

学習を中心としたイノベーション 採用プロセスモデル

浜口 幸弘

1 序

組織のイノベーション採用に関しては、これまで多くの研究が行われてきた。しかし、現場での生産工程における新技術導入に関する研究が多く、知識集約的な開発工程への新技術導入に関する研究は少ない。特に、知識集約的な開発の典型とも言えるソフトウェア開発においては、イノベーション採用に対する影響要因や組織の適応化という点で他の産業分野とは異なる特徴を持つと予想される。こうした背景を前提に、本研究ではソフトウェア開発におけるイノベーション採用を分析するために、2回の質問票調査を行ってきた。まず浜口(1999; 2000)では、ソフトウェア開発企業におけるソフトウェアの体系的再利用をとりあげ、従来の研究で指摘されているイノベーション採用に対する影響要因を検証するとともに、新たな影響要因も抽出した。同時に、再利用コンポーネントのタイプおよび運用面に伴う不確かさが採用レベルに関係することを見いだした。さらに浜口(2001)では、ソフトウェア開発企業におけるCASEツールおよびオブジェクト指向の採用を対象に、各イノベーションの採用レベルを形式的採用レベルと実質的運用レベルに区別し、従来からの有力要因および1回目の分析結果に基づき詳

細化した学習変数の観点から、採用レベルへの影響力を分析した。このとき1回目と同様、運用面に伴う不確かさの程度に応じて異なる結果が得られることを確認した。そこで本稿の目的は、このようなイノベーション採用に関する実証結果(浜口1999; 2000; 2001)と既存のイノベーション採用理論との融合をはかり、Cohen and Levinthal(1990)の提案する知識吸収能力を拡張して、学習概念を中心としたソフトウェア開発におけるイノベーション採用プロセスモデルを提案することにある。

2 組織の知識吸収能力

Cohen and Levinthal(1990)が提案した組織の吸収能力(absorptive capacity)という考え方によれば、組織が外部から新しい知識を吸収するに当たり、吸収能力が重要因子として介在し、吸収能力のレベルが高くなるほど、積極的に新しい知識を取り入れる傾向が強くなる。そして、この吸収能力を決定する因子が既存の関連知識(prior related knowledge)である。したがって、外部知識を評価し、利用できる能力は、およそ既存の関連知識のレベルを変数とする関数と見なせる。この考え方を踏襲し、Fichman and Kemerer(1997b)は質問票調査に基づき、新技

術の採用すなわちイノベーション採用における既存の関連知識の影響力を検証している。浜口（1999；2001）でも、この考え方を基礎にイノベーション採用に関する調査研究を行っているが、ソフトウェア工学に基づく開発習慣や指針、日常学習と採用時学習を取り入れている点で、Cohen and Levinthal（1990）の定義する吸収能力という概念を拡張している。

ところが、Van den Bosh et al.（1999）は、既存の関連知識に加え、組織の形態および組み合わせ能力（combinative capabilities：知識の統合力という意味）も吸収能力を決定する要因として提案している。そこで次章以降の議論を精緻にするためにも、Van den Bosh et al.（1999）の考え方と本研究の考え方を対比し、共通点と相違点を明確にしておく。

Van den Bosh et al.（1999）は、企業の知識吸収は外部の新しい知識の評価、獲得、統合からなると考え、知識吸収能力を分析するための3つの次元、すなわち効率、範囲、柔軟性を与えてい

る。効率は企業がコンポーネント知識（製品やサービス関連、生産プロセス関連、市場関連の知識）を獲得し、利用する際に発生するコストの点から評価される。範囲は企業が利用するコンポーネント知識の幅である。多様なプロジェクトを擁し、それらに有用な資源を割り当てられるほど高くなる。柔軟性は企業が追加するコンポーネント知識を入手して、現行のコンポーネント知識を再構成できる程度で評価される。事業運営の意思決定に対する自律性が高く（他の組織とは独立して知識の再構成ができる）、外部環境と直接接触する機会が多いほど高くなる（外部環境との相互やりとりを通じて、知識の再構成ができる）。そしてVan den Bosh et al.（1999）は、「知識吸収の範囲と柔軟性は吸収能力を高め、効率は吸収能力を低めるように働く」という仮定を前提にし（仮定の根拠は与えていない）、3つの次元から見た組織形態と組み合わせ能力の性質を考察している。その要点を表1に示す。

さらに、これらの概念を適用する状況として、

表1 3つの次元から見た組織形態と組み合わせ能力

<ul style="list-style-type: none"> ・組織形態 <ul style="list-style-type: none"> 職能別組織 <ul style="list-style-type: none"> 職能別に部門化されており、規模やスキルの経済性を享受 効率：高 範囲：低 柔軟性：低 ⇒ 知識吸収能力：低 事業部制組織 <ul style="list-style-type: none"> 部門の自律性が高く、部門レベルでコンポーネント知識を再構成 効率：低 範囲：低 柔軟性：高 ⇒ 知識吸収能力：中 マトリックス組織 <ul style="list-style-type: none"> 自律性を備えており、多様なプロジェクトを通して幅広い知識を吸収 効率：低 範囲：高 柔軟性：高 ⇒ 知識吸収能力：高 ・組み合わせ能力 <ul style="list-style-type: none"> システム能力 <ul style="list-style-type: none"> 公式の知識交換手段（規則、手順、情報システムなど）によって知識を吸収 効率：高 範囲：低 柔軟性：低 ⇒ 知識吸収能力：低 調整能力 <ul style="list-style-type: none"> グループのメンバー間の関係（訓練とローテーション、横の連絡機構、意思決定への参加層拡大）によって知識を吸収 効率：低 範囲：高 柔軟性：高 ⇒ 知識吸収能力：高 共同化能力 <ul style="list-style-type: none"> 組織に信念や価値観を注入することで構成員を一体化し、知識を効率的に吸収 効率：高 範囲：低 柔軟性：低 ⇒ 知識吸収能力：低

安定した知識環境（例えば、成熟した単一産業）と不安定な知識環境を与えている。前者では、組織が保有する知識を精緻化、拡張することに重点が置かれ、高い効率、狭い範囲と低い柔軟性が要求される。後者では、組織外知識の探求に重点が置かれ、低い効率、広い範囲と高い柔軟性が要求される。本稿で考察するソフトウェア開発におけるイノベーション採用では、組織外部でしばしばイノベーションが生じ、それを採用するプロセスを想定しているので、後者の知識環境に相当すると見なし、この前提のもとに組織形態と組み合わせ能力について検討する。

