Open Source as a Business

Strategy, Struggle & Success

Philipp Krenn

@xeraa

elastic

[...] allow software to be freely used, modified, and shared.

https://opensource.org/licenses

elastic

Who is
using OSS?

Who is
opening issues?

Who is
contributing back?

elastic

https://twitter.com/
geowolf/status/
9718113468232221248
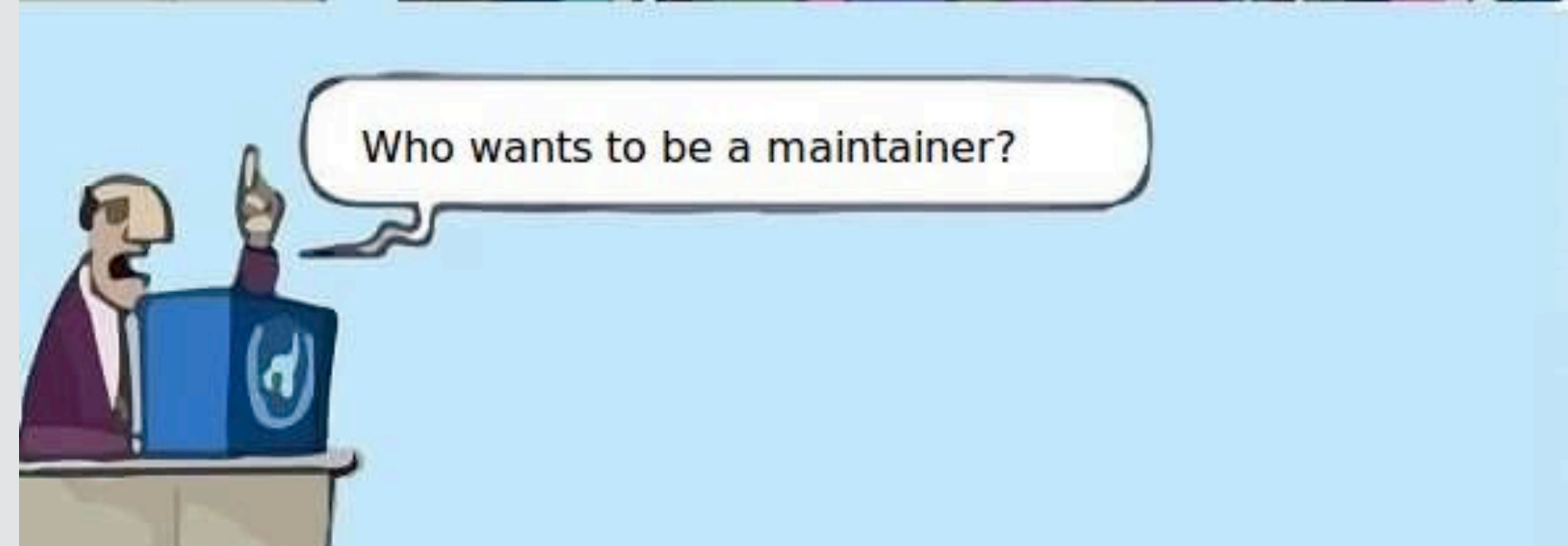
Ninety-eight-point-five percent of the code ever put into the core of Puppet was put in there by somebody I paid.

https://www.geekwire.com/2018/might-time-big-cloud-share-wealth-open-source-startups/

elastic

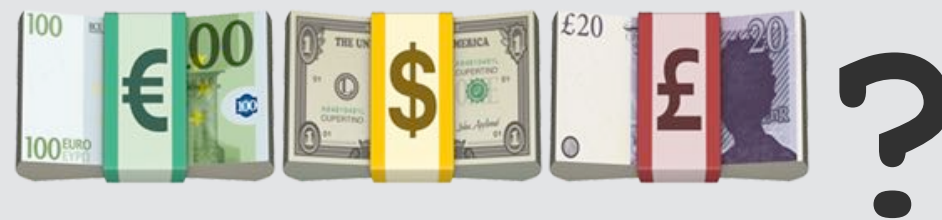# Open Source ≠ Business Model

elastic

elastic

Developer 🥑

elastic

elasticsearch

kibana

beats

logstash

Where is the 💶💵💷?

elastic

Agenda

Strategy

Struggle

Success

elastic

Strategy

Services

Support, Consulting, Training, Certification

elastic

CANONICAL

64 Quarters of Revenue Growth

$772

Great Recession

$19

Q103 → Q418

(In millions, fiscal years)

