

プログラミング演習を対象としたナレッジ・マネジメントの提案  
Knowledge Management on Programming Practice

南野謙一 阿部昭博 渡邊慶和  
Ken'ichi Minamino Akihiro Abe Yoshikazu A. Watanabe

### 1. まえがき

岩手県立大学ソフトウェア情報学部では、1年次より研究室に学生を配属し、小グループでマンツーマン方式を取るプログラミング演習を実施している。プログラミング演習において学生は、少人数の利点を活かし教員と学生だけではなく学生同士（同学年および上級生）でのコミュニケーションを行いながら、互いの持っている知識を共有し、新しい知識を創造しながら、課題を解決していくことができる。しかしながら、効果のある活発な知識の創造を促進するためには、基礎となる知識の共有および発見、学生同士で課題解決を行うためのアイデア発想を支援する必要がある。このような支援を研究室における環境として実現することにより、学生同士の主体的な学習を行うことが可能となる。

本研究では、ナレッジ・マネジメントの理論[1][2]を応用し、プログラミング演習における学生の知識創造を支援することにより、創意工夫を凝らしたソフトウェアを開発できる人材を育成することを目的としている。本稿では、課題解決を行うための知識創造を対象としたナレッジ・マネジメントを提案する。

### 2. 知識創造のモデル

本研究ではナレッジ・マネジメントを、ある組織の個人が持っている経験や知識を組織内で収集・蓄積し、共有することにより有効に活用する手法や技術であるとみなし、プログラミング演習にナレッジ・マネジメントの理論[1][2]を応用する。ナレッジ・マネジメントでは「知識」を「形式知」と「暗黙知」の2面から捉える。形式知とは本や文のように形式的に表される知識であり、暗黙知とは考え方や経験における知識などの目に見えない知識である。ナレッジ・マネジメントでは暗黙知を形式知に、形式知を暗黙知に変換するという知識の循環に重点を置く。本研究では図1に示す知識創造モデルとして、SECI (Socialization Externalization Combination Internalization) モデル[1]を適用する。SECIモデルでは4つの知識変換モードを通じた循環プロセスとして捉える。次に4つの知識変換モードの内容を示す。

- (1) 共同化：他人の持つ暗黙知を共同の体験により共有する
- (2) 表出化：対話（共同思考）により暗黙知を明確なコンセプトで表現する
- (3) 連結化：異なった形式知を組み合わせる新たな形式知を創り出す
- (4) 内面化：新たに創造された形式知を個人個人の暗黙知として定着させる

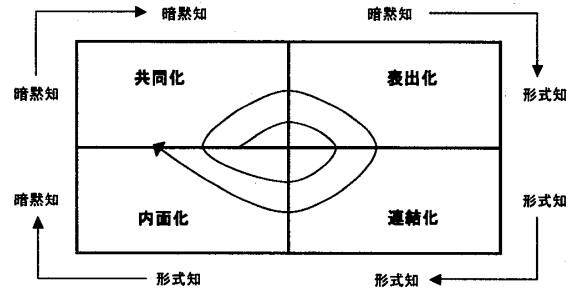


図1 知識変換の4つのモード (SECIモデル)

### 3. ナレッジ・マネジメント

#### 3.1 プログラミング演習

岩手県立大学ソフトウェア情報学部では研究室単位で、1年次後期のソフトウェア演習B、2年次前期のソフトウェア演習Cという授業においてC言語の演習を行っている。ソフトウェア演習Bで文法の基礎を、ソフトウェア演習Cでアルゴリズムとデータ構造を中心に学習する。どちらの授業においても、演習は共通のテキストの内容に沿って進められる。学生には毎回の演習後に学習した内容についてのレポート課題が出題される。レポート課題には、必須課題と発展的な内容の自由課題がある。必須課題では基礎から応用にわたる3~4題の問題が出される。本研究では、難しい発展的な課題の解決に伴う新しい知識の創造のみを対象としているのではない。課題の難易度に関わらず自己の知識を組み合わせ適用し課題を解決する過程での知識創造を対象とする。このため、各回の演習における知識や提出したレポートに存在する知識を随時蓄積していき、知識創造を支援する。個々の学生単位で知識を蓄積するのではなく、学生全員で多様な知識を蓄積する。したがって、学生は前回までの演習の知識を次回の課題解決に活かしながら演習を進めていくことができる。

本研究では課題解決を対象として、SECIモデルをプログラミング演習へ適用する。次にプログラミング演習における4つの知識変換モードの内容を示す。

- (1') 共同化：演習を共同で体験し他人のスキル（これまで学習した知識（暗黙知））を獲得する
- (2') 表出化：対話を通して課題をプログラミングのコンセプトに置き換える
- (3') 連結化：これまでに獲得した知識を組み合わせる処理の流れを組み立てる知識（形式知）を創り出す
- (4') 内面化：新たに創造された形式知を個人の暗黙知として定着させる

#### 3.2 研究室の組織

研究室には、数名の教員、各学年10名程度の学生と大学院学生が所属している。学生には、同学年の学生との横のつながりと上級生との縦のつながりがある。1~3年生の

学生が学生研究室に、4年生と大学院生が講座研究室に配属されている。学生には1人1台のワークステーションが用意されている。学生は研究室に24時間自由に入出りできる。プログラミング演習は、教員および1名のティーチングアシスタント(以下TA)が担当する。学生がレポート課題に取り組む際には、研究室という場で教員、TAに加えて同学年の学生と上級生にも教わることができる。

