



# ソフトウェアエンジニアリング 領域 (J07-SE)

Software Engineering Discipline (J07-SE)

J07

**阿草清滋** 名古屋大学

**西 康晴** 電気通信大学

**沢田篤史** 南山大学

**鷲崎弘宜** 早稲田大学

## はじめに

J07-SE は、情報専門学科カリキュラム標準 J07 におけるソフトウェアエンジニアリング領域のカリキュラムモデルである。これは情報処理学会ソフトウェアエンジニアリング教育委員会が 2006 年 4 月から 2008 年 3 月まで 2 年間にわたり議論を行って策定したものである。

本稿では、ソフトウェアエンジニアリングを取り巻く状況、J07-SE カリキュラムモデルの全体像、知識項目および科目、年次進行、今後の取り組みを概説する。

## SE を取り巻く状況

近年、企業情報システム（エンタープライズ系システム）やソフトウェア組込み製品（組込み系システム）の発展に伴い、ソフトウェアは社会に欠くべからず存在となった。我々の生活も企業の活動も、ソフトウェアなしではまったく立ちゆかなくなってきた。

しかしその一方で、ソフトウェアに起因するシステム障害や製品リコールが多発している。特に最近では、証券取引所・銀行間取引システムのような重要な社会インフラや、人命にかかわる製品にも及んでいる。このままでは、我が国の産業競争力にも消費者安全にも大きな負の影響を及ぼしてしまう。

原因についてはさまざまな議論があるが、質の高い人材を生み出すシステムが機能していないことが大きな一因である点は論を俟たない。我が国のソフトウェアの多くは、きちんとソフトウェアエンジニアリング (SE) を学んでおらず、また企業でも十分な体系的教育を受けていない技術者によって作り続けられている。これはきわめて憂慮すべき問題である。

ソフトウェアエンジニアリングとは何か。「ソフトウェアを開発する際に駆使すべき技術」<sup>1)</sup>であり、「ソフトウェアに対する工学的原則の適用」<sup>2)</sup>である。SE はコンピュータサイエンス (CS) に基づいているが、重点を置くところが異なる。科学者は自然法則の知識を拡張するのに対して、技術者は数多くの制約条件のもとに、役に立つ成果物を構築するために自然法則を適用しなくてはいけないからだ<sup>2)</sup>。しかし我が国では、CS すら学んだことのない技術者と、大学等で CS しか学ばず SE は学んでいない技術者によって多くのソフトウェアが作り続けられているという問題を抱えている。

この問題を解決するために必要なのは、多くの技術者を生み出す大学などの高等教育機関における SE 教育の質の向上である。もちろんこれまでも、意識の高い研究者は自らの講義のために実践的・先端的な SE 教育を整備してきているし、一部の大学では産学連携の SE 教育を行ってきた。いくつかの NPO 法人では SE 教育のカリキュラムや内容に対する議論が交わされている。経済産業省や文部科学省、日本経済団体連合会(経団連)はいくつかの大学や大学院に対して SE 教育支援を行っている。また経済産業省・情報処理推進機構による IT スキル標準 (ITSS)<sup>3)</sup> や組込みスキル標準 (ETSS)<sup>4)</sup>、情報システムユーザースキル標準 (UISS)<sup>5)</sup> などでも大学教育との関連性が議論されている。

情報処理学会では、1999 年からアクレディテーション委員会 SE 分科会として、SWEBOK<sup>2)</sup> や CCSE<sup>6)</sup> を始めとする SE の知識体系の調査、日本の大学の学部などで実行可能なカリキュラムモデル Jpn1 の策定、(特に米国における) SE アクレディテーション動向の調査などを行ってきた。これらの活動成果は平成 13 年 3 月に開催した情報処理学会「アクレディテーション」シンポジウ