https://investors.redhat.com/~/media/Files/R/Red-Hat-IR/documents/q418-fact-sheet.pdf
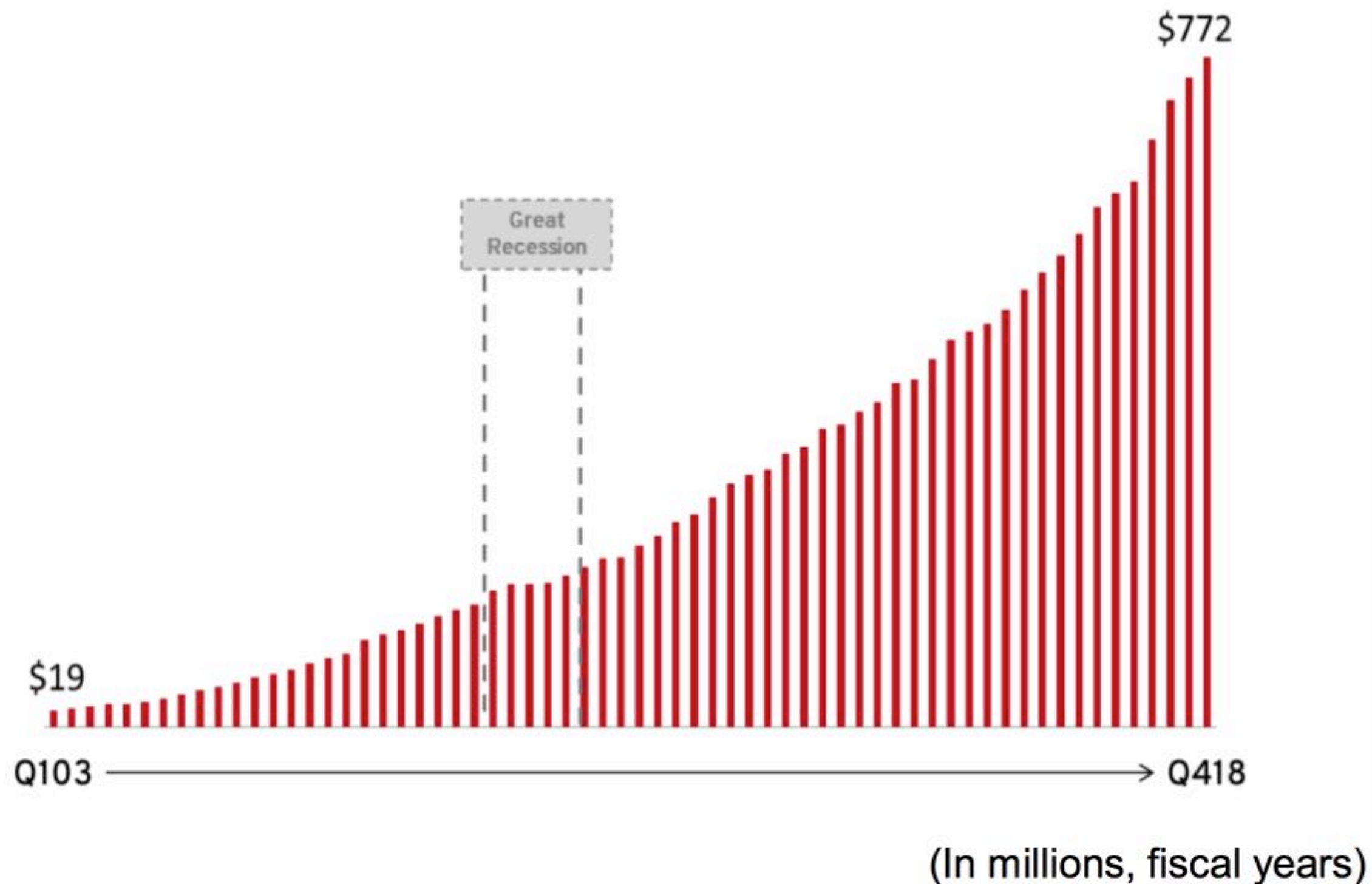
Support Problem

Ease of Use

Support Problem

Renewal Rates

elastic

Service & Training Problem

Consulting-Only Competition

Open Core

elastic

MAPR

Problem

Competing Tools

elastic

Problem

Cloud Providers

elastic

Problem

Lock-In

elastic
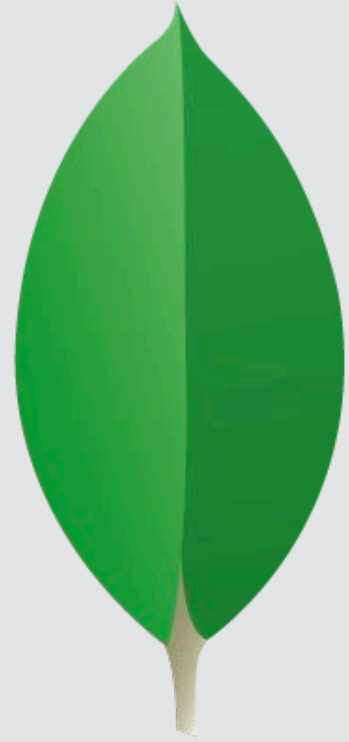
Cloud Service

elastic

WORDPRESS

Problem

Cloud Providers

The intent of open source software was not so that someone else can take the exact same software and offer it as a service.

https://www.geekwire.com/2018/might-time-big-cloud-share-wealth-open-source-startups/

elastic

Partnerships

elastic

@xeraa

Donations

For Patreon/donations to be a viable business model, have to put in just as much work as you'd have to do selling services/product, likely ending up w less from donations

