SpecifiedBy

Joan Figuerola Hurtado

Co-founder & CTO

@joanfihu

joan@specifiedby.com

Outline

- Crawling, structuring and storing data.
- Search engines: architectures, resources and evaluation.
- Recommendation engines: architectures, resources and evaluation.
- Q&A

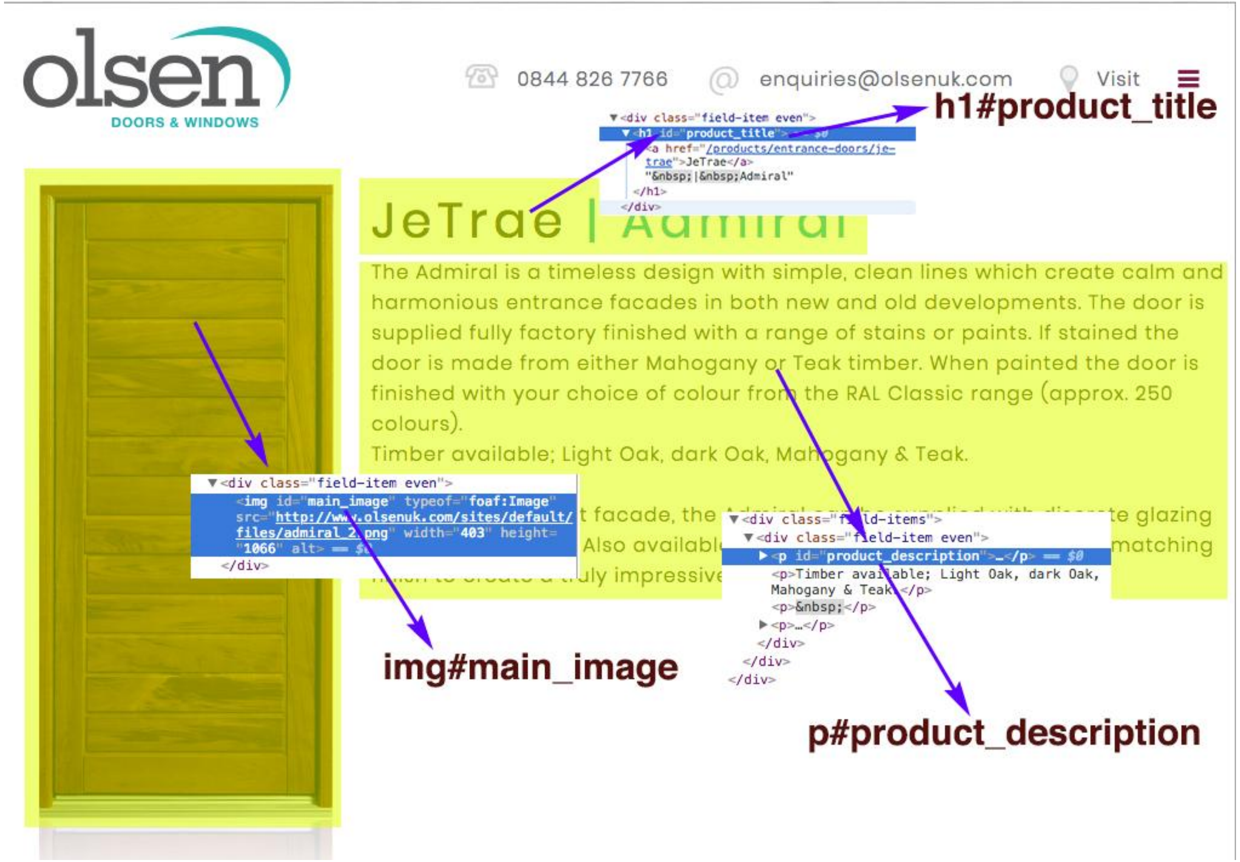
Getting Data

- Many ways to get data. Problem related.
- Scan books, manual collection, public Internet data, etc

