

**<CDM-fr>**

## **Règles d'usages de la recommandation CDM-fr**

**Spécification des métadonnées de description des formations et des cours**

**Version 0.4 du 07/10/2005**

### **Avertissement**

**Ce document est, par nature, en évolution constante puisqu'il intègre au fur et à mesure les ajouts proposés par les utilisateurs qui déploient CDM-fr. A ce titre vous êtes invités à faire parvenir vos remarques par email à [francois.ledoux@education.gouv.fr](mailto:francois.ledoux@education.gouv.fr) qui les proposera pour étude et intégration au comité de suivi.**

## Table des matières

1	Principes généraux d'application .....	3
2	Adaptabilité aux besoins des établissements .....	3
2.1	Concepts principaux .....	3
2.2	Identification et référencement d'éléments.....	3
2.3	Enumérations .....	4
2.4	Considérations linguistiques.....	4
2.5	Mise en forme des textes (infoBlockType) .....	4
3	Description des principaux concepts .....	5
3.1	<CDM>Conteneur de l'offre de formation</CDM> .....	5
3.2	<orgUnit>Entité organisationnelle</orgUnit> .....	5
3.2.1	Entités de formation .....	5
3.2.2	Entités de recherche associées aux formations.....	6
3.3	<program>Programme d'études</program>.....	6
3.3.1	Description de l'organisation d'une formation diplômante ou non .....	6
3.3.2	Découpage temporel .....	7
3.3.3	Découpage en parcours.....	8
3.3.4	Découpage par capitalisation .....	9
3.3.5	Pourquoi utiliser des SubProgram ?.....	10
3.3.6	Traitement des données d'habilitations.....	10
3.3.7	<course>Unité d'enseignement</course> .....	11
3.4	<person>Personne</person>.....	12
3.5	Supports de concepts généraux.....	12
3.5.1	Bloc d'informations .....	12
3.6	Formation continue versus formation initiale.....	12

## 1 Principes généraux d'application

Ce document décrit les règles d'usages de la recommandation qui permet d'appliquer le format CDM-fr aux différentes formations de l'enseignement supérieur français, qu'elles soient conformes au modèle LMD ou non et, qu'elles soient offertes en formation initiale ou continue. Le présent format vise à renforcer la mobilité européenne des étudiants dans le cadre du processus de Bologne en appliquant un cadre commun et inter-opérable à la description de l'offre de formations des Universités européennes.

Ce document a une valeur de préconisation ; les règles énoncées pourront évoluer en fonction des observations et remarques des établissements ou de la tutelle. Ce document complète et précise les modalités d'application du document de référence CDM traduit, dans le contexte français de l'enseignement supérieur.

## 2 Adaptabilité aux besoins des établissements

Compte tenu du fait que CDM offre une grande souplesse et que la description de l'offre de formation varie fortement selon les établissements français, la recommandation CDM-fr propose des règles d'usages communes et des choix parmi les possibilités offertes par le schéma initial. Ces règles et ces choix contribuent à la définition de protocoles stables, cohérents et partagés par tous les partenaires afin de permettre l'échange d'informations entre les établissements d'enseignement supérieur français et leur visibilité potentielle à un niveau national ou européen.

### 2.1 Concepts principaux

Lors de la description des formations il est important de s'accorder sur l'interprétation des principaux concepts et le sens exact des termes employés. C'est-à-dire s'accorder sur la signification des différents éléments qui composent CDM-fr et sur la façon de les remplir :

- **OrgUnit** : entité organisationnelle. (*définition dans le cahier de recommandations*)
- **Program** : programme d'études dispensé par un établissement. (*définition dans le cahier de recommandations*)
- **Course** : ensemble de cours (organisé en unité d'enseignement (UE) ou module ou cours (*définition dans le cahier de recommandations*))
- **Person** : Personnes impliquées dans les activités administratives ou pédagogiques relatives aux programmes d'études et aux cours. (*définition dans le cahier de recommandations*)

