

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

- ⌚ indexes 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

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								
id	full_name	summary	location_lat	location_lon	is_likely_to_jump	job_exp	cur_func	
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						
id	cand_id	job_title	industry_id	is_current	years_exp	
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	
id	name
IND_100001	IT Engineering
IND_100002	IT Support

Responsibilities: Product Team

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



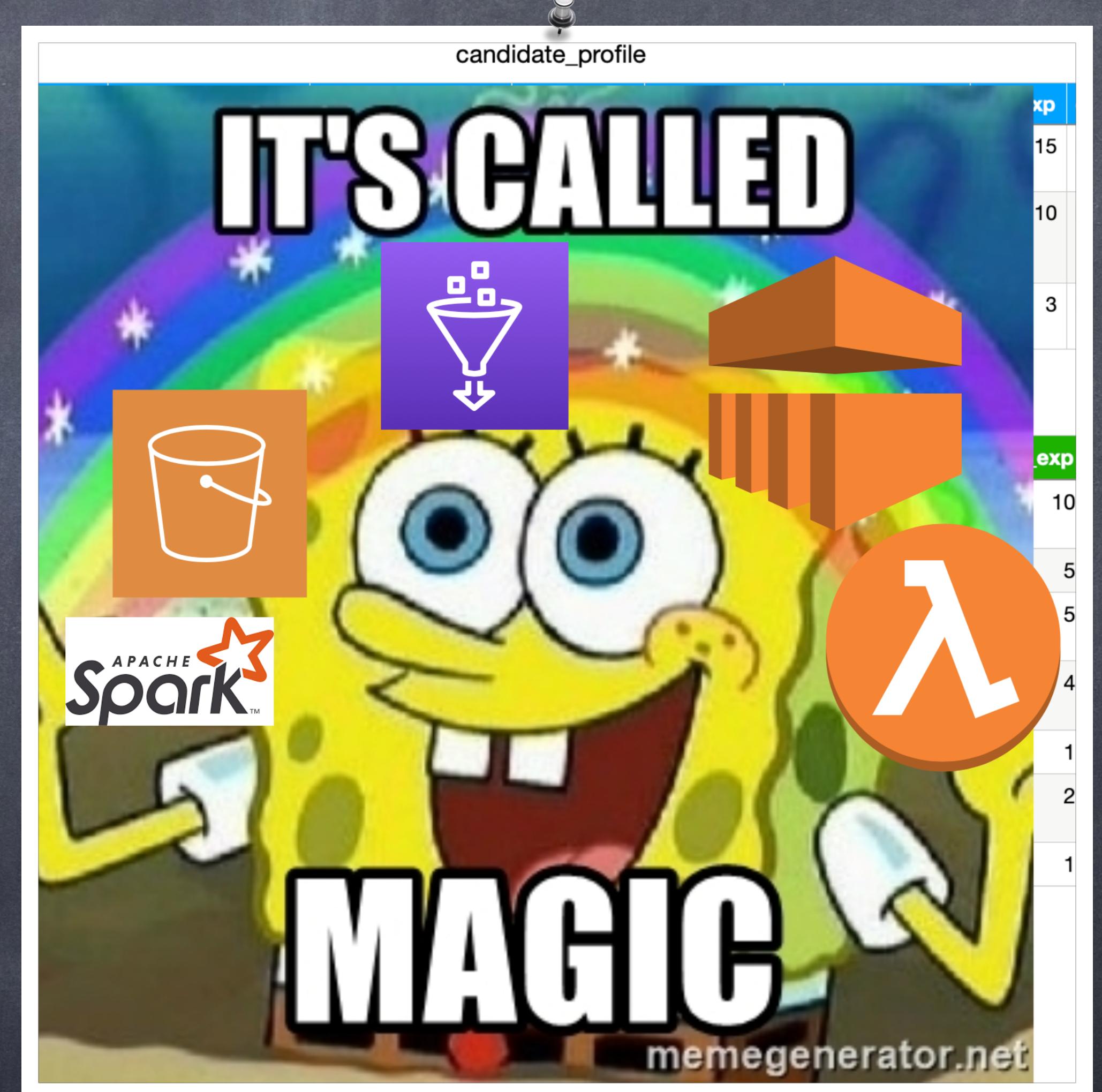
candidate_profile								
id	full_name	summary	location_lat	location_lon	is_likely_to_jump	job_exp	cur_func	
CN_10001	Olaf Oldtimer	An oldtimer in the IT Industry	40.71	74.00	TRUE	15	FN_100001	
CN_10002	Marco Midd	Worked as a Software Engineer for quite a while	40.71	74.00	TRUE	10	FN_100001	
CN_10003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

candidate_job_history						
id	cand_id	job_title	industry_id	is_current	years_exp	
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	
id	name
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)**

