The Promises and Perils of Mining Ohloh.net

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

ABSTRACT

Abstract here..

1. INTRODUCTION

Introduction here..

2. THE *BODY* OF THE PAPER

Body here..

2.1 Related websites

Sourceforge.net.. 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 collect 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.

2.2 Promise: Learn what people are using and contributing to

2.3 Peril: Multiple contributers could be the same person

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. [cite: Smushing RDF]

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

2.5 Promise: Increase awareness of your open source projects

First 63 contributers - rank 10 3504 - rank 9 10583 - rank 7 - the default rank

2.6 Promise: Estimating coding behaviour change

2.7 Find good programmers

Many projects. Commits that shrink the codebase.

2.8 Peril: Limited to 1000 queries per day

2.9 Peril: Web scraping gets more info than using API

2.10 Peril: Can't find people by programming language

Currently you can't find people who use one -or several- language(s), with spidering the whole site and scraping it yourself.

2.11 Peril: Commit stats don't separate out changed lines

Encountered in [Contiuous integration..] is that Ohloh only counts lines removed from the previous state, and lines added in the commit. The amount of overlap is unknown. Lines changed or overlap measure could be used to asses code churn

3. CONCLUSIONS

Conclusions here..

4. REFERENCES

[1] A. Litvinenko. Automatic prediction of source code contribution type. *dspace.utlib.ee*, Jan 2007.