Anonymisation des identités numériques de l'Éducation nationale

Marc Berhaut
DSII du rectorat de la région académique Bretagne
96, rue d'Antrain
CS 10503
35410 Rennes Cedex 7

Bertrand Blaessinger
DSII du rectorat de la région académique Bretagne
96, rue d'Antrain
CS 10503
35410 Rennes Cedex 7

Olivier Adam
DSII du rectorat de la région académique Bretagne
96, rue d'Antrain
CS 10503
35410 Rennes Cedex 7

Sophie Schaal
DSII du rectorat de la région académique Bretagne
96, rue d'Antrain
CS 10503
35410 Rennes Cedex 7

Résumé

Le RGPD est entré en vigueur le 25 mai 2018. Il exige que les organismes mettent en œuvre les « mesures techniques et organisationnelles appropriées » pour être en mesure de démontrer leur conformité. Il est donc plus difficile d'avoir des données représentatives de nos usagers sur les plateformes de test, de qualification et de développement de nos infrastructures, sans les mesures de sécurité nécessaires. Cependant, les activités de test et de développement doivent pouvoir se faire.

Pour répondre à cet enjeu, l'académie de Rennes a mis en œuvre une solution de production de jeux de données représentatifs de la communauté éducative et de ses structures. Ces jeux de données sont utilisés afin de qualifier l'offre de services portée par le ministère de l'Éducation nationale, l'académie de Rennes et les collectivités territoriales : EduConnect, ENT territorial et le GAR.

Aujourd'hui, l'académie de Rennes est identifiée par le ministère de l'Éducation nationale comme le fournisseur du service de production de jeux de données anonymisées et est sollicitée régulièrement pour de nombreux projets pilotés par des acteurs internes et externes, nationaux et territoriaux.

Ce document présente la solution que nous avons développée – légère et reposant uniquement sur des technologies open-source, elle implémente un procédé d'anonymisation des données réelles issues de nos SI. Et nous montrerons notamment comment elle répond aux exigences de volumétrie, de cohérence et de reproductibilité. Elle permet également une grande adaptabilité au type de SI source et aux formats et grammaires de la donnée.

Mots-clefs

Identité, RGPD, CNIL, Anonymisation, Pseudonymisation, Open-source, Données à caractère personnel, DCP, Représentativité, Irréversibilité, Qualification, Éducation nationale, Confidentialité, Outil.

1 Préambule

L'ensemble des régions académiques agrègent les données issues de leurs établissements scolaires (premier et second degré, public/privé) et des SI RH (public/privé), au sein d'un « annuaire académique fédérateur » (AAF).

Cet annuaire fédère ainsi l'ensemble des « entités » numériques de la communauté éducative de la région académique (identités élèves, représentants légaux, enseignants, personnels administratifs, définition des établissements, filières, matières enseignées, déclarations des classes et groupes d'option...). Il constitue un point d'entrée déterminant pour nombre de services numériques délivrés.

Aucun accès n'est effectué directement, mais un outillage national est délivré aux académies afin de permettre à celles-ci de réaliser des exports fichiers pour un périmètre organisationnel et/ou fonctionnel donné et conformes à une grammaire définie par un schéma directeur.

Selon le périmètre, une grammaire d'export ou une autre sera adoptée.

2 Contexte d'anonymisation

Afin de disposer de jeux de données réalistes et représentatifs pour les phases de qualification d'un projet, tout en restant conforme au RGPD, l'académie de Rennes a fait le choix d'utiliser les exports AAF après les avoir au préalable anonymisés.

« Les techniques d'anonymisation peuvent apporter des garanties en matière de respect de la vie privée et peuvent servir à créer des procédés d'anonymisation efficaces, mais uniquement si leur application est correctement conçue – **ce qui suppose que les conditions préalables (le contexte) et les objectif(s) du processus d'anonymisation soient clairement définis de façon à parvenir à l'anonymisation visée, tout en produisant des données utiles.** » – Groupe de l'Article 29 [1].

Cette citation du G29, reprise par la CNIL, pointe l'importance de bien définir le contexte et les objectifs de l'anonymisation d'un ensemble de données.

La figure suivante présente ces éléments pour la région académique Bretagne.

