

Search

**Under the Hood**

# Agenda

- ⌚ basic introduction to Elasticsearch
- ⌚ the pre-work needed to prepare the search index
- ⌚ capabilities and limitations of Elasticsearch

What is Elasticsearch ?

# Document Storage

- stores data as JSON
- row = document
- table = index
- column = field

```
PUT candidates_demo/_doc/CN_10003
{
  "fullName": "Nero Newbie",
  "summary": "A newbie in the IT Industry",
  "location": [40.71, 74.00],
  "currentFunction": "FN_100001",
  "likelyToJump": false,
  "yearsOfExperience": 3,
  "jobHistories": [
    {
      "jobTitle": "Jr. Software Engineer",
      "industry": "IND_100001",
      "industryName": "IT Engineering",
      "yearsOfExperience": 2,
      "current": true
    },
    {
      "jobTitle": "Intern",
      "industry": "IND_100001",
      "industryName": "IT Engineering",
      "yearsOfExperience": 1,
      "current": false
    }
  ]
}
```

# Search Engine

- analyzes/tokenize the document during insert
- uses this index during search to optimize the results

```
141 PUT candidates_demo/_doc/CN_10003
142 {
143   "fullName": "Nero Newbie",
144   "summary": "A newbie in the IT Industry",
145   "location": [40.71, 74.00],
146   "currentFunction": "FN_100001",
147   "likelyToJump": false,
148   "yearsOfExperience": 3,
149   "jobHistories": [
150     {
151       "jobTitle": "Jr. Software Engineer",
152       "industry": "IND_100001",
153       "industryName": "IT Engineering",
154       "yearsOfExperience": 2,
155       "current": true
156     },
157     {
158       "jobTitle": "Intern",
159       "industry": "IND_100001",
160       "industryName": "IT Engineering",
161       "yearsOfExperience": 1,
162       "current": false
163     }
164   ]
165 }
```

summary

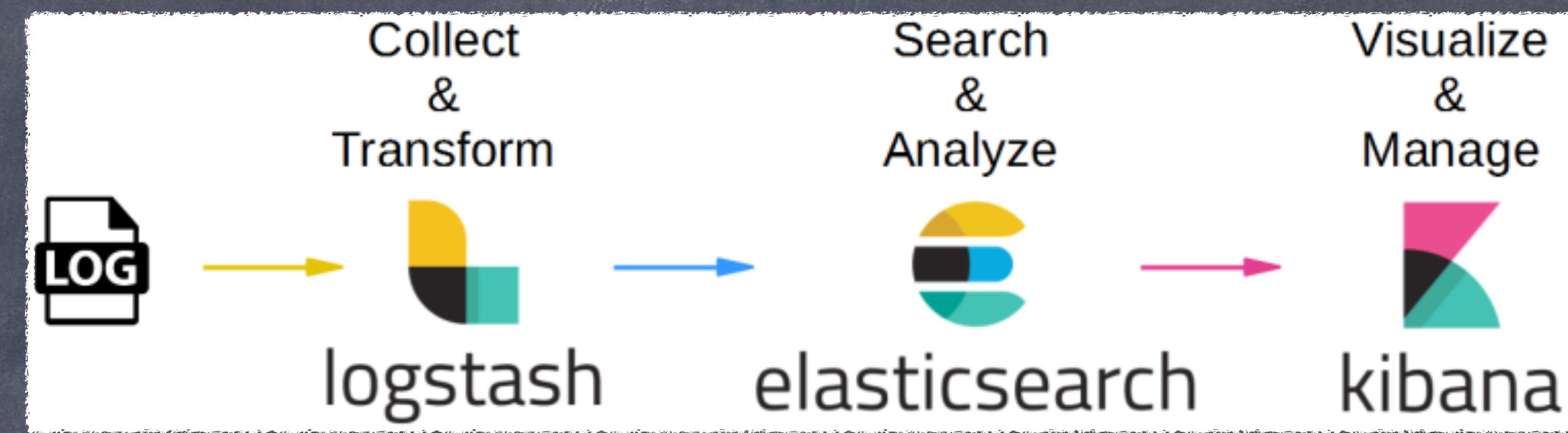
- newbie
- IT
- industry

jobHistories.industryName

- it
- it
- it e
- it en
- it eng
- ....
- it engineering

# Others....

- ⦿ **log aggregation**
- ⦿ **app monitoring**
- ⦿ **machine learning**
- ⦿ **data analytics**



Building the Search  
Database

# Responsibilities: Data Team

- ☛ **ingest**
- ☛ **profile the data**
- ☛ **cleanup**



# Responsibilities: Data Team

- ⦿ ingest
- ⦿ profile the data
- ⦿ cleanup
- ⦿ data warehouse (**Redshift**)



**Amazon Redshift**

# Responsibilities: Data Team

☛ **ingest**

☛ **profile the data**

☛ **cleanup**

☛ **data warehouse (Redshift)**

candidate_profile								
<b>id</b>	<b>full_name</b>	<b>summary</b>	<b>location_lat</b>	<b>location_lon</b>	<b>is_likely_to_jump</b>	<b>job_exp</b>	<b>cur_func</b>	
CN_10001	Olaf Oldtimer	An oldtimer in the IT Industry	40.71	74.00	TRUE	15	FN_100001	
CN_10002	Marco Midd	Worked as a Software Engineer for quite a while	40.71	74.00	TRUE	10	FN_100001	
CN_10003	Nero Newbie	A newbie in the IT Industry	40.71	74.00	FALSE	3	FN_100001	

candidate_job_history						
<b>id</b>	<b>cand_id</b>	<b>job_title</b>	<b>industry_id</b>	<b>is_current</b>	<b>years_exp</b>	
JH_10001	CN_10001	Sr. Software Engineer	IND_100001	TRUE	10	
JH_10002	CN_10001	Jr. Tech Support	IND_100002	FALSE	5	
JH_10003	CN_10002	Sr. Software Engineer	IND_100001	TRUE	5	
JH_10004	CN_10002	Jr. Software Engineer	IND_100001	FALSE	4	
JH_10005	CN_10002	Intern	IND_100001	FALSE	1	
JH_10006	CN_10003	Jr. Software Engineer	IND_100001	TRUE	2	
JH_10007	CN_10003	Intern	IND_100001	FALSE	1	

ref_industry	
<b>id</b>	<b>name</b>
IND_100001	IT Engineering
IND_100002	IT Support