ムにおいて発表し、また同じく平成13年3月に発行された「ソフトウェア工学におけるアクレディテーションに関する調査研究」調査報告書としてまとめられている。

こうした流れを踏まえ、さまざまなSE教育の取り組みによって質の高い人材を生み出す仕組みを活性化できるよう、改めて情報処理学会としてのSEの知識項目の整備やカリキュラムモデルの策定が強く望まれている。J07としてSEの知識項目やカリキュラムの参照モデルを提示することで、さまざまなSE教育の取り組みを整理できるからである。それによって強みや弱みを明らかにでき、さらなる改善を促すことができる。もちろん、SEの学科やコースの設置が容易になることは言うまでもない。

そこで我々SE教育委員会は、J07の一領域としてカリキュラムモデルを策定することとした。その内容は情報処理学会全国大会にて報告を行うとともに、J07/SE領域のWebサイト(<http://blues.se.uec.ac.jp/j07/>)に公開した。本稿では、以上の資料をまとめたものを報告する。

## J07-SEの全体像

### ミニマムセットのカリキュラムモデル

J07-SEは、大学などの高等教育機関の情報専門学科におけるSE教育のためのカリキュラムモデルである。

まず、カリキュラムそのものではない点に注意が必要である。本カリキュラムモデルは、カリキュラムを開発するために最低限必要な知識項目、それらを教えるコア科目、演習やインターンシップの仕様、学科やコースの特色を出すためのSAS科目(System and Application Specialties: システム応用・特化型科目)の例、年次進行の例から構成される。学科やコースにおいてソフトウェアエンジニアリングのカリキュラムを開発するためには、本カリキュラムモデルを基にして、知識項目の把握、コア科目の理解、演習やインターンシップの実装、SAS科目の選択や開発、年次進行の策定などを行うことが必要となる。

次に、J07-SEはSEの学科やコースにおけるミニマムセットである点に注意が必要である。本カリキュラムモデルは、日本の大学が置かれている競争的環境を踏まえ、学科やコースの特色を出す余裕を持たせるために、最低限必要な知識項目に絞ってコア科目を列挙している。そのため、学科やコースの特色を反映した科目を追加しなくてはならない。

例として、エンタープライズ系システム向けSEカリキュラムや組込みシステム向けSEカリキュラム、Webシステム向けSEカリキュラムなどの適用ドメイン的な特色を反映した学科やコース、要求開発やアーキテクチ

ャ設計、実装、検証と妥当性確認、マネジメントなど技術的な特色を反映した学科やコースなどが考えられる。J07-SEでは、カリキュラム開発者のためにSAS科目として、学科やコースの特色を反映させるための追加科目を例示している。特色を出しやすいように、演習などについては内容の詳細に踏み込まず、仕様を提示するにとどめている。

また真に実践的で充実した教育を行うためには、学部教育に許された時間数では少ないと言わざるを得ない。大学院との連携教育が可能な学科やコースは、ミニマムセットであるJ07-SEを拡張することで大学院修士課程も含めた6年間のカリキュラムモデルを構築するとよい。その際にはSE教育委員会の以前の成果物であるJpn1カリキュラムモデルが参考になるだろう。

### コンセプト

SEは、その名の通り「エンジニアリング」である。エンジニアリング教育には、主に2つの特性が求められる。1つは“実践的”であり、もう1つは“骨太”である。

従来から大学で行われている情報系の教育は、その出自や理念などにより、理論的なものが多いと言われている。しかし産業界からは、開発現場であり役に立たないという批判も多い。そこでJ07-SEは、産業界のニーズに応えるために従来の基礎一辺倒のカリキュラムから、実践的内容にも踏み込んだカリキュラムモデルを指向することとした。まず、プログラミング言語の習得にとどまらず、開発の全体像である要求開発から検証と妥当性確認、保守、プロセスやマネジメントといった開発ライフサイクルを網羅するカリキュラムモデルとなっている。次に、理論の理解にとどまらず、開発の目的である品質・生産性・コストといった要因を重視するカリキュラムモデルとなっている。また、個人に閉じた作業にとどまらず、開発に必要なマネジメントやコミュニケーション、チームダイナミクスなどを学び実践するカリキュラムモデルとなっている。さらに、開発者のモチベーションや、不具合につながる開発者のヒューマンエラーといった心理的側面も取り扱っている。

