,2)		Length of the borehole, as initially planned
tool	character varying(20)		Drilling (digging) tool/size, diameter: RC, RAB, percussion, core, SQ, PQ, HQ, NQ, BQ, AQ, mechanical shovel, hand shovel, banka, etc.
drilled_length_during_shift	numeric(10,2)		Length of borehole drilled during the shift
drilled_length	numeric(10,2)		Total length of the borehole drilled at the end of the shift
completed	boolean		Borehole finished or not
profile	character varying(10)		Section identifier
comments	character varying(254)		Comments on drilling (events, presence of water, difficulties, major facies, etc.)
invoice_nr	integer		Subcontractor invoice number
drilled_shift_destr	numeric		Drilled length during shift in destructive
drilled_shift_pq	numeric		Drilled length during shift in PQ core
drilled_shift_hq	numeric		Drilled length during shift in HQ core

3 Sans mention de schéma, bien entendu. Au passage, on note que lors de cette mission, les travaux étaient des travaux "lourds", en aucun cas une maintenance "standard": les requêtes sont le plus souvent exprimées (cf. annexe) avec mention explicite des schémas. C'est également nécessaire tant que des ON ... INSTEAD ne sont pas implémentées dans les vues homonymes des tables.

drilled_shift_nq	numeric		Drilled length during shift in NQ core
recovered_length_shift	numeric		Recovered length during shift
stbby_time1_h	numeric		Standby time hours, with machine powered on
stbby_time2_h	numeric		Standby time hours, with machine powered off
stbby_time3_h	numeric		Standby time hours, due to weather conditions
moving_time_h	numeric		Moving time hours
driller_name	character varying		Driller supervisor name
geologist_supervisor	character varying		Geologist supervisor name
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"fichette_pkey" PRIMARY KEY, btree (opid, no_fichette)			
Contraintes de clés étrangères :			
"shift_reports_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
Contient des OID: non			

2.3.3. Table dh_followup

Cette table permet un suivi rapide de l'avancement des travaux d'une campagne de sondages ou de tranchées; il s'agit grosso modo d'un équivalent du classique "tableau au mur" où chaque géologue, prospecteur, chef d'équipe, etc. met à jour en temps réel l'avancement de ses travaux.

Table « public.dh_followup » Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	character varying		Identifier, refers to dh_collars
devia	character varying(3)		Deviation survey (x: done; xx: done, data entered; xxx: data verified)
quick_log	character varying(3)		Quick geological log, typically done on hole finish, for an A4 log plot (x: done; xx: done, data entered; xxx: data verified)
log_tech	character varying(3)		Core fitting, core measurement, meters marking, RQD, fracture counts, etc. (x: done; xx: done, data entered; xxx: data verified)
log_lith	character varying(3)		Full geological log (x: done; xx: done, data entered; xxx: data verified)
sampling	character varying(3)		Hole sampling (x: done; xx: done, data entered; xxx: data verified)
results	character varying(3)		Assay results back from laboratory (x: received; xx: entered; xxx: verified)
relogging	character varying(3)		Geological log done afterwards on mineralised intervals (x: done; xx: done, data entered; xxx: data verified)
beacon	character varying(3)		Beacon or any other permanent hole marker on field (PVC pipe, concrete beacon, cement, etc.) (x: done)
in_gdm	character varying(1)		Data exported to GDM; implicitly: data clean, checked by GDM procedures (x: done)
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"dh_followup_pkey" PRIMARY KEY, btree (numauto)			
Contraintes de clés étrangères :			
"dh_followup_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
Contient des OID: non			

2.3.4. Table dh_quicklog

Cette table contient un log rapide, typiquement fait à la fin d'un sondage, ou en marchant au long d'une tranchée, le but étant d'avoir une description rapide, permettant de sortir des logs résumés tenant sur une feuille A4 par exemple, sans délai.

Table « public.dh_quicklog » Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	text		Full identifier for borehole or trench
depfrom	numeric(10,2)		Interval beginning depth
depto	numeric(10,2)		Interval ending depth
description	character varying(254)		Quick geological description, logging wide intervals and/or only representative portions
oxid	character varying(4)		Oxidation state: 0: PO, U
alt	smallint		Alteration intensity: 0: none, 1: weak, 2: moderate, 3: strong
def	smallint		Deformation intensity: 0: none, 1: weak, 2: moderate, 3: strong
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"dh_quicklog_pkey" PRIMARY KEY, btree (numauto)			
Contraintes de clés étrangères :			
"dh_quicklog_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
Contient des OID: non			

2.3.5. Table dh_core_boxes

Cette table contient les informations des boîtes de carottes.

Table « public.dh_core_boxes » Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	character varying(20)		Identifier, refers to dh_collars
depfrom	numeric(10,2)		Core box contents beginning depth
depto	numeric(10,2)		Core box contents ending depth
box_number	integer		Core box number
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"dh_core_boxes_pkey" PRIMARY KEY, btree (numauto)			
Contraintes de clés étrangères :			
"dh_core_boxes_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
Contient des OID: non			

À partir de cette table, les numéros de caisses de carottes reportés sur les logs de sondages, comme c'était le cas pour les cahiers de logs des campagnes de 1997, 2000, 2001, sont particulièrement utiles quand il s'agit de revenir rapidement sur d'anciennes carottes entreposées en carothèque.

2.3.6. Table dh_tech

Cette table contient les levés techniques et géotechniques.

Table « public.dh_tech »

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	character varying(20)		Drill hole identification
depfrom	numeric(10,2)		Interval beginning depth
depto	numeric(10,2)		Interval ending depth
drillers_depto	numeric(10,2)		Driller end-of-run depth, as mentioned on core block
drilled_len	numeric(10,2)		Interval length
reco_len	numeric(10,2)		Recovery length
rgd_len	numeric(10,2)		Rock Quality Designation "length"
diam	character varying(10)		core diameter
core_loss_cm	integer		Core loss along drilled run
joints_description	character varying		Joints description: rugosity, fillings, etc.
nb_joints	integer		Count of natural joints along drilled run
comments	character varying		
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"dh_tech_pkey" PRIMARY KEY, btree (numauto)			
"id_depto_dh_tech" btree (id, depto)			
Contraintes de clés étrangères :			
"dh_tech_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
Contient des OID: non			

2.3.7. Table dh_devia

Cette table contient les mesures de déviations au long des sondages ou tranchées. Pour le cas des tranchées, ce peuvent être soit de "vraies" mesures faites à la boussole/clinomètre, ou ce peuvent être des valeurs recalculées à partir de coordonnées relevées à la sole.

Cela permet de traiter de manière identique différents types d'ouvrages.

L'orientation initiale d'un sondage n'est pas enregistrée dans cette table, mais dans la table des têtes.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	character varying(20)		Drill hole identification related to the collars table
depto	numeric(10,2)		Depth of deviation measurement
device	character varying		Device used for deviation measurement
date	date		Date of deviation measurement
time	integer		Time of deviation measurement
azim_ng	numeric(10,2)		Hole azimuth (°) relative to geographic North
azim_nm	numeric(10,2)		Hole azimuth (°) relative to magnetic North (?)
dip_hz	numeric(10,2)		Drill hole dip relative to horizontal (°), positive down
temperature	numeric(10,2)		temperature
magnetic	numeric(10,2)		Magnetic field intensity measurement
roll	numeric(10,2)		Roll angle
comments	character varying		Various comments; concerning measurements done with Reflex Gyro, all parameters are concatenated as a json-like structure
valid	boolean	Par défaut, true	True when a deviation measurement is usable; queries should take into account only valid records
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"dh_devia_pkey" PRIMARY KEY, btree (numauto)			
Contraintes de clés étrangères :			
"dh_devia_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
Contient des OID: non			

2.3.8. Table dh_litho

La table dh_litho contient les descriptions géologiques. Les codes reprennent les conventions efficaces du BRGM, à savoir une codification sur 4 caractères alphanumériques majuscules.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	character varying(20)		Identifier, refers to dh_collars
depfrom	numeric(10,2)		Interval beginning depth
depto	numeric(10,2)		Interval ending depth
description	character varying		Geological description, naturalist style
description1	character varying		Complement to main geological description: metallic minerals
description2	character varying		Complement to main geological description: alterations
code1	character varying(4)		Conventional use is lithology code, 4 characters, uppercase. Refer to lex_codes table
code2	character varying(4)		Conventional use is supergene oxidation, 1 character, uppercase. Refer to lex_codes table
code3	character varying(4)		Conventional use is stratigraphy code, 4 characters, uppercase. Refer to lex_codes table
code4	character varying(4)		4 characters code. Refer to lex_codes table
value1	integer		Integer value. Refer to lex_codes table
value2	integer		Integer value. Refer to lex_codes table
value3	integer		Integer value. Refer to lex_codes table
value4	integer		Integer value. Refer to lex_codes table
value5	integer		Integer value. Refer to lex_codes table
value6	integer		Integer value. Refer to lex_codes table
colour	character varying		
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"dh_litho_pkey" PRIMARY KEY, btree (numauto)			
"dh_litho_id_depto" btree (id, depto)			
Contraintes de clés étrangères :			
"dh_litho_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED			
Contient des OID: non			

Les champs génériques code1 - code4 et value1 - value6 sont définis pour chaque opération (opid) dans la table

2.3.9. Table dh_struct_mesures

Mesures structurales: les mesures d'éléments structuraux peuvent être enregistrées soit en suivant les conventions usuelles, pour des carottes remises en situation, ou des mesures en tranchées, soit avec les angles mesurés au kénomètre par exemple sur des carottes.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	character varying(20)		Full identifier for borehole or trench

depto	numeric(10,2)		Measurement depth
measure_type	character varying		Type of measurement: [P: plane L: line PL: plane line PLM: plane line movement PLMS: plane line movement sure]
structure_type	character varying		Measured structure type: [VEIN, FRACTURE, C, SCHISTOSITY, FOLIATION, MYLONITE, CONTACT, VEIN_FAULT, FOLD_PAX_AX, FOLIATION_LINE, FAULT, CATACLASE, MINERALISED_STRUCTURE]
alpha_tca	numeric		Alpha angle = To Core Axis (TCA) angle, measured on core
beta	numeric		Beta angle
gamma	numeric		Gamma angle
north_ref	character varying		North reference for azimuths and directions measurements: [Nm: magnetic North, Ng: geographic North, Nu: UTM north, Nl: local grid Y axis]
direction	integer		Plane direction, 0-180°
dip	integer		Plane dip, 0-90°
dip_quadrant	character varying		Plane dip quadrant, NESW
pitch	integer		Pitch of line on plane, 0-90°
pitch_quadrant	character varying		Quadrant of pitch, NESW
movement	character varying		Relative movement of fault/c: [N: normal, I: inverse = R = reverse, D: dextral, S: sinistral]
valid	boolean		Measure is valid or not (impossible cases = not valid)
struct_description	character varying		Naturalist description of measured structure
sortgroup	character(1)		Sorting group, for discriminated of various phases: a, b, c, ...
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record

Index :
 "dh_struct_measures_pkey" PRIMARY KEY, btree (numauto)
 Contraintes de clés étrangères :
 "dh_struct_measures_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED
 Contient des OID: non

2.3.10. Table dh_density

Mesures de densité.

Table « public.dh_density »	Colonne	Type	Modificateurs	Description
opid	integer			Operation identifier
id	character varying(20)			Identifier, refers to dh_collars
depfrom	numeric(10,2)			Interval beginning depth: if not empty, density measured along an interval; otherwise, density measured on a point
depto	numeric(10,2)			Interval ending depth: if depfrom is empty, depth of poncutal density measurement
density	numeric(10,2)			Density, unitless, or considered as kg/l, or t/m3
density_humid	numeric			Density, unitless, or considered as kg/l, or t/m3, determined on humid sample
moisture	numeric			Moisture contents
method	character varying			Procedure used to determine specific gravity
datasource	integer			Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...		Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()		Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()		User (role) which created data record

Index :
 "dh_density_pkey" PRIMARY KEY, btree (numauto)
 Contraintes de clés étrangères :
 "dh_density_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED
 Contient des OID: non

Initialement, cette table avait une structure beaucoup plus compliquée, elle accueillait les valeurs brutes mesurées; en fait, il est rapidement apparu que c'était inutile, du fait du nombre de différentes méthodes de calcul de la densité, et qu'il valait mieux garder le résultat de la méthode, en mentionnant explicitement la méthode employée.

2.3.11. Table dh_sampling_grades

Les échantillons au long des sondages, avec les teneurs (qui peuvent être des moyennes, ou la première valeur, ou la moyenne d'un seul des laboratoires, etc. au choix du géologue responsable).

Table « public.dh_sampling_grades »	Colonne	Type	Modificateurs	Description
opid	integer			Operation identifier
id	character varying(20)			Identifier, refers to dh_collars
depfrom	numeric(10,2)			Sample beginning depth
depto	numeric(10,2)			Sample ending depth
core_loss_cm	numeric(5,1)			Cumulated core loss over sampled interval, in cm
weight_kg	numeric(6,2)			Sample weight kg
sample_type	character varying(8)			Sample type: DD: core sample (diamond drill), RC: percussion drilling Reverse Circulation sample, NS: not sampled, CS: channel sample
sample_id	character varying(20)			Sample identifier: refers to assay results and quality check tables
batch_id	integer			Batch identifier: refers to batch submission table: lab_ana_batches_expedition
au_specks	integer			Number of gold specks seen in drill hole or trench; typically, after panning destructive drilling chips, also gold specks seen in core drilling
au1_ppm	numeric(8,3)			Au grade 1; cf. usage definition in lex_codes for opid
au2_ppm	numeric(8,3)			Au grade 2; cf. usage definition in lex_codes for opid
au3_ppm	numeric(8,3)			Au grade 3; cf. usage definition in lex_codes for opid
au4_ppm	numeric(8,3)			Au grade 4; cf. usage definition in lex_codes for opid
au5_ppm	numeric(8,3)			Au grade 5; cf. usage definition in lex_codes for opid
au6_ppm	numeric(8,3)			Au grade 6; cf. usage definition in lex_codes for opid
ph	numeric(4,2)			pH measurement (for acidic ores)
moisture	numeric(8,4)			Moisture content (for percussion drilling samples mainly)
comments	character varying			Free comments, if any
quartering	integer			Sample quartering, if any (for percussion drilling samples split on site, mainly)
datasource	integer			Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...		Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()		Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()		User (role) which created data record

Index :
 "dh_sampling_grades_pkey" PRIMARY KEY, btree (numauto)
 Contraintes de clés étrangères :
 "dh_sampling_grades_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED
 Contient des OID: non

Cette table ne contient pas les résultats bruts des analyses chimiques, qui sont dans la table lab_ana_results. Des champs de teneurs sont rajoutés, au besoin, pour les éléments recherchés (métaux de base, etc.); initialement, il y avait pléthore de champs: n'ont été gardés que les champs utiles pour le moment à la SMI⁴.

4 Historiquement, cette table était héritée de dh_sampling, qui ne contenait que les données d'échantillonnage, alors que les données analytiques étaient dans lab_ana_results. Les teneurs étaient déterminées par requête à chaque fois. Mais l'arrivée massive de lots de données où il n'y avait aucune donnée purement analytique, mais seulement des données d'échantillons avec des teneurs (qui avaient pu être moyennées ou choisies) a fait plutôt pencher pour cette architecture, qui correspond mieux à la réalité des données initiales.

