

ソフトウェア工学科の誕生: ソフトウェア工学によるソフトウェア工学教育カリキュラムの開発

青山 幹雄† 蜂巢 吉成† 沢田 篤史† 野呂 昌満†

†南山大学 数理情報学部 情報通信学科

南山大学では、日本で最初のソフトウェア工学科を 2009 年 4 月に開設する。本稿では、カリキュラム開発とソフトウェア開発の類似性に着目し、学科設置の過程で検討した、ソフトウェア工学科のモデル、地域の要求を考慮した人材像、J07 に準拠したカリキュラム設計、などを報告する。この中で、カリキュラム設計にソフトウェア工学を応用する方法を提案する。

The Birth of Software Engineering Department: Development of Software Engineering Education Program Based on Software Engineering

Mikio Aoyama†, Yoshinari Hachisu†, Atsushi Sawada†, Masami Noro†

†Dept. of Information and Telecommunication Engineering, Nanzan University

This article reports on the design of the undergraduate software engineering education program at Nanzan University, to be initiated in April 2009. From the analogy between the curriculum development and software development, the authors discuss the model and techniques for the development of software engineering education curriculum, including model of software engineering department, necessary software engineering body of knowledge, and architecture of curriculum with consideration of standard software engineering curricula SE2004 and J07.

1. はじめに

南山大学数理情報学部は、2009 年 4 月に情報理工学部に改称し、あわせて、わが国で最初のソフトウェア工学科を開設する。

世界的に、ソフトウェア工学の専門技術者を育成する専門学科の必要性が長年議論されてきた[6, 10, 15, 16, 18, 19, 20, 23, 24, 27, 28, 32]。欧米諸国やアジア諸国ではソフトウェア工学科が設置され、専門教育が実施されている。教育内容についても、ソフトウェア工学カリキュラム標準 SE2004 が策定された[13, 17]。これに基づき、情報処理学会カリキュラム標準 J07 の一つとしてソフトウェア工学教育カリキュラムも策定された[14]。

しかし、わが国では、コンピュータ科学などの学科はすでに多数設置されているが、学科の新設や再編には様々な利害の調整が必要であることなどから、ソフトウェア工学科は実現されていない。

このような背景から、南山大学では、2000 年に開設した情報通信学科内のソフトウェア工学教育プログラムを発展させ、ソフトウェア工学科を開設するに至った。

本稿は、学科設置の過程で検討した、ソフトウェア工学科のモデル、地域の要求を考慮した人材像、J07 に準拠したカリキュラム設計について報告する。

特に、ソフトウェア工学教育のカリキュラム開発とソフトウェア開発との類似性に着目し、ソフトウェア工学の技術をカリキュラム設計へ応用する方法とその有効性を示す。

2. ソフトウェア工学科開設の背景

2.1 ソフトウェア工学科への期待

ソフトウェア工学科開設の背景は、次の、ニーズとシーズの両面がある。

(1) ソフトウェア工学科へのニーズ

- 1) 社会からのニーズ: 情報処理技術の進展に伴い、わが国では、長年、ソフトウェア開発の専門技術者の育成が求められてきた。特に、金融や運輸などの社会の基盤となる情報システムの障害が発生してきたことが、ソフトウェア工学の専門技術者の不足を浮き彫りにした。これに対して、情報科学、コンピュータ科学などの、CS (Computer Science)カリキュラム標準を基礎とする学科は多数設置されてきた。しかし、これらの学科ではソフトウェア工学に関する教育は限定される。SE (Software Engineering)カリキュラム標準を基礎とし、ソフトウェア工学の専門家を育成する学科の開設が望まれている。

- 2) 地域からのニーズ: 特に、南山大学が立地する中部圏は、自動車産業、航空産業をはじめとする製造業の集積地である。自動車を含む広範な産業でコンピュータ利用が急激に進んでいる。例えば、自動車の組込みソフトウェアは規模、複雑度ともに急速に増大している。地域におけるソフトウェア専門技術者育成のニーズが高まっている。

(2) ソフトウェア工学科のシーズ

- 1) ソフトウェア工学標準カリキュラムの策定: IEEE Computer Society と ACM によるカリキュラム標準 CC2001 の一つとして、ソフトウェア工学教育カリキュラム標準が策定された。あわせて、組込みシステムを対象とする CE (Computer Engineering) も策定された。これに基づき、わが国の教育制度を加味して、情報処理学会カリキュラム標準 J07 の一つとして、J07-SE, CE が策定された。
- 2) 南山大学におけるソフトウェア工学教育の実践: 2000 年 4 月に新設された南山大学 数理情報学部 情報通信学科では、ソフトウェア工学と通信工学の二領域を主たる知識領域と位置づけた。ソフトウェア工学を専門とする教員も 8 名を擁し、学科内のソフトウェア工学コースとして、ソフトウェア工学関連科目を体系化し、ソフトウェア開発技術者の育成を図ってきた。産業界の視点からも評価されてきた[21]。さらに、大学院数理情報研究科の修士課程では、2007 年 4 月に「ソフトウェア工学専修コース」を開設し、ソフトウェア工学の専門教育を実施している。

2.2 ソフトウェア工学科への学科再編

図 1 に示すように、数理情報学部の情報通信学科と情報システム数理学科を再編し、ソフトウェア工学科、システム創成学科、数理科学科の 3 学科に再編することになった。ソフトウェア工学科は、情報通信学科のソフトウェア工学コース、コンピュータ科学コースをまとめ、学科として独立させた。

