



Analysis of Gitlab Using Boa

Team sdmay20-43:

Diego Realpe Tobar - Data Pipeline Manager, Beloved Leader

Adrian Hamill - Data Pipeline Manager

Megan Miller - Boa Expert, R expert, Website manager

Yi-Hsien Tan - Boa Expert, Website Manager

Benjamin Carland - Boa Expert

Client/Advisor - Simanta Mitra

Website - <http://sdmay20-43.sd.ece.iastate.edu/>



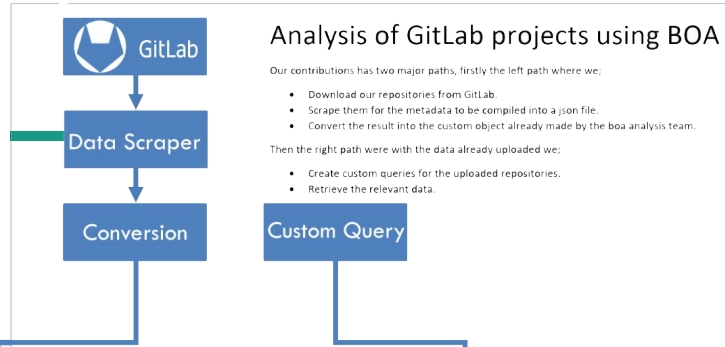
Part I: Project Plan



Project Overview

- Code Analyzer
- Integrate Boa with GitLab
- Analyze GitLab projects using Boa
- Generate reports using R with data pulled from GitLab projects

Conceptual Sketch



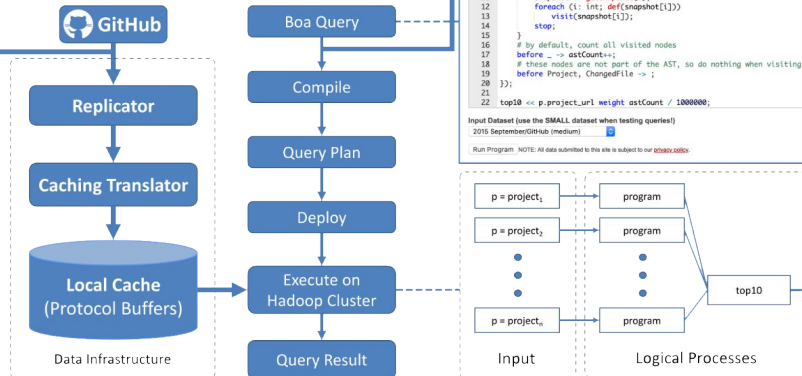
Analysis of GitLab projects using BOA

- Our contributions has two major paths, firstly the left path where we:
- Download our repositories from GitLab.
 - Scrape them for the metadata to be compiled into a json file.
 - Convert the result into the custom object already made by the boa analysis team.
- Then the right path were with the data already uploaded we:
- Create custom queries for the uploaded repositories.
 - Retrieve the relevant data.

Boa is a domain-specific language and infrastructure that eases mining software repositories.

- Its infrastructure leverages distributed computing techniques to execute queries against thousands of software projects efficiently.

Boa Architecture Overview



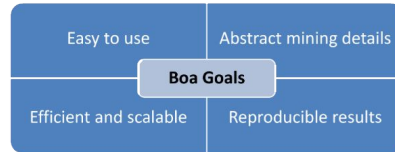
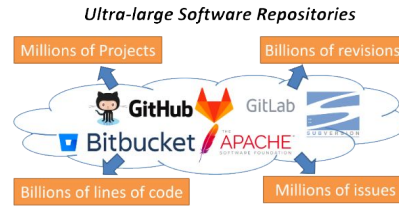
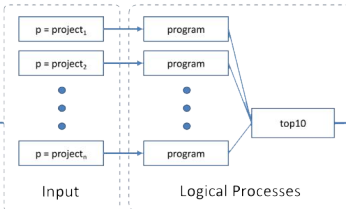
```

Run Examples
-- Select Example --
Boa Source Code
1 # What are the 10 largest projects, in terms of AST nodes?
2 # Output is in Millions of AST nodes.
3 p: Project = Input;
4 top10: output top(10) of string weight int;
5
6 astCount := 0;
7
8 visit(e, visitor {
9   # only look at the latest snapshot
10  before m: CodeRepository => {
11    snapshot := getSnapshot(m);
12    foreach (c: int; def(Snapshot{1}))
13      visit(snapshot{1});
14  }
15 }
16 # by default, count all visited nodes
17 before _ => astCount++;
18 # these nodes are not part of the AST, so do nothing when visiting
19 before Project, ChangedFile => ;
20 });
21
22 top10 << p.project_url | weight astCount / 1000000;
  
```

Input Dataset (use the SMALL dataset when testing queries!)

