Tutoriel alignement de vocabulaire : GeoNames

Auteur(s) : , Lizzie Scholtus,
Institution émettrice : Consortium MASA
contact mail du projet : olivier.marlet@univ-tours.fr

Avant-propos

Ce tutoriel a pour objectif d'expliquer comment effectuer un alignement de vocabulaire vers le référentiel de données géographiques GeoNames, à l'aide du logiciel libre OpenRefine1. Le but de cet alignement est de relier un jeu de données archéologiques contenant des lieux (généralement des sites) à leur localisation et référentiel GeoNames afin de pouvoir utiliser le websémantique et en particulier l'outil OpenArchaeo. Ce document est établi à l'aide des données provenant du fond "Jean Perrot. Du village à l'État au Proche- et Moyen-Orient" conservé à la MSH Mondes et d'un export de la base "Cruches gallo-romaines" depuis la plateforme de SIG en ligne ArkeoGIS.

Aspects techniques

Ce tutoriel est écrit en R Markdown à l'aide du package "Bookdown" créé par Yihui Xie (*bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.24., 2021).

I. Récupérer les informations nécessaires à l'alignement

Selon l'architecture du fichier de données, il est d'abord nécessaire d'isoler au sein d'une même colonne, nommée **Lieux**, les noms de lieux sur lesquels l'alignement doit être réalisé2.

En fonction de la structure de la base de données – à savoir, si elle comporte des champs systématiques ou non – et des possibles précédents nettoyages de données, il peut y avoir plusieurs formes pour décrire un même lieu. Il n'est pas forcément nécessaire de réduire ces différentes versions à un seul et unique nom puisque le référentiel GeoNames les conserve également sous la forme d'*alternate names*. Par ailleurs, conserver ces multiples termes pourra faciliter, par la suite, l'enrichissement des données sources. Il est, toutefois, utile de vérifier si ces éléments sont saisis de manière uniforme et cohérente3.

II. Aligner les termes avec les GeoNames

Pour relier les noms de lieux à leur identifiant GeoNames il existe deux possibilités qui dépendent des données sources et de la connaissance que l'on en a. Ainsi, si ces données ne concernent qu'un voire deux pays, il est possible de télécharger un export de la base de données GeoNames et de le rapprocher des données sources. Par contre, si l'emprise des données n'est pas connue ou si elle s'étend sur plus de deux pays, il peut être plus simple de moissonner directement les données GeoNames à l'aide d'une API. Ces deux méthodes sont expliquées en détail dans les lignes qui suivent.

1. Aligner les lieux avec un export de GeoNames

La méthode proposée ici s'applique pour des données situées dans un espace géographique connu et limité à un ou deux pays. Elle nécessite d'exporter un extrait de la base de données GeoNames4 et d'installer un module supplémentaire dans OpenRefine.

A. Exporter la base GeoNames

Depuis la page d'accueil du site GeoNames, il est possible d'accéder aux jeux de données GeoNames pour chaque pays en cliquant sur le lien "*Free Gazetteer Data*" dans le menu "*Download*" (figure 2.1). Toutefois, il n'est possible de télécharger ces éléments que pays par pays ; c'est pourquoi, il nous parait plus judicieux de n'utiliser cette méthode que lorsque la spatialisation des données est connue et ne dépasse pas deux pays.

Le dossier téléchargé contient un fichier *Readme* qui détail les différents champs du jeu de données, et un fichier.txt qui contient les données elles-mêmes (table 2.1). C'est ce dernier fichier que l'on ouvre dans OpenRefine.

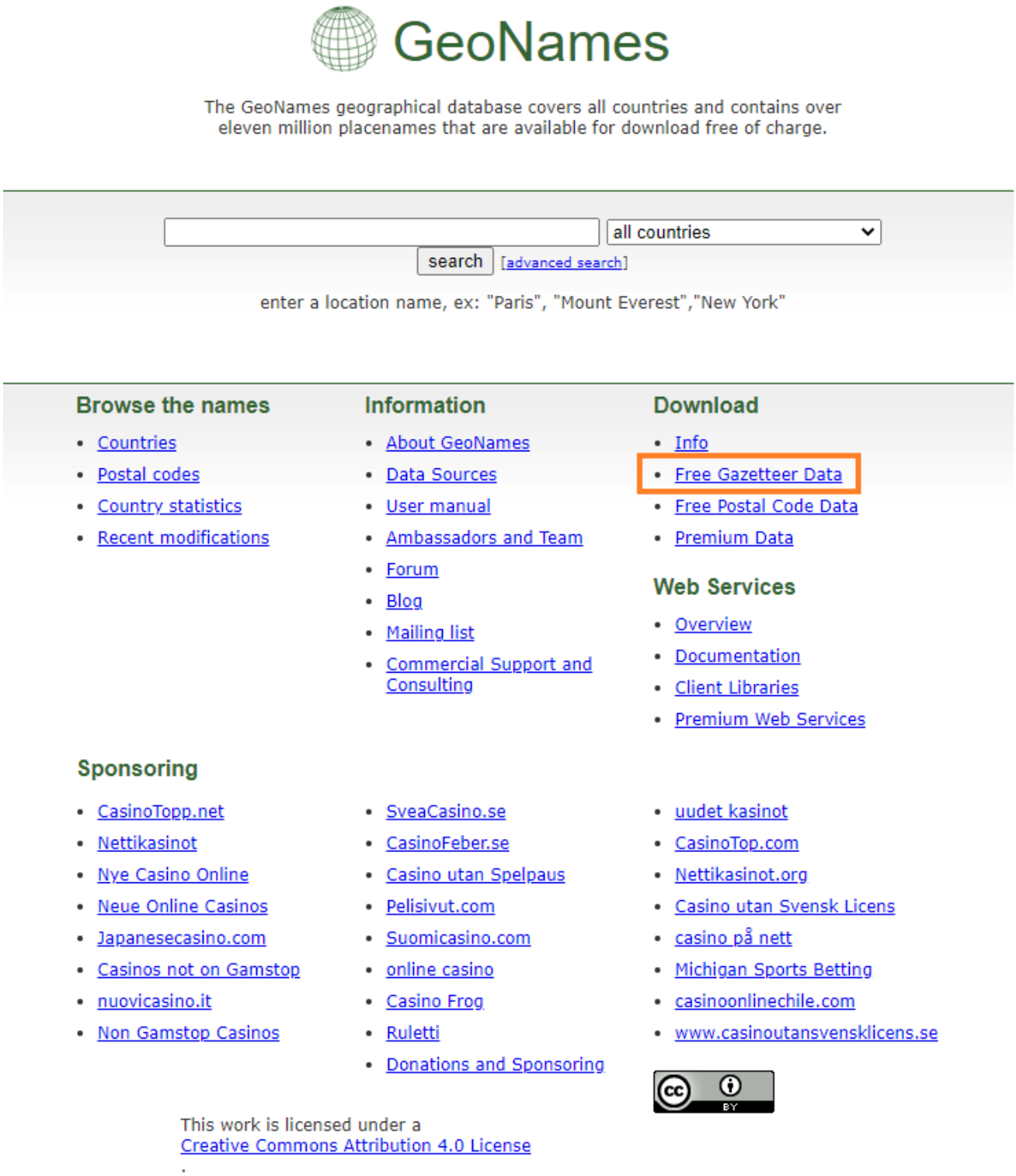

Figure 2.1: Supprimer les doublons dans la colonne ***noms_source****.*

Table 2.1: Extrait du fichier.txt GeoNames pour la France.

col 1	col 2	col 3	col 4	col 5	col 6	col 7	col 8	col 9	col 10	col 11	col 12	col 13
2659086	Col de Recon	Col de Recon	Rapenaz Col de,Recon Col de	46.30352	6.82838	T	PASS	FR	CH	84	74	744

2659815	Lucelle	Lucelle	La Lucelle Riviere,La Lucelle Rivière,Lucelle,Lutzel,Lützel	47.41667	7.50000	H	STM	FR		00		
2659933	Les Cornettes de Bise	Les Cornettes de Bise	Cornettes de Bise,Les Cornettes de Bise	46.33263	6.78458	T	PK	FR	CH	84	74	744
2659943	Lertzbach	Lertzbach	Le Lertzbach Ruisseau,Lertzbach,Ruisseau le Lertzbach	47.60479	7.54665	H	STM	FR	CH	00		
2659973	Le Cheval Blanc	Le Cheval Blanc	Le Cheval Blanc	46.05193	6.87276	T	MT	FR	CH	84	74	742
2660216	Jougnena	Jougnena	Jougnena,Jougnenaz,Jouguena,La Jougnena Ruisseau	46.71667	6.40000	H	STM	FR	CH	00		
2661811	London	London	Allondon,Allondon River,L'Allondon,La London Riviere,La London Rivière,London	46.16667	6.01667	H	STM	FR		00		
2806878	Wolfesberg	Wolfesberg	Wolfers-Berg,Wolfesberg	49.29176	6.62355	T	HLL	FR		44		
2842622	Saar River	Saar River	La Sarre Riviere,La Sarre Rivière,Saar,Saar River,Sarre	49.70000	6.56667	H	STM	FR		00		
2844740	Rosselle	Rosselle	La Rosselle Riviere,La Rosselle Rivière,La Rosselle Ruisseau,Rosel,Roselle River,Rossel,Rosselle	49.24558	6.83944	H	STM	FR	DE	00		

B. Télécharger et installer le module VIB-Bits

Le principe de cet alignement, depuis un export des données GeoNames, est de croiser les informations GeoNames et le jeu de données source à l'aide d'un élément commun – généralement le nom de lieu. Cette opération est possible sur OpenRefine en y installant une extension complémentaire qui nous permettra d'ajouter une colonne à notre jeu de données en fonction des données d'un autre projet : VIB-BITs[5].

Après avoir téléchargé (ici) l'extension VIB-BITs et avoir décompressé le fichier zip, il faut ensuite l'installer dans le dossier du programme. Il est possible d'accéder à cet espace directement depuis l'interface d'OpenRefine : à l'ouverture d'un projet choisir l'option "**Ouvrir un projet**", puis, tout en bas, l'option "**Parcourir le dossier de l'espace de travail**" (voir figure 2.2). Le dossier s'ouvre alors dans l'explorateur de fichier.

Figure 2.2: Ouvrir le dossier de l'espace de travail.

