

博 士 論 文

高度ソフトウェア専門技術者育成のための
教育研修システム開発方法論の研究

D2014MM002 古畑 慶次

指導教員 青山 幹雄

2015 年 2 月

南山大学大学院 数理情報研究科 数理情報専攻

A Development Methodology of Education and Training
Systems for Professional Software Engineers and
Its Evaluation

D2014MM002 Keiji Kobata

Supervisor Mikio Aoyama

February 2015

Graduate Program in Mathematical Sciences and Information Engineering
Graduate School of Mathematical Sciences and Information Engineering
Nanzan University

要約

企業において高度ソフトウェア専門技術者は、ソフトウェア開発を成功に導くばかりでなく、経営戦略達成のための戦力として必要不可欠である。企業では、一般に高度ソフトウェア専門技術者を研修により育成している。しかし、従来の研修カリキュラムの開発は研修担当者の知識や経験に依存しているため、企業が必要とする人材育成のゴールを達成できるか保証できない。また、研修は講義、演習が中心であるため、高度ソフトウェア専門技術者に必要な知識、スキルを効果的に習得できる教授法が取られているとは言い難い。

本研究では、教育研修をソフトウェア工学の新しい課題と捉え、研修開発にソフトウェア工学の方法論を適用する。研修をソフトウェアシステムとして捉えることにより、研修を教育研修システムとして開発する高度ソフトウェア専門技術者育成のための教育研修システム開発方法論を提案する。

本方法論では、教育研修システム開発をソフトウェア開発に対応させることにより、教育研修システム開発プロセスを研修要求定義、研修設計、研修実行、研修評価で構成する。

研修要求定義では、企業が必要とする高度ソフトウェア専門技術者の人材像、育成目標を、要求工学のゴール指向分析に基づいて、技術、実践、哲学について定義する人材育成ゴールモデルを提案する。人材育成ゴールモデルは、ゴール指向分析を用いて人材像を育成目標に展開したゴールモデルである。

研修設計では、カリキュラム、学習目標、教授法を設計する。カリキュラムは、人材育成ゴールモデルより導出する高度ソフトウェア専門技術者に必要な知識、スキルに基づいて設計する。学習目標は、**Bloom's Taxonomy** を育成目標に基づいて拡張することにより設定する。教授法については、カリキュラム設計で導出した高度ソフトウェア専門技術者に必要な知識、スキルの習得に効果的な教授法を設計し、学習目標に基づいてカリキュラムに適した教授法を選択するプロセスを提案する。

研修実行では、研修設計の成果物であるカリキュラム、学習目標、教授法に基づいて研修を実施する。研修評価では、研修結果を学習目標に対して評価する。

提案する開発方法論を企業の高度ソフトウェア専門技術者育成の教育研修システム開発に適用し、その結果から本提案方法論の有効性を示す。

Abstract

Professional software engineers are indispensable not only for the success of software projects but also for the achievement of business goals of a company. In the company, professional software engineers are generally developed through a special training program. However, the current practice of the curriculum designed for the program is rather ad hoc, and the teaching methods are mainly lectures and exercises. The performance of the educational program is hard to evaluate.

In this thesis, the author proposes a development methodology of education and training system for professional software engineers, as we consider the development of education and training system as a new challenge of software engineering. The author proposes a model of educational goals of professional software engineers, which consists of engineering, practice, and philosophy goals, derived by goal-oriented analysis in requirement engineering. The knowledge and skills necessary for professional software engineers are derived from the goals, and the curriculum is designed. The author proposes the process of the design of the teaching method from the goals using software design approach.

The author demonstrates the effectiveness of the development methodology by applying the methodology to the development of an education and training system for professional software engineers at the company.

目次

1	はじめに	8
1.1	研究の背景.....	8
1.2	研究の目的.....	8
1.3	本論文の構成.....	8
2	研究課題	9
2.1	教育研修システム開発方法論.....	9
2.2	人材要求定義.....	9
2.3	カリキュラム設計.....	10
2.4	教授法設計.....	10
3	関連研究	11
3.1	教育システム設計.....	11
3.2	カリキュラム設計.....	12
3.2.1	大学におけるカリキュラム設計.....	12
3.2.2	企業におけるカリキュラム設計.....	13
3.2.3	CDIO.....	14
3.3	教授法設計.....	16
3.3.1	最新の教授法.....	16
3.3.2	教授法設計.....	16
3.4	学習目標.....	18
4	教育研修システム開発方法論	19
4.1	教育研修開発と教育研修システム開発.....	19
4.2	教育研修システム開発方法論のフレームワーク.....	20
4.2.1	研修要求定義.....	21
4.2.2	研修設計.....	21
4.2.3	研修実行.....	22
4.2.4	研修評価.....	22
5	教育研修システムの開発プロセス	23
5.1	教育研修システム開発プロセス.....	23
5.2	研修要求定義プロセス.....	24
5.3	研修設計プロセス.....	24
5.4	研修実行プロセス.....	25
5.5	研修評価プロセス.....	25
6	研修要求定義	26
6.1	教育研修システムのステークホルダ.....	26
6.2	人材育成ゴールモデル.....	27
6.2.1	人材像.....	27
6.2.2	戦略ゴール.....	27
6.2.3	戦術ゴール.....	28

6.3	研修要求定義プロセス	28
6.4	人材像の定義.....	29
6.5	戦略ゴールの設定.....	29
6.5.1	戦略マップ.....	29
6.5.2	戦略ゴールの設定方法.....	29
6.5.2.1	達成目標の導出.....	29
6.5.2.2	戦略ゴールの導出.....	30
6.6	戦術ゴールの設定.....	31
6.6.1	戦略ゴールの展開.....	31
6.6.2	戦術ゴールの導出.....	31
7	研修設計.....	35
7.1	カリキュラム設計.....	35
7.1.1	カリキュラム設計プロセス.....	35
7.1.2	技術のカリキュラム設計.....	37
7.1.3	実践のカリキュラム設計.....	37
7.1.4	哲学のカリキュラム設計.....	39
7.2	学習目標設計.....	41
7.2.1	学習目標設計プロセス.....	41
7.2.2	技術の学習目標設計.....	43
7.2.2.1	到達目標の設定.....	43
7.2.2.2	学習目標の設計.....	43
7.2.3	実践の学習目標設計.....	44
7.2.3.1	到達目標の設定.....	44
7.2.3.2	学習目標の設計.....	44
7.2.4	哲学の学習目標設計.....	44
7.2.4.1	到達目標の設定.....	44
7.2.4.2	学習目標の設計.....	45
7.3	教授法設計.....	46
7.3.1	教授法設計のアプローチ.....	46
7.3.2	教授法設計プロセス.....	47
7.3.3	教授法の導出.....	48
7.3.4	教授法の設計.....	51
7.3.5	学習目標へのマッピング.....	54
7.3.6	教授法の選択.....	55
8	教育研修システムの評価方法.....	56
8.1	評価方法の枠組み.....	56
8.2	カリキュラム評価.....	57
8.2.1	研修設計時の評価.....	57
8.2.1.1	評価方法.....	57
8.2.1.2	評価手順.....	57
8.2.2	研修実行時の評価.....	58
8.2.2.1	評価方法.....	58
8.2.2.2	評価手順.....	58

8.3	教授法評価	59
8.3.1	研修設計時の評価.....	59
8.3.1.1	評価方法.....	59
8.3.1.2	評価手順.....	59
8.3.2	研修実行時の評価.....	59
8.3.2.1	評価方法.....	59
8.3.2.2	評価手順.....	59
8.4	育成評価	59
8.4.1	評価方法.....	59
8.4.2	評価手順.....	60
9	実際の研修への適用	61
9.1	適用した研修.....	61
9.1.1	第1期教育研修システム.....	61
9.1.2	第2期教育研修システム.....	62
9.1.3	第3期教育研修システム.....	63
9.2	カリキュラム.....	64
9.2.1	技術のカリキュラム.....	64
9.2.2	実践のカリキュラム.....	65
9.2.3	哲学のカリキュラム.....	67
9.3	教授法.....	68
9.3.1	技術の教授法.....	68
9.3.2	実践の教授法.....	70
9.3.3	哲学の教授法.....	71
9.3.3.1	リーダーシップの教授法.....	71
9.3.3.2	コミュニケーションの教授法.....	72
9.4	研修計画	74
9.4.1	技術の研修計画.....	74
9.4.2	実践の研修計画.....	74
9.4.3	哲学の研修計画.....	74
9.5	実際の研修	76
10	評価結果.....	78
10.1	カリキュラム評価.....	78
10.1.1	研修設計時の評価.....	78
10.1.2	研修実行時の評価.....	79
10.2	教授法評価	79
10.2.1	研修設計時の評価.....	80
10.2.2	研修実行時の評価.....	80
10.2.2.1	技術の教授法評価.....	80
10.2.2.2	哲学の教授法評価.....	81
10.3	育成評価.....	82
11	考察.....	83
11.1	教育研修システム開発方法論の意義	83

11.2	人材像のゴール分析方法の意義.....	83
11.3	カリキュラム設計方法の意義.....	83
11.4	教授法設計方法の意義.....	84
12	今後の課題.....	85
12.1	教授法設計方法の検討.....	85
12.2	研修のスケジュールリング技術の確立.....	85
12.3	開発方法論の他分野への適用と修了生の評価.....	85
13	まとめ.....	86
	謝辞.....	87
	参考文献.....	88
	研究業績.....	90

1 はじめに

1.1 研究の背景

企業における次世代の技術開発リーダーの育成は重要な経営課題である。特に、大規模化、複雑化するソフトウェア開発においては、ソフトウェア工学を習得し、現場の問題解決を推進できる高度ソフトウェア専門技術者は必要不可欠である。このため企業では、ソフトウェア開発を成功に導き、経営課題達成に貢献できる高度ソフトウェア専門技術者の育成は急務となっている[25]。

一般に、企業における高度ソフトウェア専門技術者は研修により育成している。しかし、研修カリキュラムは研修担当者の知識や経験に依存し、研修生の満足度や知識の理解度で研修を評価しているため、企業が必要とする人材を育成できているか評価できない。また、カリキュラムや研修結果を適切に評価できないため、カリキュラムの改善が進んでいない。研修における教授法については講師に一任されるため、伝統的な教授法である講義、演習中心の研修となり、研修時間内で高度ソフトウェア専門技術者に必要な知識、スキルを習得するには限界がある。以上のように高度ソフトウェア専門技術者を育成する枠組みは確立しているとは言い難い。

1.2 研究の目的

研修開発における学習目標、カリキュラム、教授法を教育研修システムと捉えることにより、教育研修システムの開発はソフトウェアシステムの開発に帰着できる。本稿では、研修開発をソフトウェアシステム開発として捉えることにより、企業が必要とする高度ソフトウェア専門技術者を育成する教育研修システムの開発方法論を提案する。提案する開発方法論を企業における高度ソフトウェア専門技術者の育成研修に適用し、本開発方法論の有効性を示す。

提案する開発方法論では、教育研修システム開発プロセスを研修要求定義、研修設計、研修実行、研修評価と定義し、研修開発にソフトウェア工学の原理を適用して教育研修システムを開発する。研修要求定義では、企業が必要とする人材像を要求工学のゴール指向分析を適用し、技術、実践、哲学に関する育成目標に展開する。研修設計では、学習目標、カリキュラム、教授法を設計する。学習目標は、Bloom's Taxonomyを拡張して作成する。カリキュラムは、育成目標より導出する高度ソフトウェア専門技術者に必要な知識、スキルに基づいて設計する。教授法の設計は、ソフトウェア設計の考え方を応用し、育成目標を満たす教授法を組み合わせることにより、必要な知識、スキルを習得できる教授法を設計する。

1.3 本論文の構成

本論文の構成を以下に示す。

第2章は研究課題について説明する。第3章は関連研究について示す。第4章は教育研修システム開発方法論を提案する。第5章は教育研修システムの開発プロセスを示す。第6章は教育研修システムにおける研修要求定義、第7章は研修設計、第8章は教育研修システムの評価方法について説明する。第9章は提案する開発方法論を適用した企業での研修について示す。第10章は提案する開発方法論を適用した研修の評価結果である。第11章は提案する開発方法論についての考察を述べる。第12章は今後の課題、第13章は本稿のまとめを述べる。

2 研究課題

2.1 教育研修システム開発方法論

本稿で対象とする教育研修システムは、企業における高度ソフトウェア専門技術者の育成を目的としている。高度ソフトウェア専門技術者は、企業戦略達成に必要な課題解決に貢献できる人材である。従って、高度ソフトウェア専門技術者への要求は企業固有であり、育成目標とする人材像に対する要求は複雑である。

研修開発は、研修コースを開発するプロセスである教育システム設計に従って進めるのが一般的である。しかし、教育システム設計は、分析、設計、開発、実施、評価の各フェーズを含んだ包括的な研修開発の基本概念である。そのため、各フェーズでの進め方は担当者の知識や経験に依存する。高度ソフトウェア専門技術者に対する要求は企業固有で複雑なため、従来の方法論でカリキュラム、学習目標、教授法を設計するには限界がある。

本稿では、研修開発をソフトウェアシステム開発と捉え、ソフトウェア工学の基本原理を応用することにより高度ソフトウェア専門技術者育成のための教育研修システム開発方法論を提案する。図 2.1 に示すように教育研修システム開発方法論の中核を成すのは、人材要求定義、カリキュラム設計、教授法設計である。それぞれの研究課題について以下に説明する。



図 2.1 開発方法論の研究課題

2.2 人材要求定義

高度ソフトウェア専門技術者は、企業における経営課題達成のために必要な人材である[25]。企業の経営課題は多種多様であり、解決へのアプローチも企業理念や経営戦略により大きく異なる。さらに、人材への要求は、単なる技術知識やスキルだけでなく、問題分析力や課題形成力、課題解決力、さらにはマネジメントやリーダーシップの能力など多岐にわたり、それらは互いに深く関係している。以上のように、高度ソフトウェア専門技術者の人材像に対する要求は、企業により異なり複雑である。従って、高度ソフトウェア専門技術者を育成する教育研修システムを開発するためには、目標とする人材像に対する要求を定義する必要がある。

しかし、実際の教育研修システム開発ではカリキュラム作成が優先し、人材像の要求定義について十分議論されないまま研修を実施することが多い。そのため、目標とする人材像とカリキュラムの関係が明確にできず、開発した研修により企業が必要とする人材を育成できているか評価できない。また、教育研修システムの評価、改善も進んでいないのが現実である。

本稿では、要求工学におけるゴール指向分析を適用し、高度ソフトウェア専門技術者に対する要求を体系的に定義するモデルを提案する。定義した要求である育成目標に基づいて、カリキュラム設計、学習目標設計、教授法設計を実施する。

2.3 カリキュラム設計

高度ソフトウェア専門技術者の育成には、目指す人材像の要求を満足するカリキュラムが必要である。一般に知られているカリキュラムの設計方法は、個別の教育や小規模な教育研修システムを対象にしている。そのため、本稿で対象とする高度ソフトウェア専門技術者の育成を目的とする複雑で大規模な教育研修システムのカリキュラム設計へ適用するのは困難である。

多くの企業では、社内の有識者や研修担当者の知識や経験に基づいて高度ソフトウェア専門技術者の研修カリキュラムを作成している。そのため個々の研修はそれぞれの目標を達成しても、カリキュラム全体を通して、必要とする人材を育成できるか保証できない。

本稿では、人材像、育成目標から導出する高度ソフトウェア専門技術者に必要な知識、スキルに基づいてカリキュラムを設計する方法を提案する。育成目標である人材像と習得すべき知識、スキルの関係を体系的に把握することによりカリキュラムの設計方針を明確にし、包括的にカリキュラムを作成できる設計手法を提案する。

2.4 教授法設計

高度ソフトウェア専門技術者を育成する教育研修システムでは、目標とする人材像に対する要求が複雑で企業ごとに異なるため、こうした要求に応えられる教授法が必要となる。従来は、教育研修を担当する講師に一任することが多かったため、教授法は研修内容を講義で説明し演習で知識の定着を図る講義、演習が中心であった。講義、演習を中心した教授法は、知識の習得には適しているが、高度ソフトウェア専門技術者のように経営課題に直結する難易度の高い問題の分析や解決方法の提案が求められる技術者の育成には効果的とは言い難い。

近年では、反転授業[2][9]やアクションラーニング[17]、PBL(Problem-Based-Learning)[20]といった教育研修効果の高い新しい教授法の形態が提案されている。こうした教授法は技術についての知識やスキルの習得を目的とした教育研修では効果が報告されている。しかし、高度ソフトウェア専門技術者に求められる高度な問題解決力や課題形成力、リーダーシップを習得する教育研修システムには十分対応できていない。

本稿では、カリキュラム設計で導出した高度ソフトウェア専門技術者に必要な知識、スキルの習得に効果的な教授法を、育成目標から導出する複数の教授法を組み合わせることにより設計する。さらに、カリキュラムの学習目標に適した教授法を設計した教授法より選択するプロセスを提案する。

3 関連研究

本研究の関連研究として、教育システム設計についての研究がある。また、教育研修システム開発を構成するカリキュラム設計、教授法設計、学習目標の研究について述べる。

3.1 教育システム設計

研修開発の方法論として、インストラクションデザインにおける教育システム設計が知られている[8]。教育システム設計とは、教育システムを開発するプロセスを指す。教育システムとは、学習を促進するために用いられる資源(リソース)や手続きの順序を示している。教育システムは、狭い範囲での技術的な研修コースから、広義には学習者に着目した学習環境を提供するツールに至るまで多様な形式がある。

インストラクションデザインにおける教育システム設計プロセスの最も基本的なモデルは、分析(Analysis)、設計(Design)、開発(Development)、実施(Implementation)、評価(Evaluation)の 5 つの段階を含む。図 3.1 のモデルは、その 5 つの構成要素からそれぞれの頭文字をとり、ADDIE(Analysis, Design, Development, Implementation, Evaluation)モデルと呼ばれている。一般に教育研修は、ADDIE モデルが示すように、分析、設計、開発、実施、評価のプロセスに従って開発、改善する。

分析では、教育研修の目的や学習者、組織の課題、業務内容、必要な知識などの要件を洗い出す。設計では分析結果をもとに、教育研修で用いる教材やツールなどの設計を行う。設計した教材やツールは開発で作成する。実施では、教育研修を実際に行う。評価では、教育研修全体や教材などの問題点を洗い出し、改善を行う。ADDIE モデルに代表される教育システム設計モデルは、視覚的な表現によって教育システム設計のシステムの的なアプローチを示したものである。

ADDIE モデルをはじめとする教育システム設計プロセスを実際の教育システム設計に適用するためには、各プロセスにおける問題を解決するための方法論が必要となる。単一の技術科目の研修のような小規模な教育システムは、ADDIE モデルの適用により開発プロセスや各プロセスにおける成果物が明確になり、教育システム設計の修正、改善が可能である。

しかし、本稿で対象とする高度ソフトウェア専門技術者育成のように、企業によって育成目標である人材像が異なり、人材像への要求が複雑である教育研修の開発は、従来の教育システム設計では限界がある。高度ソフトウェア専門技術者に対する要求を体系的に扱い、各プロセスでの問題解決の方法を含む研修の開発方法論が必要である。

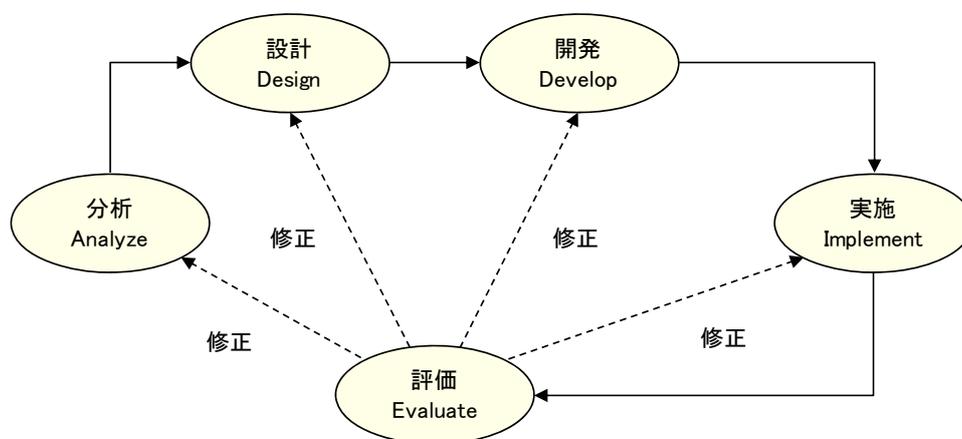


図 3.1 ADDIE モデル

3.2 カリキュラム設計

一般に、人材育成のためのカリキュラム設計は、教育目標を明確にし、その目標を達成するカリキュラムを編成する方法が知られている[7][11][18][19][22][23]。また、カリキュラムの評価方法については理論的枠組みが確立されている[12]。しかし、複雑な人材要求が求められる高度ソフトウェア専門技術者のカリキュラム設計に、一般的なカリキュラムの設計方法や評価方法を適用するには限界がある。企業が求める人材像は複雑であるため、従来の設計方法では人材像とカリキュラムの関係を明確に示すことができなかった。

カリキュラム設計の関連研究として、大学におけるカリキュラム設計[1]、企業におけるカリキュラム設計[21]、次世代のエンジニアを育成するための工学教育カリキュラムの枠組みである CDIO[6]について述べる。

3.2.1 大学におけるカリキュラム設計

カリキュラム設計の事例として、大学におけるソフトウェア工学科のカリキュラム設計がある[1]。図 3.2 は、ソフトウェア工学科のカリキュラムに対する要求である。ソフトウェア工学科の主なステークホルダを特定し、ステークホルダが期待する人材像に基づいてカリキュラム要求を導出している。導出した要求から、ソフトウェア工学カリキュラムに関する知識体系である SE2004, J07-SE/CE, SWEBOK[5]を活用し、カリキュラムを設計する。SE2004 は、カリキュラム標準 CC2001 の 1 つであるソフトウェア工学カリキュラムである。J07-SE/CE は、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)を構成する SE(ソフトウェア工学 : Software Engineering), CE(組込みシステム : Computer Engineering)のカリキュラムである。

大学におけるカリキュラム設計は、高度専門職に対する技術の知識、スキルの教育が目的であるため、経営課題の達成という視点が入っていない。本稿で対象とする高度ソフトウェア専門技術者は、企業の経営課題を解決するための人材である。求める人材像には、技術知識やスキルだけでなく、企業の課題に対応するための要求を定義する必要がある。

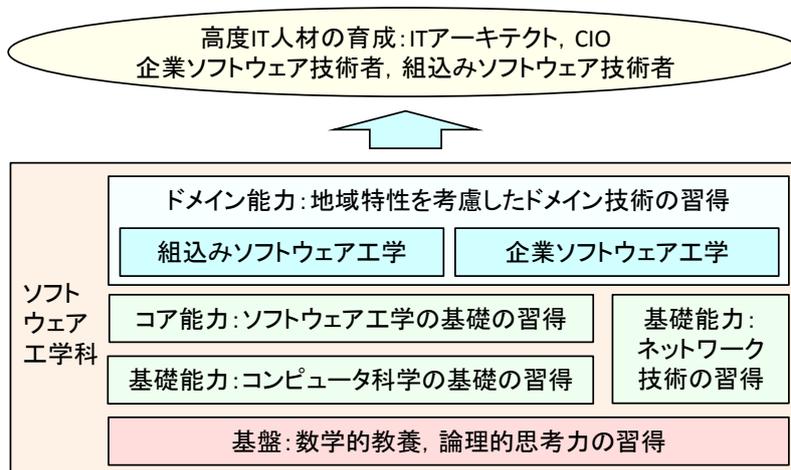


図 3.2 ソフトウェア工学科のカリキュラムに対する要求

3.2.2 企業におけるカリキュラム設計

企業における高度ソフトウェア専門技術者の育成事例としてアーキテクト資格制度がある[21]。表 3.1 は、アーキテクトとしての資格を認定するために、アーキテクトに求められるコンピテンシーをまとめた表である。アーキテクトのカリキュラムは、表 3.1 に基づいて設計する。

アーキテクトのコンピテンシーをまとめた背景には次のような状況がある。開発対象であるシステムは年々複雑化しており、ソフトウェアの開発規模は急激に増加している。それに加え、開発期間の短縮も求められている。こうしたソフトウェア開発に対応するために、アジャイル開発手法やモデル駆動開発、サービス指向アーキテクチャなどの活用が課題になっている。

上記の現状より、アーキテクトはソフトウェア開発の中で幅広い役割と技術が求められている。アーキテクチャ技術はアーキテクトに必須な技術である。それ以外にアーキテクチャに関連の強い要求工学やソフトウェアプロセス、テストと品質、ビジネスと戦略についての技術が必要となる。また、実際のソフトウェア開発では、チーム間の連携、協調がプロジェクトの成否を分けるため、コミュニケーションを含むリーダーシップの観点もコンピテンシーに考慮されている。以上のように、表 3.1 のコンピテンシーは、実際のソフトウェア開発プロジェクトでアーキテクトに求められる技術、スキルを分析して導出している。

本稿で対象とする高度ソフトウェア専門技術者には、経営課題を解決するためのコンピテンシーを考慮する必要がある。高度ソフトウェア専門技術者は経営課題を解決するための人材である。高度ソフトウェア専門技術者の育成には、経営課題を解決するコンピテンシーを定義し、コンピテンシーに必要な知識、技術、スキルを習得する体系的なカリキュラム設計が必要となる。

表 3.1 アーキテクトに必要なコンピテンシー

Topic	Subtopic
Business and strategy	Business case understanding
	Global development
	Legal issues
Requirements engineering	Product management
	Requirements engineering
Software architecture and development	Product line engineering
	Architecture and design
	SW design methods
	Configuration management
Testing and quality	Quality assurance
	Test processes and methods
Software processes	System development
	SW development processes
Leadership	Project management
	Social skills

3.2.3 CDIO

CDIO は次世代のエンジニアを育成するための工学教育カリキュラムの枠組みであり、現実のシステムや製品開発における設計過程を背景にした教育プログラムのガイドラインである[6]。この枠組みで学習することで CDIO が定義する“考え出す力 (Conceive)”，“設計する力 (Design)”，“実行する力 (Implement)”，“運営する力 (Operation)” を身に付けることを目的としている。

CDIO は、2000 年に MIT とスウェーデンの 3 つの大学が協力し考案した工学教育である。1990 年代後半、大学では、企業の現場では一般的とは言えない知識偏重型の工学教育が世界の主流であった。それに対し、企業や社会が、技術者に必要な技術や態度、自発性、創造性、技能、リーダーシップ、動機づけ、そして、チームワークなどを身につける教育の重要性から、工科系大学の教育プログラムを改革することを狙いとして CDIO が発案された。