学科再編にあたり、2007 年夏からタスクフォースを編成し、新学科のカリキュラムなどを検討した。

すでにソフトウェア工学コースを設置していたことから、ソフトウェア工学科への再編では、学科新設に比べカリキュラムなどの変更は少ないと予想された。しかし、学部の制約条件、他学科との整合性、JABEE との整合性、カリキュラム標準 J07 との整合性など多くの制約条件を満たす必要があったため、本稿で述べる体系的な設計方法が必要となった。

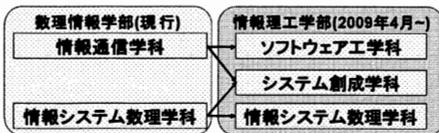


図 1 学部・学科の再編

3. ソフトウェア工学科開発の課題とアプローチ

3.1 ソフトウェア工学科開発の課題

ソフトウェア工学科を実現するためには、カリキュラムなどの教育内容に加え、その実行を担う教員などの人や設備、さらには、それを実現可能とする経営的課題、

政治的課題など、多様な課題を解決する必要がある。

これらの課題は、ソフトウェア工学教育とその実行組織であるソフトウェア工学科を一つのビジネスユニットと見なすと、情報システムにより顧客の課題を解決し、新たなビジネスを実現する活動と共通点が多い。従って、ソフトウェア工学科を実現するための様々な課題の解決にソフトウェア工学の諸技術を応用できる可能性がある[7]。

これまで、ソフトウェア工学教育の実現やその課題解決のために様々な議論があったが、工学的なアプローチやそれに基づく提案はほとんど見られない。

このような問題に対し、教育学では、カリキュラム開発として様々な技術が開発されている[5, 30]。特に、教育に情報技術を応用する諸技術が「教育工学」として体系化されている[4]。さらに、近年、工学教育の開発に工学的アプローチを試みる提案もある[5, 26]。

本稿では、ソフトウェア工学科を実現する様々な課題の中で中核的課題であるカリキュラム開発に焦点を当て、ソフトウェア工学の諸技術を応用する可能性を議論する。これは、図 2 に示すように、カリキュラムをソフトウェアと見なすことにより、カリキュラム開発とソフトウェア開発の間に多くの類似性があると考えられるからである。以降では、ソフトウェア工学を応用するアプローチをとり、ソフトウェア工学カリキュラムの開発をソフトウェア工学の枠組みで整理して示す。特に、カリキュラム開発には、現行のカリキュラムからの移行、大学の履修システムや JABEE などにより課される履修条件、学生や教員などのステークホルダー間の利害など、多様で複雑な課題がある。

なお、表 1 に示すように、実行主体が人である点でカリキュラムはソフトウェアよりビジネスプロセスと対応する点に留意すべきである。

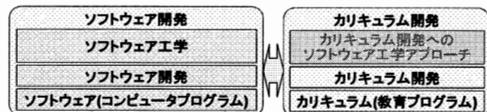


図 2 ソフトウェア開発とカリキュラム開発

表 1 ソフトウェア、ビジネスプロセス、カリキュラム

	ソフトウェア	ビジネスプロセス	カリキュラム
内容	処理過程	ビジネスの過程	教育課程
実行主体	コンピュータ	人(顧客, 遂行者)	人(学生, 教員)
出力(価値)	情報の付加価値	ビジネスの生産性, 品質向上, 期間短縮	人材の付加価値(知識, スキル)

3.2 ソフトウェア工学の視点から見たカリキュラム開発の課題とアプローチ

表 2 にウォータフォール型のソフトウェア開発プロセスの上流工程と対比して、カリキュラム開発のプロセスと成果物を示す。

これまでの、学科開設や、それに伴うカリキュラム開発で提起される主要な課題は担当教員の不足や担当の割当ての偏りなど、いわば、実装段階での課題であ

る。この課題は、開発したカリキュラムが実行可能であるためには、事前に検証する必要がある。従って、カリキュラム開発では、表 2 に示す活動を繰り返して調整する必要がある。

一方、本稿で述べるソフトウェア工学カリキュラムの開発では、ソフトウェア工学担当教員が一定数いることから、カリキュラム開発の初期段階では、担当の割当てを外して、まず、カリキュラムのアーキテクチャやモジュールを決定した。

以下では、カリキュラム開発の要求開発とアーキテクチャ設計に焦点を絞って述べる。

表 2 カリキュラム開発プロセスと成果物

開発プロセス	内容と成果物
要求開発	ゴール(人材像)、カリキュラムへの要求(知識要素、達成目標など)
アーキテクチャ設計	カリキュラムアーキテクチャ 1) カリキュラムモジュール(科目)とその関係 2) 時間割(年度、学期単位での講義割り当て)
モジュール設計	カリキュラムモジュールの設計: シラバス(カリキュラムモジュール)の内容(学修目標、講義概要、講義日程)
実装	担当教員の割当てと時間割(週間)の割当て

