



Le schéma GML

Gabarit physique v1.3.9.0 du standard de métadonnées SINP v1.3.9

Auteurs : Rémy Jomier (MNHN/SPN)

1	Introduction.....	2
2	Structure générale.....	2
2.1	Découpage des fichiers.....	2
2.2	Flexibilité de la structure du GML	2
3	Contenu du fichier.....	3
3.1	Encodage des fichiers	3
3.2	Balises	3
3.3	Contenu des balises.....	3
3.4	Vocabulaire contrôlé	4
3.5	Annotation.....	5
4	Le Modèle Logique de Données du standard (MLD).....	5
4.1	Gestion des attributs facultatifs	5
4.2	Gestion des attributs obligatoires conditionnels	5
4.3	Changements par rapport au standard	5
5	Les modèles du Standard	5
5.1.1	Modèle Logique de Données.....	5
5.1.2	Modèle Physique de Données (MPD).....	9

1 Introduction

L'objectif de ce document est de présenter l'implémentation du dictionnaire de données en Geography Markup Language (GML), le format choisi par le GT Standard de données (cf CR du 14 octobre 2013). Les fichiers GML, comme les XML, sont créés à partir d'un schéma de référence en XSD, appelé dans ce document « schéma GML ». Les classes et les attributs sont matérialisés par des balises.

L'utilisation du GML 3.2.1 permet de respecter les normes ISO (norme ISO 19136 publiée en 2007) et INSPIRE (format préconisé par INSPIRE).

2 Structure générale

2.1 Découpage des fichiers

Un fichier GML échangera zéro à plusieurs observations représentant une partie ou la totalité d'une fiche de métadonnées, ayant ou non un métacadre. Ce découpage pourra être optimisé selon les performances des plateformes dans l'échange de métadonnées.

Les échanges sont cadrés de cette façon : un système d'échange pour les métadonnées de cadres d'acquisition, un autre pour les métadonnées de jeux de données.

Dans les fichiers d'échange, la balise englobante du jeu de données est « FeatureCollection » et la balise de chaque cadre d'acquisition ou jeu de données est « featureMember ». Elles n'apparaissent pas dans le schéma GML (xsd) mais doivent être ajoutées dans les fichiers GML (voir les fichiers d'exemple).

2.2 Flexibilité de la structure du GML

La structure du gabarit GML peut être plus ou moins verrouillée :

- dans la présence de toutes les balises, même si elles sont vides. En effet, dans un GML/XML, un champ vide peut se concrétiser par une balise vide ou par une absence de balise. Ainsi, si a, b, c sont des balises d'un GML et que la balise b est vide, car l'attribut est facultatif par exemple, alors la présence de toutes les balises n'est pas obligatoire : la structure du fichier est pour autant toujours conforme.

Fichier avec les balises a, b, c :	b est facultatif et non renseigné :	ou
<code><a> </code>	<code><a>xxx </code>	<code><a>xxx</code>
<code> </code>	<code></code>	<code><c>zzz</c></code>
<code><c> </c></code>	<code><c>zzz</c></code>	(C'est la solution retenue ici, pour limiter le volume des fichiers autant que possible)

- dans l'ordre des balises. L'ordre des balises n'est pas fixé dans le XML. Par exemple : les balises a, b, c peuvent se présenter en b, a, c ou c, a, b etc ; pour autant, la structure du fichier est toujours conforme. Il a été fait le choix de verrouiller ces aspects de la structure pour faciliter la validation de la conformité des fichiers et leur utilisation. En effet, plus la structure est fixée, et plus la validation du fichier peut se faire avec des parseurs simples et plus la récupération et l'interrogation des données sont facilitées.

Si un concept n'est pas utilisé, il n'a pas à être présent (si par exemple on n'utilise pas les attributs additionnels, aucune des balises des attributs additionnels n'est nécessaire).

Dans le schéma GML, cela est représenté par la balise <xs:sequence>.

Remarque :

- Les contrôles de conformité des fichiers au gabarit peuvent être faits par les plateformes.

3 Contenu du fichier

3.1 Encodage des fichiers

Les fichiers GML seront encodés en UTF-8 sans BOM de manière à permettre aux outils de les lire de façon correcte. Chaque fichier sera en début de fichier précédé des balises suivantes : **<?xml version="1.0" encoding="UTF-8"?>**

3.2 Balises

Les balises correspondent aux classes et aux attributs du dictionnaire de données.