You can point to donation models that had it easy & I can point to traditional businesses that got lucky too
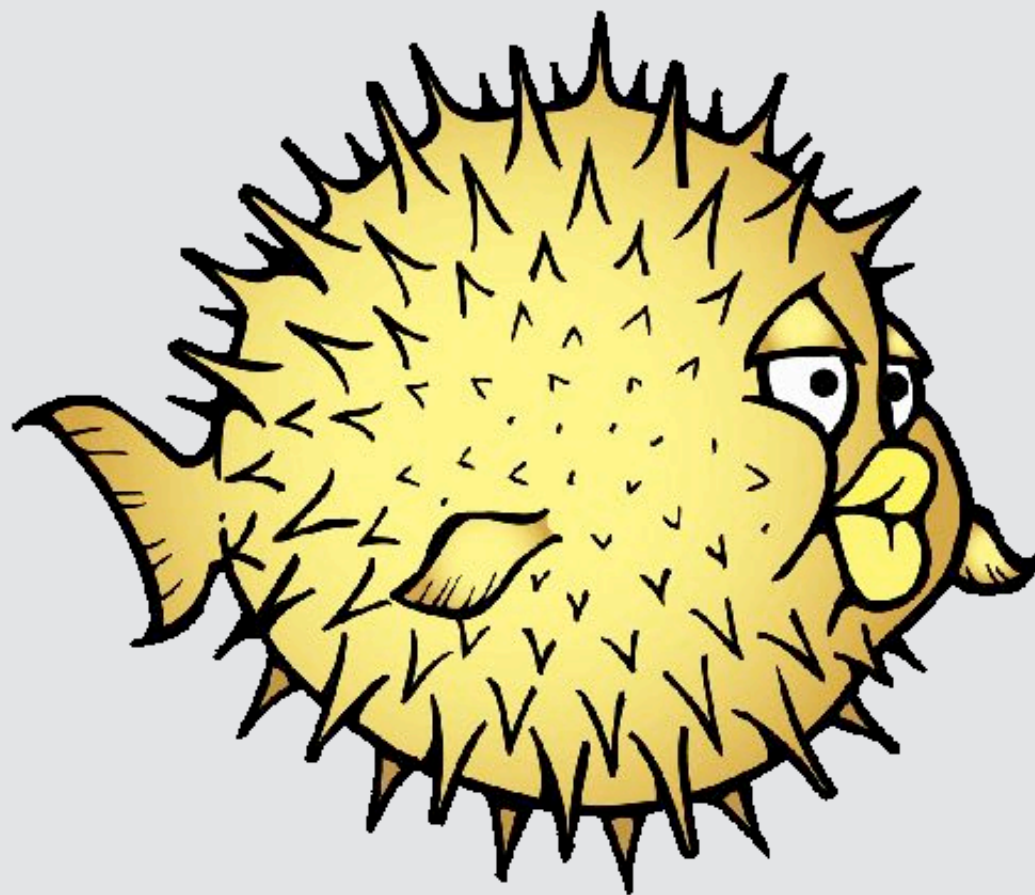
elastic

@xeraa

Certified Partners

elastic

Ads

@xeraa

Merchandise

elastic

Bounties / Crowdfunding

elastic

Cataclysm Dark Days Ahead

Struggle

Contributors ❤️

Users 🙃

Consumers 😐

elastic

[...] Amazon, on one hand, bashes the open source ecosystem and highlights the advantage of its own tools, while at the same time taking projects like Presto, which was developed in the open by Facebook, and turning it into a packaged, revenue-generating product (the newly announced Athena service).

https://dzone.com/articles/did-amazon-just-kill-open-source

elastic

[...] If you are MSFT and I ask you if you've used Redis and you say "you mean Azure Redis Cache?", that's not a good sign.

https://twitter.com/jensenharris/status/984268950136537088

elastic

redis

elastic
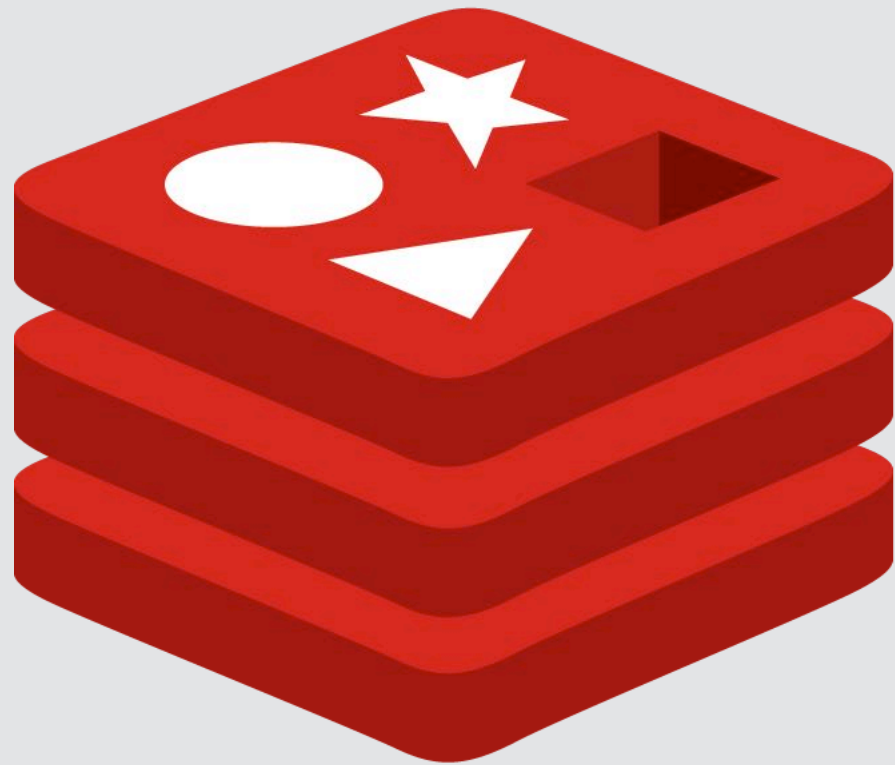
RedisLabs Modules: AGPL ➡️ Apache 2 modified with Commons Clause

