Cahier de TP
« Apache SolR»

Pré-requis :

- OS recommandé : Linux ou Windows 10
- JDK17+
- Bon éditeur XML (Notepad++, Vscode, …)
- Plusieurs machines en réseau ou Docker pour le TP Optionnel sur SolrCloud

Table des matières

Atelier 1 : Installation et démo

Récupérer une distribution de Apache Solr et la dézipper dans un répertoire de travail :

($SOLR_HOME)

Démarrer la configuration cloud

```
cd $SOLR_HOME

./bin/solr start -e cloud
```

Répondre à l'assistant en choisissant :

- 2 nœuds
- les ports proposés
- une collection de données nommée ***techproducts***
- 2 shards, 2 répliques
- ***sample_techproducts_configs*** comme configuration

Ouvrir et découvrir l'interface d'administration : http://localhost:8983/solr/

En particulier visualiser le lien cloud

Indexer des données avec :

```
./bin/post -c techproducts example/exampledocs/*
```

OU

```
./bin/solr post -c techproducts example/exampledocs/*
```

Ensuite visualiser les documents via l'interface d'admin en effectuant des recherches

- Tous les documents
- Les documents contenant le terme ***foundation***
- Les documents dont le champ ***cat*** contient ***electronics***
- Les documents contenant la phrase ***« Memory stick »***

Vous pouvez également utiliser curl pour effectuer ses requêtes

Supprimer la collection via :

```
bin/solr delete -c techproducts
```

Arrêter le cluster

```
bin/solr stop -all
```

Atelier 2 : Mise en place de cœur

2.1 Création de cœurs

Démarrer une configuration standalone de Solr

```
bin/solr start
```

Vérifier http://localhost:8983/solr

Créer un cœur nommé ***formation_managed*** (Schéma managé par SolR et mode *schemaless*)

```
bin/solr create -c formation_managed -d _default
```

Visualiser les fichiers de configuration créés dans server/solr/formation_managed/. En particulier :

- Le fichier `managed-schema`
- `SolrConfig` et la balise `schemaFactory`

Visualiser la config via :

- L'API Rest :

```
curl "http://localhost:8983/solr/formation_managed/config"
```

- Via la console d'administration

Créer un autre cœur nommé ***formation***, en créant au préalable un répertoire de configuration, en se positionnant en mode contrôle exclusif du schéma sans possibilité d'ajout de champ

Créer le répertoire et copier la configuration de base :

```
mkdir -p server/solr/configsets/formation_conf/conf

cp -r server/solr/configsets/_default/conf/*
server/solr/configsets/formation_conf/conf/
```

Remplacer la config par défaut en éditant `solrconfig.xml et en fixant la propriété schemaFactory à` ***`ClassicIndexSchemaFactory`***

Renommer le fichier *managed-schema* en *schema.xml*

Créer le coeur formation :

```
bin/solr create -c formation -d
server/solr/configsets/formation_conf/conf
```

Test de l'indexation, essayer ces 2 requêtes d'indexation

```
curl "http://localhost:8983/solr/formation_managed/update?
commit=true" \

  -H "Content-Type: application/json" \

  -d '[{"id":"1","nom":"PLB","ville":"Paris"}]'
```

ET

```
curl "http://localhost:8983/solr/formation/update?commit=true" \

  -H "Content-Type: application/json" \

  -d '[{"id":"1","nom":"PLB","ville":"Paris"}]'
```

2.2 Comprendre l'analyse

Testez l'analyse sur la chaîne « Une formation débutant sur SolR » :

- Avec le type de champ ***text_ws***
- Avec le type de champ ***text_general***
- Avec le type de champ ***text_fr***

Que constatez-vous ?

Lisez les commentaires associés à ces types de champs dans ***schema.xml***

Essayez avec :

« Overview of Documents, Fields, and Schema Design» et le champ phonétique english

Exécuter le test fourni pour visualiser les effets des annotations

2.3 Analyseur phonétique français

Définir un nouveau type de champs effectuant une analyse phonétique en français, y associer un champ dynamique et tester l'analyse

Atelier 3 : Indexation

3.1 Indexation XML

Visualiser le fichier XML fourni et dans le cœur formation modifier le fichier ***schema.xml*** pour être le plus précis sur les champs utilisés.
Effectuer la bonne configuration dans ***solrconfig.xml*** ou ***configoverlay.json*** pour interdire tout nouveau champ dans le schéma
Utiliser le fichier XML fourni pour alimenter les 2 cœurs (*formation* et *formation_managed*)

3.2 Indexation JSON

Reprendre le fichier ***slides.json*** fourni et effectuer une requête permettant d'insérer le document dans le schéma maîtrisé précédent

3.3 Indexation CSV (Optionnel)

Ajouter un document au format CSV dans les cœurs précédents

Atelier 4 : Importation de documents bureautique

Créer un nouveau cœur

Configurer le gestionnaire d’importation

Utiliser l’utilitaire *solr* ou *curl* pour indexer les documents bureautiques fournis.

Atelier 5 : Importation base de données

Démarrer l’exemple DIH.

Visualiser les différentes configurations :

- Chargement des librairies DIH
- Configuration du DataImportHandler
- Configuration de la source de données
- Emplacement du driver JDBC

Exécuter les requêtes HTTP permettant

- De visualiser les statistiques
- D’effectuer un import complet de la base

Atelier 6 : Configuration request handler et 1ères recherches

6.1 Configuration

Base bureautique :

Configurer le request handler pour :

- Travailler par défaut sur le champ *content*
- Forcer une sortie en *json*
- Utiliser par défaut le parseur lucene
- Désactiver les *searchcomponent* : *facet, morelikethis* et *highlight*
- Ajouter _score dans les champs retournés
- Informations de debug

6.2 Requêtes

Effectuer les recherches suivantes en utilisant la syntaxe lucene :

- Documents répondant à « Java »
- Documents ne répondant pas à « Java »
- Limiter les documents retournés de la première requête
- Documents dont le contenu répond à « Java »
- Documents PDF dont le contenu répond à « Java »
- Documents dont le contenu répond à « SolR»•
- Documents dont le champ titre contient administration
- Document créés après une date particulière
- Document créés après une date particulière et dont le contenu répondant « Java Elastic Search » mais pas « Administration »

Atelier 7 : Différents types de recherches

7.1 Spell-check

Voir la configuration du spell-check

7.2 Highight

Utiliser les paramètres de highlight

Atelier 8 : Agrégation de documents

8.1 Facettes

Utiliser les paramètres liés aux facettes

Atelier 9 : Recherche géo-graphique et client Java

Compléter l'application SpringBoot afin de pouvoir importer des données de géo-location dans un cœur SolR :

- Ajout des dépendances vers SolrJ
- Classe de configuration SolR créant un bean HttpClient
- Implémentation d'une classe service ajoutant un document dans un cœur à partir de la classe du modèle *Position.java*

Préparer un cœur définissant des champs géographique avec les types :

- LatLonPointSpatialField
- SpatialRecursivePrefixTreeFieldType

Effectuer ensuite les requêtes suivantes :

- Rechercher les documents via un rectangle
- Distance à partir d'un point central
- idem avec en plus des facettes
- Agrégation de type heatmap sur le champ RPT

Atelier 10 : SolrCloud

Visualiser le fichier docker-compose fourni

Démarrer le cluster et vérifier le bon démarrage de tous les processus

Ajouter une collection au cloud avec 2 shards et 1 réplique