



Profils d'unité archivistique

Date	Version
25/10/2018	3.0 (Release 8)

État du document

En projet Vérifié Validé

Maîtrise du document

Responsabilité	Nom	Entité	Date
Rédaction	MVI	Équipe Vitam	04/04/2018
Vérification	Équipe	Équipe Vitam	
Validation		Équipe Vitam	

Suivi des modifications

Version	Date	Auteur	Modifications
0.1	04/04/2018	MVI	Initialisation
1.0	05/06/2018	EVA	Relecture
1.1	06/06/18	MRE	Relecture
2.0	15/06/2018	MRE	Finalisation du document pour publication de la Release 7
2.1	10/09/2018	MVI	<p>Mise à jour pour tenir compte des fonctionnalités mises en œuvre pendant la Release 8 :</p> <ul style="list-style-type: none"> • Section 2.1 « Qu'est-ce qu'un profil d'unité archivistique? » : ajout d'un schéma. • Section 3.2 « Dans le Standard d'échange des données pour archivage (SEDA) » : ajout d'un tableau. • Section 4.1 « Administration d'un référentiel des profils d'unité archivistique » : le nom de la notice descriptive n'est plus unique dans le système (sections 4.1.1 et 4.1.3) ; précisions sur les contrôles effectués par la solution logicielle Vitam lors de l'ajout d'un schéma de contrôle à une notice descriptive et des motifs d'échec de cette opération (section 4.1.2) ; service de contrôle sur le caractère actif ou inactif de la notice descriptive lors d'un transfert d'archives implémenté (section 4.1.3). • Section 4.2 « Processus d'entrée » : précision sur les contrôles effectués lors du transfert d'unités archivistiques déclarant un profil d'unité archivistique. • Section 4.3 « Accès » : refonte de cette partie ; ajout des filtres sur les droits de modifications attribuables dans un contrat d'accès (section 4.3.2.1) ; précisions sur les contrôles effectués par la solution logicielle Vitam lors de la mise à jour d'une unité archivistique déclarant un profil d'unité archivistique (section 4.3.2.2). • Section 5 « Étapes d'élaboration d'un profil d'unité archivistique et de son schéma de contrôle » : refonte complète de cette partie : réorganisation et ajouts de sections. • Section 6 « Conseils de mise en œuvre » : ajouts de conseils sur les besoins et l'élaboration d'un profil

			d'unité archivistique. <ul style="list-style-type: none"> • Annexe 1 « Schéma de contrôle par défaut dans la solution logicielle Vitam (état au 14/09/2018) » : mise à jour du schéma. • Annexe 3 « Types JSON » : ajout de l'annexe.
2.2	30/09/2018	MRE	Relecture du document
3.0	25/10/2018	MRE	Finalisation du document pour publication de la Release 8

Documents de référence

Document	Date de la version	Remarques
NF Z44022 – MEDONA - Modélisation des données pour l'archivage	18/01/2014	
Standard d'échange de données pour l'archivage – SEDA – v. 2.1	06/2018	
Vitam - Structuration des <i>Submission Information Package (SIP)</i> – v. 6.0.	25/10/2018	

Licence

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.

Table des matières

1. Résumé.....	5
1.1 Présentation du programme Vitam.....	5
1.2 Présentation du document.....	6
2. Présentation de la notion de profil d’unité archivistique.....	7
2.1. Qu’est-ce qu’un profil d’unité archivistique ?.....	7
2.2. Pourquoi élaborer un profil d’unité archivistique ?.....	8
3. Formalisation des profils d’unité archivistique.....	10
3.1. Dans un fichier propre.....	10
3.2. Dans le Standard d’échange de données pour l’archivage (SEDA).....	10
4. Mécanismes mis en œuvre dans la solution logicielle Vitam.....	12
4.1. Administration d’un référentiel des profils d’unité archivistique.....	12
4.1.1. Import de la notice descriptive du profil d’unité archivistique.....	12
4.1.2. Ajout du profil d’unité archivistique lui-même.....	13
4.1.3. Modification de la notice descriptive d’un profil d’unité archivistique.....	14
4.1.4. Activation / Désactivation d’un profil d’unité archivistique.....	15
4.2. Processus d’entrée.....	16
4.3. Accès.....	17
4.3.1. Consultation des unités archivistiques déclarant un profil d’unité archivistique.....	17
4.3.2. Mise à jour des unités archivistiques.....	17
4.3.2.1. Modification d’un profil d’unité archivistique.....	17
4.3.2.2. Modification des métadonnées d’une unité archivistique déclarant un profil d’unité archivistique.....	18
5. Étapes d’élaboration d’un profil d’unité archivistique et de son schéma de contrôle.....	19
5.1. Phase d’analyse.....	19
5.1.1. Analyse des données.....	19
5.1.2. Modélisation des données.....	20
5.2. Phase de réalisation.....	21
5.2.1. Réalisation d’un bordereau de transfert.....	21
5.2.2. Schématisation des données à contrôler.....	22
5.2.3. Validation d’un schéma de contrôle.....	34
5.2.4. Documentation d’un profil d’unité archivistique.....	35
6. Conseils de mise en œuvre.....	36
Annexe 1 : Schéma de contrôle par défaut dans la solution logicielle Vitam (état au 14/09/2018).....	41
Annexe 2 : Exemple de notice détaillée d’un profil d’unité archivistique.....	77
Annexe 3 : Types JSON.....	78
Annexe 4 : Messages d’erreur.....	82

1. Résumé

Jusqu’à présent, pour la gestion, la conservation, la préservation et la consultation des archives numériques, les acteurs du secteur public étatique ont utilisé des techniques d’archivage classiques, adaptées aux volumes limités dont la prise en charge leur était proposée. Cette situation évolue désormais rapidement et les acteurs du secteur public étatique doivent se mettre en capacité de traiter les volumes croissants d’archives numériques qui doivent être archivés, grâce à un saut technologique.

1.1 Présentation du programme Vitam

Les trois ministères (Armées, Culture et Europe et Affaires étrangères), combinant légalement mission d’archivage définitif et expertise archivistique associée, ont décidé d’unir leurs efforts, sous le pilotage de la Direction interministérielle du numérique et du système d’information et de communication de l’État (DINSIC), pour faire face à ces enjeux. Ils ont décidé de lancer un programme nommé Vitam (Valeurs Immatérielles Transmises aux Archives Pour Mémoire) qui couvre plus précisément les opérations suivantes :

- la conception, la réalisation et la maintenance mutualisées d’une solution logicielle d’archivage électronique de type back-office, permettant la prise en charge, le traitement, la conservation et l’accès aux volumes croissants d’archives (projet de solution logicielle Vitam) ;
- l’intégration par chacun des trois ministères porteurs du Programme de la solution logicielle dans sa plate-forme d’archivage. Ceci implique l’adaptation ou le remplacement des applications métiers existantes des services d’archives pour unifier la gestion et l’accès aux archives, la reprise des données archivées depuis le début des années 1980, la réalisation d’interfaces entre les applications productrices d’archives et la plate-forme d’archivage (projets SAPHIR au MEAE, ADAMANT au MC et ArchiPél au MinArm) ;
- le développement, par un maximum d’acteurs de la sphère publique, de politiques et de plates-formes d’archivage utilisant la solution logicielle (projet Ad-Essor).

La solution logicielle Vitam est développée en logiciel libre et recourt aux technologies innovantes du Big Data, seules à même de relever le défi de l’archivage du nombre d’objets numériques qui seront produits ces prochaines années par les administrations de l’État. Afin de s’assurer de la qualité du logiciel livré et de limiter les décalages calendaires de réalisation, le projet est mené selon une conduite de projet Agile. Cette méthode dite « itérative », « incrémentale » et « adaptative » opère par successions de cycles réguliers et fréquents de développements-tests-corrections-intégration. Elle associe les utilisateurs tout au long des développements en leur faisant tester les éléments logiciels produits et surtout en leur demandant un avis sur la qualité des résultats obtenus. Ces contrôles réguliers permettent d’éviter de mauvaises surprises lors de la livraison finale de la solution logicielle en

corrigeant au fur et à mesure d’éventuels dysfonctionnements.

Le programme Vitam bénéficie du soutien du Commissariat général à l’investissement dans le cadre de l’action : « Transition numérique de l’État et modernisation de l’action publique » du Programme d’investissement d’avenir. Il a été lancé officiellement le 9 mars 2015, suite à la signature de deux conventions, la première entre les ministères porteurs et les services du Premier ministre, pilote du programme au travers de la DINSIC, et la seconde entre les services du Premier ministre et la Caisse des dépôts et consignations, relative à la gestion des crédits attribués au titre du Programme d’investissements d’avenir.

1.2 Présentation du document

Le document présente les fonctionnalités associées à la prise en compte de la notion de profil d’unité archivistique dans la solution logicielle Vitam.

Il s’articule autour des axes suivants :

- une présentation de la notion de profil d’unité archivistique ;
- une présentation de la manière dont la solution logicielle Vitam et le Standard d’échanges de données pour l’archivage (SEDA) la formalisent ;
- une présentation des mécanismes mis en œuvre dans la solution logicielle Vitam pour prendre en compte cette notion, en application du SEDA ;
- une présentation des modalités de création d’un profil d’unité archivistique et de son schéma de contrôle ;
- des recommandations aux utilisateurs de la solution logicielle Vitam sur l’élaboration d’un profil d’unité archivistique ;
- quelques conseils complémentaires de mise en œuvre.

Le présent document décrit les fonctionnalités qui sont offertes par la solution logicielle Vitam au terme de la *release 8* (novembre 2018). Il a vocation à être amendé, complété et enrichi au fur et à mesure de la réalisation de la solution logicielle Vitam et des retours et commentaires formulés par les ministères porteurs et les partenaires du programme.

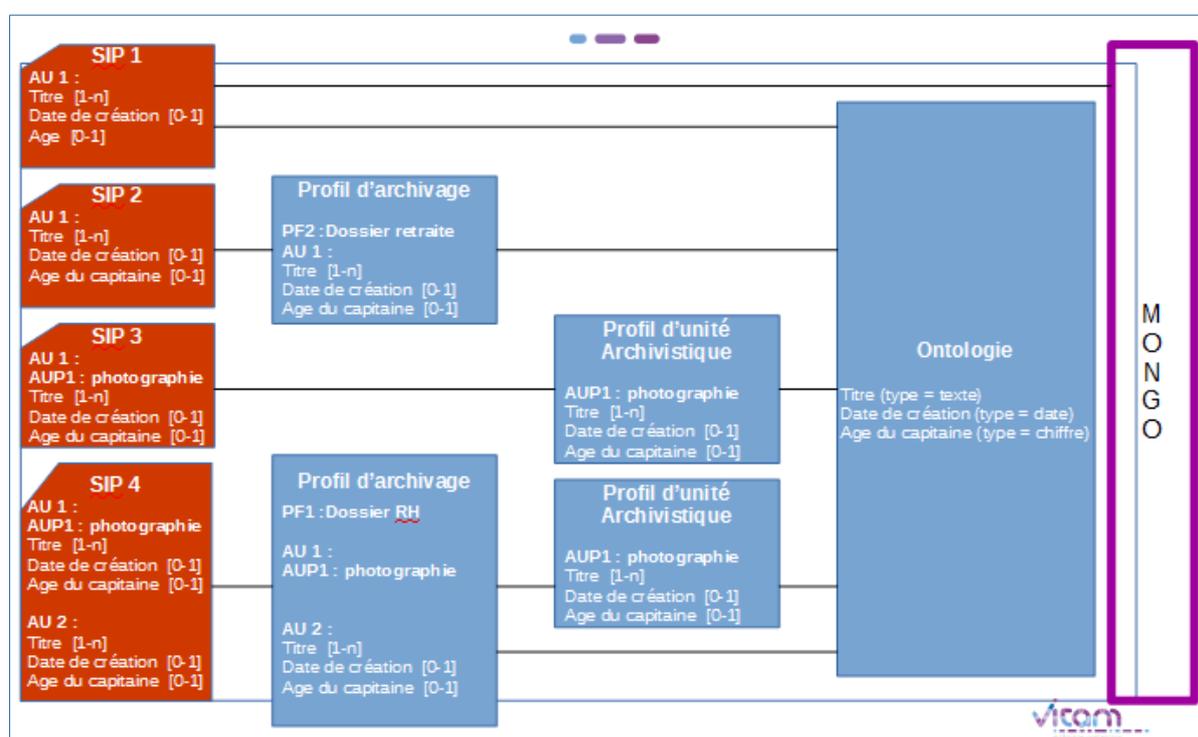
2. Présentation de la notion de profil d’unité archivistique

2.1. Qu’est-ce qu’un profil d’unité archivistique ?

Le profil d’unité archivistique permet de définir la syntaxe, la sémantique et les règles de gestion applicables à une **unité archivistique**. Il constitue une déclinaison du modèle de description propre au Standard d’échanges de données pour l’archivage (SEDA) à l’échelle d’une unité archivistique.

En d’autres termes, pour une unité archivistique, il précise :

- les éléments de description (ou vocabulaires) attendus, ainsi que leur cardinalité,
- le cas échéant, les valeurs requises,
- les règles de gestion qui lui sont associées.



À titre d’exemple, un bordereau de transfert conforme au SEDA et non associé à un profil d’archivage ou à des profils d’unité archivistique n’aura ses métadonnées contrôlées que par rapport au schéma par défaut associé au SEDA et pourra présenter une grande hétérogénéité de description, même pour une même catégorie d’archives (ex : photographies numériques, dossiers de subventions, comptes-rendus de réunion, etc.).

En revanche, un bordereau de transfert pour lequel des profils d’unité archivistique auront été définis disposera d’unités archivistiques dont la description sera homogène et propre à un type d’archives ou à une typologie documentaire en particulier, par exemple : des photographies numériques.

Le profil d'unité archivistique est spécifique à une unité archivistique, contrairement au profil d'archivage qui impose une structuration à l'ensemble des unités archivistiques présentes dans un bordereau de transfert.

Il peut également être utilisé dans un profil d'archivage pour spécifier des types particuliers d'unité archivistique, en plus du contrôle effectué par le profil d'archivage.

Il accompagne l'unité archivistique tout au long de sa prise en charge par le système d'archivage électronique en permettant de contrôler ses métadonnées aussi bien à l'entrée que lors de toute modification. Le profil d'archivage n'est utilisable qu'en entrée, tandis que le profil d'unité archivistique est utilisable en entrée et en accès.

2.2. Pourquoi élaborer un profil d'unité archivistique ?

Comme pour un profil d'archivage, définir un profil d'unité archivistique permet de :

- faciliter, voire automatiser, les contrôles sur les transferts d'archives (à l'échelle des unités archivistiques) d'un service producteur à un service d'archives, depuis un service externe vers la solution logicielle Vitam ;
- effectuer un contrôle supplémentaire sur les transferts reçus et leur contenu, s'ajoutant à l'ensemble des tâches visant à s'assurer de la conformité des données reçues par rapport à celles qui ont été transférées. Toutes ces actions sont gages de qualité des données, au moment de leur réception dans un système d'archivage électronique ;
- définir la sémantique et la syntaxe attendue pour des unités archivistiques à transférer, permettant ainsi une meilleure connaissance de celles-ci et la rédaction de spécifications sur les lots à exporter à fournir à un prestataire externe (tiers-archiviste, éditeur de solutions logicielles, etc.) ;
- imposer un modèle de description identique sur une typologie particulière d'archives, produites dans différents contextes ;
- faciliter la recherche d'unités archivistiques, voire spécifier un filtre de recherche propre à un profil d'unité archivistique en particulier.

Contrairement au profil d'archivage, les profils d'unité archivistique accompagnent les unités archivistiques tout au long de leur prise en charge dans la solution logicielle Vitam et permettent de contrôler les données, leur sémantique et leur syntaxe, lors d'une éventuelle mise à jour des unités archivistiques ou de leurs règles de gestion associées.

2.3. Quand et comment élaborer un profil d'unité archivistique ?

L'élaboration d'un profil d'unité archivistique précède l'ouverture d'un transfert à destination d'un système d'archivage électronique.

Elle peut être envisagée dans les cas suivants :

- transferts de flux applicatifs, afin de les automatiser ;

- transferts de dossiers ou de documents dits « sériels », obéissant strictement à des règles de nommage et de description uniformes (par exemple, des images numérisées par un service d’archives) ;
- transferts réguliers et récurrents d’un même type d’archives par différents services producteurs et donc volonté de disposer d’une description homogène, y compris pour faciliter les recherches sur celles-ci.

Dans ces cas-là, son utilisation peut entrer en concurrence avec celle d’un profil d’archivage.

En revanche, contrairement au profil d’archivage, qui requiert de connaître la hiérarchie des données et leur structuration, un profil d’unité archivistique peut être utilisé pour des types ou catégories d’archives dont la structuration précise ne peut être définie par défaut (absence de connaissance *a priori* du nombre de niveaux dans l’arborescence) et qui sont :

- des documents bureautiques, pris isolément,
- des documents structurés et décrits de manière non uniforme (par exemple, des boîtes mail).

Un profil d’unité archivistique peut s’utiliser indifféremment dans différents contextes. Par exemple, un profil détaillant un modèle d’« acte de naissance » peut être utilisé pour un transfert d’archives d’un SIRH, mais également d’une application de demande de retraite. Il peut alors répondre à un besoin plus transverse de contrôle et de gestion des données dans la solution logicielle Vitam.

