

## Backlog RM - Fonctionnalité #13332

Fonctionnalité # 13326 (Rejeté): [SEDA 2] Prise en compte des extensions

### [SEDA 2] Gérer les métadonnées complémentaires dans content

26/02/2020 11:39 - Elodie SOME-BLAD

<b>Statut:</b> Rejeté	<b>Début:</b> 26/02/2020
<b>Priorité:</b> 2-Sérieux	<b>Echéance:</b>
<b>Assigné à:</b> Cyril VAZQUEZ	
<b>Catégorie:</b>	
<b>Version cible:</b> Inscription Backlog	
<b>Tags RM:</b>	

**Description**

En tant que SA, je veux utiliser un schéma de métadonnées descriptives étendu pour les unités d'archive afin de correspondre à des besoins du métier des services producteurs.

**Analyse**

Implique l'extension du schéma SEDA2 dans un nouvel espace de nom (ou en remplacement de l'espace de nom "officiel" ce qui ne serait pas la bonne pratique);

Deux options :

- intégrer un schéma custom pour les clients souhaitant étendre le SEDA 2 (donc lier le namespace du bordereau reçu à une config qui fournit le chemin du fichier xsd)
- utiliser une configuration pour étendre à la volée (runtime), un peu à la manière de nos extension de profil d'archive (en modifiant la XSD DOM)

Dans les deux cas, étudier la pertinence et la valeur avant d'entamer des travaux plus complexes.

**Travaux réalisés**

Solution alternative :

- Proposer un schéma de paquet "SEDA 2.1 ouvert" qui ouvre la possibilité d'ajouter des balises "libres" à Content
- Si ajout au lieu de remplacement du schéma SEDA, ajouter de la configuration à "[medona] packageSchemas" pour indiquer le schéma à utiliser (chemin xsd en dur aujourd'hui pour SEDA 1 et 2)

Ensuite, plusieurs options :

- Traiter les champs complémentaires dans l'objet Content en mode "lax" (laxiste, oui) avec la dépendance xml : une balise est transformée
  - en chaîne si cardinalité 1 et aucun enfant
  - en tableau si cardinalité multiple
  - en objet si elle contient des enfants

Problème: que fait-on sans schéma de métadonnées en entrée si certaines archives on une chaîne, d'autre un tableau de chaînes

- Traiter les champs complémentaires avec un schéma non xml (json, interne ?) afin d'avoir une description, nécessaire pour les accès aux métadonnées en consultation et modification.

Problème: il s'agit d'ajouter des propriétés à un objet existant (Content), pas de décrire un nouvel objet, donc l'extension passe forcément par une classe "Content" étendue aussi...

- Traiter uniquement les champs complémentaires scalaires (chaîne, booléen, date, nombre) avec une cardinalité de 1, comme le socle pour les métadonnées utilisateur. Ensuite, le schéma de description des archives pourra "raccrocher" avec une correspondance exacte des noms

==> proposition sur branche feat/13332 de Maarch RM et archivesPubliques

#### Historique

#1 - 27/02/2020 12:10 - Elodie SOME-BLAD

- Description mis à jour

**#2 - 09/03/2020 16:07 - Cyril VAZQUEZ**

- Description mis à jour

- Statut changé de A traiter à R&D - A étudier

**#3 - 10/03/2020 16:30 - Cyril VAZQUEZ**

- Description mis à jour

**#4 - 10/03/2020 16:49 - Cyril VAZQUEZ**

- Description mis à jour

**#5 - 10/03/2020 17:21 - Cyril VAZQUEZ**

- Description mis à jour

**#6 - 11/08/2020 14:53 - Elodie SOME-BLAD**

- Version cible changé de 2.7 à Product Backlog

**#7 - 09/06/2021 14:57 - Emmanuel DILLARD**

- Version cible mis à Inscription Backlog

**#8 - 10/01/2022 14:30 - Cyril VAZQUEZ**

- Statut changé de R&D - A étudier à Rejeté

Rendu inutile par la conversion AIP V3