

On the Job Learning: 産学連携による 新しいソフトウェア工学教育手法

小林 隆志[†] 沢田 篤史^{††} 山本晋一郎^{†††} 野呂 昌満^{††} 阿草 清滋[†]

[†] 名古屋大学 大学院情報科学研究科 〒 464-8601 愛知県名古屋市千種区不老町

^{††} 南山大学 情報理工学部 〒 489-0863 愛知県瀬戸市せいれい町 27

^{†††} 愛知県立大学 情報科学部 〒 480-1198 愛知県愛知郡長久手町熊張茨ヶ廻間

E-mail: †tkobaya@is.nagoya-u.ac.jp

あらまし 本稿では、産学が密に連携し、企業における実際のソフトウェア開発を題材とし実践的な教育を行う新しい教育手法 On the Job Learning(OJL) を提案し、実際に 2 年間実施した経験からその実施方法や教育効果を議論する。

キーワード ソフトウェア工学教育, PBL, OJL, 産学連携

On the Job Learning: A New Software Engineering Education Method on Collaboration between Academia and Industry

Takashi KOBAYASHI[†], Atsushi SAWADA^{††}, Shinichiro YAMAMOTO^{†††}, Masami NORO^{††},
and Kiyoshi AGUSA[†]

[†] Grad. School of Information Science, Nagoya University

Furocho, Chikusa-ku, Nagoya, Aichi 464-8601 Japan

^{††} Faculty of Information Sciences and Engineering, Nanzan University,

27 Seirei-cho, Seto, Aichi 489-0863 Japan

^{†††} Faculty of Information Science and Technology, Aichi Prefectural University

Ibaragabasama, Kumabari, Nagakute, Aichi-gun, Aichi 480-1198 Japan

E-mail: †tkobaya@is.nagoya-u.ac.jp

Abstract In this report, we propose a new software engineering education method “On the job Learning (OJL)” which is a project based practical education based on collaboration between academia and industry with real software development projects. We also review the implementation and the effectiveness of our proposed education method with our two-year practice.

Key words Software Engineering Education, PBL, OJL, Collaboration between Academia and Industry

1. はじめに

IT 分野の発展は目覚しく、産業界における高度 IT 人材の不足が議論されるようになってきた [1]。我が国の国際的な競争力の低下を防ぐためにも、高度 IT 人材を排出するために効果的なカリキュラムや教育方法が望まれている。我々は、文部科学省が進める先導的 IT スペシャリスト育成推進プログラム^(注1)の一拠点のメンバとして 2006 年度か

ら活動してきた。この取り組みの中で、我々はソフトウェア工学に関する基礎的な技術力、それら技術を実践する力、変化へ適応する力を養成するための教育カリキュラムを設計し実際に実施している。

本稿では、我々が設計した教育カリキュラムの中核を担う、新しい教育手法 OJL(On the Job Learning) について説明する。OJL は、PBL に代表される教育用の仮想プロジェクトを題材とする教育と、インターンシップや OJT のように実際の業務を体験することによる実務訓練を融合し、産学連携でより実践的な教育を行うことのできるプロジェク

(注1) : http://www.mext.go.jp/a_menu/koutou/it/index.htm

ト参加型の教育手法である。産業界の協力のもと、実際のソフトウェア開発を題材にすることで、開発技術の実践的な内容を深く理解させ、それが訓練ではなく教育となるように教員が指導を行うという点が特徴である。

本論文では、OJLの実施方法を説明し他の教育手法との差異を議論するとともに、これまで複数の企業からの支援のもとで我々が実施してきた実施事例を紹介しその実施方法や教育効果について考察する。

以下では、まずIT教育における既存のプロジェクト参加型教育に関して概観し、それぞれの特徴を議論した上で、3.でOJLの概要を説明し、実施方法や教員や産業界からの人員の役割、テーマの選定方法に関して説明する。その後、4.で我々がこれまでに実施してきた事例を紹介し、5.で本OJLの有効性や課題について考察した後、6.で本稿をまとめる。

2. IT教育におけるプロジェクト参加型教育

知識の獲得状況を説明する上で有名な分類に Bloom らの4段階の知識レベル [8] がある。それぞれをIT教育に当てはめると下記のように考えると考える。1) 教科書や講義などで見聞きしたことがあり、関連する資料を調べることができる (Knowledge), 2) 言葉の意味を理解し、説明することができる (Comprehension), 3) 言葉に関する技術・技法をソフトウェア開発に利用することができる (Application), 4) 技術の背景にある制約や特性について理解し、特定の問題に対する向き・不向きを比較評価できる (Analysis)。技術者育成の観点では、Application レベル以上の知識として各ソフトウェア開発技術を獲得させることが特に重要である。