RediSearch, Redis Graph, ReJSON, ReBloom, Redis-ML

elastic

Without limiting other conditions in the License, the grant of rights under the License will not include, and the License does not grant to you, the right to Sell the Software.

https://commonsclause.com

elastic

Commons Clause

Still Open Source?

Why not AGPL?

elastic

Confusion

CC

Apache 2 modified with Commons Clause

elastic

Some rabid open-source wonks accused Redis Labs of trying to trick the community into thinking that modules were open source, because they used the word "Apache."

https://techcrunch.com/2018/09/07/commons-clause-stops-open-source-abuse/

elastic

Be aware that the debate starting up about the Common Clause license due to Redis Labs adopting it will likely be heavily tainted by large corporations who have a lot of free labor to lose suddenly pretending to be huge champions of free software ideals.

https://twitter.com/taotetek/status/1032248562116186112

elastic

Redis: BSD

"Redis will remain BSD licensed" http://antirez.com/news/120
"Redis is not "open core" http://antirez.com/news/121

elastic

Philosophy

AWS Open Source Blog

Open Source Korean Analyzer Support in Amazon Elasticsearch Service

by Pallavi Priyadarshini and Vengadanathan Srinivasan | on 05 APR 2018 | in Amazon Elasticsearch Service*, Open Source* | Permalink | 💬 Comments | ↪ Share

Need for an improved Korean analyzer in Amazon Elasticsearch Service (AES)

Processing content for search in Asian languages such as Chinese, Japanese, and Korean presents unique challenges. Typical processing done by a search engine, such as tokenizing based on whitespaces and stemming to derive root forms, is not sufficient for Asian languages, as many words are compound, with different meanings depending on their compound context. AES has improved its support for Korean language processing by adding support for the widely-used open source Korean language analyzer called the Seunjeon plugin. The launch announcement of the Seunjeon plugin support was accompanied by a blog post describing steps to use the plugin with AES. As part of the Seunjeon plugin support, we made several optimizations to the plugin to reduce its memory footprint in order to support instances running with low memory. This post covers the analysis done behind the decision to support the Seunjeon plugin, as well as details of the memory optimizations done to reduce its heap utilization.

Choosing the best Korean analysis plugin for Amazon Elasticsearch Service

We evaluated several Korean analysis plugins before deciding on the Seunjeon plugin. Below we cover the main alternatives that were considered along with the pros and cons of each:

1. **elasticsearch-analysis-openkoreantext** This plugin supports all Elasticsearch versions up to 6.1.1. It uses the *Twitter opensource korean-open-text* module, which has a very powerful Korean analyzer and a built-in dictionary, and is licensed under the Apache License Version 2.0. The drawbacks of this plugin included a less-active community and the lack of capability to add a custom inline dictionary for tokenization. Although it provides a way to add custom dictionaries as files, they would need to be added inside the plugin folder in a pre-determined location which cannot be made dynamic, making adding a custom dictionary file difficult.
2. **elasticsearch-twitter-korean** This plugin only supports Elasticsearch up to version 2.3.
3. **open-korean-text-elastic-search** This plugin is obsolete and is no longer maintained.
4. **Seunjeon** This plugin supports all Elasticsearch versions up to 6.1.1, and leverages mecab-ko-dic internally to perform text analysis. It is licensed under Apache License Version 2.0 and includes the ability to specify a custom dictionary inline, as well as a relative file path. A minor shortcoming of this plugin is the necessity to edit the plugin descriptor file to use it with the newer versions of Elasticsearch (> 5.4).

seunjeon

- Overview
- <> Source
- Commits
- Branches
- Pull requests
- Pipelines
- Issues
- Downloads

Optimizing heap utilization

#11 **MERGED** at b88a2e8   🅔 vengadan-seunje…   ⑂ heap_optimize → ⑂ heap_optimize

Overview   Commits   Activity

Author   👤 Vengadanathan

Reviewers

Description   Description:

seunjeon is occupying heap of around 560 MB Initial optimization done by us reduced the heap utilization of the plugin to ~ 174 MB (around 68% reduction) in heap usage.

Profiling and metrics:

- Each query (when compared to unoptimized version of the plugin) will take additional .8-1ms (on an average) to execute because of decompression that we do when reading the feature array from memory.
- seunjeon lexiconDict is occupying heap of around 174 MB instead of existing 560 MB

Testing: All existing UT's and newly added UT's passed

Comments (10)

👤 영호 유

I tested the performance of your code.

Considering the indexing-time, I tried to do a rather long document.