4. ソフトウェア工学教育カリキュラムの要求開発

4.1 ソフトウェア工学科カリキュラムの要求分析

ソフトウェア工学カリキュラムへの要求と制約条件を明らかにする必要がある。そのため、要求工学と同様、次の二つの課題を解決する。

- (1) ステークホルダ分析によるゴールの明確化: ステークホルダが期待する人材、すなわち、育成すべき人材像の明確化
- (2) 制約条件の明確化: カリキュラムの内容を決定する上で考慮すべき制約条件を明確にする。

4.2 カリキュラムのゴール: ステークホルダが期待する人材像

本学科の主なステークホルダは、(1)学生、(2)卒業生を受け入れる産業界、そして、(3)教育の実行主体である大学と教員である。

- (1) 学生: 本学科の学生の80%以上が東海3県(愛知、岐阜、三重)の出身である。学生の多くは、大学院進学より就職を選ぶことから、学部教育で就職を想定し、一定水準の技術を身につけるソフトウェア工学教育が期待される。
- (2) 産業界: 就職先地域は東海3県が50%以上を占める。就職先業種もソフトウェアベンダとユーザの両方であるが、ベンダが過半数である。さらに、企業ソフトウェア開発などの分野と自動車などの組込みソフトウェア分野の両方がある。後者は、地域産業界からの組込みソフトウェア技術者へのニ-

ズとも合致する。これは、企業ソフトウェア開発技術と組込みソフトウェア開発技術の両分野を習得した人材が期待されていることを意味する。

- (3) 大学と教員: カリキュラム実行主体である大学と教員からカリキュラムに対する様々な要求がある。
 - 1) 大学: 南山大学では教育モットーとして「人間の尊厳のために(Hominis Dignitati)」を掲げている。そのため、これに関連する科目が全学共通の必修となっている。さらに、学部も技術倫理科目を必修としている。専門性と人間性のバランスがとれた人材育成が目標となる。
 - 2) 学部: ソフトウェア工学科が属する情報理工学部(現、数理情報学部)では、基礎学力として数学を中心とする数理的思考力を重視している。さらに、1-2年生は学科に配属せず、学部一括教育を行っていることから、学部共通科目の割合が必然的に高まる。
 - 3) 学科: 現在の情報通信学科では主要な専門領域として通信ネットワーク技術の科目が充実している。学科再編に伴い、これらの科目は他学科開講となる。しかし、社会のネットワーク化に伴い、通信ネットワーク技術の習得が望ましいことから、履修機会を引き続き確保する必要がある。

以上の要求から育成すべき人材像を次のようにまとめることができる。

- (a) 専門性と人間性のバランスがとれた人材: 大学の掲げるモットーに基づき、現代社会の要求でもある技術者倫理やコミュニケーションスキルの習得。
- (b) 専門と基礎のバランスがとれた人材: 数学、情報科学などの基礎科目のバランスをとった習得。
- (c) ソフトウェア工学の広がりと深化に対応できる人材: ソフトウェア工学の基礎から企業ソフトウェアと組込みソフトウェア開発技術の両方を含む、包括的な習得。

4.3 カリキュラム要求フレームワーク

カリキュラムに対する要求全体の構造をカリキュラム要求フレームワークと呼ぶことにする。上述の人材像から、図 3 にソフトウェア工学科のカリキュラム要求フレームワークを示す。

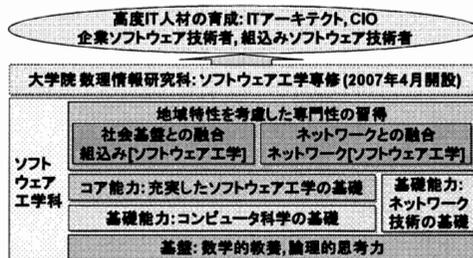


図 3 カリキュラム要求フレームワーク

4.4 カリキュラムへの制約条件

ステークホルダとして述べた、大学と教員に関する制約に加え、次の制約条件を考慮する必要がある。

- (1) 現行カリキュラムからの進化可能性: 現行カリキュラムから現実的に移行可能であること。これは教員の割当ても含むが、前述したように、ソフトウェア工学担当教員が多数いることから教員の条件は当初は外した。
- (2) 大学院連携: 図3に示すように、2007年4月に開設した大学院修士課程のソフトウェア工学専修コースと学部間での教育内容の適切な配分と継続性、一貫性を実現する。
- (3) JABEE: JABEE への準拠を保証する[11, 12]: カリキュラムの内容、達成目標、履修時間などの面で制約を課すこととなる。
- (4) 教職課程の保証。

5 ソフトウェア工学カリキュラムの設計

5.1 ソフトウェア工学カリキュラムの参照モデル

ソフトウェア工学カリキュラムに関連する知識体系として、SE2004, J07-SE/CE, SWEBOK の三つがある。表3にSE2004を示す。さらに、表4はSE2004を基礎として策定されたJ07-SEをSE2004と対比して示す。

- (1) SE2004 [8, 13, 17]: ACM/IEEE-CS が開発したカリキュラム標準 CC (Computing Curricula) 2001 に含まれるソフトウェア工学教育プログラム。
- (2) J07-SE/CE [3, 14, 25]: CC2001 に基づき、情報処理学会で策定したカリキュラム標準(J07)に含まれる SE (ソフトウェア工学: Software Engineering), あるいは、CE (組込みシステム: Computer Engineering) のカリキュラム。
- (3) ソフトウェア工学知識体系 (SWEBOK: Software Engineering Body of Knowledge)[2]

