

AN ABSTRACT OF THE THESIS OF

Rithika Kiran Naik for the degree of Master of Science in Computer Science
presented on September 15, 2015.

Title: Visualizing Contribution Patterns in Open Source

Abstract approved: _____

Dr Carlos Jensen

Open Source software gives users the freedom to copy, modify and redistribute source code without having to pay any amount of money to an organization or individual. The evolution of these softwares usually depend a lot on how the participating crowd interact and co-operate with each other. Over the past few years, open source software have become widely accepted and used hence making their study a very hot topic. To understand the working philosophy of any open source software one needs to look into a variety of factors like community interaction, strength, evolution, support and contribution patterns. Bitergia is one such organization which is highly involved in getting, providing and analyzing Software Development metrics and information for open source projects. The main idea behind this thesis work is to understand contribution patterns by the community in few open source projects by the means of various visualizations. Data visualization has gained a lot of popularity as it helps in easy analysis of huge amounts

of information quickly. We have hence taken up the task of improving the existing visualizations by Bitergia into visualizations which will help in taking informed decisions clearly. Our analysis were performed on five different datasets : Eclipse Foundation, OpenStack Foundation, Puppet Labs, Red Hat and Apache Cloud-stack.

©Copyright by Rithika Kiran Naik
September 15, 2015
All Rights Reserved

Visualizing Contribution Patterns in Open Source

by

Rithika Kiran Naik

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented September 15, 2015
Commencement June 2015

Master of Science thesis of Rithika Kiran Naik presented on September 15, 2015.

APPROVED:

Major Professor, representing Computer Science

Director of the School of Electrical Engineering and Computer Science

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Rithika Kiran Naik, Author

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank my advisor, Dr Carlos Jensen, for his patience and constant support throughout the phase of this research. All the brainstorming sessions to discuss every minute detail in progress were extremely fruitful. My sincere gratitude to Dr A, Dr B, Dr C for accepting my invitation and taking time to be a part of my Graduate committee.

My gratitude to team Bitergia as this work would not have been possible if not for their quest to learn more. So thank you Jesus M Gonzalez Barahona and Gregorio Robles for providing me with the necessary datasets and timely suggestions.

Thank you Shankar Jothi for sitting through those long sessions of programming with me, Pranathi Bhattacharji for providing me with the much needed stress relief and all my friends in Corvallis for the wonderful 2 years away from home. Special thanks to the entire Human Computer Interaction group for looking through my work and providing me with valuable inputs.

My dream of pursuing a Masters degree would not have been fulfilled if not for my parents constant encouragement and faith in me. Thank you Mom and Dad for everything !

TABLE OF CONTENTS

	<u>Page</u>
1 Introduction	1
1.1 Geography of Developers	1
1.2 Experience of Developers	2
1.3 Why Data Visualizations	3
2 Related Work	4
2.1 Introduction to the Introduction	4
3 Bitergia	5
3.1 Services provided	5
3.2 Task at hand	7
4 Methodology	11
4.1 Dataset	11
4.2 The Organizations	12
4.3 D3.js	13
4.4 Initial Iterations	14
4.5 Final Implementations	14
5 User study	15
5.1 Introduction to the Introduction	15
6 Results	16
6.1 Introduction to the Introduction	16
7 Conclusion	17
7.1 Fin	17
Bibliography	17

LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
2.1 Go figure.	4
3.1 Bitergia data forums	6
3.2 Timezone origin of messages	8
3.3 Timezone origin of messages for 2002,2007,2012	8
3.4 3D version of pyramids every two years	9
3.5 Comparison for pyramids every two years	10
4.1 Global population by timezone	12
4.2 Go figure.	14
5.1 Go figure.	15
6.1 Go figure.	16

Chapter 1: Introduction

The software industry has undergone a lot of changes over the years and one such important change was the advent of Open source software. Open source software relies heavily on its community for its growth and development [4]. The OSS community usually has a lot of developers collaborate from all over the world who work voluntarily to produce code and support the software in general. Their motivations and ethics for this have always been open for studies [5].