It took 15~20 times more time than un-patched.

```
elapsedTime =  7,326,618 us -> seunjeon
elapsedTime =  9,018,791 us -> seunjeon
elapsedTime =  8,779,136 us -> seunjeon
```

Lucene - Core / LUCENE-8231

Nori, a Korean analyzer based on mecab-ko-dic

⬇ Export ▾

Details

Type:	➕ New Feature	Status:	**RESOLVED**
Priority:	⚠ Major	Resolution:	Fixed
Affects Version/s:	None	Fix Version/s:	7.4, master (8.0)
Component/s:	None		
Labels:	None		
Lucene Fields:	New		

People

Assignee:	❓ Unassigned
Reporter:	🔵 Jim Ferenczi
Votes:	② Vote for this issue
Watchers:	⑦ Start watching this issue

Dates

Created:	28/Mar/18 21:21
Updated:	6 days ago
Resolved:	6 days ago

Description

There is a dictionary similar to IPADIC but for Korean called mecab-ko-dic:
It is available under an Apache license here:
https://bitbucket.org/eunjeon/mecab-ko-dic

This dictionary was built with MeCab, it defines a format for the features adapted for the Korean language.
Since the Kuromoji tokenizer uses the same format for the morphological analysis (left cost + right cost + word cost) I tried to adapt the module to handle Korean with the mecab-ko-dic. I've started with a POC that copies the Kuromoji module and adapts it for the mecab-ko-dic.
I used the same classes to build and read the dictionary but I had to make some modifications to handle the differences with the IPADIC and Japanese.
The resulting binary dictionary takes 28MB on disk, it's bigger than the IPADIC but mainly because the source is bigger and there are a lot of
compound and inflect terms that define a group of terms and the segmentation that can be applied.
I attached the patch that contains this new Korean module called ~~godori~~ nori. It is an adaptation of the Kuromoji module so currently
the two modules don't share any code. I wanted to validate the approach first and check the relevancy of the results. I

If you are claiming your startup/company is open source and you aren't contributing to some form of upstream... then you aren't really. Sorry not sorry.
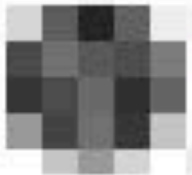
https://twitter.com/jessfraz/status/954802380125736961

elastic

🔒 Elasticsearch-rails & -ruby are abandoned ✏️

🟩 Elasticsearch

Oct '17

It was pointed out 3 months ago here 54 and nothing has changed - 66 PRs and a lot of bugs not being fixed.
Latest commit was made 5+ months ago.
elasticsearch-rails will be incompatible with an upcoming Rails 5.2 and it looks like nobody cares. How can we rely on a product with such bad support?

elastic

warkolm 🛡 Mark Walkom Elastic Team Member   Oct '17

Would you be willing to take on some of the maintainer load to assist?

2 Replies ⌄   5 Likes ♥ 🔗 ... ↩ Reply

▓▓▓▓ ▓▓▓▓▓   Oct '17

Of course not. Can't you hire 1 more developer?

                                                    ♥ 🔗 ... ↩ Reply

elastic   @xeraa

If you're unhappy with an open source project you're entitled to a full, immediate refund to the $0.00 you paid for it. Alternatively, get involved with the project to make it better.

https://twitter.com/MikeMcQuaid/status/970000490468708352

softwaredoug Doug Turnbull ✉ Oct '17

Wow! Come on man, you're getting this stuff for free. Don't be a freeloader, if you see problems share the load! That's the joy and burden of open source 🙂

5 Likes ♥ %o ••• ↩ Reply

Oct '17

Are you sure I'm using it for free? Elasticsearch is a commercial product with $104m of funding and a lot of paid subscriptions 5 , why the hell would I work for them for free?
Btw, you're walking on the streets and breathing air for free - do you clean them up and plant new trees from time to time (also for free)? Nope? Why not?

♥ %o ••• ↩ Reply

elastic

@xeraa

The first crucial aspect to address is that the Ruby client ③ and the Rails integrations ① are two completely separate projects, which together contain eight different Rubygems (not counting the X-Pack integration, with a very different degree of problematic aspects. For any further discussion, let's keep this distintion in mind.

The Ruby client has 6 open pull requests ② right now, with [5 pull requests] waiting for reply from the contributor. There are also 10 issues with questions waiting for reply ② , and 26 issues with general discussion waiting for reply ③ . The client *and* associated libraries, like `elasticsearch-extensions` are consistently updated ② , with a lot of activity from October 2017, actively tracking the changes in the Elasticsearch 6.x branch. Travis CI currently fails because of incorrect snapshot URL for the Elasticsearch 6.0 package, not with a load of actual test failures. That, to me, doesn't look like an "abandoned" project at all. If it does to our community, I'm very much in favour of a discussion about how can we improve here.

The Rails integration is a different story, and I understand the frustration expressed in the original post by ▃▃▃▃▃▃ . The project itself consists of *three* totally different Rubygems: the integration with ActiveRecord models, the persistence layer, and the Rails helpers and extensions. All three are pretty different when it comes to complexity, usage — *and* community contibutions. Just compare the download numbers between `elasticsearch-model` ② and `elasticsearch-persistence` ② : 3 million versus 600 thousands. But one of biggest sources of troubles, manifesting in the number of issues and pull requests submitted, is actually `elasticsearch-persistence`, which skews the perspective quite significantly.

