

Государев Илья Борисович,

факультет программной инженерии и компьютерной техники

Университет ИТМО

Санкт-Петербург

институт компьютерных наук и технологического образования

Российский государственный педагогический университет

им.А.И.Герцена,

Санкт-Петербург

доцент

goss@itmo.ru, igossoudarev@herzen.spb.ru,

Кочетыгов Андрей Владимирович,

факультет программной инженерии и компьютерной техники

Университет ИТМО

Санкт-Петербург

студент

ko4etygov.a@yandex.ru,

Анализ применения непрерывной интеграции среди проектов на GitHub

Аннотация: В работе рассматривается понятие непрерывной интеграции, приводится статистика применения инструментов непрерывной интеграции, а также данные о корреляции между применением непрерывной интеграции и такими характеристиками проекта, как используемый язык программирования и популярность среди пользователей.

Ключевые слова: интеграция, автоматизация, тестирование, непрерывная интеграция, языки программирования.

Gosudarev Ilya Borisovich,
associate Professor of the Computer science and technological education
institute
associate Professor of the program engineering and computer technics
faculty
candidate of pedagogical sciences,
The Herzen State Pedagogical University of Russia,
ITMO University
Saint-Petersburg
igossoudarev@herzen.spb.ru, goss@corp.ifmo.ru

Kochetygov Andrey Vladimirovich,
master student of the program “Web-technology”
software engineering and computer technics faculty
ITMO University
Saint-Petersburg
ko4etygov.a@yandex.ru,

Analysis of Continuous Integration Usage Among GitHub Projects

Abstract: The paper considers the concept of continuous integration, provides statistics on the usage of continuous integration tools, as well as data on the correlation between the usage of continuous integration and such project characteristics as the programming language used and popularity among users.

Ключевые слова: GitHub, CI, CI/CD, DevOps, integration, automatization, testing, continuous integration, programming languages.

В настоящее время требования к программным продуктам со стороны конечных пользователей и заказчиков неуклонно растут. Вместе с этим растет масштабность проектов и соответственно проектных команд. Кроме того, разделение обязанностей в командах становится все более ярко

выраженным. Все это способствует возникновению сопутствующих проблем свойственных человеческому взаимодействию в больших группах.

Одной из таких проблем является проблема интеграции отдельных частей проекта в единое целое с последующим тестированием. Наиболее эффективным решением в данном случае может стать практика непрерывной интеграции, так как она позволяет в большом объеме автоматизировать процедуру интеграции и тестирования. Она требует ответственного подхода к внедрению, но в значительной степени окупается, так как помогает эффективно решать проблемы одного из самых подверженных издержкам процесса в ходе разработке программного обеспечения.

В данном исследовании проводится анализ понятия непрерывной интеграции и факторов влияющих на ее внедрение в производственный процесс проектов.

Впервые словосочетание «Continuous Integration», упоминается в работе «Object-oriented analysis and design with applications» за авторством Грэди Буча, которая была опубликована в 1994 году [4].

Затем важность практики непрерывной интеграции рассматривается в статье «Extreme programming: A humanistic discipline of software development» за авторством Кента Бека, которая была опубликована в 1998 году [2]. Кроме того, он дал наиболее полное описание практики непрерывной интеграции в своей книге под названием «Extreme programming explained: Embrace change», опубликованной в 1999 году [3].

Приведенные исторические факты также отмечаются в других исследованиях посвященных практике непрерывной интеграции [5, 6].

Однако набирать значимую популярность практика непрерывной интеграции начала с 2000 года, после того как 10 сентября Мартин Фаулер и Мэтью Феммель опубликовали статью, которая описывает понятие непрерывной интеграции и ее особенности [1, 5, 6, 7].

Так Мартин Фаулер дает следующее определение рассматриваемой концепции: «Непрерывная интеграция – способ разработки программного обеспечения, при котором все участники группы осуществляют частую интеграцию результатов своей работы. Обычно каждый человек выполняет интеграцию ежедневно, что приводит к нескольким интеграциям в день. Результат каждой интеграции автоматически проверяется на предмет ошибок по возможности быстрее. Большинство групп находит, что данный подход ведет к значительному уменьшению количества проблем интеграции и позволяет ускорить разработку связанного программного обеспечения» [1, с. 48].

