UNIVERSITÉ DE VERSAILLES ST-QUENTIN-EN-YVELINES

université PARIS-SACLAY

Récupération des données musicales par des techniques sur des données sémi-structurées

Jingwei ZUO
Qiuhao QIAN

Sommaire

01 | Introduction

- ✓ Environnement & Techniques
- ✓ Sources de données

02 | Structure globale

- ✓ Des données envisagées
- ✓ Structure de BD
- ✓ Structure de services web

03 | Implémentation

- ✓ Récupération des sources de données
- ✓ Conversion par XSLT et DOM
- ✓ Web service exposé

04 | Problèmes & Solutions

- ✓ Sources de données
- ✓ Conversion
- ✓ Manipulation de BD

05 | Conclusion

UNIVERSITÉ DE VERSAILLES ST-QUENTIN-EN-YVELINES

université PARIS-SACLAY

01 | Introduction

✓ Environnement du travail

✓ Environnement & Techniques

- ➢ Eclipse + Tomcat 7.0 + Axis2 (Web services), JRE1.8
- ➢ Postman(Test de REST requête)
- ➢ XSLT + DOM/SAX(Transition et extraction de données démi-structurées)
- ➢ Mysql(BD pour stocker des données envisagées)
- ➢ SOA
- ➢ SoapUI for Mac(Test de services Web générés)

✓ Sources de données musicales

- MusicBrainZ
- Lastfm

UNIVERSITÉ DE VERSAILLES ST-QUENTIN-EN-YVELINES

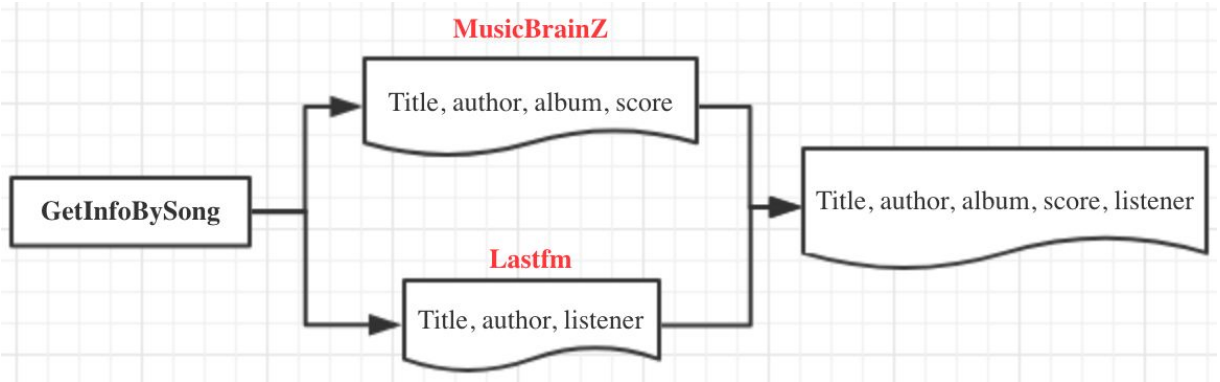
université PARIS-SACLAY

02 | Structure globale

- ✓ Des données envisagées
- ✓ Structure de BD
- ✓ Structure des services web

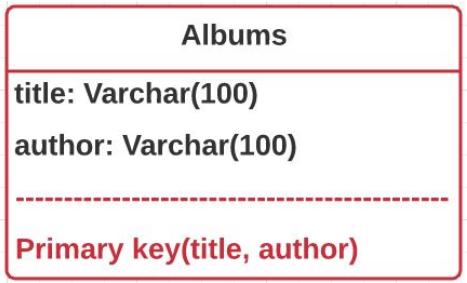
✓ Des données envisagées

- GetSongsByAuthor: Song title
- GetAlbumsByAuthor: Albums name
- GetInfoBySong: Title, author, album, score, listener