# Responsibilities: Product Team

☛ **unload the data from the Data Warehouse (Redshift) to AWS S3**



candidate_profile								
<b>id</b>	<b>full_name</b>	<b>summary</b>	<b>location_lat</b>	<b>location_lon</b>	<b>is_likely_to_jump</b>	<b>job_exp</b>	<b>cur_func</b>	
CN_10001	Olaf Oldtimer	An oldtimer in the IT Industry	40.71	74.00	TRUE	15	FN_100001	
CN_10002	Marco Midd	Worked as a Software Engineer for quite a while	40.71	74.00	TRUE	10	FN_100001	
CN_10003	Nero Newbie	A newbie in the IT Industry	40.71	74.00	FALSE	3	FN_100001	

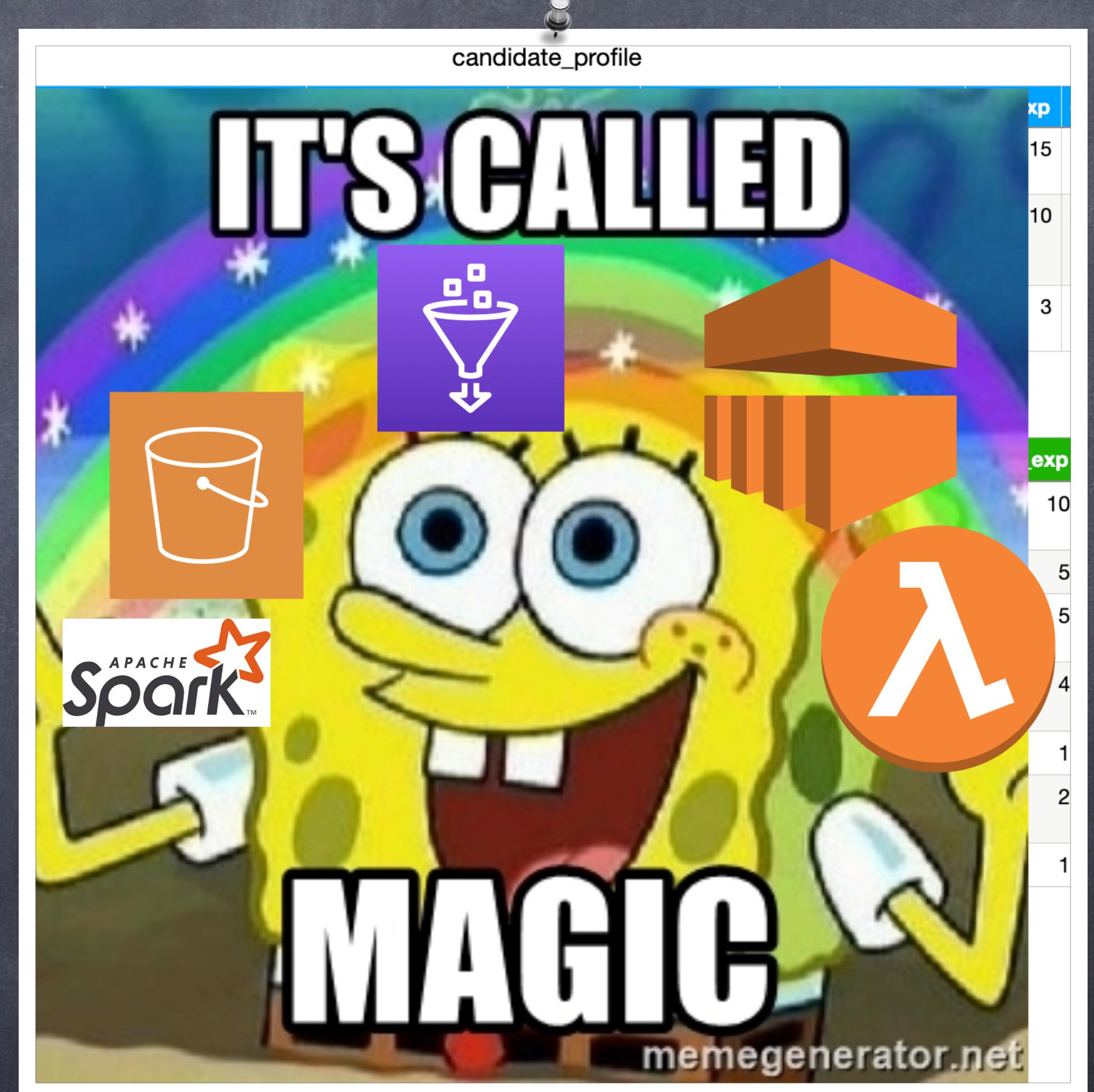
candidate_job_history						
<b>id</b>	<b>cand_id</b>	<b>job_title</b>	<b>industry_id</b>	<b>is_current</b>	<b>years_exp</b>	
JH_10001	CN_10001	Sr. Software Engineer	IND_100001	TRUE	10	
JH_10002	CN_10001	Jr. Tech Support	IND_100002	FALSE	5	
JH_10003	CN_10002	Sr. Software Engineer	IND_100001	TRUE	5	
JH_10004	CN_10002	Jr. Software Engineer	IND_100001	FALSE	4	
JH_10005	CN_10002	Intern	IND_100001	FALSE	1	
JH_10006	CN_10003	Jr. Software Engineer	IND_100001	TRUE	2	
JH_10007	CN_10003	Intern	IND_100001	FALSE	1	

ref_industry	
<b>id</b>	<b>name</b>
IND_100001	IT Engineering
IND_100002	IT Support

# Responsibilities: Product Team

- ☛ **unload the data from the Data Warehouse (Redshift) to AWS S3**
- ☛ **process the data using Spark (AWS Glue or EMR works)**