表1によれば、組織形態によって知識吸収能力の程度が異なるが、実際に、組織形態と知識吸収能力（またはイノベーション採用）の関係について調査した従来研究は見られない。そこで意思決定の階層レベルだけに注目すると、意思決定の集権化がイノベーション採用に及ぼす影響について調査した研究が幾つかある（Zmud 1982; Dewar and Dutton 1986）。しかし、イノベーション採用に関して意思決定が集権化されている組織と分権化されている組織の間で有意差は検証されていない。また表1が示すように、職能別組織は範囲と柔軟性が劣るため吸収能力が最も低くなっているが、その概念の定義から判断すると、調整能力を充実させることで、ある程度補うことは可能であろう。以上の点を総合的に評価し、本稿のモデル化では組織形態を除外する⁽¹⁾。

次に、表1の組み合わせ能力を見ると、吸収能力にとって調整能力が最も重要な要因であり、システム能力や共同化能力⁽²⁾の強化はマイナス要因になっている。この調整能力の有効性は浜口（1999；2000；2001）でも検証されている。Van den Bosh et al.（1999）はシステム能力が吸収能力にマイナスの影響を与えると仮定しているが、

自ら調査した2社の事例では、システム能力は調整能力を補完する役割を果たし、吸収能力を高める効果をあげている。このような矛盾が生じた理由は次のように説明できよう。まず「組織外から新しい知識を獲得すること」と「組織内で得られた知識を普及させること」を区別する必要がある。前者は、組織外部の新しい知識を評価して、吸収することであり、後者は、組織内で獲得された知識を一括管理して、関係者に普及させることである。Van den Bosh et al.（1999）は、前者の働きのみを吸収能力として考慮しているが、後者の働きも重要である（Fichman and Kemerer 1997b）。後者は直接吸収能力に寄与しないが、将来の採用技術に関連し得る知識を広めることになるので、間接的に吸収能力を高める可能性を持つからである。そして後者で扱われる知識は相対的に洗練化され、曖昧さが小さいので、システム能力で処理するのに適している。実際、Cohen and Levinthal（1990）によれば、組織内重視の（inward-looking）吸収能力と組織外重視の（outward-looking）吸収能力の両方が要求されると指摘しているが、システム能力は組織内重視の吸収能力に含まれよう。また浜口（1999；2000；2001）でもシステム能力の有効性は検証されている。したがって吸収能力の要素として、調整能力とシステム能力が重要であり、本稿では両方を考慮する。もっとも、新製品開発のような創造的領域ではシステム能力の相対的ウェイトは低くなるかもしれないが。

以上をまとめると、Van den Bosh et al.（1999）による吸収能力は組織形態を取り入れ、本稿の基礎をなす吸収能力は組織形態を取り入れない点に本質的相違がある。

3 学習概念の分析と結果

一般に組織学習という概念は広く扱われているが、本研究では、イノベーションの採用に関わるソフトウェア開発担当者を対象にした、組織的に実施されている技術知識の獲得、管理、普及および組織の適応化というプロセスに焦点を置く。この前提に基づき、本章ではイノベーション採用における学習概念の相対的重要性を分析する。

組織の予め保有するイノベーション関連知識が、イノベーションの採用に影響を及ぼすことは既に検証されている（Fichman and Kemerer 1997b; 浜口 1999）。ここでは関連知識以外の学習概念、すなわち組織による知識の獲得、管理、および普及が、イノベーション採用に対し、組織の開発特性を表す諸変数の中で相対的に重要なウェイトを持つことを検証する。そこで、これらの学習変数がイノベーション採用、すなわちCASEツールおよびオブジェクト指向の採用に及ぼす相対的影響度を評価する。このため、従属変数をイノベーションの形式的採用レベルおよび実質的運用レベルとし（以下で、イノベーション採用レベルと記した場合は、形式的採用レベルと実質的運用レベルの両方を指す）、独立変数の候補を考慮すべきすべての変数の場合（フルモデル）、考慮すべきすべての学習変数のみの場合（学習モデル）に分けて、それぞれ回帰分析を行う。そして独立変数による説明力（自由度調整済 R^2 ；以下、 R^2 と略す）の最大値を評価基準として比較する。各変数の定義は浜口（2001）に従うものとする。

さらに、イノベーション採用に伴う組織学習の一つである適応化（Leonard-Barton 1988；Attewell 1992）についても考察する。ソフトウェア開発に関する組織の適応化を調査した研究は少

ないが、その中で、Orlikowski（1993）は同じ種類のCASEツールを採用する特定2社を対比することにより、開発工程に起こる種々の適応は、CASE技術だけでなく、制度的（環境、組織、開発部門）背景、主要プレイヤーの意図と行動の複合的相互作用によってもたらされると論じている。これら要因の影響力の相対的強さについては触れていないが、性質の異なる要素が多数あるため、変数間の関係を導くのは難しいと考えられる。そこで本研究では扱う対象を絞り、開発工程の適応^④が行われる項目を次のように決定する。CASEツールとオブジェクト指向において、おおよそ共通する期待効果は現行の開発工程を効率化して、開発期間、品質、生産性を改善することにある。こうした効果をもたらされることで変更の可能性が生じる開発方法は、「プロジェクトの日程計画および資源配分の方法」、「各工程で生成される中間生産物の評価方法」、「プロトタイプングの実施方法」である。この前提に基づき浜口（2001）では、イノベーション採用レベルと開発方法の適応化の関係を分析し、イノベーションの採用レベルが高い組織ほど、開発方法を変更する傾向が強くなることを検証した。ここでは、最初の分析でイノベーション採用に有力な影響を及ぼすことが実証される学習変数と開発方法の変更の関係を求め、以前の検証結果と比較する。

以上の分析を行うために、浜口（2001）で収集したサンプルを用いる。詳細は浜口（2001）によるが、その概略を以下に述べておく。調査対象となる企業は、1998年度のソフトウェア開発による売上高（受注ソフトウェア開発による売上高とパッケージ開発による売上高の合計^④）が2億円以上で、かつ開発技術者の数（自社のシステムエンジニアとプログラマの合計人数）が30人以上のソフトウェア開発企業である。対象とするソフ

トウェア開発に関しては、「少なくとも主要部分は自社開発するソフトウェアの中で、売上高の最も高い開発分野に属すソフトウェア全体」を主力ソフト系列と呼んで、主力ソフト系列の代表的な開発について回答を求める。そして、2000年10月に220社の企業に質問票を送付した。依頼した回答者は、主力ソフト系列を開発する組織の管理者である。その結果、126社の有効回答が得られた。

質問票調査の結果から、主力ソフト系列の代表的な開発において、従来型開発（オブジェクト指向以前の手続き型開発）を中心とする企業は124社、オブジェクト指向開発を中心とする企業は2社であった。主力ソフトは120社が受託開発で、6社がパッケージ開発である。そして自社で担当する開発工程の範囲は、約90%の企業が全工程、約8%が分析・設計工程、残りがコーディング・テスト

