

Usage, Costs, and Benefits of Continuous Integration in Open- Source Projects



Michael Hilton Timothy Tunnell Kai Huang

Darko Marinov Danny Dig

Continuous Integration



Travis CI



Jenkins

● 継続的インテグレーション

- ▶ ビルドやテストを頻繁に繰り返し行い、効率化や省力化を図る手法
 - 専用のツールを用いることで自動化・半自動化
- ▶ 問題を早期に発見することができる

背景

- CIを採用することでの成功事例が多く報告されている
 - ▶ Flickrが1日に10回以上デプロイすることに貢献していると報告[22]
 - ▶ HPの製品グループは開発コストを78%削減したと主張[40]
- 人気サービスのTravis CI^[17]は利用プロジェクトは30万件以上
- Google Trend^[11]では、過去10年で
[Continus Integration]の検索が350%増加

[11] Google Search Trends. <https://www.google.com/trends/>. Accessed: 2016-04-24.

[17] Travis CI. <https://travis-ci.org/>. Accessed: 2016-04-21.

[22] J. Allspaw and P. Hammond. 10+ deploys per day: Dev and ops cooperation at Flickr. <https://www.youtube.com/watch?v=LdOe18KhtT4>. Accessed: 2016-04-21.

[40] J. Humble. Evidence and case studies. <http://continuousdelivery.com/evidence-case-studies/>. Accessed: 2016-04-29.

問題点

- CIシステムに関する体系的な調査は少ない（この論文が発表された時点）
- CIについての情報が少ないため具体的なメリットやデメリットが知られていない

目的

- OSSでのCIの使用方法を調べる
 - ▶ CIが開発者によって、どの程度採用されているか
 - ▶ どのCIシステムを使用しているか
 - ▶ CIをどのように使用しているか

リサーチクエスチョン

◎ CIの使用法

1. オープンソフトウェアのCIの使用率は？
2. それぞれのCIサービスの使用状況は？
3. CIを多く利用している特定のタイプはあるか？
4. いつCIを採用するか？
5. 開発者はCIの使用を継続する予定か？

リサーチクエスチョン

◎ CIのコスト

- 6.なぜCIを使用しないのか？
- 7.どのくらいの頻度でCIの設定を変更するか？
- 8.どのような理由でCIの設定を変更するか？
- 9.CIのビルド時間の平均は？

リサーチクエスト

◎ CIのメリット

10.なぜCIを利用するのか？

11.CIを利用しているプロジェクトでは、頻繁にリリースが行われるか？

12.CIを利用しているプロジェクトは、より多くのプルリクエストを受け入れるか？

13.CIビルドでプルリクエストを受け入れる時間は短縮されるか？

14.CIビルドは非マスターブランチよりもマスターで失敗が少ないか？

データの取得方法

- ◎ Github API
- ◎ Cloud-Bees[6]による問い合わせ
- ◎ GH-Torrent[39]
 - ▶ GitHubから人気の34,533プロジェクト
 - ▶ 各プロジェクトのプルリクエスト653,404件
 - ▶ CIシステムから構築された1,529,291ビルド
 - ▶ 442人へのアンケート調査

[6] CloudBees. <http://cloudbees.com/>. Accessed: 2016-04-26.

[39] G. Gousios. The GHTorrent dataset and tool suite. In MSR, 2013.

RQ1 結果

.....

●オープンソフトウェアのCIの使用率は？

Table 1: Breadth corpus projects CI usage		
Project Uses CI?	Percentage	Number of Projects
Yes	40.27%	13,910
No	59.73%	20,634

全体の約40%がCIを使用している

RQ2 結果

- それぞれのサービスの使用率は？

Usage by CI Service				
Travis	CircleCI	AppVeyor	CloudBees	Werker
90.1%	19.1%	3.5%	1.6%	0.4%
12528	2657	484	223	59