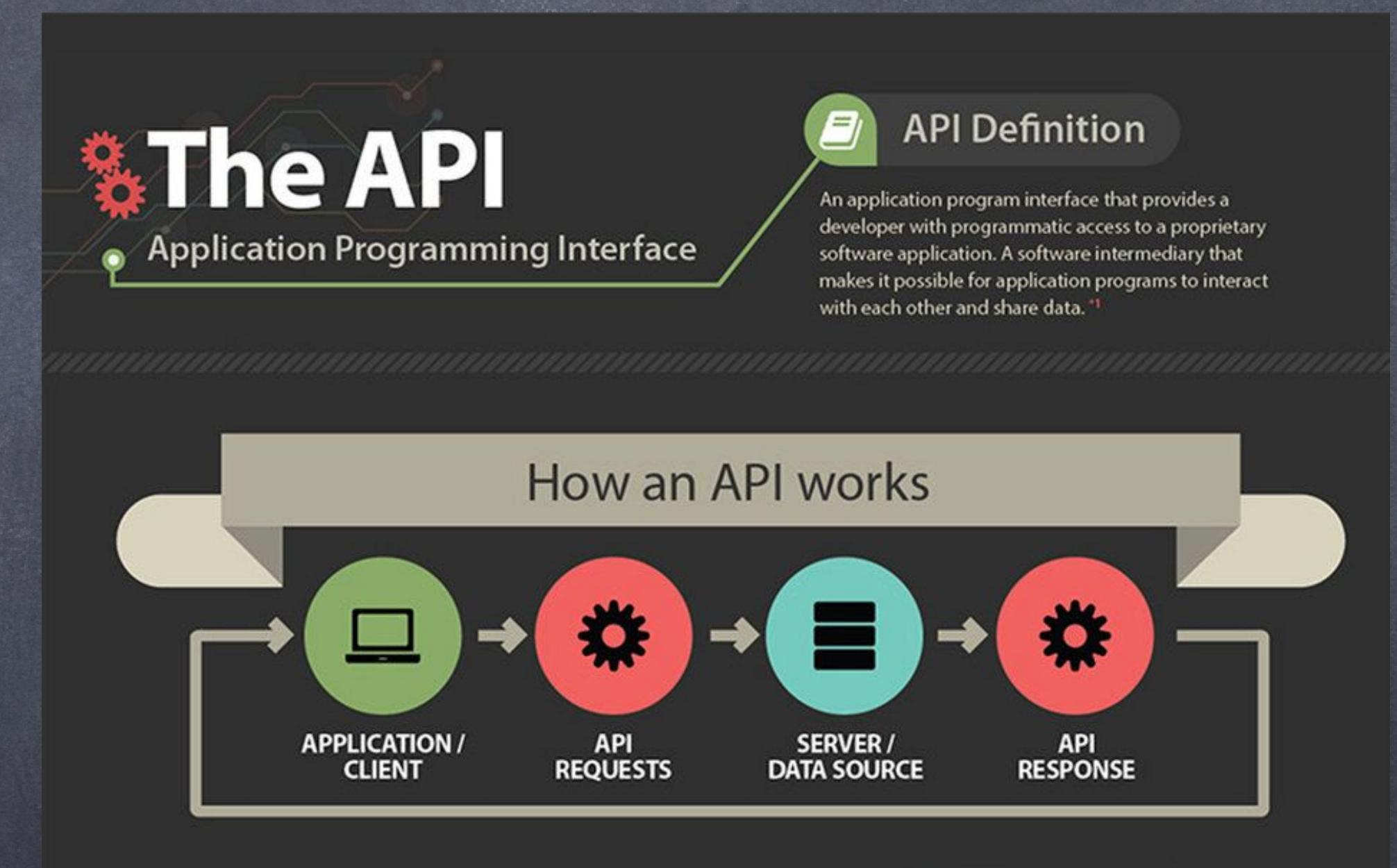
# Responsibilities: Product Team

- ☛ **unload the data from the Data Warehouse (Redshift) to AWS S3**
- ☛ **process the data using Spark (AWS Glue or EMR works)**
- ☛ **insert into Elasticsearch Service**