La rédaction d’un profil d’unité archivistique, en fonction de l’usage que l’on souhaite en faire, suppose un travail préparatoire, adapté à chaque lot ou catégorie d’archives concerné (identification des documents à transférer en tenant compte des calendriers de conservation ou des circulaires de tri existants, détermination des métadonnées utiles à la recherche, structuration de l’information dans chaque transfert, etc.).

Elle résulte d’un travail entre les différents acteurs définis dans le Standard d’échange de données pour l’archivage (SEDA), à savoir :

- le service d’archives (administrateurs fonctionnels et techniques du système d’archivage électronique, archivistes en charge de la collecte),
- le service producteur (responsables de l’administration fonctionnelle et technique de l’application concernée quand il s’agit de l’archivage d’une application),
- le service de contrôle quand il existe.

3. Formalisation des profils d’unité archivistique

3.1. Dans un fichier propre

Les profils d’unité archivistique prennent la forme de fichiers au format JSON. En termes de syntaxe et de grammaire, ils correspondent plus exactement à des **schémas de contrôle**, permettant de valider des contenus exprimés au format JSON, qui est le format de structuration des données utilisé dans la solution logicielle Vitam¹.

Ces profils d’unité archivistique doivent se conformer :

- aux règles émises par le **Standard d’échange de données pour l’archivage (SEDA)**. Ainsi, ils peuvent :
 - émettre des règles plus strictes que celles du SEDA (ex. : rendre unique un titre, alors que le standard rend possible sa répétabilité), mais pas des règles moins strictes (ex : rendre répétable le niveau de description alors qu’il est unique dans le SEDA) ;
 - rendre obligatoire l’utilisation de certains éléments et/ou valeurs. Il peut s’agir de :
 - métadonnées descriptives,
 - métadonnées de contrôle et de gestion (profil d’unité archivistique et règles de gestion),
 - métadonnées référençant un objet ou un nœud parent ;
- aux vocabulaires définis dans l’**ontologie**. Ils doivent en effet définir des éléments et types compatibles avec ceux référencés dans l’ontologie².

3.2. Dans le Standard d’échange de données pour l’archivage (SEDA)

La manière de déclarer un profil d’unité archivistique est définie dans le Standard d’échanges de données pour l’archivage (SEDA).

Le SEDA offre la possibilité de déclarer un profil d’unité archivistique dans un bordereau de transfert, au moment de la demande de transfert à un service d’archives (message ArchiveTransfer, bloc DataObjectPackage) dans les blocs **ArchiveUnit** pour chaque unité archivistique qui le nécessite.

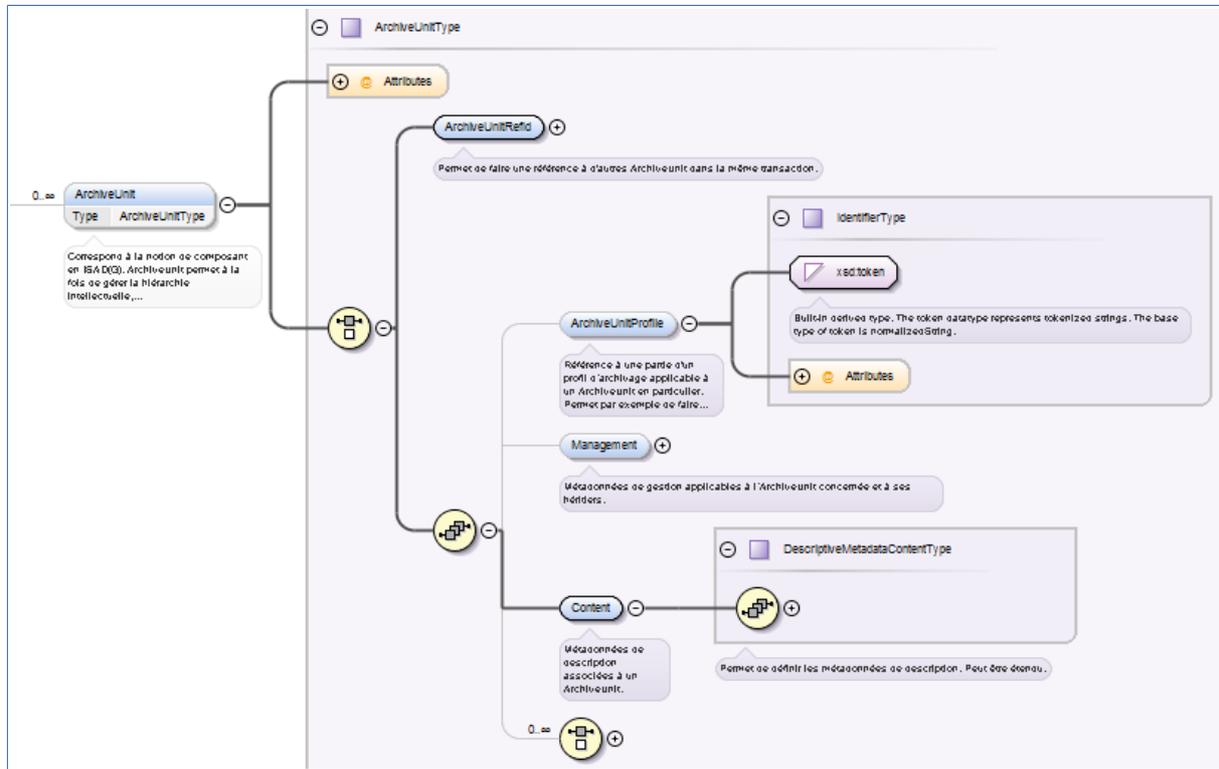
Dans le sous-bloc **ArchiveUnitProfile**, est indiqué l’identifiant du profil d’unité archivistique correspondant à l’unité archivistique que l’on souhaite contrôler, tel qu’il figure dans le référentiel des profils d’unité archivistique (dit des documents type) de la solution logicielle Vitam. Cet identifiant renvoie au référentiel des profils d’unité archivistique et correspond à l’identifiant de la notice détaillée du profil d’unité archivistique qui doit être associée à une unité

¹ Pour plus d’informations sur la rédaction d’un schéma de contrôle, consulter la section 5.2.2 « Schématisation des données à contrôler » du présent document.

² Un tableau de concordance entre les types définis dans l’ontologie et ceux présents dans un schéma de contrôle JSON est présent dans la section 5.2.2.2 « Rédaction d’un schéma de contrôle ».

archivistique³.

Le schéma ci-dessous montre la structuration des informations pour une unité archivistique :



³ Pour plus d’informations sur le référentiel des profils d’archivage, consulter le chapitre « 4.1. Administration d’un référentiel des profils d’unité archivistique ».

4. Mécanismes mis en œuvre dans la solution logicielle Vitam

La solution logicielle Vitam offre à un service d’archives ou à un service externe plusieurs fonctionnalités lui permettant de mettre en œuvre des profils d’unité archivistique :

- l’**administration d’un référentiel** des profils d’unité archivistique ;
- en entrée du système, une **vérification que les unités archivistiques** présentes dans le SIP et déclarant un profil d’unité archivistique **sont conformes à celui-ci** ;
- lors de la mise à jour des unités archivistiques, la **déclaration a posteriori d’un profil** d’unité archivistique et la **vérification que les unités archivistiques modifiées** et déclarant un profil d’unité archivistique **restent conformes** à celui-ci.

4.1. Administration d’un référentiel des profils d’unité archivistique

La solution logicielle Vitam intègre un référentiel des profils d’unité archivistique, administrable par un utilisateur doté des droits adéquats (**administrateur fonctionnel et/ou technique**) et géré dans une collection particulière⁴.

Ce référentiel est propre à chaque tenant de la solution logicielle Vitam. Il se compose de notices descriptives au format JSON, qui incluent nativement un schéma de contrôle au format JSON ou auxquelles il est possible d’associer *a posteriori* un schéma.

Il est possible de réaliser les opérations présentées ci-dessous.

4.1.1. Import de la notice descriptive du profil d’unité archivistique

Il est possible d’importer 1 à n notice(s) descriptive(s) de profil(s) d’unité archivage sous la forme d’un fichier JSON. Il s’agit d’une opération d’administration.

Cette notice descriptive comprend les informations suivantes pour un profil d’unité archivistique donné :

- identifiant système unique de la notice correspondant à un GUID (_id) attribué automatiquement par la solution logicielle Vitam ;
- identifiant (Identifiant) de la notice qui peut être attribué au profil d’unité archivistique, généré automatiquement par la solution logicielle Vitam ou par l’application à l’origine de sa création⁵. S’il est généré par la solution logicielle Vitam, cet identifiant se compose du préfixe AUP, suivi d’un tiret et d’une suite de 6 chiffres incrémentés automatiquement ;
- nom du profil d’unité archivistique, qui doit être obligatoirement renseigné dans la

⁴ Pour plus d’informations sur la modélisation de cette collection, consulter le document *Modèle de données*, chapitre « Collection ArchiveUnitProfile ».

⁵ Cette option est paramétrable dans le fichier de configuration `functional-administration.conf` (cf. *Documentation d’exploitation*, chapitre 7.2.10.2.1 « Fichier `functional-administration.conf` »). Par défaut, la solution logicielle Vitam attribue automatiquement un identifiant métier.

- solution logicielle Vitam (Name) ;
- description du profil d’unité archivistique, obligatoire (Description) ;
 - statut « Actif » ou « Inactif » (Status) ;
 - schéma de contrôle, obligatoire, mais pouvant être vide (ControlSchema).
 - Il est destiné à contenir l’ensemble des éléments qui feront l’objet d’un contrôle en entrée ou en mise à jour ;
 - Les éléments (ou propriétés) qu’il définit doivent se conformer aux vocabulaires définis dans l’ontologie, que ce soit en termes de nommage ou de typage ;
 - date de création de la notice descriptive du profil d’unité archivistique (CreationDate) ;
 - dernière date de modification de la notice descriptive du profil d’unité archivistique (LastUpdate) ;
 - si la notice descriptive du profil d’unité archivistique est active, date d’activation du profil (ActivationDate) ;
 - si la notice descriptive du profil d’unité archivistique est inactive, date de désactivation du profil (DeactivationDate) ;
 - tenant dans lequel le profil d’unité archivistique s’applique (_tenant) ;
 - version du profil d’unité archivistique (_v) ;
 - liste des vocabulaires de l’ontologie référencés dans le schéma de contrôle (Fields).

Points d’attention :

- Il est possible d’importer, en une seule fois, un référentiel complet, comprenant plusieurs notices décrivant chacune un profil d’unité archivistique. La solution logicielle Vitam ne comptabilisera qu’une seule opération, et ne prendra pas en compte dans le journal des opérations la création unitaire des différents items compris dans le référentiel importé.
 - Afin d’optimiser la traçabilité de la création des différentes notices, **il est recommandé de créer ces dernières une par une.**
- La solution logicielle Vitam permet d’importer une notice détaillée avec un schéma de contrôle déjà détaillé ou vide de tout contenu. Dans ce dernier cas, l’import d’une notice ne suffit pas pour utiliser un profil d’unité archivistique dans la solution logicielle Vitam. Il est nécessaire, après cette opération, d’ajouter le schéma de contrôle correspondant au profil d’unité archivistique lui-même et de l’associer à la notice qui le référence, afin qu’il soit pris en compte dans la solution logicielle Vitam⁶.

4.1.2. Ajout du profil d’unité archivistique lui-même

À chaque notice descriptive, il est possible d’associer un seul profil d’unité archivistique ou schéma de contrôle.

- Dès son import, la notice détaillée peut inclure un schéma de contrôle détaillé, dans la mesure où tous deux sont au format JSON. Il s’agit d’une opération d’import.

⁶ Cf section 4.1.2 « Ajout du profil d’unité archivistique lui-même ».

- Néanmoins, la notice peut être importée sans schéma détaillé. Il est alors possible, *a posteriori*, de lui adjoindre ce schéma de contrôle et de vérifier sa validité par rapport au format JSON. Il s’agit alors d’une opération de mise à jour.

L’ajout d’un schéma de contrôle dans la collection est un des prérequis indispensable pour pouvoir réaliser des contrôles entre un profil d’unité archivistique et des unités archivistiques présentes dans des bordereaux de transfert au moment de leur transfert dans la solution logicielle Vitam ou déjà prises en charge dans la solution logicielle Vitam et en cours de modification.

Lors de cette opération, la solution logicielle Vitam effectue les vérifications suivantes :

- **vérification que les vocabulaires** déclarés dans le schéma de contrôle **existent dans l’ontologie** ;
- si des types sont associés aux vocabulaires dans le schéma, **vérification que ces types correspondent au type de chacun de ces vocabulaires dans l’ontologie**⁷.

L’import ou la mise à jour d’un profil d’unité archivistique peut échouer pour les motifs suivants :

- le schéma de contrôle est invalide, c’est-à-dire qu’il n’est pas conforme au format attendu pour un schéma JSON. À noter que la solution logicielle Vitam ne vérifie pas si le schéma de contrôle est cohérent :
 - d’un point de vue métier. Par exemple, on peut rendre obligatoire un élément qui n’est pas défini dans les propriétés d’un même schéma ou imposer des cardinalités telles que 10 – 5. La solution logicielle Vitam enregistrera le schéma de contrôle en l’état.
 - par rapport au SEDA. Par exemple, la solution logicielle Vitam enregistrera le schéma de contrôle sans contrôler si ce dernier respecte les cardinalités imposées par le SEDA.
- le schéma de contrôle contient un vocabulaire inconnu de l’ontologie.

4.1.3. Modification de la notice descriptive d’un profil d’unité archivistique

La modification des champs des notices décrivant les profils d’unité archivistique est possible au moyen des API et de l’IHM standard fournies avec la solution logicielle Vitam.

Cette action provoque la création d’une nouvelle version de la notice modifiée. Elle fait l’objet d’une journalisation dans le journal des opérations.

Les champs modifiables sont :

- le nom du profil d’unité archivistique (Name) ;
- la description du profil d’unité archivistique (Description) ;
- le statut « Actif » ou « Inactif » (Status) ;
- le schéma de contrôle (ControlSchema).

⁷ Les règles propres à la cohérence des types définis dans un profil d’unité archivistique et dans l’ontologie sont définies dans la partie « 5.2.2.1.2. Propriétés d’un schéma de contrôle ».

Points d’attention :

- le statut de la notice descriptive doit être « Actif » pour pouvoir procéder à des transferts de SIP comprenant des unités archivistiques devant se conformer à un profil d’unité archivistique ;
- la solution logicielle Vitam permet de modifier le profil d’unité archivistique ou son schéma de contrôle, tant que ce dernier n’est pas utilisé par une unité archivistique en entrée ou déjà transférée dans la solution logicielle Vitam. S’il l’est, il n’est plus possible de le modifier⁸.

4.1.4. Activation / Désactivation d’un profil d’unité archivistique

La solution logicielle Vitam permet de rendre active ou inactive une notice détaillant un profil d’unité archivistique.

En fonction du statut, seront autorisés ou non :

- le transfert de SIP dont au moins une unité archivistique déclare un profil d’unité archivistique ;
- la modification d’une unité archivistique déclarant un profil d’unité archivistique.

	Catégorie d’opération	Profil d’unité archivistique	Schéma de contrôle	Résultat
CAS 1	Transfert	ACTIF	vide	Transfert de SIP dans le système non autorisé.
CAS 2		ACTIF	renseigné	Transfert de SIP dans le système autorisé.
CAS 3		INACTIF	vide	Transfert de SIP dans le système non autorisé.
CAS 4		INACTIF	renseigné	Transfert de SIP dans le système non autorisé.
CAS 5	Modification	ACTIF	vide	Modification d’unité archivistique non autorisée.
CAS 6		ACTIF	renseigné	Modification d’unité archivistique autorisée.
CAS 7		INACTIF	vide	Modification d’unité archivistique non autorisée.
CAS 8		INACTIF	renseigné	Modification d’unité archivistique non autorisée.

La modification du statut engendre la mise à jour des champs :

- Date de mise à jour ;
- Date d’activation OU date de désactivation.

⁸ Si on a besoin de modifier un profil d’unité archivistique déjà utilisé par au moins une unité archivistique, consulter la section 6 « Conseils de mise en œuvre » du présent document.

La date d’activation correspond à la date à laquelle la notice et, par conséquent, le profil d’unité archivistique sont rendus actifs. Il peut s’agir de :

- sa date d’import, si elle a un statut « Actif » au moment de son import ;
- la date correspondant à l’action d’activation, si celle-ci est postérieure à l’import de la notice dans la solution logicielle Vitam.

La date de désactivation correspond à la date à laquelle la notice et par conséquent le profil d’unité archivistique sont rendus inactifs. Il peut s’agir de :

- sa date d’import, si elle a un statut « Inactif » ou non renseigné au moment de son import ;
- la date correspondant à l’action de désactivation, si celle-ci est postérieure à l’import de la notice dans la solution logicielle Vitam.