S'il n'existe pas encore, il faut ensuite créer le dossier "**extensions**" dans ce dossier (voir figure 2.3 : **A**). Le logiciel OpenRefine doit ensuite être fermé, avant de déplacer le dossier décompressé VIB-BITs dans ce nouveau dossier. Au redémarrage du logiciel, il est possible de vérifier si l'installation a bien fonctionné en regardant si le nom de l'extension apparaît bien en haut à droite de la page (voir figure 2.3 : **B**). Par ailleurs, de nouvelles fonctions sont désormais disponibles.

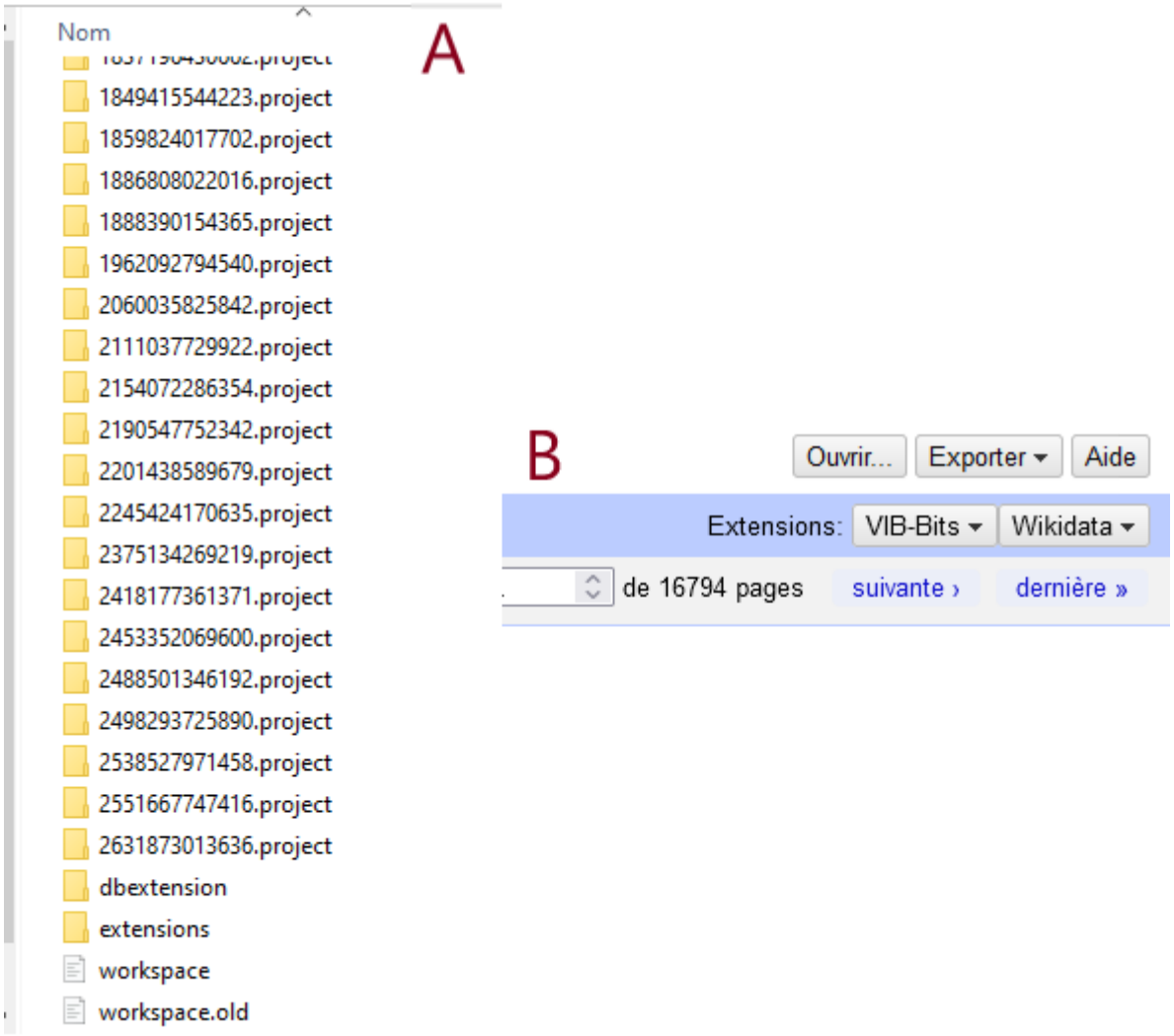

*Figure 2.3: **A** : Créer le dossier "**extensions**". **B** : Vérifier que l'extension est bien installée.*

C. Aligner les données sources

Le module VIB-BITs installé, il est désormais possible d'ajouter une colonne dans notre projet d'après les données d'un autre projet en les croisant en fonction d'une colonne commune. Pour rappel, l'objectif est ici d'importer dans un jeu de données source les informations GeoNames relatives aux lieux qu'il contient. Toutefois, pour que cette opération fonctionne, il est nécessaire que les colonnes qui serviront de clé de correspondance est le même nom dans les deux projets. Dans cet exemple, nous nommerons cette colonne **Lieu** dans les deux projets comparés, c'est-à-dire la colonne " **MAIN_CITY_NAME**" dans la base Cruche conservée sur le SIG ArkeoGIS (table 2.2) et la colonne 2 de l'export précédent pour les données GeoNames concernant la France (table 2.1).

Table 2.2: Extrait de la base Cruche exporté depuis ArkeoGIS.

SITE_AKG_ID	SITE_SOURCE_ID	SITE_NAME	MAIN_CITY_NAME	STATE_OF_KNOWLEDGE	OCCUPA
536199	A, 01, 01	Kembs, lotissement des Bateliers II	Kembs	Non renseigné	Non rense
536200	A, 01, 02	Brumath, 13 rue du Château	Brumath	Non renseigné	Non rense
536201	A, 01, 03	Strasbourg, place de l'Homme-de-Fer	Strasbourg	Non renseigné	Non rense

536202	A, 01, 04	Strasbourg, place de l'Homme-de-Fer	Strasbourg	Non renseigné	Non rense
536203	B, 01, 01	Rosheim, Baruch-Laser	Rosheim	Non renseigné	Non rense
536204	B, 01, 02	Kembs, lotissement des Bateliers II	Kembs	Non renseigné	Non rense
536205	B, 01, 03	Strasbourg, place de l'Homme-de-Fer	Strasbourg	Non renseigné	Non rense
536206	B, 02, 01	Brumath, lotissement du Tilleul	Brumath	Non renseigné	Non rense
536207	B, 02, 02	Brumath, lotissement du Tilleul	Brumath	Non renseigné	Non rense
536208	B, 03, 01	Koenigshoffen, rue Mentelin	Koenigshoffen	Non renseigné	Non rense
536209	B, 04, 01	Saint Quirin, Croix-Guillaume	Saint Quirin	Non renseigné	Non rense
536210	B, 04, 02	Reichshoffen, C.E.S., 14 rue du Cerf	Reichshoffen	Non renseigné	Non rense

Il est alors possible de créer – dans le projet contenant les données sources – une ou plusieurs nouvelles colonnes d'après des colonnes contenues dans l'export GeoNames, en sélectionnant la fonction "***Add column(s) from other projects***"[6] dans l'onglet "**Éditer la colonne**" (figure 2.4).

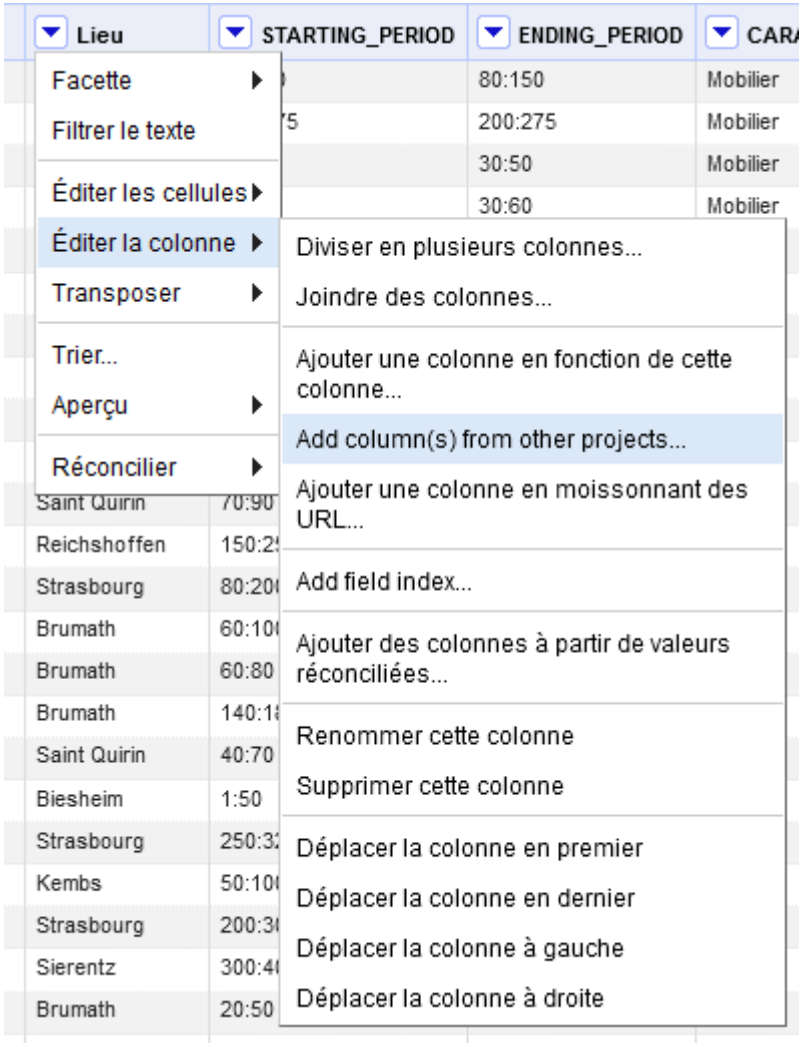

Figure 2.4: Créer une nouvelle colonne dans le projet source d'après les données GeoNames.

Cette opération ouvre une nouvelle fenêtre qui permet dans un premier temps (figure 2.5 : **1**) de définir le projet OpenRefine dans lequel se trouvent les informations à ajouter7. Dans un deuxième temps, il est demandé de préciser la colonne du second projet qui correspond aux données du projet actuel. Si ces colonnes ont bien été nommées de la même manière, la colonne de correspondance (ici **Lieu**, qui comprend les noms de communes dans la base source et dans l'export GeoNames) est automatiquement cochée8. Enfin, la troisième et dernière phase permet de sélectionner les colonnes à importer dans le projet en cours. Ici, nous sélectionnons celles indiquant les l'identifiant unique GeoNames et les coordonnées des lieux (voir table 2.1).