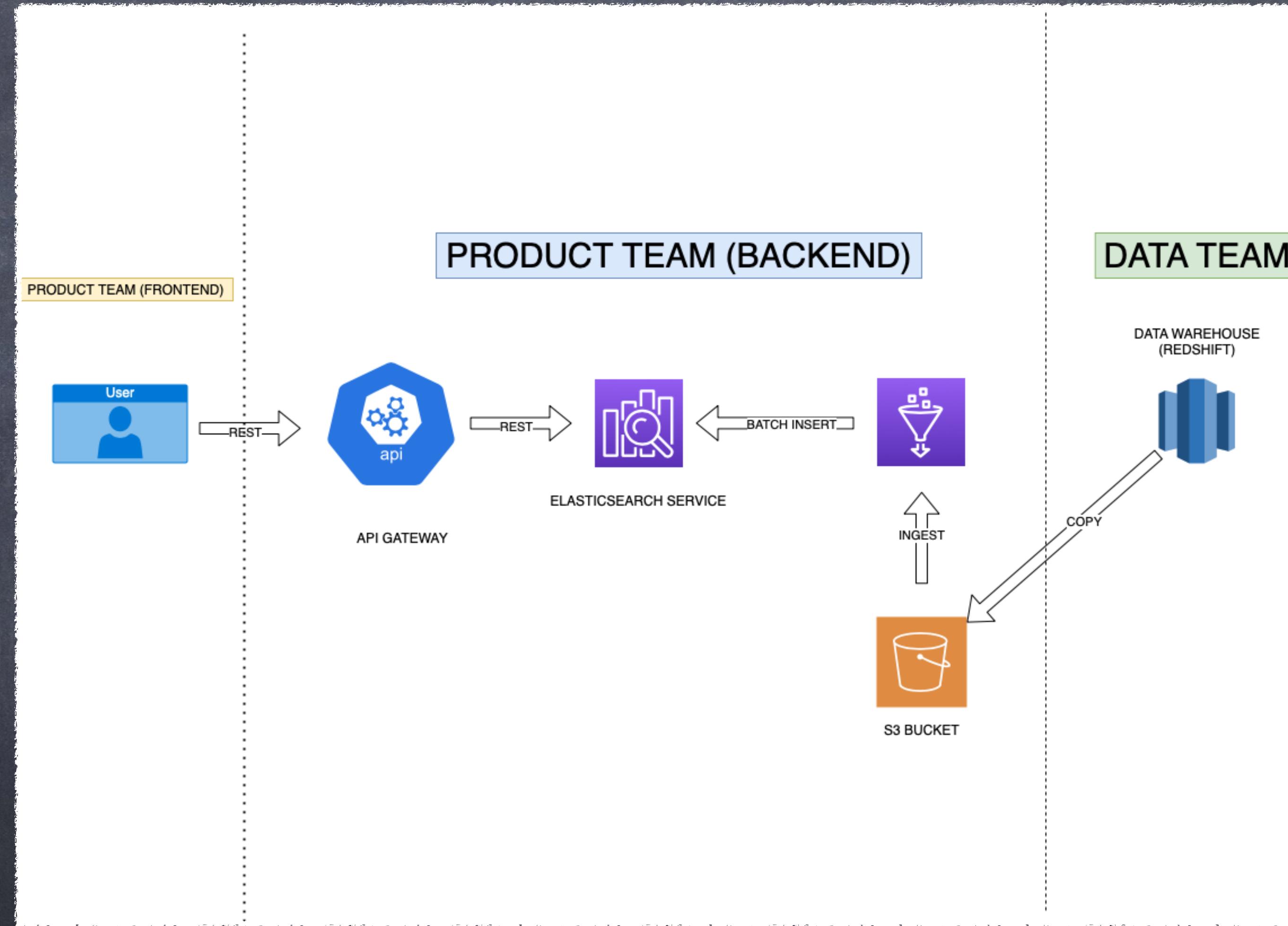
```
141 PUT candidates_demo/_doc/CN_10003
142 {
143   "fullName": "Nero Newbie",
144   "summary": "A newbie in the IT Industry",
145   "location": [40.71, 74.00],
146   "currentFunction": "FN_100001",
147   "likelyToJump": false,
148   "yearsOfExperience": 3,
149   "jobHistories": [
150     {
151       "jobTitle": "Jr. Software Engineer",
152       "industry": "IND_100001",
153       "industryName": "IT Engineering",
154       "yearsOfExperience": 2,
155       "current": true
156     },
157     {
158       "jobTitle": "Intern",
159       "industry": "IND_100001",
160       "industryName": "IT Engineering",
161       "yearsOfExperience": 1,
162       "current": false
163     }
164   ]
165 }
```

# Responsibilities: Product Team

- ☛ **unload the data from the Data Warehouse (Redshift) to AWS S3**
- ☛ **process the data using Spark (AWS Glue or EMR works)**
- ☛ **insert into Elasticsearch Service**
- ☛ **serve the data to the User via REST API**

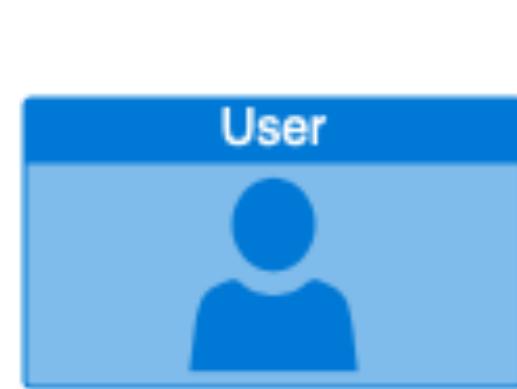


# Overall Architecture



## PRODUCT TEAM (BACKEND)

PRODUCT TEAM (FRONTEND)



REST

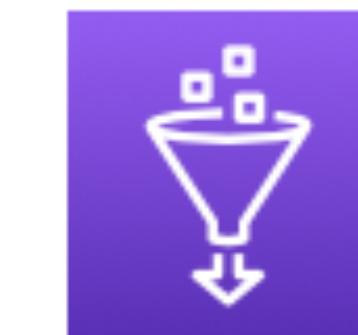


REST



ELASTICSEARCH SERVICE

API GATEWAY



BATCH INSERT



INGEST



S3 BUCKET

## DATA TEAM

DATA WAREHOUSE (REDSHIFT)



# Capabilities and Limitations

# Capabilities

- ⌚ perform complex queries (geospatial, range, nested, composite)