一方でJ07-SEは、SEの技術進化の速さを鑑みるに近視眼的な技術訓練では不十分であると考え、骨太なカリキュラムモデルを指向した。特に開発現場寄りの産業界からは、すぐに現場で使える技術を求められることもある。しかし、SEにおいて重要なのは技術の使い方よりも“ものの考え方”そのものである。そこで、モデル化を習得することで「捉える力」や「考える力」、「表現する力」などを、検証と妥当性確認やプロセス改善を習得することで「問題発見能力」や「問題解決能力」などを、プロセスやマネジメントを習得することで「段取り力」や「調

整力」などを涵養することを指向した。またソフトウェアにとどまらない一般的な工学原則も学ぶこととした。これによって、技術を単に理解するだけでなく、臨機応変に応用でき中長期的に付加価値を生み続けられる技術者の育成が期待できる。

J07-SE の特徴は、こうした内容を PBL (Project Based Learning) のみに担わせるべきではないと意図している点にある。PBL はその性格上、基本概念や技術を習得する際に、PBL で採用した開発やマネジメントのスタイルに基づく一面的な理解になりかねない。そこで J07-SE では、講義においてさまざまな基本概念や技術を提示することで、PBL が採用するスタイル以外のさまざまな開発やマネジメントのスタイルの特徴を多面的に深く理解することを指向した。

また、卒業研究によって SE の技術やマネジメントの本質に触れ、技術者として巣立った後に SE の技術やマネジメントを改善し発展させ新たに創造することも期待している。したがって J07-SE は、実践的な教育を実施するという意味で PBL を卒業研究に代えるという構成の学科やコースも許容する一方で、PBL に加えて卒業研究を行う学科やコースも許容している。J07-SE を参照する学科やコースは、ぜひ講義での理解と PBL での実践を踏まえた質の高い卒業研究指導を目指してほしい。

### カリキュラムモデルの構造

J07-SE は、講義科目と実習科目、SAS 科目、インターンシップ、卒業論文から構成される。講義科目はさらに、情報科学基礎科目と SE 科目に分けられる。情報科学基礎科目とは、論理と計算理論、離散数学、オペレーティングシステム基礎・データベース基礎といった SE の基礎となる科目である。SE 科目とは、ソフトウェア構築やソフトウェア設計、検証と妥当性確認、開発マネジメントといった SE 技術を扱う科目である。

実習科目は、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から構成される。プログラミング入門では、プログラミング言語や開発環境の習得、データ構造とアルゴリズムの理解と実装を行う。教養系で実施する一般的なプログラミング入門を想定している。プログラミング基礎実習では、設計内容に対する個人でのプログラム開発と単体・統合テストを行う。プログラミング応用実習では、要求仕様に対する個人ないしグループでのアーキテクチャ設計、ソフトウェア設計、プログラム開発、単体・統合・システムテストを行う。ソフトウェア開発実習では、システムへの要望に対するグループでのシステム開発とプロジェクトマネジメントを行う。エンタープライズ系や組込み系など学科やコースの特色に合わせて、かなり

具体的なシステムを想定した実習となる。

SAS (System and Application Specialties) 科目は、学科やコースの特色を反映させるための追加科目である。J07-SE では、システムのドメイン、システム特性、システムの構造、開発技術による SAS 科目を例示している。ただし科目名の提示にとどめ、詳細は学科やコースで定める必要がある。

インターンシップは、民間企業などとの連携により、現実のソフトウェア開発プロジェクトを教材とした実践的な学習を行う科目である。単なる例題にとどまらない現実のソフトウェア開発作業にかかわることにより、納期、予算、組織といった実社会の制約を踏まえたソフトウェア開発の実際について学ぶ。目的意識を持った内容の濃いインターンシップとするために、事前報告書を作成することでインターンシップの目的を深耕し、事後報告書や発表によって内容を咀嚼するだけでなく、インターンシップ受け入れ先からも評価を受ける必要がある。

