

ETUDE ET REFONTE DE LA BASE DE DONNEES GEOPHYSIQUES DU SHOM BDGEOS NG

Dossier de Conception Préliminaire



Réf. BDGEOS-DC-03-CS

	Nom	Société	Fonction	Date	Visa
Rédigé par :		CS			
Validé par :	Roselyne PEDROSA	CS	Responsable Qualité		
Pour application :	Frédéric ALFONSI	CS	Directeur de projet		

CS Systèmes d'Information

5 rue Brindejonn des Moulinais
Parc de la Grande Plaine
BP 15872
31506 Toulouse Cedex 5



Versions successives

ED.	RÉV.	DATE	MOTIF
01	01	03/02/2011	Création du document
01	02	03/02/2011	Intégration des premières remarques du SHOM, Intégration de la « liste des tables ».
01	03	03/02/2011	Intégrations des « Secondes remarques du shom. Ce document est accompagné d'une liste des réponses à ces questions : Reponse_de_CS_au_SHOM_sur_DC1_2.doc



Table des matières

1. GENERALITES	5
1.1 Glossaire des termes et abréviations	5
1.2 Documents de référence	5
1.3 Documents applicables	5
1.4 Documents référencés	6
2. INTRODUCTION	7
2.1 Objet du document	7
2.2 Structure du document	7
2.3 Démarche de conception	7
2.3.1.1 Etat de l'existant BDGEOS, et des exigences	7
2.3.1.2 Démarche de spécification et de conception.	8
3. LE MODELE DE DONNEES	9
3.1 Nomenclature	9
3.1.1 Conceptuelle.....	9
3.1.2 Physique	10
3.2 Thèmes du modèle.....	11
3.2.1 Utilisateurs.....	11
3.2.1.1 Vue conceptuelle.....	12
3.2.1.2 Vue physique.....	13
3.2.2 Listes de Référence.....	13
3.2.2.1 Paramétrage	14
3.2.3 Levé.....	14
3.2.3.1 Levé « seul »	14
3.2.3.2 Lots et domaines de références.....	16
3.2.3.3 Data.....	19
3.2.3.4 Stations magnétiques.....	22
3.2.4 Modèles Géophysiques	23
3.2.4.1 Modèles et domaines de références.....	23
3.2.4.2 Modèles , variables et Formats	25
3.2.5 Templates de formats des données.....	29
3.2.6 Stations Gravimétriques	31
3.2.6.1 Vue conceptuelle.....	33
3.2.6.2 Vue physique.....	34
3.2.7 Historiques.....	34
3.2.8 Historique des entrées et modifications en base.....	35
3.2.8.1 Vue conceptuelle.....	35
3.2.8.2 Vue physique.....	35
3.2.9 Journal des process	35
4. DESCRIPTION GLOBALE DE L'ARCHITECTURE	37
4.1 Architecture matérielle.....	37

4.2	Vue générale des composants logiciels	39
5.	PRESENTATION DETAILLEE DES COMPOSANTS	40
5.1	Composant applicatif	40
5.1.1	Cas d'utilisation	40
5.1.2	le Framework Symfony	41
5.1.3	Modules Fonctionnels	42
5.1.4	Détails des cas d'utilisation	43
5.2	Composant Cartographique	43
5.2.1	Introduction	43
5.2.2	Choix techniques sur le stockage des données	43
5.2.3	Choix techniques sur les composants utilisés	44
5.2.4	Préparation des données de modèles géophysiques	44
5.2.4.1	Présentation	44
5.2.4.2	Caractéristiques des fichiers à produire	44
5.2.4.3	Données d'entrée pour la génération	45
5.2.4.4	Etapes de génération d'un fichier	45
5.2.4.5	Données en sortie	45
5.2.5	Préparation des données de Lots	46
5.2.6	Préparation de la visualisation des stations gravimétriques	46
5.2.6.1	Stations de référence gravimétriques	46
5.2.6.2	Réseaux de stations gravimétriques	46
5.2.7	Visualisation et interaction avec les données géographiques	47
5.2.7.1	Présentation	47
5.2.7.2	Configuration du serveur	47
5.2.7.3	Navigation cartographique	49
5.3	Le Serveur de données Oracle	49
5.3.1	Introduction	49
5.3.2	Copie de bases de données	50
5.3.2.1	Cadre général	50
5.3.3	étapes du processus de replication offline	51
5.3.4	Chargement extraction des DONNEES CSV	52
5.3.4.1	Chargement et extraction des DONNEES CSV	53
5.3.4.2	Chargement et extraction des modèles CSV	55
6.	MATRICE DE COUVERTURE DU BESOIN	57
7.	LISTE DES TABLES	58

Table des illustrations

Liste des figures

Figure 1 : Exemple de modélisation conceptuelle de données	9
Figure 2 : Modèle physique de donnée correspondant	10
Figure 3 : Modélisation conceptuelle du domaine « Utilisateurs »	12
Figure 4 : Modélisation physique du domaine « Utilisateurs »	13
Figure 5 : Structure générique d'une table de référence.....	13
Figure 6 : Relation entre Porteur (DOM_CODBAT) et Type de porteur (DOM_TYPBAT).....	13
Figure 7 : Relation entre Ville (DOM_VILLE) et Nation (DOM_NATION).....	14
Figure 8 : Relation entre Variable (DOM_CVAL) et Unité (DOM_UNIT).....	14
Figure 9 Structure de la table de paramétrage.....	14
Figure 10 : Modélisation conceptuelle des relations entre un levé, ses supports, ses documents.....	15
Figure 11 : Modélisation physique des relations entre un levé, ses supports, ses documents.....	16
Figure 12 : Modélisation conceptuelle du rapport entre lot et listes de références.	17
Figure 13 : Modélisation physique du rapport entre lot et listes de références.	18
Figure 14 : Modélisation conceptuelle de la constitutions de données de Levés/Lots.....	20
Figure 15 : Modélisation physique de la constitutions de données de Levés/Lots.....	21
Figure 16 : Modélisation conceptuelle du domaine « Stations Magnétiques »	22
Figure 17 : Modélisation physique du domaine « Stations Magnétiques »	23
Figure 18 : Modélisation conceptuelle du rapport entre modèles et listes de références.	24
Figure 19 : Modélisation physique du rapport entre modèles et listes de références.	25
Figure 20 : Modélisation conceptuelle des modèles et de la description de leurs variables.	27
Figure 21 : Modélisation physique des modèles et de la description de leurs variables.	28
Figure 22 : Modélisation physique du domaine « Composition des Formats »	31
Figure 23 : Modélisation conceptuelle du domaine « Station gravimétrique »	33
Figure 24 : Modélisation physique du domaine « Station gravimétrique »	34
Figure 25 : Modélisation conceptuelle du domaine « Historique »	35
Figure 26 : Modélisation physique du domaine « Historique »	35
Figure 27 : Modélisation physique du domaine « Historique »	36
Figure 28 : Architecture matérielle	37
Figure 29 : Architecture Logicielle Générale	39
Figure 30 : Diagramme des cas d'utilisation généraux	40
Figure 31 : Pattern MVC selon « symfony »	41
Figure 32 : Diagramme schématique des modules fonctionnels principaux.....	42
Figure 33 : Diagramme de séquence Navigation cartographique	49

Liste des tableaux

Tableau 1 : Regroupement des droits élémentaires d'utilisation en ROLES	11
Tableau 2 : Description succincte des différents champs de la table DOM_CVAL.....	29

1. Généralités

Ce document présente la conception préliminaire de BDGEOS-NG.

1.1 Glossaire des termes et abréviations

L'ensemble des définitions des termes et des abréviations utilisés dans le document figure ci-dessous :

1.2 Documents de référence

L'ensemble des documents de référence, qu'ils proviennent du SHOM ou de CS, est récapitulé ci-dessous :

- ✓ Notice technique n°15 MHA/NP du 06 novembre 1996
- ✓ Annexe 1 à la note express n°37/MHA/NP du 21 décembre 1994
- ✓ GU2007-021: Règles de mise en œuvre d'un logiciel sur un équipement informatique propriété du SHOM
- ✓ PS2007-085: Règles d'intégration spécifiques aux bases de données dans l'environnement informatique du SHOM
- ✓ PS2007-086: Règles d'intégration spécifiques aux logiciels sous environnement UNIX au sein du SHOM
- ✓ PS2007-087: Règles d'intégration spécifiques aux logiciels sous environnement Windows au sein du SHOM
- ✓ NR2004-089: Recueil de données gravimétriques marines
- ✓ PS2007-072: Acquisition et traitement des mesures magnétiques marines

1.3 Documents applicables

L'ensemble des documents applicables, qu'ils proviennent du SHOM ou de CS, est récapitulé ci-dessous :

- ✓ Cahier des Charges Technique et Particulières du marché 10MF0029 : « Etude et refonte de la base de données géophysique du SHOM BDGEOS ses appendices, ses annexes.
- ✓ Annexe technique n°1 au marché n°10MF0029 « Etudes et refonte de la base de données géophysique du SHOM »
- ✓ Proposition Technique de CS pour l'annonce du marché (n°279 publiée le 07/08/2008 dans le BOAMP 152 B, Dép. 94.).CSSI/111.1/GG/LR/9/705 27/11/2009

1.4 Documents référencés

Les documents fournis par CS qui sont référencés dans le présent document sont

- ✓ Dossier de spécifications : BDGEOS-DS-02-CS.doc



2. Introduction

2.1 Objet du document

Ce document présente la conception préliminaire du projet BDGEOS-NG .Il constitue une description de l'architecture logicielle élaborée pour la refonte du système d'archivage des données géophysiques du SHOM. Ce document sera complété et précisé au cours des phases de développement du système.

Il est principalement construit sur la représentation par diagrammes, ou schémas des différentes composantes matérielles et logicielles du système

2.2 Structure du document

Ce document s'articule autour de 4 grands chapitres :

- Le présent chapitre « [Introduction](#) » présente l'objectif, la structure du document, et la démarche de conception.
- Le chapitre « [Le modèle de données](#) » décrit la nomenclature utilisée, et les différents schémas conceptuels et physiques des thématiques du modèle
- Le chapitre « [Description globale de l'architecture](#) » décrit la nomenclature utilisée, et les différents schémas conceptuels et physiques des thématiques du modèle
- Le chapitre « [Présentation détaillée des composants](#) » décrit la nomenclature utilisée, et les différents schémas conceptuels et physiques des thématiques du modèle

2.3 Démarche de conception

Le projet BDGEOS est une refonte d'un système déjà existant. Le CCTP et le recueil de besoin effectués en début de phase d'initialisation a permis de dégager les éléments suivants :

2.3.1.1 Etat de l'existant BDGEOS, et des exigences

Le système BDGEOS, basé sur une base de données Oracle est sous utilisé actuellement.

L'interface est basée sur un client Lourd.

La rédaction du cahier des charges témoigne d'une connaissance « physique » du modèle de données. Des requêtes sont actuellement effectuées directement sur la base Oracle par certains opérateurs

Le mode de stockage des données n'est pas adapté aux volumes de stockages envisagés à moyen et surtout long terme.

Les modalités de chargement de cette base de données, et certaines fonctionnalités réclamées pour le CCTP ne sont pas compatibles avec des « temps de serveur WEB » souhaités pour cette nouvelle version BDGEOS-NG.

Aucune exploitation spécifique (calculs ...) n'est envisagée pour les données scientifiques. Les metadonnées en revanche de certains objets « métier » doivent pouvoir être mises à jour.

Les grandes fonctionnalités attendues sont des processus d'import et de restitution de données, gourmands en temps de processus et volumes de stockage.

La description de ces données doit pouvoir être accessible rapidement, notamment à travers une visualisation cartographique.

2.3.1.2 Démarche de spécification et de conception.

Ce contexte a conduit à une démarche de recueil de besoin et une description axées sur :

- ✓ La description des différentes entités métier à partir de leur définition persistée (*i.e.* leur mode de stockage, leur description en base).
- ✓ La définition de séquences d'utilisation et la définition d'écrans. Le dossier de spécifications fonctionnelles.
- ✓ La confirmation d'une problématique importante incluant la nécessité de la prise en charge, outre le stockage lui-même, des processus d'import et d'export en background, processus détachés du fonctionnement « web ».

Sur ces bases, la conception s'est déroulée les axes suivants :

- ✓ La définition la plus exhaustive possible d'un modèle conceptuel de données décrivant la nature des attributs des entités métier et leurs relation. A l'issue de ce modèle **conceptuel**, un modèle **physique** et un dictionnaire données préliminaires ont été établis.
- ✓ Pour l'application, l'utilisation d'un outil de mapping et un framework permettant d'effectuer le lien avec la base de données tout en séparant logique métier, logique de persistance et logique de présentation. Le framework symfony.
- ✓ La description de l'architecture et des services du Serveur de BD Oracle.
- ✓ La mise en place d'un composant cartographique

3. Le modèle de données

N£ DS_mcd_D_0010 £N

T£

Le modèle de données est élaboré d'abord de manière conceptuelle afin de répondre de manière logique à la définition et aux relations logiques entre les entités.

Les grandes thématique retenues permettent d'organiser en grand schémas :

- ✓ Utilisateurs
- ✓ Levés/Lot
- ✓ Modèles géophysique
- ✓ Stations gravimétriques

C'est de cette vue « métier » que découle le modèle physique.

£T

A£ Exigence 52#C £A

3.1 Nomenclature

3.1.1 Conceptuelle

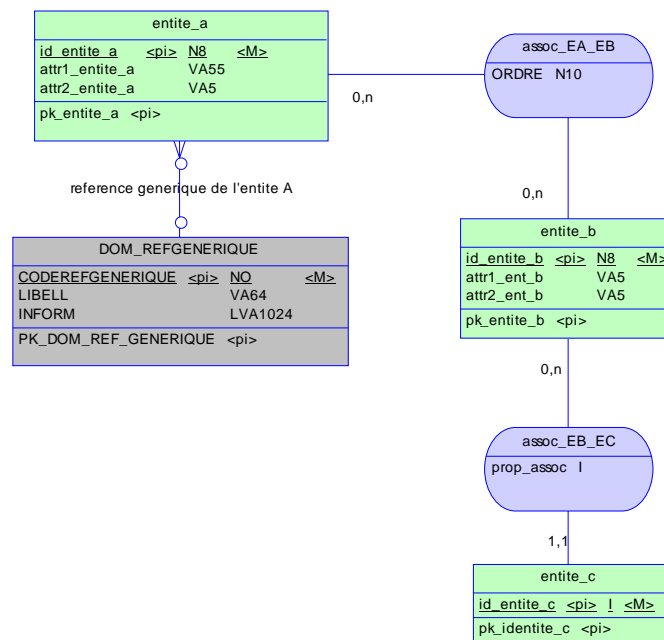


Figure 1 : Exemple de modélisation conceptuelle de données

Les relations entre les différentes entités peuvent être lues de la manière suivante.

L'entité A est reliée à une et une seule « référence générique ». Celle-ci pourrait en revanche être liée à d'autres entités.

- ✓ Les entités A et B sont associées. Cette association est caractérisée par
 - ↳ Une multiplicité de combinaisons possibles entre a et b (association n-n)
 - ↳ Un attribut qui caractérise cette association (ici ordre)
- ✓ Les entités B et C sont associées, cette association est caractérisée par
 - ↳ Une propriété de l'association
 - ↳ Un élément de l'entité C est associé une et une seule fois à un élément de l'entité B (1,1)
 - ↳ Un élément de l'entité B est associé à 0 ou plusieurs éléments de l'entité C

3.1.2 Physique

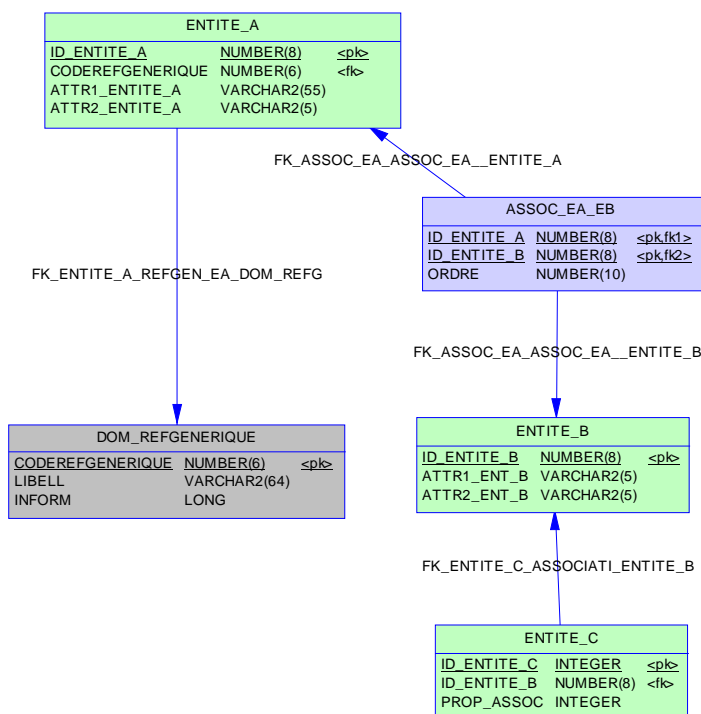


Figure 2 : Modèle physique de donnée correspondant

La traduction physique des relations conceptuelles se matérialise au niveau de la conception des clefs de chaque table.