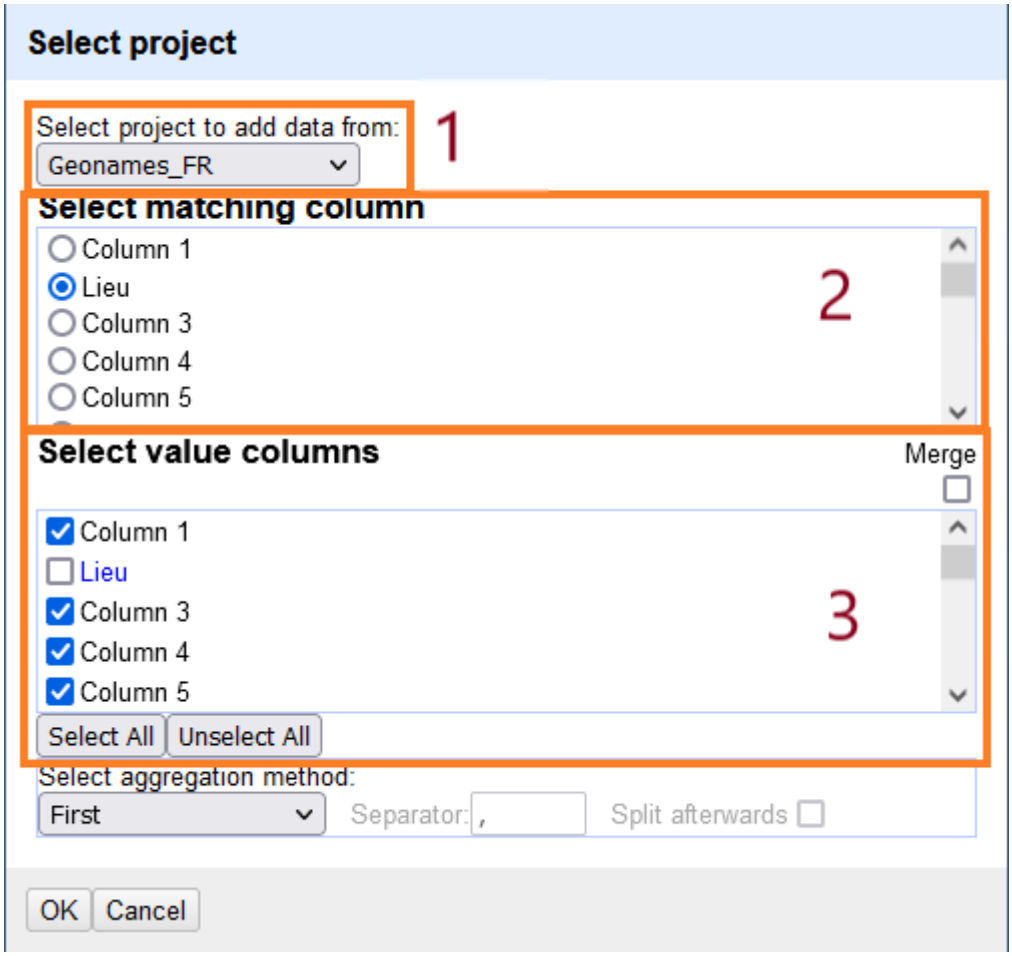

Figure 2.5: Sélectionner les colonnes de croisement de données et celles à importer.

Lorsque cette sélection est validée, les colonnes déterminées dans le fichier GeoNames sont directement importées dans le jeu de données source en fonction des termes contenus dans la colonne lieu. De plus, ces informations sont également répétées lorsqu'un même terme apparaît plusieurs fois dans les données sources, c'est le cas ici, par exemple, pour les communes de Strasbourg ou de Brumath (figure 2.6).

SITE_NAME	Lieu	Column 1	Column 5	Column 6	STARTING_PE
Kembs, lotissement des Bateliers II	Kembs	3011964	47.68948	7.50323	80:150
Brumath, 13 rue du Château	Brumath	3029771	48.73398	7.71095	200:275
Strasbourg, place de l'Homme-de-Fer	Strasbourg	2973783	48.58392	7.74553	30:50
Strasbourg, place de l'Homme-de-Fer	Strasbourg	2973783	48.58392	7.74553	30:60
Rosheim, Baruch-Laser	Rosheim	2982790	48.4971	7.47105	300:350
Kembs, lotissement des Bateliers II	Kembs	3011964	47.68948	7.50323	70:100
Strasbourg, place de l'Homme-de-Fer	Strasbourg	2973783	48.58392	7.74553	20:100
Brumath, lotissement du Tilleul	Brumath	3029771	48.73398	7.71095	60:80
Brumath, lotissement du Tilleul	Brumath	3029771	48.73398	7.71095	60:100
Koenigshoffen, rue Mentelin	Koenigshoffen	3011438	48.55855	7.68382	140:200
Saint Quirin, Croix-Guillaume	Saint Quirin				70:90
Reichshoffen, C.E.S., 14 rue du Cerf	Reichshoffen	2984136	48.93287	7.66561	150:250

Figure 2.6: Résultat du croisement de données et de l'import de colonnes.

D. Compléter les données manquantes

Comme pour toute complétion automatique de données, il est nécessaire de vérifier le résultat obtenu et il peut exister des lignes pour lesquelles le croisement de données n'a pas pu fonctionner. Plusieurs raisons peuvent expliquer ce phénomène : terme absent dans le référentiel, orthographe différente, faute(s) de frappe, différences dans la casse, etc. Dans l'exemple développé ici, aucune correspondance n'a été mise en évidence pour le site de Saint Quirin (voir figure 2.6). La commune existe pourtant dans les données GeoNames (table 2.3), mais y est orthographiée avec un trait d'union.

Table 2.3 : Recherche du terme "Quirin" dans les données exportées depuis GeoNames.

	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X
10146	2977254	Vallée de Saint-Quirin	Vallee de Saint-Quirin		48.58409	7.13206	T	VAL	FR		44	57		
10147	2977255	Saint-Quirin	Saint-Quirin	Saint-Quirin,Sen-Kiren,sheng qu yi lan,sn-kyrn,syn-qaryn,	48.61018	7.06569	P	PPL	FR		44	57	575	57
86886	6437711	Métairies-Saint-Quirin	Metairies-Saint-Quirin	57461,Metairies-Saint-Quirin,Meteri-Sen-Kiren,Métairies-Saint-Quirin,mei tai li ye sheng qu yi lan,mtry-sn-kyrn	48.65310	7.04240	A	ADM4	FR		44	57	575	57
102140	6454372	Saint-Quirin	Saint-Quirin	57623,Saint-Quirin	48.61028	7.06472	A	ADM4	FR		44	57	575	57
144191	11897789	Commune de Métairies-Saint-Quirin	Commune de Metairies-Saint-Quirin	Commune de Metairies-Saint-Quirin,Commune de Métairies-Saint-Quirin,Metairies-Saint-Quirin,Métairies-Saint-Quirin	48.64129	7.03802	A	ADMD	FR		44			
144200	11897798	Mairie de Métairies-Saint-Quirin	Mairie de Metairies-Saint-Quirin	Mairie de Metairies-Saint-Quirin,Mairie de Métairies-Saint-Quirin,Metairies-Saint-Quirin,Métairies-Saint-Quirin	48.65282	7.04186	S	ADMF	FR		44			
144207	11897910	Forêt Domaniale de Saint-Quirin	Foret Domaniale de Saint-Quirin		48.57086	7.11196	V	FRST	FR		44			

Il est donc possible de compléter les informations manquantes, comme nous venons de le faire, en recherchant des parties de termes dans le fichier exporté et ensuite reporter les informations correspondantes dans le projet source9. Il est aussi

possible de rechercher ces éléments directement sur le site geonames.org (figure 2.7) et d'y récupérer les mêmes informations, avec la possibilité toutefois de vérifier la localisation réelle du lieu (figure 2.8).

Figure 2.7: Rechercher les lieux manquant sur le site geonames.org.

