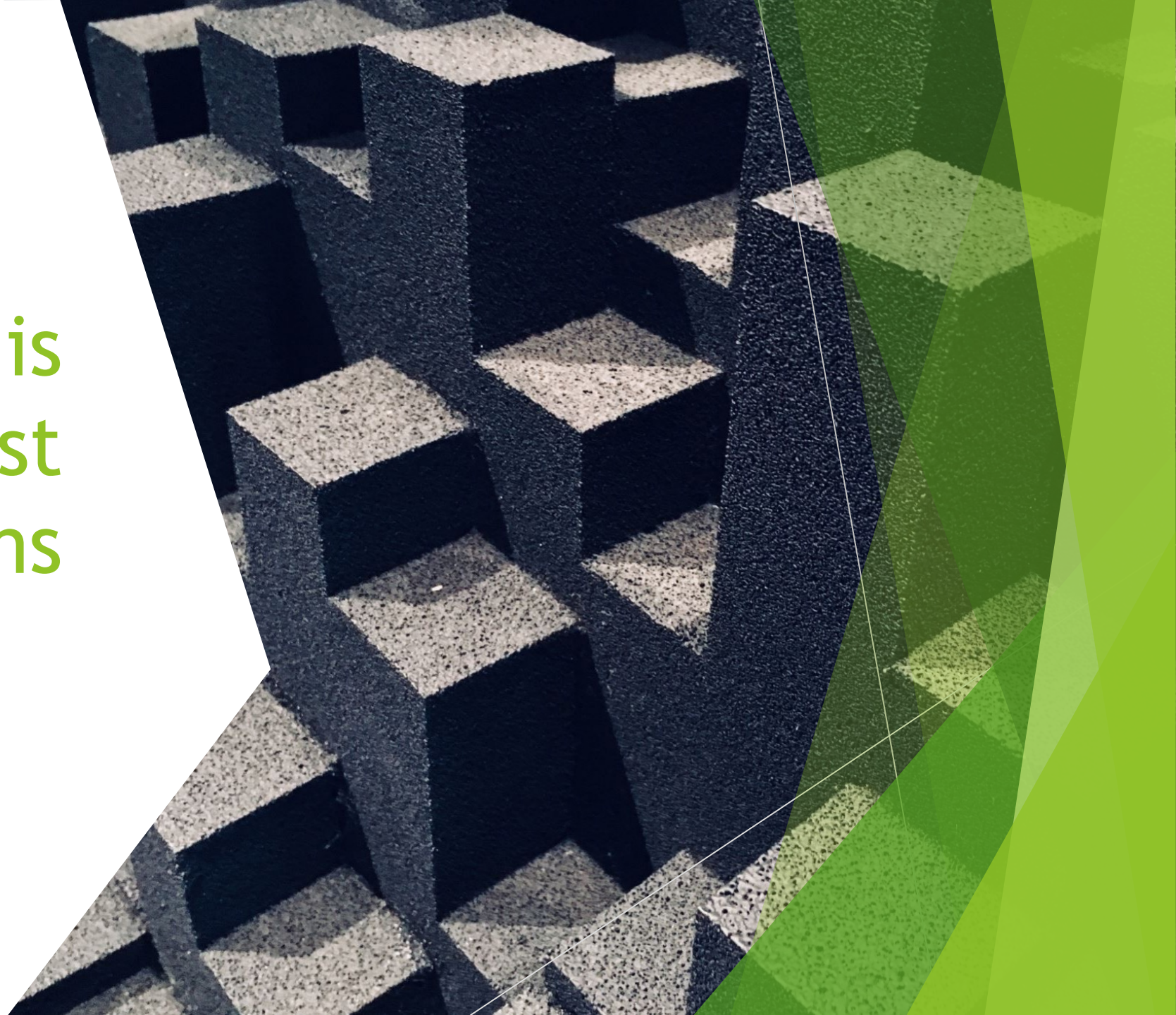
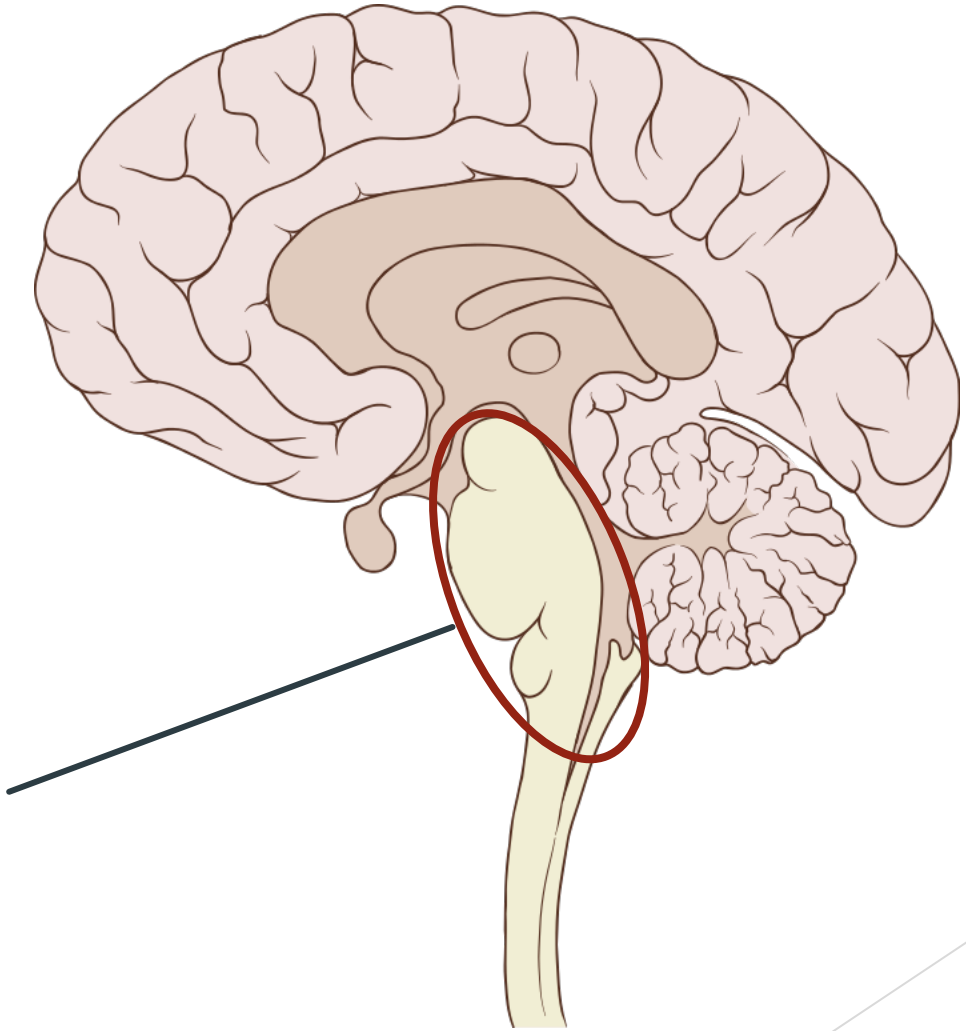
Relevance is more than just algorithms



About me

- Psychology and Computer Science
- Querqy Committer
- Team Servant
- Data & Cloud Engineer
- Product Manager (if there is no alternative)