CDIO の目標を表 3.2 に示す。従来、中心であった技術の基礎知識の習得だけでなく、製品やシステムの創造に対するリーダーシップや、社会における研究や技術開発の戦略的な影響と重要性の理解まで含んでいる。

表 3.2 CDIO の目標

<p>The Goals</p> <ol style="list-style-type: none">(1) Master a deeper working knowledge of technical fundamentals(2) Lead in the creation and operation of new products, processes, and systems(3) Understand the importance and strategic impact of research and technological development on society

CDIO のシラバスを表 3.3 に示す。CDIO は、従来実施してきた知識教育に加え、「Conceive (考え出す) - Design (設計する) - Implement (実行する) - Operate (運営する)」というプロセスを通じて、知識を活用してシステムや製品の開発ができる技術者の育成を目指している。CDIO は、次世代の技術者を育成する教育のフレームワークの 1 つであり、システムや製品開発における工学の基礎教育を提供するのが狙いである。

CDIO は、大学における工学教育カリキュラムのフレームワークを提供しているが、その内容は、次世代の高度専門職に対する技術の知識、スキル、態度についての教育プログラムのガイドラインの域を出ない。教育のプロセスは参考になるが、高度ソフトウェア専門技術者の育成には、企業ごとに異なる経営戦略の観点を導入し、必要な人材の育成を可能とするカリキュラム設計が必要である。

表 3.3 CDIO のシラバス

1 DISCIPLINARY KNOWLEDGE AND REASONING	
	1.1 KNOWLEDGE OF UNDERLYING MATHEMATICS AND SCIENCE
	1.2 CORE ENGINEERING FUNDAMENTAL KNOWLEDGE
	1.3 ADVANCED ENGINEERING FUNDAMENTAL KNOWLEDGE, METHODS AND TOOLS
2 PERSONAL AND PROFESSIONAL SKILLS AND ATTRIBUTES	
	2.1 ANALYTICAL REASONING AND PROBLEM SOLVING
	2.2 EXPERIMENTATION, INVESTIGATION AND KNOWLEDGE DISCOVERY
	2.3 SYSTEM THINKING
	2.4 ATTITUDES, THOUGHT AND LEARNING
	2.5 ETHICS, EQUITY AND OTHER RESPONSIBILITIES
3 INTERPERSONAL SKILLS: TEAMWORK AND COMMUNICATION	
	3.1 TEAMWORK
	3.2 COMMUNICATIONS
	3.3 COMMUNICATIONS IN FOREIGN LANGUAGES
4 CONCEIVING, DESIGNING, IMPLEMENTING AND OPERATING SYSTEMS IN THE ENTERPRISE, SOCIETAL AND ENVIRONMENTAL CONTEXT - THE INNOVATION PROCESS	
	4.1 EXTERNAL, SOCIETAL AND ENVIRONMENTAL CONTEXT
	4.2 ENTERPRISE AND BUSINESS CONTEXT
	4.3 CONCEIVING, SYSTEMS ENGINEERING AND MANAGEMENT
	4.4 DESIGNING
	4.5 IMPLEMENTING
	4.6 OPERATING
	4.7 LEADING ENGINEERING ENDEAVORS
	4.8 ENTREPRENEURSHIP

3.3 教授法設計

3.3.1 最新の教授法

一般に教授法は、研修を実施する上で講師がとるべき効果的な教育方法として知られている。近年では、アクションラーニング[17]やPBL(Problem-Based Learning)[20]、反転授業(Inverted Classroom)[2][9]といった教授法が注目されている。こうした新しい教授法のソフトウェア工学研修への適用が課題となっている。アクションラーニング、PBL、反転授業について以下に述べる。

(1) アクションラーニング

図 3.3 にアクションラーニングの学習プロセスを示す。アクションラーニングは、グループで現実の問題に対処し、その解決策を立案、実施していく過程で生じる実際の行動とその振り返り、気付きを通じて、個人、そしてグループ、組織の学習する力を養成するチーム学習法である。組織の現実の問題を取り上げながら、その解決策に対処していくため、問題解決と個人の能力開発、組織開発を同時に達成できる。

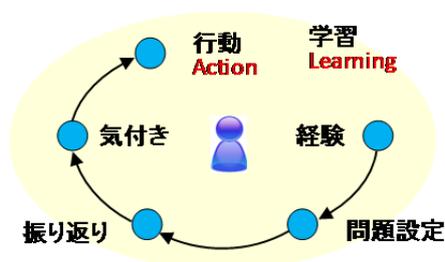


図 3.3 アクションラーニング

(2) PBL

PBL は、少人数グループによる問題発見解決型（事例解決型、事業課題解決型）の学習方法である。そのプロセスには、グループ討議、活動記録の作成、自己学習、成果報告までを含む統合的・創造的な学習に主眼を置く実践形式の学習方法である。PBL では、講師はまず研修生に課題を出す。このとき、幾つかのインストラクションは提示するが、研修生が自主的に学習して研修の準備をする。1つのテーマに対して幾つかのグループに分かれて作業を分担し、講義は行いが、研修生同士の質疑応答で研修は進行する。研修生が学習の主体であり、講師は学習を支援する立場になる。

(3) 反転授業

反転授業は、研修生が新たな学習内容を、通常は自宅でビデオ授業を視聴して予習し、研修では講義は行わず、従来であれば宿題とされていた課題について講師が個々の研修生に合わせた指導を与えたり、研修生が他の研修生と協働しながら取り組む形態の教授法である。研修時間は、学生が予習で得た知識を応用して問題を解き、実習する。

3.3.2 教授法設計

会計学の教育では、学習目標を達成するために、複数の教授法を組み合わせる方法が効果を上げている[3]。提案されている教授法設計法は、会計学の学習目標を特定し、それらを Gagné's Taxonomy に基づいて Verbal Information, Intellectual Skills-Defined Concepts, Intellectual Skills-Rules and Higher-Order Rules,

Cognitive Strategies の 4 つの学習目標に分類する。次に、会計学の講師が実施可能な教授法を研修生の学習の観点から定義する。表 3.4 に定義した教授法を示す。さらに、講師が学習目標を達成するために必要な条件を教育認知心理学の観点から導出する。表 3.5 に導出した必要条件を示す。そして、分類した 4 つの学習目標のそれぞれに対して、学習目標達成に必要な条件に貢献できる教授法を表 3.4 の実施可能な教授法より決定する。

表 3.4 実施可能な教授法

<ol style="list-style-type: none">1. Read Text2. Read worked-out example problems3. Listen to lecture/watch video4. Watch demonstration5. Listen to and participate in interactive lecture6. Answer short objective questions7. Write and answer questions8. Work short numerical problems9. Work longer, unstructured cases and problems10. Discuss issues with other students11. Conduct research12. Make oral presentations and answer questions13. Participate in demonstrations
--

表 3.5 学習目標達成に必要な条件

<ol style="list-style-type: none">1. Describe expected performance2. Facilitate recall of well-organized knowledge base3. Deliver meaningfully organized material4. Facilitate elaboration of material5. Elicit expected performance; provide practice
--

会計学は、習得科目が明確であり、知識の習得が主な学習目標である。一方、高度ソフトウェア専門技術者に対する要求は企業ごとに異なり、求める人材像には知識習得を前提とした実践的なスキルや問題分析力、課題形成力、問題解決力、リーダーシップなどが求められる。従って、高度ソフトウェア専門技術者育成研修の教授法には、人材像に対する要求から学習目標を定義し、学習目標を達成する新たな教授法の組み合わせ方法を検討する必要がある。

3.4 学習目標

学習目標の設定基準には、教育目標の分類体系である Bloom's Taxonomy[4]を用いるのが一般的である。また、教育の評価にも Bloom's Taxonomy は適用されている[12]。

表 3.6 に Bloom's Taxonomy の全体構成を示す。Bloom's Taxonomy は、アメリカ心理学会(APA)が 1948 年に大学の試験に関わる研究者を集めて、試験問題を分類して互いのコミュニケーションの円滑化を図る目的で開始した 8 年間のプロジェクトで得られた成果である。教育の目標とする領域を「頭、心、体」の認知的領域、情意的領域、精神運動的領域の 3 領域に分け、それぞれに表 3.6 に示すレベル分けを提案した。

認知的領域に関する第 1 巻が 1956 年に公刊され、情意的領域に関する第 2 巻が 8 年後の 1964 年に公刊された。しかし、精神運動的領域についての第 3 巻はこの研究者グループの手によるまとめはなく、数人の研究者による提案が 1970 年代になされたが定説には至っていない。

表 3.6 Bloom's Taxonomy の全体構成

6.0	評価	Evaluation				
5.0	統合	Synthesis	個性化	Characterization	自然化	Naturalization
4.0	分析	Analysis	組織化	Organization	分節化	Articulation
3.0	応用	Application	価値づけ	Valuing	精密化	Precision
2.0	理解	Comprehension	反応	Responding	巧妙化	Manipulation
1.0	知識	Knowledge	受け入れ	Receiving	模倣	Imitation
		認知的領域	情意的領域		精神運動的領域	

実際の教育目標の設計では、Bloom's Taxonomy を拡張して実用的なツールとして活用するための方法が知られている[18]。しかし、実際の企業研修の現場では、Bloom's Taxonomy は研修設計や研修評価に十分適用されているとは言い難い。特に、高度ソフトウェア専門技術者の育成研修に関しては人材像に対する要求が複雑であるため、学習目標の作成には、育成目標とカリキュラム内容に合わせて Bloom's Taxonomy を拡張する方法を検討する必要がある。

4 教育研修システム開発方法論

提案する教育研修システム開発方法論は、研修開発におけるカリキュラム，学習目標，教授法を開発する方法論である。教育研修を教育研修システムと捉えることにより，研修開発はソフトウェアシステムの開発に帰着できる。そこで，ソフトウェア開発プロセスを研修開発に適用し，ソフトウェア工学の原理を適用することにより研修開発の問題を解決する。

4.1 教育研修開発と教育研修システム開発

企業における教育研修とは，競争力を高めるために企業戦略達成に必要な人材を育成することである。教育研修は，経営層や開発現場，人事部門のニーズから目標とする人材像を定義し，人材像から育成目標を作成する。そして，育成目標から学習目標を設定し，学習目標に基づいてカリキュラムを作成して教育研修を実施する。従って，教育研修の入力は，目標とする人材像，研修生であり，教育研修の出力は教育研修を受講した研修生である。

Valacich によるシステムの定義[26]に従えば，教育研修は，育成目標を達成するために，カリキュラム，教授法，育成目標などの関連するコンポーネントで構成された教育研修システムと考えることができる。

図 4.1 に教育研修システムを示す。教育研修システムの入力は，目標とする人材像と研修生であり，出力は研修を受講した研修生である。教育研修システムの制約条件は，期間，コスト，品質である。教育研修システムは，決められた期間と予算で実行しなければならない。さらに，その期間と予算の中で，カリキュラムに関する知識，スキルを研修生に習得させる必要がある。これらを満たす研修の品質が教育研修システムには求められる。

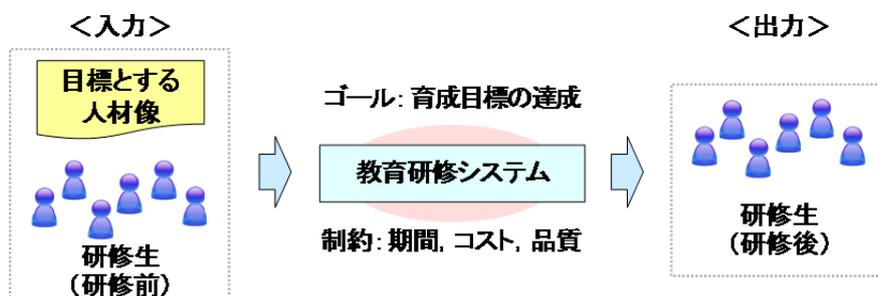


図 4.1 教育研修システム

表 4.1 は，教育研修システムを Valacich が定義する 9 つのシステム特性の観点について整理した表である。教育研修システムは，カリキュラム，教授法，研修運営部署等で構成し，育成目標を達成するためにそれぞれのコンポーネントが有機的に機能し，研修生に知識，スキルを提供する。教育研修システムには，限られた期間と予算の中で育成目標を達成する研修の品質が制約条件として求められる。

以上のように，研修を教育研修システムとして捉えることにより，教育研修システムの開発は，ソフトウェアシステムの開発に帰着できる。そこで，研修開発の対象である学習目標，カリキュラム，教授法を教育研修システムとして定義し，教育研修システムの開発方法論を提案する。

表 4.1 教育研修システムのシステム特性

No	Characteristics	教育研修システム
1	Components	カリキュラム, 教授法, 研修運営部署 等
2	Interrelated components	カリキュラム, 教授法間の関係
3	Boundary	教育研修システムと研修生, 研修生の所属組織との境界
4	Purpose	育成目標
5	Environment	研修の環境
6	Interfaces	教育研修システムとその実施環境とのインタフェース
7	Constraints	期間, 費用, 研修の品質
8	Input	目標とする人材像, 研修生(研修前)
9	Output	研修生(研修後)

4.2 教育研修システム開発方法論のフレームワーク

図 4.2 に提案する教育研修システム開発方法論のフレームワークを示す。提案する開発方法論は、研修開発をソフトウェアシステム開発と捉え、ソフトウェアの開発プロセスを適用して教育研修システムである学習目標、カリキュラム、教授法を開発する。

開発方法論のフレームワークは、研修要求定義、研修設計、研修実行、研修評価で構成する。

- (1) 研修要求定義： 人材像、育成目標を定義する
- (2) 研修設計： 学習目標、カリキュラム、教授法を設計する
- (3) 研修実行： 教育研修システムを研修生に対して実行する
- (4) 研修評価： 研修の実行結果を育成目標、学習目標で評価し、評価結果を研修要求定義、研修設計にフィードバックする

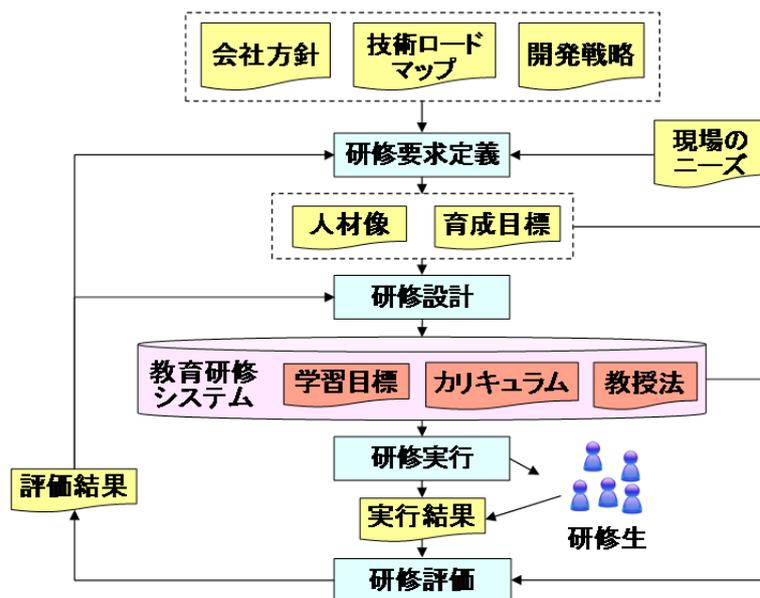


図 4.2 開発方法論のフレームワーク

4.2.1 研修要求定義

研修要求定義では、会社方針、技術ロードマップ、ソフトウェア開発戦略に対して、要求工学におけるゴール指向分析[16]を適用し、企業が必要とする高度ソフトウェア専門技術者の人材像、育成目標を設定する。

(1) 人材像

人材像は、中長期的視野に立って、企業の経営戦略達成に必要な人材を定義する。本稿で検討する高度ソフトウェア専門技術者は、経営的視点から企業の存続、発展に必要不可欠な人材である。高度ソフトウェア専門技術者は、経営戦略達成のための重要な戦力として位置づけることができる。

高度ソフトウェア専門技術者の人材像の設定では、最初に、企業の技術ロードマップに基づいてソフトウェア開発戦略を実施するに当たり必要不可欠な人材の役割を決定する。次に、会社方針に基づき、その役割に求められる要求を人材像として定義する。現場のヒアリングにより現場ニーズも人材像に反映する。

高度ソフトウェア専門技術者は、企業の経営課題を達成する人材であるので、課題解決に必要な技術知識の他、問題解決力やリーダーシップなどが求められる。そのため、人材像への要求は企業ごとに異なり、複雑となる。そこで、要求工学におけるゴール指向分析を適用することで、人材像に求められる要求を体系的に獲得し、育成目標として設定する。

(2) 育成目標

育成目標は人材像のサブゴールとして定義する。人材像と育成目標間の関係を構造的に表現することにより、高度ソフトウェア専門技術者に対する要求を体系的に獲得する。本稿では、筆者の所属する企業における高度ソフトウェア専門技術者育成研修の研究成果[14]から得られたモデルを適用し、人材像を、技術、実践、哲学の観点で展開して育成目標を設定する。

4.2.2 研修設計

研修設計では、研修要求定義で導出した人材像、育成目標に基づいて、教育研修システムを構成する学習目標、カリキュラム、教授法を設計する。

(1) 学習目標

学習目標は、人材像から展開した育成目標に対して、研修の具体的な目標を作成する。研修要求定義で設定した技術、実践、哲学の育成目標に対して、**Bloom's Taxonomy** を拡張することにより学習目標を設計する。

(2) カリキュラム

カリキュラムは、研修生が研修を通して習得すべき研修科目と研修計画である。カリキュラムの設計では、育成目標から高度ソフトウェア専門技術者に必要な知識、スキルを技術、実践、哲学について導出する。そして、導出した知識、スキルを満足する具体的な研修科目を設計する。育成目標である技術、実践、哲学の相互関係を考慮し、知識、スキルを習得する効果的な研修科目を設計する。

(3) 教授法

教授法は、カリキュラム設計で導出した高度ソフトウェア専門技術者に必要な知識、スキルに対して、育成目標から導出する教授法を適切に組み合わせることにより設計する。次に、設計した教授法を、育成目標である技術、実践、哲学の学習目標にマッピングし、学習目標の達成に適した教授法を選択する。

4.2.3 研修実行

研修実行では、研修設計で設計した学習目標、カリキュラム、教授法を教育研修システムとして実行する。教育研修システムの実行は、カリキュラムと教授法に基づいた研修の実施である。実行結果は、研修の実施結果と研修生の成果物である。

4.2.4 研修評価

研修評価では、研修の実行結果を育成目標、学習目標により評価し、評価結果は研修要求定義、研修設計に反映する。評価対象は、研修設計で作成したカリキュラムと教授法、本開発方法論を適用して開発した研修を受講した研修生である。

評価対象に対する評価方法は以下の通りである。

(1) カリキュラム評価

設計したカリキュラムが、学習目標を達成できているかを評価する。研修生の成果物を評価することによりカリキュラムを評価する。

(2) 教授法評価

設計した教授法の導入前後で、各研修科目が達成する学習目標について評価する。設計した教授法の効果は研修生のパフォーマンスにより評価する。

(3) 育成評価

本開発方法論により開発した研修を受講した研修生の能力を、研修生の成果物とパフォーマンスにより評価する。

5 教育研修システムの開発プロセス

5.1 教育研修システム開発プロセス

教育研修システムの開発プロセスを図 5.1 に示す。

企業では、ソフトウェアの開発企画部署、実際に研修を運営する研修運営部署、研修生の所属部署である研修対象部署をステークホルダとし、研修企画部署が中心となって教育研修システムを開発する。教育研修システムの開発プロセスは、研修要求定義プロセス、研修設計プロセス、研修実行プロセス、研修評価プロセスの4つのプロセスで構成する。

研修要求定義プロセスでは、研修企画部署が開発企画部署、研修対象部署から人材像についての要求を収集し、目標とする人材像を定義して育成目標を設定する。研修設計プロセスでは、研修企画部署が、育成目標に基づいて学習目標、カリキュラム、教授法を設計する。研修実行プロセスでは、研修運営部署が研修生を募集して研修を実施する。研修評価プロセスでは、研修企画部署が研修結果をレビューし、その結果を研修要求定義プロセス、研修設計プロセスにフィードバックする。

次に各プロセスの詳細について説明する。

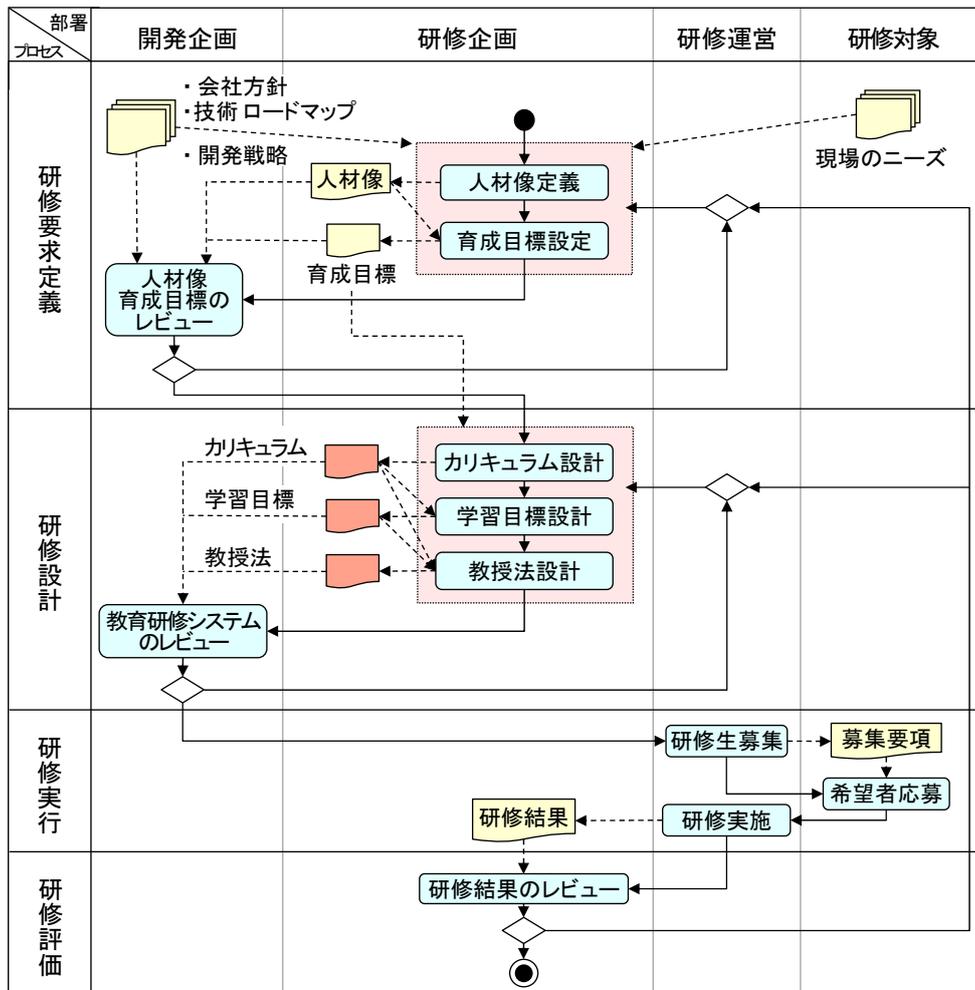


図 5.1 教育研修システム開発プロセス

5.2 研修要求定義プロセス

研修要求定義プロセスでは、会社方針、技術ロードマップ、開発戦略、開発現場のニーズから、企業で必要な高度ソフトウェア専門技術者の人材像、育成目標を研修企画部署が作成する。

(1) 人材像定義

人材像は、研修企画部署がソフトウェアの開発企画部署が保持する会社方針、技術ロードマップや開発戦略、研修対象部署の現場ニーズから必要とする人材についての要求を獲得し、中長期的視点に立って、企業に必要な高度ソフトウェア専門技術者の人材像を定義する。

会社方針は、企業活動において意志決定の判断基準となる経営方針、企業理念、事業を遂行する基本的な価値観や事業目的、社員の行動規範である。会社方針に基づいて、人材像に必要な価値観や考え方、取るべき行動を検討する。

技術ロードマップは、企業が今後開発していく製品やサービスの将来展望と開発に必要な技術要素を時系列でまとめたものである。開発戦略は、技術ロードマップに基づいた製品やサービス、それを支える要素技術についての開発方針と長期計画である。高度ソフトウェア専門技術者は経営戦略達成のための戦力であり、今後の企業にとって重要な技術開発や製品開発での活躍が期待されている。そこで、技術ロードマップ、開発戦略に基づいて、高度ソフトウェア専門技術者に求められる役割、必要とする技術領域を明確にし、人材像に反映する。

現場のニーズは、研修生の所属部署の現場マネージャや技術者とのヒアリングにより収集する。現状の開発における問題や今後の課題を解決するために必要な人材や技術について情報を集め、人材像に反映する。

(2) 育成目標設定

定義した人材像を育成目標に展開して整理する。育成目標は、教育研修システムである学習目標、カリキュラム、教授法の設計の入力情報となる。育成目標は、人材像から導出する定性的な目標と、それを達成する定量的な目標として設定する。

(3) 人材像、育成目標のレビュー

作成した人材像、育成目標は、ソフトウェア開発の開発企画部署とレビューを実施し、人材像、育成目標の会社方針、開発戦略に対する妥当性を確認する。修正が必要な場合は、研修企画部署が人材像、育成目標を再定義する。

5.3 研修設計プロセス

研修設計プロセスでは、研修要求定義プロセスで設定した育成目標から、教育研修システムを構成する学習目標、カリキュラム、教授法を研修企画部署が設計する。学習目標、カリキュラム、教授法は次の順で設計する。

(1) カリキュラム設計

カリキュラム設計は、育成目標から導出する高度ソフトウェア専門技術者に必要な知識、スキルを習得するカリキュラムを設計する。

(2) 学習目標設計

学習目標は、育成目標に基づいて Bloom's Taxonomy を拡張して作成する。学習目標は、教育研修における学習の到達レベルを 5 段階で設計する。

(3) 教授法設計

教授法設計は、育成目標から研修で実施可能な教授法を導出し、高度ソフトウェア専門技術者に必要な知識、スキルを効果的に習得できる教授法を設計する。次に、学習目標に設計した教授法をマッピングすることにより、カリキュラムに適した教授法を選択する。

(4) 教育研修システムのレビュー

設計した学習目標、カリキュラム、教授法は、ソフトウェアの開発企画部署とレビューを実施し、教育研修システムの育成目標に対する妥当性を確認する。修正が必要な場合は、研修企画部署が育成目標、カリキュラム、教授法を再設計する。

5.4 研修実行プロセス

研修実行プロセスでは、研修運営部署が研修設計プロセスで設計した教育研修システムに基づいて研修生を募集し、研修を実施する。

(1) 研修生募集

研修運営部署は、教育研修システムの学習目標、カリキュラム、教授法に基づいて研修の募集要項を作成し、研修対象部署に発行する。募集要項には、研修の狙い、到達目標、研修科目、研修スケジュール、募集人数などを記述する。