### 2.2 Identification et référencement d'éléments

Dans la description des programmes d'études ou des unités d'enseignement, une même information peut s'avérer pertinente à différents endroits. Par exemple, une même unité d'enseignement peut faire partie de plusieurs programmes d'études, les coordonnées d'une même entité organisationnelle ou d'une même personne peuvent correspondre à des unités d'enseignement différentes. Dans ces cas là, il est possible de dupliquer l'information commune, ou encore de mettre en place un mécanisme d'identification et de référencement des informations pour éviter la duplication.

Le schéma XML propose pour chacun des éléments deux attributs pour l'identification et le référencement d'éléments : *id* de type *xs.ID* et *ident* de type *xs.token* La préconisation CDM-fr est :

- *ident* doit être utilisé comme identifiant des éléments *orgUnit*, *program*, *course*, *person*.
- *refprogram*, *refcourse*, *refperson*, *reforgUnit*, doivent être utilisés en positionnant l'identifiant de l'élément référencé dans l'attribut *ref*.
- L'identifiant est l'identifiant interne unique au sein du SI de l'établissement (a priori sans valeur d'échange)

L'utilisation de *id* a été écartée pour éviter les erreurs de validation XML susceptibles d'apparaître lorsque des références orphelines existent dans le document échangé.

## 2.3 Enumérations

Pour le moment (CDM V2.2) les listes fermées contiennent des valeurs dans la langue d'origine du schéma (norvégien). Pour s'accommoder de cette situation CDM-fr préconise :

- L'utilisation d'un dictionnaire pour proposer dans les interfaces la traduction française des valeurs du schéma
- L'utilisation de listes en français dans le schéma CDM-fr dès que les instances de normalisation l'autoriseront.

## 2.4 Considérations linguistiques

CDM-fr préconise l'utilisation de l'attribut *language* (*fr-FR*) de l'élément CDM pour décrire la langue par défaut de l'élément CDM, c'est-à-dire la langue par défaut de la description de l'offre de formation.

L'affichage multilingue de certains des éléments CDM utilise :

- Pour les éléments de type *textType* l'attribut *language*  

```
<courseName>  
  <text >Modélisation de concept utilisant un schéma XML </text>  
  <text language="en-GB">Concept modelling using XML Schema</text>  
</courseName>
```
- Pour les éléments de type *infoBlockType* l'attribut *blockLang* de l'élément *altLangBlock*

Les attributs *language*, *blockLang*,..., suivent la norme *xs:language* de W3c

L'élément *altLangBlock* doit impérativement figurer dans l'élément de type *infoBlockType* de plus haut niveau possible, aucun de ses éléments fils ne contient d'élément *altLangBlock*. Néanmoins le traitement du multilinguisme implique des chemins d'accès pouvant être différents selon qu'il s'agit d'atteindre les éléments de la langue par défaut ou des éléments dans une autre langue. La réflexion reste ouverte sur ce point.

Voir la préconisation de l'usage du type *infoBlockType* dans le paragraphe (ci-dessous) sur la mise en forme des textes.

## 2.5 Mise en forme des textes (infoBlockType)

Le type *infoBlockType* permet d'insérer du texte libre. Pour structurer ou formater le texte il est nécessaire de faire appel :

- aux éléments de formatage (*italic*, *bold*) qui ne peuvent pas être combinés ou inclus l'un dans l'autre.
- aux éléments de structuration et d'enrichissement (*header*, *subBlock*, *list*, *picture*).

Pour réaliser par exemple un texte composé de paragraphes contenant chacun un titre, un infoBlock va contenir des subBlocks dont le contenu des éléments headers sera le titre du paragraphe.

```
<programDescription>
  <header>titre</header>
  Ceci est un texte de démonstration <italic>en italiques </italic>ou alors
<bold>en gras</bold> pour démontrer l'utilisation d'un élément de type infoBlockType
  <subBlock>
    <header>sous titre</header>
    Texte du paragraphe
  </subBlock>
</programDescription>
```

L'infoBlockType a d'autres fonctionnalités qui seront décrites dans leur contexte d'utilisation.

