

実システム開発を通じた社会連携型PBLの提案と実践

金田 重郎[†] 井上 明^{††}

[†] 同志社大学大学院・工学研究科・情報工学専攻 〒610-0321 京田辺市多々羅都谷

^{††} 甲南大学・情報教育研究センター 〒658-8501 神戸市東灘区岡本

E-mail: skaneda@mail.doshisha.ac.jp, inoue-a@konan-u.ac.jp

あらまし 著者らは、2000年頃より、学生のみで実業務に利用する情報システムの設計・開発・導入・保守を行う「実社会連携型PBL」の提案と実践を行ってきた。その詳細は、情報処理学会・情報システムと社会環境研究会(2007年3月)、及び、情報処理学会・論文誌(2008年2月)において報告している。本稿では、著者らが進めてきた実社会連携型PBLについて、概要を紹介する。また、上記報告以降の取り組みである、オブジェクト指向分析・設計を中心とするPBLについても簡単に紹介する。そこでは、MASPアソシエーションによって提案されている概念データモデリングと責務駆動設計を活用している。CDMが「オブジェクト指向的な考え方の入門」として効果的と考えるためである。キーワード PBL, 情報システム, 実社会連携, 開発, 導入, 保守, オブジェクト指向

A PBL Approach using Real World Application Development

Shigeo KANEDA[†] and Akira INOUE^{††}

[†] The Graduate School of Engineering, Doshisha University, Kyoto 610-0321 Japan

^{††} Education and Research Center for Information Science, Konan University, Kobe 658-8501 Japan

E-mail: skaneda@mail.doshisha.ac.jp, inoue-a@konan-u.ac.jp

Abstract The authors started the real world system development PBL approach in 2000. This paper summarizes the outline in the PBL approach. Especially, OOA (Object-oriented Analysis) and OOD (Object-oriented Design) are the main target of the current PBL projects. The PBL process employs CDM (Concept Data Modeling) proposed by MASP association and RDD (Responsibility-Driven Design) proposed by Rebecca Wirfs-Brock. The proposed approach is a good “School” for acquisition of the knowledge and expertization of OOA and OOD.

Key words PBL(Project Based Learning), Information System, OOA, OOD, System Development

1. はじめに

著者らは2000年頃より、「学生のみで、実社会における情報システムを開発・導入・維持管理する」形の、情報システム開発教育PBLを行ってきた。ジーン・レイヴ、エティエンヌ・ウェンガー著「状況に埋め込まれた学習—正統的周辺参加—[1]」に触れるチャンスがあり、深い感銘を受けた。そして、その正統的周辺参加理論をバックグラウンドとして、実践を重ねてきたものである。業務改革への信念を持ったドメインの専門家とプロのICT技術者(SE)とが作り出す共同化コミュニティこそが、そして、それだけが、学生を真に育て得ると信じたからである。もとより、正統的周辺参加が1年間の活動で達成できるはずもなく、4年生時点で初心者として参画した学部生は、そのまま修士課程に進み、深い形でプロジェクトに参加して、後輩を指導する。そのプロセスで学生は成長してゆく。以下、

その概要について述べる。(注1)。

2. 教育の対象者・教育目標

2.1 対象者

本PBLでは、企業・自治体の実務に長期間利用される情報システムを、学生が独力で開発・導入・運用する。従って、学部4年生からスタートするが、最初から、システムの中核部分の設計までを担当することは無理がある。当初は周辺部分を担当し、翌年以降、修士課程に進んで、自らは中核的な決定に参与しつつ、後輩たちと情報システムを開発する形で共同体コミュニティを形成する。実システムを開発して実業務に適用して、しかも長期間運用する実社会連携PBLにおける学生・院生の

(注1): なお、実社会連携型PBLについては、著者らは、すでに2007年3月に本研究会で報告を行っている。本稿の内容は、当該報告と重複している部分が多いことをご了解願いたい。

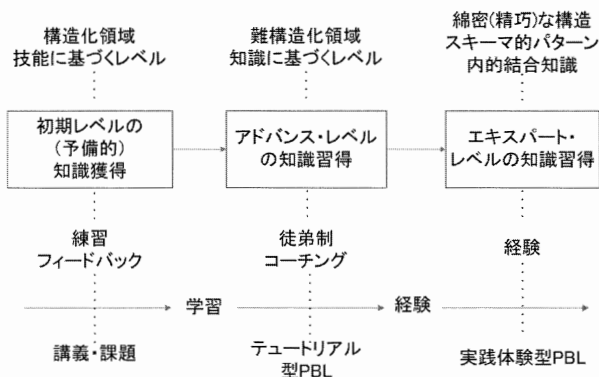


図1 ジョナッセンによる知識習得の3段階モデル (一部筆者追加)

負担は軽いものではない。この負担と学問的意義を要求される卒業研究・修士論文とを如何に両立させるかは、ひとつの課題となる。

2.2 教育の目標

周知のように、PBLは、医学教育・看護教育において始められた。医学教育・看護教育のPBLテキストを参照しても、いずれも、「テクニックを学ぶべき」とは言っていない点に著者らは着目してきた。すなわち、学生が、ICTが使われている実社会の状況に踏み込み、そこでの課題とともに活動する必要があると認識してきた。そのプロセスの中で、ICTプロフェッショナルとして日々成長するために必要な全人的姿勢を身につけることこそが、PBLの基本的目標であるとの仮説に立脚している。

PBLの目的が、対象への全人格的投与と、それによって形成される全人格的發展にあるとするなら、学内での「模擬訓練」には自ずと限界がある。学生自らが第一人称で責任を引き受け、現実社会に出て要求を分析し、それを情報システムとして具現化し、そして、学生自らによって実運用するしかない。その経験から学生は「システムを構築するということが如何なるものか」を自分のこととして初めて理解できるのである。このアプローチを著者らは2000年度より採用してきた。PBLカリキュラム開始より8年が経過し、一定のスタイルが確立されたと思われる。以下、この実社会連携型PBLカリキュラムの概要と課題・今後の方向性について述べる。