プロジェクト開発型の実習は、専門知識や技術を実践する能力を養う点で効果的であるとされ、先導的ITスペシャリスト育成推進プログラムでも各拠点がPBL型の教育プログラムを採り入れている [2,3]。我々の実施する教育カリキュラムにおいても飛行船の制御プログラムを題材としたPBLを実施している [4]。仮想開発を題材にするPBLでは、技術や開発プロセスの実践が主目的となり Analysis レベルの知識獲得は比較的困難である。Analysis レベルに到達するためには、単に座学として簡単な例題に触れることや、特定の技術を適用するために設計された仮想プロジェクトを実践するだけではなく、現実問題に近い課題を設定し、問題を理解したうえで代替案とあわせて技術解を検討し、その技術を適用させる経験が重要となる。

現実問題に近い課題を設定したプロジェクト参加型教育は多く存在し、実システム開発を題材として用いた事例 [5] や、産学連携による取り組みが効果的であること [6] 等が報告されている。海外においても、産業界におけるソフトウェア開発課題を利用した教育を行う事例に [7] 等がある。また実際の開発を題材とし教育を行う類似手法として、OJT [9] がある。OJLは、社員教育方法の1つで、実際の仕事を通じて、必要な技術、能力、知識、あるいは態度や価値観などを身に付けさせる教育訓練のことであり、多く



図1 OJLの概要

の企業で導入されている。学生に対する教育という点では、学生を一時的に企業の一員とし、就業体験をさせることで業務知識を獲得させるインターンシップも存在する。

これらプロジェクト参加型教育の中で現実問題を扱う場合は、教育効果が指導者の素質に強く依存することになる。筆者らが [7] の Melbourne 大学の実施担当者らにインタビューをした際も、指導者役の選定は大変重要かつ難しいポイントであるとの回答を得た。特定のドメインや企業における開発方法論の訓練ではなく教育を実施するという観点では、プロジェクトを運用し開発を進める実務能力と、それを題材に指導を行う教育能力の双方が必要となる。

3. OJL: On the Job Learning

3.1 概要

我々が提案する OJL (On the Job Learning) とは、産学共同による新しいプロジェクト参加型の教育手法である。仮想的な課題で開発を学ぶ PBL では実製品レベルのソフトウェア開発における品質や管理の難しさを真の意味で実感することができない一方、特定の組織における業務のあり方を身につける OJT ではそこで教授される技術や作法の普遍性、本質的な価値を学ぶ事ができない。OJL では、PBL と OJT を組み合わせ、互いにその欠点を補完しあうことでより高い教育効果を狙っている。

図1にOJLの概要を示す。OJLでは、産学それぞれの立場から明確な役割を担う要員が参画する開発プロジェクト(OJLプロジェクト)を形成する。

典型的なOJLプロジェクトには、企業側から指導者と開発者が参画し、大学側から教員と学生が参画する。企業指導者は製品レベルの実システム開発課題をプロジェクトに持ち込むとともに、プロジェクト管理者の立場から、実システム開発のために必要な開発・管理のスキルの指導を行う。企業開発者は学生とともに、システムやソフトウェアを開発する要員として必要に応じてプロジェクトに加わる。一方、大学教員は、プロジェクトにおいて学生の課題遂行を管理する役割を通じ、与えられた開発課題のドメインがもつ特徴や制約、さらに課題を解決するために用いられる開発や管理の技術を普遍的な視点からとらえ、問題と解法の本質を学生に教授する役割を担う。

表1は、OJLとほかのプロジェクト参加型の教育方法を比較したものである。実施拠点を大学内に設置することで、学生が長期間プロジェクトに参加することを可能とし、インターンシップで問題となるような、従事可能な期間が短

表1 OJL とプロジェクト参加型教育の比較

方式	対象者	指導者	題材	期間・規模	主な目的
PBL	学生・社員	教育担当者	教育用プロジェクト	6ヶ月	専門的スキルの伝授
OJT	社員	業務担当者	現実の業務	2ヶ月	社内業務への適応
インターンシップ	学生	業務担当者	現実の業務	1ヶ月	ジョブマッチング
OJL	学生・社員	業務担当者と 教育担当者が協力	現実規模のプロジェクト	18ヶ月	専門的スキルの伝授と 変化への対応力の育成

