Analyzing OSS Project Health with Heterogeneous Data Sources

Wikan Danar Sunindyo, Vienna University of Technology, Austria, and Bandung Insitute of Technology, Indonesia

Thomas Moser, Vienna University of Technology, Austria

Dietmar Winkler, Vienna University of Technology, Austria

Stefan Biffl, Vienna University of Technology, Austria

ABSTRACT

Stakeholders in Open Source Software (OSS) projects need to determine whether a project is likely to sustain for a sufficient period of time in order to justify their investments into this project. In an OSS project context, there are typically several data sources and OSS processes relevant for determining project health indicators. However, even within one project these data sources often are technically and/or semantically heterogeneous, which makes data collection and analysis tedious and error prone. In this paper, the authors propose and evaluate a framework for OSS data analysis (FOSSDA), which enables the efficient collection, integration, and analysis of data from heterogeneous sources. Major results of the empirical studies are: (a) the framework is useful for integrating data from heterogeneous data sources effectively and (b) project health indicators based on integrated data analyses were found to be more accurate than analyses based on individual non-integrated data sources.

Keywords: Computer Science, Information Systems, Open Source Software, Process and Product Metrics, Project Health Indicators, Project Management

INTRODUCTION

Current Open Source Software (OSS) projects involve a range of stakeholders, from core developers and co-developers to potential users and project investors. Typically, stakeholders, such as potential users or project investors need to know the status and the likely future performance of the project to determine whether the project is likely to sustain for a reasonable period of time in order to justify their investments into the project.

Recent research on using project data to support OSS health monitoring to provide immediate OSS project status, e.g., Sourcerer (Linstead, Bajracharya, Ngo, Rigor, Lopes, & Baldi, 2009), focus on analyzing author-topic relationships in different OSS artifacts to in-
crease understanding of the project and to raise
the awareness on the health status of a project.
Gall, Fluri, and Pinzger (2009) introduced
the Evolizer approach to analyze the software
evolution of OSS projects within Eclipse. This
analysis is useful to investigate the current
stage of OSS to be adapted continuously to
changing environments, business reorientation,
or modernization. Recent research on OSS
project status monitoring includes participation
aspects (Choi, Chengalur-Smith, & Whitmore,
2010), productivity aspects (Wahyudin & Tjoa,
2007), communication aspects (Biffl, Sunindyo,
& Moser, 2010a), and community aspects
(Kaltenecker, 2010). The research presented
in this paper is based on the concept of project
health indicators, which has been introduced
by Wahyudin, Schatten, Mustofa, Biffl, and
Tjoa (2006) for monitoring the health status
of OSS projects during development. Example
indicators that can be used by experts to assess
an OSS project are: (a) service delays on open
issues – the time it takes to fix bugs and issues
listed in the project bug reporting system; (b)
proportions of activity metrics in the community,
e.g., the volume of mailing list postings, bug
status changes per times slot, and updates in the
SVN to learn the health of relationships between
relevant activities, e.g., activities on the same
bug; and (c) communication and use intensity.
In a healthy project community, a reasonable
relationship can be expected between measures
such as the number of downloads, mailing list
postings, and developer interactions in mailing
lists (Wahyudin, Mustofa, Schatten, Biffl, &
Tjoa, 2007).

However, challenges for monitoring the
health status of OSS projects easily and fre-
tently are: (a) manual data collection and
integration from heterogeneous data sources,
i.e., data sources, which represent common
project-level concepts in various data formats
that are non-trivial to reconcile, tend to be prone
to errors and take considerable effort to integrate
(Conklin, 2006); (b) the need to correlate data on
different activities requires data integration; (c)
manual data validation of the integrated data is
hard due to different representation of common
concepts, e.g., different names for one person
in the data models involved; (d) data analyses
of individual data sources, e.g., mailing lists,
bug database (Mockus, Fielding, & Herbsleb,
2002), SVN/CVS (German, 2004), and change
logs (Chen, Schach, Yu, Offutt, & Heller, 2004)
have been shown to be weak to detect the health
status of OSS project accurately; and (e) the
large amount of data to maintain for analysis
in an OSS project over time takes significant
resources for storing.

In this paper, we propose and evaluate a
framework for OSS data analysis, FOSSDA,
which enables the efficient collection, integra-
tion, and analysis of data from heterogeneous
sources. This framework provides the following
contributions to address OSS project health
monitoring challenges: (a) a process with sem-
antic tool support to make data collection and
integration from heterogeneous data sources
more efficient; (b) adaptation of ontology-based
querying techniques to OSS project monitoring,
which makes data validation simpler and more
effective; (c) the combination of different project
metrics for analysis purposes is expected to
improve project health analysis accuracy over
the analysis based on individual data sources
only; (d) the use of an ontology to represent OSS
knowledge based on well-defined semantics
and to provide extensive querying capabilities
(Biffl, Sunindyo, & Moser, 2010b).

The empirical evaluation of the FOSSDA
approach focuses on two research issues, namely
(a) a feasibility study of FOSSDA in a pilot ap-
lication with several OSS projects and (b) an
integrated data model that can be used to derive
a health indicator model to assess OSS project
status with reasonable accuracy. Major results
show that (a) the proposed framework supported
efficient data collection and analysis compared
to the traditional approach in the study context
and (b) the integrated data model supported a
more accurate analysis of OSS project health
indicators.

The remainder of this paper is structured as
follows. Second section discusses related work
on health indicators and current frameworks
on analyzing OSS project data. Third section
Open Source Software: Strengths and Weaknesses
www.igi-global.com/chapter/open-source-software/21188?camid=4v1a

Morality and Pragmatism in Free Software and Open Source
www.igi-global.com/chapter/morality-pragmatism-free-software-open/21176?camid=4v1a