

---

# ソフトウェア開発 PBL における タスク記録の修正に基づく振り返り手法の提案

Task Record Correction Based Retrospective on Software  
Development PBL

福安 直樹\* 井垣 宏† 佐伯 幸郎‡ 眞鍋 雄貴§ 楠本 真二¶  
井上 克郎||

あらまし 受講生の主体的な活動が重視される PBL 形式の演習では、プロジェクトの振り返りが重要である。振り返りはプロジェクトで行われたタスクの正確な記録に基づいて実施される必要があるとされているが、タスクを記録するシステムの操作や記録されるデータの意味、および正確に記録することの重要性を受講生が十分に理解できていない場合、正確な記録に基づく振り返りは困難となる。本研究では、正確なタスク記録に基づく振り返りに対する受講生の理解を促すために、タスク記録の修正作業を振り返りの一環として実施する手法を提案し、演習で得られたタスク記録の修正実験を通じて本手法の有効性を確認する。

## 1 はじめに

技術的能力だけでなく、マネジメントやコミュニケーションなどの総合的な能力を備えた高度 IT 人材の育成方法として PBL (Project Based Learning) 形式によるソフトウェア開発の演習 (ソフトウェア開発 PBL, 以降 SDPBL) が注目されている [1]。PBL では、受講生による主体的な活動が重視され [2]、プロジェクトの振り返り [3] を中心とした活動により演習が進められる。

プロジェクトの振り返りは、正確な記録に基づいて行われることが重要である [4]。しかしながら、記録は受講生による入力に基づいているため、タスク管理システムの操作に十分に慣れていないことや、記録されるべき要素の意味についての認識が不十分であること、それらを正確に入力することの重要性が十分に理解できていないことなどの理由から、入力誤りなどが数多く発生し、結果として正確な記録に基づく振り返りが困難となっている。

そこで本研究では、タスク管理システムに記録されたデータの修正作業を振り返りとして実施する手法を提案する。これにより、タスク管理システムやそこに記録される要素に対する理解を深め、記録の重要性とその効果にできるだけ早期に気付かせることを目指す。

## 2 ソフトウェア開発 PBL における振り返り

PBL 形式による教育は、受講生が主体的にプロジェクトの運用方法を工夫・改善しながら試行錯誤することにより、通常の講義・演習で学んだプロジェクトマネジメントに関する知識を応用可能な技術として身につけることを目的として実施される [5]。SDPBL における試行錯誤のプロセスは、振り返りを中心としたアクティビティにより駆動され、開発と振り返りを繰り返しながら進められる。振り返りとは、プロジェクト中あるいは終了後にメンバー自身により行われるプロジェクト改善の

---

\*Naoki Fukuyasu, 和歌山大学

†Hiroshi Igaki, 大阪大学大学院

‡Sachio Saiki, 高知工科大学

§Yuki Manabe, 大阪大学大学院

¶Shinji Kusumoto, 大阪大学大学院

||Katsuro Inoue, 大阪大学大学院

表 1 タスク情報

属性	説明
a. 着手時刻	タスクに着手した日時
b. 終了時刻	タスクを終了した日時
c. 担当者	タスクの担当者名
d. 見積時間	タスク実施に必要と見積もられた時間
e. 総時間	タスク完了にかかった時間
f. 成果物	タスクで作成・編集される対象物
g. 種類	タスクの種類（実装，レビュー，試験など）
h. マイルストーン	このタスクを含む，納期の設定された開発単位

ための自発的な活動で、それまでの進め方について、良い点・問題点を自ら認識し、問題点に対する解決策をメンバー全員によるミーティングの形式で検討する [3]。

### 2.1 タスク情報の記録

振り返りでは、問題点に対する解決策の提案だけでなく、実施された解決策の評価も行われる。そのための具体的な指標や根拠は、メンバーの記憶だけに頼るのではなく、プロジェクトにおける活動の正確な記録に基づくことが重要である [4]。