4.2. Processus d’entrée

Dans le cadre du processus d’entrée d’un ensemble d’archives, suite à la réception d’un message ArchiveTransfer du SEDA, parmi les tâches et traitements internes qu’elle effectue, la solution logicielle Vitam **vérifie que les unités archivistiques déclarant un profil d’unité archivistique sont conformes à ce dernier.**

Ce contrôle s’effectue lors du processus de contrôle et traitements des unités archivistiques

- après la vérification de la conformité du type des vocabulaires présents dans les unités archivistiques présentes dans le bordereau de transfert par rapport à leur type d’indexation dans l’ontologie,
- avant les vérifications portant sur les règles de classification et de gestion.

Lors de cette étape de vérification de la conformité entre le profil d’unité archivistique déclaré dans au moins une unité archivistique présente dans le bordereau de transfert et cette dernière :

- Si l’unité archivistique est conforme à son profil d’unité archivistique, si elle correspond au modèle de données défini dans le profil d’unité archivistique, la tâche de vérification aura un statut « OK » et la solution logicielle Vitam passera à la tâche suivante, de vérification du niveau de classification ;
- Si au moins une unité archivistique n’est pas conforme à son profil d’unité archivistique pour les raisons suivantes :
 - elle ne correspond pas au profil d’unité archivistique qu’elle déclare,
 - le profil d’unité archivistique déclaré est inconnu,
 - le profil d’unité archivistique déclaré est inactif,
 - la notice descriptive du profil d’unité archivistique est active, mais son schéma de contrôle est vide,

alors le transfert du SIP échouera à la tâche de vérification de conformité aux profils d’unité archivistique, passera à la tâche suivante, mais n’aboutira pas. Un message de réponse

(ArchiveTransferReply ou ATR) indiquera que le transfert a échoué à la tâche de vérification de la conformité au profil d’unité archivistique et précisera la première erreur rencontrée dans le détail de l’événement (EventDetailData)⁹.

Exemple : message d’erreur sur la tâche de contrôle de conformité au profil d’unité archivistique.

```
<ArchiveUnit id="ID3">
  <Management>
    <LogBook>
      <Event>
        <EventTypeCode>LFC.CHECK_ARCHIVE_UNIT_PROFILE</EventTypeCode>
        <EventType>Vérification de la conformité aux profils d'unité
archivistique</EventType>
        <EventDateTime>2018-04-27T10:37:30.404</EventDateTime>
        <Outcome>KO</Outcome>
        <OutcomeDetail>LFC.CHECK_ARCHIVE_UNIT_PROFILE.KO</OutcomeDetail>
        <OutcomeDetailMessage>Échec de la vérification de la conformité aux
        profils d'unité archivistique</OutcomeDetailMessage>
        <EventDetailData>{"ArchiveUnit":"ID3","ArchiveUnitProfile":"AUP-
220001","evDetTechData":"Archive Unit Profile not
        found"}</EventDetailData>
      </Event>
    </LogBook>
  </Management>
  <Content>
    <SystemId>aeaqaabebhc7dlhabspualda2w5tlyaaaba</SystemId>
  </Content>
</ArchiveUnit>
```

4.3. Accès

4.3.1. Consultation des unités archivistiques déclarant un profil d’unité archivistique

Par défaut, lorsque l’utilisateur demande à accéder au détail d’une unité archivistique, la solution logicielle Vitam retourne l’identifiant du profil d’unité archivistique associé à celle-ci.

4.3.2. Mise à jour des unités archivistiques

4.3.2.1. Modification d’un profil d’unité archivistique

En mise à jour, la solution logicielle Vitam permet :

- d’ajouter un profil d’unité archivistique à une unité archivistique préalablement transférée dans le système ;

⁹ Pour plus d’informations sur les messages d’erreur, consulter « Annexe 4. Messages d’erreur » du présent document.

- de modifier un profil d'unité archivistique associé à une unité archivistique.

Cette opération de mise à jour n'est possible qu'aux conditions suivantes :

- l'unité archivistique doit être conforme au profil d'unité archivistique à laquelle on souhaite l'associer ;
- le profil d'unité archivistique doit être actif et disposer d'un schéma de contrôle détaillé.

Si l'une de ces conditions n'est pas présente, la modification n'aura pas lieu.

Cette opération fait l'objet d'un enregistrement dans le journal des opérations en tant qu'opération de mise à jour des règles de gestion.

Le droit de mise à jour d'un profil d'unité archivistique est paramétrable dans le contrat d'accès via la possibilité de restreindre les droits d'écriture :

- Si la valeur du champ « WritingRestrictedDesc » est égale à « true », le détenteur du contrat ne peut pas effectuer des mises à jour sur le profil d'unité archivistique ;
- Si sa valeur est égale à « false » (valeur par défaut), le détenteur du contrat peut effectuer des mises à jour sur les métadonnées descriptives, ainsi que sur les métadonnées de gestion et de contrôle de métadonnées dont la mise à jour du profil d'unité archivistique.

[4.3.2.2. Modification des métadonnées d'une unité archivistique déclarant un profil d'unité archivistique](#)

La solution logicielle Vitam permet de modifier les métadonnées d'une unité archivistique déclarant un profil d'unité archivistique, à condition que :

- l'unité archivistique reste conforme au profil d'unité archivistique qu'elle déclare,
- le profil d'unité archivistique soit actif et dispose d'un schéma de contrôle détaillé.

Il s'agit d'une opération de mise à jour.

Lors de cette opération, la solution logicielle Vitam effectue les vérifications suivantes :

- **vérification de l'association d'un profil d'unité archivistique à l'unité archivistique** que l'on souhaite modifier ;
- **vérification de la conformité** de l'unité archivistique **par rapport au profil d'unité archivistique** qu'elle déclare.

5. Étapes d’élaboration d’un profil d’unité archivistique et de son schéma de contrôle

5.1. Phase d’analyse

5.1.1. Analyse des données

Définir un profil d’unité archivistique nécessite une analyse préalable des données à archiver, type de document par type de document. Il s’agit de connaître :

- les types de documents rencontrés (ex : documents bureautiques, factures, mails, etc.) ;
- les métadonnées associées à ces données, permettant de les identifier, de les gérer et de les retrouver après transfert dans le système d’archivage électronique (métadonnées de pérennisation, de représentation et de description au sens de la norme OAIS) ;
- le contenu de données, à savoir la nature des données à archiver (catégorie ou typologie d’archives).

En prenant l’exemple d’un projet de dématérialisation d’archives, il est recommandé, lors de ce travail préparatoire, de se poser les questions suivantes¹⁰ :

Identifier le contexte	Quels sont les documents à archiver ? Pourquoi ? Existe-il un besoin légal ? La reprise de l’existant a-t-elle été prévue ?
Identifier les documents dématérialisés	Quelles sont les typologies concernées ? Quels sont les usages ? Les documents et données sont-ils nativement numériques ou ont-ils été numérisés ? Une impression papier est-elle prévue en fin de procédure ? De quelle(s) base(s) de données les données et documents sont-ils issus ? Quels sont les formats des documents ? Quelles sont les possibilités d’exports et le modèle des données ? Les documents ont-ils une valeur probante (ex. signature électronique) ?
Identifier les outils techniques	Dans le cadre d’une numérisation, s’agit-il d’une opération interne ? Le recours à un prestataire externe est-il prévu ? Une reprise du stock a-t-elle été envisagée ? L’application a-t-elle été développée en interne ? L’acquisition d’un logiciel est-elle prévue ? Quelles métadonnées ont été intégrées ? Quelles métadonnées peut-on extraire de l’application ? L’organisation a-t-elle recours à un tiers de télétransmission ? Un parapheur électronique est-il en place ? Les agents disposent-ils de la signature électronique ? Est-elle systématiquement utilisée ?
Stockage, archivage et sort final	Quelles sont les règles de gestion et plus particulièrement les règles de conservation et le sort final des documents ? Comment sont conservées les pièces dématérialisées ? Où ? Par qui (organisation, tiers archiveur) ? Existe-t-il un système d’archivage électronique ? Un coffre-fort ? Des serveurs répliqués ? Un système de gestion électronique de documents ? Quelles sont les modalités d’accès ? La destruction est-elle possible ? Quelles sont les modalités ?

À l’issue de cette première analyse, l’archiviste doit être en mesure d’avoir à sa disposition :

¹⁰ Le tableau suivant est une libre adaptation du tableau présenté dans « Fiche pratique n°13 : positionnement de l’archiviste face à un projet de dématérialisation », Association des Archivistes de France.

- une liste des métadonnées liées à chaque type de document ;
- les règles de gestion associées.

Le résultat de ce travail peut prendre la forme du tableau suivant :

Type de document ou d’unité archivistique	Métadonnées		Contenu de données
	Informations de description (métadonnées)	Règles de gestion	Documents versés en pièces jointes (oui/non)

La liste des métadonnées peut prendre la forme du tableau suivant :

Nom de la métadonnée dans le système d’origine	type	cardinalité	Valeurs imposées	Commentaires	À récupérer ?
	<i>Texte, Date, Chiffres, etc.</i>	<i>1-1, 0-1, 0-n</i>	<i>Ex : liste, valeur héritée.</i>		<i>OUI / NON</i>
Age du capitaine	Numérique	0-1		Entier entre 0 et 150	OUI

Il est nécessaire de bien évaluer les informations à retenir dans le profil d’unité archivistique, leur degré de précision (formes, valeurs, cardinalités), afin de faciliter les recherches et la gestion futures des données dans la solution logicielle Vitam.

5.1.2. Modélisation des données

À la suite de cette première analyse, il est nécessaire de se poser les questions suivantes :

- Est-il utile de récupérer l’ensemble des métadonnées et de leur valeur dans un bordereau de transfert ?
- Quelles sont les métadonnées qui sont systématiquement renseignées ?
- Quelles sont les métadonnées qui pourraient faire l’objet d’une interrogation par le moteur de recherche de la solution logicielle Vitam ?
- Quelles sont les métadonnées qu’il faudrait voir afficher dans une notice détaillant une unité archivistique ?

Les réponses à ces questions doivent orienter les choix de correspondance (ou mapping) entre les métadonnées initiales et le modèle de données du SEDA. La solution logicielle Vitam prenant en charge un bordereau de transfert conforme au SEDA, il est nécessaire de :

- faire correspondre les métadonnées à prendre en charge avec celles du SEDA ;
- le cas échéant, si aucune métadonnées proposée par le SEDA ne correspond au modèle

initial, de créer une ou plusieurs métadonnée(s) supplémentaire(s) qu’il faudra ajouter dans l’ontologie.

Le résultat de ce travail peut prendre la forme du tableau suivant :

Nom de la métadonnée dans une unité archivistique	type	cardinalité	Nom de la métadonnée dans le système d’origine	Valeurs imposées	Commentaires
	<i>Texte, Date, Chiffres, etc.</i>	<i>1-1, 0-1, 0-n</i>	<i>Métadonnées ou concaténation de plusieurs métadonnées</i>	<i>Ex : liste, valeur héritée.</i>	
Title	Texte	1-1	Nom + Prénom		Nom en majuscules
Description	Texte	0-1	Commentaires		
AgeDu Capitaine	Numérique	0-1	AgeCapitaine		Entier entre 0 et 150

5.2. Phase de réalisation

5.2.1. Réalisation d’un bordereau de transfert

Suite à ce travail préliminaire, il est recommandé de réaliser un bordereau de transfert contenant :

- soit une unité archivistique conforme à la modélisation préalablement réalisée,
- soit plusieurs unités archivistiques, chacune faisant état de différentes propositions de modélisation.

Cette étape peut être réalisée par un informaticien ou par un prestataire en charge de concevoir des exports.

Ce travail permet de :

- confirmer le besoin exprimé à la suite de l’analyse des données et de leur modélisation,
- vérifier la structuration des données, et, le cas échéant, de corriger l’unité archivistique, le profil d’unité archivistique, ainsi que les spécifications ;
- vérifier la conformité au SEDA du bordereau de transfert, avant même de vérifier la conformité d’une de ses unités par rapport à un profil d’unité archivistique ;
- aider à la rédaction du schéma de contrôle. En effet, il est recommandé de transférer ce bordereau dans une plate-forme de tests de la solution logicielle Vitam et de récupérer l’unité archivistique enregistrée au format JSON dans la base de données MongoDB.

5.2.2. Schématisation des données à contrôler

5.2.2.1. Méthode de rédaction d’un schéma de contrôle

Actuellement, il n’existe ni outil ni interface pour générer automatiquement un schéma de contrôle associé à un profil d’unité archivistique au format JSON. Un schéma de contrôle doit être écrit à partir et au moyen d’un éditeur acceptant le format JSON.

Pour en rédiger un, il est conseillé de :

- créer un fichier dans un outil de validation de schéma JSON (ex : Visual Studio Code, Oxygen) et de l’enregistrer en tant que fichier JSON ;
- En parallèle, transférer le bordereau de transfert dans la solution logicielle Vitam et récupérer l’unité archivistique enregistrée au format JSON dans la base de données MongoDB, puis l’enregistrer dans un fichier au format JSON ;
- ouvrir cette unité archivistique dans l’outil de validation de schéma JSON et l’associer au schéma de contrôle.
- rédiger au fur et à mesure le schéma de contrôle, en contrôlant sa validité par rapport à l’unité archivistique.

Cette méthode permet de vérifier directement la conformité du profil d’unité archivistique par rapport à une unité archivistique dès la phase de conception du schéma de contrôle.

5.2.2.2. Structure d’un schéma de contrôle

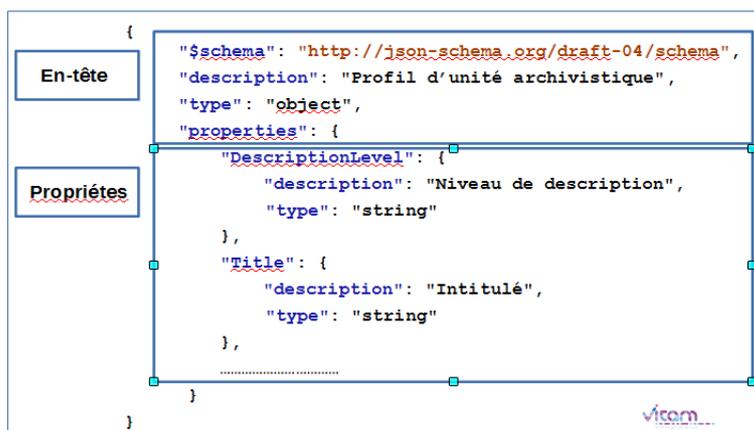
Un schéma de contrôle associé à un profil d’unité archivistique comprend deux parties¹¹ :

- une partie déclarative ou **en-tête** (obligatoire),
- une partie listant les **propriétés** du schéma de contrôle (obligatoire), c’est-à-dire énumérant les éléments du SEDA ou de l’ontologie qui feront l’objet d’un contrôle. Le contenu de la liste est laissé à l’appréciation de chacun.

Cette partie peut elle-même être décomposée en deux ou trois sous-parties :

- contexte d’utilisation de l’unité archivistique,
- règles de gestion à contrôler au moyen du profil d’unité archivistique (équivalent au contenu du bloc Management du SEDA),
- énumération des métadonnées descriptives à contrôler (équivalent au contenu du bloc Content du SEDA).

¹¹ Pour plus d’informations sur la manière de rédiger un schéma de contrôle JSON, consulter l’ouvrage suivant : DROETTBOOM Michael *Understanding JSON Schema*, Space Telescope Science Institute, 2013, 2018 (dernière version), in : <https://json-schema.org/understanding-json-schema/index.html> (dernière mise à jour : août 2018).



5.2.2.1.1. En-tête d’un schéma de contrôle

Comme dans un fichier au format XML, le schéma de contrôle JSON comporte des éléments d’information propres au fichier, permettant :

- d’identifier son format :
 - **\$schema** : référence au schéma JSON. La solution logicielle Vitam supportant la version 4 du schéma JSON, la référence doit être égale à « http://json-schema.org/draft-04/schema » ;
 - **type** : toujours égal à « object » dans l’en-tête. Le type « object » est assimilable à un dictionnaire, possédant des propriétés. En langage XML, il correspondrait à un élément englobant des sous-éléments et ne pouvant contenir de valeurs ;
 - **properties** : dictionnaire listant les propriétés associées au schéma de contrôle, et, de fait, au profil d’unité archivistique. On y énumère les éléments à contrôler ;

Exemple :

```

{
"$schema": "http://json-schema.org/draft-04/schema",
"type": "object",
"properties": {
.....
}
}

```

- de déterminer des règles propres aux objets (notion de type « object » JSON) à appliquer aux propriétés qui vont être définies dans le schéma. La solution logicielle Vitam supporte seulement la règle **additionalProperties** , de type booléen :
 - Si sa valeur est égale à « true », une unité archivistique pourra contenir des éléments (métadonnées) supplémentaires, non définis dans le schéma de contrôle, pouvant être aussi bien des métadonnées de gestion, des métadonnées descriptives, internes ou externes ;
 - Si sa valeur est égale à « false », une unité archivistique contenant des éléments supplémentaires, non définis dans le schéma de contrôle, ne sera

pas conforme à son profil et ne pourra pas être versée dans la solution logicielle Vitam ;

- de déterminer des règles génériques à appliquer aux propriétés qui vont être définies dans le schéma :
 - **required** : liste d’éléments de type texte (= string) obligatoires. Ces éléments doivent ensuite être définis en tant que propriétés du schéma de contrôle. S’il s’agit de sous-propriétés (ex : KeywordContent, sous-élément et sous-propriété du bloc Keyword), la règle ne s’appliquera pas.

