# Convergence des modèles de métadonnées

# Architecture des bases de métadonnées

## Situation actuelle

E-forest

Pastis N années

N années

MetaIFN

PDA 1 année

N années

La base MetaIFN est la base de référence des métadonnées de l’inventaire.

La base Pastis (N années) contient une exportation des métadonnées de MetaIFN mise au format PASTIS c'est-à-dire optimisée pour le terrain.

La base Pastis (1 année) correspondant à une exportation des métadonnées issues de Pastis (N année) pour alimenter le PDA pour une année de collecte

La Base de métadonnées E-forest correspondant à une exportation des métadonnées de la base Pastis (N années) complétée par la description des formulaires.

## Situation cible:

E-forest

MetaIFN

PDA 1 année

N années

L’objectif visé est la réduction du nombre de base de métadonnées afin d’utiliser la même base pour la collecte des données terrain et la consultation.

A partir de la base MetaIFN, la base E-forest est alimentée pour servir de base de métadonnées de référence pour les applications de la chaîne de collecte et de consultation/diffusion des données collectées.

## Analyse des écarts

Informations présentes dans la base de métadonnées E-forest  et absentes de MetaIFN :

* La spécification du contenu des formulaires web (et demain formulaires PDA)
* La notion de dataset. A noter que la notion de lot de données de MetaIFN pourrait être utilisée.

Informations présentes dans la base de métadonnées MetaIFN  et absente de la base ER-forest:

* Une nomenclature qui évolue en fonction du temps et de l’espace

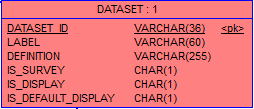
## Propositions d’évolution

Dans MetaIFN :

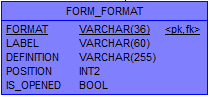
1. Intégrer un nouveau schéma afin de pouvoir intégrer des besoins applicatifs.
2. Dans ce schéma, la spécification des formulaires (web et PDA) pourrait alors être décrite.

Dans E-forest /OGAM (pour Web et PDA):

1. Introduire la notion de dataset de saisie (pour une campagne, une fraction, etc) et dataset de consultation (pouvant potentiellement regrouper plusieurs campagnes ou fractions) via l’ajout des champs « is\_survey » et « is\_display » dans la table dataset.

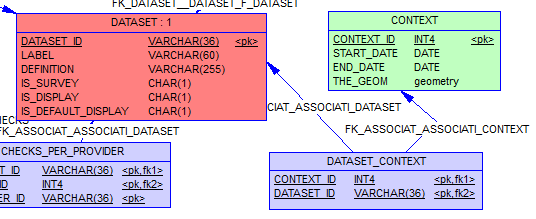


1. Utilisation de la notion de hiérarchie dans les formulaires via le champ « position » de la table FORM\_FORMAT (utile pour PDA si et ssi génération auto des formulaires).



1. Introduction de la notion de dataset temporels et spatiaux. Ceci est nécessaire pour agréger toutes les fractions de l’inventaire (sinon une base par fraction).

Création d’une table « Contexte » dans OGAM pour spécifier les attributs temporels et spatiaux des différents dataset.



1. La mise en oeuvre dans OGAM des changements d’unité en fonction du temps (stratégie MetaIFN) et en fonction de l’espace n’est pas retenue car cela implique trop de développements et une maintenance difficile, voir impossible (Exemple avec l’application Pommier).

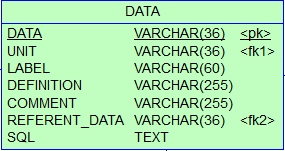
Cette notion peut être avantageusement reprise et simplifiée par la mise en place de changements de données en fonction du temps et de l’espace. Cela ne demande pas beaucoup de développement et respecte la logique actuelle d’OGAM.

Le principe est que si l’unité d’une donnée vient à changer, au lieu de d’affecter une nouvelle unité à la donnée déjà existante, la nouvelle unité est affectée à une nouvelle donnée en lien avec la précédente.

Prenons un exemple :

J’ai deux lots de données « ld06» et « ld07 ». L’un est de 2006 avec 3 codes pour la donnée « ecu06 » qui sont « brun, rouge, autre » et l’autre est de 2007 avec 4 codes pour la donnée « ecu07 » qui sont « brun, rouge, noir, autre ». Techniquement chaque donnée a une colonne correspondante en base. On retrouvera donc en base une colonne « ecu06 » suivi d’une colonne « ecu07 ».

Je souhaite visualiser les écureuils sur 2006 et 2007. Pour cela je crée un lot de données « ld0607 ». J’y ajoute les données qui m’intéressent dont la nouvelle donnée « ecu0607 » et je crée en base la colonne « ecu0607 » au même niveau que les deux autres colonnes. La donnée « ecu0607 » aura 3 codes qui sont « brun, rouge, autre ». La colonne en base est remplie par un simple script sql archivé dans le champ « sql » de la table « data ». La colonne « referent\_data » de la table « data » aura pour valeur « ecu » qui est la donnée d’origine.



Cette approche n’impacte donc aucunement le fonctionnement actuel d’OGAM et peut donc être ajoutée n’importe quand.

1. Notion d’unité de saisie et d’unité de stockage
2. Notion de filtrage des modalités en fonction d’un contexte (ou dataset de saisie)
3. Notion de groupe de champs (fieldset) et d’écran dans un formulaire pour la saisie sur pda.
4. Notion de zonage (usite/site) en complément de la notion de géométrie.
5. Notion de données de contexte pour l’édition/la sauvegarde des données d’un formulaire.

## Règles pour la prise en compte des modifications dans le cadre du projet SONGE

MetaIFN ne doit pas être modifiée (ou perturbée) par des implémentations applicatives.

Il n’est pas envisagé d’introduire des modifications de structure de la base de données E-forest qui nécessiteraient des transformations importantes dans l’application OGAM

Il est nécessaire de faire évoluer le méta modèle pour que le framework QT s’appuie sur le méta-modèle cible si cela ne remet pas en cause les charges estimées dans le projet SONGE initial.