いために開発技術の十分な実践や教育ができないことを防いでいる。学生は、大学に在ながら各 OJL プロジェクトが定める週報等の定期的な進捗報告や、工数・従事時間の管理、機密情報に関する取り扱い規則などを経験し、就業体験を超えたより実践的な学習を実施することができる。また教員が開発プロジェクトに参加をすることで、その技術や作法における普遍性、本質的な価値を教示し、特定の企業の開発スタイルに偏らない教育を実践することができる。

3.2 OJL での教育目標

OJL を通じた教育の主要な目標は次の三点である。

- 製品レベルの実システム開発の体験
- 開発や管理に関するスキルの修得
- 開発課題の特徴に応じた適用技術選択能力の獲得

一点目と二点目の目的は、製品に対する品質要求の高さ、高品質を達成するために現場で適用されている開発や管理に関わる諸技術について、実際にプロジェクトに携わり、適用することで体験をすることであり、それを通じて教科書的に学んできた技術に対する制約や特徴を深く理解することにつながると期待している。三点目は、与えられた開発課題の特徴や制約を見抜く力を身につけることであり、それを通じてその課題に適用する技術を取捨選択し、また必要に応じてその課題に最適な専用の技術を創造することのできる能力が養われることを期待している。これらの目的は、産と学、どちらかのみが主体となる体制により実現できるものではなく、産学双方の強い協調関係の上になつてはじめて実現できるものである。

なお、教育と指導の対象となるのはあくまで学生であるが、OJL は企業から参画する開発者にとってもメリットを享受できる良い機会であるといえる。学ぶ立場にある学生とチームを組み、技術を伝授する指導的な立場で開発を行いながら、指導者がプロジェクトを通じ学生をどのように使いた管理するのかを見ることで、プロジェクト管理のノウハウを学ぶ機会になる。また、学からの普遍的な視点を入れることで、企業内に閉じた開発技術や仕事の作法の本質を理解することにもつながる。

3.3 テーマ選定とそのタイプ

OJL プロジェクトを実施するにあたっては、そこで解決すべき開発課題を与え、教育目標を達成するためにとるべき手段、またそこで何に注目した指導を行うべきかについて決定しなければならない。本節では、開発課題についての、開発形態と管理形態の軸に沿った特徴に基づいて大まかに分類し、それぞれのタイプにおけるテーマ設定指針に

ついて議論する。

開発形態の軸では、表 2 に示すように、新規開発型と既存資産改良型に大別することができる。

新規開発型の OJL は、新製品などの先行的開発として設定することができる。市場・顧客・組織からの新たな要求や、新たなハードウェアデバイスの出現などに伴って必要となるソフトウェア開発を試験的に行うプロジェクトがこれにあたる。指導ポイントとして、顧客要求の獲得や分析に関わる技術やソフトウェアアーキテクチャ設計やインタフェース設計の技術が挙げられる。このタイプのプロジェクトでは、例えば要求からソフトウェアシステムが解決すべき問題の特徴や制約をとらえる技術や、製品に求められる仕様(機能・非機能)を満たすために最適なアーキテクチャを設計したり代替案の中から取捨選択したりする能力を養うようなテーマ設定が可能である。

既存資産改良型の OJL は、既存の製品など、長年にわたる保守を経て品質が劣化したソフトウェアの品質改善や再設計のプロジェクトとして設定することができる。指導ポイントとして、再利用を意識したソフトウェア資産の構築や、保守性や可搬性を高めるための設計改善などリエンジニアリングの技術が挙げられる。このタイプのプロジェクトでは、例えば、再利用を意識したプロダクトラインの構築技術、品質向上のために最適なパターン技術を選択して適用できる能力を養うようなテーマ設定が可能である。

両方の型に共通して着目すべき開発技術として検証技術が挙げられる。いずれにタイプの OJL においても製品が満たすべき品質を確保するために適切な検証の技術を選択したり組み合わせたりする能力を養うようなテーマ設定が可能である。

