

Modèle de données 2001 de la mensuration officielle vaudoise (MD.01-MO-VD)

Version 1.2.6 du 23.09.2004

Version conforme à la version 24 du modèle fédéral et formellement approuvée par la Direction fédérale des mensurations cadastrales.

D'autres informations sur le MD.01-MO ou sur Interlis peuvent être téléchargées directement à partir du site internet de swisstopo (www.swisstopo.ch) ou à l'adresse www.interlis.ch.

Version du 01.10.2006	Validation	Distribution	Classement
Remplace version du 27.09.2004		Interne/externe	6020

TRANSFER INTERLIS1;

```

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!
!! Modele de donnees 2001 de la mensuration officielle "Vaud"
!! (MD01_VD) decrit en INTERLIS version 1 (SN 612030).
!!
!!
!! Derivee de la version 24 du modele federal
!! Nom du fichier: md01movd.ili
!!
!! Modification: v1.2.6 23.09.2004 RH
!! Adaptation du commentaire pour les PL dans le topic BF
!! Version approuvee par la D+M
!! Modification: v1.2.6 26.08.2004 CPy
!! Ajout plus fin d'un attribut UFID comme LINEATTR sur
!! les objets SURFACE (voir v1.2.3)
!! Modification: v1.2.5 30.07.2004 CPy
!! Attribut Genre_format de Bord_de_plan: les valeurs
!! permettent de tenir compte de la taille et de l'orientation
!! Modification: v1.2.4 27.07.2004 CPy
!! Attribut Designation OPTIONAL
!! Modification: v1.2.3 20.07.2004 C-Plan
!! Ajout d'un attribut UFID comme LINEATTR sur tous les
!! les objets AREA et SURFACE
!! Modification: v1.2.2 07.07.2004 CPy
!! Version approuvee par la D+M
!! Modification: v1.2.1 29.06.2004 RH, CPy
!! Suite aux remarques R Balanche D+M
!! Modification: v1.2.0 04.06.2004 RH, CPy
!! Prise en compte des changements du MD.01 version 24
!! Modification: v1.1.2 26.03.2004 RH, CPy
!! Ajout d'une TABLE Mise_a_jourRP au
!! TOPIC Repartitions_plans afin de
!! supporter les mutations decentralisees
!! Modification: v1.1.1 13.05.2003 CPy
!! Change la cardinalite de Origine pour tous les points
!! particuliers (CS, OD et CO)
!! supporter les mutations decentralisees
!! Modification: v1.1.0 29.04.2003 CPy
!! Ajout d'un identifiant unique sur tous les objets afin de
!! supporter les mutations decentralisees
!! Modification: v1.0.4 20.02.2003 RH, CPy
!! Mise en conformite avec la version 24 du modele federal
!! Modification: v1.0.3 23.01.2003 CPy
!! OD: corrige Genre_OD, corrige Designation_Batiment_OD, migre
!! Designation de Nom_Objet vers Numero_Objet
!! BF: rajoute SuperficieRF A DDP
!! Modification: v1.0.2 14.11.2002 CPy
!! corrige "object" en "objet"
!! Modification: v1.0.1 13.11.2002 RH, CPy
!! Suite au COPIL du 05.11.2002
!! Modification: v1.0.0 15.08.2002 CPy
!! Version 1.0.0 approuvee par D+M
!! Modification: v0.0.5 31.07.2002 CPy
!! Suite aux remarques R Balanche D+M
!! Modification: v0.0.4 09.07.2002 CPy
!! Suite aux remarques L. Huguenin
!! Modification: v0.0.3 05.07.2002 RH, CPy
!! Modification: v0.0.2 04.07.2002 RH, CPy
!! Creation: v0.0.1 14.06.2002 RH, CPy

```

!!

MODEL MD01MOVVD

DOMAIN

```
CoordP = COORD2      480000.000      70000.000
                    850000.000      310000.000;
```

```
CoordA = COORD3      480000.000        70000.000       -200.000  
                     850000.000        310000.000        5000.000;
```

Altitude = DIM1 -200.000 5000.000;

```
Precision = [0.0 .. 700.0]; !! en cm
```

```
Fiabilite = (
    oui,  !! suffisante
    non); !! insuffisante
```

```
Statut = (
    projete,
    valable);
```

```
StandardQualite = (
    M093,
    MP74,
    NP,
    PRP,    !! Produits de remplacement provisoires
    autre); !! Pas d'objet dans la categorie autre, uniquement pour extensions
```

```
Rotation = GRADS 0.0 399.9;
```

```
GrandeurEcriture = (
    petite,
    moyenne,
    grande);
```

```
StyleEcriture = (
    normal,
    ecarte,
    autre); !! Pas d'objet dans la categorie autre, uniquement pour extensions
```

```
Materiel = (
    borne,
    borne_artificielle,
    cheville,
    tuyau,
    pieu,
    croix,
    non_materialise,
    autre); !! Pas d'objet dans la categorie autre, uniquement pour extensions
```

```
Statut_mise_a_jour_AB = (    !! Pour adresses_batiment, voir SN 612040
    projete,    !! nouvel objet en travail
                !! Etat provisoire ou non reconnu
    reel,       !! L'objet existe vraiment
    caduc);     !! L'objet n'existe plus
```

```
TypeLangue = ( !! Pour adresses_batiment, voir SN 612040
  de,      !! deutsch
  fr,      !! francais
  it,      !! italiano
  rm,      !! rumantsch
  en);     !! english
```

```
IdUnique = [0 .. 9999999999]; !!VD pour traitement mutations
```

```
Designation_Batiment = (                                !!VD
  batiment_administratif (                                !!VD
    batiment_de_l_administration,                          !!VD
    batiment_scolaire,                                    !!VD
    batiment_ecclesiastique,                              !!VD
    batiment_hospitalier,                                !!VD
    batiment_public),                                    !!VD
  maison_d_habitation (                                  !!VD
    habitation,                                           !!VD
    habitation_et_garage,                                !!VD
    habitation_affectation_mixte),                        !!VD
  batiment_agricole (                                    !!VD
    habitation_et_rural,                                  !!VD
    batiment_agricole),                                  !!VD
  batiment_commercial,                                  !!VD
  batiment_industriel,                                  !!VD
  petit_batiment_dependance (                            !!VD
    batiment,                                             !!VD
    garage),                                             !!VD
  couvert);                                              !!VD uniquement pour les objets de
type Numero_Obj et du TOPIC Objets_divers
```

```
TOPIC Points_fixesCategorie1 =
```

```
TABLE Mise_a_jourPFP1 =
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !! attribution par swisstopo
  Description: TEXT*30;
  Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.200;
  !! Pour les mises a jour futures la date a renseigner est En_vigueur.
  !! Date1 correspond aux anciennes mises a jour.
  !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
  !! En_vigueur sera declare obligatoire.
  En_vigueur: OPTIONAL DATE;
  Date1: OPTIONAL DATE; !! par ex. date de mise en service
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT IdentDN, Identification;
  END Mise_a_jourPFP1;
```

```
!! Lorsque le PFP1 est aussi point limite territoriale, il est necessaire
!! de renseigner Signe, ce point etant copie dans
!! limites_commune.point_limite_ter
```

```
TABLE PFP1 = !! point de triangulation I-IIIe ordre
  Origine: -> Mise_a_jourPFP1; !! relation 1-mc
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Numero: TEXT*12; !! attribution par swisstopo
  Geometrie: CoordP;
  GeomAlt: OPTIONAL Altitude;
  PrecPlan: Precision;
  FiabPlan: Fiabilite;
  PrecAlt: OPTIONAL Precision; !! dependant de GeomAlt
```

```
FiabAlt: OPTIONAL Fiabilite; !! dependant de GeomAlt
Accessibilite: (
    accessible,
    inaccessible);
Signe: OPTIONAL Materiel;
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Numero;
END PFP1;

TABLE PosPFP1 =
    PosPFP1_de: -> PFP1; !! relation 1-1; inscription de Numero
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 100.0 //;
    HALi: OPTIONAL HALIGNMENT // indefini = Left //;
    VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT PosPFP1_de;
END PosPFP1;

TABLE SymbolePFP1 =
    SymbolePFP1_de: -> PFP1; !! relation 1-c
    Ori: OPTIONAL Rotation // indefini = 0.0 //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT SymbolePFP1_de;
END SymbolePFP1;

TABLE Mise_a_jourPFA1 =
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
    Identification: TEXT*12; !! attribution par swisstopo
    Description: TEXT*30;
    Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
        WITHOUT OVERLAPS > 0.200;
    !! Pour les mises a jour futures la date a renseigner est En_vigueur.
    !! Date1 correspond aux anciennes mises a jour.
    !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
    !! En_vigueur sera declare obligatoire.
    En_vigueur: OPTIONAL DATE;
    Date1: OPTIONAL DATE; !! par ex. date de mise en service
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Mise_a_jourPFA1;

TABLE PFA1 = !! nivellement federal
    Origine: -> Mise_a_jourPFA1; !! relation 1-mc
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
    Numero: TEXT*12; !! attribution par swisstopo
    Geometrie: CoordP;
    GeomAlt: Altitude;
    PrecPlan: OPTIONAL Precision;
    FiabPlan: OPTIONAL Fiabilite;
    PrecAlt: Precision;
    FiabAlt: Fiabilite;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Numero; Geometrie;
END PFA1;

TABLE PosPFA1 =
    PosPFA1_de: -> PFA1; !! relation 1-1; inscription de Numero
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 100.0 //;
    HALi: OPTIONAL HALIGNMENT // indefini = Left //;
```

```

    VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT PosPFA1_de;
    END PosPFA1;

