Babelfish

Universal Code Parsing Server

Curry On

Barcelona 2017

Santiago M. Mola

Who am I?

- Lead Data Engineer at source{d}: https://sourced.tech
- Working on a pipeline to analyze all open source code found online.

Who am I not?

- Programming languages expert

The Story

2015

2015

- Project: find best developers to recruit based on their open source projects

2015

- Project: find best developers to recruit based on their open source projects
- How?
  - Fetch all Git repositories
  - Apply PageRank to contributors
  - Rank based on recent activity per language
  - ...

2015

- Project: find best developers to recruit based on their open source projects
- How?
  - Fetch all Git repositories
  - Apply PageRank to contributors
  - Rank based on recent activity per language
  - ...
- Easy

2016

Detect common libraries

2016

Detect common libraries Regexps

2016

Detect common libraries	Regexps
Ignore comments	

2016

Detect common libraries	Regexps
Ignore comments	More regexps

2016

Detect common libraries	Regexps
Ignore comments	More regexps
More libraries	

2016

Detect common libraries	Regexps
Ignore comments	More regexps
More libraries	Tokenizing, some pattern matching, +200 regexps

Data Science comes in

Data Science comes in

Dataset of identifiers used in every file in GitHub

Dataset of identifiers used in every file in GitHub

Pygments syntax highlighting + select identifiers

Dataset of identifiers used in every file in GitHub

Pygments syntax highlighting + select identifiers

Dataset of all tokens inside small Go functions

Dataset of identifiers used in every file in GitHub	Pygments syntax highlighting + select identifiers
Dataset of all tokens inside small Go functions	Write an extractor based on Go AST

Dataset of identifiers used in every file in GitHub	Pygments syntax highlighting + select identifiers
Dataset of all tokens inside small Go functions	Write an extractor based on Go AST
Dataset of tokens per block in every language	

Dataset of identifiers used in every file in GitHub	Pygments syntax highlighting + select identifiers
Dataset of all tokens inside small Go functions	Write an extractor based on Go AST
Dataset of tokens per block in every language	...

Abstract Syntax Trees

- We can use ASTs to easily extract the kind of data we need
- Most languages have one already implemented

Universal Abstract Syntax Trees

- But we need them normalized
- Writing our own grammars to produce a UAST?

Universal Abstract Syntax Trees

- But we need them normalized
- Writing our own grammars to produce a UAST?
- For 200 languages?

Middle Ground

- Leverage standard parser from each language
- Normalize as a post-processing step
- Normalizers are written in a single language (Go)

Middle Ground

- Leverage standard parser from each language
- Normalize as a post-processing step
- Normalizers are written in a single language (Go)
- **That's Babelfish**

Motivation

- *Babelfish was born as a solution for massive code analysis.*
- **Goal:** analyzing all source code from every repository in the world, for every revision.

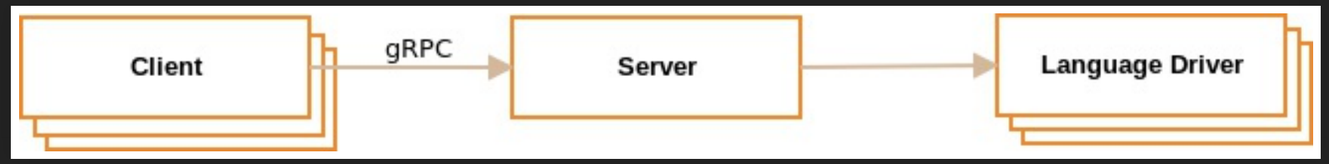
Scope

- **Scope:** parsing single files in any programming language and producing a universal abstract syntax tree.

Scope

- **Scope:** parsing single files in any programming language and producing a universal abstract syntax tree.
- **Future scope, maybe, *who knows:*** full project analysis, where source code can be analyzed with its full context, and not just per-file.

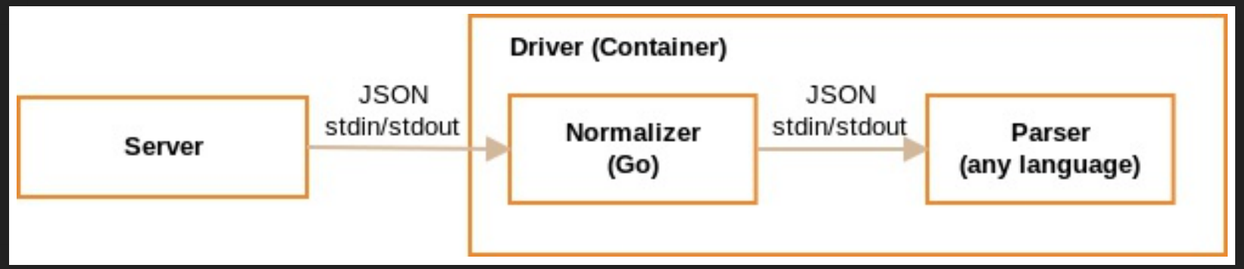
Architecture Overview



Containerize everything!

- Language drivers are packaged as Docker containers
- Server contains a lightweight container runtime
- Based on libcontainer
- No Docker, no external runtime dependencies
- Official drivers published at Docker Hub

Driver Architecture



UAST

- Minimal structural normalization
- Language-independent annotations
- Fallback to language-specific data

Clients

- gRPC: protocol code generated for any language
- Library available in Go
- C++ library + bindings in multiple languages *(early stage)*

Status

- Early stage
- Server working
- Beta drivers: Python and Java
- Early drivers: 13 more
- Early UAST Specification
- Showcase tools (tokenizer, cyclomatic complexity, npath)

We need your help

- Feedback on the UAST specs
- Language driver contributions
- Use cases implemented on top of it!

Thank you!

- Documentation: doc.bblf.sh
- GitHub: github.com/bblfsh
- Slack: doc.bblf.sh/community.html