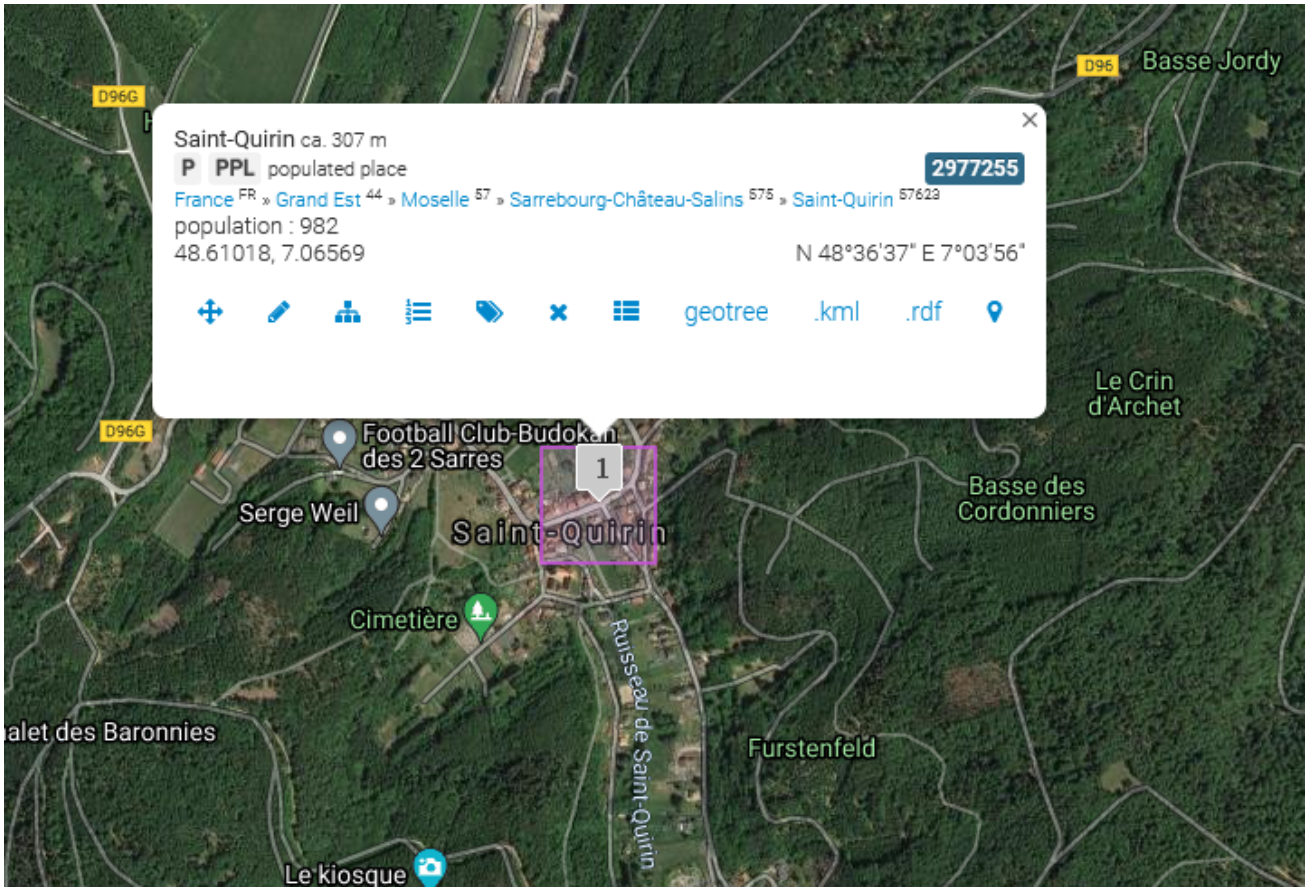

Figure 2.8: Dans GeoNames, les données s'accompagnent d'une carte de localisation.

2. Aligner les lieux en moissonnant GeoNames

La seconde méthode pour récupérer les identifiants et autres informations GeoNames repose sur le moissonnage des données hébergées sur le site à l'aide d'OpenRefine en utilisant le service d'API10, et plus particulièrement le webservice dédié à la recherche de GeoNames. Bien qu'un peu plus contraignante que la méthodologie présentée dans les lignes précédentes, cette technique à l'avantage de permettre d'aligner sur le référentiel GeoNames de grandes quantités de données dont l'étendue géographique est variée et/ou inconnue.

E. Créer un compte GeoNames

L'utilisation de l'API GeoNames ne fonctionne que pour un utilisateur connecté. Il est donc nécessaire de créer (gratuitement) un compte utilisateur (ici) en suivant les différentes étapes sur le site. Attention, ce service est limité à 20000 crédits11 par jour pour un même utilisateur. Pour des alignements plus volumineux il existe une version payante premium, mais il est aussi possible d'utiliser la première méthode présentée (partie 2.1).

Après avoir reçu le mail de confirmation de création du compte GeoNames12, il est encore nécessaire d'activer le webservice sur son compte utilisateur.

F. Préparer les données sources

Avant de pouvoir rechercher l'existence des données sources sur le site GeoNames, elles nécessitent encore quelques modifications. Même si les lieux à rechercher ont été isolés dans une colonne unique et nettoyés (voir partie 1), il faut encore en modifier l'encodage. En effet, la recherche et la moisson de données GeoNames se fait à travers la reconstitution de leurs URLs, mais ces dernières n'acceptent ni les caractères spéciaux, ni les espaces qui doivent être remplacer par leur équivalent en Encodage-pourcent.

Le logiciel OpenRefine permet d'effectuer cette opération à l'aide d'une fonction GREL13 en créant une nouvelle colonne **Lieux2** d'après les données contenues dans la colonne **Lieux** (figure 2.9). Il est alors possible de modifier le contenu de cette colonne en indiquant une expression GREL dans la fenêtre qui s'ouvre (figure 2.10). Toutefois, la fonction `escape` utilisée ici modifie les espaces par le symbole `+`. Il est alors nécessaire de le remplacer par `%20` dans un second temps.

Expression GREL :

```
escape(value, "url").replace("+", "%20")
```

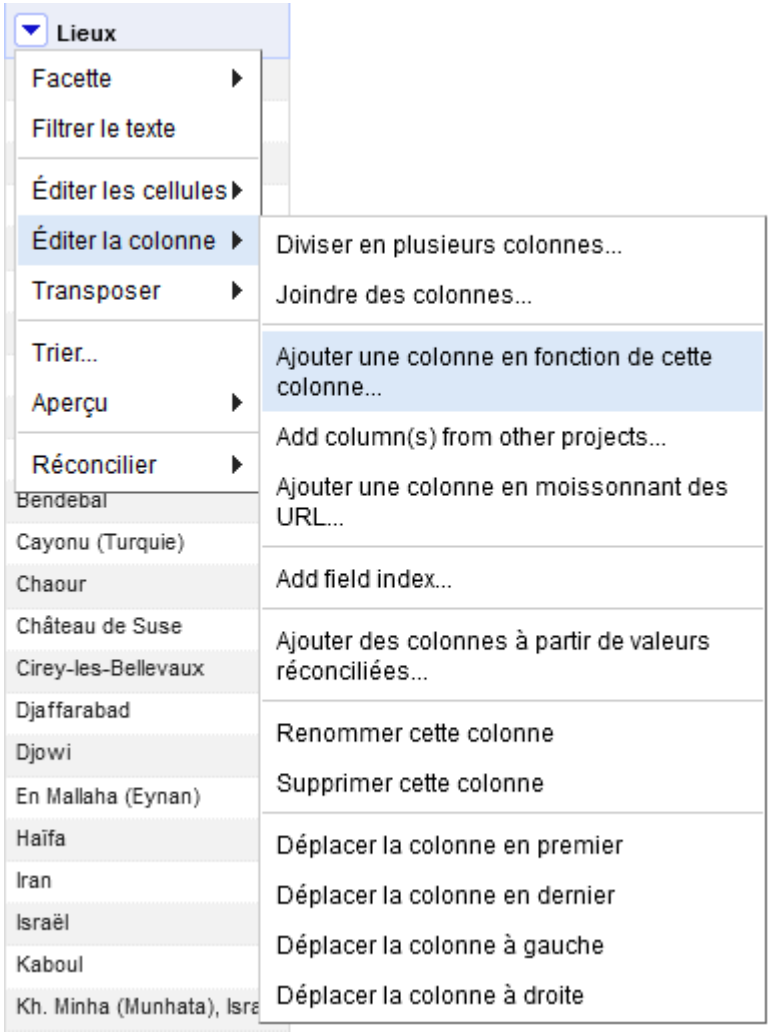

Figure 2.9: Dupliquer une colonne dans OpenRefine.

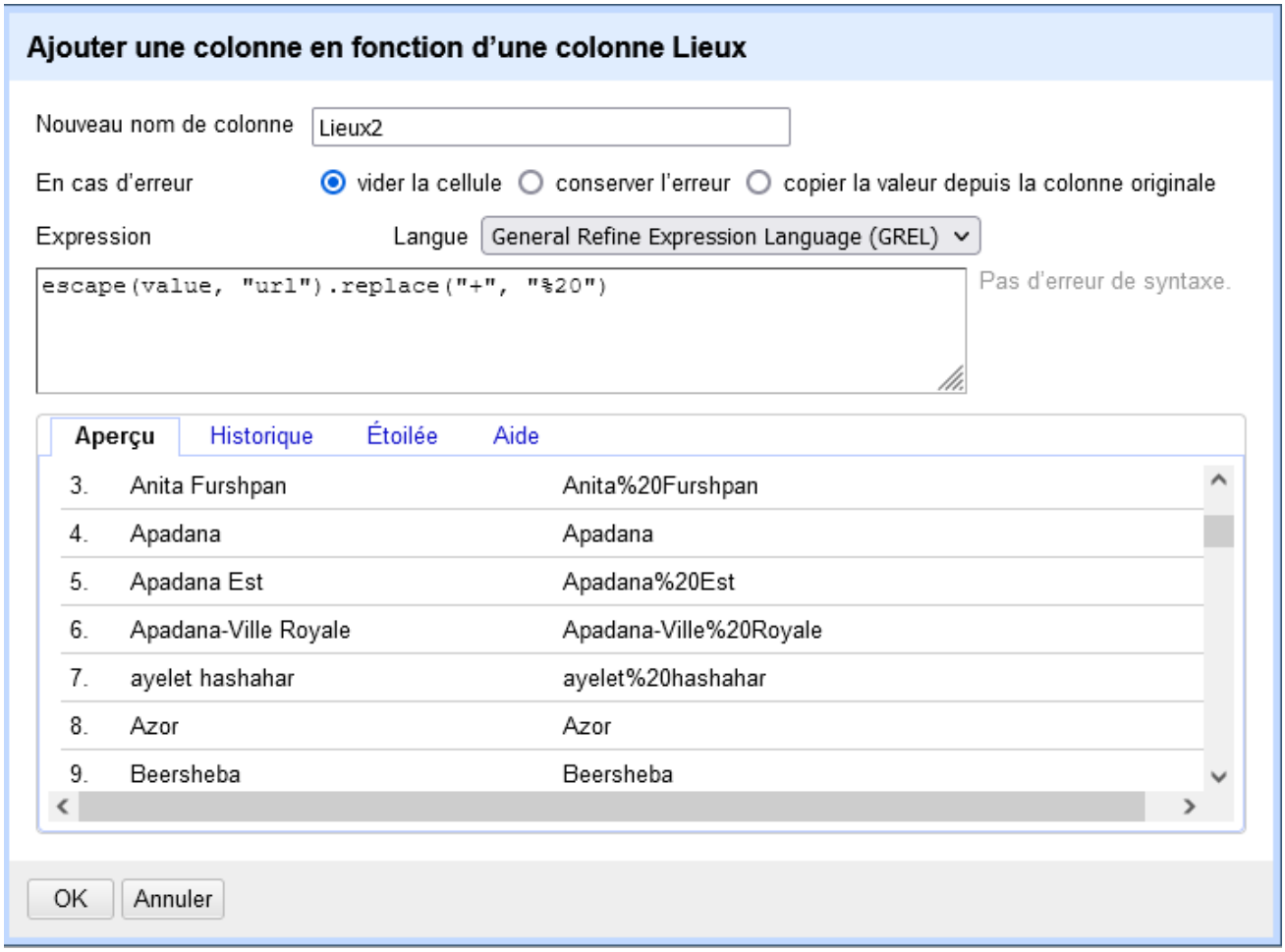

Figure 2.10: Modifier le contenu d'une colonne dupliquée.