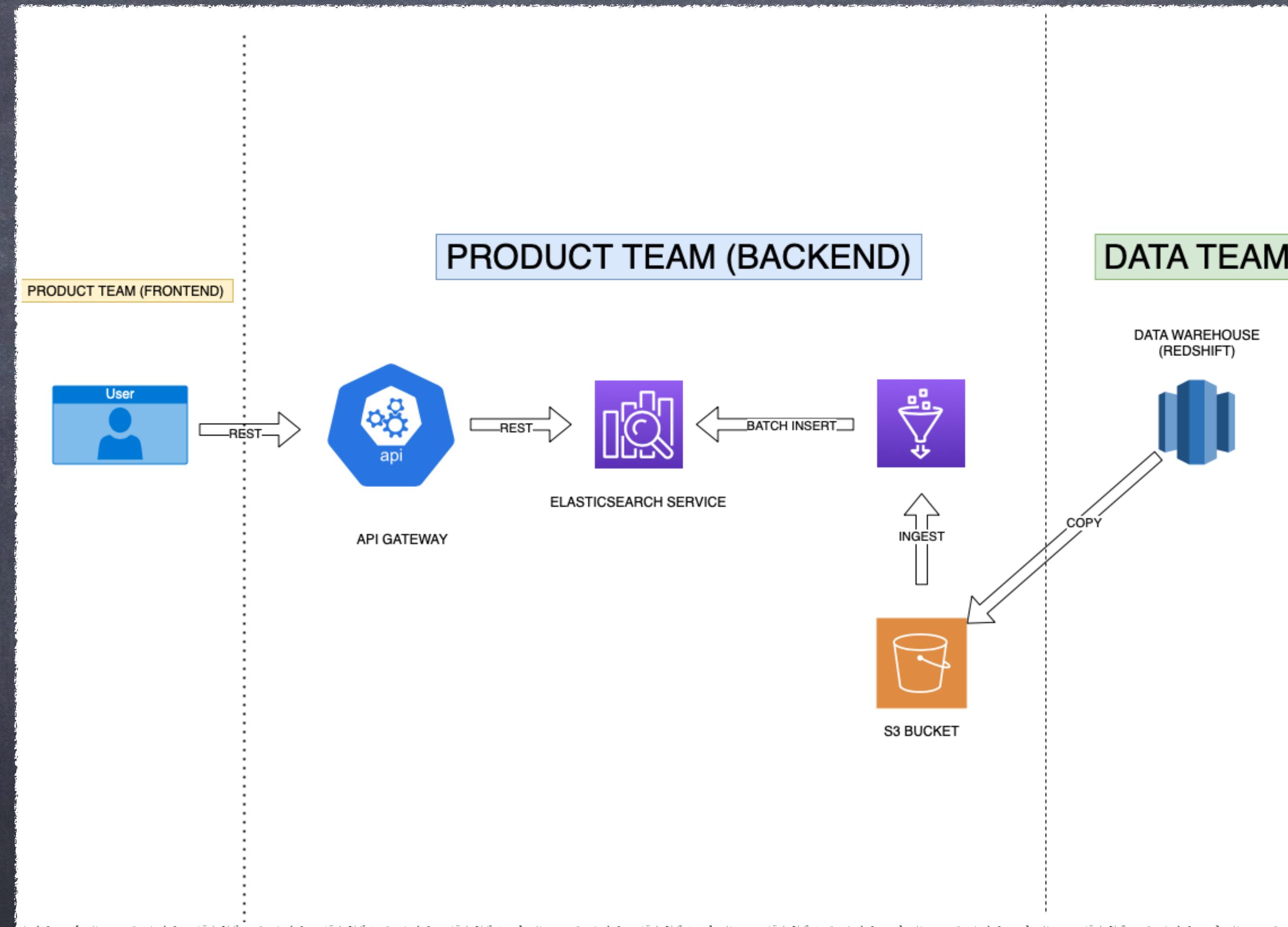


Responsibilities: Product Team

- ☛ **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**

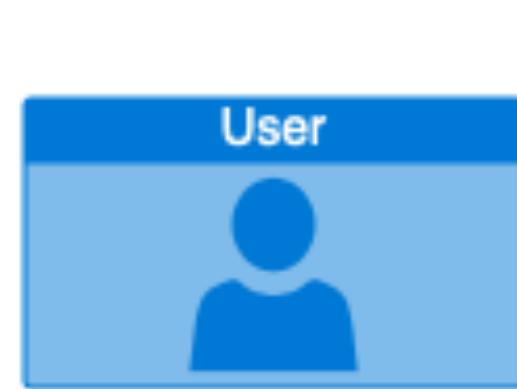
```
141 PUT candidates_demo/_doc/CN_10003
142 {
143   "fullName": "Nero Newbie",
144   "summary": "A newbie in the IT Industry",
145   "location": [40.71, 74.00],
146   "currentFunction": "FN_100001",
147   "likelyToJump": false,
148   "yearsOfExperience": 3,
149   "jobHistories": [
150     {
151       "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 }
```

Overall Architecture



PRODUCT TEAM (BACKEND)

PRODUCT TEAM (FRONTEND)



