

長期の競争、短期の競争：人材vs.ファイナンス(3)

KOIKE, Kazuo / 小池, 和男

(出版者 / Publisher)

法政大学経営学会

(雑誌名 / Journal or Publication Title)

経営志林 / The Hosei journal of business

(巻 / Volume)

50

(号 / Number)

2

(開始ページ / Start Page)

51

(終了ページ / End Page)

70

(発行年 / Year)

2013-07-31

(URL)

<https://doi.org/10.15002/00013606>

〔研究ノート〕

長期の競争、短期の競争 —人材 vs. ファイナンス (3)

小池 和男

目次	
序章	長期の競争の重要性 「経済志林」 80 - 4、萩原進教授記念号
第2章	コンビニエンス・ストアの革新
第3章	ソフトウェアの技術者たち
第4章	生産ラインの設計・構築
第5章	ファイナンス—投資銀行とヘッジ ファンド
第6章	企業のガバナンス
終章	長期の競争の要件

第3章 ソフトウェアの技術者たち

1. なぜソフトウェア開発をとりあげるか

イノベーションはあったか

非製造業からソフトウェアの世界をぜひともとりあげたい。情報化の流れは滔々としてますます大きく、とても看過するわけにはいかない。かなりの文献もある。

だが、そこにいくつかの疑問がある。まず日本のソフトウェアに、たとえ小さなイノベーションでもほとんど認められない、という議論が少なくない。だが、はたしてそうだろうか、という疑問である。このシリーズは前章のセブン—イレブン—ジャパンのように、なんらかのイノベーションをおこした事例の追及を主眼とする。そして、そのイノベーションをになう人材の形成に長期の視野が必須かどうか、その吟味にある。

ところがソフトウェアの世界では、OS operating systems など基本ソフトはもちろん、アプリケーションソフトも汎用であるかぎり、ほとんど日本の出番はないかにみえる。O

Sのソフト、また汎用アプリケーションであればワード、エクセルなどおもな製品は、すべてアメリカ製である。日本製の汎用ソフトで他国に輸出され大いに活用されている例を、わたくしは寡聞にして知らない。

その結果、ソフトウェアを開発する技術者の活躍が日本できわめて弱い、とみられている。米であれば、「英雄」ともいべきソフト技術者が、設計もプログラミング＝コーディング（米風のいい方）も両方こなす。というより設計とコーディングはいわば行きつ戻りつおこなうもので、最初に設計しても、コーディングしながらよりよい考えが浮かび、設計を直したりして高度なソフトをつくっていく。だから、ソフト技術者の腕いかんで10倍、20倍も効率が違ってくる、といわれる。ところが、日本では、設計はソフト開発の本社技術者が担当し、そのコーディングを下請けにまわす。それでは設計とコーディングの相互作用ができず、高度な仕事ができない、といわれる。（中島「2011」pp. 58 - 59）。

それに、日本のソフトウェアの注目すべき事例があまり見つからない。このシリーズはなるべくならばひとつの事例をとりあげ、それを掘り下げていくという方法をとりたい。前章もそう、セブン—イレブン—ジャパンをとりあげた。ところがソフトウェアではそれを見つけるのが容易でない。

ゲームソフトは？

というとあるいは疑問におもわれよう。かのゲームソフトの分野では、まさに日本企業が海外をリードしてきたではないか。ソフトにかぎらず日本企業一般の方式では創造性をそでてるのが苦手といわれてきたにもかかわらず、ゲー

ムソフトでは見事な成果をあげてきた。しかも、それを立ち入って調べた研究も少なくない。ただし、それらはあとでみる。なぜこれを主役にしなないか。その理由はふたつある。

第一、すぐあとでみるように、ゲームソフトの比重は日本のソフトウェア開発業界ではまことに小さく、2010年時点でわずかに売上では2%にすぎない。2%の分野を代表例としてとりあげるのは、いかがなものであろうか。前章のコンビニは小売業界の主役となった。

第二、このシリーズの主眼は人材形成にある。ところがゲームソフトの研究論文は、人材形成への観察がとぼしい。それをみると称しながら、実は肝心のキャリア、関連の深い仕事群を経験しながら技能を形成していくかどうか、それをなんら見ていない。それではここで主役にする材料がたりない。

もっともゲームソフト業界を研究した業績は、それなりに重要な含意をもつ。その点をふくめてこの章の最後にふりかえる。では、だれを主役にすればよいか。その探索には日本のソフトウェア開発をまず概観せねばなるまい。

いちじるしい伸び率

日本のソフトウェア開発業は、他国と同様いちじるしく伸びてきた。さしあたりその点を雇用の統計で一瞥しておきたい。とはいえ、その観察はととても一瞥などという簡単な手続きではすまない。

なるほど、経済産業省のふたつの統計シリーズは古くから毎年つづいている。「特定サービス産業実態調査」の「ソフトウェア業編」と、「情報処理実態調査」である。前者はソフトウェア開発業を対象とし1973年から、また後者はソフトウェアを利用するユーザー側を対象とし1969年からある。それぞれソフトの従業者数は記載されている。雇用の伸び率はよくわかるが、それはソフトウェア関係のみの数値で、他の産業分野の雇用とくらべていない。直接くらべないと真の伸びがわかりにくく、それには同じ統計でないといろいろ支障がでる。

おなじ統計でくらべるならば、産業分類や職業分類がもっともこまかい国勢調査を使うのがふつうであろう。ところが国勢調査でソフト関係の職業小分類が明示されたのは、もっともあ

表 3-1 ソフトウェアの雇用の伸び
—就業者、国勢調査

年次	1985	1990	1995	2000	2005
全産業就業者計増加率 (%) *		5.7	4.0	-1.8	-2.8
34 情報サービス業 (千人)	304	573	673	909	972
増加率 (%)		88.6	17.5	35.0	7.0
99 ソフトウェア業				668	768
増加率					11.7
100 情報処理・提供サービス業				203	205
増加率					1.1
35 インターネット 付随サービス業 (千人)					61
34 と 35 の合計 (千人)					1,034
増加率 (%)					13.8