### 3 Description des principaux concepts

#### 3.1 <CDM>Conteneur de l'offre de formation</CDM>

La présence de cet élément est impérative même pour décrire un sous élément unique.

#### 3.2 <orgUnit>Entité organisationnelle</orgUnit>

##### 3.2.1 Entités de formation

- L'organisme responsable de l'offre de formation est décrit dans le premier élément *orgUnit*
- La tutelle est indiquée dans l'attribut *responsibleBody* de l'élément *orgUnitKind*. CDM-fr préconise de n'utiliser cet attribut que pour l'organisme responsable de la formation.
- Pour décrire de façon explicite une hiérarchie d'entités organisationnelles, CDM-fr préconise d'utiliser un emboîtement d'éléments *orgUnit*. Il est toutefois possible d'exploiter la souplesse des éléments *orgUnitDescription* et *refOrgUnit* pour présenter de façon hiérarchisée des entités organisationnelles qui seraient décrites au même niveau.

```
<OrgUnit> organisme
  <orgUnitDescription>
    <refOrgUnit ref="id1"> </refOrgUnit>
    <refOrgUnit ref="id2"> </refOrgUnit>
  </orgUnitDescription>
  <orgUnit ident="id2"> ...
    <orgUnitDescription>
      <refOrgUnit ref="..."> </refOrgUnit>
    </orgUnitDescription>
  </orgUnit>
  <refOrgUnit ref="id3"> </refOrgUnit>
</OrgUnit>
<OrgUnit ident="id1"> ... </orgUnit>
<OrgUnit ident="id3"> ... </orgUnit>
```

Si un organisme souhaite présenter de façon structurée et commentée son organisation et les entités qui la composent, CDM-fr préconise l'exploitation des possibilités textuelles des éléments *subBlock (header et list)* de l'élément *orgUnitDescription*.

Par exemple :

- -liste des UFR
- -liste des IUT
- -liste des écoles
- -liste des établissements associés

Tant que nous n'avons pas de vocabulaire français dans CDM-fr validé pour *orgType* de l'élément *orgUnitKind* :

CDM-fr préconise de mettre dans l'attribut *userDefined* de l'élément *orgUnitKind* le libellé du type d'établissement ou du type de composante.

CDM-fr préconise d'utiliser exclusivement l'élément *refProgram* pour associer un élément *program* à un élément *orgUnit*.

CDM-fr préconise d'utiliser exclusivement l'élément *refCourse* pour associer un élément *course* à un élément *orgUnit* si nécessaire.

CDM-fr préconise d'utiliser exclusivement l'élément *refPerson* pour associer un élément *person* à un élément *orgUnit* si nécessaire.

En conclusion dans CDM-fr l'élément *orgUnit* ne contient aucun des éléments suivants : *program, course, person*.

### 3.2.2 Entités de recherche associées aux formations

CDM-fr préconise l'utilisation de l'élément *orgUnit* pour décrire les structures de recherche.

, Lorsque *orgUnit* décrit une structure de recherche, les formations qui lui sont associées, par exemple un master, seront reliées à l'aide d'éléments *refProgram*.

Réciproquement CDM-fr recommande d'utiliser l'élément *refOrgUnit* d'un *subBlock* de *programDescription* pour relier le master à la structure de recherche qui lui est associée.

CDM-fr préconise d'utiliser l'élément *orgUnit* pour décrire les écoles doctorales ; les laboratoires de recherche de l'école doctorale sont décrits en utilisant une structure d'emboîtement d'éléments *orgUnit* identique à celui qui est proposé pour les composantes d'enseignement.

