LATINOWARE 2018

BUSCAS TEXTUAIS RELEVANTES E RÁPIDAS COM ELASTICSEARCH

AUGUSTO XAVIER

LATINOWARE 2018
XV Congresso Latino-americano de Software Livre e Tecnologias Abertas
17 a 19 de outubro de 2018
Centro de Convenções de Foz do Iguaçu - CECONFI
Foz do Iguaçu | Paraná | Brasil
www.latinoware.org

QUEM SOU EU?

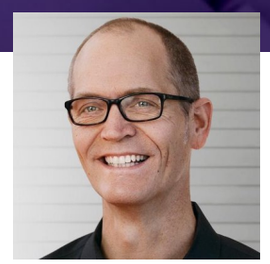
AUGUSTO CESAR BATISTA XAVIER

- MINISTÉRIO PUBLICO DE GOIÁS
- tripifyapp.com (co-fundador)
- memorialvivo.com.br (criador)
- Nerd e Pai de 3

ELASTICSEARCH

ORIENTADO A DOCUMENTOS

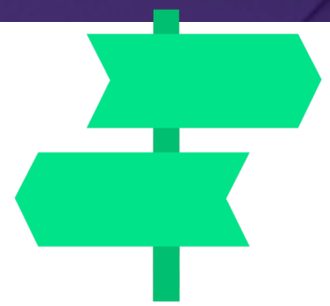
LUCENE



- 1999: Doug Cutting publica o Lucene
- 2001: Lucene entra para a Apache Software Fundation
- 2004: Shay Banon cria o Compass, em cima do Lucene
- 2010: Shay Banon publica a primeira versão do Elastic Search, sucessor do Compass

QUEM USA?

CLIENTES



NETFLIX

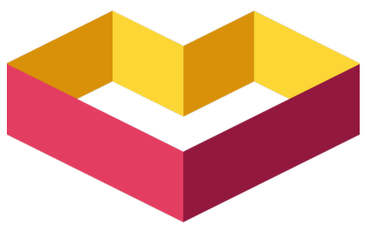
Adobe

Quora

GitHub

vimeo

FOURSQUARE



pixabay

PRINCIPAIS SOLUÇÕES CONCORRENTES

POR QUE?

ELASTIC

- FÁCIL INSTALAÇÃO
- FÁCIL CONFIGURAÇÃO
- GRANDE PODER COM POUCOS RECURSOS
- FLEXÍVEL
- Funciona com WebServices com respostas em json

SUPORTADAS PELO CORE TIME

Java

C#

Perl

python

JavaScript

php

MINISTÉRIO PÚBLICO DO ESTADO DE GOIÁS

- Migração de um sistema legado RoR para uma nova versão
- Protocolo de comunicação Legado com AMF (para Flex)
- Novo cenário: Frontend AngularJS + WS Rest + Json
- Desafio: Migração para o novo sistema mantendo os 2 simultaneamente com os mesmos dados
- Solução: Elasticsearch
- Surpresa: Desempenho e Relevância

TRIPIFYAPP.COM

- 3 Tabelas críticas: 1mi, 4mi e 11 milhões de registros.
- Algumas consultas levavam mais de 2 minutos
- Resultados pouco relevantes
- Algumas consultas dependiam de pesquisas em provedores externos, especialmente as geolocalizadas
- Desafio: melhorar o tempo de resposta
- Surpresa: Deixamos de usar WS terceiros e passamos a controlar as regras de relevância

WINDOWS

- Baixar MSI com Wizard
- Baixar ZIP
- https://www.elastic.co/guide/en/elasticsearch/reference/current/windows.html
-

MAC

- $ brew install elasticsearch
- Baixar ZIP

LINUX

- RPM (Red Hat, Centos, SLES, OpenSuSE e outros sistemas baseados em RPM)
- DEB (Debian, Ubuntu e outros sistemas baseados em Debian)

ARQUIVO YML

- https://gist.githubusercontent.com/zsprackett/8546403/raw/23b8cfe1ab08ff0c41ca795c677f5257451a6479/elasticsearch.yml
- Boa configuração padrão e simples alteração, largamente documentado

PADRÃO

PROTOCOLO REST (REPRESENTATIONAL STATE TRANSFER)

- Métodos HTTP (GET, PUT, POST, DELETE) com json
- Fácil implementação para linguagens "não suportadas”
- Protocolo muito usado e baixa curva de aprendizado

REQUISITOS

- SISTEMA DE ARMAZENAMENTO DE ARQUIVOS DIVERSOS (ODT, DOC, DOCX, JPEG, PNG, TXT, PDF)
- INDEXAR ARQUIVOS QUE SEJAM ENVIADOS VIA UPLOAD
- BUSCAR, IGNORANDO OS ACENTOS
- Para as imagens, fazer OCR

FERRAMENTAS

- RUBY ON RAILS (paperclip e elasticsearch-model)
- TESSERACT - OCR
- TIKA para extração de texto (HTML, XML, MICROSOFT OFFICE, ODT, PDF, IMAGE, EPUB, RTF, CHM, MP4, MP3, FLV, ETC)

FERRAMENTAS

Github

AUGUSTO CÉSAR BATISTA XAVIER

augustocbx@gmail.com
http://github.com/augustocbx
@augustocbx
(linkedin, face, twitter, insta)

https://tripifyapp.com

https://memorialvivo.com.br