(2) 希望者応募

研修対象部署では、部署内で研修受講希望者を募集し、希望者は募集要項に従い研修に応募する。研修受講希望者が多い場合は、研修企画部署が受講希望者に対して選抜試験を実施し、研修生を決定する。

(3) 研修の実施

研修運営部署は、教育研修システムに従って研修を実施する。研修実施後は、研修の実施結果、研修生からの提出物を研修結果としてまとめる。

5.5 研修評価プロセス

研修評価プロセスでは、研修運営部署がまとめた研修結果を研修企画部署がレビューし、学習目標、育成目標に基づいて評価する。評価結果は、研修設計プロセスにフィードバックし、学習目標、カリキュラム、教授法の設計に反映する。また、人材像、育成目標についても評価結果に基づいて検討する。検討結果は研修要求定義プロセスにフィードバックし、人材像定義、育成目標設定に反映する。

6 研修要求定義

研修要求定義では、教育研修システムのステークホルダを明確にする。そして、ゴール指向分析を適用して人材育成ゴールモデルを作成することにより人材像を定義し、育成目標を設定する。

6.1 教育研修システムのステークホルダ

図 6.1 に教育研修システムのステークホルダ図を示す。図 5.1 の教育研修システム開発プロセスより、企業における高度ソフトウェア専門技術者を育成する教育研修システム開発では、ソフトウェアの開発企画部署、研修開発を進める研修企画部署、研修を実施する研修運営部署、研修生の所属する研修対象部署がステークホルダとなる。

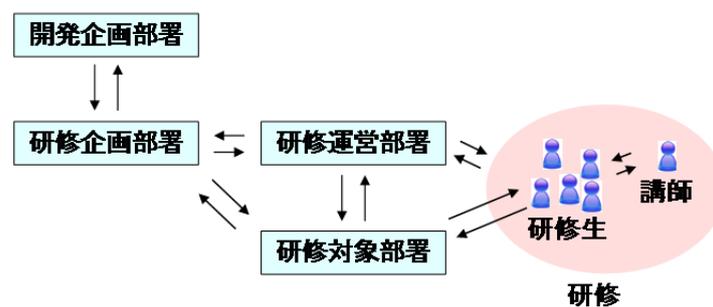


図 6.1 教育研修システムのステークホルダ

(1) 開発企画部署

企業におけるソフトウェア開発の統括部署である。会社方針や製品戦略、技術ロードマップに基づいて、ソフトウェア開発戦略に関する具体的な施策を立案する。開発企画部署は、経営視点で人材像、育成目標、教育研修システムをレビューし、高度ソフトウェア専門技術者の育成を支援、指導する。

(2) 研修企画部署

高度ソフトウェア専門技術者の教育研修システムの開発を中心となって進める部署である。開発企画部署、研修運営部署、研修対象部署と連携を取り、中長期的な視点に立って、経営的な観点と現場のニーズから会社に必要な高度ソフトウェア専門技術者の人材像、育成目標を設定し、教育研修システムを設計する。そして、研修運営部署に研修の実施を依頼する。研修後は評価結果を教育研修システムの開発にフィードバックする。

(3) 研修運営部署

研修企画部署が設計した教育研修システムに従って研修を実施する部署である。研修生を研修対象部署から募集し、教育研修システムであるカリキュラムに基づいて研修を運営する。研修後は研修結果をまとめ、研修企画部署に提出する。

(4) 研修対象部署

研修対象部署は、教育研修システムが対象とする研修生の所属部署であり、育成した高度ソフトウェア専門技術者が研修の成果を発揮する部署でもある。研修企画部署に対し、必要とする人材像についての現場のニーズを伝える。

6.2 人材育成ゴールモデル

提案する教育研修システム開発方法論では、要求工学におけるゴール指向分析[16]を適用し、目標とする人材像をゴールモデルにより体系的に分析して育成目標に展開する。本稿では、ゴール指向分析を用いて、人材像から育成目標に展開したゴールモデルを人材育成ゴールモデルと呼ぶ。

企業が必要とする高度ソフトウェア専門技術者の人材像を人材育成ゴールモデルのトップゴールとし、そのゴールを達成するサブゴールに展開することで、実行可能かつ測定可能なゴールを導出する。これらのサブゴールが人材像の育成目標となる。人材育成ゴールモデルは、人材像から段階的に育成目標を導出し、その結果をゴールグラフで表現するので、人材像と育成目標の関係を明確に表現することができる。

図 6.2 に高度ソフトウェア専門技術者の人材育成ゴールモデルを示す。人材育成ゴールモデルは、人材像と育成目標である戦略ゴールと戦術ゴールの 3 階層で構成する。

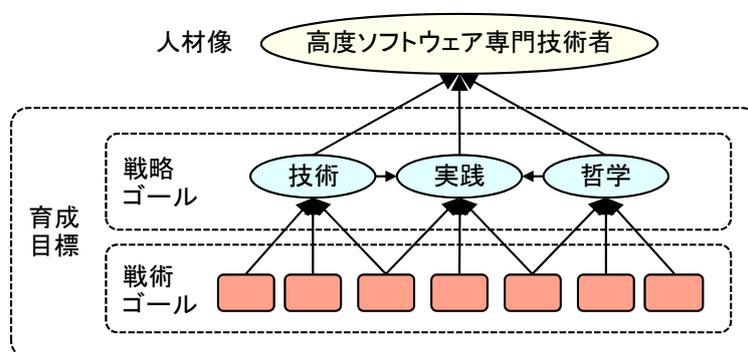


図 6.2 人材育成ゴールモデル

6.2.1 人材像

人材像は、企業が必要とする高度ソフトウェア専門技術者のあるべき姿を定義する。目標とする高度ソフトウェア専門技術者は、企業の経営戦略達成や現場の事業の存続に貢献できる人材である。長期的視野に立ち、経営陣のニーズに基づく会社方針、開発戦略と、ヒアリングから得られる現場のニーズを満たす人材像を定義する。

6.2.2 戦略ゴール

戦略ゴールは、人材像を育成の観点から詳細化した定性的な育成目標である。人材像に対する要求を包括的に扱い、得られた要求を体系的に展開することにより戦略ゴールを導出する。

本稿では、企業における高度ソフトウェア専門技術者育成の研修に関する研究成果[14]から得られた P2E (Practice, Philosophy, Engineering) モデルを適用し、人材像から技術、実践、哲学の 3 つのカテゴリーについて戦略ゴールを展開する。P2E モデルは、高度ソフトウェア専門技術者を「ソフトウェア工学を適用し、ソフトウェア開発における課題を正しい考え方で適切に解決できる技術者」であることを前提として、高度ソフトウェア専門技術者に対する要求を技術、実践、哲学で整理したモデルである。

P2E モデルに基づき、戦略ゴールを人材像の技術、実践、哲学の 3 つのサブゴールとして定義する。

(1) 技術の戦略ゴール

ゴールは、問題を解決するための高度な専門技術を習得できている状態である。

(2) 実践の戦略ゴール

ゴールは、適切な専門技術を適用して課題を正しく解決できている状態である。

(3) 哲学の戦略ゴール

ゴールは、課題を解決するために人を正しい解決に導く考え方、行動原理を備えている状態である。

6.2.3 戦術ゴール

戦術ゴールは、戦略ゴールを達成する教育研修システムが満たすべき定量的な育成目標である。戦術ゴールは、技術、実践、哲学の戦略ゴールを達成する必要かつ十分なゴールであり、実行かつ測定可能なゴールである。戦術ゴールは、人材像をゴール指向分析に基づいて戦略ゴールより展開しているのので、戦術ゴールを満たす教育研修システムを開発することにより目標とする高度ソフトウェア専門技術者の育成が可能となる。

6.3 研修要求定義プロセス

図 6.3 に研修要求定義プロセスを示す。研修要求定義プロセスは、人材育成ゴールモデルに基づいて、会社方針、技術ロードマップ、開発戦略、現場のニーズから人材像を定義し、育成目標である戦略ゴール、戦術ゴールを人材像から展開して設定する。戦略ゴールは、バランススコアカードのフレームワークである戦略マップを用いて設定する。研修要求定義プロセスについて以下に説明する。

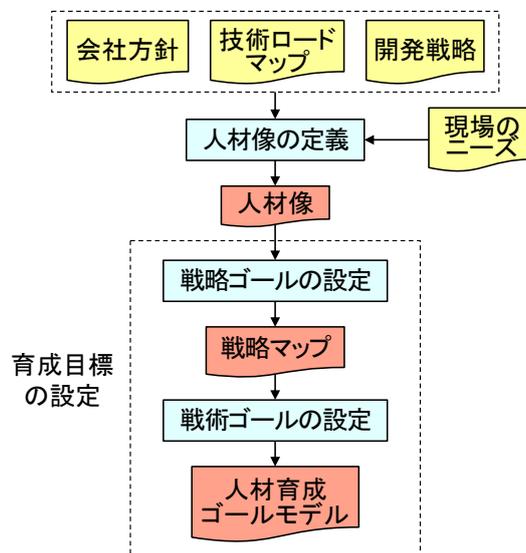


図 6.3 研修要求定義プロセス

6.4 人材像の定義

高度ソフトウェア専門技術者に対する要求を、経営、現場の観点で整理し、中長期的な視点に立って人材像を定義する。

(1) 経営の観点

組織的に決定された経営戦略を達成するために必要な高度ソフトウェア専門技術者に対する要求を整理する。開発戦略、会社方針、技術ロードマップから、高度ソフトウェア専門技術者が必要な業務、課題、役割を導出することにより、求める人材像を明確にする。

(2) 現場の観点

現場とのコミュニケーションやヒアリングにより、現場のニーズを探索、調査し、高度ソフトウェア専門技術者に対する現場の要求を整理する。

6.5 戦略ゴールの設定

育成目標である戦略ゴールは、戦略マップ[13]を適用し、P2Eモデルに基づく技術、実践、哲学の観点から人材像を展開することにより導出する。

6.5.1 戦略マップ

戦略マップは、バランストスコアカードにおいて戦略が無形の資産を価値創造プロセスにどのように結びつけるかを明示するためのフレームワークである[13]。戦略マップは、ビジネスにおける戦略のロジックを表現し、価値を創造する重要な内部プロセス及びその内部プロセスを支援するために必要な戦略を明確に示すことができる。戦略マップは戦略を視覚的に表現する形式を規定し、戦略がどのように結合するかについての1枚におさまる概念図を提供する。

6.5.2 戦略ゴールの設定方法

戦略マップにおける価値創造プロセスを、目標とする人材像を育成する高度ソフトウェア専門技術者の育成プロセスと考えれば、価値創造プロセスにおける戦略は育成目標の戦略ゴールに対応できる。そこで、戦略マップにおける戦略テーマを高度ソフトウェア専門技術者の育成、戦略目標を戦略ゴールの達成目標に設定し、高度ソフトウェア専門技術者の育成プロセスにおける戦略ゴールを戦略マップにより導出する。戦略マップを用いることにより、育成目標である戦略ゴール間の関係や人材像から戦略ゴールへの展開を構造的に表現することができる。

図 6.4 は、実際の高度ソフトウェア専門技術者の育成プロセスに対して作成した戦略マップである。戦略マップによる戦略ゴールの導出を以下に示す。

6.5.2.1 達成目標の導出

戦略ゴールにおける技術、実践、哲学の各トップゴールを達成目標と定義する。人材像より技術、実践、哲学における達成目標を導出し戦略マップに記入する。図 6.4 の戦略マップでは、人材像を「ソフトウェア開発を主導できる技術者」とし、人材像より技術、実践、哲学の達成目標を以下のように設定した。

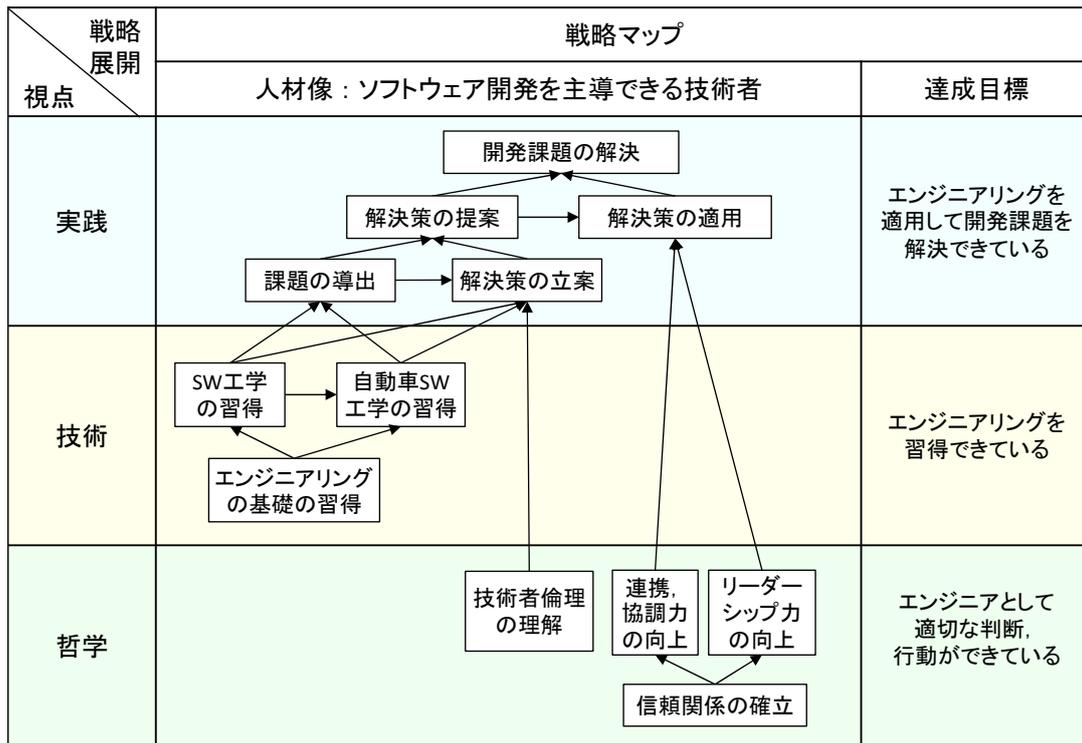


図 6.4 戦略マップによる戦略ゴールの導出

(1) 実践の達成目標

人材育成ゴールモデルにおける実践の戦略ゴールは、適切な専門技術を適用して課題を正しく解決できている状態である。そこで、人材像より実践の達成目標を、エンジニアリングを適用して開発課題を解決できている状態と定義する。

(2) 技術の達成目標

人材育成ゴールモデルにおける技術の戦略ゴールは、問題を解決するための高度な専門技術を習得できている状態である。そこで、人材像より技術の達成目標を、問題解決に必要なモデリングや最適化、分析技術を含むエンジニアリングを習得できている状態と定義する。

(3) 哲学の達成目標

人材育成ゴールモデルにおける哲学の戦略ゴールは、課題を解決するために、人を正しく解決に導く考え方、行動原理を備えている状態である。そこで、人材像より哲学の達成目標を、エンジニアとして適切な判断、行動ができている状態と定義する。

6.5.2.2 戦略ゴールの導出

高度ソフトウェア専門技術者は、経営戦略達成に貢献する技術課題を解決することが求められる。そこで、戦略マップでは、実践の視点を最上位に位置付け、続いて、技術の視点、哲学の視点の順に配置する。

戦略ゴールは、実践、技術、哲学の順に導出する。実践の戦略ゴールは、実践の戦略ゴールのトップゴールに達成目標を設定し、達成目標を展開することにより導出する。次に、実践の戦略ゴール達成に必要な技術、哲学の戦略ゴールをそれぞれの達成目標から導出し、実践における戦略ゴールとの関係を明確にする。

図 6.4 の戦略マップにおける実践、技術、哲学の戦略ゴールの詳細な導出方法について説明する。

(1) 実践の戦略ゴール

実践の戦略ゴールは、課題解決の手順に従い、達成目標である「開発課題の解決」を「解決策の提案」と「解決策への適用」に展開する。さらに、「解決策の提案」を「課題の導出」と「解決策の立案」の戦略ゴールに展開する。

(2) 技術の戦略ゴール

技術の達成目標と関連する実践の戦略ゴールから技術の戦略ゴールを導出する。技術の視点が必要な実践の戦略ゴールは「課題の導出」と「解決策の立案」である。企業理念や製品ドメイン、中長期計画に基づいて「課題の導出」、「解決策の立案」に必要な技術の戦略ゴールを「ソフトウェア工学の習得」、製品ドメインである「自動車ソフトウェア工学の習得」、「エンジニアリングの基礎の習得」を達成目標から展開する。

(3) 哲学の戦略ゴール

哲学の達成目標と関連する実践の戦略ゴールから哲学の戦略ゴールを導出する。哲学の視点が必要な実践の戦略ゴールは、「解決策の立案」と「解決策の適用」である。企業理念、経営の方針、社員の行動指針や価値観から哲学の達成目標であるエンジニアに求められる適切な判断、行動を考慮し、「解決策の立案」と「解決策の適用」に必要な哲学の戦略ゴールを「技術者倫理の理解」「連携、協調力の向上」「リーダーシップ力の向上」「信頼関係の確立」とする。

6.6 戦術ゴールの設定

戦略マップより導出した戦略ゴールを人材育成ゴールモデル上に展開し、実践、技術、哲学について戦術ゴールを導出する。

6.6.1 戦略ゴールの展開

図 6.5 は戦略マップより導出した戦略ゴールを人材育成ゴールモデルに展開した図である。戦略マップの達成目標を人材育成ゴールモデルの戦略ゴールのトップゴールとし、導出した実践、技術、哲学の戦略ゴールを人材育成ゴールモデル上に展開する。

図 6.4 の戦略マップで検討した実践と技術、実践と哲学の戦略ゴール間の関係は、図 6.5 の人材育成ゴールモデルでは戦略ゴールのトップゴールである戦略目標間の関係で表現する。戦略マップにおける戦略ゴール間の関係から、技術の戦略ゴールは実践の戦略ゴールである「課題の導出」、「解決策の立案」に対して必要な技術、ツールを提供している。また、哲学の戦略ゴールは、実践の戦略ゴールである「解決策の立案」「解決策の適用」に対して判断基準と行動原理を提供している。

6.6.2 戦術ゴールの導出

図 6.6 は、図 6.5 の戦略ゴールより戦術ゴールを展開して作成した人材育成ゴールモデルである。戦術ゴールは、戦略ゴールを達成するために必要な定量的な育成目標として導出する。

図 6.6 の人材育成ゴールモデルは、技術、実践、哲学の各戦略ゴールに対して戦術ゴールを導出している。導出した戦術ゴールが他の戦略ゴールと共有する場合は、両者間の関係を赤色の点線で示している。これにより、技術と実践、実践と哲学の戦略ゴールの関係が戦術ゴールで表現できたことになる。従って、人材育成ゴールモデルの戦術ゴールに着目した研修設計を実施すれば、カリキュラム設計、教授法設計において技術と実践、実践と哲学の関係を配慮した設計が可能となる。

図 6.6 の人材育成ゴールモデルにおける実践、技術、哲学の戦術ゴールの詳細な設定方法について説明する。

(1) 技術の戦術ゴール

技術の戦略ゴールは、技術の習得が中心である。そこで、Bloom's Taxonomy の認知的領域のレベルを適用して戦術ゴールを設定する。ソフトウェア工学、自動車ソフトウェア工学は、実践における問題解決に必要な技術であるので、Bloom's Taxonomy のレベル 2 の理解に相当する「技術を説明できること」を戦術ゴールとして設定する。また、エンジニアリングには、分析、最適化など問題解決に必要な基本技術が含まれている。そこで、エンジニアリングの基礎については、Bloom's Taxonomy のレベル 3 の応用と 4 の分析に相当する「開発の適用可能性を分析できること」を戦術ゴールとして設定する。

(2) 実践の戦術ゴール

実践の戦術ゴールは、戦略ゴールが示す課題解決のプロセスをさらに細分化し、課題解決を具体的に実施するプロセスのゴールを設定する。戦略ゴールの課題の導出には、問題の原因分析、その結果から課題の抽出が必要である。さらに、その課題に対し解決策を作成し、解決策を実行することにより戦略ゴールは達成できる。以上の観点から実践の戦術ゴールを設定する。

(3) 哲学の戦術ゴール

哲学の戦術ゴールは、トップゴールより展開した 4 つの戦略ゴールについて課題解決を正しく進めるために必要な条件を具体的に設定する。ソフトウェア開発に必要な技術者倫理は、安心・安全なシステムを提供するために技術者に必要な倫理である。関係者と連携、協調していくためには、信頼関係は必要不可欠であり、関係者との相互理解が求められる。リーダーシップを発揮するには、関係者との相互理解を前提に目標達成に向けてメンバーを先導する能力が必要である。以上の観点から哲学の戦術ゴールを設定する。

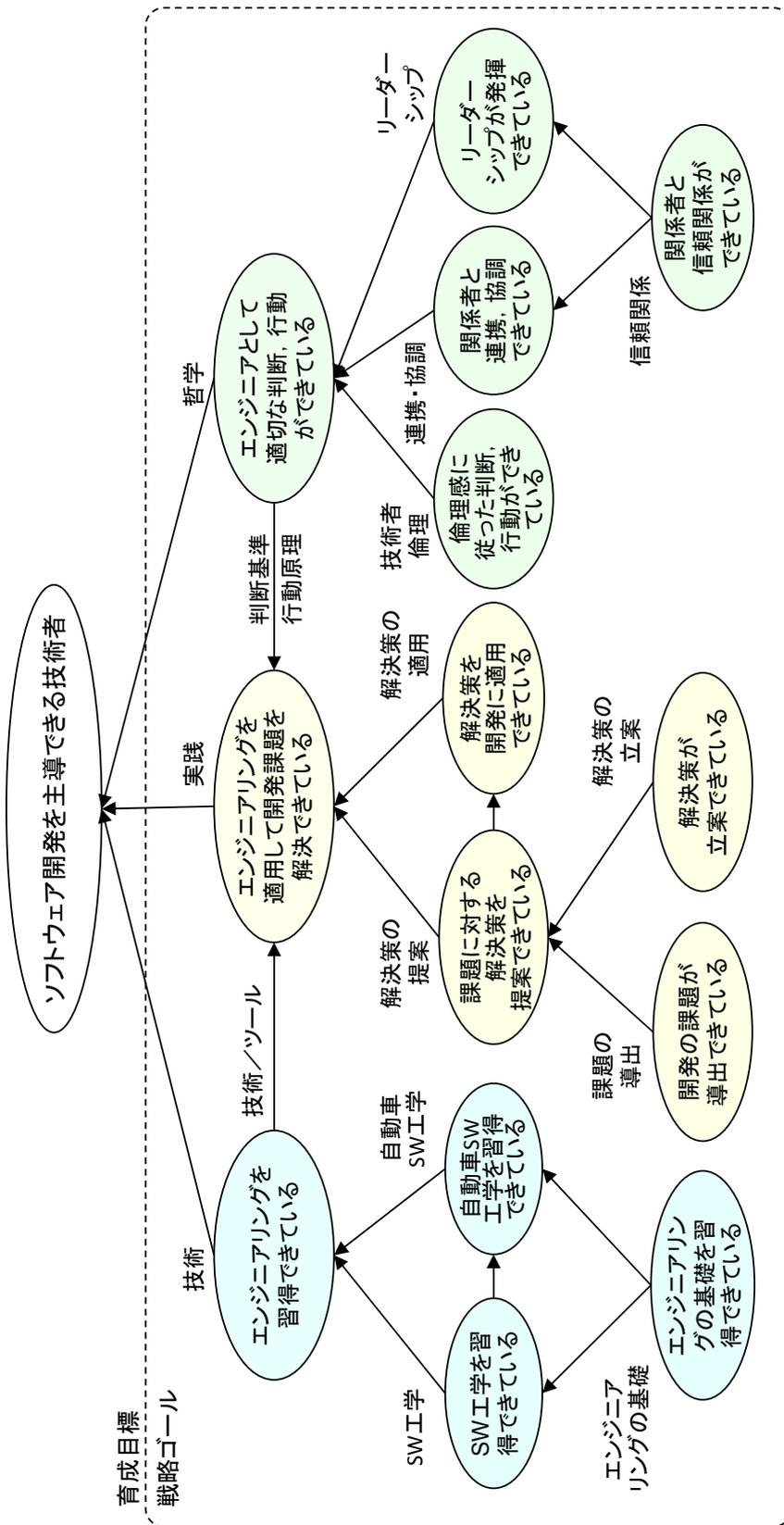


図 6.5 人材育成ゴールモデル (戦略ゴール)

