



Générateur de SIP - Mode d'emploi

Version du 20/03/2018

*La solution logicielle VITAM est publiée sous la licence CeCILL 2.1 ;
La documentation associée (comprenant le présent document) est publiée sous Licence
Ouverte V2.0.*

Valeurs Immatérielles Transférées aux Archives pour Mémoire

Organisation de la présentation

1/ Présentation du générateur

2/ Fonctionnement simple du générateur

3/ Fonctionnement avancé du générateur

- 3.1. Constitution de jeux de tests avancés et paramétrage du bordereau
- 3.2 Paramétrage du SIP en utilisant des fichiers .json
- 3.3. Rassemblement des fichiers constituant des représentations différentes d'une même unité archivistique
- 3.4. Gestion des unités archivistiques complexes, avec fichiers rattachés et également arborescence
- 3.5. Rattachement d'une unité archivistique à une unité archivistique déjà versée dans le SAE

Annexe : Rappels sur le SEDA 2.0.

1/ Présentation du générateur

Objectifs de l'outil

- Faciliter la génération des jeux de tests pour les membres de l'équipe Vitam, les ministères porteurs et les partenaires
- Générer automatiquement à partir d'une arborescence de fichiers des SIP pouvant être pris en charge par la solution logicielle Vitam :
 - Compatibles NF Z44-022 et standard SEDA v2.0 (conforme schéma .xsd du SEDA 2.0.)
 - Conformes au document de spécification des SIP propre à la solution logicielle Vitam
 - Sans recours obligatoire à un éditeur xml pour créer le bordereau
 - Permettant de générer rapidement un SIP avec des milliers d'unités d'archives et des milliers de fichiers
 - De manière fiable (ne pas copier à la main l'empreinte)

Alimentation du bordereau (1)

- Pour les fichiers

- Calcul de l'empreinte (avec l'algorithme de hachage paramétrable, par défaut SHA-512) et écriture dans le bordereau
- Calcul de la taille du fichier et écriture dans le bordereau
- Récupération dans le bordereau du nom d'origine du fichier et de sa date de dernière modification (FileInfo)
- Définition d'un usage par défaut pour les fichiers (original numérique = BinaryMaster)
- Identification du format du fichier en utilisant l'outil Siegfried
- Rassemblement dans un même groupe d'objets quand 2 fichiers constituent 2 représentations (ex. une version de conservation et une version de diffusion) d'une même unité archivistique
 - *Voir la partie fonctionnement avancée du générateur pour plus de détails*

Alimentation du bordereau (2)

- Pour l'arborescence d'unités de description
 - Création d'une arborescence d'unités archivistiques reprenant l'arborescence du système de fichiers
 - Création des liens entre unités archivistiques et fichiers numériques
 - Indication de niveaux de description par défaut : RecordGrp pour les répertoires, Item pour les fichiers
 - Alimentation automatique du bordereau avec les informations suivantes récupérées de l'arborescence de fichiers
 - *Dates (date de modification des fichiers, dates extrêmes des répertoires)*
 - *Titre (répertoire, fichier)*
 - Transformation des raccourcis vers des répertoires et des fichiers en liens vers ces répertoires et fichiers
 - Gestion des unités archivistiques complexes, avec fichiers rattachés et également arborescence (ex. message électronique)
 - Rattachement d'une nouvelle unité archivistique à une unité archivistique déjà présente dans le SAE
 - *Voir la partie fonctionnement avancée du générateur pour plus de détails*

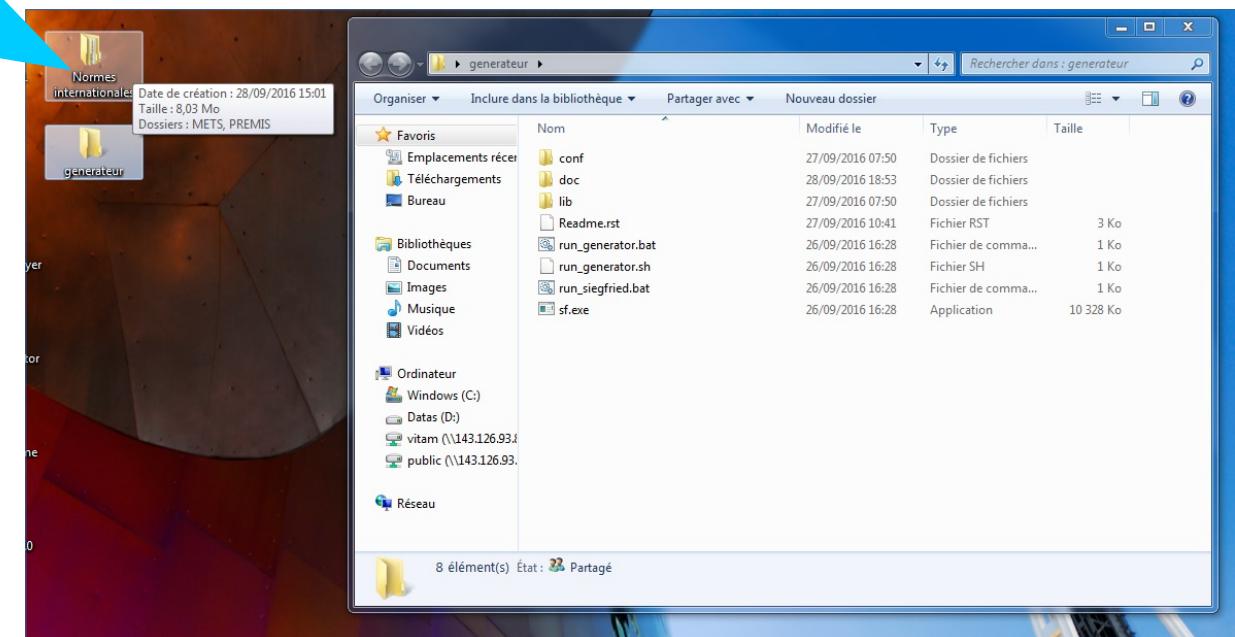
Paramétrage de l'alimentation du SIP

- **Paramétrage de l'en-tête du bordereau**
 - Alimentation de certains champs : identifiant du SIP, commentaire, contrat d'entrée, identifiant du service effectuant le transfert, identifiant du service d'archives, listes de codes, identifiant du service producteur, identifiant du service versant
 - Définition des types de fichiers que l'utilisateur ne souhaite pas mettre dans le SIP : par exemple Thumbs.db, *.vcf
- **Paramétrage des métadonnées des répertoires**
 - Alimentation possible de tous les champs existants dans l'ontologie définie dans le SEDA 2.0.
- **Autres paramétrages possibles :**
 - Rejet des fichiers non reconnus par l'outil d'identification des formats Siegfried. Par défaut, ces fichiers sont inclus dans le SIP
 - Rejet des fichiers ayant une extension contenant un caractère « URL-encoded » de type « + ». Par défaut, ces fichiers sont inclus dans le SIP
 - Choix d'un autre algorithme de calcul d'empreinte des fichiers que l'algorithme par défaut (SHA-512)

