

Library of Congress Network Development and MARC Standards Office

Comprendre PREMIS

Auteur : Priscilla Caplan

Date de publication : 1^{er} février 2009

Copyright © 2009 The Library of Congress, excepté aux États-Unis.
Les auteurs doivent être cités lorsqu'un extrait de cette publication est utilisé.

Traduction : Louise Fauduet, Clément Oury, Sébastien Peyrard, Jean-Philippe Tramoni
Bibliothèque nationale de France

COMPRENDRE PREMIS	3
1. MISE EN CONTEXTE DE PREMIS	3
1.1. QUE SONT LES METADONNEES DE PRESERVATION ?.....	3
1.2. QU'EST-CE QUE PREMIS ?.....	4
1.3. QU'EST-CE QUE LE DICTIONNAIRE DE DONNEES PREMIS ?.....	4
1.4. COMMENT DOIT-ON UTILISER PREMIS ?.....	6
1.5. DEVEZ-VOUS UTILISER PREMIS ?.....	6
2. CONVENTIONS DU DICTIONNAIRE DE DONNEES.....	7
2.1. UNITES SEMANTIQUES	7
2.2. CONTENEURS ET SOUS-UNITES.....	8
2.3. CONTENEURS D'EXTENSION.....	8
3. MODELE DE DONNEES DE PREMIS.....	9
3.1. ENTITE INTELLECTUELLE	9
3.2. ENTITE OBJET	10
3.3. ÉVENEMENTS.....	11
3.4. AGENTS	11
3.5. DROITS	12
4. LE DICTIONNAIRE DE DONNEES.....	13
4.1. EXEMPLE D'ENTREE DU DICTIONNAIRE DE DONNEES POUR UNE UNITE SEMANTIQUE.....	13
4.2. EXEMPLE D'ENTREE DU DICTIONNAIRE DE DONNEES POUR UNE UNITE DE CONTENEUR	14
5. UTILISER PREMIS	15
5.1. PREMIS EN XML	15
5.2. CONFORMITE A PREMIS	16
6. INFORMATIONS COMPLEMENTAIRES.....	17
APPENDICES	18
APPENDICE A : LISTE DE L'ENSEMBLE DES UNITES SEMANTIQUES DE PREMIS	18
APPENDICE B : EXEMPLE D'OBJET.....	22
APPENDICE C : GLOSSAIRE DES TERMES.....	25

COMPRENDRE PREMIS

Ce guide constitue une présentation relativement courte du standard de métadonnées de préservation PREMIS. Il ne vous donnera pas suffisamment d'informations pour utiliser PREMIS, mais donnera une idée de ce dont il s'agit. Ce sera suffisant pour la plupart des lecteurs. Mais pour ceux qui ont besoin de maîtriser le *Dictionnaire de données PREMIS pour les métadonnées de préservation*, ce guide peut servir d'introduction progressive qui vous familiarisera avec le dictionnaire, plus complet.

Au sein de ce guide, les termes en police Verdana sont les noms des unités sémantiques PREMIS. Les termes qui apparaissent dans le Glossaire sont en *italique* lors de leur première apparition.

1. MISE EN CONTEXTE DE PREMIS

1.1. Que sont les métadonnées de préservation ?

Si vous travaillez en bibliothèque ou en archives, vous avez probablement déjà au moins quelques connaissances en matière de métadonnées et de description des ressources. Vous savez sans doute que les métadonnées sont réparties en catégories selon leur fonction : les métadonnées descriptives aident à découvrir et identifier des ressources, les métadonnées administratives aident à les gérer et à les tracer, les métadonnées de structure documentent la façon dont des objets numériques complexes sont agencés, de manière à permettre leur visualisation, ou d'autres utilisations. De même, les *métadonnées de préservation* servent aux activités destinées à garantir l'usage à long terme d'une ressource numérique.

Le Dictionnaire de données PREMIS définit les métadonnées de préservation comme "les informations utilisées par un entrepôt comme fondement du processus de préservation numérique". Voici quelques exemples d'actions de préservation et de façons dont les métadonnées peuvent les faciliter :

- Une ressource doit être stockée de manière sécurisée afin que personne ne puisse la modifier par erreur, ou par malveillance. L'empreinte d'un fichier enregistrée sous forme de métadonnée peut être utilisée pour dire s'il a été modifié entre deux dates.
- Les fichiers doivent être stockés sur des supports qui peuvent être lus par les ordinateurs actuels. Si les supports sont abîmés ou obsolètes (comme les disquettes 8 pouces utilisées dans les années 1970), il peut être difficile, voire impossible, d'en récupérer les données. Les métadonnées peuvent faciliter la gestion des supports de stockage en gardant la trace de leur type et de leur âge, et des dates auxquelles ont eu lieu les derniers rafraîchissements.
- À long terme, même les formats de fichiers les plus courants peuvent devenir obsolètes, c'est-à-dire qu'aucun programme actuel ne peut plus les lire. Les gestionnaires de préservation doivent appliquer des *stratégies de préservation* afin de s'assurer que les ressources restent utilisables. Cela peut se traduire par la transformation de formats anciens en leurs équivalents plus récents (*migration*), ou par l'imitation de l'ancien environnement de lecture sur un matériel et des logiciels plus récents (*émulation*). Les deux stratégies, migration et émulation, nécessitent des métadonnées sur les formats de fichier originaux et sur l'environnement —matériel et programmes — permettant de les lire.

- Les actions de préservation peuvent amener à modifier les ressources originales ou la manière dont elles sont interprétées. Cela peut mettre en cause leur authenticité. Des métadonnées peuvent aider à prouver cette authenticité en documentant la *provenance numérique* de la ressource — l'historique de ses détenteurs et des modifications qui ont été autorisées.

1.2. Qu'est-ce que PREMIS ?

PREMIS est l'acronyme de "PREservation Metadata: Implementation Strategies", ou « stratégies d'implémentation des métadonnées de préservation », un groupe de travail international soutenu par l'OCLC et le RLG de 2003 à 2005. Ce groupe de travail a produit un rapport intitulé *PREMIS Data Dictionary for Preservation Metadata (Dictionnaire de données PREMIS pour les métadonnées de préservation)*, qui inclut à la fois un dictionnaire de données, et un bon nombre de considérations sur les métadonnées de préservation. Une seconde version mise à jour a été publiée en mars 2008. La Bibliothèque du Congrès maintient un schéma XML pour PREMIS.

L'Activité de maintenance de PREMIS est très dynamique, et soutenue par la Bibliothèque du Congrès. Elle inclut un site Web qui fournit des liens vers toutes sortes d'informations officielles et officieuses sur PREMIS, une liste de diffusion et un wiki pour les utilisateurs de PREMIS, ainsi qu'un comité éditorial responsable des révisions du dictionnaire de données et du schéma. L'Activité de maintenance s'efforce aussi de promouvoir PREMIS, d'organiser des tutoriels sur son utilisation, et de commanditer des études et des publications qui y sont liées, comme le présent guide.

Le terme "PREMIS" est généralement utilisé pour désigner le dictionnaire de données. Parfois, il peut désigner le schéma XML, le groupe de travail, ou l'ensemble du projet, y compris l'Activité de maintenance.

Dictionnaire de données PREMIS : www.loc.gov/premis/v2/premis-2-0.pdf

Site Internet de PREMIS : www.loc.gov/standards/premis/

Liste de diffusion du groupe d'utilisateurs de PREMIS : listserv.loc.gov/listarch/pig.html

1.3. Qu'est-ce que le dictionnaire de données PREMIS ?

Le dictionnaire de données PREMIS définit un ensemble d'unités sémantiques (voir section 2.1) qui constituent la base de ce que les entrepôts doivent connaître pour remplir leurs fonctions de préservation. Ces fonctions peuvent varier d'un entrepôt à un autre, mais comprennent généralement des actions destinées à garantir que les objets numériques restent viables (c'est-à-dire qu'ils puissent être lus sur un support) et lisibles (c'est-à-dire qu'ils puissent être affichés, lus ou interprétés de toute autre manière par des programmes applicatifs), ainsi qu'à s'assurer que les objets numériques dans l'entrepôt ne soient pas dégradés par erreur, et que les modifications appropriées effectuées sur ces objets soient documentées.

Le Dictionnaire de données n'a pas vocation à définir tous les éléments de métadonnées de préservation possibles, seulement ceux que la plupart des entrepôts auront besoin de connaître dans la plupart des cas. Plusieurs catégories de métadonnées sont exclues car elles sortent de son périmètre, notamment :

- Les métadonnées propres à un format, c'est-à-dire les métadonnées qui ne portent que sur un seul format de fichiers ou une seule famille de formats, comme l'audio, la vidéo ou les images vectorielles.

- Les métadonnées et règles métier propres à une implémentation, c'est-à-dire les métadonnées qui décrivent les principes et les pratiques d'un entrepôt particulier, tel que la façon dont il donne accès aux ressources.
- Les métadonnées descriptives. La description des ressources est évidemment pertinente dans le cadre de la préservation, mais de nombreux standards peuvent être utilisés à cette fin, comme MARC21, MODS ou Dublin Core.
- Les informations détaillées sur les supports et les machines. Ici encore, ces métadonnées sont clairement pertinentes dans le cadre de la préservation, mais leur définition est du ressort d'autres communautés.
- Les informations sur les agents (personnes, organisations et programmes), à part le minimum nécessaire à leur identification.
- Les informations sur les droits et permissions, sauf celles qui ont une incidence directe sur les fonctions de préservation.

Si l'on pense à toutes les métadonnées dont a besoin une organisation qui gère un entrepôt de préservation, on peut considérer que PREMIS définit un sous-ensemble au centre de celles-ci. D'un côté, il ne s'intéresse pas aux fonctions de recherche et d'accès ; de l'autre, il ne cherche pas à définir des métadonnées détaillées pour un format particulier. Il définit seulement les métadonnées nécessaires en général pour remplir les fonctions de préservation sur tout type de collection.