```

```
END Points_fixesCategorie1.
```

```
TOPIC Points_fixesCategorie2 =
```

```

TABLE Mise_a_jourPFP2 =
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !! par ex. numero du dossier technique
  Description: TEXT*30;
  Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.200;
  !! Pour les mises a jour futures la date a renseigner est En_vigueur.
  !! Date1 correspond aux anciennes mises a jour.
  !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
  !! En_vigueur sera declare obligatoire.
  En_vigueur: OPTIONAL DATE;
  Date1: OPTIONAL DATE; !! par ex. date de mise en service
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT IdentDN, Identification;
  END Mise_a_jourPFP2;

```

```

!! Lorsque le PFP2 est aussi point limite territoriale, il est necessaire
!! de renseigner Signe, ce point etant copie dans
!! limites_commune.point_limite_ter

```

```

TABLE PFP2 = !! point de triangulation IVE ordre
  Origine: -> Mise_a_jourPFP2; !! relation 1-mc
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Numero: TEXT*12; !! attribution par swisstopo
  Geometrie: CoordP;
  GeomAlt: OPTIONAL Altitude;
  PrecPlan: Precision;
  FiabPlan: Fiabilite;
  PrecAlt: OPTIONAL Precision; !! dependant de GeomAlt
  FiabAlt: OPTIONAL Fiabilite; !! dependant de GeomAlt
  Accessibilite: (
    accessible,
    inaccessible);
  Signe: OPTIONAL Materiel;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT IdentDN, Numero;
  END PFP2;

```

```

TABLE PosPFP2 =
  PosPFP2_de: -> PFP2; !! relation 1-1; inscription de Numero
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Left //;
  VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT PosPFP2_de;
  END PosPFP2;

```

```

TABLE SymbolePFP2 =
  SymbolePFP2_de: -> PFP2; !! relation 1-c
  Ori: OPTIONAL Rotation // indefini = 0.0 //;

```

```
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT SymbolePFP2_de;
END SymbolePFP2;
```

```
TABLE Mise_a_jourPFA2 =
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !! par ex. numero du dossier technique
  Description: TEXT*30;
  Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.200;
  !! Pour les mises a jour futures la date a renseigner est En_vigueur.
  !! Date1 correspond aux anciennes mises a jour.
  !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
  !! En_vigueur sera declare obligatoire.
  En_vigueur: OPTIONAL DATE;
  Date1: OPTIONAL DATE; !! par ex. date de mise en service
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT IdentDN, Identification;
END Mise_a_jourPFA2;
```

```
TABLE PFA2 = !! nivellement cantonal
  Origine: -> Mise_a_jourPFA2; !! relation 1-mc
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Numero: TEXT*12; !! attribution par le canton
  Geometrie: CoordP;
  GeomAlt: Altitude;
  PrecPlan: OPTIONAL Precision;
  FiabPlan: OPTIONAL Fiabilite;
  PrecAlt: Precision;
  FiabAlt: Fiabilite;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT IdentDN, Numero; Geometrie;
END PFA2;
```

```
TABLE PosPFA2 =
  PosPFA2_de: -> PFA2; !! relation 1-1; inscription de Numero
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Left //;
  VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT PosPFA2_de;
END PosPFA2;
```

```
END Points_fixesCategorie2.
```

```
TOPIC Points_fixesCategorie3 =
```

```
TABLE Mise_a_jourPFP3 =
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !! par ex. numero du dossier technique
  Description: TEXT*30;
  Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.200;
  !! Pour les mises a jour futures la date a renseigner est En_vigueur.
  !! Date1 correspond aux anciennes mises a jour.
  !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
  !! En_vigueur sera declare obligatoire.
  En_vigueur: OPTIONAL DATE;
```

```
Date1: OPTIONAL DATE; !! par ex. date de mise en service
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Mise_a_jourPFP3;
```

```
TABLE PFP3 =
    !! anciennement point de base, point intercalaire,
    !! point de polygone, point ajustage.
    Origine: -> Mise_a_jourPFP3; !! relation 1-mc
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
    Numero: TEXT*12; !! attribution par le canton
    Geometrie: CoordP;
    GeomAlt: OPTIONAL Altitude;
    PrecPlan: Precision;
    FiabPlan: Fiabilite;
    PrecAlt: OPTIONAL Precision; !! dependant de GeomAlt
    FiabAlt: OPTIONAL Fiabilite; !! dependant de GeomAlt
    Signe: Matériel; !! Seul Non materialise pas permis
    Fiche: (
        oui,
        non);
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT IdentDN, Numero; Geometrie;
END PFP3;
```

```
TABLE PosPFP3 =
    PosPFP3_de: -> PFP3; !! relation 1-1; inscription de Numero
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 100.0 //;
    HALi: OPTIONAL HALIGNMENT // indefini = Left //;
    VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT PosPFP3_de;
END PosPFP3;
```

```
TABLE SymbolePFP3 =
    SymbolePFP3_de: -> PFP3; !! relation 1-c
    Ori: OPTIONAL Rotation // indefini = 0.0 //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT SymbolePFP3_de;
END SymbolePFP3;
```

```
TABLE Mise_a_jourPFA3 =
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
    Identification: TEXT*12; !! par ex. numero du dossier technique
    Description: TEXT*30;
    Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
        WITHOUT OVERLAPS > 0.200;
    !! Pour les mises a jour futures la date a renseigner est En_vigueur.
    !! Date1 correspond aux anciennes mises a jour.
    !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
    !! En_vigueur sera declare obligatoire.
    En_vigueur: OPTIONAL DATE;
    Date1: OPTIONAL DATE; !! par ex. date de mise en service
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT IdentDN, Identification;
END Mise_a_jourPFA3;
```

```
TABLE PFA3 = !! nivellement communal, si PFP3 sans altitude
    Origine: -> Mise_a_jourPFA3; !! relation 1-mc
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
```



```

Numero: TEXT*12; !! attribution par le canton
Geometrie: CoordP;
GeomAlt: Altitude;
PrecPlan: OPTIONAL Precision;
FiabPlan: OPTIONAL Fiabilite;
PrecAlt: Precision;
FiabAlt: Fiabilite;
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Numero; Geometrie;
END PFA3;

TABLE PosPFA3 =
  PosPFA3_de: -> PFA3; !! relation 1-1; inscription de Numero
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Left //;
  VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT PosPFA3_de;
  END PosPFA3;

```

END Points_fixesCategorie3.

TOPIC Couverture_du_sol =

DOMAIN

```

Genre_CS = (
  batiment,
  revetement_dur (
    route_chemin,
    trottoir,
    ilot,
    chemin_de_fer,
    place_aviation,
    bassin,
    autre_revetement_dur (
      acces_place, !!VD
      autre)),      !!VD
  verte (
    champ_pre_paturage,
    culture_intensive (
      vigne,
      autre_culture_intensive),
    jardin,
    tourbiere,
    autre_verte),
  eau (
    eau_stagnante,
    cours_eau,
    roseliere),
  boisee (
    foret_dense,
    paturage_boise ( !! voir explications chap. 3.4
      paturage_boise_dense,
      paturage_boise_ouvert),
    autre_boisee),
  sans_vegetation (
    rocher,

```

```

        glacier_neve,
        eboulis_sable,
        graviere_decharge,
        autre_sans_vegetation (
            divers,          !!VD categorie existante et utilisee
            autre));        !!VD

TABLE Mise_a_jourCS =
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
    Identification: TEXT*12; !! par ex. numero du dossier technique
    Description: TEXT*30;
    Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
        WITHOUT OVERLAPS > 0.050;
    Valide: Statut;
    !! Pour les mises a jour futures la date a renseigner est En_vigueur.
    !! Date1 correspond aux anciennes mises a jour.
    !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
    !! En_vigueur sera declare obligatoire.
    En_vigueur: OPTIONAL DATE;
    Date1: OPTIONAL DATE; !! par ex. date de mise en service
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Mise_a_jourCS;

TABLE SurfaceCSProj=
    Origine: -> Mise_a_jourCS
        // Valide = projete //; !! relation 1-mc
    Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
        WITHOUT OVERLAPS > 0.050;
    Qualite: StandardQualite;
    Genre: Genre_CS;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END SurfaceCSProj;

TABLE NumeroBatimentProj =
    NumeroBatimentProj_de: -> SurfaceCSProj // Genre = batiment //;
        !! relation 1-mc
    Numero: TEXT*12; !! attribution par le canton (resp. la commune)
    !! RegBL_EGID si la definition du batiment correspond a celle de l'OFS
    RegBL_EGID: OPTIONAL [1..999999999];
    Designation: OPTIONAL Designation_Batiment; !!VD
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END NumeroBatimentProj;

TABLE PosNumeroBatimentProj =
    PosNumeroBatimentProj_de: -> NumeroBatimentProj; !! relation 1-mc;
        !! inscription de Numero
    Pos: CoordP;
    Ori: Rotation;
    HALi: OPTIONAL HALIGNMENT // indefini = Center //;
    VALi: OPTIONAL VALIGNMENT // indefini = Half //;
    Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosNumeroBatimentProj;

TABLE NomObjetProj =
    NomObjetProj_de: -> SurfaceCSProj; !! relation 1-mc
    Nom: TEXT*30;

```

UFID: OPTIONAL IdUnique; !!VD pour traitement mutations

NO IDENT

END NomObjetProj;

TABLE PosNomObjetProj =

PosNomObjetProj_de: -> NomObjetProj; !! relation 1-mc; inscription de Nom

Pos: CoordP;

Ori: Rotation;

HALi: OPTIONAL HALIGNMENT // indefini = Center //;

VALi: OPTIONAL VALIGNMENT // indefini = Half //;

Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;

UFID: OPTIONAL IdUnique; !!VD pour traitement mutations

NO IDENT

END PosNomObjetProj;

!! Se referer aussi aux commentaires de SymboleSurfaceCS.