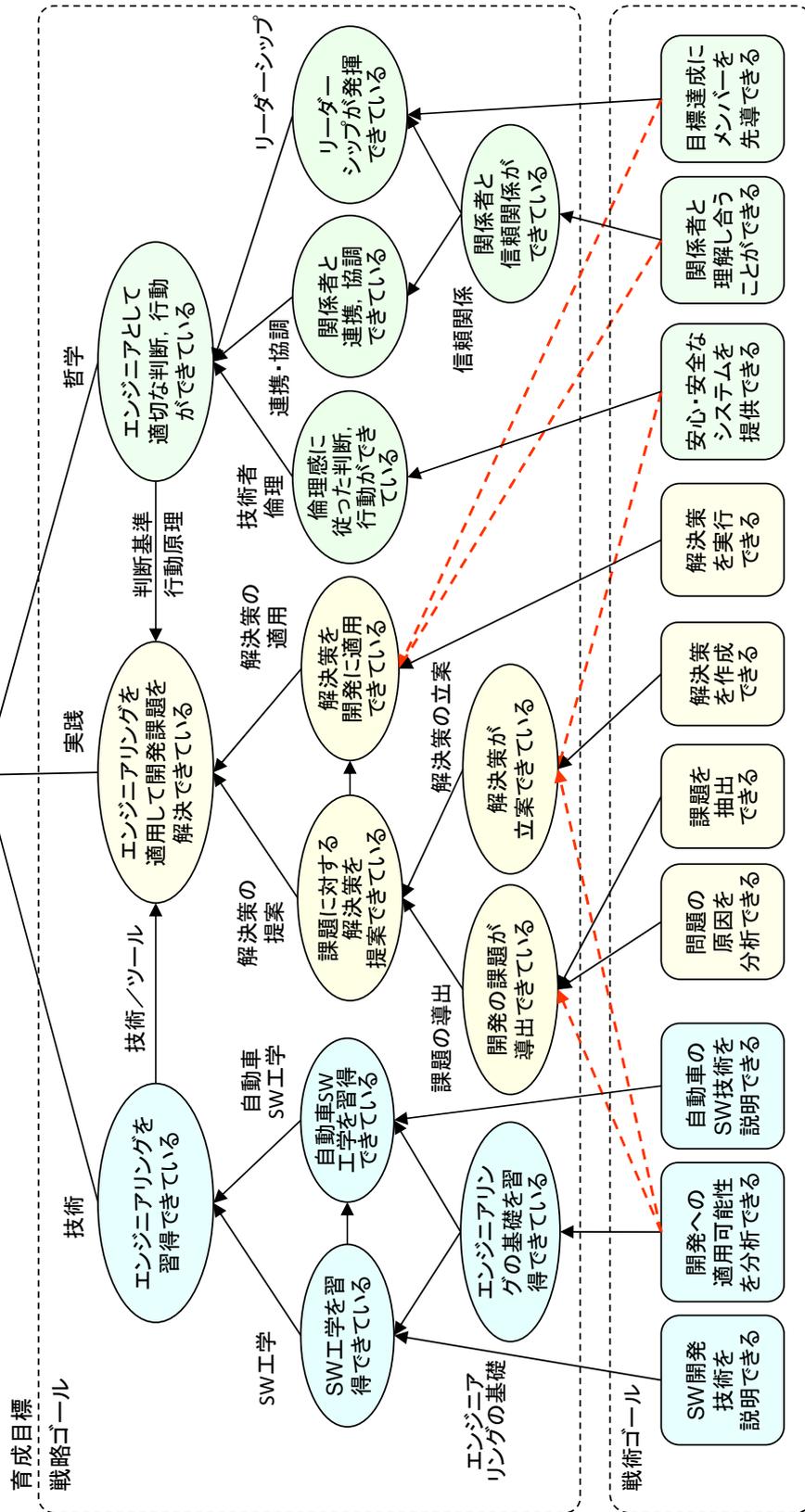


図 6.6 人材育成ゴールモデル

7 研修設計

研修設計では、教育研修システムを構成するカリキュラム、学習目標、教授法を設計する。

7.1 カリキュラム設計

人材育成ゴールモデルから高度ソフトウェア専門技術者に必要な知識、スキルを育成目標から導出し、この知識、スキルを習得するカリキュラムを設計する。

7.1.1 カリキュラム設計プロセス

図 7.1 にカリキュラム設計プロセスを示す。教育研修システムにおけるカリキュラム設計は、人材育成ゴールモデルにおける戦術ゴールから技術、実践、哲学の育成目標達成に必要な知識、スキルを導出し、その知識、スキルに基づいてカリキュラムを設計する。

図 7.2 に人材育成ゴールモデルより導出した知識、スキルを示す。人材育成ゴールモデルを用いて高度ソフトウェア専門技術者に必要な知識、スキルを導出することにより、目標とする人材に必要な知識、スキルが明確になる。育成目標と知識、スキルの関係からカリキュラム設計の指針が得られる。

本章では、人材育成ゴールモデルより育成目標である技術、実践、哲学についての知識、スキルの導出方法について説明する。具体的なカリキュラムの設計については、第 9 章の実際の研修への適用で説明する。

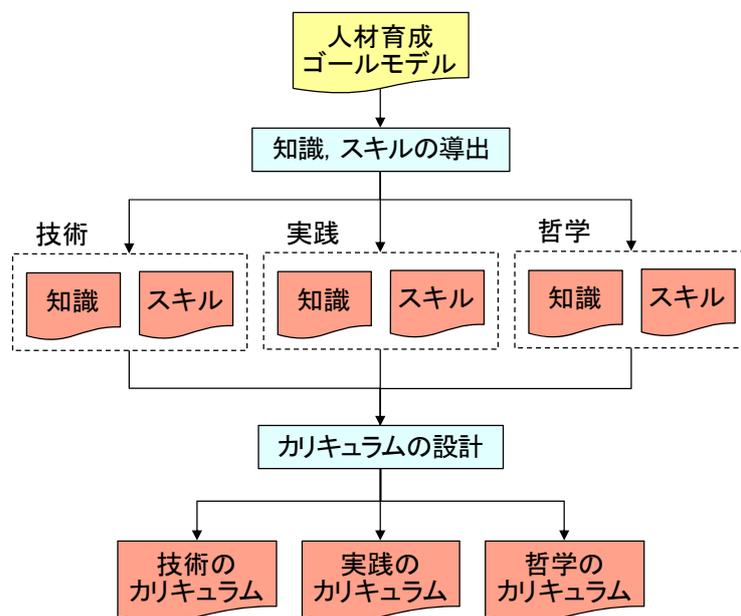


図 7.1 カリキュラム設計プロセス

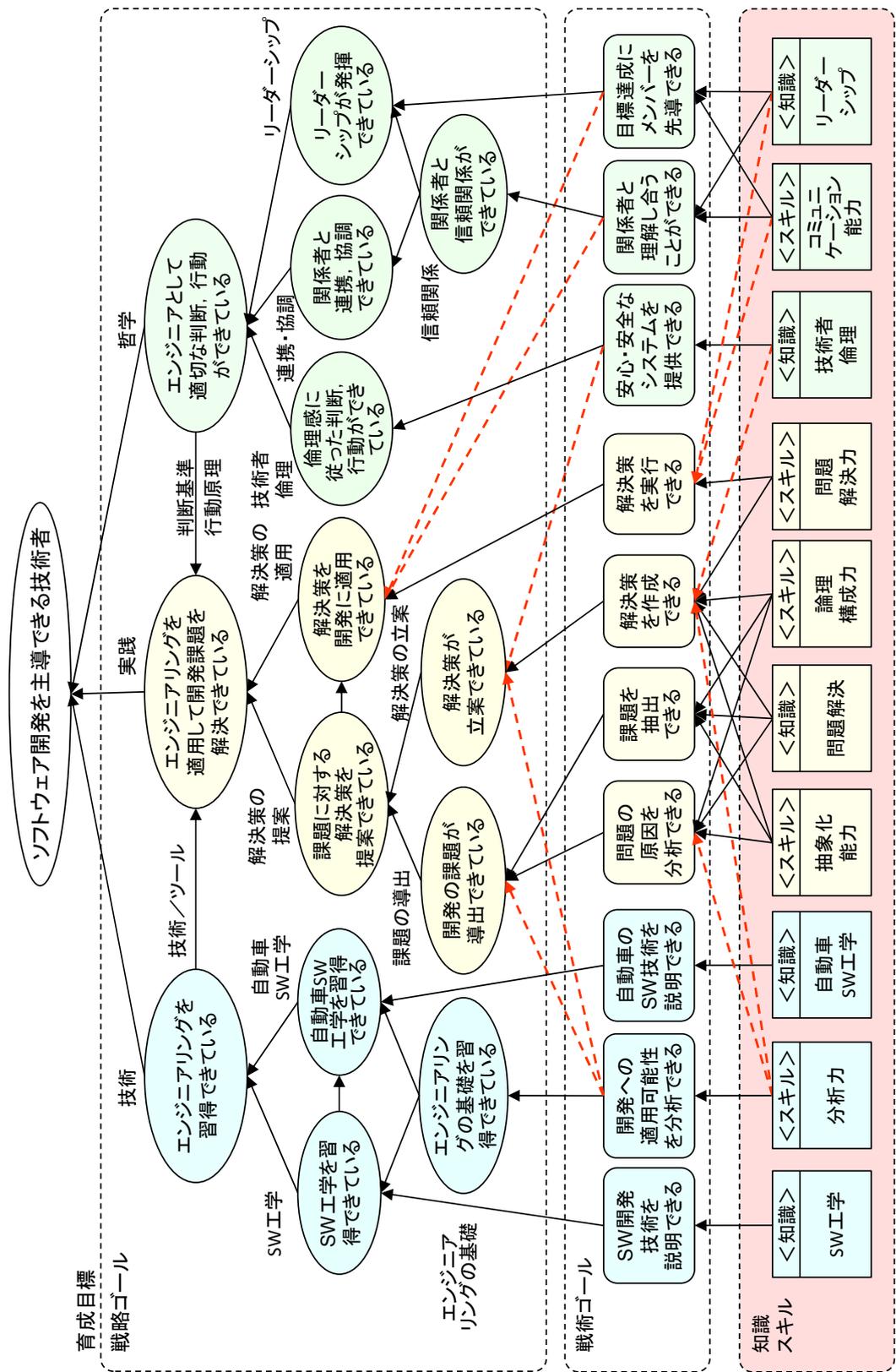


図 7.2 人材育成ゴールモデルによる知識、スキルの導出

7.1.2 技術のカリキュラム設計

図 7.2 より、技術の戦術ゴールは、知識についての戦術ゴールとその知識を活用するスキルについての戦術ゴールに分けることができる。表 7.1 は、図 7.2 から導出した技術の育成目標達成に必要な知識、スキルと戦術ゴールとの関係を示した表である。

知識についての戦術ゴールは、「ソフトウェア開発技術を説明できる」「自動車ソフトウェア技術を説明できる」である。習得すべき内容は、Bloom's Taxonomy のレベル 2 に相当し、対象となるソフトウェア工学、自動車ソフトウェア工学に対する理解が求められる。

スキルについての戦術ゴールは、「開発への適用可能性を分析できる」である。戦術ゴールの達成には、ソフトウェア工学や自動車ソフトウェア工学を開発へ適用する際の可能性を判断する分析力が求められる。また、分析力は、開発現場への技術の適用可能性の判断ばかりでなく、実践の戦術ゴールである「問題の原因を分析できる」や「解決策を作成できる」にも必要不可欠なスキルである。

表 7.1 より、技術のカリキュラムは、ソフトウェア工学、自動車ソフトウェア工学についての知識を習得し、それらの技術の開発への適用可能性を分析できるスキルの向上を目的として設計する。分析力は、表 7.1 より、実践における問題の原因分析や解決策作成にも必要なスキルでもある。そこで、技術のカリキュラムは、ソフトウェアに対する分析技術やプロセスのモデリング、設計、実装の最適化技術も取り入れ、実践における問題の原因分析、課題抽出に対応できるカリキュラムとして設計する。

表 7.1 技術に必要な知識、スキル

育成目標		技術			実践	
戦略ゴール		エンジニアリングを習得できている			エンジニアリングを適用して開発課題を解決できている	
戦術ゴール		SW開発技術を説明できる	開発への適用可能性を分析できる	自動車のSW技術を説明できる	問題の原因を分析できる	解決策を作成できる
知識	ソフトウェア工学	●				
	自動車ソフトウェア工学			●		
スキル	分析力		●		●	●

7.1.3 実践のカリキュラム設計

図 7.2 より、実践の戦術ゴールは、戦略ゴールである「解決策の提案」を達成するための戦術ゴールと、同様に「解決策の適用」を達成する戦術ゴールに分けることができる。表 7.2 は、図 7.2 から導出した実践の育成目標達成に必要な知識、スキルと戦術ゴールとの関係を示した表である。

「解決策の提案」を達成する戦術ゴールは、「問題の原因を分析できる」、「課題を抽出できる」、「解決策を作成できる」である。これらの戦術ゴールは一般的な問題解決のプロセスに相当する。そこで、解決策の提案に必要なスキルを「分析力」、「抽象化能力」、「論理構成力」、「問題解決力」とし、必要な知識を「問題解決」についての知識とする。解決策は、安心・安全なシステムの提供が前提になるので、「技術者倫理」の知識を追加する。

表 7.2 実践に必要な知識、スキル

育成目標	カテゴリー	実践				技術	哲学						
		問題の原因を分析できる	課題を抽出できる	解決策を作成できる	解決策を実行できる		エンジニアリングを習得できている	エンジニアとして適切な判断、行動ができています	安心・安全なシステムを提供できる	関係者と理解し合うことができる	目標達成にメンバーを先導できる		
戦略目標	戦略目標	エンジニアリングを適用して開発課題を解決できている											
	戦術目標	問題の原因を分析できる	課題を抽出できる	解決策を作成できる	解決策を実行できる								
知識	リーダーシップ				●						●	●	●
	技術者倫理			●							●		
	問題解決	●	●	●	●						●		
	分析力	●	●	●	●				●				
スキル	抽象化能力	●	●	●	●								
	論理構能力	●	●	●	●								
	問題解決力												●
	コミュニケーション能力												●

「解決策の適用」を達成する戦術ゴールは「解決策を実行できる」である。解決策の実行は、立案した解決策を実際の現場で実行することである。解決策を実行するためには、解決策の提案とは異なり、チームメンバーや関係部署の協力や解決策を進める上でのリーダーシップが求められる。そこで、解決の実行に必要なスキルを問題解決力、コミュニケーション能力、必要な知識をリーダーシップの知識とする。解決策の実行に必要な問題解決スキルは、技術の問題解決だけではなく、組織やマネジメントに対する問題解決も含む。

実践のカリキュラムの対象を表 7.2 の角丸長方形で囲んだ領域で示す。問題の原因分析、課題抽出、解決策の作成を通して、分析力、抽象化能力、論理構成力、問題解決力の向上を目的とするカリキュラムとして設計する。また、問題解決を通して実際の現場で必要な技術者倫理の知識や問題解決の知識が習得できるよう設計する。

解決策の実行については、開発現場での取り組みが中心となるので教育研修システムで取り扱うことは困難である。解決策実行に必要なリーダーシップの知識、問題解決力、コミュニケーション能力のスキルについては、哲学でその内容をカバーできるよう設計する。

7.1.4 哲学のカリキュラム設計

図 7.2 より、哲学の戦術ゴールは、技術者倫理、信頼関係、リーダーシップの 3 つのカテゴリーに分けることができる。信頼関係は、関係者との連携・協調、リーダーシップの戦略ゴールを達成する戦術ゴールである。表 7.3 は、図 7.2 から導出した哲学の育成目標達成に必要な知識、スキルと戦術ゴールとの関係を示した表である。

技術者倫理の戦術ゴールである「安心・安全なシステムを提供できる」には、技術者倫理に関する基本的な知識が必要である。技術者倫理の考え方や事例から得られる知識が、安心・安全なシステムを提供するための判断基準となる。技術者倫理の知識は、実践における解決策の作成にも必要な知識である。

信頼関係の戦術ゴールである「関係者と理解し合うことができる」には、相互理解のためのコミュニケーション能力が必要である。相手を理解し、自分の意見、主張を論理的にわかりやすく表現するコミュニケーション能力は、チームメンバーを目標達成に先導するリーダーシップを発揮する上でも重要なスキルである。また、コミュニケーション能力は、実践の解決策の実行にも必要な能力である。

リーダーシップの戦術ゴールである「目標達成にメンバーを先導できる」には、コミュニケーション能力の他に、リーダーシップについての知識[10][15]が必要不可欠である。リーダーの役割、行動原則、価値観を理解しておくことにより、正しいリーダーシップを実践できる。リーダーシップの知識は、関係者との相互理解においても相手の立場を理解する上で有効である。また、実践の解決策の実行にもリーダーシップの知識は必要である。

表 7.3 より、哲学のカリキュラムは、技術者倫理、リーダーシップに関する知識を習得し、コミュニケーション能力の向上を目的として設計する。技術者倫理は、課題の解決策の作成に必要な知識であることを踏まえてカリキュラムを検討する。リーダーシップの知識、コミュニケーション能力は、実践での解決策の実行に必要な知識、スキルである。哲学のカリキュラムでは、解決策実行に求められるリーダーシップの知識の習得、解決策のチームメンバーへの説明や説得に必要なコミュニケーション能力の向上を狙いとして設計する。

表 7.3 哲学に必要な知識, スキル

育成目標		哲学			実践	
戦略ゴール		エンジニアとして適切な判断, 行動ができています			エンジニアリングを適用して開発課題を解決できている	
戦術ゴール		安心・安全なシステムを提供できる	関係者と理解し合うことができる	目標達成にメンバーを先導できる	解決策を作成できる	解決策を実行できる
知識	技術者倫理	●			●	
	リーダーシップ		●	●		●
スキル	コミュニケーション能力		●	●		●

7.2 学習目標設計

研修要求定義プロセスで作成した戦略マップ、戦術ゴール、カリキュラム設計で導出した高度ソフトウェア専門技術者に必要な知識、スキルから教育研修システムの学習目標を設定する。

7.2.1 学習目標設計プロセス

図 7.3 に学習目標設計プロセスを示す。学習設計プロセスでは、6.5 節で作成した戦略マップを、戦術ゴール、高度ソフトウェア専門技術者に必要な知識、スキル、到達目標を追加して拡張する。到達目標は、研修生が研修を通して実際に到達すべき具体的な目標である。学習結果の達成度を判断できる目標として戦略マップから導出する。学習目標は、到達目標に基づいて Bloom's Taxonomy を拡張することにより設計する。

表 7.4 に拡張した戦略マップを示す。拡張した戦略マップにより技術、実践、哲学についての到達目標を設定する。技術と実践は、到達目標が一意に設定できるが、哲学については、技術者倫理、関係者との連携・協調、リーダーシップの3つの内容を含んでいる。本稿では、哲学の戦略ゴールとして、高度ソフトウェア専門技術者に最も必要とされるリーダーシップを取り上げ、学習目標を設定する。

学習目標は、拡張した戦略マップで設定した到達目標に基づいて Bloom's Taxonomy を拡張して設計する。Bloom's Taxonomy は、認知的領域、情意的領域、精神運動的領域の3つの領域に分かれている。教育研修システムの学習目標は、認知的領域を拡張して作成する。認知的領域は、6段階(知識、理解、応用、分析、統合、評価)で規定されているが、評価は研修の学習目標には必要ないため5段階で学習目標を設定する。戦略マップを拡張して設定した技術、実践、哲学の到達目標を、Bloom's Taxonomy のレベル5として位置づけ、Bloom's Taxonomy を拡張して学習目標を設定する。Bloom's Taxonomy を拡張して作成した高度ソフトウェア専門技術者の学習目標を表 7.5 に示す。

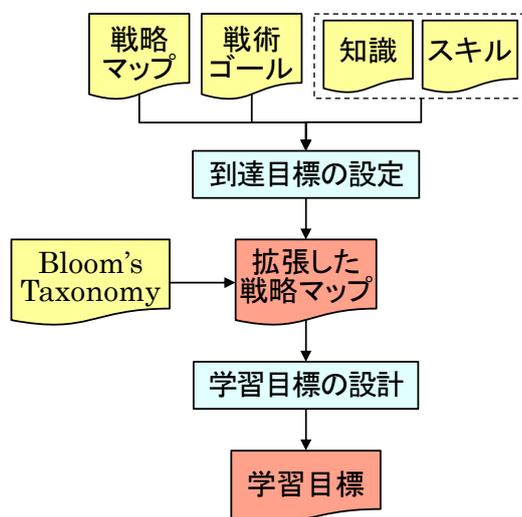


図 7.3 学習目標設計プロセス

表 7.4 拡張した戦略マップ

人材像：ソフトウェア開発を主導できる技術者					
視点	達成目標	戦略ゴール	戦術ゴール	知識／スキル	到達目標
実践	エンジニアリングを適用して開発課題を解決できている		<ul style="list-style-type: none"> 問題の原因を分析できる 課題を抽出できる 解決策を立案できる 解決策を適用できる 	<知識> ・問題解決 <スキル> ・抽象化能力 ・論理構成力 ・問題解決力 ・分析力	<ul style="list-style-type: none"> 提案した解決策が社外に貢献できる
技術	エンジニアリングを習得できている		<ul style="list-style-type: none"> SW開発技術を説明できる 開発への適用可能性を分析できる 自動車SW技術を説明できる 	<知識> ・SW工学 ・自動車SW工学 <スキル> ・分析力	<ul style="list-style-type: none"> ソフトウェア技術を開発へ適用する具体的な方法を提案できる
哲学	エンジニアとして適切な判断, 行動ができている		<ul style="list-style-type: none"> 安心・安全なシステムを提案できる 関係者と理解し合うことができる 目標達成にメンバーを先導できる 	<知識> ・リーダーシップ ・技術者倫理 <スキル> ・コミュニケーション能力	<リーダーシップ> ・リーダーシップに関する自分の考え方を理解, 納得させることができる

表 7.5 学習目標

育成目標	学習レベル	学習目標
技術	5	ソフトウェア技術を開発へ適用する具体的な方法を提案できる
	4	ソフトウェア技術の開発への適用可能性について説明できる
	3	ソフトウェア技術を開発の問題を解決する技術として説明できる
	2	ソフトウェア技術の特徴、従来技術との違いを説明できる
	1	ソフトウェア技術を専門用語を使って説明できる
実践	5	提案した解決策が社外に貢献できる
	4	提案した解決策が社内の他の部署へ貢献できる
	3	提案した解決策が自部署の問題を解決する
	2	チームの問題の解決策を提案できる
	1	問題解決の方法を説明できる
哲学	5	リーダーシップに関する自分の考え方を理解、納得させることができる
	4	リーダーシップに関する自分の考え方の意味、価値を説明できる
	3	実践で、リーダーシップについて自分の意見を説明できる
	2	リーダーシップについて自分の意見を説明できる
	1	研修で取り上げたリーダーシップについて説明できる

7.2.2 技術の学習目標設計

7.2.2.1 到達目標の設定

技術の達成目標は、「エンジニアリングを習得できている」ことである。戦略ゴールより、研修の対象となる技術は、ソフトウェア工学、自動車ソフトウェア工学とエンジニアリングの基礎である。技術の研修では、戦略ゴールより、ソフトウェア工学、自動車ソフトウェア工学を知識として習得し、エンジニアリングの基礎である分析技術、最適化技術、モデリング技術により、ソフトウェア工学の開発への適用可能性を分析できることが目標である。人材育成ゴールモデルより、技術は実践における解決策の提案に必要な技術、ツールを提供する育成目標と位置づけている。また、技術で習得する分析力は、実践における課題の導出、解決策の立案にも必要不可欠なスキルである。そこで、技術の到達目標を、実践への達成目標に貢献できるよう、「ソフトウェア技術を開発へ適用する具体的な方法を提案できる」ことに設定する。

7.2.2.2 学習目標の設計

設定した到達目標を、Bloom's Taxonomy の学習目標のレベル 5 の統合に位置付け、Bloom's Taxonomy を拡張して、技術の各レベルの学習目標を設定する。学習目標のレベル 4 の分析は、ソフトウェア技術を開発へ適用するために必要な分析ができることに対応できる。そこで、レベル 4 の学習目標を、「技術の適用可能性についての説明ができる」ことに設定する。レベル 3 の応用は、習得したソフトウェア技術を開発へ応用することと考えることができる。レベル 3 の学習目標を、「ソフトウェア技術を開発へ適用したときの効果を説明できる」こととする。学習目標のレベル 1, 2 の知識、理解は、ソフトウェア工学、自動車ソフトウェア工学の

理解度に応じた学習目標を定義する。レベル1の知識は、ソフトウェア技術を知識として知っているレベルと定義し、「ソフトウェア技術を専門用語を使って説明できる」ことを学習目標とする。レベル2の理解は、知識をより深く理解して従来の技術と比較できる状態と定義し、「ソフトウェア技術の特徴、従来技術との違いを説明できる」ことを学習目標とする。

7.2.3 実践の学習目標設計

7.2.3.1 到達目標の設定

実践の達成目標は、「エンジニアリングを適用して開発課題を解決できている」ことである。戦略ゴールより、実践の研修では、具体的な課題に対して解決策を提案し、現場に解決策を実際に適用することが求められる。そのためには、戦術ゴールより問題の原因分析、課題抽出、解決策の作成、解決策の実行を実施することで、抽象化能力、論理構成力、問題解決力、分析力のスキルと問題解決の知識を習得する必要がある。高度ソフトウェア専門技術者には高いレベルの課題解決力が求められるので、課題解決策の到達目標を、「提案した解決策が社外に貢献できる」ことに設定する。

7.2.3.2 学習目標の設計

設定した到達目標を Bloom's Taxonomy の学習目標のレベル5の統合で達成できるものとして、レベル1から4の学習目標を解決方法の有効性に基づいて設定する。レベル1の知識の学習目標は、問題解決についての知識を習得し、「問題解決の方法を説明できる」ことに設定する。レベル2, 3, 4の理解、応用、分析の学習目標は、それぞれのレベルの分析力、抽象化能力、論理構成力、問題解決力を習得したときに立案できる課題解決策の有効性に基づいて設定する。レベル2の学習目標は、提案する解決策の適用範囲を所属するチーム、レベル3は自部署、レベル4は社内として、解決策の有効性に基づいた学習目標を設定する。

7.2.4 哲学の学習目標設計

哲学は、到達目標に対して、技術者倫理、関係者との連携・協調、リーダーシップの3つの戦略ゴールを設定している。哲学の研修では、この3つの内容について、技術者倫理の知識、リーダーシップの知識を習得し、コミュニケーション能力の向上を図る。人材育成ゴールモデルより、これらの哲学の知識、スキルは実践における解決策の提案、解決策の適用に必要な判断基準、行動原理を提供する。技術者倫理、関係者との連携・協調、リーダーシップは、それぞれ独立した内容であるため、到達目標、学習目標はそれぞれ個別に設定する必要がある。

本稿では、高度ソフトウェア専門技術者が実践において最も期待されるリーダーシップを哲学で取り上げる。以降では、リーダーシップの到達目標、学習目標について説明する。

7.2.4.1 到達目標の設定

哲学の達成目標は、「エンジニアとして適切な判断、行動ができている」ことである。リーダーシップに関する戦略ゴールは、リーダーシップ力の向上と信頼関係の確立である。研修では、リーダーシップの知識の習得とコミュニケーション能力向上が目標である。

高度ソフトウェア専門技術者は、解決策を提案するだけでなく、解決策を組織に展開することにより、実際に課題を解決することが求められている。提案する解決策を開発メンバーに説明し、メンバーを目標達成に先導しなければならない。リーダーシップは、実践における解決策の適用に必要な行動原理である。そこで、哲学の到達目標を、「リーダーシップに関する自分の考え方を理解、納得させることができる」ことに設定する。リーダーシップについて自分の考え方をメンバーにしっかり理解させることができれば、チームを目標達成に向けて先導することが可能である。

7.2.4.2 学習目標の設計

設定した到達目標を、Bloom's Taxonomy の学習目標のレベル 5 の統合として位置づけ、Bloom's Taxonomy の各レベルに対応した学習目標を設定する。学習目標のレベル 4 の分析は、リーダーシップについての考え方を分析して、「リーダーシップに関する自分の考え方の意味、価値を説明できる」ことに設定する。レベル 3 の応用の学習目標は、開発現場でリーダーシップを発揮するために、「実践でリーダーシップについて自分の意見を説明できる」ことに設定する。レベル 1, 2 の知識、理解の学習目標は、学習したリーダーシップについての理解度に応じて設定する。レベル 1 の知識の学習目標は、「研修で取り上げたリーダーシップについて説明ができる」こと、レベル 2 の理解は、「リーダーシップについて自分の意見を説明できる」ことに設定する。