2.3.12. Table dh_mineralised_intervals

Cette table contient les intervalles minéralisés, avec les résultats des calculs de passes composites.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	character varying(20)		Full identifier for borehole or trench
depfrom	numeric(10,2)		Mineralised interval starting depth
depto	numeric(10,2)		Mineralised interval ending depth
mine	integer	Par défaut, 1	Take-out interval class: 1=normal interval, 2=high-grade interval
avau	numeric(10,2)		Average grade (g/t)
stva	character varying(150)		Stretch value, X m at Y g/t
accu	numeric(10,2)		Accumulation in m.g/t over mineralised interval
recu	numeric(10,2)		recovery
dens	numeric(10,2)		density
comments	character varying(100)		
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record

Index :
 "dh_mine_pkey" PRIMARY KEY, btree (numauto)
 "id_depto_dh_mine" btree (id, depto)
 Contraintes de clés étrangères :
 "dh_mineralised_intervals_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED
 Contient des OID: non

Les intervalles peuvent être définis à la main, ou automatiquement par le programme `procedure_genere_passmin.py` qui "fabrique" du SQL. Les passes peuvent être retouchées à la main, après la génération automatique, ce qui est le plus recommandé. Leur recalcul se fait alors par le programme `procedure_update_passmin.sh` (lequel ne fait qu'appeler la commande SQL `procedure_update_passmin.sql`).

2.3.13. Table dh_sampling_bottle_roll

Cette table contient les résultats d'essais minéralurgiques de type test-bouteille.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	text		Identifier, refers to dh_collars
depfrom	numeric(10,2)		Sample beginning depth
depto	numeric(10,2)		Sample ending depth
sample_id	character varying		Sample identifier: refers to assay results and quality check
au_total	numeric(10,2)		Total gold recovered
au_24h	numeric(10,2)		Gold recovered after 24 hours
au_48h	numeric(10,2)		Gold recovered after 48 hours
au_72h	numeric(10,2)		Gold recovered after 72 hours
au_residu	numeric(10,2)		Residual gold
rec_24h_pc	numeric(10,2)		Recovery after 24 hours, percent
rec_48h_pc	numeric(10,2)		Recovery after 48 hours, percent
rec_72h_pc	numeric(10,2)		Recovery after 72 hours, percent
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record

Index :
 "dh_sampling_bottle_roll_pkey" PRIMARY KEY, btree (numauto)
 Contraintes de clés étrangères :
 "dh_sampling_bottle_roll_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED
 Contient des OID: non

2.3.14. Table dh_thinsections

Cette table résume les résultats de diagnostics de lames minces.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	character varying		Identifier, refers to dh_collars
depto	numeric(10,2)		Sample taken for thin section: bottom depth
core_quarter	character varying		Optional code to identify which core quarter was taken to make thin section; useful for oriented core
questions	character varying		Interrogations concerning sample; desired diagnose
name	character varying		Result of diagnose: rock name
texture	character varying		Result of diagnose: texture
mineralogy	character varying		Result of diagnose: mineralogy
metamorphism_deformations	character varying		Result of diagnose: metamorphism and/or deformations
mineralisations	character varying		Result of diagnose: mineralisations
origin	character varying		Result of diagnose: origin: in case of highly transformed rock, protore
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in tabl
username	character varying	Par défaut, "current_user"()	User (role) which created data record
datasource	integer		Datasource identifier, refers to lex_datasource

Index :
 "dh_thinsections_pkey" PRIMARY KEY, btree (numauto)
 Contraintes de clés étrangères :
 "dh_thinsections_opid_fkey" FOREIGN KEY (opid, id) REFERENCES public.dh_collars(opid, id) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED
 Contient des OID: non

2.4. Résultats analytiques, suivi des échantillons

2.4.1. Table lab_ana_batches_expedition

Cette table contient les données relatives aux lots d'échantillons envoyés pour analyse chimique. Elle peut suffire à faire le suivi des lots d'échantillons, si tant est que le laboratoire d'analyses conserve les lots entre les expéditions et les résultats analytiques⁵.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
batch_id	integer		Batch identifier; recommended is 7-digit number, year and sequential number on 3 digits
labname	character varying(50)		Destination assay laboratory name

⁵ Ce n'est pas toujours le cas.

expedition_id	character varying(20)		Identifier of expedition (this is usually useless, if batches correspond to expeditions)
order_id	character varying(20)		Order identifier (DA number)
description	character varying		Quick description of samples: rocks, soils, core, chips, rocktypes if relevant, etc.
preparation	character varying(10)		Preparation of samples prior to expedition to lab (crushing, grinding, splitting, etc.)
process_labo	character varying(10)		Required preparation of samples in laboratory
scheme	character varying		Required assay scheme
shipment_date	date		Date of batch expedition to laboratory
sent_to_lab	boolean		Boolean: batch sent to laboratory or not
reception_date	date		Date of batch received
results_received	boolean		Boolean: results received for this batch (useful if laboratory returns results according to expedition batches (recommended); irrelevant otherwise)
lab_batches	character varying		List of laboratory batches, if any; useless if laboratory batches correspond to expedition batches
comments	character varying		Specific comments, reason for assay (control re-assay, re-sampling, routine, etc.)
samples_amount	integer		Number of samples
sample_id_first	character varying		First sample identifier; only relevant if samples in sequence
sample_id_last	character varying		Last sample identifier; only relevant if samples in sequence
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"lab_ana_batches_expedition_pkey" PRIMARY KEY, btree (numauto)			
Contient des OID: non			

2.4.2. Table lab ana batches reception

Cette table contient les métadonnées associées à un lot de données d'analyses reçues du laboratoire d'analyses. Cette table est remplie automatiquement par les scripts d'imports de données analytiques.

Table « public.lab_ana_batches_reception »	Colonne	Type	Modificateurs	Description
opid	integer			Operation identifier
labname	character varying			As in files received from laboratory: full laboratory name
client	character varying			As in files received from laboratory: client name
validated	date			As in files received from laboratory: validation date
jobno	character varying			As in files received from laboratory: job number
number_of_samples	integer			As in files received from laboratory: number of samples
project	character varying			As in files received from laboratory: project name
shipment_id	character varying			As in files received from laboratory: shipment id
p_o_number	character varying			As in files received from laboratory: P.O. number
received	date			As in files received from laboratory: reception date
certificate_comments	character varying			As in files received from laboratory: certificate comments
generic_txt	character varying			Generic text, containing information from original results file as is, unformatted
info_suppl_json	character varying			Supplementary information, serialised as a JSON (validated by json_xs)
datasource	integer			Datasource identifier, refers to lex_datasource
db_update_timestamp	timestamp without time zone		Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying		Par défaut, "current_user"()	User (role) which created data record
numauto	integer		non NULL Par défaut, nextval...	Automatic integer primary key
Index :				
"lab_ana_batches_reception_pkey" PRIMARY KEY, btree (numauto)				
Contient des OID: non				

2.4.3. Table lab ana results

Cette table contient les résultats analytiques provenant des laboratoires d'analyses, stockés de la manière la plus générique, sous forme Entité-Attribut-Valeur (EAV). Cette structure est directement calquée sur la structure du LIMS d'Intertek, laboratoire avec lequel avait été développé un protocole très simple d'import automatique de données.

Le format est désormais directement issu du système LIMS employé chez ACME et Bureau Veritas, tel que défini avec M.Sidiki Fofana lors de cette mission.

Note: nous avons mis au point un protocole, avec Intertek Djakarta, qui permettait de nourrir directement la base bdxplo, à partir du système LIMS d'Intertek. Idéalement, il faudrait normaliser ce type d'échanges, de manière à ce que les bases de données puissent parler entre elles, et éviter des échanges de données passant par des fichiers .xls ou équivalents, parfois très hasardeux (vécu douloureux...). Des discussions ont été initiées dans ce sens avec d'autres laboratoires. Un protocole existe, le ADX (Analytical Data Exchange), il a été considéré: il s'agit de XML, très générique, mais très lourd à l'usage.

Table « public.lab_ana_results »	Colonne	Type	Modificateurs	Description
opid	integer			Operation identifier
labname	character varying(10)			Analytical laboratory
jobno	character varying(20)			jcsa.pro_job, --> Intertek JobNo (VarChar(20))
orderno	character varying(40)			pj.orderno, --> Client Order No (VarChar(40))
sample_id	character varying(40)			Sample Identifier
scheme	character varying(20)			pjcsa.sch.code, --> Scheme Code (Intertek Internal Code - which probably reported to Client as well) (VarChar(20))
analyte	character varying(20)			pjcsa.analytecode, --> Analyte Code (Intertek Internal Code - which probably reported to Client as well) (VarChar(20))
value	character varying(20)			pjcsa.formattedvalue, --> Reported Value (VarChar(20))
value_num	numeric			Reported value, converted to numeric. IS becomes -999, LNR -9999, < -, > nothing
batch_id	integer			Batch identifier
sampletype	character varying			Sample type: DUP: duplicate, STD: standard, REP: repeat, etc.
unit	character varying			Unit: PPM, PPB, KG, G, %, etc.
sample_id_lab	character varying			pjc.sampleident, --> Client SampleID (VarChar(40))
valid	boolean			=> sometimes different from REAL sample_id
detlim	numeric		Par défaut, true	Analysis is considered as valid or ignored(if QAQC failed, for instance)
uplim	numeric			Lower detection limit
datasource	integer			Upper limit
db_update_timestamp	timestamp without time zone		Par défaut, now()	Datasource identifier, refers to lex_datasource
username	character varying		Par défaut, "current_user"()	Current date and time stamp when data is loaded in table
numauto	integer		non NULL Par défaut, nextval...	User (role) which created data record
Triggers :				
lab_ana_results.insert AFTER INSERT ON public.lab_ana_results FOR EACH STATEMENT EXECUTE PROCEDURE lab_ana_results_sample_id_default_value_num()				
Contient des OID: non				

2.4.4. Table lab ana columns definition

Cette table est désormais obsolète. Néanmoins, des héritages subsistent encore, qui interdisent de la supprimer pour le moment.

Table « public.lab_ana_columns_definition »

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
analyte	character varying(20)		Analyte: element, oxide, etc.
unit	character varying		Unit (PPM, PPB, etc.)
scheme	character varying(20)		Scheme: fire assay, ICP, etc.; laboratory scheme identifier
colid	text		Column identifier, used for groupings in cross-tab queries
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
Index :			
"lab_ana_columns_definition_pkey" PRIMARY KEY, btree (numauto)			
Contient des OID: non			

2.5. Contrôle analytique, contrôle qualité

2.5.1. Table qc sampling

Les échantillons du contrôle analytique: standards insérés, duplicatas, blancs.

Colonne	Type	Modificateurs	Description
sample_id	character varying(20)		
qc_type	character varying		
comments	character varying		
opid	integer		Operation identifier
batch_id	integer		
refers_to	character varying		
datasource	integer		Datasource identifier, refers to lex_datasource
weight_kg	numeric(6,2)		
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"qc_sampling_pkey" PRIMARY KEY, btree (numauto)			
"qc_sampling_sample_id" btree (sample_id)			
Contient des OID: non			

2.5.2. Tables qc standards et lex standard

qc_standards est la table de référence pour les échantillons standards.

Colonne	Type	Modificateurs	Description
qc_id	character varying(20)	non NULL	sample identification
labo	character varying(50)		sample laboratory
matrix	character varying(50)		sample composition
presentation	character varying(50)		sample presentation
au_ppm	numeric(10,3)		sample analysis value
cu_ppm	numeric(10,3)		sample analysis value
zn_ppm	numeric(10,3)		sample analysis value
pb_ppm	numeric(10,3)		sample analysis value
ag_ppm	numeric(10,3)		sample analysis value
ni_ppm	numeric(10,3)		sample analysis value
au_ppm_95pc_conf_interval	numeric		
cu_ppm_95pc_conf_interval	numeric		
zn_ppm_95pc_conf_interval	numeric		
pb_ppm_95pc_conf_interval	numeric		
ag_ppm_95pc_conf_interval	numeric		
ni_ppm_95pc_conf_interval	numeric		
opid	integer		Operation identifier
datasource	integer		Datasource identifier, refers to lex_datasource
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
Index :			
"qc_standards_pkey" PRIMARY KEY, btree (numauto)			
"lex_qc_id_key" UNIQUE, btree (qc_id)			
Contient des OID: non			

lex_standard est une autre table utilisée pour les échantillons standard; à terme, il faudra n'en garder qu'une.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
std_id	character varying	non NULL	
unit	character(5)	non NULL	
element	character(5)	non NULL	
value	numeric	non NULL	
std_dev	numeric		
interval_conf	numeric		
std_origin	character(25)		
type_analyse	character(25)	non NULL	
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"lex_standard_pkey" PRIMARY KEY, btree (numauto)			
Contient des OID: non			

2.5.3. Table lab ana qaqc results

Cette table contient les résultats de contrôles analytiques internes des laboratoires⁶.

Avec le nouveau format par lequel Bureau Veritas envoie ses résultats, cette table devient obsolète: il faudra verser les enregistrements dans lab_ana_results, à terme.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
jobno	character varying		Job number
generic_txt_col1	character varying		Generic text field, receiving any text from original file, as is
generic_txt_col2	character varying		Generic text field, receiving any text from original file, as is
generic_txt_col3	character varying		Generic text field, receiving any text from original file, as is
generic_txt_col4	character varying		Generic text field, receiving any text from original file, as is
generic_txt_col5	character varying		Generic text field, receiving any text from original file, as is
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"lab_ana_qaqc_results_pkey" PRIMARY KEY, btree (numauto)			
Contient des OID: non			

⁶ Généralement, ces contrôles sont bons... Je qualifierais cette table de "décoratrice".

2.6. Tracabilité des données

2.6.1. Table lex_datasource

La tracabilité des données importées dans la base est enregistrée dans la table `lex_datasource`, pour lexique des sources de données:

Table « public.lex_datasource »	Colonne	Type	Modificateurs	Description
	<code>opid</code>	integer		Operation identifier
	<code>filename</code>	character varying(500)		Data imported: file name with full path, to be kept for permanent reference
	<code>comments</code>	character varying		Various comments
	<code>datasource_id</code>	integer	non NULL Par défaut, nextval...	datasource field in various tables refer to this datasource_id field
	<code>db_update_timestamp</code>	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
	<code>username</code>	character varying	Par défaut, "current_user"()	User (role) which created data record
	<code>numauto</code>	integer	non NULL Par défaut, nextval...	Automatic integer primary key
Index :				
"lex_datasource_pkey" PRIMARY KEY, btree (numauto)				
Contient des OID: non				

Le champ "datasource" d'une table quelconque se réfère au champ `datasource_id` de ce lexique. Ceci est facultatif, les données entrées "à la main" via une interface utilisateur quelconque, ne font pas l'objet d'un `datasource` systématiquement, par exemple. En revanche, pour des données importées, en particulier si l'on ne maîtrise pas la qualité des données, il est capital de maintenir correctement cette table. Le champ `filename` contient le chemin et le nom de fichier importé, ce qui permet une tracabilité des données importées. L'arborescence de référence est de la forme:

/transferts/from/telle_personne/aaaa_mm_jj/(fichiers_de_données en l'état)

Cette convention est, à l'usage, la plus aisée et efficace pour retrouver rapidement des données transmises par des tiers.

