



Elasticsearch : moteur de recherche taillé pour le cloud

par David Pilato
@dadoonet et @elasticsearchfr



{ “speaker” : “David Pilato” }

\$ curl http://localhost:9200/**devoxx**/**speaker**/**dpilato**

```
{
  "nom" : "David Pilato",
  "jobs" : [
    { "boite" : "SRA Europe (SSII)", "mission" : "bon à tout faire", "duree" : 3 },
    { "boite" : "SFR", "mission" : "touche à tout", "duree" : 3 },
    { "boite" : "e-Brands / Vivendi", "mission" : "chef de projets", "duree" : 4 },
    { "boite" : "DGDDI (douane)", "mission" : "mouton à 5 pattes", "duree" : 7 } ],
  "passions" : [ "famille", "job", "deejay" ],
  "blog" : "http://dev.david.pilato.fr/",
  "twitter" : [ "@dadoonet", "@elasticsearchfr" ],
  "email" : "david@pilato.fr"
}
```



Abstract

- Un moteur ? Pourquoi faire ?
- Elasticsearch : une solution simple, complète, performante
- Et si on indexait Twitter ?

Faites du bruit sur **@DevoxxFR**
avec le hashtag **#elasticsearch** !



Un moteur ? Pour quoi faire ?

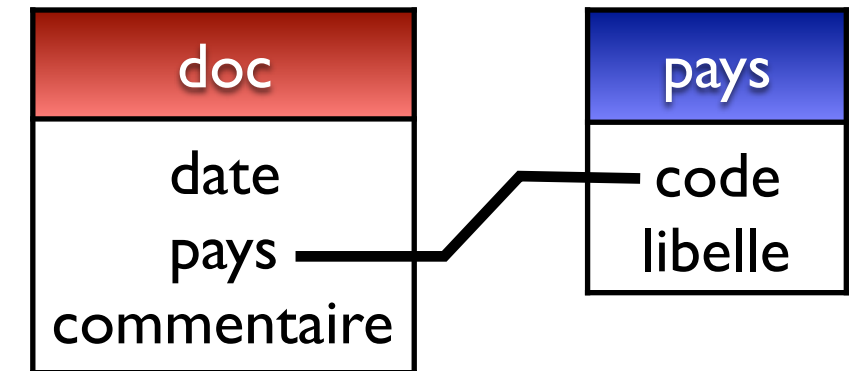
LE BESOIN



Cas d'école « SQL old school »

Un document dans une base de données :

- Un attribut **date** : *19/04/2012*
- Un attribut codifié **pays** : *FR*
- Correspondant à la table d'association code/libellé
 - **Code** : *FR*
 - **Libellé** : *France*
- Un attribut **commentaire** : *"J'observe une erreur de saisie dans la désignation commerciale du produit. Songer à téléphoner à David."*



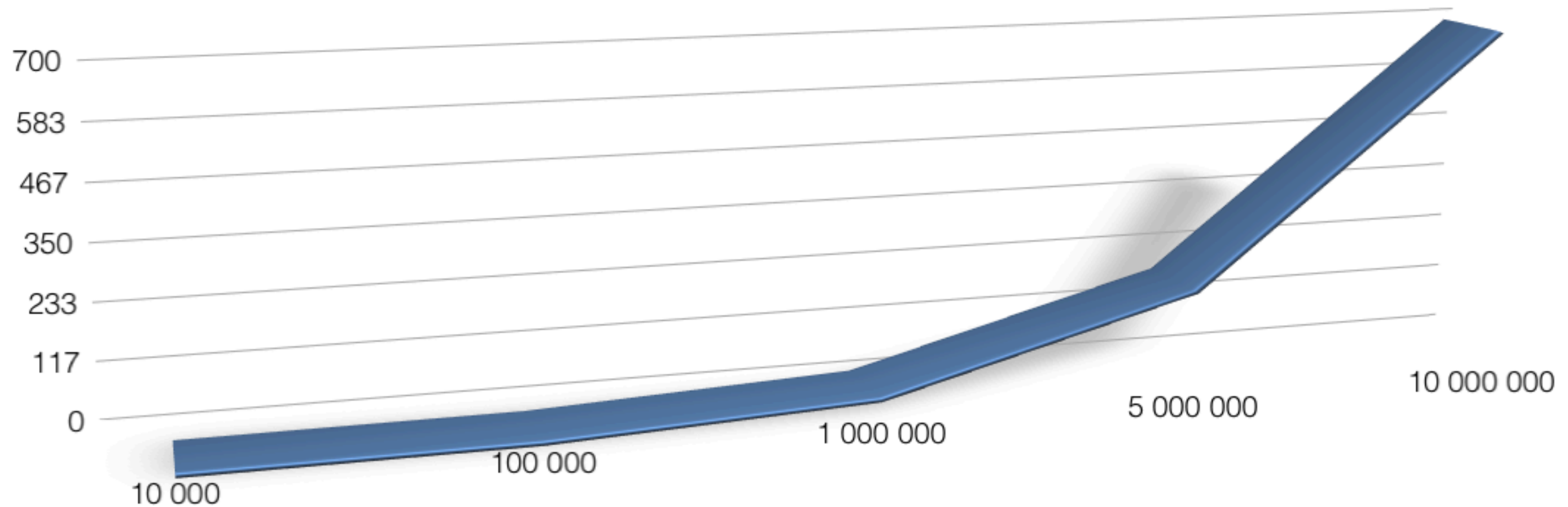
Cas d'école « SQL old school »

- Cherche moi un document de **décembre 2011** portant sur la **france** et contenant **saisie** et **david**
- En SQL :

```
SELECT
  doc.*, pays.*
FROM
  doc, pays
WHERE
  doc.pays_code = pays.code AND
  doc.date_doc > to_date('2011-12', 'yyyy-mm') AND
  doc.date_doc < to_date('2012-01', 'yyyy-mm') AND
  lower(pays.libelle) = 'france' AND
  lower(doc.commentaire) LIKE '%saisie%' AND
  lower(doc.commentaire) LIKE '%david%';
```



Performances du like ‘%’



Voir aussi : <http://www.cestpasdur.com/2012/04/01/elasticsearch-vs-mysql-recherche>



C'est quoi un moteur ?

- Un moteur de recherche est composé de :
 - un moteur d'**indexation de documents**
 - un moteur de **recherche sur les index**
- De fait, un moteur de recherche est énormément plus rapide qu'une base de données pour faire des recherches :

c'est son métier !



Your Data, your Search !



Elasticsearch



- Moteur de recherche pour la génération NoSQL
- Basé sur le standard Apache Lucene
- Masque la complexité Java/Lucene à l'aide de services standards HTTP / RESTful / JSON
- Utilisable à partir de n'importe quelle technologie
- Ajoute la couche cloud manquante à Lucene
- C'est un **moteur**, pas une **interface graphique** !



Points clés

- **Simple !** En quelques minutes (Zero Conf), on dispose d'un moteur complet prêt à recevoir nos documents à indexer et à faire des recherches.
- **Efficace !** Il suffit de démarrer des nœuds Elasticsearch pour bénéficier immédiatement de la réplication, de l'équilibrage de charge.
- **Puissant !** Basé sur Lucene, il en parallélise les traitements pour donner des temps de réponse acceptables (en général inférieurs à 100ms)
- **Complet !** Beaucoup de fonctionnalités : analyse et facettes, percolation, rivières, plugins, ...



Ranger ses données

- **Document** : Un objet représentant les données (au sens NoSQL).
Penser "recherche", c'est oublier le SGBDR et penser "Documents"

Un tweet

```
{
  "text": "Bienvenue à la conférence #elasticsearch pour #devoxxfr",
  "created_at": "2012-04-06T20:45:36.000Z",
  "source": "Twitter for iPad",
  "truncated": false,
  "retweet_count": 0,
  "hashtag": [ { "text": "elasticsearch", "start": 27, "end": 40 },
               { "text": "devoxxfr", "start": 47, "end": 55 } ],
  "user": { "id": 51172224, "name": "David Pilato",
            "screen_name": "dadoonet", "location": "France",
            "description": "Soft Architect, Project Manager, Senior Developer.\r\nAt this time, enjoying NoSQL world : CouchDB, ElasticSearch.\r\nDeeJay 4 times a year, just for fun !" }
}
```

- **Type** : Regroupe des documents de même type
- **Index** : Espace logique de stockage des documents dont les types sont fonctionnellement communs



Interagir avec Elasticsearch

API REST : `http://host:port/[index]/[type]/[_action/id]`

Méthodes HTTP : **GET, POST, PUT, DELETE**

Documents

- **curl -XPUT** `http://localhost:9200/twitter/tweet/1`
- **curl -XGET** `http://localhost:9200/twitter/tweet/1`
- **curl -XDELETE** `http://localhost:9200/twitter/tweet/1`

Recherche

- **curl -XGET** `http://localhost:9200/twitter/tweet/_search`
- **curl -XGET** `http://localhost:9200/twitter/_search`
- **curl -XGET** `http://localhost:9200/_search`

Meta données Elasticsearch

- **curl -XGET** `http://localhost:9200/twitter/_status`



Indexons un document

```
$ curl -XPUT localhost:9200/twitter/tweet/1 -d '{
  "text": "Bienvenue à la conférence #elasticsearch pour #devoxxfr",
  "created_at": "2012-04-06T20:45:36.000Z",
  "source": "Twitter for iPad",
  "truncated": false,
  "retweet_count": 0,
  "hashtag": [ { "text": "elasticsearch", "start": 27, "end": 40 },
               { "text": "devoxxfr", "start": 47, "end": 55 } ],
  "user": { "id": 51172224, "name": "David Pilato",
            "screen_name": "dadoonet", "location": "France",
            "description": "Soft Architect, Project Manager, Senior Developer.\r\nAt this time, enjoying
NoSQL world : CouchDB, ElasticSearch.\r\nDeeJay 4 times a year, just for fun !" }
}'
```

```
{
  "ok":true,
  "_index":"twitter",
  "_type":"tweet",
  "_id":"1"
}
```



Cherchons un document

```
$ curl localhost:9200/twitter/tweet/_search?q=elasticsearch
```

```
{
  "took" : 24,
  "timed_out" : false,
  "_shards" : { "total" : 5, "successful" : 5, "failed" : 0 },
  "hits" : {
    "total" : 1,
    "max_score" : 0.227,
    "hits" : [ {
      "_index" : "twitter",
      "_type" : "tweet",
      "_id" : "1",
      "_score" : 0.227, "_source" : {
        "text": "Bienvenue à la conférence #elasticsearch pour #devoxxfr",
        "created_at": "2012-04-06T20:45:36.000Z",
        "source": "Twitter for iPad",
        [...]
      }
    } ]
  }
}
```

Nb de documents
Document
Coordonnées

Pertinence



Les résultats de recherche

- Elasticsearch ne renvoie que les 10 premiers résultats (même sur plusieurs millions)
- Elasticsearch permet ensuite de se "balader" dans les résultats

```
$ curl "localhost:9200/twitter/tweet/_search?q=elasticsearch&from=10&size=10"
```

