

- Nous avons créé un micro-service SPARQL qui utilise une API externe pour interroger et récupérer des données sur des hôtels spécifiques. Le service a été configuré pour transformer les données JSON obtenues via l'API en données RDF, permettant ainsi une intégration fluide avec notre graphe de connaissances existant.
- La requête principale utilisée pour interroger l'API externe est structurée comme suit :

`api_query =`

[https://app.goflightlabs.com/retrieveHotel?access_key=\[ACCESS_KEY\]&entityId={hotel_ID}&checkin=2024-01-20&checkout=2024-01-23](https://app.goflightlabs.com/retrieveHotel?access_key=[ACCESS_KEY]&entityId={hotel_ID}&checkin=2024-01-20&checkout=2024-01-23)

- Cette requête utilise plusieurs paramètres clés :
 - **access_key** : La clé d'accès API sécurisée qui authentifie notre demande auprès du service externe.
 - **entityId** : L'identifiant unique de l'hôtel que nous souhaitons interroger.
 - **checkin** et **checkout** : Les dates d'arrivée et de départ qui nous permettent de récupérer des informations tarifaires pour des périodes spécifiques.
- La configuration du micro-service a été réalisée grâce à un fichier config.ini qui spécifie les paramètres personnalisés et la clé d'accès comme paramètres de la requête HTTP. Le mappage des données reçues de l'API externe a été défini dans un fichier profile.jsonld, qui décrit le contexte JSON-LD pour interpréter les données JSON en tant que données RDF. En outre, un fichier construct.sparql a été utilisé pour construire la requête SPARQL CONSTRUCT qui transforme les données JSON en triplets RDF selon notre modèle de données.

2. CSV :

Pour enrichir les données fournies par l'API de GoFlightLabs, nous avons utilisé un fichier CSV provenant du site web data.gouv.fr. Ce fichier CSV est une ressource précieuse car il comble les lacunes de l'API en fournissant des informations supplémentaires essentielles sur les hôtels. Il inclut des détails tels que l'adresse exacte, l'email de contact, le téléphone de contact, ainsi que la capacité d'accueil des établissements. L'intégration de ces données complémentaires nous permet d'offrir une vue plus complète et détaillée de chaque hôtel.

Afin de transformer notre fichier CSV en un fichier au format TTL (Turtle), nous avons utilisé l'outil csv2rdf. Ce processus a commencé par la création d'un fichier metadata.json, qui définit la structure et le schéma des données du CSV. Le fichier metadata.json sert de guide pour l'outil csv2rdf, lui permettant de comprendre comment interpréter et convertir les données du CSV en un format RDF compatible, spécifiquement en TTL.

Cette transformation en format TTL est une étape cruciale car elle nous permet d'interroger de manière cohérente et efficace les données issues à la fois de l'API de GoFlightLabs et du fichier CSV.