7.3 教授法設計

ソフトウェア工学の原理を応用することにより、高度ソフトウェア専門技術者に必要な知識、スキルに対して適切な教授法を設計する。教授法設計では、人材育成ゴールモデルから導出した教授法から新たな教授法を設計し、Bloom's Taxonomy を拡張して作成した学習目標に基づいて最適な教授法を選択する。

7.3.1 教授法設計のアプローチ

図 7.4 に教授法の設計方法を示す。人材育成ゴールモデルより導出した知識、スキル、カリキュラムをデータとして捉え、教授法をデータに対する手続きであるメソッドと考えれば、教授法の設計はソフトウェアの設計に帰着できる。つまり、カリキュラムで習得すべき知識、スキルをデータとし、データに対するメソッドを組み合わせることにより研修生へのサービスを提供するアプリケーション設計が教授法設計である。

教授法設計では、人材育成ゴールモデルの育成目標より、戦術ゴール達成に必要な教授法を導出する。そして、それらの教授法から、人材育成ゴールモデルより導出した高度専門ソフトウェア技術者に必要な知識、スキルを効果的に習得する新たな教授法を設計する。設計した教授法から、カリキュラムに適した教授法を学習目標に基づいて選択する。

本方法で設計した教授法による研修生の技術、実践、哲学の能力の向上は、人材育成ゴールモデルにおける戦術ゴール、戦略ゴールの達成に貢献し、高度ソフトウェア専門技術者の育成を可能とする。

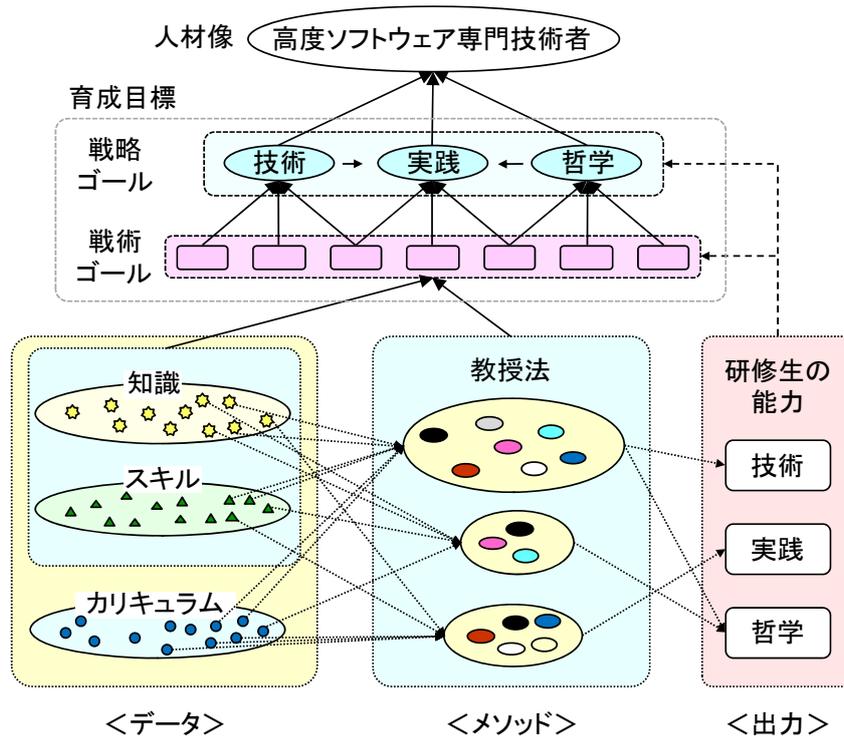


図 7.4 教授法の設計方法

7.3.2 教授法設計プロセス

図 7.5 に教授法設計プロセスを示す。本教授法設計プロセスは、人材育成ゴールモデルから導出する教授法を組み合わせることにより、新たな教授法を設計し、学習目標に基づいてカリキュラムに適した教授法を選択する。教授法設計プロセスの詳細について以下に説明する。

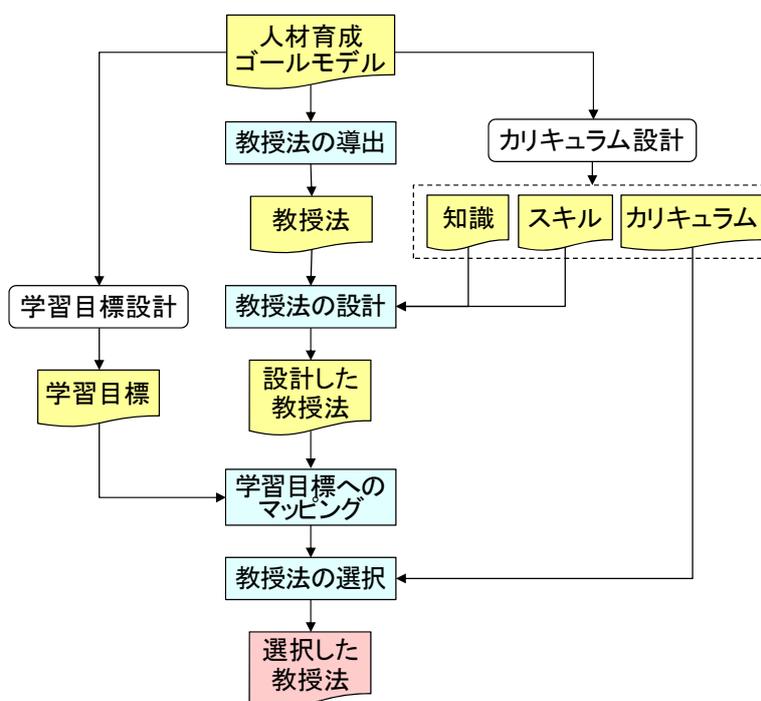


図 7.5 教授法設計プロセス

(1) 教授法の導出

人材育成ゴールモデルの育成目標である戦術ゴールの達成に貢献できる教授法を導出する。

(2) 教授法的设计

人材育成ゴールモデルから導出した教授法と高度ソフトウェア専門技術者に必要な知識、スキルについてのポートフォリオを作成し、教授法を組み合わせることにより知識、スキルの習得に適した教授法を設計する。

(3) 学習目標へのマッピング

設計した教授法を、7.2節の学習目標設計で設定した表 7.5 の学習目標にマッピングする。

(4) 教授法を選択

教授法をマッピングした学習目標に基づいてカリキュラムに適した教授法を選択する。

提案する設計方法を企業における高度ソフトウェア専門技術者育成の教授法設計に適用した。この適用例を用いて提案する教授法的设计方法を説明する。

7.3.3 教授法の導出

人材育成ゴールモデルの戦術ゴールから、高度ソフトウェア専門技術者育成研修で実施可能な教授法を、表 7.6 に示す一般的に用いられる教授法より選択する。表 7.6 には、これまでの研修で実績のある教授形態である論文作成、論文発表、ライトニングトークも教授法として追加している。

表 7.6 一般的な教授法

教授法	説明
講義	講師が作成してきた教授内容を口頭で説明する
講話	講師が自分の経験や考え方をわかりやすく説明する
演習	講義内容に関する練習問題や問題集を解く
ワークショップ	主体的に議論に参加しチームの相互作用を通じて学習する
ケーススタディ	開発の具体的な問題に取り組むことで実践的な解決法を学ぶ
ディスカッション	テーマについての知見や意見の共有化を目的に討議する
相談	研修生が課題解決の進め方や論文作成について講師に相談する
レビュー	講師が研修生のレポートや発表資料、論文を精査する
検討会	講師、研修生全員でレポートや発表資料、論文について議論する
論文作成	開発課題を導出し解決策を立案して論文にまとめる
論文発表	まとめた論文について発表する
ライトニングトーク	設定したテーマについて短いプレゼンテーションを実施する

図 7.6 に、人材育成ゴールモデルから導出した教授法を示す。育成目標である技術、実践、哲学について教授法の導出方法について以下に説明する。

(1) 技術の教授法

技術の戦術ゴールの達成には、従来の知識獲得を目的とした教授法である講義、演習、ワークショップを選択できる。知識の習得、定着には講義、演習が適している、また、適用可能性の分析には演習、ワークショップが効果的である。適用可能性の分析には、ケーススタディやディスカッションも有効である。しかし、ケーススタディやディスカッションを実施するには講義、演習、ワークショップに比べ時間がかかる。研修期間の制約と技術の研修内容、効果の観点から技術の教授法として講義、演習、ワークショップを選択する。

(2) 実践の教授法

実践の戦術ゴールを達成するためには、開発課題に対して、問題の原因分析、課題抽出、解決策の作成、解決策の実行が求められる。このように実践の戦術ゴール達成にはスキル習得中心の教授法が適している。従来の演習やワークショップでは応用範囲が狭く、実践の育成目標である現場の開発課題の解決に対応するのは困難である。そこで、実践では、研修生が実際の開発の問題を分析することにより課題を抽出し、作成した解決策を実行できる教授法を導出する。

導出した実践の教授法は、研修生の課題への取り組みを支援する教授法である。課題解決の進め方の相談、レビュー、成果報告による進め方の検討会、そして、課題抽出、課題解決の結果をまとめる論文作成、論文発表を教授法として導出した。

(3) 哲学の教授法

哲学の戦略ゴールである技術者倫理、関係者との連携・協調、リーダーシップは、それぞれ内容が異なるため戦略ゴールごとに教授法を導出する。

技術者倫理の戦術ゴールである安心・安全なシステムの提供には、講義と実際の現実の場面を想定したケーススタディが有効である。講義により技術者倫理の知識を習得し、システム開発についてのケーススタディを実施することにより技術者倫理に即した正しい考え方を身につける。

連携・協調の戦術ゴールである関係者との理解には、一線のリーダーや経営者の経験や成功体験、それに基づく考え方についての講話が有効である。理解した内容を自分自身の立場や場面に置き換えて考えるためには、ケーススタディが効果がある。連携・協調のための実践的なスキルには、自分の考え方や意見を説明したり、周りの意見に耳を傾けるコミュニケーション能力は必須である。そこで、ディスカッションとライティングトークを教授法として導出した。

リーダーシップの戦術ゴールである目標達成へのメンバーの先導には、リーダーとしての心得や考え方、実際の現場での行動原則を事例から学ぶことができる、一線のリーダーや経営者の講話が有効である。特にリーダーは、メンバーへの説明、説得が必要となるため、コミュニケーション能力の向上を目的としたライティングトークは有効なトレーニング方法である。また、ディスカッションもメンバーの先導に必要なコミュニケーション能力の向上に適した教授法である。

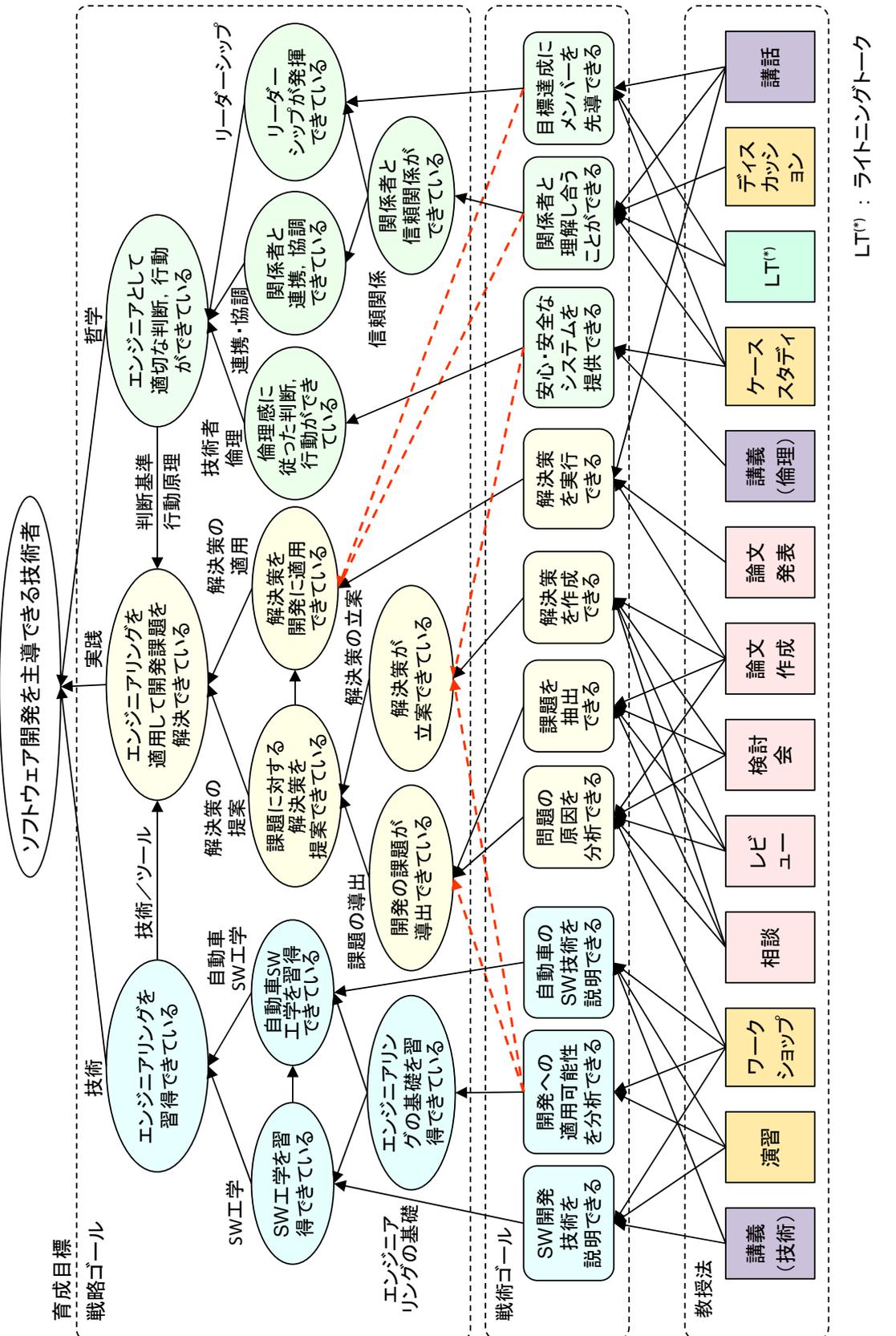


図 7.6 人材育成ゴールモデルによる教授法の導出

LT(*) : ライトニングトーク

7.3.4 教授法の設計

人材育成ゴールモデルより導出した教授法と、カリキュラム設計で導出した高度ソフトウェア専門技術者に必要な知識、スキルのポートフォリオを作成し、対象となる教育研修システムに適した教授法を設計する。

表 7.7 は、高度ソフトウェア専門技術者に必要な知識、スキルと導出した教授法から作成したポートフォリオである。表 7.7 のポートフォリオに基づいて、設計した教授法である、知識学習、実践学習、PBL について以下に説明する。

(1) 知識学習

知識学習は、知識の習得、定着を狙いとした講義と演習による教授法である。図 7.7 に知識学習の構成を示す。知識学習は、学習内容を講義形式で説明し、学習内容の定着を図るための演習を行う。講義、講話で学習内容についての知識、情報を研修生に伝え、内容についてより深く考え、理解するためにケーススタディ、ディスカッションを含む演習を実施する教授法と定義する。

ソフトウェア工学や自動車ソフトウェア工学、問題解決の知識習得の目標は、戦術ゴールより対象となる技術について説明できることである。このレベルを達成するには講義と演習を中心とした知識学習が適している。知識学習の演習は、表 7.7 のポートフォリオより、分析力や抽象化能力、論理構成力のスキルの向上も可能である。

技術者倫理の知識習得は、講義で技術者倫理の基本的な考え方や事例を学び、ケーススタディやディスカッションを通して、現実の問題へ適用する考え方を習得する。

リーダーシップの知識習得は、講義でリーダーシップの基本知識を学び、現場での実践は、講話での事例や講師の考え方を通して学習するのが効果的である。演習は、技術者倫理と同じように、ケーススタディやディスカッションを通して現場でリーダーシップを発揮するための判断基準、行動原理を習得する。

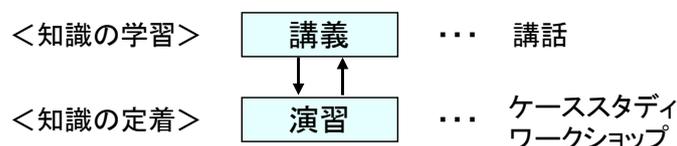


図 7.7 知識学習

(2) 実践学習

実践学習は、現場での実践に必要なスキルの向上を狙いとしたワークショップを中心とした教授法である。図 7.8 に実践学習の構成を示す。事前に研修で習得する内容を学習し、研修ではその知識を現場で実践するための考え方や具体的な方法をワークショップを通して指導する。研修では、講義や講話のように講師から研修生への一方向の情報提供は実施しない。必要な技術や知識、情報については、テキスト、書籍、DVD、インターネットなどで事前に学習する。研修では事前学習の理解度を確認し、現場で実践するための具体的な方法や考え方を習得する。

ソフトウェア工学や自動車ソフトウェア工学、問題解決の実践学習は、ワークショップを通して事前に学習した内容の効果を実際に手を動かして確認することにより、より深い知識の定着が可能である。さらに、ワークショップを通して開発における実践的な問題を扱うことにより、分析力、抽象化能力、論理構成力、問題解決力の向上を図ることができる。演習やケーススタディ、ディスカッションは、ワークショップに組

み込んで実施することも可能なので、表 7.7 では実践学習の教授法としている。技術者倫理やリーダーシップについても、ワークショップによって同様の効果が期待できる。

コミュニケーション能力についての実践学習では、自分の意見や考え方について整理、まとめることが事前学習に相当する。そして、研修では、ワークショップでのディスカッションやライトニングトークを通して自分の意見や考え方を実際に説明したり、議論する実践的な技術を習得する。

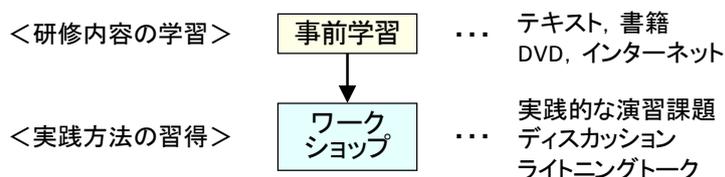


図 7.8 実践学習

(3) PBL (Problem Based Learning)

PBL は、実際の開発課題に対する問題解決力の向上を狙いとした教授法である。図 7.9 に PBL の構成を示す。PBL では、研修生が各自の開発における問題に対して、問題の原因分析、課題抽出、解決策の作成、実行に取り組み、その取り組みを支援する。この一連の課題解決の取り組みを論文作成として進める。PBL では、研修生の自主性を尊重し、研修生が主体的に論文作成に取り組めるように、相談やレビュー、検討会を定期的実施する。研修生が取り組む課題は、研修生ごとに異なるので、それぞれのテーマに合わせた具体的な支援、指導を個別に行う。研修生は、課題解決の成果を論文としてまとめ、研修の最後の論文発表会で報告する。

PBL は、開発の実践的な課題解決を扱うことにより、実際の開発に必要な分析力、抽象化能力、論理構成力、問題解決力を向上させることができる。さらに、論文発表を通して、コミュニケーション能力の向上も期待できる。

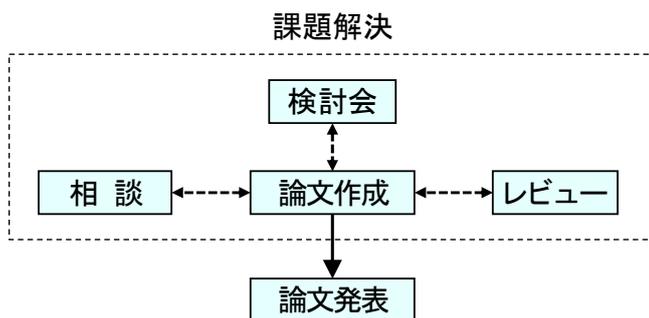


図 7.9 PBL

表 7.7 ポートフォリオによる教授法設計

知識/スキル	教授法	講義	講話	演習	ケーススタディ	ディスカッション	ワークショップ	LT(*)	相談	レビュー	検討会	論文作成	論文発表
知識	SW工学	●		●			●						
	自動車SW工学	●		●			●						
	問題解決	●		●			●						
	技術者倫理	●			●	●	●						
	リーダーシップ	●				●	●						
スキル	分析力			●	●		●					●	●
	抽象化能力			●	●		●					●	●
	論理構成力			●	●		●					●	●
	問題解決力				●	●	●					●	●
	コミュニケーション能力					●	●						●

実践学習

PBL
(Problem Based Learning)

知識学習

LT(*) : ライトニングトーク

7.3.5 学習目標へのマッピング

図 7.10 に、前節で設計した知識学習，実践学習，PBL を表 7.5 の学習目標にマッピングした結果を示す。育成目標である技術，実践，哲学に対して，それぞれの教授法の提供できる学習目標を図示した。各育成目標に対してマッピングした教授法について以下に説明する。

学習目標	育成目標		
	技術	実践	哲学
5 (統合)	ソフトウェア技術を開発へ適用する具体的な方法を提案できる	提案した解決策が社外に貢献できる	リーダーシップに関する自分の考え方を理解，納得させることができる
4 (分析)	ソフトウェア技術の開発への適用可能性について説明できる	提案した解決策が社内他の部署へ貢献できる	リーダーシップに関する自分の考え方の意味，価値を説明できる
3 (応用)	ソフトウェア技術を開発の問題を解決する技術として説明できる	提案した解決策が自部署の問題を解決する	実践で，リーダーシップについて自分の意見を説明できる
2 (理解)	ソフトウェア技術の特徴，従来技術との違いを説明できる	チームの問題の解決策を提案できる	リーダーシップについて自分の意見を説明できる
1 (知識)	ソフトウェア技術を専門用語を使って説明できる	問題解決の方法を説明できる	研修で取り上げたリーダーシップについて説明できる

図 7.10 のマッピングは、以下の通りである：

- 実践学習 (赤枠)**: 技術 (レベル 3, 4, 5) と 哲学 (レベル 3, 4, 5) の学習目標にマッピングされている。
- PBL (緑枠)**: 実践 (レベル 1, 2, 3, 4, 5) の学習目標にマッピングされている。
- 知識学習 (青枠)**: 技術 (レベル 1, 2) と 哲学 (レベル 1, 2) の学習目標にマッピングされている。

図 7.10 学習目標への教授法のマッピング

(1) 技術の学習目標へのマッピング

技術の学習目標のレベル 1, 2 に対しては知識学習を，レベル 3, 4, 5 には実践学習をマッピングした。学習目標のレベル 1, 2 におけるソフトウェア技術の体系的な技術知識の習得には知識学習が適している。一方，レベル 3, 4, 5 の応用，分析，統合の学習目標を達成するためには，ソフトウェア開発の具体的な問題を扱う必要があるため知識学習では限界がある。レベル 3, 4, 5 には，知識を現場で実践するための考え方や具体的な方法をワークショップで学習する実践学習が適している。

(2) 実践の学習目標へのマッピング

実践の学習目標に対しては，すべての学習目標に PBL をマッピングした。実践では，開発課題を抽出し，解決策を作成する能力が求められる。この能力を育成できる教授法は，実際に研修生が各自の問題に対して課題形成，課題解決に取り組む PBL が適している。

(3) 哲学の学習目標へのマッピング

哲学の学習目標に対しては、学習目標のレベル1には知識学習を、レベル2, 3, 4, 5には実践学習をマッピングした。哲学における学習目標のレベル1は、リーダーシップの知識習得を目標としているので知識学習が適している。一方、学習目標がレベル2以上になると、自分自身のリーダーシップについての考え方を説明することが求められる。特に学習目標のレベル5では、リーダーシップに関する自分の考え方を周りに理解、納得させることが目標である。学習目標のレベル2, 3, 4, 5については知識学習では限界があるため、現場での実践方法を中心に学習する実践学習をマッピングした。

7.3.6 教授法の選択

図7.10に示す教授法の学習目標へのマッピング結果に基づいて、高度ソフトウェア専門技術者を育成する教育研修システムに適した教授法を選択する。

(1) 技術の教授法

技術の教授法には、知識学習と実践学習をマッピングした。技術の育成目標は、実践における課題解決に必要なソフトウェア技術を習得し、実際の開発へのソフトウェア技術の適用可能性を判断できることである。従って、学習目標のレベル3, 4, 5にマッピングしている実践学習を教授法とすることが望ましい。

知識学習がカバーしているレベル1, 2の知識、理解の学習目標は、研修生がテキストやDVD、書籍等を使って事前に研修内容を学習してくるにより達成可能である。実践学習において事前学習を必須とし、研修の最初に理解度テスト等で学習目標のレベル1, 2の技術知識の習得を確認できれば、研修を実践学習で進めることが可能である。従って、技術の教授法には実践学習を選択する。

(2) 実践の教授法

実践の教授法は、すべての学習目標にマッピングしたPBLを選択する。

(3) 哲学の教授法

哲学の教授法には、知識学習と実践学習をマッピングした。リーダーシップについての知識は、事前にテキストや書籍などで学習することは可能である。しかし、技術者にとってリーダーシップや他の哲学で習得する知識やスキルは、技術で習得すべき知識、スキルに比べ学習経験が少ない。そこで、哲学での研修内容については、講師から研修生へ直接、知識や考え方を伝える講義、講話の必要性を検討し、カリキュラムの内容に合わせて知識学習あるいは実践学習を選択する。

8 教育研修システムの評価方法

教育研修システムであるカリキュラム，教授法に対する評価と教育研修システムを研修生のパフォーマンスで評価する方法を提案する．カリキュラム，教授法に対する評価は，研修設計時と研修実行時に実施する．教育研修システムの評価は研修実行時に実施する．

8.1 評価方法の枠組み

図 8.1 に教育研修システムの評価の枠組みを示す．教育研修システムは，育成目標に対応した多面的な評価が求められる．そこで，教育研修システムを研修設計時と研修実行時の 2 段階に分けて評価する．

研修設計時の評価は，カリキュラム，教授法を評価対象とする．提案する開発方法論により設計したカリキュラム，教授法を従来のカリキュラム，教授法と比較し評価する．

研修実行時の評価は，カリキュラム，教授法，教育研修システムを評価対象とする．教育研修システムの学習目標を評価指標として使用する．カリキュラム，教授法の評価は，提案する開発方法論により設計したカリキュラム，教授法で構成する研修を受講した研修生と従来の研修を受講した研修生のパフォーマンスを比較する．研修生のパフォーマンスは，研修受講時の成果物，研修中の発表で評価する．教育研修システムの評価は研修を受講した研修生の育成評価である．育成評価は，育成目標である技術，実践，スキルについて研修生のパフォーマンスを評価する．

