

# 実プロジェクトデータからのソフトウェアタグ作成とその分析

## Making a software tag from a real project data and its analysis

山田 慎也 †      西川 倫道 †      鵜久森 将隆 †      楠本 真二 †  
Shinya Yamada   Tomomichi Nishikawa   Masataka Ugumori   Shinji Kusumoto

### 1 まえがき

ソフトウェアの大規模化、複雑化に伴いソフトウェアの社会における重要性は年々大きくなっている。しかし大規模化、複雑化に伴う開発の困難さから、ソフトウェアが引き起こす問題も深刻化してきている。

StagE(Software Traceability and Accountability of Global Software Engineering) プロジェクト [5] ではソフトウェア開発における実証的なデータを収集し、ソフトウェアタグとしてソフトウェア製品に添付する技術の開発を行っている。ソフトウェアタグの情報により、ソフトウェア発注者は自身が発注したソフトウェアがどのような体制で、どのような開発手法を用いて開発されたものか、という開発情報を入手できる。このようにして、ソフトウェア開発過程に透明性を持たせることで社会に流通するソフトウェア品質の向上を試みている。

本研究では、ソフトウェアタグの有用性を評価することを目的として、IT Spiral[2] で提供されている実プロジェクトデータに対してソフトウェアタグの作成を試みた。更に、ソフトウェアデータ白書 [1] に記された標準的ソフトウェア開発データとの比較やソフトウェアタグデータの推移情報の分析を行った。結果として、ソフトウェアタグ情報を用いることで、実プロジェクトデータの開発プロセスの特徴を把握することができた。

### 2 ソフトウェアタグ

ソフトウェアタグとは、ソフトウェア開発におけるトレーサビリティ情報をまとめたものである。ソフトウェアタグはソフトウェアの開発途中もしくは開発後に作られるものであり、12 項目のプロジェクト情報タグと 29 項目の進捗情報タグによって構成される [4]。プロジェクト情報タグは対象となるプロジェクトに関する情報、プロジェクト中に不変なデータ、プロジェクトを特徴付ける情報を持ち、進捗情報タグは成果物や作業の進捗状況、成果物やプロセスの品質を表す情報を持つ。この情報を分析することで開発プロジェクトの様々な特徴が分かる。

本研究における実プロジェクトデータからのソフトウェアタグ作成では、プロジェクト情報 6 項目 (表 1) と、進捗情報 20 項目 (表 2) を対象とした。

ソフトウェアタグのタグデータ収集は、今回収集を行った 22 項目のうち 15 項目についてツール [3] による自動収集を行った。残りの 7 項目については手作業での収集を行った。

### 3 実プロジェクトデータ

実プロジェクトデータは、大阪大学大学院情報科学研究科を中心に実施された教育プロジェクトである IT Spiral で提供されているソフトウェア開発データである。これは、和歌山大学の学部教務システムの開発で作成されたプロジェクトデータとドキュメント類を含む。実プロジェクトデータは設計、実装、テスト等の工程において開発支援ツールを使用しており、様々な実証データが存在する。開発期間は平成 19 年 4 月から 5 月、および平成 19 年 12 月から平成 20 年の 2 月までの合計 5 か月であり、総データ量は 450MB ほどである。

表 1 収集を行ったプロジェクト情報タグデータ一覧

| 分類     | タグ項目       |
|--------|------------|
| 基本情報   | プロジェクト名    |
|        | 開発組織の情報    |
|        | 開発プロジェクト情報 |
|        | 顧客情報       |
| システム情報 | システム構成     |
| 開発情報   | プロジェクト期間   |

表 2 収集を行った進捗情報タグデータ一覧

| 分類      | タグ項目      | 収集対象データ       |
|---------|-----------|---------------|
| 要件定義    | 規模推移      | ユースケース図数      |
|         | 変更推移      | ユースケース図数      |
| 設計      | 規模推移      | UML 図数        |
|         | 変更推移      | UML 図数        |
| プログラミング | 規模推移      | 行数            |
|         | 変更推移      | 変更量           |
|         | 複雑度       | CK メトリクス      |
| テスト     | 規模推移      | テストケース数       |
|         | 変更推移      | テストケース数       |
|         | 密度        | テストケース数/行数    |
|         | 消化        | 消化テスト数        |
| 品質      | レビュー状況    | レビュー回数        |
|         | レビュー作業密度  | レビュー時間/時間     |
|         | レビュー指摘数推移 | レビュー指摘率       |
|         | 欠陥件数推移    | 欠陥件数          |
|         | 欠陥対応件数    | 欠陥対応件数        |
|         | 欠陥密度      | 欠陥件数/行数       |
|         | 欠陥指摘率     | 欠陥件数/消化テスト数   |
|         | 静的チェックの結果 | FindBugs 指摘件数 |
| 工数      | 作業工数      | 作業時間          |
|         | 生産性       | 行数/作業時間       |

† 大阪大学 大学院情報科学研究科, Graduate School of Information Science and Technology, Osaka University