2016 September@GitHub (medium)

Run Program: NOTE: All data submitted to this site is subject to our [privacy policy](#).



Examples

Programming Guide

Eclipse IDE

Client API

Publications

My Logged In

Run Examples

Job List

My Account

Log Out

Job Status View Job Output Download Job Output

Output for Job 76025

```

top10 = https://github.com/mwmg/ast-test2, 15.0
top10 = https://github.com/lehen/od-mod, 13.0
top10 = https://github.com/ernst/erlang, 11.0
top10 = https://github.com/robertmeyer/erlang, 9.0
top10 = https://github.com/erwan/erlang, 7.0
top10 = https://github.com/FreeBSD/freebsd-dep, 6.0
top10 = https://github.com/nextint/next-intent, 6.0
top10 = https://github.com/IBM-India/ibm-India, 5.0
top10 = https://github.com/ast-lab/ast-lab, 5.0
top10 = https://github.com/sumitkang/sumitkang, 3.0
  
```



Functional Requirements

Automation - After configuration setup, the solution processes and completes with a single click from the user.

Repository Analysis - The solution utilizes the metadata from GitLab repositories.

Visual Output - Query results are put into graphical formats when possible.



Technical Considerations

- Limitations of the Boa language
 - Lacks typical language types
- Difficulty setting up pipeline for Private Gitlab repos
 - Important for Security requirement
 - Turned out to be impossible because of how the GitHub implementation of Boa works



Risk Mitigation

- We maintained communication with client and within the group about all significant risks that we determined.
- The risk that some of the queries that we decided on might not be possible.
 - Having experience it was easier to identify possible issues.
- The risk that some compromises need to be made to implement the pipeline.
 - Repositories needed to be Public.



Resources and Cost Estimate

- The monetary cost of the entire project was the Mac Mini running the Boa framework.
- We found out later that it would be possible to run that on Linux, so now it runs on Linux OS.



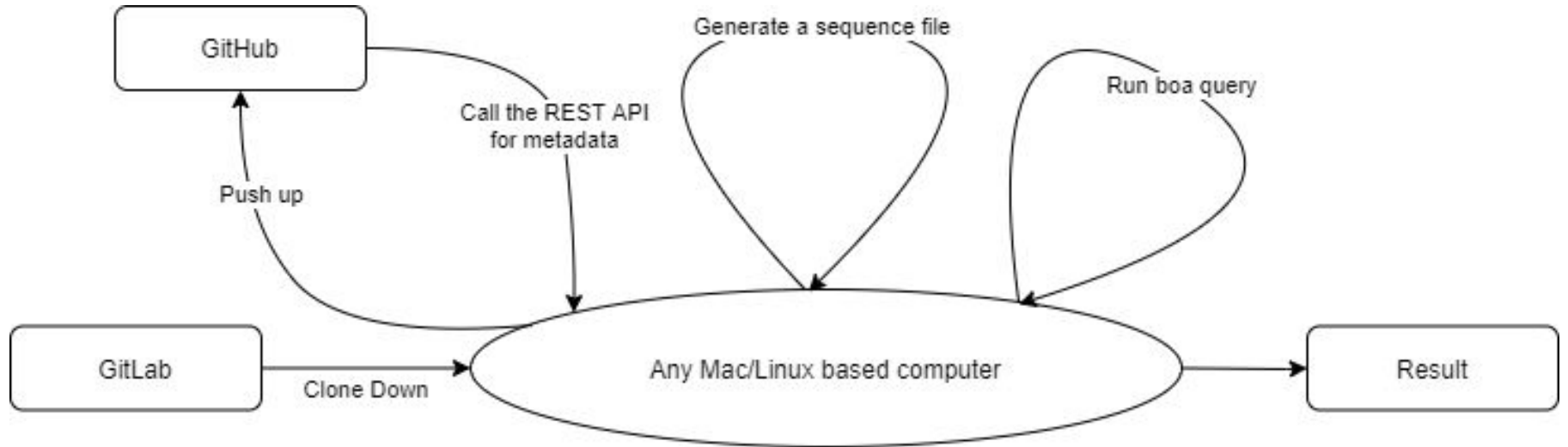
Project Milestones & Schedule

Goal	Name of Deliverable	Expected Delivery	Actual Delivery
1	Ideas for Boa Queries	Dec 1	Dec 1
2	Finalized List of Queries	Jan 20	Jan 20
3	Gather data to Query	Feb 2	Feb 15
4	Working prototype queries	Feb 15	Mar 15
5	Finished Queries *	Mar 1	Apr 21
6	Finished Product with R	Apr 1	--