Exemple : Pour se conformer au profil, l’unité archivistique devra contenir obligatoirement les éléments DescriptionLevel et Title. En revanche, l’élément DataObjectReference y est facultatif, car il n’est pas listé comme requis.

```
{
  "$schema": "http://json-schema.org/draft-04/schema",
  "type": "object",
  "additionalProperties": false,
  "required": ["DescriptionLevel", "Title"],
  "properties": {
    "_og": {"type": "string"},
    "DescriptionLevel": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

- **definitions** : liste permettant de définir des expressions à utiliser dans les propriétés du schéma de contrôle. L’équipe projet Vitam recommande son utilisation dans l’en-tête des schémas afin de déterminer :
 - les différentes cardinalités possibles ;
 - des patterns particuliers (ex : date).

Ces expressions sont destinées à être ensuite déclarées comme références dans les propriétés du schéma de contrôle qui ne sont pas de type objet.

Exemple : Ont été définies :

- différentes cardinalités pour des éléments SEDA nativement répétables : 1-1, 1-n, 0-1, 0-n ;
- une expression permettant d’empêcher la création de sous-objets dans un vocabulaire de type objet ;
- une expression précisant comment écrire une date.

```
{
  "$schema": "http://json-schema.org/draft-04/schema",
  "type": "object",
  "additionalProperties": false,
  "definitions": {
    "one-one": {"type": "array", "minItems": 1, "maxItems": 1},
    "one-many": {"type": "array", "minItems": 1},
    "zero-one": {"type": "array", "minItems": 0, "maxItems": 1},
    "zero-many": {"type": "array", "minItems": 0},
    "closed-object": {"type": "object", "additionalProperties": false},
    "simple-date": {"type": "string", "pattern": "[0-9]{4}-[0-9]{2}-[0-9]{2}"}
  }
  "properties": {
```

```
"_og": {"type": "string"},
>DescriptionLevel": {"type": "string"},
>Title": {"type": "string"}
}
}
```

- de déterminer des propriétés à exclure du schéma de contrôle, que ce soit lors d’un transfert ou lors d’une mise à jour :
 - **patternProperties** : liste les vocabulaires à exclure du contrôle qui sera effectué par le schéma de contrôle. Dans la solution logicielle Vitam, il est obligatoire de référencer :
 - tout vocabulaire interne à la solution logicielle Vitam et non attendu dans un bordereau de transfert,
 - si aucune règle de gestion n’est définie dans le schéma de contrôle, le bloc Management.

Si `additionalProperties` est égal à « false », il est obligatoire d’utiliser `patternProperties`.

Exemple : sont autorisés et ne feront pas l’objet d’un contrôle par le profil d’unité archivistique les vocabulaires commençant par un underscore, `DataObjectReference`, `ImplementationVersion`, `SedaVersion` et `Management`.

```
{
  "$schema": "http://json-schema.org/draft-04/schema",
  "type": "object",
  "additionalProperties": false,
  "patternProperties": {
    "^_": {"description": "autorise tous les champs techniques"},
    "^DataObjectReference$": {},
    "^ImplementationVersion$": {},
    "^SedaVersion$": {},
    "^Management$": {},
  }
  "properties": {
    "_og": {"type": "string"},
    "DescriptionLevel": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

À l’exception des items précédemment cités, la solution logicielle Vitam **ne supporte actuellement pas les règles** suivantes au niveau de l’en-tête :

- **minProperties** : entier indiquant le nombre minimum d’éléments attendus dans une unité archivistique ;
S’il n’est pas renseigné, le nombre minimum d’éléments attendus est égal à 0.
- **maxProperties** : entier indiquant le nombre maximum d’éléments attendus dans une unité archivistique ;
- **dependencies** : liste permettant de définir des relations de dépendance entre

éléments ;

- **additionalProperties** : liste permettant de préciser et contrôler les éléments qui ne sont pas définis dans la liste des propriétés du schéma de contrôle ;
- **anyOf** : liste imposant qu’au moins l’une des configurations déclarées soit vérifiée, quelle qu’elle soit ;
- **allOf** : liste d’éléments devant être vérifiés. Tous les éléments doivent faire l’objet d’un contrôle ;
- **oneOf** : liste imposant que les données soient exactement conformes à l’une des configurations proposées ;
- **not** : liste déclarant des éléments ne devant pas être présents dans l’unité archivistique déclarant le profil d’unité archivistique ou excluant un paramétrage en particulier.

5.2.2.1.2. Propriétés d’un schéma de contrôle

Après l’en-tête, viennent les propriétés associées au schéma de contrôle. Ces propriétés correspondent à une liste d’éléments, clés ou vocabulaires que l’on souhaite contrôler.

Si certains éléments du SEDA ne doivent pas faire l’objet d’un contrôle, il n’est pas nécessaire de les référencer dans le profil d’unité archivistique.

Généralités

Un élément (ou vocabulaire) présent dans un schéma de contrôle doit toujours avoir été préalablement créé dans l’ontologie. Dans un schéma de contrôle, il est désigné par son **nom**, qui peut être explicité par un titre et/ou une description, tous deux facultatifs et, s’ils sont utilisés, devant obligatoirement contenir des éléments textuels.

Exemple : l’élément suivant se nomme Description. Il dispose d’un titre et d’une description apportant des précisions sur son origine.

```
"Description": {
  "title": "Description",
  "description": "Element du SEDA, faisant partie du groupe DescriptionGroup."
}
```

Types

L’élément (ou vocabulaire) est associé à un **type** particulier¹².

Exemple : l’élément Description attend une chaîne de caractères, car il est de type string.

```
"Description": {
  "type": "string"
}
```

On distingue plusieurs types JSON possibles :

¹² Les types des éléments propres au SEDA sont listés dans l’annexe 3 « Types JSON ».

- « string » : texte ;
- « number » : nombre, entier ou décimal ;
- « integer » : nombre entier ;
- « boolean » : booléen dont la valeur est true ou false ;
- « object » : objet ;
- « array » : liste ou tableau de valeurs textuelles.

Leur usage est facultatif dans un schéma de contrôle, dans la mesure où l’ontologie référence un type d’indexation particulier pour chacun des vocabulaires.

Néanmoins, si un type est déclaré dans le schéma de contrôle, il doit alors être cohérent avec celui du vocabulaire tel qu’il est déclaré dans l’ontologie :

Type d’indexation dans l’ontologie	Type correspondant dans un profil d’unité archivistique		Commentaires
	Vocabulaire interne	Vocabulaire externe	
TEXT	string ou [string]	[string]	
KEYWORD	string ou [string]	[string]	
DATE	string ou [string] + pattern date	[string] + pattern date	
LONG	number ou integer [number] ou [integer]	[number] ou [integer]	
DOUBLE	number ou [number]	[number]	
BOOLEAN	boolean ou [boolean]	[boolean]	
GEO_POINT	string	[string]	L’équipe Vitam n’a pas investigué sur les usages de ces deux types d’indexation.
ENUM	[string] + pattern énumératif	[string] + pattern énumératif	

Par analogie au SEDA et au langage XML, il convient de prêter attention aux éléments suivants :

- sera qualifié en objet un élément contenant des sous-éléments, par exemple : Management, Writer, Keyword ;
- sera qualifié en tableau (ou array) :
 - un élément répétable, tel que Tag ou OriginatingAgencyArchiveUnitIdentifier,
 - un vocabulaire externe ;
- certains éléments du SEDA (PreventInheritance, NeedAuthorization, NeedReassessingAuthorization) doivent contenir un booléen.

Dans une unité archivistique, on trouvera essentiellement les types suivants :

- pour les **vocabulaires internes**, propres au SEDA, les principaux types rencontrés sont : « string », « array » et « object », auxquels s’ajoutent un unique « boolean »¹³.
- quant aux **vocabulaires externes**, ajoutés pour répondre à des besoins et transferts spécifiques, la solution logicielle Vitam les identifie systématiquement comme des array (ou tableaux), c’est-à-dire des éléments répétables. Ces tableaux peuvent inclure ensuite un type particulier de chaînes : texte, entier, décimal ou booléen.

Les éléments de type « array » et « object » ont une structuration plus complexe qu’un type simple :

- un élément de type « array » peut contenir une liste ou un tableau de valeurs textuelles (les **items**). Cette liste est obligatoire quand l’élément ne correspond pas à une chaîne de caractères textuelles (type par défaut d’un tableau). La liste permet alors de caractériser un type particulier au tableau (un entier ou un objet par exemple).

Par exemple : Le vocabulaire externe AgeDuCapitaine est doté d’un type « array » dans le schéma de contrôle d’une unité archivistique. Attendant un entier, il est caractérisé par un item de type « integer ».

```
"AgeDuCapitaine": {  
  "type": "array",  
  "items": {  
    "type": "integer"  
  }  
}
```

- un élément de type « object » définit des sous-propriétés (**properties**).

Exemple : l’élément ou vocabulaire externe RegisteredAgent est un objet qui contient les sous-propriétés ou sous-éléments FirstName, FullName, BirthName.

```
"RegisteredAgent": {  
  "items": {  
    "properties": {  
      "FirstName": {  
        "description": "C'est un string comme indiqué dans l'ontologie. Inutile de le redire ici",  
      },  
      "FullName": {  
      },  
      "BirthName": {  
      }  
    }  
  }  
}
```

Règles

Un élément peut également définir les règles suivantes :

- **\$ref** : référence à une définition déclarée dans l’en-tête du schéma de contrôle. Cette règle permet de pointer exactement vers une expression pouvant caractériser :

¹³ Les types des éléments propres au SEDA sont listés dans l’annexe 3 « Types JSON », du présent document.

- les différentes cardinalités possibles ;
- des patterns particuliers (ex : date).

Seules les propriétés du schéma de contrôle qui ne sont pas de type objet peuvent utiliser cette règle.

Exemple : L'élément Tag fait référence à l'expression one-one précisant qu'il a une cardinalité 1 – 1, tandis que _og est facultatif ou unique car sa cardinalité correspond à l'expression zero-one.

```
{
  "$schema": "http://json-schema.org/draft-04/schema",
  "type": "object",
  "additionalProperties": false,
  "definitions": {
    "one-one": {"type": "array", "minItems": 1, "maxItems": 1},
    "one-many": {"type": "array", "minItems": 1},
    "zero-one": {"type": "array", "minItems": 0, "maxItems": 1},
    "zero-many": {"type": "array", "minItems": 0},
    "closed-object": {"type": "object", "additionalProperties": false},
    "simple-date": {"type": "string", "pattern": "[0-9]{4}-[0-9]{2}-[0-9]{2}"}
  }
  "properties": {
    "_og": {"$ref": "#/definitions/zero-one"},
    "Title": {"type": "string"},
    "Tag": {"$ref": "#/definitions/one-one"}
  }
}
```

- **enum** : liste, permettant de signaler les valeurs autorisées. Au moins, l'une d'entre elle devra être présente pour se conformer au profil d'unité archivistique.

Exemple : L'élément DescriptionLevel requiert l'utilisation d'une valeur égale à : RecordGrp, SubGrp ou File. Si l'unité archivistique contient une autre valeur, elle ne sera pas valide.

```
"DescriptionLevel": {
  "type": "string",
  "enum": [
    "RecordGrp",
    "SubGrp",
    "File"
  ]
}
```

Certains types d'éléments peuvent spécifier des règles particulières :

- Un élément de type « string », correspondant à une chaîne de caractères, peut contenir les règles suivantes :
 - **pattern** : texte exprimant une expression régulière. Le type date n'existant pas dans le format JSON, cette règle est notamment utilisée pour préciser le format de date attendu.

Exemple : L'élément SigningTime attend une valeur correspondant à une expression régulière. Cette dernière désigne trois séries de chiffres compris entre 0 et 9, une série de 4 chiffres et deux de 2 chiffres, séparés par des tirets et se traduit par une date de type AAAA-MM-JJ.

```
"SigningTime": {  
  "description": "pattern: Date","type": "string",  
  "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"  
}
```

- **minLength, maxLength** : nombres entiers permettant de spécifier le minimum de caractères attendus comme valeur dans un champ et le maximum de caractères attendus dans un champ. Cette règle est notamment utile pour forcer à ce qu'un champ ait au moins une valeur renseignée (ex. champ Title du bordereau de transfert).

Exemple : L'élément DescriptionLanguage requiert l'utilisation d'une valeur ayant un minimum de un caractère. Si, dans le bordereau de transfert, le champ est vide (aucune valeur n'est renseignée), l'unité archivistique ne sera pas validée.

```
"DescriptionLanguage": {  
  "description": "End of LanguageGroup",  
  "type": "string",  
  "minLength": 1  
},
```

- **format** : liste permettant d'effectuer des contrôles prédéfinis pour un certain nombre de valeurs, notamment une date (vérification de la conformité de la valeur saisie par rapport au standard RFC3339¹⁴) ou une adresse électronique (vérification de la conformité de la valeur saisie par rapport au standard RFC5322¹⁵)

Exemple : L'élément EventDateTime attend une valeur correspondant à une date conforme à la RFC3339, c'est-à-dire se présentant sous la forme AAAA-MM-JJThh:mm:ss. Tout unité archivistique déclarant dans l'élément EventDateTime une date non exprimée sous ce format ne sera pas validée.

```
"EventDateTime": {  
  "description": "pattern: Date",  
  "type": "string",  
  "format": "date-time"  
},
```

- Un élément de type « number », correspondant à une valeur numérique, peut contenir les règles suivantes :
 - **minimum, maximum** : nombres entiers permettant de spécifier la valeur minimale et la valeur maximale attendues comme valeur dans un champ numérique.

Exemple : L'élément ChildrenNumber (exemple fictif) attend une valeur numérique comprise entre 0 et 22. Toute unité archivistique déclarant une valeur supérieure à 22 ne sera pas validée. En revanche, une unité archivistique déclarant une valeur de 0 ou de 22 sera acceptée.

```
"ChildrenNumber": {  
  "type": "array",
```

¹⁴ IETF, RFC3339. *Date and Time on the Internet : Timestamps*, in : <https://www.ietf.org/rfc/rfc3339.txt> (Dernière consultation : 25 septembre 2018).

¹⁵ IETF, RFC5322. *Internet Message Format*, in : <https://tools.ietf.org/html/rfc5322> (Dernière consultation : 25 septembre 2018).

```
"items": {  
  "type": "number",  
  "minimum": 0,  
  "maximum": 22  
}
```

- **exclusiveMinimum, exclusiveMaximum** : booléen dont la valeur est « true » (en minuscules) si on veut que la valeur attendue dans un champ numérique soit strictement supérieure à la valeur minimale définie ou strictement inférieure à la valeur supérieure définie.

Exemple : L’élément ChildrenNumber (exemple fictif) attend une valeur numérique structurellement comprise entre 0 et 22. Toute unité archivistique déclarant une valeur égale à 0, à 22 ou supérieure à 22 ne sera pas validée. En revanche, une unité archivistique déclarant une valeur de 1 ou de 21 sera acceptée.

```
"ChildrenNumber": {  
  "type": "array",  
  "items": {  
    "type": "number",  
    "minimum": 0,  
    "exclusiveMinimum": true,  
    "maximum": 22,  
    "exclusiveMaximum": true  
  }  
}
```

- Un élément de type « object » pourra contenir les règles :
 - **required** : liste d’éléments de type texte (= string) obligatoires dans un élément de type « object ». Ces éléments doivent ensuite être définis en tant que sous-propriétés de cet élément.
 - **additionalProperties**, de type booléen :
 - Si sa valeur est égale à « true », l’élément présent dans une unité archivistique pourra contenir des sous-éléments (métadonnées) supplémentaires, non définis dans le schéma de contrôle,
 - Si sa valeur est égale à « false », des sous-éléments non définis comme sous-propriétés d’un élément de type « object » et présents dans une unité archivistique ne seront pas conformes au profil d’unité archivistique.

La solution logicielle Vitam **ne supporte actuellement pas les règles** suivantes au niveau des propriétés :

- **anyOf** : liste imposant qu’au moins l’une des configurations déclarées soit vérifiée, quelle qu’elle soit.
- **allOf** : liste d’éléments devant être vérifiés. Tous les éléments doivent faire l’objet d’un contrôle.
- **oneOf** : liste imposant que les données soient exactement conformes à l’une des

configurations proposées.

- **not** : liste déclarant des éléments ne devant pas être présent dans l’unité archivistique déclarant le profil d’unité archivistique ou excluant un paramétrage en particulier.

Cardinalités

La gestion des cardinalités diffère en fonction du type de l’élément :

	Type(s)	Cardinalité(s) possible(s)
Vocabulaires internes	string, boolean, object	0 – 1, 1 – 1
	array (pouvant définir un sous-type particulier)	0 – 1, 1 – 1, 0 – n, 1 – n
Vocabulaires externes	array (pouvant définir un sous-type particulier)	0 – 1, 1 – 1, 0 – n, 1 – n

Pour un élément de type array (ou tableau), la cardinalité se paramètre au moyen de **minItems** et **maxItems**, nombres entiers permettant de spécifier le nombre minimal et le nombre maximal d’entrées attendues dans le tableau ou la liste de valeurs.