Travis CIが最も利用されている

RQ3 結果

● CIを多く利用している特定のタイプはあるか？

人気のあるプロジェクトではCIの使用率が高い

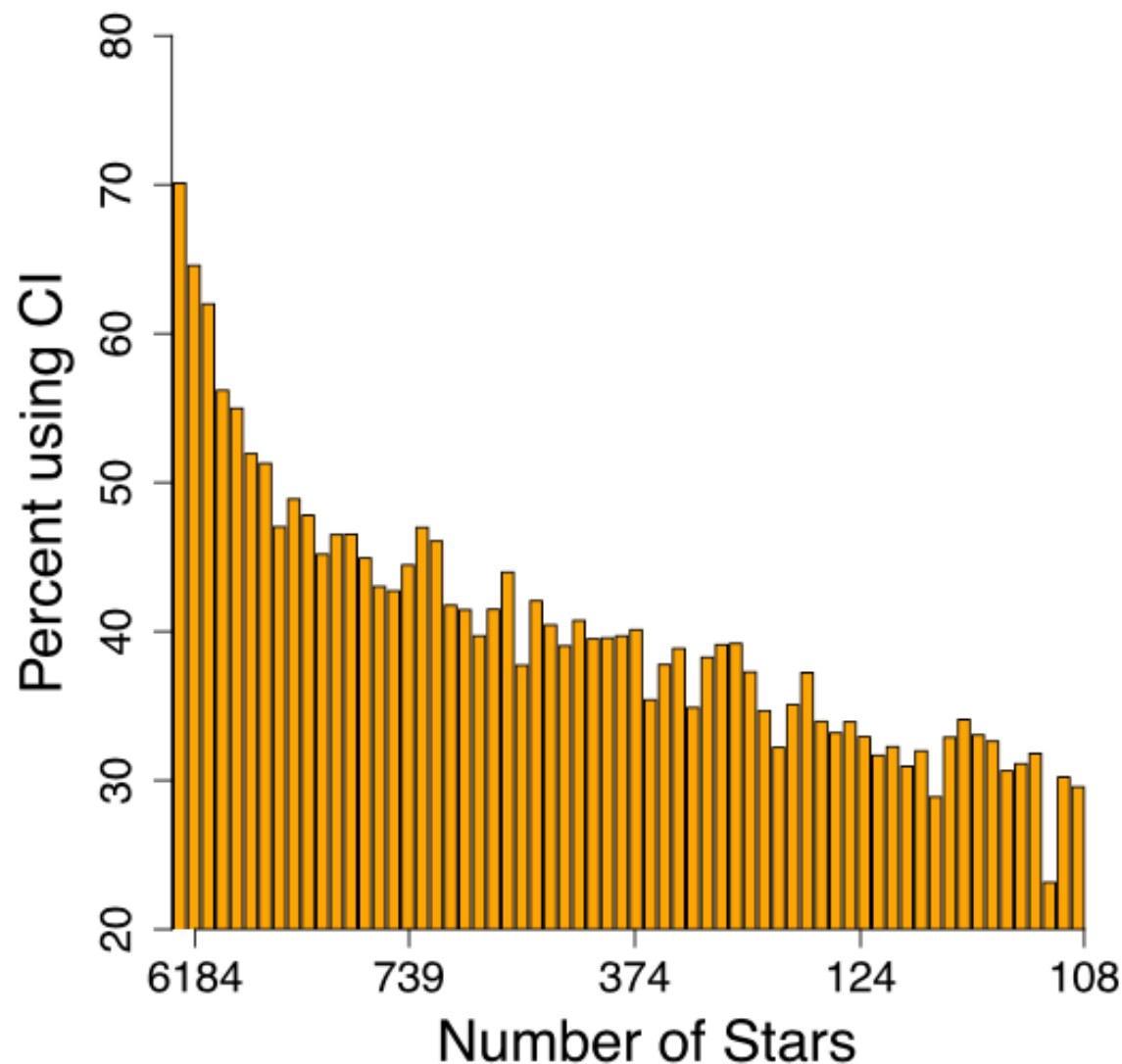
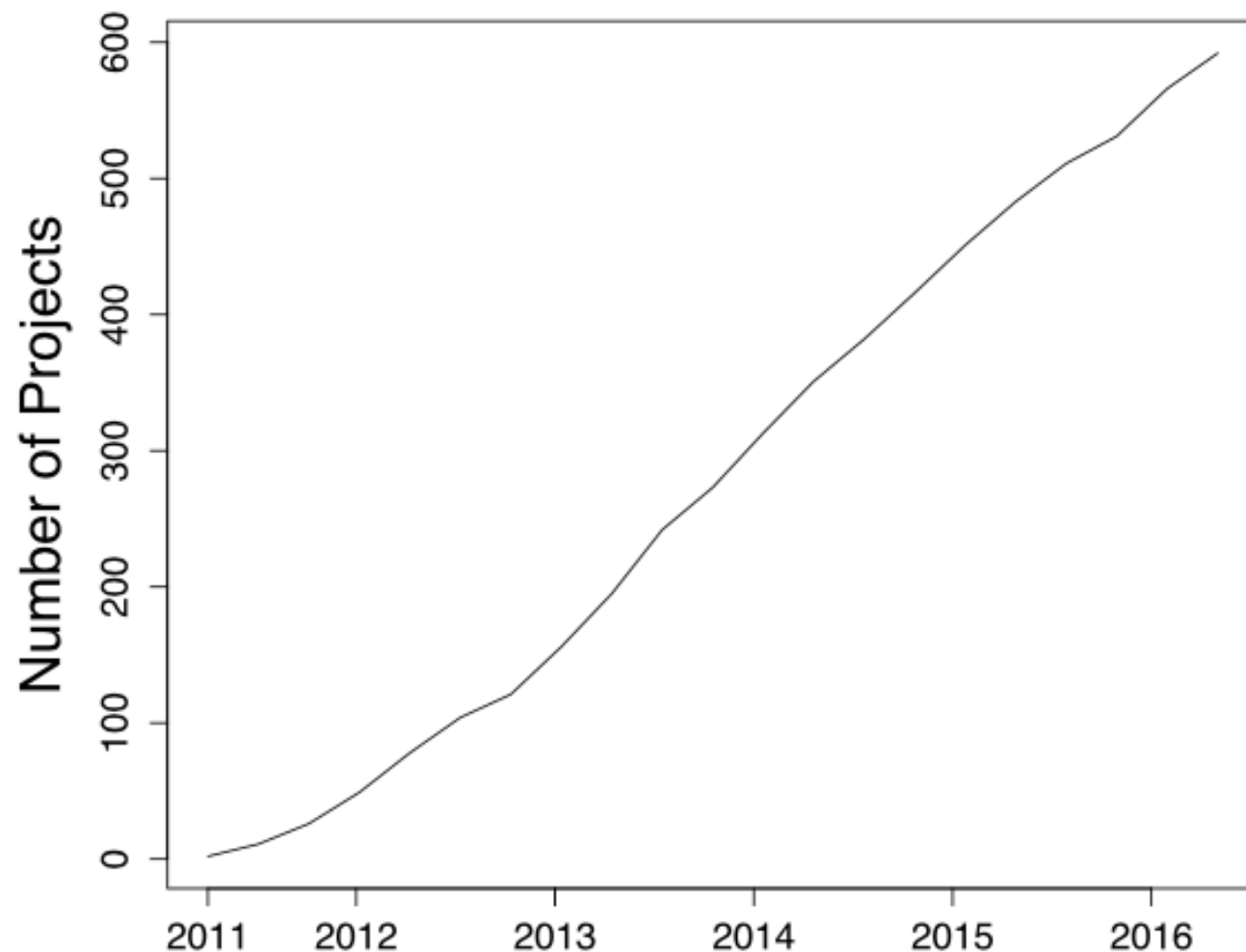