注：この中分類は2005年調査のものである。それは「99 ソフトウェア業」と「100 情報処理・提供サービス業」という小分類からなっていた。それ以前はほかに「その他情報サービス業」という小分類があった。なお、日本標準産業分類2007年とくにソフトウェア業は大幅に改訂された。

出所：*は「平成17年国勢調査 日本の人口」p.158より。

他は各国勢調査、第5巻その1全国編より。抽出集計である。

たらしい 2005 年調査からにすぎない。これでは伸び率はわからない。せめて産業分類でみるほかない。産業分類はほぼ毎回改定されながら、最新の 2005 年調査の分類区分をとると、中分類 34「情報サービス業」がなんとか前から接続できる（前は小分類あるいは細分類であった）。ソフトウェア業が前面に出てきたのは 1980 年代からであろうから、ここでは 1985 年国調からの数値を、接続できる分だけ、表 3-1 に掲げた。なお、2005 年国調から中分類 35「インターネット付随サービス業」も掲げられるようになった。おそらくそれ以前は「その他情報サービス業」という分類に含まれていたのを、分けて別掲したのであろう。

表によれば、ソフトウェア産業の雇用の伸びは依然すばらしい。なるほど前から接続できる「34 情報サービス業」の数値をみれば 2000 年以降大幅に減速した。だが、減速した 2000 年以降でも、全産業の就業者が 5 年間で 2.8% 減少したのにたいし、ソフト産業の雇用は逆に 13.8% も増加した。「34 情報サービス業、35 インターネット」の合計数値でみている。そのほうが実態をよくあらわすであろう。これをめざましい伸びといわなければ、いったいなにをいうのであろう。まさに情報化社会の明瞭な進展である。この分野を逸することはできない。

顧客ごとのソフト開発

では、日本のソフトウェア業の主役はだれか。それは個々の顧客用のソフトの開発者である。しばしば「受注ソフトの開発」といわれる。まずその点を統計数値で確かめておこう。

表 3-2 から受注ソフト開発が圧倒的であることを知る。ソフトウェア業の売り上げの、じつに 85% にのぼる。2010 年時点の数値である。そのソフトウェア業がソフトウェア開発にあたり、情報サービス業の大半をしめる。他の情報処理・提供サービス業などは、たとえば会計業務の計算サービスを個々の顧客から請け負うことをいう。ここではソフトの開発に注目する。それならば日本の特徴は、個々の企業の受注ソフトの開発がきわめて多いこととなる。

そのソフト開発の方式

そのゆえに、日本のソフト開発業にはイノベーションがない、との議論が盛んになる。なかでもマイクロソフトの米本社にも長年つとめ、主要な汎用ソフトの開発にたずさわった中島聡 [2011] は、その点を強調する。つまり日本は発注先が設計し、それを下請けがプログラミングする。ところが米のマイクロソフトでは、コーディングも設計もひとりのプログラマーがおこなう。そもそもよい設計とは、コーディングし

表 3-2 「受注ソフト」の比重
—ソフトウェア業務の内容、2010 年

	年間売上高 (百万円)	(%)
ソフトウェア業	10,164,191	100
受注ソフトの開発	8,555,656	84.9
業務用パッケージ	1,090,662	10.8
ゲーム	215,318	2.1
コンピューター基本ソフト	216,777	2.2
情報処理・提供サービス業	3,393,047	
インターネット付随サービス業	1,229,354	

注：

- 1) その事業所の主業務によって分類したものである。
- 2) 事業所規模 5 人以上にかぎる。

出所：経済産業省「平成 22 年特定サービス産業実態調査報告書 ソフトウェア業、情報処理・提供サービス業、インターネット付随サービス業編」2012 年

ながら、あるいはコーディングとは往ったり来たりしながらできるものであって、設計が前提されたばあいのコーディング方式では、とうてい創造的な仕事はできない、と力説する。

だが、それはおそらくふたつの事情によるかもしれない。第一、中島の語る状況は米でもマイクロソフト特有の方式かもしれない。米のソフトウェア開発のやり方では、やはり設計とコーディングを（日本ではプログラミングという）、それぞれ別の人が担当するのが多いらしい。だが、ソフト開発技術者のキャリアとしても分かれているかどうか。この点は、あとで事例にたちいって吟味しよう。

第二、そしてより重要な理由は、極度に大勢の人が利用するすばらしい汎用ソフトウェアの開発者だから、とおもわれる。標準となるOSのソフトを開発する。のみならず、そのうえで動く汎用アプリケーションソフトの開発を制覇している。ワードやエクセルなどである。このような広く利用される汎用アプリの開発は、まことに創造的なソフト開発技術を要求する。最初のソフトの開発はもちろん、改訂版では前のバージョンとの接続の問題、また世界中の膨大な人が使用するさまざまな機種との適合性の問題などは、はなはだ面倒な技術、工夫を要する。そうしたばあいには、すぐれて創造的なソフト開発技術者が、設計もコーディングもこなすほかないのかもしれない。

だが、そうした極度に大勢が利用する汎用ソフトのメーカーの数は、ごくわずかとなるだろう。周知のように、汎用ソフトの世界はめざましい収穫増だからである。世界でひとにぎりの勝者がのこるだけであろう。かりに汎用ソフトの開発で技術水準を判定すれば、米以外、西欧のどの国の業界もイノベーションがないとはいわないまでも、ごく少ない、といわざるをえないであろう。せいぜいドイツ製のSAPなどである。それならば、なにもひとり日本にとどまらず、米以外のどの国も敗北したことになる。だからといって、それぞれの国のソフト業界が死滅したか。まったくそうではない。その個別企業用のプログラムの開発という他の分野では、それなりに健闘しているのではないだろうか。

ただし、その人材形成について立ち入って解明した文献はとぼしい。日本のわずかな文献も1990年前後に集中し、近時はむしろ減少したかにみえる。¹⁾ それゆえ、やや立ち入って観察している1990年前後の文献から検討していく。