卒業研究は、在学中に修得したソフトウェアエンジニアリングに関する知識や技術のすべてを、初めての新しい課題に集中して独自の解を求め、その研究成果を論文または制作物の形にまとめ、発表する科目である。これを通して、研究の推進方法、関連研究の調査による知識の進歩、および、新しい独創的な考えを展開してゆく能力を養成する。

## J07-SE の知識項目および科目

### 知識項目

カリキュラムモデルにまず必要となるのは、教えるべき知識項目の整理である。SE 分野には SWEBOK (Software Engineering Body of Knowledge)<sup>2)</sup> などいくつかの知識項目が定義されているが、J07-SE では J07 の方針を踏襲し、CCSE2004 (Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering / A Volume of the Computing Curricula Series, August 23, 2004)<sup>6)</sup> を参照し、科目として取り扱いやすい単位として再体系化を行っている。表-1 に知識項目の第 2 カテゴリーまでを示す。

### 情報科学基礎科目

J07-SE では前述した知識項目を基に、8つの情報科学基礎科目を提示した。表-2 に情報科学基礎科目の一覧を示す。なお確率統計については教養系で実施する講義を想定しているが、各種 SE 技術の背景・基礎という観点で習得の必要な内容をカリキュラムモデルに含めて策定した。プログラミング基礎については、学科やコースの特色に応じたプログラミング言語を対象として科目を

コンピュータとソフトウェアの基礎			検証と妥当性確認		
CMP	コンピュータ基礎		VAV	検証と妥当性確認(V&V)	
	CMP.cf	コンピュータ科学基礎		VAV.fnd	V&Vの用語と基礎
N/A	N/A	ソフトウェア工学の基礎		VAV.rev	レビュー
確率統計				VAV.tst	テスト
FND	数理基礎・工学基礎			VAV.hct	HCIのテストと評価
	FND.mf	数理基礎		VAV.par	不具合の分析と報告
	FND.ef	ソフトウェアのための工学基礎	形式手法		
離散数学			CMP	コンピュータ基礎	
FND	数理基礎・工学基礎			CMP.fm	形式手法
	FND.mf	数理基礎	MAA	ソフトウェアのモデリングと分析	
プログラミング基礎				MAA.md	モデリングの基礎
CMP	コンピュータ基礎		ソフトウェアプロセスと品質		
	CMP.cf	コンピュータ科学基礎	FND	数理基礎・工学基礎	
	CMP.ct	構築技術		FND.ec	ソフトウェアのためのエンジニアリングエコノミクス
	CMP.tl	構築のためのツール	PRF	プロフェッショナルプラクティス	
論理と計算理論				PRF.pr	プロフェッショナルリズム
FND	数理基礎・工学基礎		EVO	ソフトウェアの進化や保守	
	FND.mf	数理基礎		EVO.pro	進化や保守のプロセス
オペレーティングシステム基礎・データベース基礎				EVO.ac	進化や保守のアクティビティ
CMP	コンピュータ基礎		PRO	ソフトウェア開発プロセス	
	CMP.cf	コンピュータ科学基礎		PRO.con	プロセスの基礎
ネットワーク基礎				PRO.imp	プロセスの実装
CMP	コンピュータ基礎		QUA	ソフトウェア品質	
	CMP.cf	コンピュータ科学基礎		QUA.cc	ソフトウェア品質の概念と文化
工学基礎				QUA.std	ソフトウェア品質に関する標準
CMP	コンピュータ基礎			QUA.pro	ソフトウェア開発プロセスの改善
	CMP.cf	コンピュータ科学基礎		QUA.pca	プロセスの保証
FND	数理基礎・工学基礎			QUA.pda	プロダクトの保証
	FND.ef	ソフトウェアのための工学基礎	ヒューマンファクター		
PRF	プロフェッショナルプラクティス		CMP	コンピュータ基礎	
	PRF.pr	プロフェッショナルリズム		CMP.cf	コンピュータ科学基礎
ソフトウェア構築				DES.hci	ヒューマン・コンピュータ・インタフェース(HCI)設計
CMP	コンピュータ基礎		開発マネジメント		
	CMP.cf	コンピュータ科学基礎	FND	数理基礎・工学基礎	
	CMP.ct	構築技術		FND.ec	ソフトウェアのためのエンジニアリングエコノミクス
	CMP.tl	構築のためのツール	PRF	プロフェッショナルプラクティス	
DES	ソフトウェア設計			PRF.psy	グループダイナミクス/心理学
	DES.con	設計に用いられる概念		PRF.com	(SEに特化した)コミュニケーションスキル
	DES.dd	詳細設計		PRF.pr	プロフェッショナルリズム
モデル化と要求開発			MAA	ソフトウェアのモデリングと分析	
MAA	ソフトウェアのモデリングと分析			MAA.af	モデルの分析の基礎
	MAA.md	モデリングの基礎		MAA.rfd	要求分析の基礎
	MAA.tm	モデルの種類		MAA.er	要求の獲得
	MAA.rv	要求の評価		MAA.rsd	要求の仕様化と文書化
ソフトウェアアーキテクチャ			MGT	ソフトウェア開発のマネジメント	
DES	ソフトウェア設計			MGT.con	マネジメントの基礎
	DES.con	設計に用いられる概念		MGT.pp	プロジェクトの計画
	DES.str	設計のパラダイム		MGT.per	プロジェクトのメンバと組織
	DES.ar	アーキテクチャ設計		MGT.ctl	プロジェクトのコントロール
	DES.ste	設計の支援ツールと評価		MGT.cm	ソフトウェア構成管理
ソフトウェア設計					
DES	ソフトウェア設計				
	DES.dd	詳細設計			