- La pertinence est calculée suivant le nombre d'occurrences plus ou moins exactes de chaque terme dans un document

```
$ curl "localhost:9200/twitter/tweet/_search?q=elasticsearch&explain=true"
```



Quelques types de recherche

Langage **QueryDSL** pour les recherches avancées

Type	Description
<u>Match All</u>	Recherche tout le contenu (pratique avec des filtres)
<u>QueryString</u>	Recherche avec analyse, jokers (syntaxe Lucene possible* +, -, FROM, TO, ^)
<u>Term</u>	Recherche d'un terme sans analyse préalable
<u>Text</u>	Recherche d'un texte avec analyse (par défaut OR sur chaque token)
<u>Wildcard</u>	Recherche avec joker (*, ?)
<u>Bool</u>	Recherche multi-critères (MUST, MUST NOT, SHOULD)
<u>Range</u>	Recherche intervalle (>, >=, <, <=)
<u>Prefix</u>	Utile pour faire de l'autocomplétion
<u>Filtered</u>	Filtrage (couplage de filtres et de queries)
<u>Fuzzy like this</u>	Permet des recherches par vraisemblance de termes
<u>More like this</u>	Permet de trouver des documents avec un minimum de termes

* http://lucene.apache.org/core/old_versioned_docs/versions/3_5_0/queryparsersyntax.html

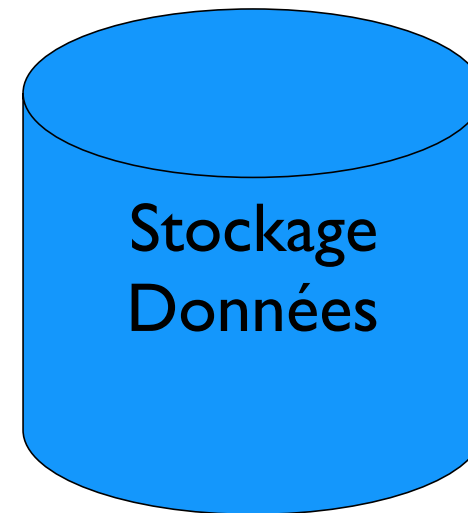


Ou "La vie est un long fleuve tranquille !"

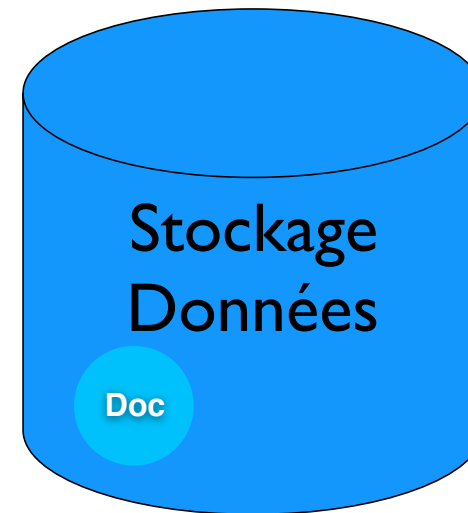
LA COLLECTE AUTOMATIQUE DE DONNÉES



La collecte



La collecte



La collecte



La collecte



La collecte



La collecte



Rivers

- CouchDB River
- MongoDB River
- Wikipedia River
- Twitter River
- RabbitMQ River
- RSS River
- Dick Rivers



La puissance des facettes ! Faites parler vos données en les regardant sous différentes facettes !

ANALYSE DES RÉSULTATS (EN TEMPS QUASI RÉEL)



Les facettes

Des tweets

ID	Username	Date	Hashtags
1	dadoonet	2012-04-18	1
2	devovxfr	2012-04-18	5
3	elasticsearchfr	2012-04-18	2
4	dadoonet	2012-04-18	2
5	devovxfr	2012-04-18	6
6	elasticsearchfr	2012-04-19	3
7	dadoonet	2012-04-19	3
8	devovxfr	2012-04-19	7
9	elasticsearchfr	2012-04-20	4

Facette "Term"

	Username	Date	Hashtags
	dadoonet	2012-04-18	1
	devoxxfr	2012-04-18	5
	elasticsearchfr	2012-04-18	
	dadoonet	2012-04-18	
	devoxxfr	2012-04-18	
	elasticsearchfr	2012-04-19	
	dadoonet	2012-04-19	3
	devoxxfr	2012-04-19	7
	elasticsearchfr	2012-04-20	4

Username	Count
dadoonet	3
devoxxfr	3
elasticsearchfr	3

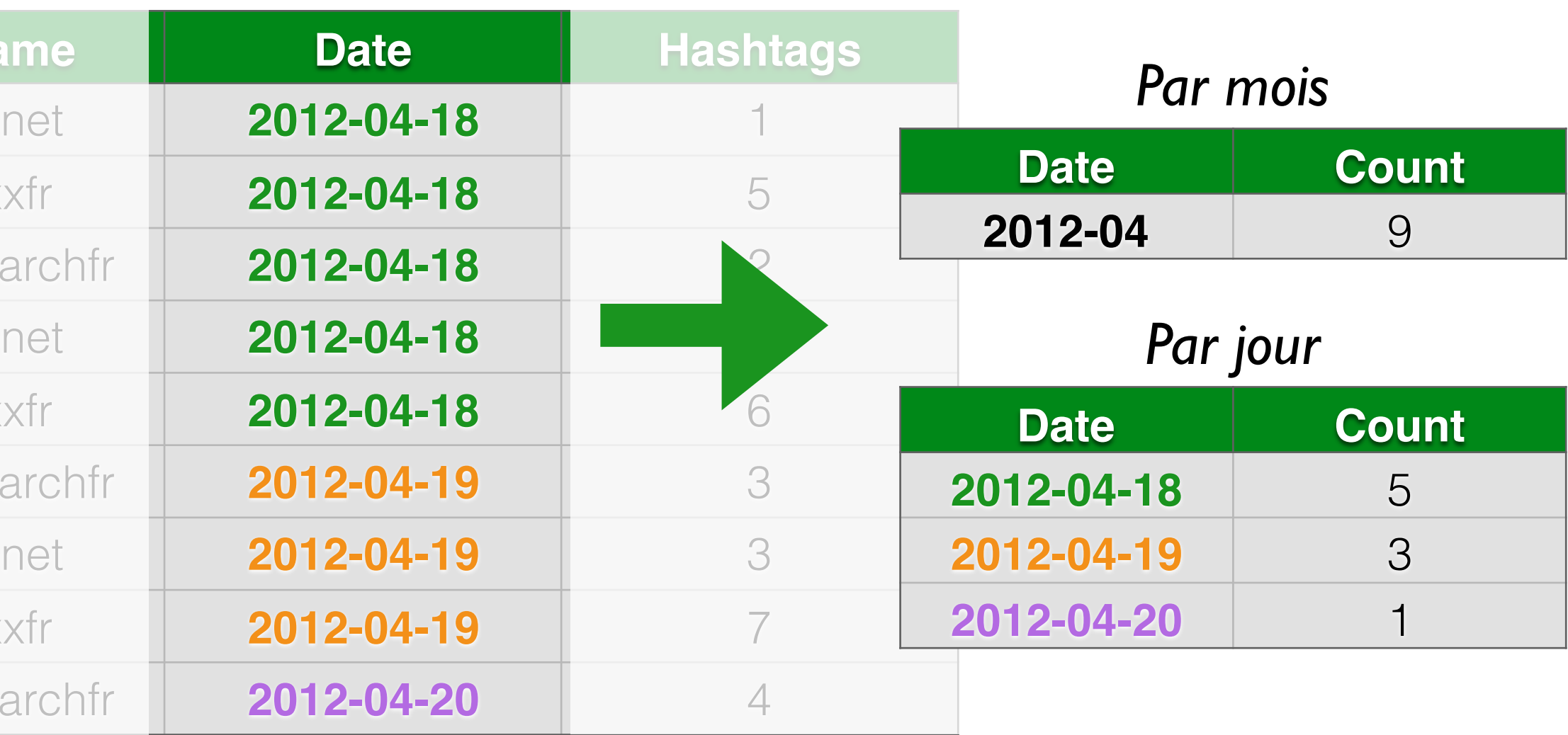
Facette "Term"

ID	Username
1	dadoonet
2	devoxxfr
3	elasticsearchfr
4	dadoonet
5	devoxxfr
6	elasticsearchfr
7	dadoonet
8	devoxxfr
9	elasticsearchfr

```
"facets" : {
  "users" : { "terms" : { "field" : "username" } }
}

"facets" : {
  "users" : {
    "_type" : "terms",
    "missing" : 0,
    "total": 9,
    "other": 0,
    "terms" : [
      { "term" : "dadoonet", "count" : 3 },
      { "term" : "devoxxfr", "count" : 3 },
      { "term" : "elasticsearchfr", "count" : 3 }
    ]
  }
}
```

Facette "Date Histogram"



Facette "Date Histogram"

ame	Date
net	2012-04-18
xfr	2012-04-18
archfr	2012-04-18
net	2012-04-18
xfr	2012-04-18
archfr	2012-04-19
net	2012-04-19
xfr	2012-04-19
archfr	2012-04-20

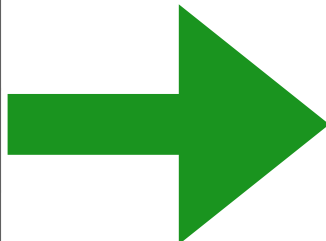
```
"facets" : {
  "perday" : {
    "date_histogram" : {
      "field" : "date",
      "interval" : "day"
    }
  }
}
```

```
"facets" : {
  "perday" : {
    "_type" : "date_histogram",
    "entries": [
      { "time": 1334700000000, "count": 5 },
      { "time": 1334786400000, "count": 3 },
      { "time": 1334872800000, "count": 1 }
    ]
  }
}
```



Facette "Ranges"

Hashtags
1
5
2
2
6
3
3
7
4



Ranges	Count	Min	Max	Moy	Total
$x < 3$	3	1	2	1.667	5
$3 \leq x < 5$	3	3	4	3.333	10
$x \geq 5$	3	5	7	6	18



Facette "Ranges"

```
"facets" : { "hashtags" : {  
  "range" : { "field" : "hashtags",  
  "ranges" : [  
    { "to" : 3 }, { "from" : 3, "to" : 5 }, { "from" : 5 }  
  ] } } }
```

```
"facets" : {  
  "hashtags" : {  
    "_type" : "range",  
    "ranges" : [  
      { "to": 3,  
        "count": 3, "min": 1, "max": 2, "total": 5, "mean": 1.667  
      },  
      { "from":3, "to" : 5,  
        "count": 3, "min": 3, "max": 4, "total": 10, "mean": 3.333  
      },  
      { "from":5,  
        "count": 3, "min": 5, "max": 7, "total": 18, "mean": 6  
      }  
    ] } }
```