2. 1990年代の日本のソフトウェア技術者

やや大規模な事例

わたくし自身が聞きとりした事例からはじめることをお許しいただきたい。というのは、わたくしのみるところ、当時の人材形成にもっとも立ち入った観察だからである。数例尋ねたが、もっともくわしく話を聞くことができた1事例を記す。すでに公刊している本に多少とも記しているけれど（小池 [1991] pp. 92 - 96）、実際の聞きとりはそれを大分うわまわる。聞きとりノートは手許にのこっており、公刊本よりも具体的により詳しく記したい。

1, 200人規模のソフト開発企業の事例である。日本のソフトウェア開発企業としては大手のひとつであろう。その上級管理者2人の話を伺った。ひとりの方に2回計2時間半、他の方には1回1時間であった。²⁾

この事例はまさに日本のソフト企業の持ち味、つまり「受注ソフト開発」が中心であった。ふたりの話し手が共通して力説するのだが、ソフト技術者にもっとも要求される技能は、いわゆるソフトの技術ではない。それはそこそこできればよい。はるかに、対象とする企業の仕事かどのように流れているか、その分析こそ重要だ。分析がすっきりしていると、明晰なソフトができる。それが下手だと開発されたソフトが冗漫になる。たくさんバグ（原意は害虫）もでる、というのである。この点は他の数例の聞きとりでも共通に強調された。

ここから明晰なソフトを開発する技能、それを形成するキャリアが導かれる。まず仕事の流れの分析力を高めるにはどうしたらよいか。仕事の流れはおなじ業種であればすくなくからず共通する。もちろん業種の中でも個別企業による差異は当然にのこる。だが、業種の共通性が大きい。そこで担当する業種をなるべく固定し、

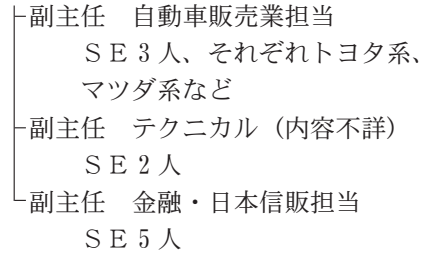
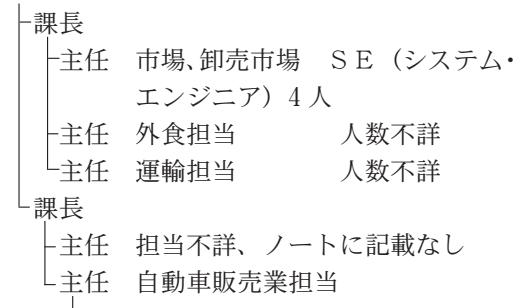
そのなかでさまざまな事例、多くの対象企業を経験する方式をとる、というのである。まことにもっともな育成法ではないだろうか。それに業種に注目するならば、それまでその業種で受注した事例の社内のソフトが蓄積されているはずだ。それらにすこし手を加え再活用していく利点がある。

ソフト技術者の実際の育て方をみよう。なによりもまずその実務、すなわち仕事経験を観察しなければならない。それには一見迂遠のようだが、この事例の組織からみていく必要がある。庶務グループなどを別に、主体のソフト開発関係はつぎの10のグループにわかれる。製造・流通、通信業、官庁、公共・金融、運輸・報道、医療、宇宙制御、通信制御、OA・端末、基本ソフト開発、以上である。つまりはなはだ広い分野におよび、おもな顧客は日本の大企業、官庁、そして大病院などである。ソフト技術者はよほどのことがないかぎり、このグループのどれかに属し、めったに他に移動しない。ある分野に特化するのには、対象事例の仕事の流れを把握するためには、まことに肝要なのだ。さらに立ち入って、とりくむ仕事を一段とこまかく観察していく。

最末端の組織

具体的に個々の仕事経験を観察するには、労働の最末端単位に立ち入る必要がある。いま製造・流通グループをとる。その下に3つの部がある。そのひとつが流通担当で、そこに30人ほどのソフト技術者が属している。その組織を下に示す。

流通担当部



ソフト開発技術者はまずはひとつのプロジェクトに属する。プロジェクトとは継続期間も人数も流動的だが、ほぼうえの主任単位の組織とみて大過ない。さしあたり採用からそのキャリアを追うことにしよう。

この企業は新卒採用が基本という。大卒とソフト技術の専門学校卒もいる。ここは全国で9校のソフト専門学校を運営しているのだが、基本的には採用は大卒中心である。それは大卒が中長期には結果的に専門学校卒より伸びるからだ、という。その理由を問えば、論理的な思考力こそが肝要で、専門学校卒はなるほどコーディングの技術が最初は大卒よりできるけれど、すぐに大卒が追いこす、というのである。

大卒といっても情報工学や理科系とはかぎらない。文科系でもよい。哲学系はとくによい。論理的に思考できるからだ。ただし、宇宙制御グループの仕事などは数学物理系にかぎられる。というのはユーザーの要求を理解するのに、数物系の素養が欠かせないからである。

文科系でもソフトのいわゆる技術、C言語などの修得は、さしあたりは入ってから研修による。2か月間全日の技術研修である。そこでフローチャートを書き、それをプログラミングすることを学ぶ。そのあと、どれかの主任のチームに配属される。さらに実務についたのちも研修があるが、はるかにそのあとの実務こそが肝要というのである。

実務の内容はうえの流通担当部の組織図でわかるように、主任ごとの単位がほぼひとつのプロジェクトチームであって、ひとつの業種をうけもつ。もっとも自動車販売のように、自動車販売と信販の複数の業種をうけもつばあいもある。それはどちらもローンをあつかうという共通の部分があるからだ。新人はある主任の

チームに配属される。そこで、ひとつの業種をマスターする。そのなかでさまざまな企業を担当する。たとえば卸売市場の担当ならば、東京都の大田市場を担当することもあれば、他のさまざまな地の卸売市場を担当することもある。つまり似た業種で多様な事例を経験する。

その業種のなかでさまざまな業務がある。販売管理、在庫管理などである。それらの業務面もなるべく多く経験していく。販売管理にせよ在庫管理にせよ、業種特有の性質があり、共通面が大きい。つまり業種こそ肝要という方式である。なぜか。