2/ Fonctionnement simple du générateur

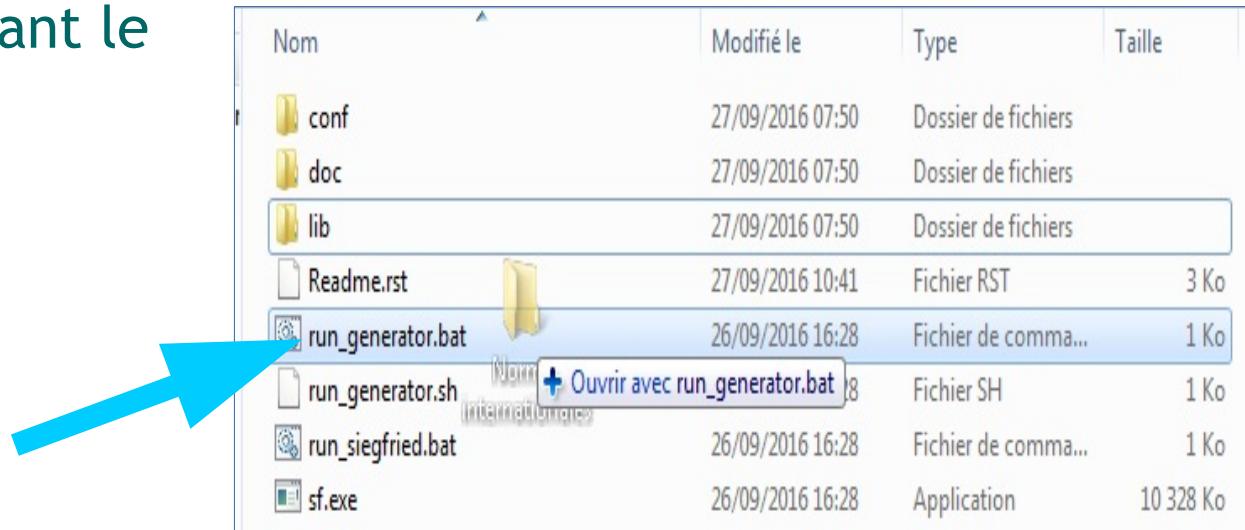
Étape 1 - Identifier le dossier à archiver

- Choisir dans le navigateur la racine de l'arborescence à intégrer dans le SIP
- Attention :
 - Si un seul fichier doit être archivé, il convient de l'enregistrer dans un répertoire
 - Le répertoire ne doit pas être enregistré dans le même répertoire que le générateur

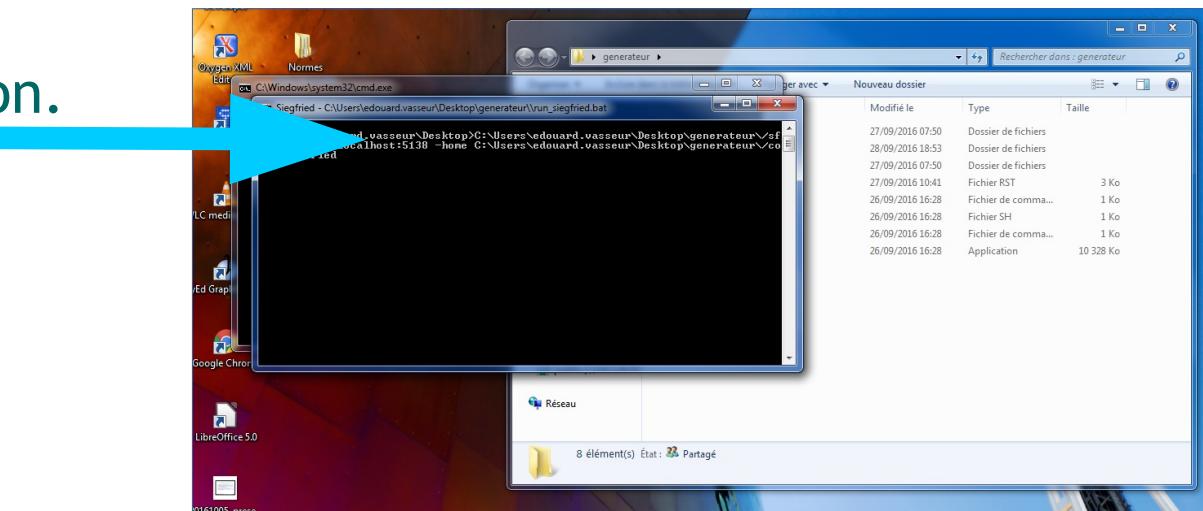


Étape 2 - Lancer l'opération de création du SIP

- Ouvrir le répertoire contenant le générateur
- Faire glisser le dossier sélectionné vers le fichier « run_generator.bat »