Hashtags
1
5
2
2
6
3
3
7
4



Usage "site marchand"

Recherche avancée

Enlever les filtres (0 actif)
Nouvelle recherche

Ancienneté

Tout

≤ 2 ans (~ 38 600)

≤ 1 an (~ 21 000)

≤ 6 mois (~ 10 100)

≤ 1 mois (~ 1 900)

≤ 1 semaine (~ 140)

Collection

Collection Standard

Collection Infinite

Collection abonnement

Toutes les collections

Type

Photos (~ 30 000)

Illustrations (~ 18 000)

Vecteurs (~ 16 800)

Vidéos (~ 950)

Tous types

Orientation

Paysage (~ 35 400)

Portrait (~ 13 500)

Carré (~ 16 800)

Tous

Options

Tailles

L (~ 64 700)

XL (~ 56 300)

XXL (~ 34 800)

XXXL (~ 56 300)

65 900 résultats

Classement par Pertinence

Par page

Ajouter au Panier

#36840357

36840357

Ajouter au Panier

6901686

Ajouter au Panier

31936984

Ajouter au Panier

22365572

Ajouter au Panier

32863249

Ajouter au Panier

24850194

Ajouter au Panier

18051275

Ajouter au Panier

14397379

Ajouter au Panier

29729704

Ajouter au Panier

Ancienneté

Tout

≤ 2 ans (~ 38 600)

≤ 1 an (~ 21 000)

≤ 6 mois (~ 10 100)

≤ 1 mois (~ 1 900)

≤ 1 semaine (~ 140)

Collection

Collection Standard

Collection Infinite

Collection abonnement

Toutes les collections

Type

Photos (~ 30 000)

Illustrations (~ 18 000)

Vecteurs (~ 16 800)

Vidéos (~ 950)

Tous types

Orientation

Paysage (~ 35 400)

Portrait (~ 13 500)

Carré (~ 16 800)

Tous

Options

Tailles

L (~ 64 700)

XL (~ 56 300)

XXL (~ 34 800)

XXXL (~ 56 300)

Connexion

Créer un compte

Mon Panier (0)

Ranges

Term

Term

Ranges

La navigation par facettes

EORI :
SIRET :
Tél :
Fax :
Mél :

Critère fixe

Domaine

Etat

non_conforme : 6
Conformité

documentaire : 5
Type Contrôle

TOP 10 Bureaux

Infractions

TOP 10 Espèces

us : 7
TOP 10 Origines

Taxes

Historique

03/2012 : 3
Historique

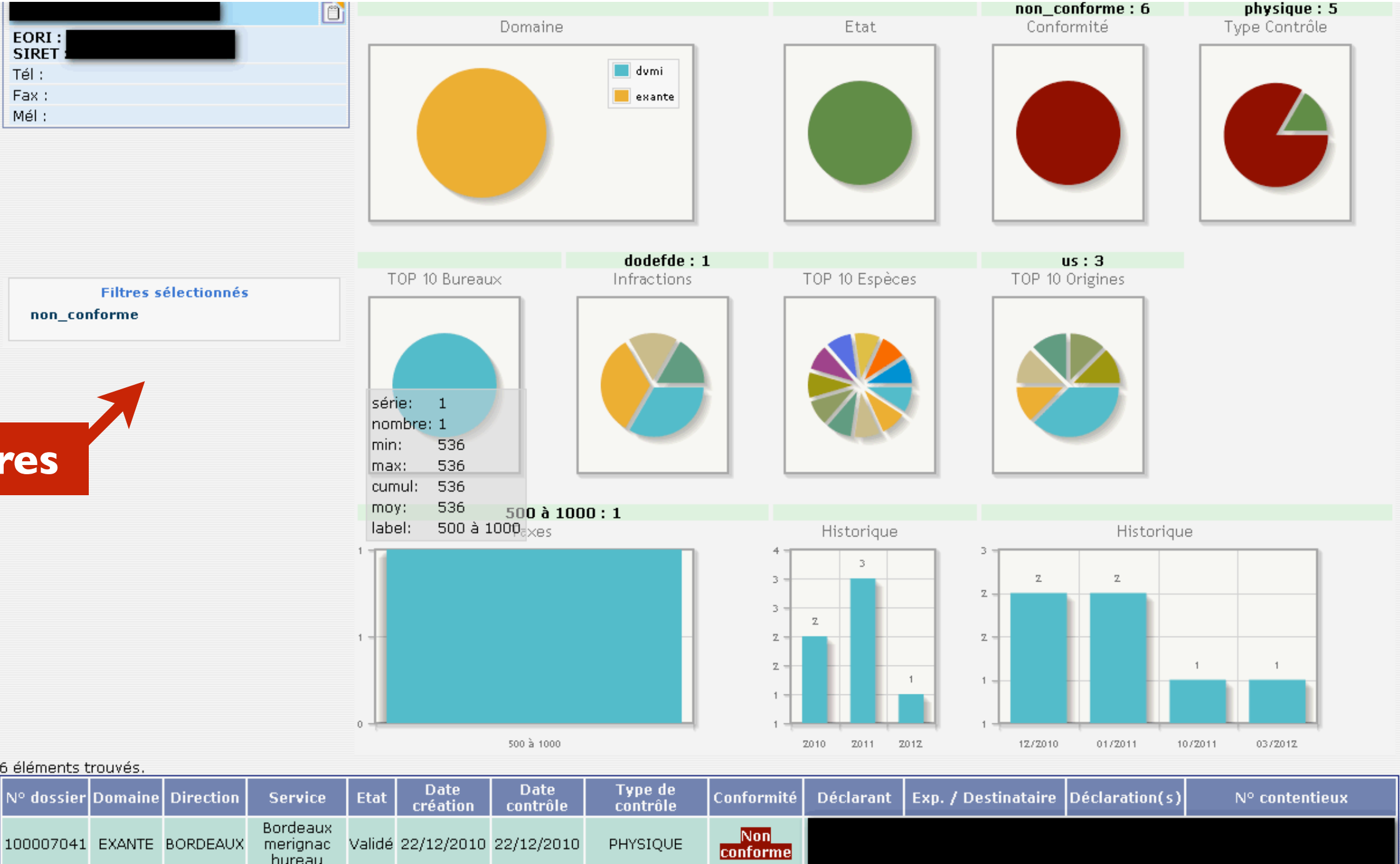
Résultats

Ranges

Date histogram

N° dossier	Domaine	Direction	Service	Etat	Date création	Date contrôle	Type de contrôle	Conformité	Déclarant	Exp. / Destinataire	Déclaration(s)	N° contentieux
100007041	EXANTE	BORDEAUX	Bordeaux merignac bureau	Validé	22/12/2010	22/12/2010	PHYSIQUE	Non conforme				

La navigation par facettes

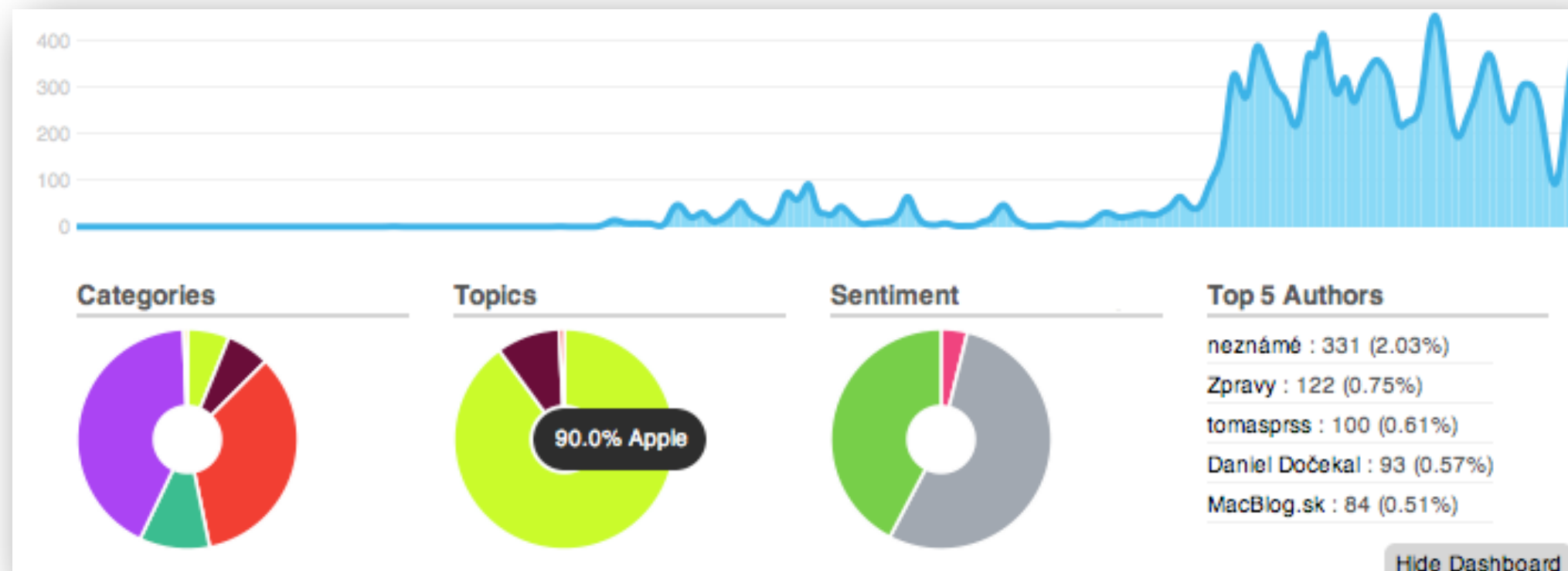


Critères



Analyse temps-réel des données

- Faire un matchAll sur l'ensemble des données
- Actualiser toutes les x secondes
- Indexer en même temps les nouvelles données



Date histogram

Term

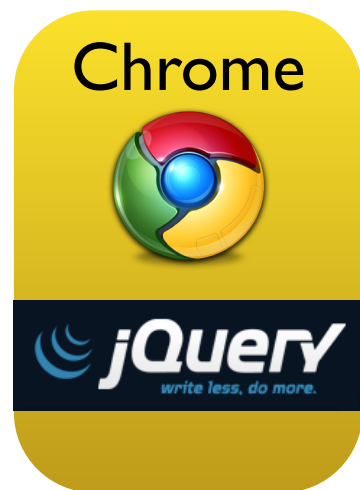


Avons-nous fait du bruit ?