ナレッジ・マネジメントを実施するには、知識創造が行える場を用意することだけではなく、学生の積極的な参加が重要な要素となる。研究室という場は学生同士のコミュニティを形成しやすいという利点がある。これを活かし、ナレッジ・マネジメントの意義、目的を研究室の全員に理解させることが重要である。知識創造を行う組織は、下位から、知識を体得するナレッジ・プラクティショナー(KP)、知識変換を促進するナレッジ・エンジニア(KE)、組織の方向を決定するナレッジ・オフィサー(KO)から構成される[1]。研究室においては、プログラミング演習の履修学生がKP、上級生およびTAがKE、教員がKOとなる。重要な役割をするのはKEであり、プログラミング演習での過去の体験を通してKPの知識創造を促進させられるかがナレッジ・マネジメントの実施において重要なポイントとなる。

#### 4. 支援環境

本研究では、知識の蓄積、発見、創造を行うことのできる環境を実現する。本研究の支援環境の特徴としては、知識入力負担の軽減、プログラムの動作イメージから関連する知識の発見、アイデアの発想法や思考法を取り入れた知識創造、についての支援を行う点にある。創意工夫を凝らしたソフトウェアを開発できる人材の育成には、適切な論理的思考に基づいたプログラミングスタイルを経験的に身に付け、それを様々な課題解決の局面に活かせる能力が必要であると考えられるからである。この支援環境は、ナレッジ・マネジメントを遂行できる組織での利用を前提としている。以下にそれぞれの点について述べる。

##### (1) 知識の蓄積

テキストで学ぶ基礎となる知識は、予め環境に蓄積しておく。これらの知識は宣言的な知識と手続き的な知識が含まれる。学生の論理的な思考を支援するため、プログラミング演習で前提とする知識まで細分化した単位として蓄積する。これにより、1つ1つの知識に対して前提と結論の関係を明確かつ詳細に定義できるようになる。課題解決の結果得られる知識は、これらの知識を組み合わせるメタな知識であり、新たな関係により基礎となる知識と結び付けられる。演習における演繹的な学習を通して得られた知識を蓄積していくことを目的とする。このような知識の入力は入力者にとって大きな負担となる。入力負担を軽減するために、基本的にはレポートとして提出したファイル(プログラムと文章のファイル)から知識の抽出を行う。予めレポートの体裁を指定しておき、課題解決を行った知識を抽出できるようにする。教員がレポートを評価し優れているものを蓄積する。その他の蓄積する知識としては、学生が他の文献などを調べて獲得したテキストにはない知識、学生同士でやり取りされる疑問点と回答などがあり、これらは学生が自ら蓄積する。

##### (2) 知識の発見

プログラミング演習における課題解決では、出題された課題を解釈、分析し、プログラムの処理の流れに翻訳する必要がある。この翻訳の過程で解決の糸口の1つになるのが、これまでに獲得したプログラムの処理の流れに関するイメージである[3]。本研究では、イメージの形成が不完全な学生のために、プログラムの処理の流れに関するイメージと知識を結び付け、知識を発見できるように支援を行う。具体的には、学生の組み立てた処理の流れのイメージから類似したイメージである知識を検索でき、また、関連した知識を検索することができる。ナレッジ・マネジメントにおいて重要なことは、ただ単に知識を発見することだけではなく、知識を登録した人を知り、コミュニケーションを交わすことにより形式知に加えて暗黙知を引き出すことができることである。本研究では、イメージを介して意思疎通をはかることのできる支援も行う。

##### (3) 知識の創造

一般的に、詰め込み型の学習を行ってきた学生による知識創造活動では、実現性を考慮していない未熟なアイデアを出したり、アイデアを出すことができなかつたりすることがある。これを解決するためには、学生同士で発想法や思考法を体験させることが有効である。本研究では、学生同士で発想法や思考法によりアイデアを出し、それをプログラムで実現できるように支援を行う。その方法としては、

(1) 発想法や思考法を体験させる前にこれまで獲得した知識の全体像を把握させ、自己の能力の範囲で行うことを理解させる、(2) 課題解決のアイデアを発想法や思考法により出させる、(3) 課題解決のアイデアをプログラムに結び付けるアイデアを発想法や思考法により出させる、といった3段階のステップを実施する。このような発想法や思考法は、演習の最終回などに行う総合的な演習で体験させると最も効果的に行うことができる。

#### 5. むすび

本研究では、ナレッジ・マネジメントの理論[1][2]を応用し、プログラミング演習における学生の知識創造を支援することにより、創意工夫を凝らしたソフトウェアを開発できる人材を育成することを目的としている。本稿では、課題解決を行うための知識創造を対象としたナレッジ・マネジメントの提案について述べた。ナレッジ・マネジメントを実践する組織作りなど取り組むべき課題はあるが、知識を蓄積し課題解決に活かすことができる環境を実現することは、学生にとっては大きな利点となる。

今後の課題としては(1) ナレッジ・マネジメントを支援するシステムの開発を進める、(2) KP, KE, KOの役割を明確にしプログラミング演習においてナレッジ・マネジメントを遂行する手法を定める、(3) ナレッジ・マネジメントの評価方法の研究を行うことである。

#### 参考文献

- [1] 野中郁次郎, 竹内弘高: 知識創造企業, 東洋経済新報社(1996)
- [2] Harvard Business Review: ナレッジ・マネジメント, ダイヤモンド社(2000)
- [3] 長谷川聡, 山住富也, 小池慎一: プログラミング教育における制御構造のイメージと理解度について, 情報処理学会論文誌, Vol.39, No.4, pp.1180-1183 (1998)