一方、管理形態の軸では、表 3 に示すように、企業からの参画者の関与の仕方に応じて、請負型と共同開発型に大別することができる。

請負型の OJL では、企業からの指導者は製品の顧客に近い立場、あるいは開発元請けの立場からプロジェクトに参加する。したがって、プロジェクトは主に学生からなるチームにより運営され、学生は管理と開発の両方の役割を担うことになる。このタイプのプロジェクトの指導ポイントとしては、要求やリスクの分析に基づく開発計画や、開発目標の設定や計測、管理のための技術が挙げられ、計画技術や管理技術の実適用における特徴や制約に関する知識、課題の特徴や開発技術の制約に応じた管理技術の選択に関する能力を養うようなテーマ設定が可能である。

表2 開発形態軸に基づくテーマ設定

開発形態	開発目的	着目する開発技術
新規開発型	先行開発・試作	要求工学, 設計, 検証
既存資産改良型	品質改善	リエンジニアリング, 検証

表3 管理形態軸に基づくテーマ設定

管理形態	指導者の役割	着目する管理技術
請負型	顧客・元請け	開発計画, 開発管理
共同開発型	プロジェクト管理	構成管理, 技術管理

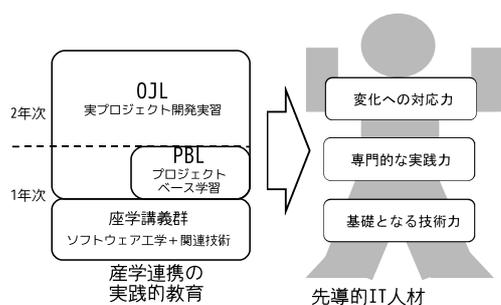


図2 OCEANの概要

Fig. 2 Overview of the OCEAN program

共同開発型の OJL では、企業からの指導者がプロジェクト管理者として参画し、学生は管理者のもとで開発を担う要員としてプロジェクトに携わる。このタイプのプロジェクトの指導ポイントとしては、構成管理や技術管理のための技術が挙げられ、製品の特徴に照らして適切な構成管理方式を選択する能力や製品の開発にとって適切な技術や技術の組合せを比較検討したり、開発のプロセスの改善を提案したりする能力を養うようなテーマ設定が可能である。

4. 実施事例と成果

我々は、先導的 IT スペシャリスト育成推進プログラムの下で、本稿で提案する OJL を教育の中核におく教育プロジェクトとして、名古屋大学を中心とし、南山大学、愛知県立大学、静岡大学、ほか連携企業 6 社が参画する「OJL による最先端技術適応能力を持つ IT 人材育成拠点の形成」(OCEAN: On the job Centered Education for Advanced engiNeers)^(注2)と銘打つ教育プロジェクトを実施してきた。

4.1 OCEAN のカリキュラム

OCEAN の概要を図 2 に示す。OCEAN では高度な専門性を持つ IT 人材に求められる素養を、ソフトウェア工学に関する基礎的な技術力とその実践力、変化への適応力と定義し、座学での講義群に、PBL および OJL (On the Job Learning) という開発実習型の科目を組み入れたカリキュラムによって、大学院修士課程の学生へこれらの素養を教授することを狙っている [4]。

座学での講義は修士課程の 1 年目に配置される。講義科目は、技術倫理、ネットワーク、ソフトウェア構築、ソフトウェア保守といった基礎科目に加え、ソフトウェア工

学を中心とした技術要素に対して、深く掘り下げた内容を教授できるように、ソフトウェア要求工学、ソフトウェアアーキテクチャ、ソフトウェア設計技術、ソフトウェアモジュール化技術、正当性検証と妥当性確認、ソフトウェアプロジェクト管理、の 6 つ必修科目を配置している。また、各開発ドメインでの知識を教授するために、組み込みシステム開発技術、分散システム開発技術、情報システム開発技術、といった講義科目を用意している。

各科目半期 1 講時ずつの授業時間 (1.5 時間 × 15 回の 22.5 時間) を割り当て、ソフトウェア工学のカリキュラムガイドライン SE2004 の知識体系のうちコンピュータサイエンス基礎と数学基礎を除く大項目を全科目でほぼカバーするように設計されている。

4.2 OJL の実施

1 年半の OJL 実施期間のうち、修士 1 年後期は座学での講義や PBL と並行して行うことから、開発問題の概要や背景知識の理解などにあて、本格的な課題への取り組みは修士 2 年次の 1 年間に行う。

表 4 はこれまで実施してきた OJL テーマである。★印は、本教育カリキュラム 1 期生 (2007 年 10 月～2009 年 3 月) と 2 期生 (2008 年 10 月～2010 年 3 月) の双方のテーマとして継続実施されたテーマである。