L'ensemble des fichiers utilisés pour construire la base de données exploration de la SMI respecte cette arborescence; il a été remis en même temps que le serveur, en juillet 2012.

Des utilitaires permettent de générer les références dans cette table.

Toujours pour la tracabilité, dans toutes les tables, un champ `db_update_timestamp` et un champ `username` enregistrent automatiquement l'horodatage et le nom de l'utilisateur lors de l'insertion d'un enregistrement⁷.

2.7. Observations de terrain

2.7.1. Table field_observations

Observations de terrain⁸, par exemple des affleurements, ou toute autre observation.

Table « public.field_observations »	Colonne	Type	Modificateurs	Description
	<code>opid</code>	integer		Operation identifier
	<code>year</code>	integer		Year when observation is done (TODO DROP COLUMN redundant with date field)
	<code>obs_id</code>	character varying		Observation identifier: usually composed of: (acronym of person)_(year)_(incremental integer)
	<code>date</code>	date		Observation date
	<code>waypoint_name</code>	character varying		If relevant, waypoint name from GPS device
	<code>x</code>	numeric(20,10)		X coordinate (Easting), in coordinate system srid
	<code>y</code>	numeric(20,10)		Y coordinate (Northing), in coordinate system srid
	<code>z</code>	numeric(20,2)		Z coordinate
	<code>srid</code>	integer		Spatial Reference Identifier, or coordinate reference system: see spatial_ref_sys from postgres extension
	<code>description</code>	text		Naturalist description
	<code>code_litho</code>	character varying(4)		Lithological code
	<code>code_unit</code>	character varying(4)		Unit code: lithostratigraphic, and/or cartographic
	<code>geologist</code>	character varying		Geologist or prospector name
	<code>icon_descr</code>	character varying		If relevant, icon description from some GPS devices/programs
	<code>comments</code>	character varying		Comments
	<code>sample_id</code>	character varying		If relevant, sample identifier
	<code>photos</code>	character varying		List of photographs pictures files, if relevant
	<code>audio</code>	character varying		Audio recording files, if relevant
	<code>timestamp_epoch_ms</code>	bigint		Timestamp of observation: as defined in GeolPDA devices, as epoch in ms
	<code>datasource</code>	integer	non NULL Par défaut, nextval...	datasource identifier, refers to lex_datasource
	<code>numauto</code>	integer	Par défaut, now()	Automatic integer primary key
	<code>db_update_timestamp</code>	timestamp without time zone	Par défaut, "current_user"()	Current date and time stamp when data is loaded in table
	<code>username</code>	character varying		User (role) which created data record
Index :				
"field_observations_pkey1" PRIMARY KEY, btree (numauto)				
Contient des OID: non				

2.7.2. Table field_observations_struct_measures

Mesures structurales faites sur le terrain⁹, en liaison avec une observation. Les conventions utilisées sont du type français; à terme, le stockage ne se fera que sous forme de matrice de rotation, qui est la manière la plus générique de stocker une orientation spatiale.

Colonne	Type	Modificateurs	Description
<code>opid</code>	integer		Operation identifier
<code>obs_id</code>	character varying		Observation identifier: refers to field_observations table
<code>measure_type</code>	character varying		Type of measurement: [P: plane L: line PL: plane line PLM: plane line movement PLMS: plane line movement sure]
<code>structure_type</code>	character varying		Measured structure type: [VEIN, FRACTURE, C, SCHISTOSITY, FOLIATION, MYLONITE, CONTACT, VEIN_FAULT, FOLD_PAX_AX, FOLIATION_LINE, FAULT, CATACLASE, MINERALISED_STRUCTURE]
<code>north_ref</code>	character varying		North reference for azimuths and directions measurements: [Nm: magnetic North, Ng: geographic North, Nu: UTM north, Nl: local grid Y axis]
<code>direction</code>	integer		Plane direction, 0-180°

⁷ Ceci a été rajouté à la suite de la mission.

⁸ Les données venant d'un GeolPDA sont automatiquement importées dans cette table.

⁹ Les données venant d'un GeolPDA sont automatiquement importées dans cette table.

dip	integer		Plane dip, 0-90°
dip_quadrant	character varying		Plane dip quadrant, NESW
pitch	integer		Pitch of line on plane, 0-90°
pitch_quadrant	character varying		Quadrant of pitch, NESW
movement	character varying		Relative movement of fault/c: [N: normal, I: inverse = R = reverse, D: dextral, S: sinistral]
valid	boolean		Measure is valid or not (impossible cases = not valid)
comments	character varying		Comments
rotation_matrix	character varying		3x3 rotation matrix, fully describing any orientation: initial state: [X axis points East, Y axis points North, Z axis points up] => measurement state = rotation applied. Corresponds to function public static float[] getOrientation (Float[] R, float[] values) from android API as described in http://developer.android.com/reference/android/hardware/SensorManager.html#getOrientation%28Float[],%20Float[],%29
geolpda_id	integer		Datasource identifier, refers to lex_datasource
geolpda_poi_id	integer		Automatic integer primary key
datasource	integer		Current date and time stamp when data is loaded in table
numauto	integer	non NULL Par défaut, nextval...	User (role) which created data record
db_update_timestamp	timestamp without time zone	Par défaut, now()	
username	character varying	Par défaut, "current_user"()	
Index :			
"field_observations_struct_measures_pkey" PRIMARY KEY, btree (numauto)			
Contient des OID: non			

2.7.3. Table field_photos

Table des photographies prises sur le terrain, en liaison avec une observation.

Table « public.field_photos »	Colonne	Type	Modificateurs	Description
opid	integer			Operation identifier
pho_id	character varying		non NULL	
obs_id	character varying			
file	character varying			
description	character varying			
az	numeric			
dip	numeric			
author	character varying			
datasource	integer			Datasource identifier, refers to lex_datasource
numauto	integer		non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone		Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying		Par défaut, "current_user"()	User (role) which created data record
Index :				
"field_photos_id" PRIMARY KEY, btree (pho_id)				
Contient des OID: non				

2.8. Pré-exploitation

2.8.1. Table grade_ctrl

Il s'agit de prélèvements considérés comme ponctuels. À la SMI, les données de pré-exploitation étant des rainurages dans des tranchées en fond de fosse, considérés comme des sondages, les données sont dans les tables des ouvrages. Cette table est donc vide pour la SMI.

Table « public.grade_ctrl »	Colonne	Type	Modificateurs	Description
id	character varying(4)			Quarry and block identification in 4 characters
num	character varying(10)			sample number
x	numeric(10,2)			X drill hole collar coordinate, projected in UTM (m)
y	numeric(10,2)			Y drill hole collar coordinate, projected in UTM (m)
z	numeric(10,2)			Z drill hole collar coordinate, projected in UTM (m)
prof	numeric(10,2)			End of sample depth
aucy	numeric(10,2)			Sample cyanidable gold grade (g/t)
autot	numeric(10,2)			Total gold grade (g/t)
litho	character varying(20)			Sample lithology in GDM or Serpine code
old_id	character varying(20)			Quarry and block old identification
numauto	integer		non NULL Par défaut, nextval...	Automatic integer primary key
aucy2	numeric(10,2)			
datasource	integer			Datasource identifier, refers to lex_datasource
opid	integer			Operation identifier
db_update_timestamp	timestamp without time zone		Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying		Par défaut, "current_user"()	User (role) which created data record
Index :				
"preex_sampling_pkey" PRIMARY KEY, btree (numauto)				
"id_preex_sampling" btree (id)				
"xyz_preex_sampling" btree (x, y, z)				
Contient des OID: non				

2.9. Géophysique au sol

On n'a jusqu'ici jamais eu l'occasion d'intégrer d'autres données brutes que du mag au sol, c'est donc la seule table qui y figure.

2.9.1. Table gpy_mag_ground

Table « public.gpy_mag_ground »	Colonne	Type	Modificateurs	Description
opid	integer			Operation identifier
srld	integer			
x	numeric(10,2)			
y	numeric(10,2)			
z	numeric(10,2)			
x_local	numeric(10,2)			
y_local	numeric(10,2)			
mag_nanotesla	double precision			
val_corr_mag_nanotesla	double precision			
db_update_timestamp	timestamp without time zone		Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying		Par défaut, "current_user"()	User (role) which created data record
numauto	integer		non NULL Par défaut, nextval...	Automatic integer primary key
datasource	integer			Datasource identifier, refers to lex_datasource
Index :				
"gpy_mag_ground_pkey" PRIMARY KEY, btree (numauto)				
Contient des OID: non				

2.10. Prélèvements de surface: prospection, géochimie ruisseau, sol

2.10.1. Table surface samples grades

Table regroupant les échantillonnages ponctuels faits en surface, que ce soient des sédiments de ruisseau, des sols, des roches, etc. À l'usage, c'est une approche nettement plus efficace que l'approche séparant ces données.

De même que dh_sampling_grades, les teneurs sont dans cette table. Dans les jeux de données existants, il n'y a

pas de données analytiques brutes; à terme, les analyses doivent être séparées et rangées dans lab_ana_results, comme les analyses de sondages ou autres.

Concernant Ity, les données de cette table concernent que des sols et des sédiments de ruisseau.

Table « public.surface_samples_grades »			
Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
sample_id	character varying		
x	double precision		
y	double precision		
z	double precision		
srld	integer		
description	character varying		
sample_type	character varying		
outcrop_id	character varying		
trend	character varying		
dip	character varying		
length_m	character varying		
width_m	character varying		
au1_ppm	double precision		
au2_ppm	double precision		
ag1	double precision		
ag2	double precision		
cu1	double precision		
cu2	double precision		
as	double precision		
pb	double precision		
zn	double precision		
k2o	double precision		
ba	double precision		
si2	double precision		
al2	double precision		
fe2x	double precision		
mno	double precision		
tio2	double precision		
p2o5	double precision		
cao	double precision		
mgo	double precision		
mo	double precision		
sn	double precision		
sb	double precision		
w	double precision		
bi	double precision		
cr	double precision		
li	double precision		
b	double precision		
v	double precision		
cr	double precision		
ni	double precision		
co	double precision		
sr	double precision		
y	double precision		
la	double precision		
ce	double precision		
nb	double precision		
be	double precision		
cd	double precision		
spp2	double precision		
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"numauto_pkey" PRIMARY KEY, btree (numauto)			
Contient des OID: non			

2.10.2. Table geoch_sampling

Cette table d'échantillons de géochimie est obsolète¹⁰.

Table « public.geoch_sampling »			
Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	character varying(20)		Identification
lab_id	character varying(50)		Analysis laboratory
labo_ref	character varying(20)		Analysis laboratory report reference
amc_ref	character varying(20)		AMC analysis report reference
recep_date	date		Report reception date by AMC
type	character varying(20)		Analysis type
sampl_index	character varying	non NULL Par défaut, nextval...	Auto increment integer
x	numeric(15,4)		X coordinate, projected in UTM (m)
y	numeric(15,4)		Y coordinate, projected in UTM (m)
z	numeric(10,4)		Z coordinate, projected in UTM (m)
soil_color	character varying(50)		Soil color
type_sort	character varying(20)		Sort of type
depth_cm	numeric(10,2)		Sample depth
reg_type	character varying(20)		Type of region
geomorphology	character varying(20)		Some region description
rock_type	character varying(50)		Lithology
comment	character varying(200)		Some comments
utm_zone	character varying(3)		UTM area
geologist	character varying(20)		geologist
float_sampl	character varying(20)		sample designation (?)
host_rock	character varying(200)		host rock
prospect	character varying(20)		
spacing	character varying(20)		
horizon	character varying(20)		
datasource	integer		Datasource identifier, refers to lex_datasource
date	date		type of survey (ex : HHGPS)
survey_type	character varying(50)		
grid_line	character varying		
grid_station	character varying		
alteration	character varying		
occ_soil	character varying		
slope	character varying		
slope_dir	character varying		
soil_description	character varying		
numauto	integer	non NULL Par défaut, nextval...	Automatic integer
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"geoch_sampling_pkey" PRIMARY KEY, btree (sampl_index)			
"sampl_index_geoch_sampling" btree (sampl_index)			
Tables enfant :: public.geoch_sampling_grades			
Contient des OID: non			

2.10.3. Table geoch_ana

Idem que précédente.

Table « public.geoch_ana »			
Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
sampl_index	integer		Sample identification related to the geoch_sampling table

¹⁰ Cette table était utilisée chez AMC, et elle n'était conservée qu'en attendant que ces données soient transférées dans surface_samples_grades en sérialisant les champs

ana_type	character varying(20)		Analysis type
unit	character varying(10)		Unit of the analysis
det_lim	numeric(6,4)		Analysis detection limit
scheme	character varying(20)		Analysis method
comment	character varying(20)		Some comments
value	numeric(10,3)		Analysis value
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
datasource	integer		Datasource identifier, refers to lex_datasource
Index :			
"geoch_ana_pkey" PRIMARY KEY, btree (numauto)			
"saml_index_geoch_ana" btree (saml_index)			
Contient des OID: non			

2.10.4. Table geoch_sampling_grades

Idem que précédente. Attention, ceci est une table héritée de geoch_sampling.

Table « public.geoch_sampling_grades »	Colonne	Type	Modificateurs	Description
opid	integer			Operation identifier
id	character varying(20)			
lab_id	character varying(50)			
labo_ref	character varying(20)			
amc_ref	character varying(20)			
recep_date	date			
type	character varying(20)			
saml_index	character varying	non NULL Par défaut, nextval...		
x	numeric(15,4)			
y	numeric(15,4)			
z	numeric(10,4)			
soil_color	character varying(50)			
type_sort	character varying(20)			
depth_cm	numeric(10,2)			
reg_type	character varying(20)			
geomorphology	character varying(20)			
rock_type	character varying(50)			
comment	character varying(200)			
utm_zone	character varying(3)			
geologist	character varying(20)			
float_saml	character varying(20)			
host_rock	character varying(200)			
prospect	character varying(20)			
spacing	character varying(20)			
horizon	character varying(20)			
date	date			
survey_type	character varying(50)			
grid_line	character varying			
grid_station	character varying			
alteration	character varying			
occ_soil	character varying			
slope	character varying			
slope_dir	character varying			
soil_description	character varying			
au_ppb	numeric			
datasource	integer	non NULL Par défaut, nextval...		Datasource identifier, refers to lex_datasource
numauto	integer	Par défaut, now()		Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, "current_user"()		
username	character varying			
Index :				
"geoch_sampling_grades_pkey" PRIMARY KEY, btree (numauto)				
Héritée de: public.geoch_sampling				
Contient des OID: non				

2.11. Permis, prospects, indices, données générales

2.11.1. Table licences

Cette table contient les permis miniers, avec uniquement les encombrements en fenêtre latitude-longitude; la gestion des vertex, le tracé des permis, le suivi des permis dans le temps, etc., sont réservés à un autre ensemble de relations.

Table « public.licences »	Colonne	Type	Modificateurs	Description
opid	integer			Operation identifier
licence_name	character varying(50)			Licence official name, as reported on legal documents
operator	character varying(20)			Operator, owner of licence
year	integer	non NULL		Year when information is valid
lat_min	numeric(10,5)	non NULL		Minimum latitude
lon_min	numeric(10,5)	non NULL		Minimum longitude
lat_max	numeric(10,5)	non NULL		Maximum latitude
lon_max	numeric(10,5)	non NULL		Maximum longitude
comments	character varying(500)			Comments
db_update_timestamp	timestamp without time zone	Par défaut, now()		Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()		User (role) which created data record
numauto	integer	non NULL Par défaut, nextval...		Automatic integer
datasource	integer			Datasource identifier, refers to lex_datasource
Index :				
"licence_id" PRIMARY KEY, btree (id)				
Contient des OID: non				

2.11.2. Table occurrences

Table des indices, au sens large; également, suivi des avancements travaux. Elle peut être vue comme une table pense-bête à long terme.