With that in mind, let's consider the technical details. For a good example, I've been actively and publicly telling people that the *ActiveRecord* pattern for the `elasticsearch-persistence` is too challenging from the development *and* usage standpoint, and that it's going to be deprecated in favour of the *Repository* ④ pattern. Now, you *can* hold it against the maintainer, in this case me, to not act on such a plan in a very long time. Doing this transition properly, which means updating the code with deprecation

Ruby on Rails Integrations Engineer

APPLY NOW

Location: Anywhere EMEA or US
Department: Engineering

At Elastic, we have a simple goal: to solve the world's data problems with products that delight and inspire. As the company behind the popular open source projects — Elasticsearch, Kibana, Logstash, and Beats — we help people around the world do great things with their data. From stock quotes to real time Twitter streams, Apache logs to WordPress blogs, our products are extending what's possible with data, delivering on the promise that good things come from connecting the dots. The Elastic family unites employees across 30+ countries into one coherent team, while the broader community spans across over 100 countries.

Right now, we are looking to add a talented Ruby Engineer with a deep understanding of Rails and real passion for open source development to collaborate within our small, but globally distributed team.

Responsibilities

Your primary responsibility will be to develop and maintain the integration between the Ruby on Rails platform and the Elastic

Pulse

Contributors

Community

Traffic

Commits

Code frequency

Dependency graph
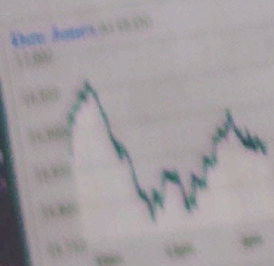
Network

Forks

Additions and **Deletions** per week

👁 Watch ⌄ | 236   ★ Star | 2,344   ⑂ Fork | 613

<> Code   ⊙ Issues **153**   ⑂ Pull requests **54**   ▥ Projects **0**   📖 Wiki   📊 **Insights**   ⚙ Settings

Pulse
Contributors
Community
Traffic
Commits
Code frequency
Dependency graph
Network
Forks

Additions and **Deletions** per week

Venture Capital

Accelerated Development vs **Business Dynamics**

elastic

Where is the ☢️⚠️☣️?

Development

Hobbled Product vs **Starving Company**

elastic

influxdb

RethinkDB

elastic

@xeraa

docker

Releases

Multiple Products vs **Proprietary Plugins**

DATASTAX

Qt

elastic

Open

Open Source vs Open Code

elastic

Elastic's Strategy
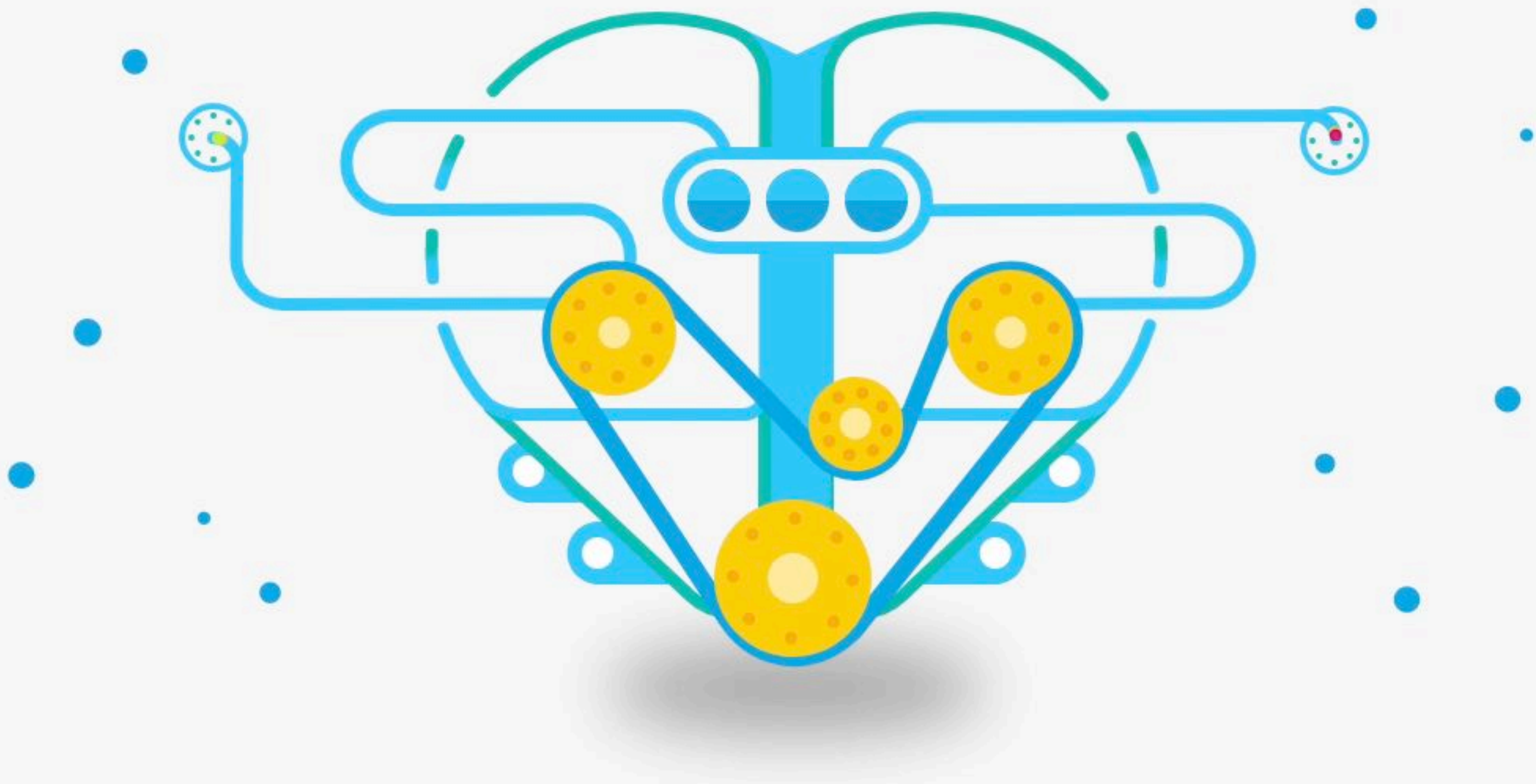
	OPEN SOURCE Free Download	BASIC Free License	GOLD Request Info	PLATINUM Request Info	ENTERPRISE Request Info
THE ELASTIC STACK					
Elasticsearch	✓	✓	✓	✓	✓
Kibana	✓	✓	✓	✓	✓
Beats	✓	✓	✓	✓	✓
Logstash	✓	✓	✓	✓	✓
APM - agents and server (Beta)	✓	✓	✓	✓	✓
X-PACK					
⌄ Security (formerly Shield)			✓	✓	✓
⌄ Monitoring (formerly Marvel)		✓	✓	✓	✓
⌄ Management		✓	✓	✓	✓
⌄ Alerting (via Watcher)			✓	✓	✓
⌄ Machine Learning				✓	
⌄ APM (Beta)		✓	✓	✓	✓