REST

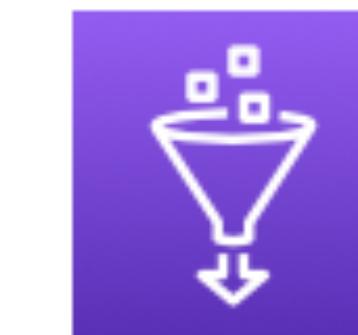


REST



ELASTICSEARCH SERVICE

API GATEWAY



BATCH INSERT



INGEST



S3 BUCKET

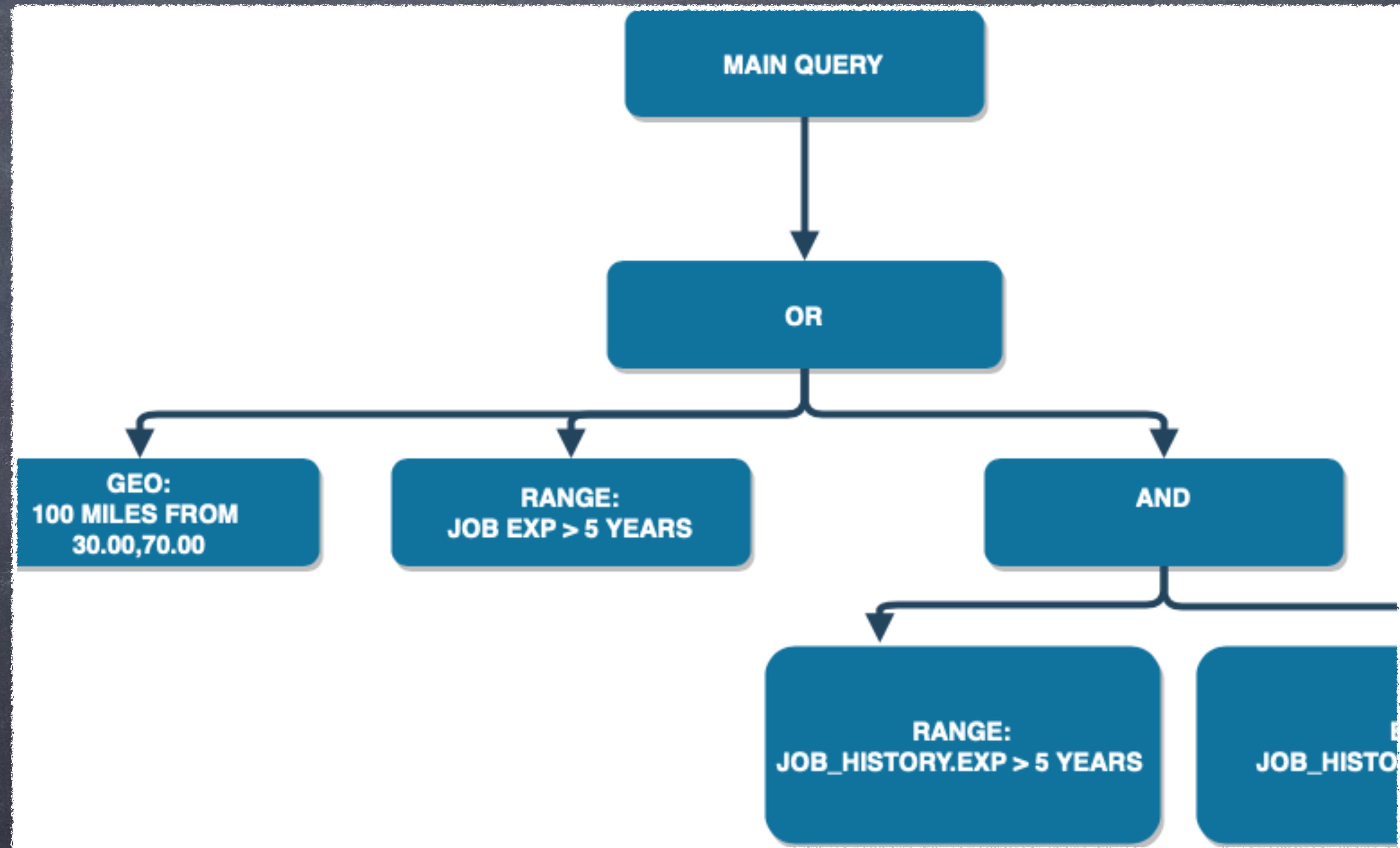
DATA TEAM

DATA WAREHOUSE (REDSHIFT)