Figure 1: CI usage of projects in GitHub. Projects are sorted by popularity (number of stars).

RQ4 結果

●いつCIを採用するか？(Travis CIからデータ収集)



CIは過去5年で成長している

CIを採用するまでの期間の中央値は1年

Figure 2: Number of projects using CI over time. Data is tabulated by quarter (3 months) per year.

RQ5 結果

●開発者はCIの使用を継続する予定か？

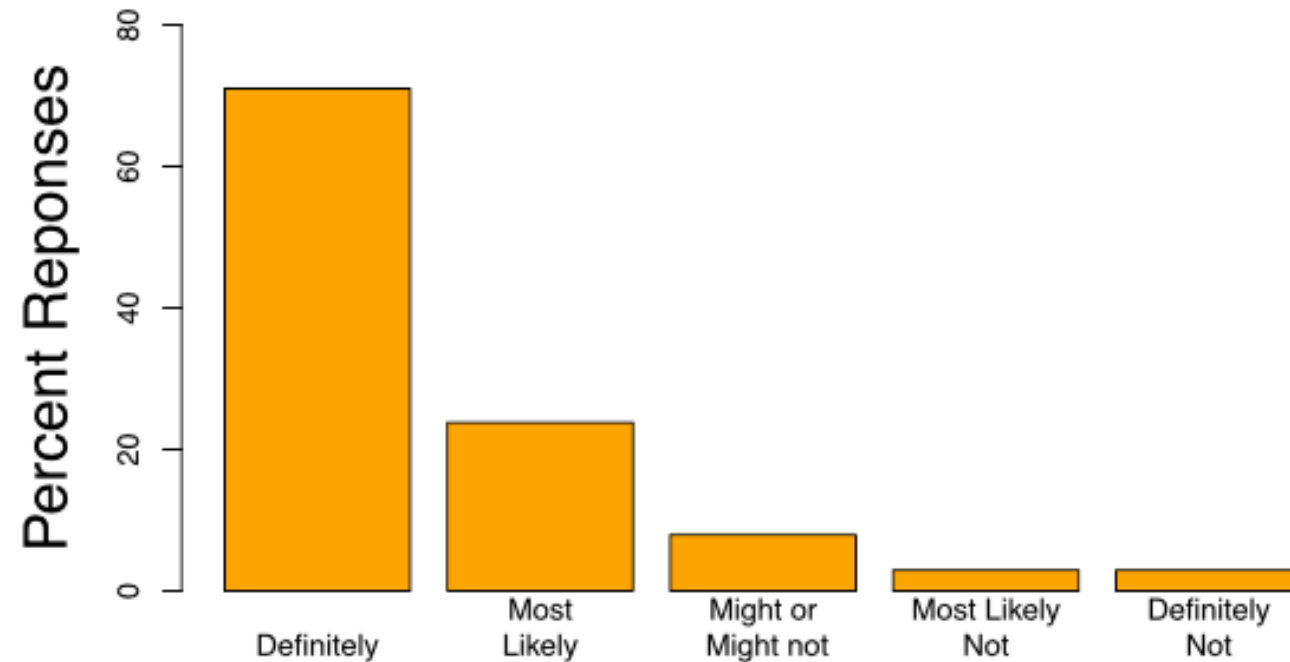


Figure 3: Answers to “Will you use CI for your next project?”

94%が「使用する」「たぶん使用する」と回答

RQ6 結果

●なぜCIを使用しないのか？

Table 4: Reasons developers gave for not using CI

Reason	Percent
The developers on my project are not familiar enough with CI	47.00
Our project doesn't have automated tests	44.12
Our project doesn't commit often enough for CI to be worth it	35.29
Our project doesn't currently use CI, but we would like to in the future	26.47
CI systems have too high maintenance costs (e.g., time, effort, etc.)	20.59
CI takes too long to set up	17.65
CI doesn't bring value because our project already does enough testing	5.88

開発者がCIに慣れていない
ことが主な原因

RQ7 結果

- どのくらいの頻度でCIの設定を変更するか？ (Travis-CI)