TABLE SymboleSurfaceCSProj =

SymboleSurfCSProj_de: -> SurfaceCSProj; !! relation 1-mc

Pos: CoordP // Pos dans SurfaceCSProj //;

Ori: OPTIONAL Rotation // indefini = 0.0 //;

UFID: OPTIONAL IdUnique; !!VD pour traitement mutations

NO IDENT

END SymboleSurfaceCSProj;

TABLE SurfaceCS =

Origine: -> Mise_a_jourCS

// Validite = valable //; !! relation 1-mc

Geometrie: AREA WITH (STRAIGHTS, ARCS) VERTEX CoordP

WITHOUT OVERLAPS > 0.050

LINEATTR =

UFID: OPTIONAL IdUnique; !!VD pour traitement mutations

END;

Qualite: StandardQualite;

Genre: Genre_CS;

UFID: OPTIONAL IdUnique; !!VD pour traitement mutations

NO IDENT

END SurfaceCS;

TABLE Numero_de_batiment =

Numero_de_batiment_de: -> SurfaceCS // Genre = batiment //; !! relation 1-

mc

Numero: TEXT*12;

!! RegBL_EGID si la definition du batiment correspond a celle de l'OFS

RegBL_EGID: OPTIONAL [1..999999999];

Designation: OPTIONAL Designation_Batiment; !!VD

UFID: OPTIONAL IdUnique; !!VD pour traitement mutations

NO IDENT

END Numero_de_batiment;

TABLE PosNumero_de_batiment =

PosNumero_de_batiment_de: -> Numero_de_batiment; !! relation 1-mc;

!! Inscription de Numero

Pos: CoordP;

Ori: Rotation;

HALi: OPTIONAL HALIGNMENT // indefini = Center //;

VALi: OPTIONAL VALIGNMENT // indefini = Half //;

Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;

UFID: OPTIONAL IdUnique; !!VD pour traitement mutations

NO IDENT

END PosNumero_de_batiment;

```

TABLE Nom_objet =
  Nom_objet_de: -> SurfaceCS; !! relation 1-mc
  Nom: TEXT*30;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Nom_objet;

TABLE PosNom_objet =
  PosNom_objet_de: -> Nom_objet; !! relation 1-mc; inscription de Nom
  Pos: CoordP;
  Ori: Rotation;
  HALi: OPTIONAL HALIGNMENT // indefini = Center //;
  VALi: OPTIONAL VALIGNMENT // indefini = Half //;
  Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosNom_objet;

!! Sur le plan du registre foncier, les surfaces de la couverture du sol
!! seront remplies soit a l'aide de trames, soit avec des symboles. Ainsi,
!! en fonction du genre, uniquement les symboles suivants sont
judicieusement
!! figures:
!! revetement_dur.bassin, vigne, tourbiere (symbole marais),
eau.eau_stagnante
!! (symbole bassin), eau.cours_eau (symbole direction du courant),
!! eau.roseliere (symbole roseliere).
TABLE SymboleSurfaceCS =
  SymboleSurfaceCS_de: -> SurfaceCS; !! relation 1-mc
  Pos: CoordP // Pos dans SurfaceCS //;
  Ori: OPTIONAL Rotation // indefini = 0.0 //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END SymboleSurfaceCS;

TABLE Point_particulier =
  Origine: OPTIONAL -> Mise_a_jourCS; !! relation 1-mc
  IdentDN: TEXT*12; !!VD relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !!VD obligatoire pour etre coherent avec IdentDN
  Geometrie: CoordP
  // pas de PFP1, PFP2, PFP3, Point_limite ou Point_limite_ter//;
  PrecPlan: Precision;
  FiabPlan: Fiabilite;
  Defini_exactement: ( !! tenir compte des tolerances selon OTEMO
    oui,
    non);
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification; Geometrie; !!VD
END Point_particulier;

TABLE PosPoint_particulier =
  PosPoint_particulier_de: -> Point_particulier; !! relation 1-c;
  !! inscription de Identification

  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Left //;
  VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT PosPoint_particulier_de;
END PosPoint_particulier;

```

END Couverture_du_sol.

TOPIC Objets_divers =

DOMAIN

```
Genre_OD = (  
  mur,  
  batiment_souterrain,  
  autre_corps_de_batiment (  
    details,      !!VD  
    mur_mitoyen,  !!VD  
    autre),       !!VD  
  eau_canalisee_souterraine,  
  escalier_important,  
  tunnel_passage_inferieur_galerie,  
  pont_passerelle,  
  quai, !! quai de chemin de fer  
  fontaine,  
  reservoir,  
  pilier,  
  couvert_independant,  
  silo_tour_gazometre,  
  cheminee,  
  monument,  
  mat_antenne,  
  tour_panoramique,  
  ouvrage_de_protection_des_rives,  
  seuil,  
  paravalanche,  
  socle_massif,  
  ruine_objet_archeologique,  
  débarcadere,  
  bloc_erratique,  
  cordon_boise,  
  ru,  
  sentier,  
  ligne_aerienne_a_haute_tension,  
  conduite_forcee,  
  voie_ferree,  
  telepherique,  
  telecabine_telesiege,  
  telepherique_de_chantier,  
  skilift,  
  bac,  
  grotte_entree_de_caverne,  
  axe,  
  arbre_isole_important,  
  statue_crucifix,  
  source,  
  point_de_reference,  
  autre (  
    piscine,      !!VD  
    terrain_de_sport, !!VD  
    autre,        !!VD  
    quai_de_chargement_a_ventiler, !!VD temporaire, uniquement pour  
rattrapage, pas de nouveaux objets  
    eau_a_ventiler, !!VD temporaire, uniquement pour  
rattrapage, pas de nouveaux objets
```

```

        acces_sentier_a_ventiler,      !!VD temporaire, uniquement pour
rattrapage, pas de nouveaux objets
        trottoir_a_ventiler,          !!VD temporaire, uniquement pour
rattrapage, pas de nouveaux objets
        bord_de_chaussee_a_ventiler,  !!VD temporaire, uniquement pour
rattrapage, pas de nouveaux objets
        berme_ilot_a_ventiler,        !!VD temporaire, uniquement pour
rattrapage, pas de nouveaux objets
        rocher_a_ventiler,            !!VD temporaire, uniquement pour
rattrapage, pas de nouveaux objets
        perre_a_ventiler,             !!VD temporaire, uniquement pour
rattrapage, pas de nouveaux objets
        barrage_a_ventiler,           !!VD temporaire, uniquement pour
rattrapage, pas de nouveaux objets
        divers_a_ventiler             !!VD temporaire, uniquement pour
rattrapage, pas de nouveaux objets
    ));

```

```

TABLE Mise_a_jourOD =
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
    Identification: TEXT*12; !! par ex. numero du dossier technique
    Description: TEXT*30;
    Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
        WITHOUT OVERLAPS > 0.050;
    Validite: Statut;
    !! Pour les mises a jour futures la date a renseigner est En_vigueur.
    !! Date1 correspond aux anciennes mises a jour.
    !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
    !! En_vigueur sera declare obligatoire.
    En_vigueur: OPTIONAL DATE;
    Date1: OPTIONAL DATE; !! par ex. date de mise en service
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT IdentDN, Identification;
END Mise_a_jourOD;

```

```

TABLE Objet_divers =
    Origine: -> Mise_a_jourOD; !! relation 1-mc
    Qualite: StandardQualite;
    Genre: Genre_OD;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    NO IDENT
END Objet_divers;

```

```

TABLE Element_surfacique =
    Element_surfacique_de: -> Objet_divers; !! relation 1-mc
    Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
        WITHOUT OVERLAPS > 0.050;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    NO IDENT
END Element_surfacique;

```

```

TABLE SymboleElement_surf = !! par ex. direction du courant pour un ru
    SymboleElement_surf_de: -> Element_surfacique; !! relation 1-mc
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 0.0 //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    NO IDENT
END SymboleElement_surf;

```

```

TABLE Element_lineaire =
    Element_lineaire_de: -> Objet_divers; !! relation 1-mc

```

```

    Geometrie: POLYLINE WITH (STRAIGHTS, ARCS) VERTEX CoordP;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Element_lineaire;

TABLE SymboleElement_lineaire = !! par ex. bac
    SymboleElement_lineaire_de: -> Element_lineaire; !! relation 1-mc
    Pos: CoordP;
    Ori: Rotation;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END SymboleElement_lineaire;

TABLE Element_ponctuel =
    Element_ponctuel_de: -> Objet_divers; !! relation 1-mc
    Geometrie: CoordP;
    Ori: Rotation;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT

END Element_ponctuel;

TABLE Nom_Objet =
    Nom_Objet_de: -> Objet_divers; !! relation 1-mc
    Nom: TEXT*30;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Nom_Objet;

TABLE PosNom_Objet =
    PosNom_Objet_de: -> Nom_Objet; !! relation 1-mc; inscription de Nom
    Pos: CoordP;
    Ori: Rotation;
    HALi: OPTIONAL HALIGNMENT // indefini = Center //;
    VALi: OPTIONAL VALIGNMENT // indefini = Half //;
    Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosNom_Objet;

TABLE Numero_Objet =
    Numero_Objet_de: -> Objet_divers; !! relation 1-mc
    Numero: TEXT*12;
    !! RegBL_EGID si la definition du batiment correspond a celle de l'OFS
    RegBL_EGID: OPTIONAL [1..999999999];
    Designation: OPTIONAL Designation_Batiment; !!VD uniquement pour les
objets surfaciques de genre batiment_souterrain
                                !!VD ou couvert_independant
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Numero_Objet;

TABLE PosNumero_Objet =
    PosNumero_Objet_de: -> Numero_Objet; !! relation 1-mc; inscription de
Numero
    Pos: CoordP;
    Ori: Rotation;
    HALi: OPTIONAL HALIGNMENT // indefini = Center //;
    VALi: OPTIONAL VALIGNMENT // indefini = Half //;
    Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations

```

```
NO IDENT
END PosNumero_Objet;
```

```
TABLE Point_particulier =
  Origine: OPTIONAL -> Mise_a_jourOD; !! relation 1-mc
  IdentDN: TEXT*12; !!VD relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !!VD obligatoire pour etre coherent avec
IdentDN
  Geometrie: CoordP
  // pas de PFP1, PFP2, PFP3, Point_limite ou Point_limite_ter//;
  PrecPlan: Precision;
  FiabPlan: Fiabilite;
  Defini_exactement: ( !! tenir compte des tolerances selon OTEM0
    oui,
    non);
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification; Geometrie; !!VD
END Point_particulier;
```

```
TABLE PosPoint_particulier =
  PosPoint_particulier_de: -> Point_particulier; !! relation 1-c;
  !! inscription de Identification
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Left //;
  Vali: OPTIONAL VALIGNMENT // indefini = Bottom //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT PosPoint_particulier_de;
END PosPoint_particulier;
```

```
END Objets_divers.
```

```
TOPIC Altimetrie =
```

```
TABLE Mise_a_jourAL =
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !! par ex. numero du dossier technique
  Description: TEXT*30;
  Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.200;
  Validite: Statut;
  !! Pour les mises a jour futures la date a renseigner est En_vigueur.
  !! Date1 correspond aux anciennes mises a jour.
  !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
  !! En_vigueur sera declare obligatoire.
  En_vigueur: OPTIONAL DATE;
  Date1: OPTIONAL DATE; !! par ex. date de mise en service
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Mise_a_jourAL;
```

```
TABLE Point_cote = !! valable ou projete
  Origine: -> Mise_a_jourAL; !! relation 1-mc
  Geometrie: CoordA;
  Qualite: StandardQualite;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT Geometrie;
END Point_cote;
```



```

TABLE PosPoint_cote =
  PosPoint_cote_de: -> Point_cote; !! relation 1-c; inscription de Geometrie
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Left //;
  VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT PosPoint_cote_de;
END PosPoint_cote;

```

```

TABLE Arete =
  Origine: -> Mise_a_jourAL; !! relation 1-mc
  Geometrie: POLYLINE WITH (STRAIGHTS) VERTEX CoordA; !! sans ARCS !
  Qualite: StandardQualite;
  Genre: (
    ligne_de_rupture,
    ligne_de_structure,
    autre); !! Pas d'objet dans la categorie autre, uniquement pour
extensions
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Arete;

```

```

TABLE Surface_vide =
  Origine: -> Mise_a_jourAL; !! relation 1-mc
  Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP;
  Qualite: StandardQualite;
  Genre: (
    surface_morte,
    autre); !! Pas d'objet dans la categorie autre, uniquement pour
extensions
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Surface_vide;

```

END Altimetrie.

TOPIC Nomenclature =

```

TABLE Mise_a_jourNO =
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !! par ex. numero du dossier technique
  Description: TEXT*30;
  Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.200;
  !! Pour les mises a jour futures la date a renseigner est En_vigueur.
  !! Date1 correspond aux anciennes mises a jour.
  !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
  !! En_vigueur sera declare obligatoire.
  En_vigueur: OPTIONAL DATE;
  Date1: OPTIONAL DATE; !! par ex. date de mise en service
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Mise_a_jourNO;

```

```

TABLE Nom_local =
  Origine: -> Mise_a_jourNO; !! relation 1-mc
  Nom: TEXT*40;
  Geometrie: AREA WITH (STRAIGHTS, ARCS) VERTEX CoordP

```

```
WITHOUT OVERLAPS > 0.200
LINEATTR =
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
END;
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Nom_local;

TABLE PosNom_local =
  PosNom_local_de: -> Nom_local; !! relation 1-mc; inscription de Nom
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HAl: OPTIONAL HALIGNMENT // indefini = Center //;
  VAl: OPTIONAL VALIGNMENT // indefini = Half //;
  Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
  Style: OPTIONAL StyleEcriture // indefini = normal //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosNom_local;

TABLE Nom_de_lieu =
  Origine: -> Mise_a_jourNO; !! relation 1-mc
  Nom: TEXT*40;
  Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.200;
  Type: OPTIONAL TEXT*30; !! attribution par le canton
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Nom_de_lieu;

TABLE PosNom_de_lieu =
  PosNom_de_lieu_de: -> Nom_de_lieu; !! relation 1-mc; inscription de Nom
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HAl: OPTIONAL HALIGNMENT // indefini = Center //;
  VAl: OPTIONAL VALIGNMENT // indefini = Half //;
  Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
  Style: OPTIONAL StyleEcriture // indefini = normal //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosNom_de_lieu;

TABLE Lieudit =
  Origine: -> Mise_a_jourNO; !! relation 1-mc
  Nom: TEXT*40;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Lieudit;

TABLE PosLieudit =
  PosLieudit_de: -> Lieudit; !! relation 1-m; inscription de Nom
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HAl: OPTIONAL HALIGNMENT // indefini = Center //;
  VAl: OPTIONAL VALIGNMENT // indefini = Half //;
  Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
  Style: OPTIONAL StyleEcriture // indefini = normal //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosLieudit;
```

END Nomenclature.

TOPIC Biens_fonds =

DOMAIN

```
Genre_Immeuble = (
  bien_fonds (
    parcelle_prive, !!VD
    DP_communal,    !!VD
    DP_cantonal),   !!VD
  DDP (
    superficie,
    source,
    concession,
    autre), !! Pas d'objet dans la categorie autre,
          !! uniquement pour extensions
  mine);
```

TABLE Mise_a_jourBF =

```
IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
Identification: TEXT*12; !! par ex. numero du dossier technique
Description: TEXT*30;
Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
  WITHOUT OVERLAPS > 0.050;
Validite: Statut;
!! Pour les mises a jour futures les dates a renseigner sont En_vigueur.
!! et Enregistrement_RF. Date1 et Date2 correspondent aux anciennes
!! mises a jour. Dans la prochaine revision du modele de donnees Date1 et
!! Date 2 seront supprimees, En_vigueur sera declare obligatoire.
En_vigueur: OPTIONAL DATE; !! traitement technique
Enregistrement_RF: OPTIONAL DATE;
Date1: OPTIONAL DATE; !! par ex. traitement technique
Date2: OPTIONAL DATE; !! attribution par le canton
                        !! par ex. introduction au registre foncier,
                        !! reconnaissance de la MO
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Mise_a_jourBF;
```

!! Comprend tous les points limites d'un bien-fonds a l'exception des points
!! limites de bien-fonds qui sont aussi PFP3.

!! Se referer aussi aux remarques sur les points limites
!! territoriaux (topic Limites_commune).

TABLE Point_limite =

```
Origine: -> Mise_a_jourBF; !! relation 1-mc
IdentDN: TEXT*12;          !!VD relation 1-m avec Domaine_numerotation
Identification: TEXT*12;   !!VD obligatoire pour etre coherent avec
```

IdentDN

```
Geometrie: CoordP;
PrecPlan: Precision;
FiabPlan: Fiabilite;
Signe: Matériel;
Defini_exactement: ( !! tenir compte des tolerances selon l'OTEMO
  oui,
  non);
```

!! Suite au declassement d'une ancienne borne speciale de limite
territoriale

!! en un point limite uniquement (voir aussi explications chap. 3.11).

```

    Anc_borne_speciale: ( !! indication de la materialisation
        oui,
        non);
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification; Geometrie; !!VD
END Point_limite;

TABLE PosPoint_limite =
    PosPoint_limite_de: -> Point_limite; !! relation 1-c;
        !! inscription de Identification
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 100.0 //;
    HALi: OPTIONAL HALIGNMENT // indefini = Left //;
    VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT PosPoint_limite_de;
END PosPoint_limite;

TABLE SymbolePoint_limite =
    SymbolePoint_limite_de: -> Point_limite; !! relation 1-c
    Ori: OPTIONAL Rotation // indefini = 0.0 //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT SymbolePoint_limite_de;
END SymbolePoint_limite;

TABLE ImmeubleProj =
    Origine: -> Mise_a_jourBF
        // Validite = projete //; !! relation 1-mc
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
    Numero: TEXT*12;
    !! Identificateur du systeme electronique d'informations foncieres
    EGRIS_EGRID: OPTIONAL TEXT*14;
    !! Contenu de l'attribut: doit etre litigieux si le Bien_fondsProj,
    !! le DDPProj, ou la MineProj est litigieux;

    Validite: (
        en_vigueur, !! prevu en vigueur
        litigieux);
    !! incomplet si par ex. l'immeuble est situe en partie
    !! a l'exterieur du perimetre.
    Integralite: (
        complet,
        incomplet);
    Genre: Genre_Immeuble;
    !! Superficie_totale sera renseignee uniquement dans le cas des parties
    !! d'immeubles. Cela signifie que plusieurs objets Bien_fondsProj, DDPProj
    !! ou MineProj forment un objet ImmeubleProj.
    Superficie_totale: OPTIONAL DIM2 1 999999999;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT Origine, IdentDN, Numero;
END ImmeubleProj;

TABLE PosImmeubleProj =
    PosImmeubleProj_de: -> ImmeubleProj; !! relation 1-m; inscription Numero
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 100.0 //;
    HALi: OPTIONAL HALIGNMENT // indefini = Center //;
    VALi: OPTIONAL VALIGNMENT // indefini = Half //;
    Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
    Ligne_auxiliaire: OPTIONAL POLYLINE WITH (STRAIGHTS) VERTEX CoordP;
    !! trait de rappel pour numero d'immeuble