となっている。

この有効回答126社のうち、CASEツールの採用について重要度は低いと回答した企業は20社であり、オブジェクト指向の採用について重要度は低いと回答した企業は6社であった。したがって、形式的な採用レベルの分析対象はCASEツールが106社、オブジェクト指向が120社となる。さらに、CASEツールを利用している（試験的利用も含む）企業67社がCASEの実質的運用レベルの分析対象となる。一方、オブジェクト指向を採用している（試験的採用も含む）企業92社が、管理・運用体制の確立、分析・設計工程の標準化、再利用の制度化の分析対象となり、このうち制度的再利用を実施している（試験的实施も含む）企業61社が部品の品質基準の確立に関する分析対象となる。なお、他のオブジェクト指向言語を採用せず、Visual Basicのみ採用し、これ

表2 CASEツール採用の説明モデル

説明変数	採用種類		管理運用		再利用	
	β		β		β	
主力ソフト技術者の数	.332**					
最新技術採用時期	.198*					
中間物の測定基準			.182*			
モジュール化推進	.107				.207*	
同領域技術交流会	.188*	.242**				
パイロットプロジェクト	.168*	.238**	.232*	.263**		
社内知識管理ユニット	.133					
技術支援ユニット			.387**	.429**	.374**	.435**
知識伝達手段の公式化					.258**	.260**
F値	11.06**	8.70**	9.92**	13.11**	10.32**	13.14**
adjusted-R ²	.326	.182	.288	.268	.304	.275

注 各列左側：フルモデル 各列右側：学習モデル
 採用種類：採用しているCASEツールの対象タスクの規模（106社）
 管理運用：CASEツールの管理・運用体制の確立（67社）
 再利用：CASEツールを利用した再利用の制度化（67社）
 採用規模：オブジェクト指向の組織的採用規模（120社）
 管理運用：オブジェクト指向の管理・運用体制の確立（92社）
 分析設計：オブジェクト指向開発の分析・設計工程の標準化（92社）
 再利用：オブジェクト指向における再利用の制度化（92社）
 品質基準：オブジェクト指向再利用部品の品質基準確立（61社）
 *：p<0.05 **：p<0.01 β ：標準化偏回帰係数（片側検定）

表3 オブジェクト指向採用の説明モデル

説明変数	採用規模		管理運用		分析設計	
	β		β		β	
主力ソフト技術者の数	.215**					
分散システム開発	.158*					
C言語採用	.245**					
モジュール化推進			.178*			
開発方法の標準化					.277**	
チーム情報共有	.164*					
同領域技術交流会	.128	.148*	.171*	.191*		
パイロットプロジェクト			.317**	.358**	.236*	.366**
社内知識管理ユニット	.213**				.238**	.260**
技術支援ユニット					.124	.127
F値	7.70**	5.03**	8.88**	11.35**	14.58**	15.42**
adjusted-R ²	.221	.064	.206	.185	.374	.322

説明変数	再利用		品質基準	
	β		β	
計画の標準化			.299**	
モジュール化推進			.221*	
開発方法の標準化	.285**			
技術導入関与（部門長）			.174	
同領域技術交流会			.290**	.329**
パイロットプロジェクト			.165	
社内知識管理ユニット	.122		.147	
技術支援ユニット	.337**	.369**		
知識伝達手段の公式化	.153*	.184*		
F値	12.40**	9.11**	9.79**	5.93**
adjusted-R ²	.278	.215	.370	.198

を組織的にオブジェクト指向と認めている企業10社も上記92社の中にも含めるものとする。

表2、表3および表4が分析結果である。表2と表3からわかるように、イノベーション採用の説明変数として有力な学習変数は、「パイロットプロジェクトの設置」、「同領域技術者間のコミュニケーション推進度」、「社内の技術知識を管理し、普及させるユニットの設置」、「新技術の教育、推進を担う技術支援ユニットの設置」、「技術知識を伝達する手段の公式化」である。これら学習変数の組み合わせから得られる相対的説明力（フルモ

デルのR²に対する学習モデルのR²）を見ると、オブジェクト指向の採用規模を除いて大きいことがわかる。したがって、これらの学習変数は影響要因として相対的に重要な位置を占めると考えられる。

CASE ツールよりオブジェクト指向の方が新しい技術なので、オブジェクト指向に関して組織はまだ適応途上にあると考えることもできる。しかし、表4に示すように対象サンプル全体に対する適応化企業（指定した3つの開発方法のうち1つでも変更した企業）の割合を比較すると、

表4 開発方法の変更

CASE 開発方法	日程計画と 資源配分	工程中の中間 生産物の測定	プロトタイプ イピング	(社)
変更なし	44	55	60	
変更あり	23	12	7	
オブジェクト 開発方法	日程計画と 資源配分	工程中の中間 生産物の測定	プロトタイプ イピング	
変更なし	73	82	64	
変更あり	19	10	28	

CASE

オブジェクト指向

開発方法	採用 種類	管理 運用	再利用	採用 規模	管理 運用	分析 設計	再利用	品質 基準
変更なし	28 4.61	28 2.11	28 2.93	40 4.47	40 2.28	40 2.13	40 2.10	22 1.91
1つでも 変更あり	39 5.51**	39 2.59*	39 3.87*	52 4.88*	52 2.48	52 2.48	52 2.40	39 2.51*

注 各上段は企業数，下段は平均スコア

CASE

開発方法	パイロット プロジェクト	社内知識管 理ユニット	同領域の 技術交流会	技術支援 ユニット	知識伝達手 段の公式化
変更なし	28 2.79 ^a	28 1.68	28 3.21	28 .14	27 .81
1つでも 変更あり	39 2.38	39 1.79	39 3.16	39 .33 ^a	39 .82

オブジェクト指向

開発方法	パイロット プロジェクト	社内知識管 理ユニット	同領域の 技術交流会	技術支援 ユニット	知識伝達手 段の公式化
変更なし	40 2.23	40 1.55	40 2.93	40 .25	40 .78
1つでも 変更あり	52 2.42	52 1.77	52 3.15 ^a	52 .40 ^a	50 .78

注 a: p<0.10 各上段は企業数，下段は平均スコア

CASE ツールが 58.2%，オブジェクト指向が 57.8% となり，同じ位の割合を示している（有意水準を 10% にしても差はない）。したがって，両者を同じように扱うことの問題は小さいと考え，議論する。表 4 によれば，抽出された 5 つの学習変数と適応化変数の間には，イノベーション採用レベ

ルと適応化変数の間ほど有意な関係は見られない。そこで有意水準を緩めて 10% で考えると，「技術支援ユニットの設置」が両イノベーションの適応化変数に対して有意な差を示している（全体に明確な有意差が見られないのは，後述するように，適応化の背景が複雑なことによると推測される）。