Capabilities and Limitations

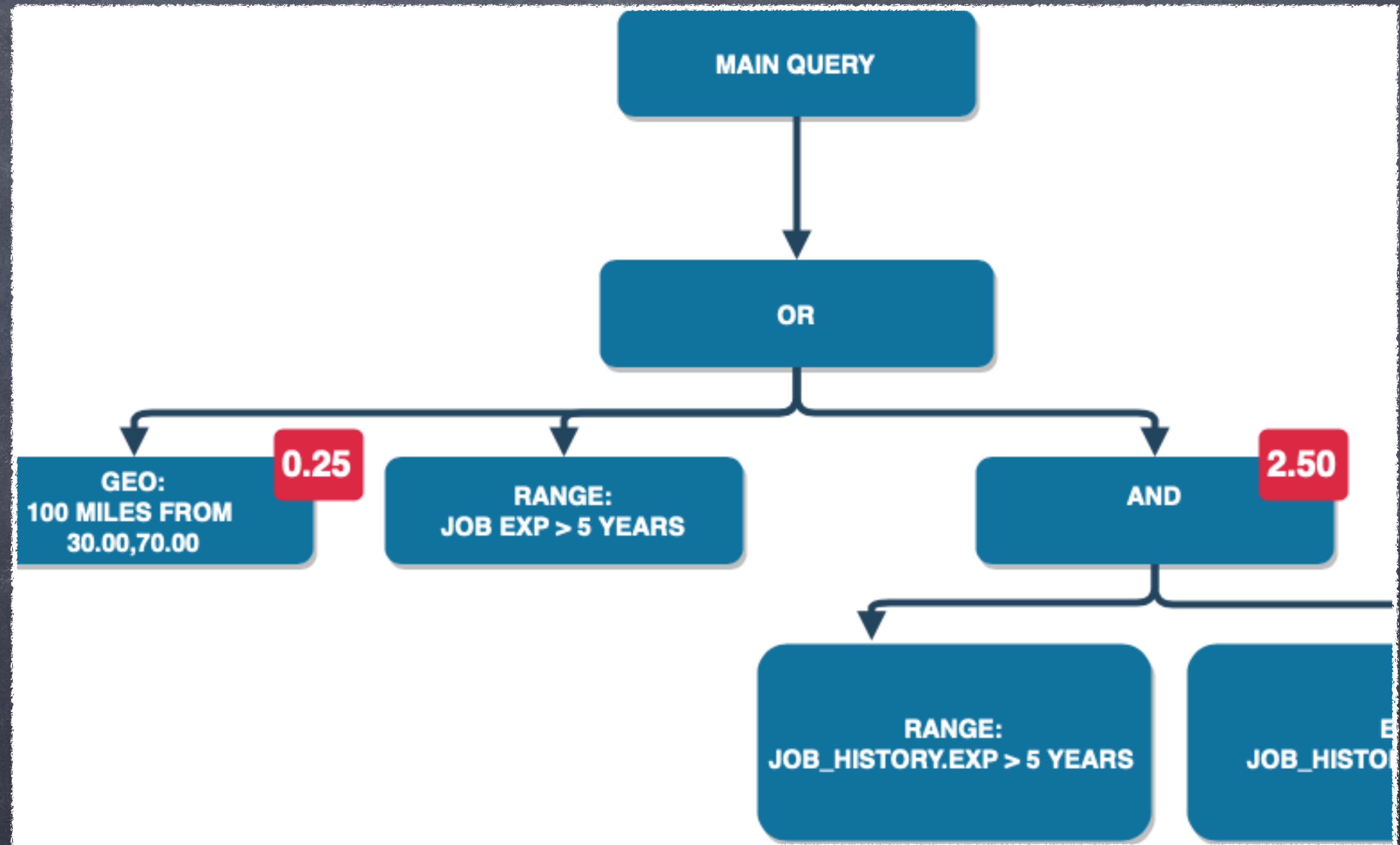
Capabilities

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



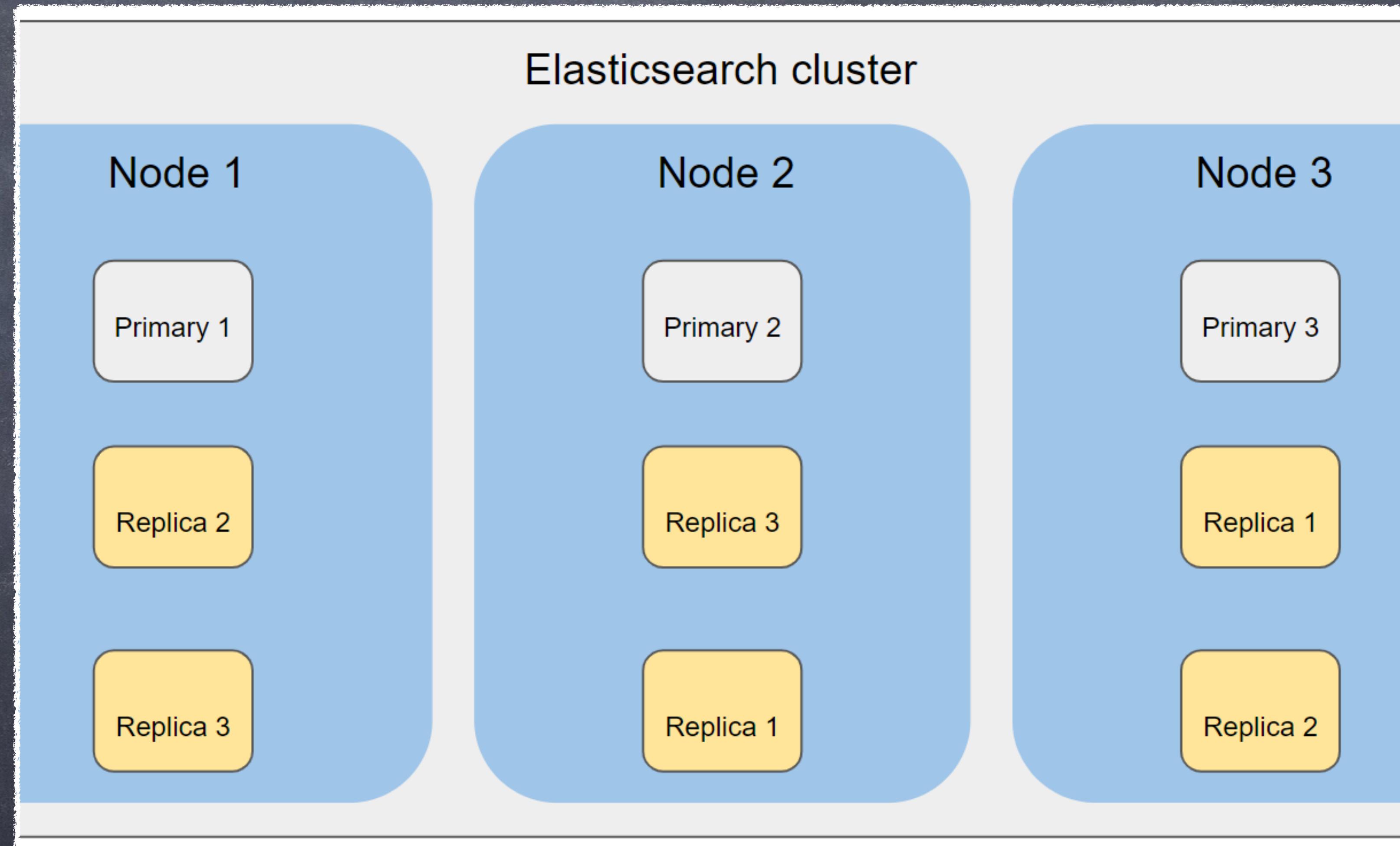
Capabilities