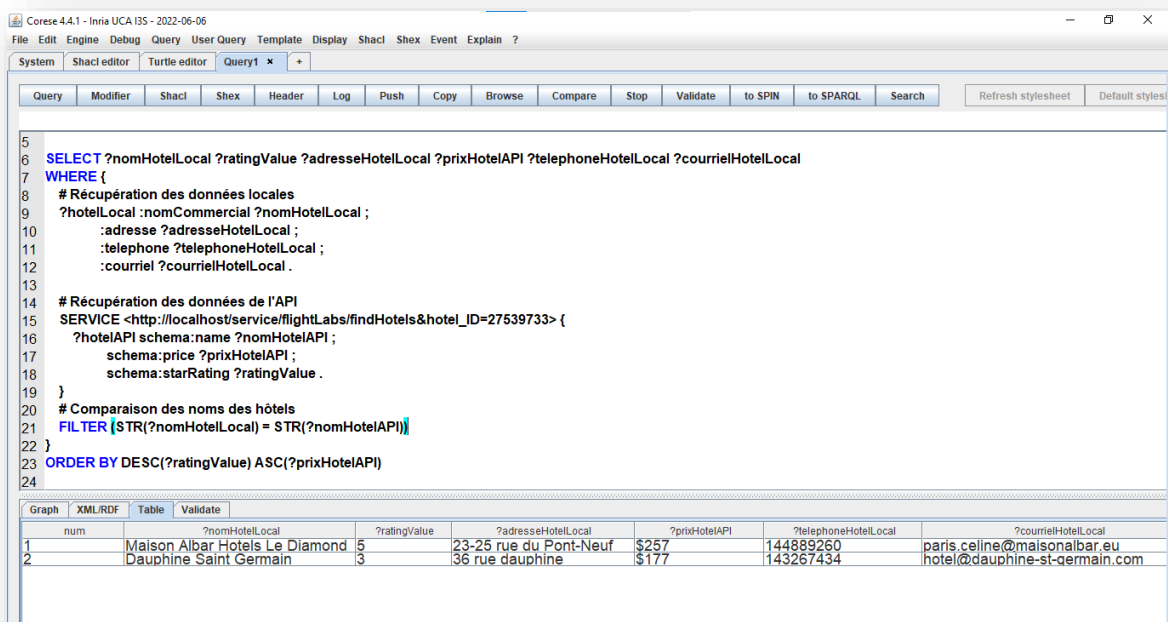
Après une génération de code via l'outil **csv2rdf** un échantillon d'une instance (ligne en csv) de l'hôtel **HÔTEL IBIS PARIS NANTERRE** se présente comme suit :

```
_:bnode_442 <http://example.org/vocab/dateClassement> "30/03/2016";
<http://example.org/vocab/datePublication> "30/03/2016";
<http://example.org/vocab/departement> "92";
<http://example.org/vocab/classement> "[3 étoiles]";
<http://example.org/vocab/nomCommercial> "HÔTEL IBIS PARIS NANTERRE";
<http://example.org/vocab/adresse> "36/38 AVENUE DES CHAMPS PIERREUX";
<http://example.org/vocab/codePostal> "92000";
<http://example.org/vocab/commune> "NANTERRE";
<http://example.org/vocab/telephone> "147241991";
<http://example.org/vocab/courriel> "H1554@accor.com";
<http://example.org/vocab/siteInternet> "www.accorhotels.com/fr/hotel-1554-ibis-nanterre-la-defense/index.shtml";
<http://example.org/vocab/capaciteAccueil> "246";
<http://example.org/vocab/nombreChambres> "123";
<http://example.org/vocab/typologieEtablissement> "HÔTEL";
<http://example.org/vocab/categorie> "-";
<http://example.org/vocab/mention> "-";
<http://example.org/vocab/typeSejour> "-";
<http://example.org/vocab/nombreEmplacements> "-";
<http://example.org/vocab/nombreUnitesHabitation> "-";
<http://example.org/vocab/nombreLogements> "-";
<http://example.org/vocab/geo> "48.888473 2.212579" .
```

3. Use cases :

- Comparateur de Prix d'Hôtels pour Voyageurs d'Affaires** : Les professionnels en déplacement recherchent souvent des hôtels qui non seulement offrent un bon rapport qualité-prix, mais qui sont également situés à proximité de leurs lieux de travail ou de réunions d'affaires. Le système utilise la requête SPARQL pour combiner des données locales sur les équipements des hôtels (comme les salles de conférence, le Wi-Fi haut débit) avec des informations dynamiques sur les prix et la disponibilité obtenues via l'API. Les utilisateurs peuvent filtrer les hôtels en fonction de leur distance par rapport à un point d'intérêt spécifié, visualiser les classements et comparer les prix pour les dates sélectionnées, facilitant ainsi la prise de décisions éclairées pour leurs réservations.
- Analyse de Marché pour les Agences de Voyages** : Les agences de voyages utilisent ce service pour analyser le marché hôtelier dans une région spécifique, en se basant sur des données actualisées sur les classifications par étoiles et les gammes de prix. Elles peuvent identifier les tendances telles que les périodes de l'année où les prix sont susceptibles de fluctuer, ce qui leur permet de proposer des offres compétitives à leurs clients. En intégrant les données de l'API avec des informations telles que les avis des clients et les services offerts, les agences peuvent créer des packages personnalisés et des recommandations pour différents segments de clients, qu'il s'agisse de voyageurs économiques ou de clients de luxe.
- Gestion Immobilière pour les Plateformes de Location de Vacances** : Les plateformes de location de vacances peuvent intégrer ce service pour fournir à leurs utilisateurs des données complètes sur les logements, en associant les caractéristiques des propriétés locatives (comme l'emplacement, la taille et les commodités) avec des informations externes sur les hôtels avoisinants. Cela permet à ces plateformes de mettre en évidence des propriétés avec des avantages compétitifs, comme être moins cher que les hôtels voisins tout en offrant des commodités comparables. Cela aide également les propriétaires de logements à positionner leurs prix de manière stratégique par rapport aux hôtels locaux en fonction de leur classement et de leurs services.