Exemple : La métadonnée `OriginatingAgencyArchiveUnitIdentifier` pourra être absente ou présente une fois dans le bordereau. Une unité archivistique qui déclarera plus de 2 fois la métadonnée `OriginatingAgencyArchiveUnitIdentifier` ne sera pas acceptée.

```
"OriginatingAgencyArchiveUnitIdentifier": {
  "description": "Part of IdentifierGroup",
  "type": "array",
  "minItems": "0",
  "maxItems": "1"
}
```

La métadonnée `ArchivalAgencyIArchiveUnitIdentifier` devra être présente une fois dans le bordereau. Une unité archivistique qui déclarera plus de 2 fois la métadonnée `ArchivalAgencyIArchiveUnitIdentifier` ou qui n’en a pas ne sera pas acceptée.

```
"ArchivalAgencyIArchiveUnitIdentifier": {
  "description": "Part of IdentifierGroup",
  "type": "array",
  "minItems": "1",
  "maxItems": "1"
}
```

La métadonnée `OriginatingAgencyArchiveUnitIdentifier` devra être présente au moins une fois dans le bordereau.

```
"OriginatingAgencyArchiveUnitIdentifier": {
  "description": "Part of IdentifierGroup",
  "type": "array",
```

```
"minItems": "1"
}
```

La métadonnée ArchivalAgencyIArchiveUnitidentifier pourra être absente ou présente une fois ou plusieurs fois dans le bordereau.

```
"ArchivalAgencyIArchiveUnitidentifier": {
  "description": "Part of IdentifierGroup",
  "type": "array"
}
```

Il est également possible de faire référence à une définition présente dans l’en-tête pour qualifier la cardinalité souhaitée. Cela vaut uniquement pour les propriétés qui ne sont pas incluses dans des objets.

Exemple : Language a pour cardinalité 0 – 1 et Tag 0 – n.

```
{
  "$schema": "http://json-schema.org/draft-04/schema",
  "type": "object",
  "additionalProperties": false,
  "definitions": {
    "one-one": {"type": "array", "minItems": 1, "maxItems": 1},
    "one-many": {"type": "array", "minItems": 1},
    "zero-one": {"type": "array", "minItems": 0, "maxItems": 1},
    "zero-many": {"type": "array", "minItems": 0},
    "closed-object": {"type": "object", "additionalProperties": false},
    "simple-date": {"type": "string", "pattern": "[0-9]{4}-[0-9]{2}-[0-9]{2}"}
  }
  "properties": {
    "_og": {"type": "string"},
    "DescriptionLevel": {"type": "string"},
    "Title": {"type": "string"},
    "Language": {"description": "Langue des documents", "$ref": "#/definitions/zero-one"},
    "Tag": {"description": "Tag", "$ref": "#/definitions/zero-many"},
  }
}
```

Pour les éléments de type « string », « object » ou « boolean », leur seule déclaration en tant que propriétés vaut pour une cardinalité 0 – 1. S’il faut les rendre obligatoire, il est nécessaire de les lister dans la règle **required** :

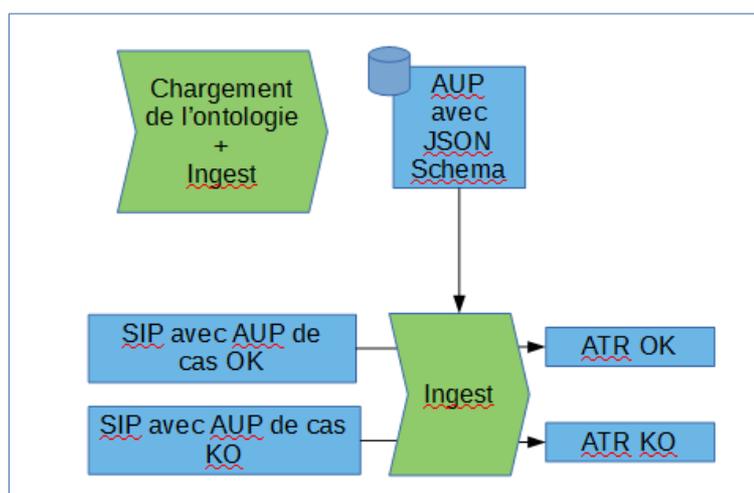
- présente dans l’en-tête pour les éléments listés comme propriétés du schéma de contrôle ;
- s’il s’agit de sous-propriétés (ex : KeywordContent, sous-élément et sous-propriété du bloc Keyword), la règle ne s’appliquera pas. Il faut alors déclarer une règle required au niveau de l’objet Keyword.

Exemple : Pour se conformer au profil, l’unité archivistique devra contenir obligatoirement les éléments DescriptionLevel et Title. En revanche, l’élément DataObjectReference y est facultatif, car il n’est pas listé

comme requis.

```
{
  "$schema": "http://json-schema.org/draft-04/schema",
  "type": "object",
  "additionalProperties": false,
  "required": ["DescriptionLevel", "Title"],
  "properties": {
    "_og": {"type": "string"},
    "DescriptionLevel": {"type": "string"},
    "Title": {"type": "string"}
  }
}
```

5.2.3. Validation d’un schéma de contrôle



Après avoir rédigé le schéma de contrôle, conforme à une unité archivistique, il est conseillé de procéder comme suit, sur une plate-forme de tests dans un premier temps :

- création et import d’une notice descriptive de profil d’unité archivistique ;
- le cas échéant, mise à jour de l’ontologie avec création de nouveaux vocabulaires ;
- dans la notice descriptive du profil d’unité archivistique, ajout du schéma de contrôle ayant fait l’objet du précédent travail.

Point d’attention : si cela n’a pas déjà été fait, il faudra veiller à ajouter un élément ArchiveUnitProfile dans les propriétés du schéma de contrôle ;

- mise à jour du bordereau de transfert pour renseigner, au niveau de chaque unité archivistique devant faire l’objet d’un contrôle, l’élément ArchiveUnitProfile avec l’identifiant attribué à ce dernier par la solution logicielle Vitam ;
- transfert du bordereau dans la solution logicielle Vitam.

Cette méthode permet de vérifier la conformité du profil d’unité archivistique créé par rapport à une unité archivistique « témoin ».

Il est conseillé de prévoir, dans cette phase de tests, des unités archivistiques non conformes,

afin de vérifier la conformité du profil d’unité archivistique à l’expression des besoins initiale.

5.2.4. Documentation d’un profil d’unité archivistique

Il est fortement recommandé de documenter le profil d’unité archivistique réalisé en réalisant sa propre modélisation, au moyen d’un schéma UML et d’un tableau récapitulatif :

Élément	Cardinalité	Type	Définitions et commentaires	Exemple	Obligatoire / Recommandé
DescriptionLevel	1.1	token	Pièce		
Title	1.1	string	Journal des transmissions		
...					

6. Conseils de mise en œuvre

À l’issue de cette première phase de réalisation de fonctionnalités concernant les profils d’unité archivistique, l’équipe projet Vitam est en mesure de fournir quelques recommandations de mise en œuvre :

Intitulé	Description	Niveau de recommandation
Élaboration d’une notice descriptive de profil d’unité archivistique		
Import des notices descriptives	Afin d’optimiser la traçabilité de la création des différentes notices, il est recommandé de créer ces dernières une par une.	Recommandé
Import des notices descriptives incluant un schéma de contrôle	La solution logicielle Vitam permet d’importer une notice descriptive incluant un schéma de contrôle. Néanmoins, ce schéma doit alors comporter des caractères d’échappement (« / »), ce qui rend son inclusion dans la notice peu pratique. De fait, il n’est pas recommandé d’importer une notice descriptive incluant un schéma de contrôle.	Non recommandé
Import du schéma de contrôle dans une notice descriptive	Il est obligatoire d’importer une notice descriptive dans la solution logicielle Vitam, puis de lui adjoindre un schéma de contrôle valide et dont les vocabulaires sont conformes à ceux qui sont déclarés dans l’ontologie.	Obligatoire
Modification des profils d’unité archivistique : notice descriptive	La solution logicielle Vitam permet de modifier les notices descriptives des profils d’unité archivistique. Cette action n’est pas recommandée quand au moins une unité archivistique déclare ce profil d’unité archivistique dans la solution logicielle Vitam.	Non recommandé
Modification des profils d’unité archivistique : désactivation des profils	La solution logicielle Vitam permet de désactiver un profil d’unité archivistique. Avant de désactiver un profil d’unité archivistique, il est recommandé de vérifier que : <ul style="list-style-type: none"> • aucun transfert nécessitant un contrôle par rapport à celui-ci n’est en cours ou programmé ; • aucune unité archivistique déjà présente dans le système ne déclare ce profil et ne fait l’objet de modifications en cours. Cette action n’est pas recommandée quand le profil d’unité archivistique a déjà été utilisé en transfert comme en mise à jour.	Recommandé
Modification des profils d’unité	La solution logicielle Vitam permet de modifier les schémas de contrôle associés aux notices descriptives.	Autorisé

archivistique : schéma de contrôle	Cette action est possible tant que le profil d’unité archivistique n’est pas déclaré par une unité archivistique.	
Modification des profils d’unité archivistique : schéma de contrôle	Il est interdit de modifier les schémas de contrôle associés aux notices descriptives quand au moins une unité archivistique déclare ce profil d’unité archivistique dans la solution logicielle Vitam.	Interdit
Modification des profils d’unité archivistique : schéma de contrôle	Si l’on souhaite modifier un schéma de contrôle déjà utilisé par au moins une unité archivistique, il est recommandé de : <ul style="list-style-type: none"> • créer un nouveau profil d’unité archivistique, comportant les modifications souhaitées ; • rechercher les unités archivistiques utilisant le profil d’unité archivistique obsolète ; • remplacer ce profil par le nouveau profil dans les unités archivistiques sélectionnées. 	Recommandé
Besoin d’un profil d’unité archivistique		
Transfert d’archives sérielles depuis une application	Dans le cadre d’un transfert de documents dits « sériels », non arborescents, produits par une application métier : il est fortement recommandé d’élaborer des profils d’unité archivistique.	Recommandé
Transfert d’archives sérielles (autres cas)	Dans le cadre d’un transfert de documents dits « sériels », obéissant <u>strictement</u> à des règles de description uniformes (par exemple, des images numérisées par un service d’archives), non nécessairement produits par une application métier : il est recommandé d’élaborer des profils d’unité archivistique.	Recommandé
Transfert ponctuel d’un type d’archives	Dans le cadre d’un transfert de documents très rarement ou très ponctuellement archivés (par exemple, un récolement de bibliothèque, un annuaire de service sous forme de tableur), il est déconseillé d’élaborer des profils d’unités archivistiques.	Déconseillé
Transfert d’archives issues de différents services producteurs	Un profil d’unité archivistique peut s’utiliser indifféremment dans différents contextes. Par exemple, un profil détaillant un modèle d’« acte de naissance » peut être utilisé pour un transfert d’archives d’un SIRH, mais également d’une application de demande de retraite. Il peut alors répondre à un besoin plus transverse de contrôle et de gestion des données dans la solution logicielle Vitam. Dans ces cas-là, il est conseillé de faire attention à ne pas trop restreindre le profil d’unité archivistique, afin	Conseillé

	de l’adapter au mieux à ses différents cas d’utilisation.	
Élaboration d’un schéma de contrôle		
Modélisation des données	Lors de la phase d’analyse et de modélisation des données, il est conseillé de chercher des correspondances (ou mapping) entre les métadonnées initiales et le modèle de données du SEDA, pris en charge nativement par la solution logicielle Vitam, avant de chercher à créer des métadonnées supplémentaires dans l’ontologie.	Recommandé
Modélisation des données	Si le profil d’unité archivistique a vocation à être utilisé dans différents contextes, il est recommandé d’adopter un modèle assez souple, notamment au niveau des cardinalités, afin que le profil d’unité archivistique puisse s’adapter à ces différents contextes.	Recommandé
Modélisation des données	Lors de la modélisation des données, il faut prendre en compte la possibilité de la mise à jour des unités archivistiques déclarant le profil d’unité archivistique. Si l’on sait que les modifications à apporter seront importantes, il est recommandé de prévoir une modélisation assez souple, notamment au niveau des cardinalités, afin que les unités archivistiques restent conformes au même profil d’unité archivistique.	Recommandé
Modélisation des données	Après avoir analysé les données à contrôler et avant de commencer à rédiger un schéma de contrôle, il est recommandé de réaliser un bordereau de transfert contenant : <ul style="list-style-type: none"> • soit une unité archivistique conforme à la modélisation préalablement réalisée, • soit plusieurs unités archivistiques, chacune faisant état de différentes propositions de modélisation. Puis, il est conseillé de le transférer sur une plateforme de tests et de récupérer un enregistrement d’unité archivistique dans la base de données MongoDB, afin qu’il serve de base de contrôle lors de la rédaction du schéma de contrôle.	Recommandé
Éléments à déclarer dans un schéma de contrôle – en-tête	La solution logicielle Vitam effectue un contrôle sur l’ensemble des informations d’une unité archivistique enregistrées dans la base de données. Certaines de ces informations étant techniques, elles n’ont aucune	Obligatoire

	<p>vocation à être contrôlées par un schéma de contrôle. Pour les écarter de son contrôle, il est obligatoire de les détailler dans une règle « patternProperties », d'autant plus si « additionalProperties » est égal à « false ».</p> <p>Sont à déclarer : les vocabulaires commençant par un underscore, ImplementationVersion, SedaVersion, ainsi que Management et DataObjectReference s'ils ne sont pas précisés en tant que propriétés dans le schéma de contrôle.</p>	
Éléments à déclarer dans un schéma de contrôle – en-tête	<p>Dans l'en-tête du schéma de contrôle, il est recommandé de déclarer une règle « additionalProperties » avec une valeur égale à « false », sans quoi le schéma de contrôle permettra à une unité archivistique de contenir des éléments supplémentaires qu'il ne définit pas.</p>	Recommandé
Éléments à déclarer dans un schéma de contrôle – en-tête	<p>L'équipe projet Vitam recommande l'utilisation des définitions dans l'en-tête des schémas afin de déterminer :</p> <ul style="list-style-type: none"> • les différentes cardinalités possibles ; • des patterns particuliers (ex : date). <p>Ces expressions sont destinées à être ensuite déclarées comme références dans les propriétés du schéma de contrôle qui ne sont pas de type objet. Elles évitent ainsi de répéter une même information à plusieurs endroits dans le schéma de contrôle.</p>	Recommandé
Éléments à déclarer dans un schéma de contrôle – en-tête	<p>La solution logicielle Vitam ne supporte actuellement pas les règles suivantes au niveau de l'en-tête : minProperties, maxProperties, dependencies, additionalProperties (liste), anyOf, allOf, oneOf, not.</p> <p>De fait, leur utilisation dans un schéma de contrôle n'est pas recommandée.</p>	Non recommandé
Éléments à déclarer dans un schéma de contrôle – propriétés	<p>Il est recommandé de déclarer dans un profil d'unité archivistique la liste des propriétés (= éléments XML du bordereau de transfert) que l'on souhaite voir contrôler.</p> <p>Si certains éléments du SEDA ne doivent pas faire l'objet d'un contrôle, il n'est pas nécessaire de les référencer dans le profil d'unité archivistique.</p>	Recommandé
Éléments à déclarer dans un schéma de	<p>L'usage des types d'indexation est facultatif dans un schéma de contrôle, dans la mesure où l'ontologie</p>	Recommandé

contrôle – propriétés	référence un type d’indexation particulier pour chacun des vocabulaires.	
Éléments à déclarer dans un schéma de contrôle – propriétés	Un vocabulaire externe doit être préalablement créé dans l’ontologie avant d’être utilisé dans un profil d’unité archivistique. Dans ce profil, si on indique son type, il faut obligatoirement le qualifier de tableau (ou array).	Obligatoire
Éléments à déclarer dans un schéma de contrôle – propriétés	Le type date n’existant pas dans le format JSON, il est obligatoire d’utiliser un pattern pour préciser le format de date attendu.	Obligatoire
Éléments à déclarer dans un schéma de contrôle – propriétés	Si on souhaite forcer à ce qu’un élément de type textuel ait au moins une valeur renseignée dans une unité archivistique, en entrée ou lors d’une mise à jour, il est recommandé d’utiliser la règle « minLength » et de lui associer une valeur égale à 1.	Recommandé
Éléments à déclarer dans un schéma de contrôle – propriétés	La solution logicielle Vitam ne supporte actuellement pas les règles suivantes au niveau des propriétés : anyOf, allOf, oneOf, not. De fait, leur utilisation dans un schéma de contrôle n’est pas recommandée.	Non recommandé