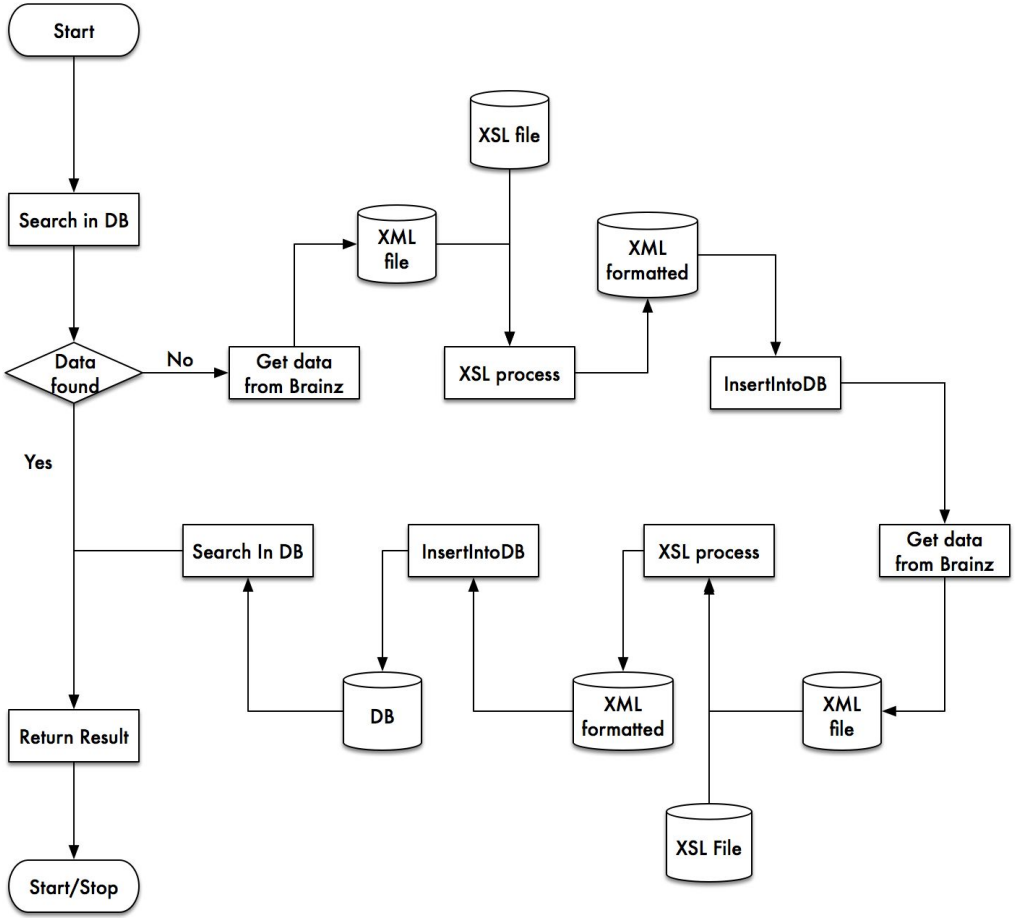


✓ Structure de base de données

- Deux tables:
- Table1: Songs (Title, author, album, score_brainz, listeners_lastfm)
- Table 2: Albums(Title, author)



✓ Structure de processus



03 | Implémentation

- ✓ Récupération des sources de données
- ✓ Conversion par XSLT et DOM/SAX
- ✓ Web service exposé

MusicBrainZ

- Source de musique plus complète
- REST API
- 3rd party libraries supporté de Java

Lastfm

- Sources de musique sont principalement copiées depuis MusicBrainZ
- REST API

MusicBrainZ, REST API

getSongsByAuthor:
http://musicbrainz.org/ws/2/recording/?query=artistname:"Luca Barbarossa"
MBID, Title, author, album

getAlbumsByAuthor:
http://musicbrainz.org/ws/2/release/?query=artist:Dave
MBID, Title, author, status, language, aritst_name, data, country

getInfo:
http://musicbrainz.org/ws/2/recording/?query=recording:"Sufriendo de amor"
Title, length, author, album, score

