PBL 研修による組込みソフトウェア開発 技術者育成の有効性

Effectiveness of Project-Based Learning for Development Engineers of Embedded Software

三 浦 元^{※1}
Hajime MIURA

湯 川 高 志^{※2}
Takashi YUKAWA

山 崎 誠^{※3}
Makoto YAMAZAKI

The present paper analyzes relation between implementation and effectiveness of PBL (Project Based Learning). PBL has many variation on both process and team configuration, according to what to be learned. But essentially PBL gives good solution clearing the gap between 'can' and 'know' for novices. Novices easily make failures and these are important because 'every failure is a stepping stone to success'. To clear the gap using PBL and to get know-how applicable to an actual development, it is important recognizing and fixing failure in the process, and retrospecting them in a postmortem process. A PBL instructor should not prevent learners to make slight failures, and should guide them to make a small feedback loop for the failure in the process. The retrospection process is also required for the learners to organize their experiences into abstract know-how.

Keywords : Engineering Education, Project-Based Learning, Software Development, Embedded Systems

キーワード：技術者教育, PBL, ソフト開発, 組み込みシステム

1. まえがき

経済産業省による組込みソフトウェア産業実態調査が2004年に開始されてから6年が経過した。この調査により判明した組込みソフトウェア技術者の不足人数

は2007年度（調査は2006年）をピークに減少の傾向にある（図1）。しかし、真の課題は不足人数の充足にあるのではなく、高いスキルを持つ技術者の確保にある（図2）。なぜなら大規模化の傾向にある組込みソ

出典：経済産業省2009年版組込みソフトウェア産業実態調査報告書

図1 組込みソフトウェア技術者数の推移

平成22年11月11日受付

※1 株式会社テクノホロン

※2 長岡技術科学大学

※3 長岡工業高等専門学校

出典：経済産業省2009年版組込みソフトウェア産業実態調査報告書

図2 スキルレベル別不足率

図3 組込みソフトウェアの規模と開発工期
(経済産業省2006年版～2010年版組込みソフトウェア産業実態調査報告書から作成)

ソフトウェア開発においていかに品質を確保するかは競争力の面からも収益力の面からも非常に重要な課題であり、品質の向上には技術者の開発スキル向上が必須であるためである。また、大規模ソフトウェア開発であってもそれを一定の期間内で終えなければならない(図3)が、こういった開発を円滑に進めるためにはプロジェクト管理スキルが重要となる。つまり、品質を軸とした開発スキル、およびプロジェクト管理スキルをともに向上させていくことが企業現場で求められているのである。

本論文では、財団法人にいがた産業創造機構(NICO)が2006年から開催してきているPBL(project based learning)から得られた効果的な組込みソフトウェア研修についての知見を報告する。なお、以降では「初学者」という用語をリーダー、中堅、初心者といったそれぞれのスキルレベルで、新たな技術を習得しなければならない技術者という意味合いで使用する。

2. 組込みソフトウェア技術者の育成手段

技術者のスキルアップを図るための手段として、自習、集合研修、OJT(on the job training)、PBLなどが実践されている。自習と集合研修は短期間に目的の知識を得る手段として適しているが、初学者にとっては、学んだ状態、つまり「わかる」状態と実務で「使える」状態との間にあるギャップの乗り越えが課題となる。「わかった」うえで実務または実務に近い状態で実際に使い、その結果を反芻することではじめて「使える」状態となるためである。

OJTとPBLは実践型のトレーニングという点で、このギャップの乗り越えを容易にする利点を持つ。

OJTは組織の開発文化や要素技術を習得するのに適している。しかしながら、近年の企業の開発現場では初学者に失敗を経験させる余裕を失っており、初学者は、失敗による教訓の習得も、失敗に直面することから生まれるブレークスルーの体験もできにくくなってきている。このため、実務に直結したOJTは、失敗しないまでに基礎技術を「使える」ようになった技術者がそれらを基盤としてさらなる技術を習得するのに適しているといえる。

一方、PBLは現実から隔離された開発であり、失敗が許される安全な砂場であるといえる。PBLはまた、同じ課題を複数のプロジェクトチームで進行することで互いにベンチマークできるという特性も持つ。これらは実務でのOJTには無い特性である。

OJTとPBLはその特性の違いにより単純に優劣を比較できるものではないが、現在OJTでは困難になっている「失敗を安全に体験できる」という点から、「わかった」段階にある初学者が基礎学習の効果を高める段階ではPBLに優位性があるといえるだろう。