Evolution of Human Brains

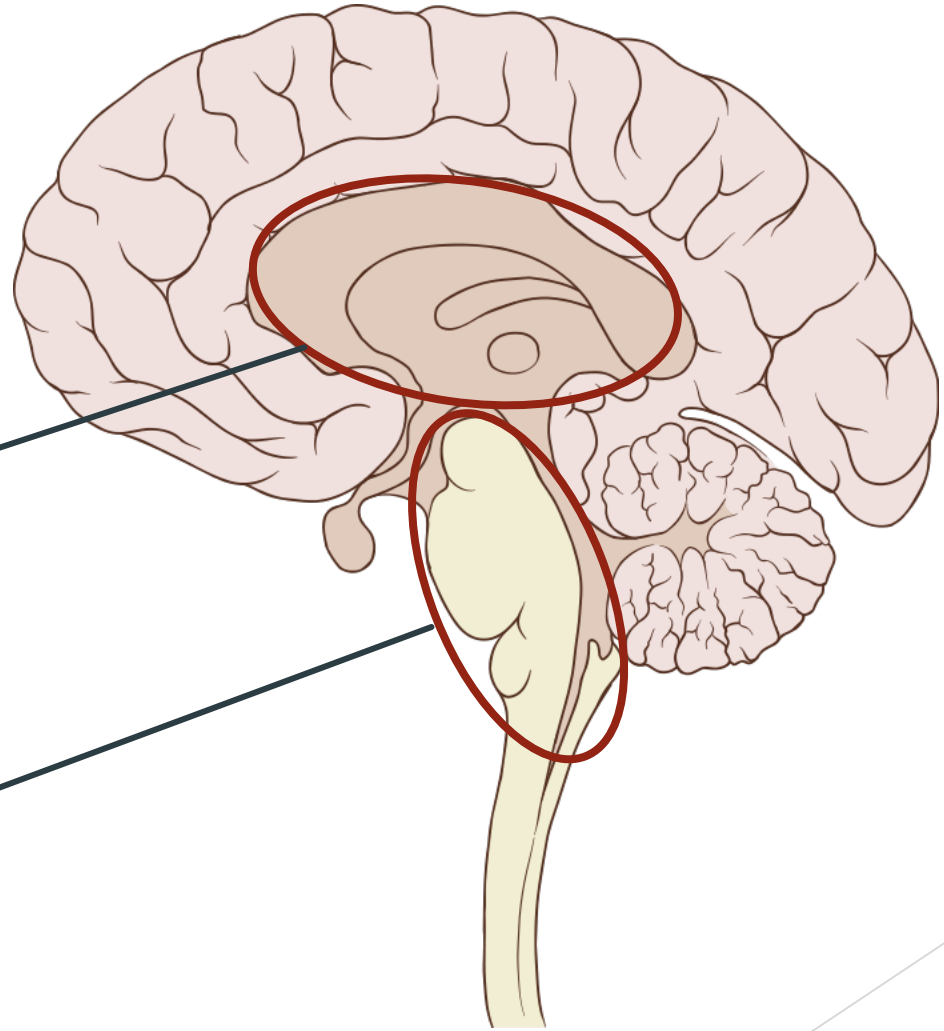


Brainstem:
Basic functionalities to survive

Evolution of Human Brains

Limbic system (emotion):
Making basic functionalities **contextual**

Brainstem:
Basic functionalities to survive

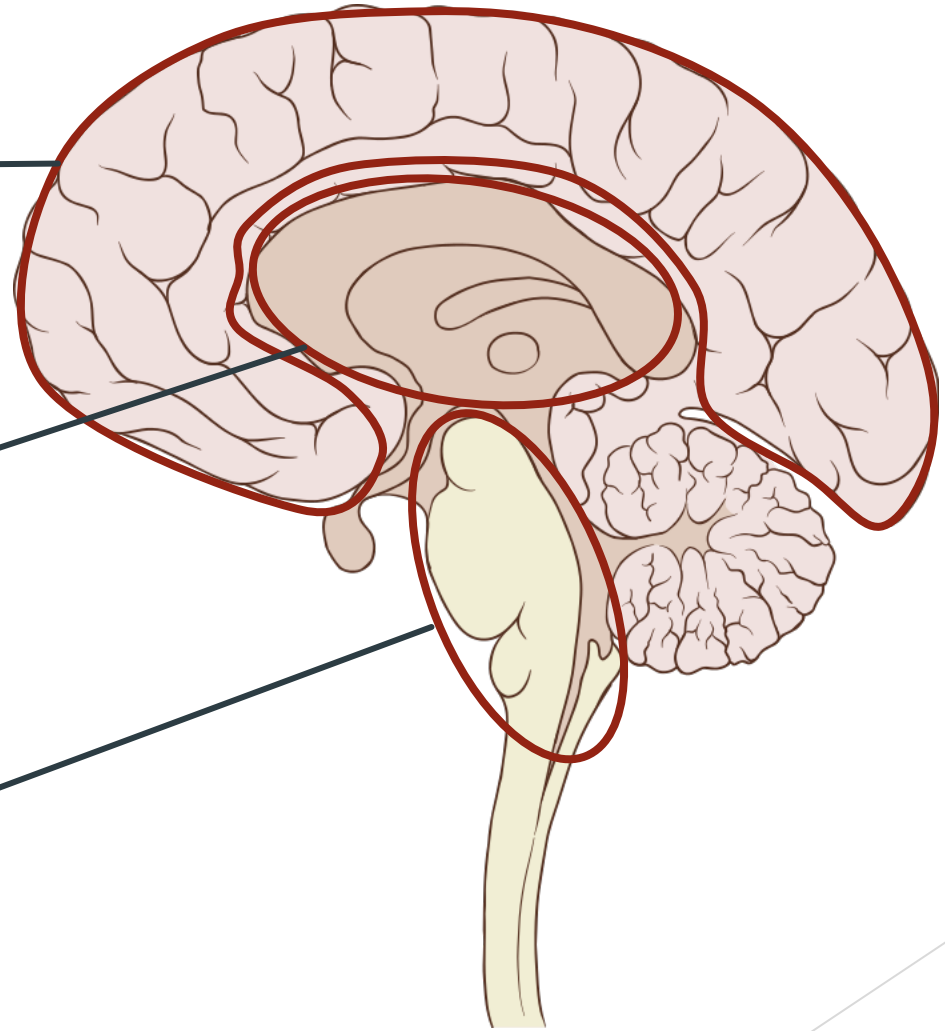


Evolution of Human Brains

Cortex (**intelligence**):
Learning, thinking, judging

Limbic system (emotion):
Making basic functionalities **contextual**

Brainstem:
Basic functionalities to survive

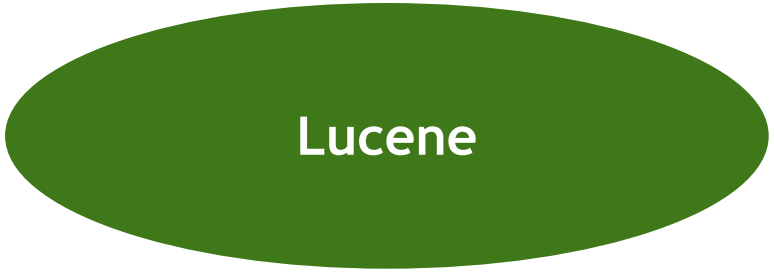


Evolution of Human Brains

- Interplay of intelligence and basic functionalities not always optimal
- E.g. panic due to (harmless) spiders is primarily regulated by emotion – cannot be overruled by cognition