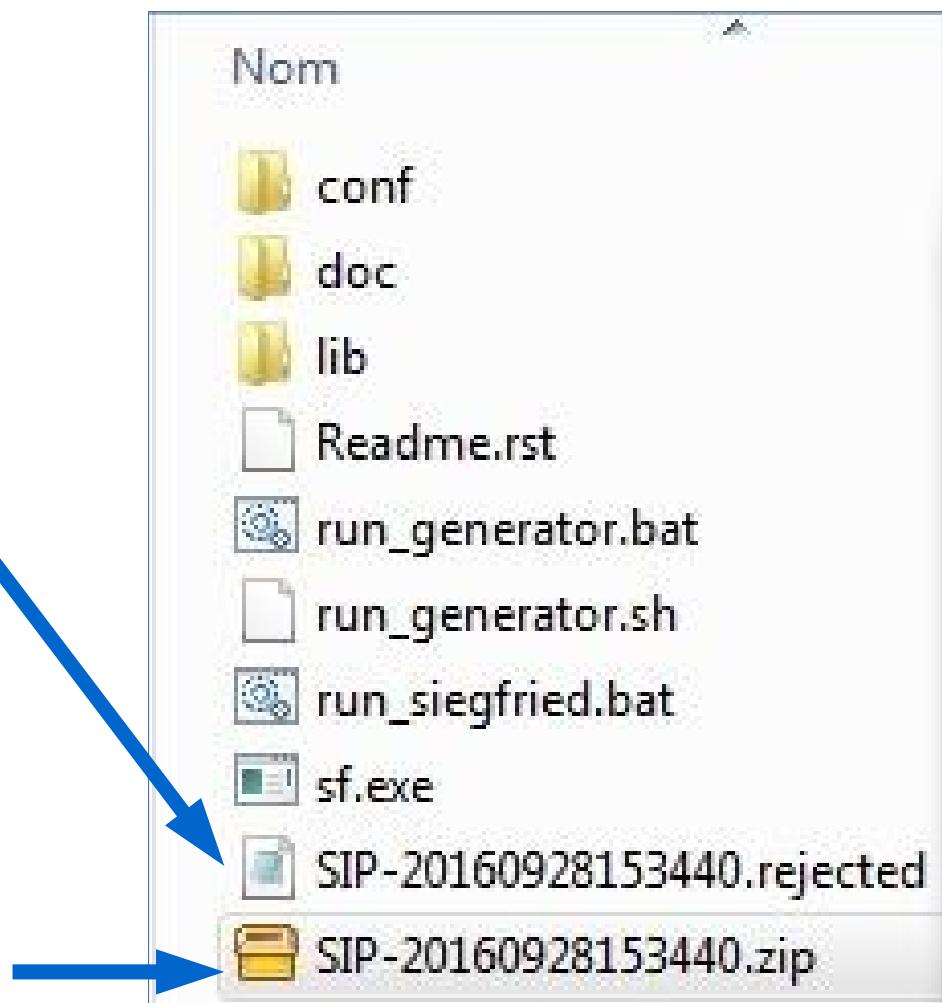


- Deux consoles s'ouvrent. La 2^e console trace les erreurs survenues pendant l'opération. Elle se ferme à la fin de l'opération et tapant sur la touche « entrée »



Étape 3 - où retrouver le SIP généré ?

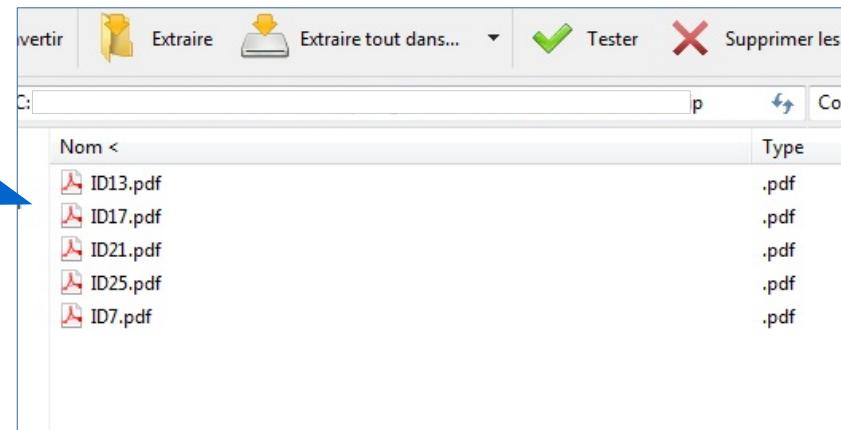
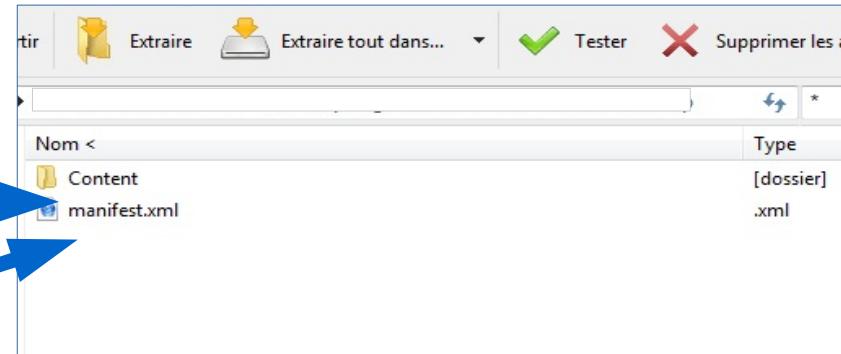
- 2 fichiers ont été créés dans le répertoire contenant le générateur :
 - Le 1^{er} « .rejected » relève toutes les erreurs survenues lors de la création du SIP
 - *S'il n'y a pas d'erreur la taille du fichier est 0 Ko*
 - Le 2^e est le SIP à proprement parler avec une extension .zip
- Le SIP peut être renommé



Étape 4 - Consulter le SIP

- Ouvrir le zip et constater qu'il contient bien :

- Un fichier manifest.xml qui correspond au bordereau
- Un répertoire « content » contenant l'ensemble des fichiers de l'arborescence, renommés et à plat



NB : le bordereau peut être extrait pour enrichissement (ex. avec un éditeur xml) et réinjecté dans le SIP ensuite (en supprimant la version d'origine)

Le SIP est prêt !

3/ Fonctionnement avancé du générateur

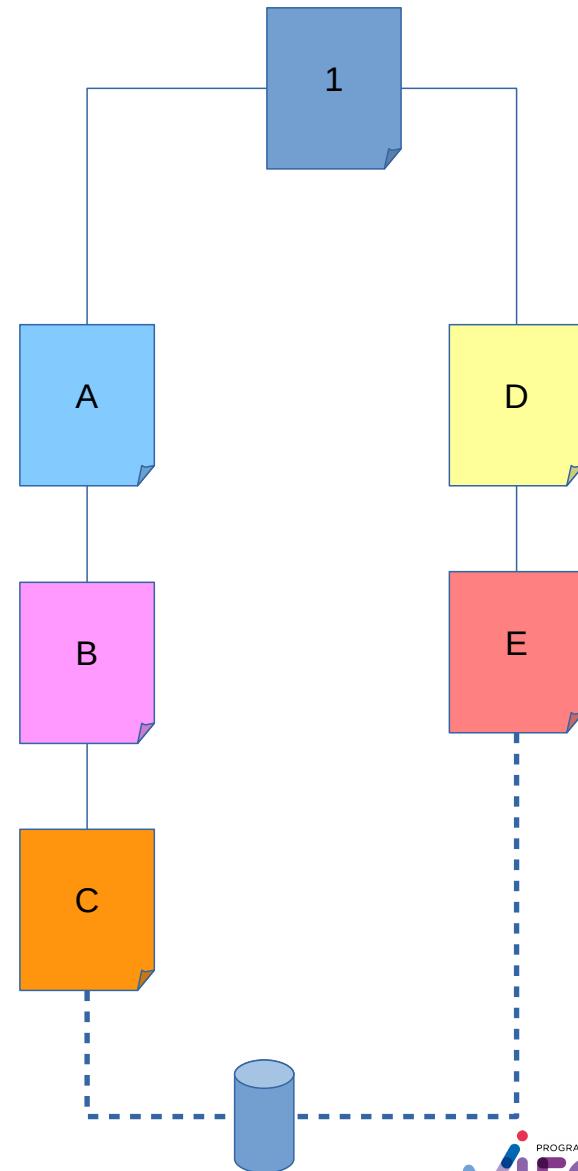
3.1. Constitution de jeux de tests avancés