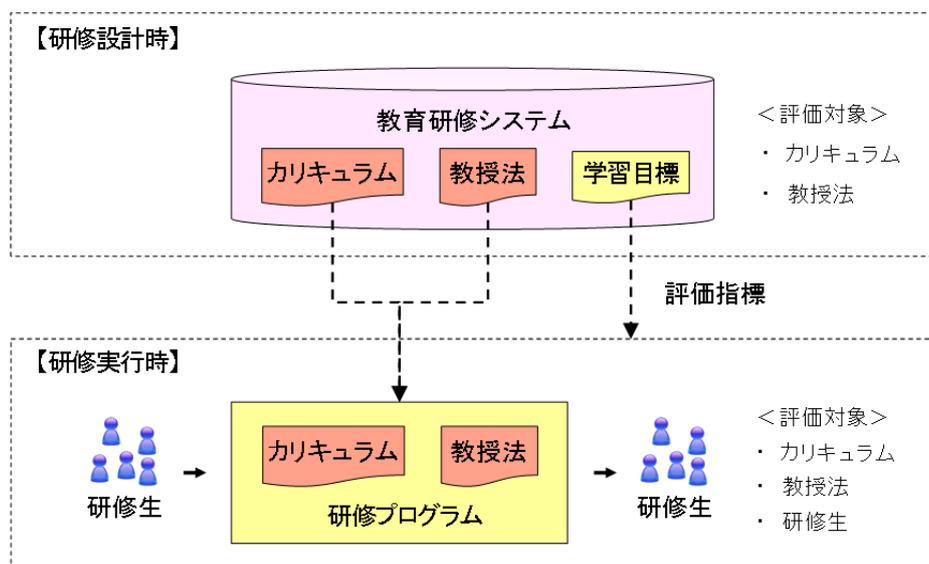


図 8.1 教育研修システム評価の枠組み

表 8.1 に，教育研修システムの評価方法の一覧を示す．教育研修システムの評価方法として，カリキュラム評価，教授法評価，そして，教育研修システムの評価である育成評価について説明する．

表 8.1 教育研修システムの評価方法

評価方法	評価対象	評価時期	評価指標	評価基準
カリキュラム評価	カリキュラム	設計時	ラーニングユニット	提案する開発方法論で開発したカリキュラム
		実行時	論文の有効性	学習目標
教授法評価	教授法	設計時	教授法の達成度	学習目標
		実行時	研修生の成果物	学習目標 (研修内容に従って設定)
育成評価	研修生	実行時	研修生のパフォーマンス	学習目標 (研修内容に従って設定)

8.2 カリキュラム評価

カリキュラム評価は、研修設計時と研修実行時に実施する。

8.2.1 研修設計時の評価

8.2.1.1 評価方法

研修設計時のカリキュラムは、対象とするカリキュラムが育成目標である技術、実践、哲学について適切に設計されているかを評価する。カリキュラムの単位は0.5日の研修を1ラーニングユニット(LU)として定義する。評価指標として、式(1)に示すように、基準とするカリキュラムに対するLU数の比である E_c を定義する。

$$E_c = \frac{\sum \{ \text{評価対象であるカリキュラムのLU} \}}{\sum \{ \text{基準とするカリキュラムのLU} \}} \quad (1)$$

カリキュラム評価は、育成目標である技術、実践、哲学について実施する。各育成目標に対して、評価対象であるカリキュラムと基準とするカリキュラムのLU数の比である E_c を算出し評価する。本稿では、提案する開発方法論により設計したカリキュラムを基準のカリキュラムとして評価を実施する。

8.2.1.2 評価手順

研修設計時のカリキュラムの評価手順を以下に示す。

- (1) 評価するカリキュラムの決定
評価対象とする教育研修システムのカリキュラムを決定する。
- (2) 基準とするカリキュラムの決定
提案する開発方法論により設計した教育研修システムから基準とするカリキュラムを決定する。
- (3) E_c の算出と評価
評価対象のカリキュラムについて式(1)から E_c を算出し、カリキュラムの評価を行う。

8.2.2 研修実行時の評価

8.2.2.1 評価方法

研修実行時のカリキュラムは、カリキュラムを適用した研修を受講した研修生のパフォーマンスで評価する。高度ソフトウェア専門技術者の育成目標は、実践の育成目標である開発課題を解決できることに集約できる。そこで、研修実行時における研修生の論文、発表資料の有効性を式(2)で定義する E_p で評価する。 E_p は実施した研修コースにおいて研修生が作成した論文、発表資料の評価の平均値である。

$$E_p = \frac{\sum \{ \text{論文, 発表資料の評価} \}}{\text{研修生の人数}} \quad (2)$$

8.2.2.2 評価手順

研修実行時のカリキュラム評価の対象である論文、発表資料の評価手順を以下に示す。

(1) 評価基準の決定

表 8.2 に論文の評価基準を示す。論文、発表資料は、解決策の有効性に基づいて 5 段階で評価する。

表 8.2 論文の評価基準

評価	説明
5	解決策が社会に貢献できる
4	解決策が社内で有効である
3	解決策が自部署で有効である
2	解決策がグループで有効である
1	解決策が個人の小改善レベルである

(2) 評価者の決定

論文、発表資料の評価は、大学教授、社外のコンサルタント、ソフトウェア担当の役員が行う。

(3) 論文、発表資料の評価

評価者が研修生の論文を事前に査読し、論文発表を聞いて、論文、発表資料を表 8.2 の評価基準に基づいて評価する。論文、発表資料の有効性 E_p を式(2)により算出し評価する。

8.3 教授法評価

教授法の評価は、研修設計時と研修実行時に実施する。

8.3.1 研修設計時の評価

8.3.1.1 評価方法

研修設計時の教授法は、設計した教授法の研修科目が達成する学習目標で評価する。

8.3.1.2 評価手順

研修設計時の教授法評価は、カリキュラムの全研修科目に対して実施する。評価手順を以下に示す。

(1) 評価するカリキュラムの決定

対象とする教育研修システムのカリキュラムを決定する

(2) 学習目標の導出と評価

評価対象とするカリキュラムのすべての研修科目に対して、それぞれの科目が教授法により達成する学習目標を求め、学習目標の分布、平均値を評価する。

8.3.2 研修実行時の評価

8.3.2.1 評価方法

研修実行時の教授法は、設計した教授法の研修科目を受講した研修生のパフォーマンスで評価する。研修生のパフォーマンスは、研修の事後レポート、研修中の発表内容により評価する。評価指標は研修科目の学習内容に基づいて学習目標から作成する。

8.3.2.2 評価手順

研修実行時の教授法の評価手順を以下に示す。

(1) 評価する研修科目の決定

対象とする研修科目を決定する。

(2) 研修生のパフォーマンスの評価

対象とする研修科目の評価指標を学習目標から作成し、研修科目を受講した研修生のパフォーマンスを評価する。

8.4 育成評価

育成評価は、提案する開発方法論により開発した教育研修システムの評価である。研修実行時に研修生に対する評価を実施する。

8.4.1 評価方法

研修実行時に、対象とする教育研修システムで構成する研修を受講した研修生のパフォーマンスを、学習目標に対する達成度で評価する。評価指標は研修科目の学習内容に基づいて学習目標から作成する。育成評価は、育成目標である技術、実践、哲学について実施する。

8.4.2 評価手順

教育研修システムの育成評価の評価手順を以下に示す。

(1) 評価対象の決定

育成目標，研修科目から，研修生のパフォーマンスを評価する対象を以下とする。

技術： 研修の事後レポート

実践： 論文

哲学： 研修の事後レポート，ライトニングトーク

(2) 評価指標の作成と評価

研修科目についての評価指標を学習目標より作成し，研修生のパフォーマンスを上記の(1)で決定した評価対象について評価する。

9 実際の研修への適用

9.1 適用した研修

提案した開発方法論の有用性を確認するため、筆者の所属する企業で実施している高度ソフトウェア専門技術者を育成する教育研修システムの開発に提案する開発方法論を適用した。高度ソフトウェア専門技術者を育成する教育研修システムを図9.1に示す。本教育研修システムは、2004年にカリキュラムを開発して研修を開始し、2013年までの10年間で計9回の研修を実施している。教育研修システムは、研修内容によって次の3期に分けることができる。

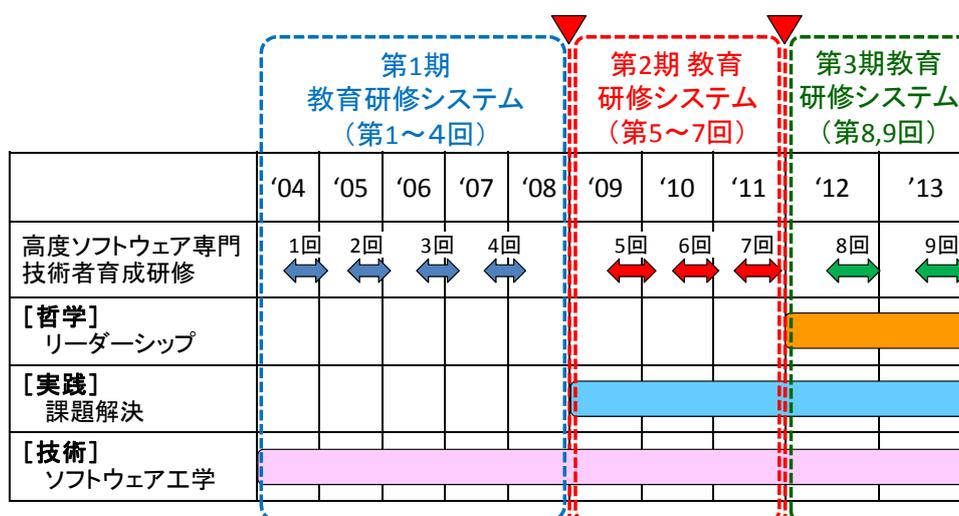


図 9.1 教育研修システム

9.1.1 第1期教育研修システム

開講した当初の教育研修システムである。2004年から2008年の間に4回の研修を実施した。表9.1に第1期教育研修システムの概要を示す。

表 9.1 第1期教育研修システム

項目	内容
目指す人材像	ソフトウェア工学を実践できる技術者
研修の狙い	ソフトウェア開発の課題を解決できる人材の育成
対象受講者	<ul style="list-style-type: none"> 開発経験5年以上の実務リーダークラス 部長推薦
研修内容	<ul style="list-style-type: none"> ソフトウェア工学に関する講義 課題発表
実施日数	16日間（7ヶ月）

(1) 目指す人材像, 研修の狙い

高度ソフトウェア専門技術者育成研修は、2004年に目指す人材像を「ソフトウェア工学を実践できる技術者」とし、「ソフトウェア開発の課題を解決できる人材の育成」を研修の狙いとしてカリキュラムを構築し研修を開始した。ソフトウェアの開発現場では多くの課題が山積しており、現場の課題解決ができる技術者が必要不可欠であった。

第1期教育研修システムの開発に当たっては、課題解決が現場で進まないのはソフトウェア工学の知識が不十分なことが原因であると当時の状況を分析し、ソフトウェア工学を実践できる技術者の育成を目指した。

(2) 対象受講者

ソフトウェア工学を実践できる技術者として、ソフトウェア開発を5年以上経験し、ソフトウェア開発を担当する現場のリーダークラスである中堅技術者を研修の対象とした。研修生は、今後のソフトウェア開発の中核を担う人材となるよう、研修の受講には本人の希望に加え、上長の推薦を必須としている。

(3) 研修内容

研修内容はソフトウェア工学に関する講義と課題発表である。ソフトウェア工学の研修科目は、ソフトウェア工学概論、メトリクスとソフトウェア開発プロセスに準じた技術科目である、要求定義、リアルタイムタスク設計、ソフトウェア設計・実装、テスト技術である。研修は、講義と演習が中心である。課題発表は、研修で習得した内容を研修生が各自の課題に適用し、今後どのように展開していくか、到達目標や方法論について発表する。

9.1.2 第2期教育研修システム

第1期の教育研修システムを改善した教育研修システムである。2009年から2011年の間に研修を3回実施した。表9.2に第2期教育研修システムの概要を示す。対象受講者は第1期と同様である。

表 9.2 第2期教育研修システム

項目	内容
目指す人材像	デンソーを主導できるリーダー
研修の狙い	開発の課題を解決できるリーダーの育成
研修内容	<ul style="list-style-type: none">ソフトウェア工学に関する講義論文作成（課題形成、課題解決）課題発表
実施日数	20日間（8ヶ月）

(1) 目指す人材像, 研修の狙い

第1期の教育研修システムの実行結果より、目指す人材像を「デンソーを主導できるリーダー」、研修の狙いを「開発の課題を解決するリーダーの育成」へと変更した。

開発現場は、第1期教育研修システムを開発した当時の状況とは大きく変化した。品質問題や開発コストの増大、技術力の低下などの直近の開発上の問題に加え、次世代の新製品、新サービスの提供に向けた技術開発が現場の課題となっていた。さらに、対象とするシステムは大規模化し、ソフトウェアはより複雑となったため、現場の課題解決には複数の組織の協力が必要となってきた。そこで、目指す人材像をこうした社内の開発を主導できるリーダーと定義し、研修の狙いも開発の課題解決ができるリーダーへと変更した。

(2) 研修内容

第1期の教育研修システムでは、課題発表に研修で習得した内容が十分活用できていなかった。研修生は開発における問題解決の経験が少ないため、問題の原因が把握できず適切な解決策の検討ができなかったことが原因であった。研修生は講師の指導を受けずに課題形成、課題解決を進めていたため、スキル不足により十分な結果が得られなかった。また、問題を正しく分析できても、研修で習得する従来のソフトウェア工学の知識だけでは課題解決に対応できなかった。

そこで、開発における問題の原因分析を実際に行い、課題形成、課題解決を通してその成果を論文にまとめる研修科目である論文作成を第2期の教育研修システムに追加した。さらに、習得するソフトウェア工学に最新のソフトウェア技術の研修を追加することで課題解決に対応した。

9.1.3 第3期教育研修システム

第2期の教育研修システムを改善した教育研修システムである。提案する教育研修システム開発方法論を適用し、第8回、9回の教育研修システムを開発し実行した。表9.3に第3期教育研修システムの概要を示す。対象受講者は、第1,2期と同様である。

表 9.3 第3期教育研修システム

項目	内容
目指す人材像	ソフトウェア開発を主導できる技術者
研修の狙い	開発の課題を解決できる人材の育成
研修内容	<ul style="list-style-type: none">・ 技術 : ソフトウェア工学 自動車ソフトウェア工学・ 実践 : 論文作成 (課題形成, 課題解決)・ 哲学 : リーダーシップ
実施日数	25日間 (9ヶ月)

(1) 目指す人材像, 研修の狙い

第3期の教育研修システムでは、目指す人材像を「ソフトウェア開発を主導できる技術者」、研修の狙いを「開発の課題を解決できる人材の育成」と変更した。

育成する人材が活躍する領域をソフトウェア開発と明確にし、目指す人材像もリーダーだけでなく、専門職のエキスパートも含む技術者とした。これからの開発ではリーダーだけでなく、開発を主導するすべての技術者に高度ソフトウェア専門技術者が必要である。

(2) 研修内容

本稿で提案するカリキュラム設計方法に基づいて、技術、実践、哲学の研修科目を設計した。技術は、ソフトウェア工学、自動車ソフトウェア工学、実践は、課題形成、課題解決を実施する論文作成、哲学はリーダーシップに関する研修内容とした。

9.2 カリキュラム

提案するカリキュラム設計方法を用いて、第9回の教育研修システムの技術、実践、哲学のカリキュラムを設計した。設計した技術、実践、哲学のカリキュラムを以下に示す。

9.2.1 技術のカリキュラム

筆者の所属する企業では経営戦略上、最も重要な製品は車載システムである。自動車に搭載される車載システムは、高い品質と安全性が必要である。また、車載システムは、近年、多機能化によりシステムは大規模化、複雑化するばかりでなく、市場の要求により従来より短い期間での開発が求められている。こうした車載システムの特徴から、高度ソフトウェア専門技術者に必要な技術のカリキュラムである研修单元、研修科目を導出する。

表 9.4 に技術の研修单元、研修科目を示す。大規模化、複雑化するシステムへ対応するため、ソフトウェア工学の知識として「ソフトウェア開発技術」、「ソフトウェア開発マネジメント技術」を研修单元とした。また、自動車ソフトウェア工学の知識として品質と安全性を確保する「品質・安全技術」を研修单元として設定した。分析力は、研修单元を「分析技術」とし、技術の各研修单元に必要なスキルとして位置付けた。

表 9.4 に 3 つの研修单元「ソフトウェア開発技術」「ソフトウェア開発マネジメント技術」「品質・安全技術」より導出した研修科目と研修内容を示す。各研修单元の研修科目、研修内容について以下に説明する。

(1) ソフトウェア開発技術

「ソフトウェア開発技術」は、カリキュラム設計で導出したソフトウェア工学の知識に対応する。筆者の所属する企業が自動車メーカーに部品を直接納入する一次サプライヤであることを考慮し、大規模化、複雑化する車載システム開発に必要な上流工程の技術として SWEBOK[5]より「要求工学」「アーキテクチャ」を導出した。また、車載システムは車種、仕向けに対して多くのバリエーションに対応していく必要がある。そこで、「プロダクトライン」を研修科目に設定し、事業、技術、プロセスの視点からプロダクトラインを習得できる研修内容とした。

(2) ソフトウェア開発マネジメント技術

「ソフトウェア開発マネジメント技術」は、車載システムのソフトウェアの品質を維持し、より短い開発期間でソフトウェアを開発するために必要な技術である。そこで、研修科目は、開発プロセス、プロジェクトマネジメントに着目して導出した。プロジェクトのメカニズムを理解して開発プロセスを設計する「プロセス設計」、開発を改善する考え方として「TPS (Toyota Production System) とリーン開発」についての研修科目を設定した。

(3) 品質・安全技術

「品質・安全技術」は、品質、安全性を確保する技術として自動車ソフトウェア工学に求められる技術である。品質技術は、ソフトウェアの定量的な品質管理に着目し、「検証技術」、「欠陥工学」、「メトリクス」の研修科目に展開した。品質保証の視点から欠陥についての知識、品質保証活動、メトリクスについて習得する。安全技術については、設計段階での安全性確保についての科目として「安全設計」を設定した。

表 9.4 技術の研修科目

育成目標	知識/スキル		研修単元	研修科目	研修内容	
技術	知識	ソフトウェア工学	ソフトウェア開発技術	要求工学	要求工学知識体系に基づく要求工学	
				アーキテクチャ	システム/ソフトウェアアーキテクチャ	
				プロダクトライン	事業, 技術, プロセスの視点からのSPLE	
			ソフトウェア開発マネジメント技術	プロセス設計	プロジェクトのメカニズムとプロセス設計	
					TPSとリーン開発	組込み技術者のためのトヨタ生産方式
		自動車ソフトウェア工学	品質・安全技術	検証技術	ソフトウェアの価値と品質保証戦略	
				欠陥工学	欠陥エンジニアリングと品質保証活動の重要性	
				メトリクス	ソフトウェアメトリクスの活用と現場への適用	
	安全設計			車載製品に求められる安全設計の考え方		
	スキル	分析力	分析技術	技術の各研修単元に必要なスキル		

9.2.2 実践のカリキュラム

実践のカリキュラムは、カリキュラム設計で導出した高度ソフトウェア専門技術者に必要な実践の知識、スキルを習得するために、研修生が開発現場における現実の問題を取り上げ、課題形成、課題解決を進め、その成果を論文としてまとめるカリキュラムとした。このカリキュラムを進めるため必要な研修科目を「課題解決」「分析技術」「文書作成技術」の3つの研修単元で整理し、研修科目を導出した。

表 9.5 に育成目標である実践に必要な知識、スキルと研修単元の関係を示した。「課題解決」は開発現場の問題から課題形成、課題解決を進める実践的な研修単元である。実践に求められる知識、スキルの習得をすべて含んでいる。「分析技術」は課題形成における問題の原因分析に必要な技術の研修単元である。問題解決の知識を基本とし、分析力、抽象化能力の習得を目的とする。有効な解決策を検討するためには、問題の原因を正しく把握する必要がある。研修生は、現場での原因分析の経験が少ないため「分析技術」の研修単元を設定した。「文書作成技術」は論文を作成するための研修単元である。「課題解決」の成果を論文としてまとめるテクニカルライティングの技術を習得する。「文書作成技術」の研修では、技術文書の作成を通して、抽象化能力、論理構成力のスキルを習得する。

表 9.6 に3つの研修単元「課題解決」「分析技術」「文書作成技術」より導出した研修科目と研修内容を示す。研修単元から研修科目への展開について以下に説明する。

(1) 課題解決

「課題解決」の研修単元では、研修生の開発における現実の問題を取り上げ、問題の原因分析、課題形成、課題解決を通して解決策を作成し、その結果を論文としてまとめる。研修科目は「論文作成」とし、研修生は問題の原因分析から論文作成までをこの研修科目の中で進める。「論文作成」は研修生がそれぞれ個別のテーマで進めるため、研修ではその取り組みを相談やレビューで支援、サポートする。育成目標である実践に必要な知識、スキルは「論文作成」の課題解決を通して習得する。

表 9.5 実践の研修単元

育成目標	知識/スキル		研修単元		
			課題解決	分析技術	文書作成技術
実践	知識	問題解決	●	●	
	スキル	分析力	●	●	
		抽象化能力	●	●	●
		論理構成力	●		●
		問題解決力	●		

表 9.6 実践の研修科目

研修単元	研修科目	研修内容
課題解決	論文作成	課題形成, 課題解決の支援
分析技術	統計的手法	ソフトウェアメトリクスと統計手法
	モデリング	ソフトウェア工学とモデリング
	課題解決技術	課題解決に向けたアプローチとツール
文書作成技術	技術文書	技術文書の読み方, 書き方
	論文の基礎	論文の読み方, 書き方

(2) 分析技術

「分析技術」は、課題解決における問題の原因分析に必要な技術である。技術のカリキュラムでは、技術共通の研修単元として検討したが、実践のカリキュラムでは、課題形成、課題解決に必要な研修単元として扱う。研修科目は、実際の問題の分析に必要な技術である「統計的手法」、「モデリング」を設定した。「課題解決技術」の研修科目では、課題解決のアプローチ方法や考え方について学習する。

(3) 文書作成技術

「文書作成技術」は、論文を作成するための技術である。「論文作成」の研修科目で作成した課題の解決策を論文としてまとめる時に必要な技術である。研修科目には、「技術文書」、「論文の基礎」を設定した。「技術文書」では、テクニカルライティング、構造的な文書を書く技術を学ぶ。「論文の基礎」では技術論文についての基礎知識と論文作成の作法を学ぶ。

9.2.3 哲学のカリキュラム

表 9.7 に哲学の研修単元, 研修科目を示す. リーダーシップの役割を, Kotter のリーダーシップ論[15]より, 変化に対応するための方向性の設定, 人心の統合, 動機付けと定義し, リーダーシップを「マインドセット」「技術戦略」「マネジメント」の 3 つの研修単元で学習することとした. また, リーダーシップに必要なコミュニケーションに着目し, コミュニケーション能力の研修単元を「連携・協調」と設定した.

リーダーシップの研修単元である「マインドセット」はリーダーとしての心構えである. 自ら変化に対応することの必要性を認識し, リーダーの役割である方向性の設定, 人心の結合, 動機付けの重要性とその具体的な行動について理解する. 「技術戦略」の研修単元では, 組織が進むべき方向性の設定に必要な技術戦略についての知識を習得する. 「マネジメント」の研修単元では, マネジメントについて理解することにより複雑な組織へ対応する考え方を学び, 人心の統合, 動機付けについて理解する. 本来, マネジメントとリーダーシップは異なる概念だが, 現実には補完し合う行動体系である. 複雑さを増し, 変化し続ける環境でリーダーとして課題解決を進めるためにはマネジメントの習得も必要である.

「連携・協調」の研修単元では, 現場でのリーダーシップの発揮を考慮に入れ, コミュニケーション能力を向上させるために自分の意見, 考え方を簡潔にわかりやすく相手に伝える技術を習得する.

表 9.7 哲学の研修科目

育成目標	知識/スキル		研修単元	研修科目	研修内容
哲学	知識	リーダーシップ	マインドセット	リーダーの心得	技術開発リーダーの行動哲学
				世界の組込み開発	世界で活躍できる技術者の条件
			技術戦略	技術戦略と経営	ビジネスモデルと競争原理/ネット資本論
				技監(役員)講話	プロセス改革と開発戦略
			マネジメント	プロジェクトマネジメント	ソフトウェアプロジェクトマネジメントと現場力
				ITマネジメント	業務システム開発から学ぶPM力
	日本の組込み開発	ソフトウェア品質と日本の品質管理			
	スキル	コミュニケーション能力	連携・協調	コミュニケーション	ライトニングトーク

表 9.7 にリーダーシップについての研修単元である「マインドセット」「技術戦略」「マネジメント」「連携・協調」より導出した研修科目を示す. 各研修単元の研修科目, 研修内容について以下に説明する.

(1) マインドセット

「マインドセット」の研修単元では, 研修生にリーダーシップを発揮するための動機付けを行う. リーダーとして変化に対応する意味を理解し, リーダーとしての考え方, 行動を学ぶ科目として「リーダーの心得」を設定した. また, 世界で活躍できる技術者に成長する自覚を促す科目として「世界の組込み開発」を設定した. 自分自身への動機付けにより, リーダーシップの役割を担う当事者意識を醸成し, リーダーとしての行動を意識させる.