Des choix d'implémentation ont été faits pour implémenter un Modèle Logique de Données, qui lui-même est traduit en Modèle Physique de données (le schéma GML).

Les choix d'implémentation sont présentés au chapitre 4.

Les définitions de chaque élément (classe, attribut, énumération, CodeList) sont ajoutées dans le schéma GML.

Pour l'attribut géographique, le GML rend obligatoire l'échange d'un identifiant unique de l'objet. Cet identifiant n'est pas défini, mais a certaines contraintes : il doit être unique, et commencer par une lettre, sont également permis les lettres, chiffres, et le caractère « _ ». Il conviendra aux plateformes R/T de diffuser le leur s'il existe ou d'en générer un s'il n'existe pas.

De manière à garder le même chemin d'accès aux cadres d'acquisition et jeux de données, les cadres d'acquisition et les jeux de données sont traités séparément, sans imbrication, au sein de deux fichiers différents.

Pour conserver un traitement homogène, chaque cadre d'acquisition pourra porter l'identifiant permanent de son cadre d'acquisition parent (ou métacadre).

3.3 Contenu des balises

Afin d'éviter des problèmes majeurs dans le traitement des fichiers XML, les données de type chaîne de caractères devront **systématiquement** être écrites comme suit :

<![CDATA[Chaîne de caractères]]>

3.4 Vocabulaire contrôlé

Le vocabulaire contrôlé représente les valeurs de référence à utiliser pour renseigner un champ. Il s'agit d'une liste de valeurs interne comme le vocabulaire contrôlé de territoire ou typeFinancement. Les listes fermées de valeurs sont des énumérations, représentées par des *Enumeration* en langage UML.

Ces différentes énumérations peuvent faire l'objet de mises à jour plus ou moins importantes.

Il est fait référence aux vocabulaires contrôlés dans le fichier GML, ce qui permet potentiellement de vérifier que la valeur transmise dans le champ est une valeur autorisée par l'énumération. Elle permet en outre d'accéder aux valeurs sans avoir recours au dictionnaire de données PDF.

Concrètement, cela peut se traduire de deux façons :

- Soit la nomenclature est inscrite directement dans le schéma. La nomenclature est directement consultable mais la mise à jour de la nomenclature implique de créer une nouvelle version du schéma GML pour prendre en compte les modifications.

- Soit la nomenclature est appelée via une URL : le fichier fait le lien avec le référentiel disponible dans une forme interrogeable (en XML par exemple).

Avec cette solution, la prise en compte des mises à jour des listes de vocabulaire se fait automatiquement. Cependant, elle est plus lourde à mettre en place et à utiliser :

- la ressource doit être disponible sur la plateforme nationale sous une forme exploitable pour le parseur, ce qui n'est actuellement pas le cas.

- il est alors nécessaire d'utiliser un parseur spécifique pour valider la donnée ou de développer du code pour récupérer la liste de valeurs et la comparer au fichier.

Une solution plus légère possible : l'URL peut aussi simplement diriger vers une page informative sur le référentiel à utiliser.

En prenant en compte les particularités des référentiels, et de l'opérationnalité de la mise en œuvre, il est proposé pour cette première version d'intégrer les référentiels à faible volumétrie et faible mise à jour directement dans le gabarit. En attendant que les référentiels plus complexes soient en format interrogeables par URL, les liens ne seront pas intégrés ou dirigeront vers une page informative sur les référentiels à utiliser (à voir avec la plateforme nationale).

Dans un premier temps, les liens URL ne dirigeront pas vers les référentiels sous une forme lisible machine. A terme, il est prévu de rendre les référentiels disponibles dans un format callable par URL de la plateforme nationale afin de faciliter la vérification de conformité des fichiers. Les nomenclatures plus simples disparaîtront ensuite du schéma pour être mises à disposition dans un format accessible pour les machines. Les énumérations disparaîtront au profit des CodeLists.

La validation de conformité du fichier peut être effectuée au niveau des plateformes régionales et thématiques et en complément au niveau de la plateforme nationale.

Remarques :

- Dans le cadre du Système d'Information sur l'Eau (SIE), le Sandre a fait le choix d'inclure dans les XSD les vocabulaires contrôlés à faible volumétrie.

- Les nomenclatures dans le schéma GML seront de plus, disponibles en CSV pour intégration dans les bases de données.

