

An Empirical Study on the Survival Rate of GitHub Projects

Adem Ait, Javier Luis Cánovas Izquierdo, Jordi Cabot

19th International Conference on Mining Software Repositories (MSR 2022)

発表者 田中智達

背景

- ・ OSS（オープンソースソフトウェア）
 - ・ 現代のソフトウェア産業において重要な役割
 - ・ 誰もが貢献できる共同開発モデル
 - ・ 開発者がプロジェクトを中止すると持続可能性に深刻な問題が生じる^[7,18]
- ・ OSSプロジェクトの選定
 - ・ 2億ものリポジトリからアクティブなプロジェクトを探索する必要がある
 - ・ 分析データと処理メカニズムが必要だが、提供されているフィルタリングやクエリ条件では不十分

目的・アプローチ

- 目的
 - OSSプロジェクト生存率の向上と正の相関を持つ要因を特定
- アプローチ
 - 初期開発動向と生存率を分析
 - 時系列分析を行い、時間に依存するデータパターンを探索
 - 生存時間分析を行い、与えられた期間内での生存率を探索

Research Questions

RQ1：プロジェクトの活動は時間と共にどのように変化するのか

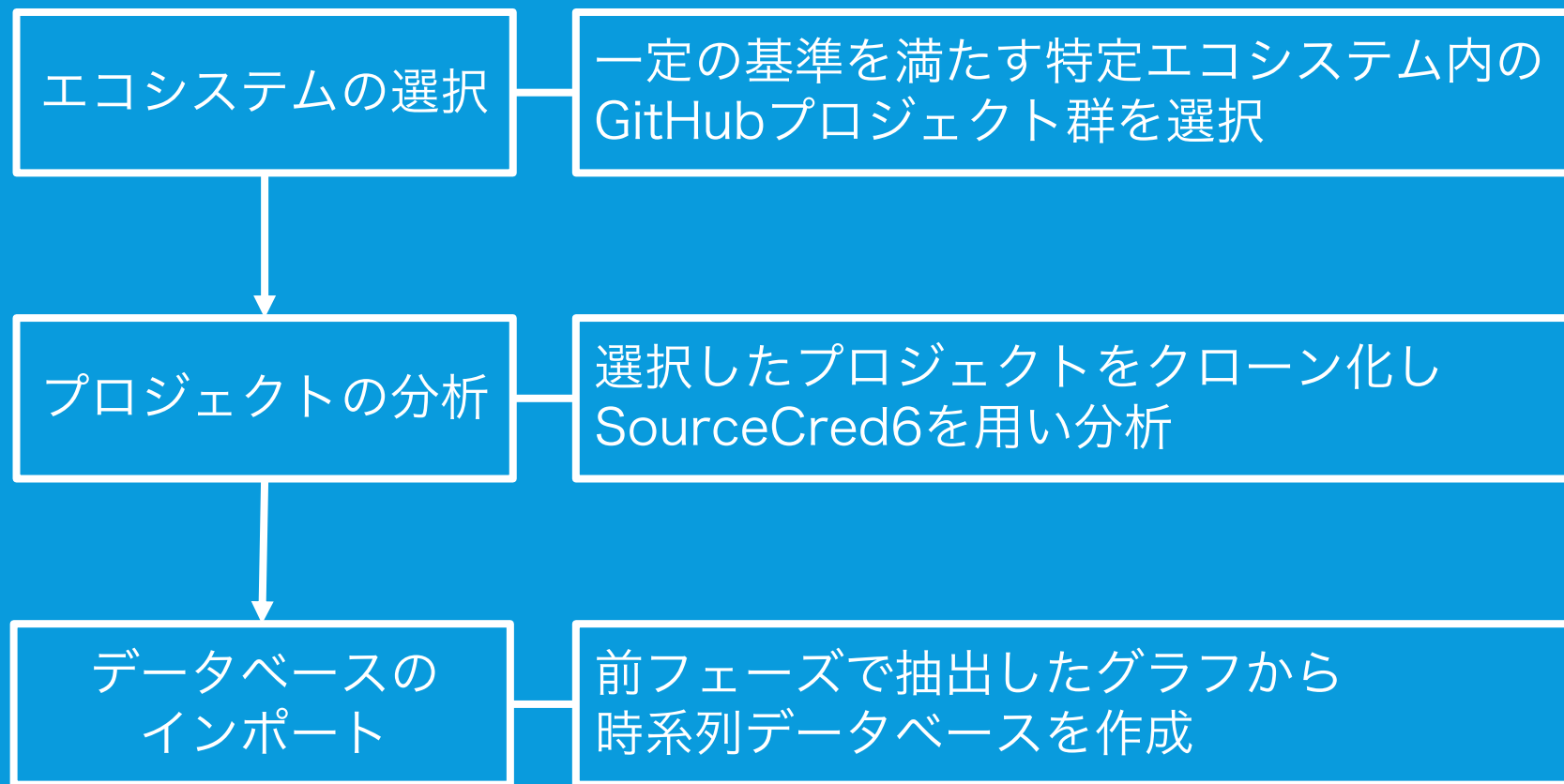
- ・プロジェクトがどのフェーズにあるかを分析
- ・プロジェクトの変遷や共通の進化パターンの理解