4. Federrated Query :



The screenshot shows the Corese 4.4.1 interface with a SPARQL query and its results. The query is as follows:

```

5 SELECT ?nomHotelLocal ?ratingValue ?adresseHotelLocal ?prixHotelAPI ?telephoneHotelLocal ?courrielHotelLocal
6 WHERE {
7   # Récupération des données locales
8   ?hotelLocal :nomCommercial ?nomHotelLocal ;
9     :adresse ?adresseHotelLocal ;
10    :telephone ?telephoneHotelLocal ;
11    :courriel ?courrielHotelLocal .
12
13   # Récupération des données de l'API
14   SERVICE <http://localhost/service/flightLabs/findHotels&hotel_ID=27539733> {
15     ?hotelAPI schema:name ?nomHotelAPI ;
16     schema:price ?prixHotelAPI ;
17     schema:starRating ?ratingValue .
18   }
19
20   # Comparaison des noms des hôtels
21   FILTER (STR(?nomHotelLocal) = STR(?nomHotelAPI))
22 }
23 ORDER BY DESC(?ratingValue) ASC(?prixHotelAPI)
24
  
```

The results table shows the following data:

num	?nomHotelLocal	?ratingValue	?adresseHotelLocal	?prixHotelAPI	?telephoneHotelLocal	?courrielHotelLocal
1	Maison Albar Hotels Le Diamond	5	23-25 rue du Pont-Neuf	\$257	144889260	paris.celine@maisonalbar.eu
2	Dauphine Saint Germain	3	36 rue dauphine	\$177	143267434	hotel@dauphine-st-germain.com

Cette requête SPARQL est extrêmement utile pour les plateformes qui souhaitent offrir à leurs utilisateurs une expérience complète en matière de recherche d'hébergements. En se connectant à une API externe, la requête permet de lister tous les hôtels d'une zone spécifique, tout en enrichissant les données locales existantes avec des informations complémentaires fournies par l'API, telles que les prix actualisés et les évaluations des établissements.

Elle effectue un mapping intelligent entre les données issues d'un fichier CSV local, qui contient des informations telles que le nom, l'adresse, le téléphone, et l'email de l'hôtel, et celles récupérées via l'API, assurant ainsi une correspondance précise entre les enregistrements locaux et distants. La requête priorise les résultats en fonction du classement par étoiles et du prix, permettant ainsi aux utilisateurs de facilement identifier et contacter les hôtels répondant à leurs critères de qualité et de budget. En somme, cette approche hybride offre une richesse de données pertinente et à jour, cruciale pour les décisions de réservation des utilisateurs.

5. Screenshot :

Query `http://localhost/service/flightLabs/findHotels&hotel_ID=27539733`

```

1 prefix : <http://ns.inria.fr/sparql-micro-service/api#>
2 prefix xsd: <http://www.w3.org/2001/XMLSchema#>
3 prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
4 PREFIX schema: <http://schema.org/#>
5 prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>
6 SELECT * WHERE {
7   ?sub ?pred ?obj .
8 }

```

110 results in 0.313 seconds

Could not render results with the Timeline plugin, the results currently are rendered with the Table plugin. See [Timeline documentation](#) for more information.