表3 SE2004 ソフトウェア工学教育知識(SEEK)

知識領域	時間	主な内容
計算機基礎	172	プログラミング、データ構造とアルゴリズム、OS、データベース、ソフトウェア構築技術
数学、工学基礎	89	離散数学、文法、統計、計測
職業、スキル	35	心理学、コミュニケーションスキル、倫理、法令遵守、標準、ソフトウェアのインパクト
モデル化と分析	53	モデル化の原理と言語、文法と意味論、構造と振舞い、ドメイン、要求工学、品質特性
ソフトウェア設計	42	設計原理、手法(オブジェクト指向など)、トレードオフ、アーキテクチャ、パターン、HCI 設計、詳細設計、ツール、評価
V&V	10	評価目標、尺度、レビュー、試験、HCI 評価
進化	10	プロセス、モデル、進化の原則、プログラム理解とリバースエンジニアリング
プロセス	13	ライフサイクルモデル、プロセスのモデル化、プロセスの分析と管理、標準
品質	16	社会への影響、品質属性、標準、プロセスとプロダクトの保証
開発管理	19	管理モデル、計画、人と組織、開発管理、構成管理
合計	459	

SE2004 で指摘されているように、SWEBOK は実務者の知識体系として定義されていることから、学部教育で習得すべき知識体系とはギャップがある[17]。従って、本稿では、SE2004 と J07 を教育カリキュラム参照モデルとし、SWEBOK を参考とした。

さらに、J07 のソフトウェア工学(SE)[3]を中核とし、組込みソフトウェア開発技術も習得できるように、J07-CE[25]の組込みソフトウェア科目を組み込んだ。

表4 SEEK と J07 のソフトウェア工学 BOK

SEEK		J07 ソフトウェア工学 BOK		主な内容
知識領域	時間	時間	数†	
計算機基礎	172	90	4	コンピュータとソフトウェアの基礎、OS 基礎・データベース基礎、ネットワーク基礎、ソフトウェア構築
数学、工学基礎	89	90*	4	確率統計*、離散数学、論理と計算理論、工学基礎
職業、スキル	35	0	0	ソフトウェア開発管理に含まれる
モデル化と分析	53	45	2	モデル化と要求開発、形式手法
ソフトウェア設計	42	90	4	ソフトウェアアーキテクチャ、ソフトウェア設計、ソフトウェア構築、ヒューマンファクター
V&V	10	23	1	V&V
進化	10	23	1	ソフトウェアプロセスと品質
プロセス	13			
品質	16			
開発管理	19	23	1	開発マネジメント
合計	459	383	17	注: †科目数, *確率統計は時間数なし
実習		90	4	プログラミング入門、プログラミング基礎実習、プログラミング応用実習、ソフトウェア開発実習

5.2 ソフトウェア工学カリキュラムの設計ゴール

育成すべき人材像と参照モデルからカリキュラム設計への具体的ゴールとして、次の7項目を挙げる。

- (1) 教養と専門のバランス
- (2) SE2004 と J07-SE を基礎とした体系的ソフトウェア工学教育の実現(特に、基礎から応用まで一貫した教育)
- (3) J07-SE に J07CE を組み入れることによる企業ソフトウェアと組込みソフトウェアの両方の領域を統合したカリキュラムの実現
- (4) JABEE への準拠
- (5) 現行のカリキュラム(As-Is)からの円滑な移行可能
- (6) 現行のスタッフで実行可能
- (7) 大学院との連携による学部・大学院一貫教育

5.3 ソフトウェア工学カリキュラムの設計

J07-SE を軸にして、ソフトウェア工学カリキュラムを設計した結果を現行カリキュラム(ソフトウェア工学コース)、J07-SE/CE と対比して表5に示す。表の1枠が1科目と対応する(J07-CE は科目と対応しない)。

(1) 設計プロセス

この設計では、次のプロセスに従った。なお、(a)~(c)のプロセスは繰り返し行い、学科カリキュラム案がまとまった時点で、(d)の合同レビューと改組のための検討と手続きを行うプロセスで学科間などでの調整を繰り返す。

返した。また、この期間、コンサルタント会社による外部からの評価を取り入れた。

- (a) 前節で述べた設計ゴールをめざし、J07-SEを参照モデルとして、現行のカリキュラムとのFit/Gap分析を行い、共通部分と差分を抽出
- (b) 制約条件を考慮し、J07-SEとの差分を吸収し、かつ、J07-CEの組込みソフトウェア開発を組み込んだカリキュラム案を作成
- (c) 作成カリキュラムの学科タスクフォースでのレビューと学科会議でのレビュー
- (d) 3学科タスクフォース合同レビュー
- (e) 学部教授会へ提案