表-1 J07-SEの知識項目(第2カテゴリまで)

開発すべきである。本カリキュラムモデルでは、例として C 言語を用いた科目を提示している。

### SE 科目

J07-SE では前述した知識項目を基に、9つの SE 科目を提示した。表-3 に SE 科目の一覧を示す。

### 実習科目とインターンシップ、卒業研究

J07-SE では、4つの実習科目とインターンシップ、卒業研究を提示した。表-4 に実習科目、インターンシップ、卒業研究の一覧を示す。なおプログラミング入門については、教養系で実施する一般的な演習を想定しているため、科目内容は策定していない。またソフトウェア開発実習については、例としてエンタープライズ系と組込み

コンピュータとソフトウェアの基礎  
確率統計  
離散数学  
プログラミング基礎  
論理と計算理論  
オペレーティングシステム基礎・データベース基礎  
ネットワーク基礎  
工学基礎

表-2 J07-SE の情報科学基礎科目の一覧

ソフトウェア構築  
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  
検証と妥当性確認  
形式手法  
ソフトウェアプロセスと品質  
ヒューマンファクター  
開発マネジメント

表-3 J07-SE の SE 科目の一覧

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  
インターンシップ  
卒業研究

表-4 J07-SE の実習科目、インターンシップ、卒業研究の一覧

系の 2 種を示す。

### SAS 科目

J07-SE では SAS 科目の例を提示した。SAS 科目の例は、システムのドメイン、システムの特長、システムの構造、開発技術などに分類している。ただし J07-SE では SAS 科目例の内容は定めていない。表-5 に SAS 科目の例一覧を示す。

#### システムのドメインによる SAS 科目

Web システム(ネットワーク中心・Web システム)  
企業情報システム・ERP (情報システムとデータ処理)  
金融・電子商取引システム  
物流・小売システム  
通信・ネットワークシステム  
マルチメディア・エンターテインメントシステム  
(ゲーム・娯楽システム)  
ユビキタスシステム(モバイルシステム)  
航空・車システム  
工業プロセス制御システム  
生命医学システム(バイオメディカルシステム)  
科学技術計算システム