3. 実社会連携型PBLの概要

3.1 全人格的教育の必要性

ICT分野を取り巻く環境は、ドッグイヤーといわれるほどに技術進化が激しい。学生時代に学んだ知識が、実社会に出た時にはすでに陳腐化することも多い。その一方で、学生が4年間あるいは6年間という時間で、ICTに関するすべての知識を理解することは不可能である。

したがって、ICT技術者の本質的な能力とは、コンピュータサイエンスやソフトウェアエンジニアリングなどの技術的知識に加え、時代や環境が変わっても、自分で課題を見つけ出し、足りない知識を身につけ、最善と思われる方法で課題を解決し

ていく全人的な力が必要と考える。医学に例えると、解剖学や生理学といった基礎的知識を習得するだけでなく、患者の様子や症状から原因を探り、その時代で適用可能な最新かつ最適な治療を実施する能力に等しい。即ち、我々が最終的に目指しているPBLカリキュラムとは、定型業務や請負開発をこなすソフトウェア技術者の大量育成ではなく、GoogleやYahooといった新たな社会的価値観を提言できるソフトウェア技術者の育成カリキュラムである。そのためには、ドキュメントの書き方やシステム開発手順を経験させるだけでなく、学びのおもしろさや、何よりICTが与える社会的インパクトやそれが実現するプロセスを「体感」させる必要がある。

3.2 正統的周辺参加

著者らが啓発を受けて来たのは、ジーン・レイヴ、エティエンヌ・ウエンガー著「状況に埋め込まれた学習—正統的周辺参加—」(産業図書, 1993年)である。構成主義学習理論のひとつとも考えられる正統的周辺参加理論では、「共同体」への周縁的な参加から、学習者が徐々に知識を獲得していくプロセスを重視する。つまり、共同体への参加によって、学習の意味づけがなされ、知識が獲得できるとする。現在、ICT教育PBLとして実践されているもの多くは、「学内において教師や研修担当者がある事例のシナリオやテーマを提示し、学生がその内容に沿って、詳細設計から、プログラミング、テストといったシステム開発の一連のプロセスを経験する、その成果の社会的な実利用は一部」というモデルが多い。図1には、構成主義学習理論のジョナッセンのモデルに基づいてPBLを分類したものである。従来型のPBLは、基本的にはテュートリアル型PBLに該当する。これらのモデルと、本稿における社会連携型PBLの違いは以下の点である。

- 社会的実践から得られる責任感、目的意識が異なる。即ち、全人格的投与を要求し、それによる成長を期待する。従前のICT教育PBLは、アドバンス・レベルの知識習得であり、本稿の社会連携型PBLはエキスパート・レベルの知識獲得である。
- 「本物の社会」に参加することで、ICT技術者に必要な、コミュニケーション能力、観察力、表現力などの育成も視野に

システム名称	システム概要	連携先	担当学生・院生	ドメイン専門家	開発年度	主な開発環境	稼働状況	システムの品質
イベント情報公開システム	京都のイベント情報を公開するWebページを自動生成。	M新聞社支局	修士課程院生	新聞記者	2000	Xi(Baykita)&Java Script	当該支局で3年間稼働	数回のバグ修正のみ
発達相談プログラムシステム	幼児教育分野における臨床心理士の活動を支援。	A保育園	学部学生・修士課程院生	臨床心理士	2002	Nucleus, MySQL, PHP	A保育園で2年間稼働	(同上)
広報情報公開システム	自治体の広報情報を統一的に管理して、当該自治体のWebサイトを自動生成。	自治体広報課	学部学生・修士課程院生	自治体広報課職員	2004	Xi(Baykita)&Java Script	全機能がはいが一部機能は3年間稼働	(同上)
災害情報共有システム	C土木事務所管内における道路・河川沿い災害状態を一括して管理。	自治体C土木事務所	学部学生・修士課程院生	土木事務所職員	2005	PHP&Smarty	当該土木事務所で2年間以上稼働。ただし、非常時システムのため、訓練稼働が主であり、暴風雨災害時に災害対応発令に利用したのは1回のみ。	災害時のトラフィックに耐えられないことがわかり、ロードバランスをいれた大規模なシステムに全面更改を1回実施。7台のサーバで運用。
幼児の発達記録システム	保育園における幼児の発達を保育者が記録し、グラフなどを作成して、保育にフィードバック。	A保育園, P保育園	学部学生・修士課程院生	幼児教育専門家	2006	Ruby on Rails	P保育園において長期運用中(現在2年目)	統計機能などの機能追加を続行中
道路管理支援システム	Y土木事務所管内における道路管理業務をWeb-GISで一括管理。	自治体Y土木事務所	学部学生・修士課程院生	土木事務所職員	2006	Ruby on Rails	Y土木事務所にて稼働中(現在2年目)	データマイニング等などの機能を追加中

表 1 PBLにより開発された実用システムから

入れることが可能である。

つまり、社会連携型PBLでは、教師が「何を学ばせたいか」ではなく、学習者自身が「何を学ぶ必要があるか」に気付くことが重要なファクターとなる。著者らは社会連携型PBL方法論の開発を目指して、2000年度より、幾つかのシステムを開発してきた。開発したシステムの一部を表1に示す。システム規模は、約2万行から4万行である。これらの経験を経て、現在は、以下のような社会連携型PBLのスタイルが確立されている。