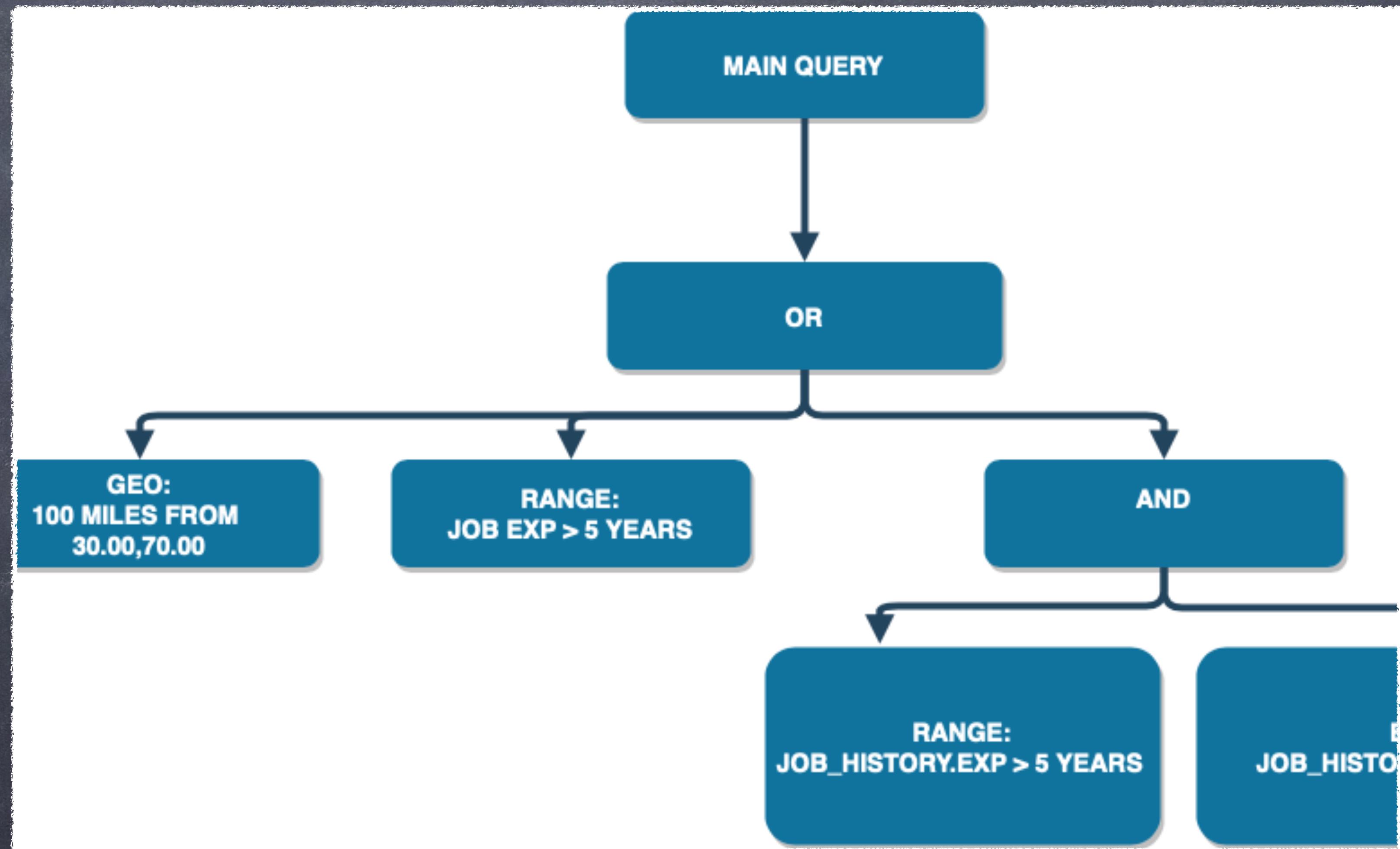
```
67 GET candidates_demo/_search
68 {
69   "explain": false,
70   "query": {
71     "bool": {
72       "should": [
73         {
74           "constant_score": {
75             "filter": {
76               "term": {
77                 "likelyToJump": {
78                   "value": true
79                 }
80               }
81             },
82             "boost": 1
83           }
84         },
85         {
86           "constant_score": {
87             "filter": {
88               "range": {
89                 "yearsOfExperience": {
90                   "gte": 15
91                 }
92               }
93             },
94             "boost": 1.5
95           }
96         }
97       ],
98     }
99   }
100 }
```

```
67 GET candidates_demo/_search
68 {
69   "explain": false,
70   "query": {
71     "bool": {
72       "should": [
73         {
74           "constant_score": {
75             "filter": {
76               "term": {
77                 "likelyToJump": {
78                   "value": true
79                 }
80               }
81             },
82             "boost": 1
83           }
84         },
85         {
86           "constant_score": {
87             "filter": {
88               "range": {
89                 "yearsOfExperience": {
90                   "gte": 15
91                 }
92               }
93             },
94             "boost": 1.5
95           }
96         }
97       ]
98     }
99   }
200 }
```

```
70 # GET candidates_demo/_search
71 {
72     "took" : 3,
73     "timed_out" : false,
74     "_shards" : {
75         "total" : 5,
76         "successful" : 5,
77         "skipped" : 0,
78         "failed" : 0
79     },
80     "hits" : {
81         "total" : {
82             "value" : 2,
83             "relation" : "eq"
84         },
85         "max_score" : 2.5,
86         "hits" : [
87             {
88                 "_index" : "candidates_demo",
89                 "_type" : "_doc",
90                 "_id" : "CN_10001",
91                 "_score" : 2.5,
92                 "_source" : {
93                     "fullName" : "Olaf Oldtimer",
94                     "summary" : "An oldtimer in the IT Industry",
95                     "location" : [
96                         40.71,
97                         74.0
98                     ],
99                     "currentFunction" : "FN_100001",
100                    "likelyToJump" : true,
101                    "yearsOfExperience" : 15,
102                    "jobHistories" : [
103                        {
104                            "company" : "Oldtimer Solutions",
105                            "function" : "Software Developer",
106                            "startYear" : 1985,
107                            "endYear" : 2005,
108                            "details" : [
109                                "Worked on various projects from mainframe to client-server environments.",
110                                "Migrated systems from COBOL to C/C++ and Java."],
111                            "titles" : [
112                                "Junior Developer", "Associate Developer", "Senior Developer"]
113                        }
114                    ]
115                }
116            }
117        ]
118    }
119}
```