Согласно имеющимся данным, около 40% проектов из рассматриваемой выборки уже применяют в своей производственной деятельности непрерывную интеграцию. Данная статистика применения представлена в таблице 1, а соответствующая статистике визуализация на рисунке 1 [5].

Таблица 1 – Статистика применения непрерывной интеграции

Применяется	Процент	Кол-во проектов
Да	40.27%	13910
Нет	59.73%	20634

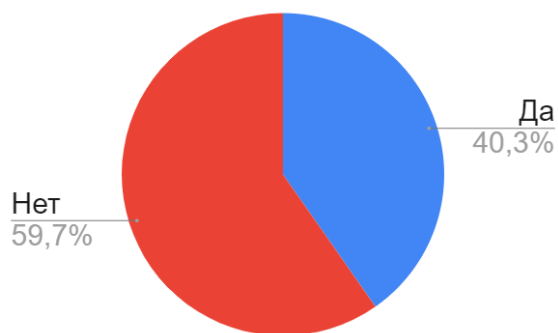


Рисунок 1 – Статистика применения непрерывной интеграции

Около 90% проектов, применяющих непрерывную интеграцию, используют для этого систему Travis CI, а около 14% проектов сразу несколько систем. Статистика использования система представлена в таблице 2, а соответствующая статистике визуализация на рисунке 2 [5].

Таблица 2 – Статистика используемых систем непрерывной интеграции

Проекты	<i>Travis</i>	<i>CircleCI</i>	<i>AppVeyor</i>	<i>CloudBees</i>	<i>Werker</i>
Доля	90.1%	19.1%	3.5%	1.6%	0.4%
Кол-во	12528	2657	484	223	59

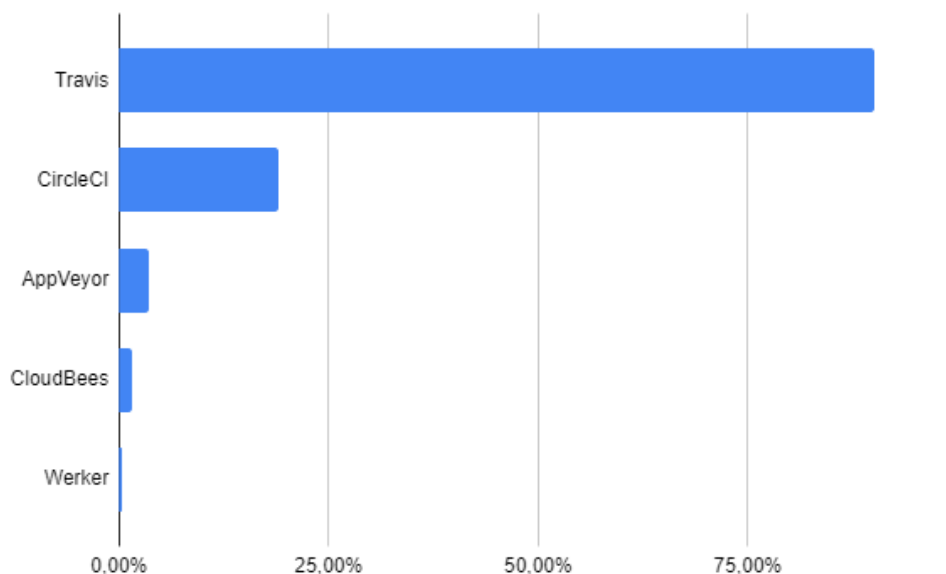


Рисунок 2 – Статистика применения сервисов непрерывной интеграции

Сервис *GitHub* предоставляет возможность положительной оценки проектов с помощью специализированной метрики – звезд. Когда пользователь сервиса ставит проекту звезду, то таким образом он изъявляет свой интерес и симпатию к проекту. Таким образом, количество звезд у проекта позволяет судить о его популярности среди пользователей сервиса.