Constituer des jeux de tests « avancés »

Organiser l'arborescence en fonction du comportement attendu

Ex. : avoir un fichier à plusieurs endroits dans l'arborescence

- *Créer 1 répertoire (1) comprenant 2 sous-répertoires (A et D)*
- *Prendre le 1^{er} sous-répertoire (A) et y mettre 1 seul sous-sous-répertoire (B), dans lequel on créé un sous-sous-sous-répertoire (C) dans lequel on positionne le ou les fichiers*
- *Prendre le 2^e sous-répertoire (D) et y mettre 1 seul sous-sous-répertoire (E), dans lequel on positionne le ou les mêmes fichiers que dans le sous-sous-sous-répertoire n°1*



3.2. Paramétrage du SIP en utilisant des fichiers .json

Paramétrage de l'en-tête du bordereau (1)

- Il est possible de paramétrer l'en-tête du bordereau en utilisant le fichier `ArchiveTransferConfig.json` enregistré dans le répertoire `conf`



Nom	Modifié le	Type
siegfried	27/09/2016 07:50	Dossier de fichiers
ArchiveTransferConfig.json	28/09/2016 11:17	JSON file
logback.xml	26/09/2016 16:28	XML Document
playbook_BinaryDataObject.json	26/09/2016 16:28	JSON file

Paramétrage de l'en-tête du bordereau (2)

- Copier ce fichier dans le répertoire racine de l'arborescence
- Ouvrir le fichier .json avec un éditeur xml ou un éditeur de texte comme Notepad++
- Modifier les valeurs des champs voulus en respectant la ponctuation (guillemets) :
 - Comment : chaîne de caractère décrivant le transfert
 - MessageIdentifier : identifiant du transfert
 - ArchivalAgreement : contrat d'entrée utilisé
 - CodeListVersions : listes de codes utilisés dans le bordereau
 - ArchivalAgency : identifiant du service d'archives destiné à recevoir le transfert
 - TransferringAgency : identifiant de l'opérateur de transfert à l'origine du transfert
 - ManagementMetadata.OriginatingAgencyIdentifier : identifiant du service producteur
 - ManagementMetadata.SubmissionAgencyIdentifier : identifiant du service versant
- Indiquer les types de fichiers que l'utilisateur souhaite exclure du SIP (champ ignore_patterns)
- Sauvegarder les modifications

Paramétrage de l'en-tête du bordereau (3)

```
{  
    "Comment" : "2eme SIP",  
    "MessageIdentifier" : "MessageIdentifier0",  
    "ArchivalAgreement" : "ArchivalAgreement0",  
    "CodeListVersions" : {  
        "ReplyCodeListVersion" : "ReplyCodeListVersion0",  
        "MessageDigestAlgorithmCodeListVersion" : "MessageDigestAlgorithmCodeListVersion0",  
        "MimeTypeCodeListVersion" : "MimeTypeCodeListVersion0",  
        "EncodingCodeListVersion" : "EncodingCodeListVersion0",  
        "FileFormatCodeListVersion" : "FileFormatCodeListVersion0",  
        "CompressionAlgorithmCodeListVersion" : "CompressionAlgorithmCodeListVersion0",  
        "DataObjectVersionCodeListVersion" : "DataObjectVersionCodeListVersion0",  
        "StorageRuleCodeListVersion" : "StorageRuleCodeListVersion0",  
        "AppraisalRuleCodeListVersion" : "AppraisalRuleCodeListVersion0",  
        "AccessRuleCodeListVersion" : "AccessRuleCodeListVersion0",  
        "DisseminationRuleCodeListVersion" : "DisseminationRuleCodeListVersion0",  
        "ReuseRuleCodeListVersion" : "ReuseRuleCodeListVersion0",  
        "ClassificationRuleCodeListVersion" : "ClassificationRuleCodeListVersion0",  
        "AuthorizationReasonCodeListVersion" : "AuthorizationReasonCodeListVersion0",  
        "RelationshipCodeListVersion" : "RelationshipCodeListVersion0"  
    },  
    "ArchivalAgency" : {  
        "Identifier" : "Identifier4"  
    },  
    "TransferringAgency" : {  
        "Identifier" : "Identifier5"  
    },  
    "ManagementMetadata.OriginatingAgencyIdentifier" : "Service_producteur",  
    "ManagementMetadataSubmissionAgencyIdentifier" : "Service_versant",  
    "ignore_patterns" : ["Thumbs.db","pagefile.sys"]  
}
```

Paramétrage des métadonnées d'un répertoire (1)

- Il est possible de paramétriser l'indexation du répertoire en utilisant le fichier `ArchiveUnitMetadata.json` enregistré dans le répertoire doc



Nom	Modifié le	Type
Arborescence.rst	27/09/2016 11:05	Fichier RST
Architecture_logicielle.rst	27/09/2016 11:07	Fichier RST
ArchiveUnitMetadata.json	26/09/2016 16:28	JSON file
Configuration.rst	27/09/2016 11:47	Fichier RST
fichiermetatadavaleurdescriptionlevel.txt	28/09/2016 15:53	Document texte

Paramétrage des métadonnées d'un répertoire (2)

- Copier ce fichier dans le répertoire que vous voulez indexer
- Ouvrir le fichier .json avec un éditeur xml ou un éditeur de texte comme Notepad++
- Identifier les champs que vous voulez enrichir parmi la liste de tous les champs disponibles dans l'ontologie fournie par le SEDA
- Modifier les valeurs des champs qui vous intéressent en respectant la ponctuation (guillemets)
 - Voir slides correspondantes pour la liste des champs
 - Les champs DescriptionLevel et Title sont obligatoires
- Supprimer les champs que vous ne souhaitez pas voir apparaître dans l'indexation du répertoire
 - Voir slide correspondante pour les points d'attention
- Sauvegarder les modifications