(2) 設計内容

表5 新カリキュラムと現行、J07-SE/CEとの対比

現カリキュラム	SEカリキュラム	J07-SE	J07-CE
ソフトウェア工学	ソフトウェア工学基礎	ソフトウェアプロセスと品質 開発マネジメント	ソフトウェア工学
情報通信学実習 I ソフトウェア開発技術	ソフトウェア工学実習 ソフトウェア開発技術 I	モデル化と要求開発 ソフトウェアアーキテクチャ ソフトウェア設計	
形式手法		形式手法	
ネットワークソフトウェア工学	ソフトウェア開発技術 II ソフトウェア工学応用	V&V ヒューマンファクター	組込みシステム設計 HCI
情報通信学演習 I, II, III	ソフトウェア工学演習 I, II, III, IV	ソフトウェア開発実習	
卒業研究	卒業研究 I, II	卒業研究	
数理情報学概論	情報理工学概論		
情報表現論	基礎演習		
プログラミング I プログラミング実習 I			
プログラミング II	プログラミング基礎	プログラミング基礎	プログラミング
プログラミング実習 II	プログラミング基礎実習	プログラミング基礎実習	
プログラミング III プログラミング実習 III	プログラミング応用 プログラミング応用実習	ソフトウェア構築 プログラミング応用実習	
オブジェクト指向プログラミング	オブジェクト指向プログラミング		
プログラミング言語 プログラミング言語処理	プログラミング言語		
アルゴリズム論	アルゴリズム論		アルゴリズム
データベース	データベース	OS基礎・データベース基礎	データベースシステム OS
OS	計算機アーキテクチャとOS	コンピュータとソフトウェアの基礎	コンピュータのアーキテクチャと構成
コンピュータアーキテクチャ			
通信ネットワークアーキテクチャ	通信ネットワーク基礎	ネットワーク基礎	テレコミュニケーション
システム工学概論 I*	システム理論*	工学基礎	
情報通信学特別講義 A 情報技術倫理	情報理工学特別講義 A 情報技術倫理		社会的な観点と職業専門人としての問題
確率・統計	確率・統計	確率統計	確率・統計
情報数学	情報数学	離散数学	離散数学
数理論理学	数理論理学	論理と計算理論	

注*: 情報システム数理学科(現),システム創成学科(新)で開講

このソフトウェア工学カリキュラムでは、次の点を考慮した。

- 1) ソフトウェア工学科目の集約と基礎科目の設置
J07-SEではソフトウェア工学を細分化している。はじめてソフトウェア工学に接する学生のために、ソフトウェア工学の入門的で基礎的な内容を集約した「ソフトウェア工学基礎」科目を設定した。
- 2) 企業ソフトウェア開発と組込みソフトウェア開発の統合:「ソフトウェア開発技術 I, II」ならびに、「ソフトウェア工学応用」科目を設定し、二つの開発技術の習得を図り、あわせて、基礎から応用への橋渡しを図った。
- 3) 実習、演習の充実: 演習、実習を充実し、自ら考える機会を提供する。現行の3年次に研究室に配属し、年間を通して演習を行う科目は継続する。

(3) カリキュラムの検証

表3からソフトウェア工学の専門科目は、J07-SEで示している科目内容を包含できているといえる。さらに、適切な講義時間を保証する必要がある。表6に、ソフトウェア工学カリキュラムの授業時間を SE2004, J07-SEで規定している授業時間と対比して示す。SE2004, J07-SEと対比し、必要な講義時間を保証できると考えている。JABEEで要求される講義時間も確保できることを確認している。

表6 授業時間評価

SE 2004 SEEK	J07-SE SEBOK	ソフトウェア工学科			
		学部共通		学科	
知識領域	時間	時間	数	時間	数
計算機基礎	172	90	4		203 9
数学、工学基礎	89	90*	4	293	13 45 2
職業スキル	35	0	0	45	2 23 1
ソフトウェア工学	163	203	9		180 8
専門合計	459	383	17	338	15 451 20
演習卒研		(135)	(6)		135 6
合計		(518)	(23)	338	15 586 26

注:()は参考。演習、卒業研究は通年(2科目相当)で計算。

5.4 大学院ソフトウェア工学専修コースとの連携

2007年4月に大学院にソフトウェア工学専修コースを設置し、高度ソフトウェア開発人材の育成を実践している。このコースには通常の大学院修士課程とITスペシャリスト育成コース[22, 29]が含まれるが、講義科目は共通である。

ソフトウェア工学科カリキュラム設計では、大学院との連携も考慮した。表7に、ソフトウェア工学科カリキュラムと大学院ソフトウェア工学専修コースの講義カリキュラムとを、J07-SEと対比して示す(確率・統計などの数学関連の基礎科目と卒業研究等は除く)。