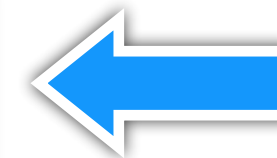
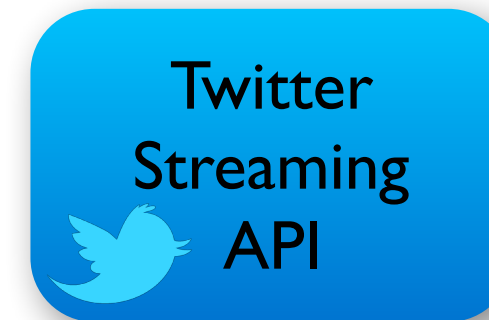
DÉMONSTRATION



Démonstration : architecture



Twitter
River



```
$ curl -XPUT localhost:9200/_river/twitter/_meta -d '
{
  "type" : "twitter",
  "twitter" : {
    "user" : "twitter_user",
    "password" : "twitter_passowrd",
    "filter" : { "tracks" : ["devovxxfr"] }
  }
}'
```



Un peu plus de technique : partitions / répliquions / scalabilité

ARCHITECTURE



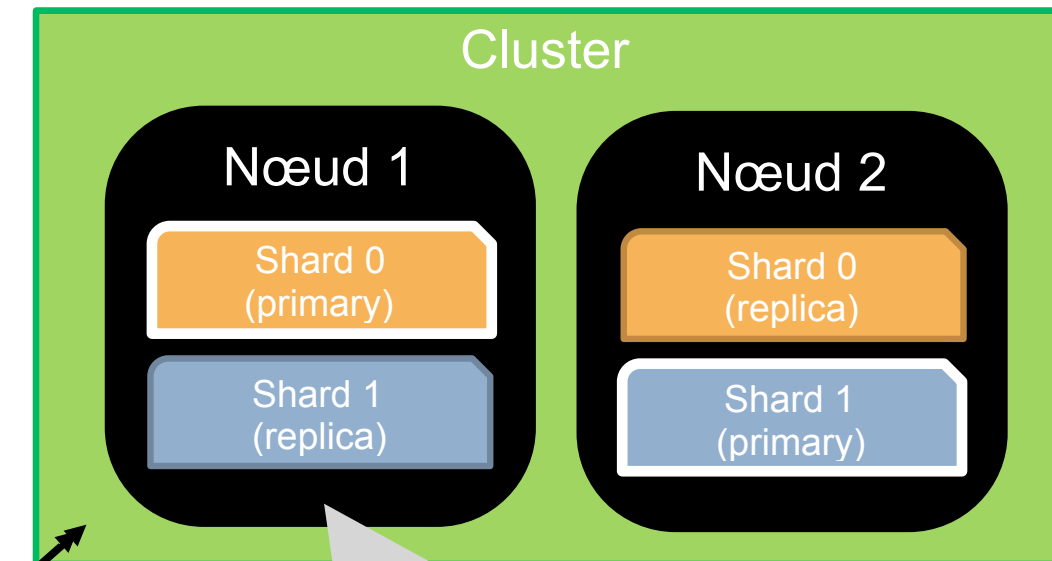
Lexique

- **Nœud (node)** : Une instance d'Elasticsearch (~ machine ?)
- **Cluster** : Un ensemble de nœuds
- **Partition (shard)** : permet de découper un index en plusieurs parties pour y distribuer les documents
- **Réplication (replica)** : recopie d'une partition en une ou plusieurs copies dans l'ensemble du cluster
- **Partition primaire (primary shard)** : partition élue "principale" dans l'ensemble du cluster. C'est là que se fait l'indexation par Lucene. Il n'y en a qu'une seule par shard dans l'ensemble du cluster.
- **Partition secondaire (secondary shard)** : partitions secondaires stockant les replicas des partitions primaires.



Créons un index

```
$ curl -XPUT localhost:9200/twitter -d '{  
  "index" : {  
    "number_of_shards" : 2,  
    "number_of_replicas" : 1  
  }  
'
```