Table « public.occurrences »	Colonne	Type	Modificateurs	Description
opid	integer			Operation identifier
name	character varying(50)			Occurrence name
status	character varying(10)			Status: OCCUR = occurrence ; OREB = orebody ;
description	character varying(254)			MINE = active mine ; MINED = exploited, depleted mine
				Occurrence description: geological context, significant figures at current stage of exploration or exploitation
w_done	character varying(254)			Exploration work done, codified field: PROSpection (rock sampling on surface), SOIL geochemistry, MAPping, DECAPage, TRENches, DRILL Holes
w_todo	character varying(254)			Exploration work to be done, codified field: PROSpection (rock sampling on surface), SOIL geochemistry, MAPping, DECAPage, TRENches, DRILL Holes
code	character varying(20)			
zone	character varying			
geom	geometry			
comments	character varying			
datasource	integer			Datasource identifier, refers to lex_datasource
numauto	integer			Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()		Current date and time stamp when data is loaded in table

```

username | character varying | Par défaut, "current_user"() | User (role) which created data record
Contraintes de vérification :
"chk_status" CHECK (status::text = ANY (ARRAY['OCCUR'::character varying, 'OREB'::character varying, 'MINE'::character varying, 'MINED'::character varying,
'MCO'::character varying, 'DISTRICT'::character varying]::text[]))
"enforce_dims_geom" CHECK (ndims(geom) = 2)
"enforce_geotype_geom" CHECK (geometrytype(geom) = 'POINT'::text OR geom IS NULL)
"enforce_srid_geom" CHECK (srid(geom) = 4326)
Contraintes de clés étrangères :
"occurrences_opid_fkey" FOREIGN KEY (opid) REFERENCES operations(opid) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED
Contient des OID: non

```

2.11.3. Table ancient workings

Table contenant les localisations d'anciens travaux miniers, les exploitations d'orpailleurs, diamineurs, etc.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
gid	integer	non NULL Par défaut, nextval...	Identifier
description	character varying(254)		Full description
the_geom	geometry		Geometry, used in GIS
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval('an	Automatic integer

Index :
 "ancient_workings_pkey" PRIMARY KEY, btree (gid)
 "ancient_workings_the_geom_gist" gist (the_geom)

Contraintes de vérification :
 "enforce_dims_the_geom" CHECK (ndims(the_geom) = 2)
 "enforce_geotype_the_geom" CHECK (geometrytype(the_geom) = 'POINT'::text OR the_geom IS NULL)
 "enforce_srid_the_geom" CHECK (srid(the_geom) = 4326)

Contient des OID: non

2.11.4. Table mag declination

Cette table contient les déclinaisons magnétiques pour les opérations; elle sera rapidement obsolète, et remplacée par un appel à igrf-2010.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
mag_decl	numeric		
date	date		
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record

Index :
 "mag_declination_pkey" PRIMARY KEY, btree (numauto)

Contraintes de clés étrangères :
 "mag_declination_opid_fkey" FOREIGN KEY (opid) REFERENCES operations(opid) ON UPDATE CASCADE ON DELETE CASCADE DEFERRABLE INITIALLY DEFERRED

Contient des OID: non

2.11.5. Table index geo documentation

Table géoréférençant des documents quelconques, permettant une consultation par SIG de divers documents (rapports, articles, mémos, croquis, etc.) à partir de leur rectangle d'encombrement. Cette table n'est pas utilisée à la SMI.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	integer	non NULL Par défaut, nextval...	
title	character varying(100)		
filename	character varying		
lat_min	numeric(20,8)		
lat_max	numeric(20,8)		
lon_min	numeric(20,8)		
lon_max	numeric(20,8)		
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record

Index :
 "index_geo_documentation_pkey" PRIMARY KEY, btree (numauto)

Contient des OID: non

2.11.6. Table lex codes

Cette table est un lexique général, où les valeurs codées de champs sont déréférencées, pour tous champs de toutes tables¹¹.

Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
tablename	text		
field	text		
code	character varying		
description	character varying		
comments	character varying		
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record

Index :
 "lex_codes_pkey" PRIMARY KEY, btree (numauto)

Contient des OID: non

2.11.7. Table formations group lithos

Cette table permet de regrouper des lithologies différentes en sur-ensembles plus simples. Typiquement, elle est utilisée avec dh_litho pour représenter des coupes, en générant par jointure des codes simples d'unités lithologiques homogènes, à partir de codes lithologiques qui peuvent être disparates et compliqués.

¹¹ L'intégrité référentielle est, pour le moment, impossible à implémenter sur cette table: ceci est dû à des données très disparates parmi les opérations.

Table « public.formation_group_lithos »			
Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
formation_name	character varying(25)		
code_litho	character varying(4)		
datasource	integer		
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"formations_group_lithos_pkey" PRIMARY KEY, btree (numauto)			
Contient des OID: non			

2.11.8. Table baselines

Lignes de base, typiquement utilisé pour des grilles de géochimie ou de sondages. Également utilisé pour la génération automatique de coupes sériées.

Table « public.baselines »/*{*/			
Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
id	integer		Baseline identifier
location	character varying		
x1	numeric(10,3)		Baseline starting point x coordinate
y1	numeric(10,3)		Baseline starting point y coordinate
z1	numeric(10,3)		Baseline starting point z coordinate
x2	numeric(10,3)		Baseline ending point x coordinate
y2	numeric(10,3)		Baseline ending point y coordinate
z2	numeric(10,3)		Baseline ending point z coordinate
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	Current date and time stamp when data is loaded in table
username	character varying	Par défaut, "current_user"()	User (role) which created data record
datasource	integer		Datasource identifier, refers to lex_datasource
Index :			
"baselines_pkey" PRIMARY KEY, btree (numauto)			
Contient des OID: non			

2.11.9. Table topo points

Données brutes de mesures de points, telles que fournies par les équipes de topographes. Cette table sert ensuite, par jointure, à comparer avec les coordonnées d'ouvrages et de les mettre à jour, éventuellement.

Table « public.topo_points »			
Colonne	Type	Modificateurs	Description
opid	integer		Operation identifier
location	character varying(20)		Topographical zone
num	numeric(10,0)		Topographical point number
x	numeric(10,3)		X coordinate, projected in UTM (m)
y	numeric(10,3)		Y coordinate, projected in UTM (m)
z	numeric(10,3)		Z coordinate, projected in UTM (m)
survey_date	date		
topo_survey_type	character varying		
coordsys	character varying		
surveyor	character varying		
id	character varying(20)		Full identifier for borehole or trench, including zone code with type and sequential number
datasource	integer		Datasource identifier, refers to lex_datasource
numauto	integer	non NULL Par défaut, nextval...	Automatic integer primary key
db_update_timestamp	timestamp without time zone	Par défaut, now()	
username	character varying	Par défaut, "current_user"()	User (role) which created data record
Index :			
"topo_points_pkey" PRIMARY KEY, btree (numauto)			
Contient des OID: non			

2.12. Tables de PostGIS

2.12.1. Table geometry columns

Table « public.geometry_columns »			
Colonne	Type	Modificateurs	Description
f_table_catalog	character varying(256)	non NULL	
f_table_schema	character varying(256)	non NULL	
f_table_name	character varying(256)	non NULL	
f_geometry_column	character varying(256)	non NULL	
coord_dimension	integer	non NULL	
srid	integer	non NULL	
type	character varying(30)	non NULL	
Index :			
"geometry_columns_pk" PRIMARY KEY, btree (f_table_catalog, f_table_schema, f_table_name, f_geometry_column)			
Contient des OID: oui			

2.12.2. Table spatial_ref_sys

Table « public.spatial_ref_sys »				
Colonne	Type	Modificateurs	Stockage	Description
srid	integer	non NULL	plain	
auth_name	character varying(256)		extended	
auth_srid	integer		plain	
srtext	character varying(2048)		extended	
proj4text	character varying(2048)		extended	
Index :				
"spatial_ref_sys_pkey" PRIMARY KEY, btree (srid)				
Contient des OID: non				

3. Procédures principales pour l'entretien de la base de données

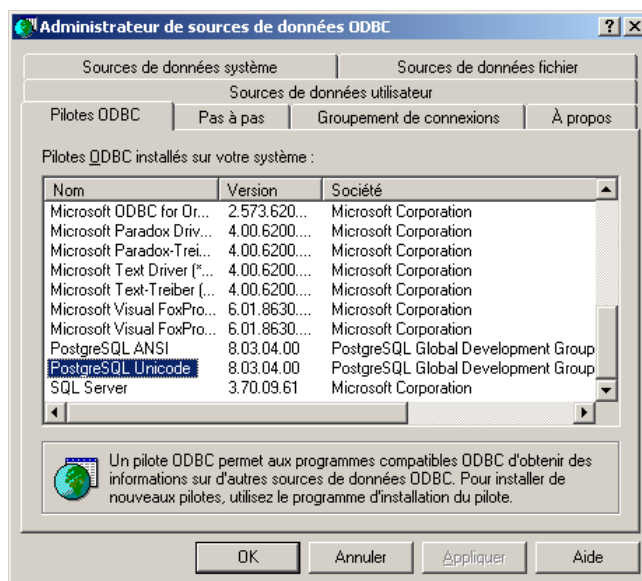
Un certain nombre de fonctionnalités sont implémentées automatiquement dans la base de données: des fonctions internes, un ensemble de vues, des programmes externes s'adressant directement à la base, et des utilitaires divers. On liste ici les principales procédures à suivre. Tous ces points ont été explicités au personnel au cours de la mission.

3.1. Connexion à la base de données

3.1.1. Se connecter au serveur via le middleware ODBC

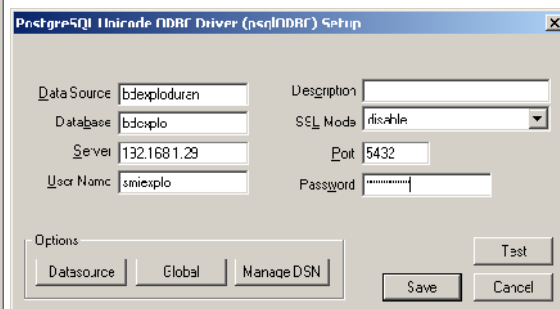
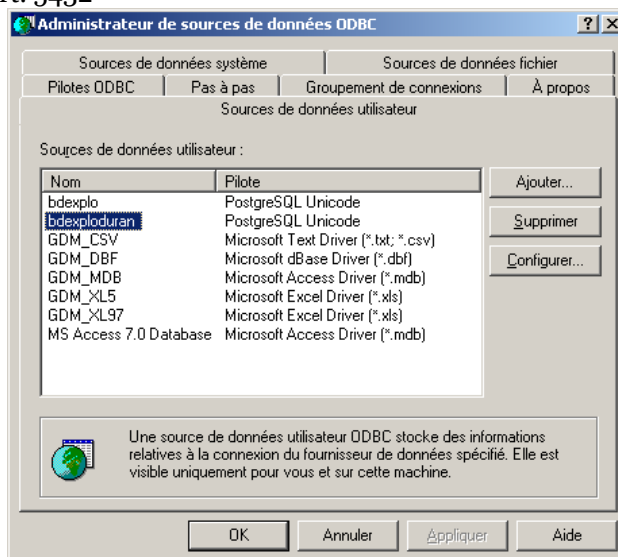
Les pilotes ODBC (Open DataBase Connectivity) sont disponibles sur le site de référence: <http://psqlodbc.projects.pgfoundry.org/>. Il faut choisir un pilote avec l'encodage UTF-8, ou Unicode.

Même les anciennes versions de pilotes fonctionnent parfaitement. La configuration suivante est opérationnelle, ce malgré un pilote relativement ancien et une version majeure différente entre le serveur (9.0) et le pilote (8.0):



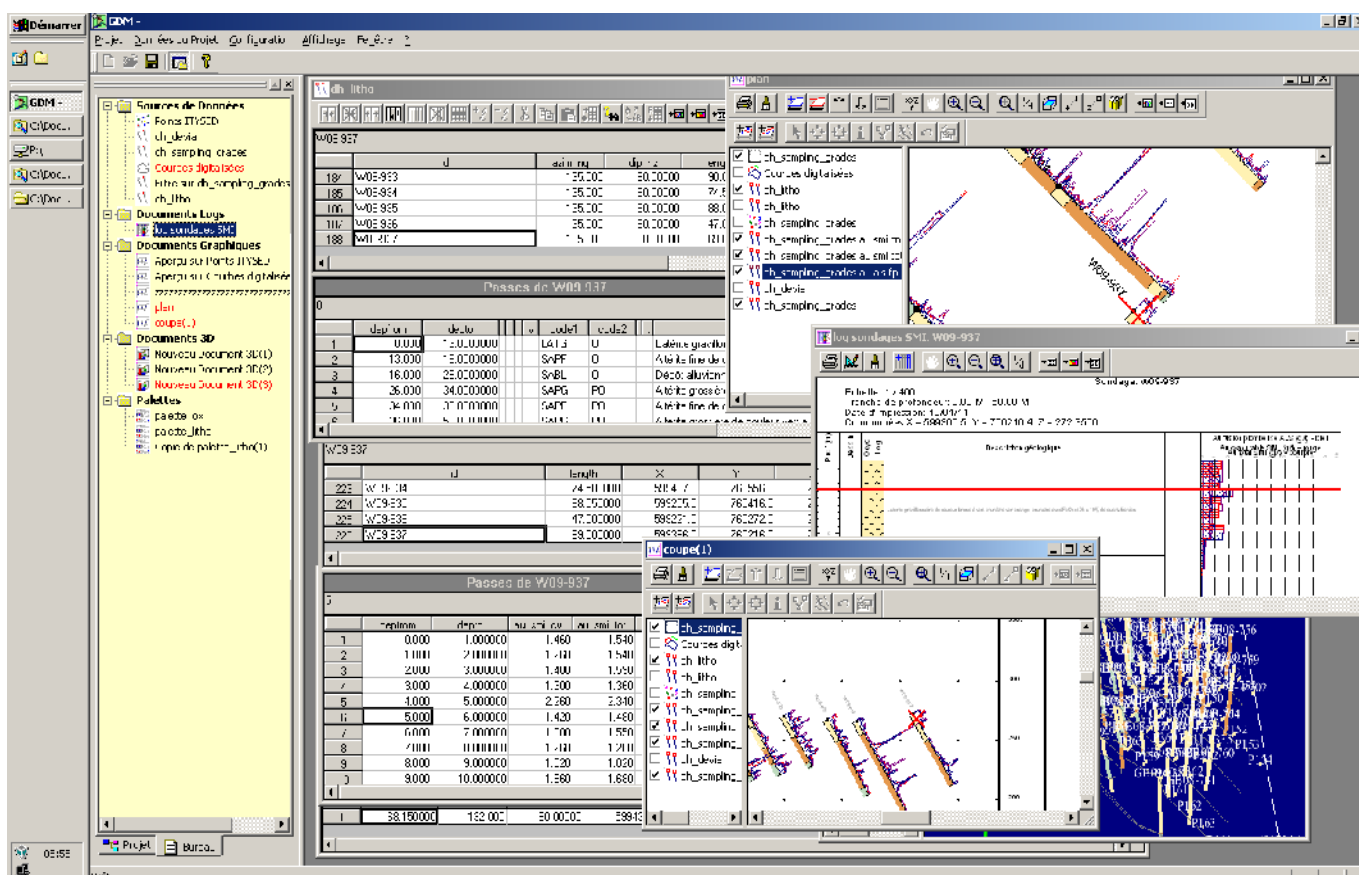
Les paramètres suivants permettent d'accéder à la base:

- database: bdexplo
- server: 192.168.1.29
- user name: (nom d'utilisateur)
- password: (mot de passe)
- port: 5432

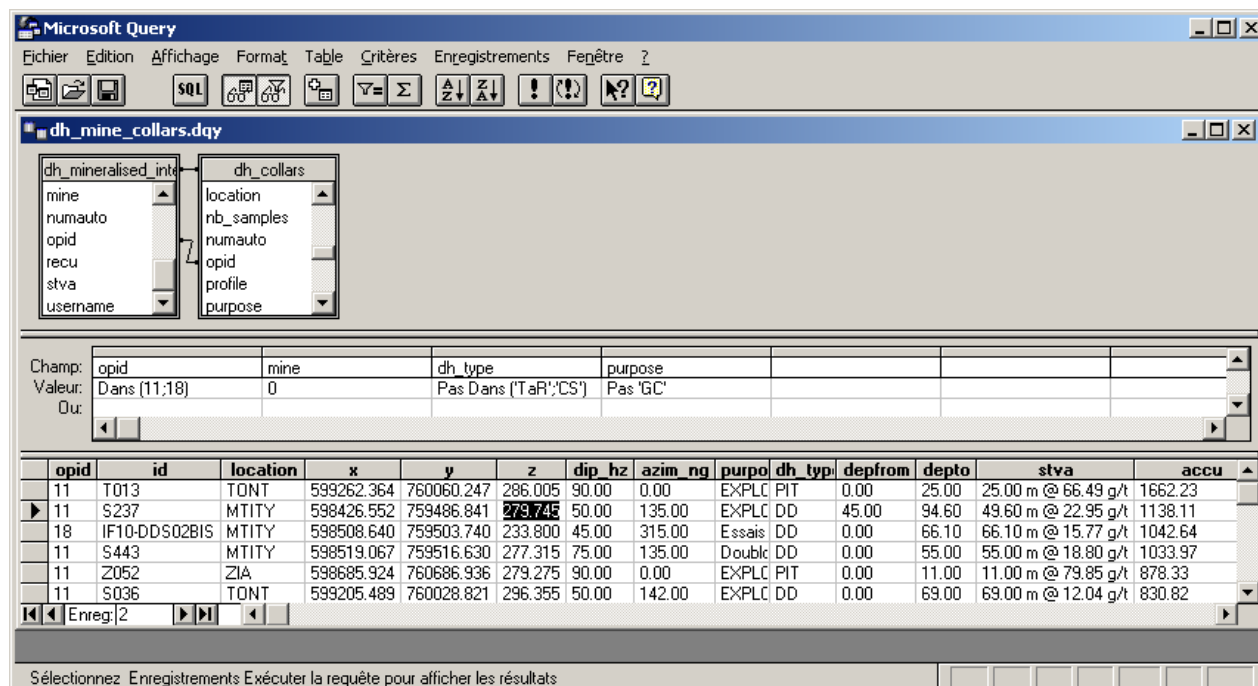


N'importe quel programme client pouvant utiliser une connexion ODBC peut se connecter et travailler avec la base bdxplo, via une connexion ainsi paramétrée.

La capture d'écran suivante montre par exemple une session de travail GDM sur l'opid 11, où toutes les données proviennent de la base de données, via ODBC.



Un autre exemple d'application cliente: une requête par MS Query, passant par la même connexion ODBC:

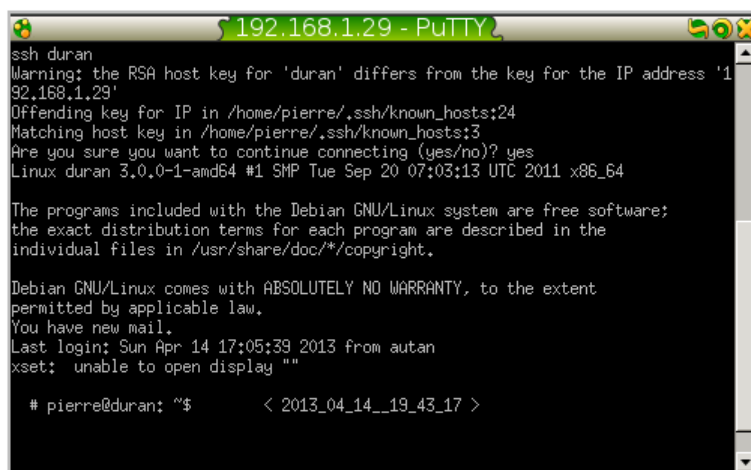


3.1.2. Se connecter au serveur pour travail en ligne de commandes

Depuis une machine sous Windows, lancer un émulateur de terminal, par exemple PuTTY (disponible ici: <http://the.earth.li/~sgtatham/putty/latest/x86/putty.exe>), avec pour paramètres de connexion:

- Host name: 192.168.1.29
- Port: 22
- Connexion type: SSH
- Window / translation: Remote character set: UTF-8

À l'ouverture de session, entrer l'identifiant et le mot de passe. On a une fenêtre du type:



Une fois connecté sur le serveur, on a l'invite du shell (le shell par défaut est le bash (bourne again shell)), où l'on peut entrer n'importe quelle commande Unix valide. La touche tabulation permet la complétion des commandes, des noms de répertoires et des noms de fichiers.

Avec les options définies dans `.bashrc` `.inputrc`, il suffit de commencer à taper le début d'une commande, puis d'appuyer sur la flèche vers le haut pour rappeler automatiquement la dernière commande tapée commençant par les lettres déjà entrées.

Pour interagir avec la base de données, on utilise la commande `psql`, avec pour argument le nom de la base¹²:

```
psql bdexplo
```

Si l'on est loggé depuis un système différent (pas en SSH), il faut préciser le nom ou l'adresse IP du serveur par l'option `-h`:

```
psql -h 192.168.1.29 bdexplo
```

Si l'on veut se connecter sous un autre rôle (utilisateur), on utilise l'option `-U`: par exemple, ici, loggé en tant que **pierre** sur le système **autan**, on ouvre une connexion à la **base bdexplo** du système **duran**, en tant **qu'utilisateur postgres**¹³:

```
# pierre@autan: ~$ < 2013_04_14_16_44_07 >
psql -h duran -d bdexplo -U postgres
Mot de passe pour l'utilisateur postgres :
psql (9.0.6)
Saisissez « help » pour l'aide.

bdexplo=#
```

L'invite en mode superutilisateur est marquée par un `#`: il ne faut **pas** se connecter en tant que `tel`, sauf cas exceptionnel.

Lors d'un travail en `psql`, il faut toujours cadrer le travail dans une transaction: non seulement cela assure une sécurité maximale, quelle que soit la durée et la complexité des travaux faits, mais cela permet de refaire tout instantanément, en cas d'erreur, sans danger.

¹² Le nom de la base peut être mentionné avec l'option `-d`, si on veut donner les options dans le désordre.

¹³ Les commandes à taper sont toujours indiquées en **gras**.

Par exemple (ce qui est entré par l'utilisateur est marqué en **gras**):

```
# pierre@autan: ~$          < 2013_10_12__08_34_15 >
psql bdexplo
psql (9.0.6)
Saisissez « help » pour l'aide.

--[local] bdexplo=>
BEGIN TRANSACTION;
BEGIN                                     <= début d'une transaction
--[local] bdexplo=>
DELETE FROM public.dh_litho;           <= bourde
DELETE 151874                           <= aïe
--[local] bdexplo=>
ABORT TRANSACTION;
ROLLBACK                                <= tout est annulé
--[local] bdexplo=>
BEGIN TRANSACTION;
BEGIN                                     <= début d'une autre transaction
--[local] bdexplo=>
DELETE FROM public.dh_litho WHERE opid IS NULL; <= voilà ce qu'on voulait vraiment faire
DELETE 1                                 <= voilà qui est mieux
--[local] bdexplo=>
COMMIT;
COMMIT                                  <= les modifications sont réellement inscrites, une fois qu'on est sûr
--[local] bdexplo=>
```

Pour ce faire, le mode de travail classique est d'avoir côte à côte une fenêtre avec l'interface psql, et une fenêtre d'éditeur de texte où l'on tape les requêtes, avec coloration syntaxique, auto-complétion, toutes ces facilités. Le travail se fait par copiers-collers intensifs entre les deux fenêtres.

3.2. Tracabilité des données

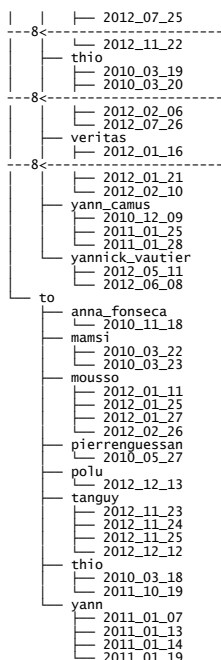
À chaque insertion d'un enregistrement dans la base, le nom de l'utilisateur et le timestamp sont enregistrés, pour chaque enregistrement, dans les champs username et db_update_timestamp.

Quand c'est nécessaire, un enregistrement est créé dans la table lex_datasource, avec un identifiant numérique et un fichier d'origine des données. Les programmes d'import des données analytiques gèrent automatiquement ces métadonnées, et permettent de revenir à la source de données; il faut bien sûr conserver les fichiers de données, donc les avoir bien rangés dans une arborescence adaptée avant de les importer. D'expérience, l'arborescence la plus adaptée est du type:

```
projet/transferts/from/personne/aaaa_mm_jj/fichiers
```

Tous les fichiers de données à partir desquels la base exploration de la SMI a été historiquement constituée sont stockés dans une telle arborescence:

```
~/smi/transferts/
├── from
│   ├── charles
│   │   ├── 2010_02_05
│   │   ├── 2010_03_06
│   │   └── 2010_03_24
│   ├── 8-----
│   ├── charlot
│   │   ├── 2012_05_29
│   │   ├── 2012_08_08
│   │   └── 2012_08_09
│   ├── 2010_03_12
│   ├── 2010_03_22
│   ├── 2011_02_14
│   ├── 2011_02_28
│   ├── jjk
│   │   ├── 2010_03_18
│   │   └── 2011_10_17
│   ├── 8-----
│   ├── 2012_04_19
│   ├── mamsi
│   │   ├── 2010_03_15
│   │   ├── 2010_03_18
│   │   └── 2010_03_22
│   ├── 8-----
│   ├── 2011_10_17
│   ├── 2011_11_15
│   ├── 2011_11_21
│   ├── maxime_dupere
│   │   └── 2010_11_24
│   ├── mohamed_nasraoui
│   │   ├── 2011_07_08
│   │   └── 2011_10_07
│   ├── moussso
│   │   ├── 2011_11_26
│   │   └── 2011_12_22
│   ├── 8-----
│   ├── 2012_04_18
│   ├── 13h17
│   ├── 13h55
│   ├── 2012_04_19
│   ├── 2012_04_21
│   ├── pierre_ng
│   │   └── 2010_03_13
│   ├── 8-----
│   ├── 2012_01_16
│   ├── polu
│   │   ├── 2011_10_18
│   │   ├── 2012_08_16
│   │   ├── 2012_10_31
│   │   └── 2012_12_13
│   ├── soro
│   │   └── 2011_01_19
│   ├── 8-----
│   ├── 2012_03_23
│   ├── 2012_03_25
│   └── tanguy
│       └── 2012_07_10
```



3.2.1. Création d'une nouvelle source de données (datasource)

Le programme:

```
gll_bdexplo_new_datasource.r
```

appelé avec un fichier en argument, permet de créer un datasource complet. Par exemple, au moment d'importer le fichier dev.xls dans la base, pour lui attribuer un identifiant datasource:

```
# pierre@autan: ~ < 2013_08_02__08_02_10 >
gll_bdexplo_new_datasource.r
~/smi/transferts/from/pierre_ng/2010_03_13/exploration/dac/dev.xls

ok, proposed datasource_id 1264 free in database
ok, file /home/pierre/smi/transferts/from/pierre_ng/2010_03_13/exploration/dac/dev.xls open
Générer l'enregistrement dans public.lex_datasource:
INSERT INTO public.lex_datasource (opid, filename, datasource_id) VALUES (11,
'/home/pierre/smi/transferts/from/pierre_ng/2010_03_13/exploration/dac/dev.xls', 1264);
(y/n) ?y
journal_sql: ["--2-Aug-2013/8:03:13" {INSERT INTO public.lex_datasource (opid, filename, datasource_id)
VALUES (11, '/home/pierre/smi/transferts/from/pierre_ng/2010_03_13/exploration/dac/dev.xls', 1264);}
Commit changes to database (undoable)?y
New datasource generated:
1264
# pierre@autan: ~ < 2013_08_02__08_02_10 >
```

3.2.2. Suppression de tous les enregistrements correspondant à un datasource

Cas typique: un fichier de résultats d'analyses a été importé deux fois dans la base: il faut supprimer un des deux jeux de données. Un cas plus complexe: un fichier .xls a été importé, qui comportait huit onglets, dont les informations ont été réparties dans une demi-douzaine de tables. Il s'avère qu'on avait importé une mauvaise version de ce fichier, et qu'un fichier meilleur a été trouvé entretemps. Il s'agit, là aussi, de supprimer le jeu de données.

Comme on a eu recours au système de traçabilité des données importées par le système du datasource, on peut retrouver aisément ces données par leur datasource, et on peut les supprimer.

Le programme qui fait cette tâche est:

```
gll_bdexplo_delete_datasource_and_dependant_information.r
```

Il faut l'appeler avec en arguments les identifiants (numériques) des datasources qu'on veut éliminer. Par exemple, on veut supprimer les enregistrements correspondant aux datasources 1275 et 999999:

```
# pierre@autan: ~/smi/transferts/from/sidiki_fofana/2013_08_20 < 2013_08_20__10_37_39 >
gll_bdexplo_delete_datasource_and_dependant_information.r 1275 999999
Gll preferences loaded:
dbhost: "autan"
dbname: "bdexplo"
user: "pierre"
```

```

tmp_schema: "tmp_imports"
Current working directory:
/home/pierre/heaume_pierre/pchgeol/operations/smi/transferts/from/sidiki_fofana/2013_08_20/tt/
Connected to database bdxplo hosted by autan, logged in as role pierre
Connected to database bdxplo hosted by autan, logged in as role pierre
Datasources to be deleted: 1275 999999
-- Table lab_ana_batches_reception contient 1 enregistrements correspondant au datasource 1275 pour l'opid 18
-- Table lab_ana_results contient 297 enregistrements correspondant au datasource 1275 pour l'opid 18
-- lab_ana_batches_reception 1
DELETE FROM public.lab_ana_batches_reception WHERE opid = 18 AND datasource = 1275; -- (1 records should be deleted)
-- lab_ana_results 297
DELETE FROM public.lab_ana_results WHERE opid = 18 AND datasource = 1275; -- (297 records should be deleted)
DELETE FROM public.lex_datasource WHERE opid = 18 AND datasource_id = 1275;
DELETE FROM public.lex_datasource WHERE opid = 18 AND datasource_id = 999999;
Nothing has been actually done on the database: you must manually run the SQL statements above

# pierre@autan: ~/smi/transferts/from/sidiki_fofana/2013_08_20 < 2013_08_20__10_37_39 >

```

Par mesure de sécurité, ce programme ne fait que montrer le SQL avec les commandes DELETE, qu'il faut ensuite "jouer" dans un psql ou équivalent.