Web Crawlers

- Scrapy: open source, Python, Non-Blocking and efficient
- Identify Target Domain and Pages
- False Dead Ends & Infinite Loops
- CSS Selectors
- Post Processing & Storage

CSS Selectors



Post Crawl Processing & Storage

Post Crawl Processing:

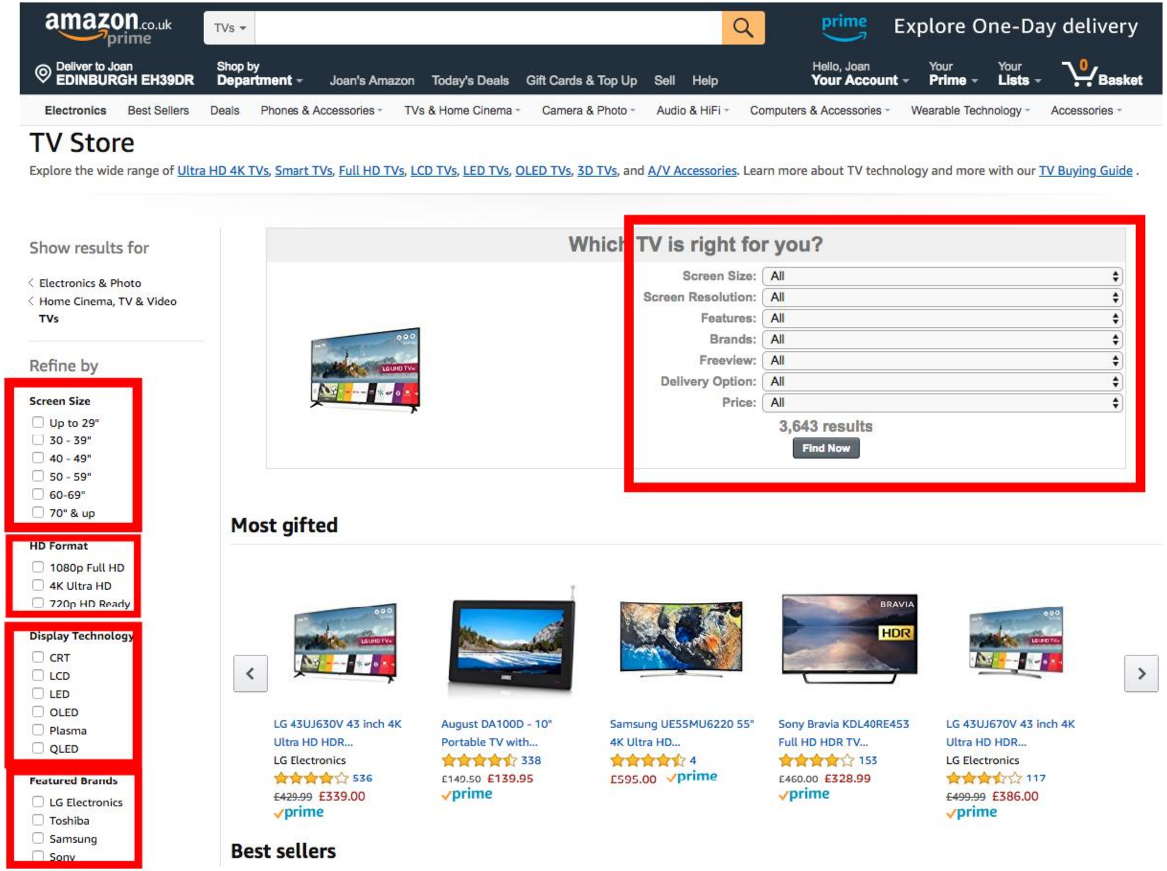
- Remove HTML tags
- Resize images
- Classify products
- Convert PDF to text
- Extract properties
- Deduplication
- Etc.

Storage:

Many built-in storage pipelines:

- MySQL
- CSV
- JSON
- S3
- XML

Search Engines



ElasticSearch & SQL

Elasticsearch is a search engine based on Lucene. It provides a distributed, multitenant-capable full-text search engine with an HTTP web interface and schema-free JSON documents.