Annexe 1 : Schéma de contrôle par défaut dans la solution logicielle Vitam (état au 14/09/2018)

```
{
  "$schema": "http://vitam-json-schema.org/draft-04/schema#",
  "id": "http://example.com/root.json",
  "type": "object",
  "additionalProperties": true,
  "anyOf": [
    {
      "required": [
        "_id",
        "Management",
        "DescriptionLevel",
        "Title"
      ]
    },
    {
      "required": [
        "_id",
        "Management",
        "DescriptionLevel",
        "Title_"
      ]
    }
  ],
  "properties": {
    "_id": {
      "type": "string"
    },
    "_og": {
      "type": "string"
    },
    "Management": {
      "id": "http://example.com/root.json/ArchiveUnit/Management",
      "additionalProperties": true,
      "properties": {
        "AccessRule": {
          "type": "object",
          "id": "http://example.com/root.json/ArchiveUnit/Management/AccessRule",
          "properties": {
            "Rules": {
              "type": "array",
              "id": "http://example.com/root.json/ArchiveUnit/Management/AccessRule/Rules",
              "description": "An explanation about the purpose of this instance.",
              "items": {
                "id": "http://example.com/root.json/ArchiveUnit/Management/AccessRule/Rules/Item/0",
                "type": "object",
                "properties": {
                  "Rule": {
                    "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
                    "title": "The Rule schema.",
                    "type": "string"
                  },
                  "StartDate": {
                    "title": "The Startdate schema.",
                    "description": "pattern: Date",
                    "anyOf": [
                      {
                        "type": "string",
                        "pattern": "^[0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2}"
                      }
                    ]
                  }
                }
              }
            }
          }
        }
      }
    }
  }
}
```

```

    },
    "EndDate": {
      "title": "The Enddate schema.",
      "description": "pattern: Date",
      "anyOf": [
        {
          "type": "string",
          "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
        }
      ]
    }
  }
},
"Inheritance": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/AccessRule/Inheritance",
  "type": "object",
  "properties": {
    "PreventInheritance": {
      "id": "http://example.com/root.json/ArchiveUnit/Management/AccessRule/Inheritance/PreventInheritance",
      "description": "An explanation about the purpose of this instance.",
      "title": "The PreventInheritance schema.",
      "type": [
        "string",
        "boolean"
      ]
    },
    "PreventRulesId": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/AccessRule/Inheritance/PreventRulesId",
      "items": {
        "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
        "title": "The PreventRulesId 0 schema.",
        "type": "string"
      }
    }
  },
  "anyOf": [
    {
      "required": [
        "PreventInheritance"
      ]
    },
    {
      "required": [
        "PreventRulesId"
      ]
    }
  ]
},
"AppraisalRule": {
  "type": "object",
  "id": "http://example.com/root.json/ArchiveUnit/Management/AppraisalRule",
  "properties": {
    "Rules": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/AppraisalRule/Rules",
      "description": "An explanation about the purpose of this instance.",
      "items": {
        "id": "http://example.com/root.json/ArchiveUnit/Management/AppraisalRule/Rules/Item/0",
        "type": "object",
        "properties": {
          "Rule": {

```



```

    ]
  }
]
}
},
"ArchiveUnit": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/ArchiveUnit",
  "properties": {
    "ArchiveUnitRefId": {
      "description": "An explanation about the purpose of this instance.",
      "title": "The Archiveunitrefid schema.",
      "type": "string"
    }
  },
  "type": "object"
},
"ClassificationRule": {
  "type": "object",
  "id": "http://example.com/root.json/ArchiveUnit/Management/ClassificationRule",
  "properties": {
    "Rules": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/ClassificationRule/Rules",
      "description": "An explanation about the purpose of this instance.",
      "items": {
        "id": "http://example.com/root.json/ArchiveUnit/Management/ClassificationRule/Rules/Item/0",
        "type": "object",
        "properties": {
          "Rule": {
            "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
            "title": "The Rule schema.",
            "type": "string"
          },
          "StartDate": {
            "title": "The Startdate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "^([0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2})"
              }
            ]
          },
          "EndDate": {
            "title": "The Enddate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
              }
            ]
          }
        }
      }
    },
    "ClassificationLevel": {
      "type": "string"
    },
    "ClassificationOwner": {
      "type": "string"
    },
    "ClassificationAudience": {
      "type": "string"
    }
  }
}

```

```

},
"ClassificationReassessingDate": {
  "anyOf": [
    {
      "type": "string",
      "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"NeedReassessingAuthorization": {
  "type": "boolean"
},
"Inheritance": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/ClassificationRule/Inheritance",
  "type": "object",
  "properties": {
    "PreventInheritance": {
      "id": "http://example.com/root.json/ArchiveUnit/Management/ClassificationRule/Inheritance/PreventInheritance",
      "description": "An explanation about the purpose of this instance.",
      "title": "The PreventInheritance schema.",
      "type": [
        "string",
        "boolean"
      ]
    }
  },
  "PreventRulesId": {
    "type": "array",
    "id": "http://example.com/root.json/ArchiveUnit/Management/ClassificationRule/Inheritance/PreventRulesId",
    "items": {
      "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
      "title": "The PreventRulesId 0 schema.",
      "type": "string"
    }
  }
},
"anyOf": [
  {
    "required": [
      "PreventInheritance"
    ]
  },
  {
    "required": [
      "PreventRulesId"
    ]
  }
]
},
      "anyOf": [ {"required":["ClassificationLevel", "ClassificationOwner"]} ]
},
"DisseminationRule": {
  "type": "object",
  "id": "http://example.com/root.json/ArchiveUnit/Management/DisseminationRule",
  "properties": {
    "Rules": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/DisseminationRule/Rules",
      "description": "An explanation about the purpose of this instance.",
      "items": {
        "id": "http://example.com/root.json/ArchiveUnit/Management/DisseminationRule/Rules/Item/0",

```

```

"type": "object",
"properties": {
  "Rule": {
    "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
    "title": "The Rule schema.",
    "type": "string"
  },
  "StartDate": {
    "title": "The Startdate schema.",
    "description": "pattern: Date",
    "anyOf": [
      {
        "type": "string",
        "pattern": "^([0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2})"
      }
    ]
  },
  "EndDate": {
    "title": "The Enddate schema.",
    "description": "pattern: Date",
    "anyOf": [
      {
        "type": "string",
        "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
      }
    ]
  }
}
},
"Inheritance": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/DisseminationRule/Inheritance",
  "type": "object",
  "properties": {
    "PreventInheritance": {
      "id":
"http://example.com/root.json/ArchiveUnit/Management/DisseminationRule/Inheritance/PreventInheritance",
      "description": "An explanation about the purpose of this instance.",
      "title": "The PreventInheritance schema.",
      "type": [
        "string",
        "boolean"
      ]
    },
    "PreventRulesId": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/DisseminationRule/Inheritance/PreventRulesId",
      "items": {
        "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
        "title": "The PreventRulesId 0 schema.",
        "type": "string"
      }
    }
  }
},
"anyOf": [
  {
    "required": [
      "PreventInheritance"
    ]
  },
  {
    "required": [
      "PreventRulesId"
    ]
  }
]
}

```

```
]
}
}
},
"NeedAuthorization": {
  "description": "An explanation about the purpose of this instance.",
  "title": "The Needauthorization schema.",
  "type": "boolean"
},
"ReuseRule": {
  "type": "object",
  "id": "http://example.com/root.json/ArchiveUnit/Management/ReuseRule",
  "properties": {
    "Rules": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/ReuseRule/Rules",
      "description": "An explanation about the purpose of this instance.",
      "items": {
        "id": "http://example.com/root.json/ArchiveUnit/Management/ReuseRule/Rules/Item/0",
        "type": "object",
        "properties": {
          "Rule": {
            "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
            "title": "The Rule schema.",
            "type": "string"
          },
          "StartDate": {
            "title": "The Startdate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "^([0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2})"
              }
            ]
          },
          "EndDate": {
            "title": "The Enddate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
              }
            ]
          }
        }
      }
    }
  }
},
"Inheritance": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/ReuseRule/Inheritance",
  "type": "object",
  "properties": {
    "PreventInheritance": {
      "id": "http://example.com/root.json/ArchiveUnit/Management/ReuseRule/Inheritance/PreventInheritance",
      "description": "An explanation about the purpose of this instance.",
      "title": "The Preventinheritance schema.",
      "type": [
        "string",
        "boolean"
      ]
    },
    "PreventRulesId": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/ReuseRule/Inheritance/PreventRulesId",

```

```

    "items": {
      "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
      "title": "The PreventRulesId 0 schema.",
      "type": "string"
    }
  },
  "anyOf": [
    {
      "required": [
        "PreventInheritance"
      ]
    },
    {
      "required": [
        "PreventRulesId"
      ]
    }
  ]
}
},
"StorageRule": {
  "type": "object",
  "id": "http://example.com/root.json/ArchiveUnit/Management/StorageRule",
  "properties": {
    "Rules": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/StorageRule/Rules",
      "description": "An explanation about the purpose of this instance.",
      "items": {
        "id": "http://example.com/root.json/ArchiveUnit/Management/StorageRule/Rules/Item/0",
        "type": "object",
        "properties": {
          "Rule": {
            "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
            "title": "The Rule schema.",
            "type": "string"
          },
          "StartDate": {
            "title": "The Startdate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "^[0-8]{1}[0-9]{3}-[0-9]{2}-[0-9]{2}"
              }
            ]
          },
          "EndDate": {
            "title": "The Enddate schema.",
            "description": "pattern: Date",
            "anyOf": [
              {
                "type": "string",
                "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
              }
            ]
          }
        }
      }
    }
  }
},
"FinalAction": {
  "enum": [
    "RestrictAccess",

```

```

"Transfer",
"Copy"
]
},
"Inheritance": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/StorageRule/Inheritance",
  "type": "object",
  "properties": {
    "PreventInheritance": {
      "id": "http://example.com/root.json/ArchiveUnit/Management/StorageRule/Inheritance/PreventInheritance",
      "description": "An explanation about the purpose of this instance.",
      "title": "The PreventInheritance schema.",
      "type": [
        "string",
        "boolean"
      ]
    },
    "PreventRulesId": {
      "type": "array",
      "id": "http://example.com/root.json/ArchiveUnit/Management/StorageRule/Inheritance/PreventRulesId",
      "items": {
        "description": "NonEmptyTokenType + FIXME: attribute (xsd:ID)",
        "title": "The PreventRulesId 0 schema.",
        "type": "string"
      }
    }
  },
  "anyOf": [
    {
      "required": [
        "PreventInheritance"
      ]
    },
    {
      "required": [
        "PreventRulesId"
      ]
    }
  ]
},
"UpdateOperation": {
  "id": "http://example.com/root.json/ArchiveUnit/Management/UpdateOperation",
  "properties": {
    "SystemId": {
      "description": "An explanation about the purpose of this instance.",
      "title": "The Systemid schema.",
      "type": "string"
    },
    "ToDelete": {
      "id": "http://example.com/root.json/ArchiveUnit/Management/UpdateOperation/ToDelete",
      "properties": {
        "ArchiveUnitRefId": {
          "id":
"http://example.com/root.json/ArchiveUnit/Management/UpdateOperation/ToDelete/ArchiveUnitRefId",
          "items": {
            "description": "An explanation about the purpose of this instance.",
            "title": "The 0 schema.",
            "type": "string"
          },
          "type": "array"
        },
        "FullUpdate": {
          "description": "An explanation about the purpose of this instance.",

```



```

    "type": [
      "string",
      "number"
    ]
  }
},
"OriginatingAgencyArchiveUnitIdentifier": {
  "description": "Part of IdentifierGroup",
  "type": "array",
  "items": {
    "type": [
      "string",
      "number"
    ]
  }
},
"TransferringAgencyArchiveUnitIdentifier": {
  "description": "End of IdentifierGroup",
  "type": "array",
  "items": {
    "type": [
      "string",
      "number"
    ]
  }
},
>Description": {
  "description": "All of DescriptionGroup",
  "type": [
    "null",
    "string",
    "array"
  ]
},
"CustodialHistory": {
  "description": "All of CustodialHistoryGroup",
  "type": "object",
  "properties": {
    "CustodialHistoryItem": {
      "type": "array",
      "items": {
        "type": "string"
      }
    }
  },
  "CustodialHistoryFile": {
    "type": "object",
    "properties": {
      "DataObjectReferenceId": {
        "description": "xsd:IDREF",
        "type": "string"
      },
      "DataObjectGroupReferenceId": {
        "description": "xsd:IDREF",
        "type": "string"
      }
    }
  }
},
"required": [
  "CustodialHistoryItem"
]
},
>Type": {
  "description": "All of TypeGroup",
  "type": [

```

```
"string",
"object"
]
},
"DocumentType": {
"description": "All of DocumentTypeGroup",
"type": "string"
},
"Language": {
"description": "Start of LanguageGroup",
"type": "array",
"items": {
"type": "string"
}
},
>DescriptionLanguage": {
"description": "End of LanguageGroup",
"type": "string"
},
>Status": {
"description": "All of StatusGroup",
"type": "string"
},
"Version": {
"description": "All of VersionGroup",
"type": "string"
},
"Tag": {
"description": "Start of KeywordsGroup",
"type": "array",
"items": {
"type": "string"
}
},
"Keyword": {
"description": "End of KeywordsGroup",
"type": "array",
"items": {
"type": "object",
"properties": {
"KeywordContent": {
"type": "string"
},
"KeywordReference": {
"type": "string"
},
"KeywordType": {
"description": "How TODO ? listVersionID (attribute) + String: [@listVersionID + $, Only string]?",
"enum": [
"corpname",
"famname",
"geogname",
"name",
"occupation",
"persname",
"subject",
"genreform",
"function"
]
}
}
}
},
"Coverage": {
"type": "object",
```

```
"properties": {
  "Spatial": {
    "type": [
      "string",
      "array",
      "number"
    ]
  },
  "Temporal": {
    "type": [
      "string",
      "array",
      "number"
    ]
  },
  "Jurisdictional": {
    "type": [
      "string",
      "array",
      "number"
    ]
  }
},
"OriginatingAgency": {
  "description": "All of OriginatingAgencyGroup",
  "properties": {
    "Identifier": {
      "type": "string"
    },
    "OrganizationDescriptiveMetadata": {
      "additionalProperties": true,
      "type": "object"
    }
  },
  "type": "object"
},
"SubmissionAgency": {
  "description": "All of SubmissionAgencyGroup",
  "properties": {
    "Identifier": {
      "type": "string"
    },
    "OrganizationDescriptiveMetadata": {
      "additionalProperties": true,
      "type": "object"
    }
  },
  "type": "object"
},
"AuthorizedAgent": {
  "description": "All of AuthorizedAgentGroup",
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "FirstName": {
        "type": "string"
      },
      "BirthName": {
        "type": "string"
      },
      "GivenName": {
        "type": "string"
      }
    }
  },
  "type": "array"
}
```

```
"Gender": {
  "type": "string"
},
"BirthDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"BirthPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    }
  }
}
```

```
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"Nationality": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Corpname": {
  "type": "string"
}
},
"anyOf": [
  {
    "anyOf": [
      {
        "required": [
          "FirstName"
        ]
      },
      {
        "required": [
          "BirthName"
        ]
      },
      {
        "required": [
          "GivenName"
        ]
      },
      {
        "required": [
          "Gender"
        ]
      },
      {
        "required": [
          "BirthDate"
        ]
      },
      {
        "required": [
          "DeathDate"
        ]
      },
      {
        "required": [
          "DeathPlace"
        ]
      }
    ]
  }
]
```

```
    },
    {
      "required": [
        "Nationality"
      ]
    }
  ]
},
{
  "anyOf": [
    {
      "required": [
        "Corpname"
      ]
    }
  ]
}
]
}
},
"Writer": {
  "type": [
    "array",
    "object"
  ],
  "items": {
    "type": "object",
    "properties": {
      "FirstName": {
        "type": [
          "string",
          "null"
        ]
      },
      "BirthName": {
        "type": [
          "string",
          "null"
        ]
      },
      "GivenName": {
        "type": [
          "string",
          "null"
        ]
      },
      "Gender": {
        "type": [
          "string",
          "null"
        ]
      },
      "BirthDate": {
        "description": "pattern: Date",
        "anyOf": [
          {
            "type": "string",
            "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}[Z]?$"
          },
          {
            "type": "string",
            "format": "date-time"
          }
        ]
      }
    }
  }
},
```

```
"BirthPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^\\[0-9\\]{4}-\\[0-9\\]{2}-\\[0-9\\]{2}\\[Z]?\\$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"Nationality": {
  "items": {
    "type": "string"
  }
},
```

```
"type": "array"
},
"Identifiant": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Function": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Activity": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Position": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Role": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Mandate": {
  "type": "array",
  "items": {
    "type": "string"
  }
}
}
},
"Addressee": {
  "description": "All of AddresseeGroup (AudienceGroup)",
  "items": {
    "type": "object",
    "properties": {
      "FirstName": {
        "type": [
          "string",
          "null"
        ]
      },
      "BirthName": {
        "type": [
          "string",
          "null"
        ]
      },
      "GivenName": {
        "type": [
          "string",
          "null"
        ]
      },
      "Gender": {
```

```
"type": [
  "string",
  "null"
],
"BirthDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"BirthPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    }
  }
},
```

```
"PostalCode": {
  "type": "string"
},
"City": {
  "type": "string"
},
"Region": {
  "type": "string"
},
"Country": {
  "type": "string"
}
},
"Nationality": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  },
  "type": [
    "array",
    "string"
  ]
},
"Corpname": {
  "type": "string"
}
},
"anyOf": [
  {
    "anyOf": [
      {
        "required": [
          "FirstName"
        ]
      },
      {
        "required": [
          "BirthName"
        ]
      },
      {
        "required": [
          "GivenName"
        ]
      },
      {
        "required": [
          "Gender"
        ]
      },
      {
        "required": [
          "BirthDate"
        ]
      },
      {
        "required": [
          "DeathDate"
        ]
      }
    ]
  }
]
```

```
    },
    {
      "required": [
        "DeathPlace"
      ]
    },
    {
      "required": [
        "Nationality"
      ]
    }
  ]
},
{
  "anyOf": [
    {
      "required": [
        "Corpname"
      ]
    }
  ]
}
]
}
},
"Recipient": {
  "description": "All of RecipientGroup (AudienceGroup)",
  "items": {
    "type": "object",
    "properties": {
      "FirstName": {
        "type": [
          "string",
          "null"
        ]
      },
      "BirthName": {
        "type": [
          "string",
          "null"
        ]
      },
      "GivenName": {
        "type": [
          "string",
          "null"
        ]
      },
      "Gender": {
        "type": [
          "string",
          "null"
        ]
      },
      "BirthDate": {
        "description": "pattern: Date",
        "anyOf": [
          {
            "type": "string",
            "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
          },
          {
            "type": "string",
            "format": "date-time"
          }
        ]
      }
    }
  }
}
```

```

]
},
"BirthPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"Nationality": {
  "items": {