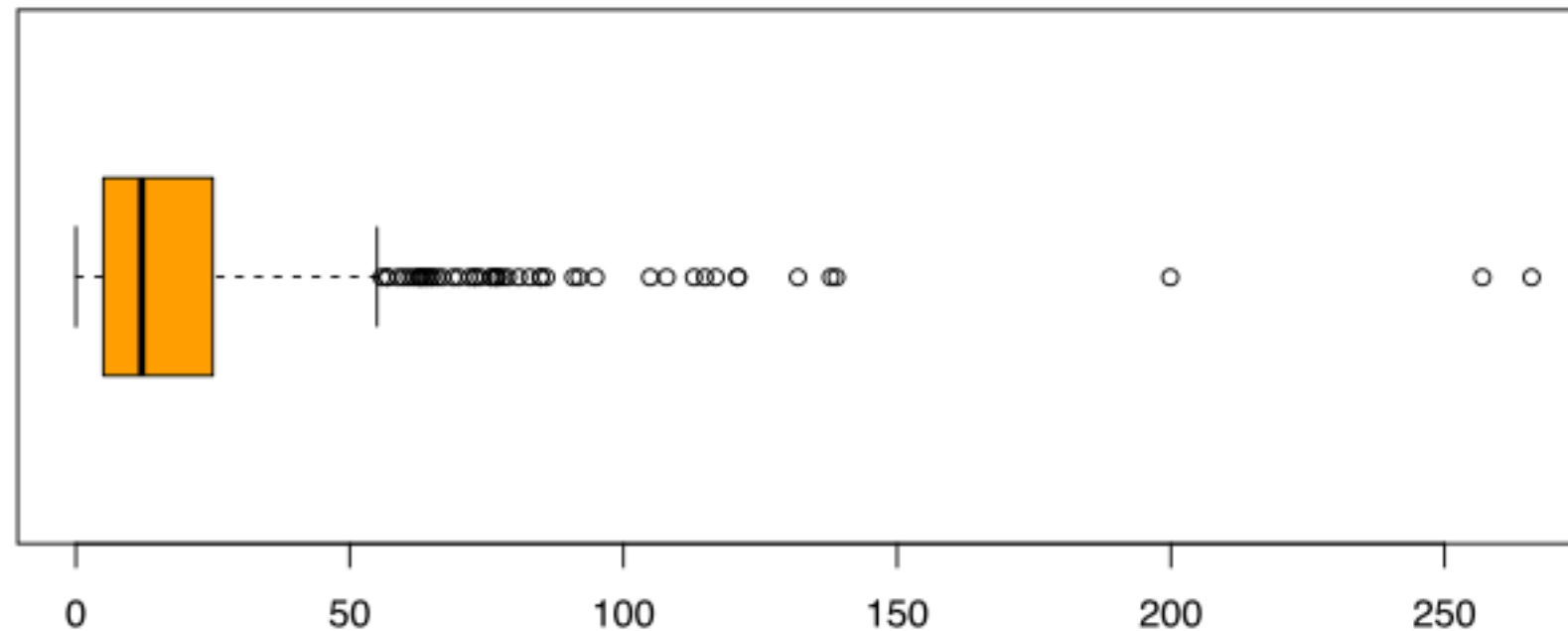


Figure 4: Number of changes to CI configs, median number of changes is 12

ほとんどのプロジェクトは設定をほとんど変更しない

しかしプロジェクトによっては頻繁に変更する

RQ8 結果

● どのような理由でCIの設定を変更するか？(Travis-CI)

Table 5: Reasons for CI config changes

Config Area	Total Edits	Percentage
Build Matrix	9718	14.70
Before Install	8549	12.93
Build Script	8328	12.59
Build Language Config	7222	10.92
Build Env	6900	10.43
Before Build Script	6387	9.66
Install	4357	6.59
Whitespace	3226	4.88
Build platform Config	3058	4.62
Notifications	2069	3.13
Comments	2004	3.03
Git Configuration	1275	1.93
Deploy Targets	1079	1.63
After Build Success	1025	1.55
After Build Script	602	0.91
Before Deploy	133	0.20
After Deploy	79	0.12
Custom Scripting	40	0.06
After Build Failure	39	0.06
After Install	14	0.02
Before Install	10	0.02
Mysql	5	0.01
After Build Success	3	0.00
Allow Failures	2	0.00