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



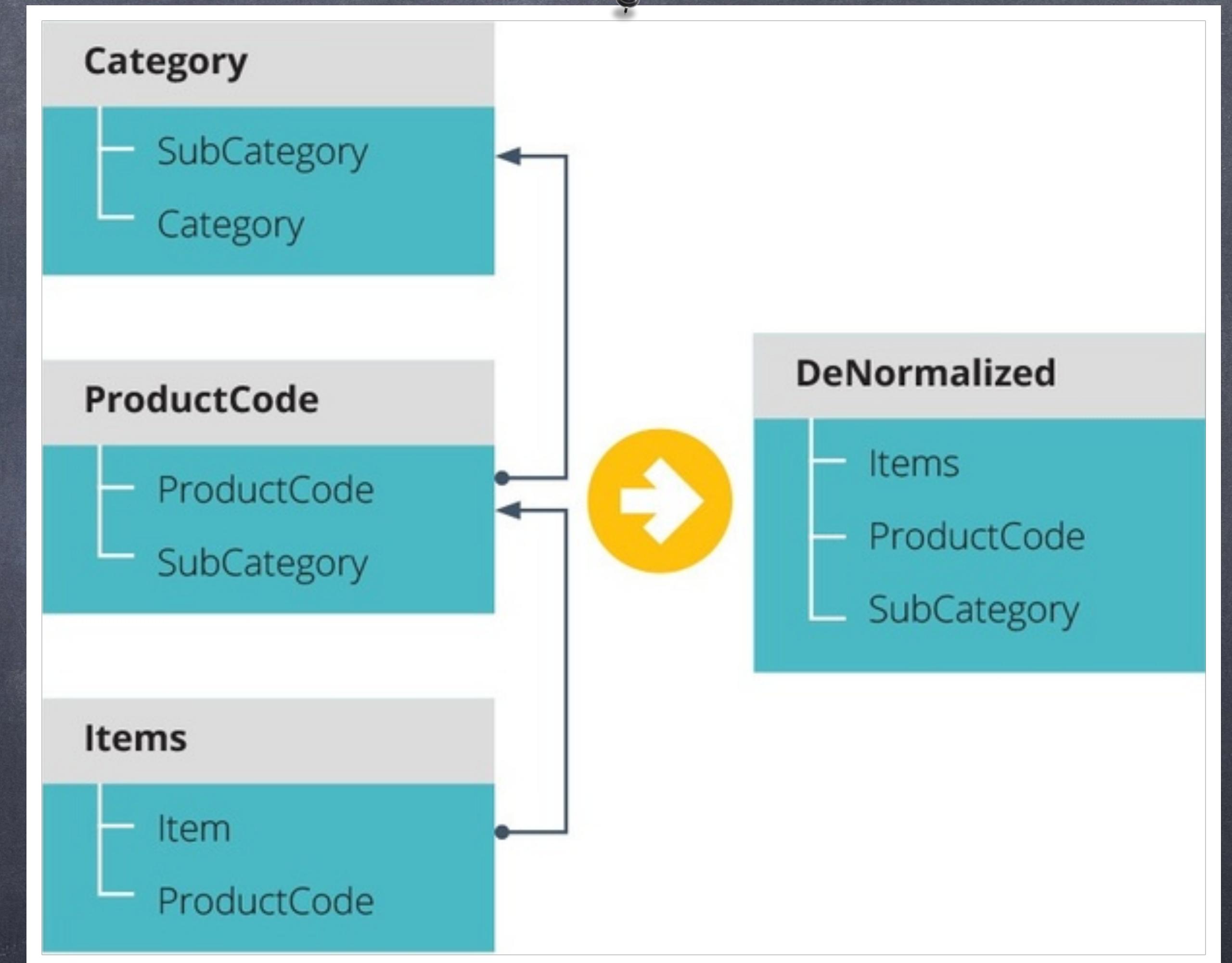
Capabilities

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



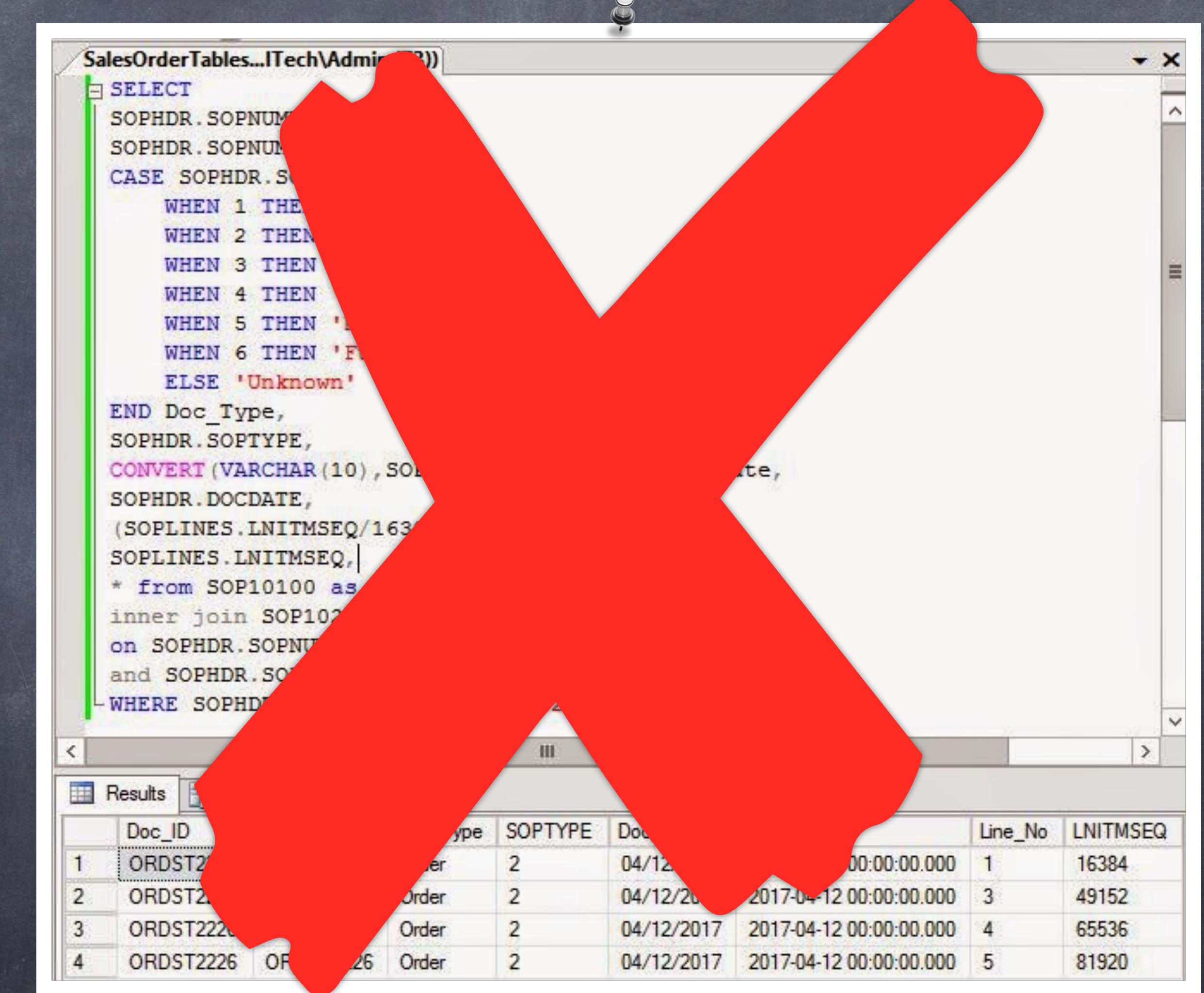
Limitations

- ⌚ 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 SOPHDR.SOPNUMR. The results pane below shows four rows of data corresponding to the query.

```
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
ORDST2226	Order	2	04/12/2017	1	16384
ORDST2226	Order	2	04/12/2017	3	49152
ORDST2226	Order	2	04/12/2017	4	65536
ORDST2226	Order	2	04/12/2017	5	81920

"The End."

-LM Bibera