Evolution of Search Technologies

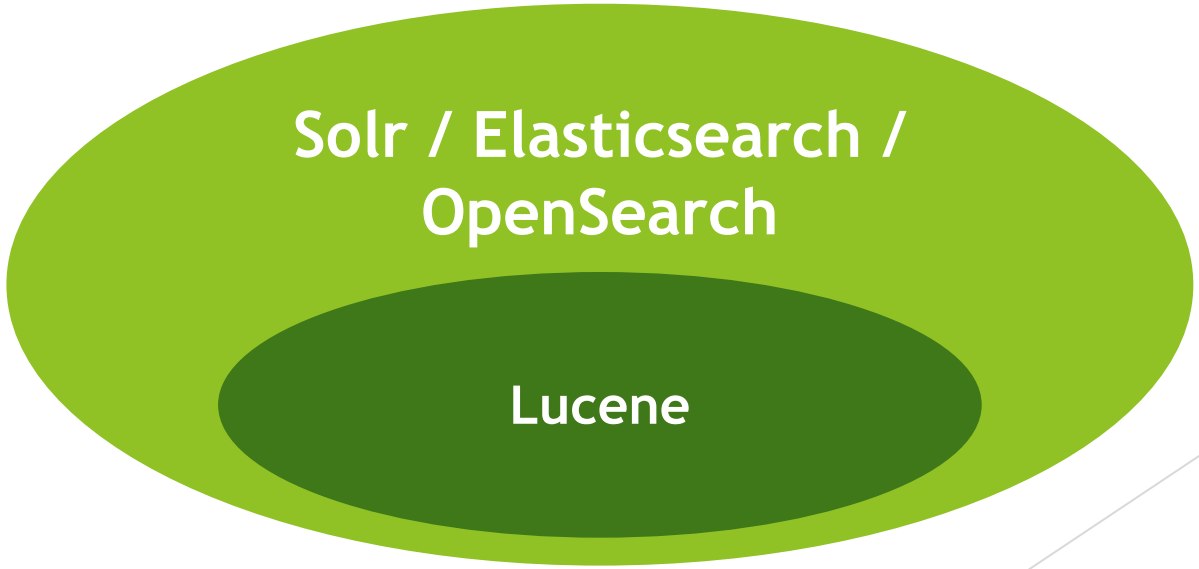
Basic functionalities



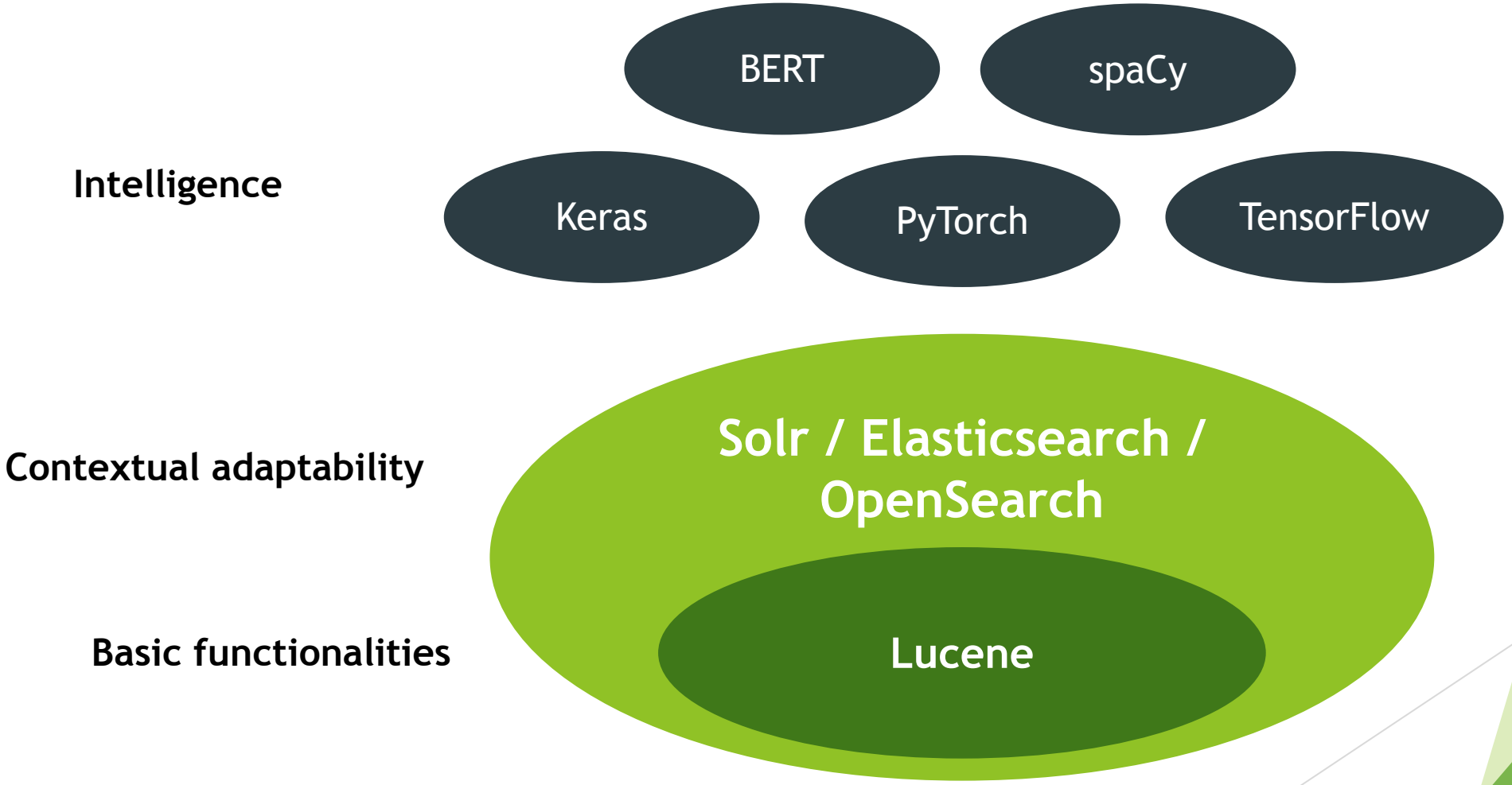
Evolution of Search Technologies

Contextual adaptability

Basic functionalities



Evolution of Search Technologies

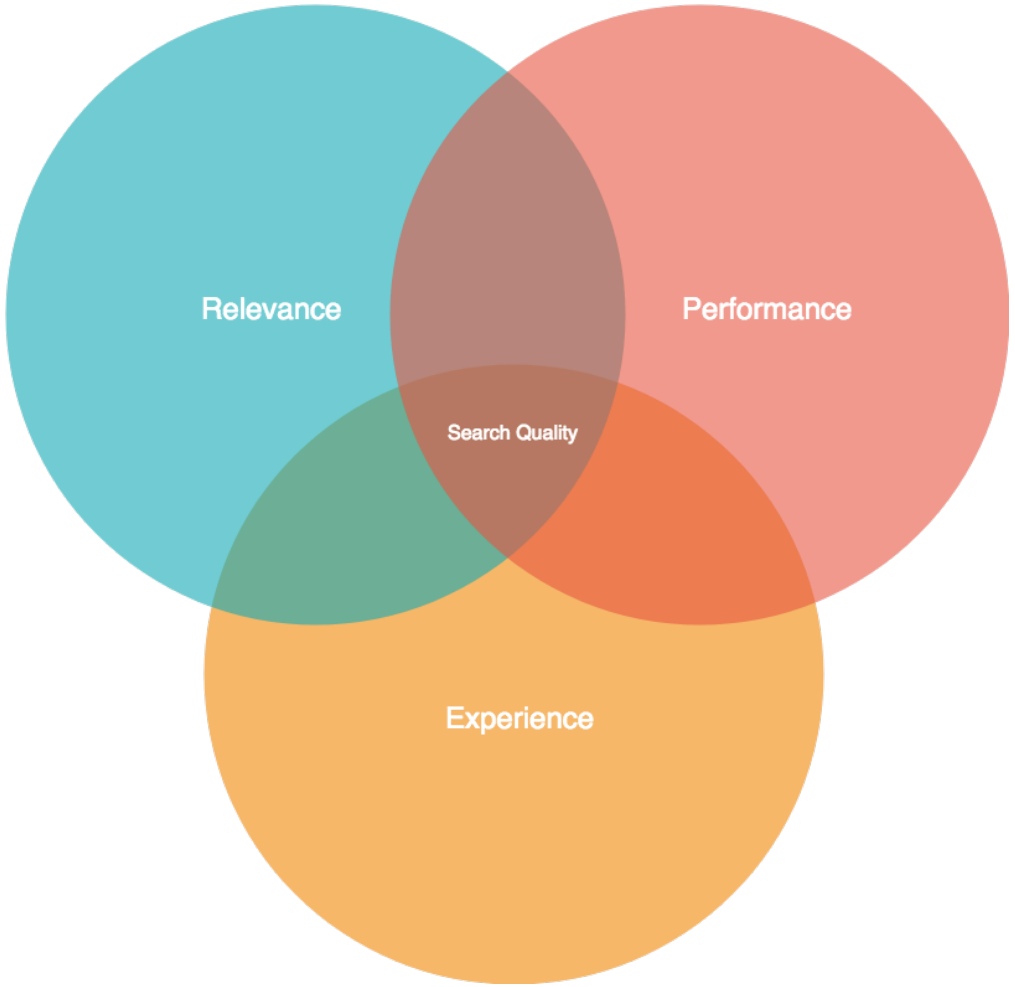


Evolution of Search Technologies

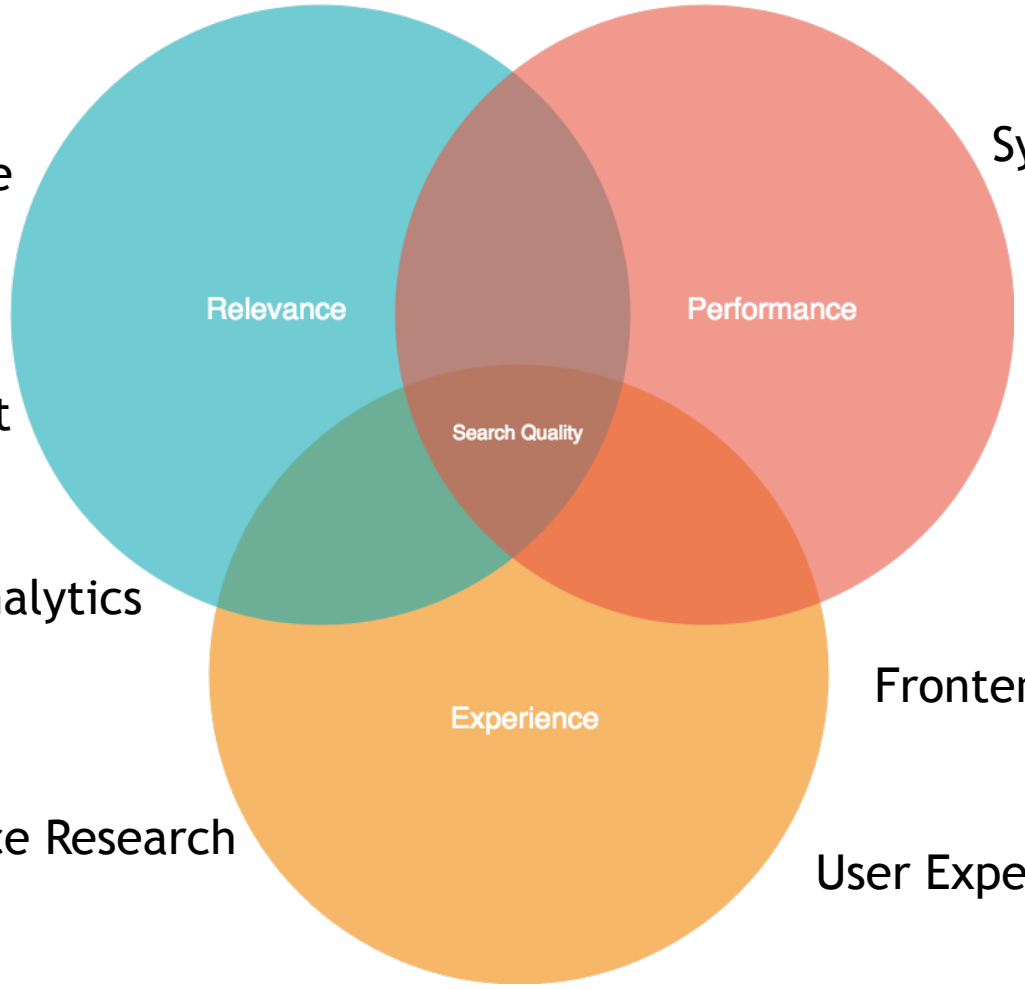
- Integrating intelligence from external tools is difficult
  - Short iterations
  - Quality evaluation
- Communication (humans & technologies)
  - Many different kinds of expertise required
  - Monolithic design & interdependencies

Search Quality

An Introduction to Search Quality (Irwin, 2018)



Kinds of expertise



Data Engineering

Data Science

System/Cloud Engineering

Backend Engineering

Search Management

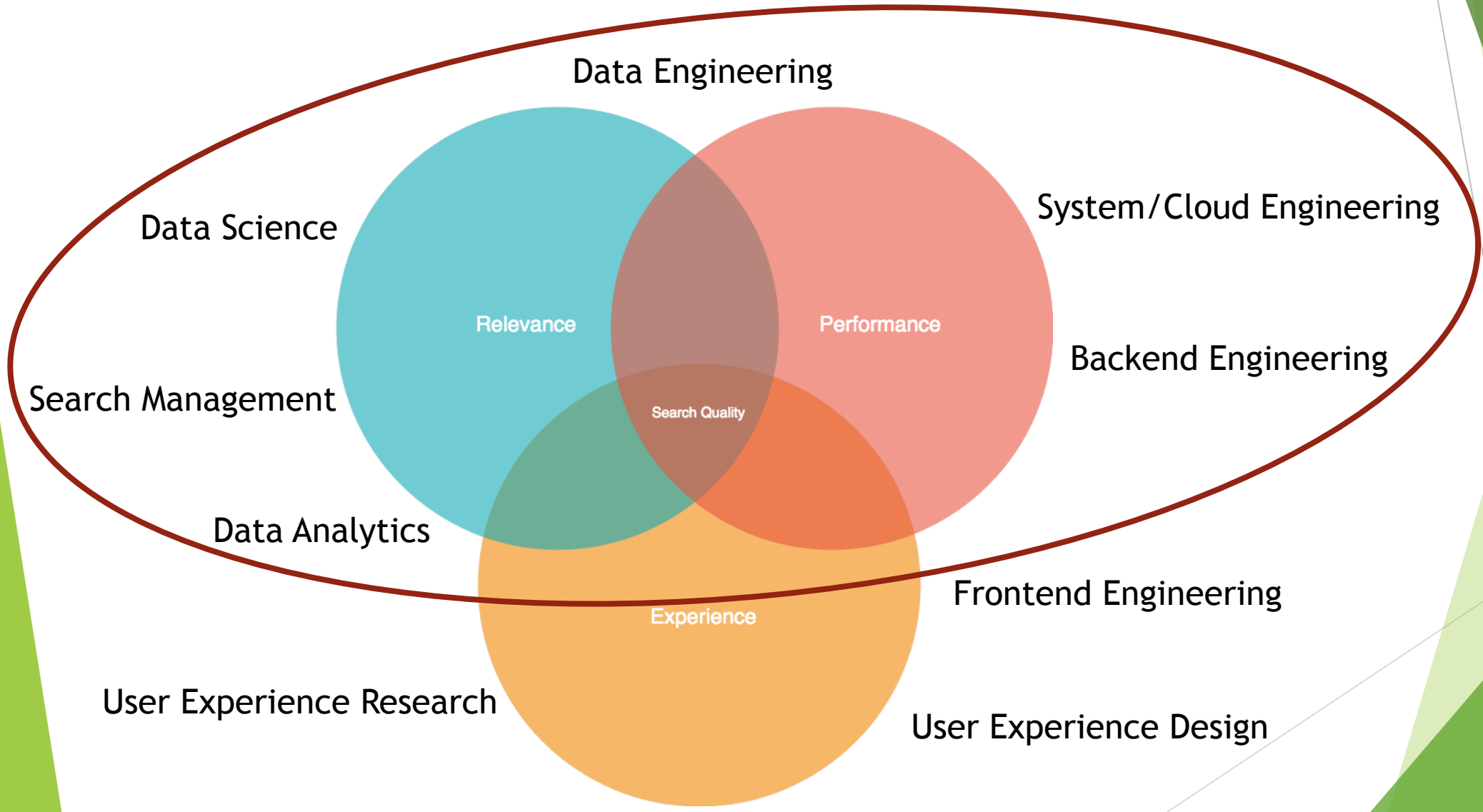
Data Analytics

Frontend Engineering

User Experience Research

User Experience Design

Kinds of expertise



Silos Are the Enemy of AI Powered Search (Turnbull, 2020)

- "[...] you have limited resources. You must make hard **tradeoffs**. [...] A minor increase in search relevance might cause performance to plummet"
- "Do your engineers **understand** your data scientists requirements early enough (and vice versa)? Or do they only realize after months of **siloed work** that they built the wrong thing?"