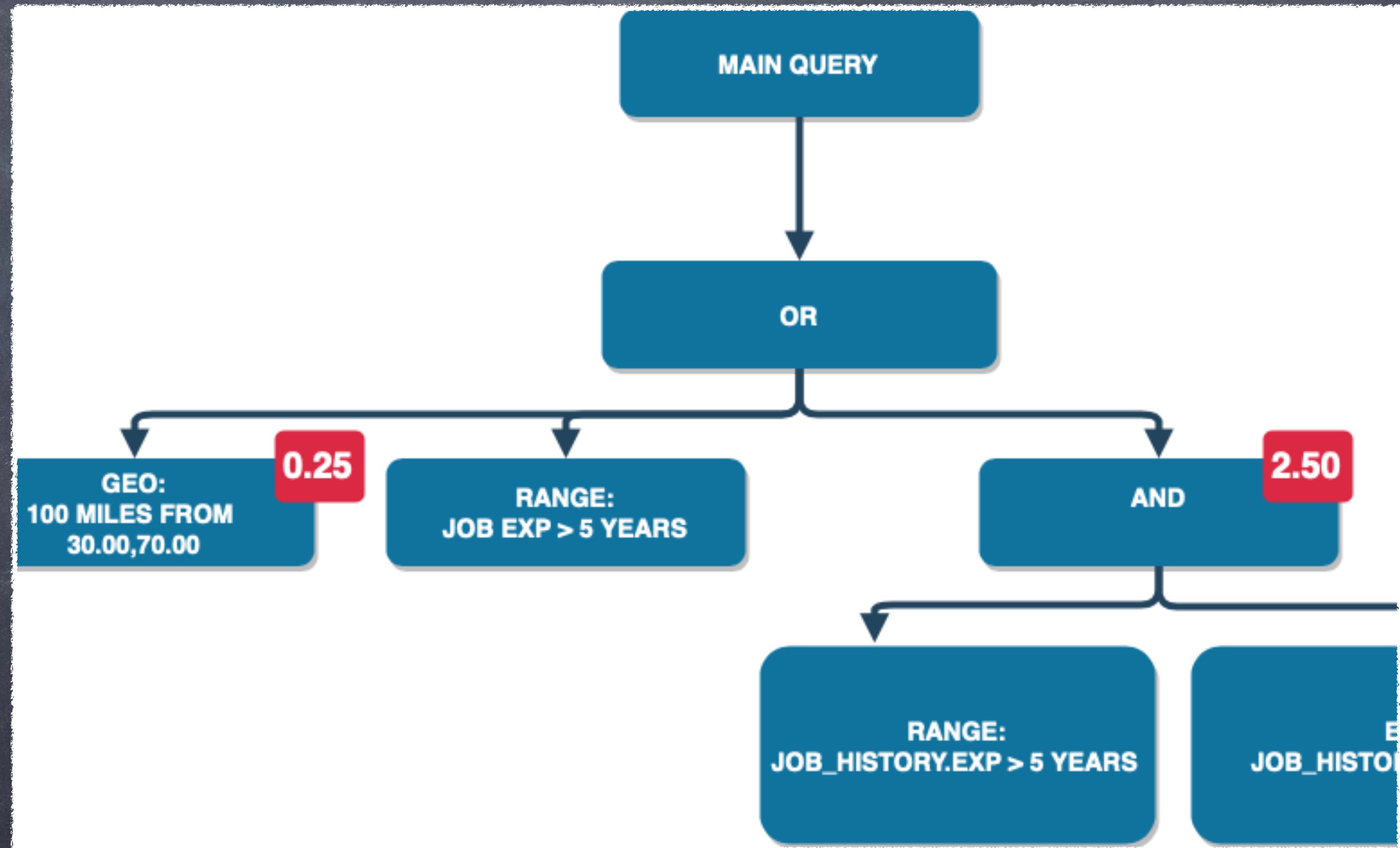
# Capabilities

- perform complex queries (geospatial, range, nested, composite)



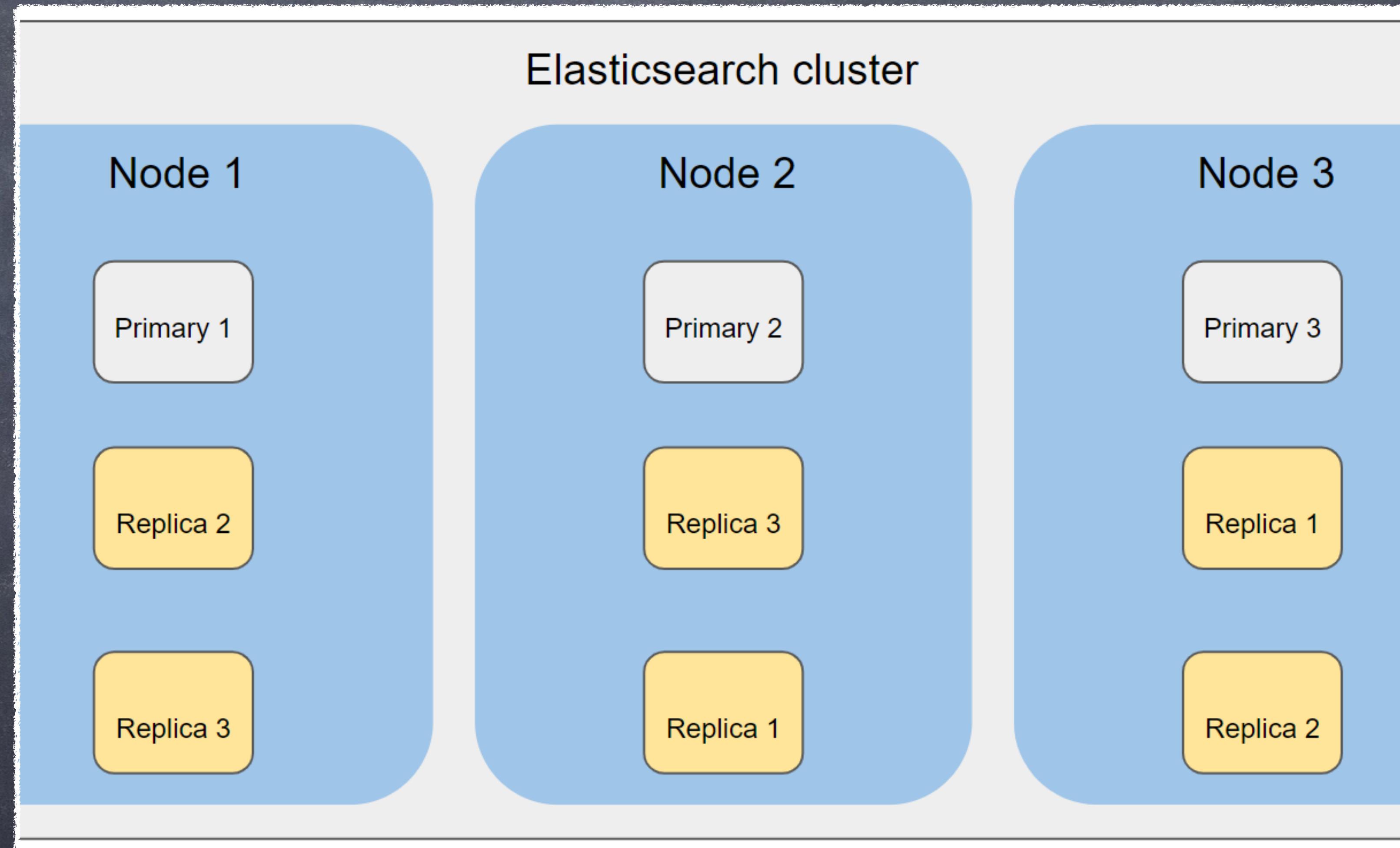
# Capabilities

- ⦿ **perform complex queries (geospatial, range, nested, composite)**
- ⦿ **add weights to influence the score**