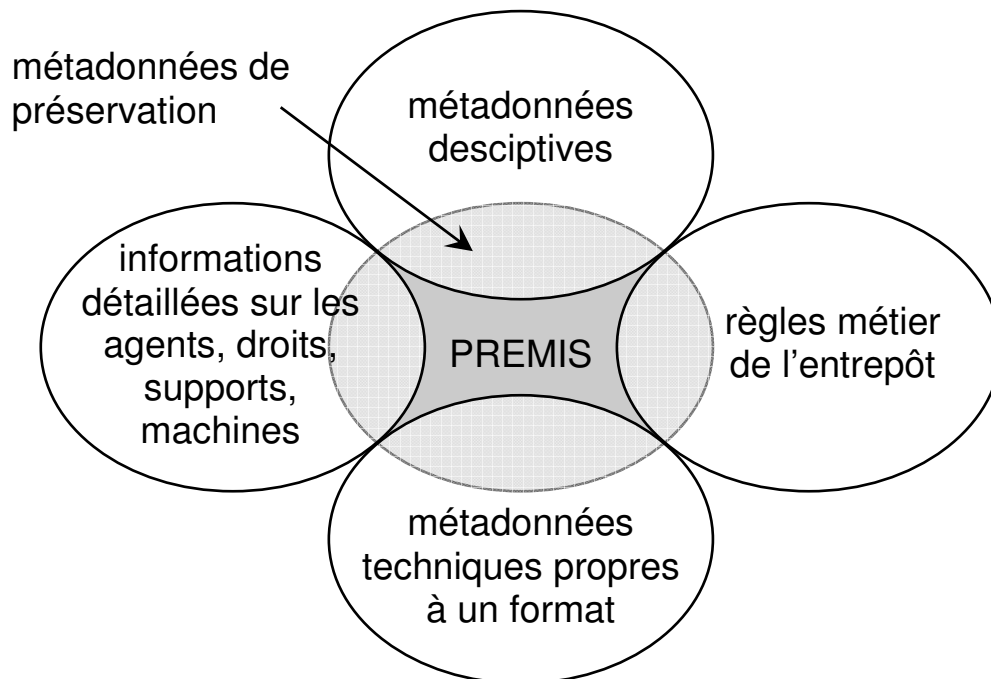


Figure 1. PREMIS comme sous-ensemble des métadonnées de préservation en général.

La figure 1 montre toutes les métadonnées pertinentes pour la préservation sous la forme du cercle gris au centre du diagramme. Le cercle inclut certaines métadonnées descriptives, certaines règles métier, certaines métadonnées techniques détaillées, et des informations détaillées sur les agents, les droits, les supports et les machines. PREMIS est le petit noyau

qui constitue le cœur des métadonnées de préservation, représenté ici en gris plus foncé, et qui exclut tous ces autres types de métadonnées.

1.4. Comment doit-on utiliser PREMIS ?

Le dictionnaire de données PREMIS définit ce qu'un entrepôt de préservation doit savoir. Il est important de noter que l'accent est mis sur le système de l'entrepôt et sa gestion, non sur les auteurs du contenu numérique, les personnes qui scannent ou convertissent de toute autre manière du contenu analogique en contenu numérique, ou le personnel qui évalue et commercialise des ressources électroniques. Les usages principaux de PREMIS sont tournés vers la conception d'entrepôts, leur évaluation, et l'échange de paquets d'information archivés entre entrepôts de préservation.

Il est conseillé que les personnes qui conçoivent et/ou développent des applications logicielles pour un entrepôt de préservation utilisent PREMIS comme un guide sur les informations à collecter et enregistrer dans l'application, ou à porter à l'attention des gestionnaires de l'entrepôt.

D'autre part, les personnes qui prévoient de mettre en œuvre un entrepôt de préservation pourront utiliser PREMIS comme une liste de points à évaluer dans un programme envisagé. Les systèmes qui peuvent fonctionner avec le dictionnaire de données PREMIS seront plus aptes à préserver des ressources d'informations sur le long terme.

Un entrepôt en fonctionnement aura parfois besoin d'exporter des paquets d'informations stockés pour qu'ils soient versés dans un autre entrepôt. Par exemple, il arrive qu'une organisation dépositaire migre d'un système d'entrepôt à un autre, ou qu'un client veuille passer d'un tiers archiveur à un autre. PREMIS fournit un ensemble commun d'éléments de données qui peut être compris par l'entrepôt qui exporte comme par celui qui importe, en particulier si le schéma XML de PREMIS est utilisé.

1.5. Devez-vous utiliser PREMIS ?

Cela dépend. La plupart des personnels de bibliothèques, d'archives, de musées et d'autres institutions culturelles ne sont pas directement impliqués dans la préservation du numérique. Si c'est votre cas, il vous suffit de savoir que PREMIS est un dictionnaire de données pour les métadonnées de préservation. Si votre poste comporte une part de responsabilité touchant à quelque aspect de la préservation numérique, vous trouverez sans doute utile de vous familiariser avec PREMIS. Si vous êtes impliqué dans l'évaluation ou la mise en œuvre d'un entrepôt ou d'un système de préservation dans une institution donnée, il est recommandé que vous ayez une bonne connaissance de PREMIS. Pensez à suivre un tutoriel si vous le pouvez : l'Activité de maintenance de PREMIS en propose régulièrement.

Si vous travaillez à des projets de numérisation, vous vous demandez peut-être si vous devez créer des métadonnées PREMIS, qui seront utilisées plus tard. La plupart des éléments dans PREMIS sont conçus pour être fournis automatiquement par l'application de l'entrepôt de préservation. (Ce qui ne veut pas dire que les applications actuellement disponibles les fournissent, bien entendu). Cependant, il est recommandé que vous enregistriez certaines informations, si possible :

Les restrictions. Sont définies comme *restrictions* toutes les caractéristiques d'un objet destinées à en restreindre l'accès, l'usage ou la migration. Il est difficile de documenter automatiquement les restrictions, car le programme utilisé à cette fin peut se voir interdire l'analyse de l'objet ; si l'on sait qu'un fichier comporte des restrictions, il est donc important de les consigner. PREMIS définit des unités sémantiques pour le type de restriction, leur cible

(les actions qui sont limitées) et les clefs (mots de passe et autres mécanismes pour contourner les restrictions).

La provenance. La *provenance numérique* documente l'historique des détenteurs et des modifications d'un objet numérique. Si votre institution a créé l'objet, les circonstances de sa création sont évidemment une partie importante de sa provenance. Le nom et la version de l'application qui l'a créé, et la date à laquelle elle l'a fait, peuvent souvent être extraits de l'en-tête du fichier, mais pas systématiquement ; il est donc recommandé de consigner cette information. PREMIS permet d'enregistrer l'historique des modifications sous la forme d'informations sur des Événements, décrites ci-dessous. Cependant, les types d'événements PREMIS sont essentiellement conçus pour décrire les actions qui ont lieu après la soumission à un entrepôt pour versement. Pour garder trace des événements qui ont lieu avant le versement, tels que l'acquisition et la prise en compte de l'objet, vous pourriez avoir besoin d'inventer vos propres types d'événements.

Les propriétés significatives. Les *propriétés significatives* sont les caractéristiques d'un objet qu'il faut maintenir au fil des actions de préservation. Par exemple, s'il s'agit d'un document, les mots et les images sont-ils les seuls à être indispensables, ou les polices, l'arrière-plan, le formatage, et les autres éléments de mise en page sont-ils tout aussi importants ? L'idée de propriétés significatives est l'un des concepts les plus importants en préservation du numérique, et l'un des moins bien compris. Un certain nombre d'initiatives se consacrent à mieux modéliser et décrire les propriétés significatives, mais elles en sont encore à leurs débuts. Néanmoins, toute institution qui crée ou acquiert des collections numériques pour une communauté d'utilisateurs a intérêt à réfléchir attentivement aux caractéristiques des collections qui sont importantes pour cette communauté, et essayer de les consigner pour de futurs usages.

Les droits. L'information sur les droits de propriété intellectuelle n'est pas spécifique à la préservation, bien sûr, mais savoir ce qu'on peut faire avec un objet est très important dans le processus de préservation. Vous devriez prendre soin de consigner toute l'information sur les droits disponible, y compris le statut juridique, les conditions de la licence d'utilisation et les permissions particulières.

2. CONVENTIONS DU DICTIONNAIRE DE DONNEES

2.1. Unités sémantiques

Le dictionnaire de données PREMIS définit des unités sémantiques, et non pas des éléments de métadonnées. Cette distinction est subtile mais importante. Une unité sémantique est un élément d'information ou de connaissance. Un élément de métadonnées est une manière définie de représenter cet élément d'information dans un enregistrement de métadonnées, un schéma ou une base de données. PREMIS ne précise pas comment les métadonnées doivent être représentées dans tout système ; il se borne à définir ce qu'un système a besoin de savoir et doit être en mesure d'exporter vers d'autres systèmes. Ainsi, si l'on est un puriste de PREMIS, on doit penser en termes d'unités sémantiques plutôt abstraites. Les unités sémantiques PREMIS ont une correspondance directe avec les éléments de métadonnées définis dans le schéma XML PREMIS ; elles peuvent avoir une correspondance moins directe avec des métadonnées dans d'autres schémas.

Les noms des unités sémantiques PREMIS sont des chaînes de caractères en casse mixte. Ces noms sont constitués de mots concaténés sans espace séparateur, mais identifiés par une

majuscule en début de mot. Exemples : `objectIdentifier`, `relatedEventIdentification`. Dans ce document, les noms des unités sémantiques PREMIS sont écrits dans la police Verdana.