- Document Indexing
- TFIDF Scoring by default
- Field Boosting
- Reranking
- Autocomplete
- Misspelling errors
- Related searches
- Stemming
- Geolocation searches
- Etc

ES search is good for full text searches but difficult to work with with complex queries -> ES for full text -> SQL for the rest.

Quality Score

Data rating of the listing. Not a user review.

Title:

- Malvern - Tilt & Turn (Bad)
- Malvern - Tilt & Turn Stainless Steel Security Window W: 1500mm **(Good)**

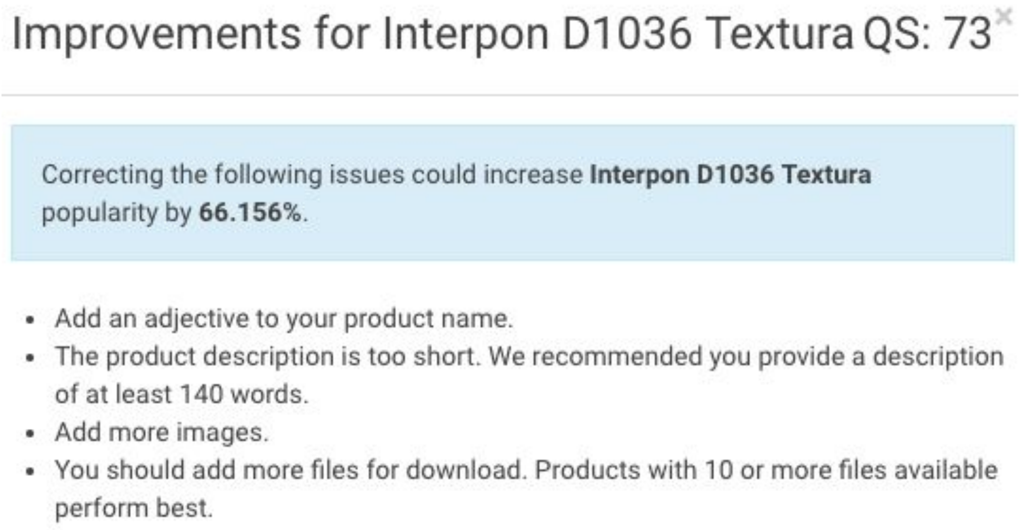
Images:

- High Resolution (Good)
- > 3 images (Good)

Files:

- Has Files (Good)
- > 1 files (Good)
- Has 3D model (Good)

QS(P) = St*Wt + Si*Wf + Sf*Wf + … + Sx*Wx



Improvements for Interpon D1036 Textura QS: 73

Correcting the following issues could increase **Interpon D1036 Textura** popularity by **66.156%**.

- Add an adjective to your product name.
- The product description is too short. We recommended you provide a description of at least 140 words.
- Add more images.
- You should add more files for download. Products with 10 or more files available perform best.

Popularity Score & Reranking

Measures how much users have engaged with a product.

- User A **clicks** on Product AA
- User A **downloads** a file from Product AA
- User B **clicks** on Product AA
- User B **compares** Product AA with Product BB