- Deux modifications mineures ont vu le jour : un ajout dans la nomenclature des stades de vie a été réalisé, la valeur correspondant au statut de post-larve, de même pour la nomenclature des statuts d'observation, la valeur « Ne sait pas » a été ajoutée.

3.5 Annotation

Une balise de documentation peut être insérée en annotation pour présenter les caractéristiques du schéma. Il est proposé de rajouter une balise d'annotation présentant des informations sur la création et le sujet du schéma.

Ces informations sont en en-tête du fichier de schéma XSD mais elles n'apparaissent pas dans les fichiers de données XML et GML. Ci-dessous la balise proposée :

```
<xs:annotation>
  <xs:documentation source="nom">Métadonnées des cadres d'acquisition</xs:documentation>
  <xs:documentation source="versionDictionnaire">1.3.9</xs:documentation>
  <xs:documentation source="versionSchemaXSD">1.3.9.0</xs:documentation>
  <xs:documentation source="auteurs">SINP</xs:documentation>
  <xs:documentation source="statutDoc">Validé</xs:documentation>
  <xs:documentation source="description">Le but du standard métadonnées "Cadres d'acquisition" est de permettre
  l'échange d'informations sur les métadonnées de cadres d'acquisition entre les acteurs du SINP.</xs:documentation>
</xs:annotation>
```

4 Le Modèle Logique de Données du standard (MLD)

Le passage du MLD au MPD se fait sans option d'implémentation mais selon les règles des formats techniques, ici selon les règles du standard GML notamment.

4.1 Gestion des attributs facultatifs

Les attributs facultatifs peuvent être absents du XML envoyé.

4.2 Gestion des attributs obligatoires conditionnels

Afin de gérer au mieux les attributs obligatoires conditionnels dépendant d'autres attributs, les balises englobantes suivantes sont créées :

- ProtocoleType
- ActeurType

Les autres attributs obligatoires conditionnels ne sont pas facultatifs, il faut se référer aux règles pour savoir quand ils sont potentiellement non renseignés.

4.3 Changements par rapport au standard

Non applicable : la relation entre un cadre et un métacadre est déjà prévue au niveau du standard.

5 Les modèles du Standard

5.1.1 Modèle Logique de Données

Le MLD produit à partir d'Enterprise Architect est disponible en format propriétaire EA. Notez bien l'absence de trait entre le sujet d'observation et le regroupement, et l'absence de trait récursif sur le concept de regroupement, par rapport au modèle logique du standard. Ces différences sont liées aux choix d'implémentation, mais n'impactent pas les informations présentes ou les relations prévues. Les modèles sont présentés à partir de la page suivante.

Name: Articulation des métadonnées des cadres d'acquisition / jeux de données
Package: «applicationSchema» Métadonnées
Version: 1.3.9
Author: rjomier



Name: Cadres d'acquisition
 Package: «applicationSchema» Métadonnées
 Version: 1.3.9
 Author: rjomier

«invariant»
 (Doit être utilisé 1 à n fois si "publications" est utilisé dans le concept CadreAcquisition)

«dataType»
Concepts::ReferenceTemporelle

dateLancement: Date
 + dateCloture: Date [0..1]

«dataType»
Concepts::Publication

+ URLPublication: CharacterString [0..1]
 # referencePublication: CharacterString

«featureType»
Concepts::CadreAcquisition

libelle: CharacterString
 # voletSINP: VoletSINPValue [0..*]
 # objectif: ObjectifsValue [1..*]
 # description: CharacterString
 # identifiantCadre: CharacterString
 + niveauTerritorial: NiveauTerritorialValue [0..1]
 # territoire: TerritoireValue [1..*]
 + precisionGeographique: CharacterString [0..1]
 + motCle: CharacterString [0..*]
 + typeFinancement: TypeFinancementValue [0..1]
 # acteurPrincipal: ActeurType
 + acteurAutre: ActeurType [0..*]
 # dateCreationMtd: Date
 + dateMiseAJourMtd: Date [0..1]
 + descriptionCible: CharacterString [0..1]
 + cibleEcologiqueOuGeologique: CharacterString [0..1]
 + idMetaCadreParent: CharacterString [0..1]
 # estMetaCadre: Boolean
 + referenceBiblio: Publication [0..*]
 + idTPS: CharacterString [0..1]

NOTE : Les attributs sans cardinalité sont tous obligatoires et de cardinalité 1.
 Pour les repérer plus facilement, tous les attributs obligatoires sont précédés du caractère "#". Les attributs non obligatoires sont précédés du caractère "+".