```

```

    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosImmeubleProj;

TABLE Bien_fondsProj =
    Bien_fondsProj_de: -> ImmeubleProj // Genre = parcelle_prive, DP_communal
ou DP_cantonal //; !! relation 1-mc
    !! Numero_ PartieNumeroImmeuble si necessaire partie d'immeubleProj
    PartieNumeroImmeuble: OPTIONAL TEXT*12;
    Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
    // Geometrie uniquement PFP1, PFP2, PFP3, Point_limite ou
    Point_limite_ter //
    WITHOUT OVERLAPS > 0.050
    LINEATTR =
        Genre_ligne: OPTIONAL (
            !! indefini pour en vigueur et complet
            litigieux,
            incomplet);
        UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    END;
    Superficie: DIM2 1 999999999;
    SuperficieRF: DIM2 1 999999999; !!VD pour concordance avec le RF
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Bien_fondsProj;

!! Si un DDPProj n'a pas de surface, alors il n'existe aussi pas d'objet.
!! Une inscription avec une localisation dans PosImmeubleProj est tout de
meme
!! possible.
TABLE DDPProj =
    DDPProj_de: -> ImmeubleProj // Genre = superficie, source, concession ou
autre //;
    !! relation 1-mc
    !! Numero_ PartieNumeroImmeuble si necessaire partie d'immeubleProj
    PartieNumeroImmeuble: OPTIONAL TEXT*12;
    Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
    // Geometrie uniquement PFP1, PFP2, PFP3, Point_limite ou
    Point_limite_ter //
    WITHOUT OVERLAPS > 0.050
    LINEATTR =
        Genre_ligne: OPTIONAL (
            !! indefini pour en vigueur et complet
            litigieux,
            incomplet);
        UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    END;
    Superficie: DIM2 1 999999999;
    SuperficieRF: DIM2 1 999999999; !!VD pour concordance avec le RF
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END DDPProj;

!! Si une MineProj n'a pas de surface, alors il n'existe pas non plus
d'objet.
!! Une inscription avec une localisation dans PosImmeubleProj est tout de
meme
!! possible.
TABLE MineProj =
    MineProj_de: -> ImmeubleProj // Genre = mine //; !! relation 1-mc
    !! Numero_ PartieNumeroImmeuble si necessaire partie d'immeubleProj

```

```
PartieNumeroImmeuble: OPTIONAL TEXT*12;
Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
// Geometrie uniquement PFP1, PFP2, PFP3, Point_limite ou
Point_limite_ter //
WITHOUT OVERLAPS > 0.050
LINEATTR =
  Genre_ligne: OPTIONAL (
    !! indefini pour en vigueur et complet
    litigieux,
    incomplet);
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
END;
Superficie: DIM2 1 999999999;
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END MineProj;

TABLE Immeuble =
  Origine: -> Mise_a_jourBF
  // Validite = valable //; !! relation 1-mc
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Numero: TEXT*12;
  !! Identificateur du systeme electronique d'informations foncieres
  EGRIS_EGRID: OPTIONAL TEXT*14;
  !! Contenu de l'attribut: doit etre litigieux si le bien-fonds, le DDP
  !! ou la Mine est litigieux;
  Validite: (
    en_vigueur,
    litigieux);
  !! incomplet si par ex. l'immeuble est situe en partie
  !! a l'exterieur du perimetre.
  Integralite: (
    complet,
    incomplet);
  Genre: Genre_Immeuble;
  !! Superficie_totale sera renseignee uniquement dans le cas des parties
  !! d'immeubles. Cela signifie que plusieurs objets Bien_fonds, DDP ou Mine
  !! forment un objet Immeuble
  Superficie_totale: OPTIONAL DIM2 1 999999999;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Numero;
END Immeuble;

TABLE PosImmeuble =
  PosImmeuble_de: -> Immeuble; !! relation 1-m; inscription de Numero
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Center //;
  VALi: OPTIONAL VALIGNMENT // indefini = Half //;
  Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
  Ligne_auxiliaire: OPTIONAL POLYLINE WITH (STRAIGHTS) VERTEX CoordP;
  !! trait de rappel pour numero d'immeuble
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosImmeuble;

TABLE Bien_fonds =
  Bien_fonds_de: -> Immeuble // Genre = parcelle_prive, DP_communal ou
DP_cantonal //; !! relation 1-mc
  !! Numero_PartieNumeroImmeuble si necessaire partie d'immeuble
  PartieNumeroImmeuble: OPTIONAL TEXT*12;
```

```

Geometrie: AREA WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
// Geometrie uniquement PFP1, PFP2, PFP3, Point_limite ou
Point_limite_ter //
WITHOUT OVERLAPS > 0.050
LINEATTR =
  Genre_ligne: OPTIONAL (
    !! indefini pour en vigueur et complet
    litigieux,
    incomplet);
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
END;
Superficie: DIM2 1 999999999;
SuperficieRF: DIM2 1 999999999; !!VD pour concordance avec le RF
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Bien_fonds;

!! Si un DDP n'a pas de surface, alors il n'existe aussi pas d'objet.
!! Une inscription avec une localisation dans PosImmeuble est tout de meme
!! possible.
TABLE DDP =
  DDP_de: -> Immeuble // Genre = superficie, source, concession ou autre //;
    !! relation 1-mc
  !! Numero_PartieNumeroImmeuble si necessaire partie d'immeuble
  PartieNumeroImmeuble: OPTIONAL TEXT*12;
  Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
  // Geometrie uniquement PFP1, PFP2, PFP3, Point_limite ou
  Point_limite_ter //
  WITHOUT OVERLAPS > 0.050
  LINEATTR =
    Genre_ligne: OPTIONAL (
      !! indefini pour en vigueur et complet
      litigieux,
      incomplet);
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  END;
  Superficie: DIM2 1 999999999;
  SuperficieRF: DIM2 1 999999999; !!VD pour concordance avec le RF
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END DDP;

!! Si une mine n'a pas de surface, alors il n'existe aussi pas d'objet.
!! Une inscription avec une localisation dans PosImmeuble est tout de meme
!! possible.
TABLE Mine =
  Mine_de: -> Immeuble // Genre = mine //; !! relation 1-mc
  !! Numero_PartieNumeroImmeuble si necessaire partie d'immeuble
  PartieNumeroImmeuble: OPTIONAL TEXT*12;
  Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
  // Geometrie uniquement PFP1, PFP2, PFP3, Point_limite ou
  Point_limite_ter //
  WITHOUT OVERLAPS > 0.050
  LINEATTR =
    Genre_ligne: OPTIONAL (
      !! indefini pour en vigueur et complet
      litigieux,
      incomplet);
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  END;
  Superficie: DIM2 1 999999999;

```

```
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Mine;
```

```
END Biens_fonds.
```

```
TOPIC Conduites =
!! selon la Loi federale sur les installations de transport par conduites
!! de combustibles ou carburants liquides ou gazeux
```

```
DOMAIN
```

```
    Matiere = (
        petrole,
        gaz,
        autre); !! Pas d'objet dans la categorie autre, uniquement pour
extensions
```

```
TABLE Mise_a_jourCO =
```

```
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
    Identification: TEXT*12; !! par ex. numero du dossier technique
    Description: TEXT*30;
    Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
        WITHOUT OVERLAPS > 0.050;
    Validite: Statut;
    !! Pour les mises a jour futures la date a renseigner est En_vigueur.
    !! Date1 correspond aux anciennes mises a jour.
    !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
    !! En_vigueur sera declare obligatoire.
    En_vigueur: OPTIONAL DATE;
    Date1: OPTIONAL DATE; !! par ex. date de mise en service
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT IdentDN, Identification;
    END Mise_a_jourCO;
```

```
TABLE Element_conduite =
```

```
    Origine: -> Mise_a_jourCO; !! relation 1-mc
    Exploitant: TEXT*30;
    Qualite: StandardQualite;
    Genre: Matiere;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    NO IDENT
    END Element_conduite;
```

```
TABLE PosElement_conduite =
```

```
    PosElement_conduite_de: -> Element_conduite; !! relation 1-mc;
        !! inscription de Exploitant
    Pos: CoordP;
    Ori: Rotation;
    Hali: HALIGNMENT;
    Vali: VALIGNMENT;
    Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    NO IDENT
    END PosElement_conduite;
```

```
TABLE Element_surfacique =
```

```
    Element_surfacique_de: -> Element_conduite; !! relation 1-mc
    Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
```



```

WITHOUT OVERLAPS > 0.050
LINEATTR =
  Genre_ligne: OPTIONAL (
    visible);
    !! L'attribut Genre_ligne doit etre principalement reference comme
    !! indefini a l'exception de la valeur visible.
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
END;
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Element_surfacique;

TABLE Element_lineaire =
  Element_lineaire_de: -> Element_conduite; !! relation 1-mc
  Geometrie: POLYLINE WITH (STRAIGHTS, ARCS) VERTEX CoordP;
  Genre_ligne: OPTIONAL (
    visible);
    !! L'attribut Genre_ligne doit etre principalement reference comme
indefini
    !! a l'exception de la valeur visible.
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Element_lineaire;

TABLE Element_ponctuel =
  Element_ponctuel_de: -> Element_conduite; !! relation 1-mc
  Geometrie: CoordP;
  GeomAlt: OPTIONAL Altitude;
  Ori: Rotation;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Element_ponctuel;

TABLE Signal =
  Origine: -> Mise_a_jourCO; !! relation 1-mc
  Numero: TEXT*12;
  Exploitant: TEXT*30;
  Geometrie: CoordP;
  Qualite: StandardQualite;
  Genre: Matiere;
  Genre_point: (
    balise,
    plaquette_borne,
    autre); !! Pas d'objet dans la categorie autre, uniquement pour
extensions
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Signal;

TABLE PosSignal =
  PosSignal_de: -> Signal; !! relation 1-c; inscription de Numero
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Left //;
  VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT PosSignal_de;
END PosSignal;

TABLE Point_particulier =
  Origine: OPTIONAL -> Mise_a_jourCO; !! relation 1-mc

