

Empirical Study of Restarted and Flaky Builds on Travis CI

Thomas Durieux, Claire Le Goues, Michael Hilton, Rui Abreu

Proceedings of the 17th International Conference on Mining Software Repositories, pages 254-264

発表者：辻 裕真

背景

- ・ 継続的インテグレーション(CI)はソフトウェア開発のベストプラクティスとみなされ広く普及している.
- ・ CIビルドが予期せず失敗した際, 開発者は手動で再びビルドを行う.(restarted build)
- ・ restarted buildのプロセスとワークフローに与える影響を理解し, 開発者のrestarted buildに対する負荷を減らすことが目的.

Restarted build

- CIのワークフロー中にビルドが予期せず失敗した際に、プロジェクトに対して変更を加えずに手動でもう一度ビルドを行うこと.
- 一般的に環境の不安定さが原因であると思われる予期しない動作を確認した時にrestarted buildを行う.
- restarted buildを行い出力が変化していた場合は、失敗の原因がシステムの不安定さやCI環境の問題であると分かる.

Research Questions

RQ1 : 開発者はどのくらいの頻度でrestarted buildを行うか.

RQ2 : restarted buildが行われるプロジェクトに特性はあるか.

RQ3 : 開発者がrestarted buildを行うのはなぜか.

RQ4 : 不安定なビルドが開発ワークフローにどんな影響を与えるか.

データ収集

- 2019年10月8日から2019年12月19日までのTravisCIのビルド情報をリアルタイムで収集し, 171,057のリポジトリから, 3,286,773のビルドを収集.
- 少なくとも1つのrestarted buildを含むリポジトリのコミット履歴を抽出し, 22,345のリポジトリから56,522のrestarted buildを収集.

結果

RQ1：開発者はどのくらいの頻度でrestarted buildを行うか.

Table 1: Overview of the collected data.

Metric Name	Value
Start Study	Tue, 08 Oct 2019
End Study	Thu, 19 Dec 2019
# Builds	3,286,773
# Jobs	9,215,866
# Logs	3,007,857
# Repositories	171,057
# Passed builds	2,306,130 (70%) 
# Failed builds	583,415 (18%) 
# Errored builds	367,963 (11%) 
# Canceled builds	29,265 (0.9%) 
Avg. # builds per repo	19
Avg. # jobs per repo	52

Table 2: Overview of the restarted builds.

Metric Name	Value
# Restarted Builds	56,522 (1.7%) 
# Restarted Jobs	125,461 (1.4%) 
# Restarted Logs	61,783 (2.1%) 
# Restarted Repositories	22,345 (13%) 
# Restart Passed builds	9,707 (17%) 
# Restart Failed builds	27,006 (48%) 
# Restart Errored builds	17,621 (31%) 
# Restart Canceled builds	2,186 (3.9%) 

- 全てのビルド(3,286,773)のうち56,522(1.72%)がrestarted buildであった.
- restarted buildは一般的なタスクではないことが分かった.

結果

RQ1：開発者はどのくらいの頻度でrestarted buildを行うか.

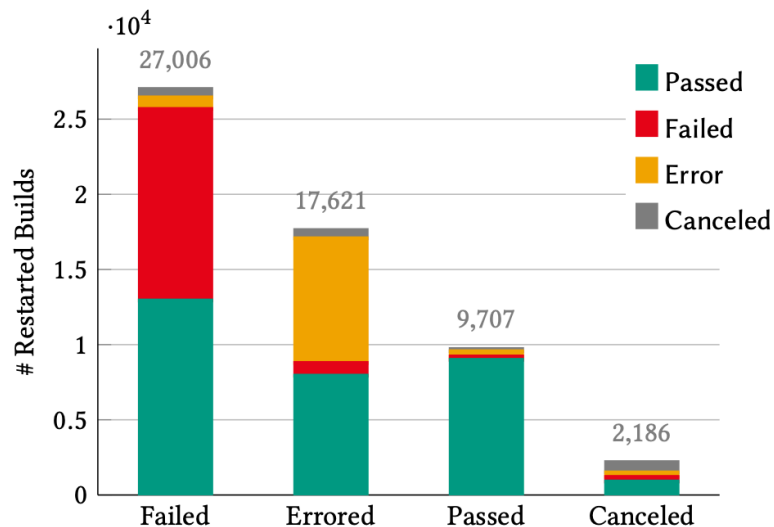


Figure 3: Evolution of the build status between original builds and restarted builds.

- 図はrestarted buildの元の状態と、restart後の状態を示している.
- 失敗/エラーのあったビルドの46.77%がrestart後に成功していることが分かる.
- この変化はビルドが不安定であることを示している.
- 主にテストが不安定であることが多かった.

結果

RQ2 : restarted buildが行われるプロジェクトに特性はあるか.