RQ2：生存率はどのくらいか

- ・放棄されたプロジェクトを特定し計算
- ・プロジェクトの生存可能性の理解

データセット

- ・以下の3段階を踏んで作成



詳細：エコシステムの選択

- 一定基準を満たす特定のエコシステムを選択
 1. 2016年に作成されたか
 2. 2016年に最低でも一度は更新されているか
 3. 関心のあるエコシステムか
- 上記を満たす4つのエコシステムを選択
 - NPMパッケージ
 - Rパッケージ
 - WordPressプラグイン
 - Laravelパッケージ

詳細：エコシステムの選択

- ・ 特定のエコシステムに焦点を当てるメリット
 - ・ 様々なソフトウェアドメインや技術を多様に表現が可能
 - ・ 各エコシステムに存在する可能性がある特定の課題とパターンについての洞察を得られる
 - ・ データセットのサイズも管理しやすくなり、 妥当な期間内に多数のプロジェクトを収集して分析することが可能
 - ・ 異なるドメインにわたる生存率と開発パターンを比較対照できるため、 プロジェクトの成功または失敗に寄与する要因をより広く理解可能

詳細：エコシステムの選択

- ・ GitHub Search APIへのクエリにより収集
 - ・ 合計1,127件のプロジェクトを獲得

Topic	In 2016	全期間
NPMパッケージ	280	4,691
Rパッケージ	199	1,993
WordPressプラグイン	540	7,223
Laravelパッケージ	108	1,424
合計	1,127	15,331

表1：エコシステムごとのGitHubリポジトリ数と作成期間

詳細：プロジェクトの分析

- ・ 選択したプロジェクトをクローン化し分析
 - ・ SourceCred6を用いるとCollaboration graphが構築
 - ・ ノードはリポジトリの資産
 - ・ ユーザー，コミット，コメント，Issue，PRなど
 - ・ エッジはその資産間の関係性
 - ・ ユーザーがコミットを作成する，コメントが課題に属するなど
 - ・ 時間的な情報も含有
 - ・ コミットやIssueの作成のタイムスタンプなど