Paramétrage des métadonnées d'un répertoire (3)

```
{ "Content": { "DescriptionLevel": "RECORD_GRP", "Title": [ { "Value": "Titre francais", "Lang": "fr" }, { "Value": "English title", "Lang": "en" } ], "FilePlanPosition": "Valeur de filePlanPosition", "SystemId": "Valeur de SystemID)", "OriginatingSystemId": "Valeur de OriginatingSystemId", "ArchivalAgencyArchiveUnitIdentifier": "Valeur de archivalAgencyArchiveUnitIdentifier", "OriginatingAgencyArchiveUnitIdentifier": "Valeur de originatingAgencyArchiveUnitIdentifier", "TransferringAgencyArchiveUnitIdentifier": "Valeur de transferringAgencyArchiveUnitIdentifier", "Description": [ { "Value": "Description francaise", "Lang": "fr" }, { "Value": "English Description", "Lang": "en" } ], "Type": { "Value": "Valeur du type", "Lang": "fr" }, "DocumentType": { "Value": "fr" }, "Language": "FR", "DescriptionLanguage": "FR", "Status": "Valeur de Status", "Version": "Valeur de version", "Tag": [ "XML Tag 1 (de type xml:token)", "XML Tag 2 (de type xml:token)" ], "Coverage": { "Spatial": [ { "Value": "Valeur de Spatial", "Lang": "fr" } ] } }}
```

Paramétrage des métadonnées d'un répertoire (4)

- Champs modifiables
 - DescriptionLevel : niveau de description
 - *Voir slide correspondante pour la liste des valeurs*
 - Title : titre
 - FilePlanPosition : position dans le plan de classement
 - SystemId : identifiant fourni par le SAE
 - OriginatingSystemId : identifiant fourni par le système de production
 - ArchivalAgencyArchiveUnitIdentifier : identifiant fourni par le service d'archives (ex. cote)
 - OriginatingAgencyArchiveUnitIdentifier : identifiant fourni par le service producteur
 - TransferringAgencyArchiveUnitIdentifier : identifiant fourni par le service versant
 - Description : présentation détaillée du contenu
 - Type : type d'information au sens de l'OAIS

Paramétrage des métadonnées d'un répertoire (5)

- **Champs modifiables**

- DocumentType : type de document
- Language : langue des archives
- DescriptionLanguage : langue des descriptions
- Status : statut du document
- Version : version du document
- Tag : indexation
- Coverage : couverture (Spatial : géographique ; Temporal : chronologique ; Jurisdictional : administrative)
- OriginatingAgency : identifiant du service producteur
- SubmissionAgency : identifiant du service versant
- Writer : rédacteur du document
- Source : pour les documents numérisés, référence au document source
- Event : description d'un événement survenu pendant le cycle de vie du document
- Gps : coordonnées géographiques définies par l'utilisateur

Paramétrage des métadonnées d'un répertoire (6)

- Valeurs possibles pour le champ Niveau de description
 - Fonds : mettre FONDS
 - Sous-fonds : mettre SUBFONDS
 - Classe : mettre CLASS
 - Collection : mettre COLLECTION
 - Série : mettre SERIES
 - Sous-série : mettre SUBSERIES
 - Groupe d'archives : mettre RECORD_GRP
 - Sous-groupe d'archives : mettre SUB_GRP
 - Dossier : mettre FILE
 - Pièce : mettre ITEM

Paramétrage des métadonnées d'un répertoire (7)

- Points d'attention sur la suppression des champs
 - Toujours mettre une virgule à la fin de chaque champ, sauf à la fin du dernier champ
 - les champs Title et Description peuvent avoir plusieurs valeurs. Attention dans ce cas à la syntaxe du fichier lors de la suppression d'un champ, notamment au nombre de caractères spéciaux délimitant le champ (voir diapositive suivante)

Paramétrage des métadonnées d'un répertoire (8)

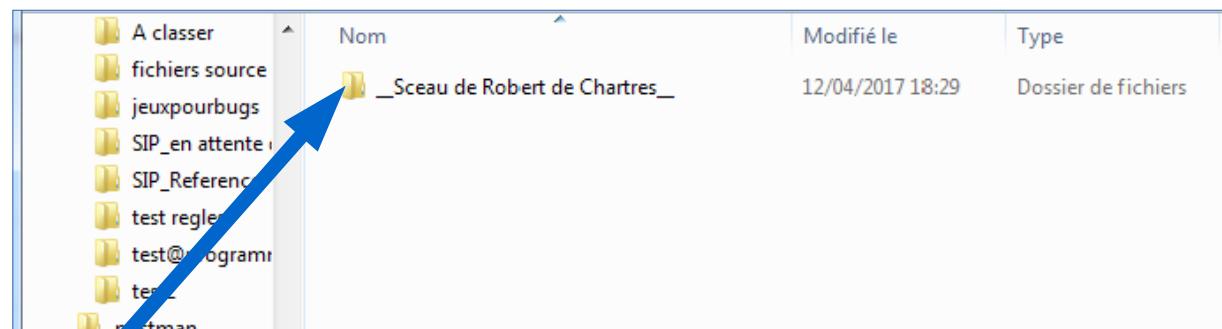
- Lors de la suppression de champs, faire attention aux caractères spéciaux les délimitant : {}, []

```
{ "Content" : {  
    "DescriptionLevel" : "FILE",  
    "Title" : [ {  
        "Value" : "Documentation sur le standard METS  
    } ],  
    "FilePlanPosition" : "3.1.1.",  
    "OriginatingSystemId" : "24561",  
    "Description" : [ {  
        "Value" : "Documentation récupérée sur le site de la Bibliothèque du Congrès"  
    } ],  
    "Language" : "EN",  
    "OriginatingAgency" : {  
        "Identifier" : {  
            "Value" : "CodeVitam"  
        }  
    },  
    "StartDate" : "2016-09-26T15:34:08.284+02:00",  
    "EndDate" : "2016-09-26T15:34:08.284+02:00"  
}
```

3.3. Rassemblement des fichiers constituant des représentations différentes d'une même unité archivistique

Rassemblement des fichiers constituant des représentations différentes d'une même unité archivistique (1)