3.3 社会連携型PBLのプロセス

本稿で示す社会連携型PBLカリキュラムの教育プロセスの概要を図2に示す。また、表2には、教育目標・内容・計画・学習活動・評価など、カリキュラムの概要をまとめて示した。全ての活動は、PBLの学習理論のプロセスにしたがって進めている。

本学習モデルは、バロウズのPBL学習モデルに準拠している(B. マジェンダ, 竹尾恵子「PBLのすすめ—教えられる学習から自ら解決する学習へ」, 学習研究社(2004))。バロウズは、PBLの学習プロセスを「Problem Posed」「Identify what we need to know」「Learn it」「Apply it」「Evaluate results」と定義している。意図的に、学生へは「提案書の作成」「外部設計・内部設計書作成」「プログラミング」「テスト」といったウォーターフォール型の作業手順は明示していない。

ある特定の開発手法を、教師から示される手順通りに実施するのであれば、PBLではなく通常の演習形式でも可能である。重要な点は、ウォーターフォールなどの作業手順はあくまでも「システム開発プロセス」であり、「学生を教育するためのプロセス」ではない、ということである。結果的には、PBL活動の中で仕様書なども作成しているが、「仕様書ありき」ではない。「なぜドキュメントが必要なのか」「どのような内容を記述すればいいか」と言った、PBLのポイントである「システム開発を

表 2 社会連携型PBLカリキュラムの概要

	概要
対象	情報工学・工学系学生3・4回生及び、大学院生
期間	通年(1年間)
人数	1グループ4名~7名程度
担当者	教員2名(システムエンジニア経験者), 連携先組織担当者(1名~3名程度)
教育目標	ICT技術者に求められるエキスパートレベルのスキル・知識の習得
内容	学生が社会(企業・自治体など)と連携し、実システムを開発する。その活動を通じて、ICTに関するスキル、問題解決力や応用力、コミュニケーション力などを獲得する
学習活動	PBLの学習プロセスに従ってシステム開発を実施する。「問題に出会う」「解決方法を論理的に考える」「グループ作業」「自主学習」「制作」「要約」の各ステップにしたがって学習活動を進める。一般的な活動スケジュールは下記の通りである。講義は実施しないが、自主的な勉強会は適宜実施する。 <ul style="list-style-type: none"> ■前年度10月頃から3月: 連携先組織の決定(教員が実施する) ■4月: テーマ選定, ミーティング開始, 共同研究契約締結 ■5月~6月: 要求分析, 基本仕様書作成, 作業進捗確認(グループ活動) ■7月~8月: 各自担当箇所制作, 単体テスト ■9月: システム結合, テスト・修正 ■10月~2月: ユーザによる評価
評価	ICTに関するスキル(制作したシステムの機能)の評価。ユーザによるシステム評価, 問題解決力(PSI値)
教科書	特に指定した教科書は用いない。適宜、活動の中で必要と思われる書籍・参考書を紹介する

成功させるには何を学ぶ必要があるか」を学生自身に気付かせ、必要な知識を獲得させるようにした。以下に具体的なカリキュラムの進め方を説明する。

【STEP1:「問題に出会う」】 まず、「本物の課題」を提供する実践の場となり、学生のPBL活動に協力可能な組織を探す。PBLでは、少人数グループ形式で学習活動が実施される。この活動を牽引するファシリテータの態度や理解度が、グループ

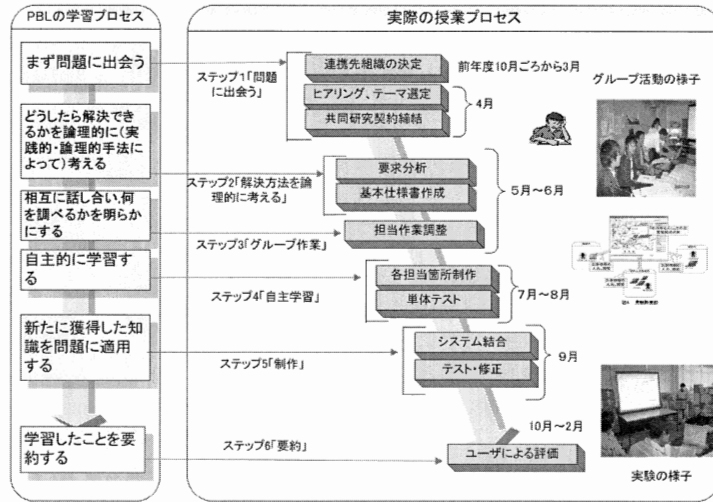


図2 社会連携型 PBL カリキュラム

全体の活動に影響を与える。つまり、ファシリテータである連携先組織（対象ドメイン）の担当者の熱意や振舞いが、システム開発に大きな影響を与える。

そのため、連携先組織を決定する重要な要素は、連携先の「現場担当者（後述する2人のファシリテータの一人となるべき方）」が、この活動に意義を感じるかである。現場担当予定者が、ICTシステム構築・導入に向かって、強い期待と熱意を持って受け入れ、自らもその活動に参加する意欲があるかどうか、が、連携先組織を決定する最大の判断ポイントとなる。

連携先決定後、教員は、連携先組織の担当者として、学生が開発するシステムの開発テーマをある程度絞り込む。例えば、「イベント情報に関するシステム」や「災害対策システム」などである。これは、1) 学生が構築するシステムの概要を、連携先組織へ明示することで、活動の協力を得やすくする、2) システム開発経験の無い学生に対し、実社会のシステムを全く制限無しで開発させるには、難易度が高すぎることへの配慮、からである。事前に教員が決める範囲は、あくまでも「開発の参考になるポイント」であり、学生には、詳細な仕様や機能は提示しない。