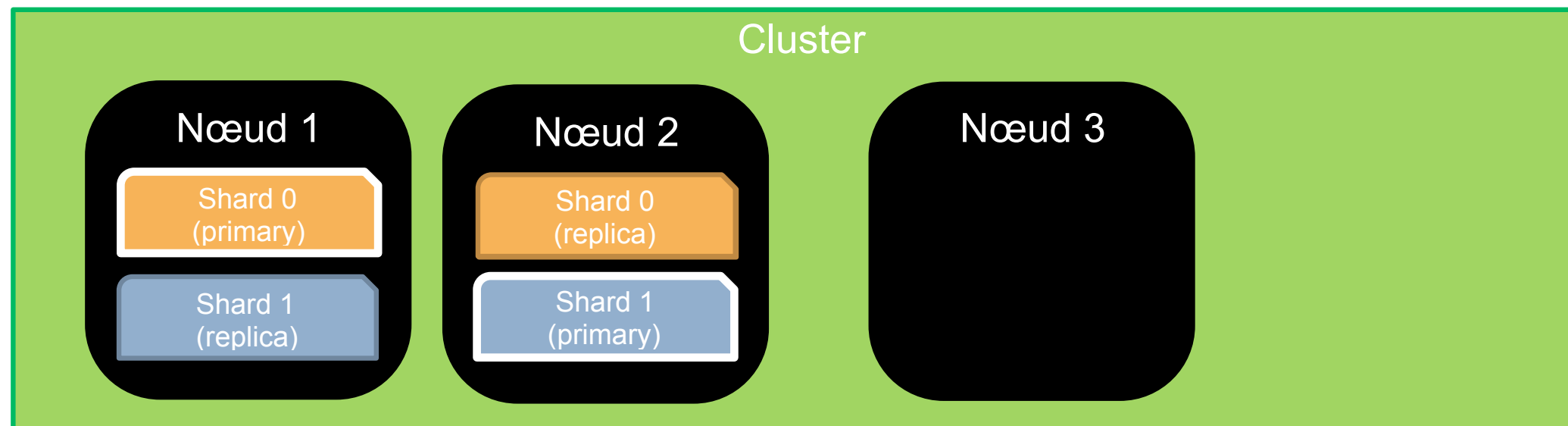


réplication respectée

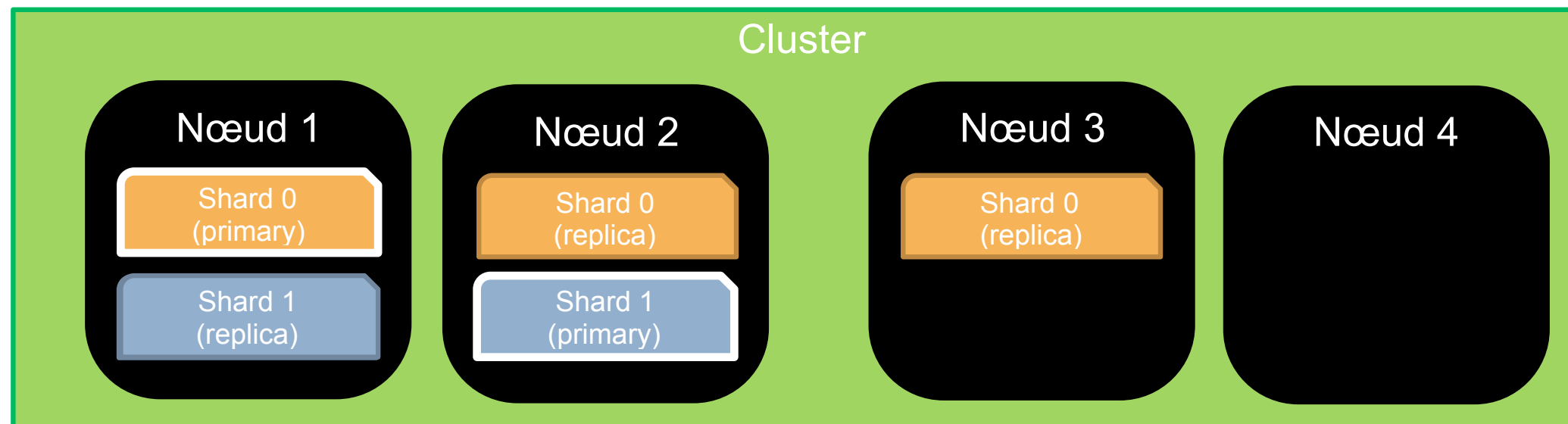
Client
CURL



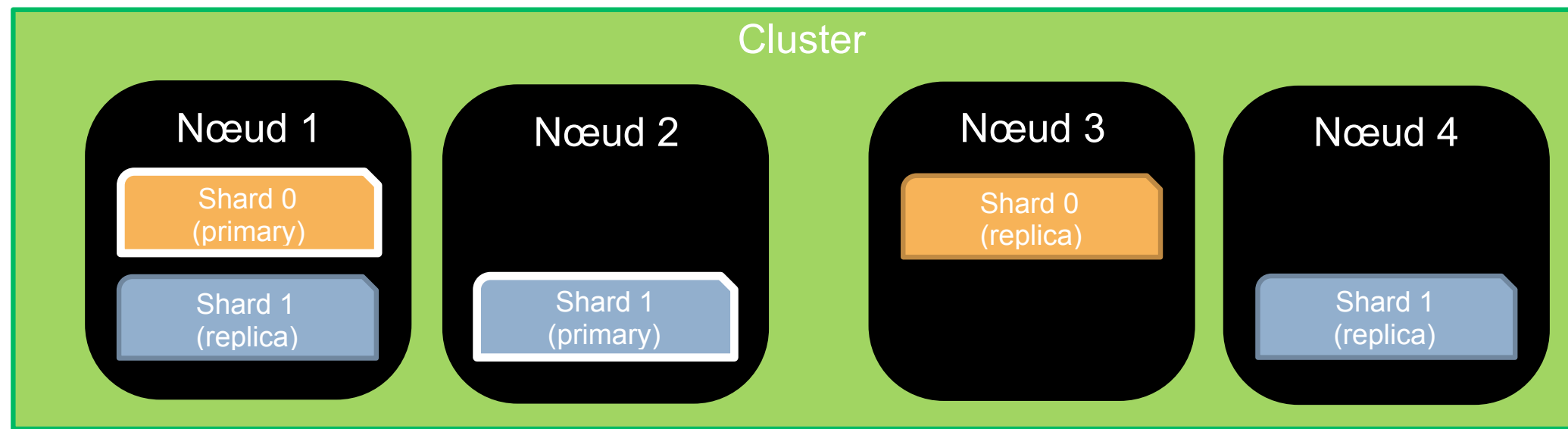
Réallocation dynamique



Réallocation dynamique



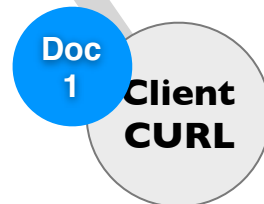
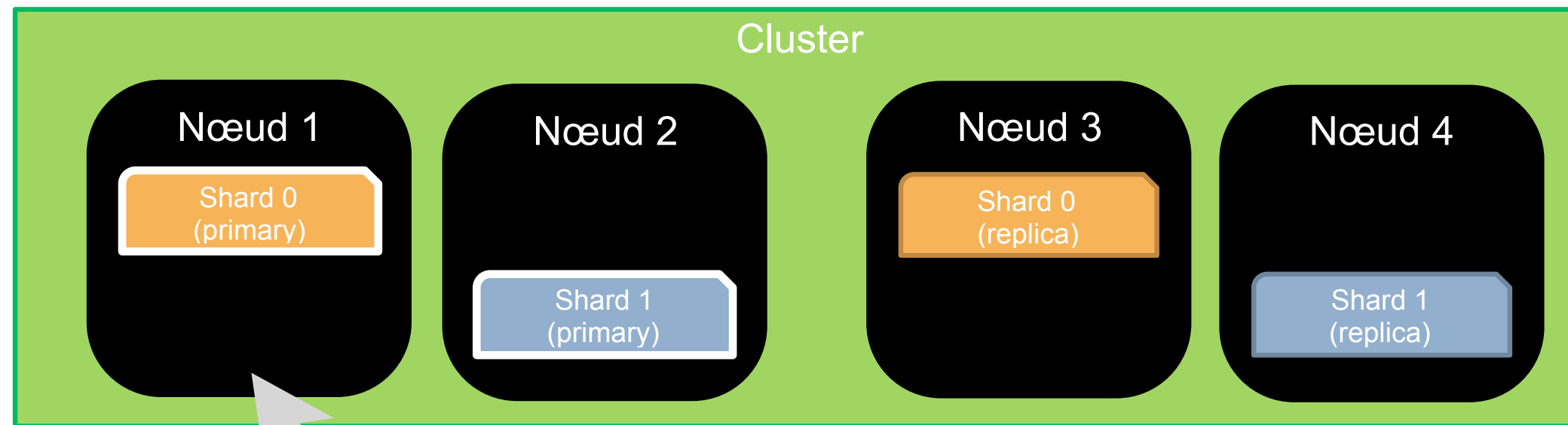
Réallocation dynamique



Le **tuning**, c'est trouver le bon équilibre entre le nombre de **nodes**, **shards** et **replicas** !



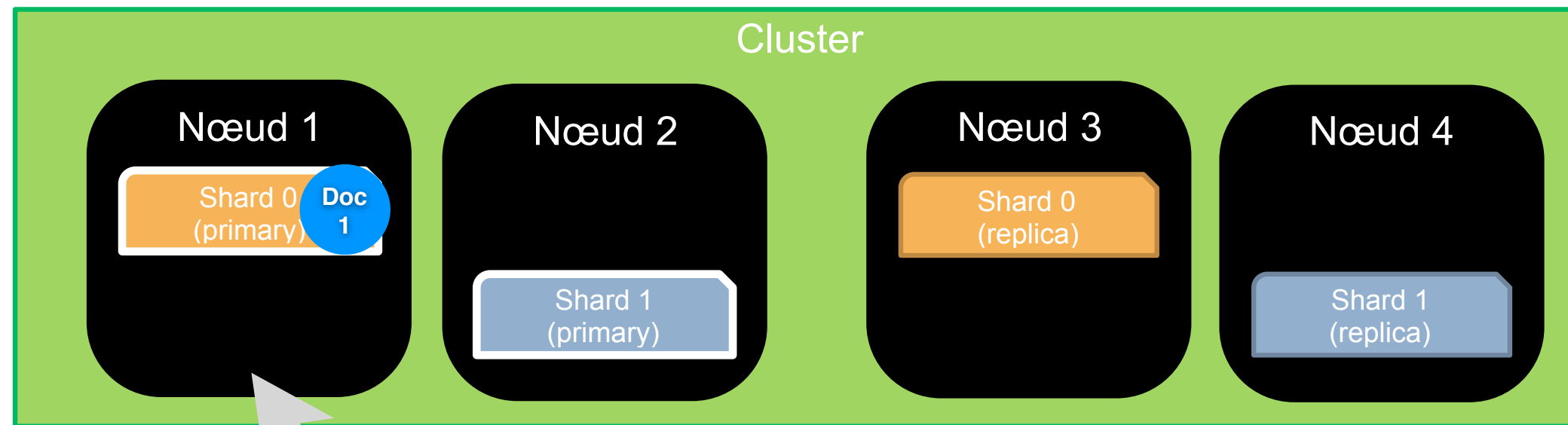
Indexons un document



```
$ curl -XPUT localhost:9200/twitter/tweet/1 -d '{
  "text": "Bienvenue à la conférence #elasticsearch pour #devoxxfr",
  "created_at": "2012-04-06T20:45:36.000Z",
  "source": "Twitter for iPad",
  ...
}'
```



Indexons un document

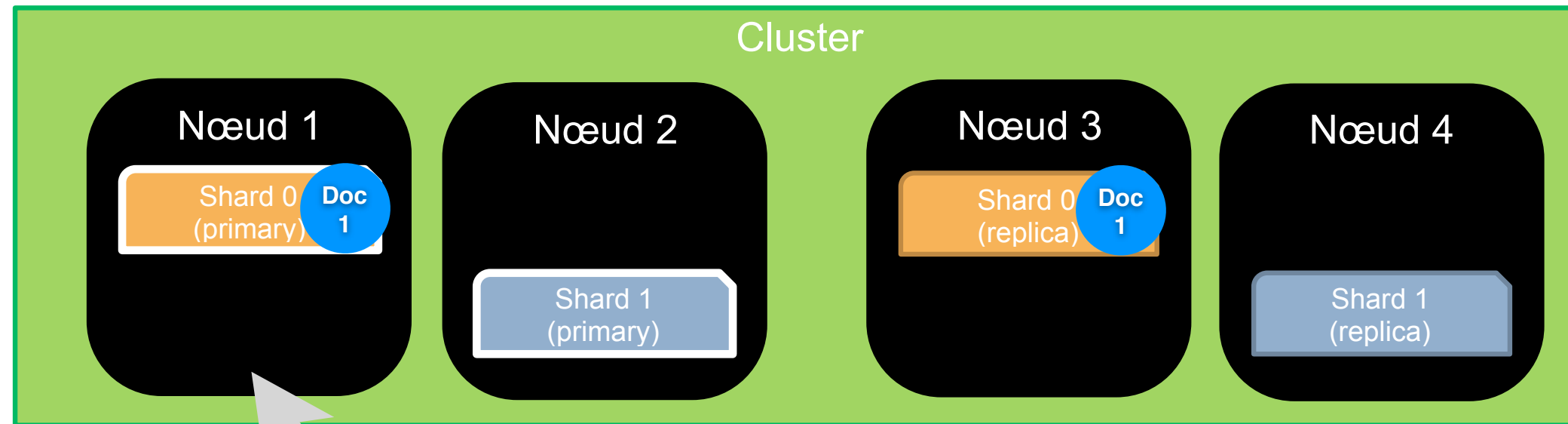


Client
CURL

```
$ curl -XPUT localhost:9200/twitter/tweet/1 -d '{
  "text": "Bienvenue à la conférence #elasticsearch pour #devoxxfr",
  "created_at": "2012-04-06T20:45:36.000Z",
  "source": "Twitter for iPad",
  ...
}'
```



Indexons un document

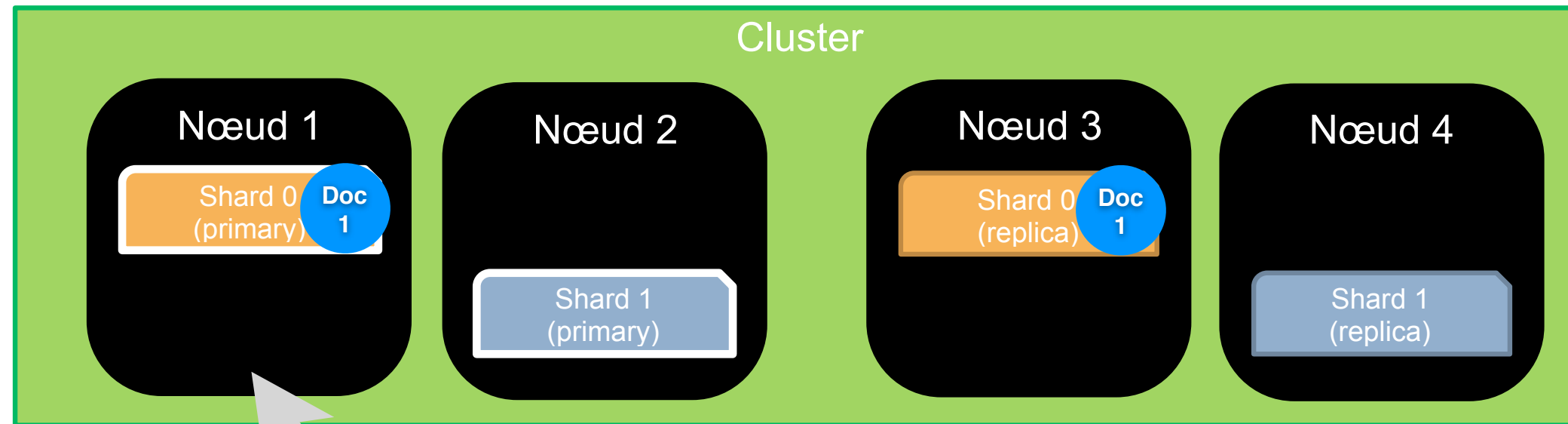


Client
CURL

```
$ curl -XPUT localhost:9200/twitter/tweet/1 -d '{
  "text": "Bienvenue à la conférence #elasticsearch pour #devoxxfr",
  "created_at": "2012-04-06T20:45:36.000Z",
  "source": "Twitter for iPad",
  ...
}'
```



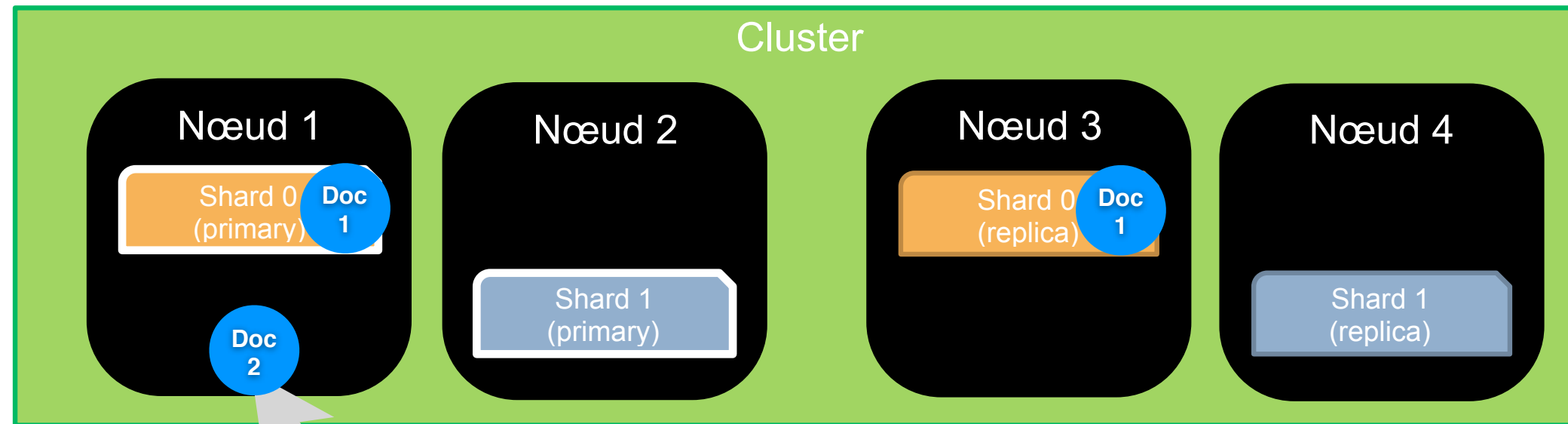
Indexons un 2^{ème} document



```
$ curl -XPUT localhost:9200/twitter/tweet/2 -d '{
  "text": "Je fais du bruit pour #elasticsearch à #devoxxfr",
  "created_at": "2012-04-06T21:12:52.000Z",
  "source": "Twitter for iPad",
  ...
}'
```