ランタイム・環境・除外などを
指定するビルドマトリックス

ビルドする前にインストールす
る依存ライブラリ

依存関係によるバージョン
変更

RQ9 結果

● CIのビルド時間の平均は？

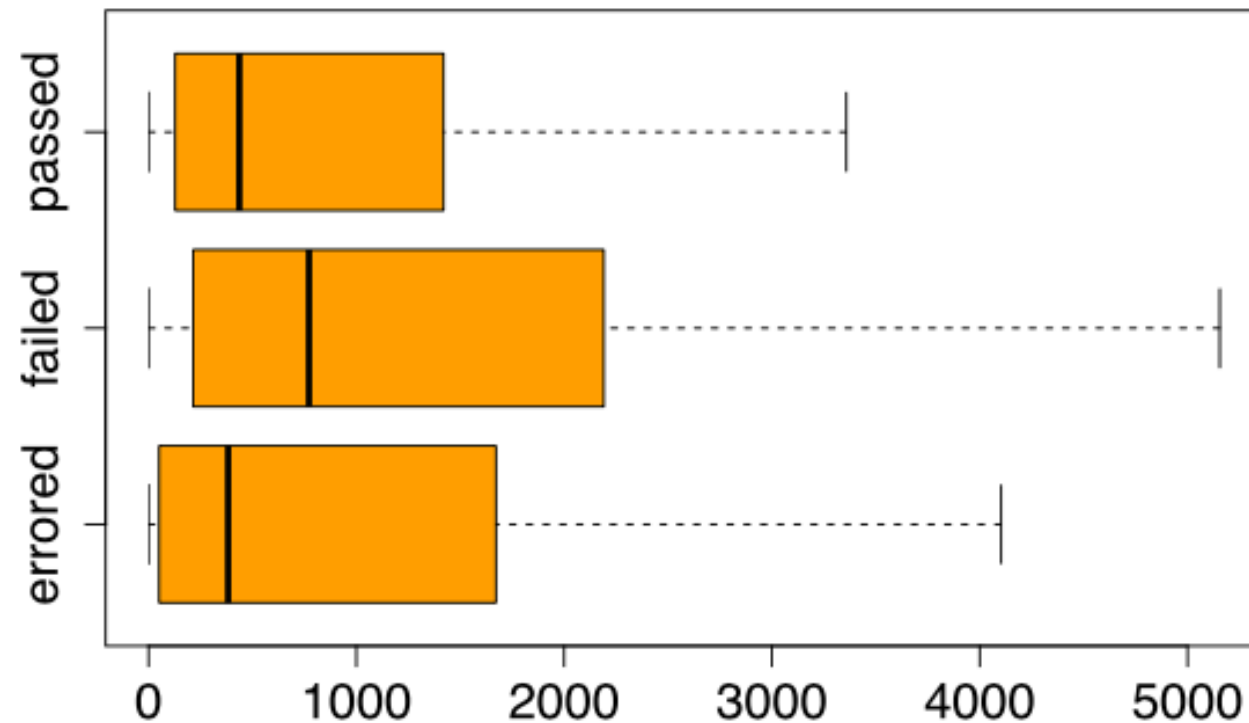


Figure 5: Build time distribution by result, in seconds

平均時間は500秒

RQ10 結果

●なぜCIを利用するのか？

Table 6: Reasons for using CI, as reported by survey participants

Reason	Percent
CI makes us less worried about breaking our builds	87.71
CI helps us catch bugs earlier	79.61
CI allows running our tests in the cloud, freeing up our personal machines	54.55
CI helps us deploy more often	53.32
CI makes integration easier	53.07
CI runs our tests in a real-world staging environment	46.00
CI lets us spend less time debugging	33.66

自分のビルドが壊れていることへの心配を減
らす

バグを早くにみつけることを助ける

RQ11 結果

- CIを利用しているプロジェクトでは、頻繁にリリースが行われるか？

Table 7: Release rate of projects	
Uses Travis	Versions Released per Month
Yes	.54
No	.24

CIを使用しているプロジェクトのほうが月に平均54回
リリースしておりリリース回数が多い

RQ12 結果

- CIを利用しているプロジェクトは、より多くのプルリクエストを受け入れるか？

Table 8: Comparison of pull requests merged for pull requests that had or did not have CI information