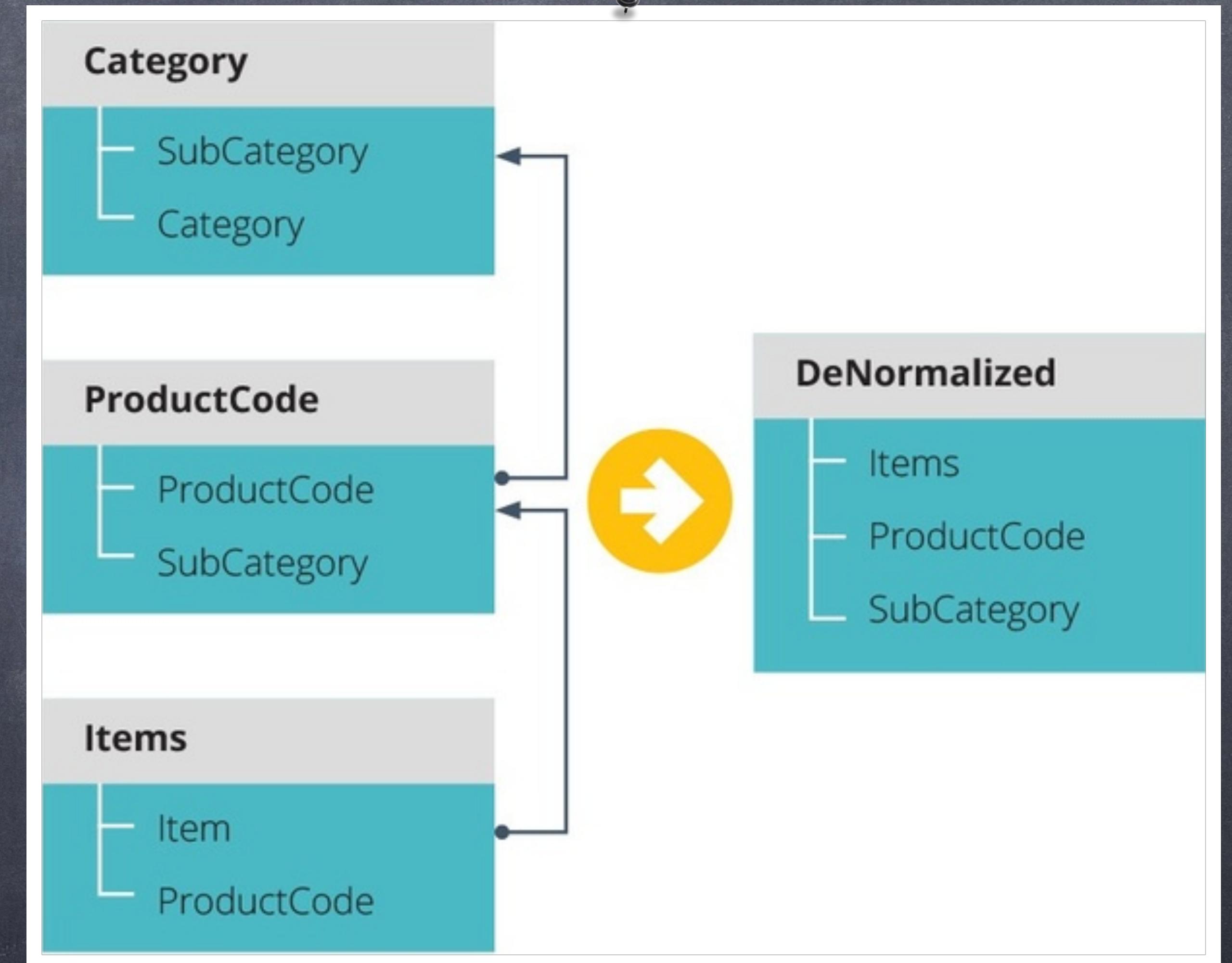
# Capabilities

- ⦿ **perform complex queries (geospatial, range, nested, composite)**
- ⦿ **add weights to influence the score**
- ⦿ **infinite scalability**