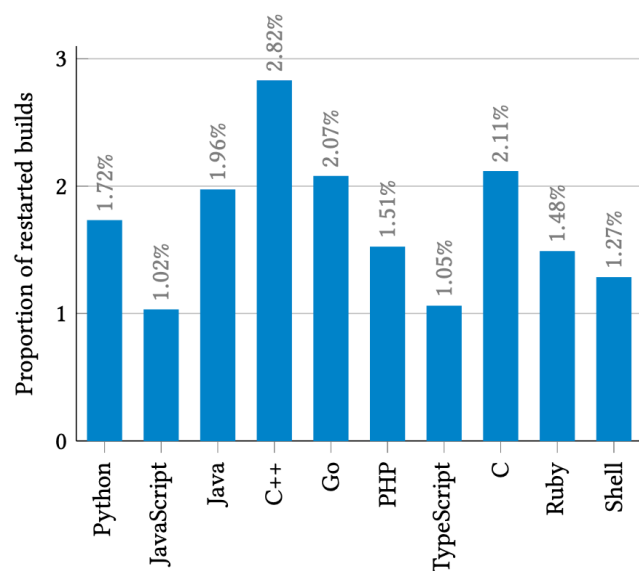


Figure 5: The restarted rate per language.

- 図は言語ごとのrestarted buildが行われる割合.
- JavaScriptは1.02%, TypeScriptは1.05%であるがその他のプロジェクトは1.27~2.82%であった.
- この違いを説明するために言語ごとのビルドに関する情報を分析する.

結果

RQ2 : restarted buildが行われるプロジェクトに特性はあるか.

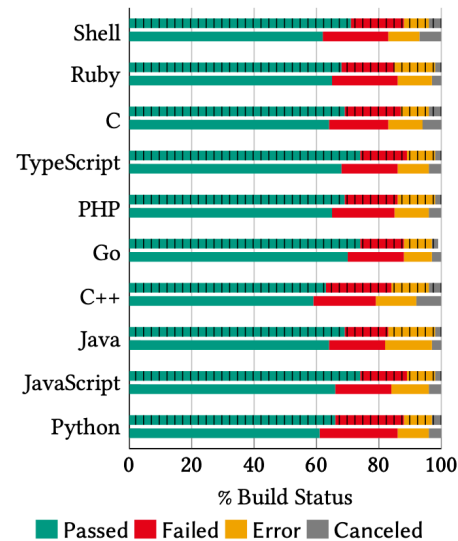


Figure 6: Comparison of the distribution of build status between the projects with restarted builds (plain box) vs. the other projects (boxes with vertical lines).

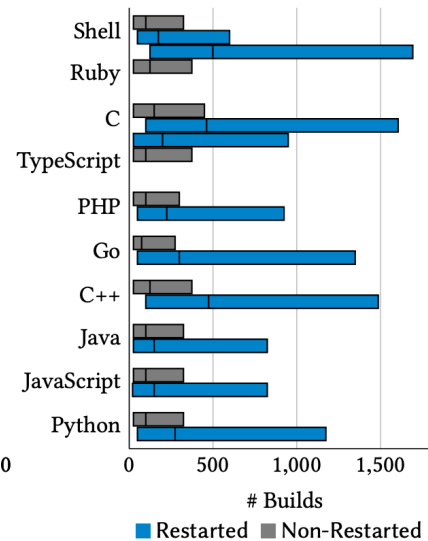


Figure 7: Box plot of the number of builds for the projects that have restarted builds (blue box) vs. the projects that do not have (gray box). It shows that the projects with restated builds have more builds that the other projects.

- ビルドの成功率やビルド数に関しては言語間に大きな違いは見られなかった.

結果

RQ2： restarted buildが行われるプロジェクトに特性はあるか.

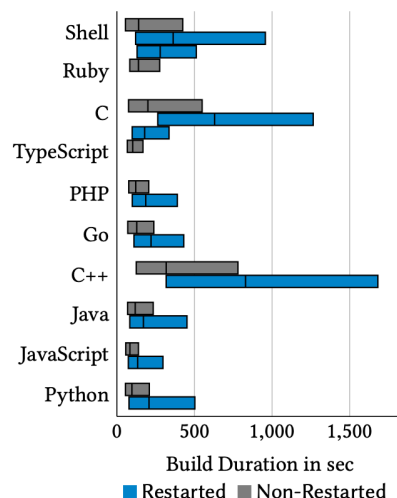


Figure 8: Box plots of the execution time of projects that have restarted builds (blue box) vs. the projects that do not have (gray box). It shows that the execution time of projects that have restarted builds are much higher.

Table 4: Number of stars of 21,775 projects that have restarted builds compared to the 153,666 that do not have a restarted build. The number of projects differs from Table 1 and Table 2 because we lack data on the repositories that are no longer accessible since initial data collection.