ソフト技術者の仕事

その説明には、ソフト技術者の仕事をその流れにそって見る必要がある。³⁾ まず「要求分析」である。この事例では別の言葉でよぶが、ここではすべてやや一般的なことばをもちいる。「要求分析」とは発注側すなわち顧客がなにを求めているか、それを明らかにするにある。顧客も自分の要求をかならずしも十分に認識しているとはかぎらないので、相談、折衝がある。顧客の要求を聞き、それを明晰にする。それは大変な作業で、顧客の要求を聞きながら、それを実現できる方式、それを改善できる方式を提案することを意味している。ソフト開発のベテランたちは、こぞってこれこそがもっとも重要、と口をそろえて強調する。

ついで「基本設計」である。ソフトの流れを設計する。さきの要求分析と基本設計のもっとも肝心な必要技能は、顧客の業務の流れの分析となる。明晰な業務の分析であれば、よりすくないコードでソフトの設計ができる。バグも少なくなる。他方、その理解があさければ、冗漫なソフトの設計となり、当然にバグも多くなる。

のち詳細設計となる。小さな単位にわけて設計する。モジュール設計とよんだりする。そのあとコーディングすなわちプログラミングとなる。もちろんテストがあり、手直しがあり、できたあとで保守がくる。

以上の工程に、人はどのように配属されるか。新人はまずコーディングに配置される。モジュール設計書にしたがいコーディングしてい

く。人により異なるが目途としていえば、1年目はプログラミングに従事する。2年目に詳細設計、4-5年目には基本設計にすすむ。もっとも事態はすくなくならず流動的である。というのは、プロジェクトごとの期間はさまざまで、短いものは3か月、長いものは5年などとなる。

しかもひとつのプロジェクトでも、その段階で必要人員が大幅にかわる。初期は4、5人でも、最盛期には20-25人にもなる。それゆえ、人はまず課のなかで移動する。たとえば自動車販売ならば、トヨタ担当とマツダ担当で助けあい、さらに信販にも出向く。ときに主任間も移動する。

主任、課長クラスとなると、自分もコーディングしながら、顧客との折衝を担当する。ここではまさに顧客の要求内容を明確にすることが肝要である。そのためにこそ、顧客の業務内容、その論理的な分析を修得しておかねばならない。それはひとつの業種の多様な事例の担当からうまれた知恵である。その知識がないと顧客との折衝、説得がうまくいかない。ここに要求される技能が形成される。あと業種をかわることもあるが、それは個人の希望と、そのときの人員事情による。人員事情とは、人手不足の部や、あらたな分野の発足などである。

技能の形成

さらにキャリアのはじめだけでなく、キャリアの間にもたびたびの研修O f f J Tがある。あらたな言語、その他の技術の修得に、毎週1回3時間の研修がある。基本技能はいわゆる情報処理1級試験の受験勉強でほぼ修得するが、それをこえた技能をこうした社内の研修で習得する。講師は社内のベテランであり、毎回その修得の確認試験がある。その成績は社内資格の昇格のひとつの条件とされている。

そうじて3つの傾向が鮮明である。

第一、対象とする業種、企業の仕事の流れの分析、その明晰化の重視である。そのためにひとつの業種のなかで多様な事例を経験する。そして類似の業種を経験して、さらにその技能を高めていく。

第二、技術的な修練であって、プログラミン

グなどから詳細設計、基本設計、さらに顧客との折衝など、しだいにより高度な仕事にとりくむキャリアである。たとえばもっとも高度な顧客の要求分析の技能修得は、つぎのようにしておこなわれる。顧客との折衝の場に、プロジェクトリーダーがつぎのリーダー候補を同行する。そしてそのやりとりを記録させる、などという方式をとる。この特徴は、マイクロソフトとは違い、設計者とプログラミングする人がひとまず別とおもわれる。前者を「SE システム・エンジニア」、後者を「プログラマー」と日本ではよぶ。だが、キャリアとしてみれば、設計者SEもかつてはプログラミングをおこなっていた。以上は社内の実務経験であった。

第三、C言語などあらたな技術の習得である。それはおもにさきにもふれてきた社内研修ではげむのであった。

こうした方式はまことに効率的とおもわれる。他社から経験者を採用する方式と対比してみよう。ソフト技術の経験者、たとえばC言語などを習得した人などは、むしろ容易に他社あるいはソフト技術者の労働市場から採用できよう。だが、技能の肝心の内容が、対象となる業種の仕事の流れの分析、それもその業種のなかの企業による微妙な違いの把握となると、その業種につき、そうした経験をもつ中途採用者を見つけるのは容易ではあるまい。しかも、かりに経歴がかなり具体的にわかったとしても、その真の仕事の実力をその人が勤める会社の外部からかなり正確に把握するのはむづかしい。その仕事ぶりをたびたび見ていないと無理であろう。いやいっしょに仕事していないと、その真の実力はなかなか誤差少なく把握できまい。

さまざまな情報処理の社会的な資格は、それにたいする各ソフト開発企業のとおり扱いからみるかぎり、いわばその職業の入口、せいぜい中の下クラスを意味するにすぎない。医師や弁護士資格となんらかわるところはない。ではいったい、ソフト開発の技能の根幹、業種なり企業の仕事の流れの分析の力を、どのように測り表示することができるのであろうか。そうじて企業内部での人材形成方式、その認知こそが、個々の技術者の技能レベル、その向上度を知る

のにはるかに有効かつ効率的であろう。

ただし、こうした特徴はたとえ当時は効率的でも、なにぶん20年も前の状況であった。その後、環境もソフト開発の技術も変化したかもしれない。それによってこれらの特徴はかわったのであろうか。それとも依然保持されているのだろうか。最近の人材形成の状況を知るよい文献がなかなかみつからない。一見それにふさわしいタイトルをもつ文献がないではない。たとえば三輪 [2001] である。なかでも3人のベテランに面接している。だが、その関心はキャリアの各段階での心理的な傾向の認定にあって、キャリアの実際そのものの情報はすくない。

