Branch Use in Practice

A Large-Scale Empirical Study of 2,923 Projects on GitHub

Weiqin Zou*, Weiqiang Zhang*, Xin Xia†, Reid Holmes‡, Zhenyu Chen*
*State Key Lab of Novel Software Technology, Nanjing University, China
†Faculty of Information Technology, Monash University, Australia
‡Department of Computer Science, University of British Columbia, Canada
wqzou, zhangweiqiang@smail.nju.edu.cn, xin.xia@monash.edu, rtholmes@cs.ubc.ca, zychen@nju.edu.cn

Abstract—**Branching is often used to help developers work in parallel during distributed software development. Previous studies have examined branch usage in practice. However, most studies perform branch analysis on industrial projects or only a small number of open source software (OSS) systems. There are no broad examinations of how branches are used across OSS communities. Due to the rapidly increasing popularity of collaboration in OSS projects, it is important to gain insights into the practice of branch usage in these communities. In this paper, we performed an empirical study on branch usage for 2,923 projects developed on GitHub. Our work mainly studies the way developers use branches and the effects of branching on the overall productivity of these projects. Our results show that: 1) Most projects use a few branches (<5) during development; 2) Large scale projects tend to use more branches than small scale projects. 3) Branches are mainly used to implement new features, conduct version iteration, and fix bugs. 4) Almost all master branches have been requested by contributors to merge their contributions; 5) There always exists a branch playing a more important role in merging contributions than other branches; 6) Almost all commits of more than 75% branches are included in the master branches; 7) The number of branches used in a project has a positive effect on a project's productivity but the effect size is small, and there is no statistically significantly difference between personal projects and organizational projects.**

Index Terms—**branch use, GitHub, exploratory study**

I. Introduction

Along with the rapid growth of both project scale and team size in modern software development, there comes an important and challenging problem: enabling developers to collaborate and develop projects in parallel without interfering with one another [5]. One common method to address this problem is with branches within version control systems [3]. Many advanced version control systems, such as Git[1] and SVN[2], have provided good support for the feature of branches.

When developers plan to perform specific tasks such as bug fixing or feature implementation without affecting the main stream development, they often create a branch. Then they will work on this new branch independently without interfering with other developers. After they finish coding and testing, they then merge their branch that they were changing back into the branch they originally branched from, or they invite another developer to help perform the merge for them [32]. In this way, branching makes it possible for developers to work on their own workspace without being disturbed or disturbing others unnecessarily.

With the above benefits of branching, many OSS projects (such as Python) and commercial companies (such as Microsoft) adopt branching strategies to facilitate the process of software development [5]. However, branching has a cost. Some developers do not fully understand branching and abusing of branches can hinder development [1]. This can result in problems such as integration failures and release delays if branches are used incorrectly.

To help developers better use branches, some researchers have created branching best practices [55], [40], [1]. Others try to learn the branch usage in practice and its potential impact on software development [43], [39]. There also exist approaches that propose solutions to problems introduced by using branches [5], [36]. Unfortunately, most existing research studies are either largely based on researchers' own experience, are targeted at a small number of OSS projects, or are limited to individual industrial projects. To the best of our knowledge, there are no large-scale empirical studies on developers' behaviours of branch usage in practice. As such, we do not yet have an overview about how branches are used in practice across the breadth of OSS communities.

Fortunately, GitHub[3] makes it possible to deeply investigate branch usage across a large number of practical projects. GitHub is a platform based on Git, which provides code hosting and distributed collaboration [26]. As of January 2017 more than 50 million repositories were hosted on GitHub.

In this paper, we investigated the current state of branch usage in OSS communities. Specifically, we conducted an empirical analysis on 2,923 GitHub projects that have been developed over at least five years. We first obtained an overview of branch usage in GitHub. Then we studied the purposes of branching. Next, we investigated the roles that branches take in coping with contributions by others. We also studied the commits flowing from non-master branches into master branches. Finally, we studied the impact of using branches on the overall productivity of projects. Our major contributions are listed as follows:

[1] http://git-scm.com/

[2] http://subversion.apache.org/

[3] http://github.com/