3.3. Import de données analytiques

3.3.1. Import d'un fichier

Les résultats analytiques ont été fournis par Bureau Veritas sous plusieurs formats. On a convenu d'un format le plus approprié, et on a programmé l'outil d'import automatique de fichiers à ce format.

gll_bdxplo_import_lab_data_eav.r

est le programme à appeler pour importer directement en base un fichier d'analyse au bon format.

Exemple de début de fichier de données:

```

Labname: ACME ANALYTICAL LABORATORIES LTD.
Client: Societe Des Mines D'Ity (SMI)
Validated: 31-May-13
Job_number: ABJ13000003
Number_of_samples: 6
Project: None Given
Shipment_id: BV130425SMI
P_O_number: 2013 04 25
Received: 29-May-13

Jobno,Orderno,Sampletype,Sampleid,Scheme,Analyte,Value,Unit,DetLim,UpLim
ABJ13000003,2013 04 25,Rock Pulp,0001V,1D,Mo, 1,PPM,1,2000
ABJ13000003,2013 04 25,Rock Pulp,0001V,1D,Cu, 15,PPM,1,10000
ABJ13000003,2013 04 25,Rock Pulp,0001V,1D,Pb, 6,PPM,3,10000
ABJ13000003,2013 04 25,Rock Pulp,0001V,1D,Zn, 36,PPM,1,10000
ABJ13000003,2013 04 25,Rock Pulp,0001V,1D,Ag,<0.3,PPM,0.3,100
ABJ13000003,2013 04 25,Rock Pulp,0001V,1D,Ni, 4,PPM,1,10000
ABJ13000003,2013 04 25,Rock Pulp,0001V,1D,Co, 2,PPM,1,2000
ABJ13000003,2013 04 25,Rock Pulp,0001V,1D,Mn, 92,PPM,2,10000
ABJ13000003,2013 04 25,Rock Pulp,0001V,1D,Fe, 0.80,%,0.01,40
...

```

Pour l'importer, il faut d'abord enregistrer le fichier dans un répertoire approprié. On peut utiliser l'utilitaire mkdiraujourd'hui pour faire automatiquement un répertoire du jour courant:

```

# pierre@autan: ~ < 2013_08_20__10_37_39 >
cd smi/transferts/from/sidiki_fofana/

# pierre@autan: ~/smi/transferts/from/sidiki_fofana < 2013_08_20__10_37_39 >
mkdiraujourd'hui
Make directory with current date YYYY_MM_DD
drwxr-xr-x 2 pierre pierre 4.0K août 20 10:37 2013_08_20

```

Il faut aller dans le répertoire contenant le fichier à importer:

```

# pierre@autan: ~/smi/transferts/from/sidiki_fofana < 2013_08_20__10_37_39 >
cd 2013_08_20/

```

Puis il faut lancer le programme gll_bdxplo_import_lab_data_eav.r avec pour argument le nom de fichier à importer; l'import se fait alors automatiquement, après des vérifications du format des données du fichier, et les tables lex_datasource, lab_ana_results, lab_ana_batches_reception sont automatiquement remplies:

```

# pierre@autan: ~/smi/transferts/from/sidiki_fofana < 2013_08_20__10_37_39 >
gll_bdxplo_import_lab_data_eav.r ABJ13000003.CSV
Gll preferences loaded:
dbhost: "autan"
dbname: "bdxplo"
user: "pierre"
tmp_schema: "tmp_imports"
Current working directory:
/home/pierre/heaume_pierre/pchgeol/operations/smi/transferts/from/sidiki_fofana/2013_08_20/
password: **
New datasource_id generation...
Ok, proposed datasource_id 1275 free in database
Reading data from
/home/pierre/heaume_pierre/pchgeol/operations/smi/transferts/from/sidiki_fofana/2013_08_20/ABJ130000

```

```

03.CSV...
Check: header data ok
Generation of data input into database...
New datasource generated: 1275
SQL statement to be run:

INSERT INTO public.lex_datasource (opid, filename, datasource_id) VALUES (18,
'/home/pierre/smi/transferts/from/sidiki_fofana/2013_08_20/ABJ13000003.CSV', 1275);
INSERT INTO public.lab_ana_results (opid, labname, jobno, orderno, sample_id, scheme,
analyte
...
Done...
committed => end.
# pierre@autan: ~/smi/transferts/from/sidiki_fofana < 2013_08_20__10_37_39 >

```

Une fois revenu à l'invite, tout est fait. Si une seule erreur est rencontrée durant le traitement (manque un champ dans le format de fichier, erreur d'intégrité au niveau de la base, erreur disque, fichier tronqué, etc.), l'ensemble du processus est interrompu, et le contenu de la base est inchangé.

Pour les autres formats qui étaient utilisés auparavant, on a mis à jour les programmes qui importaient les données, mais leur usage est déconseillé. Notamment, le contrôle et la gestion des erreurs possibles est bien plus faible. Et ils sont beaucoup moins génériques que `gll_bdexplo_import_lab_data_eav.r`.

Ils s'utilisent de la même manière, en donnant comme argument le fichier à importer:

Le programme suivant:

```
gll_bdexplo_procedure_import_fichier_veritas_fa_bdexplosmi.py 14
```

était utilisé pour importer les données d'analyse par fusion plombeuse (FA pour Fire Assay) de Bureau Veritas au format en .csv "columnnaire", par opposé au format Entité-Attribut-Valeur (EAV). Par exemple, début d'un tel fichier:

```

Batch_No,u100556
CLIENT,Societe des Mine d'Ity
No of SAMPLES,228
DATE RECEIVED,040513
DATE COMPLETED,170513
PROJECT,
CERTIFICATE COMMENTS,Finished
PO NUMBER,63771
IDENT,Au,
UNITS,ppm,
SCHEME,FA001
DETECTION LIMIT, 0.01,
42600,0.07
42601,0.03
42602,0.14
42603,0.74
42604,0.22
42605,0.17
42606,0.10
...

```

Ce programme s'utilise avec le fichier à importer comme argument:

```

# pierre@autan: ~/smi/transferts/from/kalvin/2013_06_23$ < 2013_06_23__18_33_52 >
procedure_import_fichier_veritas_bdexplosmi.py u100556.csv

```

Le programme suivant:

```
gll_bdexplo_import_acme_file_multielt.r
```

avait commencé à être développé pour importer les données multi-élémentaires venant d'ACME. Il n'a pas été terminé, pour les raisons invoquées plus haut; il ne doit pas être utilisé.

3.3.2. Import d'une série de fichiers

Pour importer **toute une série de fichiers .csv**, il faut appeler le **programme d'import** dans une **boucle for**. Par exemple, pour importer tous les fichiers .csv situés dans le répertoire courant:

```

# smiexplo@durant: /home/pierre$ < 2013_08_06__19_54_01 >
for i in *csv; do procedure_import_fichier_veritas_bdexplosmi.py $i; done

```

Ceci lance le programme d'import pour chaque fichier .csv rencontré dans le répertoire courant, successivement.

¹⁴ Ce programme était auparavant nommé `procedure_import_fichier_veritas_bdexplosmi.py`

3.4. Import d'un fichier .csv quelconque dans une table temporaire

Il est parfois utile d'importer un fichier de données quelconque, afin de travailler avec dans la base de données. L'utilitaire suivant:

```
csv2sql
```

permet de faire cela: à partir d'un fichier .csv¹⁵ exemple.csv, il va créer une table temporaire nommée tmp_exemple dans le schéma tmp_imports, qui est dédié à ces tables.

Cet utilitaire est le plus dépouillé, rapide et générique possible: aussi, tous les champs sont importés comme des textes (varchar). Par la suite, si l'on veut requêter sur des champs contenant des valeurs numériques, il faudra les convertir (type cast) à l'aide de ::numeric.

Exemple: import du fichier tt130812_smi_explo_suivi_sondages.csv:

```
# pierre@autan: ~/smi/transferts/from/marie_cecile/2013_08_13 < 2013_08_13__20_48_22 >
csv2sql tt130812_smi_explo_suivi_sondages.csv
Import of data from tt130812_smi_explo_suivi_sondages.csv into tmp_imports.tmp_tt130812_smi_explo_suivi_sondages table from bdexplo database.
Input file lines count: 1524 tt130812_smi_explo_suivi_sondages.csv
NOTICE: la table « tmp_tt130812_smi_explo_suivi_sondages » n'existe pas, poursuite du traitement
DROP TABLE
CREATE TABLE
DELETE 0
Record count of table tmp_imports.tmp_tt130812_smi_explo_suivi_sondages: 1523
Overview of table tmp_imports.tmp_tt130812_smi_explo_suivi_sondages (2 first records):
date | shift | mois | rig | id | completed | id_projet | location | dh_type | planned_depth | azim_ng | dip_hz | len_destr | len_pq | len_hq |
len_nq | drilled_shift | depth_tot | reco_m | reco_pc_shift | reco_m_tot | reco_pc_tot | standby_afm | standby_sfm | meteo | moving | observation |
bentonite_thr_kg | pure_bore_kg | huile_l | sc_mud_l | k-ion_l | sc_mud_p_kg | cap_23_l | gs_kg
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2013-01-10 | Jour | 1 | NSM_5261 | S568 | 1 | P3595 | VERSEST | DD | 120 | 135 | 60 | 22.4 | 3.6 |
22.4 | 22.4 | 21.6 | 96.43% | 21.6 | 96.43% | 1 | 120 | 135 | 60 | 22.4 | 3.6 |
2013-01-10 | Nuit | 1 | NSM_5261 | S568 | 0 | P3595 | VERSEST | DD | 120 | 135 | 60 | 31.7 | 36.3 |
31.7 | 54.1 | 31.65 | 99.84% | 53.25 | 98.43% | 1.25 | 120 | 135 | 60 | 31.7 | 36.3 |
(2 lignes)
...
(and 2 last records):
2013-08-01 | Jour | 8 | TM120_5309 | DA13-008 | 1 | P32005 | DAAPLEU | DD | 150 | 135 | 50 | 12.6 | 3.6 |
16.2 | 16.2 | 14.6 | 90.12% | 14.6 | 90.12% | 1 | 150 | 135 | 50 | 12.6 | 3.6 |
2013-08-01 | Nuit | 8 | TM120_5309 | DA13-008 | 0 | P32005 | DAAPLEU | DD | 150 | 135 | 50 | 36.3 | 36.3 |
36.3 | 52.5 | 35.55 | 97.93% | 50.15 | 95.52% | 1 | 150 | 135 | 50 | 36.3 | 36.3 |
(2 lignes)
If you saw your data, it is ok in tmp_imports.tmp_tt130812_smi_explo_suivi_sondages. To browse it:
echo "SELECT * FROM tmp_imports.tmp_tt130812_smi_explo_suivi_sondages;" | psql -X -h duran -d bdexplo | less
# pierre@autan: ~/smi/transferts/from/marie_cecile/2013_08_13 < 2013_08_13__20_48_22 >
```

Pour traiter ces données et les importer correctement dans la table de données définitive (shift_reports), les commandes SQL suivantes ont été utilisées:

```
UPDATE tmp_imports.tmp_tt130812_smi_explo_suivi_sondages SET shift = 'J' WHERE shift = 'Jour';
UPDATE tmp_imports.tmp_tt130812_smi_explo_suivi_sondages SET shift = 'N' WHERE shift ILIKE 'Nuit';
UPDATE 716

INSERT INTO public.shift_reports (
  opid, date, shift, rig, id, peg_number, planned_length, drilled_length_during_shift, drilled_length,
  completed, comments, drilled_shift_destr, drilled_shift_pq, drilled_shift_hq, drilled_shift_nq, recovered_length_shift,
  stby_time1_h, stby_time2_h, stby_time3_h, moving_time_h, no_fichette
)
(SELECT
  18, date::date, shift, rig, id, id_projet, planned_depth::numeric, drilled_shift::numeric, depth_tot::numeric,
  completed::boolean, observation, len_destr::numeric, len_pq::numeric, len_hq::numeric, len_nq::numeric, reco_m::numeric,
  standby_afm::numeric, standby_sfm::numeric, meteo::numeric, moving::numeric, no_fichette::numeric
FROM tmp_imports.tmp_tt130812_smi_explo_suivi_sondages
);
INSERT 0 1523
```

Noter les type casts ::numeric.

3.5. Calculs de passes minéralisées

Une fois des résultats analytiques importés en base, le programme suivant:

```
procedure_genere_passmin_vers_bdexplo.sh
```

est un simple "wrapper" shell qui appelle le programme:

```
procedure_genere_passmin.py
```

et envoie directement sa sortie dans la base de données, le tout en une seule commande.

procedure_genere_passmin.py génère les passes minéralisées, en fonction de critères que l'on peut modifier au besoin, en éditant /usr/bin/procedure_genere_passmin.py vers la fin du fichier, vers la ligne 350:

```
generate_mineralised_intervals(opid = opid_, criteria='TRUE', mineralised_intervals_class=0,
  cutoff=0.5, flag_dilution=True, max_barren_length=10, flag_accu_mini=False, accu_mini=0,
  gradefield='au6_ppm', insert_records=False, delete_records=False)
```

15 NB: séparateur = virgule, CSV veut dire Comma Separated Values, comma = virgule

```
generate_mineralised_intervals(opid = opid_, criteria='TRUE', mineralised_intervals_class=1,
cutoff=0.5, flag_dilution=True, max_barren_length=2, flag_accu_mini=False, accu_mini=0,
grade_field='au6_ppm', insert_records=False, delete_records=False)

generate_mineralised_intervals(opid = opid_, criteria='TRUE', mineralised_intervals_class=2,
cutoff=2, flag_dilution=True, max_barren_length=1, flag_accu_mini=False, accu_mini=0,
grade_field='au6_ppm', insert_records=False, delete_records=False)
```

Ces lignes appellent trois fois la fonction principale, `generate_mineralised_intervals()`, avec des arguments différents, pour définir trois classes de passes minéralisées:

- classe mine = 0 passes larges, bien diluées
- classe mine = 1 passes normales
- classe mine = 2 petites passes, riches

Si l'on souhaite par exemple calculer des passes minéralisées avec une **teneur de coupure de 5 g/t Au sur le champ au6_ppm**, une **dilution** en tolérant au **maximum 2m de stérile** dans un intervalle, et une **accumulation minimale de 10 m.g/t Au** pour chaque passe, le tout devant être rangé dans les **passes riches de classe 2**, on écrira:

```
generate_mineralised_intervals(opid = opid_, criteria='TRUE', mineralised_intervals_class=2,
cutoff=2, flag_dilution=True, max_barren_length=2, flag_accu_mini=True, accu_mini=10,
grade_field='au6_ppm', insert_records=False, delete_records=False)
```