以下ではいくつかのテーマに対してその内容や実施体制、成果物に関して紹介する。

4.3 実施事例 1: トヨタ自動車株式会社との事例

トヨタ自動車株式会社の協力の下で実施した OJL では、車載ソフトウェアのシステムバリエーションを構築・管理する際の支援手法およびツールを開発した。

車載ソフトウェアは、対象となる車種のグレードやモデル、各国における仕様等の差異から、多様なシステムバリエーションを管理する必要がある。現時点でのこれらの構成管理は、版管理と、ifdef などのコンパイルオプションによる管理を併用しているが、プロジェクトや担当者によって方法が異なり、きちんとルール化されていない。近年では、複数の機能を 1 つの ECU に組み込むことや、製造コストの観点から動的に機能を選択することなど、新しい要求が出てきており、構成管理の効率化が望まれている。

この事例では、それらの問題を把握できるよう、対象ドメインの知識獲得を行ったあと、構成管理対象となるソースコードを開発した。構成管理対象には、Adaptive Cruise Control を選択し、車両への搭載前にシミュレーションを行うための実装として、Simulink 上で S-Function として動作するプログラム (約 3,500 行) を実装した。その後、開発コードを対象に支援方式の検討を行ない、1) コンパイルスイッチ間の関係抽出、2) 関係定義に基づくコード検証、を行う Eclipse プラグインの開発を行った。

支援ツールはコンパイルスイッチ間の関係の解析結果をもとにその関係図と SMT 式を生成する。関係図として可視化することで、人間がコンパイルスイッチ間の関係について理解しやすくなる。また、コンパイルスイッチ間の親

(注2) : <http://www.ocean.is.nagoya-u.ac.jp/>

表4 これまでに実施した OJL テーマ

テーマ名 (協力企業)	実施大学
システム構成管理技術の研究 (トヨタ自動車株式会社)★ [4.3 で紹介]	名古屋大学 愛知県立大学
次世代運転支援システムの情報表示系機能を検証するツールの開発 (株式会社デンソー)★	名古屋大学
ラベルライターの外国語向けソフトウェア開発 (ブラザー工業株式会社)	名古屋大学 愛知県立大学
インクジェット複合機での表示機能 (ブラザー工業株式会社)	名古屋大学 愛知県立大学
次世代LANプログラム開発 (株式会社オートネットワーク技術研究所)★	名古屋大学
半導体製造装置のFA通信ソフトウェア開発 (東京エレクトロンソフトウェア・テクノロジー株式会社)	南山大学
自動販売機制御ソフトウェアの再開発 (富士電機リテイルシステムズ株式会社)★	南山大学
組み込みソフトウェアに向くコンポーネント仕様の研究 (アイシン精機株式会社)★ [4.4 で紹介]	名古屋大学
Java ソースコードのCDI(Code Inspection)の開発 (株式会社キャナリーリサーチ)	南山大学
多軸制御装置のためのNC言語コンパイラの開発 (株式会社エヌエスティー)	静岡大学
X線画像処理システムの開発 (富士フイルムイメージテック株式会社)	静岡大学
Optimized Link State Routing and Localizationの開発 (三菱電機株式会社)	静岡大学
大学事務システムの開発 (大学主導)	南山大学
Multicore 対応リアルタイム OS の開発 (大学主導)★ [4.5 で紹介]	名古屋大学

子関係や排他関係といった関係を利用し、制約上衝突しているコンパイルスイッチがないか、現在宣言しているコンパイルスイッチ群が十分であるかを検証することができる。

この事例では、開発したツールおよび、その教育効果もさることながら、産学連携で教育や研究を実施するうえで利用できる車載ソフトウェアのソースコードは非常に稀であるため、当該 OJL で作成した上記 ACC のソースコードへの注目が高く、現時点で複数箇所から借用の依頼がある。

4.4 実施事例 2: アイシン精機株式会社との事例

アイシン精機株式会社の協力の下で、車載ソフトウェアに代表される組み込みソフトウェアに対するコンポーネント化を促進するための支援方法およびツールの開発を行った。

車載ソフトウェアに代表される組み込みソフトウェアでは、厳しい時間制約、リソース制約、デバッグの難しさなどから保守性や再利用性が低い記述を強いられているという問題がある。例えばグローバル変数の使用は、組み込みソフトウェア以外のソフトウェアにおいてはモジュールの独立性を下げ、ソフトウェアの品質を下げる原因として避けられるのが一般的となっている。しかし、グローバル変数はパフォーマンスやメモリ使用量の予測の容易さなどから組み込みソフトウェアの開発の現場では使用されることも少なくない。