3. 財団法人にいがた産業創造機構（NICO）のPBL

多くのPBLは、同じスキルレベルにある集団のなかで役割を定め、それぞれの役割をこなす進め方を取る。NICOが2006年から開催している組込みソフトウェア技術者向けPBL研修では、異なるスキル段階にある社会人受講者がより実践的な体験を得られるように、スキルレベルに応じて開発全体を統括するチームリーダー、サブシステムをとりまとめるサブリーダー、サブシステムの実装を担当するプログラマの3つの役割に分かれ、チームを構成する。標準的なチーム構成ではリーダー1名、サブリーダー2名、プログラマ4名の7名構成となる。構成メンバーはスキルレベルに応じて、事前にプロジェクト管理技術、開発技術、プログラミングのいずれかの集合研修を受講している。また、同一チームには同じ会社のメンバーが入らないように配慮している。これはPBLのチーム運営のなかに暗黙の文化が侵入することを排除するためである。

NICOのPBLでは、DCモーターで動作する小型ローバーロボットのソフトウェアをC言語で作成することを課題としている。実命令行にして1,000行ほどの規模のものである。これについて見積もり、計画立案、仕様の分析、設計、実装、テスト、評価の全工程をチームで遂行する。最後の「評価」はローバーロボットを使ったチーム対抗レースである。なお、本PBLは期間として10日間を設定しているが、プロジェクトチームが開発に使用できるのは実質9日間である。

NICO PBLは、前記の「砂場」としての重要性を意識し、受講者が適度な（立ち直り可能な、現場でも教訓として活かせる）失敗体験を得られるように運営面で意図している。言い換えると、NICO PBLの目的は座学で得た知識を実際に使って開発を試行するなかで「教訓」を得て、それを実務に持ち帰ることであり、受講者個々人が役に立つ・質の良い教訓を得ることを

重要な効果と位置づけている。

4. NICOでのPBL研修の進め方と成果

2007年、2008年、2009年のPBLで、受講者が研修終了日に自身でピックアップした「教訓」の分布を図4～6に示す。2006年については、教訓の集計を行っていない。

2006年および2007年のPBLでは、受講者が失敗とそこからの復帰を体験することに主眼を置き、講師は定期的なリーダーレビューでの設計内容指導と明らかに致命的な失敗の指摘を除き、できるだけチームに干渉しないようにしていた。プロジェクトの事後検証すなわちポストモルテム（postmortem）にあてた時間は4時間である。ポストモルテムは主に成果物などについて数値指標をまとめ、それらをチームで評価したうえで個人ごとに教訓を書き出すかたちで実施した。

2007年に受講者が得た教訓（図4）にはアンチパターン、すなわち「べからず」が見られず感想の点数が相対的に多い。アンチパターンは「仕様書などで間違っていると思われる部分を自己解決しない」「リーダーが全体の状況把握をせず自分の作業に没頭してはならない」（2008年）「モジュールテストの質を落とすとつまらないバグの流出だけでなく、後半での重大バグ流出につながる」「修正要求を放置しておいてはいけない」（2009年）といったものであり、実務の中で失敗しないための最も直接的な教訓である。感想は「実装メンバーが最初にRTOSの講義を受けられたのは非常に良かった」（2007年）といったものであり、PBLでの対象、体制、環境に特化した教訓で、抽象度が低い。つまり教訓として汎化されておらず、そのままでは現場に持ち帰って活用できないものである。ポジティブな教訓は「問題が発生したときに、すぐに相談するなどのコミュニケーション能力が大事。コミュニケーションがすぐにとれるような雰囲気にすること」（2007

図4 2007年のPBLで受講者がまとめた教訓

年)「問題が起きそうなところほど早めに検証しておく」(2008年)「分析と仕様書の作成をしっかりすれば、プログラム作成に時間をかけないで済む」「一つ一つのモジュールは単機能におさめる」(2009年)といったものであり、こうしたほうが良い、あるいはこうすべきであるという内容の、いわば未来志向の教訓項目である。

また、この年はPBL進行中に過去チームの分析結果を見せていない。つまり参考とするものがない状態で初めてチームを組むメンバーで新規プロジェクトを遂行するという形を取った。従って環境的にはより失敗しやすい状態にあったといえる。

2007年のPBLでは、設計作業を各自のPCで行い、その印刷物でレビューをしていた。

図5は2008年の教訓のまとめである。2007年のPBLの運用から、講師がやりすぎと思えるくらい頻繁に干渉してもチームで失敗体験は得られることが感触として得られた。そこでプロジェクトの進行をチームに任せつつ、成果物、状況へのフィードバックとして設計

面では改善方針や受講者が見逃した設計ミスとそれにより発生が予想される不具合とを、プログラムについてはC言語として危険な用法を指摘するようにし、プロジェクト管理面ではリーダーが問題・課題局面で何をすべきかを頻繁に指導するようにした。また、過去の設計成功例、設計失敗例を見せるようにした。