CI Usage	% Pull Requests Merged
Using CI	23
Not Using CI	28

CI情報のないプルリクエストの方がマージされる可能性が高い

CI情報ありのプルリクエストは問題をマージする前に問題を識別できると考えられる

RQ13 結果

- CIビルドでプルリクエストを受け入れる時間は短縮されるか？

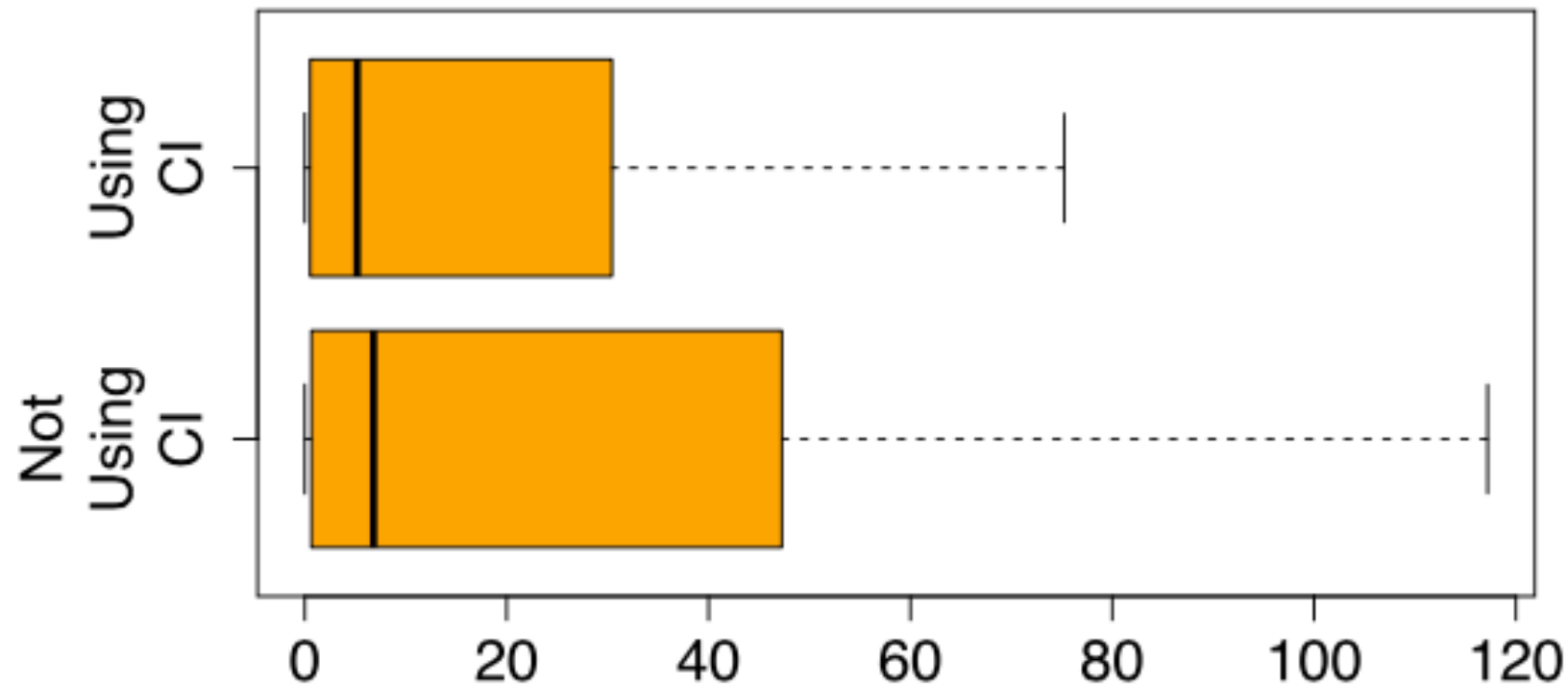


Figure 6: Distribution of time to accept pull requests, in hours

CIのビルド状況により、プル要求に迅速に対応できる

CI情報なしより中央値1.6時間早く対応する

RQ14 結果

.....

- CIビルドは非マスターブランチよりもマスターで失敗が少ないか？

Table 9: Percentage of builds that succeed by pull request target

Pull Request Target	Percent Passed Builds
Master	72.03
Other	65.36

マスターブランチに対するプルリクエストは、ビルドを成功する確率が高いことがわかった

まとめ

- プロジェクト・ビルド・開発者へのアンケートを調査することで14個のRQを調査した
- CIが人気を集めている理由を解明した
 - ▶ リリースの頻度が約2倍高い
 - ▶ プル要求を早く受け入れる
 - ▶ 破損したビルドを提供することへの心配を減らす

所感

- CIについての研究は、まだまだされていないことが多いと感じた
- CIを使用しているプロジェクトと利用していないプロジェクトでの不具合報告の数の差は？内容の差は？
- 減らせるバグと減らせないバグに特徴はあるのか？