G. Moissonner les GeoNames

Les termes à rechercher sont désormais prêts, il ne reste qu'à les insérer au sein de l'URL de requête puis de moissonner le résultat. Cette opération peut être réaliser intégralement en utilisant la fonction "**Ajouter une colonne en moissonnant des URL**" dans OpenRefine (figure 2.11). Comme pour la duplication de colonne précédente, cette fonction ouvre une nouvelle fenêtre au sein de laquelle il est possible de modifier le contenu de chaque ligne à l'aide d'une expression GREL (figure 2.12) – et donc de recomposer l'URL voulue. Ensuite, l'execution de cette commande lancera le moissonnage du site et le résultat de la requête est stocké dans une nouvelle colonne que l'on aura nommée **GeoNames**.

Il est à noter que dans l'expression GREL disponible ci-dessous, il faut remplacer la mention `NOM_UTILISATEUR` par le nom indiqué lors de la création du compte utilisateur (partie 2.2.1).

Expression GREL :

```
"http://api.geonames.org/search?q="+ value +
"&maxRows=10&type=rdf&username=NOM_UTILISATEUR"
```

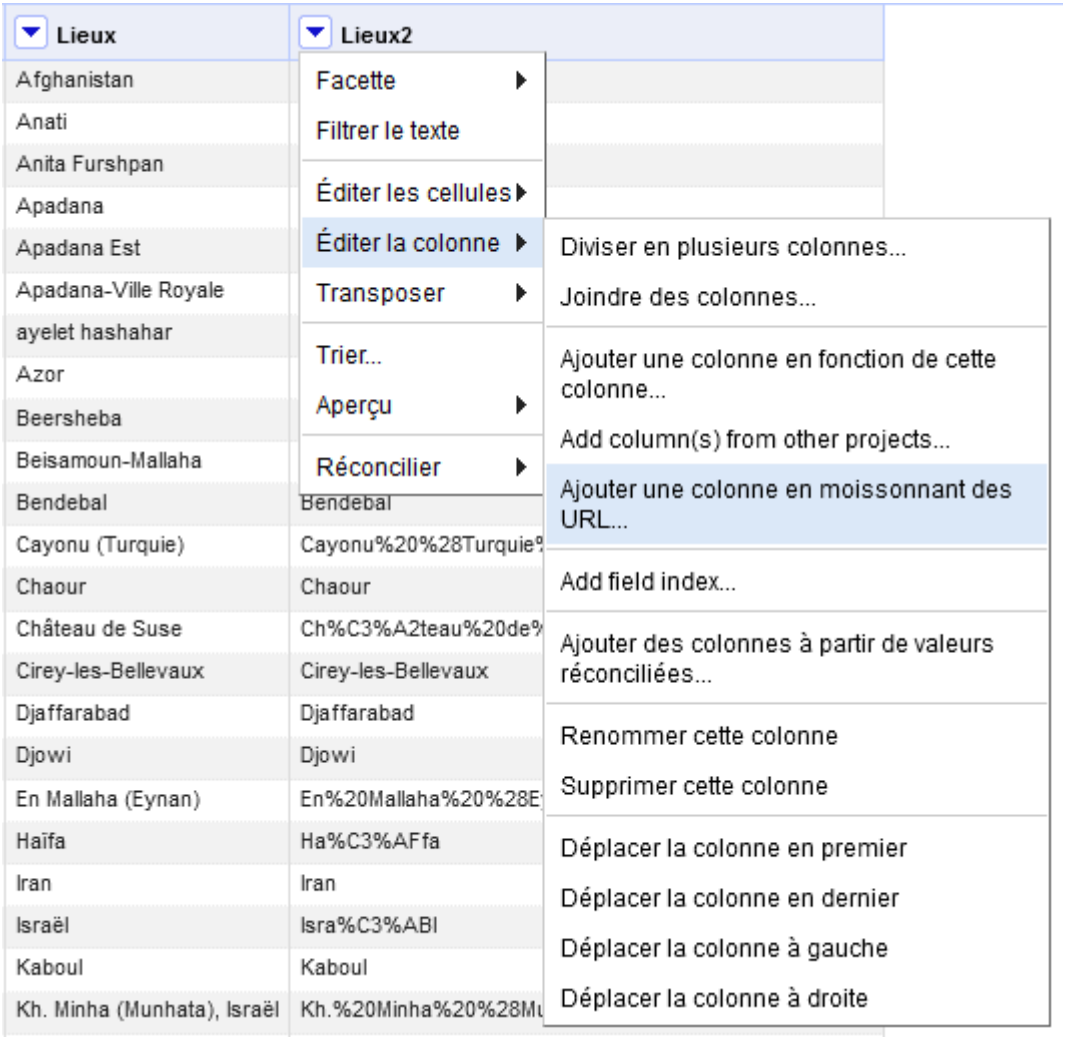

*Figure 2.11: Utiliser la fonction "**Ajouter une colonne en moissonnant des URL**".*

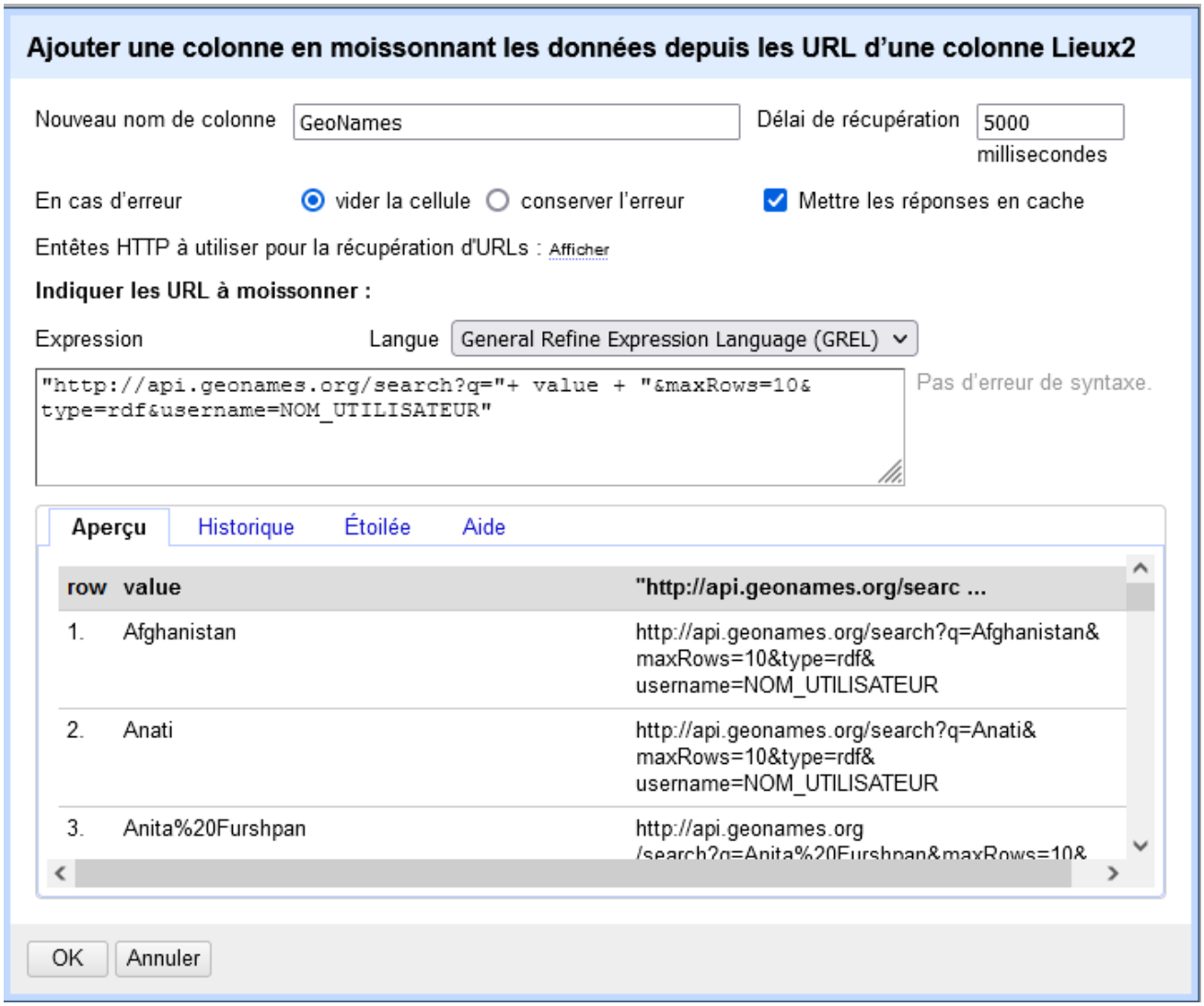

Figure 2.12: Reconstituer l'URL et moissonner.