ソフトウェア工学科カリキュラムではソフトウェア工学の基礎に重点を置き、専門性の高い内容は大学院の講義に委ねることとした。

一方、ソフトウェア開発に技術に関する内容をより深く学べるよう、大学院では、ソフトウェア開発技術の科目

を拡充した。あわせて、ソフトウェアプロジェクト管理などの、開発管理に関する科目も強化した。

表 7 大学院ソフトウェア工学カリキュラムとの対比

大学院 SE 専修	SE カリキュラム	J07-SE	J07-CE
ソフトウェア工学概論	ソフトウェア工学基礎	ソフトウェアプロセスと品質	ソフトウェア工学
ソフトウェア保守			
ソフトウェアプロジェクト管理		ソフトウェア開発マネジメント	
ソフトウェア要求工学	ソフトウェア開発技術 I	モデル化と要求開発	
ソフトウェアアーキテクチャ		ソフトウェアアーキテクチャ	
ソフトウェア設計技術		ソフトウェア設計	
		形式手法	
ソフトウェアモジュール化技術	ソフトウェア開発技術 II		組込みシステム設計
V&V		V&V	
実践的ソフトウェア開発技術	ソフトウェア工学応用	ヒューマンファクター	HCI
ソフトウェア構築	プログラミング基礎	プログラミング基礎	プログラミング
	プログラミング基礎実習	プログラミング基礎実習	
	プログラミング応用	ソフトウェア構築	
	プログラミング応用実習	プログラミング応用実習	
	オブジェクト指向プログラミング		
	プログラミング言語		
	アルゴリズム論		アルゴリズム
(データベース)	データベース	OS 基礎・データベース基礎	データベースシステム OS
組込みシステム開発技術 I	計算機アーキテクチャと OS		
組込みシステム開発技術 II		コンピュータとソフトウェアの基礎	コンピュータのアーキテクチャと構成
IT ネットワーク	通信ネットワーク基礎	ネットワーク基礎	テレコミュニケーション
情報システム開発技術	システム理論* 情報理工学特別講義 A	工学基礎	社会的な観点と職業専門人としての問題
IT 技術倫理と社会	情報技術倫理		

6 評価と議論

本稿で提示したカリキュラムの内容と開発方法の二つの観点から、評価と議論を行う。

6.1 ソフトウェア工学科への反応

ソフトウェア工学科は受験生から次の反応を得た。

(1) 学科イメージ調査: 高校生への Web アンケート Web による調査では概ね好意的な反応を得た。

(2) 一般入試受験者数の増大

受験者数が減少する傾向にあつて、初年度である 2009 年 2 月の一般入試受験者数は前年度より増加となった。

6.2 グローバルベンチマーキング

表 8 に海外のソフトウェア工学の学位プログラムの事例を示す。米国では、Rochester Institute of Technology が 1996 年に最初のソフトウェア工学の学部教育 (Bachelor's of Software Engineering) を開設した。

ABET のア krediteーションもソフトウェア工学で習得している[1]。

世界的に見て、コンピュータ科学、情報科学に比べソフトウェア工学の学位プログラムは少ない。2007 年時点では、米国で 30 校以上、カナダで 20 校以上がソフトウェア工学の学位を提供している[18]。しかし、この中には、近年、ソフトウェア工学の学位プログラムを開設した例が多く、増加の傾向にあると言える。

また、国により取組みの違いが伺える。例えば、カナダやオーストラリアでは、大学数に比較してソフトウェア工学を提供している大学の比率が高い。

表 8 ソフトウェア工学プログラムの事例

大学名	学科名	学位名	ア krediteーション
Rochester Institute of Technology (USA)	Dep. of SE.	B. of SE	ABET (SE)
U. of Texas at Dallas (USA)	CS Dep.	B. of CS, SE	ABET (SE)
Drexel U. (USA)	College of Eng. & College of Inf. Science & Technology*	B. of SE	ABET (SE)
Carnegie Mellon U. (USA)	School of CS (BSi.in CS)	Minor in SE	
Carleton U. (Canada)	Dep. of Systems & CE, Dep. of Electronics*	B. of SE	CEAB (SE)
U. of Victoria (Canada)	Dep. of Electrical & C E, Dep. of CS*	B. of SE	CEAB (SE)
U. of Waterloo (Canada)	Dep. of Electrical & CE, School of CS*	B. of SE	CEAB (SE)
U. Edinburgh (UK)	School of Informatics	B. of SE	
De Montfort U. (UK)	Faculty of Technology (CS & Eng.)	B. of SE	
Monash U. (Australia)	Faculty of IT	B. of SE	
The Royal Melbourne Institute of Technology (Australia)	School of CS & IT	B. of SE	

注*: 複数学科の共同運営による学位プログラム。

略語: CE: Computer Engineering, CS: Computer Science, IT: Information technology

本稿で述べたソフトウェア工学科のカリキュラムをグローバルな視点から評価するために、ソフトウェア工学科の優れた実現例である University of Victoria(UV)のソフトウェア工学プログラム(Bachelor's of Software Engineering) [31]と比較する。この学位プログラムは同国のア krediteーション CEAB (Canadian Engineering Accreditation Board)をカナダ西部地域で最初に取得している。

表 9 に UV のソフトウェア工学カリキュラムとの対比を示す。大学制度の違いもあり、UV では 4 年次で多くの講義科目を設定している。これらの科目は、ソフトウェア工学の応用や、基礎科目の内容を深化したものとなっていることから、応用に関する科目が多数提供されていることが分かる。基礎科目に関しては、当学科のカリキュラムは同水準にあると評価できる。

6.3 わが国におけるソフトウェア工学教育カリキュラム

本稿で示したソフトウェア工学カリキュラムは J07-SE を基本的に満たすように設計したが、以下の点で、検討すべき課題がある。

(1) ソフトウェア工学教育コアカリキュラムのモデル

ソフトウェア工学の卒業生に対して、企業ソフトウェア開発と組込みソフトウェア開発の両分野の知識の習得が期待されている。特に、わが国では、この要求は高い。J07, ならびに、その基礎となったCC2001では、SEとCEに分離されているため、カリキュラム開発では、この二つの標準を考慮した。今後、ソフトウェア工学のカリキュラムのモデルとして、組込みソフトウェアを組み込んだカリキュラムのモデルが必要であると思われる。