開発テーマ決定後、教員・連携先組織担当者・学生を含め、現場ヒアリングを実施する。打ち合わせ初期段階では、教員と連携先組織担当者同士が主に話し合う。学生には、議事録係として打ち合わせを記録する役割を与える。学生に、仕様書や提案書作成からスタートさせるのではなく、議事録を作成させることから始める理由は、「徐々に開発作業に参加させる」という目的からである。これは、いきなり古参者と同じ作業をするのではなく、新参者は、単純で失敗に対する損失の小さい責任の軽い活動からはじめる、という正統的周辺参加理論に従っている。このような実践共同体の一部に加わっていくプロセスを通じて、新参者は、自分達の置かれている状況とやるべき事柄を理解する。

また、開発初期段階では、対象ドメインの事務所などへ積極的に出向くようにする。実際の現場の雰囲気や仕事の様子から、学習者に対し「遊び」ではない責任感を感じさせる。それに加え、打ち合わせやプレゼンテーションを通じて、システム開発には他者への説得やアピールが必要不可欠であることを学ばせる狙いがある。表1に示した実践の中でも、連携相手先の幹部・組織内へのプレゼンテーションでは、イントロのみを教員が行い、あとのシステム紹介などは学生にやらせている。

さらに、上記作業と平行して、共同研究契約を締結する。共同研究契約を正式に結ぶ理由は、守秘や学生の事故に対する問題もあるが、大きな目的は、連携先の担当者の正式業務として認めてもらうことである。これによって、連携先組織担当者も一人称で活動に参加しやすくなる。

STEP2:「解決方法を論理的に考える」次に、グループ活動によるミーティングをおこない、要求分析や各自の担当箇所を決定する。基本仕様の策定には、システム全体のイメージ図とおおまかな画面遷移図を何度も更新しながら、仕様を確定していった。また、できるだけ早い段階で、動作はしないがシステムの雰囲気を確認する画面だけの「プリ・プロトタイプ」を作成する。画面イメージを早期に作成することで、メンバー全員がより早い段階でシステムのイメージがしやすくなり、作業効率のアップ、意見の食い違いを最小限に抑えるなど、開発を進めていくうえで効果的であった。

ドキュメント類は、XP (eXtreme Programming) の考え方を取り入れ、必要最小限のものにとどめる。具体的には、基本仕様書、ユーザ・インターフェース設計書（実際には、プリ・プロトタイプとして連携先との打ち合わせに利用したHTMLによる内部処理とは無関係の「紙芝居」をそのまま利用することが多い）、E-R図、DB設計書、などである。

本PBLでは、「ドキュメント作成能力」を、PBLを通じて学

生に習得して欲しい最重要スキルと位置づけていない。確かに、各種のドキュメント作成能力は重要である。しかしながら、ドキュメント作成能力は1回経験しただけでは理解できない。ドキュメント作成能力は、ジョナッセンの知識習得モデルでは、「初期レベルの知識」であり、その習得には、先に述べたように、繰り返し練習する必要があると考える。つまり、ドキュメント作成スキルは、PBL以外の他の演習で学習すべき項目であり、PBLでは、「システム開発を成功させるためには、何をどのように進めていけばよいか」といったシステム開発を進めるための論路的思考を意識させるようにしている。

【STEP3:「グループ学習」】開発作業は、グループ活動として実施する。学生は、分担してシステムの開発を行う。基本的には、マイクロソフトの同期安定化プロセスを縮小したような開発形態となる。学生は、自分が担当になった機能のプログラミングを行う。ただ、全く個人で活動するのではない。他の機能との連携が必要になるために、頻繁に他のメンバーとの調整が求められる。また、学生同士だけでなく、学外者とも意見交換をしなければならない。これらのコミュニケーション活動を、システム開発作業に組み込むことによって、ヒューマン・スキルの向上を図るようにしている。

以上のようなグループ活動を円滑に進めるために、開発作業用のWikiを用いた。Wikiは、意見交換や作業の進捗管理など、情報蓄積・共有を進めるうえで有効なツールであった。また、複数人での開発を進めるためにバージョン管理システムも利用している。グループ活動が基本となるPBLの実践には、このような共同作業用のシステム環境は必須である。

【STEP4:「自主学習」】ICT教育PBLでの自主学習とは、2つの側面がある。一つが、本やネット、友人からプログラミングやサーバ構築など、自分に必要な知識を調べて獲得する能力である。二つ目が、誰かの指示を待つのではなく、グループがひとつの自立的組織として形成されていく活動である。学生の作業担当は、技術的ファシリテータである教員が指示する。しかし、実際に作業が進むにつれて、ファシリテータの指示が無くとも学生間で、自主的に作業の調整を行うようになっていった。例えば、技術的スキルの高い学生が、他のメンバーへ作業の指示やサポートを自主的に行うようになった。

活動が進むにつれて、「教員（ファシリテータ）」「学生」という2つの立場ではなく、その中間的な「サブファシリテータ」ともいえる学生が育つ。これは、できるだけ自分達で解決しようとする意欲、責任感が高まっていた結果と思われる。学生の活動を観察すると、技術的スキルの習得に関して、一人で本やネットで調べることもあるが、他のメンバーへの質問や助け合いが頻繁に見られる。つまり、相互活動によって、すでに多くの知識を獲得している学生は、他者へ教えることによって知識の再確認と定着が行える。一方、教えられる側の学生は、新たな知識を獲得する。

このようにPBLを通じて、最新技術を常に身につけ、最適解を作り出す力といった、ICT技術者に求められる、自己獲得型の知識習得や自立的組織の形成を経験させるようにしている。