本稿で対象とする SDPBL では、活動内容を記録するタスク管理システムおよびプロダクトの開発履歴を管理する版管理システムに記録されたデータを利用することにより振り返りを実施することを想定している。プロジェクトは納期の設定されたいくつかの開発単位（マイルストーン）に分割され、さらにマイルストーンの達成に必要なタスクがタスク管理システムに登録される（タスク記録）。ここでそれぞれのタスクは、1人の担当者が1つの成果物に対して1度に行う作業（実装，レビュー，試験など）を表す。それぞれのタスクに記録される属性（タスク情報）は、表1に示す通りである。一方、版管理システムに対しては、check-out, commit, update, diff, log といった操作がチームのメンバーにより行われる。

### 2.2 振り返りにおける課題

より効果的な振り返りを実現するためには、活動の正確な記録に基づいて実施することが重要である。しかしながら、我々がこれまでに実施した SDPBL での経験において、タスク管理システム上で正確な記録が十分に残されていない事例が少なからず見られている。これは、

- 多くの受講生にとって初めて本格的に利用するタスク管理システムの使い方や版管理システムとの連携に対する理解が十分でないこと（How）
- タスク管理システムに記録されるべき各タスク情報そのものに対する理解が十分でないこと（What）
- タスクの内容を記録するという行為が、開発プロセスにおけるシステムの実装という目的のよりはっきりとした活動に比べて軽視されやすく、その目的や効果に対する理解が十分でないこと（Why）

の3点が主な原因として考えられる。これらの知識は、演習前の講義においても説明を行っているが、実際に演習を実施した際の我々の観測では必ずしも正確に認識できているとはいえず、通常の講義だけでは習得の難しい知識の1つであると考えられる。したがって、これらに対する理解を PBL の中で実体験を通じて早期に促すことが課題である。

## 3 タスク記録の修正とそれに基づく振り返り

2.2 節で述べた課題に対して、本研究では、受講生自身によるタスク記録の修正に基づく振り返り手法を提案する。本手法を SDPBL における振り返りの一環とし

表 2 タスク記録誤りの分類

カテゴリ	分類
1. タスク情報誤り	a. 着手時刻誤り b. 終了時刻誤り c. 担当者誤り d. 見積時間誤り e. 総時間誤り f. 成果物誤り g. 種類誤り h. マイルストーン誤り
2. タスク粒度誤り	a. タスクの粒度が大きすぎる b. タスクの粒度が小さすぎる
3. タスク記録漏れ	a. 実施されたにもかかわらず記録されていないタスク b. 実施されていないにもかかわらず記録されているタスク

て実施することにより、タスク管理システム / 版管理システムやタスク情報そのものについての受講生らの理解促進を目指す。さらに、タスク記録の修正作業によりプロジェクトの活動状況に関する正確なデータの重要性を認識させ、タスクの内容をより正確に記録するモチベーションを受講生に与えることを目標とする。

### 3.1 タスク記録の修正

タスクは、チームのメンバーがタスク管理システムに対して行う作成・着手などの操作によって管理され、タスク記録にはタスク情報の操作ログが記録されている。本稿で想定するタスクの基本的な実行プロセスは次の通りである。まず、成果物や種類、マイルストーンを設定したタスクをタスク管理システム上でタスク記録として作成し、見積時間の入力や担当者割当を行う。タスクを割り当てられた担当者は、タスクに着手した後に check-out や update を実施して手元のソースコードを最新の状態にする。実装のタスクでは成果物の作成・編集を行って commit を実行する。レビューのタスクでは、diff による差分の確認や、log によるコメントの確認に基づいて実施し、報告書の作成等を行う。最後にタスクにかかった総時間を入力して終了する。

タスク管理システムにタスク記録を登録する際に発生する可能性のある誤りを表 2 に示す。タスク情報誤りは、表 1 に示したタスクの各属性値に関する誤りである。タスク粒度誤りは、記録されたタスクの粒度が、1 人の担当者が 1 つの成果物に対して 1 度に行う作業という基準を満たさない場合である。タスク記録漏れは、タスク管理システムに登録せずに実施された作業や、本来タスクではないものを登録した場合に相当する。