```

```
"type": "string"
},
"type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  },
  "type": [
    "array",
    "string"
  ]
},
"Corpname": {
  "type": "string"
}
},
"anyOf": [
  {
    "anyOf": [
      {
        "required": [
          "FirstName"
        ]
      },
      {
        "required": [
          "BirthName"
        ]
      },
      {
        "required": [
          "GivenName"
        ]
      },
      {
        "required": [
          "Gender"
        ]
      },
      {
        "required": [
          "BirthDate"
        ]
      },
      {
        "required": [
          "DeathDate"
        ]
      },
      {
        "required": [
          "DeathPlace"
        ]
      },
      {
        "required": [
          "Nationality"
        ]
      }
    ]
  },
  {
    "anyOf": [
      {
```

```

        "required": [
            "Corpname"
        ]
    }
]
}
],
},
"Source": {
    "description": "All of SourceGroup",
    "type": "string"
},
"RelatedObjectReference": {
    "description": "All of RelationGroup",
    "type": "object",
    "properties": {
        "IsVersionOf": {
            "type": "array",
            "items": {
                "type": "object",
                "properties": {
                    "ArchiveUnitRefId": {
                        "description": "TYPE: xsd:IDREF",
                        "type": "string"
                    },
                    "DataObjectReference": {
                        "type": "object",
                        "properties": {
                            "DataObjectReferenceId": {
                                "description": "TYPE: xsd:IDREF",
                                "type": "string"
                            },
                            "DataObjectGroupReferenceId": {
                                "description": "TYPE: xsd:IDREF",
                                "type": "string"
                            }
                        }
                    }
                }
            },
            "RepositoryArchiveUnitPID": {
                "description": "NonEmptyTokenType",
                "type": "string"
            },
            "RepositoryObjectPID": {
                "description": "NonEmptyTokenType",
                "type": "string"
            }
        }
    }
},
"Replaces": {
    "type": "array",
    "items": {
        "type": "object",
        "properties": {
            "ArchiveUnitRefId": {
                "description": "TYPE: xsd:IDREF",
                "type": "string"
            },
            "DataObjectReference": {
                "type": "object",
                "properties": {
                    "DataObjectReferenceId": {
                        "description": "TYPE: xsd:IDREF",
                        "type": "string"
                    }
                }
            }
        }
    }
}

```

```

    },
    "DataObjectGroupReferenceId": {
      "description": "TYPE: xsd:IDREF",
      "type": "string"
    }
  },
  "RepositoryArchiveUnitPID": {
    "description": "NonEmptyTokenType",
    "type": "string"
  },
  "RepositoryObjectPID": {
    "description": "NonEmptyTokenType",
    "type": "string"
  }
}
},
"Requires": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "ArchiveUnitRefId": {
        "description": "TYPE: xsd:IDREF",
        "type": "string"
      },
      "DataObjectReference": {
        "type": "object",
        "properties": {
          "DataObjectReferenceId": {
            "description": "TYPE: xsd:IDREF",
            "type": "string"
          },
          "DataObjectGroupReferenceId": {
            "description": "TYPE: xsd:IDREF",
            "type": "string"
          }
        }
      },
      "RepositoryArchiveUnitPID": {
        "description": "NonEmptyTokenType",
        "type": "string"
      },
      "RepositoryObjectPID": {
        "description": "NonEmptyTokenType",
        "type": "string"
      }
    }
  }
},
"IsPartOf": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "ArchiveUnitRefId": {
        "description": "TYPE: xsd:IDREF",
        "type": "string"
      },
      "DataObjectReference": {
        "type": "object",
        "properties": {
          "DataObjectReferenceId": {
            "description": "TYPE: xsd:IDREF",

```

```

        "type": "string"
    },
    "DataObjectGroupReferenceId": {
        "description": "TYPE: xsd:IDREF",
        "type": "string"
    }
},
"RepositoryArchiveUnitPID": {
    "description": "NonEmptyTokenType",
    "type": "string"
},
"RepositoryObjectPID": {
    "description": "NonEmptyTokenType",
    "type": "string"
}
}
},
"References": {
    "type": "array",
    "items": {
        "type": "object",
        "properties": {
            "ArchiveUnitRefId": {
                "description": "TYPE: xsd:IDREF",
                "type": "string"
            },
            "DataObjectReference": {
                "type": "object",
                "properties": {
                    "DataObjectReferenceId": {
                        "description": "TYPE: xsd:IDREF",
                        "type": "string"
                    },
                    "DataObjectGroupReferenceId": {
                        "description": "TYPE: xsd:IDREF",
                        "type": "string"
                    }
                }
            },
            "RepositoryArchiveUnitPID": {
                "description": "NonEmptyTokenType",
                "type": "string"
            },
            "RepositoryObjectPID": {
                "description": "NonEmptyTokenType",
                "type": "string"
            }
        }
    }
},
"CreateDate": {
    "description": "pattern: Date",
    "anyOf": [
        {
            "type": "string",
            "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
        },
        {
            "type": "string",
            "format": "date-time"
        }
    ]
},

```

```
{
  "type": "string",
  "format": "date-time-vitam"
}
],
},
"TransactedDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
    },
    {
      "type": "string",
      "format": "date-time"
    },
    {
      "type": "string",
      "format": "date-time-vitam"
    }
  ]
},
"AcquiredDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
    },
    {
      "type": "string",
      "format": "date-time"
    },
    {
      "type": "string",
      "format": "date-time-vitam"
    }
  ]
},
"SentDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
    },
    {
      "type": "string",
      "format": "date-time"
    },
    {
      "type": "string",
      "format": "date-time-vitam"
    }
  ]
},
"ReceivedDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
    },
    {

```

```
"type": "string",
"format": "date-time"
},
{
"type": "string",
"format": "date-time-vitam"
}
]
},
"RegisteredDate": {
"description": "pattern: Date",
"anyOf": [
{
"type": "string",
"pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
},
{
"type": "string",
"format": "date-time"
},
{
"type": "string",
"format": "date-time-vitam"
}
]
},
"StartDate": {
"description": "pattern: Date",
"anyOf": [
{
"type": "string",
"pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
},
{
"type": "string",
"format": "date-time"
},
{
"type": "string",
"format": "date-time-vitam"
}
]
},
"EndDate": {
"description": "pattern: Date",
"anyOf": [
{
"type": "string",
"pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
},
{
"type": "string",
"format": "date-time"
},
{
"type": "string",
"format": "date-time-vitam"
}
]
},
"Event": {
"type": [
"array",
"object"
],

```

```
"items": {
  "type": "object",
  "properties": {
    "EventIdentifier": {
      "description": "NonEmptyTokenType",
      "type": "string"
    },
    "EventType": {
      "description": "NonEmptyTokenType",
      "type": "string"
    },
    "EventDateTime": {
      "description": "pattern: Date",
      "anyOf": [
        {
          "type": "string",
          "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
        },
        {
          "type": "string",
          "format": "date-time"
        },
        {
          "type": "string",
          "format": "date-time-vitam"
        }
      ]
    },
    "EventDetail": {
      "description": "TextType",
      "type": "string"
    }
  }
},
"Signature": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "Signer": {
        "type": "array",
        "items": {
          "type": "object",
          "properties": {
            "FirstName": {
              "type": "string"
            },
            "BirthName": {
              "type": "string"
            },
            "FullName" : {
              "type": "string"
            },
            "GivenName": {
              "type": "string"
            },
            "Gender": {
              "type": "string"
            },
            "BirthDate": {
              "description": "pattern: Date",
              "anyOf": [
                {
                  "type": "string",
```

```

    "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
  },
  {
    "type": "string",
    "format": "date-time"
  }
]
},
"BirthPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^([0-9]{4}-[0-9]{2}-[0-9]{2})$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {

```

```
    "type": "string"
  }
}
},
"Nationality": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Corpname": {
  "type": "string"
},
"SigningTime": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
    }
  ]
},
"Function": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Activity": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Position": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Role": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Mandate": {
  "type": "array",
  "items": {
    "type": "string"
  }
}
},
"anyOf": [
  {
    "anyOf": [
      {
        "required": [
          "FirstName"
        ]
      }
    ]
  }
]
```

```
]
},
{
  "required": [
    "BirthName"
  ]
},
{
  "required": [
    "FullName"
  ]
},
{
  "required": [
    "GivenName"
  ]
},
{
  "required": [
    "Gender"
  ]
},
{
  "required": [
    "BirthDate"
  ]
},
{
  "required": [
    "DeathDate"
  ]
},
{
  "required": [
    "DeathPlace"
  ]
},
{
  "required": [
    "Nationality"
  ]
}
],
{
  "anyOf": [
    {
      "required": [
        "Corpname"
      ]
    }
  ]
}
],
},
"Validator": {
  "type": "object",
  "properties": {
    "FirstName": {
      "type": "string"
    },
    "BirthName": {
      "type": "string"
    }
  }
},
}
```

```
"FullName" : {
  "type": "string"
},
"GivenName": {
  "type": "string"
},
"Gender": {
  "type": "string"
},
"BirthDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"BirthPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"DeathDate": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}$"
    },
    {
      "type": "string",
      "format": "date-time"
    }
  ]
},
"DeathPlace": {
  "description": "LocationGroup",
  "type": "object",
  "properties": {
    "Geogname": {
      "type": "string"
    }
  }
}
```

```
    },
    "Address": {
      "type": "string"
    },
    "PostalCode": {
      "type": "string"
    },
    "City": {
      "type": "string"
    },
    "Region": {
      "type": "string"
    },
    "Country": {
      "type": "string"
    }
  }
},
"Nationality": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Identifier": {
  "items": {
    "type": "string"
  },
  "type": "array"
},
"Corpname": {
  "type": "string"
},
"ValidationTime": {
  "description": "pattern: Date",
  "anyOf": [
    {
      "type": "string",
      "pattern": "^[0-9]{4}-[0-9]{2}-[0-9]{2}"
    }
  ]
},
"Function": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Activity": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Position": {
  "type": "array",
  "items": {
    "type": "string"
  }
},
"Role": {
  "type": "array",
  "items": {
    "type": "string"
  }
}
```

```
},
"Mandate": {
  "type": "array",
  "items": {
    "type": "string"
  }
}
},
"anyOf": [
  {
    "anyOf": [
      {
        "required": [
          "FirstName"
        ]
      },
      {
        "required": [
          "BirthName"
        ]
      },
      {
        "required": [
          "FullName"
        ]
      },
      {
        "required": [
          "GivenName"
        ]
      },
      {
        "required": [
          "Gender"
        ]
      },
      {
        "required": [
          "BirthDate"
        ]
      },
      {
        "required": [
          "DeathDate"
        ]
      },
      {
        "required": [
          "DeathPlace"
        ]
      },
      {
        "required": [
          "Nationality"
        ]
      }
    ]
  },
  {
    "anyOf": [
      {
        "required": [
          "Corpname"
        ]
      }
    ]
  }
]
```

```

    ]
  }
]
},
"Masterdata": {
  "description": "Type: CodeType (NonEmptyTokenType)",
  "type": "object",
  "properties": {
    "Value": {
      "description": "CodeType value",
      "type": "string"
    }
  }
},
"ReferencedObject": {
  "type": "object",
  "properties": {
    "SignedObjectId": {
      "description": "xsd:IDREF",
      "type": "string"
    },
    "SignedObjectDigest": {
      "description": "String type: (FIXME: Check xsd:base64Binary OR xsd:hexBinary, can specify Format for those ?) V + FIXME: Attributes (How to do)",
      "type": "object"
    }
  }
}
}
}
},
"Gps": {
  "type": "object",
  "properties": {
    "GpsVersionID": {
      "type": "string"
    },
    "GpsAltitude": {
      "type": "integer"
    },
    "GpsAltitudeRef": {
      "type": "string"
    },
    "GpsLatitude": {
      "type": "string"
    },
    "GpsLatitudeRef": {
      "type": "string"
    },
    "GpsLongitude": {
      "type": "string"
    },
    "GpsLongitudeRef": {
      "type": "string"
    },
    "GpsDateStamp": {
      "type": "string"
    }
  }
}
}
}

```

Annexe 2 : Exemple de notice détaillée d’un profil d’unité archivistique

Nota bene : les cas présentés ci-dessous visent à vérifier la bonne mise en œuvre des mécanismes relatifs aux profils d’unité archivistique dans la solution logicielle Vitam.

Notice sans schéma de contrôle :

```
{
  "Name": "Facture",
  "Description": "Document type d'une facture associée à un dossier de marché",
  "Status": "ACTIVE",
  "ControlSchema": "{}"
}
```

Notice incluant un schéma de contrôle :

```
{
  "Identifiant": "AUP_NoticeANDSchema_1",
  "Name": "AUP_NoticeANDSchema",
  "Description": "Ce document type définit le schéma de contrôle d'une unité archivistique incluant un schéma de contrôle avec des vocabulaires externes",
  "Status": "ACTIVE",
  "ControlSchema": "{ \"$schema\": \"http://vitam-json-schema.org/draft-04/schema#\", \"id\": \"http://example.com/root.json\", \"type\": \"object\", \"additionalProperties\": false, \"properties\": { \"_id\": { \"type\": \"string\" }, \"_og\": { \"type\": \"string\" }, \"ArchiveUnitProfile\": { \"description\": \"champ obligatoire\", \"type\": [ \"string\", \"array\" ] }, \"DescriptionLevel\": { \"description\": \"champ obligatoire - valeur = item\", \"type\": \"string\", \"enum\": [ \"Item\" ], \"minItems\": 1, \"maxItems\": 1 }, \"Title\": { \"description\": \"champ obligatoire - valeur non imposee\", \"type\": \"string\", \"minLength\": 1, \"minItems\": 1, \"maxItems\": 1 }, \"MyKeyword\": { \"description\": \"mapping ES initial = KEYWORD, mises à jour possibles = TEXT\", \"type\": [ \"array\" ], \"items\": { \"type\": \"string\" } }, \"MyText\": { \"description\": \"mapping ES initial = TEXT, mises à jour possibles = KEYWORD\", \"type\": [ \"string\" ] }, \"MyDate\": { \"description\": \"mapping ES initial = DATE, mises à jour possibles = KEYWORD ou TEXT\", \"anyOf\": [ { \"type\": \"string\", \"pattern\": \"^[0-9]{4}-[0-9]{2}-[0-9]{2}$\" }, { \"type\": \"string\", \"format\": \"date-time\" }, { \"type\": \"string\", \"format\": \"date-time-vitam\" } ] }, \"MyBoolean\": { \"description\": \"mapping ES initial = BOOLEAN, mises à jour possibles = KEYWORD ou TEXT\", \"type\": [ \"array\" ], \"items\": { \"type\": \"boolean\" } }, \"MyLong\": { \"description\": \"mapping ES initial = LONG, mises à jour possibles = KEYWORD ou TEXT ou DOUBLE\", \"type\": [ \"array\" ], \"items\": { \"type\": \"number\" } }, \"MyDouble\": { \"description\": \"mapping ES initial = DOUBLE, mises à jour possibles = KEYWORD ou TEXT\", \"type\": [ \"array\" ], \"items\": { \"type\": \"number\" } }, \"MyGeoPoint\": { \"description\": \"mapping ES initial = GEO_POINT, mises à jour possibles = KEYWORD ou TEXT\", \"type\": [ \"string\" ] }, \"MyEnum\": { \"description\": \"mapping ES initial = ENUM, mises à jour possibles = KEYWORD ou TEXT\", \"type\": [ \"array\" ], \"items\": { \"type\": \"string\" } } } }"
}
```

Annexe 3 : Types JSON

Pour les éléments propres au SEDA, le tableau suivant précise les types de certains d’entre eux :

	string	number	boolean	object	array
_up	x				
_og	x				
ArchiveUnitProfile	x				
Management				x	
AccessRule				x	
AppraisalRule				x	
StorageRule				x	
ReuseRule				x	
ClassificationRule				x	
Rule	x				
StartDate	x ¹⁶				
EndDate	x ¹⁷				
FinalAction ¹⁸	x				
Inheritance				x	
PreventInheritance			x		
PreventRulesId					x ¹⁹
DescriptionLevel	x				
Title	x				x
FilePlanPosition	x				x
SystemId	x				x
OriginatingSystemId	x				x
ArchivalAgencyArchiveUnitIdentifier	x				x
OriginatingAgencyArchiveUnitIdentifier	x				x
TransferringAgencyArchiveUnitIdentifier	x				x
Description	x				x
CustodialHistory				x	
CustodialHistoryItem	x				x
Type	x				
DocumentType	x				