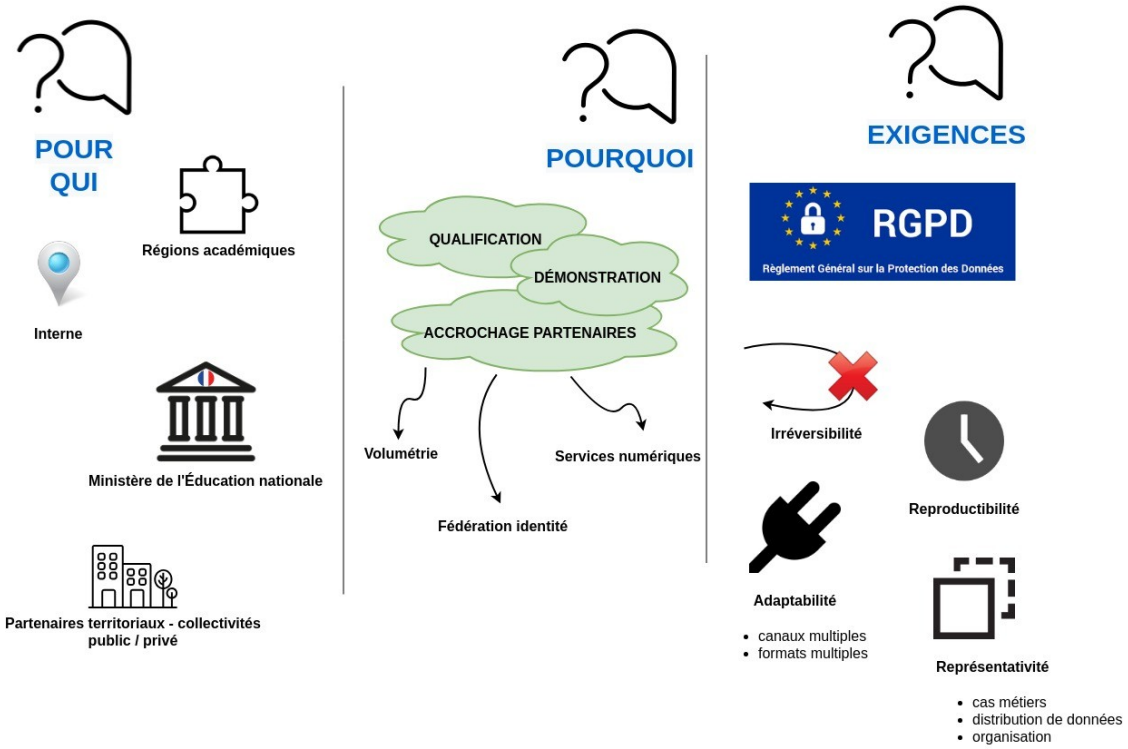

Figure 1 - Contexte d'anonymisation, objectifs et exigences

3 Traitement des données — anonymisation

3.1 Flux des données

Le processus d'anonymisation s'appuie sur des fichiers d'exports produits par un outillage national depuis l'Annuaire Académique Fédérateur (AAF).

La volumétrie de l'ensemble de données peut être très conséquente et, dans la mesure où la manipulation de fichiers à plat sur un système de fichiers n'est pas efficiente, le processus commence par importer les fichiers d'exports AAF dans une base de données de type XML implémentée par le produit BaseX. Cette technologie apporte les bénéfices suivants :

- indexe le contenu des fichiers au moment de l'import rendant les recherches extrêmement rapides (index fulltext supporté),
- supporte le langage d'interrogation standard XQuery 3.1,
- offre une interface d'interrogation par API REST HTTP,
- est open-source.

Un outil de type ETL (« Extract Transform & Load ») interroge les données indexées dans la base XML BaseX et les anonymise pour produire des fichiers à l'image des exports AAF originaux (même grammaire), comme cela est représenté par la figure suivante :

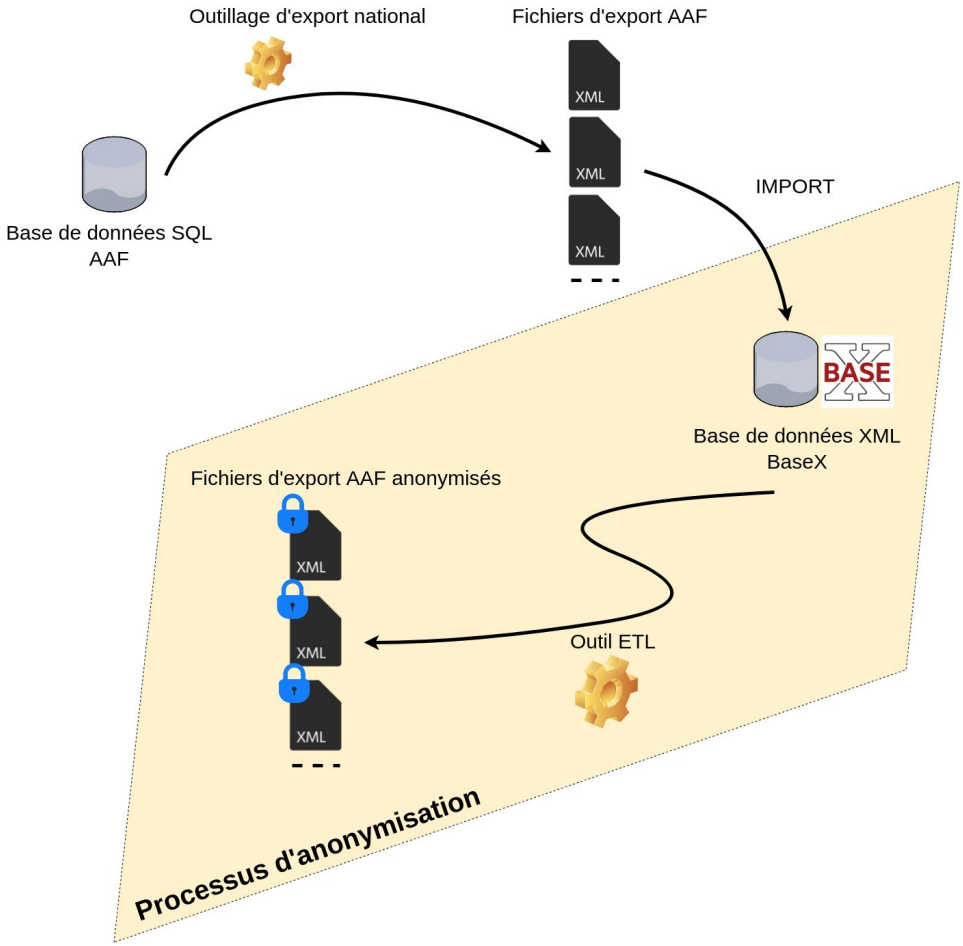

Figure 2 - Flux des données

3.2 Techniques d'anonymisation

3.2.1 Avis de la CNIL

La CNIL, par le biais du groupe de travail de l' « Article 29 » (G29 [1]), énonce deux grandes familles de techniques d'anonymisation :

- la « randomisation »,
- la « généralisation ».

Chacune possède ses forces et faiblesses au regard des trois critères d'évaluation suivants :

- **L'individualisation** : est-il toujours possible d'isoler un individu ?
- **La corrélation** : est-il possible de relier entre eux des ensembles de données distincts concernant un même individu ?
- **L'inférence** : peut-on déduire de l'information sur un individu ?

3.2.2 Techniques mises en œuvre par le processus

L'outillage développé au sein de l'ETL met en œuvre des techniques d'anonymisation de chaque famille.