All Code Open

Not Open Source —
Apache 2 & Elastic EULA

elastic

Training

Consulting

cloud

elastic

@xeraa
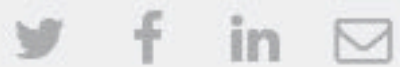
cloud
Enterprise

elastic

@xeraa

elastic

Sold Out

? Help

elastic

Home

All

Apparel

Gadgets

Newsletter   About us   FAQ's

https://elastic.shop

Sold Out

? Help
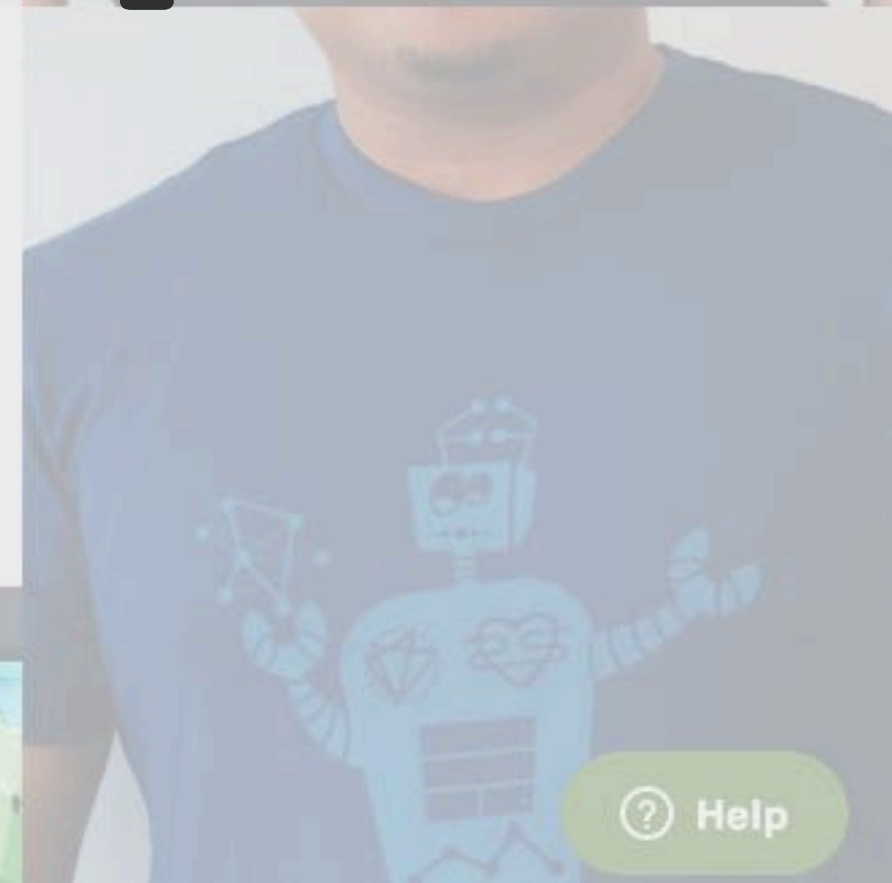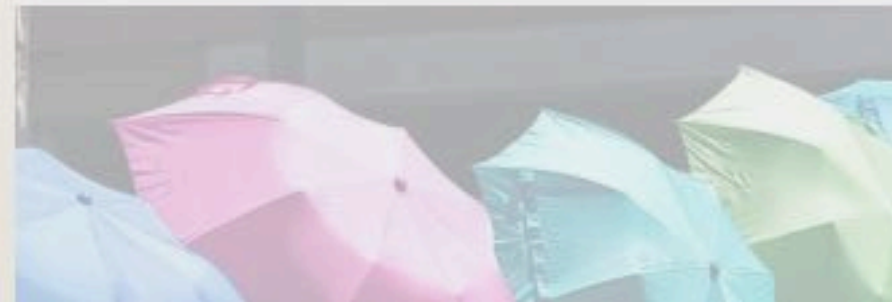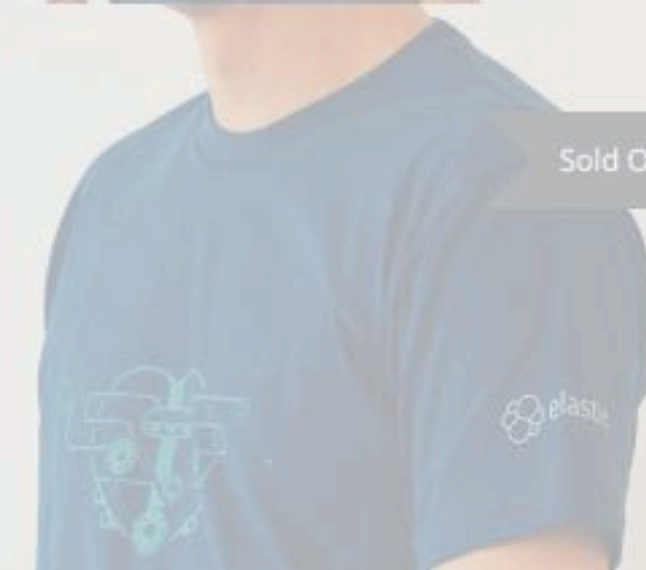
Success