よって技術支援ユニットの設置が開発方法の変更にある程度の役割を果たしていると言える。また表に記していないが、ソフトウェア工学の基本原理解を表す変数「プロジェクトの日程計画および資源配分方法の標準化度」、「要求分析から設計に至る開発方法の標準化度」、「ソフトウェアのモジュール化推進の程度」、「各工程で生成される中間生産物を評価する測定基準の確立」に関しては、イノベーションの採用レベルとは有意な関係を持つが（浜口 2001）、適応化変数とはまったく有意な関係はない。したがって両イノベーションにとって、採用レベルと適応化レベルは少なからず関係があるものの、適応化は採用とは異なる側面を持つと考えられる。この点については次の章で考察する。

4 学習から見たイノベーション採用プロセス

前章からイノベーション採用における学習概念の相対的重要性が示された。本章と次章ではその結果を取り入れつつ、関連知識の保有という学習概念を中心に、浜口（1999；2000；2001）および既存の理論との融合をはかってゆく。既存の理論の参照に当たっては、主としてソフトウェア開発に関する文献を対象にする。というのは、工場のような生産現場への新技術導入とソフトウェア開発への新技術導入では、複雑さ及び不確かさという点で意味が異なるからである。前者は繰り返し作業に基づく生産工程での技術的変化であるが、後者はより創造性を要する知識集約的な開発工程での技術的変化である。それぞれの学習の主体とそのプロセスは明らかに異なる。したがって、イノベーション採用を分析するに当たっては、どのような活動領域（ユーザ）に採用するのか注意する必要がある。

まず1980年以降^⑤に刊行された4ジャーナル Management Science, Organization Science, MIS Quarterly, Research Policy に掲載の論文を対象に、「ソフトウェア開発におけるイノベーション採用」に関するすべての論文を参照したところ、最新の6つのソフトウェア開発方法—トップダウン開発、構造化設計、構造化レビュー、チーフプログラマーチーム、ユニット開発フォルダ、構成管理—(Zmud 1982；1984)、データベース設計ツールとテクニック (Nilakanta and Scamell 1990)、オブジェクト指向技術 (Fichman and Kemerer 1997b) がイノベーションとして取り上げられている。影響力を検証された学習概念をすべて挙げると、既存の関連知識、学習関連の規模、技術支援ユニットの設置である。次に、ソフトウェア開発以外の分野についても吟味しておく。この分野でイノベーション採用を扱った文献において、採用に影響を及ぼす要因を具体的に明示あるいは検証しているものには、Ettlie et al. (1984), Dewar and Dutton (1986), Cooper and Zmud (1990), Swanson (1994) がある。そして影響力の強い学習概念が示される場合、すべて上記に含まれている。

そこで、本研究のこれまでの結果も合わせ、イノベーション採用プロセスにおける開発レベルでの学習の役割と手段をまとめると、次のようになる。ただし、学習関連の規模は基本的に担当要員数を表すので除外し、既存の関連知識は、浜口（1999；2000；2001）の結果に従い広くとらえ、ソフトウェア工学に基づく開発習慣や開発指針も含める。また知識の伝達手段は何らかの形で公式化されるとする。

(1) イノベーション採用以前において、既存の関連知識を普及させる可能性（その時点では、将来のイノベーションと関連するか否かわからな

いが)を増す。

既存の関連知識、公的機会による技術者間の交流、社内知識管理ユニット

- (2) イノベーションの試験的採用時、経験を通して得た知識やノウハウを普及させる。

パイロットプロジェクト(ワーキンググループ)、公的機会による技術者間の交流、社内知識管理ユニット

- (3) イノベーションの正式な採用後、従来の開発形態や開発方法の適応化を促す。

技術支援ユニット

このとき、教育訓練方法という概念は含めないものとする。というのは、(1)と(2)の場合において、イノベーション採用レベルと教育訓練方法との間に一定の関係が見いだされないからである(浜口 1999; 2001)。実際、1998年の調査では、若手技術者および中堅技術者について各教育重点領域(分析設計やプログラミングのテクニック/ソフトウェア工学の理論/開発ツール利用方法の習熟)とイノベーション採用レベルの関係を分析し、2000年の調査では、若手技術者の日常時教育訓練方法(OJT重視~教育訓練プログラム重視)および革新技術採用時の社員教育(採用前の教育~採用後のOJT)とイノベーション採用レベルの関係を分析したが、意味ある結果は得られなかった。

さて、(3)のイノベーション採用に伴う適応化を調査研究している文献は少ないが、その流れを概括し、本稿との関係を考察する。Rice and Rogers (1980)は、採用者はしばしば技術を適応したり、技術を取り巻く手順を適応すると述べている。Leonard-Barton (1988)も、適応化は複雑で繰り返し起こるプロセスであり、組織の適応化と技術の適応化の両方が起こり得る(一方だけの場合もあり得る)と論じているが、Rice and

Rogers (1980)同様、適応化に影響を及ぼす要因を明示してはいない。その後、Orlikowski (1993)は同じ種類のCASEツールを採用する2社を比較することにより、開発工程に起こる適応化プロセスは、CASE技術だけでなく、制度的(環境、組織、開発部門)背景、主要プレイヤー(管理者、開発者)の意図と行動の複合的相互作用によってもたらされると指摘している。しかし、これら要因の影響力の相対的強さは示していない。したがって、特定要因のみ強く、他は無関係の可能性もあり得る。こうした経緯から分かるように、適応化に影響を及ぼす有力要因を見いだすことは、イノベーション採用レベルに影響を及ぼす要因を抽出することに比べて難しいと言える。難しさの理由は、適応化の背景が複雑なことによる。すなわち、適応化する場合、イノベーションの採用レベルが高い組織ほど、それだけ効果または影響も大きくなるので、開発方法を変更する状況があり得るし、イノベーション採用レベルを一層高めるため、それに適した開発方法に変更する状況も考えられる。また適応化しない場合、既にイノベーション技術に適した開発方法を実施しているので、変更の必要がない状況もあるし、不適合があっても、技術的あるいは制度的に現行方法を変更できない状況もあり得る。

ところが、Tyre and Orlikowski (1994)では、新たな成果が得られている。特定技術に対して適応化のプロセスを明らかにしている研究はないが、Tyre and Orlikowski (1994)は、3つの組織(そのうち1つがソフトウェア開発組織)に対し、それぞれ特定技術の採用に関する事例研究を行っている。そしてすべての事例において、主たる適応活動は技術採用直後に集中して起こり、その後の適応活動は急減することを示している。さらに、このような組織の適応活動をもたらす要因を次の