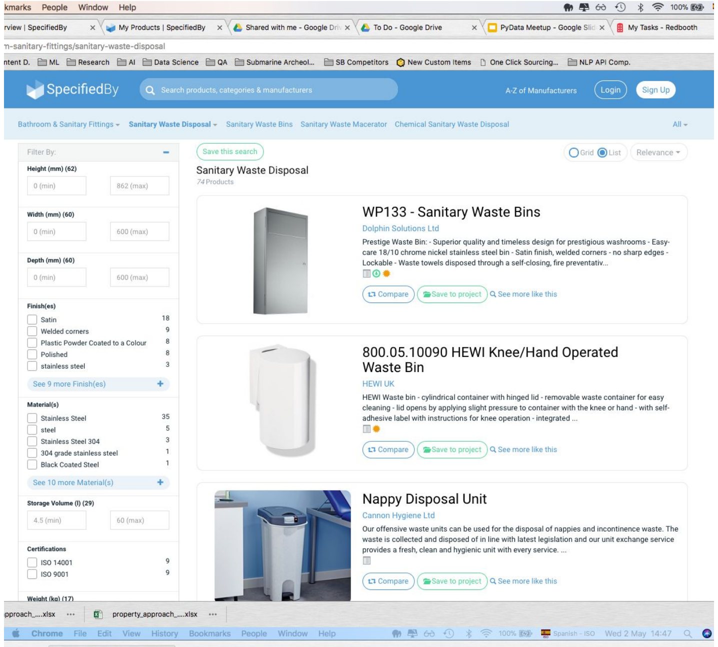
Ps(A) = #clicks*Wc + #downloads*Wd + #comparisons*Wc

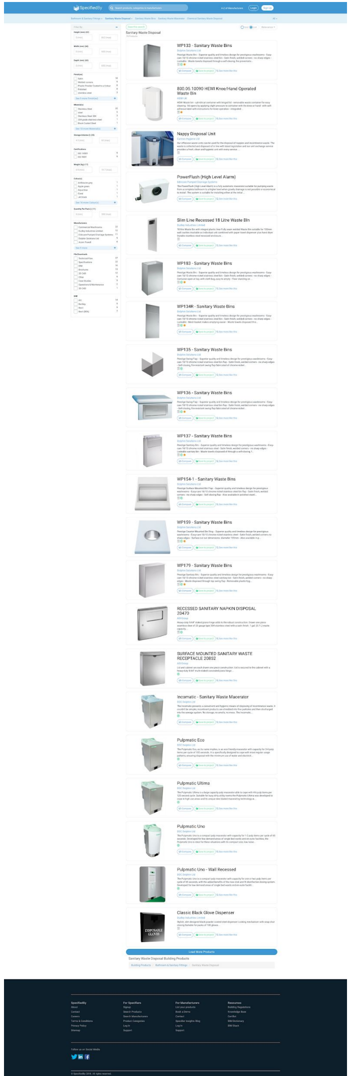
Reranking:

S(query, document) = tfIdf*Wtfidf + quality_score*Wqs + popularity_score*Wps

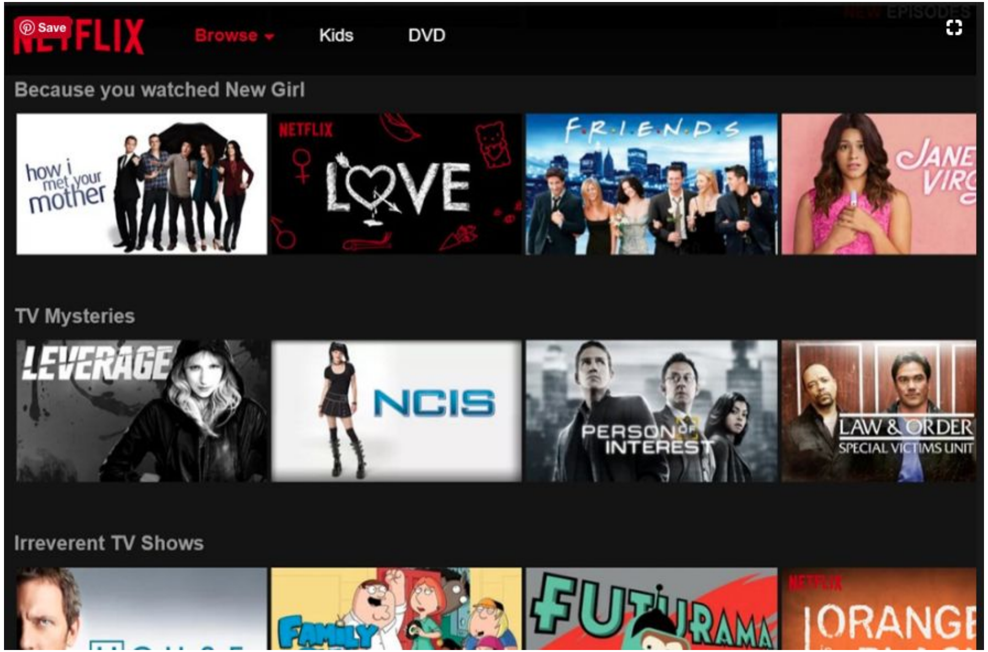
Metrics

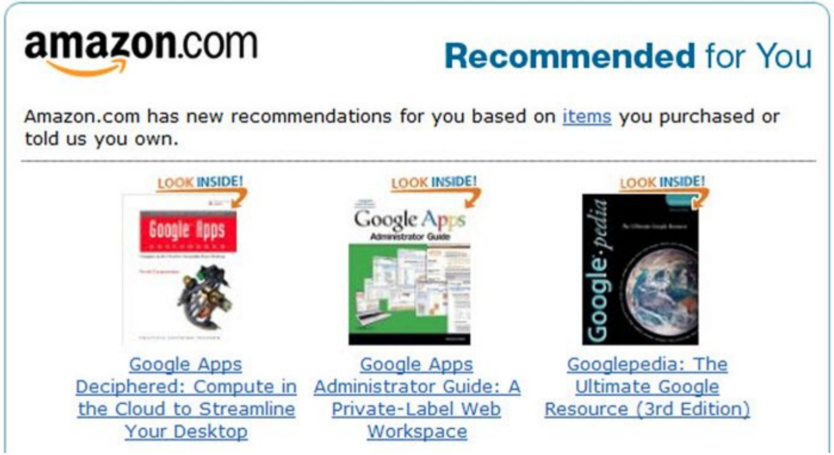
Search Click-Through-Rate(CTR): (clicks / searches) * 100
Product Click-Through-Rate(CTR): (clicks / impressions) * 100
Average Click Position: SUM(click_positions) / clicks
Viewability





Recommendations Engines





Content Based Filtering

Features: text, category, images, sound, etc
Vectorize/Embeddings: Word2Vec, CNN, Binarize Labels, etc
Calculate Similarity: cosine, euclidean, etc

	Item 1	**Item 2**	**Item 3**	**Item 4**
Item 1	1	0.8	0.1	0.3
Item 2	0.8	1	0.6	0.5
Item 3	0.1	0.6	1	0.9
Item 4	0.3	0.5	0.9	1

Collaborative Filtering

Sparks serendipity (Wow! factor)
Features: ratings from users to items. Explicit (i.e star reviews) and implicit (i.e clicks) ratings

	Item 1	Item 2	Item 3	Item 4
User 1	5	3	-	1
User 2	4	-	-	1
User 3	1	1	-	5
User 4	1	-	-	4
User 5	-	1	5	4

Matrix Factorisation

	Item 1	Item 2	Item 3	Item 4
User 1	4.97	2.98	**2.18**	0.98
User 2	3.97	**2.40**	**4.59**	0.99
User 3	1.02	0.93	**5.32**	4.93
User 4	1	**0.85**	**4.59**	3.93
User 5	1.36	1.07	4.89	4.12

Resources

- https://www.coursera.org/specializations/recommender-systems
- http://www.quuxlabs.com/blog/2010/09/matrix-factorization-a-simple-tutorial-and-implementation-in-python/
- http://lenskit.org/
- https://grouplens.org/
- http://www.mmds.org/
- https://spark.apache.org/docs/2.2.0/mllib-collaborative-filtering.html
- http://scikit-learn.org/
- https://radimrehurek.com/gensim/
- https://github.com/tensorflow/models

Attention Span For Personalisation

Time spend engaging with an item. More expressive than clicks/taps.

https://arxiv.org/abs/1608.00147

Cornell University Library

We g

arXiv.org > cs > arXiv:1608.00147

Search or Article

(Help | Advanced sea

Computer Science > Information Retrieval

Attention Span For Personalisation

Joan Figuerola Hurtado

(Submitted on 30 Jul 2016)

A click on an item is arguably the most widely used feature in recommender systems. However, a click is one out of 174 events a browser can trigger. This paper presents a framework to effectively collect and store data from event streams. A set of mining methods is provided to extract user engagement features such as: attention span, scrolling depth and visible impressions. In this work, we present an experiment where recommendations based on attention span drove 340% higher click-through-rate than clicks.