詳細：プロジェクトの分析

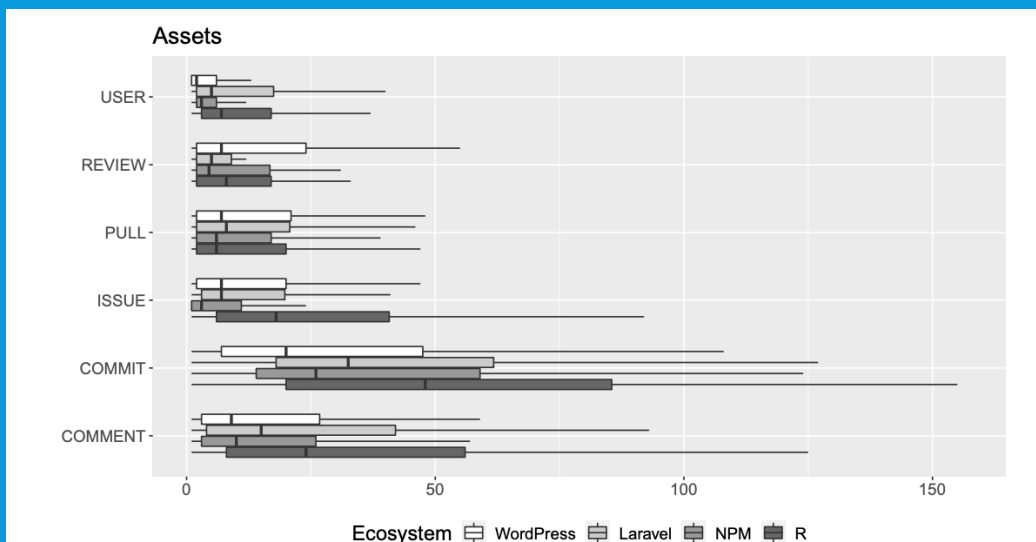


図2：エコシステム毎の資産

表2：資産の総数

TOPIC	COMMITTS	PRs	ISSUES	COMMENTS	REVS.	USERS
npm-package	41,767	8,858	3,947	22,961	4,549	4,319
r-package	48,965	2,202	7,032	23,366	1,260	2,689
wordpress-plugin	134,426	16,152	27,348	103,999	16,878	17,163
laravel-package	11,070	1,841	1,561	5,310	228	2,005

RQ1 プロジェクトの活動は時間と共にどのように変化するのか

- プロジェクトを3つの状態に分類
 - Alive状態：リポジトリにアクティビティがある
 - Running状態：コーディング活動あり
 - Zombie状態：コーディング活動なし
 - Dead状態：リポジトリにアクティビティがない

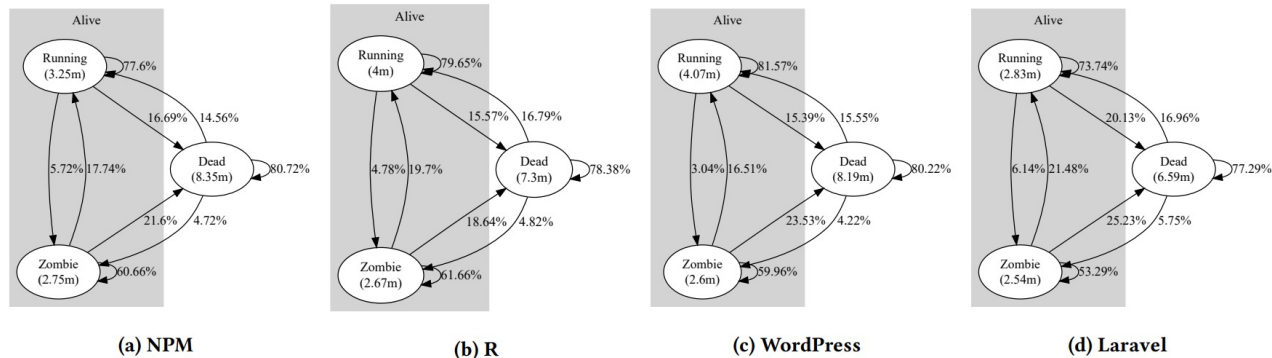


Figure 4: State machines representing the evolution paths in each ecosystem. State labels indicate the name of the status and the average time (in months). Transition labels indicate the probability to change status each month.