【STEP5:「制作（獲得した知識の問題）」】システム構築は、プ

ロトタイプを作成し、ユーザへのレビューを行い、機能の追加・修正を実施する。これはプロトタイプ開発手法とほぼ同じである。システム制作段階で重要なポイントは、使用する開発言語やアーキテクチャといった技術的な問題ではなく、「ファシリテータの振る舞い」である。学生は、プロトタイプを作成し、ユーザへレビューする。その意見や要望を次のリリースへ反映する。しかし、システム開発初心者は、作業時間の見積りや技術的難易度を判断する能力が低いために、ユーザからの要望を全て実装しようとする傾向が見られる。その結果、開発工数が膨大になる危険性が高い。事実、我々が過去に実施したPBLのいくつかでは、技術的ファシリテータのアドバイスが少ない事例において、予定していた開発期間が大幅に延びる危険性があった。

したがって、技術的ファシリテータが、時間的制約と技術的難易度から、実装すべき機能や要望の優先度や範囲を判断する。また、対象ドメイン側の担当者も、数ある要求事項の中から、本質的に必要な機能はどれかを選定し、学生のシステム構築活動を利用者側の立場からサポートする役割も担ってもらい。このように教員や対象ドメイン担当者は、プロジェクト・マネージャーとして活動をサポートする。学生は、このようなファシリテータの振る舞いから、システム開発に求められる「決断」や「判断のポイント」を理解する。

【STEP6:「要約」】完成したシステムは、実際に一定期間ユーザに利用してもらう。ただ、この時点では、開発者である学生が「完成した」と考えているだけであり、実際に利用すると、各種のバグと改善点が明らかになる。事実上の運用テストである。社会連携型PBLでは、これらのユーザ利用による評価を重視している。その理由は、システム開発の基本的原則である「システム開発が成功したかどうかは開発者ではなくユーザが判断する」ことを理解させるためである。

ユーザの利用によって、開発の問題・修正、認識の違いが明らかになる。つまり、「自分達に足りなかった点」が明確になり、学習してきた知識の獲得レベルや、今後どのようなスキルを身につける必要があるかが認識できる。自分が担当した機能以外のところを学生同士で相互評価させることも可能である。しかし、そのような「身内同士」の評価では、学習意欲が高まらない。

したがって、これまでの実践では、可能な限り、開発に参加したメンバー以外での評価を取るようにした。例えば、ある自治体と開発した災害情報共有システムでは、開発メンバーである土木事務所だけでなく、周辺市町村の防災担当者を含めた広範囲な実験と評価をおこなった。この「自分に足りない知識を認識する」ことに気付かせ、他の科目や授業への意識付けとすることが、社会連携型PBLの目的のひとつでもある。

4. 学習効果

4.1 問題解決能力の評価

上記のPBLによって、ソフトウェア技術者に求められる「問題解決力」が増加しているかどうか問題となる。そこで、問題解決力の評価として、医学・看護学でも用いられ、その評価

の妥当性が認められている PSI (Problem Solving Inventory : 問題解決力尺度) を用いた。PSI は問題指向的認知ならびに、特異的な問題解決技能に焦点を合わせた 35 項目からなる自記式の調査項目である。PSI では、設問に対し 6 段階 (回答 1=よくあてはまる, 回答 2=あてはまる, 回答 3=少しあてはまる, 回答 4=あまりあてはまらない, 回答 5=あてはまらない, 回答 6=まったくあてはまらない) で回答をおこない、それぞれの項目の回答 (1-6) を足したものが PSI 値となる。得点が低いほど問題解決的思考・自信を持っていると評価される。ただし、いくつかの項目は否定的に述べられており、その項目では採点を逆にする。

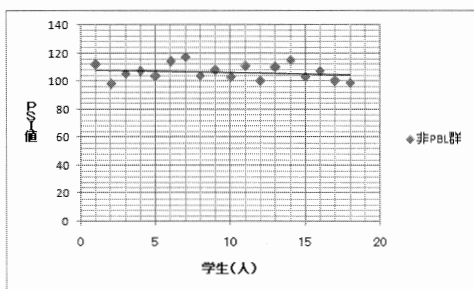


図 3 PSI 測定結果-非 PBL 群

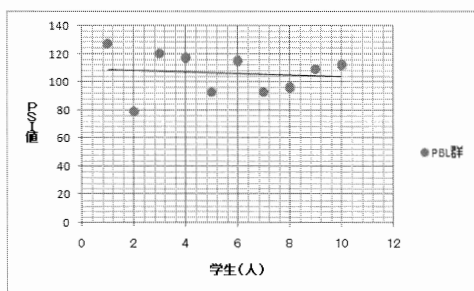


図 4 PSI 測定結果-PBL 群

評価手法と対象は以下のとおりである。

対象: 社会連携型 PBL を経験した学習者 (「PBL 群」と以下呼ぶ) 10 名, PBL を経験していない学習者 (「非 PBL 群」と以下呼ぶ) 18 名, (卒業生・修了生がいるため, 全ての PBL 参加者が回答しているわけではない。)

調査期間: 2007 年 1 月 24 日～1 月 27 日

調査方法: Web アンケートシステムを用いた電子的調査。属性情報は性別と学年のみの匿名回答

図 3, 図 4 に, PBL 群, 非 PBL 群の PSI 測定結果を示す。PSI 値は値が小さいほど, 問題解決能力が高いとされる。PSI の平均値は, PBL 群が非 PBL 群に対し, 低い値となった。ただし, t 検定の結果では, 有意な差は見られなかった。しかし, PSI の結果と学生への観察評価から, いくつかの点で社会連携型 PBL の特徴と課題が明らかになった。

- PBL 群と非 PBL 群を比較すると, PSI 値のばらつきに

違いがある。非 PBL 群の PSI 値はばらつきが小さいのに対して, PBL 群のばらつき大きい。一般的に, グループ学習は, 伸びる学生は (坐学に比して) 伸びる割合が大きい, 他者に依存するフリー・ライダーとなって, 十分に学習効果が上がらない学習者が現れるといわれる。社会連携型 PBL においても同様の現象が出ていると考えられる。