Indexons un 2^{ème} document

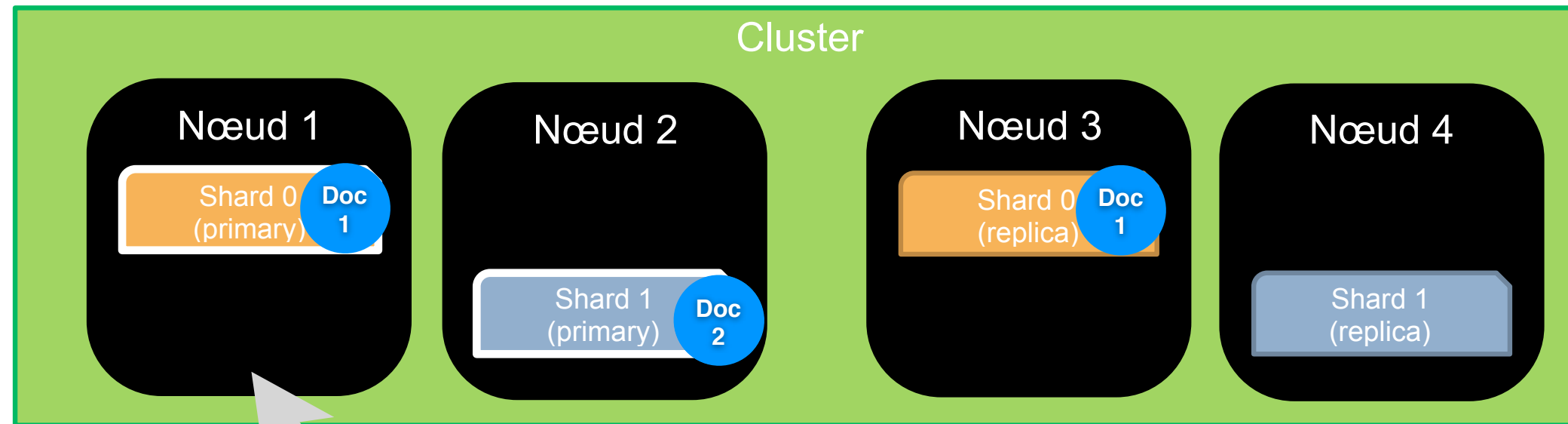


Client
CURL

```
$ curl -XPUT localhost:9200/twitter/tweet/2 -d '{
  "text": "Je fais du bruit pour #elasticsearch à #devoxxfr",
  "created_at": "2012-04-06T21:12:52.000Z",
  "source": "Twitter for iPad",
  ...
}'
```



Indexons un 2^{ème} document

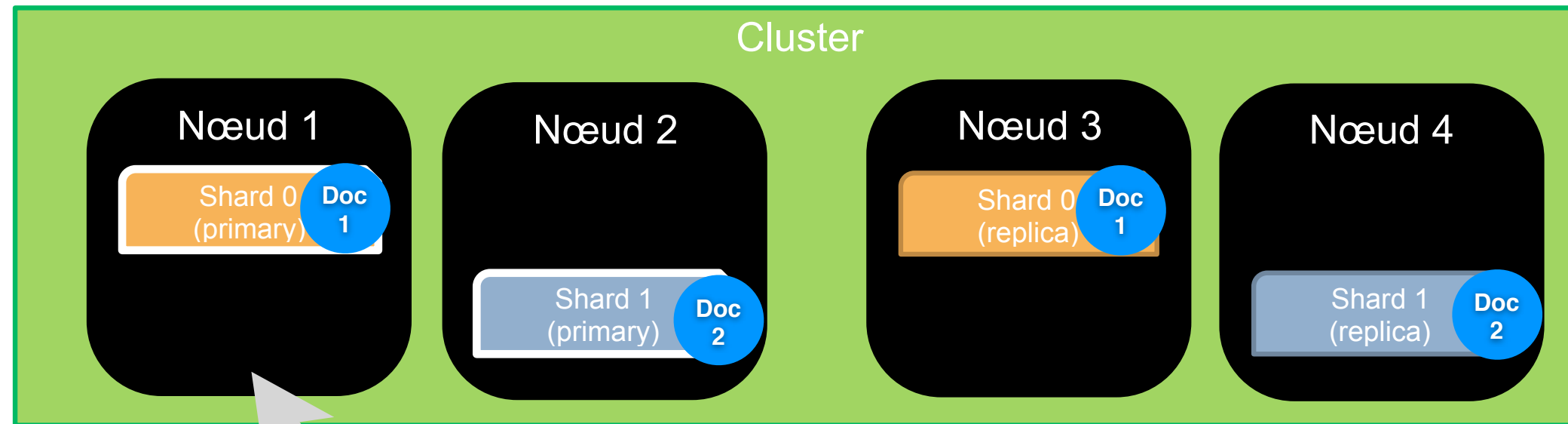


Client
CURL

```
$ curl -XPUT localhost:9200/twitter/tweet/2 -d '{
  "text": "Je fais du bruit pour #elasticsearch à #devoxxfr",
  "created_at": "2012-04-06T21:12:52.000Z",
  "source": "Twitter for iPad",
  ...
}'
```