Silos Are the Enemy of AI Powered Search (Turnbull, 2020)

- "[...] you have limited resources. You must make hard **tradeoffs**. [...] A minor increase in search relevance might cause performance to plummet"
- "Do your engineers **understand** your data scientists requirements early enough (and vice versa)? Or do they only realize after months of **siloed work** that they built the wrong thing?"

- ⚠ Fostering end-to-end understanding is limited
- ⚠ Tradeoffs are not always desired

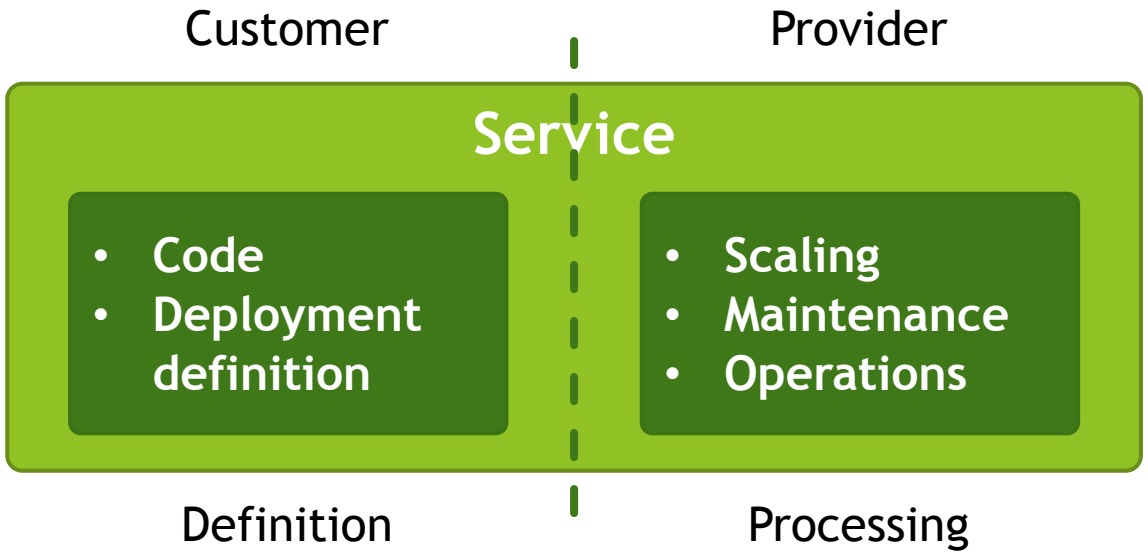
Core Questions

- How can we reduce the need for end-to-end understanding?
- How can we reduce interdependencies?
- How can we make relevance engineering organizationally scalable?

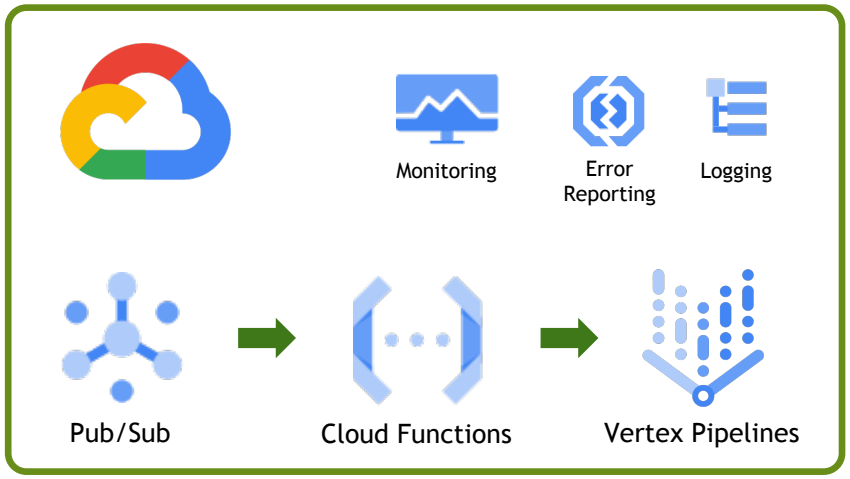
ML Ops and Serverless Technologies

What is serverless about?

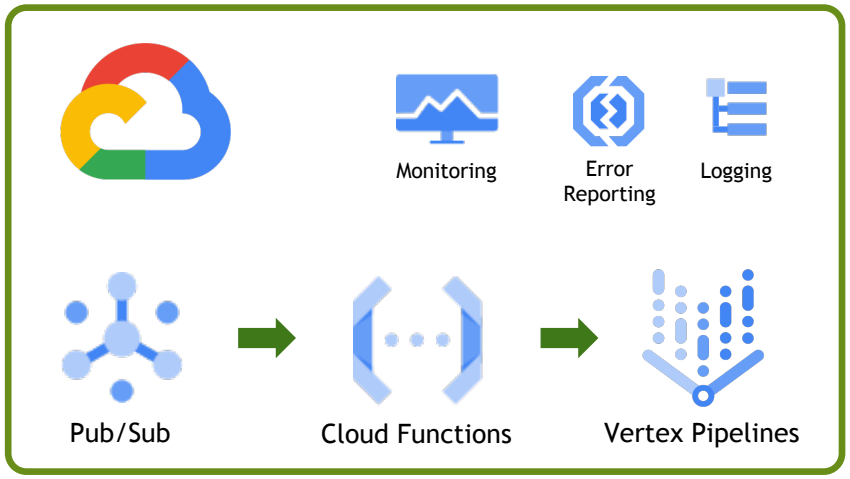
- Separating definition and processing



ML Ops


Monitoring
Error Reporting
Logging
Pub/Sub
Cloud Functions
Vertex Pipelines

ML Ops



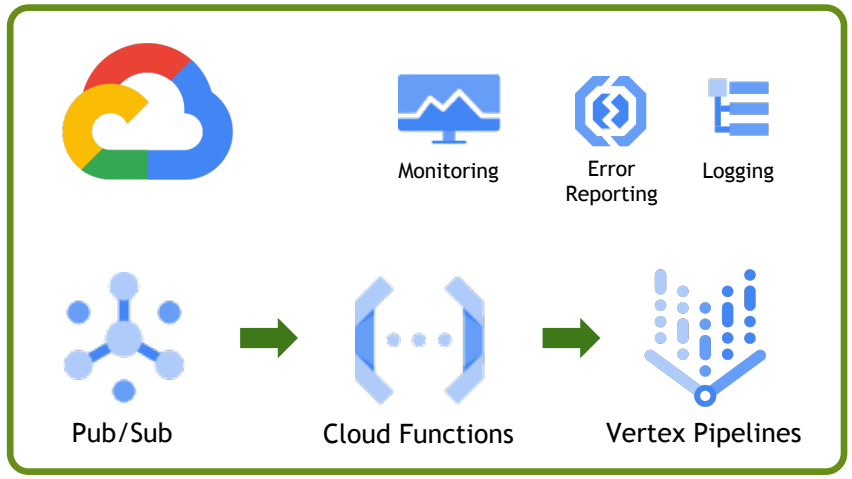
Git Repo

- Generic code
- Terraform modules
- CICD templates

ML Ops Engineer

Terraform

ML Ops



Git Repo

- Generic code
- Terraform modules
- CICD templates

ML Ops Engineer

Terraform

Git Repo

- Code
- Dockerfiles
- Pipeline definition

Data Scientist

Query Rewriting and Querqy

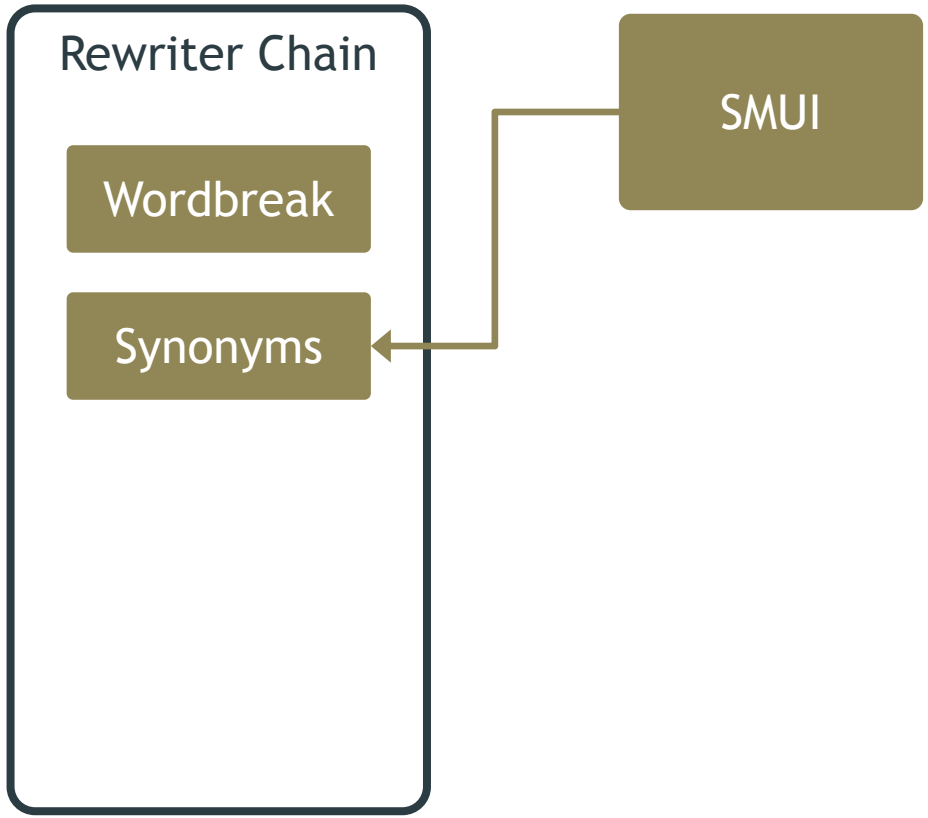
What is Query Rewriting?

```
must(apple, smartphone, case) ->
    must(
        should(
            apple,
            iphone
        ),
        should(
            smartphone,
            iphone
        ),
        case
    )
```