1.1 Geography of Developers

Its intriguing to note that though studies have been conducted to understand the motivations of these developers, understanding the range of developers contributing, be it geographically or age wise still has a lot of scope. Greogorio Robles and Gonzalez-Barahona were among the first ones to look into the geographic factors that influenced developer contributions. They collected data of all the registered users from sourceforge as it was the largest libre software web based platform which had data about geographic locations. After performing several techniques on this data they came up with a distribution of developers in different regions of the world[1]. The problem here was that the information about location or geography was not stored clearly in sourceforge. After this came the study by Takhteyev Y ,

Hilt A and they studied open source data made available by Github, which was becoming a pioneering source management software. But there were problems here as well as the dataset was either incomplete or was not categorizable by TLDs [2]. Then came the idea of looking at timezones and IP's of the systems of the users who contribute, both collected through sourceforge. But the issue with looking at IP addresses was that the dataset is not available to all and agreements needed to be signed for the same[7]. Studies involving looking at mailing lists data to understand developer geography has not quite seen the light of day yet. If the data can be accessed then the same studies could show effective results as well. Results from the above mentioned studies all show that North American and European countries particularly USA are the major areas where developers are mostly located and work from. A survey by Ghosh et al[8] showed the influence of French, American and German developers but the survey that was conducted by Tuomi et al[9] showed varied results leaning a lot towards US, UK, Canada and Germany. If the area is further widened and looked up on continent wise then North America takes the lead followed by Europe and clusters of Asia, Australia, South America and Africa and this was from the study by Gregorio Robles and Gonzalez-Barahona on sourceforge data[1].

1.2 Experience of Developers

In 2002, a study was conducted on two open source projects, Apache and Mozilla, to understand open source development. One of the parameters that was looked at

was developer participation where they looked at the frequency of contributions over a period of time. They concluded that very few developers who are not a part of the core group contribute regularly in Apache and Mozilla showed that the core group was very active as well[10]. The professors of Notre Dame University studied the parameter on a larger group of varied open source projects for better understanding of open source project development. They noted that developers in open source projects are not as well connected as developers in other collaborative networks. Also developers were more in number in general in larger more successful projects than in smaller fairly unknown projects[5].

1.3 Why Data Visualizations

In 1973, the Anscombe's Quartet showed how four sets with nearly identical statistical properties but appeared very different when graphed [11]. This led to a serious thought that understanding data by just looking at it would not be useful or correct. Hence visualizations were given a lot of importance. As the proverb says, "A picture is worth a 1000 words", visualizations made headlines for having thrown light on datasets in a completely new way. As the amount of data will only increase in the near future with each of it having a different value to it, a common way of unifying the data with the its underlying meaning was much needed. This is how data visualizations became popular. This is the very reason why we decided to work with them as well. Comparing, contrasting, reviewing data becomes easier when one views various visualizations.

Chapter 2: Related Work

I have done some excellent research [?].

2.1 Introduction to the Introduction

A rectangular box with a thin black border, containing the word "Box" in a serif font.

Figure 2.1: Go figure.

Chapter 3: Bitergia

Bitergia was founded by four professors from the Universidad Rey Juan Carlos, Spain in July 2012. They were researchers who were very much involved in understanding the open source community and its growth for long. This was the main idea behind starting Bitergia, the software development analytics company. The focus here was to have data analytics techniques to mine information about project performance, track developer actions, find areas of improvement and identify risks involved in further development.

3.1 Services provided

As providing software metrics was the goal here, they started off by building tools that would help them achieve the same by extracting data from multiple forums. Few such tools developed are -

1. CVSAnalY : This tool collects and organizes data from various code management systems like CVS, Subversion and Git.

2. RepositoryHandler : This is a python library which handles code repositories using a common interface.

3. MailingListStats : This is a command line based tool which retrieves information from mailing lists. It supports files in mbox and web accessible formats.

4.Bicho : Another command line tool that retrieves information from issue tracking systems like Bugzilla, Jira, Sourceforge, Allura, Github, Google Code, Launchpad etc.

5.CMetrics : This is a library used to measure size and complexity for C files in projects.

6.Sibyl : This tool is used to extract information from websites that have a Question-and-Answer format like Askbot with further support planned for Stack-Overflow.

These bunch of tools together are known as the *MetricsGrimoire Library* which is accesible for all in Github.