この事例では、まず、実際の製品ソースコード分析し、時間及び空間効率を犠牲にすることなく再利用性を確保する方法を検討した。上述のソフトウェアは、グローバル変数によって構成された黑板に対して、各処理モジュールが知識源としてアクセスし処理する黑板アーキテクチャであることに着目し、1) 各知識源に相当するモジュールの仕様外の入出力、2) 黑板を介さない知識源同士の値の受け渡し、を禁止するための支援ツールを作成した。

支援ツールの開発では、まず黑板アーキテクチャに基づく黑板・知識源の仕様定義方法を定め、次にこの仕様に基づいて、1) コードテンプレートを生成する支援ツール

BB-Generator、2) ソースコードが仕様を満たすように黑板(グローバル変数)へアクセスしているかを確認するツール CX-Checker [12] の2つを並行開発した。

CX-Checker は、黑板アーキテクチャ仕様以外にも、コーディング規約をチェックできる Eclipse プラグイン開発された。MISRA-C や GNU コーディングスタンダードといった代表的なものだけでなく、社内独自のコーディング規約に対して柔軟に対応できるように、チェックするルールに対する高い柔軟性を実現している。ソースコードの抽象構文木を XML によってマークアップした中間形式を採用し、XML 文書に対する XPath 記述によるルール記述や、構文要素の複雑な関係を利用するための DOM 及び独自の API を利用した Java 言語によるルール記述をサポートする。

開発は、企業側4名、学生2~5名(1期生が2名、後半より2期生3名も参加)、教員3名によって、3フェーズのインクリメンタル開発によって実施された。毎週の全体ミーティングおよび週報による報告に加え、メールベースでの議論・指導がおこなわれた。また、開発されたツールを実際にアイシン・コムクルーズ株式会社において使用して頂く導入試験や、黑板アーキテクチャに基づく制約を満たすようにリファクタリングを行ったソースコードを、実車に組み込み、走行試験を行うなど、製品や業務に直結した開発を行った。

4.5 実施事例 3: 学内組織主導での事例

この事例では、NPO 法人 TOPPERS プロジェクトの協力を得て名古屋大学大学院情報科学研究科 附属組み込みシステム研究センターが主導して、汎用性と拡張性を備えたトレースログ可視化ツール TLV を開発した [10]。

TLV(TraceLog Visualizer) は、トレースログを一般化した標準形式と各種フォーマットへの変換ルールを定め、その汎用性を確保している。また、トレースログの可視化表現を指示する仕組みを抽象化し、可視化ルールとして形式化することで、高い拡張性を実現している。当該 OJL で作

成したこのツールは、その完成度と有用性から Toppers プロジェクトにおいて、会員向け早期リリースをされる [11] など実用性の面でも高い評価を受けている。

TLV の開発では、企業出身者 2 名と教員 1 名がプロジェクトマネージャを務め、学生 3 名と企業出身者 2 名が開発実務を担当した。進捗の報告は、週に 1 度のミーティングと週報の提出により行われた。開発は初期段階でプロトタイプを作成し、そのプロセスと成果物から、GUI の評価や要求の再抽出、アプリケーションドメイン分析を通じた設計/実現方法を検討し、その後、ユースケース駆動によるアジャイル開発によって開発が行われた。ツールは C# で実装され、総行数は約 18500 行であった。

5. 考 察

教育手法の効果を定量的に議論することは非常に困難であり、現時点では OJL という教育方式の客観的かつ定量的な効果を示すことはできない。しかしながら、参加している教員や実際に指導を受けた学生の意見から、企業の実際のプロジェクトマネージャの管理の下、実務者と一緒に開発作業を行うこと、特に品質や納期、工数に関する意識の差に関しては高い学修効果があることが見受けられた。また、企業側担当者からも、「インターンシップでは、企業側ができる範囲で作業をしてもらうことになる。それに対し OJL では、教員がともに参加し教員からの知見を反映しながら実施するため企業としても、大変やりがいがある」「学生が対象ということで、OJL テーマのレベルを心配していたが、基礎知識があり、学習意欲が高いため順調にプロジェクトを実施することができている」という意見があるなど、OJL 方式の一定の効果は確認できる。