- **Randomisation**
  - Utilisation de **générateurs aléatoires** de nombre, date, UUID (« Universal Unique Identifier » [3]), de login et mot de passe.
  - Utilisation d'un générateur aléatoire d'**identités fictives** (nom, prénom, adresse postale) basé sur un projet open-source « Randomuser.me ». Ce projet nous a permis de constituer un dictionnaire d'un million d'hommes et femmes dans lequel le générateur puise aléatoirement.
  - **Bruit** : le générateur de date est implémenté de telle façon que le millésime de naissance respecte un intervalle produisant une indétermination supplémentaire et garantissant un respect raisonnable de la distribution des données de l'ensemble.

- **Généralisation**
  - Le dictionnaire d'identités fictives modifie l'échelle des attributs de localité des identités en apportant une répartition nationale plutôt que régionale.

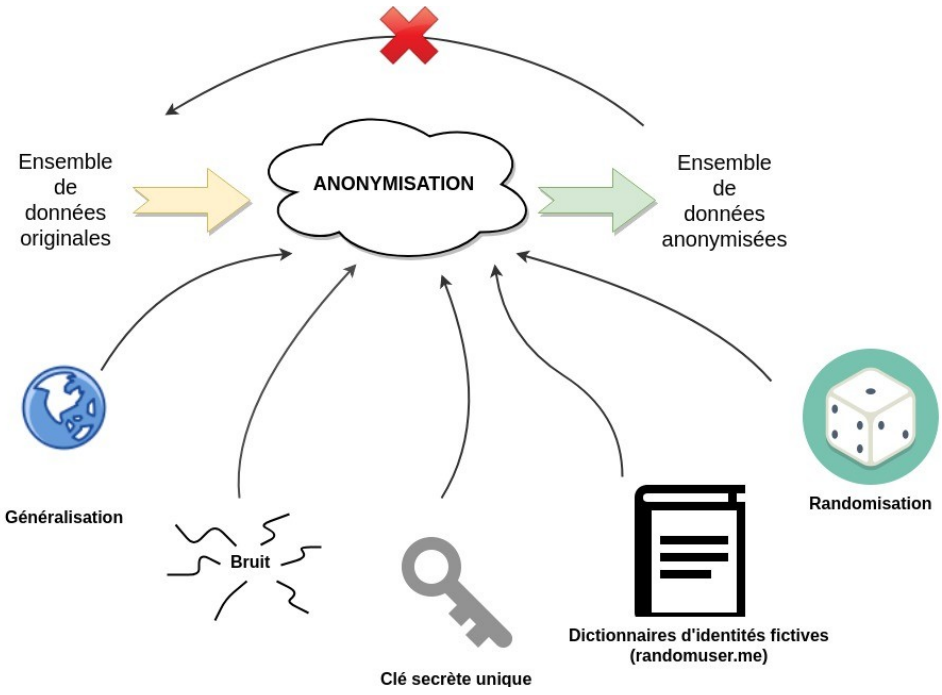

Figure 3 - Techniques d'anonymisation

3.2.3 Mesures supplémentaires de protection implémentées

De plus, une clé secrète unique, dite « clé d'anonymisation », est utilisée à chaque traitement afin d'atteindre les exigences supplémentaires suivantes :

– **Reproductibilité** de l'anonymisation d'une itération à l'autre.

L'outil ETL qui implémente le processus d'anonymisation conserve dans sa base de données des enregistrements de type « audit » afin d'être en mesure d'anonymiser de la même manière (nom, prénom, date de naissance, login, identifiant universel, adresse postale…) une entité AAF au fil des demandes.

Cela permet également d'anonymiser de façon identique une entité présente dans des exports AAF multiples, produits à intervalles réguliers dans le temps ou avec des grammaires de fichier différentes (e.g. un représentant légal ayant un enfant scolarisé en premier degré et un second dans le second degré).

Afin de ne pas conserver explicitement en base de données la relation entre une entité AAF originale et son entité anonymisée, l'ETL implémente un mécanisme reposant sur la fourniture d'un **identifiant opaque d'anonymisation**. Cet identifiant peut être retrouvé d'une itération sur l'autre par le biais de l'association {algorithme de construction, clé d'anonymisation, données identifiantes} tel que cela est représenté par la figure 4 ci-après.

– **Irréversibilité** de l’anonymisation des données

L’algorithme de production de l’identifiant opaque d’anonymisation ne conserve pas explicitement le lien entre la signature d’une entité AAF originale et son identifiant opaque. Une signature est obtenue à partir des attributs identifiants de l’entité. C’est l’empreinte « salée » de la signature qui sera conservée, le salage étant obtenu par application de la clé sur un algorithme dédié.

L’accès aux tables d’audit de la base de données ne permettra pas d’identifier la personne par des moyens susceptibles d’être raisonnablement mis en œuvre, que ce soit par le responsable du traitement ou par une tierce personne.

À noter :

– Une même personne physique peut être référencée dans l’AAF par plusieurs entités à la fois, chacune ayant un identifiant d’annuaire différent. L’algorithme de production de l’identifiant opaque d’anonymisation est capable de produire le même identifiant pour l’ensemble de ces entités en se basant sur la gestion d’une « signature » de l’entité. Ainsi, **toutes les entités se rapportant à une même personne physique seront anonymisées de la même façon** (nom, prénom, date de naissance, login...).

– La clé secrète est stockée dans un endroit sécurisé accessible uniquement par le responsable de traitement et distinct de la base de données de l’outil ETL.

– La clé secrète est unique. Chaque demande d’anonymisation d’un ensemble de données issues de l’AAF sera opérée avec une clé différente, propre au demandeur. Ainsi, une même donnée originale se verra attribuer un identifiant opaque d’anonymisation différent et sera par conséquent anonymisée de façon différente, en fonction de la demande traitée. Cela contribue à réduire la corrélation des données entre ensembles de données anonymisées.