H. Nettoyer les données obtenues

Ce moissonnage récolte l'ensemble des données contenues dans GeoNames pour tous les homonymes d'un terme et les stocke au sein d'une même cellule (figure 2.13). Il est donc nécessaire de séparer les différents résultats pour un même terme, d'effectuer un nettoyage des cellules, mais aussi une sélection des données pertinentes au projet. Dans le cadre qui nous intéresse, parce qu'il s'agit d'aligner des termes et aussi de vérifier si ce lien est correcte, les éléments retenus sont l'URI GeoNames, le nom et le pays que l'on extrait en dupliquant la colonne **GeoNames** et en en transformant les valeurs à l'aide des expressions GREL qui suivent.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<rdf:RDF xmlns:cc="http://creativecommons.org/ns#" xmlns:dcterms="http://purl.org/dc/terms
  <gn:Feature rdf:about="https://sws.geonames.org/1149361/">
    <rdfs:isDefinedBy rdf:resource="https://sws.geonames.org/1149361/about.rdf"/>
    <gn:name>Afghanistan</gn:name>
    <gn:officialName xml:lang="af">Afganistan</gn:officialName>
    <gn:officialName xml:lang="ak">Afganistan</gn:officialName>
    <gn:officialName xml:lang="am">አፍጋኒስታን</gn:officialName>
    <gn:alternateName xml:lang="an">Afganistán</gn:alternateName>
    <gn:officialName xml:lang="ar">أفغانستان</gn:officialName>
    <gn:alternateName xml:lang="arz">افغانستان</gn:alternateName>
    <gn:officialName xml:lang="as">আফগানিস্তান</gn:officialName>
    <gn:alternateName xml:lang="ast">Afghanistán</gn:alternateName>
    <gn:officialName xml:lang="az">Əfqanistan</gn:officialName>
    <gn:alternateName xml:lang="bcl">Apganistan</gn:officialName>
    <gn:officialName xml:lang="be">Афганістан</gn:officialName>
    <gn:officialName xml:lang="bg">Афганистан</gn:officialName>
    <gn:officialName xml:lang="bm">Afiganistan</gn:officialName>
    <gn:officialName xml:lang="bn">আফগানিস্তান</gn:officialName>
    <gn:officialName xml:lang="bo">অফগানিস্তান</gn:officialName>
    <gn:alternateName xml:lang="bpy">আফগানিস্তান</gn:alternateName>
    <gn:officialName xml:lang="br">Afghanistan</gn:officialName>
    <gn:officialName xml:lang="bs">Afganistan</gn:officialName>
    <gn:officialName xml:lang="ca">Afganistan</gn:officialName>
    <gn:officialName xml:lang="ce">Овхlан мохк</gn:officialName>
    <gn:alternateName xml:lang="ckb">ئەفغانستان</gn:alternateName>
    <gn:alternateName xml:lang="crh">Afğanistan</gn:alternateName>
    <gn:officialName xml:lang="cs">Afghánistán</gn:officialName>
    <gn:alternateName xml:lang="csb">Afganistón</gn:alternateName>
    <gn:alternateName xml:lang="cv">Афганистан</gn:alternateName>
    <gn:officialName xml:lang="cy">Afghanistan</gn:officialName>
    <gn:officialName xml:lang="da">Afghanistan</gn:officialName>
    <gn:officialName xml:lang="de">Afghanistan</gn:officialName>
    <gn:alternateName xml:lang="diq">Efğanıstan</gn:alternateName>
    <gn:alternateName xml:lang="dv">ﷺ</gn:alternateName>
    <gn:officialName xml:lang="dz">[illegible]</gn:officialName>
    <gn:officialName xml:lang="ee">Afghanistan nutome</gn:officialName>
    <gn:officialName xml:lang="el">Αφγανιστάν</gn:officialName>
    <gn:officialName xml:lang="en">Afghanistan</gn:officialName>
    <gn:officialName xml:lang="eo">Afganujo</gn:officialName>
    <gn:officialName xml:lang="es">Afganistán</gn:officialName>
    <gn:officialName xml:lang="et">Afganistan</gn:officialName>
    <gn:officialName xml:lang="eu">Afganistan</gn:officialName>
    <gn:alternateName xml:lang="ext">Afganistán</gn:alternateName>
    <gn:officialName xml:lang="fa">افغانستان</gn:officialName>
    <gn:officialName xml:lang="ff">Afganistaan</gn:officialName>
    <gn:officialName xml:lang="fi">Afganistan</gn:officialName>
    <gn:officialName xml:lang="fo">Afganistan</gn:officialName>
    <gn:officialName xml:lang="fr">Afghanistan</gn:officialName>
    <gn:alternateName xml:lang="frp">Afganistan</gn:alternateName>
    <gn:alternateName xml:lang="fur">Afghanistan</gn:alternateName>
    <gn:officialName xml:lang="fy">Afghanistan</gn:officialName>
```


Figure 2.13: Extrait du résultat du moissonnage d'un terme.

- Diviser les différentes possibilités d'alignements.

OpenRefine propose une fonction permettant de diviser le contenu des cellules d'une même colonne en fonction d'un symbole (ou d'une suite de caractères) servant d'élément séparateur. Il nous faut donc ajouter ce symbole – ici `|` – entre chaque résultat possible en transformant les cellules (figure 2.14) et en y appliquant une expression GREL de remplacement.

Expression GREL :

```
value.replace(/(<\/gn:Feature>)(.*)/, "$1|$2")
```

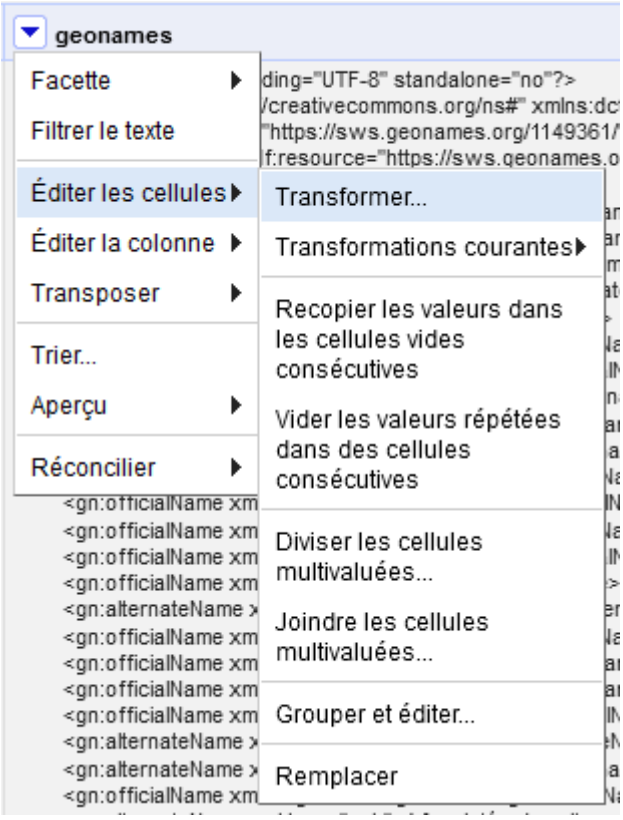

Figure 2.14: Effectuer une transformation sur l'ensemble des cellules d'une même colonne.

On peut ensuite utiliser la fonction "**Diviser les cellules multivaluées**" (figure 2.15) pour séparer ces différents résultats. Il faut encore supprimer les espaces qui ont pu être ajouter par cette opération à l'aide des transformations préenregistrées dans OpenRefine (figure 2.16).

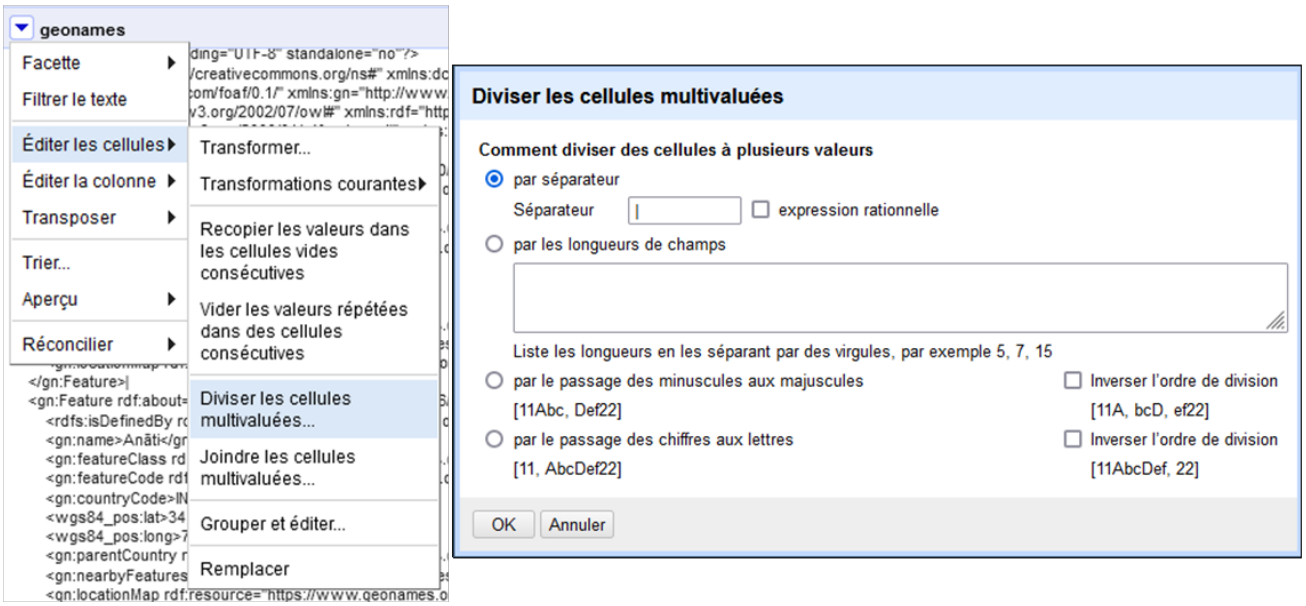

Figure 2.15: Diviser les cellules en fonction d'un séparateur.

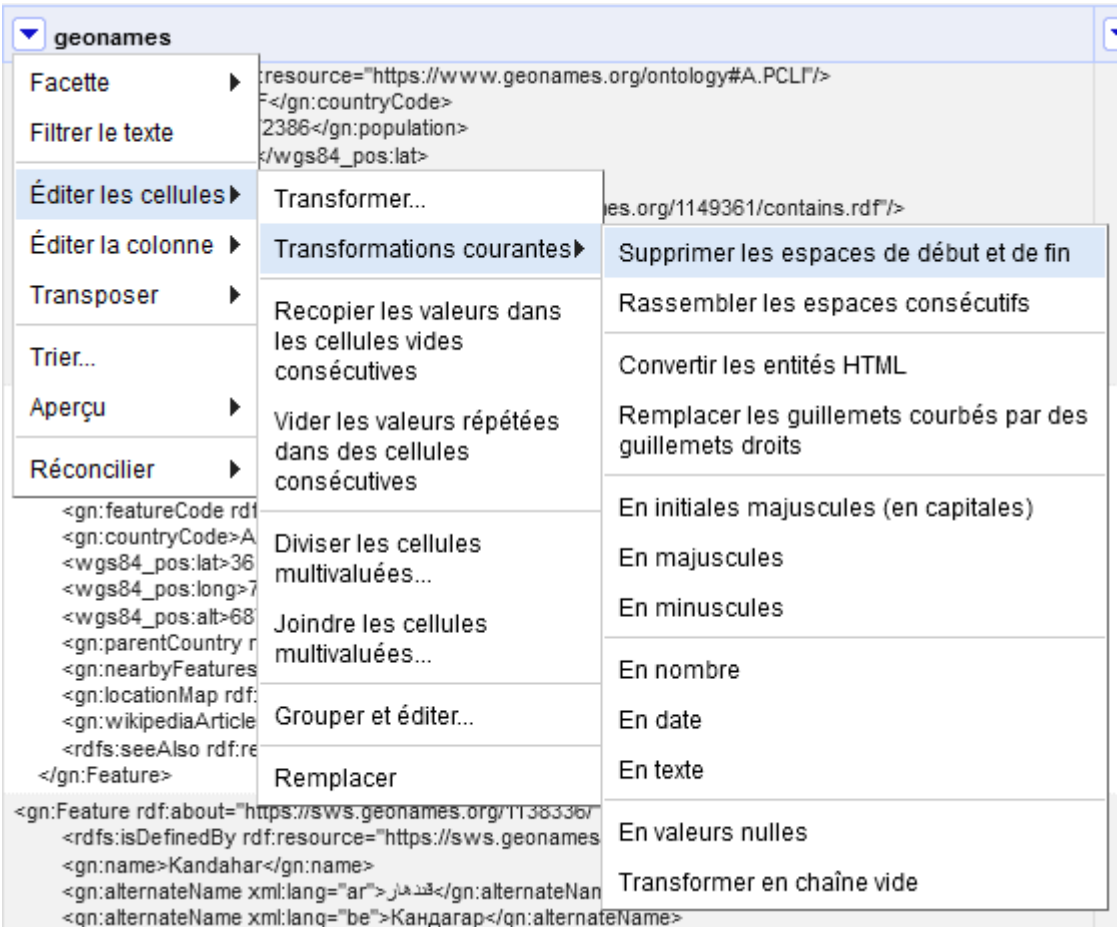

Figure 2.16: Supprimer les espaces en début et fin de cellule.

- Récupérer l'URI GeoNames.

Expression GREL :