(2) カリキュラムの制約と盛り込むべき内容の妥当性

J07-SE/SE2004では、ソフトウェア工学の主要な技術要素を幅広く取り入れている。そのため、ソフトウェア開発マネジメントや形式手法が単一の科目として提示されている。しかし、学部教育で、これらの科目を単一科目として設定することは、時間数の制約などの面で負担となる。本稿で述べたカリキュラムでは、大学のモットーに基づく教養や数学などの基礎教育を重視していることから、一部の科目はまとめることとした。今後、学部におけるソフトウェア工学の実践が増え、コアとすべきカリキュラムの妥当性の議論を深める必要がある。

(3) 学部と大学院との分担

カリキュラム内容の妥当性と共に、学部と大学院間でのカリキュラムの分担と整合性について理解を深める必要がある。

6.4 カリキュラム開発へのソフトウェア工学アプローチについて

本稿では、カリキュラム開発へソフトウェア工学の枠組みを応用する方法について述べた。現在のカリキュラム開発では、教育目標とする内容だけではなく、JABEEで定める学習履修目標や学習時間の確保、学部の制約など、多様な制約を満たす必要がある。この問題は、ソフトウェア開発と類似点が多く、かつ、複雑であることから、本稿で述べたソフトウェア工学のアプローチは有効であると考えられる。

一方、教育の分野では、カリキュラム開発の方法について研究されている[5, 26]。これらの研究においても、工学的アプローチが提起されているが、方法論として体系化されるまでには至っていない。特に、大学教育では、カリキュラムが多様であることから、いわば、個別ソフトウェア開発に近いと考えられ、ソフトウェア工学の諸技術の活用が期待できる。

7 今後の課題

まず、開設したソフトウェア工学科が社会から評価されるよう、カリキュラムの効果的な実践を図り、あわせて、学生や社会からのフィードバックを図り、継続的改善の仕組みを確立する。

また、本稿で提案したカリキュラム開発へのソフトウェ

ア工学の応用について理解を深め、技術を整理する。

表9 Univ. of Victoria のソフトウェア工学科との対比

大学院SE専修	SEカリキュラム	J07-SE	UV SE
ソフトウェア工学概論	ソフトウェア工学基礎		Eng. SW Sys.
ソフトウェア保守		ソフトウェアプロセスと品質	SW Process
ソフトウェアプロジェクト管理		開発マネジメント	SW Evolution SW Quality Eng. Applied Cost Eng. Management of SW Dev. Inf. & Knowledge Management Human Factors in Eng.
	ソフトウェア工学実習		
ソフトウェア要求工学	ソフトウェア開発技術I	モデル化と要求開発	Requirements Eng. & Formal Spec.
ソフトウェアアーキテクチャ		ソフトウェアアーキテクチャ	SW Arch. & Sys. SW Arch.
ソフトウェア設計技術		ソフトウェア設計形式手法	Component-Based SE Software Dev. Methods
ソフトウェアモジュール化技術	ソフトウェア開発技術II		SW for Embedded & Mechatronics Sys. Embedded Sys. Security Eng.
V&V		V&V	Reliability Eng.
実践的ソフトウェア開発技術	ソフトウェア工学応用	ヒューマンファクター	HCI Network-centric Computing Distributed Sys. & Internet Media Applications CSCW Practice in Inf. Security
PBL	ソフトウェア工学演習I, II, III, IV	ソフトウェア開発実習	Directed Studies
	卒業研究I, II	卒業研究	Technical Project
	情報理工学概論		Eng. Fundamentals
	基礎演習		
ソフトウェア構築	プログラミング基礎 プログラミング基礎実習 プログラミング応用 プログラミング応用実習 オブジェクト指向プログラミング プログラミング言語 アルゴリズム論	ソフトウェア構築	Fundamentals of Programming I Fundamentals of Programming II Object-Oriented SW Dev. Algorithms & Data Structures
(データベース)	データベース	OS基礎・データベース基礎	Data Mining
組込みシステム開発技術I	計算機アーキテクチャとOS		Computer Arch. & Assembly Lang. Intro. to OS
組込みシステム開発技術II		コンピュータとソフトウェアの基礎	Logic & Foundations
ITネットワーク	通信ネットワーク基礎	ネットワーク基礎	Network Security
情報システム開発実践特論	システム理論*	工学基礎	Control Theory & Sys. Topics in SE
IT技術倫理と社会	情報理工学特別講義A 情報技術倫理		Social & Professional Issues

UV SEの略称: Arch: Architecture, Dev.: Development, Eng.: Engineering, Inf.: Information, Intro.: Introduction, Lang.: Language, SE: Software Engineering, SW: Software, Spec.: Specification(s), Sys.: System(s)

8 まとめ

ソフトウェア工学科の開設にあたって解決すべき課題を示し、その中核的課題であるカリキュラム開発へソフトウェア工学の諸技術を応用する方法を提案した。

今後、ソフトウェア工学教育の実践を通して、カリキュラムの継続的改善と教育の充実を図る必要がある。

現代社会においてソフトウェア工学の専門技術者を育成するソフトウェア工学科の開設は大学の社会的責務である。本稿で述べたソフトウェア工学科の開設を契機にわが国におけるソフトウェア工学科の設置が進むことを期待する。