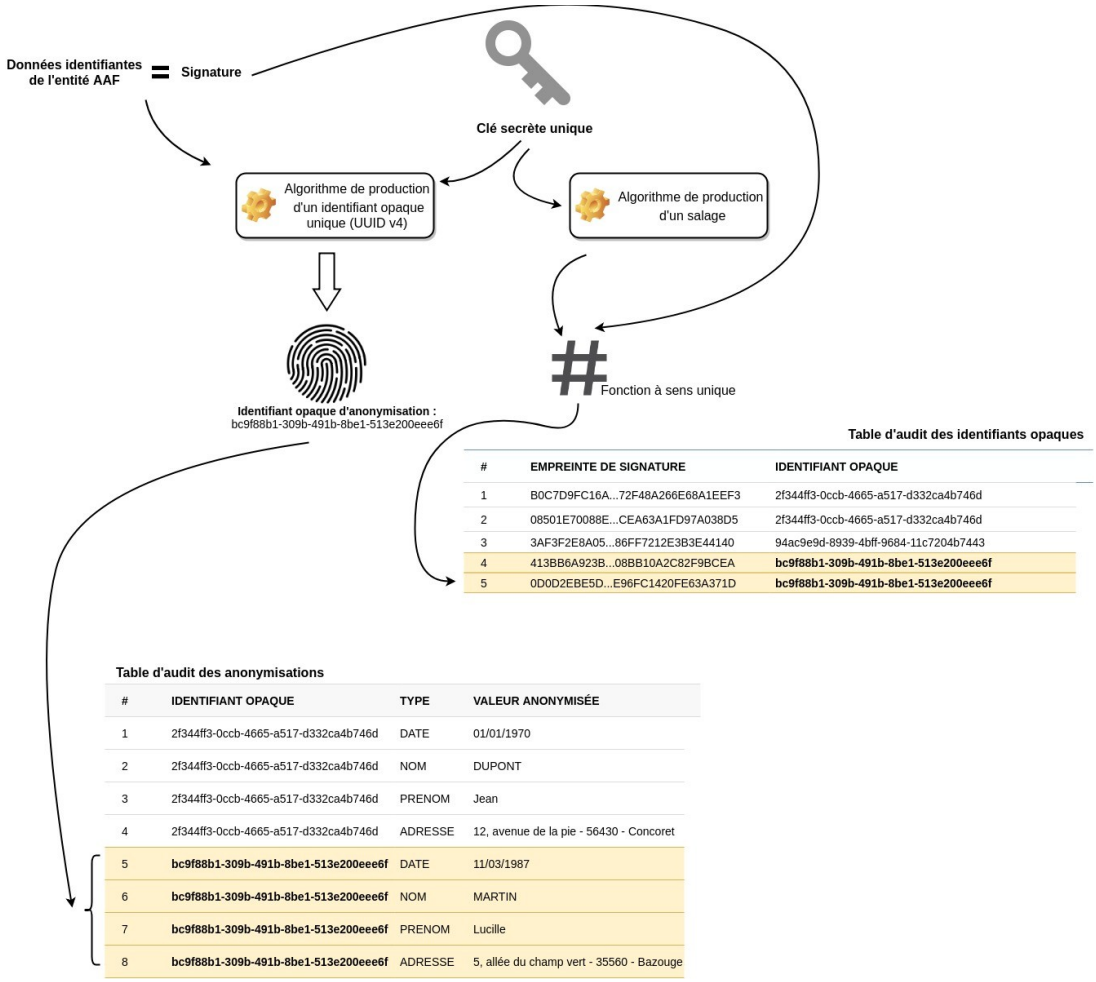

Figure 4 - Mécanismes de reproductibilité et d'irréversibilité

3.3 Caractère potentiellement identifiable des données anonymisées

3.3.1 Sécurisation des données originales

« Dans le cas où un responsable du traitement des données n'efface pas les données originales (identifiables) au niveau des événements individuels et transmet une partie de cet ensemble de données (par exemple après avoir supprimé ou masqué les données identifiables), l'ensemble de données résultant constitue encore des données à caractère personnel. » – G29 [1].

Afin de prendre en considération cet aspect juridique de l'avis rendu par le G29, et dans la mesure où les données originales sont nécessaires – indispensables au fonctionnement des services numériques développés par l'académie, elles sont stockées dans un environnement sécurisé à plusieurs niveaux.