## 3.3 <program>Programme d'études</program>

Travail en cours : Organisation de l'élément *program*

### 3.3.1 Description de l'organisation d'une formation diplômante ou non

En France, dans le cadre du LMD et hors LMD, co-existent deux modes de description de l'organisation des formations : une organisation en semestres et années et/ou une organisation en parcours.

L'élément *program* de CDM a été conçu pour décrire un modèle d'enseignement basé sur la capitalisation de modules.

L'élément *programStructure* de l'élément *program* est utilisé pour décrire l'organisation d'une formation, soit selon un axe temporel (en semestres ou années), soit en terme de parcours, soit en terme de modules capitalisables. Il référence les différentes possibilités d'achever la formation considérée.

L'élément *programStructure* ne contient que des éléments de type *refProgram* ou des éléments textuels sauf s'il appartient à un élément *program* ou *subprogram* de dernier niveau ; dans ce cas il contient des *subBlock* regroupant un header et la liste des *refCourse*. La structuration en *subBlock* au dernier niveau permet de traiter les groupes d'UE par options ou par toute autre propriété de regroupement. Comme CDM-fr préconise d'utiliser des *refCourse*, une même UE peut apparaître dans plusieurs regroupements. Bien que le nom de l'élément soit *refProgram* celui-ci peut référencer aussi bien un *program* qu'un *subProgram*. Le *subProgram* est nécessairement un élément fils d'un élément *program*. CDM-fr préconise d'utiliser un élément header de l'élément *programStructure* pour déclarer le mode d'organisation de la structure décrite .

### 3.3.2 Découpage temporel

CDM-fr préconise l'utilisation d'un élément *subProgram* pour présenter une partie d'une formation correspondant à une unité de temps (année, semestre). La référence à ce *subprogram* sera passée à l'aide d'un élément *refProgram* dans l'élément *programStructure* de la formation

**Avertissement : l'exemple ci dessous est susceptible d'évoluer rapidement**

```
<program >
  <programName>
    <text/>
  </programName>
  <programStructure>
    <header>Master</header>
    <refProgram ref="M"/>
  </programStructure>
  <subProgram ident="M">
    <programName>
      <text/>
    </programName>
    <programStructure>
      <header>découpage par années du Master</header>
      <refProgram ref="an1"/>
      <refProgram ref="an2"/>
    </programStructure>
  </subProgram>
  <subProgram ident="an2">
    <programName>
      <text/>
    </programName>
    <programStructure>
      <header>découpage par semestres de l'année 2</header>
      <refProgram ref="s1"/>
      <refProgram ref="s2"/>
    </programStructure>
  </subProgram>
  <subProgram ident="s1">
    <programName>
      <text/>
    </programName>
    <programStructure>
      <subBlock>
        <header>UE obligatoires</header>
        <refCourse/>
      </subBlock>
    </programStructure>
  </subProgram>
</program >
```

```
        <refCourse/>
    </subBlock>
    <subBlock>
        <header>UE optionnelles</header>
        <refCourse/>
        <refCourse/>
    </subBlock>
</programStructure>
</subProgram>
<subProgram ident="s2">
    <programName>
        <text/>
    </programName>
    <programStructure>
        <subBlock>
            <header>UE obligatoires</header>
            <refCourse/>
            <refCourse/>
        </subBlock>
        <subBlock>
            <header>UE optionnelles</header>
            <refCourse/>
            <refCourse/>
        </subBlock>
    </programStructure>
</subProgram>
<subProgram ident="an1">
    <programName>
        <text/>
    </programName>
    <programStructure>
        <subBlock>
            <header>UE obligatoires</header>
            <refCourse/>
            <refCourse/>
        </subBlock>
        <subBlock>
            <header>UE optionnelles</header>
            <refCourse/>
            <refCourse/>
        </subBlock>
    </programStructure>
</subProgram>
```

### 3.3.3 Découpage en parcours

Dans ce cas là, on fait le choix de décrire la formation en parcours uniquement. Chaque parcours est représenté par un subprogram de premier niveau.