2.2. Conteneurs et sous-unités

Certaines unités sémantiques sont définies comme des *conteneurs* ; ce qui signifie qu'elles ne possèdent pas une valeur par elles-mêmes, mais qu'elles servent pour regrouper des unités sémantiques liées. Par exemple, lorsqu'on enregistre un identifiant en PREMIS, on doit dire de quel type d'identifiant il s'agit (par exemple « DOI », « ISBN », « attribué par un système local »). Le conteneur `objectIdentifier` est utilisé pour regrouper les deux sous-unités : `objectIdentifierType` et `objectIdentifierValue`.

Les conteneurs fournissent une structure hiérarchique au dictionnaire de données, structure qui, dans la version 2.0, se reflète dans la numérotation des unités sémantiques :

1.1 `objectIdentifier` (M, R)

1.1.1 `objectIdentifierType` (M, NR)

1.1.2 `objectIdentifierValue` (M, NR)

Cet extrait du dictionnaire de données montre immédiatement que l'unité sémantique `objectIdentifier` est obligatoire (M) et répétable (R). Comme il y a des unités sémantiques définies au dessous de celle-ci, on peut en déduire qu'`objectIdentifier` ne possède pas une valeur en soi, mais sert de conteneur pour les éléments composants `objectIdentifierType` et `objectIdentifierValue`. Vu que `objectIdentifierType` et `objectIdentifierValue` sont non répétables (NR) à l'intérieur du conteneur, on doit répéter la structure du conteneur dans son intégralité pour pouvoir enregistrer deux identifiants différents.

2.3. Conteneurs d'extension

Un *conteneur d'extension* est un type particulier de conteneur pour lequel aucune sous-unité n'est définie. Ce type de conteneur est conçu pour fournir un moyen d'enregistrement de métadonnées non-PREMIS. De cette façon, PREMIS peut être étendu pour inclure des métadonnées qui seraient hors du périmètre ou absentes du dictionnaire de données. Les conteneurs d'extension ont le suffixe « Extension » dans la dernière partie de leur nom.

Par exemple, les métadonnées techniques spécifiques aux formats ne sont pas incluses dans PREMIS, mais cette information est très importante pour la conservation numérique. Le conteneur d'extension `objectCharacteristicsExtension` fournit un moyen pour enregistrer les métadonnées techniques définies par d'autres dictionnaires de données, tels que la norme Z39.87 pour décrire les images matricielles.

Si on est familier avec XML, il est évident que le dictionnaire de données PREMIS a été conçu pour être compatible avec XML. Les unités sémantiques PREMIS peuvent être implémentées comme des éléments XML ; les unités de conteneurs sont des éléments qui ne prennent comme contenu que d'autres éléments ; les unités d'extension sont les conteneurs des éléments définis par un schéma externe. Des précisions sur PREMIS et XML sont développées dans la section 5.1 ci-dessous.

3. MODELE DE DONNEES DE PREMIS

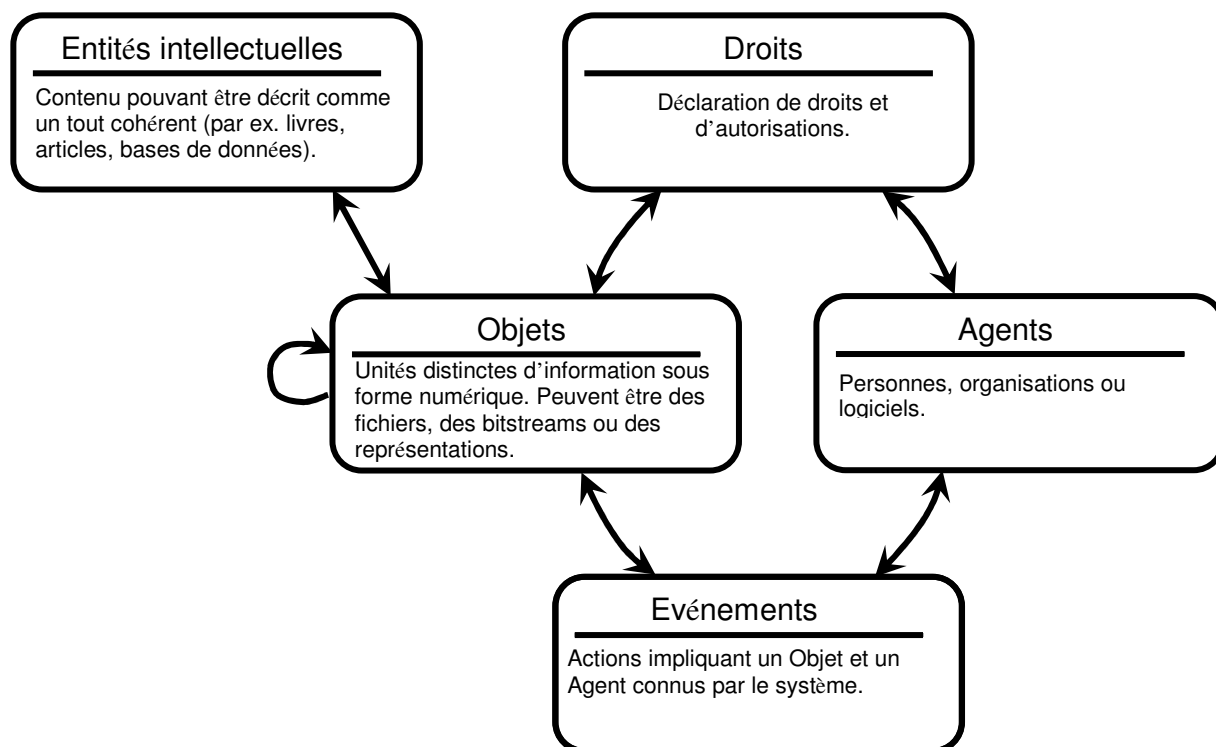


Figure 2. Modèle de données de PREMIS

Un des grands principes sous-jacents de PREMIS est la nécessité d'exprimer très clairement ce que l'on décrit. PREMIS définit cinq types d'éléments (appelés « entités ») que l'on peut décrire : Entités Intellectuelles, Objets, Agents, Événements et Droits. C'est ce que l'on appelle le modèle de données de PREMIS, présenté dans la figure 2 ci-dessus.

3.1. Entité intellectuelle

Les *Entités Intellectuelles* sont d'ordre conceptuel, et peuvent être qualifiées d'« entités bibliographiques ». PREMIS définit une Entité Intellectuelle comme « un ensemble de contenus que l'on considère comme une unité intellectuelle distincte à des fins de gestion ou de description : par exemple, un livre, une carte, une photographie, une base de donnée particulière ». En réalité, PREMIS ne définit pas de métadonnées pour décrire des Entités Intellectuelles car de très nombreux standards ou normes sont déjà disponibles pour les métadonnées descriptives.

PREMIS recommande que chaque objet au sein d'un système de préservation soit associé à son entité intellectuelle en incluant un identifiant de l'entité intellectuelle dans les métadonnées de l'objet. Ainsi, par exemple, si l'on veut préserver une copie de l'ouvrage *Bouddhisme : le livre numérique. Une introduction en ligne*, on pourra utiliser l'ISBN pour le relier à l'Entité Intellectuelle dans la description PREMIS de l'ouvrage.

3.2. Entité Objet

Les *Objets* sont généralement stockés et gérés par l'entrepôt de préservation. La plupart des éléments de PREMIS sont destinés à la description d'objets numériques. On peut enregistrer comme informations :

- un identifiant unique de l'objet (type et valeur) ;
- l'information d'intégrité, comme l'empreinte numérique (*message digest*) et l'algorithme utilisé pour la calculer ;
- la taille de l'objet ;
- le format de l'objet, qui peut être indiqué directement ou par un lien vers un répertoire de formats ;
- le nom original de l'objet ;
- des informations sur sa création ;
- des informations sur les éventuelles restrictions portant sur cet objet ;
- des informations sur ses propriétés significatives ;
- des informations sur son environnement (voir *infra*) ;
- où il est stocké, et sur quel support ;
- des informations sur sa signature numérique ;
- les relations de cet objet avec d'autres objets et d'autres types d'entités.

Plusieurs unités sémantiques sont définies pour garder trace de l'*environnement* d'un objet, c'est-à-dire le dispositif matériel et logiciel nécessaire pour y donner accès, et les dépendances possibles vis-à-vis d'autres objets. Par exemple, un fichier PDF peut être lu par plusieurs versions d'*Adobe Acrobat* et d'*Adobe Reader*, de même que par d'autres programmes, sous licence libre ou commerciale. Ceux-ci sont eux-mêmes utilisables sur divers systèmes d'exploitation et requièrent une configuration matérielle minimum (vitesse du processeur, mémoire et disque). Comme *Adobe Reader* n'est pas une application indépendante mais un greffon (*plug-in*) de navigateur, elle ne fonctionne qu'avec certaines versions de certains navigateurs : par exemple, *Reader 9* pour Mac OS fonctionne avec le navigateur Safari version 2.0.4 ou plus récente.

L'information sur l'environnement est essentielle pour certaines stratégies de préservation. On peut facilement indiquer avec PREMIS des informations d'environnement élémentaires : on peut par exemple se contenter de décrire un environnement au fonctionnement éprouvé ; mais il peut être difficile et chronophage de donner des informations exhaustives. Quand des informations détaillées concernent des classes d'objets plutôt que des instances locales d'objets, comme c'est le cas avec les informations d'environnement, il peut être utile de les agréger dans des répertoires et de les partager à destination de tous les entrepôts. PREMIS permet aux entrepôts de faire des liens avec des informations stockées dans des répertoires externes lorsque c'est préférable à un stockage local.

PREMIS définit en fait trois types d'objets différents et demande aux personnes en charge de l'implémentation de les distinguer. On différencie les objets *fichier* (« *file* »), les objets *représentation* (« *representation* »), et les objets *train d'octets* (« *bitstream* »).