ように4つ指摘している。

- 生産性のパフォーマンスが優先される状況

一旦、新技術が使えるようになると、不確実性を伴う適応化よりも、技術を使った生産活動の方が優先される。

- 新技術を採用した業務のルーティン化

ユーザが新技術の経験を積むにつれ、新技術の利用方法がルーティン化し、以降の適応化が阻まれる。

- 新技術に対する期待の変化

採用前と採用後で、新技術に対する期待が変わる。

- 採用担当者の配置転換による専念度低下

新技術の使用が安定すると、しばしば採用チームの要員は配置転換され、以降の適応化が阻まれる。

1番目は、新技術に伴う不確実性の問題と考えられる。採用する新技術が複雑なほど、組織は適切な対応ができないという主張（Van de Ven 1986）とも共通する。この問題は浜口（1999；2000；2001）でも言及しており、次章で議論する。

2番目は、新技術のユーザはその利用方法のルーティン化を志向する傾向が強いということだが、逆に、仕事の単調化を招く可能性もある。実際、この見方を裏付ける文献はないので、ここでは除外する。3番目については、新技術に対する期待が採用前後でいつも変わるとは限らない。確かに、Tyre and Orlikowskiも有力な要因と見なしていないので除外する。4番目は、新技術の採用に際して、専門の技術支援担当者（グループ）を設けないことから生じる問題である。前章での分析結果から、技術支援ユニットの設置が適応化を促進し得るので、考慮すべき要因と言える。したがって、本研究で提案するイノベーション採用モデルの適応化の部分には、技術採用に伴う不確実性と

技術支援ユニットを取り入れることにする。

ところで、上述の適応化パターンに従えば、組織における技術的適応のプロセスには漸進的かつ連続的なものが存在するという従来からのイノベーション研究の主張（例えば、Ettlie et al. 1984；Leonard-Barton 1988）に合致しない問題が生じてくる。しかし、Tyre and Orlikowski（1994）は、こうした適応化パターンは特定の組織的アプローチや特定技術に依存するのではなく、技術的適応のプロセスに固有の現象と述べており、その一般性を主張している。また、この適応化パターンは、組織の適応化は連続して行われるものではないという点で、行動理論による主張（例えば、Katz 1982；Kiesler and Sproull 1982）とも矛盾しない。さらにFichman and Moses（1999）も、イノベーション採用における漸進的アプローチについて、理論的に有効性は認められているが、現場ではほとんど実践されていないと論じている。これらの点を考慮し、本稿ではTyre and Orlikowskiが指摘する適応化要因をほぼ支持する立場をとる⁶⁾。

5 不確実性を取り入れたイノベーション採用モデルの提案

浜口（1999；2000；2001）を通じて、組織がイノベーションを採用してある形態（特定の実施状況）を実現するとき、以下のようにイノベーションの性質に応じて、 R^2 の大きさに関して特有の結果が得られた。

(1) 実現が比較的容易な形態 ((2)に比べ、 R^2 は相対的に小さい)

イノベーションが扱いやすい、あるいは形式的な採用形態のとき、大抵の企業は実現可能である。

例：オブジェクト指向の形式的な採用規模
(2) 実現にある程度の困難を伴うが、不確実さは小さい形態

実現には難しい面もあるが、取り組む範囲は絞られ、努力や目標は特定される。

例：採用する CASE ツールの種類

オブジェクト指向での再利用の実施や品質保証

(3) 実現に困難を伴い、不確実さが大きい形態
(2)に比べ、 R^2 は相対的に小さい)

採用に関わる工程範囲や活動範囲が広くなる、または効果の予測が難しくなると、大きな不確実性を伴う。

例：再利用部品の作成および適応化のため

の評価基準の設定

設計、要求仕様部品の再利用

CASE ツールの管理運用体制の確立

オブジェクト指向の管理運用体制の確立

ここでの議論は、(3)のように不確実性が説明力に影響を及ぼす(弱める)点である。まず(3)の事例に含まれる不確実性の特徴を考察する。

ソフトウェアの体系的再利用を行う場合、再利用部品を蓄積して開発に利用できるリポジトリを作成するには長期間を要し、かつ部品の利用回数を予測するには十分な経験を要するため、再利用部品を作成したり、適応化するためのコストベネフィットを短期的に評価することは困難である(Gaffney and Durek 1989; Koltun and Hudson 1991; Frakes and Terry 1996)。また再利用の対象となる部品がコード、設計、要求仕様となるにつれ、再利用プロセスを一層広い工程範囲に組み込む必要が生じ、不確実性は増すことになる(Coulange 1998)。CASE ツールの中で最も複雑な統合化 CASE ツールの場合、Kemerer (1992)

によれば、統合化 CASE の機能を十分に使いこなして達成されるパフォーマンスが、従来技術によるパフォーマンスを上回るには一定期間(6か月から1年位)が必要とされる。従来の学習曲線モデル(手作業で繰り返し仕事を行う労働者のパフォーマンスを予測するために考案された)が、統合化 CASE の学習曲線モデル(繰り返しのでない仕事を行う知的労働者のパフォーマンスを予測)に適用できない点に、CASE を評価する難しさがある。またオブジェクト指向開発の場合、方法論自体が新しく、管理運用の対象は要求分析からテストに至る全工程に及ぶため、各工程自体の不確実性と工程間を調整する際の不確実性が相乗して大きくなる。

以上の例を考慮すると、組織がイノベーションを採用して、ある形態を実現する時に伴う不確実性は、試行錯誤しながら全体の統合や調整を行う際に直面する不確実性とイノベーションに対する投資効果の不確実性からなると考えられる。前者は、スキルやノウハウの獲得、蓄積により減じられる可能性をもつ。一方後者は、イノベーション採用後、どれ位の期間が経過すれば投資を上回る利益が得られるかに関する不確実性である。組織の技術環境が整うまで長期間を要したり、複雑高度な技術ほど訓練や使用による学習効果が現れる時期を見通せないことから生まれるものである。その不確実性は、知識やノウハウの獲得によって容易に解消される訳ではない。

次に、従来研究を見ると、不確実性は曖昧な概念であり、イノベーション採用の中で具体的概念として扱っている文献は極めて少ない(Tyre and Hauptman 1992)。その中で、Tushman and Anderson (1986)の取り上げている不確実性は、技術提供側の改良に対する需要の予測誤差として測定され、組織が技術を採用するときに直