また、学生の学修結果の評価方法に関しても、OJL に特化した評価方法が必要である。現段階では半期毎の報告会と最終報告において、プロジェクトの開発成果を報告するだけでなく、そこから何を学んだかを報告させ評価するようにしている。

重要な観点は、学生が実問題を対象とした実践の開発を通して何を学んだかであり、「なにを」ではなく「どう」開発したかに重点をおく。このため、開発対象の新規性や有用性よりも、期間内に開発問題をどう解いたか、座学で学んだ技術をどのように適用したのかの説明を求める。また、Application レベルの知識を獲得できているかを確認するために、「なぜ」その方式や技術を選択/実施したのかの説明も求め、対象問題を抽象化し得失を検討できているかを確認している。

定例ミーティングや半期毎の報告会、最終報告書において、どのように技術解を求めたかを振り返らせることは、学生自身が体験した内容を Application レベルの知識として洗練するために効果的であり、高い学修効果があるものと考えている。

6. おわりに

本稿では、PBL に代表される教育用の仮想プロジェクトを題材とする教育と、インターンシップや OJT のように実際の業務を体験することによる実務訓練を融合した、新しい教育手法 OJL を提案した。プロジェクト参加型の教育手法との差異を明確にし、我々が 2 年間実施してきた事例をもとにその有用性を議論した。

今後は、履修生に対する追跡調査などを行うことで、OJL による教育効果を明確にするとともに、有効な評価基準や方法を検討していく必要がある。また、ソフトウェア分野に限らず他の分野へ適用範囲を拡大すること、特定の IT 職種の教育に特化した教育手法として洗練することも検討課題である。

謝 辞

OJL による教育の実践は、文部科学省研究拠点整備等補助金（先導的 IT スペシャリスト育成推進プログラム）による助成のもとで行われた。OJL は連携企業関係各位の教育に対する熱意により支えられている。ここに記して謝意を表す。

文 献

- [1] 山下 徹：“高度 IT 人材育成への提言～国際競争力の復権にむけて” NHK 出版 2007.
- [2] 文部科学省：“先導的 IT スペシャリスト育成推進プログラム 8 拠点のプロジェクト概要 — 世界最高水準の高度 IT 人材育成を目指して —”，2008.
- [3] 坂本 憲昭，峯 恒憲，日下部 茂，深瀬 光聡，荒木 啓二郎，福田 晃：“大規模な産学連携による高度 ICT 人材教育におけるインターンシップの役割とその効果”，情報処理学会論文誌，Vol. 49, No. 10, pp. 3388–3398, 2008
- [4] 沢田篤史，小林隆志，金子伸幸，中道上，大久保弘崇，山本 晋一郎：“飛行船制御を題材としたプロジェクト型ソフトウェア開発実習”，組込みシステムシンポジウム 2008 論文集，pp. 5–14, 2008.
- [5] 井上明，金田重郎：“実システム開発を通じた社会連携型 PBL の提案と評価”，情報処理学会論文誌，Vol. 49, No. 2, pp. 930–943, 2008.
- [6] 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.
- [7] T. Mahmood, K. Lister, S. Karunsekera and E. Kazmierczak *Industry based Learning in Software Engineering Successes and Challenges Proc. Workshop on Software Engineering Education*, pp 16–21 2007.
- [8] B. Bloom, M. Englehart, E. Furst, W. Hill and D. Krathwohl: *Taxonomy of Educational Objectives: The Classification of Educational Goals. Handbook I: Cognitive Domain*, Longmans, 1956.
- [9] 寺沢 弘忠：“管理者のための OJT の手引” 日本経済新聞社, 1999.
- [10] 後藤隼式，本田晋也，長尾卓哉，高田広章：“トレースログ可視化ツールの開発” 情報処理学会研究報告 No.2009-EMB-012-13 2009
- [11] NPO 法人 TOPPERS プロジェクト：“トレースログ可視化ツールの早期リリースについて” <http://www.toppers.jp/press/release-0904-3.pdf>
- [12] 大須賀俊憲，小林隆志，間瀬順一，渥美紀寿，山本晋一郎，鈴木延保，阿草清滋：“CX-Checker: C 言語プログラムのためのカスタマイズ可能なコーディングチェッカ” ソフトウェアエンジニアリングシンポジウム 2009 (採録決定)