Business Is
Optional

elastic

elastic
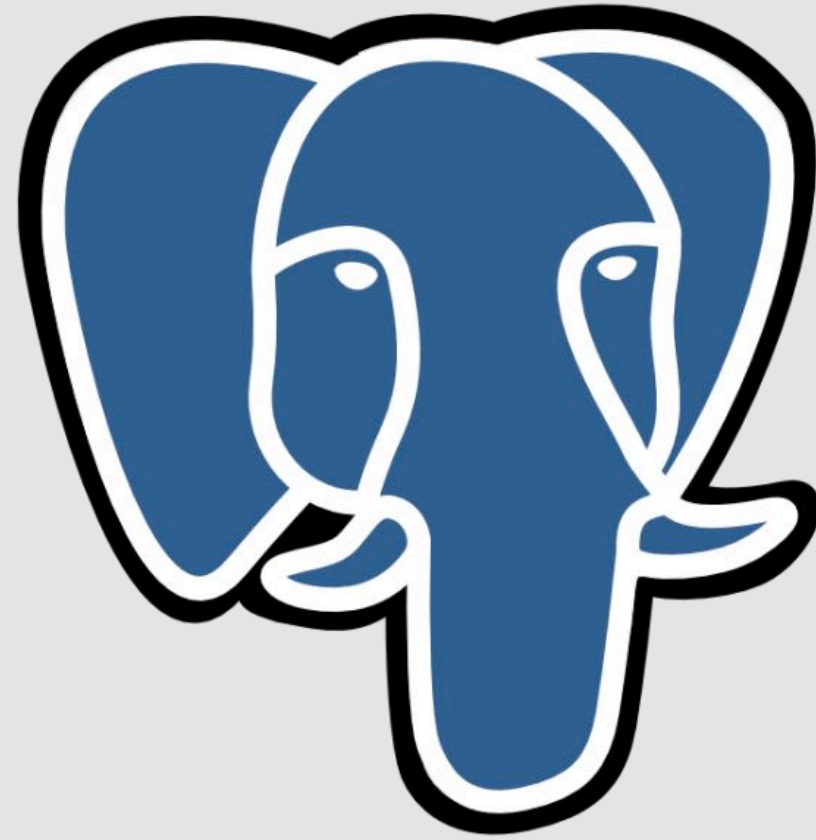
mattklein123
Engineer @lyft
Jun 17, 2017 · 6 min read

Optimizing impact: why I will not start an Envoy platform company

The following are my own views, and not that of my employer, Lyft.

Although almost unbelievable to me, Envoy has gained exceptional traction since it was open sourced about 9 months ago. (If unfamiliar with Envoy see here and here for more information). Very early on I received inbound requests from investors and entrepreneurs who were attracted to a new solution in the burgeoning area of microservice networking and management and who wanted to discuss the potential for commercial opportunities around the software. Over the last few months, especially as excitement for the software has increased (a substantial partnership with Google, the launch of Istio, uptake and excitement of other large companies, etc.), the amount of inbound requests and push by the investment community to figure out a

Business Is
Complicated

elastic

It saddens me that closed source companies are acclaimed for merely putting a piece of code on GitHub, while commercial open source companies get the worse shit for any action that involves protecting their work.

https://twitter.com/icecrime/status/1032155227368185856

elastic

Questions?
Disagreement?

Philipp Krenn   @xeraa

elastic

@xeraa