Un objet fichier, c'est exactement ce que l'on imagine : un fichier informatique, par exemple un fichier PDF ou JPEG.

Un objet représentation est constitué de l'ensemble des objets fichier requis pour constituer une Entité Intellectuelle. Imaginons par exemple que l'on veuille préserver une page Web, comme la page d'accueil de son institution, à une date donnée. Il y a de fortes chances que la page d'accueil, telle qu'on la voit dans un navigateur, soit composée de nombreux fichiers différents – un ou plusieurs fichiers HTML, une poignée d'images GIF ou JPEG, peut-être de

l'audio ou une animation Flash. Il y a aussi probablement une feuille de style pour générer l'image que l'on a à l'écran. Il faut qu'un navigateur rassemble tous ces fichiers pour donner à lire la page d'accueil, donc si un entrepôt veut préserver une page d'accueil lisible, il doit avoir des informations sur tous ces fichiers, et sur la façon de les assembler. L'objet représentation permet à l'entrepôt non seulement d'identifier l'ensemble des fichiers reliés, mais aussi de décrire les caractéristiques du tout (par exemple la page Web) qui peuvent être différentes de celles des parties.

Les objets train d'octets sont des sous-ensembles de fichiers. On définit un objet train d'octets comme des données (octets) au sein d'un fichier qui a) ont des propriétés communes dans une perspective de préservation, et b) qui ne sont pas indépendantes car elles nécessitent l'addition d'un en-tête de fichier ou d'une autre structure. Donc, si l'on a par exemple un fichier dans le format AVI (audio-video interleaved), on peut vouloir distinguer le train d'octets audio du train d'octets vidéo, et les décrire comme des objets trains d'octets séparés.

Certaines unités sémantiques définies dans le dictionnaire de données PREMIS sont applicables aux trois types d'objets, tandis que d'autres sont applicables uniquement à l'un ou à deux d'entre eux. Avoir différents types d'objets contraint à réfléchir de façon approfondie à ce que l'on décrit et à être aussi précis que possible, ce qui est important dans la perspective d'un traitement automatisé.

3.3. Événements

L'entité *Événement* agrège des informations sur les actions ayant affecté ou affectant des objets dans un entrepôt. Il est essentiel d'enregistrer les événements de façon précise et fiable pour conserver la trace de la provenance numérique d'un objet, ce qui est ensuite important pour prouver l'authenticité de l'objet.

On peut enregistrer comme informations à propos d'un événement :

- un identifiant unique de l'événement (type et valeur) ;
- le type d'événement (création, versement, migration) ;
- la date et l'heure à laquelle l'événement s'est déroulé ;
- une description détaillée de l'événement ;
- un code indiquant le résultat de l'événement ;
- une description plus détaillée du résultat ;
- les agents impliqués dans l'événement et leurs rôles ;
- les objets impliqués dans l'événement et leurs rôles.

Chaque système d'entrepôt doit décider de lui-même quels événements enregistrer pour documenter de façon permanente l'histoire d'un objet. PREMIS recommande que les actions qui modifient un objet soient toujours enregistrées, et l'entrée `eventType` du Dictionnaire de données fournit une liste de départ des types d'événements importants pour encourager les entrepôts à enregistrer ces événements de façon systématique.

3.4. Agents

Les Agents peuvent être des individus, des organisations ou des applications logicielles. PREMIS ne définit qu'un nombre minimal d'unités sémantiques nécessaires à l'identification d'agents, dans la mesure où il y a plusieurs standards et normes externes qui peuvent être utilisés pour enregistrer une information plus détaillée (Pour un échantillon de ceux-ci, voir « Metadata standards and specifications for describing people and their interests » à l'adresse www.ukoln.ac.uk/metadata/resources/people/). Un entrepôt peut choisir d'utiliser un standard

distinct pour noter des informations additionnelles sur des agents, ou utiliser l'identifiant de l'agent pour pointer vers des informations externes.

Le dictionnaire de données inclut :

- un identifiant unique de l'agent (type et valeur),
- le nom de l'agent,
- la désignation du type d'agent (personne, organisation, logiciel).

À chaque fois qu'on fait référence à un agent vis-à-vis d'un événement ou d'une déclaration de droits, le rôle de l'agent doit être noté également. Tout agent peut avoir plusieurs rôles. Par exemple, je pourrais être l'auteur et le détenteur des droits d'une œuvre, l'auteur (mais pas le détenteur des droits) d'une seconde œuvre, et le déposant d'une troisième. Dans le modèle de données de PREMIS, un entrepôt devrait m'assigner un identifiant unique et donner la référence de cet identifiant pour chaque enregistrement d'événement ou chaque déclaration de droits où j'agis en tant qu'agent, ainsi que mon rôle dans ce contexte particulier. Le rôle d'un agent vis-à-vis d'un événement ou d'une déclaration de droits est considéré comme un propriété de l'entité événement ou de l'entité droits, pas de l'agent lui-même.

3.5. Droits

La plupart des stratégies de préservation impliquent de faire des copies à l'identique et des versions dérivées d'objets numériques. Or, ces actions sont limitées par le droit d'auteur aux détenteurs des droits. L'*entité Droits* agrège les informations sur les droits et les permissions qui touchent directement à la préservation des objets dans l'entrepôt. Chaque déclaration de droits PREMIS exprime deux choses : elle identifie les actions que l'entrepôt a le droit de réaliser, et elle indique le fondement de la revendication de ce droit.

Par exemple, un entrepôt peut posséder une version numérisée d'un livre publié en 1848 et qui est donc dans le domaine public. L'entrepôt peut donc faire tout ce qu'il veut de cette copie numérisée, en raison de son statut de propriété intellectuelle. Un autre entrepôt possède un objet copié à partir d'un CD publié, dont l'emballage indique que les copies de sauvegarde sont autorisées, mais restreint les droits d'accès et d'usage.

On peut enregistrer comme informations dans une déclaration de droits :

- un identifiant unique de la déclaration de droits (type et valeur) ;
- si ce(s) droit(s) s'appuient sur le *copyright*, une licence, ou sur la législation ;
- des informations plus détaillées sur le statut de *copyright*, les termes de la licence, ou sur la législation, en fonction des cas ;
- les actions que la déclaration de droits autorise ;
- toute limitation sur ces actions ;
- la durée de validité de la permission, ou la durée pendant laquelle la déclaration est valable ;
- l(es) objet(s) au(x)quel(s) la déclaration s'applique ;
- les agents impliqués dans la déclaration de droits, et leurs rôles.

La plupart des informations sont conçues pour être actionnables (c'est-à-dire enregistrées sous une forme contrôlée qui puisse être exploitée par un programme informatique). La déclaration de droits PREMIS est assertive, elle ne sert pas à enregistrer des informations détaillées à partir desquelles on peut déterminer des droits. C'est-à-dire que PREMIS ne définit pas de types précis d'information sur les auteurs, la date et le lieu de publication, ou sur la notification de *copyright*, contrairement à la spécification *copyrightMD* de la California Digital Library (www.cdlib.org/inside/projects/rights/schema/).

L'objectif de copyrightMD est d'aider les individus à déterminer les droits de façon dynamique, tandis que l'objectif de l'entité Droits de PREMIS est de fournir une information actionnable à destination des systèmes d'entrepôts de préservation.

4. LE DICTIONNAIRE DE DONNEES

4.1. Exemple d'entrée du dictionnaire de données pour une unité sémantique

La figure 3 montre l'entrée du dictionnaire de données pour l'unité sémantique `size`, qui est un composant ou sous-unité du conteneur appelé `objectCharacteristics`. L'unité sémantique `size` elle-même n'a pas de sous-unités. Cette entrée du dictionnaire de données comprend la définition de l'élément et la justification (« rationale ») de sa présence dans les métadonnées de base PREMIS, ainsi que des exemples et des notes sur la façon dont les valeurs peuvent être obtenues et utilisées. Toutes ces informations sont destinées à aider les utilisateurs à employer l'élément de manière appropriée.

Les deux lignes « Object Category » (Catégorie d'objet) et « Applicability » (Applicabilité) sont utilisées ensemble pour préciser si l'unité sémantique est appropriée pour décrire des représentations, des fichiers ou et/ou des trains d'octets. Ici, on indique que l'unité sémantique `size` n'est pertinente que pour des fichiers et des trains d'octets. Enfin, le tableau définit un ensemble de règles d'utilisation : « Data constraint » (Contraintes sur les données), « Repeatability » (Répétabilité) et « Obligation ».

La ligne « Contraintes sur les données » spécifie des restrictions sur les valeurs qu'une unité sémantique peut prendre. Dans cet exemple, la valeur de `size` doit être un entier. Le fait que la valeur doive appartenir à un vocabulaire contrôlé est une autre contrainte de données fréquente. Parfois, les termes du vocabulaire sont précisés dans le dictionnaire de données et parfois ils ne sont pas, mais dans tous les cas le nom du vocabulaire utilisé doit être enregistré. Le Dictionnaire de données ne définit pas d'unités sémantiques pour les éléments de vocabulaire, mais une solution à base de schémas XML est en cours de développement par l'Activité de maintenance de PREMIS.

La ligne « Répétabilité » indique si l'unité sémantique peut être répétée.