謝辞: ソフトウェア工学科の実現に支援頂いた関係各位に感謝する。

参考文献

- [1] ABET, <http://www.abet.org/>.
- [2] A. Abran and J. M. Moore (Executive Eds.), *Guide to the Software Engineering Body of Knowledge, 2004 Version*, <http://www.swebok.org/>.
- [3] 阿草 清滋 ほか, ソフトウェアエンジニアリング領域 (J07-SE), 情報処理, Vol. 49, No. 7, Jul. 2008, pp. 743-749.
- [4] 赤堀 侃司, 教育工学への招待, ジャストシステム出版部, 2002.
- [5] 天野 正輝, 評価を生かしたカリキュラム開発と授業改善, 晃洋書房, 2006.
- [6] 青山 幹雄, ソフトウェア技術者のグローバルスタンダード化, 情報処理, Vol. 39, No. 11, Nov. 1998, pp. 1144-1147.
- [7] 青山 幹雄 ほか, ソフトウェア工学科の設計, 電子情報通信学会 知能ソフトウェア工学研究会, 2009年3月[発表予定].
- [8] J. M. Atlee, et al., Reflections on Software Engineering 2004, the ACM/IEEE-CS Guidelines for Undergraduate Program in Software Engineering, *Software Engineering Education in the Modern Age*, LNCS Vol. 4309, Springer, 2006, pp. 11-27..
- [9] S. T. Frezza, et al., Creating an Accreditable Software Engineering Bachelor's Program, *IEEE Software*, Vol. 23, No. 6, Nov./Dec. 2006, pp. 27-35.
- [10] C. Ghezzi and D. Mandrioli, The Challenges of Software Engineering Education, *Software Engineering Education in the Modern Age*, LNCS Vol. 4309, Springer, 2006, pp. 115-127.
- [11] JABEE, <http://www.jabee.org/>.
- [12] 情報処理学会 アク্রেディテーション委員会, <http://jabee.ipsj.or.jp/>.
- [13] Joint Task Force for Computing Curricula 2005, *Computing Curricula 2005: The Overview Report*, Sep. 2005, <http://www.acm.org/education/curricula-recommendations>.
- [14] 兼宗 進, 寛 捷彦(編), 情報専門学科カリキュラム標準 J07, 情報処理, Vol. 49, No. 7, Jul. 2008, pp. 719-774.
- [15] 経済産業省, 大学等における IT 教育実態調査報告書(情報系学科卒業生の視点), 2004年3月, http://www.meti.go.jp/policy/it_policy/jinzai/pdf/it_kyouikujittai.pdf.
- [16] J. Kramer, Is Abstraction the Key to Computing?, *CACM*, Vol. 50, No. 4, Apr. 2007, pp. 37-42.
- [17] T. C. Lethbridge, et al., SE 2004: Recommendations for Undergraduates Software Engineering Curricula, *IEEE Software*, Vol. 23, No. 6, Nov./Dec. 2006, pp. 19-25 [Software Engineering 2004 とは何か, 日経エレクトロニクス, 2007年6月4日号, pp. 105-114].
- [18] T. C. Lethbridge, et al. Improving Software Practice through Education: Challenges and Future Trends, *Proc. ICSE 2007, Future of Software Engineering*, IEEE CS, May 2007, pp. 14-28.
- [19] L. N. Long, The Critical Need for Software Engineering Education, *CrossTalk*, Vol. 21, No. 1, Jan. 2008, pp. 6-10.
- [20] N. R. Mead, Software Engineering Education: How Far We've Come and How Far We Have to Go, *Proc. CSEET (Conf. on Software Engineering Education and Training) '08*, IEEE CS, Apr. 2008, pp. 18-22.
- [21] 三菱総合研究所, 「産業競争力向上の観点からみた大学活動評価手法」を用いた試行評価結果, <http://www.univinfo.jp/rating/>.
- [22] 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, http://www.mext.go.jp/a_menu/koutou/it/.
- [23] B. Meyer, Software Engineering in the Academy, *IEEE Computer*, Vol. 34, No. 5, May 2001, pp. 28-35.
- [24] 日本経済団体連合会, 産学連携による高度な情報通信人材の育成強化に向けて, 2005, <http://www.keidanren.or.jp/japanese/policy/2005/039/honbun.pdf>.
- [25] 大原 茂之, コンピュータエンジニアリング領域 (J07-CE), 情報処理, Vol. 49, No. 7, Jul. 2008, pp. 750-758.
- [26] O. Rompelman and E. De Graaff, The Engineering of Engineering Education: Curriculum Development from a Designer's Point of View, *European J. of Engineering Education*, Vol. 31, No. 2, May 2006, pp. 215-226.
- [27] 産業構造審議会 情報経済分科会 情報サービス・ソフトウェア小委員会 人材育成 WG, 高度 IT 人材の育成をめざして, 2007年7月20日, http://www.meti.go.jp/press/20070720006/03_houkokusho.pdf.
- [28] M. Shaw, Software Engineering Education: A Roadmap, *Proc. ICSE 2000, Future of Software Engineering*, ACM, Jun. 2000, pp. 371-380.
- [29] 田口 潤, 「高度 IT 人材育成」その理想と現実, 日経コンピュータ, 2007年6月11日号, pp. 98-103.
- [30] L. Tanner, et al., *Curriculum Development, 4th ed.*, Pearson Education, 2007.
- [31] University of Victoria, *Bachelor of Software Engineering*, <http://www.bseng.uvic.ca/>.
- [32] 山下 徹(編著), 高度 IT 人材育成への提言, NHK 出版, 2007年.