#### システムの特長による SAS 科目

高信頼性・高可用性システム/組込み  
(ディペンダブルシステム)  
高信頼性・高可用性システム/エンタープライズ  
(ディペンダブルシステム)  
高セキュリティ情報システム  
高安全性組込みシステム

#### システムの構造による SAS 科目

ハードウェア制御・リアルタイムシステム  
(組込み・リアルタイムシステム)  
トランザクションシステム  
制御モデル開発  
経営情報システム  
レガシーシステムと派生開発  
OSS によるシステム開発  
エージェント・人工知能システム

#### 開発技術による SAS 科目

ソフトウェア構築・応用  
モデル化と要求開発・応用  
ソフトウェアアーキテクチャ・応用  
ソフトウェア設計・応用  
検証と妥当性確認・応用  
形式手法・応用  
ソフトウェアプロセスと品質・応用  
ヒューマンファクター・応用  
開発マネジメント・応用

表-5 J07-SE の SAS 科目の例の一覧

講義科目名	1年 前期	1年 後期	2年 前期	2年 後期	3年 前期	3年 後期	4年 前期	4年 後期
コンピュータとソフトウェアの基礎	○							
確率・統計		○						
離散数学		○						
プログラミング基礎		○						
プログラミング入門		○						
論理と計算理論			○					
オペレーティングシステム基礎・データベース基礎			○					
ソフトウェア構築			○					
ネットワーク基礎				○				
モデル化と要求開発				○				
ソフトウェア設計				○				
プログラミング基礎実習				○				
ソフトウェアアーキテクチャ					○			
検証と妥当性確認					○			
ソフトウェアプロセスと品質					○			
プログラミング応用実習					○			
インターンシップ(夏期休暇に実施)					○			
形式手法						○		
開発マネジメント						○		
ヒューマンファクター						○		
ソフトウェア開発実習						○		
工学基礎							○	
卒業研究							○	○

表-6 J07-SEの年次進行の例

## 年次進行の例

J07-SEではカリキュラムモデルとして、年次進行の例を提示した。この年次進行は、3つの視点からボトムアップ的な年次進行を意図している。1つ目として、前半の年次で情報科学基礎科目を学ぶことでシステムや計算機、情報技術を理解し、後半の年次にSE科目を学ぶ素地を養う。2つ目として、学生にとって具体的にイメージしやすいプログラムコードの作成から講義を始め、モデル化と要求開発や形式手法など抽象度の高い講義を後半に行うことで、抽象化の必要性を理解させ適切な抽象化を習得させる。3つ目として、構築など個人での作業を前半に習得させ、プロセスやマネジメントなどチームやプロジェクトでの作業を後半に学ぶことで、チームやプロジェクトでの開発に必要な基本概念や技法の重要性を認識させる。これらによって、スムーズかつ適切な理解を促すことが期待できる。

ただしJ07-SEの年次進行例はあくまで例示であり、カリキュラム開発時には学科やコースの特色に応じて適切な年次進行を策定する必要がある。表-6にJ07-SEの年次進行の例を示す。なお例において、講義科目と実

習科目の履修に3年半で約500時間かけることを想定している。

## おわりに

本稿では、ソフトウェアエンジニアリングを取り巻く状況、J07-SEカリキュラムモデルの全体像、知識項目および科目、年次進行を概説した。今後の取り組みとしては、J07-SEを基にしたカリキュラム例の策定、複数の大学や大学院のSEカリキュラムの比較、ITSS/ETSS/UISSなどとの連携が考えられる。多面的な開発やマネジメントのスタイルを理解できるようなPBLの例を提示することも必要だろう。また産業界や学会からの意見を募り、さらに実践的かつ骨太なカリキュラムを目指して改善していくとともに、大学院修士課程を含む6年でのカリキュラムモデルの検討も視野に入れたい。