La ligne « Obligation » indique si une valeur pour l'unité sémantique est obligatoire (requis) ou facultative. La notion d'obligation est potentiellement source de confusion, parce que PREMIS précise clairement qu'il n'impose à aucun entrepôt le stockage de quelque information que ce soit. Une unité sémantique obligatoire n'a pas nécessairement à être enregistrée et stockée dans l'entrepôt. En revanche, l'entrepôt doit être en mesure de générer la valeur de l'unité sémantique lorsque cela est nécessaire, comme dans le cas d'un échange avec un autre entrepôt. Par exemple, dans le cas peu probable où un entrepôt stockerait exclusivement des images TIFF 6.0, cet entrepôt n'aurait pas besoin d'enregistrer les informations de format pour chaque objet. Néanmoins, la gestion de cet entrepôt devrait savoir que ses objets fichier sont des images TIFF 6.0, et pourrait fournir cette information si elle le devait. (Voir la section 5.2. Conformité à PREMIS).

Semantic unit (Unité sémantique)	1.5.3 size (taille)		
Semantic Components (Composants sémantiques)	Aucun		
Definition (Définition)	Taille en octets du fichier ou du train de bits stocké dans l'entrepôt.		
Rationale (Justification)	La taille est utile pour s'assurer qu'un nombre exact d'octets a été récupéré du stockage, et qu'une application a suffisamment d'espace pour déplacer ou traiter des fichiers. La taille peut également être utilisée lors de la facturation du stockage.		
Data constraint (Contrainte sur les données)	Entier		
Object category (Catégorie d'objet)	Représentation	Fichier	Train d'octets
Applicability (Applicabilité)	Non applicable	Applicable	Applicable
Examples (Exemples)		2038937	
Repeatability (Répétabilité)		Non répétable	Non répétable
Obligation		Facultatif	Facultatif
Creation / Maintenance notes (Commentaires de création / mise à jour)	Générés automatiquement par l'entrepôt.		
Usage notes (Commentaires d'utilisation)	Cette unité sémantique étant définie comme la taille en octets, il est inutile d'enregistrer une unité de mesure. Toutefois, à des fins d'échange de données, l'unité de mesure doit être déclarée ou comprise par les deux partenaires.		

Figure 3 : Extrait du dictionnaire de données pour l'unité sémantique size (taille).

4.2. Exemple d'entrée du dictionnaire de données pour une unité de conteneur

La figure 4 montre dans le dictionnaire de données le début de l'entrée pour objectCharacteristics, conteneur de size. On peut repérer qu'il s'agit d'un conteneur, car il a des composants sémantiques et la contrainte de données est « container ». Notez que les composants sémantiques inclus peuvent être unitaires, comme size, ou des conteneurs eux-mêmes, comme format.

Semantic unit (Unité sémantique)	1.5 objectCharacteristics (caractéristiques de l'objet)		
Semantic Components (Composants sémantiques)	1.5.1 compositionLevel (niveau de composition) 1.5.2 fixity (intégrité) 1.5.3 size (taille) 1.5.4 format (format) 1.5.5 creatingApplication (application de création) 1.5.6 inhibitors (restrictions) 1.5.7 objectCharacteristicsExtension (extension pour les caractéristiques de l'objet)		
Definition (Définition)	Propriétés techniques d'un fichier ou train d'octets qui sont applicables à l'ensemble ou à la plupart des formats.		
Rationale (Justification)	Certaines propriétés techniques importantes s'appliquent à tout objet quel que soit son format. La définition détaillée de propriétés spécifiques à un format est hors du périmètre de ce dictionnaire de données, bien que de telles propriétés puissent être incluses dans <i>objectCharacteristicsExtension</i> .		
Data constraint (Contrainte sur les données)	Conteneur		
Object category (Catégorie d'objet)	Représentation	Fichier	Train d'octets
Applicability (Applicabilité)	Non applicable	Applicable	Applicable
Repeatability (Répétabilité)		Répétable	Répétable
Obligation		Obligatoire	Obligatoire
Usage notes (Commentaires d'utilisation)	Les unités sémantiques incluses dans <i>objectCharacteristics</i> doivent être traitées comme un ensemble d'informations qui se rapportent à un objet unique et à un seul <i>niveau de composition</i> de cet objet. L'élément <i>objectCharacteristics</i> peut être répété.		

Figure 4 : Extrait du Dictionnaire de données pour l'unité sémantique objectCharacteristics.

5. UTILISER PREMIS

5.1. PREMIS en XML

Lorsque PREMIS est utilisé pour l'échange, on suppose (sans l'imposer) qu'il est représenté en XML. L'Activité de maintenance de PREMIS fournit un schéma XML reflétant le Dictionnaire de données afin de donner une description simple d'objets, d'événements, d'agents et de droits. La figure 5 montre un fragment de métadonnées PREMIS utilisant le schéma XML.

En pratique, la plupart des systèmes de préservation utilisent d'ores et déjà des formats XML pour importer et exporter des données. Beaucoup utilisent METS (Metadata Encoding and Transmission Standard, standard d'encodage et d'échange de métadonnées), autre standard maintenu par la Bibliothèque du Congrès, comme conteneur XML permettant de rassembler différents types de métadonnées. Il est possible d'utiliser PREMIS à l'intérieur de METS, mais ce n'est pas tout à fait simple pour deux raisons. Premièrement, METS divise l'information en différentes sections selon qu'il s'agit de métadonnées techniques, de droits ou de provenance. Le schéma PREMIS, conformément au Dictionnaire de données, possède

des sections pour les objets, les droits, les événements et les agents. Ces deux structures ont une correspondance partielle, mais incomplète, en particulier pour l'information sur les agents. Deuxièmement, PREMIS et METS sont en partie redondants ; par exemple, tous deux définissent un champ particulier pour consigner l'empreinte numérique (*checksum*). Si l'on utilise les deux formats, il faut décider si l'on enregistre ces éléments redondants dans des sections PREMIS, METS, ou les deux.

Bien sûr, si chaque système de préservation devait prendre ses propres décisions, il pourrait y avoir de très grandes variations dans la manière de représenter les données, ce qui poserait des problèmes d'interopérabilité. Par conséquent, plusieurs travaux visant à définir des bonnes pratiques d'utilisation conjointe de PREMIS et METS sont en cours. Cf. le site Web de l'Activité de maintenance de PREMIS pour un état d'avancement de ces travaux.

```
<event>
  <eventIdentifieur>
    <eventIdentifieurType>DAITSS</eventIdentifieurType>
    <eventIdentifieurValue>10012</eventIdentifieurValue>
  </eventIdentifieur>
  <eventType>Validation</eventType>
  <eventDateTime>2008-05-06T10:40:22-04:00</eventDateTime>
  <eventOutcomeInformation>
    <eventOutcome>Invalid</eventOutcome>
    <eventOutcomeDetail>ill-formed DateTime value</eventOutcomeDetail>
  </eventOutcomeInformation>
</event>
```

Schéma 5: Fragment de PREMIS en XML.

5.2. Conformité à PREMIS

Le Dictionnaire de données PREMIS comprend une section sur ce qui fait qu'un entrepôt est conforme à PREMIS. Il y a trois exigences principales :

- 1) Si l'entrepôt implémente (stocke ou exporte) un élément de données censé être une unité sémantique PREMIS, cet élément de données doit avoir la même définition, obéir aux mêmes contraintes sur les données et être appliqué de la même manière que l'unité sémantique définie dans PREMIS.
- 2) Si l'entrepôt implémente une unité sémantique PREMIS, il peut être plus exigeant que ce que PREMIS exige en termes de répétabilité et de caractère obligatoire de l'élément, mais en aucun cas plus libéral. En d'autres termes, une unité sémantique répétable peut être rendue non répétable dans une implémentation donnée, mais non l'inverse, et un élément obligatoire ne peut pas être rendu facultatif.
- 3) Si l'entrepôt exporte des informations pour qu'elles soient exploitées par un autre entrepôt, il doit fournir des valeurs pour toutes les unités sémantiques obligatoires dans le Dictionnaire de données. Cette obligation est toutefois relative, dans la mesure où les entrepôts n'ont pas à fournir les unités sémantiques obligatoires d'entités qu'ils n'utilisent pas. En d'autres termes, un entrepôt est libre d'utiliser ou non les Agents PREMIS, mais s'il les utilise, alors `agentIdentifier` est obligatoire. De la même manière, un entrepôt peut ne pas utiliser les objets trains d'octets (*bitstream*), et en ce cas il n'a pas à fournir l'identifiant du train d'octets qui serait obligatoire dans le cas contraire.

Ces exigences devraient être mises en regard d'un ensemble de choses qui ne sont PAS exigées pour être en conformité avec PREMIS. Comme indiqué plus haut, un entrepôt n'a pas l'obligation d'utiliser tous les types d'entités définies dans le modèle de données PREMIS. Il n'a pas non plus d'obligation de stocker les métadonnées en utilisant l'intitulé des unités sémantiques PREMIS, ou de suivre les contraintes définies par PREMIS pour certains éléments de données. En d'autres termes, peu importe la façon dont un entrepôt « reconnaît » une valeur PREMIS, que ce soit en l'entreposant avec le même intitulé ou sous un intitulé différent, en générant une valeur à partir d'une autre valeur source, en faisant un lien vers un répertoire de formats, par inférence, par défaut ou par quelque autre moyen que ce soit. À partir du moment où l'entrepôt peut fournir une valeur PREMIS conforme lorsqu'on la lui demande, il est conforme à PREMIS.

À l'avenir, l'activité de maintenance de PREMIS pourra reconsidérer les exigences de conformité et les rendre plus strictes, mais en l'état actuel de la version 2.0 il est relativement simple pour un entrepôt d'être conforme à PREMIS.

Néanmoins, plus grand est le nombre d'unités sémantiques qu'un entrepôt sait gérer, plus il retire de bénéfices de l'utilisation de PREMIS. Le Dictionnaire de données PREMIS a été conçu afin d'identifier les éléments minimaux dont la plupart des entrepôts auront besoin pour préserver leurs contenus numériques sur le long terme. Un entrepôt de préservation responsable devrait étudier PREMIS de manière approfondie et avoir des raisons justifiées de ne pas implémenter telle ou telle partie du Dictionnaire de données.