```

```

Identification: OPTIONAL TEXT*12;
Geometrie: CoordP
  // pas de PFP1, PFP2, PFP3, Point_limite ou Point_limite_ter//;
PrecPlan: Precision;
FiabPlan: Fiabilite;
Defini_exactement: (  !! tenir compte des tolerances selon OTEMO
  oui,
  non);
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT Geometrie;
END Point_particulier;

```

```

TABLE PosPoint_particulier =
  PosPoint_particulier_de: -> Point_particulier; !! relation 1-c;
  !! inscription de Identification
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Left //;
  VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT PosPoint_particulier_de;
END PosPoint_particulier;

```

END Conduites.

TOPIC Domaines_numerotation =

```

DOMAIN
  Abreviation_cantonale = ( !! ordre de l'OFS, completee avec FL et CH
    ZH, BE, LU, UR, SZ, OW, NW, GL, ZG, FR, SO, BS, BL, SH,
    AR, AI, SG, GR, AG, TG, TI, VD, VS, NE, GE, JU, FL, CH);

```

!! La cle utilisateur definie ici et les surfaces associees peuvent
correspondre
!! a une commune, a une partie ou a une aggregation de plusieurs communes et
!! encore event. au canton et/ou a la Suisse (resp. au FL) dans leur
ensemble

```

!! (se referer au document explications).
TABLE Domaine_numerotation =
  Ct: Abreviation_cantonale; !! univoque pour la Suisse (inclus le FL)
  NumeroDN: TEXT*10;
  DossierTech: TEXT*12;
  En_vigueur: OPTIONAL DATE;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT Ct, NumeroDN; !! constitue la cle utilisateur IdentDN
END Domaine_numerotation;

```

!! Les geometries, des domaines de numerotation decoupant le territoire
!! selon la meme unite logique (par ex: perimetre des communes), doivent
!! creer entre elles une partition du territoire (AREA).

```

TABLE GeometriedN =
  GeometriedN_de: -> Domaine_numerotation; !! relation 1-m
  Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.050;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END GeometriedN;

```

TABLE PosDomaine_numerotation =

```

PosDomaine_numerotation_de: -> Domaine_numerotation; !! relation 1-mc;
                                !! inscription de NumeroDN
Pos: CoordP // position dans GeometrieDN //;
Ori: OPTIONAL Rotation // indefini = 100.0 //;
HALi: OPTIONAL HALIGNMENT // indefini = Center //;
VALi: OPTIONAL VALIGNMENT // indefini = Half //;
Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosDomaine_numerotation;

END Domaines_numerotation.

TOPIC Limites_commune =

TABLE Mise_a_jourCOM =
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !! par ex. numero du dossier technique
  Description: TEXT*30;
  Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.050;
  Validite: Statut;
  !! Pour les mises a jour futures la date a renseigner est En_vigueur.
  !! Date1 correspond aux anciennes mises a jour.
  !! Dans la prochaine revision du modele de donnees Date1 sera supprime et
  !! En_vigueur sera declare obligatoire.
  En_vigueur: OPTIONAL DATE;
  Date1: OPTIONAL DATE; !! par ex. date de mise en service
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT IdentDN, Identification;
  END Mise_a_jourCOM;

  !! Contient tous les points d'appui qui definissent une limite territoriale.
  !! (nationale, cantonale, de district et communale) avec les regles
  !! complementaires suivantes:
  !! - Les PFP1, PFP2 et PFP3 qui se situent sur une limite territoriale sont
  !!   aussi contenus ici; lors de la reprise du topic Points_fixes, les
  !!   attributs restent inchanges;
  !! - Bornes limites territoriales materialisees a l'aide de pierre speciale
  !!   (cf. Chap 3.11 des explications): Attribut Borne_territoriale = oui;
  TABLE Point_limite_ter =
    Origine: -> Mise_a_jourCOM; !! relation 1-mc
    Identification: TEXT*12; !!VD obligatoire car IDENT
    Geometrie: CoordP;
    PrecPlan: Precision;
    FiabPlan: Fiabilite;
    Signe: Matériel;
    Borne_territoriale: ( !! indication de la materialisation
      oui,
      non);
    Defini_exactement: ( !! tenir compte des tolerances selon OTEMO
      oui,
      non);
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT Identification; Geometrie;
    END Point_limite_ter;

  TABLE PosPoint_limite_ter =
    PosPoint_limite_ter_de: -> Point_limite_ter; !! relation 1-c;

```

```

!! inscription de Identification
Pos: CoordP;
Ori: OPTIONAL Rotation // indefini = 100.0 //;
HALi: OPTIONAL HALIGNMENT // indefini = Left //;
VALi: OPTIONAL VALIGNMENT // indefini = Bottom //;
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT PosPoint_limite_ter_de;
END PosPoint_limite_ter;

TABLE SymbolePoint_limite_ter =
  SymbolePoint_limite_ter_de: -> Point_limite_ter; !! relation 1-c
  Ori: OPTIONAL Rotation // indefini = 0.0 //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT SymbolePoint_limite_ter_de;
END SymbolePoint_limite_ter;

TABLE Commune =
  Nom: TEXT*30;
  NoOFS: [1 .. 9999];
  NumCom: [1 .. 388]; !!VD numero de commune cantonal
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT NoOFS; NumCom; !!VD
END Commune;

TABLE Limite_communeProj =
  Origine: -> Mise_a_jourCOM
  // Validite = projete //; !! relation 1-mc
  Limite_communeProj_de: -> Commune; !! relation 1-mc
  Geometrie: POLYLINE WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
  // Geometrie uniquement Point_limite_ter //
  WITHOUT OVERLAPS > 0.050;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Limite_communeProj;

TABLE Limite_commune = !! pour rendre possible des exclaves
  Origine: -> Mise_a_jourCOM
  // Validite = valable //; !! relation 1-mc
  Limite_commune_de: -> Commune; !! relation 1-m
  Geometrie: AREA WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
  // Geometrie uniquement Point_limite_ter
  si Genre_ligne = en_vigueur ou litigieux //
  WITHOUT OVERLAPS > 0.050
  LINEATTR =
    Genre_ligne: (
      en_vigueur, !! limite exacte de la mensuration officielle
      litigieux, !! limite litigieuse
      provisoire, !! limite definitive mais qualitativement insuffisante
      indefini); !! par ex. jonction inconnue dans un lac
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  END;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Limite_commune;

END Limites_commune.

TOPIC Limites_district =

```

```
TABLE Partie_limite_district =
  Geometrie: POLYLINE WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
  // Geometrie uniquement Point_limite_ter //;
  Valide: (
    en_vigueur, !! limite exacte de la mensuration officielle
    litigieux, !! limite litigieuse
    provisoire, !! limite definitive mais qualitativement insuffisante
    indefini); !! par ex. jonction inconnue dans un lac
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  NO IDENT
  END Partie_limite_district;

END Limites_district.
```

```
TOPIC Limites_canton =
```

```
TABLE Partie_limite_canton =
  Geometrie: POLYLINE WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
  // Geometrie uniquement Point_limite_ter //;
  Valide: (
    en_vigueur, !! limite exacte de la mensuration officielle
    litigieux, !! limite litigieuse
    provisoire, !! limite definitive mais qualitativement insuffisante
    indefini); !! par ex. jonction inconnue dans un lac
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  NO IDENT
  END Partie_limite_canton;

END Limites_canton.
```

```
TOPIC Limites_nationales =
```

```
TABLE Partie_limite_nationale =
  Geometrie: POLYLINE WITH (STRAIGHTS, ARCS) VERTEX CoordP BASE
  // Geometrie uniquement Point_limite_ter //;
  Valide: (
    en_vigueur, !! limite exacte de la mensuration officielle
    litigieux, !! limite litigieuse
    provisoire, !! limite definitive mais qualitativement insuffisante
    indefini); !! par ex. jonction inconnue dans un lac
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  NO IDENT
  END Partie_limite_nationale;

END Limites_nationales.
```

```
TOPIC Repartitions_plans =
```

```
TABLE Mise_a_jourRP = !!VD pour traitement mutations
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !! par ex. numero du dossier technique
  Description: TEXT*30;
  Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.050;
  Valide: Statut;
```

```
En_vigueur: OPTIONAL DATE;
Date1: OPTIONAL DATE; !! par ex. date de mise en service
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Mise_a_jourRP;

TABLE Plan =
  Origine: -> Mise_a_jourRP; !!VD pour traitement mutations
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Numero: TEXT*12;
  DossierTech: TEXT*12;
  En_vigueur: OPTIONAL DATE;
  Approbation_DM: OPTIONAL DATE; !!VD
  CodePlan: TEXT*9; !!VD code decrivant les methodes d'acquisition
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT IdentDN, Numero;
END Plan;

TABLE Geometrie_plan =
  Geometrie_plan_de: -> Plan; !! relation 1-m
  Geometrie: AREA WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.050
    LINEATTR =
      UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    END;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Geometrie_plan;

TABLE Posplan =
  Posplan_de: -> Plan; !! relation 1-mc; inscription de Numero
  Pos: CoordP // Pos dans Geometrie_plan //;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Center //;
  VALi: OPTIONAL VALIGNMENT // indefini = Half //;
  Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Posplan;

END Repartitions_plans.
```