- Exemple d'une photographie présente dans 2 formats : un format .tiff de conservation et un format .jpeg de diffusion
- Étape 1 : enregistrer les 2 fichiers dans un même répertoire
- Étape 2 : renommer ce répertoire en ajoutant comme préfixe 2 Underscores (_) et en ajoutant comme suffixe 2 Underscores (_)



Rassemblement des fichiers constituant des représentations différentes d'une même unité archivistique (2)

- Étape 3 : préfixer le nom de chaque fichier de la manière suivante :
 - 2 Underscores (_) + le type d'usage + 1 Underscore (_) + le numéro de version
 - Exemple : __BinaryMaster_1_mon nom de fichier.extension
 - Les types d'usages possibles sont :
 - *BinaryMaster* = *original numérique*
 - *Dissemination* = *diffusion*
 - *Thumbnail* = *vignette*
 - *TextContent* = *texte brut*
- Étape 4 : lancer la création du SIP



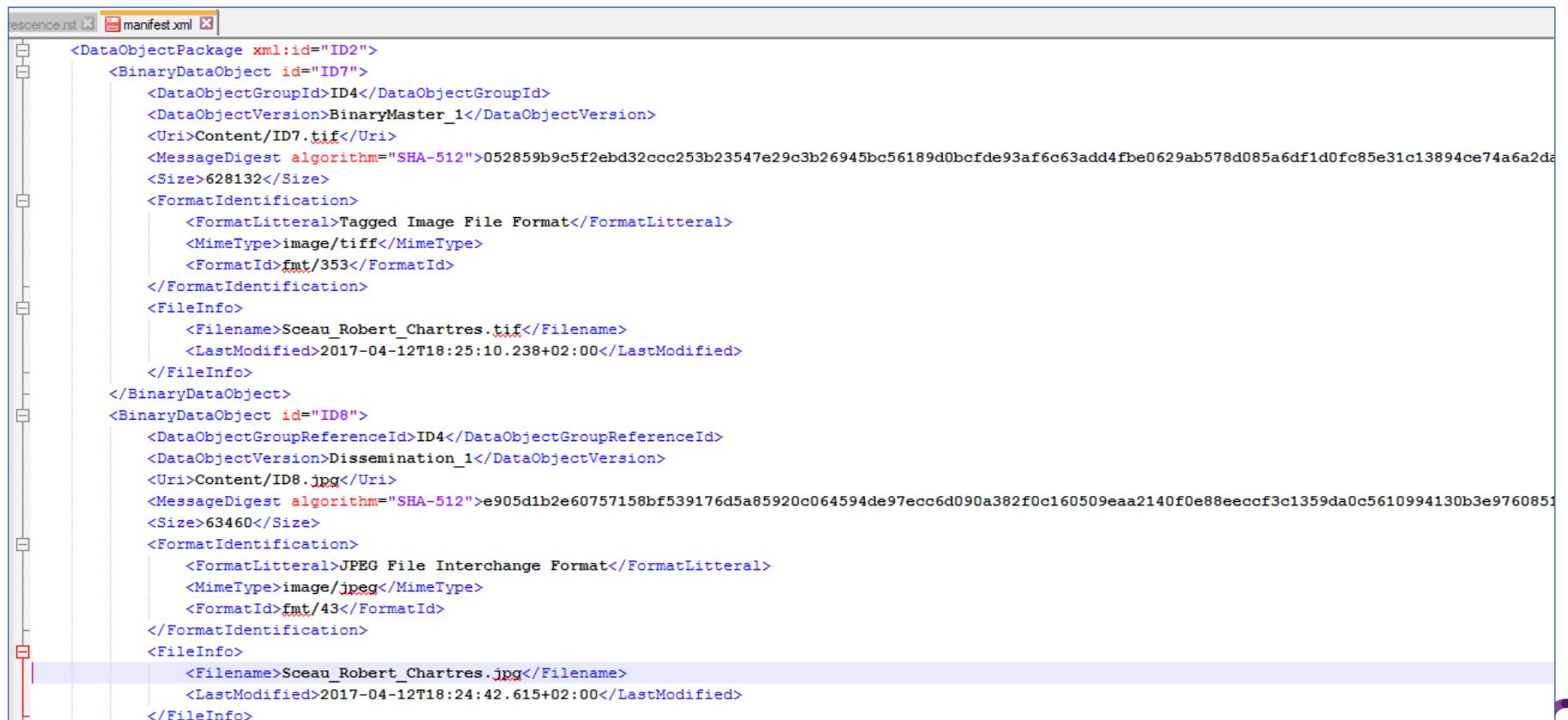
Rassemblement des fichiers constituant des représentations différentes d'une même unité archivistique (3)

- L'unité de description est unique (ici ArchiveUnit id= « ID5 »)
- Elle a pour titre le nom du répertoire (ici Sceau de Robert de Chartres)), sans les Underscores

```
</ArchiveUnit>
<ArchiveUnit id="ID5">
    <Content>
        <DescriptionLevel>Item</DescriptionLevel>
        <Title>Sceau de Robert de Chartres</Title>
        <Description>C:\Users\edouard.vasseur\Desktop\test GOT\__Sceau de Robert de Chartres__</Description>
        <TransactedDate>2017-04-12T18:25:10</TransactedDate>
    </Content>
    <DataObjectReference>
        <DataObjectGroupId>ID4</DataObjectGroupId>
    </DataObjectReference>
</ArchiveUnit>
</DescriptiveMetadata>
<ManagementMetadata>
    <OriginatingAgencyIdentifier>Service_producteur</OriginatingAgencyIdentifier>
    <SubmissionAgencyIdentifier>Service_versant</SubmissionAgencyIdentifier>
</ManagementMetadata>
</DataObjectPackage>
<ArchivalAgency>
    <Identifier>Identifier4</Identifier>
</ArchivalAgency>
<TransferringAgency>
    <Identifier>Identifier5</Identifier>
</TransferringAgency>
</ArchiveTransfer>
```

Rassemblement des fichiers constituant des représentations différentes d'une même unité archivistique (4)