Le CODEREFGENERIQUE de la table de référence, qui est sa clef primaire (pk) , et est déporté vers l'entité A. Ce code devient clef étrangère (fk) dans la table entité A.

L'association n-n entre A et B se traduit par la création d'une table d'association.

Sa clef primaire est composée des deux clefs primaires de chacune de ces entités. Celles ci sont respectivement clefs étrangères vers entité A et entité B.

L'association 1-n entre B n'entraîne pas la création d'une table d'association.

La clef primaire de l'entité B est déportée dans l'entité A ou elle constitue une clef étrangère.

L'attribut d'association devient lui aussi attributs de la table entité B.

Le découpage conceptuel en domaines et sa représentation en diagrammes « séparés » ne représente que les attributs intrinsèques d'une entité, et les relations aux entités de ce domaine logique. La vue « physique » elle, contient toutes les références qui se matérialisent notamment par des clefs étrangères. Certaines représentations d'entités physiques contiennent donc davantage de champs que ceux directement en rapport avec leurs homologues triées représentées sur un schéma partiel.

Dans la suite de ce document, l'attention est portée sur les relations entre entités, qui participent à la description du fonctionnel du système. Les attributs intrinsèques, ne relevant pas d'une clef primaire ou étrangère, ne seront détaillées que dans le dictionnaire de données.

3.2 Thèmes du modèle

3.2.1 Utilisateurs

La description des rôles (en terme de fonctionnalités et droits) est décrite dans le dossier de spécifications BDGEOS-DS-02-CS.doc

Les droits des utilisateurs ne sont pas définis en terme de « profils ». Leurs droits sont une combinaison de **rôles**, avec **accès à des données** en fonction du **niveau de protection autorisé**.

Les prérogatives ne sont pas hiérarchisées, le niveau de protection l'est.

L'ensemble de ces droits est regroupé en **rôles**, un utilisateur pouvant être titulaire de plusieurs rôles :

- ✓ Consultant : Un utilisateur qui consulte le fait en vue d'export. Les droits sont donc
 - ↳ Recherche
 - ↳ Voir/Consulter
 - ↳ Exporter (Levé/Lot, Modèle, Station gravimétrique)
- ✓ Gestionnaire de Levé/Lot
 - ↳ Charger / modifier levé-lot
- ✓ Gestionnaire de station gravimétrique
 - ↳ Charger / modifier station (fiche)
- ✓ Gestionnaire de Modèles
 - ↳ Charger / modifier modèle
- ✓ Synchroniseur
 - ↳ Générer Dump de la base publique SHOM
 - ↳ Intégrer dump dans la base confidentielle
- ✓ Gestionnaire des utilisateurs

Rq : Un utilisateur du système est au moins consultant.

Tableau 1 : Regroupement des droits élémentaires d'utilisation en ROLES

	Consultant	Gestionnaire Levé	Gestionnaire Station Gravimétrique	Gestionnaire Modèle	Synchroniseur	Gestionnaire Utilisateurs
Visualiser	x					
Rechercher	x					
Exporter	x					
Charger / modifier Lot-Levé		x				
Charger Modifier/Station Gravimétrique			x			
Charger Modifier Modèle				x		
Générer Dump					x	
Restaurer Dump					x	
Gérer les utilisateurs						x

3.2.1.1 Vue conceptuelle

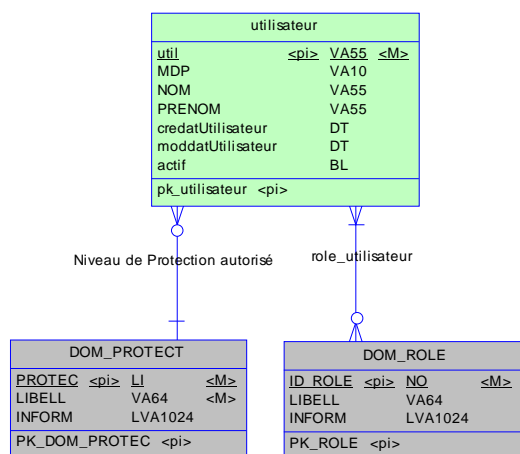


Figure 3 : Modélisation conceptuelle du domaine « Utilisateurs »

Rq : Il ne s'agit ici que du mot de passe, naturellement celui-ci est crypté.

3.2.1.2 Vue physique

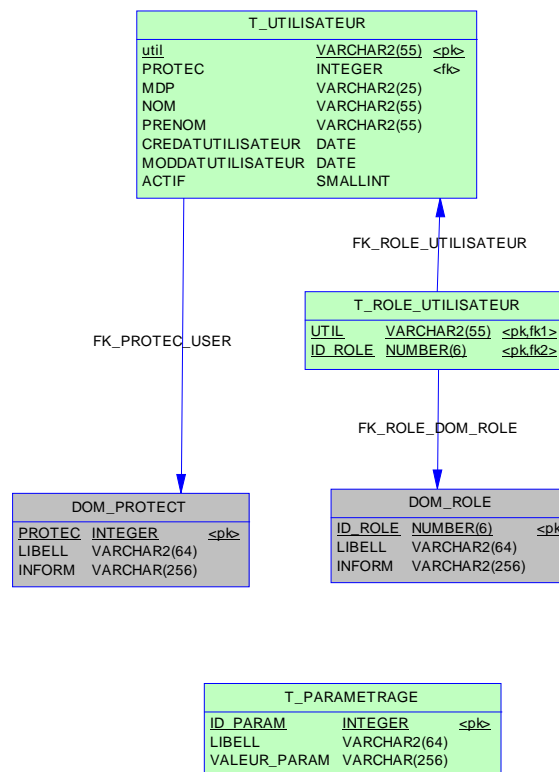


Figure 4 : Modélisation physique du domaine « Utilisateurs »

3.2.2 Listes de Référence

Pour des raisons de commodités de lecture, ce paragraphe est identique à son équivalent dans le dossier de spécifications. La description complète de la liste des domaines de référence est incluse au dictionnaire de données (cf annexe).

Les tables de domaines de références nécessaires au système sont celles du CCTP. Certaines ont été rajoutées pour rendre générique l'utilisation (ajout, suppression, modification) des listes. La structure globale d'une table de référence est la suivante :

DOM_REFGENERIQUE			
CODEREFGENERIQUE	<pi>	NO	<M>
LIBELL		VA64	
INFORM		LVA1024	
PK_DOM_REF_GENERIQUE	<pi>		

Figure 5 : Structure générique d'une table de référence

Certaines informations complémentaires sont parfois nécessaires, pour compléter des informations, ou spécifier des relations entre tables domaines de référence.

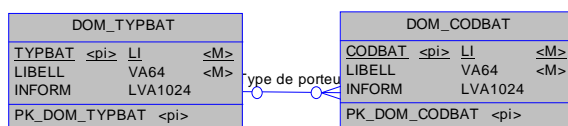


Figure 6 : Relation entre Porteur (DOM_COBAT) et Type de porteur (DOM_TYPBAT)

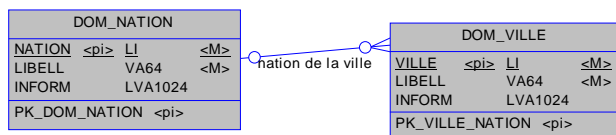


Figure 7 : Relation entre Ville (DOM_VILLE) et Nation (DOM_NATION)

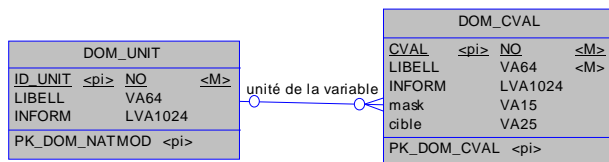


Figure 8 : Relation entre Variable (DOM_CVAL) et Unité (DOM_UNIT)

La gestion des variables nécessite en supplément

Un masque éventuel de saisie (pour les dates et heures par exemple)

Une cible, qui permet de spécifier quel est le champ destinataire de la variable dans le cas où elle est en entrée de modèle ou levé (pour éventuellement concaténer des champs dates).

3.2.2.1 Paramétrage

Un certain nombre de variables de configuration de l'application doivent pouvoir être lues par le serveur d'application. Il s'agit entre autres des éléments de l'arborescence de stockage (racine, répertoire d'export...). Ces éléments sont dans une table dédiée :

T_PARAMETRAGE		
ID_PARAM	INTEGER	<pi>
LIBELL	VARCHAR2(64)	
VALEUR_PARAM	VARCHAR(256)	

Figure 9 Structure de la table de paramétrage

3.2.3 Levé

La modélisation de ce thème peut logiquement se scinder en plusieurs groupes :

- ✓ La relation entre un levé, ses contraintes référentielles, ses annexes et documents
- ✓ La relation entre un Lot et ses contraintes référentielles
- ✓ La constitution en terme de données
- ✓ Le rapport aux stations magnétiques.

3.2.3.1 Levé « seul »

N£ DS_mcd_D_0020 £N

T£

Un levé est caractérisé par une référence à un niveau de protection.

Des supports (successifs) sont associés à ce levé. Un support ne concerne qu'un levé est également caractérisé par une validité de référence.

£T

AE Exigence 27#PC EA

NE DS_mcd_D_0030 EN

T£

De même, des documents peuvent être rattachés à un levé. Un document ne concerne qu'un levé. (Rq : Un document peut également être rattaché à un modèle géophysique, mais c'est exclusif, (cf § 3.2.4.2)

£T

AE Exigence 27#PC EA

3.2.3.1.1 Vue conceptuelle

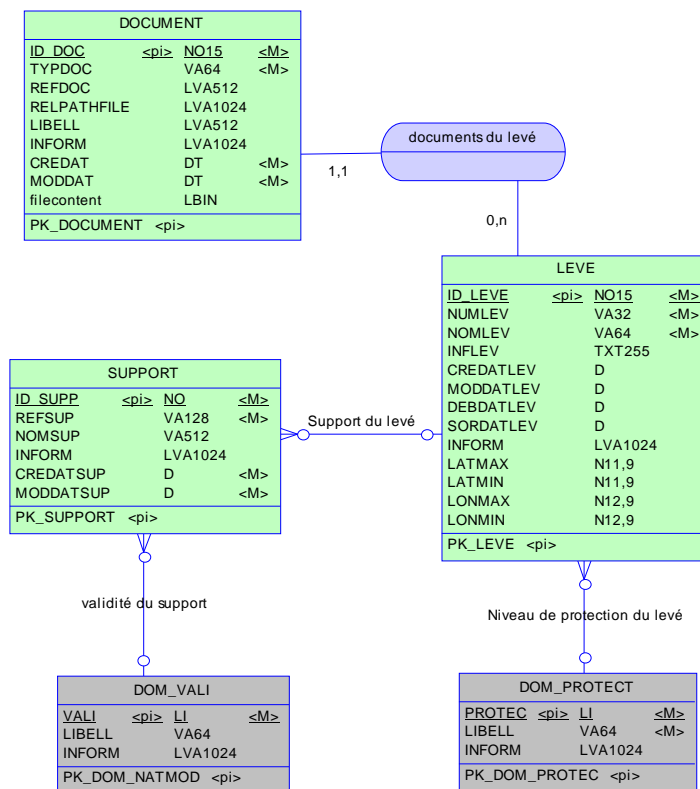


Figure 10 : Modélisation conceptuelle des relations entre un levé, ses supports, ses documents.

3.2.3.1.2 Vue physique

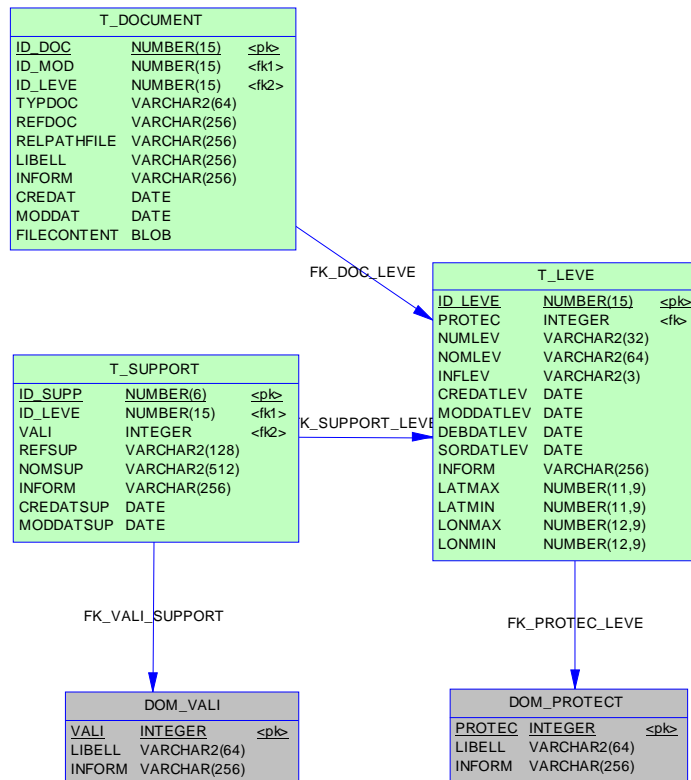


Figure 11 : Modélisation physique des relations entre un levé, ses supports, ses documents.

3.2.3.2 Lots et domaines de références

Ce paragraphe montre les relations de dépendance entre un lot et des listes de références.

Ces relations 1-n se traduisent par un ajout d'attribut (code) pour chaque table de référence et chaque relation, dans la table lot.

Ainsi un lot est relié de trois façons différentes à la liste de référence des corrections (DOM_TRCOR) :

- ✓ Données corrigées des accélérations verticales,
- ✓ Données corrigées de la dérive,
- ✓ Données corrigées de l'Eötvös

Il y aura donc trois codes relatifs à cette table de référence dans la table Lot. TRCOE, DOM_TRCOR, DOM_TRCOR2 dans le diagramme physique ci dessous. Le renommage éventuel de ces clefs s'effectue sur le modèle physique au moment de générer le script.

3.2.3.2.1 Vue conceptuelle

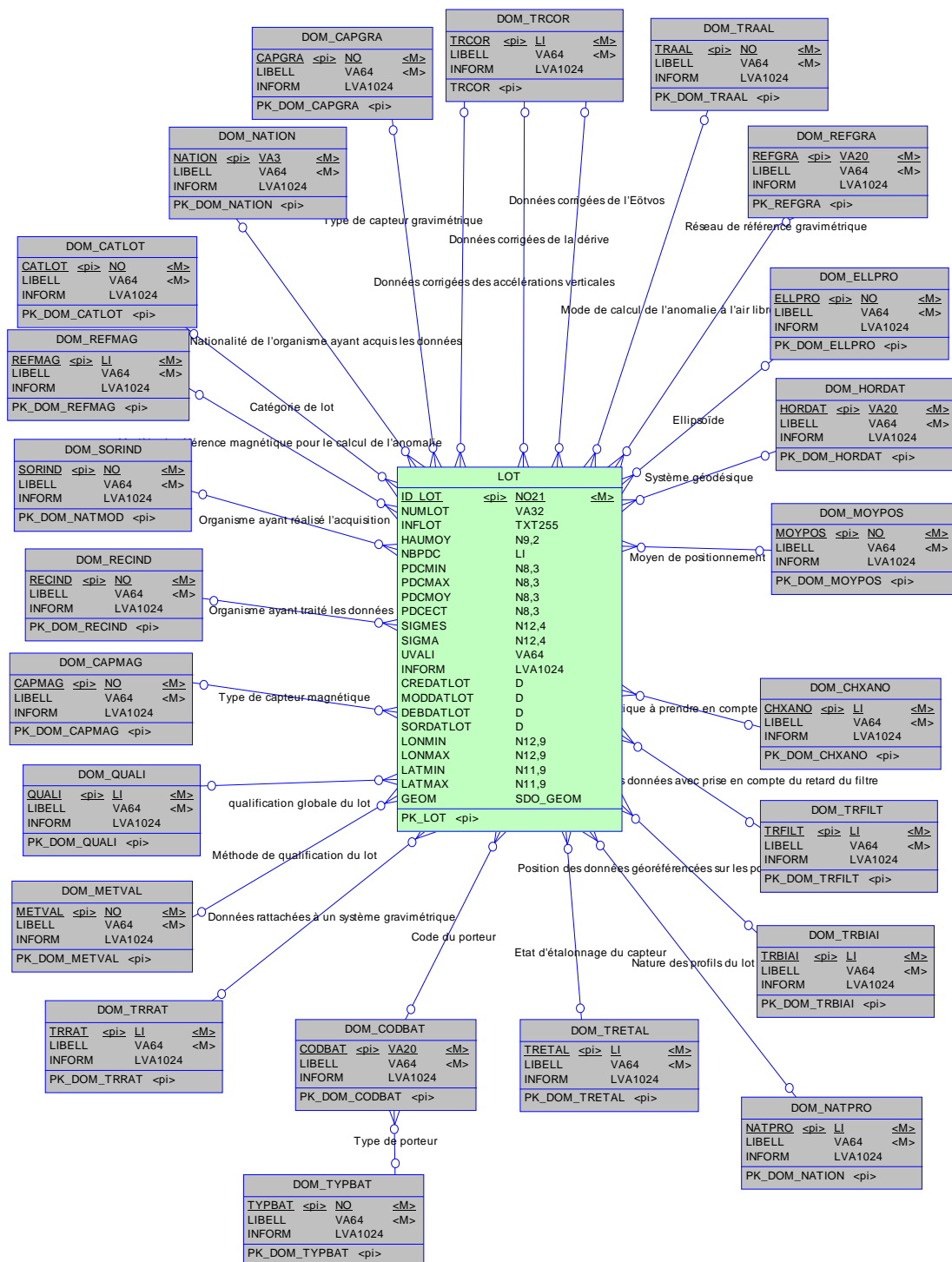


Figure 12 : Modélisation conceptuelle du rapport entre lot et listes de références.