```
value.parseXml().select('gn|Feature')[0].xmlAttr('rdf:about')
```

- Récupérer le nom.

Expression GREL :

```
value.parseXml().select('gn|name')[0].ownText()
```

- Récupérer le nom en français s'il existe dans les données.

Expression GREL :

```
value.parseXml().select('gn|alternateName[xml:lang="fr"]')[0].ownText()
```

- Récupérer le pays[14].

Le nom du pays parent n'est pas spécifié directement dans les données moissonnées, on y trouve uniquement le code et l'URI du-dit pays. Il est donc d'abord nécessaire de récupérer cette URI dans une colonne **URI_pays**, puis de se servir de ces données pour reconstituer l'URL à moissonner en utilisant la fonction "**Ajouter une colonne en moissonnant des URL**". Enfin, on peut appliquer sur ce résultat une expression GREL pour en extraire le nom d'après GeoNames ou une autre expression pour directement récupérer ce nom en français.

Expression GREL : Récupérer l'URI du pays

```
value.parseXml().select('gn|parentCountry')[0].xmlAttr('rdf:resource')
```

Expression GREL : Reconstituer l'URL à moissonner[15]

```
"http://api.geonames.org/get?geonameId="  +  value.find(/\d+/)[0]  +
"&username=NOM_UTILISATEUR"
```

Expression GREL : Récupérer le nom

```
value.parseXml().select('name')[0].ownText()
```

Expression GREL : Récupérer le nom en français

```
value.parseXml().select('alternateName[lang="fr"]')[0].ownText()
```

I. Comparer les termes ressources aux termes GeoNames

Puisque le moissonnage de données GeoNames propose plusieurs alignements possibles avec un même terme source, il nous faut encore comparer ces deux informations pour ne conserver que les résultats qui correspondent effectivement à nos termes sources. Bien que la présence de multiples possibilités résulte de l'existence d'homonymes, ces derniers sont généralement écrits différemment. De plus, lors de l'alignement de pays, le moissonnage de données GeoNames propose aussi un alignement avec les grandes villes du-dit pays. Il nous faut donc supprimer ces éléments, ou tout du moins les signaler dans le jeu de données.

Cette opération s'effectue en plusieurs étapes :

- Les différentes possibilités pour un même lieu sont consignées dans une même entrée16. Pour pouvoir comparer toutes les lignes, il convient d'abord de dupliquer la colonne **Lieux**, puis de recopier, au sein cette nouvelle colonne **Lieux2**, les valeurs des cellules précédentes dans les cellules vides en utilisant l'option "**Recopier les valeurs dans les cellules vides consécutives**" (figure 2.17).

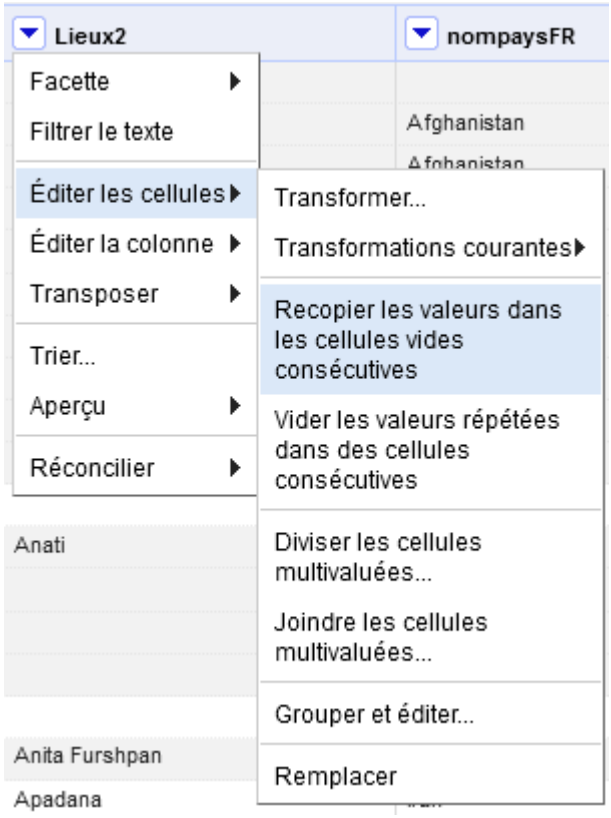

Figure 2.17: Recopier les valeurs dans les cellules vides consécutives.

- Il est ensuite possible de comparer les lieux entre eux, ou plus précisément de comparer leur orthographe, et de consigner le résultat au sein d'une nouvelle colonne appelée **comparaison**. Si le nom GeoNames et le terme source correspondent exactement, ils seront identiques. De même si le terme source correspond au nom GeoNames en français. Si aucun alignement n'a pu être établi, la mention "absent" sera indiquée dans la comparaison. Dans les autres cas, les éléments seront considérés comme différents.

Expression GREL :[17]

```
if(cells["Lieux2"].value == cells["geonames_Nom"].value, "Identique",
if(cells["Lieux2"].value == cells["geoname_nom_FR"].value, "Identique",
if(cells["geonames_Nom"].value == Null, "Absent",
"Différent")))
```

- Il est aussi utile de signaler dans cette colonne **comparaison** les alignements identifié comme différents – c'est-à-dire dont le terme source ne correspond ni au nom GeoNames, ni au nom français dans GeoNames – et pour lesquels le terme n'existe pas en français dans GeoNames. en effet, cette absence pourrait expliquer l'impossibilité d'aligner automatiquement le lieu. Pour ce faire, il est nécessaire d'effectuer une facette sur la colonne **comparaison** (figure 2.18), permettant de séparer les lignes du jeu de données en fonction de l'information saisie dans cette colonne. Il est alors possible de sélectionner toutes les lignes pour lesquelles la comparaison indique "Différent"[18] et d'y appliquer une nouvelle transformation à l'aide d'une expression GREL modifiant l'information "Différent" par "Non traduit" lorsque la cellule correspondante dans la colonne **geoname_nom_FR** est vide.

Expression GREL :

```
if(cells["geoname_nom_FR"].value == Null, "Non traduit", value)
```

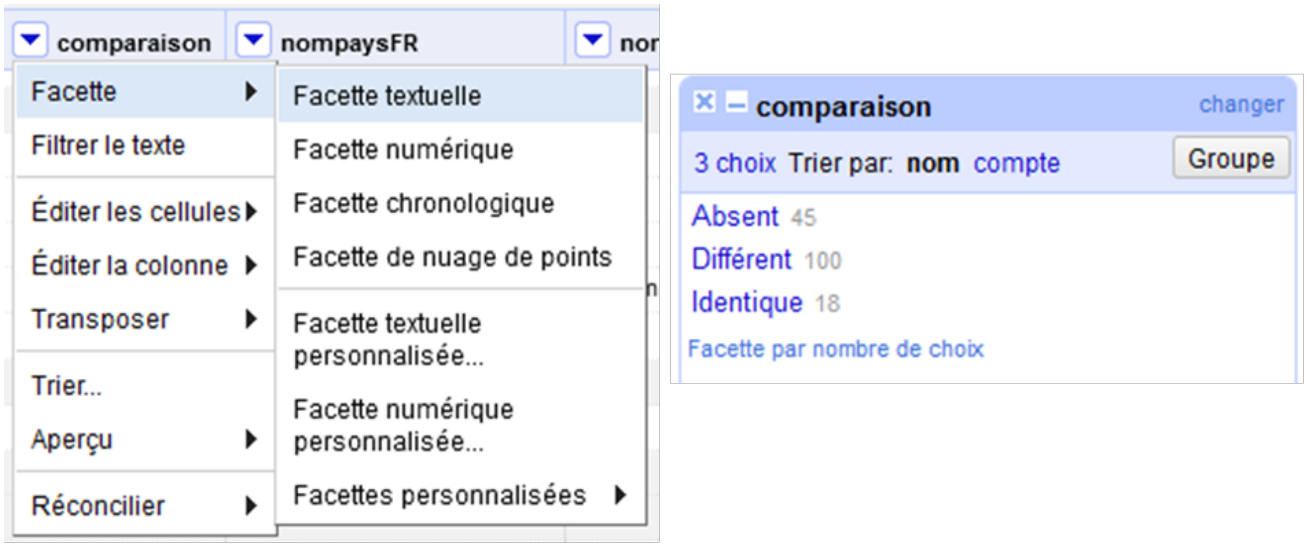

*Figure 2.18: Créer une facette sur la colonne **comparaison** pour séparer les lignes en fonction de ses données.*

- Afin de préserver l'organisation des données, et toujours parce que les différentes propositions sont enregistrées dans une même entrée pour un même terme, il faut encore distinguer dans la colonne **comparaison** les propositions considérées comme différentes lorsqu'un lieu est effectivement saisi dans les données sources en lui attribuant la mention "Non conforme". L'objectif de cette manipulation est d'éviter de supprimer du jeu de données un terme source parce qu'il n'aurait pas eu de correspondance dans les GeoNames. Il s'agit cette fois de créer une facette sur la colonne **Lieux** pour distinguer les cellules vides des cellules non vides (figure 2.19). Ces dernières sont ensuites sélectionnées et une transformation est appliquée sur la colonne **comparaison** permettant d'indiquer la mention "Non conforme" lorsque l'alignement est considéré comme différent et qu'un terme est effectivement inscrit dans la colonne **Lieux**.

Expression GREL :