6. INFORMATIONS COMPLEMENTAIRES

Le site Web de l'Activité de maintenance de PREMIS (www.loc.gov/standards/premis/) comprend des ressources utiles à tous, y compris des liens vers le Groupe des Utilisateurs de PREMIS (PIG : PREMIS Implementers Group), des tutoriaux à PREMIS, des schémas, des outils et des actualités. Il comprend également une section « Resources: Articles and Presentations » (« Ressources : articles et présentations ») qui fournit des publications sur PREMIS et les sujets qui y sont liés. Les ressources les plus utiles pour des informations générales sont listées ci-dessous :

Sur les métadonnées de préservation en général :

“Preservation Metadata” (PDF:209K/21pp.)

Brian Lavoie (OCLC) and Richard Gartner (Oxford)

Rapport commun d'OCLC, d'Oxford Library Services, et de la Digital Preservation Coalition. Publié dans DPC Technology Watch Report No. 05-01: September 2005.

<http://www.dpconline.org/docs/reports/dpctw05-01.pdf>

Sur les différences entre PREMIS version 1.0 et 2.0:

"PREMIS with a Fresh Coat of Paint: Highlights from the Revision of the PREMIS Data Dictionary for Preservation Metadata "

Brian Lavoie, *D-Lib Magazine*, May/June 2008.

<http://www.dlib.org/dlib/may08/lavoie/05lavoie.html>

Sur l'utilisation conjointe de PREMIS et METS :

"Battle of the Buzzwords: Flexibility vs. Interoperability When Implementing PREMIS with METS"

Rebecca Guenther, *D-Lib Magazine*, July/August 2008.

<http://www.dlib.org/dlib/july08/guenther/07guenther.html>

APPENDICES

Appendice A : Liste de l'ensemble des unités sémantiques de PREMIS

Liste tirée du Dictionnaire de données PREMIS pour les métadonnées de préservation, version 2.0.

Comment lire cette liste

1.2 objectCategory (catégorie de l'objet) (M, NR)

L'unité sémantique objectCategory est obligatoire (M : Mandatory) et non répétable (NR : Not Repeatable). Elle s'applique à tous les types d'objets (file : fichier, representation : représentation, bitstream : train d'octets).

La traduction française de son intitulé est « catégorie d'objet ».

1.3 preservationLevel (niveau de préservation) (O, R) [representation, file]

1.3.1 preservationLevelValue (valeur du niveau de préservation) (M, NR) [representation, file]

L'unité sémantique preservationLevel est optionnelle (O : Optional) et répétable (R : Repeatable). Elle peut être utilisée uniquement pour des objets qui sont des représentations (« representation ») ou des fichiers (« file »). Il s'agit d'une unité de type conteneur car au moins une unité sémantique lui est subordonnée, preservationLevelValue, qui s'applique elle aussi uniquement aux représentations (« representation ») et aux fichiers (« file »).

Unités sémantiques de l'entité « objet »

1.1 objectIdentifier (identifiant de l'objet) (M, R)

1.1.1 objectIdentifierType (type d'identifiant de l'objet) (M, NR)

1.1.2 objectIdentifierValue (valeur de l'identifiant de l'objet) (M, NR)

1.2 objectCategory (catégorie de l'objet) (M, NR)

1.3 preservationLevel (niveau de préservation) (O, R) [representation, file]

1.3.1 preservationLevelValue (valeur du niveau de préservation) (M, NR) [representation, file]

1.3.2 preservationLevelRole (fonction du niveau de préservation) (O, NR) [representation, file]

1.3.3 preservationLevelRationale (justification du niveau de préservation) (O, R) [representation, file]

1.3.4 preservationLevelDateAssigned (date d'attribution du niveau de préservation) (O, NR) [representation, file]

1.4 significantProperties (propriété significative) (O, R)

1.4.1 significantPropertiesType (type de propriété significative) (O, NR)

1.4.2 significantPropertiesValue (valeur de la propriété significative) (O, NR)

1.4.3 significantPropertiesExtension (extension pour les propriétés significatives) (O, R)

1.5 objectCharacteristics (caractéristiques de l'objet) (M, R) [file, bitstream]

1.5.1 compositionLevel (niveau de composition) (M, NR) [file, bitstream]

1.5.2 fixity (intégrité) (O, R) [file, bitstream]

- 1.5.2.1 messageDigestAlgorithm (algorithme de l’empreinte numérique) (M, NR) [file, bitstream]
- 1.5.2.2 messageDigest (empreinte numérique) (M, NR) [file, bitstream]
- 1.5.2.3 messageDigestOriginator (origine de l’empreinte numérique) (O, NR) [file, bitstream]
- 1.5.3 size (taille) (O, NR) [file, bitstream]
- 1.5.4 format (format) (M, R) [file, bitstream]
 - 1.5.4.1 formatDesignation (dénomination du format) (O, NR) [file, bitstream]
 - 1.5.4.1.1 formatName (nom du format) (M, NR) [file, bitstream]
 - 1.5.4.1.2 formatVersion (version du format) (O, NR) [file, bitstream]
 - 1.5.4.2 formatRegistry (répertoire de formats) (O, NR) [file, bitstream]
 - 1.5.4.2.1 formatRegistryName (nom du répertoire de formats) (M, NR) [file, bitstream]
 - 1.5.4.2.2 formatRegistryKey (identifiant dans le répertoire de formats) (M, NR) [file, bitstream]
 - 1.5.4.2.3 formatRegistryRole (fonction du répertoire de formats) (O, NR) [file, bitstream]
 - 1.5.4.3 formatNote (notes sur le format) (O, R) [file, bitstream]
- 1.5.5 creatingApplication (application d’origine) (O, R) [file, bitstream]
 - 1.5.5.1 creatingApplicationName (nom de l’application d’origine) (O, NR) [file, bitstream]
 - 1.5.5.2 creatingApplicationVersion (version de l’application d’origine) (O, NR) [file, bitstream]
 - 1.5.5.3 dateCreatedByApplication (date de création par l’application) (O, NR) [file, bitstream]
 - 1.5.5.4 creatingApplicationExtension (extension pour l’application d’origine) (O, R) [file, bitstream]
- 1.5.6 inhibitors (restrictions) (O, R) [file, bitstream]
 - 1.5.6.1 inhibitorType (type de restriction) (M, NR) [file, bitstream]
 - 1.5.6.2 inhibitorTarget (cible de la restriction) (O, R) [file, bitstream]
 - 1.5.6.3 inhibitorKey (clef de la restriction) (O, NR) [file, bitstream]
- 1.5.7 objectCharacteristicsExtension (extension pour les caractéristiques de l’objet) (O, R) [file, bitstream]
- 1.6 originalName (nom d’origine) (O, NR) [representation, file]
- 1.7 storage (stockage) (M, R) [file, bitstream]
 - 1.7.1 contentLocation (emplacement du contenu) (O, NR) [file, bitstream]
 - 1.7.1.1 contentLocationType (type d’emplacement du contenu) (M, NR) [file, bitstream]
 - 1.7.1.2 contentLocationValue (valeur de l’emplacement du contenu) (M, NR) [file, bitstream]
 - 1.7.2 storageMedium (support de stockage) (O, NR) [file, bitstream]
- 1.8 environment (environnement) (O, R)
 - 1.8.1 environmentCharacteristic (caractéristique de l’environnement) (O, NR)
 - 1.8.2 environmentPurpose (fonction de l’environnement) (O, R)
 - 1.8.3 environmentNote (note sur l’environnement) (O, R)
 - 1.8.4 dependency (dépendance) (O, R)
 - 1.8.4.1 dependencyName (nom de la dépendance) (O, R)
 - 1.8.4.2 dependencyIdentifier (identifiant de la dépendance) (O, R)
 - 1.8.4.2.1 dependencyIdentifierType (type d’identifiant de la dépendance) (M, NR)

- 1.8.4.2.2 dependencyIdentifierValue (valeur de l'identifiant de la dépendance) (M, NR)
- 1.8.5 software (logiciel) (O, R)
 - 1.8.5.1 swName (nom du logiciel) (M, NR)
 - 1.8.5.2 swVersion (version du logiciel) (O, NR)
 - 1.8.5.3 swType (type de logiciel) (M, NR)
 - 1.8.5.4 swOtherInformation (autres informations sur le logiciel) (O, R)
 - 1.8.5.5 swDependency (dépendance logicielle) (O, R)
- 1.8.6 hardware (matériel) (O, R)
 - 1.8.6.1 hwName (nom du matériel) (M, NR)
 - 1.8.6.2 hwType (type de matériel) (M, NR)
 - 1.8.6.3 hwOtherInformation (autres informations sur le matériel) (O, R)
- 1.8.7 environmentExtension (extension pour l'environnement) (O, R)
- 1.9 signatureInformation (informations sur les signatures) (O, R) [file, bitstream]
 - 1.9.1 signature (signature) (O, R)
 - 1.9.1.1 signatureEncoding (encodage de la signature) (M, NR) [file, bitstream]
 - 1.9.1.2 signer (signataire) (O, NR) [file, bitstream]
 - 1.9.1.3 signatureMethod (méthode de la signature) (M, NR) [file, bitstream]
 - 1.9.1.4 signatureValue (valeur de la signature) (M, NR) [file, bitstream]
 - 1.9.1.5 signatureValidationRules (règles de validation de la signature) (M, NR) [file, bitstream]
 - 1.9.1.6 signatureProperties (propriété de la signature) (O, R) [file, bitstream]
 - 1.9.1.7 keyInformation (informations sur la clé) (O, NR) [file, bitstream]
 - 1.9.2 signatureInformationExtension (extension pour les informations sur la signature) (O, R) [file, bitstream]
- 1.10 relationship (relation) (O, R)
 - 1.10.1 relationshipType (type de relation) (M, NR)
 - 1.10.2 relationshipSubType (sous-type de relation) (M, NR)
 - 1.10.3 relatedObjectIdentification (identification de l'objet en relation) (M, R)
 - 1.10.3.1 relatedObjectIdentifierType (type d'identifiant de l'objet en relation) (M, NR)
 - 1.10.3.2 relatedObjectIdentifierValue (valeur de l'identifiant de l'objet en relation) (M, NR)
 - 1.10.3.3 relatedObjectSequence (ordre de l'objet en relation) (O, NR)
 - 1.10.4 relatedEventIdentification (identification de l'événement en relation) (O, R)
 - 1.10.4.1 relatedEventIdentifierType (type d'identifiant de l'événement en relation) (M, NR)
 - 1.10.4.2 relatedEventIdentifierValue (valeur de l'identifiant de l'événement en relation) (M, NR)
 - 1.10.4.3 relatedEventSequence (ordre de l'événement en relation) (O, NR)
- 1.11 linkingEventIdentifier (identifiant de l'événement lié) (O, R)
 - 1.11.1 linkingEventIdentifierType (type d'identifiant de l'événement lié) (M, NR)
 - 1.11.2 linkingEventIdentifierValue (valeur de l'identifiant de l'événement lié) (M, NR)
- 1.12 linkingIntellectualEntityIdentifier (identifiant de l'entité intellectuelle liée) (O, R)
 - 1.12.1 linkingIntellectualEntityIdentifierType (M, NR) (type d'identifiant de l'entité intellectuelle liée)
 - 1.12.2 linkingIntellectualEntityIdentifierValue (valeur de l'identifiant de l'entité intellectuelle liée) (M, NR)
- 1.13 linkingRightsStatementIdentifier (identifiant de la déclaration de droits liée) (O, R)