3.2.3.2.2 Vue physique

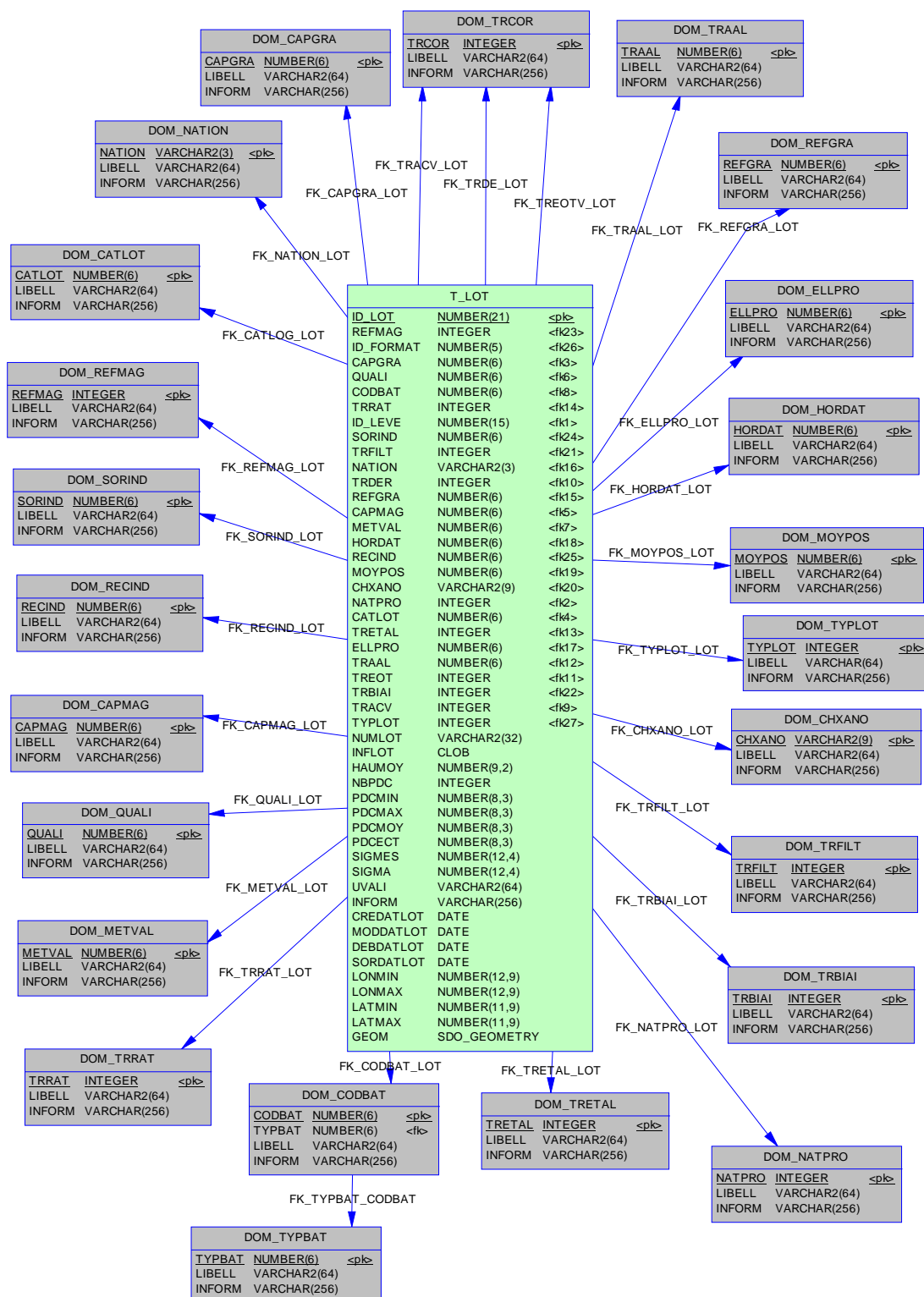


Figure 13 : Modélisation physique du rapport entre lot et listes de références.

3.2.3.3 Data

Ce paragraphe précise les relations entre entités métier Levé, lot, Suite, magnétisme, gravimétrie. Il est assez proche du CCTP

Un levé est constitué d'un ou plusieurs lots de type magnétisme ou gravimétrie, composés eux-même de suites. Ces suites sont représentées par une suite de points ordonnés. Par leur nature ces points porteurs de données scientifiques sont dans une table gravimétrie ou magnétisme.

La validité et la nature de la suite sont contraintes par les listes de référence idoines.

3.2.3.3.1 Vue conceptuelle

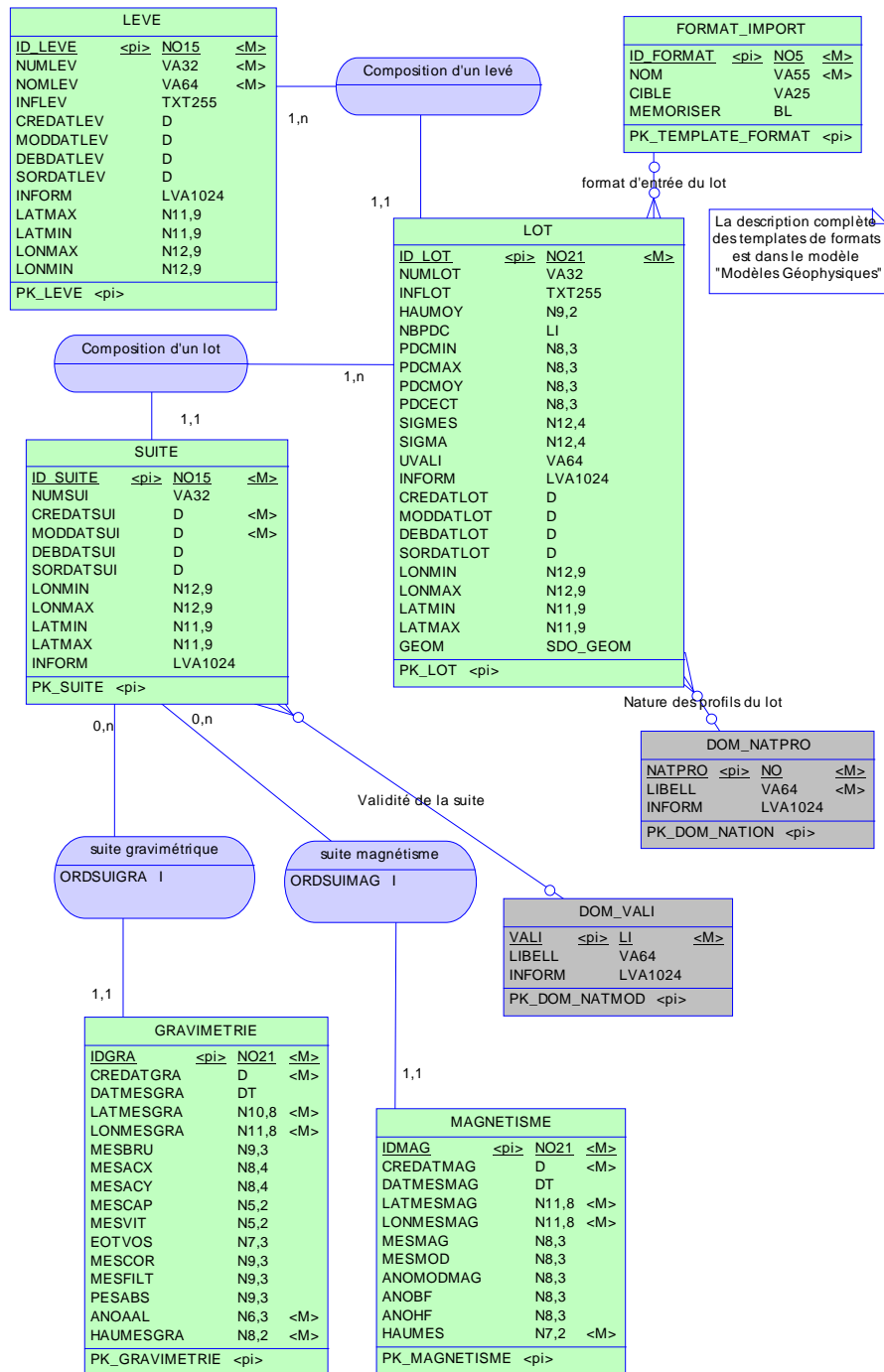


Figure 14 : Modélisation conceptuelle de la constitution des données de Levés/Lots.

3.2.3.4 Stations magnétiques

Dans la pratique l'import de données d'observatoires ou de stations magnétiques s'effectue toujours associées à un levé. L'import de mesure pour une station magnétique est donc toujours lié à un rattachement entre un levé et une station.

Une station magnétique est soit un observatoire, soit une station mobile. Ce fonctionnel peut se modéliser sous la forme suivante, en distinguant deux tables de stations : observatoire (DOM_OBSMAG) ou station mobile (STAMAG_MOBILE). La table STAMAG_DESC sert alors de lien entre la station, le levé, et les mesures associées.

3.2.3.4.1 Vue conceptuelle

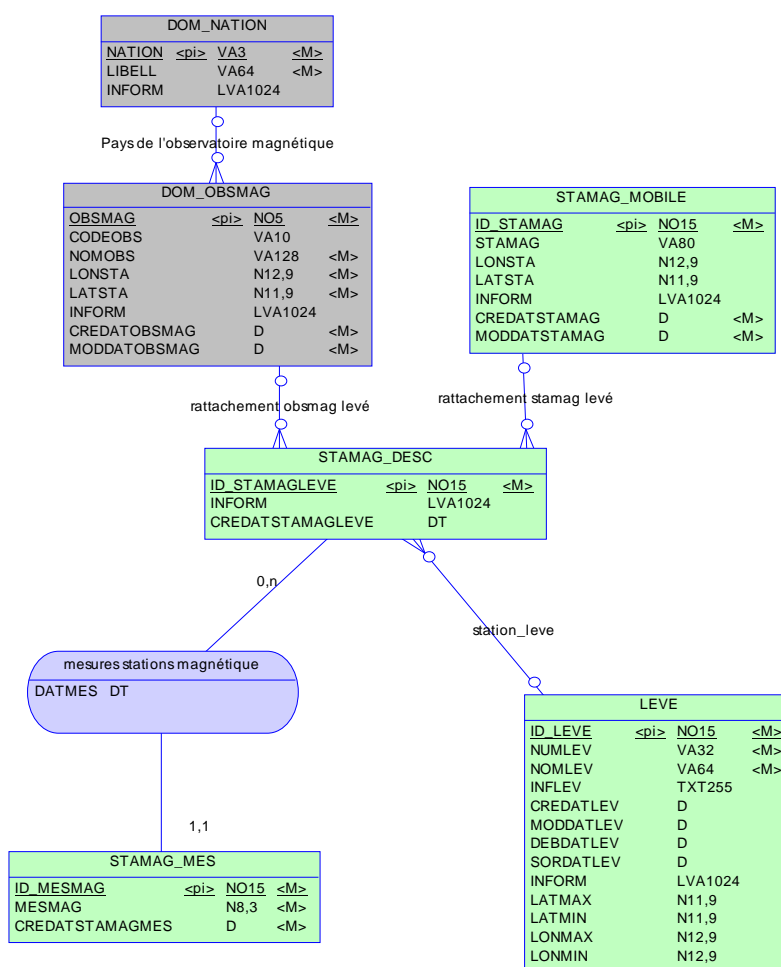


Figure 16 : Modélisation conceptuelle du domaine « Stations Magnétiques »

3.2.3.4.2 Vue physique

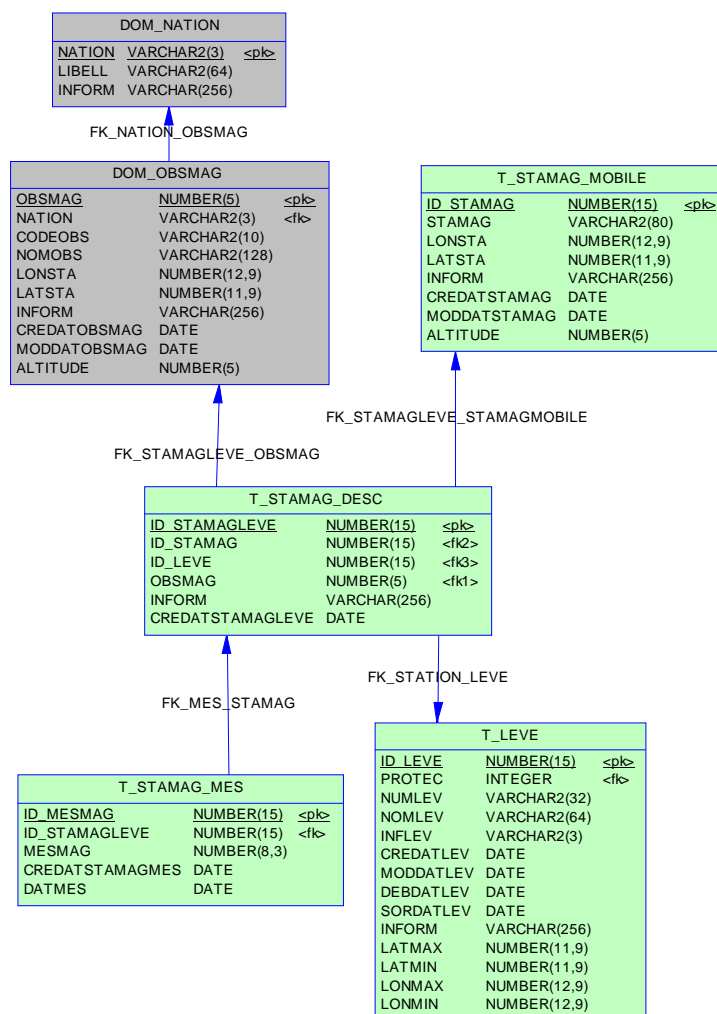


Figure 17 : Modélisation physique du domaine « Stations Magnétiques »

3.2.4 Modèles Géophysiques

3.2.4.1 Modèles et domaines de références

De même que pour les lots, un certain nombre de propriétés sont contraintes par des listes de référence.

3.2.4.1.1 Vue conceptuelle

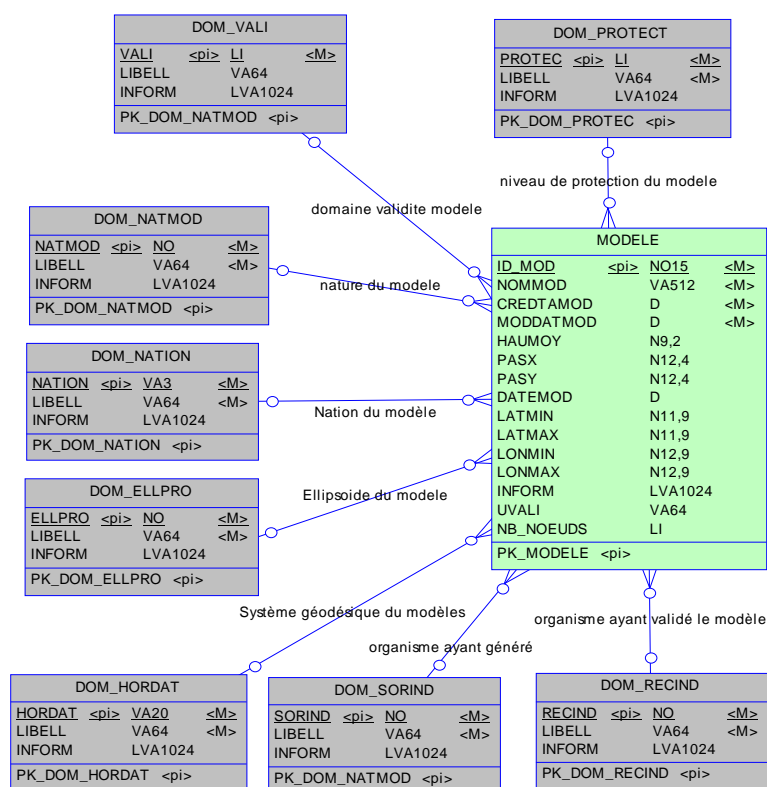


Figure 18 : Modélisation conceptuelle du rapport entre modèles et listes de références.