ただ近時の文献、藤田、生稲 [2010] は、日本のソフト開発での受注型分野の重要性を充分強調する。そして受注ソフトの重要な技能は顧客の業務の理解こそ、と指摘している。うへの1990年ごろのわたくしの記述が、いままも妥当なことを裏づけているかにおもわれる(p. 225)。ただし、残念ながら人材形成、キャリアへの観察はみられない。

日本情報システム・ユーザー協会調査

日本情報システム・ユーザー協会(JUAS) [2011] の4冊本は、まさに「IS人材のためのキャリア形成のヒント」と題し、この文章の関心に近いかにみえる(ISとはinformation systemsのこと)。とりわけもっともページ数の多い第2巻は(260p.)、IT部門の50人のベテランのキャリアを紹介している。その紹介は各人A4版4ページにもおよび、一見参考になるかにおもえる。

ところが、この文章が関心をもつ事項、すなわち日本のIT技術者が設計のみならずプログラミングも同時に行うのかどうか、その点については、あまり記載がない。キャリアの各段階でどのようなプロジェクトに従事したかは、かなり書かれている。だが、そのプロジェクトのなかでどのような役割をになったかは、あまり書かれていない。むしろ人生訓風のことばに多くのスペースがあてられている。

ただし、なかでインタビューの資料としての事前アンケートへの記入がある。各人のスキ

ルの自己評価である。「事業戦略策定」からはじまり「システム監査」にいたるまで、30項目への記入がもとめられている。そこからプログラミングと設計など、ソフト開発のプロセスの節目とおもわれる事項をとりだし、それへの回答を一覧表にしたのが表3-3である。

仕事内容をあらわす言葉づかいは、すべてもとの資料によっている。そして言葉の定義は一切ない。また答えた50人のベテランをいかに選んだかも、まったく説明がない。勤務先も姓名の明記もあれば、たんにサービス業、A氏などと記すこともある。

ただ、そこに登場する人はほぼ40歳代中心、すなわち働き盛りである。それに協会のこの調査は、経済産業省の担当課がスポンサーであることを明記している。わたくしもかつて情報関連の似た組織の調査委員会の長をごく短いなが

らつとめた経験があり、情報業界での経済産業省後援の威力を知っている。IT業界が大いに協力したとおもわれる。それを用いよう。

この日本情報システム・ユーザー協会JUSはITのユーザーの組織で、実際にプログラミングを開発するいわゆる「ベンダー企業」はごくすくない。いわば発注先である。ただし、ここで面接された人たちは発注先のIT部門のスタッフである。50人のうち入社以来IT一本、あるいは他部門経験がせいぜい2、3年の人、つまりIT部門中心の人は29人、のこる21人の大半は、やはりIT部門の経験が長い。他の業務もすくなくならず経験している。ごく一部に他の業務が中心でIT部門勤務が短い人もある。そこで表3-3はIT部門中心29人とその他21人にわけた。

スキルを5のレベルにわけている。記入要綱

表3-3 50人のベテランの仕事経験

仕事の種類	自己評価によるスキルレベル				
	レベル1	レベル2	レベル3	レベル4	レベル5
「システム要求定義」					
IT中心	1		4	9	15
他部門も	1	1	1	11	7
「業務プロセスの詳細設計」					
IT中心	1		5	13	10
他部門も	3		5	7	6
「アプリケーションの分析設計」					
IT中心	2		5	12	10
他部門も	4	6	7	4	
「アプリケーションの開発」					
IT中心	3		5	18	3
他部門も	7	5	7	2	
「保守実施」					
IT中心	6		11	10	2
他部門も	6	4	8	3	
「運用実施」					
IT中心	3	2	13	10	1
他部門も	6	5	6	4	

出所：日本情報システム・ユーザー協会（JUS）「IS人材のためのキャリア形成のヒント 第二巻 ISプロフェッショナル紹介編」

によれば、レベル1は「未経験」、レベル2は「上位者の指導の下で遂行できる」、レベル3は「独力で遂行できる」、レベル4は「指導できる」、レベル5は「社内をリードできる」とされる。ここでの関心からすれば、レベル4以上に注目すべきであろう。

最近の状況

この調査結果を摘録したのが表3-3である。

表からつぎのことが読みとれよう。第一、プログラミングが相当にできる人は、IT部門中心の人にほぼかぎられるようだ。ここで「プログラミング」とは、表では「アプリケーションの開発」となろう。「相当にできる人」とは、レベル4以上、すなわち「指導ができる」レベル以上とする。そうするとIT部門中心の計29人のなかで、21人である。とはいえ3人はまったく経験がなく、他の5人は「どうやらできる」レベルにすぎない。つまり、プログラミングはまったくのベンダー、すなわちソフト専門企業まかせが少数ながらある。ここからいえるのは、一部の日本企業IT部門では、ソフト開発の大半をITの下請け企業に任せている、という姿である。それでも大半は発注側も、プログラミングを指導できるレベルまでマスターしている。

第二、他方、IT部門中心でない人たちは、プログラミングがあまりできない。レベル4以上は21人中わずか2人にすぎない。すなわち下請けにまかせている。したがってIT中心グループもあわせ全体としてみると、ほぼ半数が下請けまかせ、という姿がうかびあがってくる。これをそのまま解すると、中島[2011]の指摘が妥当する部分はすくなくとも半分におよぶ。

第三、これにたいし「要求分析」や「業務プロセスの詳細設計」は、IT部門中心であれ、そうでないばあいであれ、大半の人がレベル4以上である。すなわち発注側の大半が、設計を自分でもこなせることがわかる。とはいえ、発注側のスタッフがソフトの技術をもっているもプログラムはまかせ、という部分が多かろう。その意味では、中島の指摘はかなり当たってい

る、といわねばなるまい。

第4、保守、運用は、IT中心であれ、そうでない人であれ、おそらくはふたつのグループにわかれる。保守運用をかなり熟知するグループと否である。ソフトは開発のあとの保守、運用が相当に重要だが、それを社内のスタッフでおこなうばあいと、外部のソフト専門企業にゆだねるばあいの、ふたつのタイプにわかれるのであろう。

