Anne Bugner (stagiaire école ENC)
27/07/2023

Récapitulatif de l'avancée du Projet Couleur BnF Manuscrits

Extraction des données d'Agorha

Chantier terminé - écriture de scripts Python[1] :

- **Premier type de scripts : pour l'extraction** toutes les **notices de type « Œuvre »** des bases 88 et 89 d'Agorha, en format **JSON-LD** (à partir du **code HTML** de la page des résultats et des identifiants de connexion personnels),

    - **Concaténation de tous les fichiers JSON-LD** en **un seul JSON-LD** pour chaque base. (total des scripts : 4 : 2 pour chaque base).

    - Extraction des données en **tout autre format disponible sur Agorh**a (JSON, rdf, rdf3, ttl), sous réserve de posséder des accès en tant que contributeur·ice.

- **Deuxième type de scripts : pour l'extraction des mêmes notices depuis l'API d'Agorha** (interrogation des bases **directement dans l'URL**, pour des résultats en format **JSON** exclusivement). **Ajustements** réalisés pour que la **procédure d'identification** ne crée pas de conflits de requêtes. (total des scripts : 2, 1 pour l'URL de chaque base.)

- **À terme**, Omeka S interrogera l'API. Mais pour travailler d'ores et déjà au schéma de transformation des données Agorha, le choix a été fait d'**entrer l'URL de téléchargement manuellement**, et de se concentrer sur l'automatisation plus tard.

---

1. ***Python*** est le **langage de programmation** choisi pour opérer ces tâches. Les instructions d'ouverture et lecture des fichiers JSON, d'extraction des informations intéressantes, de leur mémorisation et leur ré-implémentation au sein de documents tableur (CSV) seront donc formulées selon les règles de sa syntaxe.
La version du logiciel utilisée est ici **Python 3.11.1**.

TRANSFORMATION DES DONNÉES EXTRAITES

Chantiers passés

- **Analyse des données issues d'Agorha** et définition des **propriétés Omeka correspondantes** (recensement des informations ; compression de la syntaxe en Cidoc-CRM en 1 propriété choisie parmi les standards **Dublin Core et dérivés, schema.org et Cidoc-CRM**). *Document lié : tableur mapping.*

- **Installation et configuration d'Omeka S en local,** sur ordinateur personnel Windows.

- Choix des **modules complémentaires : *Advanced Resource Template*** (permet de contrôler plus finement l'édition de propriétés, et d'exporter facilement les modèles de ressources en CSV) ; ***CSV Import*** (pour la création automatisée des Contenus de chaque base) ***EasyAdmin*** (tâches de gestion et informations plus accessibles) ; ***Bulk Edit*** (pour des modifications portées sur les Contenus à grande échelle) ***Data Visualization*** (cartographie, graphiques...) ***IIIF Viewer, Mirador*** (pour l'implémentation future d'annotations IIIF)

- **Élaboration dans Omeka S de 4 modèles de ressources**, correspondant au **4 types de notices** envisagés pour le futur. Export pour sauvegarde desdits modèles en local.

- **Implémentation manuelle de contenus dans Omeka S,** pour tester les premiers rendus des sites et de l'interaction des notices. **2 étapes** sont nécessaires à la création d'un contenu :

    - La *saisie des valeurs* renseignant ses propriétés (création ex nihilo) ;

    - L'*ajout des liens internes* au sein de l'ensemble des contenus créés (modification des contenus existants)

- **Configuration des pages HTML** des 2 Sites Omeka S (interface envisagée pour les utilisateurs)

- **Analyse des disparités de saisie au sein des bases Agorha et redéfinition de la grammaire de saisie des champs, avec les contributeur.ices scientifiques des bases** (S. Mirabaud, F. Pacha-Miran, T. Knapowska, Ph. Renard, L. Checri).