3.2.4.1.2 Vue physique

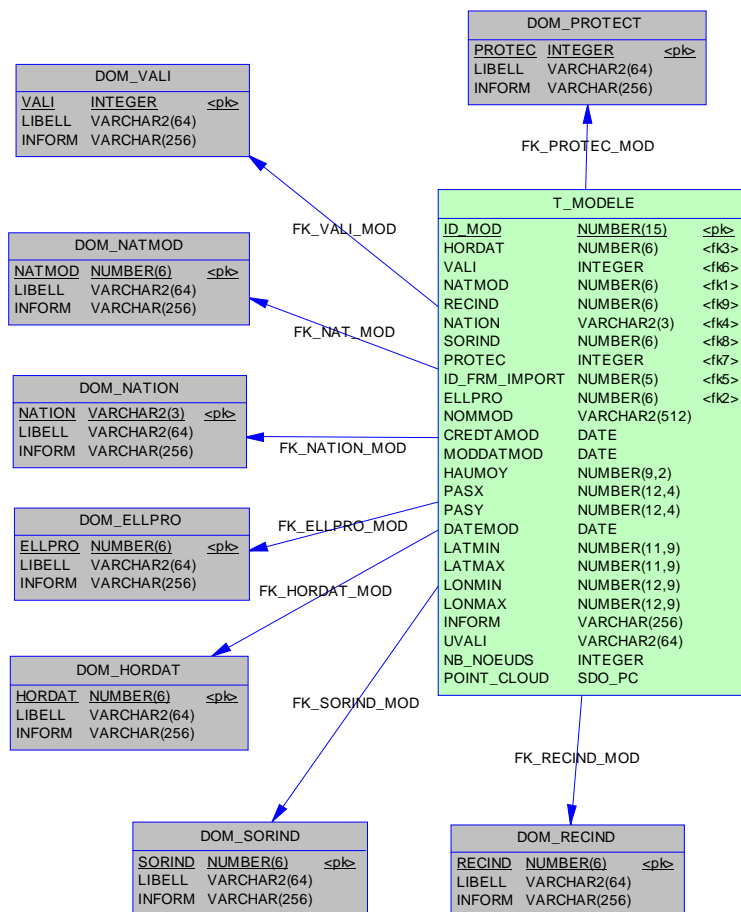


Figure 19 : Modélisation physique du rapport entre modèles et listes de références.

3.2.4.2 Modèles , variables et Formats

3.2.4.2.1 Variables

La conception de ce domaine diffère sensiblement de celle proposée dans le CCTP.

D'un point de vue logique, le port des données pour un modèle peut être envisagé comme suit :

- ✓ Un modèle géophysique est constitué de nœuds (coordonnées du globe) (table NODEMOD), ceux ci étant éventuellement ordonnés.
- ✓ Chaque nœud est porteur de une à plusieurs (table DATAMOD) valeur .
- ✓ Chaque valeur correspondant un type de variable scientifique typée , (table DOM_CVAL). Ainsi modélisé, le nombre de variables scientifique portée par un nœud n'est en théorie pas limité (représenté par une ligne d'enregistrement d'une table, et non pas une colonne comme proposé dans le CCTP).

Si ce mode de représentation est satisfaisant d'un point de vue conceptuel, d'un point de vue performance de requêtage et représentation cartographique ; il est limité par la volumétrie.

Ainsi un modèle global à la seconde d'arc représente plus de 839 000 000 000 de nœuds. Ce nombre d'enregistrements est à multiplier par le nombre de variables scientifiques possibles (jusqu'à 20 prévues par le CCTP). Ceci ne représentant qu'un seul modèle, la fréquence pouvant être pluriannuelle.

Les phases préliminaires du projet ont déjà permis de soulever la questions des espaces disque disponibles nécessaire à de tels volumes. Du point de vue méthode de stockage, ce « point par point, valeur par valeur » n'est pas non plus adapté. Cette partie « rouge » du modèle ne sera donc pas géré physiquement. Les solutions mises en place sont décrites dans le paragraphe consacré au serveur de données oracle (cf § [5.2](#)) (représenté schématiquement sur la vue physique par la table BLKTAB.

3.2.4.2.2 Vue conceptuelle

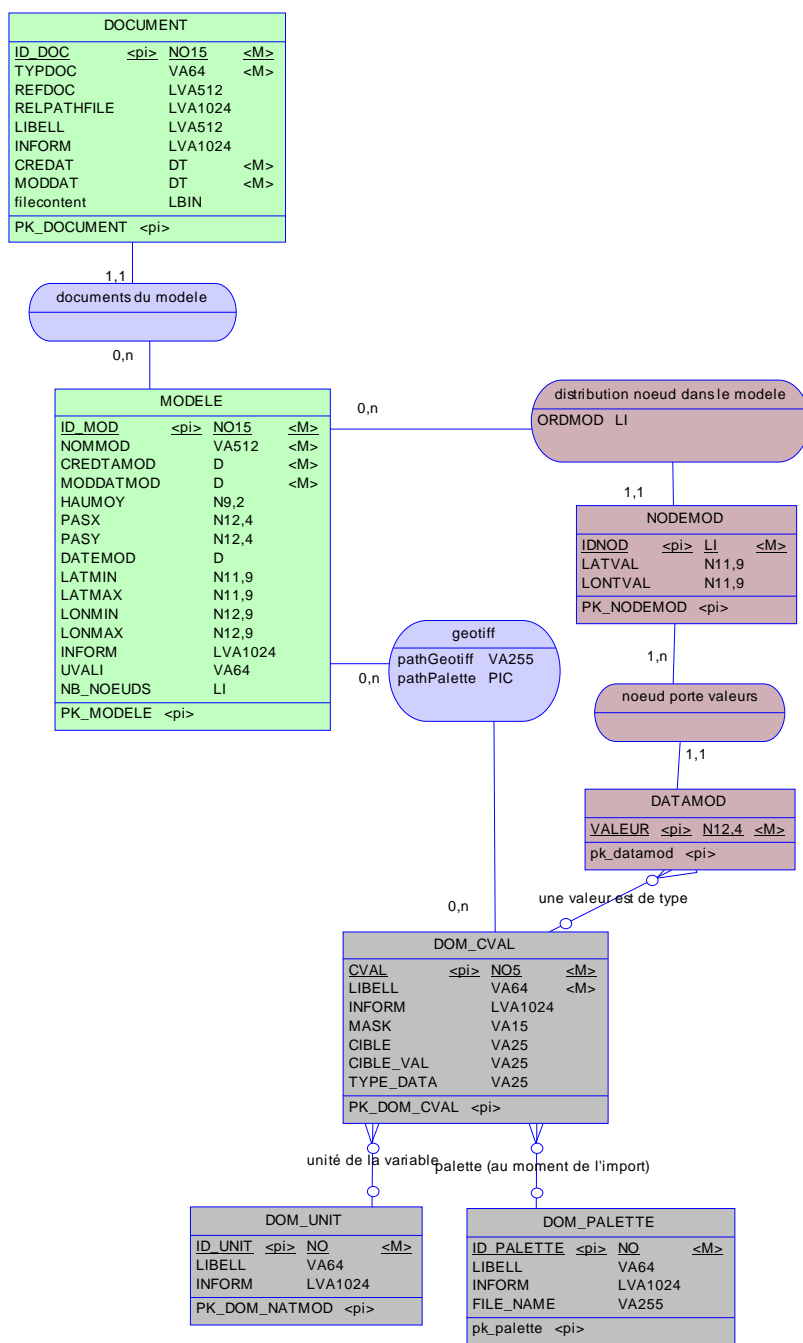


Figure 20 : Modélisation conceptuelle des modèles et de la description de leurs variables.

3.2.4.2.3 Vue physique

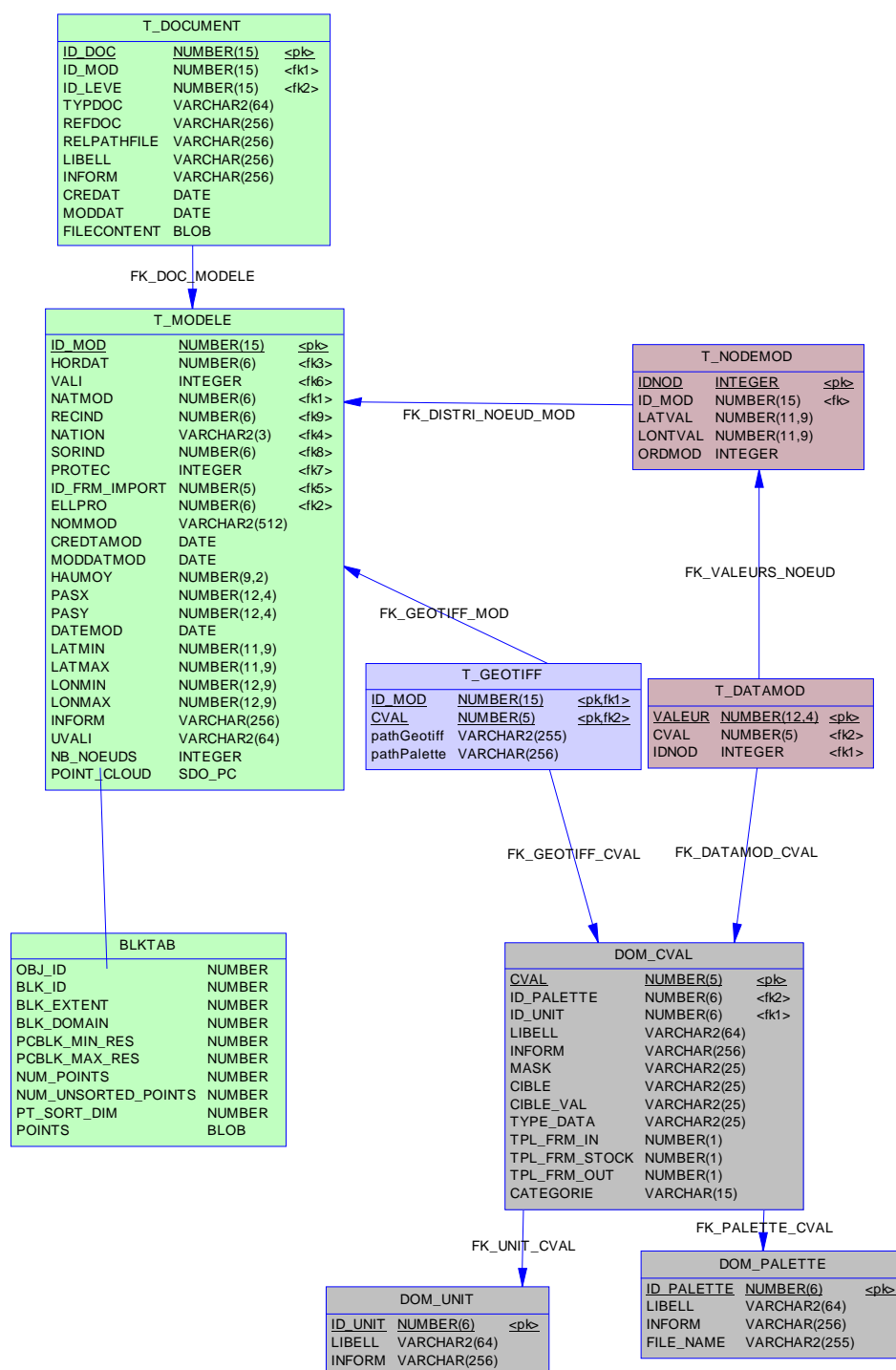


Figure 21 : Modélisation physique des modèles et de la description de leurs variables.

3.2.5 Templates de formats des données.

La variabilité de l'information portée par les modèles (ou les levés) conduit à la modélisation de la notion de Templates de formats. Une vision conceptuelle de formats de levés d'imports uniquement s'avère insuffisante au regard d'une problématique triple :

- ✓ un seul fichier (et donc un seul type de format) correspond à l'entrée des données d'un modèle (ou d'un lot)
- ✓ un seul mode de stockage correspond à ce fichier en entrée, il s'agit d'établir une correspondance entre source et cible. Le type de données de cette cible, le mode de stockage pouvant varier (modèle, lot gravi, lot mag)..
- ✓ A ce mode de stockage, plusieurs formats d'exports peuvent correspondre au cours de l' « exploitation » de telle ou telle entité (T_MODELE_FRM_EXPORT, T_LOT_FRM_EXPORT).

La table DOM_CVAL se trouve donc enrichie d'attributs qui ne sont pas scientifiquement intrinsèques à la variable mesurée. Les caractéristiques des principaux champs sont :

Tableau 2 : Description succincte des différents champs de la table DOM_CVAL

CODE	Description
CVAL	Identifiant de la variable
ID_PALETTE	Identifiant de la palette de couleur associée (n'a de sens que pour les variables relatives à un modèle)
ID_UNIT	Identifiant de l'unité associée (NA pour les dates)
LIBELL	Description courte.
INFORM	Description longue
MASK	Masque de format de la variable. Ceci est important surtout pour les éléments de dates et de coordonnées (HH, MM, DDD, DDD,dddd)
CIBLE	Domaine d'application de la variable. Il s'agit ici, pour les imports POINT_CLOUD à partir des EXTERNAL_TABLE de spécifier la table cible.(T_MODELE, T_GRAVIMETRIE, T_MAGNETISME)
CIBLE_VAL	Champs de destination de la variable : un champ spécifique de la base pour les données de lot, une référence vers des types spécifiques Oracle pour les modèles).
TYPE_DATA	Type Oracle de la variable (nécessaire pour les imports massifs d'Oracle)
TPL_FRM_IN	Détermine si cette variable peut-entrer dans la composition d'un format d'import
TPL_FRM_STOCK	Détermine si cette variable peut-entrer dans la composition d'un format de stockage (Seul un format de date timestamp, ou des coordonnées décimales peuvent entrer dans une forme de stockage).
TPL_FRM_OUT	Détermine si cette variable peut-entrer dans la composition d'un

format d'export. Les formats de date de sortie ne sont pas tous autorisés, de même que les formats de coordonnées)

Catégorie à laquelle appartient la variable (Dates, coordonnées, magnétisme, gravimétrie ...) Cela permet de les différencier pour des blocs d'interface.

CATEGORIE

Remarques :

- ✓ Il s'agit là d'une problématique plus physique que conceptuelle métier (elle a été conçue directement sur le modèle physique).
- ✓ Les spécifications fonctionnelles n'ont pas conduit à la nécessité de pérenniser les formats d'exports. Cette information doit cependant être conservée en base entre le moment où l'utilisateur définit un format d'export, et le moment où le système (serveur oracle) l'utilise pour la procédure d'export proprement dite.