- Les 2 objets sont regroupés dans un même groupe d'objet dans le bordereau (ici DataObjectPackage xml id = « id2 »)
- Le nom du fichier est le nom d'origine, sans le préfixe rajouté (ici Sceau_Robert_Chartres.tif)
- La balise DataObjectVersion indique l'usage de chaque fichier (ici id7 est le BinaryMaster)



```
<DataObjectPackage xml:id="ID2">
    <BinaryDataObject id="ID7">
        <DataObjectGroupId>ID4</DataObjectGroupId>
        <DataObjectVersion>BinaryMaster_1</DataObjectVersion>
        <Uri>Content/ID7.tif</Uri>
        <MessageDigest algorithm="SHA-512">052859b9c5f2ebd32ccc253b23547e29c3b26945bc56189d0bcfde93af6c63add4fbe0629ab578d085a6df1d0fc85e31c13894ce74a6a2da</MessageDigest>
        <Size>628132</Size>
        <FormatIdentification>
            <FormatLitteral>Tagged Image File Format</FormatLitteral>
            <MimeType>image/tiff</MimeType>
            <FormatId>fmt/353</FormatId>
        </FormatIdentification>
        <FileInfo>
            <Filename>Sceau_Robert_Chartres.tif</Filename>
            <LastModified>2017-04-12T18:25:10.238+02:00</LastModified>
        </FileInfo>
    </BinaryDataObject>
    <BinaryDataObject id="ID8">
        <DataObjectGroupId>ID4</DataObjectGroupId>
        <DataObjectVersion>Dissemination_1</DataObjectVersion>
        <Uri>Content/ID8.jpg</Uri>
        <MessageDigest algorithm="SHA-512">e905d1b2e60757158bf539176d5a85920c064594de97ecc6d090a382f0c160509eaa2140f0e88eeccf3c1359da0c5610994130b3e9760851</MessageDigest>
        <Size>63460</Size>
        <FormatIdentification>
            <FormatLitteral>JPEG File Interchange Format</FormatLitteral>
            <MimeType>image/jpeg</MimeType>
            <FormatId>fmt/43</FormatId>
        </FormatIdentification>
        <FileInfo>
            <Filename>Sceau_Robert_Chartres.jpg</Filename>
            <LastModified>2017-04-12T18:24:42.615+02:00</LastModified>
        </FileInfo>
    </BinaryDataObject>

```

3.4. Gestion des unités archivistiques complexes, avec fichiers rattachés et également arborescence

Gestion des unités archivistiques complexes, avec fichiers rattachés et également arborescence (1)

- Exemple d'un registre numérisé :
 - À un registre correspond une unité archivistique
 - À cette unité archivistique peut être associé un objet, un fichier multipages
 - Mais cette unité archivistique peut également avoir pour filles d'autres unités archivistiques correspondant à chaque page du registre => elle a donc également une arborescence d'unités archivistiques « filles »
- Comment utiliser le générateur ?
 - Utiliser la méthode décrite pour rassembler des fichiers constituant des représentations différentes d'une même unité archivistique
 - Enregistrer dans le répertoire qui a pour préfixe et suffixe les 2 Underscores l'arborescence de répertoires et de fichiers à intégrer

Gestion des unités archivistiques complexes, avec fichiers rattachés et également arborescence (2)

- Étape 1 : rajouter un répertoire dans le répertoire précédentement créé
- Étape 2 : lancer le générateur



Gestion des unités archivistiques complexes, avec fichiers rattachés et également arborescence (3)

- Dans le manifeste, le répertoire inséré dans le répertoire précédemment créé est devenu une unité archivistique fille de l'unité archivistique correspondant au répertoire

```
<!--
</ArchiveUnit>
<ArchiveUnit id="ID5">
    <Content>
        <DescriptionLevel>Item</DescriptionLevel>
        <Title>Sceau de Robert de Chartres</Title>
        <Description>C:\Users\edouard.vasseur\Desktop\test GOT\__ Sceau de Robert de Chartres __</Description>
        <TransactedDate>2017-04-12T18:25:10</TransactedDate>
    </Content>
    <DataObjectReference>
        <DataObjectGroupId>ID4</DataObjectGroupId>
    </DataObjectReference>
    <ArchiveUnit id="ID8">
        <ArchiveUnitRefId>ID7</ArchiveUnitRefId>
    </ArchiveUnit>
</ArchiveUnit>
<ArchiveUnit id="ID7">
    <Content>
        <DescriptionLevel>RecordGrp</DescriptionLevel>
        <Title>autre sceau</Title>
        <Description>C:\Users\edouard.vasseur\Desktop\test GOT\__ Sceau de Robert de Chartres __\autre sceau</Description>
        <StartDate>2017-04-12T18:24:42</StartDate>
        <EndDate>2017-04-12T18:25:10</EndDate>
    </Content>
    <ArchiveUnit id="ID12">
        <ArchiveUnitRefId>ID11</ArchiveUnitRefId>
    </ArchiveUnit>
```



3.5./ Rattachement d'une unité archivistique à une unité archivistique déjà versée dans le SAE

Rattachement d'une unité archivistique à une unité archivistique déjà présente dans le SAE (1)

- Exemple d'un sous-dossier à rajouter dans un dossier
 - Le dossier constitue une unité archivistique déjà présente dans le SAE
 - Le manifeste décrivant le sous-dossier doit préciser à quelle unité archivistique le rattacher
- Comment utiliser le générateur ?
 - Constituer l'arborescence de répertoire souhaitée pour le SIP
 - Enregistrer un fichier nommé ArchiveUnitMetadata.json dans le répertoire correspondant au sous-dossier et décrire l'unité archivistique déjà versée dans le SAE et à laquelle rattacher le sous-dossier :
 - *Titre*
 - *Niveau de description*
 - *Identifiant de l'unité archivistique dans le SAE (information disponible dans le détail de l'unité archivistique dans l'IHM standard)*
 - Générer le SIP en utilisant la méthode de fonctionnement simple

Rattachement d'une unité archivistique à une unité archivistique déjà présente dans le SAE (2)

- Étape 1 : rajouter le fichier **ArchiveUnitMetadata.json** dans le répertoire



Nom	Modifié le	Type
__Sceau de Robert de Chartres__	14/04/2017 08:29	Dossier de fichiers
ArchiveUnitMetadata.json	09/05/2017 12:14	JSON file

Rattachement d'une unité archivistique à une unité archivistique déjà présente dans le SAE (3)

- Étape 2 : mettre à jour le fichier ArchiveUnitMetadata.json en indiquant le titre, le niveau de description et l'identifiant dans le SAE de l'unité archivistique à laquelle rattacher l'unité présente dans le bordereau
- Étape 3 : générer le SIP



```
ArchiveUnitMetadata.json
1
2
3
4
5     "VitamFather": [
6         {
7             "id": "aeaqaaaaaihnjdarabzvqak3m2b3qoqaaaq",
8             "Title": [
9                 {
10                     "value": "Sénat",
11                     "lang": "fr"
12                 }
13             ],
14         }
15     ]
16 }
```