Exemple d'utilisation pour calculer l'ensemble des passes minéralisées sur la base:

```
# smiexplo@duran: ~$ < 2013_08_14__19_22_27 >
procedure_genere_passmin_vers_bdexplo.sh
DELETE 1078
INSERT 0 1078
DELETE 0
INSERT 0 2631
DELETE 0
INSERT 0 1311
```

Pour mettre à jour les sommes d'accumulations rapportées aux collets, on exécute l'instruction SQL suivante:

```
UPDATE public.dh_collars SET accusum = calcul_accusum FROM (SELECT opid, id, sum(accu) AS
calcul_accusum FROM dh_mineralised_intervals WHERE mine = 0 GROUP BY opid, id) AS tmp WHERE
(dh_collars.opid = tmp.opid AND dh_collars.id = tmp.id);
```

Pour recalculer des passes minéralisées, si tant est qu'on les modifie à la main (ce qui est conseillé; la génération automatique de passes n'est pas une pratique conseillée), on exécute le script:

`procedure_update_passmin.sh`

qui appelle simplement, via une commande shell, le script SQL `procedure_update_passmin.sql`, dont le code suit¹⁶:

```
# pierre@autan: ~$ < 2013_08_14__20_17_05 >
cat geolllibre/bdexplo/procedure_update_passmin.sql
--###PROCEDURE_UPDATE_PASSMIN: Procédure pour maj passes minéralisées### (correction: teneurs négatives zérotées):

UPDATE public.dh_mineralised_intervals
SET stva = tmp2.stretch_value, avau = tmp2.avg_au, accu=tmp2.accu
FROM (
SELECT
tmp.opid, tmp.id, tmp.depfrom, tmp.depto, tmp.mine,
replace(to_char(mineralised_length, 'FM99990.99 m @ '), '. m @ ', ' m @ ') || trim(replace(to_char(avg_au, 'FM9999990.99 '), '. ', ' ')) || ' g/t Au'
AS stretch_value, avg_au, accu
FROM (
SELECT
dh_mineralised_intervals.opid, dh_mineralised_intervals.id, dh_mineralised_intervals.depfrom, dh_mineralised_intervals.depto,
dh_mineralised_intervals.mine, sum(dh_sampling_au.au_ppm), count(dh_sampling_au.au_ppm),
sum(au_ppm*(dh_sampling_au.depto - dh_sampling_au.depfrom)) AS accu,
sum(dh_sampling_au.depto - dh_sampling_au.depfrom) AS mineralised_length,
sum(au_ppm*(dh_sampling_au.depto - dh_sampling_au.depfrom)) / sum(dh_sampling_au.depto - dh_sampling_au.depfrom) AS avg_au
FROM
(
-- c'est ici qu'on choisit le champ au à moyenner
SELECT *, greatest(0, au6_ppm) AS au_ppm FROM dh_sampling_grades
WHERE sample_type IS NULL OR sample_type NOT IN ('DUP', 'NS')
)
AS dh_sampling_au,
dh_mineralised_intervals
WHERE
dh_mineralised_intervals.stva IS NULL
AND
(dh_mineralised_intervals.opid = dh_sampling_au.opid
AND dh_mineralised_intervals.id = dh_sampling_au.id
AND (
dh_mineralised_intervals.depto >= dh_sampling_au.depto
AND
dh_mineralised_intervals.depfrom <= dh_sampling_au.depfrom
)
)
GROUP BY dh_mineralised_intervals.opid, dh_mineralised_intervals.id,
dh_mineralised_intervals.depfrom, dh_mineralised_intervals.depto, dh_mineralised_intervals.mine
)
AS tmp
)
AS tmp2
WHERE
dh_mineralised_intervals.opid = tmp2.opid
AND dh_mineralised_intervals.id = tmp2.id
```

¹⁶ Nota: ce petit bout de SQL date (à quelques modifications minimales près) de 1996, rédigé à Yamfo et Kenyasi, au Ghana.

```
AND dh_mineralised_intervals.depfrom = tmp2.depfrom
AND dh_mineralised_intervals.depto = tmp2.depto
AND dh_mineralised_intervals.mine = tmp2.mine;
```

3.6. Vérifications de cohérence des données

Le programme suivant:

```
gll_bdexplo_generation_checks_vapas.r
```

va chercher toute une série de requêtes contenues dans

```
bdexplo_verifs.sql
```

qu'on met dans ~/bdexplo.

Il génère un gros fichier .html horodaté:

```
bdexplo_verifs_smi_2013_09_10_23_28_07.html
```

a, par exemple, été généré le 10 septembre.

Il contient les résultats des requêtes de vérification, c'est-à-dire des enregistrements qui sont incohérents, ou suspects de l'être.

Pour chaque requête de vérification, une boîte scrollable est générée dans le .html, avec le SQL de la requête suivi de son résultat. Ceci permet, lors des phases de corrections de données, de jouer les requêtes de vérifications, au fur et à mesure des corrections.

Un extrait de bdexplo_verifs_smi_2013_09_10_23_28_07.html:

Report generated on 10-Sep-2013/17:36:59+2:00

Most queries listed here are designed to outline problems in the dataset. They should return zero records, meaning that the dataset is right.

1. Check drill holes and trenches data

1.1. Collars information: dh_collars table

id duplicates in dh_collars

```
SELECT id AS dh_collars_id_non_uniq, COUNT(id) FROM dh_collars GROUP BY id HAVING COUNT(id)>1;
```

```
dh_collars_id_non_uniq | count
-----+-----
(0 ligne)
```

(query #1)

Duplicate samples referring to a blank sample:

```
SELECT * FROM qc_sampling WHERE qc_type = 'DUPLICATE' AND refers_to IN (SELECT sample_id FROM
qc_sampling WHERE qc_type = 'BLANK');
```

sample_id	qc_type	comments	opid	batch_id	refers_to	datasource	weight_kg	numauto	db_update
12737	DUPLICATE		11		12736	727		6453	2013-08-17 21
13243	DUPLICATE		11		13242	768		6472	2013-08-17 21
14384	DUPLICATE		11		14383	903		6518	2013-08-17 21
14726	DUPLICATE		11		14727	903		6533	2013-08-17 21
14877	DUPLICATE		11		14878	903		6539	2013-08-17 21
15085	DUPLICATE		11		15084	769		6547	2013-08-17 21
15127	DUPLICATE		11		15126	769		6549	2013-08-17 21
15250	DUPLICATE		11		15249	769		6554	2013-08-17 21

(query #68)

Il y a des incohérences sur certaines opérations qui n'en sont pas pour d'autres: par exemple, certaines opérations imposent une nomenclature des ouvrages de type GLO_D005 pour le cinquième sondage carotté (D) du projet GLO (cela occasionne moult sources d'erreurs). À Ity, ceci n'est pas relevant.

3.7. Création des vues standard

Le programme suivant:

```
gll_bdexplo_views_delete.r
```

détruit les vues standard de la base; le programme suivant:

```
gll_bdexplo_views_create.r
```

recrée ces vues.

Ces deux programmes sont particulièrement utiles lors de changements de structure de la base.

3.8. Création d'un compte utilisateur sur le système et sur la base

Exemple de la fabrication complète de compte pour l'utilisateur "marie_cecile".

- Il faut d'abord créer un compte sur le système:
- s'authentifier en tant que root, c'est-à-dire le superutilisateur du système:

```
# pierre@duran: ~$ < 2013_08_11_17_27_40 >
su
Mot de passe : (<= entrer le mot de passe de root, cf. feuille; il n'y a pas d'écho à l'écran, par
sécurité)
root@duran:/home/pierre#
```

- création du compte utilisateur:

```
root@duran:/home/pierre# adduser marie_cecile
Ajout de l'utilisateur « marie_cecile » ...
Ajout du nouveau groupe « marie_cecile » (1012) ...
Ajout du nouvel utilisateur « marie_cecile » (1011) avec le groupe « marie_cecile » ...
Création du répertoire personnel « /home/marie_cecile »...
Copie des fichiers depuis « /etc/skel »...
Nouveau mot de passe : (<= entrer le mot de passe; il n'y a pas d'écho à l'écran, par sécurité)
Retapez le nouveau mot de passe : (<= entrer le mot de passe; il n'y a pas d'écho à l'écran, par sécurité)
passwd : le mot de passe a été mis à jour avec succès
Modification des informations relatives à l'utilisateur marie_cecile
Entrez la nouvelle valeur ou « Entrée » pour conserver la valeur proposée
Nom complet []: Marie-Cécile Febvey
N° de bureau []:
Téléphone professionnelle []:
Téléphone personnel []:
Autre []:
chfn : nom contenant des caractères non ASCII : Marie-Cécile Febvey
Cette information est-elle correcte ? [O/n]
root@duran:/#
```

- Il faut ensuite créer le même utilisateur dans la base de données hébergée par le serveur:
- s'authentifier en tant que postgres, c'est-à-dire le superutilisateur du cluster de base de données:

```
root@duran:/# su postgres
X11 connection rejected because of wrong authentication. 17
postgres@duran:/#
```

- lancer psql, l'interpréteur en ligne de commande pour interagir avec la base de données:

```
postgres@duran:/# psql bdexplo
psql (9.0.6)
Saisissez « help » pour l'aide.
```

- création du rôle (utilisateur) dans la base:

```
postgres=# CREATE USER marie_cecile;
CREATE ROLE
```

- assigner à l'utilisateur le même mot de passe pour la base que sur le système¹⁸:

```
bdexplo=# ALTER ROLE marie_cecile PASSWORD '*****'; (ici, le mot de passe apparaîtrait en clair; il a été masqué ici)
```

- octroyer des droits à l'utilisateur: ceci se fait en le faisant appartenir à des rôles qui sont des groupes:

```
bdexplo=# GRANT data_admin TO marie_cecile;
GRANT ROLE
```

3.9. Traitement d'un ensemble de commandes SQL par lot (batch)

Pour "jouer" un ensemble de commandes SQL contenues dans un fichier texte, on utilise psql avec l'option -f avant le nom du fichier SQL. Par exemple, la commande suivante exécute les instructions SQL contenues dans le fichier script_majbd_2012_11_08.sql sur la base bdexplo, en tant que superutilisateur postgres¹⁹:

```
psql -d bdexplo -U postgres -f script_majbd_2012_11_08.sql
```

3.10. Documentation sur le moteur de base

Le document de référence en .pdf est, à l'usage, très complet et très pratique: doc_postgresql90.pdf

3.11. Sauvegarde intégrale de la base de données

La commande suivante:

¹⁷ Ce message est sans importance: il signifie que, depuis le terminal où l'on travaille, on n'a pas le droit d'afficher des programmes graphiques exécutés par le serveur: c'est sans objet.

¹⁸ On pourrait fort bien assigner un mot de passe différent; mais l'articulation des divers programmes GNU et de psql simplifie notablement les authentifications, si les mots de passes sont identiques.

¹⁹ Il s'agissait là de commandes de maintenance importante, qui requerraient des droits élevés.

```
pg_dump -U postgres -Fc bdexplo > database_duran_9_0_bdexplo_`date +%Y_%m_%d_%H%M`.pg_dump
```

Crée un fichier horodaté dans le répertoire courant, qui contient une sauvegarde complète (un "dump") de la base, au format adapté à pg_restore. Comme la commande se fait en tant que superutilisateur de la base (postgres), il faudra fournir, le moment venu, le mot de passe adéquat.

Par exemple, le fichier suivant:

```
database_duran_9_0_bdexplo_2013_08_05_14h22.pg_dump
```

est une sauvegarde complète qui a été créée le 5 août 2013.

Pour la restauration, on aura recours à pg_restore. Attention! Il est très vivement découragé de procéder à des restaurations sur la base de travail. Il faut plutôt restaurer la sauvegarde dans une nouvelle base, puis travailler par export des portions de données. Il est bien souvent beaucoup plus facile de travailler à partir des exports .csv.

3.12. Sauvegarde des tables principales de la base en .csv

Le daemon cron²⁰ du serveur fait tous les soirs une sauvegarde des principales tables de la base:

```
# smiexplo@duran: ~$ crontab -e
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
57 19 * * *      sauvegarde_bdexplo_smi_csv
```

C'est la commande

```
sauvegarde_bdexplo_smi_csv
```

qui est appelée par le crontab. Elle exporte tous les jours les tables suivantes dans des fichier .csv:

```
baselines
def_sampling_grades_fields
dh_collars
dh_density
dh_devia
dh_litho
dh_mineralised_intervals
dh_sampling_bottle_roll
dh_sampling_grades
dh_struct
dh_struct_measures
dh_tech
lab_ana_batches_expedition
lab_ana_batches_reception
lab_ana_qaqc_results
lab_ana_results
lex_datasource
lex_drill_hole_type
lex_codes
occurrences
operations
qc_sampling
sections_array
sections_definition
shift_reports
surface_samples_grades
topo_points
```

Attention au fait que certaines tables pourraient ne pas être sauvegardées. Il est conseillé de vérifier sauvegarde_bdexplo_smi_csv et d'y ajouter, au besoin, les noms de tables supplémentaires à sauvegarder automatiquement. Pour ce, il faut ajouter une ligne de la forme:

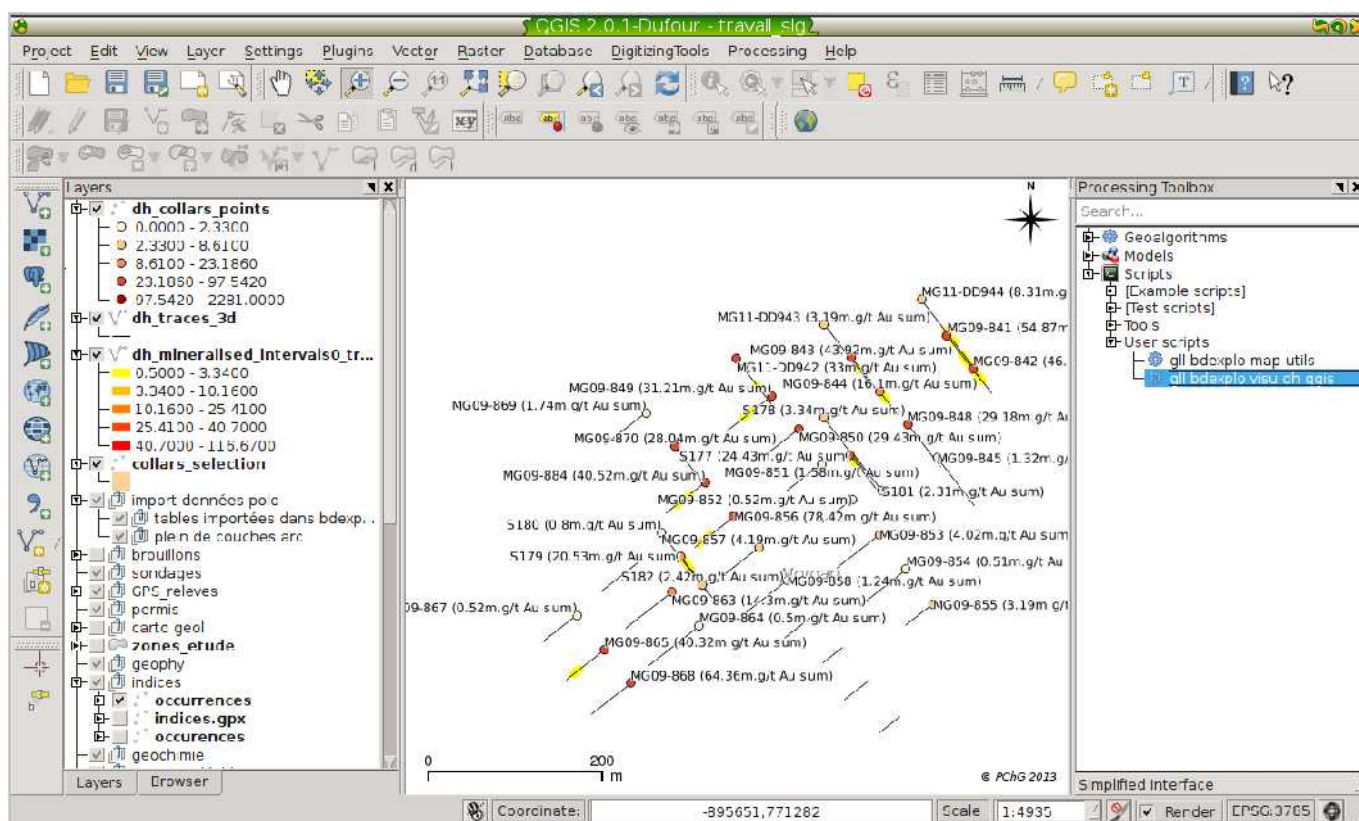
```
echo "COPY (
SELECT * FROM public.table WHERE opid = 11 OR opid = 18
) TO stdout WITH CSV HEADER" | psql -X bdexplo > table.csv
```