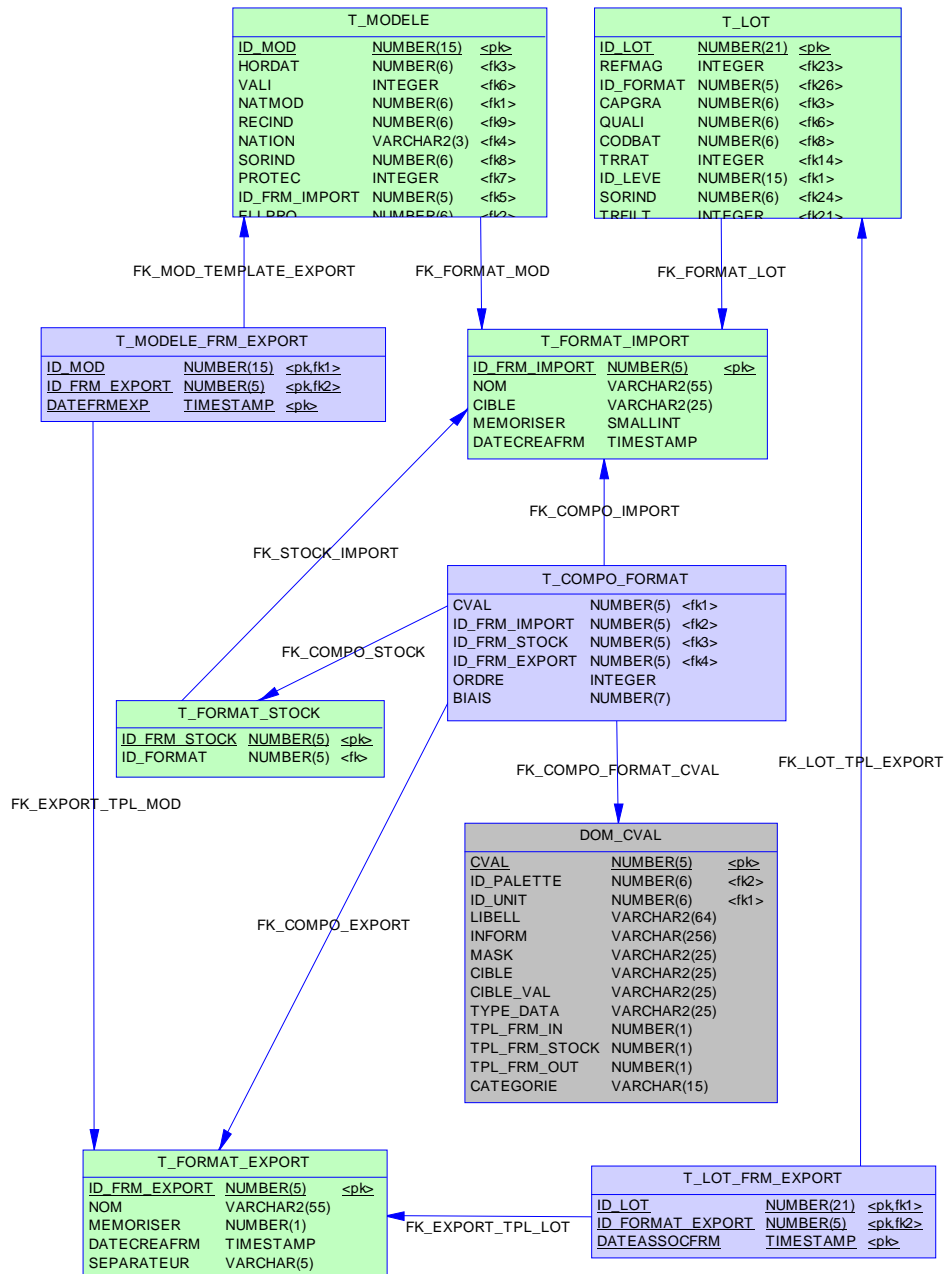


Figure 22 : Modélisation physique du domaine « Composition des Formats »

3.2.6 Stations Gravimétriques

D'un point de vue fonctionnel la notion de station est toujours accompagnée et parfois « confondue » avec la notion de fiche station. Ces deux tables sont cependant nécessaires car la table Station (STAGRA_DESC) représente les informations « immuables » : il s'agit d'un point nommé et géoréférencé. La fiche (STAGRA_FICHE) représente l'évolution « qualitative » dans le temps, des informations de cette station.

Chaque évolution de station est caractérisée par une nouvelle fiche :

- ✓ Un nouvel indice

- ✓ Des nouvelles mesures
- ✓ Un nouveau fichier accompagnateur (en general fiche.pdf) localisé sur le file_system

Une fiche correspond donc à une station et une seule station, qui elle même est associée à une ou plusieurs fiches indicées.

Une fiche est visitée une ou plusieurs fois.

Chaque fiche fait éventuellement référence à une ou plusieurs autres fiches. Ce rattachement « fille mère » étant effectué à une certaine date, et commenté.

NB : La notion de rattachement de station ou fiche à un levé, esquissé dans le CCTP à été invalidé au cours du recueil de besoin. C'est la notion de visites, plus significative, et plus réaliste qui la remplace « simplement ».

3.2.6.1 Vue conceptuelle

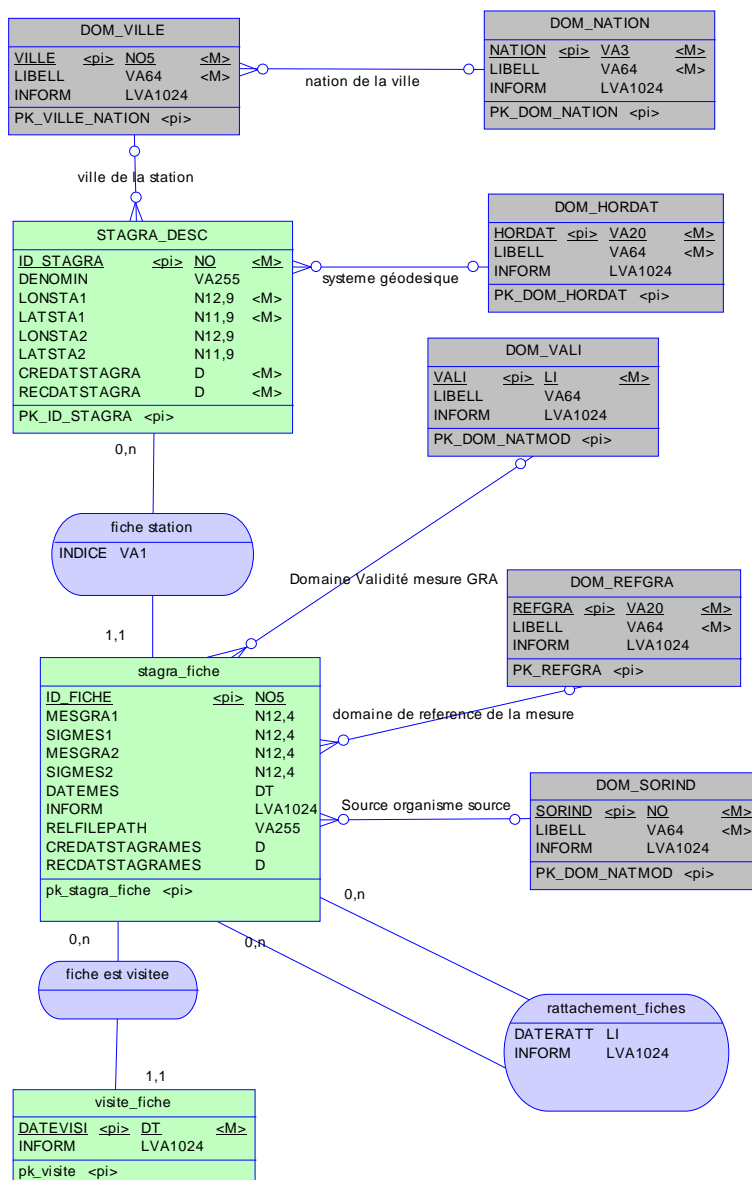


Figure 23 : Modélisation conceptuelle du domaine « Station gravimétrique ».

3.2.6.2 Vue physique

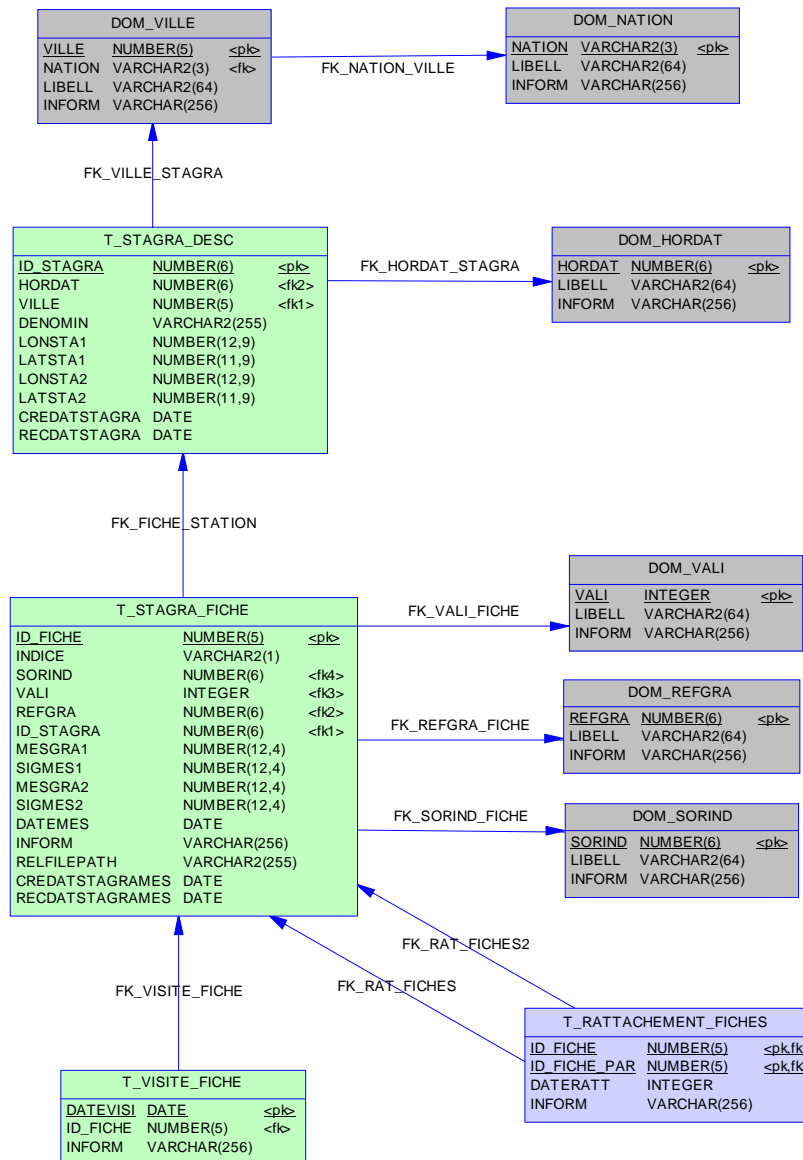


Figure 24 : Modélisation physique du domaine « Station gravimétrique »

3.2.7 Historiques

Les besoins de bilan et de « synchronisations » entre zones publiques et zones CD imposent la mise en place d'une table d'« historique ». Le besoin en terme de synchronisation nécessite une structure assez générique permettant de lister toutes les modifications sur toutes les tables. Celle-ci eut être utilisée, filtrée, pour la construction des bilans (cf BDGEOS-DS-02-CS.doc).

3.2.8 Historique des entrées et modifications en base

3.2.8.1 Vue conceptuelle

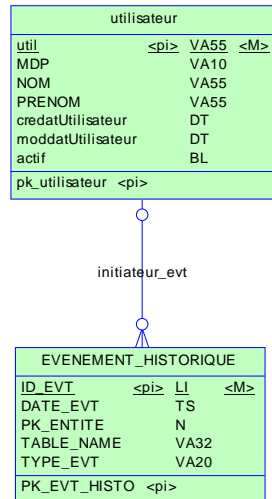


Figure 25 : Modélisation conceptuelle du domaine « Historique »

3.2.8.2 Vue physique

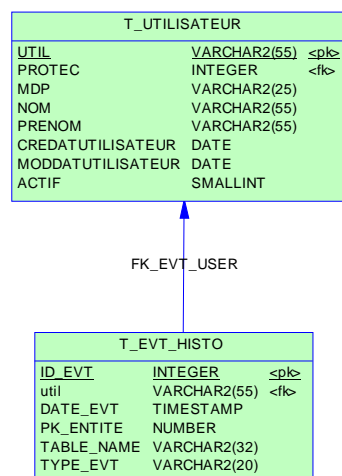


Figure 26 : Modélisation physique du domaine « Historique »

3.2.9 Journal des process

Afin de suivre les étapes majeures des processus d'import et d'export, une table de liste ces évènements en les datant. Bien que l'architecture client-léger ne permette pas de suivre en

mode « barre de progression », l'intégralité d'un processus, un « scruteur » permet de mettre à jour l'IHM et de diffuser ces étapes.

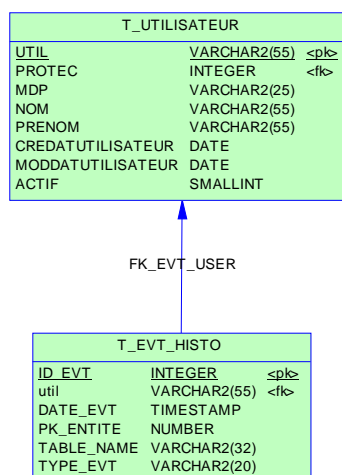


Figure 27 : Modélisation physique du domaine « Historique »

4. Description globale de l'architecture

4.1 Architecture matérielle

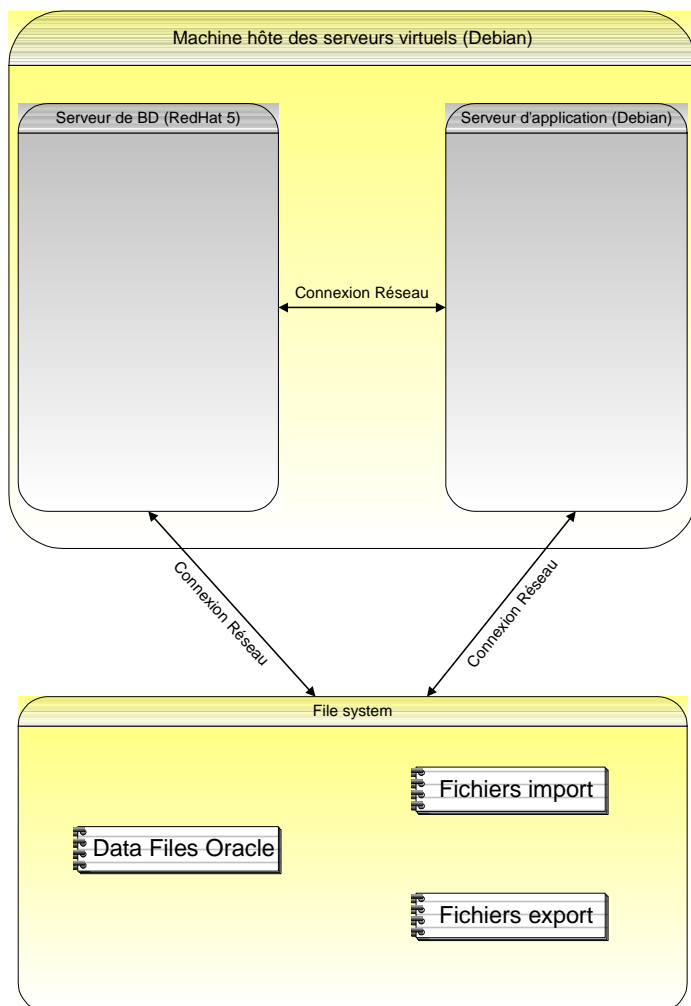


Figure 28 : Architecture matérielle

N£ DS_archi_F_0010 £N

T£

L'architecture matérielle envisagée est constitué

- D'un serveur de données
- D'un serveur d'application
- D'un serveur de fichiers

Ces éléments, sont visibles les uns pour les autres pour via le réseau

L'architecture matérielle est constituée d'éléments classiques d'une architecture client léger web :

ET

[A£ Exigence 48#C £A](#)

- ✓ Un serveur de bases de données, Oracle, prenant en charge le stockage des données et les processus de chargement et d'export massifs.
- ✓ Un serveur d'application (Apache). Chargé de la gestion métier, la gestion des formulaires faisant le lien entre base de données, opérateur web et formulaires, la visualisation cartographique.

La prise en compte des volumes de données manipulées, n'est pas compatibles avec ces seuls éléments d'architecture web : Les volumes des fichiers des fichiers d'entrée nécessiteraient des disponibilités de réseau, et des temps d'exécution de scripts impossibles dans ce genre d'environnement. Un éléments supplémentaire de l'architecture est donc introduit

- ✓ Un « file system »: Il est visible par tous les autres composants. Ainsi que par les opérateurs. Il sert pour le stockage ou le dépôt de :
 - ↳ La base de données Oracle pour ses DataFiles
 - ↳ Les fichiers à importer. l'opérateur doit pouvoir y déposer ses fichiers en entrée.
 - ↳ Les fichiers d'export volumineux ; l'opérateur doit pouvoir y accéder pour récupérer ses exports.