Indexons un 2^{ème} document

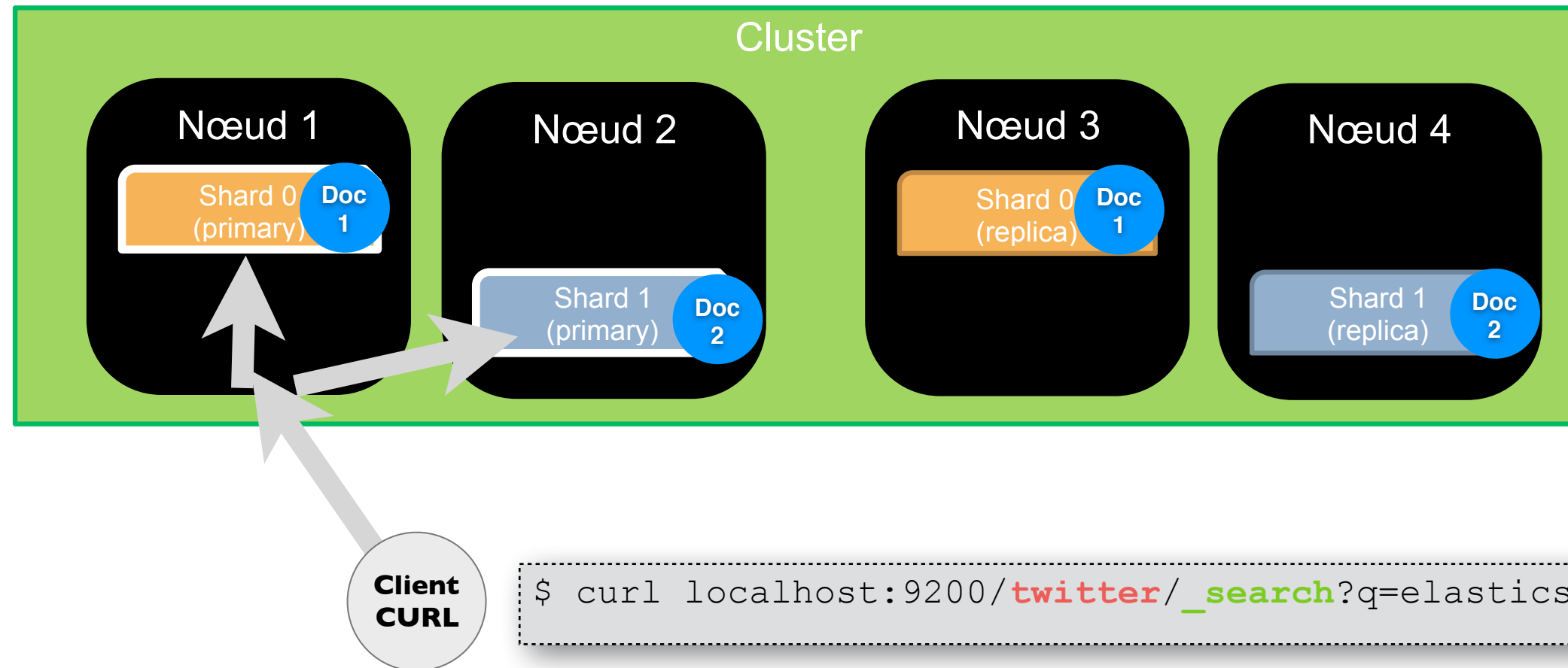


Client
CURL

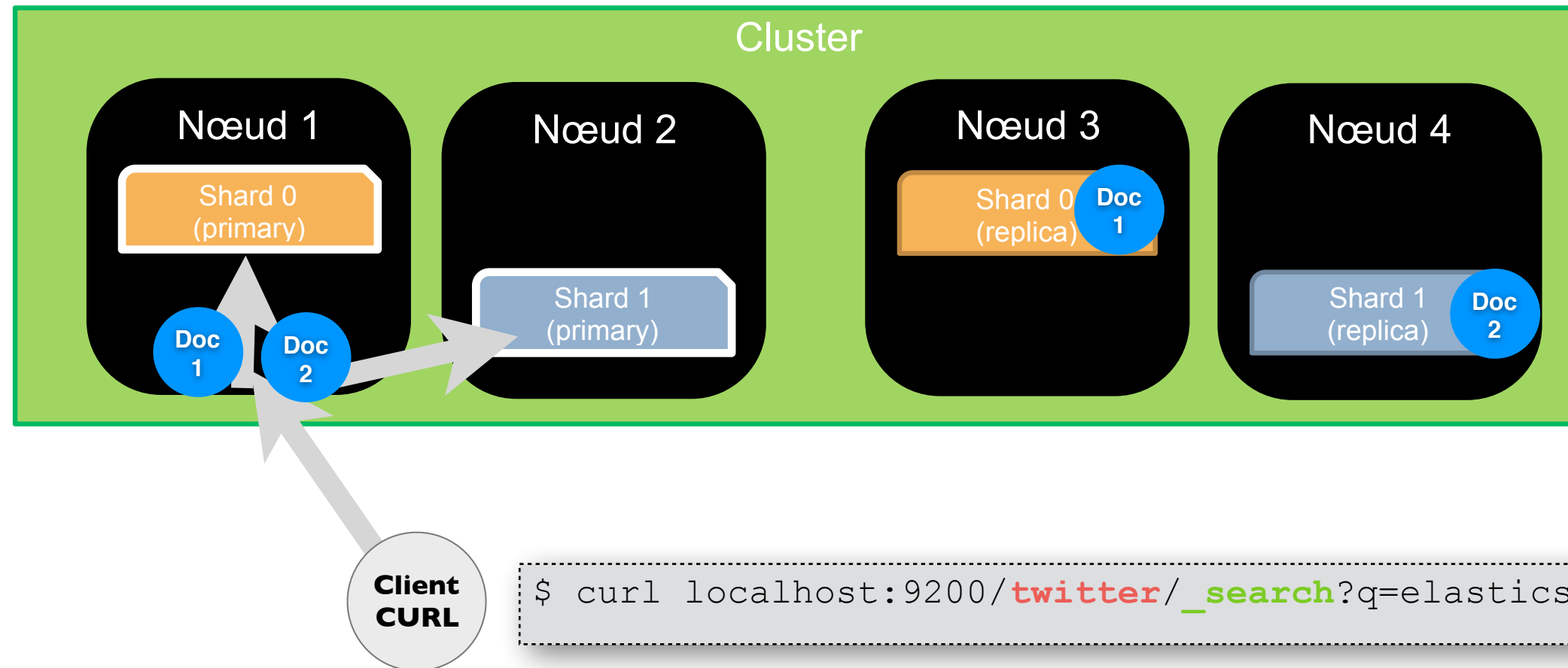
```
$ curl -XPUT localhost:9200/twitter/tweet/2 -d '{
  "text": "Je fais du bruit pour #elasticsearch à #devoxxfr",
  "created_at": "2012-04-06T21:12:52.000Z",
  "source": "Twitter for iPad",
  ...
}'
```