3.13. Vues cartographiques

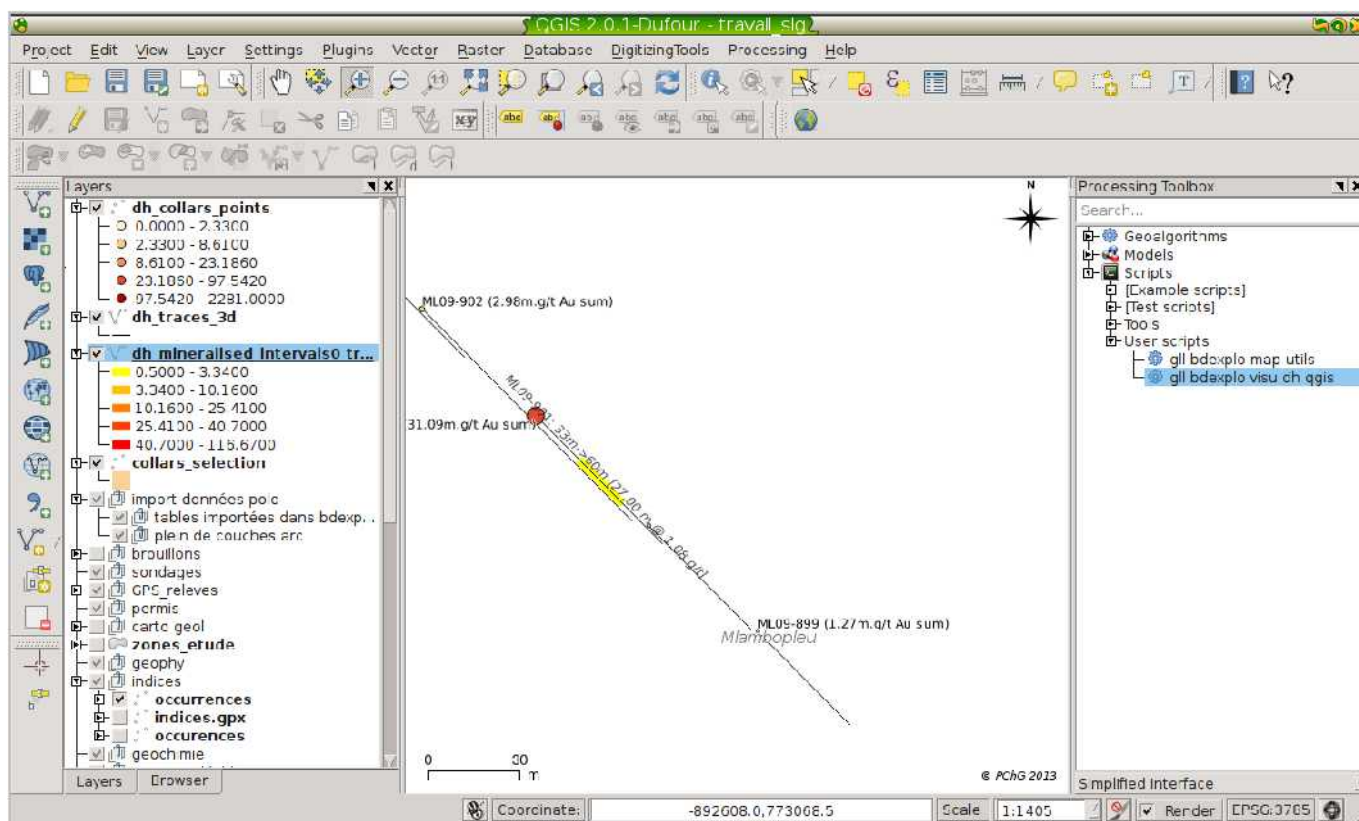
Les vues destinées à être cartographiées sont des vues dynamiques, où les géométries sont créées en temps réel à partir des données. La vue dh_collars_points crée des points cartographiables à partir des champs x, y de la table dh_collars. La vue dh_traces_3d crée des lignes cartographiables, à partir cette fois des champs x, y, z, azimuth, ng,

²⁰ Un daemon est un programme qui tourne tout le temps, en arrière-plan. Le daemon cron est le service crond, chargé de faire exécuter par le système toutes tâches définies et planifiées à l'avance. Cron vient de chronos, le dieu du temps.

dip_hz, length de la même table. La vue dh_mineralised_intervalso_traces_3d crée des lignes cartographiables au long des intervalles minéralisés de classe 0, à partir des données de dh_collars et de dh_mineralised_intervals. La capture d'écran suivante montre une combinaison de ces couches, à titre d'exemple, dans la région de Morgan:



Mêmes couches, détail sur un sondage, vers Mlambopleu:



3.14. Utilitaires, trucs et astuces

Pour créer un répertoire à la date d'aujourd'hui :

```
# pierre@autan: ~ < 2013_08_20_10_37_39 >
cd smi/transferts/from/sidiki_fofana/
```



```
# pierre@autan: ~/smi/transferts/from/sidiki_fofana < 2013_0_39 >
mkdir aujourd'hui
Make directory with current date YYYY_MM_DD
drwxr-xr-x 2 pierre pierre 4.0K août 20 10:37 2013_08_20

# pierre@autan: ~/smi/transferts/from/sidiki_fofana < 2013_0_39 >
cd 2013_08_20/

# pierre@autan: ~/smi/transferts/from/sidiki_fofana/2013_08_20 8_20__10_37_39 >
```

La commande:

```
cd
```

sans argument ramène dans son "home": ~ en abrégé

La complétion automatique se fait par la touche tabulation (<tab>); la séquence de touches suivante:

```
cd tra<tab>ce<tab>b<tab><entrée>
```

donne la commande suivante, tapée rapidement et sans erreur:

```
# smiexplo@duran: ~$ < 2013_08_14__17_55_55 >
cd transferts/certificats/bv/
```

Seules les caractères en gras ont été tapés, tout le reste a été complété.

La commande:

```
ls
```

liste les 10 derniers fichiers par ordre de date:

```
# smiexplo@duran: ~/transferts/certificats/bv$ < 2013_08_14__17_55_55 >
ls
drwx----- 2 smiexplo smiexplo 4.0K août 6 19:17 2013_07_15_2
drwx----- 2 smiexplo smiexplo 4.0K août 6 19:20 2013_08_01
drwx----- 2 smiexplo smiexplo 4.0K août 6 19:21 2013_08_03
drwx----- 2 smiexplo smiexplo 4.0K août 6 19:25 2013_08_06
drwx----- 2 smiexplo smiexplo 4.0K août 6 19:25 2013_08_05
drwx----- 2 smiexplo smiexplo 4.0K août 6 20:09 2013_08_04
drwx----- 2 smiexplo smiexplo 4.0K août 6 20:55 2013_08_02_to_import
drwx--x--x 2 smiexplo smiexplo 4.0K août 12 16:41 2013_05_24
drwxr-xr-x 2 smiexplo smiexplo 4.0K août 14 10:48 2013_08_13
drwxr-xr-x 2 smiexplo smiexplo 4.0K août 14 17:27 2013_08_14

# smiexplo@duran: ~/transferts/certificats/bv$ < 2013_08_14__17_55_55 >
```

La commande:

```
ll
```

liste tous les fichiers:

```
# smiexplo@duran: ~/transferts/certificats/bv/2013_08_13$ < 2013_08_14__17_55_55 >
ll
total 100
-rwxr--r-- 1 smiexplo smiexplo 3279 août 12 13:52 u100783.csv
-rwxr--r-- 1 smiexplo smiexplo 1138 août 13 10:57 u100783-csv.zip
-rwxr--r-- 1 smiexplo smiexplo 9393 août 13 10:56 u100783-pdf.zip
-rwxr--r-- 1 smiexplo smiexplo 3351 août 13 08:20 u100784.csv
-rwxr--r-- 1 smiexplo smiexplo 1108 août 13 10:56 u100784-csv.zip
-rwxr--r-- 1 smiexplo smiexplo 9281 août 13 10:51 u100784-pdf.zip
-rwxr--r-- 1 smiexplo smiexplo 3295 août 13 08:20 u100785.csv
-rwxr--r-- 1 smiexplo smiexplo 1150 août 13 10:56 u100785-csv.zip
-rwxr--r-- 1 smiexplo smiexplo 9265 août 13 10:56 u100785-pdf.zip
-rwxr--r-- 1 smiexplo smiexplo 3200 août 13 08:20 u100786.csv
-rwxr--r-- 1 smiexplo smiexplo 1205 août 13 10:56 u100786-csv.zip
-rwxr--r-- 1 smiexplo smiexplo 9411 août 13 10:56 u100786-pdf.zip
-rwxr--r-- 1 smiexplo smiexplo 3290 août 13 08:20 u100787.csv
-rwxr--r-- 1 smiexplo smiexplo 1149 août 13 10:56 u100787-csv.zip
-rwxr--r-- 1 smiexplo smiexplo 9314 août 13 10:56 u100787-pdf.zip
```

Une requête pour voir le dernier import d'analyses:

```
SELECT * FROM public.lab_ana_results WHERE db_update_timestamp = (SELECT max(db_update_timestamp)
FROM public.lab_ana_results);
```

Une requête pour voir les teneurs graphiquement, en même temps que les passes minéralisées:

dh_sampling_mineralised_intervals_graph_au6

Exemple:

```
--192.168.1.29 bdeplot=>
SELECT * FROM dh_sampling_mineralised_intervals_graph_au6 WHERE id = 'S526' AND depto BETWEEN 105 AND 160;
mineralised_interval
```

opid	id	depfrom	depto	sample_id	weight_kg	core_loss_cm	aumaxi_ppm	graph_aumaxi
18	S526	104.00	105.00	05515	4.20		0.150	#
18	S526	105.00	106.00	05516	3.90		0.110	#
18	S526	106.00	107.00	05517	3.60		0.220	#
18	S526	107.00	108.00	05518	3.60		0.150	#
18	S526	108.00	109.00	05519	4.10		0.250	#
18	S526	109.00	110.00	05520	3.90		0.130	#
18	S526	110.00	111.00	05521	3.30		0.270	#
18	S526	111.00	112.00	05522	3.70		0.390	##
18	S526	112.00	113.00	05523	5.20		0.200	#
18	S526	113.00	114.00	05525	3.70		0.160	#
18	S526	114.00	115.00	05526	3.40		0.110	#
18	S526	115.00	116.00	05527	3.40		0.420	##
18	S526	116.00	117.00	05528	3.60		0.255	#
18	S526	117.00	118.00	05529	3.60		0.420	##
18	S526	118.00	119.00	05530	3.40		0.650	###
18	S526	119.00	120.00	05531	1.90		0.410	##
18	S526	120.00	121.00	05532	1.80		0.270	#
18	S526	121.00	122.10	05533	2.10		0.260	#
18	S526	122.10	123.30					
18	S526	123.30	124.00	05534	1.40	120.0	3.290	#####
18	S526	124.00	125.10	05535	1.60		16.010	#####
18	S526	125.10	126.10					
18	S526	126.10	127.00	05536	2.40	100.0	100.000	#####
18	S526	127.00	128.65	05537	3.80		1.210	#####
18	S526	128.65	129.50	05538	3.40		0.280	#
18	S526	129.50	130.40	05539	4.20		4.230	#####
18	S526	130.40	131.50	05540	1.60		8.690	#####
18	S526	131.50	132.05	05541	2.00		0.270	#
18	S526	132.05	133.25					
18	S526	133.25	134.05					
18	S526	134.05	135.00	05542	2.50	120.0	0.430	##
18	S526	135.00	136.00	05543	2.50		4.870	#####
18	S526	136.00	137.00	05544	2.50		6.500	#####
18	S526	137.00	138.00	05545	2.30		3.260	#####
18	S526	138.00	139.00	05546	2.60		4.060	#####
18	S526	139.00	139.85	05547	2.30		4.750	#####
18	S526	139.85	140.65	05549	2.20		14.990	#####
18	S526	140.65	142.00	05550	3.20	80.0	11.560	#####
18	S526	142.00	143.00	05551	2.70		1.430	#####
18	S526	143.00	144.00	05552	2.80		0.590	###
18	S526	144.00	145.00	05553	2.50		1.990	#####
18	S526	145.00	146.00	05554	2.30		1.410	#####
18	S526	146.00	147.00	05555	2.70		0.230	#
18	S526	147.00	148.00	05556	2.80		0.320	##
18	S526	148.00	149.00	05557	2.30		0.380	##
18	S526	149.00	150.25	05558	3.40		0.670	###
18	S526	150.25	150.95					
18	S526	150.95	152.00	05559	3.20	70.0	1.330	#####
18	S526	152.00	153.00	05560	2.50		1.130	#####
18	S526	153.00	154.00	05561	2.70		0.780	#####
18	S526	154.00	155.50	05562	4.10		0.530	###
18	S526	155.50	156.50					
18	S526	156.50	158.00	05563	4.30	100.0	0.360	##
18	S526	158.00	159.65	05564	2.00	45.0	0.100	#
18	S526	159.65	160.00	05566	3.00		0.060	#

(55 lignes)

3.15. Récupérer les dernières versions des programmes

Les programmes développés sont maintenus à jour sur github à l'adresse:

<https://github.com/pierrechoux/geolllibre.git>

Il est préférable de les obtenir par les commandes suivantes:

```
mkdir ~/geolllibre
cd ~/geolllibre
git clone https://github.com/pierrechoux/geolllibre.git
```

Par la suite, il suffit de se remettre dans ce répertoire et de taper:

```
git pull
```

pour que tous les programmes soient mis à jour automatiquement.

Alternativement, on peut aller chercher l'archive:

<https://github.com/pierrechoux/geolllibre/archive/master.zip>

et placer le contenu du .zip dans:

```
~/geolllibre
```

Pour dézipper le zip directement sur place:

```
cd ~/geolllibre
unzip master.zip
mv geolllibre-master/* .
```