TOPIC RepartitionNT =

```
TABLE Niveau_tolerance =
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !! par ex. numero du dossier technique
  Geometrie: AREA WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.200
    LINEATTR =
      UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    END;
  En_vigueur: OPTIONAL DATE;
  Genre: (
    NT1,
    NT2,
    NT3,
    NT4,
    NT5);
```

```
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Niveau_tolerance;
```

```
TABLE PosNiveau_tolerance =
  PosNiveau_tolerance_de: -> Niveau_tolerance; !! relation 1-mc;
  !! inscription de Genre
  Pos: CoordP // Pos dans Niveau_tolerance //;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Center //;
  VALi: OPTIONAL VALIGNMENT // indefini = Half //;
  Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosNiveau_tolerance;
```

```
END RepartitionNT.
```

```
TOPIC Zones_glissement =
```

```
TABLE Glissement =
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  Identification: TEXT*12; !! par ex. numero du dossier technique
  Nom: OPTIONAL TEXT*30; !! Nom particulier
  Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.200;
  En_vigueur: OPTIONAL DATE;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Glissement;
```

```
TABLE PosGlissement =
  PosGlissement_de: -> Glissement; !! relation 1-mc; inscription de Nom
  Pos: CoordP // Pos dans Glissement //;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Center //;
  VALi: OPTIONAL VALIGNMENT // indefini = Half //;
  Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosGlissement;
```

```
END Zones_glissement.
```

```
TOPIC NPA_Localite = !! voir norme SN 612040;
  !! Localites sous la responsabilite des cantons
  !! NPA sous la responsabilite de la Poste
```

```
TABLE Mise_a_jourLoc = !! Norme SN = ChoseActualisable
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  !! MD01: en plus de SN
  Identification: TEXT*12; !! par ex. numero du dossier technique
  !! MD01: en plus de SN
  Description: TEXT*30;
  Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
    WITHOUT OVERLAPS > 0.050; !! MD01: en plus de SN
  Valide: Statut;
```

```

    En_vigueur: DATE;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT IdentDN, Identification;
    END Mise_a_jourLoc;

TABLE Groupement_de_Localite = !! Norme SN 612040: GroupementLocalite
    Vide: OPTIONAL TEXT*1; !! Cet attribut est neceessaire uniquement
    !! pour respecter les regles syntaxiques d'INTERLIS
1
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    NO IDENT
    END Groupement_de_Localite;

TABLE Texte_Groupement_de_Localite =
    Texte_Groupement_de_Localite_de: -> Groupement_de_Localite;!! relation 1-m
    Texte: TEXT*200;
    Langue: TypeLangue;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT Texte_Groupement_de_Localite_de, Langue;
    END Texte_Groupement_de_Localite;

!! Les localites reelles forment une AREA
TABLE Localite =
    Origine: -> Mise_a_jourLoc; !! relation 1-mc
    Localite_de: OPTIONAL -> Groupement_de_Localite; !!relation c-m
    Validite: Statut_mise_a_jour_AB;
    En_cours_modification: (oui, non);
    Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
        WITHOUT OVERLAPS > 0.500;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    NO IDENT
    END Localite;

!! Exemple pour Texte, Texte_abrege et Texte_index:
!! Texte (Nom complet): La Chaux-de-Fonds
!! Texte_abrege (ecriture presente sur une adresse postale): La Chx-de-Fds
!! Texte_index (pour index): Chaux-de-Fonds
TABLE Nom_localite =
    Nom_localite_de: -> Localite; !! relation 1-m
    Texte: TEXT*40;
    Texte_abrege: OPTIONAL TEXT*18;
    Texte_index: OPTIONAL TEXT*16;
    Langue: TypeLangue;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT Nom_localite_de, Langue;
    END Nom_localite;

TABLE PosNom_localite = !! MD01: en plus de SN
    PosNom_localite_de: -> Nom_localite; !! relation 1-mc; inscription de
Texte
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 100.0 //;
    HAli: OPTIONAL HALIGNMENT // indefini = Center //;
    VAlI: OPTIONAL VALIGNMENT // indefini = Half //;
    Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    NO IDENT
    END PosNom_localite;

TABLE Mise_a_jourNPA6 = !! Norme SN = ChoseActualisable
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation

```



```
!! MD01: en plus de SN
Identification: TEXT*12; !! par ex. numero du dossier technique
!! MD01: en plus de SN
Description: TEXT*30;
Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
WITHOUT OVERLAPS > 0.050; !! MD01: en plus de SN
Validite: Statut;
En_vigueur: DATE;
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Mise_a_jourNPA6;

!! Il faut eviter d'avoir un NPA6 avec Validite=reel sur une localite avec
!! une Validite<>reel.
TABLE NPA6 =
  Origine: -> Mise_a_jourNPA6; !! relation 1-mc
  NPA6_de: -> Localite; !! relation 1-m
  !! Si plusieurs codes postaux a six chiffres sont affectes a une meme
  !! localite, une surface doit etre affectee a chacun d'entre eux et
  !! doit etre incluse en totalite dans les limites de la surface de
  !! la localite.
  !! Les NPA6 reels sont de type AREA
  Geometrie: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
  WITHOUT OVERLAPS > 0.500;
  Validite: Statut_mise_a_jour_AB;
  En_cours_modification: (oui, non);
  NPA: [1000 .. 9999];
  Chiffres_supplementaires: [0 .. 99];
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT NPA, Chiffres_supplementaires;
  END NPA6;

END NPA_Localite.

TOPIC Adresses_des_batiments = !! voir SN 612040;
!! Responsabilite des communes

!! Relation geometrique entre Entree_batiment et NPA6
!! Relation geometrique entre Entree_batiment et Localite
!! Relation geometrique entre Localisation et Groupement_de_Localite

TABLE Mise_a_jourBAT = !! Norme SN = ChoseActualisable
  IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
  !! MD01: en plus de SN
  Identification: TEXT*12; !! par ex. numero du dossier technique
  !! MD01: en plus de SN
  Description: TEXT*30;
  Perimetre: OPTIONAL SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
  WITHOUT OVERLAPS > 0.050; !! MD01: en plus de SN
  Validite: Statut;
  En_vigueur: DATE;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
  IDENT IdentDN, Identification;
  END Mise_a_jourBAT;

TABLE Localisation =
  Origine: -> Mise_a_jourBAT; !! relation 1-mc
  Principe_numerotation: (
    aucun_numero,
    quelconque,
```

```

        croissant,
        impair_a_gauche,
        pair_a_gauche);
Numero_localisation: OPTIONAL TEXT*12;
Attributs_provisoires: (oui, non);
Est_designation_officielle: (oui, non);
Validite: Statut_mise_a_jour_AB;
En_cours_modification: (oui, non);
Genre: (
    Lieu_denomme,
    Rue,
    Place);
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Localisation;

!! Exemple pour Texte, Texte_abrege et Texte_index:
!! Texte (Nom complet): Conrad-Ferdinand-Meyer-Strasse
!! Texte_abrege (écriture présente sur une adresse postale): CF Meyer Str
!! Texte_index (pour index): Meyer CF Str
TABLE Nom_localisation =
    Nom_localisation_de: -> Localisation; !! relation 1-m
    Texte: TEXT*60;
    Texte_abrege: OPTIONAL TEXT*24;
    Texte_index: OPTIONAL TEXT*16;
    Langue: TypeLangue;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT Nom_localisation_de, Langue;
END Nom_localisation;

TABLE PosNom_localisation = !! MD01: en plus de SN
    PosNom_localisation_de: -> Nom_localisation; !! relation 1-mc;
    !! inscription de Texte
    Indice_deb: OPTIONAL [1 .. 60] // indefini = 1 //;
    Indice_fin: OPTIONAL [1 .. 60] // indefini = dernier caractere //;
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 100.0 //;
    HALi: OPTIONAL HALIGNMENT // indefini = Center //;
    VALi: OPTIONAL VALIGNMENT // indefini = Half //;
    Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
    !! Trait de rappel
    Ligne_auxiliaire: OPTIONAL POLYLINE WITH (STRAIGHTS) VERTEX CoordP;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosNom_localisation;

TABLE Lieu_denomme =
    Lieu_denomme_de: -> Localisation // Genre = Lieu_denomme //;
    !! relation 1-mc
    Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
        WITHOUT OVERLAPS > 0.500;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Lieu_denomme;

!! La geometrie correspond a l'axe de la rue,
!! voir explication chap.3.18.2
!! Troncons de rue principaux. Les acces prives ne sont pas saisis
!! dans le modele federal
TABLE Troncon_rue =
    Troncon_rue_de: -> Localisation // Genre = Rue ou Place //;