2008年のPBLでの最大の改善点は、ポストモルテム工程の最初に振り返り手法によるグループワークを実施し、それに続けて前年と同じ数値指標の収集と評価、教訓の抽出を行うようにしたことである。振り返りには複数の手法があるが、ここでは参考文献⁹⁾にある「タイムライン」手法を採用した。このためポストモルテムのための時間を5.5時間に拡張した。図5に見られるように「教訓」の数が大幅に増え、図4に見られなかったアンチパターンも見られるようになったが、これらの改善は全工程を俯瞰しなおし、工程の中での失敗がどこを基点に発生したか、それはどこでどのような作業を行うことで、あるいは作業をどのように改善することで予防できたか、また、どのポイントで何を

図5 2008年のPBLで受講者がまとめた教訓

図6 2009年のPBLで受講者がまとめた教訓

したことが結果に結びついたか、をチームで議論したことによるものと考えられる。具体的には「時間に追われてレビューがおざなりになったが、振り返ってみるといくつもの不具合がレビュー不足に起因することを実感できた。レビューは重要であり、きちんと実施すべきものである」といった教訓が生まれている。つまり、失敗を過程の面・他の事象との関連面から分析し、予防策を議論することで経験を汎化できたことになる。質的にも教訓の表現が実務にも適用できる抽象度の高いものとなっている。

2008年のPBLからは各チームでホワイトボードを自由に使用できるようにし、設計時にホワイトボードが活発に利用された。これは「全員の意識が共通していないと作業の進展に支障が生じる」(2008年)「分析と構造設計では共通認識を基に作業を進めることで間違いが少なくなる」(2009年)などの教訓となっており、設計内容だけでなく設計過程の共有化という面でメリットが大きかったと見られる。これが教訓のうち「コミュニケーションに関する教訓」を大きく増やした原因のひとつであろう。図5に見るように、図4に対して2008年では教訓の項目数が全般に増加している。

図6は、2009年PBL終了時に受講者がまとめた教訓の集計である。2009年のPBLはリーマンショックの影響を受け社会人受講者の申し込みが4名だけであったため、長岡技術科学大学、長岡高専の学生からプログラマとして参加してもらうことで研修を成立させた。

2009年のPBLは進め方とポストモルテムのメニューで2008年を踏襲し、そのうえで受講者の知識が不十分であると感じられた状況で講師が設計文書などを前に設計手法やプロジェクト管理の基本(原則)について簡単な指摘・指導を行うという、新たなフィードバックを加えている。形態としては実践の中に講義を短サイクルで挟んだともいえる。グラフの傾向として2008年の教訓の傾向と同じといえるが、感想の件数が減り、アンチパターンの件数が増えたことが特徴的といえる。これは受講者が基本に対する自分達の誤った方向(失敗)を認識できていることを示している。

研修は、受講者がそこで学んだことを企業に戻って応用できてこそ価値がある。図7は、NICOが2010年7月に2009年以前の社会人の全受講者に対して行った追跡アンケートの結果である。PBLでの経験が業務に役立っている領域として「コミュニケーション」(2007年・プログラマ)「設計技術」(2008年・サブリーダー)「管理技術」(2008年・リーダー、2009年・リーダー)といった回答が得られている。

図7からは、2008年の受講者全般で教訓が業務で役立っていると実感している様子が見える。これは調査に時間的に近い(印象が新しい)だけではなく、失敗

図7 修了者への追跡アンケート結果

／成功の経験を振り返りにより認識し、定着できたことによるものと考えられる。2009年の結果については、社会人の参加人数がもともと非常に少なかったこともあり、参考程度にとらえるべきであろうが、「リーダーとして参加」の並びを見ると、2007年以前と2008年以降とで明確な差があると判断できる。

5. 考察

図4～7に見える変化とそれぞれのPBL実施時の運営側での施策について考察する。

まず、ポストモルテム工程で成果物についてバグ指摘数といった解剖学的な数値指標をまとめるだけでは、PBLでの経験を十分に汎化できない。他のプロジェクトにも経験を応用できるようにするためには結果を得るまでの過程を振り返り、さまざまな事象や行動がどのように関連して結果に結び付いたのか、本来どのタイミングでどうすべきであったのか、といった知見を「再発見」することが重要である。

