

Bases de données
documentaires et distribuées,
<http://b3d.bdpedia.fr>

Interrogation avec les requêtes
Lucene

Interrogation

- ▶ Elasticsearch s'appuie sur le système d'indexation **Lucene**, dont le rôle est essentiellement de créer les index inversés, et d'implanter les algorithmes de parcours brièvement introduits dans la session précédente.
- ▶ Lucene propose un langage de recherche basé sur des combinaisons de mot-clés, langage étendu et raffiné par Elasticsearch (cf plus tard)
- ▶ La première méthode pour transmettre des recherches est de passer une expression en paramètre à l'URL :

```
http://localhost:9200/nfe204/movies/_search?q=alien
```

```
http://localhost:9200/nfe204/movies/_search?q=alien,coppola
```

```
http://localhost:9200/nfe204/movies/_search?q=alien,coppola,1994
```

```
# utilisable dans l'interface Kopf, onglet Rest
```

la réponse d'ES

```
{
  "took" : 1,
  "timed_out" : false,
  "_shards" : {
    "total" : 1,
    "successful" : 1,
    "failed" : 0
  },
  "hits" : {
    "total" : 1,
    "max_score" : 0.558217,
    "hits" : [ {
      "_index" : "nfe204",
      "_type" : "movies",
      "_id" : "movie:2",
      "_score" : 0.558217,
      "_source":{ "title": "Alien", "year": 1979,
        "genre": "Science-fiction",
        "summary": "... ",
        "country": "USA",
        "director": "Ridley Scott",
        "actors": [ ["Sigourney Weaver"] ] }
    } ]
  }
}
```

Autre méthode : en *postant* un document-requête

On fournit un document décrivant la requête (méthode POST) :

```
{
  "query": {
    "query_string" : {
      "query" : "alien,coppola,1994"
    }
  }
}
```

Un premier pas vers une interrogation plus structurée.

Termes

- ▶ Notion de base : le **terme**
- ▶ c'est un mot au sens usuel
- ▶ ou une séquence de mots entre apostrophes

On peut interroger un index avec :

Neo apprend

Puis :

"Neo apprend"

Première recherche : documents avec "Neo", "apprend" ou les deux

Termes (suite)

- ▶ la recherche d'un terme s'effectue toujours sur un champ.
- ▶ La syntaxe complète pour associer le champ et le terme est :

```
champ:terme
```

- ▶ si non précisé, c'est le champ par défaut qui est utilisé
- ▶ pratique courante : concaténer toutes les chaînes de caractères en un champ “_all” général, défini par défaut
- ▶ Nos requêtes deviennent :

```
_all:Neo _all:apprend
```
- ▶ et

```
_all:"Neo apprend"
```

Termes (suite)

- ▶ Les valeurs des termes (dans la requête) et le texte indexé sont tous deux soumis à des transformations spécifiées dans le schéma.
- ▶ Une transformation simple est de tout transcrire en minuscules.

```
_all:"Neo APPREND"
```

- ▶ Les transformations appliquées à la requête ET au texte indexé doivent être cohérentes : si les termes sont transformés en majuscules, et le texte indexé en minuscules, on n'aura jamais de résultat !

Termes (suite)

On peut spécifier des termes (pas des séquences) incomplets

- ▶ le '?' indique un caractère inconnu ; "opti?al" désigne "optimal", "optical", etc.
- ▶ le '*' indique n'importe quelle séquence de caractères ; "opti*" pour toute chaîne commençant par "opti"
- ▶ Approximation avec ~ : rechercher "optimal" et "optimal~" (0 et 1 résultat, "optical")
La similarité est évaluée par une distance d'édition (nb opérations pour passer de "optimal" à "optical")
- ▶ Les intervalles sont exprimés avec [] (bornes comprises) ou { } bornes exclues

`year: [1984 T0 2010]`

Requêtes Booléennes

- ▶ Les critères peuvent être combinés avec des **opérateurs Booléens** :
AND, **OR** et **NOT**
- ▶ Attention : majuscules

```
year:[1990 TO 2005] OR title:M*  
year:[1990 TO 2005] AND NOT title:M*
```

- ▶ Par défaut, c'est un **OR** qui est appliqué
- ▶ Recherche sur plusieurs critères ramène l'union des résultats sur chaque critère pris individuellement

En conclusion

Ce qui précède constitue l'essentiel des opérations d'interrogation avec un moteur de recherche.

- ▶ On est loin de la finesse d'interrogation d'un langage comme SQL.
- ▶ On fait peu mais on fait bien : systèmes extrêmement efficaces, même pour de très grands volumes.

Surtout, deux différences essentielles :

- ▶ Ce sont des recherches **approchées** : on trouve tous les documents "proches de" la requête
- ▶ Le résultat est **ordonné** par pertinence décroissante.

Comme ça marche ? Voir la suite...