```

```

!! relation 1-mc
Geometrie: POLYLINE WITH (STRAIGHTS, ARCS) VERTEX CoordP;
!! Au lieu de DIRECTED POLYLINE, Point_depart fixe la direction
Point_depart: OPTIONAL CoordP; !! MD01: en plus de SN
!! Au lieu de ORDERED Troncon_rue
Ordre: [1 .. 999]; !! Sequence de troncons de rue
Est_axe: (oui, non);
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT Troncon_rue_de, Ordre;
END Troncon_rue;

!! Aussi pour batiment projete
TABLE Entree_batiment =
  Origine: -> Mise_a_jourBAT; !! relation 1-mc
  Entree_batiment_de: OPTIONAL -> Localisation;
  !! relation c-mc
  Validite: Statut_mise_a_jour_AB;
  En_cours_modification: (oui, non);
  Attributs_provisoires: (oui, non);
  Est_designation_officielle: (oui, non);
  Pos: CoordP
  // Pos a l'int. de CS.Batiment, elements OD (bat souterrain, etc)//;
  !! Niveau est necessaire lorsque plusieurs entrees se trouvent a
différents
  !! niveaux. Approximativement niveau au-dessus du sol
  Niveau: OPTIONAL [-99 .. 99]; !! [m]
  !! Le numero de maison est constitue d'un numero,
  !! lequel peut-etre accompagne d'une lettre a, b, c.
  !! Entre le numero et la lettre, pas de blanc, de souligne ou de
  !! trait d'union.
  !! Lorsque le numero de maison est defini, alors pour Localisation et
  !! Entree_batiment:
  !! - la localisation et le numero doivent etre ensemble unique
  !!   pour une validite = reel
  !! - le principe de numerotation ne doit pas etre aucun_numero.
  Numero_maison: OPTIONAL TEXT*12; !! z.B. Numero de police
  !! Dans_batiment est utile pour definir si le numero est rattache a un
  !! objet de la CS ou a un objet divers.
  Dans_batiment: (CS, OD); !! MD01: en plus de SN
  !! Identificateur du batiment du RegBL, lorsque disponible,
  !! voir explication chap. 3.18.2
  RegBL_EGID: OPTIONAL [1..999999999];
  !! Identificateur de l'entree du batiment du RegBL, lorsque disponible,
  !! voir explication chap. 3.18.2
  RegBL_EDID: OPTIONAL [0..99]; !! MD01: en plus de SN
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Entree_batiment;

TABLE PosNumero_maison = !! MD01: en plus de SN
  PosNumero_batiment_de: -> Entree_batiment; !! relation 1-mc
  Pos: CoordP;
  Ori: OPTIONAL Rotation // indefini = 100.0 //;
  HALi: OPTIONAL HALIGNMENT // indefini = Center //;
  VALi: OPTIONAL VALIGNMENT // indefini = Half //;
  Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
  UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosNumero_maison;

TABLE Nom_batiment =

```

```

    Nom_batiment_de: -> Entree_batiment; !! relation 1-mc
    Texte: TEXT*40;
    Texte_abrege: OPTIONAL TEXT*24;
    Texte_index: OPTIONAL TEXT*16;
    Langue: TypeLangue;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT Nom_batiment_de, Langue;
    END Nom_batiment;

```

```

TABLE PosNom_batiment = !! MD01: en plus de SN
    PosNom_batiment_de: -> Nom_batiment; !! relation 1-m
        !! inscription de Texte
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 100.0 //;
    HALi: OPTIONAL HALIGNMENT // indefini = Center //;
    VALi: OPTIONAL VALIGNMENT // indefini = Half //;
    Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
    !! Trait de rappel
    Ligne_auxiliaire: OPTIONAL POLYLINE WITH (STRAIGHTS) VERTEX CoordP;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    NO IDENT
    END PosNom_batiment;

```

```

TABLE Description_batiment =
    Description_batiment_de: -> Entree_batiment; !! relation 1-mc
    Texte: TEXT*100;
    Langue: TypeLangue;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
    IDENT Description_batiment_de, Langue;
    END Description_batiment;

```

END Adresses_des_batiments.

TOPIC Bords_de_plan =
 !! Les objets mentionnes dans l'Ordonnance technique sur la mensuration
 !! officielle doivent etre geres.

DOMAIN

Type_echelle = [1 .. 1000000];

```

Genre_description = (
    voisins,                !! commune, district, canton ou pays
    plan_voisin,            !! plans voisins en situation
    plan_synoptique,        !! plans voisins, commune, district, canton ou
    pays                    !! dans un plan synoptique
    direction_route,
    no_CN,
    noOFS,
    autre (
        cartouche (         !!VD afin de peupler le cartouche
            commune,
            no_plan,
            code_plan),
        autre));
    uniquement pour extensions
    !! Pas d'objet dans la categorie autre,

```

Type_ligne = (

```

        standard,
        autre); !! Pas d'objet dans la categorie autre, uniquement pour
extensions

        Genre_symbole = (
            flecheNord,
            autre); !! Pas d'objet dans la categorie autre, uniquement pour
extensions

        Genre_croix = (
            croix_coord,
            croix_filet,
            marque_filet,
            autre); !! Pas d'objet dans la categorie autre,
uniquement pour extensions

        Genre_format = ( !!VD afin de permettre la production de plusieurs
formats
            plan_entier_paysage,
            plan_entier_portrait,
            demi_plan_paysage,
            demi_plan_portrait);

TABLE Bord_de_plan =
    IdentDN: TEXT*12; !! relation 1-m avec Domaine_numerotation
    Identification: TEXT*32;
    Type_bord_de_plan: TEXT*20; !! definition du type de bord de plan
    Numero_du_plan: TEXT*12;
    Nom_commune: TEXT*30;
    Nom_geometre: OPTIONAL TEXT*30;
    Date_etablissement: DATE;
    Nom_geometre_conservateur: OPTIONAL TEXT*30;
    Date_MAJ: OPTIONAL DATE;
    Nombre_echelle: Type_echelle;
    Origine_plan: CoordP;
    E_Azimet: Rotation; !! Azimet 100 est E
    Nombre_echelle_plan_synoptique: OPTIONAL Type_echelle;
    Origine_plan_synoptique: OPTIONAL CoordP;
    Avec_reseau_coord: (
        oui, !! livre avec
        non); !! a generer
    Format_plan: Genre_format; !!VD
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT IdentDN, Identification;
END Bord_de_plan;

TABLE Description_plan =
    Origine: -> Bord_de_plan; !! relation 1-mc
    Description: TEXT*30;
    Genre: Genre_description;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Description_plan;

TABLE PosDescription_plan =
    PosDescription_plan_de: -> Description_plan; !! relation 1-m;
    !! inscription de Description
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 100.0 //;
    HALi: OPTIONAL HALIGNMENT // indefini = Center //;

```

```
    VALi: OPTIONAL VALIGNMENT // indefini = Half //;
    Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END PosDescription_plan;

TABLE Indication_coordonnees =
    Indication_coordonnees_de: -> Bord_de_plan; !! relation 1-mc
    Description: TEXT*12;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Indication_coordonnees;

TABLE PosIndication_coord =
    PosIndication_coord_de: -> Indication_coordonnees; !! relation 1-m;
                        !! inscription de Description

    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 100.0 //;
    HALi: OPTIONAL HALIGNMENT // indefini = Center //;
    VALi: OPTIONAL VALIGNMENT // indefini = Half //;
    Grandeur: OPTIONAL GrandeurEcriture // indefini = moyenne //;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT PosIndication_coord_de, Pos;
END PosIndication_coord;

TABLE Element_lineaire =
    Element_lineaire_de: -> Bord_de_plan; !! relation 1-mc
    Geometrie: POLYLINE WITH (STRAIGHTS, ARCS) VERTEX CoordP;
    Genre: Type_ligne;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Element_lineaire;

TABLE Ligne_coordonnees =
    Ligne_coordonnees_de: -> Bord_de_plan; !! relation 1-mc
    Geometrie: POLYLINE WITH (STRAIGHTS) VERTEX CoordP;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Ligne_coordonnees;

TABLE Surface_representation =
    Surface_representation_de: -> Bord_de_plan; !! relation 1-mc
    Geometrie: SURFACE WITH (STRAIGHTS, ARCS) VERTEX CoordP
        WITHOUT OVERLAPS > 0.050;
    Choix_representation: (
        completement_represente,
        partiellement_represente);
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END Surface_representation;

TABLE SymboleBord_de_plan =
    SymboleBord_de_plan_de: -> Bord_de_plan; !! relation 1-mc
    Pos: CoordP;
    Ori: OPTIONAL Rotation // indefini = 0.0 //;
    Genre: Genre_symbole;
    UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
NO IDENT
END SymboleBord_de_plan;

TABLE Croix_filet =
```

```
Croix_filet_de: -> Bord_de_plan; !! relation 1-mc
Pos: CoordP;
Ori: OPTIONAL Rotation // indefini = 0.0 //;
Genre: Genre_croix;
UFID: OPTIONAL IdUnique; !!VD pour traitement mutations
IDENT Croix_filet_de, Pos;
END Croix_filet;
```

```
END Bords_de_plan.
```

```
END MD01MOVD.
```

```
FORMAT FREE;
!!FORMAT FIX WITH LINESIZE = 107, TIDSIZE = 16;
```

```
CODE
  BLANK = DEFAULT, UNDEFINED = DEFAULT, CONTINUE = DEFAULT;
  TID = ANY;
END.
```