しかしこれだけでは不十分である。講義で得た知識をPBLの実践中で確認する小さなフィードバックループを多く回すことも重要である。これは2008年度(図5)から2009年度(図6)への変化として現れている。フィードバックでは基本の確認をすること、知識の適用内容・適用手法の是非に気付くこと(失敗の認識など)、どうすべきかのアクションを実施することを含む。フィードバックループはチーム内のレビューでも回しうるが、これにはチームメンバー全員が十分なスキルを持っている必要がある。PBLは受講者が事前に講義を受講してきていることが前提であるが、受講から数ヶ月から1年の時間を経ているためせっかくの知識が白紙に近くなっている場合が多い。そのような状態では経験の少ないチームメンバー相互での指摘よりも講師による指導がより有効なのは言うまでもない。ただし、指導の際に講師は基本の確認を促すにとどめ、受講者に答えを探させることが重要である。そしてこのことが振り返りの際に反芻されることで相乗効果が生まれると考えられる。

つまり、プロジェクト中に講師が介入することでフィードバックループを頻繁に回すことと、ポストモルテム工程で成果物だけでなく工程を振り返ることと

が抽象度の高い・再利用性の高い教訓を得ることにつながり、その結果が図7で見られるようなPBLでの体験が実務で役立っているという回答の増加につながっていると言える。

6. まとめ

品質の良いソフトウェアを得るために、要素技術だけでなく開発技術およびプロジェクト管理技術が重要であり、開発技術では特に上流工程の技術（設計技術）で技術者がいかにスキルアップするかが課題となる。PBLはOJTの前段階として、座学で得た知識を実際に使ってみる場として要素技術・開発技術・プロジェクト管理技術のいずれにおいても有効である。

PBLは受講者にとって経験の浅い管理技術、開発技術や要素技術を実際に使用する場となる。初学者は知識の不十分さ、知識適用の慣れの無さから失敗しやすいものであるが、PBLでは安心して失敗することができ、何をしたらまずいいのか、どうしたらうまく行くのか、何に心がけるべきなのかといった教訓を得ることができる。

本論文では、NICOが4年間にわたって実施した組込みシステム開発PBLに対し、その実施形態と学習効果との関係を分析した。その結果、PBLでは講師が介入することでフィードバックループを多く回すこと、ポストモルテムでの振り返りにより工程全般を反芻させることなどにより、受講者がその経験を認識し、実務に応用できるだけの汎化した「教訓」が得られることが明らかとなった。

謝 辞

本研究の一部は、総務省の戦略的情報通信研究開発推進制度（SCOPE）から財政的支援を受けた。

図1および2は、経済産業省「2010年版組込みソフトウェア産業実態調査報告書」より引用している。また、図3は同省2006年版～2010年版の「組込みソフトウェア産業実態調査報告書」に掲載されたデータから作成した。

図4から図6はNICO PBLのデータから、図7はNICOによる修了者追跡アンケート結果から作成した。

以上の財政支援、およびデータの利用について御礼を申し上げる。

参 考 文 献

- 1) 経済産業省商務情報政策局 情報政策ユニット情報処理振興課, 2004年版組込みソフトウェア産業実態調査報告書, 2004, Webページ, <http://sec.ipa.go.jp/reports/>, 参照日: 2010-8-1 (1)～(7)のURLおよび参照日は同じ)
- 2) 経済産業省 商務情報政策局 情報政策ユニット情報処理振興課, 2005年版組込みソフトウェア産業実態調査報告書, 2005, Webページ
- 3) 経済産業省商務情報政策局 情報政策ユニット情報処理振興課, 2006年版組込みソフトウェア産業実態調査報告書, 2006, Webページ
- 4) 経済産業省商務情報政策局 情報政策ユニット情報処理振興課, 2007年版組込みソフトウェア産業実態調査報告書, 2007, Webページ
- 5) 経済産業省商務情報政策局 情報政策ユニット情報処理振興課, 2008年版組込みソフトウェア産業実態調査報告書, 2008, Webページ
- 6) 経済産業省商務情報政策局 情報処理振興課, 2009年版組込みソフトウェア産業実態調査報告書, 2009, Webページ
- 7) 経済産業省商務情報政策局 情報処理振興課, 2010年版組込みソフトウェア産業実態調査報告書, 2010, Webページ
- 8) 井上 明: PBL情報教育の学習効果の検証, 情報処理学会研究報告, 情報システムと社会環境研究報告, 25, pp.123-130, 2007
- 9) Esther Derby, Diana Larsen著, 角 征典 訳: アジャイルレトロスペクティブズ 強いチームを育てる「ふりかえり」の手引き, オーム社, 2007

著 者 紹 介

三浦 元

1984年3月新潟大学大学院理学研究科修了。株式会社東京精密、株式会社インテックソリューションパワーを経て株式会社テクノホロン取締役。特定非営利活動法人組込みソフトウェア管理者・技術者育成研究会理事。財団法人にいがた産業創造機構プロジェクトマネージャ
Hajime.Miura@techno-holon.co.jp