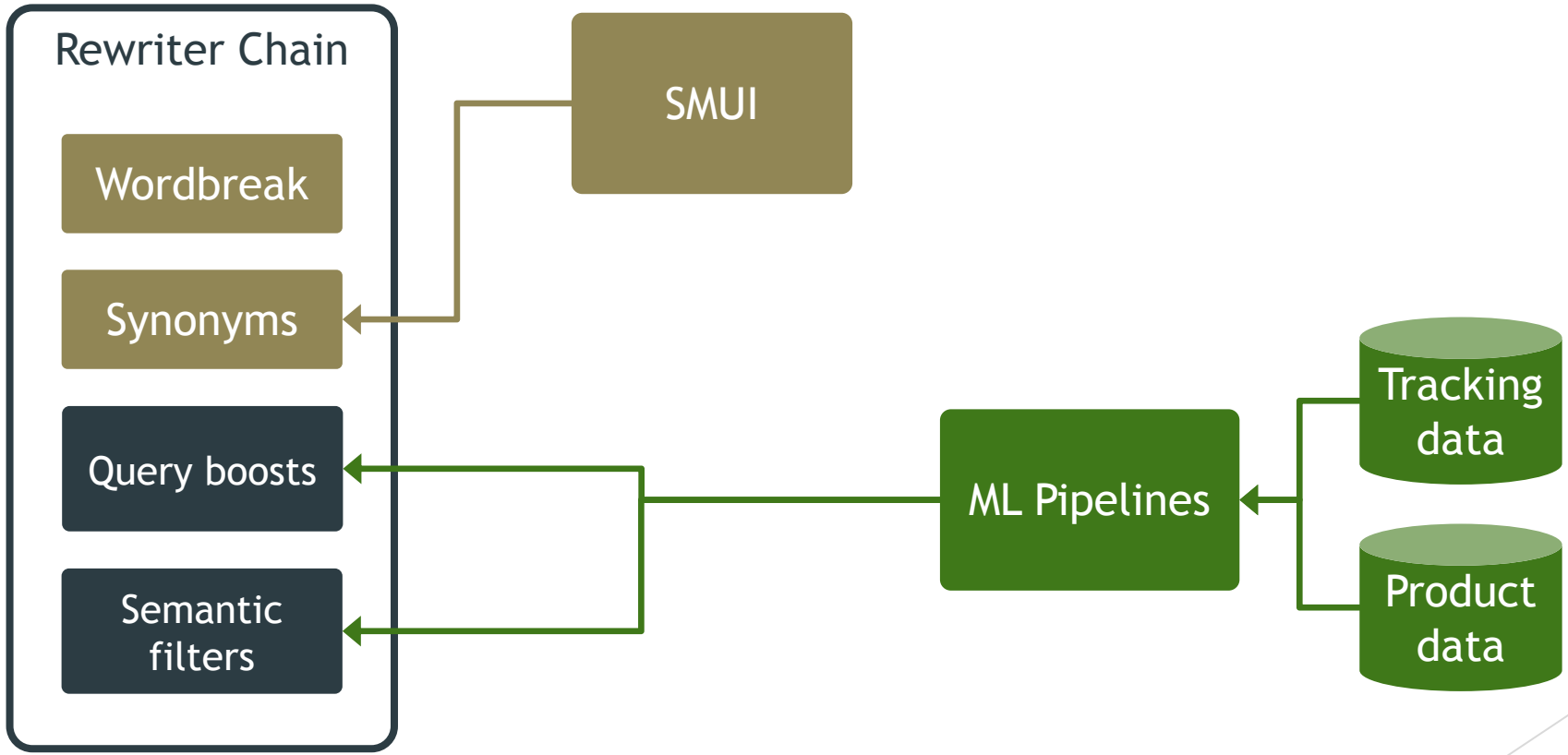
What is Querqy?

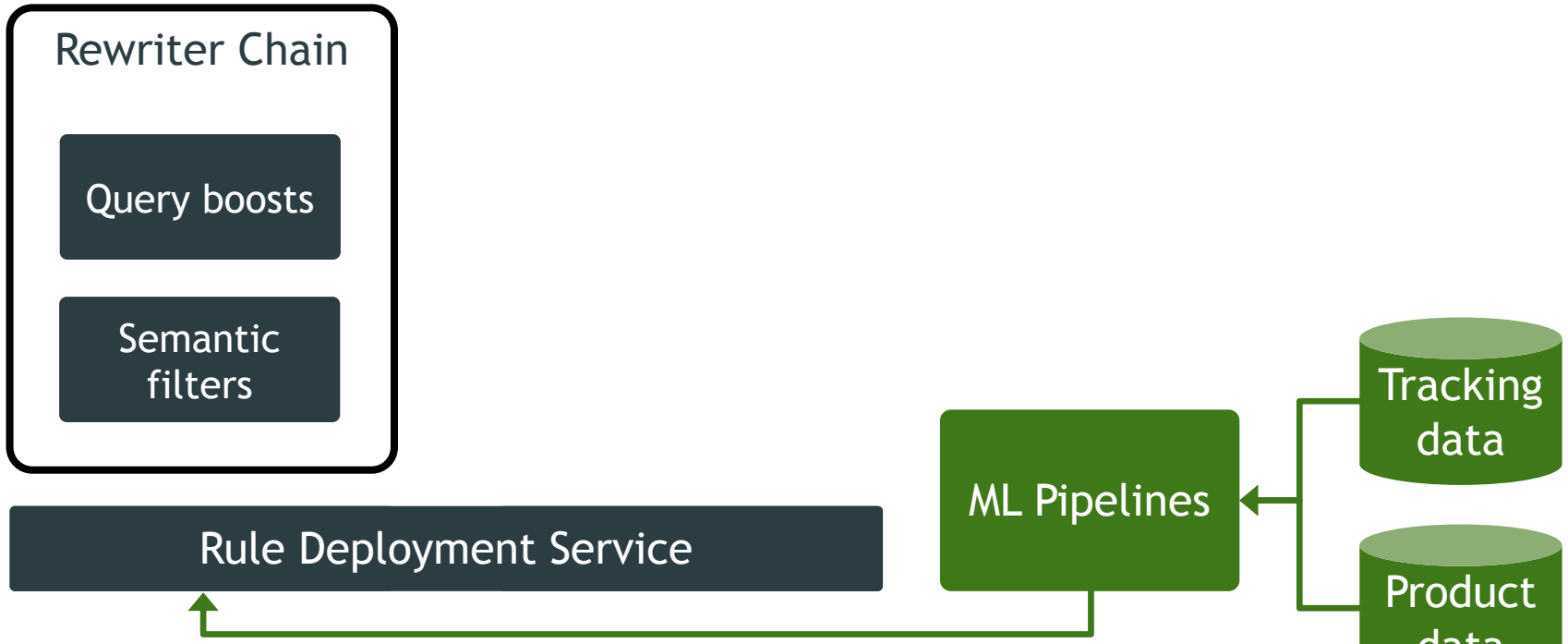
```
apple smartphone =>
    SYNONYM: iphone
    UP: brand:apple

"apple smartphone" =>
    FILTER: brand:apple
    FILTER: product_type:smartphone

smartphone AND NOT (case OR backcover) =>
    FILTER: product_type:smartphone
```