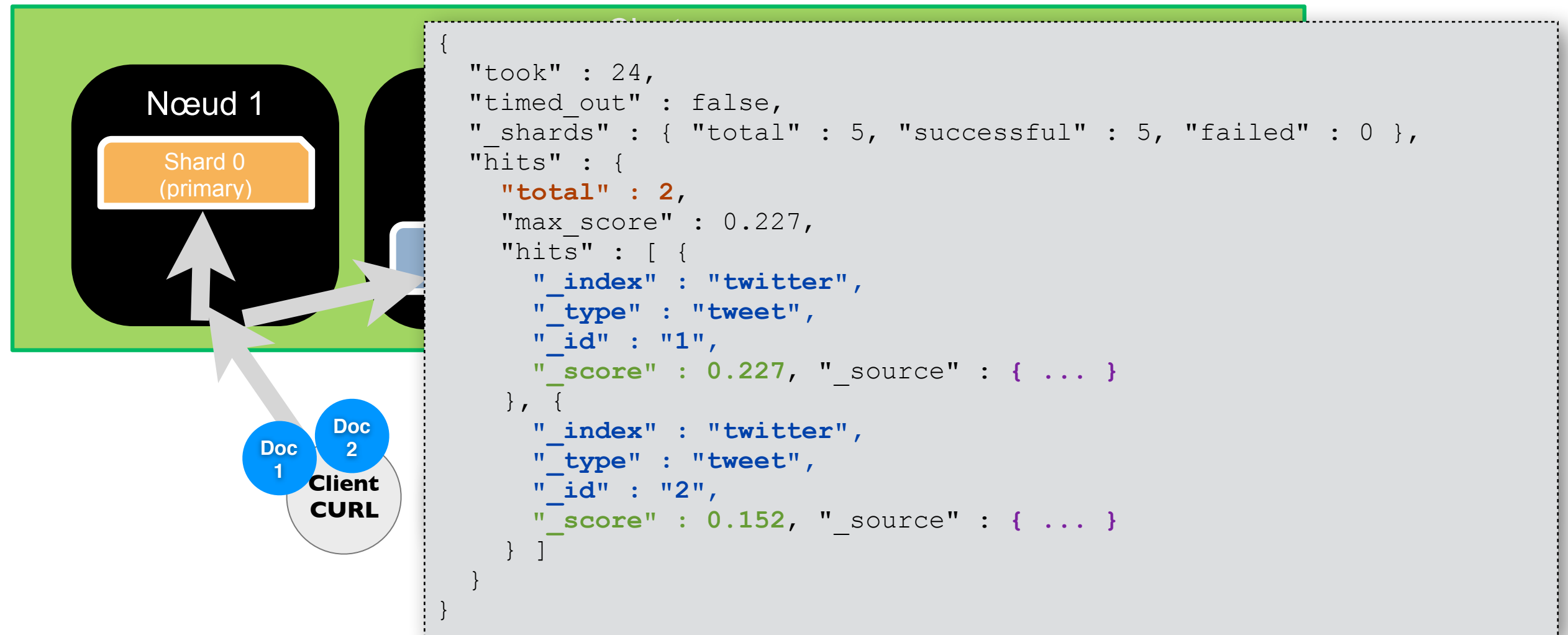
Cherchons



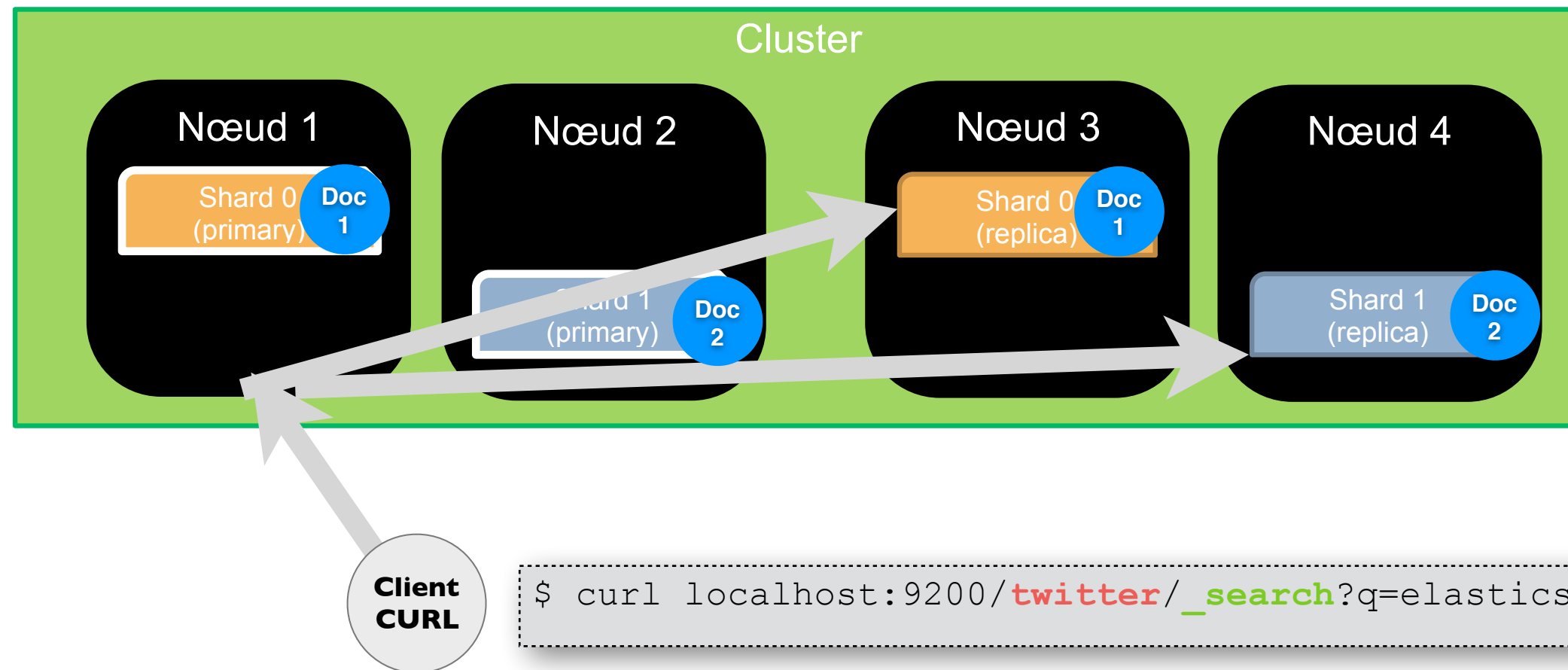
Cherchons



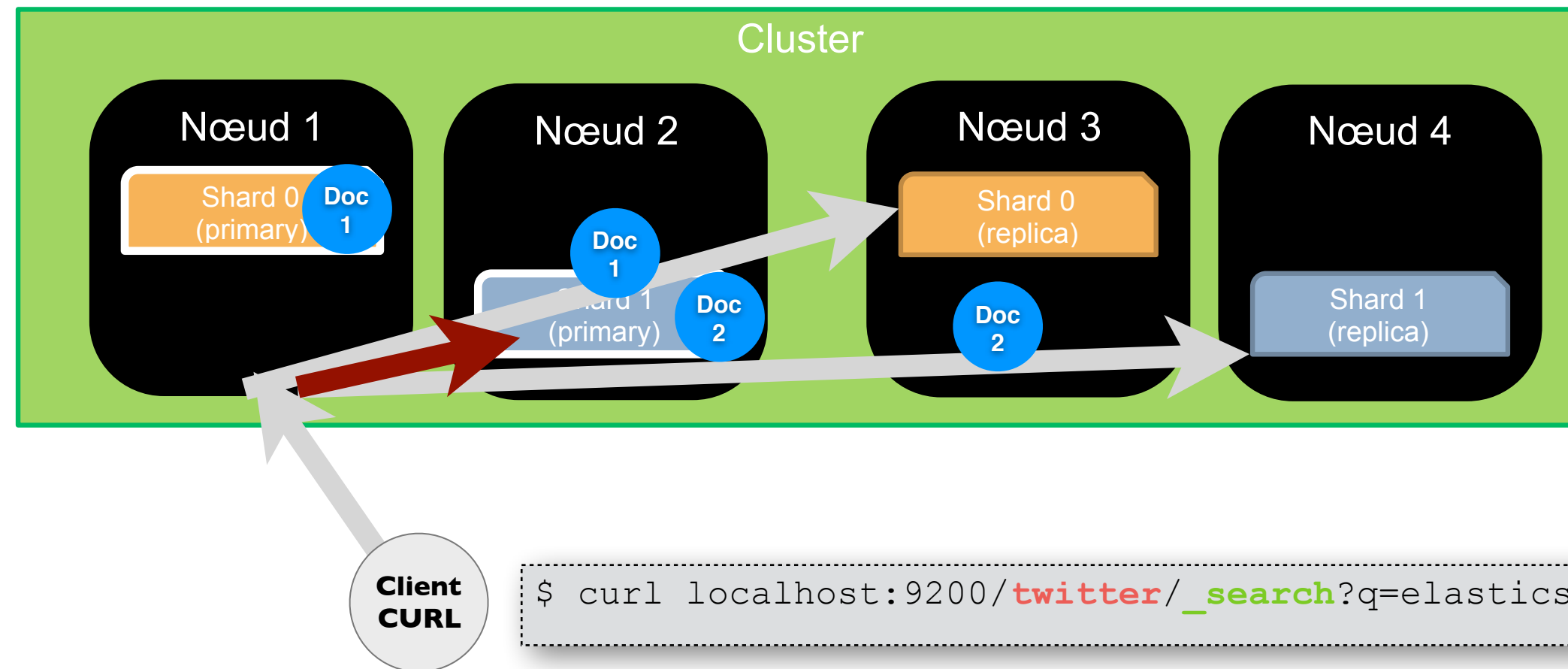
Cherchons



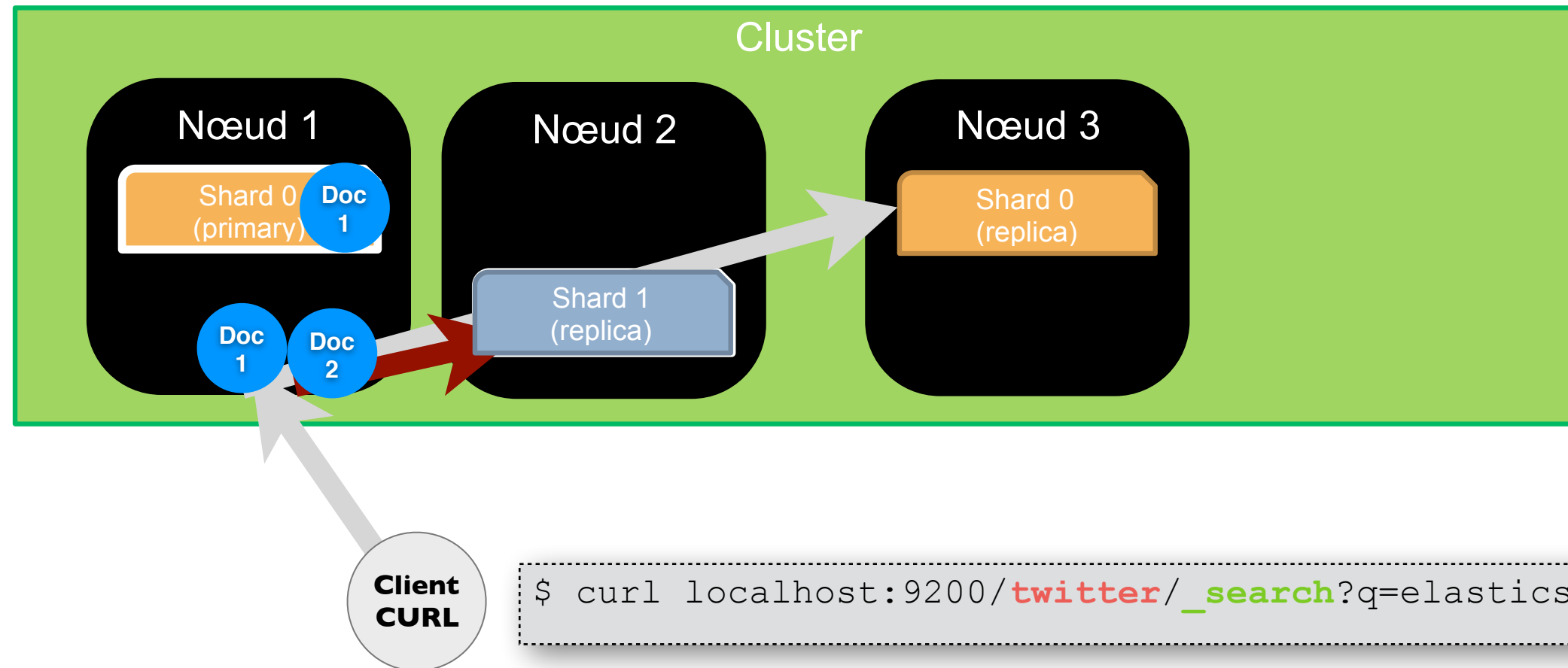
Cherchons encore



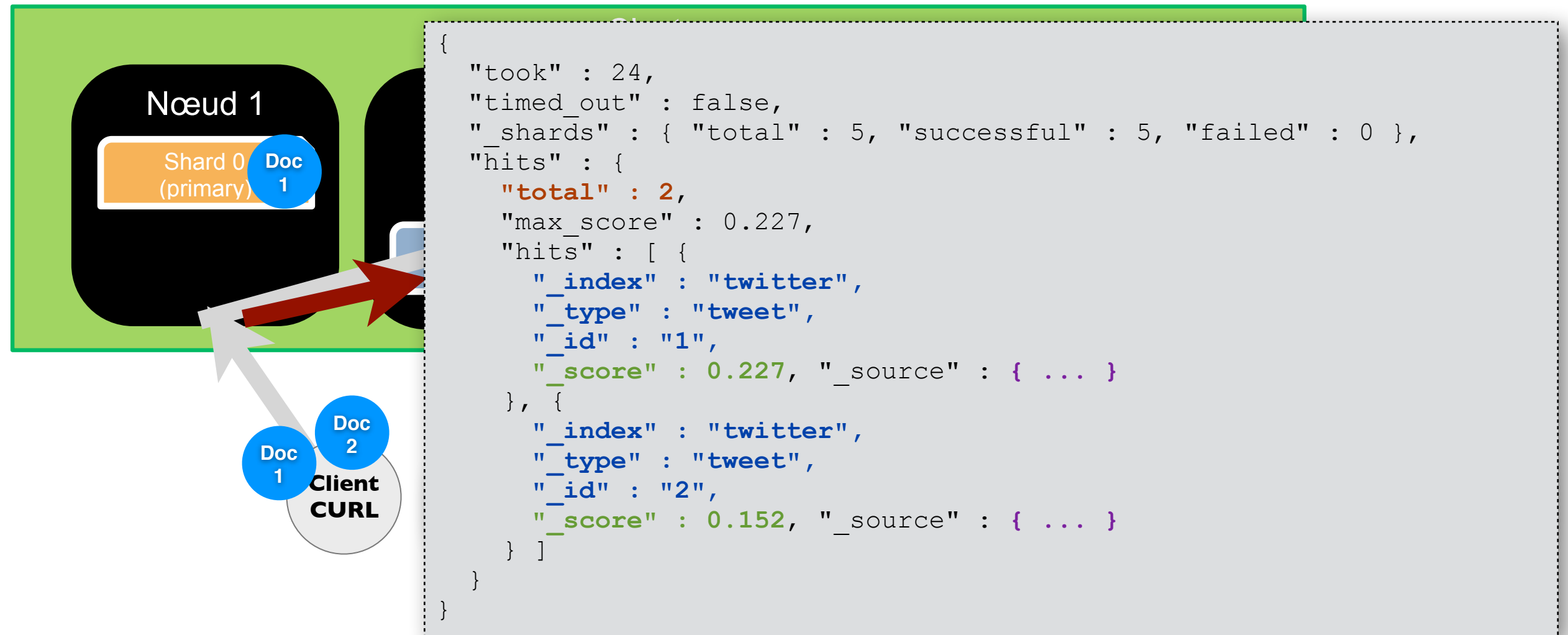
Cherchons encore



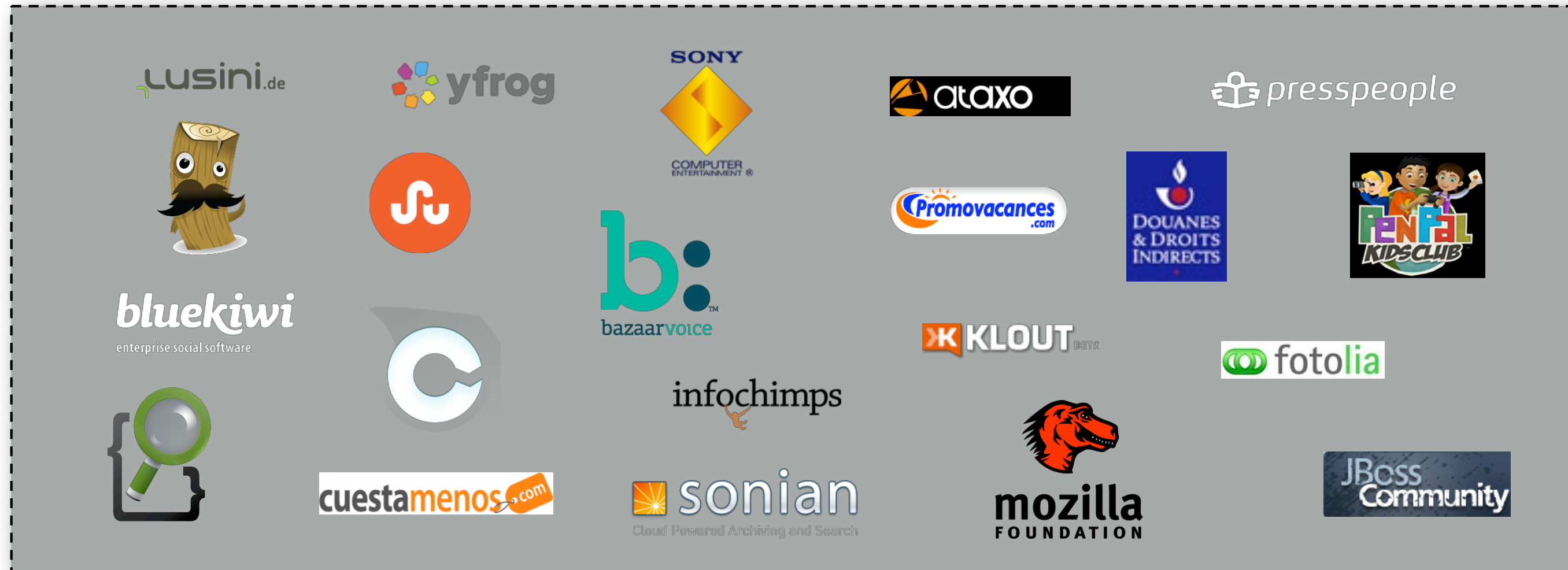
Cherchons encore



Cherchons encore



Elasticsearch : la communauté



~50 contributeurs directs au projet (+ de 2000 watchers et + de 280 forks)



Rejoignez le mouvement !

@ElasticsearchFR



Posez aussi vos questions sur elasticsearch-fr@googlegroups.com

QUESTIONS ?