J07-SEを策定した情報処理学会ソフトウェアエンジニアリング教育委員会のメンバを以下に示す。このメンバは産学間のバランスを保ち実践と理論の協調を図るとともに、世代間のバランスを保ち継続的な活動を可能にすることを意図している。また情報処理学会ソフトウェ

ア工学研究会 (SIGSE) と密接に連携しており、先端的な技術動向を広く普及啓蒙することが視野に入っている。

J07-SE 策定にはこのメンバの献身的活躍が不可欠なものであった。記して謝意を表したい。

委員長：阿草清滋(名古屋大)

幹事：羽生田栄一(豆蔵, 情報処理学会ソフトウェア工学研究会主査), 玉井哲雄(東大), 佐伯元司(東工大), 深澤良彰(早大), 榊原彰(日本 IBM), 沢田篤史(南山大), 鷺崎弘宜(早大), 西康晴(電通大)

委員：青木利晃(北陸先端大), 飯田元(奈良先端大), 石川冬樹(NII), 位野木万里(東芝ソリューション), 大西淳(立命館大), 大森久美子(NTT 情報流通基盤総合研究所), 片山徹郎(宮崎大), 小林隆志(名古屋大), 田口研治(NII), 野中誠(東洋大), 藤井拓(オーグス総研), 松下誠(阪大), 山本里枝子(富士通研)

#### 参考文献

- 1) Pressman, R. S. : Software Engineering – A Practitioner’s Approach, McGraw-Hill (2005).
- 2) ISO/IEC/JTC1/SC7 : ISO/IEC TR 19759 : 2005, Software Engineering – Guide to the Software Engineering Body of Knowledge (SWEBOK), ANSI (2007).
- 3) 情報処理推進機構 : IT スキル標準 V3 (2008). <http://www.ipa.go.jp/jinzai/its/>
- 4) 情報処理推進機構ソフトウェア・エンジニアリング・センター : 組込みスキル標準 ETSS2007 (2007). <http://sec.ipa.go.jp/download/report.php>
- 5) 経済産業省 : 情報システムユーザースキル標準～IS 機能の可視化による組織力向上のために～ Ver1.1 (2007). [http://www.meti.go.jp/policy/it\\_policy/jinzai/](http://www.meti.go.jp/policy/it_policy/jinzai/)

- 6) The Joint Task Force on Computing Curricula, IEEE Computer Society, ACM : Software Engineering 2004, Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, A Volume of the Computing Curricula Series (2004). <http://sites.computer.org/ccse/>  
(平成 20 年 6 月 6 日受付)

#### 阿草清滋(正会員)

agusa@is.nagoya-u.ac.jp

名古屋大学教授。所属は大学院情報科学研究科情報システム学専攻ソフトウェア論講座。専門はソフトウェア工学で, depository centered な高信頼性 Web アプリケーション開発手法に関する研究を進めている。

#### 西 康晴(正会員)

nishi@se.ucc.ac.jp

電気通信大学講師。所属は電気通信学部システム工学科。専門はソフトウェア工学で, テストやプロセス改善, 品質, リスクマネジメントなどの研究と教育を進めている。ソフトウェアテスト技術振興協会 (ASTER) 理事長, 組込みソフトウェア管理者・技術者育成研究会 (SESSAME) 副理事長などを務める。

#### 沢田篤史(正会員)

sawada@nanzan-u.ac.jp

南山大学教授。所属は数理情報学部情報通信学科。現在, 大学院の IT スペシャリストコースを担当。専門はソフトウェア工学で, 組込みシステムのためのソフトウェア開発方法論や開発支援環境に興味を持って研究を進めている。

#### 鷺崎弘宜(正会員)

washizaki@waseda.jp

早稲田大学准教授。所属は基幹理工学部情報理工学科。専門はソフトウェア工学で, 特に設計, 再利用, 品質保証を中心として高信頼ソフトウェアシステムの高効率開発技術, 知識体系, 方法論の研究と教育を進めている。