Extensions::AttributsAdditionnels

nomAttribut: CharacterString
 # definitionAttribut: CharacterString
 # valeurAttribut: CharacterString

«dataType»
Types::ActeurType

+ organisme: CharacterString [0..1]
 + idOrganisme: CharacterString [0..1]
 # roleActeur: RoleActeurValue
 + nomPrenom: CharacterString [0..1]
 + mail: CharacterString [0..1]

Peut être utilisé 1 à n fois pour contactDispositifAutre

«enumeration»
Nomenclatures::NiveauTerritorialValue

1
2
3
4
5
6
7

«enumeration»
Nomenclatures::TerritoireValue

METROP
GUF
MTQ
GLP
MAF
BLM
SPM
MYT
REU
NCL
WLF
PYF
CLI
EPA
SUBANT
TADL
HORSFR

«enumeration»
Nomenclatures::RoleActeurValue

1
2
3
4
5
6
7

«enumeration»
Nomenclatures::ObjectifsValue

1
2
3
4
5
6
7

«enumeration»
Nomenclatures::TypeFinancementValue

1
2
3
4

«enumeration»
Nomenclatures::VoletSINPValue

1
2
3

Name: Jeux de données
 Package: «applicationSchema» Métadonnées
 Version: 1.3.9
 Author: rjomier

Note : tous les attributs pour lesquels une cardinalité n'est pas mentionnée (il n'y a pas de [0..1] ou [0..*] ou [1..*] après le nom de l'attribut) sont OBLIGATOIRES.

 Ils sont tous précédés du caractère "#"

«enumeration»
 Nomenclatures::
 RoleActeurValue

1
2
3
4
5
6
7

«enumeration»
 Nomenclatures::
 TerritoireValue

METROP
GUF
MTQ
GLP
MAF
BLM
SPM
MYT
REU
NCL
WLF
PYF
CLI
EPA
SUBANT
TADL
HORSFR

«enumeration»
 Nomenclatures::
 MethodeRecueilValue

1
2
3
4
5
6
7
8
9
10
11
12

«enumeration»
 Nomenclatures::
 ObjectifJddValue

1.1
1.2
1.3
1.4
1.5
2.1
2.2
2.3
2.4
2.5
3.1
3.2
3.3
3.4
3.5
3.6
3.7
4.1
4.2
5.1
5.2
5.3
5.4
5.5
6.1
6.2
6.3
6.4
7.1
7.2

«dataType»
 Concepts::
 ConformiteCoherence

dateControle: Date
protocole: ProtocoleType
organismeControle: CharacterString
typeControle: TypeControleValue

«dataType»
 Concepts::
 NonConformeNonCoherent

nomAttribut: CharacterString
valeurInitiale: CharacterString

Extensions::
 AttributsAdditionnels

nomAttribut: CharacterString
definitionAttribut: CharacterString
valeurAttribut: CharacterString

«dataType»
 Types::
 ProtocoleType

typeProtocole: TypeProtocoleValue
+ uRL: CharacterString [0..1]
+ description: CharacterString [0..1]
libelle: CharacterString

«featureType»
 Concepts::
 JeuDeDonnees

libelle: CharacterString
libelleCourt: CharacterString
description: CharacterString
dateCreation: Date
+ dateRevision: Date [0..1]
typeDonnees: TypeDonneesValue
identifiantJdd: CharacterString
pointContactJdd: ActeurType [1..*]
+ motCle: CharacterString [0..*]
territoire: TerritoireValue [1..*]
domaineMarin: Boolean
domaineTerrestre: Boolean
identifiantCadre: CharacterString
objectifJdd: ObjectifJddValue
+ empriseGeographique: BoundingBox [0..1]
+ methodeRecueil: MethodeRecueilValue [0..*]
+ protocoles: ProtocoleType [0..*]
pointContactPF: ActeurType

«dataType»
 Concepts::
 BoundingBox

0..1 # borneOuest: CharacterString
borneEst: CharacterString
borneSud: CharacterString
borneNord: CharacterString

«enumeration»
 Nomenclatures::
 TypeDonneesValue

1
2
3
4
5

«enumeration»
 Nomenclatures::
 TypeProtocoleValue

0
1
2
3
4

«enumeration»
 Nomenclatures::
 TypeControleValue

1
2

«dataType»
 Concepts::
 BaseProduction

contactBaseProduction: ActeurType [0..*]
baseProduction: CharacterString

«dataType»
 Types::
 ActeurType

+ organisme: CharacterString [0..1]
+ idOrganisme: CharacterString [0..1]
roleActeur: RoleActeurValue
+ nomPrenom: CharacterString [0..1]
+ mail: CharacterString [0..1]