面するものではない。一方、Tyre and Hauptman (1992) は、組織の技術的变化に伴う不確実性を、専門技術上の複雑さ (technical complexity) と組織全体のシフト (systemic shift) という2つの視点からとらえ、この不確実性と新技術への対応方法が生産現場における技術的变化に及ぼす影響力を分析している⁽⁷⁾。前者の不確実性は、導入設備の相対的新しさ、あるいは相対的精緻さを意味する。具体的には、質問項目から判断して、新設備の基礎をなす技術の不確実度（組織内外を通じての未知性）と見なせよう。後者の不確実性は、新設備が組織の確立した概念や生産システムへ与える影響度を意味する。具体的には、質問項目から判断して、組織が新設備に対して予め保有する関連知識（または経験に基づくノウハウ）の程度と見なせる。後者は、Cohen and Levinthal (1990) による「イノベーション関連知識の保有」という概念と一致している。本稿では、この概念を不確実性とは考えずに、組織の吸収能力を決定する中心概念として扱う立場をとる。したがって、Tyre and Hauptman (1992) の不確実性は、専門技術上の複雑さに帰着できよう。しかし、この概念では前述の事例で示した不確実性を表すには精密さを欠く。そこで、組織外から情報やノウハウの入手が難しいイノベーション（例えば、組織内外を通じて新しい技術、既知の技術であっても組織独自の利用方法が要求される技術）を対象に、前述の例に基づき、Tyre and Hauptman (1992) による専門技術上の複雑さを次のように改め、組織がイノベーションを採用して、ある形態を実現する時に伴う2つの不確実性を定義する（2つ合わせて、実現の不確実性と呼ぶ）。その測定に際して、いずれの変数も技術に依存するので具体化は難しいが、例を示しておく。

・統合（調整）の不確実性

イノベーションの適用範囲の広さから生じる調整の必要性、あるいは組織内の既存の技術、方法または体制との統合を行う必要性の程度によって表される。例えば、管理運用を行うとき、採用に関わる工程範囲が広くなるほど、不確実性は相乗的に増加する。

測定変数の例：採用工程の範囲、既存技術との関係性（独立性）

・効果の不確実性

投資に対する学習効果が現れるのに要する期間、または効果の評価基準を確立できるまでの経験蓄積量で表される。例えば、再利用部品のコスト評価基準を設定するとき、部品の利用回数を予測するには十分な経験を要するので不確実性は大きい。

測定変数の例：コスト評価基準を確立する際の必要データ量、技術環境の整備期間

一方、組織外から情報やノウハウの入手が可能なイノベーションに対しては、不確実性よりもそれを入手できる手段の整備が問題になるので、「組織外の技術情報を獲得して普及させるユニットの設置」という変数を設定して対応する。

イノベーションの採用プロセスの中で、組織の適応化（開発方法の適応化）をもたらす背景は単純でないが、本稿での議論に基づき、学習を中心としたイノベーション採用プロセスに関するモデル⁽⁸⁾を図1のように提案する。イノベーション採用レベルに影響を及ぼし得る要因は、実現の不確実性、既存の関連知識、学習関連の規模、知識の獲得／管理／普及および技術支援ユニットの設置であり、開発方法の適応化に影響を及ぼし得る要因は、実現の不確実性、技術支援ユニットの設置およびイノベーション採用レベル（開発方法の適応化との相互作用が考えられる）となる。そして、実現の不確実性が大きくなるにつれ、これら変数

学習を中心としたイノベーション採用プロセスモデル

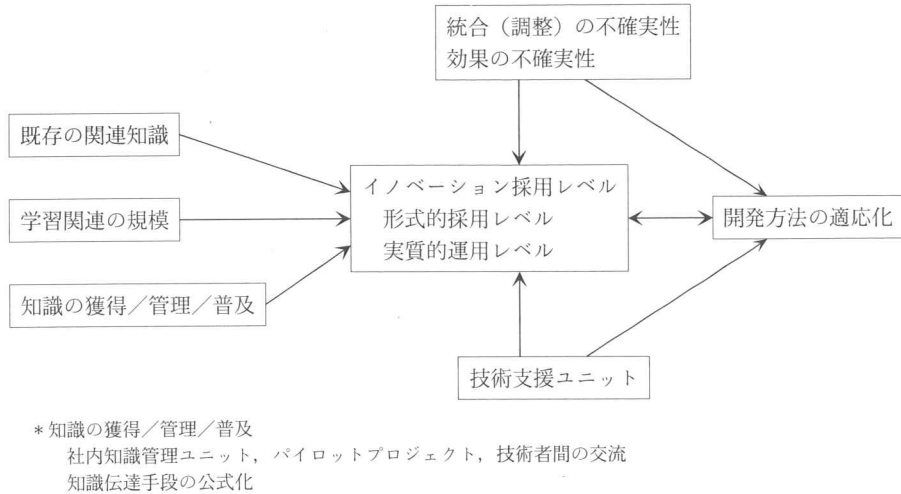


図1 学習を中心としたイノベーション採用プロセスモデル

間の影響関係は弱められる。またパイロットプロジェクトや技術支援ユニットの設置は実現の不確実性、特に、統合（調整）の不確実性を低める働きをもつ。一般に組織にとって、複雑さを伴う新しい考え方や機会に対して構成員の注意を喚起することは難しい（Van de Ven 1986）が、パイロットプロジェクトや技術支援ユニットは、これを容易にする可能性を持っている。以上から、次の2つの命題を導くことができる。

命題A：実現の不確実性が小さい、または低減されるとき、既存の関連知識、学習関連の規模、知識の獲得／管理／普及および技術支援ユニットは、イノベーション採用レベルを高める働きをもつ。また技術支援ユニットは開発方法の適応化を促す。さらに、イノベーション採用レベルが高くなるほど、開発方法の適応化が促される（逆の影響関係も成り立つ）。

命題B：実現の不確実性が大きく、かつ低減されないとき、既存の関連知識、学習関連の規模、知識の獲得／管理／普及および技術支援ユニットは、イノベーション採用レベル

に弱い影響しか及ぼさない。また技術支援ユニットも開発方法の適応化に弱い影響しか及ぼさない。イノベーション採用レベルと開発方法の適応化の相互作用は弱くなる。

ただし、適応化や不確実性に関しては、それを考慮した調査研究が極めて少ないので、これらの命題は修正の余地を残している。

6 提案モデル適用の範囲

最後に、本稿で提案したモデルが適用し得る一般的技術のタイプを示し、これに基づき、本稿で取り上げた3つのソフトウェア技術への適用性を確認する。

イノベーションの採用に関して、Ettlie et al. (1984) や Dewar and Dutton (1986) は、漸進的イノベーションと抜本的イノベーションという2つの分類を与えているが、この分類は曖昧さを持っている。一方、Tushman and Anderson (1986) は、3つの産業に関する技術革新の歴史を実証分析し、プロダクト技術とプロセス技術について技術変化を3つのタイプに分類している。

プロセス技術を対象に3つの技術変化をあげると次のようになる。1つ目は、漸進的な技術能力強化の変化（incremental competence-enhancing change）であり、従来プロセスのパフォーマンスが徐々に改善される変化である。2つ目は、技術能力強化の不連続性（competence-enhancing discontinuities）であり、従来プロセスのパフォーマンスが革新的に改善される、または従来プロセスが取って代わられる変化を意味する。新しいプロセスの構築には既存の知識、スキルまたはノウハウが利用される。3つ目は、技術能力破壊の不連続性（competence-destroying discontinuities）であり、既存の知識やスキルが利用できないような革新的な変化であり、企業の新規参入を招く傾向が強い。このように前者の分類に比べて曖昧さはある程度改善されている。