Metric	Restarted	Non-Restarted
# Stars	8,256,003	20,837,112
Average	379	136
Median	3	1

- ビルドの実行時間に関して2つの言語は他の言語と比べて非常に高速であった.
- これは2つの言語が他の言語よりも単純で予期しない動作を起こしにくいということを示している.
- restarted buildを含むプロジェクトは人気であることが分かった.

結果

RQ3：開発者がrestarted buildを行うのはなぜか.

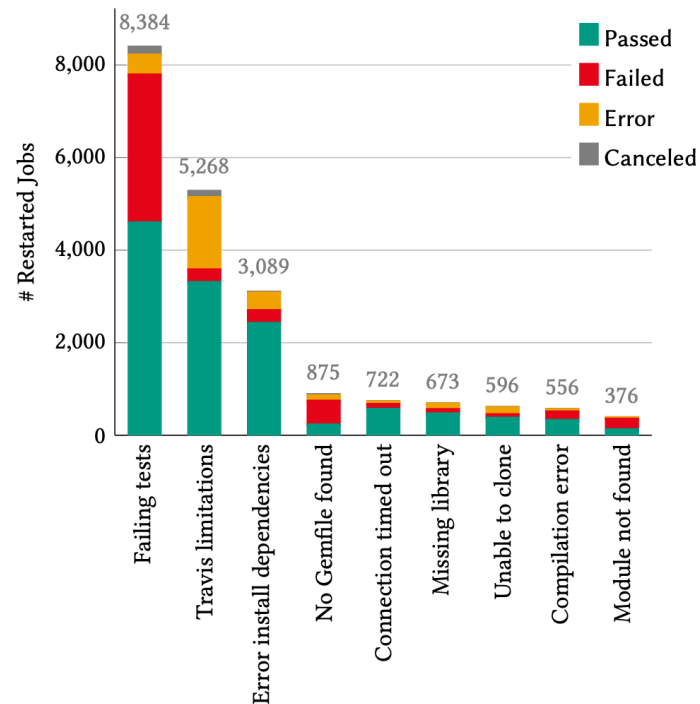


Figure 9: Reasons of failures of the restarted builds. The state of the restarted build is represented by the colors, green for passing, red for failure, orange for error, and gray for canceled.

- 正規表現を使用してログからビルドが失敗した原因を抽出.
- 19,730のビルド失敗の原因が抽出できた.
- 最も多いビルド失敗の理由はテストの失敗であった.
- 続いてTravisCIに関連する障害が多かった.

結果

RQ4：不安定なビルドが開発ワークフローにどんな影響を与えるか.

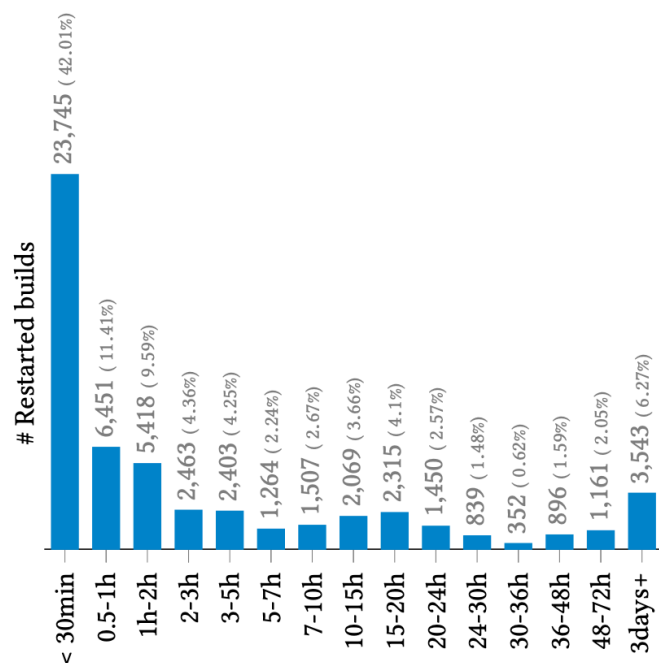


Figure 10: Amount of time between the original builds and the restarted builds.

- 図はビルドが失敗してから restarted buildされるまでの時間を表している.
- 42%が30分以内にrestartされ
63%が2時間以内にrestartされている.
- 開発者はrestarted buildのために他の作業が中断し, タスクを切り替える必要がある.

結論

- ・ プログラミング言語によってrestarted buildの頻度が異なっていた.
- ・ ビルド失敗の主な原因はテストの失敗やTravisCIの障害に関するもので、失敗の理由によって再開後のビルドの成功率も大きく異なっていた.

所感

- restarted build自体がたまにしか行われないこと(1.72%)だったのでそこまで調査に意味があるのか分からない.
- ビルドが失敗/エラーだった場合に, 手動でビルドすることで46.77%が成功するという事に驚いた.