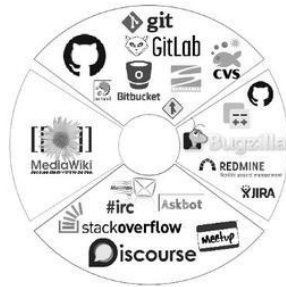


Figure 3.1: Bitergia data forums

Now that they had the tools ready to extract the data, the next step was to build a framework which would help them to analyze and then visualize the data which they would obtain from the MetricsGrimoire tools. This is known as the *VizGrimoire Library* and has two important tools in it -

1.vizGrimoireR : After extracting the needed data, this tool helps in analyzing the dataset to produce data which then can be used to answer specific type of

questions.

2.vizGrimoireJS : Having the right kind of data in hand led to the need to have a dashboard kind of environment to showcase the data in a meaningful way. This tool hence creates dashboards with reports and visualizations of the data.

3.2 Task at hand

We were presented with two cases which needed improvised visualizations. The existing visualizations were not feasible to answer different kinds of questions that one can pose. We had to work through various visualizations in order to finalize on a nearly good enough solution. We will first talk about each of the two cases presented to us and then discuss the solutions that we fixed upon in the next chapter.

CASE 1 -Where do Developers work ?

Location has always been of importance in any area of work. In the software industry, distance always was not a matter of concern because of the nature of work involved, with this being even more true for Open Source Software as its developers are assumed to be contributing according to their convenience [2]. Considering this, the group at Bitergia decided to investigate the overall spread of developers across the world for a few projects. Their main database was the data they collected from github and mailing lists for those organizations. There were certain assumptions involved like the timezones mentioned in the mail tools were correct and that they correspond to the geographical areas to some extent. They came up with these

two basic visualizations as a start to analyzing the data.

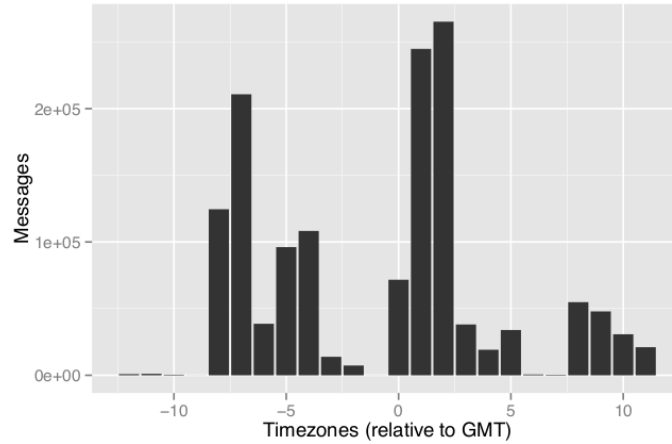


Figure 3.2: Timezone origin of messages

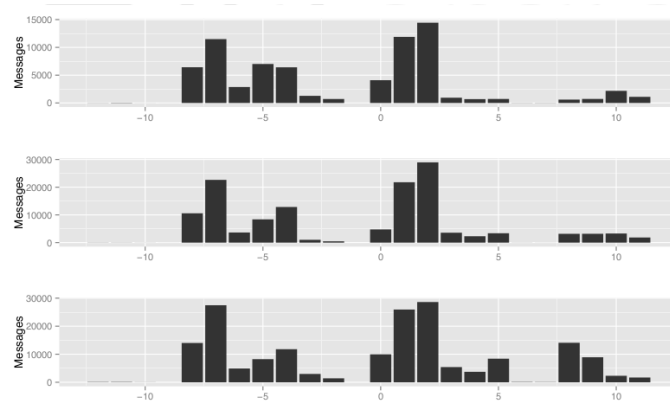


Figure 3.3: Timezone origin of messages for 2002,2007,2012

The following was the division of geography with respect to timezone -

1. America: GMT-8 to GMT-2 (US/Canada: -8 to -4)
2. Europe/Africa/Middle East: GMT to GMT+5
3. East Asia/Australia: GMT+8 to GMT+11

The visualizations being very basic didn't help solve or understand a lot about the developer spread for an organization. They were more oriented in giving an overview of messages per timezone. It was not clear about the relation between commits and authors and location tied together which needed more than a birds' eye view type of a visualization.