4.2 Vue générale des composants logiciels

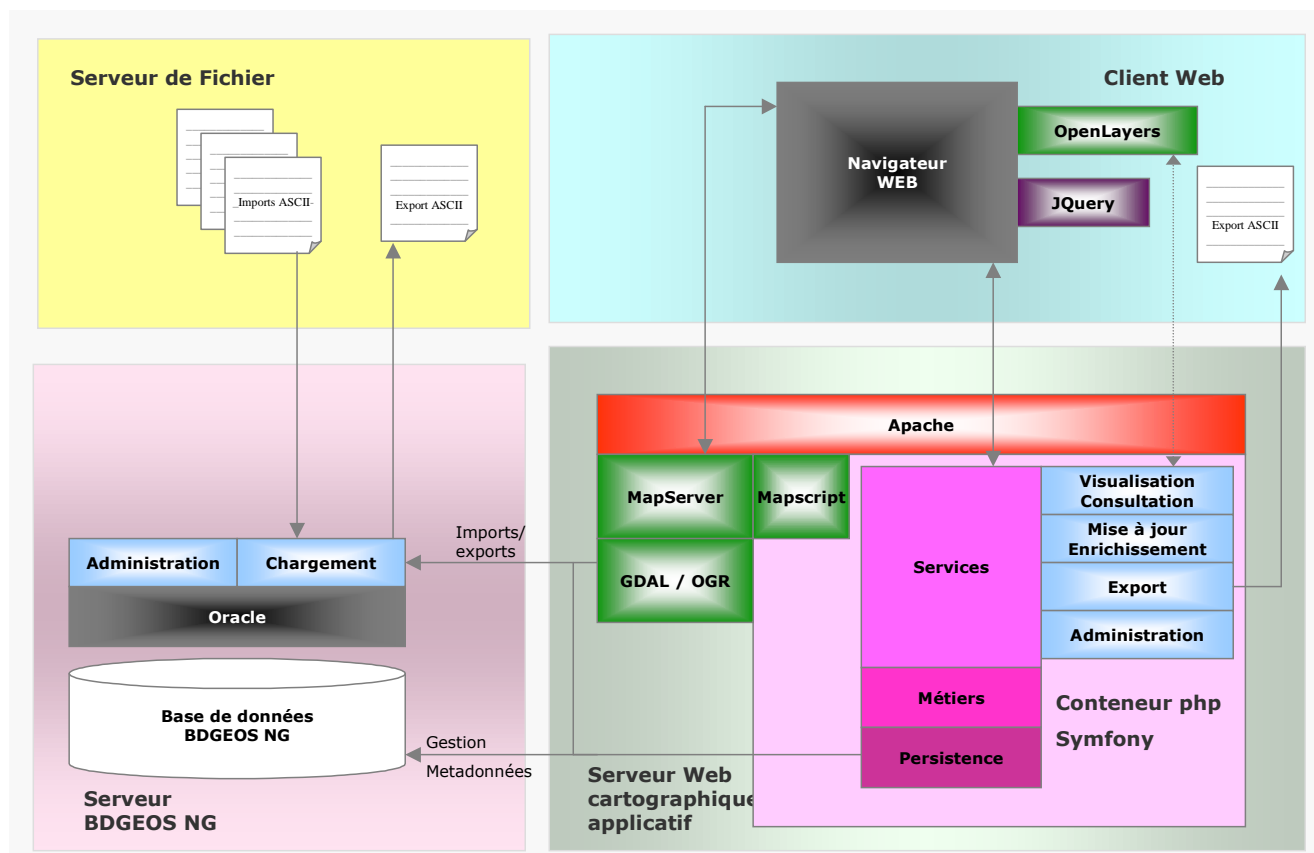


Figure 29 : Architecture Logicielle Générale

Les composants logiciels se répartissent sur deux de ces éléments d'architecture

- ✓ Un serveur Oracle : Il fournit nativement les capacités et fonctionnalités chargés
 - ↳ Du stockage
 - ↳ De la réplication entre les deux zones de déploiement non connectées.
 - ↳ Des imports et exports massifs.
- ✓ Un Composant applicatif (basé sur le framework symfony) chargé d'effectuer
 - ↳ La logique métier
 - ↳ Le lien avec la base de données
 - ↳ Les fonctionnalités interfacées avec l'opérateur (Visualisation, mise à jour ...)
- ✓ Un composant cartographique pour
 - ↳ La construction de l'information géographique nécessaire à la visualisation cartographique
 - ↳ La représentation cartographique.

5. Présentation détaillée des composants

5.1 Composant applicatif.

5.1.1 Cas d'utilisation

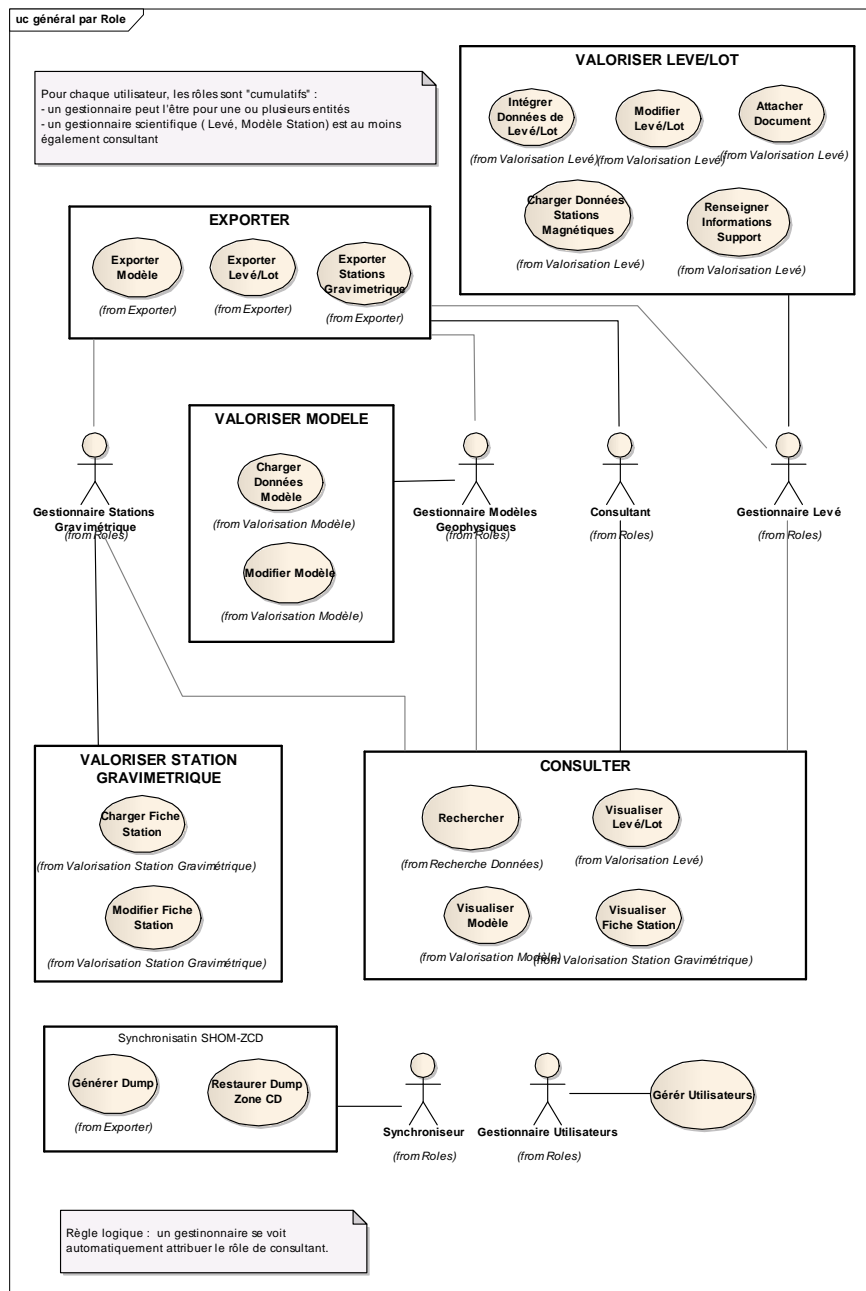


Figure 30 : Diagramme des cas d'utilisation généraux

5.1.2 le Framework Symfony

L'application est construite sur une architecture Apache/php/oracle.

La partie Apache/php étant la partie serveur de l'application, permettant d'effectuer

L'interface avec le serveur Oracle

Les processus métier

L'interface avec le client léger (IHM).

L'organisation du code php choisi est celle proposée par le framework Symfony. (<http://www.symfony-project.org/>).

Il constitue une plate-forme implémentant déjà les pratiques éprouvées du web :

Mise en place du design pattern MVC : Model Vue Controller

- ✓ La vue expose le modèle en page web disponibles pour les interactions avec l'utilisateur.
- ✓ Le contrôleur répond aux sollicitations de l'utilisateur, effectue les contrôles nécessaires et déclenche les changements de vue ou du modèle.
- ✓ Le modèle représente la logique métier. De plus, il est construit pour séparer modèle de classes et accès à la base.

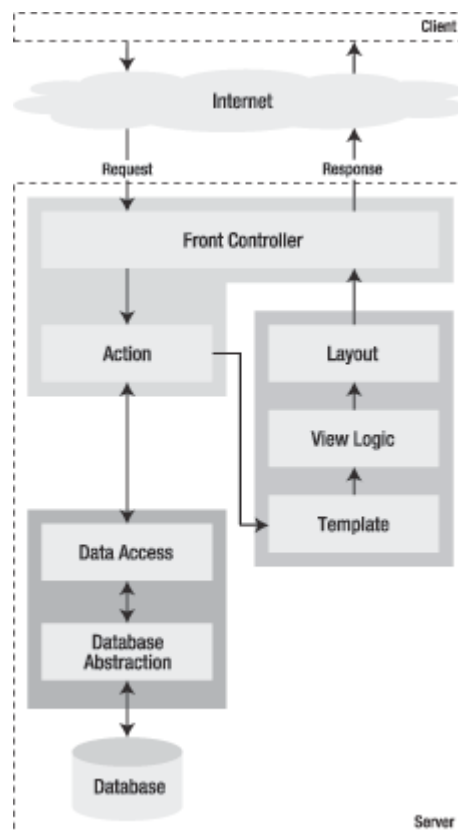


Figure 31 : Pattern MVC selon « symfony »

5.1.3 Modules Fonctionnels

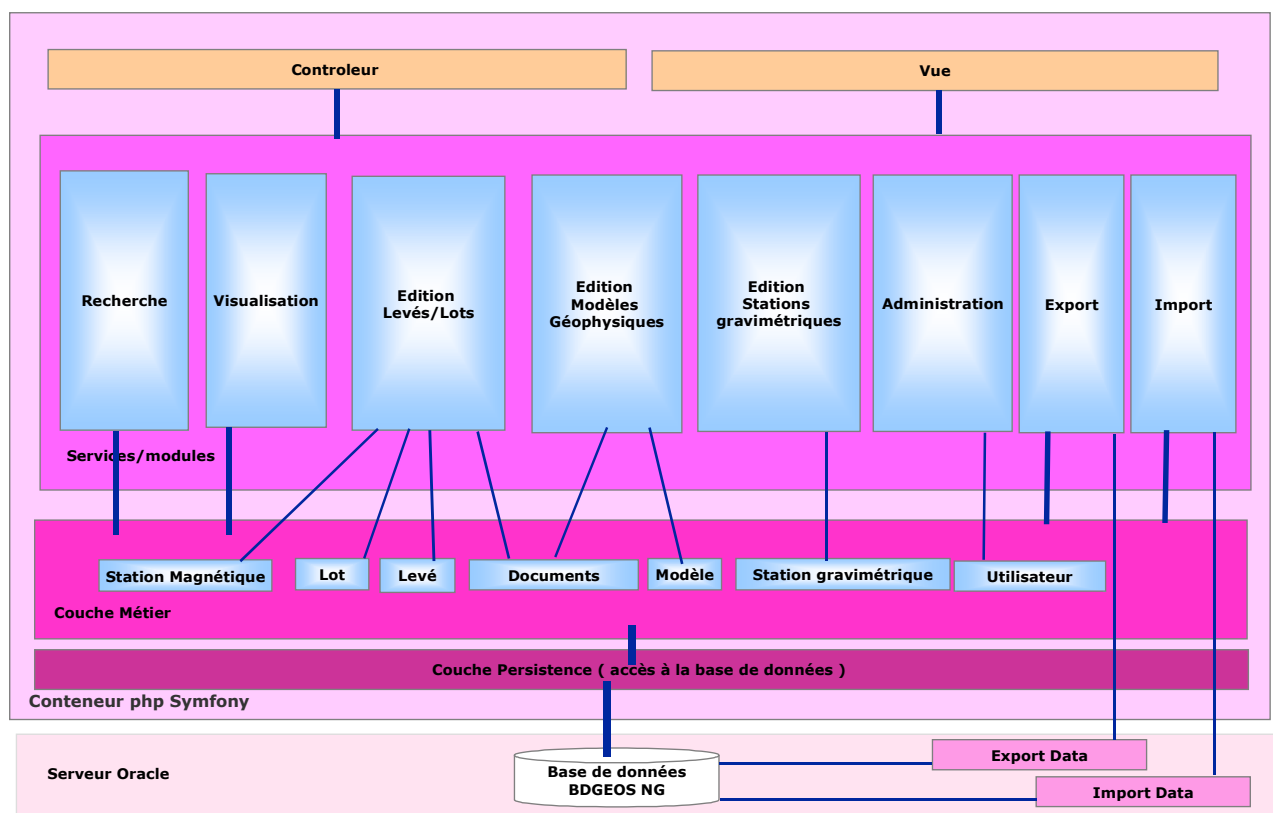


Figure 32 : Diagramme schématique des modules fonctionnels principaux

En s'appuyant sur le pattern MVC de Symfony, l'organisation métier s'articule donc autour d'une couche métier et d'une couche de services.

- ✓ La couche « métier » est constituée d'entités métier « images » des classes issues de la modélisation. Les principales sont donc
 - ↳ Levé, Lot, stations magnétiques, documents pour les levés
 - ↳ Modèles géophysiques et documents
 - ↳ Stations gravimétriques.
- ✓ La couche de persistance, se charge de l'accès à la base de données, et de persister les informations des classes métier.
- ✓ Une couche de services(modules) regroupant les grandes fonctionnalités.

Le principe de fonctionnement et de séparation des tâches est le suivant :

- ✓ L'accès à la base de données est faite par la couche de persistance (qui contient donc les ordres de cloture de transaction)
- ✓ La couche métier discute avec cette couche de persistance et expose aux services les objets métiers donc elle à besoin.

- ✓ Les services récupèrent les paramètres envoyés par le client web et envoient les résultats grâce à la vue et au contrôleur.

En principe les services n'accèdent qu'à une sélection d'objets métier, relatifs à leur domaines. Certains services transverses (qui seront affinés au cours du développement) accèdent à l'ensemble (recherche, visualisation, Import, Export ...)

Les services d'import et d'export accèdent à l'ensemble des objets métier, notamment pour sélectionner ou décrire les informations de templates de formats.

La partie « ImportData » ou « ExportData » est en partie déployée ou effectuée directement sur le serveur de base de données pour ce qui concerne la partie « chargement massif ». (cf § [5.3.4](#))

5.1.4 Détails des cas d'utilisation

La conception des diagrammes d'activité relatifs aux cas d'utilisation est dans le dossier de spécifications fonctionnelles.

5.2 Composant Cartographique

5.2.1 Introduction

N£ DS_carto_F_0120 £N

T£

Le composant cartographique permet de :

- ✓ Construire l'information géographique nécessaire à la visualisation cartographique : mises à part les couches de fond de carte, les données à visualiser font partie de la BDGEOS NG : stations de référence gravimétriques, lots et modèles. Toutes ces données comportent une composante géographique. Mais pour être performant, le système doit pouvoir accéder de façon optimale aux données. Pour cela, il est nécessaire de préparer les données en amont.
- ✓ Visualiser et manipuler les cartes nécessaires à la représentation cartographique sur le client Web.

£T

A£ Exigence 23#PC £A

A£ Exigence 20#C £A

5.2.2 Choix techniques sur le stockage des données

Les choix techniques adoptés pour le stockage des données géographiques sont les suivants :

- ✓ Les modèles géophysiques peuvent compter des milliards de points. Ils seront stockés dans des fichiers GeoTIFF sous forme d'image.
- ✓ Les données vectorielles permettant de représenter les Lots sous forme de lignes doivent être stockées dans un champ SDO_GEOMETRY dans la table Oracle.
- ✓ Concernant les stations gravimétriques, la volumétrie n'impose pas de création de champ SDO_GEOMETRY. Leur localisation est suffisante.

- ✓ Les données de fond de carte utilisées seront stockées sous forme de Shapefile.

5.2.3 Choix techniques sur les composants utilisés

Pour la préparation des données vectorielles, la manipulation des champs de type SDO_GEOMETRY se fera directement à partir de procédures stockées PL/SQL dans Oracle.

Pour la préparation des fichiers GeoTIFF permettant de représenter les modèles, un module basé sur la bibliothèque GDAL sera développé.

Pour la génération des cartes, c'est le serveur cartographique MapServer qui sera utilisé. Il permet notamment l'affichage des rasters GeoTIFF, des données vectorielles Oracle SDO_GEOM et des fichiers Shapefile.

Pour l'interface utilisateur dans le client Web, c'est la bibliothèque OpenLayers qui sera utilisée.

5.2.4 Préparation des données de modèles géophysiques

5.2.4.1 Présentation

Il s'agit de générer un fichier image GeoTIFF qui représente une variable d'un modèle.

Ce fichier image est généré au moment de l'import du modèle dans la base BDGEOS NG. A noter qu'un fichier ne peut représenter qu'une variable d'un modèle. Un modèle à N variables conduit donc à la génération de N fichiers.