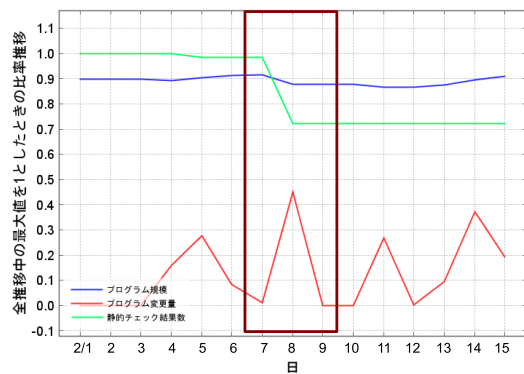


図1 平成20年2月の各メトリクス比率推移

#### 4 ソフトウェアタグを用いた分析

実プロジェクトデータから作成したソフトウェアタグを用いて、ソフトウェアの品質・プロジェクトの開発状況を把握できるかどうかを確認する。まず、タグ項目の最終実績値について評価する。本来は契約時にユーザ・ベンダ間で必要なタグ項目について決めた予想値と最終実績値を比較すべきである。しかし、対象プロジェクトではそれらの値が無い場合、プロジェクトの特性値(規模、品質など)を標準値と比較する。次に、タグデータの推移を見ることで、不自然な部分を抽出して内容を確認する。

##### 4.1 標準的ソフトウェア開発データとの比較

作成したソフトウェアタグの値とソフトウェアデータ白書2008[1]との比較を行った結果を表3に示す。対象としたデータはソフトウェアタグの作成元となった実プロジェクトデータと同じく新規開発を行った公務系ソフトウェア46プロジェクトのデータである。

この結果から一般的なプロジェクトに比べ今回の実プロジェクトはテスト密度が低いことが分かる。バグ密度も低い値が出ているが、より一層のテストを実施することでバグ密度は上昇した可能性がある。

##### 4.2 タグデータの推移に関する分析

作成したソフトウェアタグの値を分析することで、実プロジェクトデータ開発過程における特徴的な操作がいくつか明らかになった。

一例として、プログラム規模、プログラム変更量、静的チェックの警告数から明らかになったことを示す。図1に、平成20年2月1日から2月15日までのプログラムの規模、変更量、および静的チェックの警告数の推移を示す。実プロジェクトデータは開発期間が5か月であり、平成20年2月はテスト工程に入っている。ゆえにコーディングはほとんど行われないはずだが、2月8日はプログラムの変更量が大きくプログラム規模も減少がみられる。静的チェックの警告数も他の日ではあまり変化がみられないが、ここでは大きく変化している。そこで、この部分の詳細を分析した。具体的には、このプログラムはSubversionを用いてバージョン管理を行っているため、この日にどのような変更が行われたのか調べるためにコミット時のコメントを見た。すると不要クラスの削除を行ったという記述があった。したがって不要クラスの削除によりソースコード規模および静的チェッ

表3 ソフトウェア開発データ白書の値との比較

| 項目名             | タグの値   | データ白書記載の中央値 |
|-----------------|--------|-------------|
| プログラム規模 (LOC)   | 26,033 | 64,000      |
| バグ密度 (件/KLOC)   | 0.73   | 1.63        |
| テストの密度 (件/KLOC) | 23.7   | 36.1        |

クの警告数が大きく減少したと考えられる。

このように、開発途中の特徴的な操作はソフトウェアタグを分析することで明らかになる。これは望ましくない操作についても同様である。例を挙げると、開発後期に仕様変更が生じたことによってソースコードの大幅な修正が行われているという場合にもソフトウェアタグからの特定が可能である。

#### 5 まとめ

本研究では、実プロジェクトデータからのソフトウェアタグの作成を行い、タグデータに基づく開発プロジェクトの分析を行った。まず標準的ソフトウェア開発データと今回収集したタグデータとの比較を行い、対象ソフトウェアの品質について考察した。次に作成したソフトウェアタグの推移情報を用いて実プロジェクトデータの開発状況を分析した。その結果、開発途中の特徴的な操作を特定できることを示した。

今後の課題としてソフトウェアタグ作成を支援するシステムの実装が挙げられる。現時点におけるソフトウェアタグの作成は、いくつかの部分においてツールによる支援があるものの手作業での値の収集を行わなければならない項目も存在するため作成のコストは高い。ゆえにソフトウェアタグ作成コストを小さくする支援システムを開発し、他プロジェクトに適用を行い実用性を示していく。

#### 謝辞

本研究は文部科学省「次世代IT基盤構築のための研究開発」(研究開発領域名:ソフトウェア構築状況の可視化記述の開発と普及)の委託に基づいて行われた。また、一部日本学術振興会科学研究費補助金基盤研究(C)(課題番号: 20500033)の助成を受けている。

#### 参考文献

- [1] IPA/SEC:ソフトウェア開発データ白書2008, 日経BP社, 2008.
- [2] IT Spiral, <http://it-spiral.ist.osaka-u.ac.jp/>
- [3] 西川, 鶴久森, 山田, 楠本: “ソフトウェアタグデータ収集システムの試作”, ソフトウェア信頼性研究会第5回ワークショップ論文集, pp.57-65, 2009.
- [4] StagE プロジェクト:ソフトウェアタグ規格第1.0版, 2008. <http://www.stage-project.jp/kanri/data/dl/20081029151602.pdf>
- [5] StagE プロジェクト, <http://www.stage-project.jp/>