CASE 2 - Experience of Developers A lot has been discussed about the motivation for developers to join or stay or leave an open source project. The most sort after work being the onion model which describes the general roles that developers take and how they gravitate eventually [6, 3]. The Bitergia team focused on calculating age in project for active developers at a certain time. They again had access to datasets which gave them the data needed to calculate the same. As a result of their work the following visualizations were built -

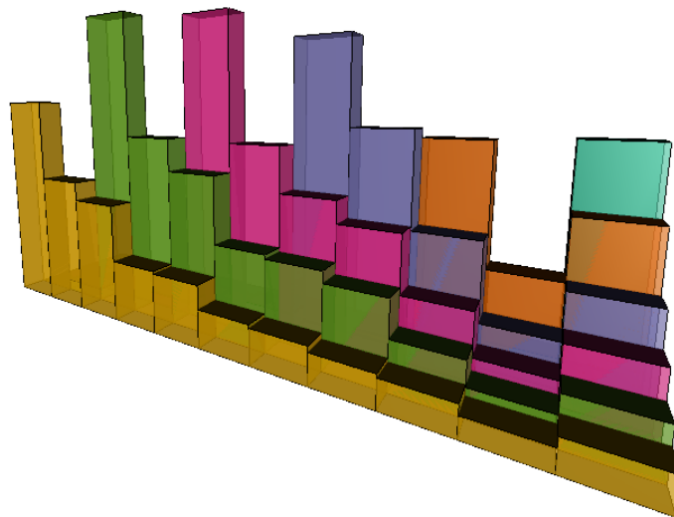


Figure 3.4: 3D version of pyramids every two years

The problem here was that, the 3D visualization, though colorful, was a little

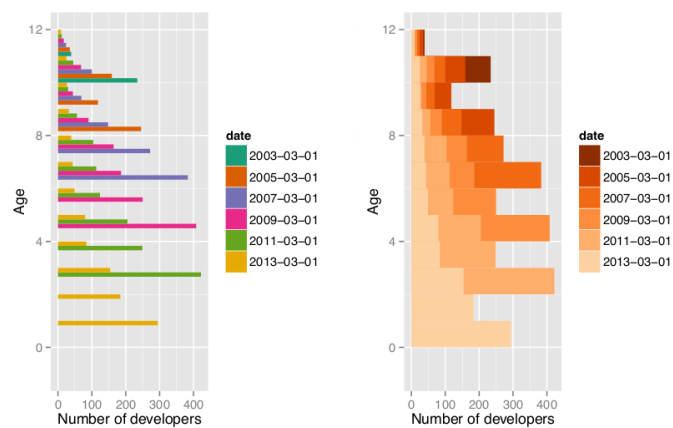


Figure 3.5: Comparison for pyramids every two years

hard to relate to understand the dynamics. WHAT ELSE IS WRONG ?????? My visuals are the same!

Chapter 4: Methodology

To help explore the previously explained cases, our task was to look into improving the visualizations to help people understand the data in a better way. To carryout this process we took datasets of 5 open source projects for the cases explained in the previous chapter. We started off with different prototypes for each case and improvised by taking timely suggestions from the team. We build initial visualizations and followed an iterative process of development.

4.1 Dataset

As Bitergia already had a large database for a few open source projects, they were very co-operative and gave us full access to all the data that they had in hand for the 5 organizations we were looking into. The files were all in the json format and this was an ideal format for us to work with as well. There was some manipulation of data involved for a few visualizations. The first step was to identify the format of the data and then brainstorm for ideal visualizations depending on the possible ways to understand the data. All the data collected was for February,2015 and has data from 2010 to 2015. The timezone files have data from timezone -12 to +11. One important field added to the timezone files, was the approximate population for each of the timezones which was taken from the below graph.