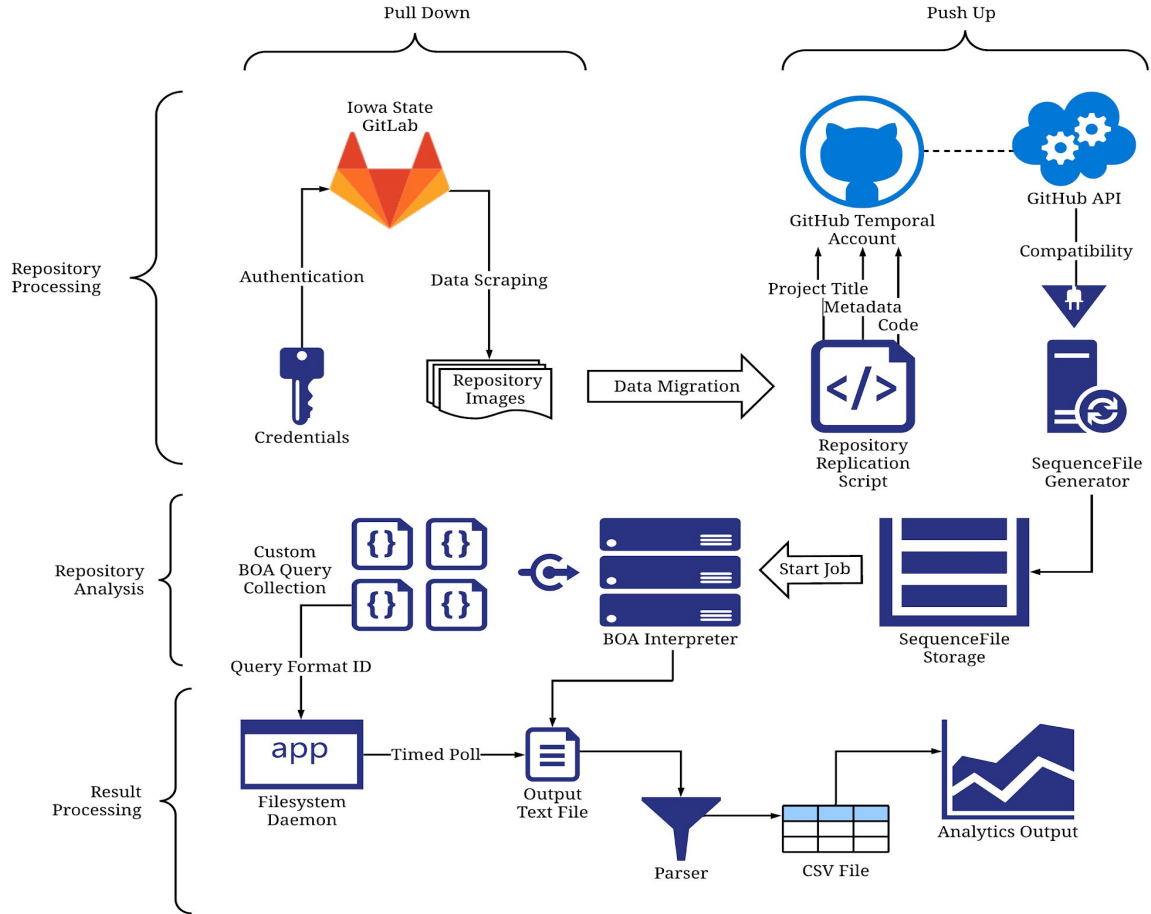
Part II: System Design

Functional Decomposition





Detailed Design





Technologies Used

- Boa Language
- Bash Scripting
- Git/Hub
- Rscript
- FSWatch
- VirtualBox
- Eclipse



Testing Implemented

Boa Queries were tested by comparing their output to an expected output for the repositories provided. There were issues in the testing and we could not deploy an automated version of Boa script testing since the language is not developed enough to do this.

On the backend, testing was done by setting up a Virtual Environment where specific functions of the code were tested separately and then ran together to test correct I/O.



Prototype Implementations

- First prototype was to bypass GitHub altogether, we attempted to generate SequenceFiles from GitLab directly which was finally not possible
- Second prototype tried to analyze micro aspects of project code with Boa + SpotBugs, this ended up not being feasible or compatible with Boa and had to be taken out.



Engineering Standards & Design Practices

- Privacy and Security were a big focus
 - A lot of time was spent looking for an automated way to export GitLab projects to private
- Quality
 - Component test assured us that the data pipeline, queries and R code were working as intended.



Part III: Conclusion



Team Contributions

- Ben: Research and development of AST pertaining queries
- Megan: Aided in queries, created the R graphical scripts
- Yi-Hsien: Took part in investigating and developing Boa queries
- Diego: Aided in backend functionalities, developed the output parsing script, general administrative duties, managed client/teammates
- Adrian: Developed the setup functionalities, led the “Pull Down, Push Up” solution and the execution of the Boa scripts via CLI.



Future Prospects

The work done here is a significant baseline, the skeleton of this conversion tool is almost completely finished. In the future, others will be able to work out the last connecting link to these query execution and complete the range of tests available to users.



Questions