RQ1 プロジェクトの活動は時間と共にどのように変化するのか

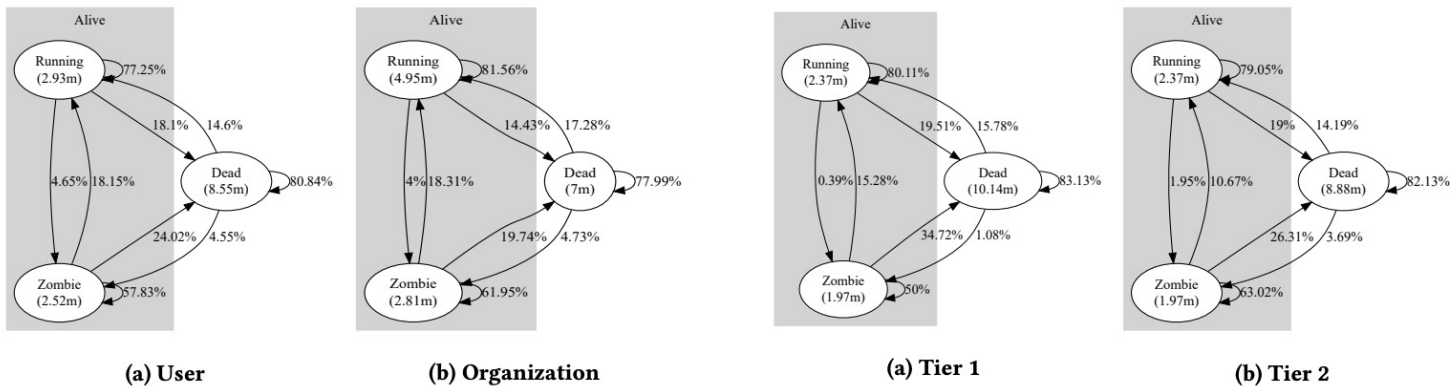


Figure 5: State machines representing the evolution paths grouped by project type. State labels indicate the name of the status and the average time (in months). Transition labels indicate the probability to change status each month.

資料

図5：プロジェクトの種類別での進化経路を表すステートマシン

図6：コミュニティの規模別での進化経路を表すステートマシン

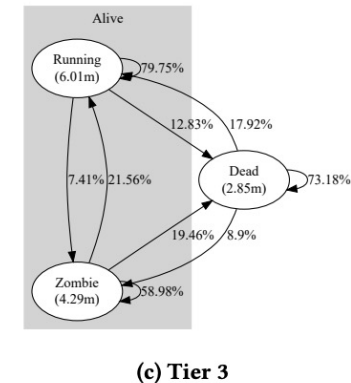
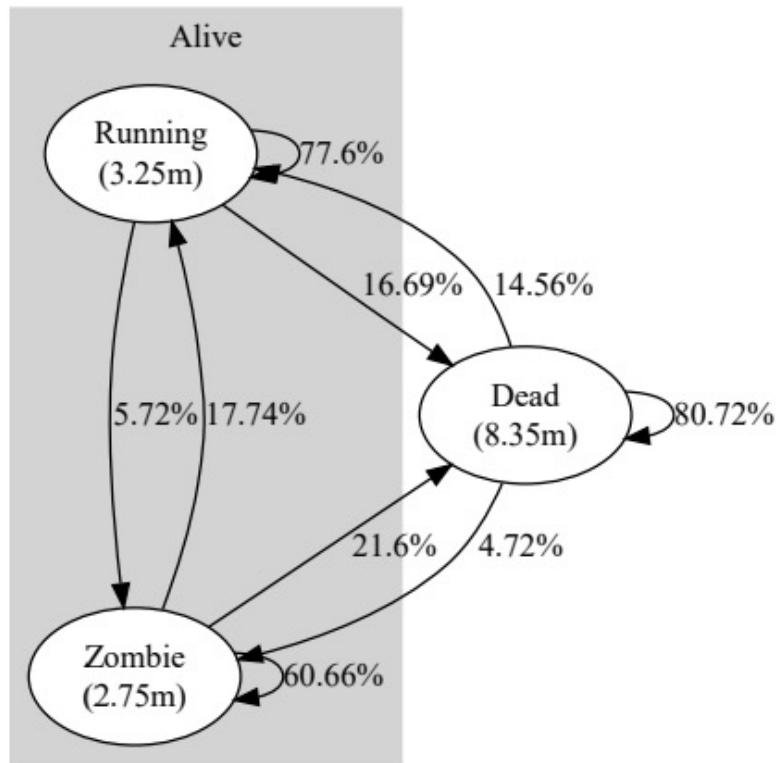


Figure 6: State machines representing the evolution paths grouped by community size. State labels indicate the name of the status and the average time (in months). Transition labels indicate the probability to change status each month.