以上のことから、つぎのように推測できる。設計とプログラミングをひとりで受け持つ方式ではないようだ。まずはプロミング、しだいに詳細設計、基本設計、そして要求分析と、うけもつ仕事が高度化していく。ただし、高度化した仕事の担当者は、すくなくともその半数は相当のプログラミングの技能を身につけている。こうした特徴がどれほど米とかさなるのか、それともくい違うのか、それを次節で検討したい。

3. 米のソフト技術者

マイクロソフトの事例

米ではやや一般的な記述はあるにしても、多少とも具体的な説明があるのは結局マイクロソフトとなるようだ。Cusumano [1995、訳1996]とZackary [1994、訳1994]であろうか。しかし、あまり具体的ではない。Cusumano [1995]は大部な著書であり、わざわざ人材開発の章を設けながら、すべて記述はごく一般的で、具体的な例示にとぼしい。これにたいし人材開発の章はとくに設けていないながら、Zackary [1994]の方が話はより具体的である。NT New technology という基本ソフトの開発にしぼって、その開発者たちの群像を立ち入って描いているからであろう。それを頼りにするほかあるまい。そこからわかることを指摘しておきたい。

第一、設計者は同時にプログラミングも担当する。これは日本の状況とは大いに違う。それはマイクロソフトの明示的な方針である。(Zackary, 訳上, p.172など。またCusumano, 訳 [1995] 上 p.162など)。この点は米のマイクロソフトに10数年つとめた中島も強調するところである(中島 [2011] pp.58-9)。

第二、そのことは優秀なプログラマーを選ぶ、という方針を意味する。プログラマーの上手下手で10 - 20倍も効率に差がでることを何回も指摘する。(Cusumano, 訳上 p. 10, p. 95 など)

だが、第三、それは最初から平等に仕事を分担することではけっしてない。4、5人ほどからなる小さなチームに分ける。その点は他とかわらない。新人ないし技能の低いものは、そのチームのなかでもっともやさしい部分の設計とプログラミングを担当する。そしてチームのリーダーのつくった仕様書すなわち基本設計に沿って、プログラミングしていくのである(Zackary, 訳上 p. 182, Cusumano, 訳上 p. 104 など)。

第4、マイクロソフトの特色は、「プロジェクト管理」、「テスト」とプログラミングの担当者をはっきりとわけている。プロジェクト管理とは、営業の要望をプログラマーにつたえ、その間の調整をはかる役割である。自分でプログラミングするわけではない。さらにテスト担当とプログラミング担当を最初からわけ、ほとんど1対1で対応させる。できたプログラミングをすぐにテストする。うまくいかなかったばあいの原因を指摘し、その修正をすぐに促す(Cusumano, 訳、第3、5章、Zackary, 訳上 pp. 138 - 139, 訳下 p. 31 など)。

そのサラリー、はげしい長時間労働

そのサラリーは、レベルもきわめて高くかつ成果につよく依存するもの、とおもわれよう。なにしろ、すぐあとにみるように目をみはる長時間労働である。しかもそれこそ創造的な仕事への報酬である。当然の想定とおもわれよう。だが、基本的には米のごくふつうの企業のホワイトカラーサラリーとあまりかわらない。基本給は社内資格給 (pay-for-job grade) である。社内資格 (job grade) の数はすくない。範囲給 (range rate) で査定つきの定期昇給制だが、その範囲給の大きさは不詳である。マイクロソフトの特徴は、ふたつある。ひとつは、その範囲給の高さが同業他社にくらべむしろ低い。けっして高いとはいえない。それがビル・ゲイ

ツの方針と明記されている (Cusumano, 訳上, p. 130)。

もうひとつは、かなりの株式オプションがつく。それがマイクロソフトの激しい株価上昇によって莫大な報酬となる。さらに、技能の低いソフト技術者を解雇するようだ。下位5%ほどを解雇する方針との記述がある (Zackary, 訳下, p. 242)。ただし、それがどれほど実施されているかどうかは不詳である。ただ、優秀なソフト技術者偏重の傾向はこうしたことからもうかがわれよう。

そして、すぐれたソフト技術者を極度に活用する。つまりものすごい長時間労働である。100万都市シアトル近郊に職場があり、職と住は東京にくらべはるかに近接しているのに、帰宅は遅い。ときに職場に泊まり込む。なにしろ米は西欧とならび、日本の常識とはまったく違い、技術者のみならず大卒正規ホワイトカラーには残業手当を一切はらわない。したがって労働時間の統計はない。さまざまな文献が日本のSEは長時間労働と非難するが、日本など問題にならない長時間労働といっても過言ではなからう。日本風にいえば、おどろくべき「超サービス労働」である。それでは燃え尽き症となり、途中退職、離婚がたえない。

さらに技能の修得は基本的に実務であり、なにも研修は用意されていない。新人はチームに入り、そこで先輩のメンターが指名され、わからないことはかれに聞いておぼえていく。その点では日本と基本的にかわらない。情報処理試験などの受験、そこでの資格取得は米でみられない。まずそうした制度がない。設けよとの声はあるが、いかに日本の「常識」、欧米こそ資格社会との「先入主」が実態とはずれていることか。いろいろな国のさまざまな産業の職場をみていくと、この方がまことに自然で、それがマイクロソフトにも実践されている。実務こそという原則は日本以上に貫徹されているのだ。

マイクロソフト以外の事例

これだけみると、肝心の点で日米の差異がかわだっているかにおもえる。だが、同じ米でも同業他社はやや異なるようだ。ソフトの設計者

とプログラマーが別の部署に属する企業の存在の記述が Cusumano [1995] (訳 [1996] 上 p. 105) にもある。ただし、これを確かめるよい文献をみつけることができなかった。McConnel [2004] (訳「2005」) や Perry, Ermel, & Shields [1994] (訳「1995」) などの文献があるが、その技能形成や報酬はあまりに一般的な記述にとどまり、具体的なことはわからない。