L'intérêt de disposer d'un fichier GeoTIFF est multiple :

- ✓ il peut utiliser la compression,
- ✓ il peut être tuilé et pyramidé pour des performances d'accès optimales,
- ✓ il est directement utilisable par MapServer pour générer les cartes,
- ✓ c'est le format raster le plus performant pour être utilisé avec MapServer.

Ce module est écrit en C/C++ et repose sur la bibliothèque GDAL.

5.2.4.2 Caractéristiques des fichiers à produire

Format de fichier	GeoTIFF
Support BIGTIFF	Oui
Compression	Sans perte : Deflate ou LZW
Tuilage	Oui
Pyramidage	Oui
Nombre de bandes	1
Résolution maximale	1 pixel par minute d'arc terrestre
Taille maximale	21600 x 10800 pixels.

	Remarque : on est très loin de la limite maximale supportée par l'option BIGTIFF, ce qui laisse la possibilité technique de représenter des modèles de résolution beaucoup plus fine dans le futur.
Nombre d'octets par pixel	1 (256 valeurs possibles)
Palette de couleur	Palette de 256 couleurs dont une valeur utilisée pour les pixels non renseignés.
Système de coordonnées	EPSG:4326 (Coord. géographiques sur ellipsoïde WGS84)

5.2.4.3 Données d'entrée pour la génération

- ✓ Fichier ASCII contenant les données du modèle
- ✓ Format de lecture du fichier modèle
- ✓ Emprise géographique du modèle
- ✓ Variable à représenter
- ✓ Type de Palette à utiliser
- ✓ Résolution (taille du pixel)
- ✓ Nom du fichier GeoTIFF en sortie

5.2.4.4 Etapes de génération d'un fichier

1. Création d'une grille régulière à partir des données lues dans le fichier ASCII :
 - 1.1. Création d'une grille du type de la variable sur l'emprise du modèle
 - 1.2. Parcours du fichier ASCII du modèle. Pour chaque ligne
 - 1.2.1. Extraction de : latitude, longitude, variable
 - 1.2.2. Mise à jour des statistiques (Min, Max)
 - 1.2.3. Ecriture de la cellule de la grille. Dans le cas où un modèle est d'une résolution supérieure à la minute d'arc, conserver la dernière une seule des valeurs contenue dans ce carré est suffisant pour la visualisation.
2. Conversion de la grille en valeur de pixels en fonction du type de palette :
 - 2.1. Parcours de la grille, pour chaque cellule
 - 2.1.1. Calcul de la valeur de la cellule avec un codage sur un octet en fonction des statistiques et du type de palette
3. Conversion de la grille en fichier GeoTIFF avec GDAL et application de la palette
4. Pyramidage du fichier GeoTIFF avec GDAL

5.2.4.5 Données en sortie

- ✓ Fichier GeoTIFF
- ✓ Code de retour de la fonction

5.2.5 Préparation des données de Lots

La représentation vectorielle des Lots se fait sous forme de lignes. Ces lignes sont constituées par les points dont les coordonnées sont celles des données composant la suite qui compose un Lot. Ces points sont reliés de façon chronologique.

Il existe au final un objet ligne par Lot. Il est stocké dans le champ SDO_GEOM de la table Lot.

Ce module est une procédure PL/SQL exécutée après chaque import.

5.2.6 Préparation de la visualisation des stations gravimétriques

5.2.6.1 Stations de référence gravimétriques

Les stations de référence gravimétriques sont représentées sur la carte sous forme de points.

La volumétrie n'impose pas de création de champ SDO_GEOMETRY. La localisation est suffisante.

Une vue SQL dédiée à une utilisation purement cartographique sera utilisée pour représenter dynamiquement les stations sur la carte. La géométrie SDO_GEOM sera calculée à la volée dans la requête SQL de la vue à partir des données de localisation latitude / longitude.

Cette vue remontera également tous les champs discriminants pour la représentation : le réseau d'appartenance, la précision, ...

Toutes ces données se trouvent dans les tables STAGRA_DESC et STAGRA_FICHE.

Il n'y a donc pas d'étape spécifique de préparation des données pour les stations puisque la représentation cartographique se fera dynamiquement via la vue SQL.

5.2.6.2 Réseaux de stations gravimétriques

Les réseaux de stations gravimétriques sont représentées sur la carte sous forme de lignes mettant en relation les liens entre fiches

La volumétrie n'impose pas de création de champ SDO_GEOMETRY. La localisation est suffisante.

Une vue SQL dédiée à une utilisation purement cartographique sera utilisée pour représenter dynamiquement les liens entre les fiches stations sur la carte. La géométrie SDO_GEOM sera calculée à la volée dans la requête SQL de la vue à partir des données de localisation latitude / longitude à partir des stations mères / filles.

Toutes ces données se trouvent dans les tables STAGRA_DESC, STAGRA_FICHE et RATTACHEMENT_FICHES.

Il n'y a donc pas d'étape spécifique de préparation des données pour les stations puisque la représentation cartographique se fera dynamiquement via la vue SQL.

5.2.7 Visualisation et interaction avec les données géographiques

5.2.7.1 Présentation

La visualisation des cartes se fait par l'intégration d'OpenLayers. Cette bibliothèque offre les services de base qui permettent de prendre en charge :

- ✓ l'affichage de la carte dans le navigateur,
- ✓ les interactions utilisateur côté client (outils de navigation et de sélection),
- ✓ les appels au serveur cartographique.

Le serveur cartographique MapServer permet, après configuration, de :

- ✓ répondre aux requêtes OpenLayers du client Web,
- ✓ de générer des cartes sous forme d'images à partir de données d'origines variées :
 - ↳ fichiers raster GeoTIFF (comme les modèles géophysiques, par exemple),
 - ↳ données vectorielles issues de requêtes Oracle Spatial, (comme les Lots et les stations)
 - ↳ fichiers vectoriels de type Shapefile (traits de côtes, ...).

Toutes les interactions entre le client cartographique et le serveur se font via http.

5.2.7.2 Configuration du serveur

5.2.7.2.1 Configuration générale de la carte

La configuration générale de la carte consiste à paramétrer le service Web cartographique assuré par MapServer.

La configuration se fait en éditant un fichier Texte qui est communément appelé MapFile (ou fichier .map).

Ce fichier contient :

- ✓ des paramètres généraux :
 - ↳ titre,
 - ↳ emprise géographique,
 - ↳ formats d'image,
 - ↳ systèmes de coordonnées,
 - ↳ répertoire de publication,
 - ↳ légende,
 - ↳ ...
- ✓ une succession de description des couches composant la carte :
 - ↳ source des données,
 - ↳ classes de représentation (symbole, couleur, ...),
 - ↳ visibilité par défaut,

- ↗ plage d'échelle de visibilité,
- ↗ ...

Dans le cadre de BDGEOS, les couches suivantes seront référencées :

- ✓ Stations gravimétriques,
- ✓ Réseaux de stations gravimétriques,
- ✓ Lots de données magnétisme,
- ✓ Lots de données gravimétrie,
- ✓ Modèle géophysique,
- ✓ Traits de côtes.

5.2.7.2.2 Configuration particulière de la carte

Il est nécessaire de prévoir une configuration particulière de la carte. En effet, on n'affiche surtout pas toutes les données d'une couche en même temps sur la carte.

On n'affiche que les objets des couches répondant aux critères de sélection.

Pour cela, la simple configuration du fichier MapFile n'est pas suffisante. Il faut prendre en charge un moyen de filtrage.

Ce moyen de filtrage est véhiculé par des paramètres passés dans les URLs qui sont émises par le client vers le serveur cartographique.

5.2.7.3 Navigation cartographique

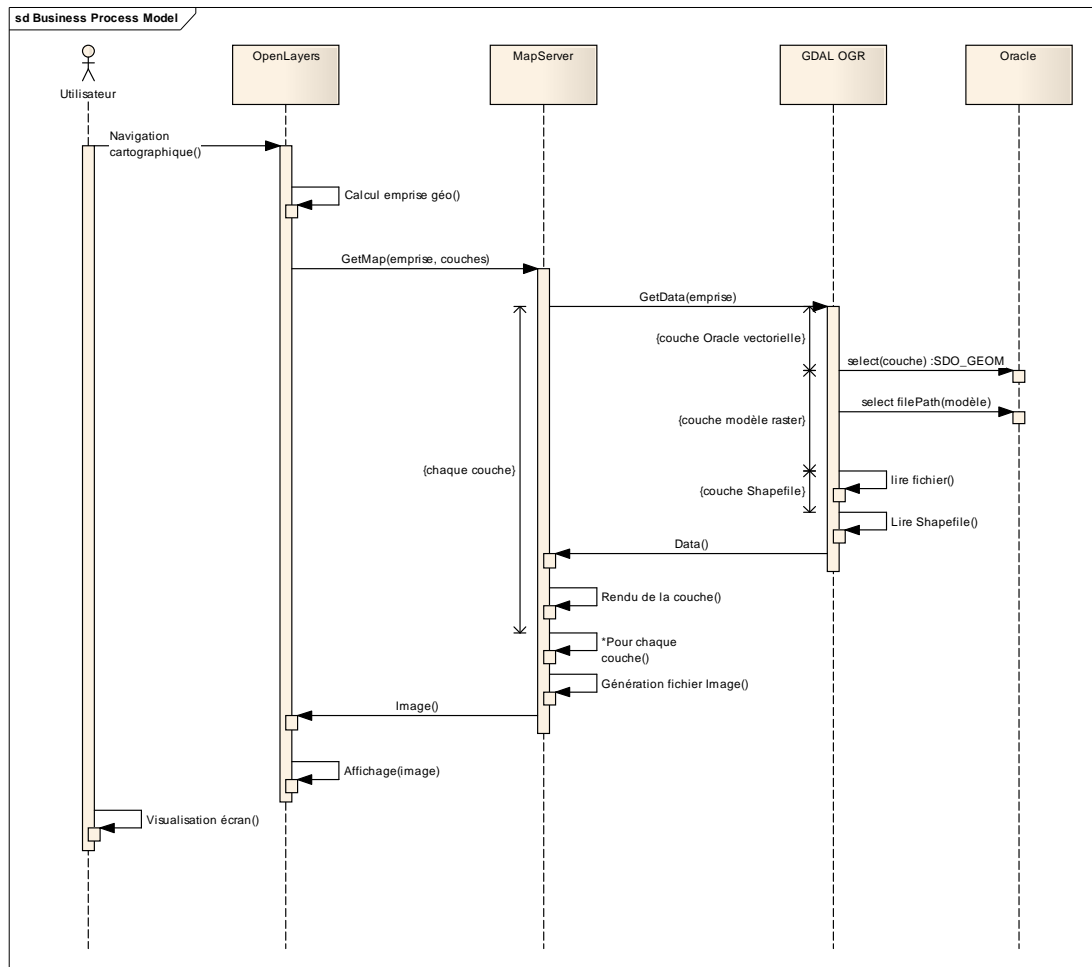


Figure 33 : Diagramme de séquence Navigation cartographique

Ce diagramme est simplifié car, pour plus de clarté, il ne mentionne pas dans le passage des paramètres les identifiant des objets répondant aux critères de filtrage (ex : identifiant du modèle, numéro de lot, ...).

5.3 Le Serveur de données Oracle.

5.3.1 Introduction

N£ DS_DataBase_F_0010 £N

T£

L'utilisation de la version Oracle 11g permet de mettre en place la gestion des nuages de points pour le traitement des modèles, les « secure files » afin d'optimiser la gestion et la possibilité de compression des « BLOBs » associés au nuages de points des données modèles.

L'utilisation des procédures stockées plsql , des « external tables », des techniques de « bulk collect » permet d'optimiser les processus de chargement et d'extractions des données de la base SHOM.

£T

A£ Exigence 50#PC £A

A£ Exigence 49#PC £A

De plus, les connections au travers une interface Web de type PhP sont facilitées par l'apparition d'un nouveau driver pour PhP.

5.3.2 Copie de bases de données

N£ DS_DataBase_F_0020 £N

T£

Ce chapitre Copie de base de donnée décrit les différentes étapes et mécanismes mis en place pour assurer la copie de bases de données

£T

5.3.2.1 Cadre général

On distingue deux bases de données non interconnectées :

- Une base source : La Base principale ou base publique,
- Une base cible : La Base dupliquée ou base privée, en zone confidentielle du SHOM

La base principale ou base publique (réseau du SHOM) est enrichie au fur et a mesure de l'arrivée de données.

N£ DS_DataBase_F_0030 £N

T£

La base dupliquée (réseau isolé) est une copie initiale de la base principale, mise a jour ponctuellement par export physique de la base principale.

Sur la Base publique toutes les créations et modification de données (insert, update) sont tracées par un trigger associé à chaque table. Ces modifications sont historiés dans la table historique des IU (Insert, Update) .

£T

N£ DS_DataBase_F_0040 £N

T£

A un instant Tcn donné, un utilisateur peut décider d'extraire toutes les données modifiées depuis le dernier chargement Tcn-1. Un processus Oracle DataPump va extraire l'ensemble des lignes des tables du schéma de la base publique qui ont subies des modifications entre Tcn-1 et Tcn.

Un fichier Oracle Datapump est généré dans un espace dédié.

£T

N£ DS_DataBase_F_0050 £N

T£

L'utilisateur a la responsabilité de transférer manuellement ce fichier pour l'utiliser en vue du chargement des données sur la Base Privée.

L'utilisateur dépose le fichier sur un espace dédié et lance un shell script lui permettant de recharger les données publique sur la base privée.

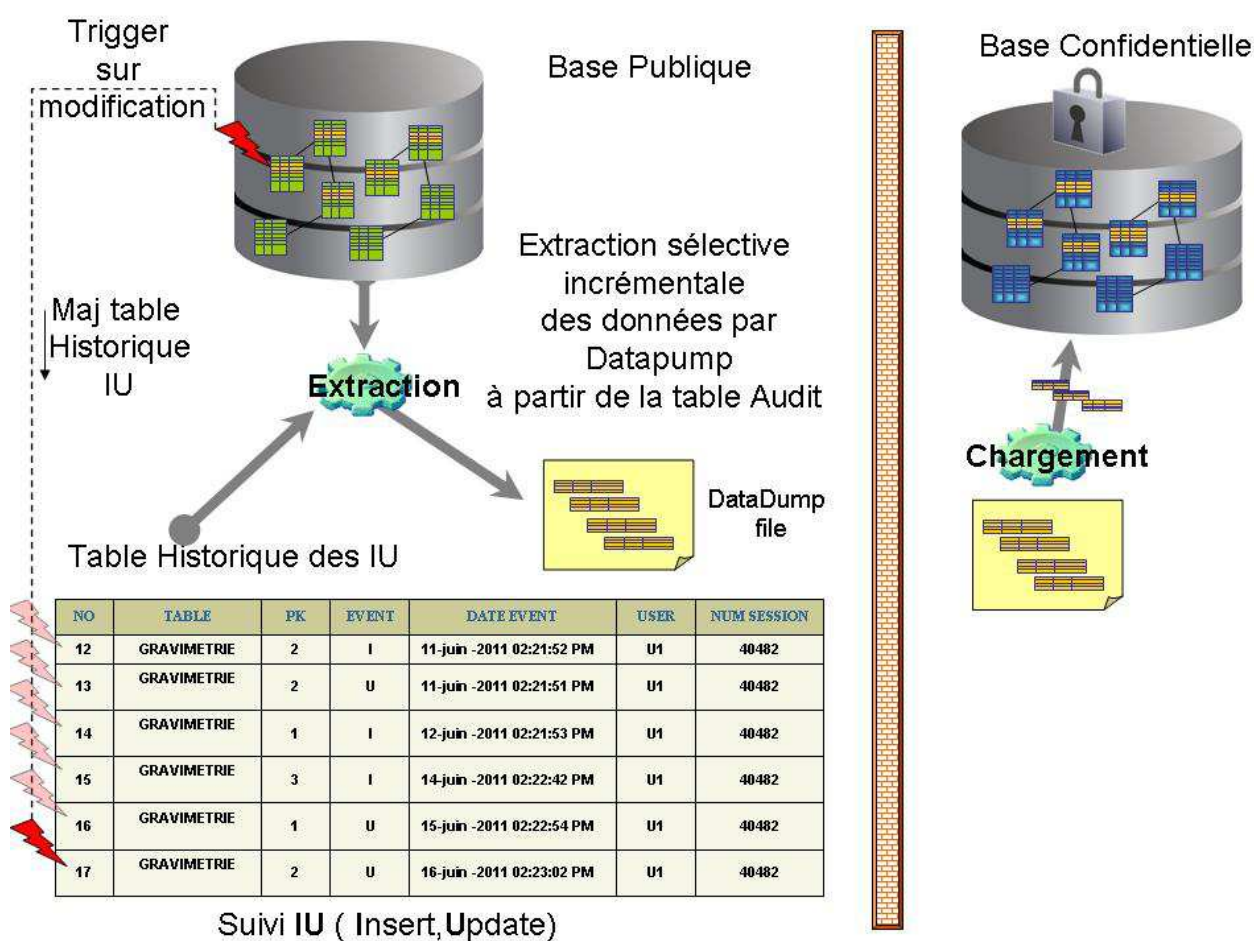
£T

N£ DS_DataBase_F_0060 £N

T£

Ces données sont consultables sur la base Privée, en cas de modifications sur celles ci, aucun contrôle n'est effectué au cas ou cette même donnée viendrait à être réimporter à partir d'un fichier venant de la base Publique. Dans ce cas les modifications effectuées sur la base privée sont écrasées.