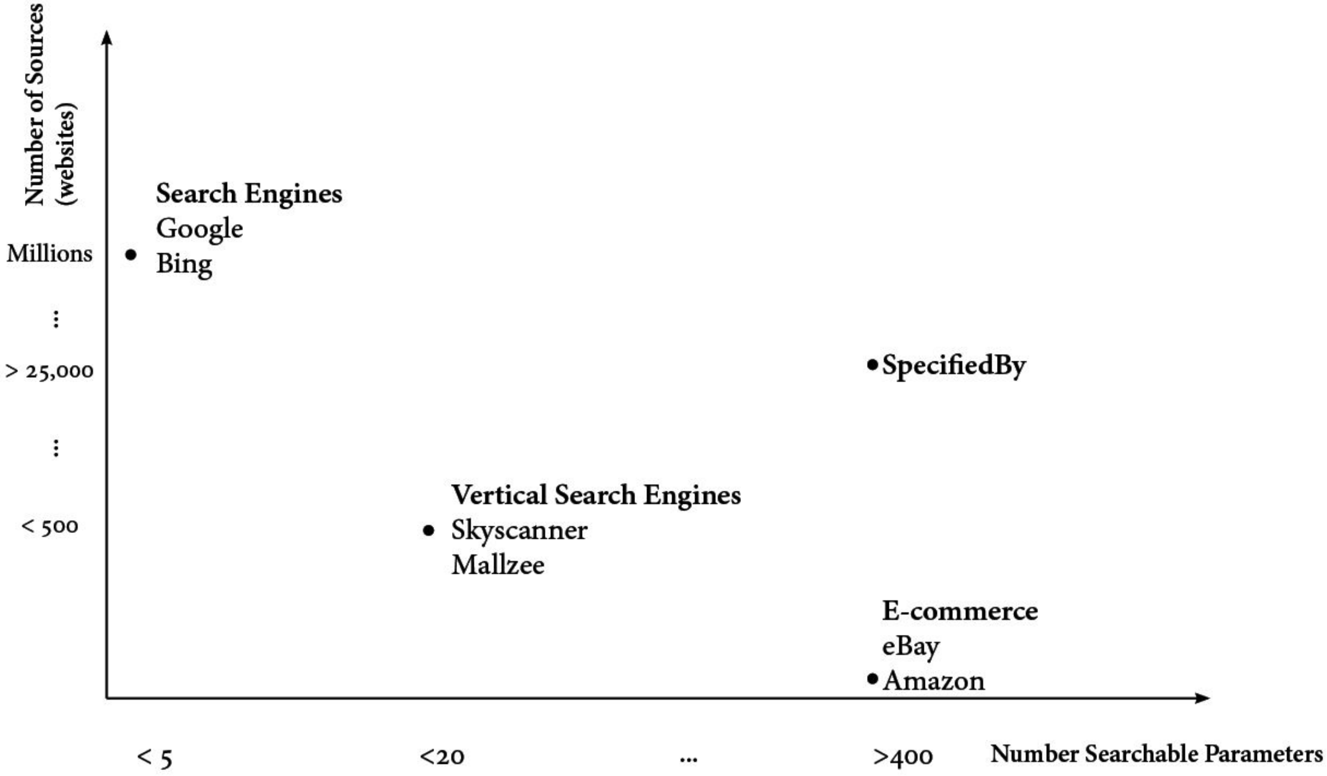
Subjects: **Information Retrieval (cs.IR)**
Cite as: arXiv:1608.00147 [cs.IR]
(or arXiv:1608.00147v1 [cs.IR] for this version)