Cusumano & Yoffie [1998] が人材の採用と報酬につき短いながら言及している事例もある。インターネット関連の急成長企業である。94年数人で起業、2000人規模まで急成長、99年他社に合併された。その事例では同業大企業から、すでに技能をしっかりと身につけた人材を中途採用するのがふつうであった。技術者なら20歳代終わり、マーケティングなどなら30歳代終わり、もとのサラリーを上回る高給でひきぬいた、という (訳 [1999] pp. 77 - 80)。

これほど急成長の事例ならば、できあがった人材を中途採用するのは充分うなずける。というより、ほかにあまりよい方法もあるまい。ただし、すんなりともとの大企業でのサラリーを上回ったのは一般職で、上級職のサラリーの払い方はわりとマイクロソフトに似ている。基本給は低く、ボーナスを高額にしている。それにしても立ち入った具体例への言及はまったく見られない。

シリコンバレーの1事例、1992年

1992年とかなり前なら、わたくしがスタンフォードでソフト開発技術者3名別々に短い時間ながら聞きとりしたことがある。うちもっともくわしく聞いた1例は、すでに小池「1993」に記した。もっとも、わずか3ページにすぎない (pp. 74 - 76)。さいわいその聞きとりノートがのこっており、公刊分よりもやくわしく記しておく。

ただし、この聞きとりは不十分であった。というのは、ソフトウエア労働の吟味が主眼ではなかった。そのときの目的は米の経営エリートにあった。その有無、その育て方つまり「特急組」があるかどうか、そしてそのキャリアが複数職能型か、あるいは主にひとつの職能という

専門型か、それを調べるにあったからだ。そしてそれぞれ1時間ていど1回かぎりの聞きとりにすぎない。

聞きとりの対象はスタンフォード大学ビジネススクールの「エグゼクティブ・クラス」のメンバーであった。エグゼクティブ・クラスとは、会社派遣のエリート中年社員たちである。1年間かなりの金額を住居費も含め会社が払い、このコースのあと、その企業の事業所長などかなり高い役割につく人たちであった。ただし、わたくしがそうした人たちへの聞きとりを思いわたったのは学年のほとんど終わり、すでに一部の人しか学校にのこっていなかった。当時わたくしはその教員で、わたくしの後期のクラスにエグゼクティブ・クラスの学生が参加したことで思いついたにすぎない。以上の制約つきではあるけれど、参考までに記しておく。もっともくわしく聞くことができた1例を記すにとどまる。

プログラムから設計へ

その事例は従業員数725名、中規模のソフト開発企業である。主要製品はいずれも汎用アプリケーションのソフトである。そのひとつ、企業のプレゼン用ソフト開発のチームに焦点をすえて説明しよう。話し手はソフト開発の企業を以前2社経験している。2社目はヒューレッドパカードで、企業向けプレゼン用の汎用ソフト開発を提案してうけいれられず、9年前独立し、いまの会社をつくった。つまり相当に経験のあるソフト開発技術者である。

企業向けプレゼン作成用の汎用アプリケーション開発のチームに注目する。その開発チームは、プロジェクトの初期は8人、5人の開発者と3人のテスト担当者である。それにマニュアル書作成担当1人がくわわる。最盛期にはそれが20 - 25人にもふえる。その点はマイクロソフトとかわらない。チームはおなじソフトにふたつある。奇数バージョン用と偶数バージョン用である。ほぼ1年半ごとにバージョンをアップする。つまりひとつのチームは、ほぼ3年は固定する。けっして短期ではない。

新人はどれかのチームに配属される。もっと

も同業他社からの経験者採用も多く、初任配属はさまざまである。経験のあさいものは、まずコーディングすなわちプログラミングをうけもつ。ときに、やさしい部分の詳細設計とそのプログラミングにつく。さらにその汎用アプリケーションのなかで、しだいにより高度な部分をうけもつ。途中テスト担当にもなる。こうした点ではマイクロソフトとやや異なり、日本の話に似てくる。そして5,6年でプロジェクトリーダーになる。プロジェクトリーダーになっても自分でプログラミングもする。その点ではマイクロソフトと似る。このようにマイクロソフトと似る点、日本と似る点の双方がある。日本だけが特殊とみるまでもない。

以上を要するに、マイクロソフトはやや短期の視野におもえる。すぐれたソフト技術者を育てるよりもそうした人をみだし、それを活用する。そして、あまりの長時間労働ゆえに燃え尽き離職する技術者も当然にでる。だが、マイクロソフトが米のすべてではない。同業他社ではそれなりに仕事をわけ、その間をやさしい職務からより高度な職務へと経験していくようだ。それなりに長期を要しよう。その点では日本とあまりかわるまい。

そうじて日米をつうじ最先端と目される分野でも、その中堅層の人材形成には、ときにやや長期を要することを否定できまい。

4. 受注型ソフトへのつよみ

米でも受注ソフトが多い

うえで日米を多少とも対比してきた。日本は受注ソフト開発の分野が主流であった。そしてそれが日本の特徴とおもわれてきた。もっとも、それを日本の特徴といいきるには他国の統計との対比が必要だが、寡聞にしてよい統計を知らない。統計とはべつに周知の米の事例をみれば、じつは案外に受注ソフトが多く、しかも増大しつつある。

なるほど一見日本とはまったく逆に、基本ソフトとアプリケーションの汎用ソフトがまことにつよい。だが、そのことは受注ソフト開発がアメリカで乏しいことを毫も意味しない。長年

各国のソフト業界をしらべてきたクスmanoは、しだいに売上や収益にしめる割合からみて、米でも特定顧客用すなわちカスタマイズされたソフトが高まってきた、という。汎用のソフト、ましてや標準となるソフトの開発は、収穫逓増ゆえにごく少数のメーカーが勝利者となるにすぎない、マイクロソフトなどである。しかし、受注ソフト業だけをとりだした数値は不詳だが、メンテなどをふくめたいわゆる「サービス」分野をみれば、法人向け営業のソフト企業売り上げの7割をしめる、という(Cusumano[2004]訳[2004] p.74)。しかも増加しつつある。規模からみても、上位企業のなかにサービス中心の企業がかなりの割合をしめるにいたった。