Согласно имеющимся данным, между популярностью проекта и применением непрерывной интеграции существует положительная зависимость. Так применение непрерывной интеграции на популярных проектах достигает 70%, в то время как среди менее популярных лишь 23-40%. Описанная положительная зависимость представлена на рисунке 3 [5].

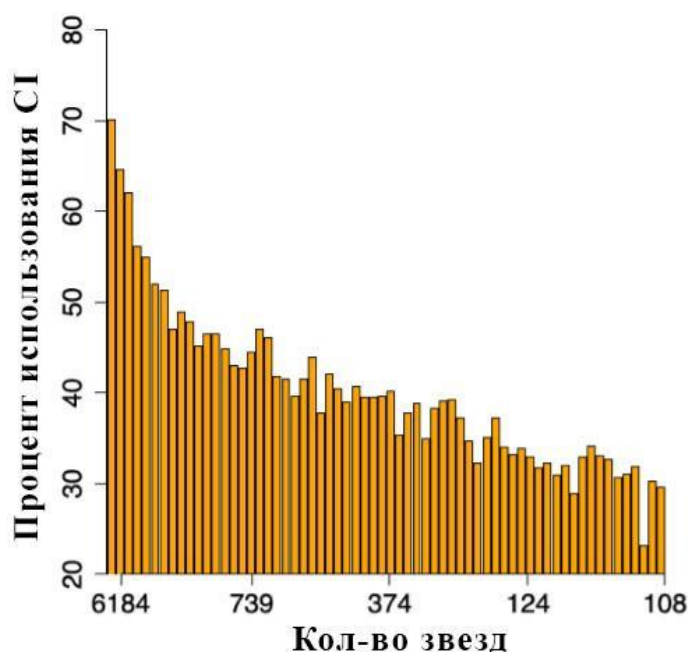


Рисунок 3 – Зависимость между популярностью проекта и применением непрерывной интеграции [5]

В то же время применение непрерывной интеграции не зависит от популярности используемого языка программирования. Однако проекты, использующих языки программирования с динамической типизацией, такие как Ruby, PHP, CoffeeScript, Clojure, Python и JavaScript, обладают более высоким процентом применения непрерывной интеграции [5].

Приведенная ранее статистика указывает на то, что применение непрерывной интеграции востребовано на проектах, где критически важна стабильность основной кодовой базы. Таким образом данный факт позволяет рекомендовать применение непрерывной интеграции на проектах, обладающих высоким спросом на рынке, а также использующих языки программирования с динамической типизацией для обеспечения повышенной стабильности основной кодовой базы.

Литература

1. Дюваль, П. М. Непрерывная интеграция: улучшение качества программного обеспечения и снижение риска / П. М. Дюваль, С. М. Матиас III, Э. Гловер // Вильямс. — Москва, 2008. — 240 с.

2. Beck, K. Extreme Programming: A Humanistic Discipline of Software Development in Fundamental Approaches to Software Engineering. Springer / K. Beck // Addison-Wesley. — Santa Clara, 1998.
3. Beck, K. Extreme Programming Explained: Embrace Change / K. Beck, C. Anders // Addison-Wesley. — Boston, 1999.
4. Grady, B. Object-oriented analysis and design with applications / B. Grady, R. A. Maksimchuk, M. W. Engel, B. J. Young, J. Conallen, K. A. Houston // Addison-Wesley. — Boston, 1994.
5. Hilton, M. Usage, Costs, and Benefits of Continuous Integration in Open-Source Projects / M. Hilton, T. Tunnell, K. Huang, D. Marinov, D. Dig // Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering. — IEEE, 2016. — P. 426–437.
6. Lai, S. Applying Continuous Integration for Increasing The Maintenance Quality And Efficiency of Web App / S. Lai // International Journal of Software Engineering & Applications (IJSEA). — 2019. — Vol. 10. — № 1. — P. 37–50.
7. Mårtensson, T., Continuous Integration is Not About Build Systems / T. Mårtensson, P. Hammarström, J. Bosch // Proceedings of the 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA). — IEEE, 2017. — P. 1-9.