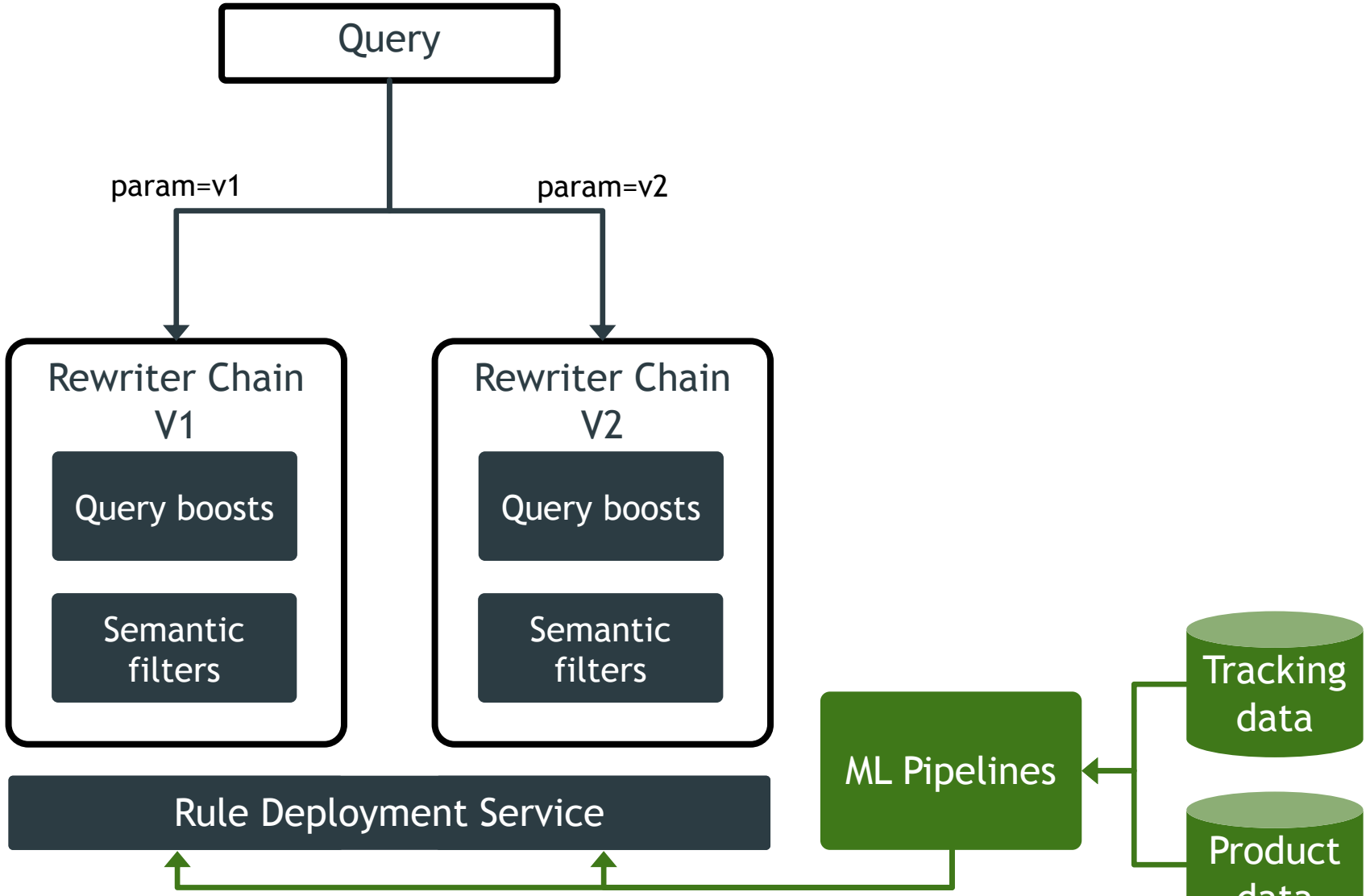
Querqy Rewriters

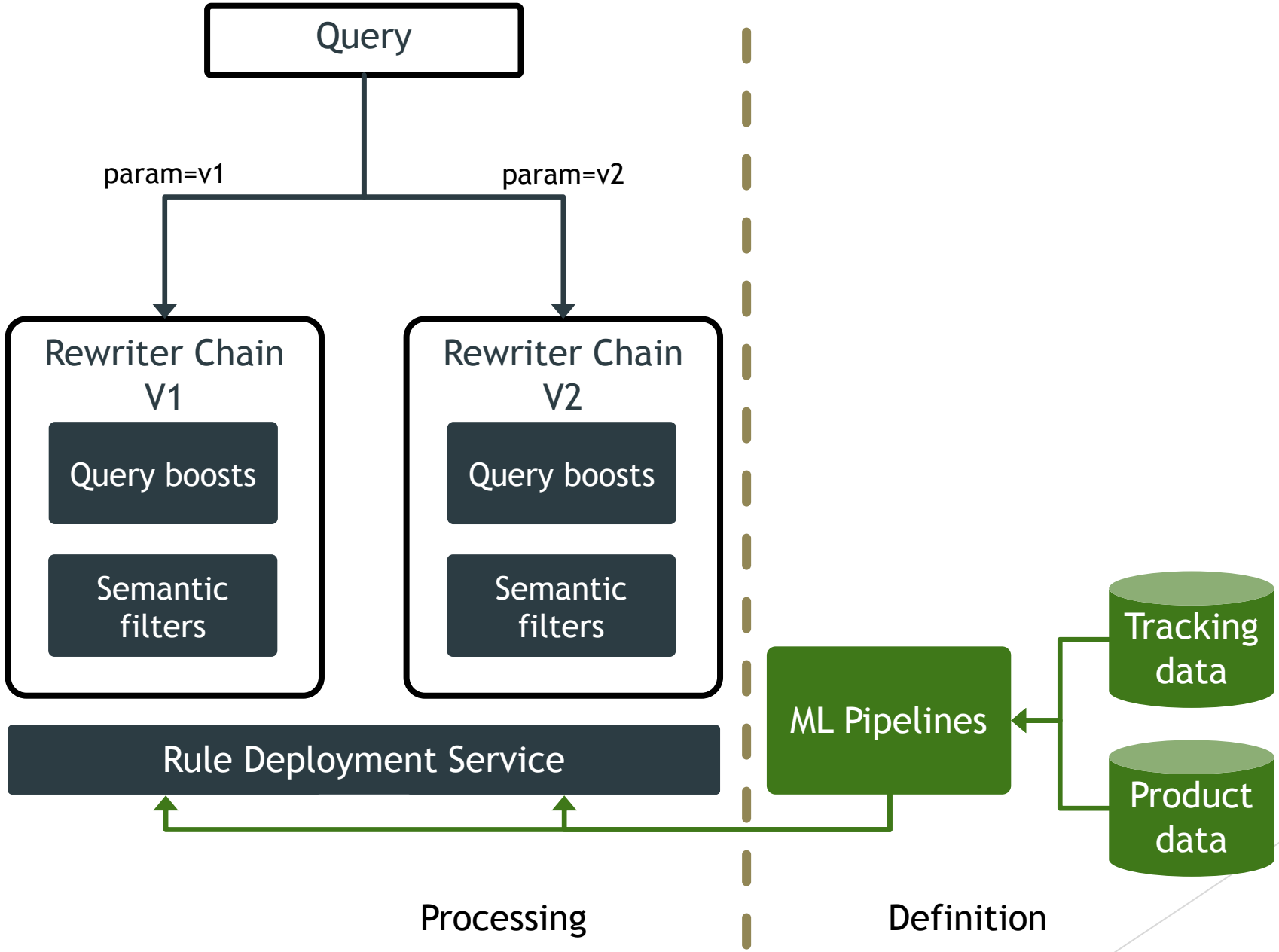


Querqy Rewriters



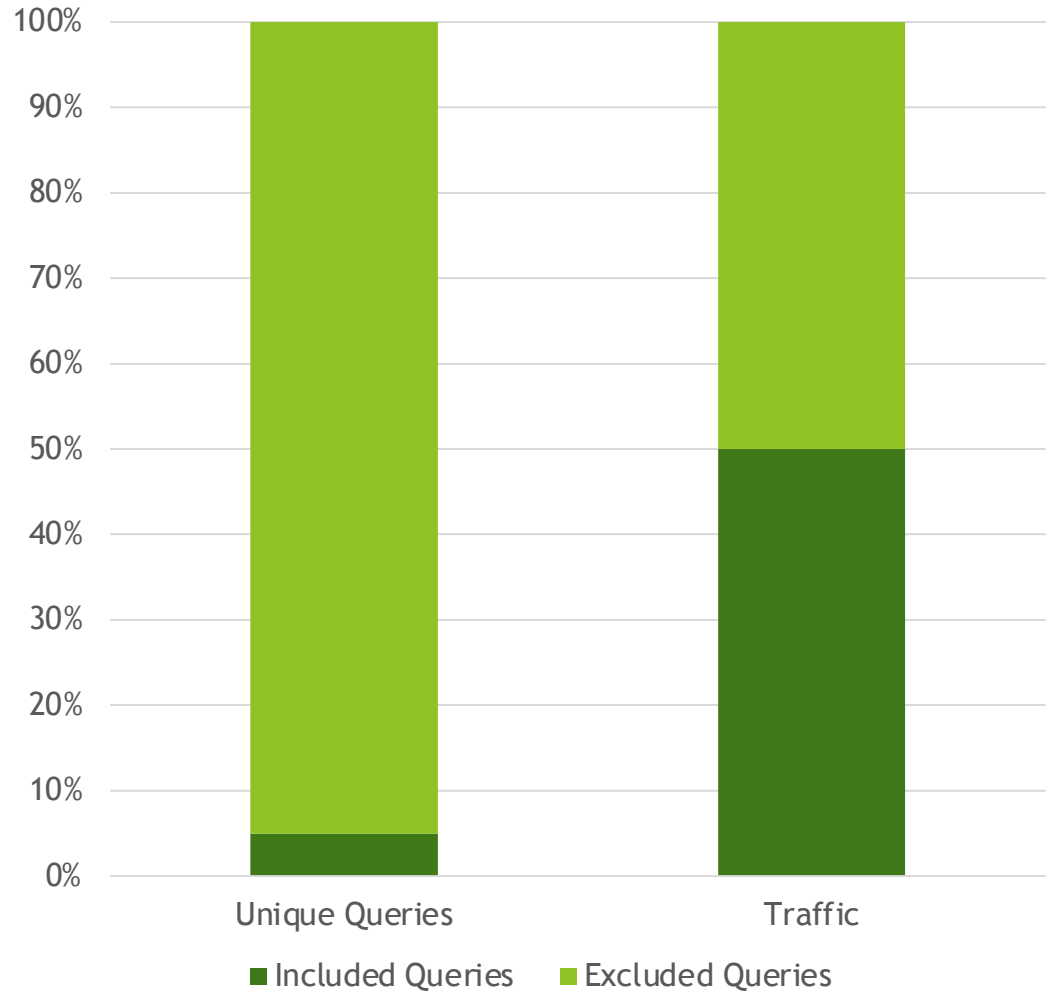

Rewriter Chain
Query boosts
Semantic filters
Rule Deployment Service
ML Pipelines
Tracking data
Product data


Query
param=v1
param=v2
Rewriter Chain V1
Query boosts
Semantic filters
Rewriter Chain V2
Query boosts
Semantic filters
Rule Deployment Service
ML Pipelines
Tracking data
Product data


Query
param=v1
param=v2
Rewriter Chain V1
Query boosts
Semantic filters
Rewriter Chain V2
Query boosts
Semantic filters
Rule Deployment Service
ML Pipelines
Tracking data
Product data
Processing
Definition