1.13.1 linkingRightsStatementIdentifierType (type d'identifiant de la déclaration de droits liée) (M, NR)

1.13.2 linkingRightsStatementIdentifierValue (valeur de l'identifiant de la déclaration de droits liée) (M, NR)

Unités sémantiques de l'entité événement

2.1 eventIdentifier (identifiant de l'événement) (M, NR)

2.1.1 eventIdentifierType (type d'identifiant de l'événement) (M, NR)

2.1.2 eventIdentifierValue (valeur de l'identifiant de l'événement) (M, NR)

2.2 eventType (type d'événement) (M, NR)

2.3 eventDateTime (date de l'événement) (M, NR)

2.4 eventDetail (détails sur l'événement) (O, NR)

2.5 eventOutcomeInformation (informations sur le résultat d'un événement) (O, R)

2.5.1 eventOutcome (résultat d'un événement) (O, NR)

2.5.2 eventOutcomeDetail (détails sur le résultat d'un événement) (O, R)

2.5.2.1 eventOutcomeDetailNote (note de détail du résultat d'un événement (O, NR)

2.5.2.2 eventOutcomeDetailExtension (extension détaillant le résultat d'un événement) (O, R)

2.6 linkingAgentIdentifier (identifiant de l'agent lié) (O, R)

2.6.1 linkingAgentIdentifierType (type d'identifiant de l'agent lié) (M, NR)

2.6.2 linkingAgentIdentifierValue (valeur de l'identifiant de l'agent lié) (M, NR)

2.6.3 linkingAgentRole (rôle de l'agent lié) (O, R)

2.7 linkingObjectIdentifier (identifiant de l'objet lié) (O, R)

2.7.1 linkingObjectIdentifierType (type d'identifiant de l'objet lié) (M, NR)

2.7.2 linkingObjectIdentifierValue (valeur de l'identifiant de l'objet lié) (M, NR)

2.7.3 linkingObjectRole (rôle de l'objet lié) (O, R)

Unités sémantiques de l'entité agent

3.1 agentIdentifier (identifiant de l'agent) (R, M)

3.1.1 agentIdentifierType (type d'identifiant de l'agent) (M, NR)

3.1.2 agentIdentifierValue (valeur de l'identifiant de l'agent) (M, NR)

3.2 agentName (nom de l'agent) (O, R)

3.3 agentType (type d'agent) (O, NR)

Unités sémantiques de l'entité déclaration de droits

4.1 rightsStatement (déclaration de droits) (O, R)

4.1.1 rightsStatementIdentifier (identifiant de la déclaration de droits) (M, NR)

4.1.1.1 rightsStatementIdentifierType (type d'identifiant de la déclaration de droits) (M, NR)

4.1.1.2 rightsStatementIdentifierValue (valeur de l'identifiant de la déclaration des droits) (M, NR)

4.1.2 rightsBasis (fondement des droits) (M, NR)

4.1.3 copyrightInformation (informations sur le *copyright*) (O, NR)

4.1.3.1 copyrightStatus (statut de *copyright*) (M, NR)

4.1.3.2 copyrightJurisdiction (aire d'application du *copyright*) (M, NR)

4.1.3.3 copyrightStatusDeterminationDate (date d'établissement du statut de *copyright*) (O, NR)

- 4.1.3.4 copyrightNote (note sur le *copyright*) (O, R)
- 4.1.4 licenseInformation (informations sur la licence) (O, NR)
 - 4.1.4.1 licenseIdentifiant (identifiant de la licence) (O, NR)
 - 4.1.4.1.1 licenseIdentifiantType (type d'identifiant de la licence) (M, NR)
 - 4.1.4.1.2 licenseIdentifiantValue (valeur de l'identifiant de la licence) (M, NR)
 - 4.1.4.2 licenseTerms (termes de la licence) (O, NR)
 - 4.1.4.3 licenseNote (note sur la licence) (O, R)
- 4.1.5 statuteInformation (informations sur le texte législatif) (O, R)
 - 4.1.5.1 statuteJurisdiction (aire d'application du texte législatif) (M, NR)
 - 4.1.5.2 statuteCitation (référence du texte législatif) (M, NR)
 - 4.1.5.3 statuteInformationDeterminationDate (date d'établissement des informations sur le texte législatif) (O, NR)
 - 4.1.5.4 statuteNote (note sur le texte législatif) (O, R)
- 4.1.6 rightsGranted (autorisations accordées) (O, R)
 - 4.1.6.1 act (action) (M, NR)
 - 4.1.6.2 restriction (restriction) (O, R)
 - 4.1.6.3 termOfGrant (durée de validité des autorisations) (M, NR)
 - 4.1.6.3.1 startDate (date de début) (M, NR)
 - 4.1.6.3.2 endDate (date de fin) (O, NR)
 - 4.1.6.4 rightsGrantedNote (note sur les autorisations accordées) (O, R)
- 4.1.7 linkingObjectIdentifiant (identifiant de l'objet lié) (O, R)
 - 4.1.7.1 linkingObjectIdentifiantType (type d'identifiant de l'objet lié) (M, NR)
 - 4.1.7.2 linkingObjectIdentifiantValue (valeur de l'identifiant de l'objet lié) (M, NR)
- 4.1.8 linkingAgentIdentifiant (identifiant de l'agent lié) (O, R)
 - 4.1.8.1 linkingAgentIdentifiantType (type d'identifiant de l'agent lié) (M, NR)
 - 4.1.8.2 linkingAgentIdentifiantValue (valeur de l'identifiant de l'agent lié) (M, NR)
 - 4.1.8.3 linkingAgentRole (rôle de l'agent lié) (M, NR)
- 4.2 rightsExtension (extension pour les droits) (O, R)

Appendice B : Exemple d'Objet

L'exemple ci-dessous est la simplification d'un exemple conçu par la Bibliothèque du Congrès pour un tutoriel PREMIS. Il montre les unités sémantiques PREMIS et les valeurs utilisées pour décrire une image TIFF.

La LoC utilise des identifiants handle (identifiants créés et gérés par une application appelée le système Handle) pour les fichiers stockés dans l'entrepôt (cf. 1.1 objectIdentifiant et 1.10.3 relatedObjectIdentification). L'objet décrit est reconnu dans l'entrepôt par le biais de son identifiant unique (1.1 objectIdentifiant) mais son nom d'origine avant le versement était « 001h.tif » (1.6 originalName). Nous savons qu'il s'agit d'un fichier grâce à la catégorie de l'objet (1.2 objectCategory).

Le fichier possède une empreinte MD5 calculée par le système de gestion de contenus de la LoC (1.5.2 fixity). Son format est décrit de deux manières différentes, par un type MIME et un numéro de version (1.5.4.1 formatDesignation) et en pointant vers le répertoire de formats PRONOM qui détient une copie de la spécification TIFF.

La LoC considère qu'il y a eu deux applications d'origine pour ce fichier. Un élément non déterminé (probablement un fichier RAW sans nom) a été créé sur un scanner par l'application ScandAll et a été transformé en TIFF en utilisant Adobe Photoshop (1.5.5 creatingApplication). Il n'y a pas de cryptage ou d'autre type de restriction que la LoC ait jugé pertinent de consigner explicitement (1.5.6 inhibitors).

Le fichier est stocké sur disque à un emplacement identifié par son chemin dans l'arborescence (1.7.1 contentLocation); le type de support correspondant est décrit très précisément, sans doute afin que les gestionnaires de l'entrepôt puissent déterminer le moment où les données nécessiteront une migration vers un type de support plus récent (1.7.2 storageMedium).

La LoC a pris la peine de consigner un environnement recommandé pour visualiser le fichier : utiliser Adobe Acrobat version 5.0 sur un processeur Intel x86 fonctionnant sous Windows XP. La vitesse du processeur doit être au moins de 60 MHz et il doit y avoir au moins 32 Mo de mémoire, et de préférence 64 Mo (1.8 environment).