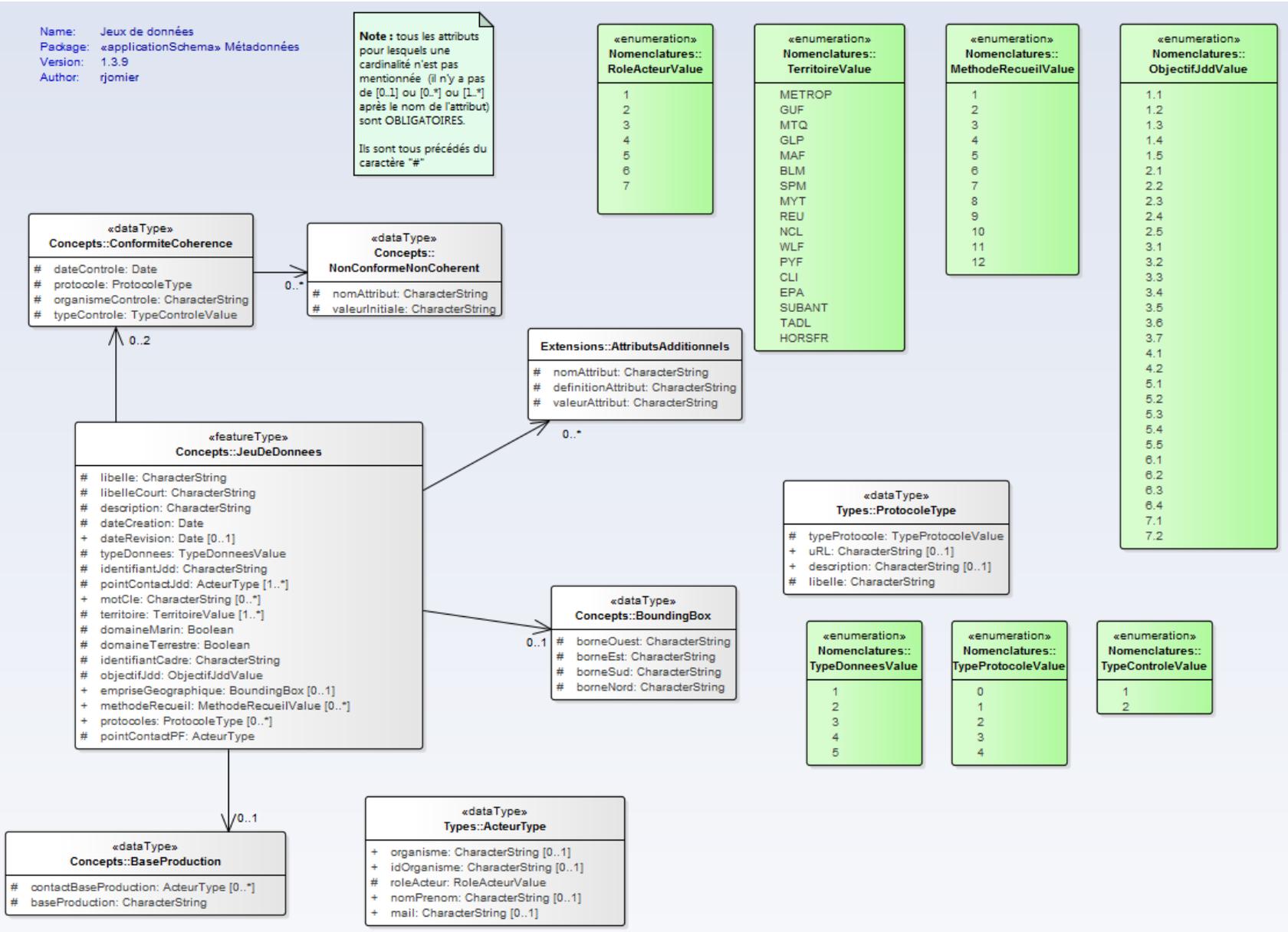
0..2

0..*

0..*

0..1

0..1



5.1.2 Modèle Physique de Données (MPD)

Le MPD est disponible en schéma GML (mtd_CA_1_3_9.xsd et Mtd_jdd_1_3_9.xsd).