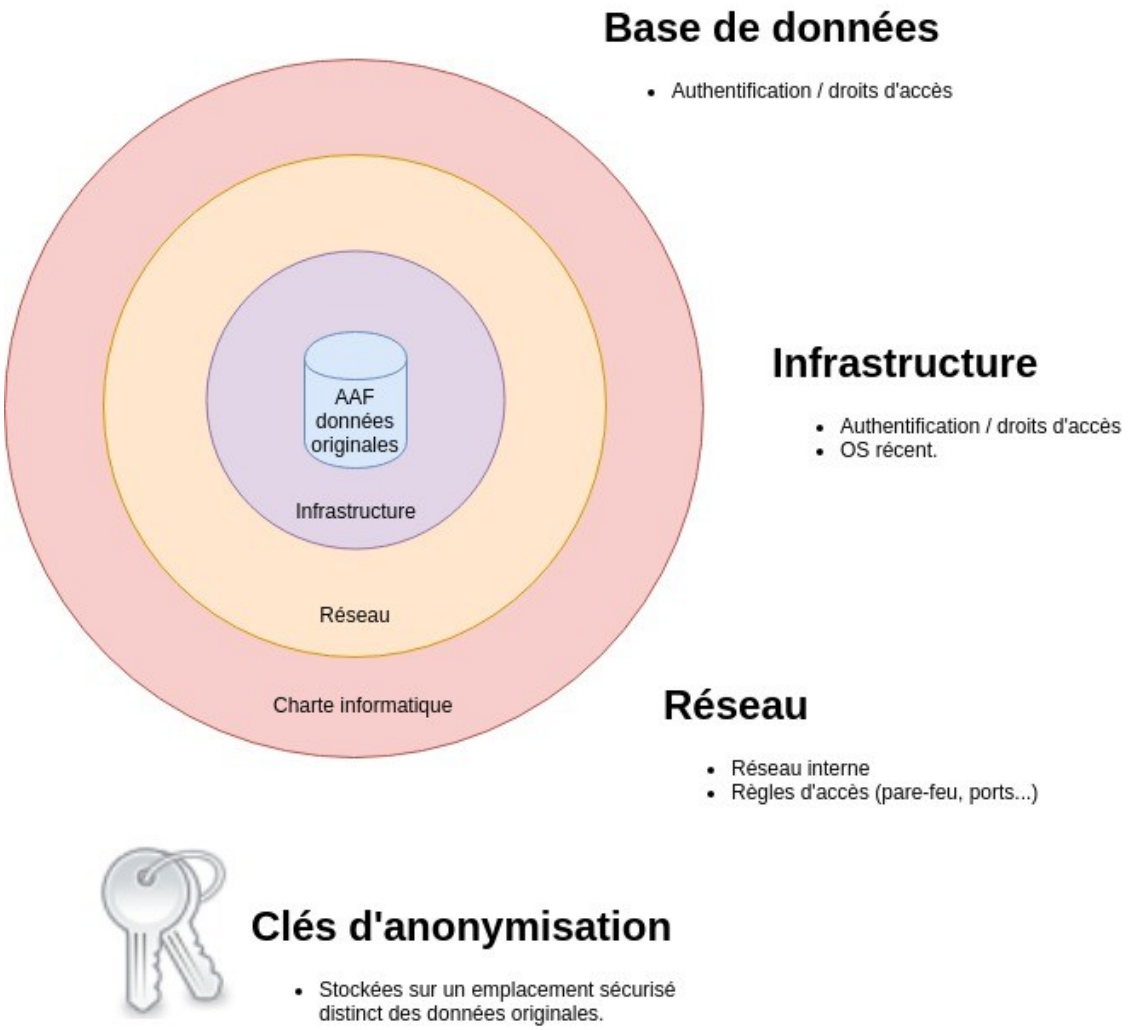

Figure 5 - Sécurisation

3.3.2 Pseudonymat

Dans la mesure où, par nécessité, la qualification des services numériques exige une représentativité forte des situations de terrain, les enregistrements des ensembles de données anonymisées peuvent être soumis à l'individualisation et la corrélation.

Dans la mesure également où les données originales ne sont pas supprimées mais conservées dans un endroit sécurisé, ou que d'autres ensembles de données publics (par exemple registres de mairie sur ses effectifs scolaires…) permettraient la corrélation, les techniques énoncées précédemment se rapportent plus à de la « pseudonymisation » en se référant aux définitions de la CNIL [1].

Néanmoins, le RGPD, dans son article 25 [2], cite nommément la pseudonymisation parmi les mesures techniques et organisationnelles appropriées pour la protection des données que doit mettre en œuvre le responsable du traitement.

4 Présentation de l'outillage

L'outil ETL qui implémente le processus d'anonymisation est un produit propriétaire de l'académie de Rennes, dont le développement en langage **Java** sous **licence GNU LGPL** a débuté il y a une dizaine d'années, initialement pour répondre au besoin d'alimentation d'un annuaire applicatif d'espace numérique de travail. Il est en constante évolution depuis.

À l'instar de tout ETL, ses missions sont d'extraire de la donnée à partir d'une source, la transformer pour ensuite la charger dans une cible.

4.1 Connecteurs

Le produit supporte de multiples connecteurs d'entrée (extraction) :

- itérer sur le contenu d'un fichier CSV,
- lister par page des fichiers sur un système de fichiers avec définition possible d'un filtre de recherche (reproduction d'un 'ls'),
- rechercher des patterns dans un fichier texte (reproduction d'un 'grep'),
- rechercher des entrées dans un annuaire LDAP (résultats paginés),
- rechercher des documents sur une instance d'ECM Nuxeo,
- obtenir une date aléatoire,
- obtenir une entité identité aléatoire (motorisé par l'API open source randomuser.me),
- obtenir un mot de passe aléatoire (avec des critères de longueur et de symboles : chiffre, lettre, caractère spécial, casse),
- rechercher des tuples dans une base de données,
- charger un document XML (et être en capacité de rechercher ses éléments via requêtage XPATH),
- interroger un web service quelconque via une API REST,
- requêter un cluster Elastic avec pagination des résultats,
- requêter une base de données XML BaseX avec pagination des résultats,

et de sortie (chargement), pour :

- écrire dans un fichier CSV,

- provisionner un annuaire LDAP,
- provisionner une instance d'ECM Nuxeo,
- provisionner une base de données relationnelle,
- chiffrer des données avec un algorithme symétrique ou asymétrique dans un fichier,
- envoyer des notifications par courriel (avec attachements possibles),
- produire un fichier dans un format quelconque (HTML, XML, CSV, propriétaire...) en se basant sur un template Freemarker,
- alimenter un cluster Elastic,
- alimenter un web service quelconque via API REST.

4.2 Transformation de la donnée

Les mécanismes de transformation s'appuient sur les technologies suivantes :

- XSLT (eXtensible Stylesheet Language).
- Freemarker (projet Apache – moteur de templating se présentant sous la forme d'une librairie Java).

4.3 Principe général

Deux modes d'utilisation de l'ETL sont possibles : avec ou sans templating Freemarker intermédiaire.