- PBL 群の中で, 学部生と, 活動の中心となった大学院修士課程学生との PSI 値を比較した。その結果, 修士課程の学生は, 学部学生に比べ平均的に PSI 値が低いことがわかった (つまり, 問題解決に自信を持っている)。学部学生では PBL を経験したからといって, すぐに問題解決力が養成されるわけではないと思われる (学部学生で 1 名, 極めて優れた PSI 値を示して。この学生はアルバイトとして, 既に何万ステップも実用プログラムを書いている学生ではないと思われるが, 匿名なので, 定かではない)。修士学生は, 何回か PBL を経験している。つまり, 1 度の PBL 経験ではなく, PBL の繰り返し, 優れた PSI 値を生むように思われる。

- PSI 調査で, 自由記述欄を設けた。ここでは, 大学全体の講義のあり方について聞いた。PBL 群のメンバーは, 「理論と実践のバランスのとれた講義を期待する」「講義のレベルと自分のレベルに差があった時には, どうしてよいのか分からなかった」といった, 社会性が感じられる記述が多い。これに対して, 非 PBL 群のメンバーの中には, 「講義をもっと面白くして欲しい」「今日は面白い講義あるかな」といった, 稚拙な表現が散見される。PBL 群のほうが一般的に謙虚で論理的記述である。PBL では, 学習者が主体的に知識を獲得する経験を積む。また, 他者との意見交換や説得・表現活動が常に求められる。このような知識発見型学習の経験が, 客観的分析力・表現力を高めるものと思われる。つまり, PBL が一種の社会性を養成しているように感じられる。すなわち, この社会連携型 PBL は, 大学と自治体が連携して実施する「インターンシップ」であるとも思える。

- 学生へのヒアリング並びに観察評価から, 例えば, 社会連携型 PBL を経験した学生の中には, 「作っているだけで, 深い知識が身についたという感じがしない」という感想をもらすケースがあることが分かった。一見, PBL に否定的な意見のように思える。しかし, この感想は, 言われたことを部分的に担当しただけで終わった学生からは聞かれない。むしろ, 自ら積極的に PBL に取り組み, 作成プログラム量も多く, 打ち合わせでも頻繁に発言した学生に限定される。しかも, この問題意識は, プロジェクトの初期段階では発生しない。開発の最終段階で, 細かいバグ・フィックスや手直しの段階でこのような要望が生じてくる。学生が言う「深い知識」とは, 例えば, 「もったときちんとオブジェクト指向について勉強したかった」「デザインパターンについて系統的に学びたい」といったものであった。これは, 1) PBL でシステム開発を行う, 2) 分からないながらも作成する, 3) 作業が進むにつれて本来は身につけているはずの基礎知識やスキルが, 獲得できていないことに気がつき始める, という状態と思われる。つまり, 「身についた感じがしない」ではなく, 知識の内的結合性が高まった結果, 「身についたいな

かったことが PBL によって気がついた」とも言える。これまで自分達が学習してきた知識の不十分さを認識し、新たな知識獲得意欲が生まれたと考えられる。

以上のように、PSI 調査結果とこれまでの実践から、1)PBL によって学習者全員が等しく PSI 値が高まるのではなくフリーライダーへの対応が必要、2)PBL は 1 回だけではなく数回の経験が望ましい、3)PBL とは別に要素技術を学習できる環境を整備する、ことが必要であることが分かった。また、本 PBL で経験する知識獲得型学習や他者とのコミュニケーション活動によって、客観的分析力や表現力、知識の内的結合力を獲得するものと思われる。

4.2 中間総括

以上の経験から、以下のようなことを中間総括として感じた。

- 3 万行を越えるようなプログラムを開発してしまうと、事実上、連携先組織への導入が困難となる。詳細な設計書、機能仕様書、あるいは運用マニュアルまでとはとても作成できないために、担当者以外は手を入れることができず、保守管理能力に限界があるためである。大学の規模では PBL で実践できる範囲に限界があることを理解しておく必要がある。

- 開発方法論の訓練として、ラショナル統一プロセスのようにユースケースを書いて、名詞・動詞を取り出すことも何度か試みたが、本システム規模では、本当のところ、それらの手法がありがたいという実感を得にくいように感じられる。また、受託開発的な PBL のみでは、アプリケーションのパッケージ化のような視点が弱くなる恐れもある。処理系屋としての技術や視点もあわせて指導する必要があると感じられた。

- 社会連携型 PBL を実践することで、優秀な学生ほど、高度な知識への渴望が強くなる。デザインボタンであれ、オブジェクト指向であれ、コンパイラの設計であれ、「きちんとした理論的なことを分かりたい」「手法のみでなく、その背後にある思想が知りたい」との希望が強くなる。これら要望に応え得る授業や研究の実践が、大学全体のカリキュラムに求められる。PBL さえやっていれば、後は多少手抜きでも学生は育つ、ということはあるまい。むしろ、「本当に将来の業務でも意味のある、基礎となり得る講義内容」が問われる。

- 最後に、PBL は単なる職業訓練ではないことを理解しておかなければならない。PBL の学習プロセスでは、あいまいな問題をいかに一般化・抽象化できるかが問われる。それがエキスパート・レベルのスキルであろう。今後は、「PBL 2.0」とも言える、企業などの受託開発から踏み出し、個人や社会に求められる全く新しいサービスを創出できる人材を育成するような PBL をさらに進めていく必要があると考えている。

- 卒業研究、修士論文との関係も重大な課題である。これらはなんらかの学問的な切り込みを必要とするからである。現在のところ、PBL で得た問題意識を利用して、ソフトウェア開発の方法論にするなどの形で、一定量のジャーナル論文文化に「やっと成功し始めた」ところである [15][14][13][12]。