NB : pour simplifier la présentation, seule la section du fichier ArchiveUnitMetadata.json permettant de déclarer l'unité archivistique présente dans le SAE est présentée

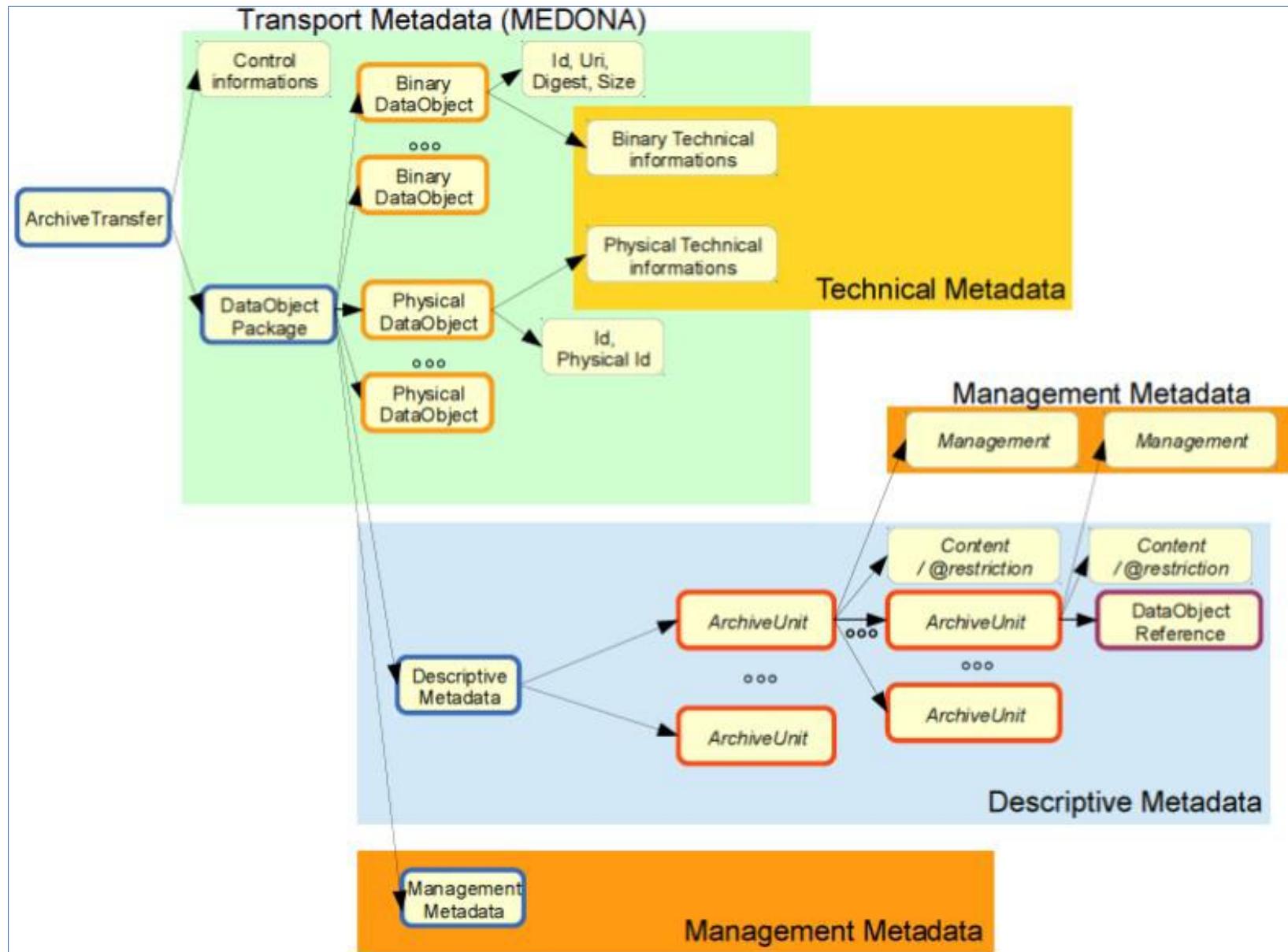
Rattachement d'une unité archivistique à une unité archivistique déjà présente dans le SAE (4)

- Dans le manifeste :
 - l'unité archivistique présente dans le système a été créée
 - Elle déclare pour fille l'unité archivistique présente dans le manifeste et à laquelle on souhaite la rattacher dans le SAE

```
<DescriptiveMetadata>
  <ArchiveUnit id="ID4">
    <Management>
      <UpdateOperation>
        <SystemId>aeaqaaaaaihnjdarabzvqak3m2b3qoqaaaq</SystemId>
      </UpdateOperation>
    </Management>
    <Content>
      <DescriptionLevel>RecordGrp</DescriptionLevel>
      <Title xml:lang="fr">Sénat</Title>
    </Content>
    <ArchiveUnit id="ID20">
      <ArchiveUnitRefId>ID3</ArchiveUnitRefId>
    </ArchiveUnit>
  </ArchiveUnit>
  <ArchiveUnit id="ID3">
    <Content>
      <DescriptionLevel>RecordGrp</DescriptionLevel>
      <Title>test_rattachement</Title>
      <Description>C:\Users\edouard.vasseur\Desktop\test_rattachement</Description>
      <StartDate>2017-04-12T18:25:10</StartDate>
      <EndDate>2017-04-12T18:25:10</EndDate>
    </Content>
    <ArchiveUnit id="ID7">
      <ArchiveUnitRefId>ID6</ArchiveUnitRefId>
    </ArchiveUnit>
  </ArchiveUnit>
```

Annexe sur le SEDA

Structuration du bordereau (message de type ArchiveTransfer)



Vocabulaire

- <DataObjectPackage> : englobe tous les objets et leurs MD de description et de gestion
- <BinaryDataObject> : objet correspondant à des fichiers binaires
- <PhysicalDataObject> : objet correspondant à quelque chose de physique (un carton, un cd-rom, etc.)
- <DataObjectGroup> : groupe d'objets données
- <DataObjectGroupId> : identifiant d'un groupe d'objets données
- <DataObjectVersion> : usage/version de l'ensemble intellectuel
- <Uri> : chemin permettant d'accéder au fichier
- <MessageDigest> : empreinte du fichier
- <Descriptive MetaData> : métadonnées de description
- <Management Metadata> : métadonnées de gestion
- <ArchiveUnit> : unité d'archives
- <Content> : contenu de l'unité d'archives
- <DataObjectReference> : référence interne à un objet-donnée ou à un groupe d'objet-données