sub	pred	obj
http://example.org/hotel/104926371	schema:distance	1.46 miles from downtown
http://example.org/hotel/104926371	schema:identifier	104926371
http://example.org/hotel/104926371	schema:name	Hotel Libertel Gare de l'Est Francais
http://example.org/hotel/104926371	schema:price	\$240
http://example.org/hotel/104926371	schema:priceRange	\$240 for 1 night
http://example.org/hotel/104926371	schema:starRating	3
http://example.org/hotel/104926371	rdf:type	schema:Hotel
http://example.org/hotel/104926371	geo:coordinates	2.3570
http://example.org/hotel/104926371	geo:coordinates	48.876
http://example.org/hotel/104926371	geo:lat	

1) Querying the the API

System Shacl editor Turtle editor Query1 x +								
Query Modifier Shacl Shex Header Log Push Copy Browse Compare Stop Validate to SPIN to SPARQL Search Refresh stylesheet Default styles								
<pre> 1 PREFIX ex: <http://example.org/vocab/> 2 3 SELECT ?nom ?classement ?adresse ?ville ?telephone ?siteWeb 4 WHERE { 5 ?hotel ex:nomCommercial ?nom ; 6 ex:classement ?classement ; 7 ex:adresse ?adresse ; 8 ex:codePostal ?codePostal ; 9 ex:commune ?ville ; 10 ex:telephone ?telephone ; 11 ex:courriel ?courriel ; 12 ex:siteInternet ?siteWeb ; 13 ex:geo ?geo . 14 }</pre>								
Graph	XML/RDF	Table	Validate					
num	?nom		?classement	?adresse		?ville	?telephone	?siteWeb
1	THE FIVE HA"TEL		(3 A)toiles	3 RUE FLATTERS		PARIS	143317421	www.thefivehotel.com
2	HA"TEL TAMARIS		(3 A)toiles	14 RUE DES MARAICHERS		PARIS	143728548	www.hotel-tamaris.fr
3	HA"TEL SPLENDE ROYAL		(5 A)toiles	18 rue du Cirque		PARIS	143871010	www.splendideroyal.fr
4	LOUEST HA"TEL		(3 A)toiles	3 rue du Rocher		PARIS	143875749	ouest-hotel-paris.com
5	HOTEL ROYAL SAINT MICHEL		(4 A)toiles	3 boulevard Saint Michel		PARIS	144070606	www.hotelrovalsaintmichel.com
6	HA"TEL DES BEAUX ARTS		(2 A)toiles	2 rue Toussaint Ferron		PARIS	144242260	www.hotel-beaux-arts.fr
7	PARC HA"TEL PARIS		(3 A)toiles	60 RUE BEAUNIER		PARIS	145407702	www.parc-hotel-paris.com
8	HA"TEL LE RICHEMONT		(3 A)toiles	17 rue Jean Coily		PARIS	145828484	www.vacancier.com
9	AMBIANCE HA"TEL		(1 A)toile	8 rue André Citroën		LA QUEUE-EN-BRIE	145940808	ambiance-hotel.net
10	HA"TEL VILLA SOREL		(2 A)toiles	20 rue Georges Sorel		BOULOGNE-BILLANCOURT	146049158	www.villasorel.com
11	HA"TEL DU NORD ET DE LEST		(3 A)toiles	49 RUE DE MALTE		PARIS	147007170	www.hotel-nord-est.com
12	HA"TEL CAMPANILE PARIS SUD POR...		(3 A)toiles	73 avenue Aristide Briand		ARCUEIL	147408709	www.campanile.com/fr/hotels/camp
13	HOLIDAY INN PARIS OPA%RA GRAN...		(4 A)toiles	30 32 boulevard Poissonnière		PARIS	147702555	www.holidayinn.com/grands-blvds
14	HA"TEL AMIRAL		(2 A)toiles	90 rue de l'Amiral Roussin		PARIS	148285389	www.hotelamiral.com

2) Querying the the RDF turtle file

NB: Veuillez revenir vers le dossier principal afin d'avoir une vision plus claire des captures d'écran, si vous voulez accéder à des fichiers particuliers ou si vous voulez voir la hiérarchie du projet.

Lien *github* du projet : <https://github.com/alihaitam/mini-projet-data-lifting>