Dans les header noter des indications d'enchaînement.

```
<program >
    <programName>
        <text/>
    </programName>
```

```
<programStructure>
  <refProgram ref="parcours1"/>
  <refProgram ref="parcours2"/>
</programStructure>
<subProgram ident="parcours1">
  <programName>
    <text/>
  </programName>
  <programStructure>
    <subBlock>
      <header>UE obligatoires</header>
      <refCourse/>
      <refCourse/>
    </subBlock>
    <subBlock>
      <header>UE optionnelles</header>
      <refCourse/>
      <refCourse/>
    </subBlock>
  </programStructure>
</subProgram>
<subProgram ident="parcours2">
  <programName>
    <text/>
  </programName>
  <programStructure>
    <subBlock>
      <header>UE obligatoires</header>
      <refCourse/>
      <refCourse/>
    </subBlock>
    <subBlock>
      <header>UE optionnelles</header>
      <refCourse/>
      <refCourse/>
    </subBlock>
  </programStructure>
</subProgram>
</program>
```

### 3.3.4 Découpage par capitalisation

**Avertissement : l'exemple ci dessous est susceptible d'évoluer rapidement**

```
<program >
  <programName>
    <text/>
  </programName>
  <programStructure>
    <refProgram ref="parcours1"/>
    <refProgram ref="parcours2"/>
  </programStructure>
  <subProgram ident="parcours1">
    <programName>
```

```
        <text/>
    </programName>
    <programStructure>
        <subBlock>
            <header>UE obligatoires</header>
            <refCourse/>
            <refCourse/>
        </subBlock>
        <subBlock>
            <header>UE optionnelles</header>
            <refCourse/>
            <refCourse/>
        </subBlock>
    </programStructure>
</subProgram>
<subProgram ident="parcours2">
    <programName>
        <text/>
    </programName>
    <programStructure>
        <subBlock>
            <header>UE obligatoires</header>
            <refCourse/>
            <refCourse/>
        </subBlock>
        <subBlock>
            <header>UE optionnelles</header>
            <refCourse/>
            <refCourse/>
        </subBlock>
    </programStructure>
</subProgram>
</program>
```

### 3.3.5 Pourquoi utiliser des SubProgram ?

- permet de préciser à un niveau inférieur le contenu des éléments de description : objectifs pédagogiques, qualification, localisation des enseignements, responsables etc. qui peuvent être différents d'un parcours à l'autre
- permet pour les M1 et M2 de préciser des conditions d'admission différentes,
- permet pour les masters à finalités professionnelles de préciser le nombre de places
- permet pour les masters à finalités recherche de préciser les critères pédagogiques d'admission
- permet d'isoler un parcours ou une année puisque chaque subProgram a une référence et un type
- permet d'afficher uniquement un parcours ou une année donnée correspondant à des critères de recherche ....

### 3.3.6 Traitement des données d'habilitations

Ces informations sont traitées dans l'élément programDescription de l'élément program. Elles sont toutes contenues dans un subBlock dont l'attribut userDefined porte la valeur « DES » et un header contenant le texte « Information sur les habilitations ». Ces informations concernent obligatoirement les formations « habilitées », ces informations ont une forme qui doit être respectée .