さて前章で提案したモデルは既存の関連知識を基礎にしているので、技術能力の破壊をもたらすような革新的技術には適用できない。したがって、1つ目と2つ目の技術変化にのみ適用できる。そこで本稿で取り上げた3つのソフトウェア技術に対し、この分類概念を当てはめ、モデルの適用妥当性を確認する。

ソフトウェア開発組織による再利用は、従来技術で対応する形態から最新の再利用向けツールを駆使した形態まで多岐にわたるが、基本的に従来の開発技術の枠組みが基礎になっている（McClure 1992）。再利用によって著しくパフォーマンスが向上した組織がある一方（Banker et al. 1993）、その難しさから効果を上げられない組織も多い（Card and Comer 1994）。したがって多数は漸進的な技術能力強化の変化であり、残り少数が技術能力強化の不連続性に相当する。

CASEは開発工程の一部（または全部）を支援する開発支援型の技術である。Finlay and

Mitchell（1994）は特定企業1社をとりあげ、CASEツールの導入により生産性、納期、品質の点で明らかな改善があったことを報告している。Banker et al.（1993）は2社の事例を示しつつ、CASEツールの導入がソフトウェアの再利用に大きな貢献を果たしたと述べている。また、Orlikowski（1993）が調査した2社においても、概ね肯定的な結果が出ている。一方、Blackburn et al.（1996）の調査によれば、ソフトウェアの開発時間を短縮する点で、CASEツールは日本、米国ともに低い評価しか得られていない。Gillies and Smith（1994）も、CASEツールはコーディング作業の部分的な自動化には役立つが、より高度な工程では期待されるほど使い勝手が良くないと指摘している。さらに、Frakes and Fox（1995）は実証結果を通して、CASEツールはソフトウェアの再利用を促進する上で、必ずしも有効な要因ではないと述べている。このようにパフォーマンスの評価は分かれており、CASEツールの採用は高々、技術能力強化の不連続性と言えよう。

オブジェクト指向プログラミング言語の製品が商用化されたのは、80年代後半からである。CASEツールと比べると、新しい技術と言える。オブジェクト指向ではデータと操作をカプセル化してオブジェクトとして扱い、オブジェクト間のメッセージのやり取りで処理が進行する。オブジェクト指向の定義は、オブジェクト、クラス、メッセージ受け渡し、カプセル化、継承、多相性という機能を備えていることであり（Firesmith and Eykholt 1995）、これらの特徴はソフトウェア工学の目標である抽象化、モジュラリティ、再利用、拡張性などを促進する。Fichman and Kemerer（1992；1993；1997a）は、オブジェクト指向的方法論は構造化分析のような従来型方法論に対して、分析レベルおよび設計レベルともに、抜本的

な変化を遂げていると説明している。したがって従来方法に慣れ親しんだ開発者ほどオブジェクト指向を修得するのに相当の労力を強いられる。このように既存の知識が直接利用できる訳ではないので、表面的には技術能力破壊の不連続性に相当するが、上述のようにソフトウェア工学の目標を達成するための特徴を備えており、開発の基本原則が変わることはない。実際、浜口（2001）の調査結果でも、オブジェクト指向の採用を契機に新規参入した企業は見られない。パフォーマンスの調査例はほとんどないが、オブジェクト指向の採用は技術能力強化の不連続性におさまるであろう。

以上の議論から再利用、CASE ツール、オブジェクト指向の場合、本稿のモデルの適用範囲内にあると見なせる。

7 むすび

組織によるイノベーションの採用プロセスをモデル化するに当たり、留意すべきことは、イノベーションを採用する活動領域（例えば、生産工程か開発工程か）、イノベーションに関する知識と既存技術（または方法）を支える知識との関連性という2つの条件によって、その影響要因が変わり得るということである。本稿では、ソフトウェア開発におけるイノベーション採用について考察し、学習概念を中心とした採用プロセスのモデルを提案した。このモデルがそれ以外の分野に適用可能か否かは、少なくともこれらの条件を考慮する必要がある。イノベーション一般に適用できる汎用モデルの提案は難しいと思われるが、学習の重要性は変わらないだろう。また、採用プロセスモデルの構築で難しいのは、適応化の背景やその要因を示すことにある。しかし適応化の要因や時間的変遷については、調査研究が少ないのが実情であ

る。適応化のプロセスは、前述のように Tyre and Orlikowski（1994）によるパターンが考えられるので、イノベーション採用後の経過期間を幾つかに区切って調査する必要がある。

注

- (1) ソフトウェア開発企業を考えると、プロジェクトチームが開発の中心になることから、事業部制もしくはマトリックス組織の形態をとっている場合がほとんどである（浜口 1996）。
- (2) 共同化能力は吸収能力を高める可能性もあるが、強い企業文化によって構成員を同質化すると、多様な知識を吸収することが困難になり、むしろ吸収能力にマイナスになると考えている。
- (3) Leonard-Barton（1988）が指摘するように、適応化として組織の適応化と技術の適応化が考えられるが、本稿では組織の適応化（特に、開発方法の適応化）のみを対象にしている。Leonard-Barton（1988）の事例にあるように、複雑な技術の場合、適応化は常に可能な訳ではないからである。
- (4) ソフトウェア開発のみの売上高で、その他、情報処理サービス等による売上高は除いている。
- (5) 浜口（2001）で示しているように、イノベーション採用と学習の関係が論じられ始めたのは80年代以降からである。
- (6) Tyre and Orlikowski（1994）は3つの新技術を対象にしているが、技術の性質から適応化に長期間を要する必要性があれば、ほとんどの適応活動が採用直後に行われることには疑問が生じる。著者らはこの問題について言及していないが、不確実性と技術支援ユニットの適応化要因としての重要性については、本研究でも検証されているので問題ないと思う。
- (7) 特定技術を対象にしたものではなく、現行設備の日常的アップグレードから統合化生産システムの採用にいたる多様な技術的プロセスの変化をサンプルとしている。したがって、技術の複雑さをパラメータとして、組織の新技術対応方法（独立変数）による技術的变化（従属変数）の説明を行っているのではない。
- (8) Mendelson（2000）は、組織のIQが組織のパフォーマンスに影響を及ぼすことを実証している。本稿のモデル化でも組織の学習能力という概念として考慮したが、浜口（2001）で示したように、