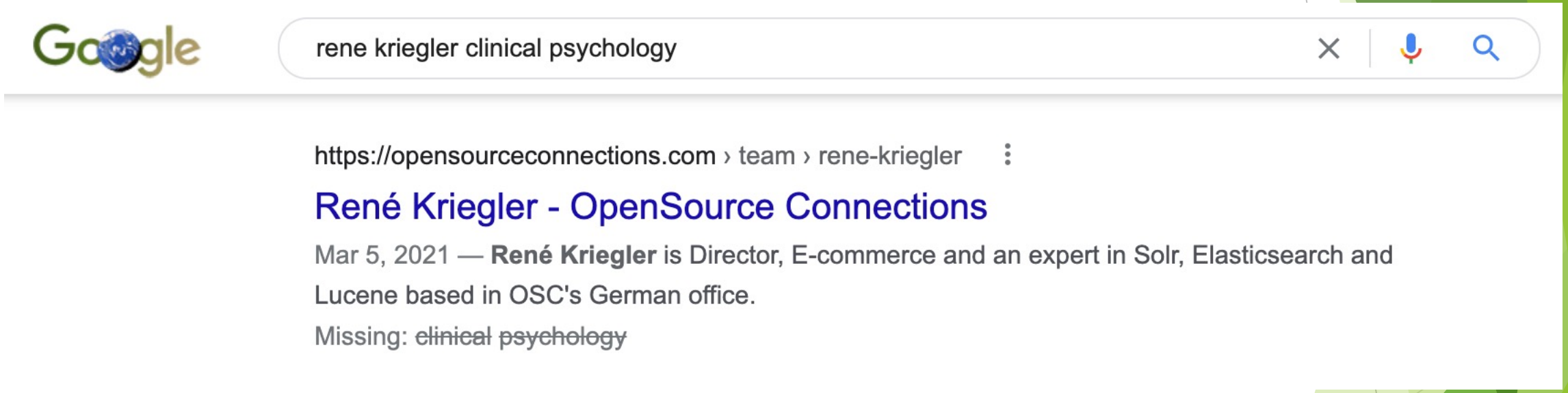
Limitations

- Queries / terms need to be known
- Only supports rewriting / pre-processing



More Advanced Rewriting

Query Relaxation



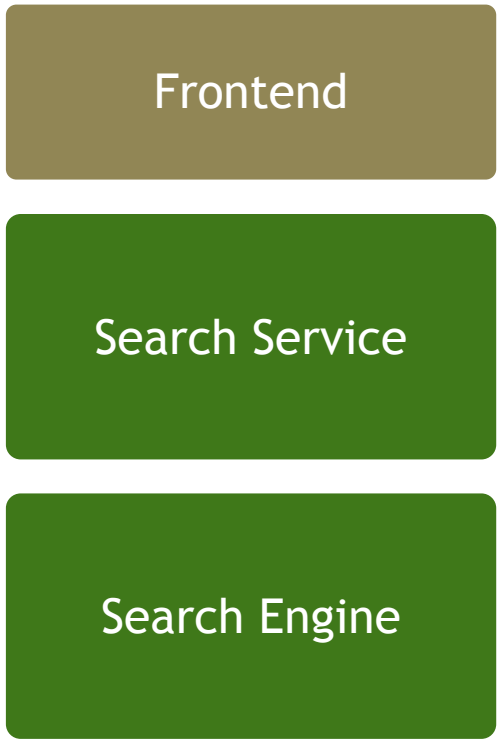
Query relaxation - a rewriting technique between search and recommendations (Kriegler, 2019)

11 - MNN / Word2vec plus wordshape

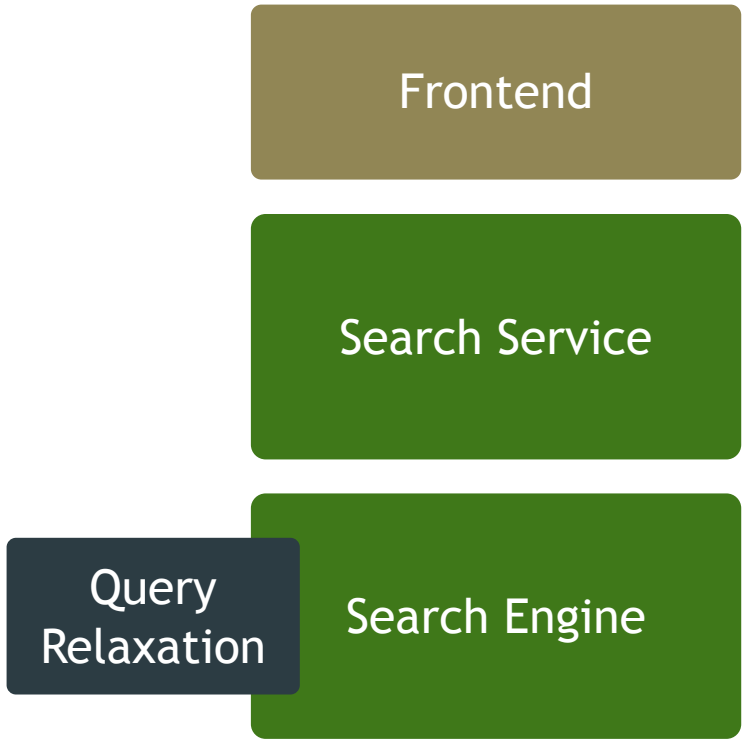
Predict the term to be dropped using a multi-layer neural network (MNN) with Word2vec embeddings and wordshape features as input.

Judgment Type	Best previously seen relaxed query						Any previously seen relaxed query					
Data set	FREQ			COOC			FREQ			COOC		
Metric	P	R	F1	P	R	F1	P	R	F1	P	R	F1
0 - Drop random term	0.46	0.46	0.46	0.46	0.46	0.46	0.61	0.61	0.61	0.47	0.47	0.47
1 - Drop shortest term	0.38	0.38	0.38	0.48	0.48	0.48	0.54	0.54	0.54	0.49	0.49	0.49
2 - Drop shortest non-alphabetical term	0.52	0.05	0.09	0.45	0.04	0.08	0.55	0.05	0.09	0.46	0.04	0.08
3 - use 2, fallback to 1	0.40	0.40	0.40	0.49	0.49	0.49	0.56	0.56	0.56	0.50	0.50	0.50
4 - Drop most frequent term	0.25	0.17	0.20	0.44	0.35	0.39	0.56	0.38	0.45	0.45	0.36	0.40
5 - Drop least frequent term	0.79	0.79	0.79	0.60	0.60	0.60	0.90	0.90	0.90	0.61	0.61	0.61
6 - Drop term with highest entropy	0.29	0.27	0.28	0.43	0.41	0.42	0.45	0.43	0.44	0.44	0.42	0.43
7 - Drop term with lowest entropy	0.32	0.32	0.32	0.29	0.29	0.29	0.46	0.46	0.46	0.30	0.30	0.30
8 - keep most similar query (Word2vec)	0.82	0.81	0.82	0.61	0.61	0.61	0.91	0.90	0.90	0.63	0.62	0.62
9 - keep most similar query ('Query2vec')	0.66	0.07	0.13	0.64	0.11	0.18	0.87	0.10	0.18	0.65	0.11	0.19
10 - MNN, W2V embeddings as input	0.85	0.85	0.85	0.68	0.68	0.68	0.90	0.90	0.90	0.69	0.69	0.69
11 - like 10, plus wordshape features	**0.87**	**0.87**	**0.87**	**0.69**	**0.69**	**0.69**	**0.93**	**0.93**	**0.93**	**0.71**	**0.71**	**0.71**

Re-Ranking

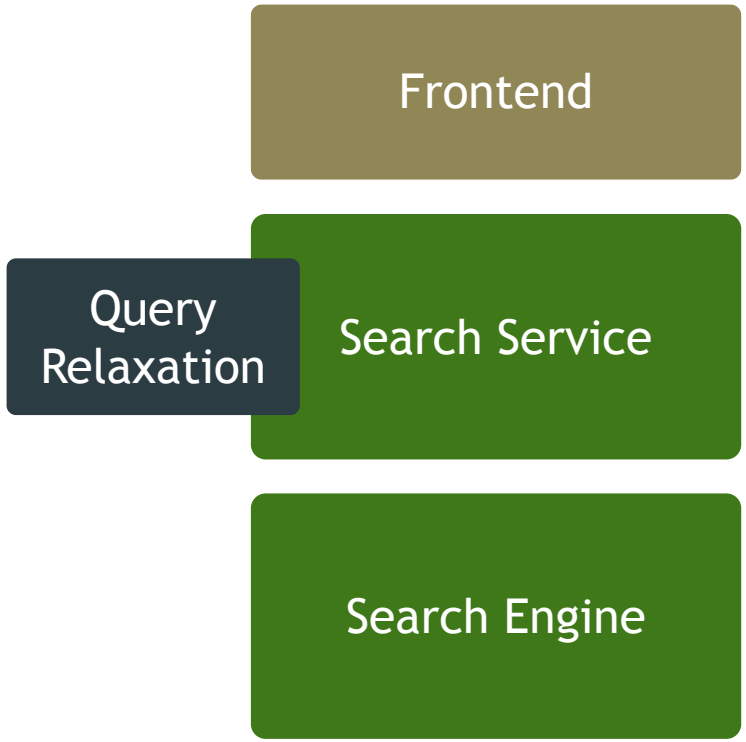


Query Relaxation



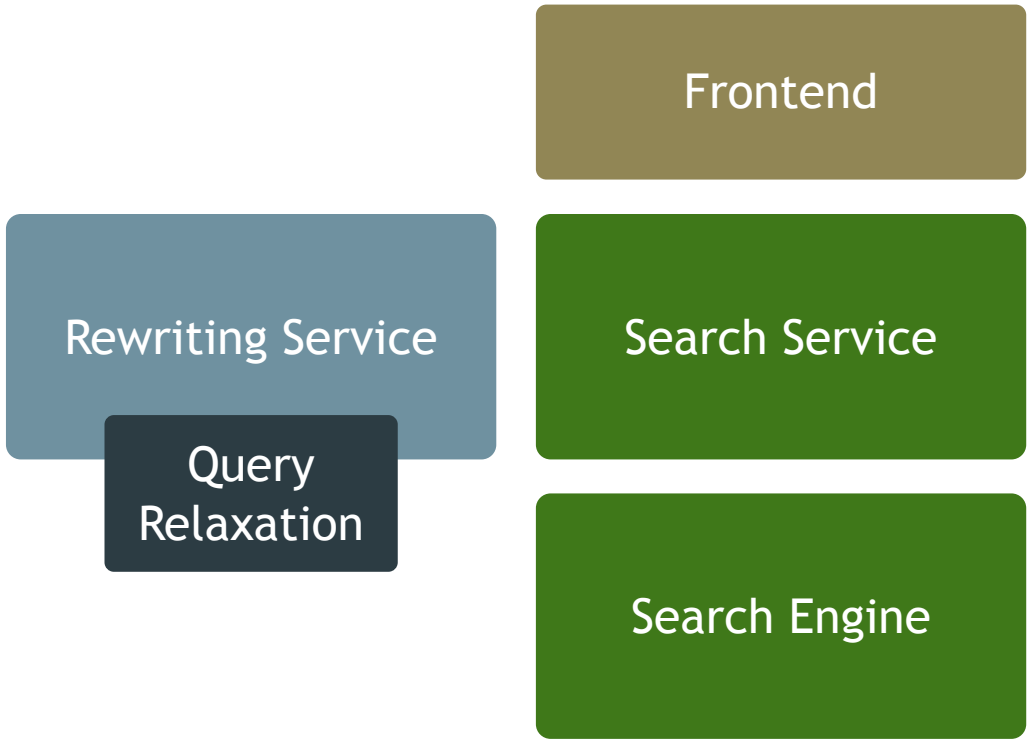
- Load and deployments interfere with indexing
- Monolithic design
- No release without search engineer