    - Redéfinition de *l'ordre d'inscription des informations* au sein d'un même champ ;

- Ajout de *labels* permettant de les distinguer les unes des autres ;

- Généralisation de la *création de notices-mères* désignant l'ensemble d'un polyptyque ou d'un manuscrit, et de l'indication des relations hiérarchiques entre ces notices et leurs notices-filles ;

- Définition de *l'emplacement d'informations supplémentaires* (indexation des motifs iconographiques recensés dans une image, état de conservation, opérations de restauration, matériaux supposés, éléments chimiques identifiés lors d'une analyse)

- **Objectif : normaliser le contenu des champs d'information dans Agorha,** afin de faciliter au maximum leur extraction automatique.

- **Côté Agorha *:***

  - **Création et correction selon les nouvelles consignes établies**. (Sont à jour : les notices créées par Teresa Knapowska depuis le 15/06/2023 sur la base Manuscrits ; les notices gérées par Philomène Renard, sur la base INHA.)

  - **Correction manuelle personnelle** d'une sélection randomisée de notices de la base Manuscrits (manuscrits grecs et syriaques principalement) pour **donner aux scripts Python des données de test suffisamment variées dans un délai immédiat**.

- **Écriture de scripts Python**, pour **extraire chaque information** visée dans les notices téléchargées en **JSON** :

  - Renseignement de chaque **chemin d'accès interne au fichier** de la notice,

  - **Enregistrement de la (ou les) valeur(s) associées** au sein de **variables** distinctes, correspondant aux Propriétés Omeka visées.

  - **1 script pour 1 modèle de ressource**.

  - **Attribution d'informations sémantiques supplémentaires** (répartition des auteurs entre *« Créateurs »*, *« Contributeurs »* et *« Inspirations »* en fonction de la valeur *« rôle »* assignée en amont

  - Distinction des expressions et des bornes chronologiques chiffrées pour les **dates**

- **Implémentation de valeurs « Null »** lorsque l'information n'est pas renseignée dans la notice JSON traitée)

- **Ajout d'une fonction de tri** entre les notices *Image* (= 1 unité picturale, soit 1 panneau ou 1 illustration sur un feuillet) et les notices *Oeuvre* (1 unité documentaire, soit 1 polyptyque ou 1 manuscrit dans son ensemble) *Critères de tri retenus :* présence ou non d'un bloc « Matérialité » au sein de la notice Agorha ; d'un bloc « Imprimé/manuscrit » ; de la mention d'un feuillet (« f. ») ou des expressions « Pelliot » et « Mexicain » dans le titre.

- **Partition du fichier JSON** en **deux listes de notices** (une pour chaque type), et **application des scripts de transformation** à chaque liste. Création des classes (ensemble des propriétés de chaque ressource, et des fonctions nécessaires à leur définition) et instanciation de ces classes à partir du fichier JSON général de chaque base.[2]

- **Tenté mais non abouti :** extraire les propriétés depuis les **fichiers JSON-LD**.

  - Mise au point de toute l'architecture logicielle nécessaire (Triple Store, scripts Python avec librairie rdflib, requêtes SPARQL)

  - **Or l' API** renverra automatiquement des **JSON seuls**,

  - **Le mapping entre JSON et JSON-LD** est encore en **version beta** (selon le SNR de l'INHA) et donc susceptible de causer des bugs.

- **Écriture de scripts** pour la création des modèles de ressource ***Motif Iconographique*** **et** ***Couche(s) de matériaux***

  - Même procédé que pour ***Oeuvre*** et ***Image*** **:** extraction des informations dans les fichiers JSON, implémentation au sein de variables.

  - **Création de la classe** ***Motif,*** de ses propriétés et contraintes (renseigner obligatoirement un motif nommé)

  - **Instanciation[3] de la classe** ***Motif*** à partir des blocs « Matérialité » des notices Agorha

2. De fait, il ne s'agit pas es fichiers JSON obtenus via l'API pour chaque base, mais de la **concaténation de toutes les notices** ***corrigées selon les dernières consignes de saisie*** **;** l'absence de cette correction faussant la création de scripts, puisqu'il est plus difficile de cerner si les bugs rencontrés proviennent d'erreurs de programmation, ou des disparités de saisies.