RQ1 プロジェクトの活動は時間と共にどのように変化するのか



どの状態か
(上記状態の月単位での平均期間)

月毎に状態が変化する確率

RQ1 プロジェクトの活動は時間と共にどのように変化するのか

- ・ 存続している間、断続的活動を行う
 - ・ Dead状態の平均期間は、Running状態の平均期間の約2倍ある
 - 短い集中期間と長い非活動期間で発展する
- ・ プロジェクトの種類（ユーザーor組織）
 - ・ 平均状態期間、状態遷移はわずかな変化
 - ・ Running状態の平均期間のみ、組織の方が長い
- ・ コミュニティの規模
 - ・ 状態遷移はわずかな変化
 - ・ 規模と共にRunning・Zombie状態の平均期間が増加し、Dead状態の平均期間が減少

RQ2

生存率はどのくらいか

- ・データセットで考慮した期間の終了時の状態を調査
 - ・全エコシステムにおいて、半数以上が本研究の考慮される期間内にDead状態へ

Topic	Projects	Dead	Alive
NPMパッケージ	280	189	91
Rパッケージ	199	98	101
WordPressプラグイン	540	360	180
Laravelパッケージ	108	58	50

表5：Dead, Aliveそれぞれの状態のプロジェクト分布（件数）

RQ2

生存率はどのくらいか

- ・データセットで考慮した期間の終了時の状態を調査
 - ・1年目に多くがDead状態になる
 - ・分析対象の半数以上が，最初の4年間でDead状態

Topic	1	2	3	4	5	6
NPMパッケージ	56	50	36	28	18	1
Rパッケージ	8	13	13	27	34	3
WordPressプラグイン	144	69	60	48	33	6
Laravelパッケージ	21	11	9	13	4	0