Query Relaxation



- ⚠️ Load and deployments interfere with other features
- ⚠️ Monolithic design
- ⛔ No release without search engineer

Query Relaxation

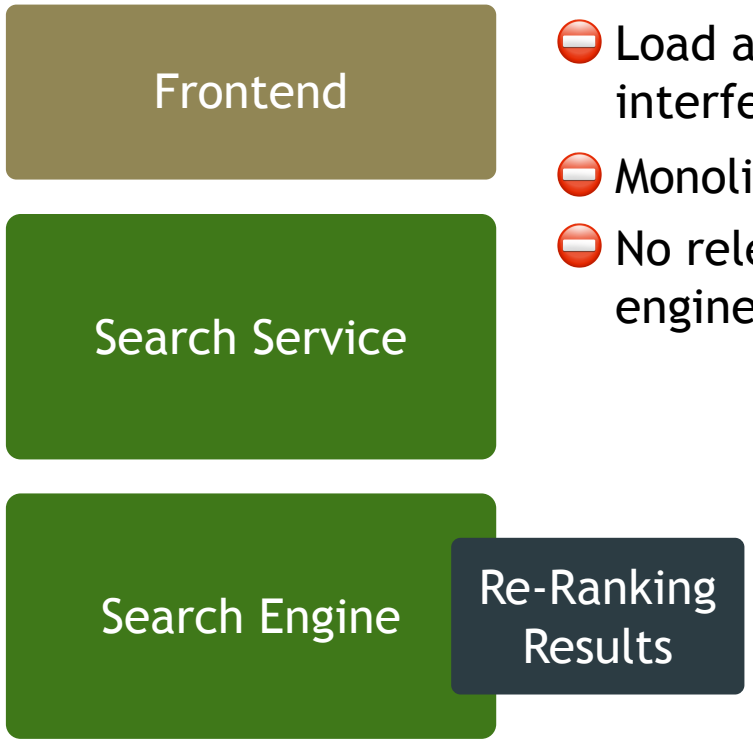


Independent development

Independent releasing

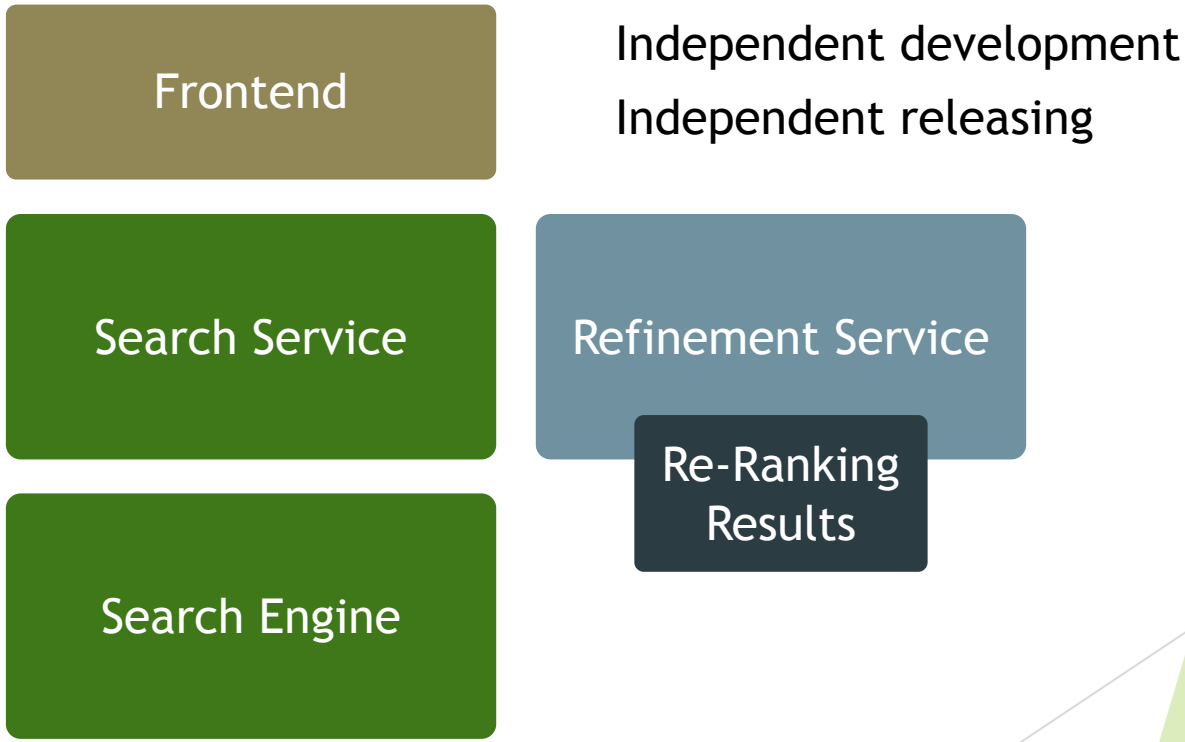
Re-Ranking

Re-Ranking



- Load and deployments interfere with indexing
- Monolithic design
- No release without search engineer

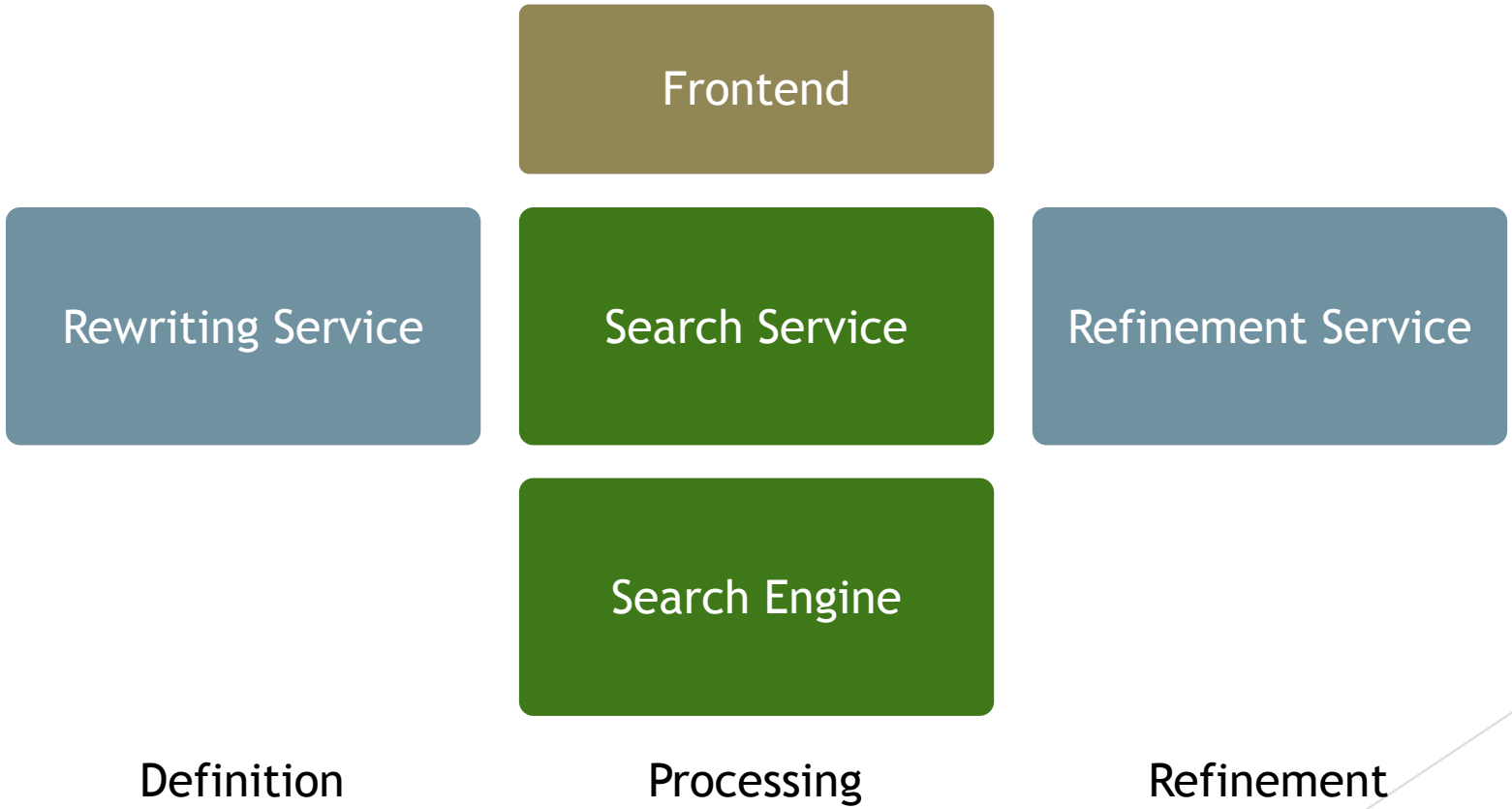
Re-Ranking



Independent development

Independent releasing

Re-Ranking


Frontend
Rewriting Service
Search Service
Refinement Service
Search Engine
Definition
Processing
Refinement

Thank you!