このタスク記録誤りの分類に基づいて、タスク記録の修正を行う手順を以下に示す。この修正は、自身の記憶やタスク管理システムの操作ログ、commit 時のコメントの他、版管理システムに対して行われたチームのメンバーによるすべての操作ログ (check-out, commit, update, diff, log の各操作がタイムスタンプや担当者とともに記録されたもの) を利用しながら行われるものとする。

Step1: タイムスタンプや担当者、成果物名を参考に、版管理システムの操作ログを各タスク記録に対応付ける。

Step2: タスク記録漏れの修正を行う。

(2-1) 版管理システムの操作ログに対応するタスク記録が存在しない場合 (3-a), 新たなタスク記録を追加する。

(2-2) タスク記録に対応する版管理システムの操作ログが存在しない場合で、実際に作業として実行されていないと判断できる場合 (3-b) はそのタスク記録を削除する。

Step3: 1 人の担当者が 1 つの成果物に対して 1 度に行う作業という基準に基づい

てタスク粒度誤りを修正する。

(3-1) 複数の作業が行われていると判断できるもの(2-a)について、タスク(と対応する版管理システムの操作ログ)を分割する。判断のための指標として、複数の commit が行われている場合や、途中で担当者が変更されている場合、一旦終了したタスクが再開されている場合などが挙げられる。

(3-2) 同じ担当者が設定されている複数のタスク記録において、まとめて1度の作業と判断できる場合(2-b)、それらのタスク記録を1つに結合する。

Step4: 分割後のすべてのタスク記録(追加されたタスクや分割されていないタスクも含む)について、タスク情報誤りの修正を必要に応じて実施する。

(4-1) 着手時刻(1-a)を修正する。そのタスクのために update(または check-out)されたと思われる時刻と着手時刻に乖離がある場合などが判断基準となる。

(4-2) 終了時刻(1-b)を修正する。そのタスクの成果物が commit された時刻と終了時刻に乖離がある場合などが判断基準となる。

(4-3) 担当者(1-c)、属するマイルストーン(1-h)を修正する。

(4-4) 総時間(1-e)を修正する。修正後の着手・終了時刻を参考に、そのタスク以外の作業を行っていた時間を除く実質の作業時間に修正する。

(4-5) 成果物(1-f)、種類(1-g)を修正する。成果物は commit されたプロダクトなどを確認する。

なお、Step4 では、見積時間誤り(1-d)に対する修正は行っていない。見積時間は、タスク実行前にどのように考えていたのかを記録する属性であり、タスク終了後の修正時にこれを補完することに意味はないと考えるからである。

### 3.2 タスク記録の修正に基づく振り返り

タスク記録の修正手順により修正された値は、チームの活動を評価するための判断材料として利用される。着手時刻、終了時刻、担当者は、同時に複数のタスクを実行していないかなどの分担状況の確認などに、見積時間、総時間、マイルストーンは、納期が守れなかった場合の原因の確認などに、成果物、種類は、必要十分なタスクが実行できているかといった品質の確認などに利用される。また、すべてのタスクを記録し、記録されたタスクの粒度をそろえることにより、メンバー間や種類ごとといった比較が可能となる。客観的なデータに基づいてKPTなどの振り返り手法を実施することで、プロジェクトの問題点を認識し運用方法を工夫・改善する。

タスク記録の修正を振り返りの一部として実施することで、次のような効果が期待できる。

**How** の理解 Step1, Step2 の作業を通じて、タスク管理システムと版管理システムとの関係を理解する。また、Step4 の特に(4-1), (4-2) からタスクの実行プロセスを理解する。

**What** の理解 Step4 における各タスク情報の修正作業を通じてそれぞれにどのような値を入力すればいいのかを理解する。また Step3 はタスク管理システムで管理すべきタスクの粒度の理解につながる。

**Why** の理解 修正されたタスク記録を使って客観的なデータに基づく問題点の認識を経験することで、正確に記録することの意味や目的を理解する。

SDPBL の初期にこのような振り返りを経験することで、SDPBL の主目的である、プロジェクトの運用方法について工夫・改善しながら試行錯誤することにできるだけ早期に専念できるようにする。