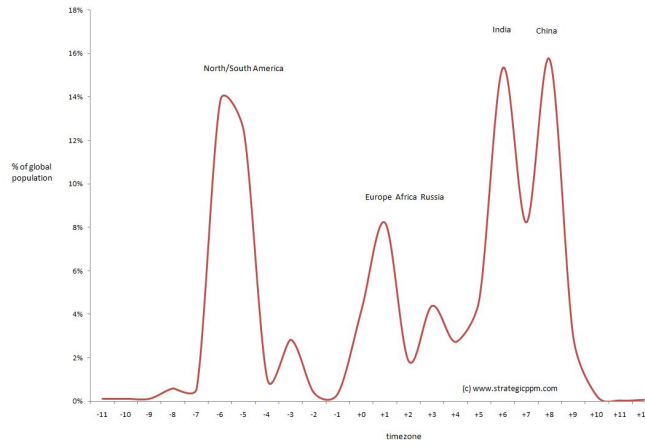


Figure 4.1: Global population by timezone

Also in order to build a timezone based visualization, a geojson file was needed which would draw the map timezone wise and this file was found. As geojson files have large overhead and would take a lot of time to load, converting it to topojson format was better. Topojson is a command line application which not only reduces the size of geojson files, it also gives multiple options like simplification, conversion etc. It can be installed easily using NPM.

4.2 The Organizations

1. Eclipse Foundation : This is a member supported open source community focused on building a platform comprised of extensible frameworks, tools and run-times for building, deploying and managing softwares across the lifecycle.

2. OpenStack : It is one of the most active open source products currently available for all types of cloud environments. It controls a large number of

resources for computation, storage ,networking for a datacenter.

3. Puppet Labs : This is an open source tool which manages various stages of IT infrastructure inclusive of provisioning, patching, configuration and management of operating systems across cloud structures.

4. RedHat RDO: This is a community of people who are interested to use and deploy OpenStack on the RedHat Linux, Fedora ditributions. It is a sub community of people who use RedHat which is one of the largest contributors to Linux.

5. Citrix Apache Cloudstack : This is an open source software which deploys and manages networks of virtual machines as a IaaS(Infrastructure as a Service) platform. It supports VMware, KVM, XenServer, Hyper-V.

4.3 D3.js

As the visualizations needed flexibility the most ideal tool in hand was D3.js. D3 is a javascript library built by Mike Bostock which has the capacity to build interactive, dynamic and a big variety of visualizations. It is largely based on HTML5, CSS and SVG. It makes rendering the visualizations on any browser extremely convenient and easy as well. Due to the large developer support available for the library, it was an ideal choice to build the final visualizations in.It accepts external data in the form of json,CSV or TSV. Being a javascript library, it follows the same syntax hence making data manipulation quick. D3 API has hundreds of functions which make gives it its community. This library is widely used by big

corporates like Datameer, The New York Times, OpenStreetMap etc.

4.4 Initial Iterations

4.5 Final Implementations



Figure 4.2: Go figure.

Chapter 5: User study

I have done some excellent research [?].

5.1 Introduction to the Introduction



Box

Figure 5.1: Go figure.

Chapter 6: Results

I have done some excellent research [?].

6.1 Introduction to the Introduction

Box

Figure 6.1: Go figure.

Chapter 7: Conclusion

Wow, that really was excellent.

7.1 Fin

This is the end, my only friend, the end.

Bibliography

- [1] Robles G , Barahona M Jesus. Geographic Location of Developers at SourceForge, *MSR*,2006.
- [2] Takhteyev Y , Hiltz A. Investigating the Geography of Open Source Software through Github, 2010.
- [3] Ye Y , Kishida K. Towards an Understanding of the motivation of Open Source Developers. In *ICSE*,2003.
- [4] Sethanandha B.D , Massey B, Jones W. Managing open source contributions for software project sustainability, In *PICMET*, 2010.
- [5] Madey G, Freeh V, Tynan R. The Open Source Software Development Phenomenon: An analysis based on social network theory, In *8th Americas Conference on Information Systems*, 2002.
- [6] Crowston K , Howison J. The Social Structure of Free and Open Source Software Development. In *First Monday*, 2005.
- [7] Von Engelhardt S, Freytag A, Schulz C. On the Geographic Allocation of Open Source Software Activities. In *Jena Economic Research Papers*, 2010.
- [8] Ghosh R. Understanding free software developers:Findings from the FLOSS study, In *MIT Press,Boston*, 2005.
- [9] Tuomi I. Evolution of Linux credits file: Methodological challenges and reference data for open source research, In *First Monday*, 2004.
- [10] Mockus A, Fielding R, Herbsleb J. Two case studies of open source software development:Apache and Mozilla, In *TOSEM*, 2002.
- [11] <https://en.wikipedia.org/wiki/Anscombe>