Ce subBlock est décomposé en 4 subBlock :

- Header= Domaine mention spécialité. Décomposé lui-même en 4 subBlock :
  - Header = Domaine
  - Header = Mention
  - Header= Spécialité
  - Header = Numéro d'habilitation
- Header = Finalité
- Header = Etablissements partenaires, ce subBlock contient des refOrgUnit qui référencent les différents établissements partenaires de la formation considérée
- Header = Dates , décomposé en 3 subBlock
  - Header = Date d'habilitation
  - Header = Date d'échéance
  - Header= Durée

### 3.3.7 <course>Unité d'enseignement</course>

#### Principes

- tous les éléments pédagogiques sont traités comme des éléments course qu'ils soient une UE ou bien un EC si la formation est organisée de cette façon.
- il est donc nécessaire dans ce cas de typer les éléments course en UE ou EC pour différencier deux usages du même élément CDM-fr *course* (bien que ce ne soit pas utile pour une simple présentation de la ressource).
- CDM-fr préconise de créer un nouvel élément courseCode français dont l'attribut codeSet est de type courseCodeValueSet et contient une liste de 2 valeurs UE et EC. L'élément courseCode contient le code d'identification d'une UE lorsque le codeSet a la valeur UE ou le code d'identification d'une EC lorsque le codeSet vaut EC.

**Avertissement un schéma CDM-fr v2 (voir le N° de version Norvégien) sera fourni prochainement.**

- CDM-fr préconise de décrire la composition des UE à partir d'EC à l'aide de références. L'élément course ne dispose pas de l'équivalent « program structure », pour faire référence aux EC qui composent une UE, il faut utiliser CourseDescription (de type infoBlock). La technique décrite plus haut pour programStructure s'applique ici de la même façon : c'est-à-dire utiliser les subBlock pour gérer au besoin les notions d'obligatoire, optionnelles etc... Toutefois pour distinguer le subBlock de description de la structure il est nécessaire de le typer. CDM-fr préconise d'utiliser l'attribut userDefined de l'élément subBlock en lui donnant la valeur courseStructure.

#### Exemple

```
<course ident="ue1">
  <courseName>
    <text>UE</text>
  </courseName>
  <courseCode codeSet="UE">MATH007</courseCode>
  <courseDescription>
    <subBlock userDefined="courseStructure">
      <header>éléments constitutifs obligatoires</header>
      <refCourse ref="ec1"/>
      <refCourse ref="ec2"/>
    </subBlock>
    <subBlock userDefined="courseStructure">
      <header>éléments constitutifs optionnels</header>
      <refCourse ref="ec3"/>
      <refCourse ref="ec4"/>
    </subBlock>
  </courseDescription>
</course>
<course ident="ec1">
  <courseName>
    <text>L'EC n°1</text>
  </courseName>
  <coursecode codeSet="EC">MATH009</coursecode>
</course>
<course ident="ec2">
```

```
<courseName>
  <text>L'EC n°2</text>
</courseName>
<coursecode codeSet="EC">MATH010</coursecode>

</course>
</CDM>
```

### 3.4 <person>Personne</person>

Travail en cours

### 3.5 Supports de concepts généraux

Travail en cours

#### 3.5.1 Bloc d'informations

Le bloc d'informations est un concept général pour exprimer de façon flexible des informations complexes et complètes. Le contenu du bloc d'informations varie du simple texte à une structure de chapitres imbriqués (comme dans un livre) avec un texte, des images et des références. Dans le schéma XML, le bloc d'informations est spécifié par le type de données complexe *infoBlockType*. Dans l'élément *infoBlockType*, le texte peut être renseigné librement entre les sous-éléments (mixed = true). Toutefois, CDM-fr préconise de définir tout texte dans un élément subBlock, cet élément représentant la notion de paragraphe de texte. Seul cet élément subBlock contenant du texte pourra être mixé.

### 3.6 Formation continue versus formation initiale

**Avertissement : une demande est en cours auprès de l'équipe norvégienne pour identifier ou se situe la FC et comment ils procèdent pour rentrer les codes**

L'élément Targetgroup recueille à priori ces informations. Il est à noter que les formations sont qualifiées par la demande et non par l'offre en formation continue ou initiale. C'est donc le statut du demandeur qui permet de définir FC/FI.

**Avertissement : un exemple sera fourni rapidement. Il est à vérifier la codification Cise des différents régimes de formations.**