£T



5.3.3 étapes du processus de replication offline

N£ DS_DataBase_F_0070 £N

T£

Les étapes du processus de réplication off line sont décrites dans ce chapitre 5.3.3

ET

- ✓ initialisation de la base de données

L'utilisateur va initialiser la base de données source et en faire une copie complète des données publique pour les importer sur la base cible dans l'espace privée.

- ✓ **génération code trigger**

Une procédure automatique de génération de trigger, séquence et table d'audit est mise en place pour suivre les modifications sur la Base de données Source

Lors de l'installation du schéma de la base source, le code plsql createSeqTriggerFortrackChange.sql génère le code de déploiement des triggers de suivi de modification sur les tables du schéma de la base source (liste des tables du schéma source : select table_name from cat where table_type='TABLE' and TABLE_NAME not like 'T_%' and TABLE_NAME not like 'MDRT%' and TABLE_NAME not like 'BIN%';). On obtient un code Auditspool.sql qui va créer l'ensemble des triggers.

- ✓ modification des données pendant un intervalle de temps

L'utilisateur va exploiter les données déjà présentes en base en effectuant des modifications qui vont être tracées dans la table des historiques de modifications alimentée par les triggers précédemment mis en place.

- ✓ Extraction des modifications

Au bout d'un certain moment, l'utilisateur décide d'extraire les données modifiées pendant la période d'exploitation pour alimenter la base cible non interconnecté.

une procédure ExportQueryTables écrite en plsql utilisant l'API oracle Datapump permet de faire un export sélectif des données modifiées : ExportQueryTables (SchemaName in varchar2, myDirectory in varchar2) .

L'export se base sur un filtre sur la clé primaire des tables modifiées, la table d'audit AUDIT_LOG contient les mise a jour par les triggers de suivi de modification. Après lancement, on trouve en sortie un fichier .dmp et un fichier log associé sur l'espace correspond au nom de la DIRECTORY oracle passée en paramètre à la procédure ExportQueryTables.

le fichier dmp contient uniquement les données modifiées durant la période couverte par le contenu de la table AUDIT_LOG.

- ✓ Transfert et chargement sur la base cible

Après transfert du fichier dump sur la cible, L'utilisateur exécute un shell script DATAPUMP qui recharge les données sur la base de données cible en mode append (TABLE_EXISTS_ACTION=APPEND).

Lors de l'exécution de ce processus les utilisateurs seront déconnectés afin d'éviter tout problème de lock.

Afin de prévenir des collisions dans la gestion des identifiants des données publiques et privés, un système de séquence intercalées pair, impair est mis en place sur les deux bases.

Base source - create sequence test_seq start with 1 increment by 10;
base cible - create sequence test_seq start with 2 increment by 10;

5.3.4 Chargement extraction des DONNEES CSV

N£ DS_DataBase_P_0080 £N

T£



La problématique des volumétries a été initialisée au cours du recueil de besoin et des différentes réunions. Les exigences maximales décrites dans le CCTP sont peu compatibles avec les possibilités techniques des systèmes informatiques actuels.

Indépendamment des formats, Le système devra intégrer des jeu de recette comprenant : :

- ✓ Un modèle global, résolution 1 minute d'arc, 8 valeurs
- ✓ Un modèle local 10°10°, résolution 5 secondes d'arc, 4 valeurs.
- ✓ Un lot/Levé de 1500000 points

£T

A£ Exigence 35#PC £A

A£ Exigence 36#PC £A

A£ CCTP_CR03_2.1#PC £A

5.3.4.1 Chargement et extraction des DONNEES CSV

N£ DS_DataBase_F_0090 £N

T£

Les étapes de chargement et extraction des données « CSV » sont les suivantes

£T

1. Initialisation et Définition des formats

L'ensemble de la définition des formats des fichiers CSV susceptibles d'être chargés dans les bases SHOM est stockée dans les tables de définition des formats. Les formats fixes sont initialisés dans la base. L'utilisateur peut définir et modifier la définition d'un format libre de fichier CSV.

2. Génération d'une « External table » par format de fichier

A partir de la définition d'un format CSV, un processus permet de générer la définition d'une table de mapping de ce format par l'intermédiaire de la création d'une external table oracle générée à partir de la définition des formats de fichiers des tables des formats (FORMAT, COMPOSITION_FORMAT, ...etc.).

3. Lecture du fichier CSV

Le mécanisme dit d'external table permet de mapper la structure du fichier CSV en entrée et de pouvoir accéder à ce fichier en lecture par des requêtes requête SQL via des directives sqlloader.

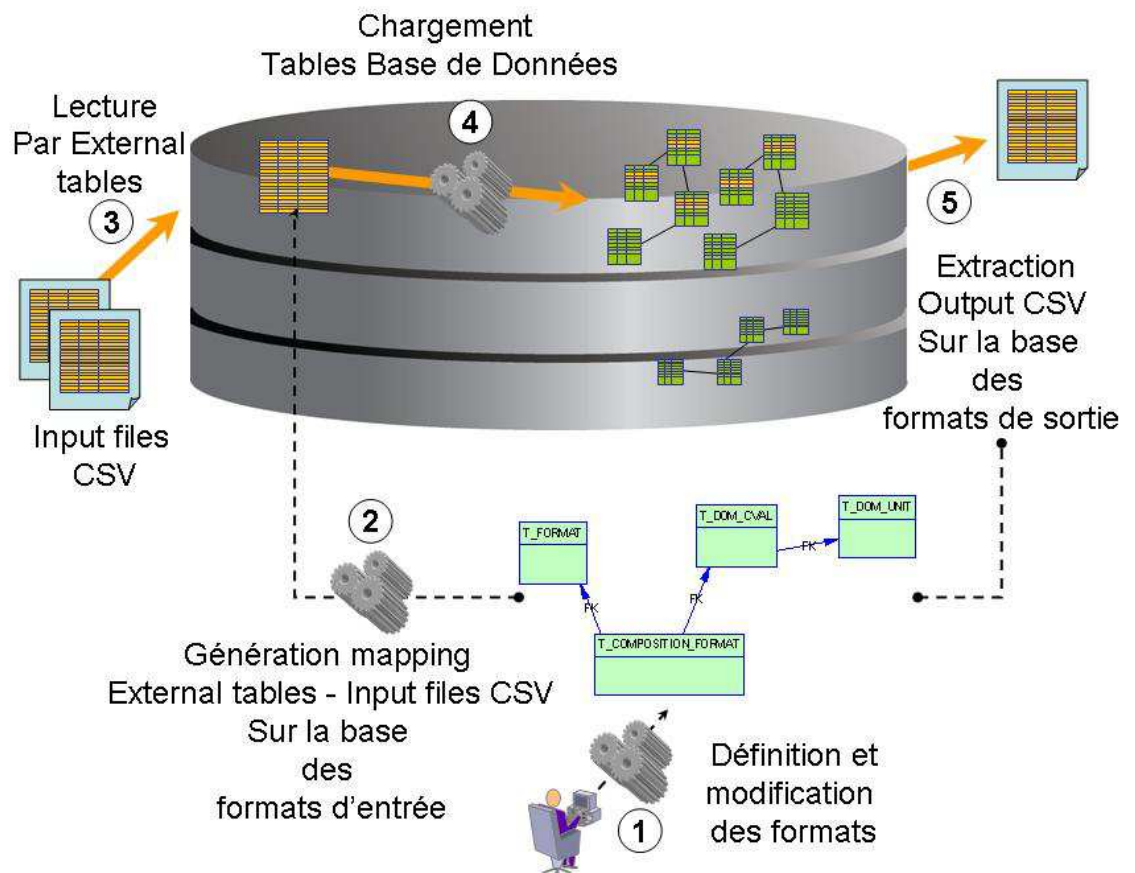
4. Chargement des données dans les tables de la base de données

Une fois que le processus de lecture est mise en place, les données lues dans le fichier CSV sont chargées dans la base de données. Des techniques de bufferisation et de bulk collect PLSQL sont mise en place pour optimiser le processus de chargement des données.

5. Extraction des données au format CSV

A partir des données stockées en base un fichier CSV est extrait correspondant à un type de format définies par l'utilisateur





Exemple de génération d'une création d'une external table à partir de la définition d'un format de type GRAVIMETRIE :

Begin

Execute immediate

```
(create table external_tab_GRAVIMETRIQUE(
'DATMES VARCHAR2(10),
'HEUMES NUMBER,
'LONMES NUMBER,
'LATMES NUMBER,
'ROUTE NUMBER,
'MESVIT NUMBER,
'PESABS NUMBER,
'ANOALL NUMBER,
'NUMPRO VARCHAR2(10)
)
organization external
(type oracle_loader
default directory external_file
access parameters (
records delimited by newline
BADFILE "bad_%a_%p.bad"
logfile "log_%a_%p.log"
discardfile "dis_%a_%p.dis"
fields terminated by "

```



```

        'missing field values are null' ||
        (
'DATMES,'||
'HEUMES,'||
'LONMES,'||
'LATMES,'||
'ROUTE,'||
'MESVIT,'||
'PESABS,'||
'ANOALL,'||
'NUMPRO' ||
        )
        'location reject ("demo1.dat") limit unlimited' ||
        ;
end;
/

```

5.3.4.2 Chargement et extraction des modèles CSV

N£ DS_DataBase_F_0100 £N

T£

Les étapes de chargement et extraction des modèles « CSV » sont les suivantes possède les caractéristiques et se déroule suivant les étapes décrites dans ce chapitre [5.3.4.2](#)

£T

Le traitement des fichiers modèles, de part leur volumétrie importante requiert l'utilisation de techniques de stockage particulière dite de « point cloud »

Le stockage des données des modèles s'appuie sur le nouveau type ORACLE « point cloud » ou nuage de point de la version oracle 11G, ce type SDO_PC est dédié au traitement des nuages de point comme les données SHOM de type MODELE.

On distingue différents étapes dans le traitement d'un modèle :

1. Chargement du fichier input dans une external table

Soit un fichier input de plusieurs millions de lignes de type données de modèle, constitué de n colonnes de valeur VAL_Dn.

Ce fichier est chargé dans une external table oracle , cette external table est constituée des champs suivant RID, VAL_D1, VAL_D2 ... VAL_Dn.

2. Initialisation et création d'une table « point Cloud »

A partir de la table input de type oracle external table, une table de nuage de point est créée et initialisée. Cette étape passe par l'utilisation des procédures Oracle Spatial 11G sdo_pc_pkg.create_pc et sdo_pc_pkg.init.

A l'issue de cette étape, une table T_MODELE de type nuage de point est créée, elle est associée avec une table de block de points qui va répartir les données en entrée en block de points stockées sous la forme de BLOB oracle(Binary Large Object).

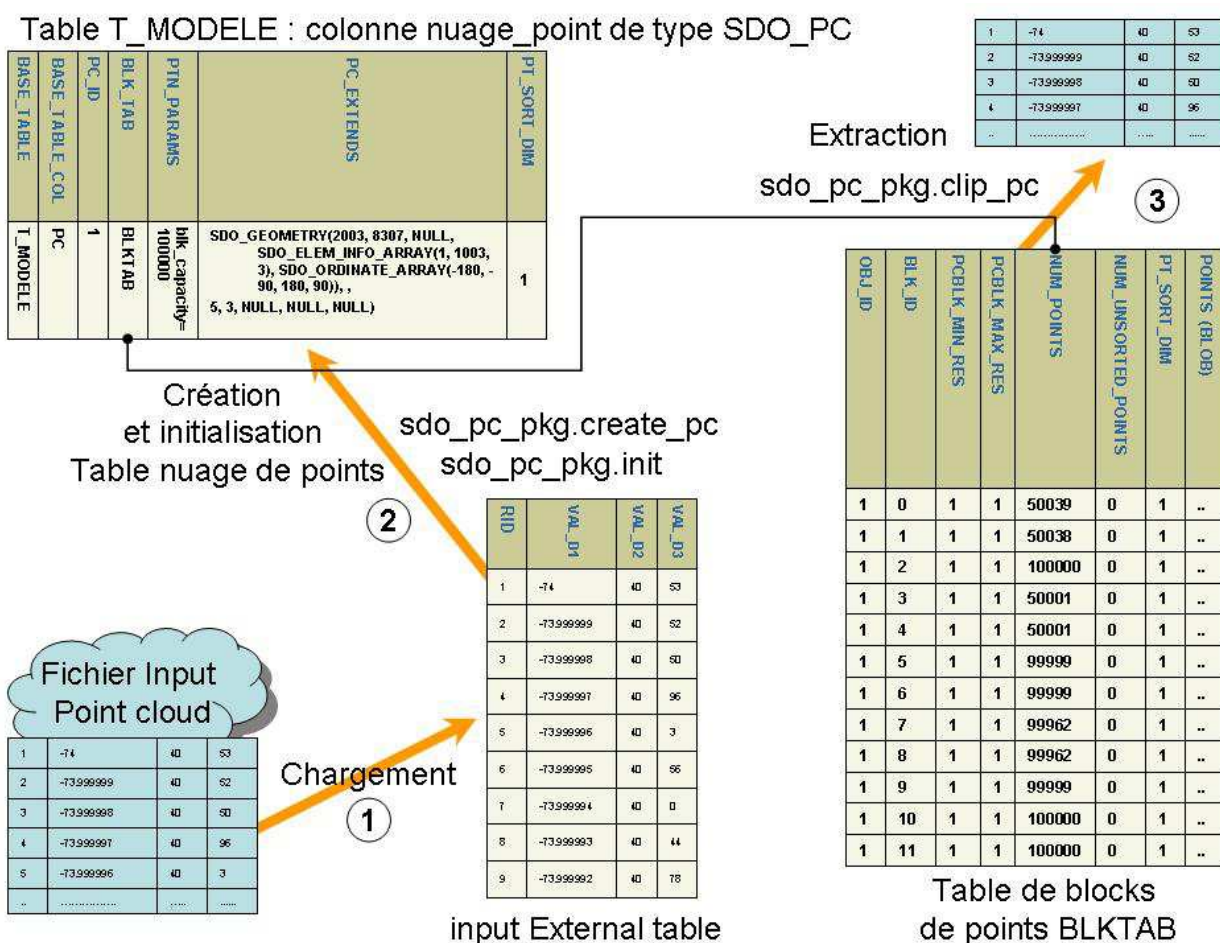
Le paramètre Blk_capacity va conditionner la répartition des données input en block de points. (colonne NUM_POINTS de la table BLKTAB = nombre de point par blocks) .

Dans la figure ci dessous, on a un exemple d'un fichier input d'un million de lignes qui vont se repartir dans la table BLKTAB sous la forme de BLOB. Cette figure permet de visualiser le contenu de la colonne NUAGE_POINT de type SDO_PC table T_MODELE (ID_MODELE number, NUAGE_POINT sdo_pc, ... autres colonnes ...). le type SDO_PC type est un type composé dont le champ PC_EXTENT représente l'empreinte de nuage de point.

La table T_MODELE pointe vers la table BLKTAB par l'intermédiaire du champ PC_ID et BLK_TAB.

3. Extraction des données « point cloud »

Oracle spatial fournit une procédure sdo_pc_pkg.clip_pc qui permet d'extraire un sous ensemble du nuage de point, on a la possibilité d'extraire une sous partie ou l'intégralité d'un modèle à partir d'une emprise géographique.



4. Utilisation de SecureFiles Oracle

La version Oracle 11G propose la notion de stockage SecureFiles applicable aux données de type LOB. Les SecureFiles Oracle permettent de compresser les données.

Il sera envisagé d'utiliser les SecureFiles au regard de test sur le temps de chargement, de test sur le gain d'espace potentiellement obtenu sur les données chargées. La compression étant fortement dépendante du type de données, la décision d'activation sera conditionnée par une série de tests représentatifs sur des données opérationnelles.

6. Matrice de couverture du besoin

Peu d'exigences détaillées numérotés sont « couvertes » dans ce document. Il s'agit plutôt de la conception du modèle de données. Il traduit les énoncés qui sont dans la partie générale de description des relations entre entités, schémas préliminaires de tables. Les tâches « d'inspiration et de modification du modèle » pour « améliorer l'adéquation de la base au besoin » ne sont pas décrites comme des exigences.

Une grande partie de la couverture des exigences du CCTP est décrite dans le Document de spécifications

La matrice de couverture des besoins fera l'objet d'un document à part

7. Liste des tables

Cf le document

[BDGEOS-DC-03-CS_ANNEXE1_TABLES.doc](#)

