The Promises and Perils of Mining Ohloh.net

Henk Poley
Vrije Universiteit Amsterdam
hpy300@cs.vu.nl

Rahul Premraj
Vrije Universiteit Amsterdam
rpremraj@cs.vu.nl

ABSTRACT

Ohloh.net is a website that indexes open source software projects. It contains data on approx. 300.000 projects, and 330.000 source committers. In this paper we try to find a scientific use case for the data exposed through the Ohloh website & API, and a review of existing papers.

1. INTRODUCTION

Introduction here..

1.1 Related websites

Sourceforge.net has recently (May 2009) acquired Ohloh.net. Changes to either Ohloh or Sourceforge at not fleshed out. Though further orientation on finding projects on Ohloh's side, and hosting projects on Sourceforge's side seems obvious.

Ohloh.net has surpassed *Freshmeat.net* since November 2008, according to compete.com site analysis. Alexa.com does not show this change, but does show wild changes in their statistics aggregation.

DistroWatch.com moves in more or less the same open source space, but collects only high level projects that integrate software together into desktop OSes. This site is less interesting to developers. It has about half the vistors as Freshmeat.net, and was surpassed by Ohloh in September 2008.

2. PROMISES AND PERILS

2.1 Promise: Learn what people are using and contributing to

In the paper ”RDFohloh, a RDF wrapper of Ohloh” Sergio Fernández worked on exposing the accounts on Ohloh as Friend-Of-A-Friend (FOAF) data in RDF format through the Ohloh API [3].

2.2 Peril: One person could be represented as multiple contributers

For example a search for *torvalds* will return 15 results. One is Linus Torvalds account. Another is a fan. And 13 are Linus Torvalds commits to projects that are not mentioned on his personal account. The paper ”Smushing RDF instances” worked on combining ('smushing') accounts and committers that represent the same person [4].

2.3 Promise: Journaling lets you keep people up-to-date on what you're coding

This requires the project maintainers to keep a weblog hosted on Ohloh itself. Also, other people can break into the timeline by mentioning (linking) the project in their posts. There are currently 934 projects with one or more journal entries, either by maintainers or Ohloh users mentioning the project.

2.4 Promise: Increase awareness of your open source projects

Questionable if this is this is a matter of fact, at the moment.

Kudorank starts at 7 and declines over time. Only receiving a 'kudo' from someone elses Ohloh account increases this score.

kudorank	#contributers
10	63
9	3504
8	10583
7	26465

With this we can estimate the amount of users with increased exposure at around 14.000 (kudorank 8, 9 & 10). Set against the total user and committers base of about 330.000 gives us a 4.2%.

2.5 Promise: Estimating coding behaviour change

Ohloh indexes commit size on projects with public repositories. Looking at changes in the commit size could give an indication of a coding philosophy change.

This was researched by Amit Deshpande & Dirk Riehle in ”Continuous Integration in Open Source Software Development”. Their research hypotheses was: The average commit size in any year is greater than the average commit size during the next year. This was found to be non existant at 95% confidence level. [2]

2.6 Promise: Find good programmers

Many projects, so 1000 query limit is a factor. Commits that shrink the codebase (cite?). Commits that fix bugs, look for common ticket references.

2.7 Peril: Limited to 1000 queries per day

The Ohloh.net website has to say ”Bandwidth will initially be limited to 1,000 requests per API key per day”.

2.8 Peril: Web scraping gets more info than using API

”The design concept is that for each web page on Ohloh, there may be an equivalent XML-formatted version of the page. Currently, only a small subset of the Ohloh site is available as XML, but more data will become available over time.”

2.9 Peril: Can’t find people by programming language

Currently you can’t find people who use one -or several- language(s), without spidering the whole site and scraping it yourself. With the 1000 queries per day limit, one would need about a year (330 days) to scrape the entier user base.

2.10 Peril: Commit stats don’t separate out changed lines

Encountered in [2] is that Ohloh only counts lines removed from the previous state, and lines added in the commit. The amount of overlap is unknown, there is no count of which lines are only slightly changed. The lines changed or overlap measure could be used to asses code churn.

3. CONCLUSIONS

Conclusions here..

4. ACKNOWLEDGMENTS

The structure of the paper was inspired by ”The Promises and Perils of Mining Git” [1].

5. REFERENCES

[1] C. Bird, P. C. Rigby, E. T. Barr, D. J. Hamilton, D. M. German, and P. Devanbu. The promises and perils of mining git. pages 1–10, Apr 2009.

[2] A. Deshpande and D. Riehle. Continuous integration in open source software development. *IFIP International Federation for Information Processing*, 275:273–280, 2008.

[3] S. Fernandez. Rdfohloh, a rdf wrapper of ohloh. *1st workshop on Social Data on the Web (SDoW2008)*, Jan 2008.

[4] L. Shi, D. Berrueta, S. Fernandez, L. Polo, and S. Fernandez. Smushing rdf instances: are alice and bob the same open source developer?