16 Avec un pattern date.

17 Idem.

18 Simple énumération.

19 Tableau contenant des éléments de type string.

Programme Vitam – Profils d’unité archivistique – v 3.0

Language	x ²⁰				x
DescriptionLanguage	x ²¹				
Status	x				
Version	x				
Tag	x				x
Keyword				x	x
KeywordContent	x				
KeywordReference	x				
KeywordType	x				
Coverage				x	
Spatial	x				x
Temporal	x				x
Jurisdictional	x				x
OriginatingAgency				x	
SubmissionAgency				x	
Identifier	x				
AuthorizedAgent				x	x
Writer				x	x
Addressee				x	x
Recipient				x	x
Transmitter				x	x
Sender				x	x
FirstName	x				
BirthName	x				
FullName	x				
GivenName	x				
Gender	x				
BirthDate	x ²²				
DeathDate	x ²³				
BirthPlace				x	
DeathPlace				x	
Geogname	x				
Address	x				
PostalCode	x				
City	x				
Region	x				

20 Le SEDA attend plus précisément un pattern langue.

21 Idem.

22 Avec un pattern date.

23 Idem.

Programme Vitam – Profils d’unité archivistique – v 3.0

Country	x				
Nationality	x				
Corpname	x				
Identifier	x				
Function	x				
Activity	x				
Position	x				
Role	x				
Mandate	x				
RelatedObjectReference				x	
IsVersionOf				x	
Replaces				x	
Requires				x	
IsPartOf				x	
References				x	
ArchiveUnitRefId	x				
DataObjectReference				x	
DataObjectReferenceId	x				
DataObjectGroupReferenceId	x				
RepositoryArchiveUnitPID	x				
RepositoryObjectPID	x				
CreatedDate	x				
TransactedDate	x				
AcquiredDate	x				
SentDate	x				
ReceivedDate	x				
RegisteredDate	x				
StartDate	x				
EndDate	x				
Event				x	
EventIdentifier	x				
EventTypeCode	x				
EventDataTime	x				
EventDetail	x				
OutCome	x				
OutComeDetail	x				
OutComeDetailMessage	x				
EventDetailData	x				
Signature				x	

Programme Vitam – Profils d’unité archivistique – v 3.0

Signer				x	
Validator				x	
ValidationTime	x				
MasterData	x				
ReferencedObject				x	
SignedObjectId	x				
SignedObjectDigest	x				
Gps				x	
GpsVersionId	x				
GpsAltitude	x				
GpsAltitudeRef	x				
GpsLatitude	x				
GpsLongitude	x				
GpsLongitudeRef	x				
GpsDateStamp	x				

Annexe 4 : Messages d’erreur

Nota bene : les cas présentés ci-dessous n’ont pas vocation à être exhaustif.

	Message retourné par la solution logicielle Vitam	Explication	Résolution
Import ou mise à jour d’un profil d’unité archivistique			
1	{ "Duplicate Field" : "The archive unit profile identifier AUP_IDENTIFIER_0 already exists in database" }	L'utilisateur a importé à deux reprises une notice descriptive ayant le même identifiant, sur une plateforme où l'identifiant est géré par le front-office.	Modifier l'identifiant de la seconde notice descriptive de profil d'unité archivistique.
2	{ "Mandatory Fields" : "The field Name is mandatory,The field ControlSchema is mandatory" }	La notice de profil d'unité archivistique ne contient pas deux champs obligatoires, qui sont : <ul style="list-style-type: none"> Name (= intitulé) ControlSchema (schéma de contrôle). À noter que l'élément peut être vide, mais doit obligatoirement être présent. 	Rajouter dans le fichier JSON correspondant à la notice les deux éléments obligatoires : Name et ControlSchema, ce dernier pouvant ne contenir aucune valeur.
3	{ "archiveUnitProfileCheck" : "The field AgeDuCapitaine specified in the schema is not declared in ontology" }	La notice du profil d'unité archivistique contient un schéma de contrôle. Ce dernier définit un vocabulaire externe qui n'a pas été signalé dans l'ontologie.	Ajouter dans l'ontologie le vocabulaire externe avant d'importer notice et schéma de contrôle où ce vocabulaire externe est présent.
4	{ "Invalid JSON schema" : "The field ControlSchema is not a json schema" }	La notice du profil d'unité archivistique contient un schéma de contrôle. Ce dernier n'est pas valide par rapport au format JSON attendu.	Il faut corriger le schéma de contrôle et vérifier sa validité par rapport au format JSON, avant de réimporter notice et schéma de contrôle.
5	{ "archiveUnitProfileCheck" : "The field ControlSchema is not a json schema,The archive unit profile name contains bad fields in its schema" }	Lors de l'ajout d'un schéma de contrôle à une notice descriptive, cette erreur indique que le schéma n'est pas valide par rapport au format JSON.	Il faut corriger le schéma de contrôle et vérifier sa validité par rapport au format JSON, avant de réimporter le schéma de contrôle dans la notice descriptive.
6	{ "archiveUnitProfileCheck" : "The field AUP_mail is used by an archiveUnit" }	Le schéma de contrôle n'est plus modifiable dès qu'il est utilisé par au moins une unité archivistique enregistrée dans le système.	
Import d'une unité archivistique déclarant un profil d'unité archivistique			
7	{ "ArchiveUnit":"ID3","ArchiveUnitProfile":"The declared manifest Photographies ArchiveUnitProfile status is not active","evDetTechData":"The	Le profil d'unité archivistique de l'unité archivistique dont l'identifiant est ID3 est inactif.	Activer le profil d'unité archivistique intitulé « Photographies ».

	declared manifest Photographies ArchiveUnitProfile status is not active"}}		
8	{"ArchiveUnit":"ID3","ArchiveUnitProfile":"The declared manifest Photographies ArchiveUnitProfile does not have a controlSchema","evDetTechData":"The declared manifest Photographies ArchiveUnitProfile does not have a controlSchema"}	Le profil d’unité archivistique de l’unité archivistique dont l’identifiant est ID3 ne contient pas de schéma de contrôle.	Ajouter un schéma de contrôle au profil d’unité archivistique intitulé « Photographies ».
9	{"ArchiveUnit":"ID3","ArchiveUnitProfile":"TOBECOMPLETED","evDetTechData":"Archive Unit Profile not found"}	Le profil d’unité archivistique de l’unité archivistique dont l’identifiant est ID3 n’a pas été trouvé dans le référentiel de la solution logicielle Vitam. Il y a eu : <ul style="list-style-type: none"> • soit une erreur de saisie, • soit un oubli d’intégrer le profil d’unité archivistique dans la solution logicielle Vitam. 	Il faut : <ul style="list-style-type: none"> • soit corriger l’identifiant du profil d’unité archivistique déclaré pour l’unité archivistique dont l’identifiant est ID3, • soit ajouter dans le référentiel le profil d’unité archivistique, puis saisir son identifiant dans l’unité archivistique en erreur.
10	{"ArchiveUnit":"ID3","ArchiveUnitProfile":"AUP-000001","evDetTechData":{"val idateUnitReport":{"level":"error","schema":{"loadingURI":"#","pointer":"/properties/DescriptionLevel"},"instance":{"pointer":"/DescriptionLevel"},"domain":"validation","keyword":"enum","message":"instance value (\\\"RecordGrp\\\") not found in enum (possible values: [\\\"Item\\\"])\",\"value\":\"Record Grp\",\"enum\":{\"Item\"}}}}	L’unité archivistique dont l’identifiant est ID3 contient une propriété ou élément DescriptionLevel avec « RecordGrp » pour valeur, alors que son profil d’archivage impose une valeur égale à « Item » pour cet élément.	Modifier l’élément DescriptionLevel de l’unité archivistique en remplaçant « RecordGrp » par « Item », qui est la valeur attendue et contrôlée par le profil d’unité archivistique.
11	{"ArchiveUnit":"ID3","ArchiveUnitProfile":"AUP-000001","evDetTechData":{"val idateUnitReport":{"level":"error","schema":{"loadingURI":"#","pointer":"/properties/Writer/items"},"instance":{"pointer":"/Writer/0"},"domain":"validation","keyword":"r	Absence d’une métadonnée attendue (Identifiant).	Corriger le bordereau de transfert pour rajouter la métadonnée attendue.

	<p>required\", \"message\": \"object has missing required properties</p> <p>([\\\"Identifieur\\\"])\", \"required\": [\\\"BirthName\\\", \\\"FirstName\\\", \\\"Identifier\\\", \\\"missing\\\": [\\\"Identifieur\\\"]}]\"}</p>		
12	<p>{\"ArchiveUnit\": \"ID3\", \"ArchiveUnitProfile\": \"AUP-000001\", \"evDetTechData\": {\"validateUnitReport\": {\"level\": \"error\", \"schema\": {\"loadingURI\": \"#\", \"pointer\": \"\", \"instance\": {\"pointer\": \"\", \"domain\": \"validation\", \"keyword\": \"additionalProperties\", \"message\": \"object instance has properties which are not allowed by the schema:</p> <p>[\\\"ArchivalAgencyArchiveUnitIdentifier\\\"]\", \"unwanted\": [\\\"ArchivalAgencyArchiveUnitIdentifier\\\"]}]\"}</p>	<p>Présence dans le bordereau de transfert d’une métadonnée non autorisée dans le schéma (ArchivalAgencyArchiveUnitIdentifier)</p>	<p>3 solutions :</p> <ul style="list-style-type: none"> • modifier la valeur du champ additionalProperties dans l’en-tête du schéma de contrôle pour autoriser les métadonnées supplémentaires • déclarer dans le schéma de contrôle l’objet correspondant à la métadonnée ArchivalAgencyArchiveUnitIdentifier • modifier le manifeste du SIP pour retirer la métadonnée correspondante
13	<p>{\"ArchiveUnit\": \"ID3\", \"ArchiveUnitProfile\": \"AUP-000003\", \"evDetTechData\": {\"validateUnitReport\": {\"level\": \"error\", \"schema\": {\"loadingURI\": \"#\", \"pointer\": \"\", \"instance\": {\"pointer\": \"/Tag\", \"domain\": \"validation\", \"keyword\": \"maxItems\", \"message\": \"array is too long: must have at most 1 elements but instance has 3 elements\", \"maxItems\": 1, \"found\": 3}]\"}</p>	<p>L’élément Tag, présent dans l’unitulé archivistique dont l’identifiant est ID3 est répété à trois reprises, alors que le profil d’unité archivistique n’en requiert qu’une seule occurrence.</p>	<p>Enlever au moins deux des trois éléments Tag dans l’unité archivistique.</p>
14	<p>{\"ArchiveUnit\": \"ID6\", \"ArchiveUnitProfile\": \"AUP-000018\", \"evDetTechData\": {\"validateUnitReport\": {\"level\": \"error\", \"schema\": {\"loadingURI\": \"#\", \"pointer\": \"\", \"instance\": {\"pointer\": \"/Age/0\", \"domain\": \"validation\", \"keyword\": \"maximum\", \"message\": \"numeric instance is greater than the required</p>	<p>L’unité archivistique doit l’identifiant est ID6 contient un vocabulaire Age avec une valeur égale à « 158 », alors que sa valeur maximale doit être égale à 150.</p>	<p>Corriger l’unité archivistique, en précisant un âge inférieur à 150.</p>

	<p>maximum (maximum: 150, found: 158)\" ,\"maximum\":150,\"found\":158}}]\"}</p>		
15	<pre> {"ArchiveUnit":"ID6","ArchiveUnitProfile":"AUP-000018","evDetTechData":{"val idateUnitReport": [{"level":"error","schema":{"loadingURI":"#","pointer":"/properties/Beneficiary"},"instance":{"pointer":"/Beneficiary"},"domain":"validation","keyword":"minItems","message":"array is too short: must have at least 1 elements but instance has 0 elements","minItems":1,"found":0}}]} </pre>	<p>Le vocabulaire Beneficiary présent dans l’unité archivistique dont l’identifiant est ID6 est absent, alors que le profil d’unité archivistique attend au moins une occurrence de ce vocabulaire, ainsi qu’un nombre minimal de caractères.</p>	<p>Ajouter au moins un vocabulaire Beneficiary dans l’unité archivistique, contenant une chaîne de caractères minimale.</p>
16	<pre> {"ArchiveUnit":"ID26","ArchiveUnitProfile":"AUP-000002","evDetTechData":{"val idateUnitReport": [{"level":"error","schema":{"loadingURI":"#","pointer":"","instance":{"pointer":"","domain":"validation","keyword":"additionalProperties","message":"object instance has properties which are not allowed by the schema: [\\"DataObjectReference\\",\\"Management\\"]"},"unwanted":["DataObjectReference","Management"]}]}]} </pre>	<p>L’en-tête du profil déclare la valeur « false » dans le champ additionalProperties. Or l’objet JSON correspondant au bloc DataObjectReference du schéma SEDA n’est pas déclaré dans le schéma de contrôle et l’unité archivistique déclare un groupe d’objets dans le bordereau de transfert. Donc le schéma de l’unité archivistique n’est pas conforme à son schéma de contrôle.</p>	<p>2 solutions :</p> <ul style="list-style-type: none"> • modifier la valeur du champ additionalProperties dans l’en-tête du schéma de contrôle • déclarer dans le schéma de contrôle l’objet correspondant au bloc DataObjectReference du schéma SEDA
17	<pre> {"ArchiveUnit":"ID3","ArchiveUnitProfile":"AUP-000005","evDetTechData":{"val idateUnitReport": [{"level":"warning","schema":{"loadingURI":"#","pointer":"/properties/Management/properties/AppraisalRule"},"domain":"syntax","message":"the following keywords are unknown and will be ignored: [FinalAction]"},"ignored":["FinalAction"]}, {"level":"error","schema":{"loadingURI":"#","pointer":"","instance":{"pointer":"","domain":"val </pre>	<p>Présence dans le bordereau de transfert d’une métadonnée non autorisée dans le schéma</p>	<p>3 solutions :</p> <ul style="list-style-type: none"> • modifier la valeur du champ additionalProperties dans l’en-tête du schéma de contrôle pour autoriser les métadonnées supplémentaires • déclarer dans le schéma de contrôle l’objet correspondant à la métadonnée ChildrenNumber • modifier le manifeste du SIP pour retirer la métadonnée correspondante

	<p>idation\", \"keyword\": \"additional Properties\", \"message\": \"object instance has properties which are not allowed by the schema: [\\\"ChildrenNumber\\\"]\", \"unwanted\": [\\\"ChildrenNumber\\\"]}\"}</p>		
18	<pre>{ "ArchiveUnit": "ID3", "ArchiveUnitProfile": "AUP-000005", "evDetTechData": { "validationUnitReport": { "level": "warning", "schema": { "loadingURI": "#", "pointer": "/properties/Management/properties/AppraisalRule", "domain": "syntax", "message": "the following keywords are unknown and will be ignored: [FinalAction], ignored: [FinalAction]", "level": "error", "schema": { "loadingURI": "#", "pointer": "", "instance": { "pointer": "", "domain": "validation", "keyword": "anyOf", "message": "instance failed to match at least one required schema among 1, nrSchemas: 1, reports: {/anyOf/0": { "level": "error", "schema": { "loadingURI": "#", "pointer": "/anyOf/0", "instance": { "pointer": "", "domain": "validation", "keyword": "required", "message": "object has missing required properties ([\\\"ChildrenNumber\\\"])\", required: [ChildrenNumber, DescriptionLevel, Management, RecruitmentDate, StartDate, Title], missing: [ChildrenNumber]}}}}}}}</pre>	<p>Absence d'une métadonnée attendue (ChildrenNumber).</p>	<p>Corriger le bordereau de transfert pour rajouter la métadonnée attendue.</p>
19	<pre>{ "ArchiveUnit": "ID3", "ArchiveUnitProfile": "AUP-000005", "evDetTechData": { "validationUnitReport": { "level": "warning", "schema": { "loadingURI": "#", "pointer": "/properties/Management/properties/AppraisalRule", "domain": "syntax", "message": "the following keywords are unknown and will be ignored:</pre>	<p>La valeur du champ ChildrenNumber est incompatible avec le type déclaré dans le profil d'unité archivistique.</p>	<p>Corriger le manifeste du SIP pour rendre la valeur du champ ChildrenNumber compatible avec le type déclaré dans le profil d'unité archivistique.</p>

<pre>[FinalAction]\,"ignored": ["FinalAction"]}, {"level":"error","schema": {"loadingURI":"#", "pointer":\ "/properties/ChildrenNumber"},\ "instance": {"pointer":"/ChildrenNumber" }, "domain":"validation", "key word":"type", "message":"inst ance type (array) does not match any allowed primitive type (allowed: [["integer","number"])\," found":"array", "expected": ["integer", "number"]}, {"level":"warning", "schema": {"loadingURI":"#", "pointer":\ "/properties/Management/properti es/AppraisalRule"}, "domain":" syntax", "message":"the following keywords are unknown and will be ignored: [FinalAction]\,"ignored": ["FinalAction"]}]}"}</pre>		
--	--	--