4.3.1 Mise en œuvre sans templating ou avec templating XSLT

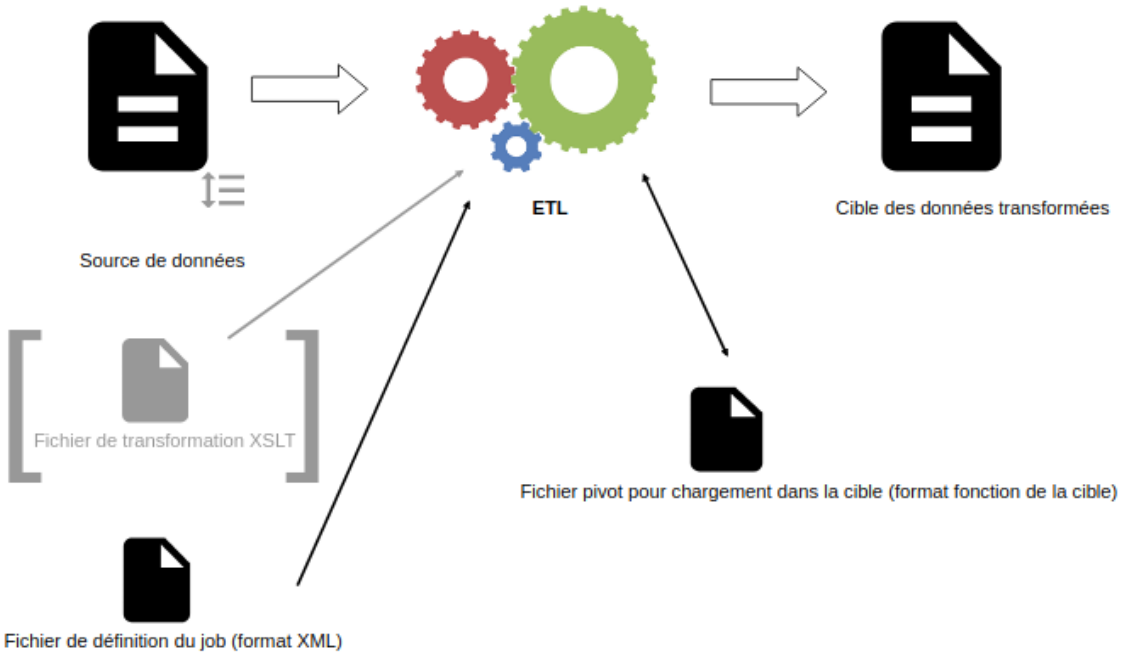

Figure 6 - Absence de templating ou avec templating XSLT

L'ETL itère sur chaque entrée de la source de données, applique la transformation et charge la donnée transformée dans la cible.

Un fichier XSLT peut être optionnellement utilisé pour appliquer une transformation sur chaque entrée du fichier source. Un fichier pivot est alors généré pour chaque entrée ainsi transformée puis chargé dans la cible.

En l'absence de fichier de transformation XSLT, un fichier pivot faisant fonction de template, variabilisé, peut être défini. Les variables seront résolues par les données issues de chaque entrée de la source. Le pivot est ensuite chargé dans la cible.

Une syntaxe particulière permet d'exécuter des traitements apparentés à des fonctions au sein du pivot avant son chargement dans la cible :

- récupérer une entrée d'un annuaire LDAP,
- contrôler l'unicité d'une entrée dans un annuaire LDAP sur plusieurs critères/attributs,

– générer un identifiant unique,

– chiffrer des données,

– normaliser des données pour respecter un format précis (uniformiser la casse et la gestion des accents, caractères spéciaux…).

Ce mode de fonctionnement présente l'avantage d'être simple et direct.

4.3.2 Mise en œuvre avec templating Freemarker

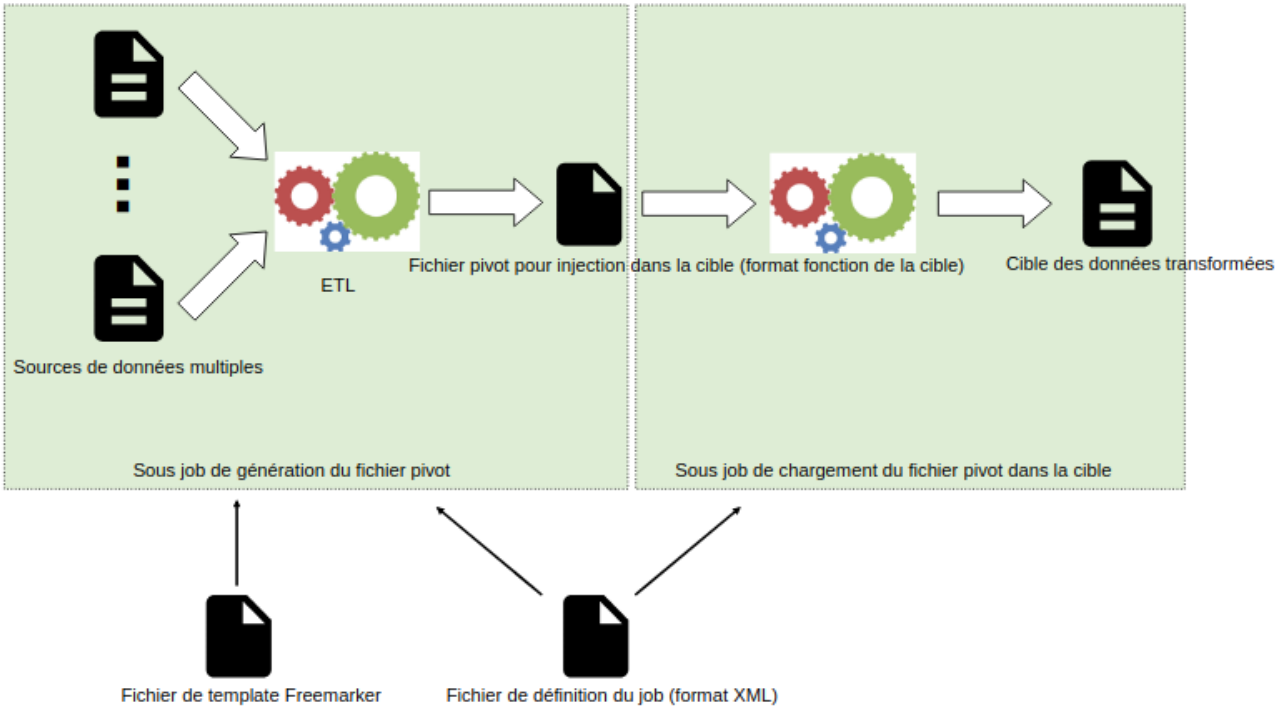

Figure 7 - Avec templating Freemarker

Deux jobs constituent les opérations d'extraction, transformation et chargement dans la cible.