(2) 技術戦略

「技術戦略」の研修単元は、リーダーシップに求められる方向性の設定に必要な研修科目に展開した。「技術戦略と経営」の研修科目では、製品を取り巻くマーケットの市場原理、競争原理、ビジネスモデルについて習得し、技術開発の方向性について検討する。さらに、実際に社内でソフトウェアの開発戦略やソフトウェア開発のプロセス改革を担当してきた役員と、技術開発のリーダーシップについて議論する研修科目である「技監（役員）講話」を設定した。

(3) マネジメント

「マネジメント」の研修単元では、リーダーシップにおける人心の統合、動機付けに関する研修科目に展開した。現場、プロジェクトマネジメント、日本的品質管理の観点から人と組織を動かすリーダーシップについて学習する。また、リーダーシップとマネジメントの違いを明確にし、リーダーシップの本質を理解する。

「プロジェクトマネジメント」では、ソフトウェアプロジェクトマネジメントの特徴を理解し、リーダーシップがチームメンバーの動機付けに及ぼす影響と現場力について学習する。「IT マネジメント」では、大規模な業務システム開発のマネジメントからプロジェクトマネジメントの効果を理解し、リーダーの役割、行動を理解する。「日本の組込み開発」は、ソフトウェア品質、日本的品質管理の観点から具体的なリーダーシップについて学習する。

(4) 連携・協調

「連携・協調」の研修単元では、「コミュニケーション」の研修科目を設定した。ライトニングトークを導入することによりコミュニケーション能力の向上を図った。研修では、短いプレゼンテーションを繰り返し実施することにより、自分の主張を効果的に相手に伝える技術を学習する。また、プレゼンテーションのテーマをリーダーシップの内容に設定することで、リーダーシップについて考える機会を提供する。

9.3 教授法

提案する教授法設計方法を用いて、第 9 回の教育研修システムの教授法を設計した。設計した技術、実践、哲学の教授法について以下に説明する。

9.3.1 技術の教授法

技術の教授法は、7.3 節の教授法設計より実践学習を選択した。図 9.2 に技術の教授法を示す。

技術の実践学習は、事前学習、研修、事後学習で構成する。研修では、学習する技術を現場で実践するための具体的な方法や考え方をワークショップで習得する。ワークショップに必要な知識は事前学習で理解しておく。事後学習では、学習した技術の実際の開発への適用について具体的に検討する。

技術の実践学習における事前学習、研修、事後学習について以下に説明する。

(1) 事前学習

事前学習では、研修で学習する技術知識の習得、事前課題を自己学習で取り組む。知識の習得は、事前配布したテキストを読み、内容のまとめと質問事項の整理を行う。事前課題は、学習する技術についての演習問題や研修のワークショップで使用する演習課題である。事前に、演習問題や演習課題に取り組むことにより、習得する技術についての問題意識を高める。

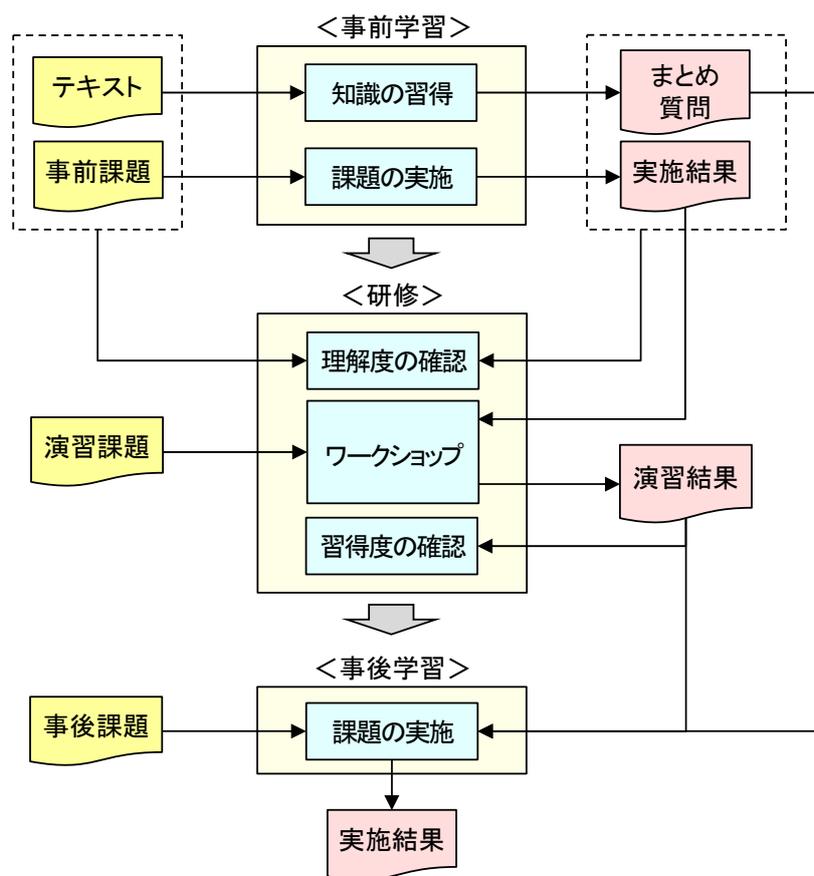


図 9.2 技術の教授法

(2) 研修

研修では、理解度の確認、ワークショップ、習得度の確認を行う。

理解度の確認は、事前学習で学習した内容についての簡単なテストやテキストについての Q&A、事前課題の確認により、研修生の習得する技術に対する理解度を確認する。理解が不十分な点に関しては、講義や Q&A を実施することで、ワークショップの開始前に研修生の学習する技術についての理解度を揃えておく。

ワークショップでは、習得する技術の現場への具体的な適用方法についての演習課題に取り組む。事前課題の結果を前提とした演習課題に取り組む場合もある。課題の内容に応じてグループワークやディスカッションを実施し、技術の本質を理解し、現場へ適用するための考え方を習得する。

習得度の確認は、ワークショップで実施した演習結果のウォークスルーや研修の最後にテストを実施することにより、研修生の理解度を確認する。ウォークスルーやテスト結果の解説では、ワークショップで学習した内容について整理する。

(3) 事後学習

事後学習では、ワークショップの演習結果を発展させ、習得した技術を実際の開発へ適用する課題について具体的に検討する。また、事前学習でまとめた質問について、研修で得た知見を整理してレポートにまとめる。

9.3.2 実践の教授法

実践の教授法は、7.3 節の教授法設計より PBL を選択した。図 9.3 に実践の教授法を示す。

実践のカリキュラムでは、研修期間を通して研修生が担当するソフトウェア開発の問題に対して、課題形成、課題解決を進め、その成果を論文にまとめ発表する。実践の教授法である PBL は、研修生が個別に進める課題形成、課題解決、論文作成の進捗に合わせて、相談やレビュー、検討会を適切な時期に実施することにより、研修生の取り組みを支援する。研修の最後には、研修生がまとめる論文についての発表会を実施する。

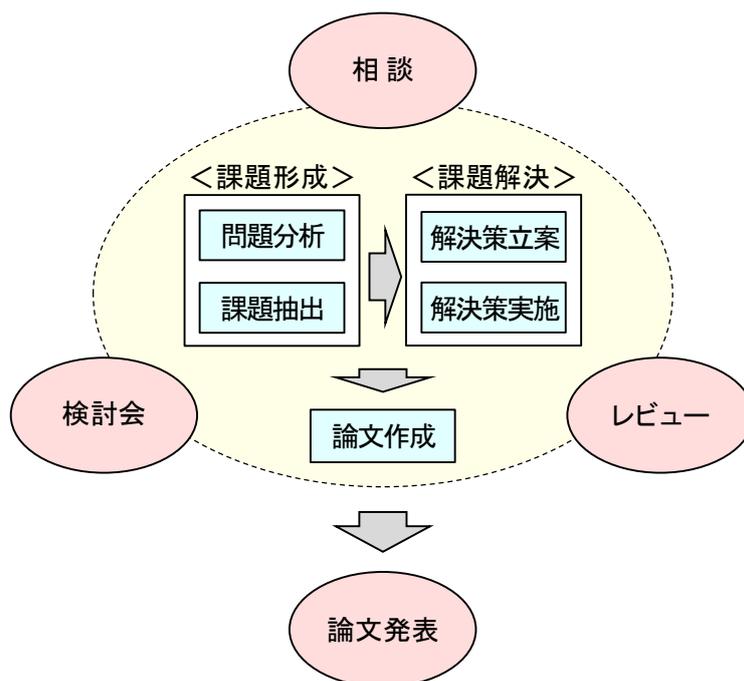


図 9.3 実践の教授法

実践の教授法における、相談、レビュー、検討会、論文発表について以下に説明する。

(1) 相談

実践における課題形成、課題解決、論文作成は、研修生が各自のテーマに対して研修と自己学習で個別に取り組む。相談は研修生から希望があれば随時実施する。相談では、研修生が課題解決を進める上での問題について講師とディスカッションを行う。講師は、問題の原因分析から論文作成に関わるすべての内容について研修生を支援する。研修時間中は相談の時間を設定するが、研修時間外はメールや研修用に構築した学習管理システムで対応する。

(2) レビュー

課題形成、課題解決、論文作成の期間の中間時期に、成果についてのレビューを講師と個別に実施する。レビューでは、進捗と課題解決の進め方の問題点を確認し、各フェーズでのゴールを決め、取り組みの方針と具体的な進め方について決定する。

(3) 検討会

課題形成、課題解決の各フェーズの最後には、研修生全員で検討会を実施する。課題形成の検討会では、問題の原因分析と課題の抽出の結果を発表し、課題解決の進め方について全員で議論する。課題解決の検討会では、課題形成から解決策の作成、実行までの取り組み内容を発表し、論文としてまとめるに当たり、不足している内容を明確にして進め方を確認する。

(4) 論文発表

論文発表は研修の最後に実施する。論文、発表についての評価、講評は、ソフトウェア担当の役員、大学教授、実践を担当した社外講師に依頼する。事前に論文を配布し、論文発表会では発表内容について、経営戦略の視点、技術の視点、教育の視点で議論を行う。発表会には、研修生の上司である部長クラスの管理職も出席し、提案する課題解決策の現場での取り組みについても検討する。

9.3.3 哲学の教授法

哲学の教授法は、7.3 節の教授法設計よりカリキュラムの内容に応じて知識学習と実践学習を選択する。リーダーシップについての研修は、技術の研修とは異なり研修生にとっては初めての内容が多いので、講師が研修生に直接指導する講義、講話による知識学習を選択した。コミュニケーションの教授法は、ライトニングトークを導入し、プレゼンテーションを実際に研修で行う実践学習として設計した。

9.3.3.1 リーダーシップの教授法

図 9.4 にリーダーシップの教授法を示す。リーダーシップの教授法は、講話、講義を中心とした知識学習として設計する。リーダーシップの教授法では、事前学習で講話資料、講義資料の予習や事前課題に取り組む。研修は講師による講話や講義が中心である。講話、講義後は、その内容についてのディスカッション、演習を実施する。事後学習では研修でのディスカッションや演習結果についての事後課題に取り組む。

リーダーシップの教授法における事前学習、研修、事後学習について以下に説明する。

(1) 事前学習

事前学習で講話資料、講義資料について予習し、内容についての質問事項をまとめる。講話や講義、ディスカッションや演習で使用する課題がある場合は事前学習で取り組む。

(2) 研修

研修では、講師による講話や講義を実施し、研修内容についてのディスカッション、演習を行う。事前学習でまとめた質問事項、事前課題は、研修時に講師に回答を確認する。ディスカッションでは、講話や講義の内容に関する課題について、講師を交えて議論、演習を進めることで、リーダーシップについての理解を深める。ディスカッション、演習のテーマとして事前課題の結果を使用する場合もある。

(3) 事後学習

事後学習では、研修でのディスカッションや演習の結果を参考にして事後課題に取り組む。事後課題は、リーダーシップについての課題である。課題に対するリーダーシップについての自己の考え方を研修内容に基づいて整理する。また、事前学習でまとめた質問については、講師から得た回答と研修で得た知見を整理してレポートとしてまとめる。

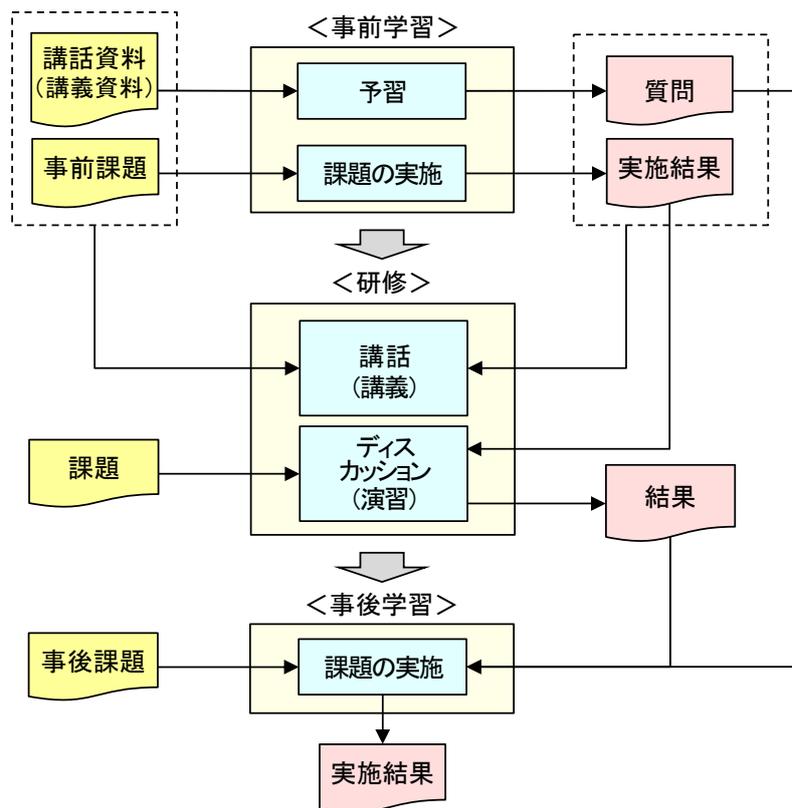


図 9.4 リーダーシップの教授法

9.3.3.2 コミュニケーションの教授法

図 9.5 にコミュニケーションの教授法を示す。コミュニケーションの教授法は、ライトニングトークによるプレゼンテーションの実践学習として設計した。

研修では、研修生がリーダーシップについてのライトニングトークを実際に行い、発表の構成や内容、プレゼンテーションスキルについて指導と振り返りを実施する。ライトニングトークは、リーダーシップに必要なコミュニケーション能力の向上を図るために導入した。そこで、ライトニングトークのテーマは、リーダーの心得、考え方、判断基準などリーダーシップを発揮する上で最低限考えておくべき内容とした。コミュニケーションの研修は、研修期間中に数回実施する。

コミュニケーションの教授法における事前学習と研修について以下に説明する。

(1) 事前学習

事前にテーマを展開し、研修で使用するライトニングトークの資料を作成する。前回の振り返り結果がある場合は、その内容を発表資料に反映する。

(2) 研修

研修では、全員がライトニングトークを実施し、講評、指導の後、各自振り返りを行う。

ライトニングトークの時間は 3 分とする。研修生、講師は、各ライトニングトークを発表後の 1 分間で評価する。ライトニングトークはビデオカメラで撮影し、振り返りに使用する。

講評／指導では，講師が研修生のライトニングトークで気づいた点について指導する．発表内容の構成や論理展開，発表資料の作り方，プレゼンテーションスキルについて，研修生が実際に使用した発表資料を例に具体的に説明する．

振り返りでは，評価結果，ビデオ，講師の講評／指導に基づいて，各自が自分のライトニングトークを振り返り，改善点を振り返り結果にまとめる．振り返り結果は，次回のライトニングトークに反映する．

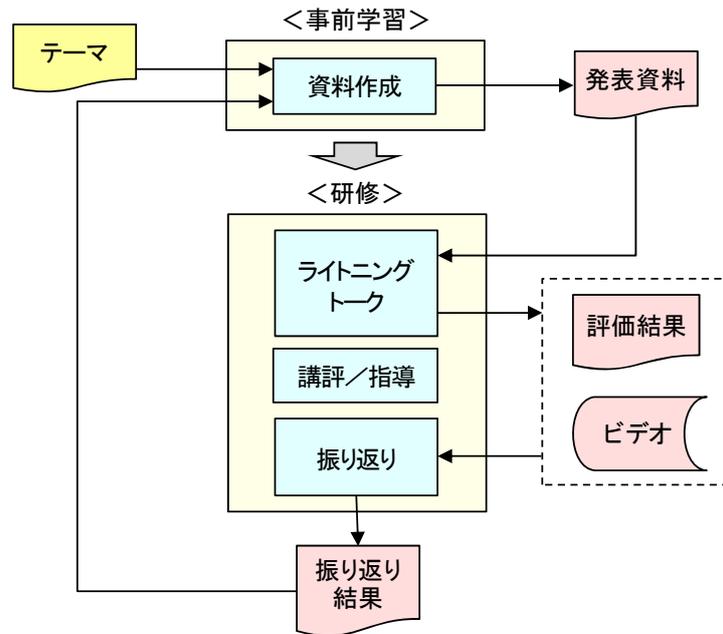


図 9.5 コミュニケーションの教授法

9.4 研修計画

表 9.8 に第 9 回の教育研修システムの研修日程を示す。図 9.6 は研修スケジュールである。高度ソフトウェア専門技術者育成の教育研修システムを 9 か月間、全 25 日間の研修として開発した。研修では、育成目標である技術、実践、哲学の研修科目を並行して実施する。

研修は、実践の課題形成、課題解決の研修科目である論文作成を中心に計画した。技術、実践の研修科目は、研修内容が論文作成における、課題形成、課題解決、論文作成に反映できるよう実施時期を設定した。

技術、実践、哲学の研修計画について以下に説明する。

9.4.1 技術の研修計画

技術の研修科目は、実践の課題形成、課題解決に必要な技術、ツールを提供する。そこで、実践の課題形成が終了する中間発表までに課題形成、課題解決に必要なソフトウェア開発技術や品質技術に関する研修科目を実施する。要求工学、アーキテクチャ、プロダクトライン、検証技術、メトリクス、欠陥工学の研修科目を中間発表前に実施するよう計画した。

中間発表後は、ソフトウェア開発のマネジメント技術や設計知識に関する研修科目を実施する。プロセス設計や TPS、リーン開発など効率的に開発を進めるためのマネジメント技術と安全設計に関する設計知識の研修科目を中間発表後に計画した。

9.4.2 実践の研修計画

実践の研修計画は、課題解決が研修全体を通して効果的に実施できるよう開講式から中間発表までは課題形成、中間発表から課題発表会までは課題解決、論文作成の期間とした。分析技術の研修科目である統計的手法、モデリングは、課題形成の期間に実施する。分析技術の課題解決技術と文章作成技術の研修科目は、課題解決、論文作成の期間に実施するよう計画した。

開講式後に実施するポジションペーパー発表会は、研修生が担当する開発の問題とその原因、開発課題、解決策、予想される効果について発表する。ポジションペーパーにより研修生の課題解決に対する認識を確認する。課題形成では、ポジションペーパーをベースラインとして問題の原因分析、課題形成に取り組み、その内容は中間発表で発表する。中間発表から課題発表会までは、課題に対する解決策を作成し、その成果を論文にまとめる。論文については課題発表会で発表する。

9.4.3 哲学の研修計画

哲学の研修科目は、リーダーシップに関する研修科目とライトニングトークを導入したコミュニケーションの研修科目で構成する。

技術、実践の研修科目が研修の前半に集中するため、リーダーシップに関する研修は中間発表後の研修の後半に計画した。但し、リーダーとしてのマインドセットに必要なソフトウェア担当役員（技監）の講話や、課題解決へのモチベーション向上を狙いとした研修科目は研修の初期に実施する。

コミュニケーションの研修は、研修全体を通して 8 回実施する。研修を通して、ライトニングトークに継続的に取り組むことによりコミュニケーション能力の向上を図る。

表 9.8 研修日程

日程	内容	
	AM	PM
第1回	開講式	ポジションペーパー発表会
第2回	実践(論文作成)	哲学(技監講話)
第3回	技術(アーキテクチャ(1))	
第4回	技術(アーキテクチャ(2))	
第5回	哲学(LT)	実践(論文作成) 哲学(日本の組込み開発)
第6回	技術(要求工学)	
第7回	哲学(LT)	実践(論文作成) 技術(メトリクス)
第8回	実践(統計的手法)	
第9回	哲学(LT)	実践(論文作成) 技術(欠陥工学)
第10回	実践(モデリング)	技術(プロダクトライン)
第11回	実践(モデリング)	実践(論文作成)
第12回	哲学(LT)	実践(論文作成) 技術(検証技術)
第13回	中間発表	
第14回	実践(課題解決技術)	
第15回	哲学(LT)	実践(論文作成) 技術(TPSとリークン開発)
第16回	技術(プロセス設計)	
第17回	哲学(LT)	実践(論文作成) 技術(安全設計)
第18回	実践(技術文書)	
第19回	哲学(LT)	実践(論文作成) 哲学(技術戦略と経営)
第20回	実践(論文作成) 実践(論文の基礎)	
第21回	実践(論文の基礎) 哲学(プロジェクトマネジメント)	
第22回	実践(論文作成)	
第23回	哲学(世界の組込み開発) 哲学(リーダーシップ)	
第24回	哲学(LT)	実践(論文作成) 哲学(ITマネジメント)
第25回	課題発表会 修了式	

LT:ライトニングトーク

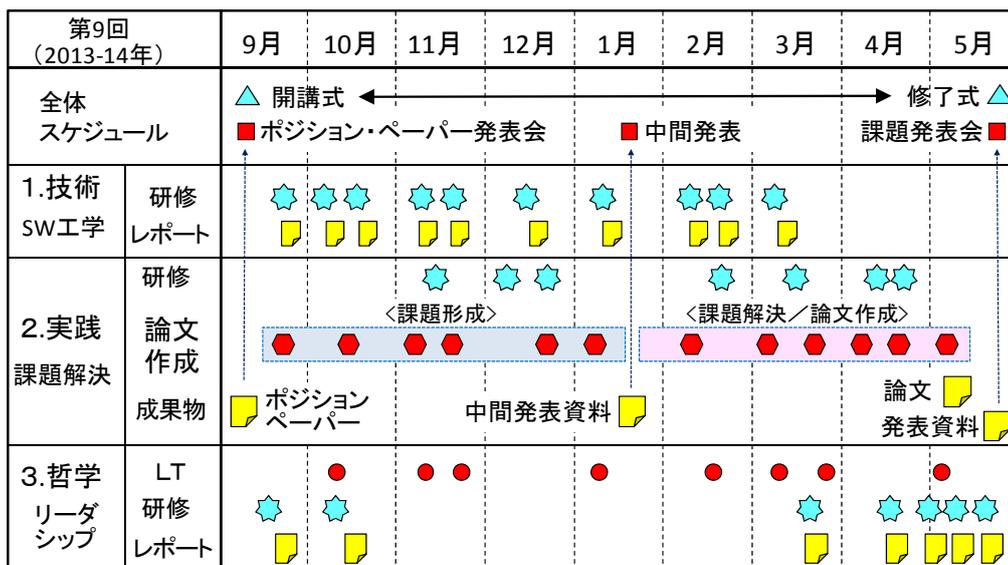


図 9.6 研修スケジュール

9.5 実際の研修

実際の研修の様子を図 9.7, 図 9.8, 図 9.9, 図 9.10 に示す.



図 9.7 講義



図 9.8 ワークショップ

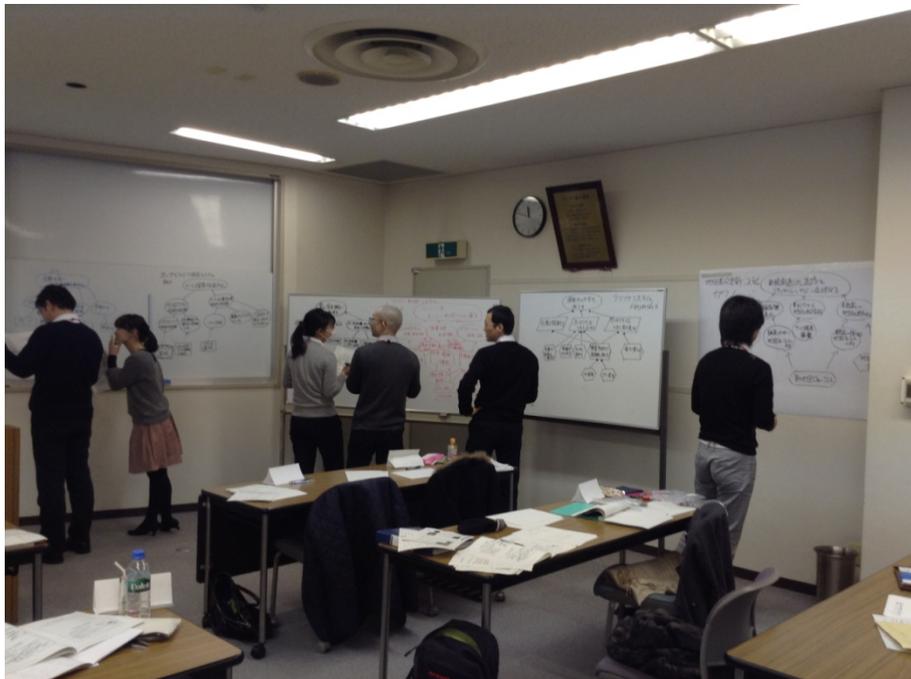


図 9.9 演習



図 9.10 ウォークスルー

10 評価結果

第8章の教育研修システムの評価方法に従い、開発した教育研修システムを評価した結果を示す。

10.1 カリキュラム評価

カリキュラム評価は、これまで開発した第1～3期の教育研修システムのカリキュラムに対して実施した。各期の最後に実施した最も完成度の高い第4回、第7回、第9回の教育研修システムを評価した。

10.1.1 研修設計時の評価

カリキュラムの各育成目標である、技術、実践、哲学の研修時間を基準とするカリキュラムの研修時間と比較することにより、育成目標に対するカリキュラムの有効性を評価した。

(1) 評価方法

カリキュラム評価の基準は、本稿で提案する教育研修システム開発方法論を適用して開発した第9回の教育研修システムのカリキュラムとする。第4回、第7回、第9回の教育研修システムのカリキュラムを8.2節の式(1)を用いてラーニングユニットにより E_c で評価する。評価は、育成目標である技術、実践、哲学の研修科目について実施する。

(2) 評価結果

図10.1に評価結果を示す。高度ソフトウェア専門技術者の育成目標である技術、実践、哲学を3軸とし、各教育研修システムの育成目標について E_c の値を示した。

第4回のカリキュラムでは技術の育成目標しか考慮されていないが、第7回のカリキュラムでは実践の育成目標の内容が追加されている。さらに第9回のカリキュラムでは、哲学の育成目標を検討した教育研修システムとなっている。図10.1よりカリキュラムは育成目標を達成するよう改善されていることがわかる。

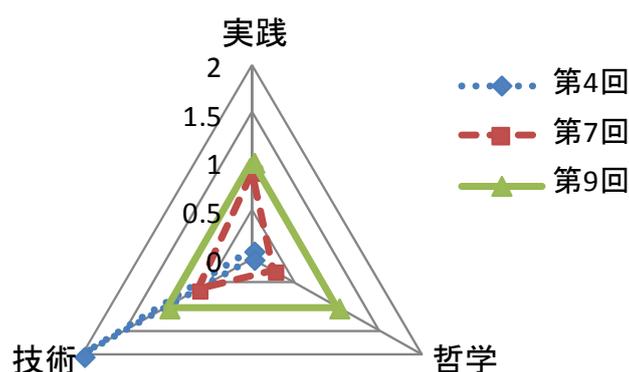


図 10.1 カリキュラム評価

10.1.2 研修実行時の評価

カリキュラムを受講した研修生のパフォーマンスを比較することにより、カリキュラムの有効性を評価した。

(1) 評価方法

受講生のパフォーマンスは、研修生が作成した論文、発表資料の有効性で評価する。論文、発表資料の有効性は8.2節の式(2)の E_p で評価する。表 10.1 は、評価対象とする教育研修システム実行時の研修生の人数である。