Le fichier TIFF fait partie d'une représentation dont l'identifiant interne est *R200802948*. Cette information est donnée par le troisième bloc d'informations sur les relations (1.10 relationship). Il y a deux autres fichiers dans cette représentation, identifiés par des identifiants Handle. Nous savons qu'il doit s'agir de fichiers car le type de relation consigné est « frère ». Il nous faut consulter la description des objets *loc.music/gottlieb.09602* et *loc.music/gottlieb.mets* pour plus d'informations sur ces derniers, par exemple pour savoir de quel type de fichiers il s'agit.

Nous n'avons pas d'information descriptive sur l'entité intellectuelle correspondant à cette représentation. Toutefois, nous avons un lien vers son identifiant, un LCCN (1.12 linkingIntellectualEntityIdentifier). En toute logique, si l'on recherchait cet LCCN, nous trouverions une description bibliographique de l'entité intellectuelle en question (pas dans cet exemple toutefois car le LCCN a été forgé).

Unité sémantique	Valeur
1.1 objectIdentifier (identifiant de l'objet)	
1.1.1 objectIdentifierType (type d'identifiant de l'objet)	hdl
1.1.2 objectIdentifierValue (valeur de l'identifiant de l'objet)	loc.music/gottlieb.09601
1.2 objectCategory (catégorie de l'objet)	file
1.3 preservationLevel (niveau de préservation)	
1.3.1 preservationLevelValue (valeur du niveau de préservation)	full
1.5 objectCharacteristics (caractéristiques de l'objet)	
1.5.1 compositionLevel (niveau de composition)	0
1.5.2 fixity (intégrité)	
1.5.2.1 messageDigestAlgorithm (algorithme de l'empreinte numérique)	MD5
1.5.2.2 messageDigest (empreinte numérique)	36b03197ad066cd719906c55eb68ab8d
1.5.2.3 messageDigestOriginator (origine de l'empreinte numérique)	localDCMS
1.5.3 size (taille)	20800896
1.5.4 format (format)	
1.5.4.1 formatDesignation (dénomination du format)	
1.5.4.1.1 formatName (nom du format)	image/tiff
1.5.4.1.2 formatVersion (version du	6.0

COMPRENDRE PREMIS

format)	
1.5.4.2 formatRegistry (répertoire de formats)	
1.5.4.2.1 formatRegistryName (nom du répertoire de formats)	PRONOM
1.5.4.2.2 formatRegistryKey (identifiant dans le répertoire de formats)	fmt/10
1.5.4.2.3 formatRegistryRole (fonction du répertoire de formats)	specification
1.5.5 creatingApplication (application d'origine)	
1.5.5.1 creatingApplicationName (nom de l'application d'origine)	ScandAll21
1.5.5.2 creatingApplicationVersion (version de l'application d'origine)	4.1.4
1.5.5.3 dateCreatedByApplication (date de création par l'application)	1998-10-30
1.5.5 creatingApplication (application d'origine)	
1.5.5.1 creatingApplicationName (nom de l'application d'origine)	Adobe Photoshop
1.5.5.2 creatingApplicationVersion (version de l'application d'origine)	CS2
1.5.5.3 dateCreatedByApplication (date de création par l'application)	1998-10-30T08:29:02
1.5.6 inhibitors (restrictions)	
1.5.6.1 inhibitorType (type de restriction)	none
1.6 originalName (nom d'origine)	001h.tif
1.7 storage (stockage)	
1.7.1 contentLocation (emplacement du contenu)	
1.7.1.1 contentLocationType (type d'emplacement du contenu)	filepath
1.7.1.2 contentLocationValue (valeur de l'emplacement du contenu)	amsserver/
1.7.2 storageMedium (support de stockage)	IBM DS4000 System 1740-5208
1.8 environment (environnement)	
1.8.1 environmentCharacteristic (caractéristique de l'environnement)	recommended
1.8.2 environmentPurpose (fonction de l'environnement)	render
1.8.2 environmentPurpose (fonction de l'environnement)	edit
1.8.5 software (logiciel)	
1.8.5.1 swName (nom du logiciel)	Adobe Acrobat
1.8.5.2 swVersion (version du logiciel)	5.0
1.8.5.3 swType (type de logiciel)	renderer
1.8.5 software (logiciel)	
1.8.5.1 swName (nom du logiciel)	Windows
1.8.5.2 swVersion (version du logiciel)	XP
1.8.5.3 swType (type de logiciel)	OperatingSystem
1.8.6 hardware (matériel)	
1.8.6.1 hwName (nom du matériel)	Intel x86
1.8.6.2 hwType (type de matériel)	processor
1.8.6.3 hwOtherInformation (autres informations sur le matériel)	60 MHz minimum
1.8.6 hardware (matériel)	
1.8.6.1 hwName (nom du matériel)	64 MB RAM
1.8.6.2 hwType (type de matériel)	memory
1.8.6.3 hwOtherInformation (autres informations sur le matériel)	32 MB minimum
1.10 relationship (relation)	

1.10.1 relationshipType (type de relation)	structural
1.10.2 relationshipSubType (sous-type de relation)	has sibling
1.10.3 relatedObjectIdentification (identification de l'objet en relation)	
1.10.3.1 relatedObjectIdentifierType (type d'identifiant de l'objet en relation)	hdl
1.10.3.2 relatedObjectIdentifierValue (valeur de l'identifiant de l'objet en relation)	loc.music/gottlieb.09602
1.10.3.3 relatedObjectSequence (ordre de l'objet lié)	0
1.10 relationship (relation)	
1.10.1 relationshipType (type de relation)	structural
1.10.2 relationshipSubType (sous-type de relation)	has sibling
1.10.3 relatedObjectIdentification (identification de l'objet en relation)	
1.10.3.1 relatedObjectIdentifierType (type d'identifiant de l'objet en relation)	hdl
1.10.3.2 relatedObjectIdentifierValue (valeur de l'identifiant de l'objet en relation)	loc.music/gottlieb.mets
1.10.3.3 relatedObjectSequence (ordre de l'objet lié)	0
1.10 relationship (relation)	
1.10.1 relationshipType (type de relation)	structural
1.10.2 relationshipSubType (sous-type de relation)	is included in
1.10.3 relatedObjectIdentification (identification de l'objet en relation)	
1.10.3.1 relatedObjectIdentifierType (type d'identifiant de l'objet en relation)	LocalRepository
1.10.3.2 relatedObjectIdentifierValue (valeur de l'identifiant de l'objet en relation)	R200802948
1.10.3.3 relatedObjectSequence (ordre de l'objet lié)	0
1.12 linkingIntellectualEntityIdentifier (identifiant de l'entité intellectuelle liée)	
1.12.1 linkingIntellectualEntityIdentifierType (type d'identifiant de l'entité intellectuelle liée)	LCCN
1.12.2 linkingIntellectualEntityIdentifierValue (valeur de l'identifiant de l'entité intellectuelle liée)	2007-86121

Appendice C : glossaire des termes

Ce glossaire rassemble des définitions qui ont été mobilisées plus tôt dans ce guide. Ces définitions peuvent être moins formelles que celles apparaissant dans le Dictionnaire de données PREMIS pour les métadonnées de préservation (*PREMIS Data Dictionary for Preservation Metadata*).

actionnable : Fait d'être enregistré sous une forme contrôlée qui peut être exploitée par un programme informatique.

agent : Personne, organisation ou programme informatique qui joue un rôle dans un événement (event) ou une déclaration de droits (rights).

conteneur d'extension : Type particulier d'*unité de conteneur* PREMIS pour laquelle aucune sous-unité n'est définie, mais qui sert d'emplacement pour des métadonnées non-PREMIS.

COMPRENDRE PREMIS

émulation : *Stratégie de préservation* qui consiste à reproduire un environnement de consultation ancien sur un matériel et/ou logiciel plus récent.

entité : Dans le cadre de PREMIS, chose que l'on souhaite décrire. Les types d'entités PREMIS sont les *Entités intellectuelles, Objets, Agents, Evénements* et *Droits*.

entité droits : Entité PREMIS agrégeant des informations sur les droits et les permissions portant sur les objets d'un système de préservation.

entité événement : Entité PREMIS regroupant des informations sur les actions portant sur des objets de l'entrepôt.

entité intellectuelle : Ensemble de contenus traités comme un tout à des fins de gestion et de description ; similaire à une « entité bibliographique » en bibliothéconomie.

environnement : Matériel, logiciel et autres objets nécessaires pour la lecture d'un objet.

métadonnées de préservation : Métadonnées facilitant les activités visant à garantir qu'une ressource numérique reste utilisable sur le long terme.

migration : *Stratégie de préservation* consistant à créer une version d'un fichier numérique dans un format de fichier plus récent.

objets : Documents numériques stockés et manipulés dans un système de préservation. PREMIS définit trois types d'objets : les fichiers, les trains d'octets et les représentations.

objet train d'octets : Type particulier d'objet PREMIS : données contenues dans un fichier possédant des caractéristiques communes à des fins de préservation et qui n'ont pas d'autonomie propre.

objet fichier : Type d'objet PREMIS : un fichier informatique, comme un fichier PDF ou JPEG.

propriétés significatives : Caractéristiques d'un objet qui doivent être préservées au fil des opérations de préservation.

provenance numérique : Documentation de l'historique des détenteurs et des changements survenus sur une ressource numérique.

restrictions : Fonctionnalités d'un objet numérique destinées à en restreindre l'accès, l'utilisation ou la *migration*.

représentation : Type d'objet PREMIS : ensemble des objets fichiers requis pour représenter une *entité intellectuelle*.

stratégies de préservation : Techniques employées pour garantir que les ressources numériques restent utilisables sur le long terme ; la *migration* et l'*émulation* sont deux stratégies de préservation courantes.

unités de conteneur (ou conteneurs) : *Unités sémantiques* qui ne comprennent pas directement de valeur mais ont pour fonction de regrouper des sous-unités liées.

unités sémantiques : unité d'information ou de connaissance.