Lastfm, REST API

getSongsByAuthor: **artist_getTopAlbums(artist, api_key)**
http://ws.audioscrobbler.com/2.0/?method=artist.gettoptracks&artist=wason&api_key=fdc873135eefe53670783bd15d92eed5
Title, author

getAlbumsByAuthor: **artist.getTopTracks(artist, api_key)**
http://ws.audioscrobbler.com/2.0/?method=artist.gettopalbums&artist=cher&api_key=fdc873135eefe53670783bd15d92eed5
MBID,Title, author

getInfo: **track.search(track, api_key) ou Track.getInfo(track, artist, api_key)**
http://ws.audioscrobbler.com/2.0/?method=track.search&track=Believe&api_key=fdc873135eefe53670783bd15d92eed5 OU
http://ws.audioscrobbler.com/2.0/?method=track.getInfo&track=Believe&artist=xxx&api_key=fdc873135eefe53670783bd15d92eed5
Title, author, listener OU Title, author, listener, Album, play_numbers, etc.

Résultats exposé aux clients

getSongsByAuthor:
Title, author

getAlbumsByAuthor:
Title, author

getInfoBySong, **fusionner des attributs depuis des deux sources**
Title, author, album, score_brainz, listeners_lastfm

Conversion par XSLT et DOM/SAX

getSongsByAuthor & GetInfoBySong

Node	Content
xml	version="1.0" encoding="utf-8"
Brainz_1	
xmlns:ext	http://musicbrainz.org/ns/ext#-2.0
xmlns:n	http://musicbrainz.org/ns/mmd-2.0#
song	
song_title	Believe
Artist	Ministers De-La-Funk
Album	2020
Score	100
listeners	
song	
song	

Primary key

Node	Content
xml	version="1.0" encoding="utf-8"
lastfm_3	
song	
song_title	Believer
Artist	Imagine Dragons
Album	
Score	
listeners	184571
song	
song	