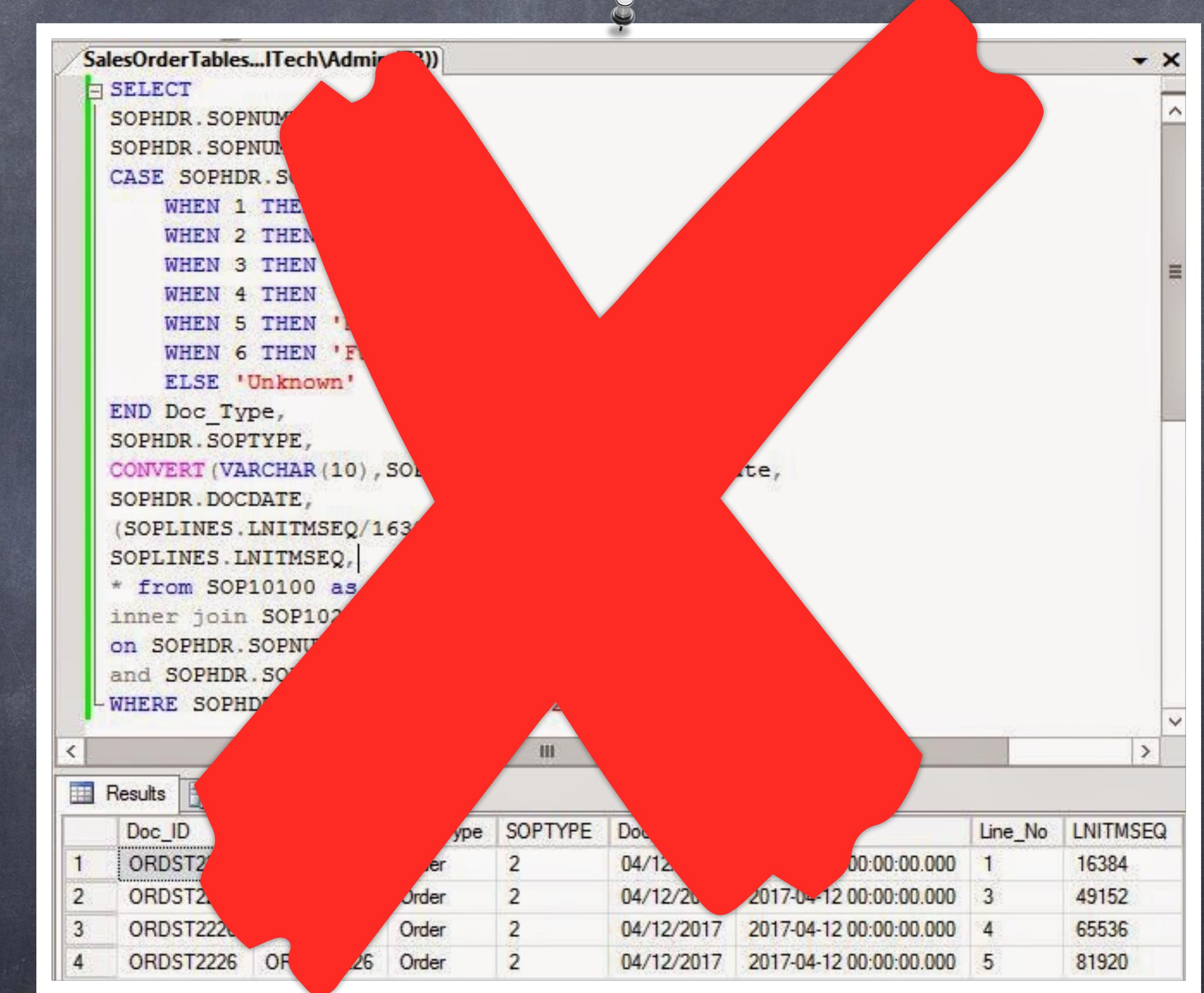
# Limitations

- ⌚ joins  
(denormalization)



# Limitations

- ad hoc queries  
(update, recreate  
and re-insert the  
index every time we  
have a new query  
to support)



The image shows a screenshot of a SQL query editor window titled "SalesOrderTables...ITech\Admin (3)". The query is a complex SELECT statement involving multiple tables and a CASE statement to determine a Doc\_Type based on SOPNUMR values. The results pane below the query shows four rows of data, each with a Doc\_ID starting with "ORDST2". A large red 'X' is drawn across the entire window, indicating a limitation or error.

```
SELECT
    SOPHDR.SOPNUMR,
    SOPHDR.SOPNUMR,
    CASE SOPHDR.SOPNUMR
        WHEN 1 THEN 'Order'
        WHEN 2 THEN 'Order'
        WHEN 3 THEN 'Order'
        WHEN 4 THEN 'Order'
        WHEN 5 THEN 'Order'
        WHEN 6 THEN 'Order'
        ELSE 'Unknown'
    END Doc_Type,
    SOPHDR.SOCTYPE,
    CONVERT(VARCHAR(10), SOPHDR.DOCDATE),
    (SOPLINES.LNITMSEQ/16384) AS Line_No,
    SOPLINES.LNITMSEQ,
    * from SOP10100 as
    inner join SOP10200
    on SOPHDR.SOPNUMR = SOP10200.SOPNUMR
    and SOPHDR.SOPNUMR = SOP10200.SOPNUMR
    WHERE SOPHDR.SOPNUMR > 1000000000000000000
```

Doc_ID	Type	SOCTYPE	DocDate	Line_No	LNITMSEQ
1 ORDST2226	Order	2	04/12/2017	1	16384
2 ORDST2226	Order	2	04/12/2017	3	49152
3 ORDST2226	Order	2	04/12/2017	4	65536
4 ORDST2226	Order	2	04/12/2017	5	81920

Dev Tools

History Settings Help

Console

```
1 DELETE candidates_demo
2 PUT candidates_demo
3 { [REDACTED]
69
70 POST /_aliases
71 { [REDACTED]
81
82 PUT candidates_demo/_doc/CN_10001
83 { [REDACTED]
107
108 PUT candidates_demo/_doc/CN_10002
109 { [REDACTED]
140
141 PUT candidates_demo/_doc/CN_10003
142 { [REDACTED]
166
167 GET candidates_demo/_search
168 { [REDACTED]
201
202 GET candidates_demo/_search
203 { [REDACTED]
232
233 GET candidates_demo/_analyze
234 { [REDACTED]
237
238 GET candidates_demo/_analyze
239 { [REDACTED]
243
244 GET candidates_demo/_analyze
245 { [REDACTED]
:
1 { [
2 "took" : 10,
3 "timed_out" : false,
4 "_shards" : {
5   "total" : 20,
6   "successful" : 20,
7   "skipped" : 0,
8   "failed" : 0
9 },
10 "hits" : {
11   "total" : {
12     "value" : 10000,
13     "relation" : "gte"
14 },
15   "max_score" : 1.0,
16   "hits" : [
17     {
18       "_index" : ".kibana_1",
19       "_type" : "_doc",
20       "_id" : "config:7.4.2",
21       "_score" : 1.0,
22       "_source" : {
23         "config" : {
24           "buildNum" : 26506
25         },
26         "type" : "config",
27         "updated_at" : "2020-08-22T14:21:30.014Z"
28       }
29     },
30     {
31       "_index" : ".kibana_1",
32       "_type" : "_doc",
33       "_id" : "ui-metric:Kibana_home:sampleDataDecline",
34       "_score" : 1.0,
35       "_source" : {
36         "ui-metric" : {
37           "count" : 3
38         },
39         "type" : "ui-metric",
40         "updated_at" : "2020-08-23T11:26:57.748Z"
41       }
42     }
43   ]
44 }
```

"The End."

-LM Bibera