学習能力はイノベーション採用の有力要因になり得ないので、この概念は含めないものとする。むしろ新技術開発のような創造的な領域で有効な要因になる可能性はあるだろう。

参考文献

- Attewell, P., "Technology Diffusion and Organizational Learning: The Case of Business Computing," *Organization Science*, Vol. 3, No. 1, 1992
- Banker, R. D., R. J. Kauffman and D. Zweig, "Repository Evaluation of Software Reuse," *IEEE Transactions on Software Engineering*, Vol. 19, No. 4, April, 1993, 379-389
- Blackburn, J. D., G. D. Scudder and L. N. Van Wassenhove, "Improving Speed and Productivity of Software Development: A Global Survey of Software Developers," *IEEE Trans. Software Engineering*, Vol. 22, No. 12, 1996, 875-885
- Card, D. and E. Comer, "Why do so many reuse programs fail," *IEEE Software*, Sept., 1994, 114-115
- Cohen, W. M. and D. A. Levinthal, "Absorptive Capacity: A New Perspective on Learning and Innovation," *Administrative Science Quarterly*, 35, March, 1990, 128-152
- Cooper, R. B. and R. W. Zmud, "Information Technology Implementation Research: A Technological Diffusion Approach," *Management Science*, Vol. 36, No. 2, 1990, 123-139
- Coulange, B., *Software Reuse*, Springer, 1998
- Dewar, R. D. and J. E. Dutton, "The Adoption of Radical and Incremental Innovations: An Empirical Analysis," *Management Science*, Vol. 32, No. 11, 1986, 1422-1433
- Ettlie, J. E., W. P. Bridges and R. D. O'Keefe, "Organization Strategy and Structural Differences for Radical versus Incremental Innovation," *Management Science*, Vol. 30, No. 6, 1984, 682-695
- Fichman, R. G. and C. F. Kemerer, "Object-Oriented and Conventional Analysis and Design Methodologies," *IEEE Computer*, October, 1992, 23-39
- Fichman, R. G. and C. F. Kemerer, "Adoption of Software Engineering Process Innovations: The Case of Object Orientation," *Sloan Management Review*, Winter, 1993, 7-22
- Fichman, R. G. and C. F. Kemerer, "Object Technology and Reuse: Lessons from Early Adopters," *IEEE Computer*, October 1997a, 47-59
- Fichman, R. G. and C. F. Kemerer, "The Assimilation of Software Process Innovations: An Organizational Learning Perspective," *Management Science*, Vol. 43, No. 10, 1997b, 1345-1363
- Fichman, R. G. and S. A. Moses, "An Incremental Process for Software Implementation," *Sloan Management Review*, Winter, 1999, 39-52
- Finlay, P. N. and A. C. Mitchell, "Perceptions of the Benefits from the Introduction of CASE: An Empirical Study," *MIS Quarterly*, December, 1994, 353-369
- Firesmith, D. G. and E. M. Eykholt, *Dictionary of Object Technology*, SIGS Books, 1995
- Frakes, W. B. and C. J. Fox, "Sixteen Questions about Software Reuse," *Communications of the ACM*, Vol. 38, No. 6, 1995, 75-87
- Frakes, W. B. and C. Terry, "Software Reuse: Metrics and Models," *ACM Computing Surveys*, Vol. 28, No. 2, June, 1996
- Gaffney, J. E. and T. A. Durek, "Software Reuse-Key to Enhanced Productivity: Some Quantitative Models," *Information on Software Technology*, 31, 5, 1989, 258-267
- Gillies, A. C. and P. Smith, *Managing Software Engineering*, Chapman & Hall, 1994
- Katz, R., "The Effects of Group Longevity on Project Communication and Performance," *Administrative Science Quarterly*, 27, 1982, 81-104
- Kiesler, S. and L. Sproull, "Managerial Response to Changing Environments: Perspectives on Problem Sensing from Social Cognition," *Administrative Science Quarterly*, 27, 1982, 548-570
- Kemerer, C. F., "How the Learning Curve Affects CASE Tool Adoption," *IEEE Software*, May, 1992, 23-28
- Koltun, P. and A. Hudson, "A Reuse Maturity Model," in *the 4th Annual Workshop on Software Reuse*, 1991
- Leonard-Barton, D., "Implementation as Mutual Adaptation of Technology and Organization," *Research Policy*, Vol. 17, 1988, 251-267
- McClure, C. L., *The Three Rs of Software Automation: Re-engineering, Repository, Reusability*, Prentice Hall Inc., 1992
- Mendelson, H., "Organizational Architecture and Success in the Information Technology Indus-

- try," *Management Science*, Vol. 49, No. 4, 2000, 513-529
- Nilakanta, S. and R. W. Scamell, "The Effect of Information Sources and Communication Channels on the Diffusion of Innovation in a Database Development Environment," *Management Science*, Vol. 36, No. 1, 1990, 24-40
- Orlikowski, W. J., "CASE Tools as Organizational Change: Investigating Incremental and Radical Changes in Systems Development," *MIS Quarterly*, September, 1993, 309-340
- Rice, R. and E. Rogers, "Reinvention in the Innovation Process," *Knowledge: Creation, Diffusion, Utilization*, 1, 4, 1980, 499-514
- Swanson, E. B., "Information Systems Innovation among Organizations," *Management Science*, Vol. 40, No. 9, 1994, 1069-1092
- Tushman, M. L. and P. Anderson, "Technological Discontinuities and Organizational Environments," *Administrative Science Quarterly*, 31, 1986, 439-465
- Tyre, M. J. and O. Hauptman, "Effectiveness of Organizational Responses to Technological Change in the Production Process," *Organization Science*, Vol. 3, No. 3, 1992, 301-320
- Tyre, M. J. and W. J. Orlikowski, "Windows of Opportunity: Temporal Patterns of Technological Adaptation in Organizations," *Organization Science*, Vol. 5, No. 1, 1994, 98-118
- Van den Bosh, F. A. J., H. W. Volberda and M. de Boer, "Coevolution of Firm Absorptive Capacity and Knowledge Environment: Organizational Forms and Combinative Capabilities," *Organization Science*, Vol. 10, No. 5, 1999, 551-568
- Van de Ven, A. H., "Central Problems in the Management of Innovation," *Management Science*, Vol. 32, 1986, 590-607
- Zmud, R. W., "Diffusion of Modern Software Practices: Influence of Centralization and Formalization," *Management Science*, Vol. 28, No. 12, 1982, 1421-1431
- Zmud, R. W., "An Examination of Push-Pull Theory Applied to Process Innovation in Knowledge Work," *Management Science*, Vol. 30, No. 6, 1984, 727-738
- 通産省, 情報サービス企業台帳, 1997, 1998, 1999, 2000
- 浜口幸弘, ソフトウェアの開発管理, 明治学院大学経済研究第 104 号, 1996
- 浜口幸弘, ソフトウェア技術知識の再利用, 明治学院大学経済研究第 115 号, 1999
- 浜口幸弘, イノベーションとしてのソフトウェア再利用, 明治学院大学経済研究第 118 号, 2000
- 浜口幸弘, ソフトウェア開発におけるイノベーション, 明治学院大学経済研究第 121 号, 2001

(2001 年 10 月 5 日経済学会受理)