## 4 タスク記録の修正実験

我々の研究グループでは、主に博士前期課程の1年次に在籍する大学院生を対象とした高度なソフトウェア技術者の育成を目指す教育プログラム [6] において、SDPBL

タスク名: AppDAO の実装	
着手時刻: 2011-11-11T12:53:59	
終了時刻: 2011-11-11T13:08:39	11/Nov/2011:11:53:09 DX update /trunk r72
担当者: DX	11/Nov/2011:13:06:35 DX commit r74
見積時間: 0 時間	11/Nov/2011:13:12:02 DX commit r75
総時間: 0 時間	11/Nov/2011:13:37:36 DZ update /trunk r76
成果物: AppDAO.java	11/Nov/2011:14:09:01 DZ commit r81
種類: 実装	
マイルストーン: MS5	図 2 AppDAO.java に関する Subversion 操作ログ

図 1 誤りを含むタスク記録

「開発実践演習」を開講している [7]。そこで、2011 年度の開発実践演習で得られたタスク記録について、3.1 節に示した手順によるタスク記録の修正実験を行った。

#### 4.1 開発実践演習受講生によるタスク記録の修正

2011 年度の開発実践演習では、34 名の受講生が 6 チーム（1 チームあたり 5~6 名）に分かれ、ウェブアプリケーションの詳細設計に基づいて実装・レビュー、単体試験、結合試験の各工程を行うおよそ 1.5 ヶ月のプロジェクトを遂行した。開発におけるプロダクトは版管理システム Subversion により管理され、開発の各タスクはあらかじめ設定された 8 つのマイルストーンに基づいてタスク管理システム Trac に記録された。

演習終了後に、各チームのメンバー 1 名ずつに自分のチームのタスク記録に対して 3.1 節に示した手順により修正作業を行ってもらった。ただし、見積時間、総時間については修正は行わず、入力値が初期値のまま変更されていないものを誤りとして計測した。Step4 では (4-4) において総時間を修正することになっているが、今回の実験が演習の 1 ヶ月後に行われており、いつ作業をしていたのかを正確に思い出すのが困難であったためである。

実験の結果、Trac に登録された 6 チームの計 1,419 件のタスク記録の中で、1 つでも修正が行われたものは全体の 40.1% にあたる 570 件であった。誤りを含むタスク記録数の修正前タスク記録数に占める割合は最も修正が少ないチームで 29.1%、多いところでは 60.3% にも達しており、6 チームで延べ 827 件の修正が行われた。

図 1 はタスク情報誤りとタスク粒度誤り、タスク記録漏れのすべての誤りを含むタスク記録の例である。なお、担当者名と成果物名は実際のものから変更されている。被験者は、成果物名に基づいて取得した、このタスクと関係のあると思われる Subversion 操作ログ（図 2）より、図 1 が 2 つのタスク記録を 1 枚のチケットとして記録していることと、このタスクに関連するもう 1 つのタスク記録が漏れていること、さらに複数のタスク情報が間違っていることを確認した。被験者による具体的な修正内容は以下の通りである。

C1: 漏れたタスク記録の追加 図 2 に対して、担当者 DZ が行ったタスクの記録が存在しなかったため、11/Nov/2011:14:09:01 の Subversion 操作ログから commit の詳細を精査し、「AppDAO の修正」というタスク記録を追加した。

C2: タスク記録の分割 図 2 の commit r74 と commit r75 は r74 が AppDAO の初期実装に関するもので、r75 は AppDAO のレビューが行われた後の修正に相当するものであることが他のタスク記録により分かった。そのため、図 1 のタスク記録を「AppDAO の実装」と「AppDAO の修正」の 2 つに分割した。結果として、AppDAO.java は一度実装されてから二度修正されたことが確認できた。

C3: タスク情報の修正 C2 でタスク記録が分割されたため、それぞれのタスク情報の修正を行った。「AppDAO の実装」については、着手時刻が実際の作業開始時刻と推定される 11/Nov/2011:11:53:09（update が行われているため）に、終了時刻が commit r74 の時刻に修正された。さらに、分割された「AppDAO