Un premier job regroupe les opérations d'extraction et de transformation en s'appuyant sur le moteur de templating Freemarker.

Le fichier de template Freemarker est en capacité de requêter les multiples sources et d'opérer des contrôles ou opérations de réconciliation des données afin de produire le fichier pivot qui servira ensuite au chargement. Les itérations sont implémentées par des directives Freemarker au sein du template.

Le fichier pivot contient l'ensemble des données à charger à la différence du modèle de mise en œuvre sans Freemarker où le pivot contient les données à charger en lien avec une entrée de la source unique à chaque itération.

Le second job opère le chargement du pivot dans la cible après avoir exécuté les éventuelles fonctions embarquées dans ce dernier.

Ce deuxième mode offre le double avantage de bénéficier de la flexibilité de Freemarker afin d'implémenter des contrôles avancés et de produire un pivot qui peut être qualifié avant son chargement dans la cible.

4.4 Définition des jobs

Un job désigne un processus complet d'extraction, transformation et chargement. Dans le cas de l'utilisation avec Freemarker, la cible est le fichier pivot qui servira d'entrée ensuite.

Une multitude de jobs peuvent participer à une opération sur un système d'information (synchronisation, provisioning, migration, mise à niveau, production de statistiques...).

Il est décrit par le biais d'un fichier XML respectant une grammaire précise.

4.4.1 Template XML d'un job

```
<?xml version="1.0" encoding="UTF-8"?>
<ldapOmatic>
    <variables>
        <variable name="une_premiere_variable">V1</variable>
        <variable name="une_seconde_variable">V2</variable>
    </variables>
    <jobs>
        <job name="tous">
            <execute-job name="un premier job"/>
            <execute-job name="un second job"/>
        </job>

        <job name="un premier job" asynch="true">
            ...
        </job>
        <job name="un second job" asynch="true">
            ...
        </job>
```

```
    </jobs>
</ldapOmatic>
```

Un job est identifié par un attribut « name ».

La balise « execute-job » permet de créer un regroupement en référençant des sous-jobs, par exemple pour regrouper les jobs « atomiques » qui participent à une opération macroscopique.

L'attribut « asynch » permet de définir si le job doit être exécuté au sein d'un nouveau thread de façon asynchrone ou bien séquentiellement (mode par défaut). Dans l'exemple ci-dessus, si le job parent « tous » est sollicité, les deux sous-jobs « un premier job » et « un second job » seront exécutés en parallèle, chacun au sein d'un thread différent.

La balise « variables » permet de définir des variables qui seront ensuite accessibles depuis la définition de chaque job.

4.4.2 Exemple de définition de job sans templating Freemarker

```
<job name="exemple-job-sans-freemarker">
    <source type="ldap" name="Ldap A">
        <driver>com.sun.jndi.ldap.LdapCtxFactory</driver>
        <uri>ldap://***.in.ac-rennes.fr/ou=personnes,dc=...</uri>
        <login>***</login>
        <passwd>***</passwd>
        <query>(&amp;(objectClass=person)(title=...))</query>
        <attributeList>uid,prenom,nom,mail...</attributeList>
    </source>
    <destination type="ldap" name="Ldap B">
        <driver>com.sun.jndi.ldap.LdapCtxFactory</driver>
        <uri>ldap://***.in.ac-rennes.fr</uri>
        <login>***</login>
        <passwd>***</passwd>
        <pivot>pivot.xml</pivot>
    </destination>
</job>
```

Le fichier pivot, dans l'exemple donné, serait un fichier XML statique respectant le formalisme pour le chargement dans une cible de type annuaire. Il est variabilisé afin de recevoir les attributs de chaque élément du résultat renvoyé par la source.

Exemple de fichier pivot

```
<?xml version="1.0" encoding="UTF-8"?>
<ldapOmatic>
    <variables>
        <!--Les variables permettent d'éviter de renseigner
plusieurs fois des valeurs et aussi de calculer plusieurs fois
des fonctions-->
    </variables>
    <entries>
        <entry verifyIfExist="(&amp;(objectClass=ENTPerson)(uid=
%uid%))">
            <dn>uid=%uid%,ou=personnes,dc=...</dn>
            <attributes>
                <attr name="objectClass" modifyMode="replace">
                    <value>top</value>
                    <value>person</value>
                    <value>organizationalPerson</value>
                </attr>
                <attr name="uid" modifyMode="ignore">
                    <value>%uid%</value>
                </attr>
                <attr name="prenom" modifyMode="replace">
                    <value>%prenom%</value>
                </attr>
                <attr name="nom" modifyMode="replace">
                    <value>%nom%</value>
                </attr>
                <attr name="mail" modifyMode="replace">
                    <value>%mail%</value>
                </attr>
                <attr name="ListeRouge" modifyMode="replace">
                    <value>N</value>
                </attr>
```

```
                <attr name="Profils" modifyMode="replace">
<value>ldap://***.in.ac-rennes.fr/ou=personnes,dc=...?uid?one?
(&amp;(objectClass=ENTProfil)(member=uid=%uid%))</value>
                    <value>cn=personnel-tous</value>
                </attr>
                <attr name="Jointure" modifyMode="replace">
                    <value>(UUID)(/UUID)</value>
                </attr>
            </attributes>
        </entry>
    </entries>
</ldapOmatic>
```

Les attributs de chaque élément du résultat renvoyé par la source sont référencés, de la même façon que les variables du job, par un encadrement avec le caractère % (par exemple %nom%).