- **Création des classes *Couche de Motif* et *Couches Support*** pour le modèle de ressources *Couche(s) de matériaux,* qui diffèrent dans leur mode de renseignement des propriétés *Localisation de la couche, épaisseur de la couche, couches liées, Objet parent* (les Couches de Motif étant associées à un Motif iconographique, et les Couches Support à une Image ou une Oeuvre)

- **Écriture de la fonction d'attribution des instances à chaque classe** ; soit de tri entre les blocs « Matérialité » d'Agorha, en cernant s'ils portent sur un motif ou sur un support d'ensemble de l'œuvre source ;

- **Tests et corrections des bugs** de chaque script ; intégration de tous les scripts (5 pour chaque base) au sein d'1 seul document.

- Formation rapide aux **documents CSV** par le SNR (Federico Nurra) : alignement de la grammaire de saisie des données sur les habitudes de l'INHA (séparateurs en '' en fin de cellule, § pour des valeurs distinctes au sein d'une même cellule)

- **Tests d'importation automatique par CSV** sur les installations Omeka S de Federico Nurra et moi-même.

  - **Rencontre du même bug** (pour l'instant insoluble) lors de l'importation sur ces deux installations : Omeka S déplie le fichier CSV envoyé, prend note de toutes les propriétés et valeurs stockées dedans, mais ne peut les implémenter dans ses propres contenus.

  - **Hypothèse :** *CLI Path*[4] incompatible avec une installation sur système Windows. Le forum d'utilisateurs d'Omeka S évoque ce problème, sans que les solutions appliquées (à des environnements sur Linux, dans leur cas) ne soient reproductibles.

  - Autre hypothèse : nous sommes confrontés à une faiblesse du logiciel d'installation Laragon lorsque celui-ci est utilisé sur Windows.

---

3. Détermination des *instances de classe*, soit de chaque unité documentaire qui rejoindra la classe. Toute instance de classe est appelée à partager le même ensemble de propriétés.

4. Le *CLI Path* est un paramétrage obligatoire d'Omeka S, dont la fonction est de faire le lien entre les différentes applications structurant Omeka S, en PHP, et les dossiers en local de l'ordinateur propriétaire (où sont stockées nos bases de données, téléchargées depuis Agorha.) Or le renseignement d'emplacements de dossiers se fait dans le langage du *système de l'ordinateur* (*Windows*, dans notre cas) tandis que PHP semble lire en priorité la syntaxe de son *propre* langage, qui est différente et plus proche de celle des *systèmes Linux*. Les applications PHP d'Omeka S ne parviendraient donc pas à communiquer avec les fichiers contenant les databases.

- Résolution du bug renvoyée à plus tard pour mon installation personnelle.

Chantiers à venir

- **Formation PHP** pour résoudre au moins ce problème de communication entre installation Omeka S et fichiers CSV.

- **Installer Omeka S sur un poste Linux** (que je possède, bien qu'il ait, en bon matériel d'étudiante, plusieurs bugs), installer les modules *Advanced Resource Template* et *CSV Import*, et obtenir une structure fonctionnelle pour un import CSV sur Omeka S. Installation du module *Bulk Edit* pour implémentation a posteriori des liens internes entre chaque contenu (liens hiéarchiques entre Oeuvre et Image, Image et Motif, Motif et Couches ; liens documentaires entre les Couches participant au même mélange, fonctionnalité souhaitée par Sigrid Mirabaud.)