Primary key

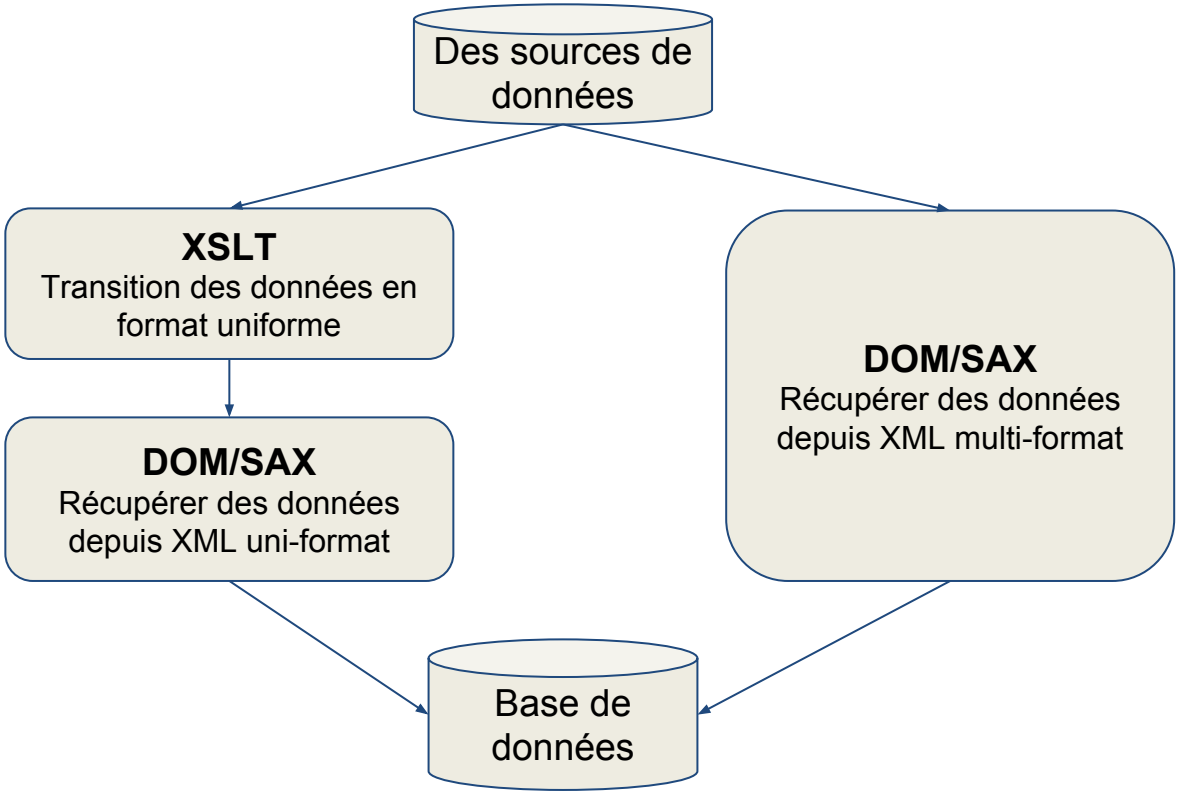
Conversion par XSLT et DOM/SAX

getAlbumsByAuthor:

Node	Content
?-? xml	version="1.0" encoding="utf-8"
▼ e Brainz_2	
ⓐ xmlns:ext	http://musicbrainz.org/ns/ext#-2.0
ⓐ xmlns:n	http://musicbrainz.org/ns/mmd-2.0#
▼ e Album	
e Album_title	Alma mía
e Artist	Wason
▶ e Album	

Node	Content
?-? xml	version="1.0" encoding="utf-8"
▼ e lastfm_2	
▼ e album	
e Album_title	Alma Mía
e Artist	Wason
▶ e album	

Conversion par XSLT et DOM/SAX



04 | Problèmes & Solutions

✓ Des attributs des sources sont différents

✓ -> Choisir des attributs en commun

✓ -> Fusionner des attributs

✓ Namespace de données venant de MusicBrainZ et Lastfm

✓ -> Dans notre script XSLT, il faut ajouter des namespaces

```
<xsl:stylesheet version="2.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:n = "http://musicbrainz.org/ns/mmd-2.0#" xmlns:ext="http://musicbrainz.org/ns/ext#-2.0">

<xsl:for-each select="n:recording">
  <song>
       <song_title>
        <xsl:value-of select="n:title"/>
       </song_title>
      <Artist>
        <xsl:value-of select="n:artist-credit/n:name-credit/n:artist/n:name"/>
      </Artist>
```

✓ Namespace OpenSearch dans le XML retourné par Lastfm

✓ -> Pré-traitement du fichier XML avant de faire la transition XSLT, pour enlever une partie de données(OpenSearch)

```
<?xml version="1.0" encoding="UTF-8" ?>
<lfm status="ok">
    <results>
        <opensearch:Query role="request" startPage="1"></opensearch:Query>
        <opensearch:totalResults>634211</opensearch:totalResults>
        <opensearch:startIndex>0</opensearch:startIndex>
        <opensearch:itemsPerPage>30</opensearch:itemsPerPage>
        <trackmatches>
            <track>
                <name>Believer</name>
                <artist>Imagine Dragons</artist>
                <url>https://www.last.fm/music/Imagine+Dragons/_/Believer</ur
                <streamable>FIXME</streamable>
                <listeners>184571</listeners>
                <image size="small">https://lastfm-img2.akamaized.net/i/u/34s
                <image size="medium">https://lastfm-img2.akamaized.net/i/u/64
                <image size="large">https://lastfm-img2.akamaized.net/i/u/174
                <image size="extralarge">https://lastfm-img2.akamaized.net/i/
                <mbid></mbid>
            </track>
```

✓ Web service client, connexion avec des services

DB_test.java | *services.xml

```
1  <service name="GetAlbumByAuthor" >
2    <Description>
3      Please Type your service description here
4    </Description>
5    <messageReceivers>
6      <messageReceiver mep="http://www.w3.org/2004/08/wsdl/in-only"
7      <messageReceiver  mep="http://www.w3.org/ns/2004/08/in-out"
8    </messageReceivers>
9    <parameter name="ServiceClass" locked="false">service.GetAlbumBy
10 </service>
```

DB_test.java | services.xml

```
1  <service name="GetAlbumByAuthor" >
2    <Description>
3      Please Type your service description here
4    </Description>
5    <messageReceivers>
6      <messageReceiver mep="http://www.w3.org/ns/wsdl/in-only"
7      <messageReceiver  mep="http://www.w3.org/ns/wsdl/in-out"
8    </messageReceivers>
9    <parameter name="ServiceClass" locked="false">service.GetAl
10 </service>
```

La version de messageReceiver par défaut est trop ancienne. Il faut remplacer le lien par celui montré dans l'image au-dessous

✓ **Upload des fichiers sur le serveur de Web service**

✓ **-> Des fichiers XSLT sont stockés à local**

Proposition de solutions:

1. Lire des URLs de nos scripts XSLT dans le code(e.g. Github)

   www.github.com/XXX/transition.xslt

2. Ecriture en dur des scripts XSLT dans le code

-> Meilleure solution?