表6：消滅したプロジェクトの分析（年，件数）

RQ2 生存率はどのくらいか

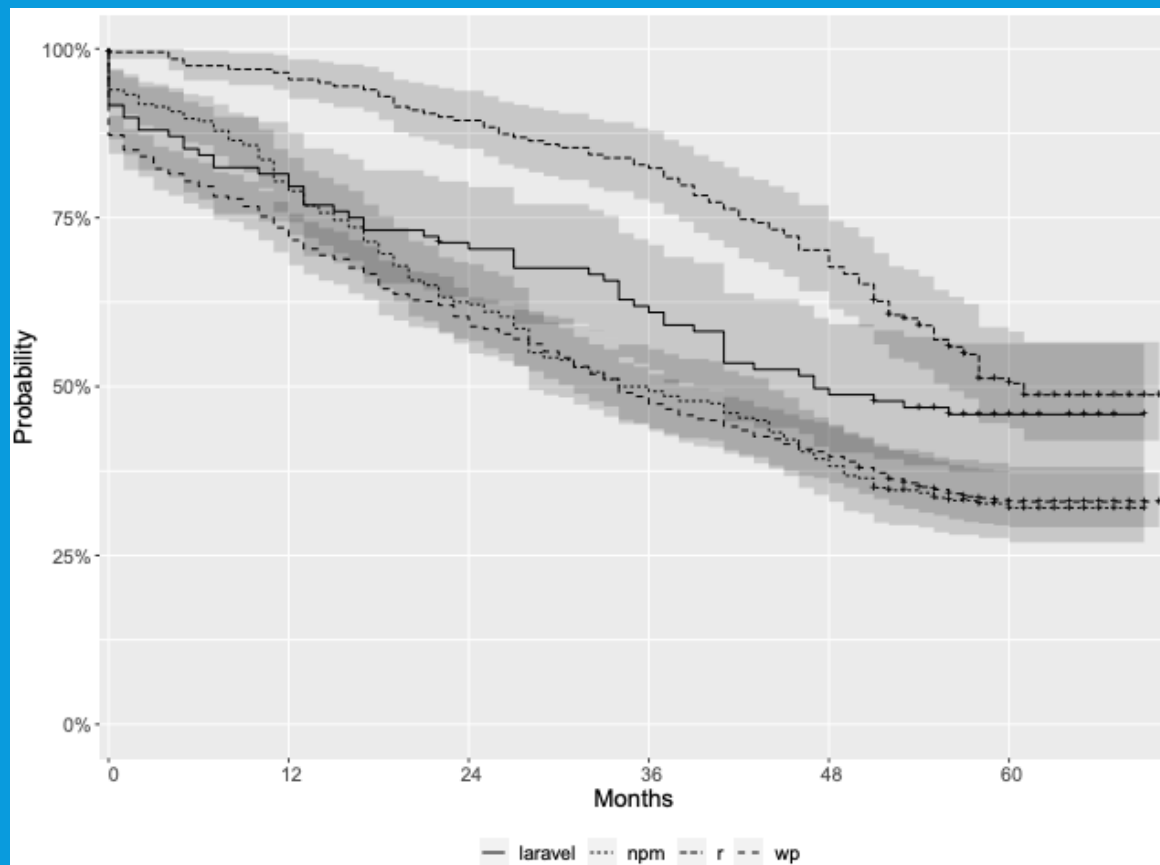
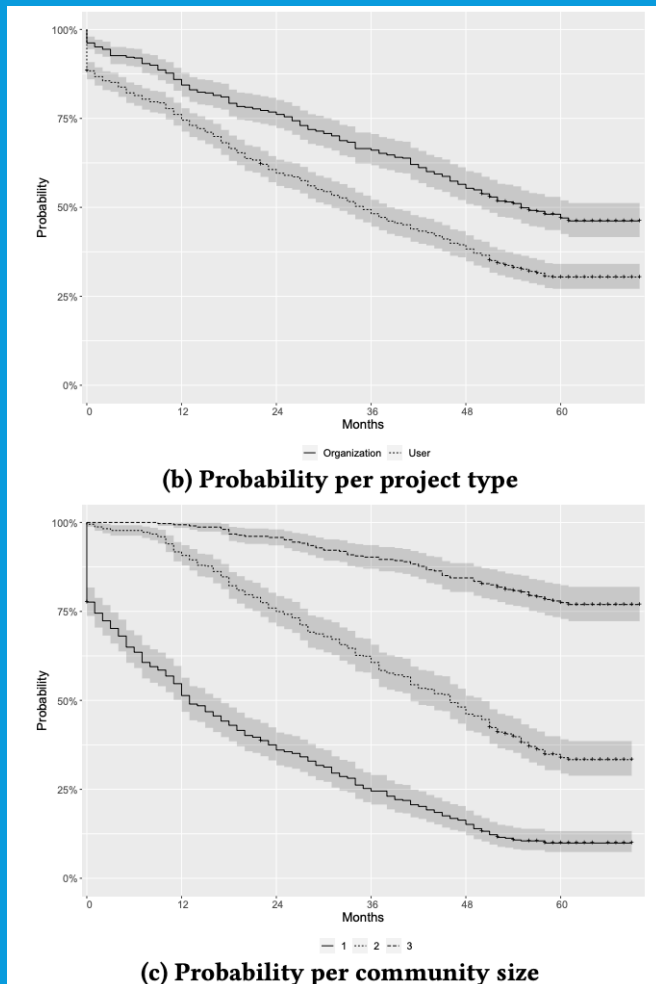


図7-a：プロジェクトごとの生存率

RQ2 生存率はどのくらいか



- ・プロジェクトの種類, コミュニティの規模別調査
- ・組織プロジェクトの方が生存率が高い
- ・コミュニティの規模が大きいと生存率が高い

上図7-b：プロジェクトの種類別生存率
下図7-c：コミュニティの規模別生存率

まとめ

- 分析結果
 - 典型的な開発プロセスは、集中期間と長期の非活動期間から構成
 - 多くのプロジェクトが最初の1年で活動が停止し、半数以上が4年以上存続できない
- オープンソースプロジェクトを扱う上で伴うリスクの認識を高めるために有用な研究