の修正」については、種類が“実装”から“修正”に変更され、着手時刻・終了時刻が修正された。さらにこのタスク記録については、見積時間・総時間が共に0（初期値）となっていたため、誤りであるということを記録した。

6名の被験者による修正作業の結果、特に多かったのは(1-a)着手時刻誤り123件、(1-b)終了時刻誤り56件、(1-d)見積時間誤り120件、(1-e)総時間誤り292件、(2-a)タスクの粒度が大きすぎるケース155件であった。これらの項目は納期や分担といった評価基準に基づいてプロジェクトの状態を判断する上で重要な要素であり、誤った記録は効果的な振り返りに影響を与える可能性がある。

#### 4.2 考察

タスク記録の修正作業を通じてタスク管理システムに対する理解を深めることが可能かどうかについての考察を目的として、実験の被験者6名に対してアンケートを実施した。その結果、修正作業により、ほとんどすべての被験者が“1人の担当者が1つの成果物に対して1度に行う作業”という粒度を深く理解できたと回答した。また、必要なタスクを実施できているかという観点においてタスク記録漏れが重大なミスであり、さらにその修正が困難であることが多くの被験者により認識されていることが確認できた。これらは、タスク粒度の概念やタスク記録の重要性に対する理解という点において、2.2節で述べたWhatおよびWhyの課題の改善に対応していると考えられる。さらに、Howの課題については、時間に関する誤りが版管理システムのログを対応付けることで修正できることに気付くなど、タスク管理システムと版管理システムの連携に対する理解についての一定の改善が見られた。

### 5 おわりに

SDPBLにおける振り返りとして、タスク管理システムに記録されたデータの修正作業を実施する手法を提案した。タスク記録の修正作業を通じて、タスクを記録するシステムの操作や記録されるデータの意味、および正確に記録することの重要性についての理解を促し、SDPBLの主目的である、プロジェクトの運用方法について工夫・改善しながら試行錯誤することによってできるだけ早期に専念できるようにする。

2011年度の開発実践演習におけるタスク記録について、一部の受講生に修正実験を実施してもらった結果、本手法を通じて、タスク管理システムの仕組みや版管理システムとの連携（How）、記録されるデータやタスクの粒度（What）、正確に記録することの意味（Why）を受講生が理解できることを確認できた。今後は、実際のSDPBLにおける振り返りへの導入を行って、本手法の効果を確認する予定である。

謝辞 本研究は、日本学術振興会科学研究費補助金若手研究(B)(課題番号:24700030)の助成を得た。

#### 参考文献

- [1] 独立行政法人 情報処理推進機構. IT人材白書 2011, 2011.
- [2] 松澤芳昭, 大岩元. 産学協同の Project-based Learning によるソフトウェア技術者教育の試みと成果. 情報処理学会論文誌, Vol. 48, No. 8, pp. 2767–2780, 2007.
- [3] 天野勝. プロジェクトファシリテーション実践編振り返りガイド第20版, 2011.
- [4] A. Cockburn. *Agile Software Development*. Addison-Wesley, 2003.
- [5] H. Batatia, A. Ayache, and H. Markkanen. Netpro: an Innovative Approach to Network Project Based Learning. In *Proc. International Conference on Computers in Education (ICCE'02)*, pp. 382–386, 2002.
- [6] M. Barker and K. Inoue. IT SPIRAL: A Case Study in Scalable Software Engineering Education. In *Proc. 22nd IEEE-CS Conference on Software Engineering Education and Training*, pp. 53–60, 2009.
- [7] 井垣宏, 柿元健, 佐伯幸郎, 福安直樹, 川口真司, 早瀬康裕, 崎山直洋, 井上克郎. 実践的ソフトウェア開発演習支援のためのグループ間比較にもとづくプロセスモニタリング環境. 日本教育工学会論文誌, Vol. 34, No. 3, pp. 289–298, 2010.