05 | Conclusion

✓ Résultats

REST API

- Récupérer des données depuis MusicBrainZ et lastfm
- Pré-traitement sur des données avant de faire la transition

Transitions et extraction

- Des transitions sur des données ver un format unique par XSLT
- Extractions des données à insérer dans la DB par DOM/SAX

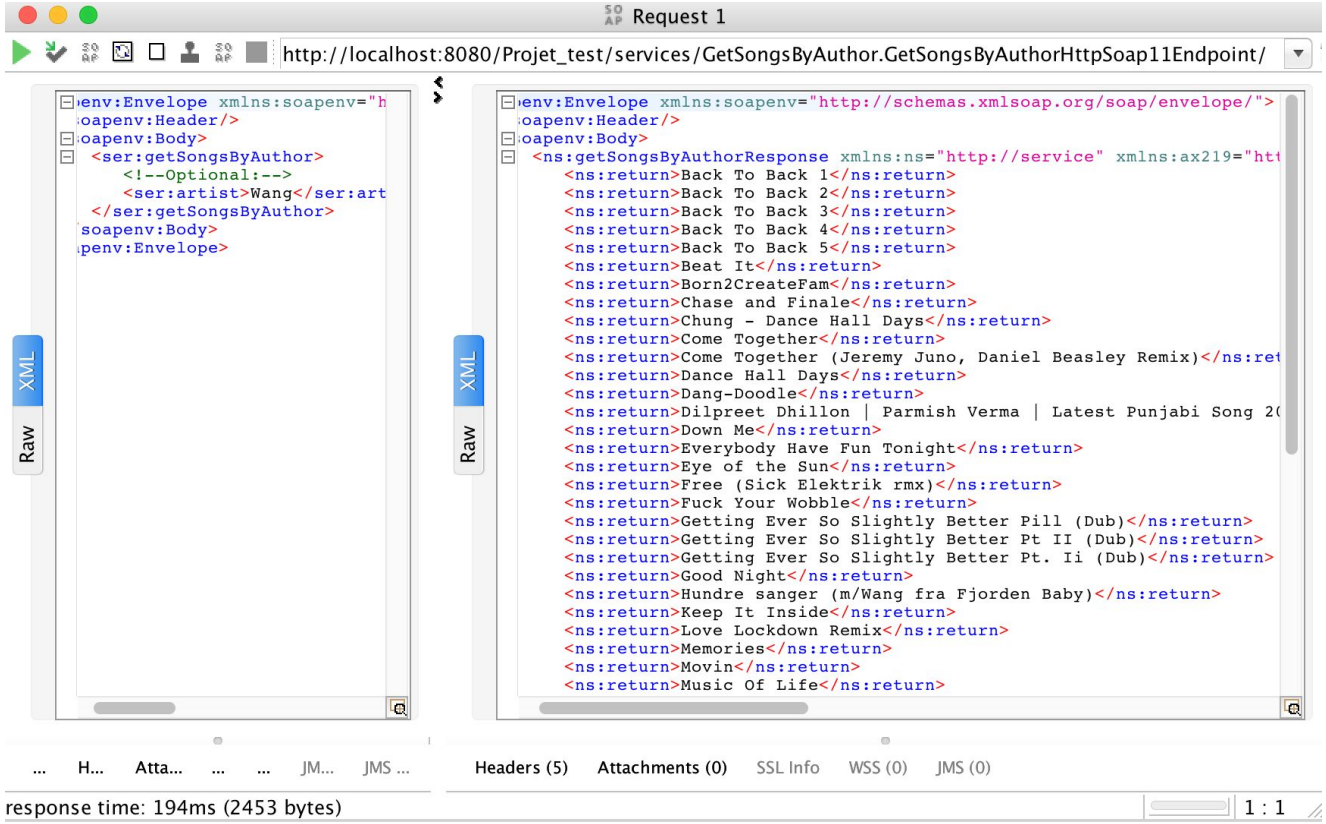
Résultats

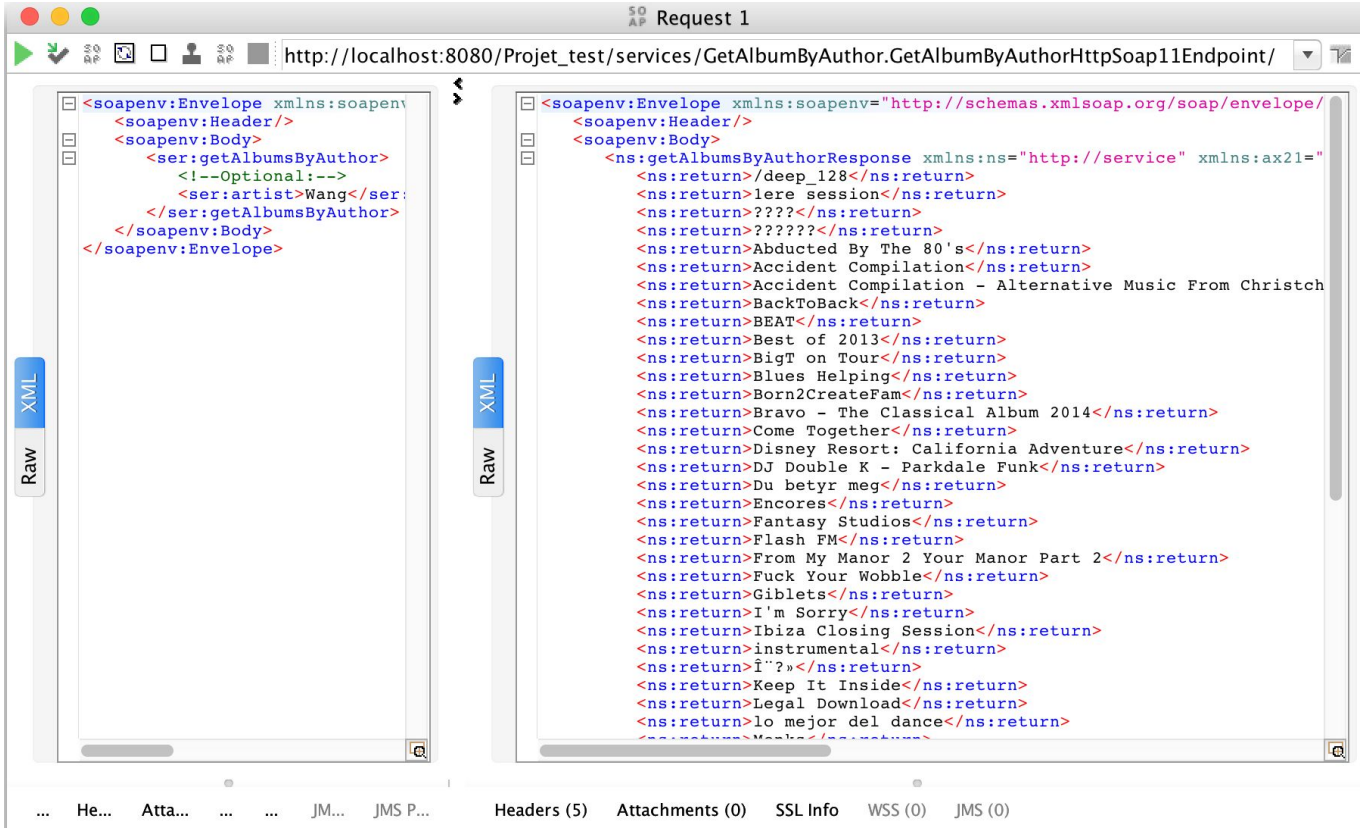
- Analyse sur des données récupérées depuis le Web
- Définition des structure du processus. e.g fusionner des attributs, etc.