5. オブジェクト指向の深い理解を目指して

以上、2007 年度までの経験を踏まえて、2008 年度からは新

しい形の実社会連携型の PBL に着手している。PBL に着手している。この試みは、システム・エンジニア育成 PBL における、学習項目の明確化である。本稿の締めくくりとして、その一端を紹介したい。著者らは、PBL に、新たな目的を設定した、それは、「オブジェクト指向分析 (OOA)、オブジェクト指向設計 (OOD) の利用と理解」である。具体的には以下の 2 つのアプローチを併用している。

5.1 概念データモデリングの採用

MASP アソシエーション [4] が提唱する「概念データモデリング (以下、「CDM」と称する) [5][6][19] は、中村善太郎の提唱する「かなめの『もの』とかなめの『こと』」[10]を取り出す手法であり、KDDI、JFE スチールなどの情報システム統合に大きな寄与があったと言われる [7][8][9]。CDM では、静的モデル、動的モデル、組織間連携図などの図を用いて、対象業務の中にある「情報の流れ」をデータモデルとして「写し取る」。著者らも、PBL のプロセスとして、自治体担当者と一緒に、モデリングを行うことを 2007 年度より開始している。実際に自治体担当者やってみると、学生にとっても、自治体担当者にとっても、現状のビジネスの形を明確に意識することが可能となる。道路管理システム、税務共同化などの中で、分析手法として用いて来た [13][12][20]。

学生が実際に CDM を経験する中では課題も見える。それは、必ずしもすべての学生が概念的なモデリングに対して、「親しみ」を感じるとは限らないことである。しかし、少なくとも、CDM は、「オブジェクト指向的な捉え方」のよい教育の場になると感じている。例えば、料理の例で言えば「レシピ」というものを初学者は (現実社会にレシピという「もの」が存在するので) オブジェクトと捉えがちである。しかし、レシピは明らかに View (情報) であって、その背後には、「料理」というオブジェクトが存在する。レシピは、料理オブジェクトの属性値であるべきであろう。Java のコーディングから「オブジェクト」を意識させるのではなく、このように、「オブジェクト指向的な捉え方」を学習させることはオブジェクト指向のトレーニングとして意味があると感じている。

しかしながら、上流工程の「モデリング段階のみの PBL」には一定の限界がある。本来、技術者の醍醐味は「自らが判断した技術的選択に、実社会で責任を取る」ことである。実社会連携 PBL は、1) モデリング段階から顧客と連携し、顧客の問題解決能力を高め、2) そして、実際にシステムを開発・導入・運用する形であるべきと思われる。それは、初期コンサルティング段階から、モデリング手法などを用いた顧客教育、ソリューション導出、情報システム開発・導入・維持管理までの一連のプロセスである。1 年間、モデリングの PBL を行った結果、「やはり作らないといけない」との思いも強くした。従って、オブジェクト指向分析のみではなく、オブジェクト指向設計も取り入れる必要性を痛感した。

5.2 責務駆動設計の採用

そこで、2008 年度からは、PBL のシステム開発において、オブジェクト指向設計技術の理解と習得を目的のひとつとして位置づけた。著者らは、責務駆動設計 [11] を採用している。

UML(クラス図, シーケンス図など)とCRCカードなどを書か
せて, 設計を行っている. 本来, オブジェクト指向分析と, オ
ブジェクト指向設計では, 同じオブジェクトと言っても, 捉え
方は異なる. しかし, 責務駆動設計には, 「かなめの『こと』」
は, 従来の MVC モデルとはことなり, むしろ『M(モデル)』
側に実装する視点があると感じられ, CDM との相性もよいと
感じたからである.

それでもなお, 大学という学問の府である以上, 忘れてはな
らないことがあると考えたい. それは, 「一般性」「新規性」で
ある. 実社会連携 PBL を, 単なる「請負開発」とすることは
できない. そのプロセスを, 「処理系屋の目」「ソフトウェア工
学屋の目」で眺めながら, 実社会において本当に「役に立つ」
一般的方法論にまで高め, ジャーナル論文文化まで学生とともに
進めること. それができて初めて, 完全な意味での実社会連携
PBL と言えるのかも知れない. それが現状の著者らにできて
いるとは思えない. しかし, その試みは続けて行きたいと考え
ている^(注2).

6. おわりに

本稿では, ICT 専門教育のための PBL として, 実際に社
会と連携し実システムを開発する社会連携型 PBL の実施結果
について報告した. 本実践は, PBL が持つ「学ぶ意欲の喚起」
「問題解決力の習得」「知識の統合化」といった特徴に着目し, 学
部教育の4年間, そして修士教育の2年間に, ICT プロフェッ
ショナルとして求められる本質的能力をいかに獲得させるかを
目的としている. これまで, 多数の社会連携型 PBL を実施し
た. これらの PBL では, PBL の学習プロセスである, 「問題に
出会う」「解決方法を論理的に考える」「グループ学習」「自主
学習」「制作」「要約」の各 STEP に合致するように進めた. そ
して, 本稿では紙幅の関係で説明を省略したが, ICT 専門家育
成のための PBL には, 以下のプラクティスが必要であると考え
ている. 1) 技術系(SE)と応用ドメイン分野の2人のファシリ
テータ, 2) ドメイン専門家に負担をかけないボランティア方
式, 3) PBL を核としたカリキュラム編成, 4) 時間的余裕の確
保. これらの項目を考慮して PBL を実施すべきと考えている.

ICT プロフェッショナルをいかに育成するか. この課題は非
常に大きい. 繰り返しになるが, ICT の進化は今後も急速に拡
大していく. ある時点でのピンポイント的な知識獲得ではなく,
課題解決力や自己学習能力の育成がソフトウェア技術者に求め
られている. 大学における ICT 教育の歴史は, ようやく 40 年
ぐらいにさしかかったところである. 一方, 長い歴史を有する
医学教育から生まれた PBL には, 「何をいかに学ぶか」が集約
されている. PBL は, 手順の理解ではなく, プロフェッショナル
としての能力を育成するための教育手法として位置づけるこ
とが重要であると考えている.

