

HEG-796-22-040

AIPs et OCFL

Jan Krause-Bilvin

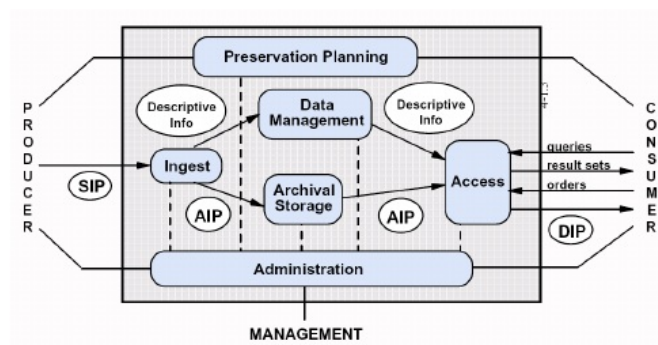
2022-04-11

Thème de cette session

- Archival Information Packages (AIP) : paquets d'information archivistiques.
- Oxford Common File Layout (OCFL) : une spécification pour les paquets d'information.

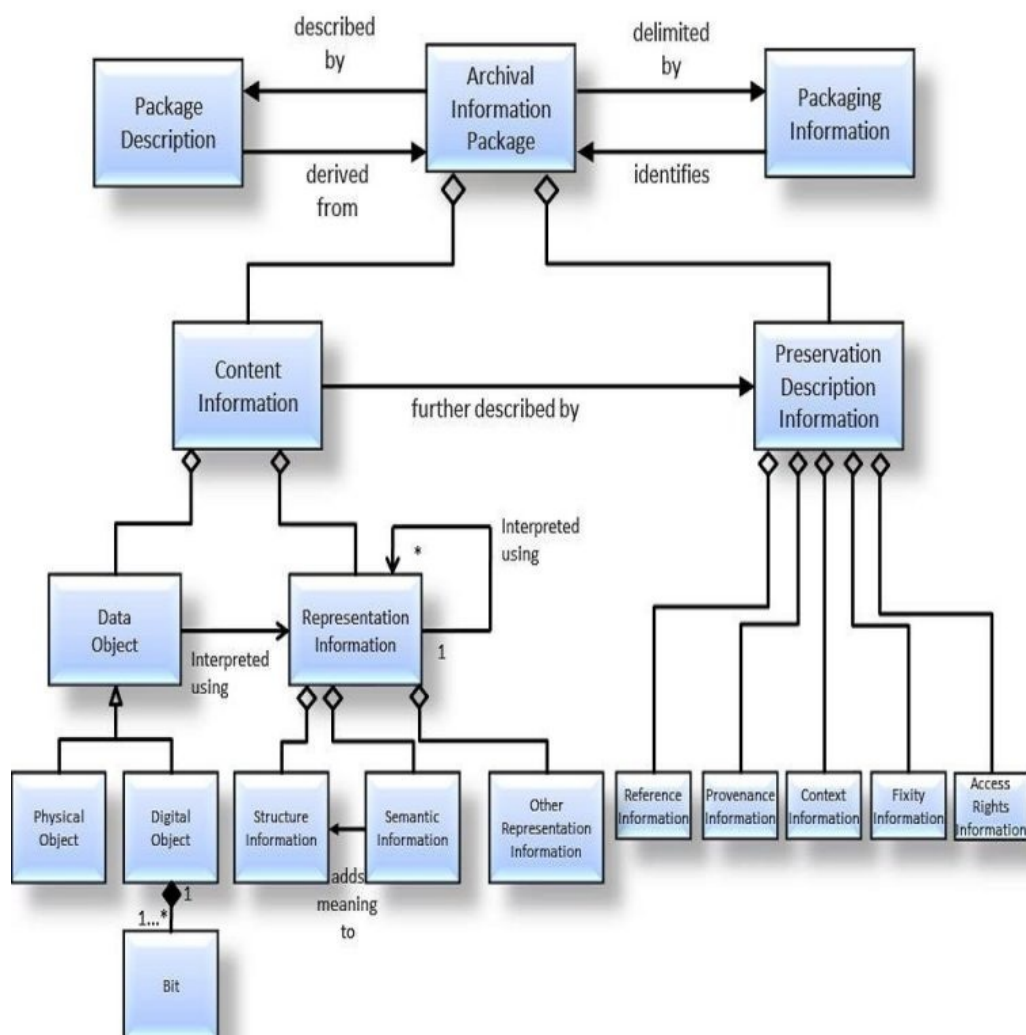
AIPs

Rappel - schéma d'un système de préservation OAIS :



OAIS schema

- Les AIPs sont autonomes : ils contiennent toutes les données et métadonnées d'une "unité archivistique".
- Au coeur du module de stockage OAIS, ils sont structurés de façon à ce que celui-ci puisse remplir ses fonctions.



OAIS schema

Dans Fedora Commons

- Par défaut, chaque container LDP est stocké comme un seul AIP.
- Il est possible d'attribuer la fonction "archival unit" à un container LDP.
- Ceci a pour effet que ce container et tout ses enfants (définis par *ldp:contains*) sont "physiquement" stockés dans le même AIP.

OCFL

[Oxford Common File Layout](#) est une spécification. Selon [ocfl.io](#) :

This Oxford Common File Layout (OCFL) specification describes an application-independent approach to the storage of digital information in a structured, transparent, and predictable manner. It is designed to promote long-term object management best practices within digital repositories.

Toujours selon [ocfl.io](#), ses bénéfices :

- **Completeness**, so that a repository can be rebuilt from the files it stores
- **Parsability**, both by humans and machines, to ensure content can be understood in the absence of original software

- **Robustness** against errors, corruption, and migration between storage technologies
- **Versioning**, so repositories can make changes to objects allowing their history to persist
- **Storage diversity**, to ensure content can be stored on diverse storage infrastructures including conventional filesystems and cloud object stores

En pratique, OCFL définit:

- la hiérarchie de stockage
 - i.e. l'organisation de paquets sur le media
- le format des paquets d'information
 - i.e. la structure des paquets eux-mêmes

La hiérarchie de stockage OCFL

Elle doit être déterministe. Dans le cas de Fedora Commons, la règle déterministe pour calculer le chemin des paquets est la suivante:

```
hash := sha256( fedoraId )
chemin := hash[0:3]/hash[3:6]/hash[6:9]/hash
```

Exemple:

```
fedoraId = records/acv/dossiers/D1
chemin = 536/2a8/fe0/5362a8fe0af7fd17596d076f943f179...
```

La structure des paquets OCFL

- un répertoire par version v1, v2, v3, ...
- dans chaque répertoire de version, il y a:
 - un fichier manifeste, *manifest.json*, comprenant:
 - listing des fichiers (avec leur chemin) composant la version courrante (ainsi que pour les versions antérieures)
 - le checksum correspondant à chaque fichier
 - au début du manifeste, le checksum permet de localiser tout fichier, même si il stocké dans une version précédent
 - un répertoire "content" fichiers ajoutés ou modifiés dans la version courrante

Exemple:

```
[object root]
├── 0=ocfl_object_1.0
├── inventory.json
├── inventory.json.sha512
├── v1
│   ├── inventory.json
│   ├── inventory.json.sha512
│   └── content
│       ├── empty.txt
│       ├── foo
│       │   └── bar.xml
│       └── image.tiff
├── v2
│   ├── inventory.json
│   ├── inventory.json.sha512
│   └── content
│       ├── foo
│       │   └── bar.xml
└── .
OCFL package
```

Démo : exemple d'un dossier dans Fedora Commons.