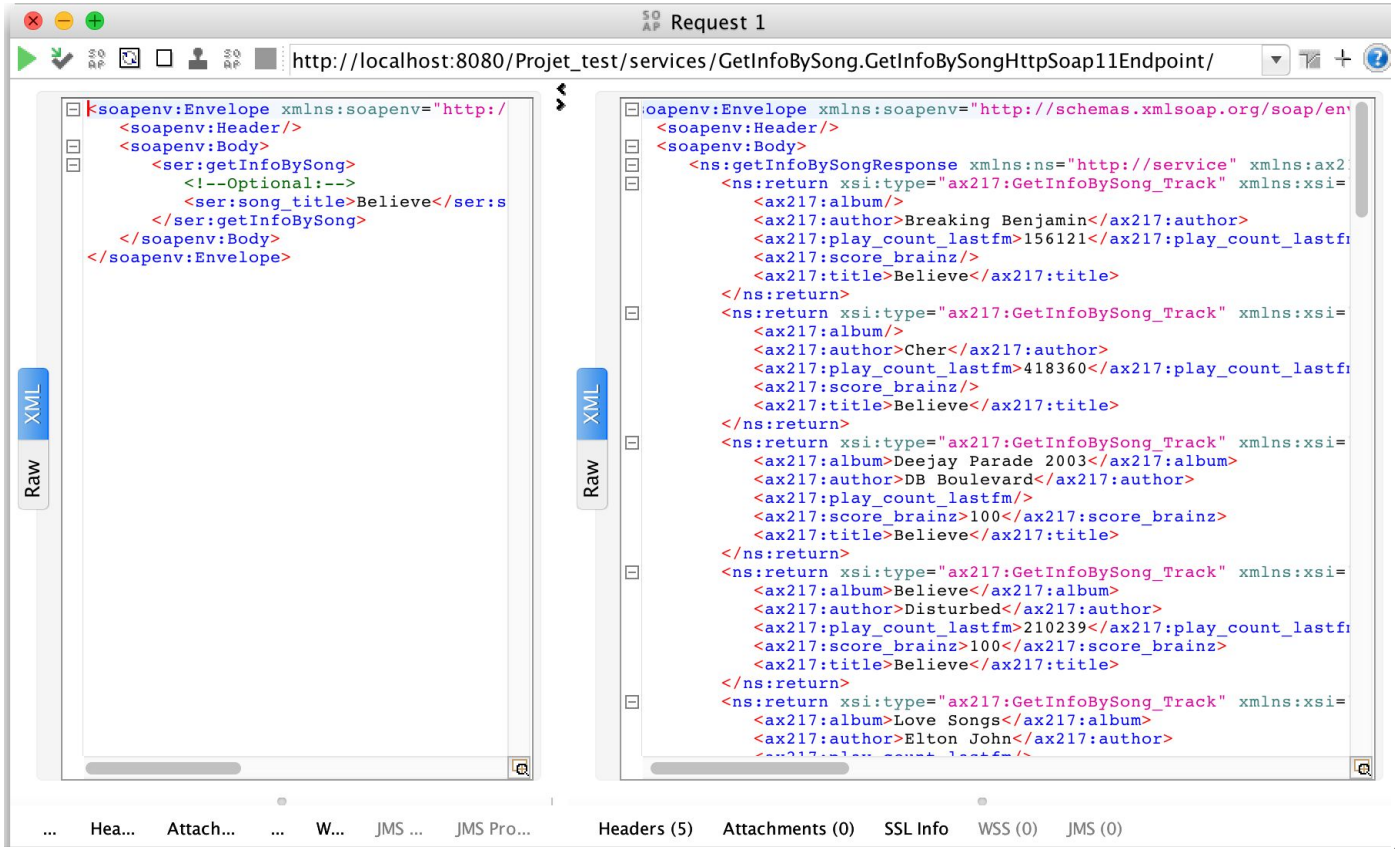
Analyse sur des données

- 3 services répondant aux besoins sont implémenté et testé

Exposer des 3 services web







UNIVERSITÉ DE VERSAILLES ST-QUENTIN-EN-YVELINES

université PARIS-SACLAY

Merci

THANK YOU

Questions?