(注2): 処理系屋としての視点を持つためには, 対象システムが一定量以上の規
模を持っている必要がある. このような規模になると, 当然, PBL により大学
が開発することは難しい. 結果として, 巨大システムの開発検討に参加する中
から, 処理系屋の目でミドルウェアを抽出し, これを小規模なレベルで検証するア
プローチを探らざるを得ないと考えている.

文 献

- [1] ジーン・レイヴ, エディエンヌ・ウェンガー(著), 佐伯胖(訳), 「状況に埋め込まれた学習—正統的周辺参加」, 産業図書, 1993 年 11 月
- [2] 井上明, 金田重郎, 「実システム開発を通じた社会連携型 PBL の提案と実践」, 情報処理学会, 論文誌, Vol.49, No.2, pp.930-943, 2007 年 2 月
- [3] 井上明, 金田重郎, 「実システム開発を通じた社会連携型 PBL の提案と実践」, 情報処理学会, 情報システムと社会環境研究会, 情報処理学会研究報告・2007-IS-99, pp.115-122, 2007 年 3 月
- [4] 特定非営利法人 技術データ管理支援協会 (MASP), <http://www.masp-assoc.org/>
- [5] 手島歩三, 「概念データモデル設計によるソフトウェアのダウンサイジング」, 日本能率協会マネジメントセンター, 1994 年 11 月.
- [6] 手島歩三 「ビジネス情報システム工学概説—概念データモデリングに基づく情報システム構築と運営—」, 技術データ管理支援協会 (MASP)・内部資料 (非売品), 2006.
- [7] 経営情報学会 システム統合特設研究部会 [編], 「成功に導くシステム統合の論点」日科技連, 2005 年 10 月.
- [8] 前掲書, p.121, 「KDDI の事例-概念データモデルによるシステム統合-
- [9] 杉原明, 白崎俊行, 森弘之, 「J-Smile を支える IT イノベーション (メソドロジ) —柔軟なシステム構造, 短工期開発を実現する設計開発方法—」, JFE 技報, No.14, pp.25-28, 2006 年 11 月.
- [10] 中村善太郎, 「もの・こと分析で成功するシンプルな仕事の構想法」, 日刊工業新聞社, 2003 年 11 月.
- [11] レベッカ・ワーフスブラック / アラン・マクキーン(著), 藤井拓(監訳), 「オブジェクトデザイン ロール, 責務, コラボレーションによる設計技法」, 翔泳社, 2007 年 9 月.
- [12] 吉澤憲治, 吉田和正, 井上明, 芳賀博英, 金田重郎, 「論理思考プロセス (TOC) と概念データモデリング (CDM) に基づく業務分析手法の提案」, 情報処理学会論文誌, 2009 年 2 月 (発行予定).
- [13] 吉田和正, 吉澤憲治, 芳賀博英, 金田重郎, 「概念データモデリングとプロブレムフレームを用いた情報システム実装手法」, 情報処理学会論文誌, 2009 年 2 月 (発行予定).
- [14] 白井由希子, 糠野亜紀, 新谷公明, 井上明, 芳賀博英, 金田重郎, 「「気になる子」の保育を支援するための巡回相談支援システムの提案と評価」, 情報処理学会論文誌, 2009 年 2 月 (発行予定).
- [15] 「保育者の保育傾向を抽出できる発達記録システムの提案」仁木賢治, 糠野亜紀, 新谷公明, 芳賀博英, 金田重郎, 情報処理学会論文誌, 2009 年 2 月 (発行予定).
- [16] 古畑貴志, 吉澤憲治, 小野孝司, 寺田守正, 吉田和正, 矢野高一, 中村喜輝, 佐野嘉紀, 井上明, 金田重郎, 「Web-GIS を用いた道路管理システム「京都市守くん」の開発」, 情報処理学会, 第 69 回全国大会, 3ZA-9, 3 月, 2007 年.
- [17] 吉澤憲治, 古畑貴志, 小野孝司, 寺田守正, 吉田和正, 矢野高一, 中村喜輝, 佐野嘉紀, 井上明, 金田重郎, 「Web-GIS を用いた道路管理業務支援システム「京都市守くん」の開発」情報処理学会研究報告・2007-IS-99, pp.39-44, 3 月, 2007 年.
- [18] 寺田守正, 「道路管理を科学する」, 国土交通省近畿地方整備局管内技術研究発表会, 2003 年.
- [19] 金田重郎, 吉田和正, 吉澤憲治, 「概念データモデリングへの意味論からの接近」, 情報システムと社会環境研究会, 2009 年 3 月発表予定.
- [20] 矢野高一, 吉田和正, 吉澤憲治, 上仲良幸, 川上拓也, 金田重郎, 「二層構造アーキテクチャを有する地方税共同化システムに関する基礎的検討」, 情報システムと社会環境研究会, 2009 年 3 月発表予定.
- [21] 古畑貴志, 吉澤憲治, 小野孝司, 寺田守正, 吉田和正, 矢野高一, 中村喜輝, 佐野嘉紀, 井上明, 金田重郎, 「Web-GIS を用いた道路管理システム「京都市守くん」の開発」, 情報処理学会, 第 69 回全国大会, 3ZA-9, 3 月, 2007 年.
- [22] 吉澤憲治, 古畑貴志, 小野孝司, 寺田守正, 吉田和正, 矢野高一, 中村喜輝, 佐野嘉紀, 井上明, 金田重郎, 「Web-GIS を用いた道路管理業務支援システム「京都市守くん」の開発