L'élément XML suivant indique à l'ETL quelle entité de la destination il faut créer ou mettre à jour si celle-ci existe déjà.

```
<entry verifyIfExist="(&amp;(objectClass=ENTPerson)(uid=%uid%))">
```

L'élément XML suivant indique à l'ETL la requête LDAP à exécuter afin d'affecter une valeur à l'attribut courant.

```
<value>ldap://***.in.ac-rennes.fr/ou=personnes,dc=...?uid?one?
(&amp;(objectClass=ENTProfil)(member=uid=%uid%))</value>
```

L'élément XML suivant montre le formalisme pour solliciter une fonction de l'ETL au moment du chargement du pivot dans la cible. Ici, la fonction de génération d'un identifiant unique universel est sollicitée pour affecter une valeur à l'attribut « Jointure ».

```
<value>(UUID) (/UUID)</value>
```

4.4.3 Liste des fonctions pouvant être sollicitées au sein d'un pivot ou template :

- obtenir un objet date (heure courante) ou faire des calculs de date,
- obtenir un UUID,
- formater une chaîne de caractères sur le modèle de *String::format()*,
- normaliser une chaîne de caractère en appliquant des polices pré-définies,
- convertir des coordonnées de géolocalisation du format Lambert 93 en GPS,
- contrôler l'unicité d'un attribut LDAP,
- obtenir l'empreinte SHA-n d'une chaîne de caractères et encodage en base64,
- chiffrer une chaîne de caractères avec un algorithme symétrique (p. ex. AES 256) ou asymétrique (p. ex. RSA 1024/2048) et encodage en base64,
- obtenir un mot de passe aléatoire en précisant les éléments constituants (lettre, minuscule, majuscule, chiffre, caractères spéciaux, longueur),
- obtenir un incrément,
- obtenir un contenu (plain texte, HTML, XML...) à partir d'un template freemarker.

4.5 Packaging

Le packaging se présente sous la forme d'une archive ZIP. Le moteur de l'ETL fait l'objet d'un livrable. Chaque script métier (composé des jobs) fait l'objet d'un livrable séparé (addon du moteur) dont la gestion sur le dépôt de codes sources et le déploiement sont indépendants.

Nous utilisons l'outil de scripting *Ansible* pour le déploiement du produit (moteur et addons).

Ansible est également utilisé pour piloter l'exécution des scripts (jobs). Il nous permet de gérer simplement le déploiement et l'exécution des scripts sur les différentes plates-formes que nous exploitons (développement, pré-production & production).

4.6 Supervision

À l'exécution, le moteur de l'ETL instrumente chaque job avec un « bean JMX » permettant de recueillir les informations suivantes en temps réel :

- le nom du job exécuté (tel que défini dans le fichier XML d'entrée),
- le nom du thread d'exécution,

- la durée d'exécution,
- le pourcentage de progression,
- la description de l'opération en cours de traitement,
- le statut (en attente, en cours d'exécution, terminé),
- un « traffic light » pour signaler la présence de warning et/ou erreur au cours des traitements.

Par ailleurs, un agent *Jolokia* est embarqué dans le moteur permettant l'interrogation simplifiée de ces sondes JMX via le protocole HTTP.

En s'appuyant sur ce bridge JMX – HTTP, un tableau de bord web léger a été implémenté (HTML – JavaScript).

4.7 Utilisations actuelles

Le produit ETL est utilisé dans le cadre de nombreux projets et services numériques délivrés par l'académie de la région Bretagne, parmi lesquels :

- identification des usagers via leur compte EduConnect,
- fourniture de ressources pédagogiques numériques au sein des établissements via le projet national GAR « Gestionnaire d'accès aux ressources »,
- alimentation quotidienne de l'annuaire applicatif de l'ENT (« Espace Numérique de Travail »),
- alimentation de notre référentiel documentaire afin de permettre le déploiement de services métiers,
- production de jeux de données anonymisées afin de permettre la mise en place de démonstrateur et/ou phase de qualification,
- fourniture d'accès à l'ENT aux personnes extérieures à l'académie en faisant la demande.

4.8 Synthèse et évolutions

Le dispositif d'anonymisation a atteint les objectifs énoncés au chapitre §2 Contexte d'anonymisation, faisant de l'académie de Rennes le fournisseur du service d'anonymisation pour de nombreuses académies et partenaires.

Depuis sa création, de constantes évolutions ont été apportées au produit ETL afin d'enrichir ses fonctionnalités et augmenter ses performances (gain de rapidité d'exécution et diminution de l'empreinte mémoire). Une « APIsation » du produit a aussi été réalisée afin de pouvoir le solliciter en mode API REST HTTP en plus du mode CLI.

Adapter son architecture logicielle afin de faciliter les contributions et sa redistribution open-source est également un objectif important. Pour accentuer cet aspect, l'utilisation d'un framework faisant partie des standards de l'industrie est à l'étude. Cela permettrait d'obtenir une modularité complète sur le moteur et ses addons et ainsi faciliter les contributions de la communauté de développeurs.

Une publication sur la plate-forme GitHub est en cours.

Bibliographie

[1] G29 (groupe de travail « article 29 » - organe consultatif européen indépendant sur la protection des données et de la vie privée). Avis sur les techniques d'anonymisation, 05/2014 ; https://www.cnil.fr/sites/default/files/atoms/files/wp216_fr.pdf.

[2] RGPD – article 25 ; http://www.privacy-regulation.eu/fr/25.htm.

[3] Wikipédia - Universal Unique Identifier ; https://fr.wikipedia.org/wiki/Universal_Unique_Identifier.