- **Import test de CSV dans Omeka S** et édition des liens internes dans HeidiSQL (gestion de la base de données d'Omeka S depuis le logiciel d'installation Laragon)

- Intégrer à chaque site, sur Omeka S, **une page de bibliographie** (HTML)

- Intégrer à chaque site également un **glossaire des termes techniques** (telles que les méthodes d'analyses physico-chimiques) **vers qui renverront les termes concernés au sein des Contenus** (modification des pages HTML de chaque Contenu par script)

- **Mettre en valeur les termes thésaurisés**, et renvoyer vers la page de leur thesaurus Agorha (modification des pages HTML de chaque Contenu par script)

- Faire en sorte que le **bilinguisme français/anglais** ne soit plus un souci (la base INHA est particulièrement concernée). Pour l'instant, Omeka S ne gère aucune traduction automatiquement, et les propriétés comme valeurs associées ne peuvent être configurées que dans 1 langue.

Préparation des visualisations

Chantiers passés

- **Écriture d'un document d'équivalences** (format JSON) entre toutes les expressions désignant une **période historique** ou temporelle thesaurisées dans Agorha, et leurs

bornes chronologiques. **Objectif :** que chaque document puisse indiquer une période de création **en chiffres,** pour des visualisations **chronologiques**.

- **Constitution d'une liste des fonctionnalités visées**, et détermination de leur **ordre de priorité** ;

    - Idées provenant de **lectures scientifiques** (travaux d'histoire matérielle mettant ces problématiques en application, travaux d'analyse des professionnels du patrimoine).

    - Et de **premiers échanges avec des professionnels du patrimoine** (laborantins suisses invités pour une campagne d'analyses XRF, Museum d'Histoire naturelle)

- **Préparation d'entretiens avec de potentiels usagers des sites :**

    - **Définition extensive des publics :** doctorant.es en histoire de l'art, en histoire médiévale et moderne, enseignant.es-chercheurs, conservateur.ices des bibliothèques et des musées, professionnels de la restauration du patrimoine, des Monuments historiques, étudiants en histoire et histoire de l'art, guides-conférencier.es professionnels, ingénieurs d'applications Web et/ou d'Humanités numériques.

    - **Constitution d'un carnet d'adresses idéal**, de **listes de questions** (1 pour chaque public), de pistes **d'ateliers collaboratifs** (autour de 1 version beta d'Omeka S, installée en local, et soumise à l'observation directe des intéressé.es, et de comparaison avec des applications similaires existantes)

    - ***Restent à définir*** : si le **planning** le permet, et le cas échéant, selon quelles **modalités** (nombre d'ateliers et de participants, nombre d'entretiens, durées, questions prioritaires à aborder)

- **Écriture d'un document associant :**

- les différents **types** de *créateurs*

- les différents degrés de **certitude** d'*analyse chimique*

- les différents degrés de **certitude** liées à la *date et au lieu de création*

…. et **3 palettes de couleurs de visualisation** (exprimées en *hexadécimal*, soit le standard HTML).
Objectif : renseigner le degré d'implication des noms cités, ainsi que le **degré de certitude** des contributeur·ices scientifiques **directement dans l'affichage** (avec légende)

Chantiers à venir

- Réflexion sur les **transformations des données Agorha à effectuer en amont** pour permettre la **visualisation des degrés de certitude** (implémentation de propriétés supplémentaires dans Omeka, mais qui ne seraient pas affichées avec les autres dans les Contenus)

- **Variations de la configuration de l'affichage des contenus dans les sites** (par œuvre seulement ? par pigment ? En élaborant des collections pour chacune de ces informations?)

- **Recherches** relatives aux **annotations IIIF** (pour l'instant mises en pause, puisque les corpus concernés dépendent encore des chantiers de Biblissima)

- **Élaboration d'un document** JSON recensant l'arborescence de **chaque composant chimique** envisageable **pour chaque *couleur* recensée.**
Objectif : faciliter les jointures entre chaque information au sein des notices *Couches*.

- Recherches relatives à la **configuration des moteurs de recherche**, et de leur potentiels d'interactivité avec les utilisateur.ices

- **Formation complémentaire** à **Javascript** (solo pour l'instant, ne sera sans doute pertinente qu'à moyen terme) pour **étendre les capacités** proposées par le module « Data Visualization » d'Omeka S