```
if(value == "Différent", "Non conforme", value)
```

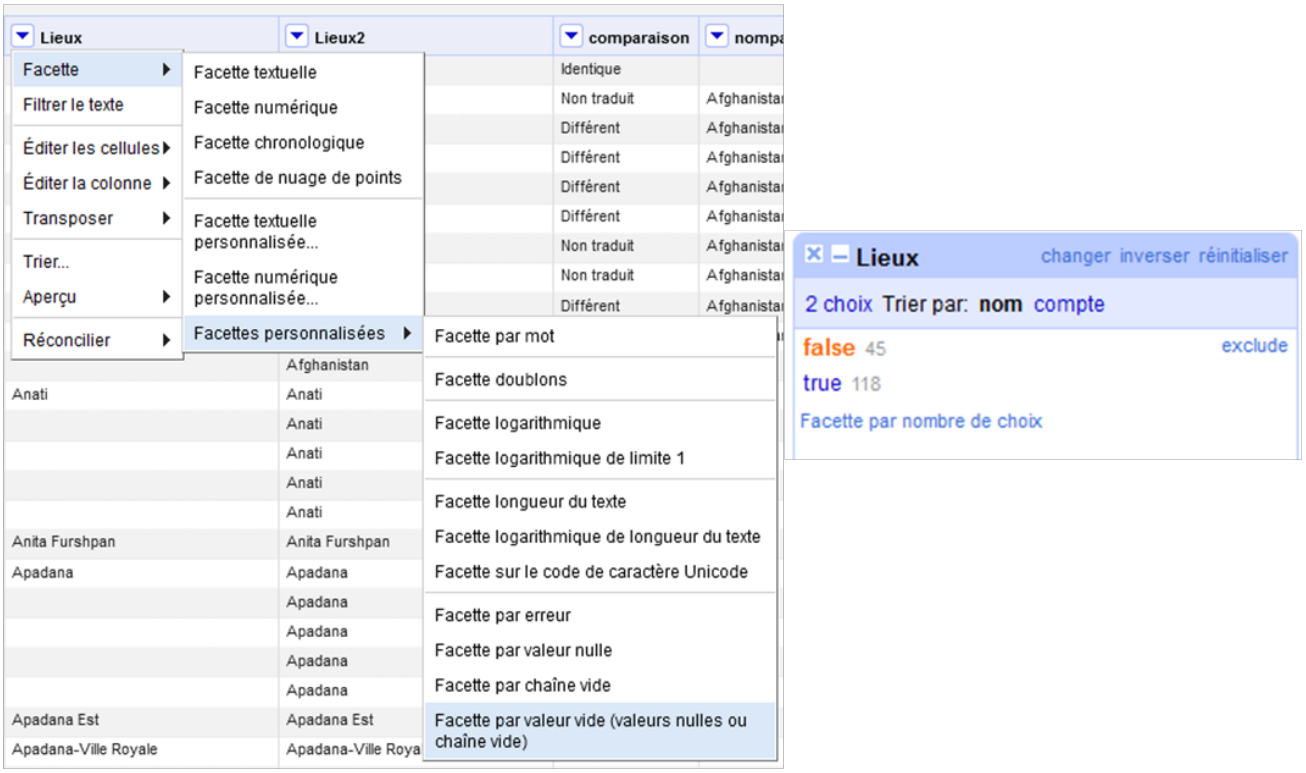

*Figure 2.19: Créer une facette sur la colonne **Lieux** pour séparer les cellules vides des cellules non vides.*

- Il est finalement possible de faire une nouvelle facette textuelle sur la colonne **comparaison** pour supprimer toutes les lignes restantes pour lesquelles l'alignement est encore "Différent". Cette opération peut s'effectuer dans l'onglet "**Toutes**" puis "**Éditer les lignes**" et "**Supprimer les lignes correspondantes**" (figure 2.20).

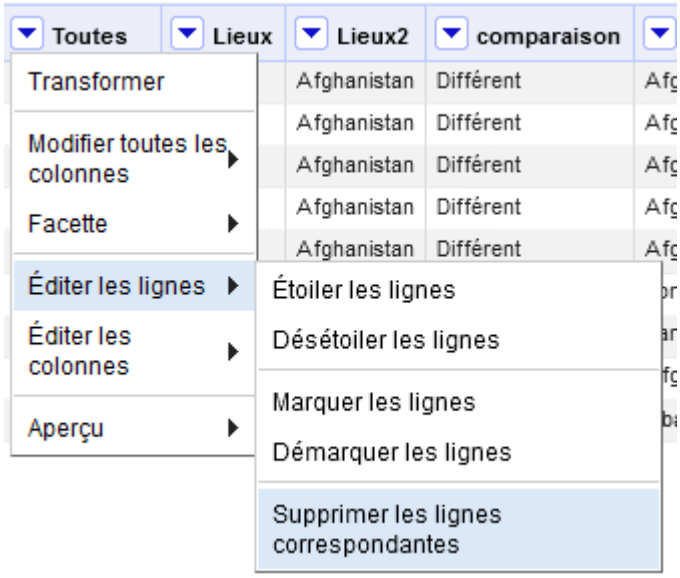

Figure 2.20: Supprimer les lignes pour lesquelles l'alignement est "Différent".

III. Vérifier les résultats

Quelle que soit la méthode employée pour aligner le jeu de données source avec le référentiel GeoNames, il est nécessaire de vérifier le résultat obtenu. Il est préférable que cette opération soit réalisée par l'auteur du jeu de données ou par une personne les connaissant suffisamment pour pouvoir détecter un alignement éventuel sur un homonyme.

IV. Ressources et références

- Blaney, J. (2017). “Introduction to the Principles of Linked Open Data”. *Programming Historian*. https://programminghistorian.org/en/lessons/intro-to-linked-data
- Hooland, S. van, Verborgh, R., & Wilde, M. D. (2013). “Nettoyer ses données avec OpenRefine”. *Programming Historian*. https://programminghistorian.org/fr/lecons/nettoyer-ses-donnees-avec-openrefine
- *GeoNames*. Consulté 21 janvier 2018, à l’adresse http://www.geonames.org/
- *OpenRefine user manual | OpenRefine*. Consulté 8 octobre 2021, à l’adresse https://docs.openrefine.org//
- Saby, M. *Tutoriel OpenRefine 3.4?: nettoyer, préparer et transformer des données - 06/11/2020*. Consulté 8 octobre 2021, à l’adresse https://msaby.gitlab.io/tutoriel-openrefine
- Williamson, E. P. (2017). “Fetching and Parsing Data from the Web with OpenRefine”. *Programming Historian*. https://programminghistorian.org/en/lessons/fetch-and-parse-data-with-openrefine

- 1 Pour voir comment installer et utiliser OpenRefine lire ce tutoriel
- 2 Pour les cas où les fichiers sont les moins structurés et où ces noms se trouvent au sein de phrases explicatives plutôt que dans des champs systématiques, il est obligatoire d’effectuer un nettoyage des données au préalable en divisant les phrases en fonction des espaces et en utilisant les facettes et regroupements proposés par OpenRefine comme cela est expliqué dans la partie 5 du tutoriel précédemment cité.
- 3 Il s’agit ici de vérifier qu’il n’y ait pas, par exemple, deux localisations au sein d’une même cellule comme “Paris, France” qui pourrait fausser la recherche
- 4 Cette base de données exportable est mise à jour quotidiennement
- 5 L’ensemble de ce chapitre est écrit à l’aide du tutoriel de M. Saby.
- 6 Ce module supplémentaire VIB-BITs, ajouté à l’installation de base d’OpenRefine, n’est disponible qu’en anglais, ce qui explique que cette fonction et ses dépendances ne soient pas en français comme les autres fonctions.
- 7 Attention, il est obligatoire que ces données se situent au sein d’un **projet OpenRefine**, cela signifie que les données GeoNames exportées précédemment doivent d’abord être importées dans OpenRefine. Ici ce projet se nomme Geonames_FR.
- 8 Lorsque les colonnes de correspondances ne possèdent pas un nom identique, la fusion des jeux de données ne fonctionne pas. Cette fenêtre permet en effet de sélectionner la colonne clé dans le second projet, mais pas dans celui actuellement ouvert.
- 9 On peut utiliser, dans OpenRefine, les facettes pour retrouver plus facilement les lignes manquantes, ainsi que le tri et le remplissage automatique de cellules répétées pour compléter plus rapidement le fichier (pour plus de détail sur ces éléments voir le tutoriel de M. Saby ici et ici)
- 10*Application Programming Interface* ou interface de programmation d’applications en français, plus de détail sur Wikipedia.
- 11 Un crédit correspond dans la plupart des cas à une requête. Plus d’informations ici.
- 12 Si aucun mail n’est arrivé au bout de deux minutes, il est possible qu’il est été considéré comme un SPAM.
- 13 L’acronyme GREL correspond à *General Refine Expression Language*, un langage d’expression développé d’après JavaScript pour OpenRefine. Pour plus d’informations voir le manuel d’OpenRefine.
- 14 Cette étape n’est utile que lorsque l’on ne recherche pas des pays directement ou lorsque les lieux requêtés sont situés dans différents pays.
- 15 Il faudra encore remplacer la formule `NOM_UTILISATEUR` pour le nom indiqué lors de la création du compte GeoNames (voir partie 2.2.1).
- 16 Plus d’informations sur la différence entre ligne et entrée sont disponibles dans le manuel d’OpenRefine ou dans la partie 5.18 du tutoriel de M. Saby.
- 17 Pour utiliser la fonction `if` sur les valeurs contenues dans une colonne donnée, il est nécessaire d’indiquer le nom de cette colonne entre `[]` et entre `""`, mais attention, OpenRefine est sensible à la casse.
- 18 Attention, pour réaliser cette opération, il est important d’avoir sélectionner l’option “**voir en lignes**” en haut à gauche et non en “**entrées**”.