Submission history

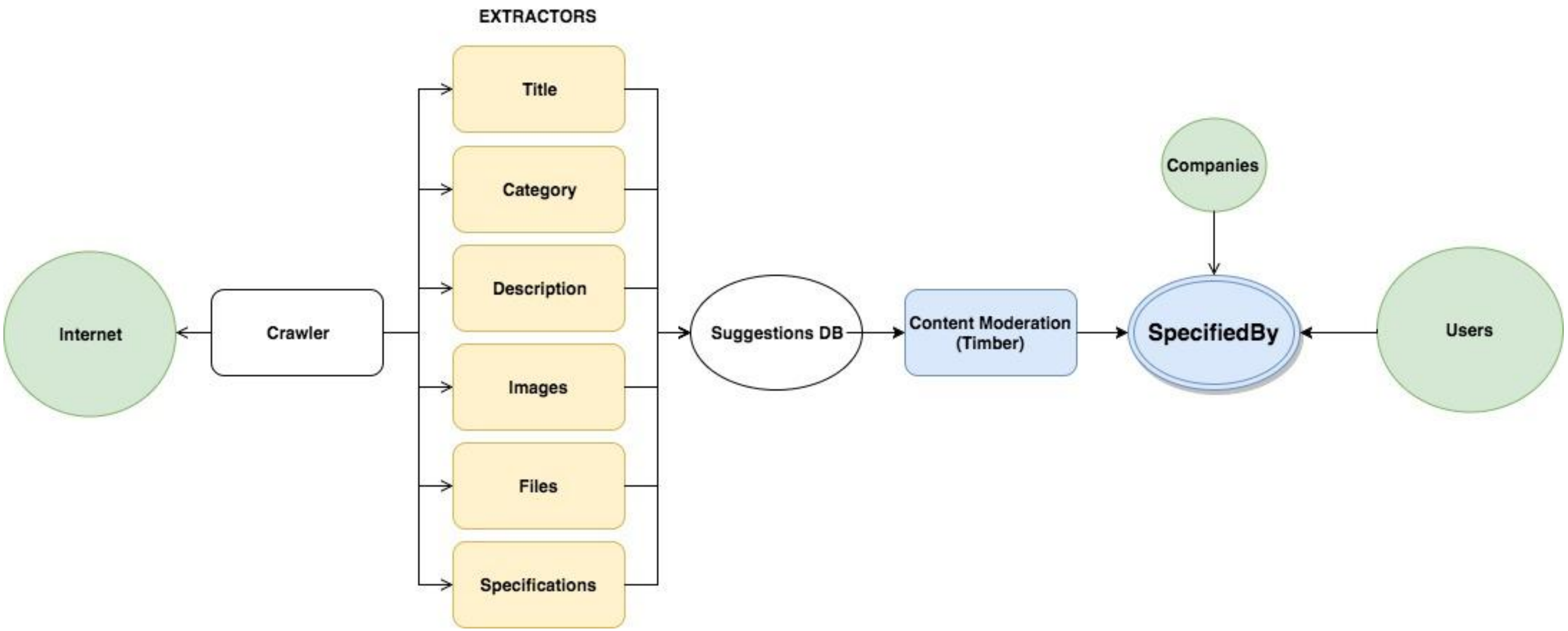
From: Joan Figuerola Hurtado [view email]
[v1] Sat, 30 Jul 2016 16:53:05 GMT (470kb)

Thank you.

Challenge


Number of Sources
(websites)
Millions
> 25,000
< 500
Search Engines
Google
Bing
SpecifiedBy
Vertical Search Engines
Skyscanner
Mallzee
E-commerce
eBay
Amazon
< 5
<20
...
>400
Number Searchable Parameters

Solution


EXTRACTORS
Title
Category
Description
Images
Files
Specifications
Internet
Crawler
Suggestions DB
Content Moderation
(Timber)
Companies
SpecifiedBy
Users