表 10.1 研修生の人数

研修	第4回	第7回	第9回
人数	23人	9人	14人

(2) 評価結果

図 10.2 に評価結果を示す。研修設計時に育成目標を達成するよう設計した教育研修システムのカリキュラムは、研修の実施結果である研修生の論文、発表資料の有効性も高いことがわかる。第9回の教育研修システムは本稿で提案する開発方法論を適用して開発していることから、図 10.2 は提案する教育システム開発方法論の有効性を示している。

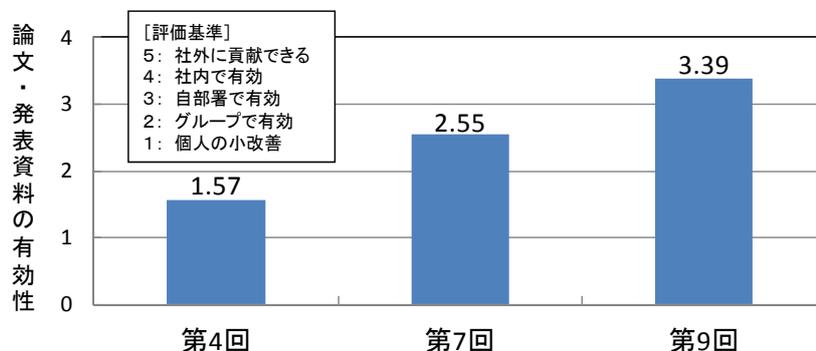


図 10.2 論文、発表資料の評価

10.2 教授法評価

提案する教授法設計方法の妥当性を確認するため、第3期の第8回と第9回の教育研修システムの教授法を研修設計時、研修実行時に評価した。第9回の教育研修システムは、第8回の教育研修システムと同じ研修科目を実施している。第9回の教育研修システムは、技術と哲学の研修科目について提案する教授法設計方法を適用した。

10.2.1 研修設計時の評価

研修設計時に技術と哲学の研修科目について教授法を評価した。

(1) 評価方法

第8回と第9回の教育研修システムにおける技術と哲学の教授法を、表7.5の学習目標により評価する。各研修科目に対して教授法が達成する学習目標を評価し、学習目標の科目数の分布を第8回と第9回の教育研修システムで比較する。

(2) 評価結果

図10.3に評価結果を示す。第8回の教育研修システムでは教授法が達成する学習目標の平均は2.0であった。第8回の教授法設計は従来の教授法を継続している。提案する教授法設計を導入して設計した第9回の教育研修システムの学習目標の平均値は3.5と向上した。

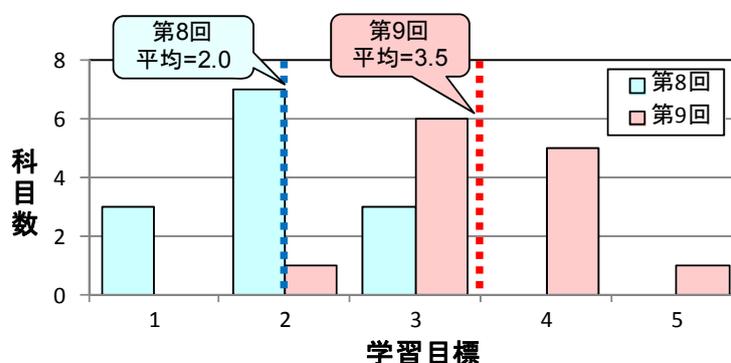


図 10.3 研修設計時の教授法評価

10.2.2 研修実行時の評価

研修実行時に技術と哲学の研修科目について教授法を評価した。

10.2.2.1 技術の教授法評価

(1) 評価方法

技術の教授法は、第9回の教育研修システムに実践学習を適用した要求工学、プロダクトライン、プロセス設計の研修科目を、知識学習が中心である第8回の教育研修システムの同科目と比較し評価する。教授法は、研修生のパフォーマンスを事後レポートで評価する。評価基準は、表7.5の学習目標を各科目の技術内容に対応して作成した評価指標を使用する。

(2) 評価結果

図10.4に評価結果を示す。提案する教授法設計方法を導入することにより、事後レポートの評価値として平均1.19(64.5%)の向上があった。技術の研修は学習目標3,4,5を達成するために、7.3節の教授法設計より実践学習を選択した。図10.4より技術の研修については、学習目標3,4,5を達成するには実践学習が適していると言える。

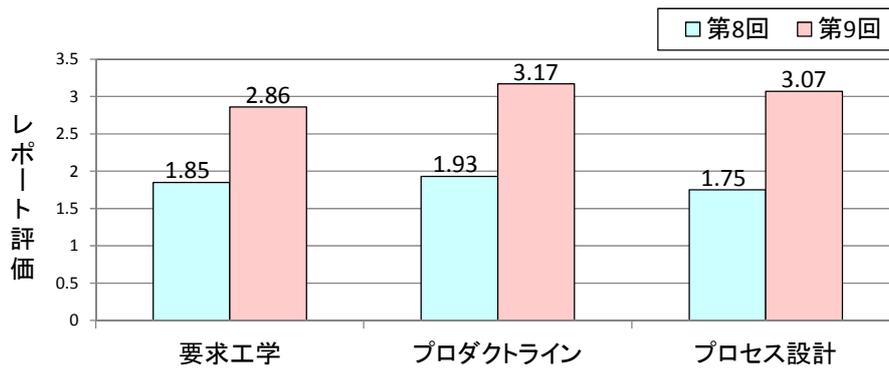


図 10.4 技術の教授法評価

10.2.2.2 哲学の教授法評価

(1) 評価方法

哲学の教授法は、第9回の教育研修システムで研修科目として実施したコミュニケーションを評価する。教授法はコミュニケーションの研修にライトニングトークを導入した実践学習を評価する。ライトニングトークは、表7.5の学習目標に基づいて作成した、主張の妥当性、主張の説得力、コミュニケーション能力について3段階で評価する。コミュニケーションの研修は第9回の教育研修システムの研修期間中に合計8回実施した。

(2) 評価結果

評価結果を図10.5に示す。研修期間中に実施した8回のコミュニケーションの研修科目の第1回と第8回のライトニングトークの評価結果を示す。第1回と第8回では、平均として0.91(72.9%)の研修生のパフォーマンスの改善が見られた。特にコミュニケーション能力の改善が90.5%と最大であった。コミュニケーションの研修については、実践学習を繰り返し実施する教授法が高い学習目標の達成には効果的である。

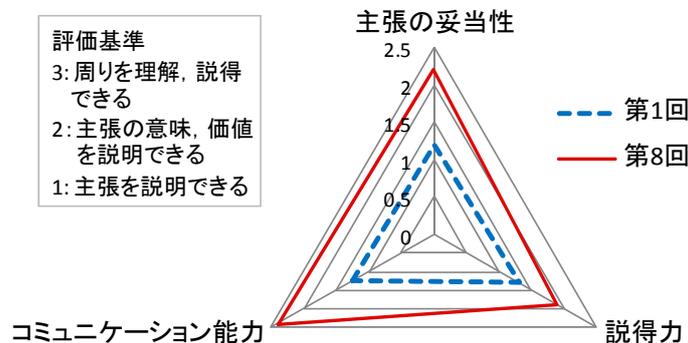


図 10.5 哲学の教授法評価

10.3 育成評価

第9回の教育研修システムに対して実行時に育成評価を実施した。

(1) 評価方法

提案する開発方法論を適用して開発した第9回の教育研修システムの研修を受講した研修生のパフォーマンスを評価する。研修生のパフォーマンスは各育成目標の研修科目に対する研修生の成果を、8.4.2節に示す評価対象に対して評価する。評価基準は、表7.5の学習目標を各科目の技術内容に対応して作成した評価指標を使用する。

(2) 評価結果

図10.6に評価結果を示す。第9回の教育研修システムの研修設計時のカリキュラム評価の結果も同じ軸上に示した。研修設計時のカリキュラムは、表7.5の学習目標に対するカリキュラムの達成能力で評価した。

育成評価の結果より、提案する開発方法論を適用した教育研修システムはどの育成目標についても学習目標の学習レベル3を達成している。学習目標の定義より、開発した教育研修システムは、自部署の開発課題を解決できる高度ソフトウェア専門技術者を育成できていると言える。しかし、育成評価の値は、どの育成目標に対してもカリキュラム評価の値までは達していない。これは、テキストの難易度や講師の教え方、教授法などのカリキュラム以外に問題があったためと考えられる。

提案する開発方法論を適用することにより、カリキュラム評価と育成評価に同じ評価指標を使用でき、評価結果を同じ評価指標で比較して評価できる。カリキュラム評価と育成評価の差は、カリキュラム設計以降の問題として捉えることができ、教育研修システム開発の改善点の分析が可能となる。

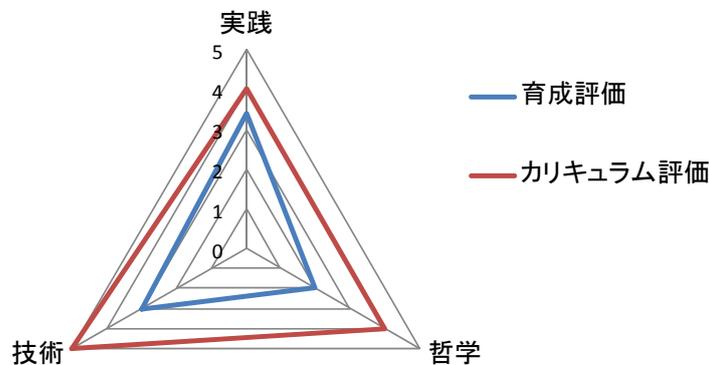


図 10.6 育成評価

11 考察

11.1 教育研修システム開発方法論の意義

従来研修開発方法で、高度ソフトウェア専門技術者を育成する研修を開発するには次の2つの課題がある。

- (1) 目標とする人材像は企業固有である
- (2) 人材像に求められる要求は複雑である

提案した教育研修システム開発方法論は、上記の課題に対して教育研修をソフトウェアと考えることにより、教育研修システムの開発をソフトウェアシステムの開発の問題に帰着する。これにより、上記(1)(2)の人材像を満たす教育研修システムの開発にソフトウェア工学の技術が応用できる。特に、要求工学におけるゴール指向分析を適用することにより、高度ソフトウェア専門技術者の人材像に対する複雑な要求を体系的に獲得できる。

提案した開発方法論によって、従来は担当者の経験に依存していた教育研修の開発が体系的に遂行できるようになった。さらに、高度ソフトウェア専門技術者に対する要求は、教育研修システムに対する要求として扱うことができるので、提案した開発方法論により複雑な人材要求に対する教育研修システムを統一的に開発することを可能にした。

11.2 人材像のゴール分析方法の意義

従来研修開発[1][7][11][18][19][22][23]では、人材像を定義するための具体的な方法論は確立されていない。実際は、研修の企画担当者が、経営層や現場の技術者など研修開発に関わるステークホルダと話し合いを通して、目標とする人材像を決めるのが一般的である。そのため、高度ソフトウェア専門技術者の人材像を体系的に定義できない。さらに、人材像に求められる要求は複雑であるため、従来方法では技術に関する一面的な育成目標となり、企業の人材に対する要求を包括的に反映するのは困難である。

提案した開発方法論では、企業固有の高度ソフトウェア専門技術者の人材像を要求工学におけるゴール指向分析を適用して定義する。人材に求められる複雑な要求に対して、人材像を技術、実践、哲学の3つのサブゴールに分けることにより、人材像を人材育成ゴールモデルとして構造化し、かつ視覚化することを可能にした。さらに、人材育成ゴールモデルのサブゴールを実現するタスクを特定することによって、人材像を達成するカリキュラム、教授法への設計へと展開できる。これにより、目標とする人材像に基づいた研修開発を可能にした。

11.3 カリキュラム設計方法の意義

従来カリキュラム設計方法では、高度ソフトウェア専門技術者に求められる知識やスキルを人材像から導出していないため、カリキュラムを通して習得すべき内容が一般的なソフトウェア技術となっている。これでは、求める人材像に適したカリキュラムを実現できないだけでなく、企業が求める高度ソフトウェア専門技術者を育成できるカリキュラムかどうか保証できない。

提案したカリキュラム設計方法では、人材育成ゴールモデルより導出した高度ソフトウェア専門技術者に必要な知識、スキルに基づいてカリキュラムを包括的に設計できる。これにより、求める人材像、育成目標と研修生が習得する知識、スキルの関係が明確になり、人材像に適したカリキュラムを合理的に設計できる。

近年、カリキュラム設計法としてCDIOアプローチが注目されている[6]。CDIOアプローチは、技術習得のシラバスが実践中心に定義されているので、大学のカリキュラム設計には有効である。しかし、カリキュラムを通して習得する内容は、一般的な技術者育成の範囲に限られており、高度ソフトウェア専門技術者に求められる企業固有の人材要求を満たすことはできない。提案したカリキュラム設計方法では、人材育成ゴールモデ

ルを適用することで、企業における高度ソフトウェア専門技術者に対する要求からカリキュラムを設計する。習得内容は、技術に加えて課題解決やリーダーシップについての人材要求も満たすカリキュラム設計が可能である。

カリキュラムに関する技術の知識体系として、IT スキル標準や SWEBOK[5]がある。両者とも、ソフトウェア技術の知識体系として、技術のカリキュラム設計時に参照できる。しかし、高度ソフトウェア専門技術者に必要な実践、哲学のカリキュラムの内容は含んでいない。

提案したカリキュラム設計方法は、多面的な高度ソフトウェア専門技術者の人材要求に対応できるよう、育成目標を技術から実践、哲学まで拡張している点に意義がある。

11.4 教授法設計方法の意義

教授法設計方法の関連研究として、会計分野の教授法選択に関する研究がある[3]。会計学の教授法設計は、習得内容が知識に限定されているため教授法の選択は比較的容易である。一方、高度ソフトウェア専門技術者には、人材像に対する要求より知識だけでなく、問題の原因分析や解決方法の提案を自ら創出することが求められる。高度ソフトウェア専門技術者育成の教育研修システムには、会計分野の知識中心の教授法とは異なる教授法が必要である。

企業研修における問題解決を目的にした教授法として GE の Work Out が知られている[24]。Work Out では、研修生が社内の課題を設定し、課題に対する解決法の提案をゴールとする課題解決のアプローチをとる。Work Out は、高度ソフトウェア専門技術者の育成目標である実践と共通の問題を扱っている。しかし、高度ソフトウェア技術者に求められる要求は、実践の課題解決に加えて哲学のリーダーシップなども含む。Work Out のアプローチだけでは、高度ソフトウェア専門技術者に対する要求を満たすには不十分である。一般的に、有効な問題解決型の教授法として PBL が広く知られている[20]。しかし、PBL も Work Out と同様に、哲学を含む多様な人材要求を満たすには限界がある。

提案した教授法設計方法は、技術、実践、哲学の異なる育成目標に対して、有効な複数の教授法を適切に組み合わせることが可能な点に意義がある。

12 今後の課題

本研究を発展させていくために、次の3点を今後の課題としてあげる。

12.1 教授法設計方法の検討

提案した教授法設計方法では、人材育成ゴールモデルより教授法を導出し、カリキュラム設計で導出した高度ソフトウェア専門技術者に必要な知識、スキルとのポートフォリオを作成する。そして、ポートフォリオを活用して教育研修システムに適した教授法を設計する。

知識、スキル、教授法は、同じ人材育成ゴールモデルの戦術ゴールから導出している。今回提案した教授法設計方法は、知識、スキルと教授法の関係に基づいて教授法を設計している。さらに戦術ゴールと知識、スキル、教授法の関係を活用できれば、より効果的な教授法の設計が可能となる。

12.2 研修のスケジュールリング技術の確立

開発した高度ソフトウェア専門技術者の第9回の教育研修システムは、全25日間の研修コースである。研修コースは、育成目標である技術、実践、哲学の研修科目を並行して実施し、最後に論文を提出後、課題発表を実施する。今回の研修コースは、実践の論文作成を中心に育成目標間の関係に着目して研修科目のスケジュールリングを実施した。

さらに効果的な研修コースを構築するためには、研修科目に着目したスケジュールリング技術の確立が必要である。技術、実践、哲学の育成目標間の関係だけでなく、研修科目間の関係や育成目標と研修科目の内容に着目して体系的なスケジュールリング技術が確立できれば、より効果的な教育研修システムの開発が可能となる。教育研修システムを育成目標であるサブシステムと、研修科目であるモジュールから構成するシステムと捉えれば、研修科目のスケジュールリングはシステムの動的な振る舞いの設計と考えることができ、工学的な設計手法が適用できる可能性がある。

12.3 開発方法論の他分野への適用と修了生の評価

本論文では、提案した教育研修システムの開発方法論を企業における高度ソフトウェア専門技術者の育成研修に適用し、その有効性を研修結果で評価した。本開発方法論をより発展させるために、他分野への適用と研修の修了生に対する評価方法の確立が必要である。

開発方法論の他分野への適用として、ソフトウェア工学分野における、アーキテクト、品質保証技術者、中堅技術者の教育研修システムに対して提案した開発方法論の有効性を検討する。さらに、ソフトウェア工学以外の科目に対して提案した開発方法論の適用を進める。他分野への適用により完成度を高め、普遍性の高い開発方法論を構築する。

本論文では、提案した開発方法論を適用した教育研修システムの実行結果で評価した。しかし、企業における高度ソフトウェア専門技術者育成の最終的な目標は、研修の修了生が開発現場の課題を実際に解決し、経営戦略の達成に貢献することである。現在の教育研修システムの評価の枠組みを拡張し、修了生の現場での行動変容や課題解決の評価結果をフィードバックすれば、包括的な教育研修システムの評価が可能となる。

13まとめ

本研究では、高度ソフトウェア専門技術者の効果的な育成を目指した教育研修システムの開発方法論を提案した。提案した開発方法論は、教育研修をソフトウェア工学の新たな課題として捉え、ソフトウェア工学の原理と技術を応用することにより、目標とする高度ソフトウェア専門技術者の人材像から体系的に教育研修システムを開発する。

教育研修システムの開発プロセスは、ソフトウェア開発プロセスに対応させることにより、研修要求定義、研修設計、研修実行、評価で構成した。研修要求定義では、ゴール指向分析に基づき育成すべき人材像から育成目標を人材育成ゴールモデルとして導出する方法を提案した。研修設計では、高度ソフトウェア専門技術者に必要な知識、スキルを導出し、カリキュラムを設計する方法を提案した。教授法設計では、人材育成ゴールモデルから導出した教授法を組み合わせ、育成目標に適した教授法を設計する方法を提案した。

提案した教育研修システム開発方法論を企業における高度ソフトウェア専門技術者の育成研修に適用し、その有効性を示した。評価結果から、提案した開発方法論は、育成すべき人材像を達成する研修開発に有用であることを確認した。

本論文で提案した開発方法論は、関連研究と照らして、次の点でソフトウェア工学と教育工学において意義があると考えられる。

ソフトウェア工学においては、研修システムをソフトウェアシステムとして捉えることにより、研修システムの開発にソフトウェア工学の技術を拡張、応用できることを示した。これは、ソフトウェア工学の拡張に貢献するものといえる。

教育工学においては、ソフトウェア工学の技術を導入し、カリキュラム設計、教授法設計における工学的な方法論の導入を可能とすることにより、教育工学の発展に資するものと考えられる。

謝辞

本研究は、南山大学大学院数理情報研究科数理情報専攻青山研究室において、青山幹雄教授のご指導の下に実施されたものです。本研究の遂行にあたり、終始手篤いご指導、ご援助を賜りました、青山幹雄教授に心より厚く御礼申し上げます。

また、本論文の審査を頂きました、南山大学大学院数理情報研究科数理情報専攻、野呂昌満教授、阿草清滋教授におかれましては、本論文について詳細なアドバイスを頂きました。ここに深甚な謝意を表します。

本論文について共に研究し、研修開発の現場における提案方法論の実用性を検討して下さいました、株式会社デンソーの足立久美担当次長、株式会社デンソー技研センターの上杉卓司課長には、深く感謝の意を表します。さらに、本研究について多くの有益なご助言を頂きました南山大学、菊島靖弘客員教授に感謝の意を表します。

社会人学生として研究の機会を下さいました株式会社デンソー技研センター、湯川晃宏社長、本研究を進めるに当たりご支援頂きました株式会社デンソー、村山浩之技監に心から御礼申し上げます。

最後に、日々の研究活動を心身両面に渡って支えてくれた家族に心から感謝します。

参考文献

- [1] 青山 幹雄, 蜂巢 吉成, 沢田 篤史, 野呂 昌満, ソフトウェア工学科の誕生, 情報処理, Vol. 50, No. 9, Sep. 2009, pp. 906-913.
- [2] J. Bergmann, and A. Sams, Flip Your Classroom, International Society for Technology Education, 2012.
- [3] S. E. Bonner, Choosing Teaching Methods Based on Learning Objectives, Issues in Accounting Education, Vol. 14, No. 1, Feb. 1999, pp. 11-39.
- [4] P. Bourque, L. Buglione, A. Abran, and A. April, Bloom's Taxonomy Levels for Three Software Engineer Profiles, Proc. of IEEE STEP 2004, Sep. 2004, pp. 123-129.
- [5] P. Bourque, and R. E. Fairley, SWEBOK, Version 3.0, IEEE Computer Society, 2014, <http://www.computer.org/portal/web/swbok/v3guide>.
- [6] E. F. Crawley, J. Malmqvist, S. Östlund, D. Brodeur, and K. Edström, Rethinking Engineering Education, 2nd ed., Springer, 2014.
- [7] H. J. C. Ellis, and G. W. Hislop, Techniques for Providing Software Engineering Education to Working Professionals, Proc. of 34th ASEE/IEEE Frontiers in Education Conference, Vol. 2, Oct. 2004, pp. 19-24.
- [8] R. M. Gagné, W. W. Wager, K. C. Golas, and J. M. Keller, Principles of Instructional Design, 5th ed., Wadsworth Pub Co, 2004 [岩崎 信, 鈴木 克明 (監訳), インストラクショナルデザインの原理, 北大路書房, 2007].
- [9] G. C. Gannod, J. E. Burge, and M. T. Helmick, Using the Inverted Classroom to Teach Software Engineering, Proc. of ICSE '08, ACM, May 2008, pp. 777-786.
- [10] Harvard Business Review (ed.), Developing Hidden Capability in Your People, Harvard Business School Press, 2006 [DIAMOND ハーバード・ビジネス・レビュー編集部 (監訳), 人材育成の戦略, ダイアモンド社, 2007].
- [11] O. Hazzan, T. Lapidot, and N. Ragonis, Guide to Teaching Computer Science, Springer, 2011.
- [12] 梶田 叡一, 教育評価, 第2版補訂2版, 有斐閣, 2010.
- [13] R. S. Kaplan, and D. P. Norton, Strategy Maps: Converting Intangible Assets into Tangible Outcomes, Harvard Business School Press, 2004.
- [14] K. Kobata, T. Uesugi, H. Adachi, and M. Aoyama, Software Engineering Education Program for Software Professionals of High Competency at DENSO, Proc. of APSEC 2013, IEEE Conference Publishing Services, Dec. 2013, pp. 117-122.
- [15] J. P. Kotter, John P. Kotter on What Leaders Really Do, Harvard Business School Press, 1999 [黒田 由貴子, 有賀 裕子 (訳), リーダーシップ論, 第2版, ダイアモンド社, 2012].
- [16] A. van Lamsweerde, Goal-Oriented Requirements Engineering: A Guided Tour, Proc. of IEEE RE 2001, Aug. 2001, pp. 249-262.
- [17] M. J. Marquardt, Optimizing the Power of Action Learning, Building Teams and Transforming Organizations, 2nd ed., Nicholas Brealey, 2011.
- [18] R. J. Marzano, and J. S. Kendall, The New Taxonomy of Educational Objectives, 2nd ed., Corwin, 2007 [黒上 晴夫, 泰山 裕 (訳), 教育目標をデザインする, 北大路書房, 2013].
- [19] 中原 淳, 研修開発入門, ダイアモンド社, 2014.
- [20] M. J. O'Grady, Practical Problem-Based Learning in Computing Education, ACM Transactions on Computing Education, Vol. 12, No. 3, Article 10, Jul. 2012, 16 pages.
- [21] F. Paulisch, and P. Zimmerer, A Role-Based Qualification and Certification Program for Software Architects: An Experience Report from Siemens, Proc. of ICSE 2010, IEEE Computer Society, pp.

21-27.

- [22] M. Shaw, Software Engineering Education: A Roadmap, The Future of Software Engineering, ACM Press, 2000, pp. 373-380.
- [23] D. Tanner, and L. Tanner, Curriculum Development: Theory into Practice, 4th ed., Prentice Hall, 2007.
- [24] D. Ulrich, S. Kerr, and R. Ashkenas, The GE Work-Out, McGraw-Hill, 2002 [高橋 透, 伊藤 武志 (訳), GE 式ワークアウト, 日経 BP, 2003].
- [25] 山下 徹, 高度 IT 人材育成への提言, NHK 出版, 2007.
- [26] J. Valacich, J. Geroge, and J. Hoffer, Essentials of Systems Analysis and Design, 5th ed., Prentice Hall, 2011.

研究業績

- [1] Keiji Kobata, Takuji Uesugi, Hisayoshi Adachi, and Mikio Aoyama, Software Engineering Education Program for Software Professionals of High Competency at DENSO, Proc. of 20th Asia-Pacific Software Engineering Conference (APSEC 2013), IEEE Conference Publishing Services, Dec. 2-5, 2013, Bangkok, Thailand, pp. 117-122 [Best Paper Award in Industry Track].
- [2] 古畑 慶次, 上杉 卓司, 足立 久美, 青山 幹雄, 高度ソフトウェア専門技術者育成のためのカリキュラム開発方法論の提案と評価, ソフトウェアエンジニアリングシンポジウム 2014 (SES2014) 論文集, Sep. 1-2, 2014, 情報処理学会, 芝浦工業大学, pp. 15-20.
- [3] Keiji Kobata, Takuji Uesugi, Hisayoshi Adachi, and Mikio Aoyama, A Curriculum Development Methodology for Professional Software Engineers and Its Evaluation, Proc. of IEEE Int'l Conference on Teaching, Assessment, and Learning for Engineering (TALE 2014), IEEE, Dec. 8-10, 2014, Wellington, New Zealand, 8 pages.
- [4] 古畑 慶次, 上杉 卓司, 足立 久美, 青山 幹雄, 高度ソフトウェア専門技術者育成のためのゴール指向に基づく教授法設計方法の提案, ソフトウェア工学の基礎 XXI (第 21 回ソフトウェア工学の基礎ワークショップ(FOSE 2014)), レクチャーノート/ソフトウェア学 Vol. 40, 近代科学社, Dec. 11-13, 2014, 日本ソフトウェア科学会, 霧島国際ホテル, pp. 251-260.
- [5] 今枝 誠, 古畑 慶次, デンソーにおける人づくり価値づくり物づくり, 日科技連出版, 2013.