Cusumano [2004] はソフト業界の将来の姿をつぎのように描く。a. 汎用ソフトをつくる「製品」企業、b. メンテやカスタム化を中心とする「サービス」企業、c. そしてその中間の「ハイブリッド」企業、以上3タイプであり、うちcが将来の中心になる、と予測する。なお「サービス」とはそれぞれの顧客にたいし、その汎用ソフトのメンテナンス、新たなバージョンの適用、さらにそのインストールを請け負うのみならず、それぞれの顧客ごとにプログラムを変える。カスタマイズするのである。それもほんの一部を変えるとはかぎらない。クスmanoによれば、20-50%ほどを変える、というのだ。

クスmanoはその理由をつぎのように説明する。業界標準となる「製品」をつくりだすのはいわばベストセラーをあてる出版界のようなもので、あたれば凄いいけれど波がはげしい。他方、「サービス」は堅実に稼ぐから、というのである。

わたくしの考えでは、そうした理由も否定できないが、はるかに別の理由こそが肝要ではないか。別の理由はふたつある。ひとつは「組み込みソフト」の急激な増加である。他は、というより根源的な理由は、それぞれの企業の独自性こそ競争力の基盤、ということである。前者から説明しよう。

「組み込みソフト」の激増

「組み込みソフト」とは、いまやほとんどの製品に組み込まれたソフトをいう。早い話がテ

レビは、組み込んだソフトなしには消費者は使えない。携帯電話はいうまでもあるまい。延岡 [2006] (第8章7節 p. 228) によれば、テレビの設計開発工数に占めるソフトの割合は、松下電器の事例でみて1980年代10%にもおよばなかった。それが2005年には7割にのぼる。

いやそうした家電など電気製品にももちろんとどまらない。自動車はよく知られているように、エンジンはもとより、駆動系、ブレーキ系などほとんどの部分にソフトが入っている。飛行機、電車はいうまでもない。

この傾向はいうまでもなく消費者向けの製品にとどまるはずがない。工作機械など資本財でも、その設計開発にソフトがまことに重要な部分を占める。そして、そうしたソフトは汎用プログラムをどれほど転用できようか。もともと商品は各企業の独自性を売りものにしていく。まさに商品差別化である。それならばソフトも当然に独自性を要求されよう。すなわち受注ソフトの盛行である。

組み込みの受注ソフトが、いまのソフト開発のうちどれほどをしめているかは、政府統計のかぎりではわからない。つぎの国勢調査ではわかるように標準産業分類が改訂されているけれど、まだ数値はわからない。だが、かなりを占める、とみるのが自然であろう。そしてその勢いは、商品差別化を否定しないかぎり、まずつよまる一方であろう。

そして製品の企業独自性を活かそうとしたら、その新製品の機能上の特性をよく把握してソフトを開発するほかない。それには新製品のハードな設計の初期からの連携を必須としよう。遠くインドなどに発注すれば、連携がうすれ、かえって手直しがふえ、コストが高まろう、と延岡 [2006] (p. 230) も指摘している。

各企業の独自性こそ競争力の源泉

この点を製品開発にかぎらず、広くビジネスの分野につき強調するのが、もうひとつの理由である。およそ仕事をすすめるうえでの、個別企業の独自性の重視である。競争力の根源はそこにある。したがって、独自性をよくいかすソフトでないと、市場に生き残れない。法人向け

ソフトではカスタム化は必須ではないだろうか。

重要ゆえにあえてくりかえそう。トヨタ生産方式ははっきりした独自性がある。その点はだれも疑いが無い。それが普及し普通名詞のように各企業に広がっても、なお多少の独自性はこる。とりわけこの方式は最終的には職場の中堅層という大衆の実践によるところ大なのだ。職場の大衆の活動が企業をとわずまったく共通する、とはとうてい思えない。それならば企業ごとの独自性はどうしてもこる。それを汎用ソフトによってくまなく表現できようか。汎用ソフトはあるていどは有効に働くであろうが、その独自性の根幹を汎用ソフトが十分に反映できようか。

またトヨタ生産方式はたえず進化する。その変化を、汎用ソフトはその汎用性ゆえに、すぐさま追いかけることができるのだろうか。カスタム化が欠かせない。そしてその論理は、なにもトヨタや日本企業にかぎらず、米企業でもかわるまい。

その点を無視すると、SAPのような罠に陥る。SAPを例にあげたのは、いまなお有力なソフトメーカーだからであり、それほどの有名メーカーでもなお問題をのこすことをしめしたいのだ。周知のようにSAPは、人事をはじめ企業活動の多くの分野を総合した企業運営ソフトを売りものにしていく。ドイツが母国だが、欧州、米にも多くの顧客をもっている。そうした標準型の汎用ソフトを多様な企業のさまざまな分野にまで適用すると、うえで指摘した各企業の独自のよさが消えてしまうのではないか。

どうしたらよいか。そのすくなくとも一部を、個別企業用に変えることだ。いわゆるカスタマイズすることだ。それこそ受注ソフトの仕事である。だが、SAPはそれを拒んだ。業種ごとのソフトを24種類も用意し、それにしたがうよう顧客にもとめた。各企業の仕事の単位の定義も、それにあわせるように求めた。個別企業用に改修するのにSAPの許可を要求した、と仄聞した。それでは各企業の真のよさが充分いかせるかどうか。あるいはその良さが同業他社に伝わるおそれがこる。

また、クスマノのいう「ハイブリッド」タイプへの、ささやかな疑問ものころ。カスタム化するばあい、どうして汎用ソフトを前提にする方が有利なのか、その説明がCusumano [2004]にあまりみられない点である。もしソフトが特定業界用ならば、事例をとおしてすくなくならず共通する点もあり、そうした特定業界用のいわば準「汎用」ソフトの活用の理があろう。だが、それならば、その業界を長年あつかってきた、カスタム化専門の開発企業なら、つまりまさに日本の受注ソフト開発企業なら、その業界のおもな事例をあつかった経験をぎっしりと蓄積していよう。それらを活用する方が、すくなくならず有利ではないだろうか。変化もすばやくこなせよう。すくなくとも「ハイブリッド」型がサービス専門型より有利、とはいいい難いのではないだろうか。その点をクスマノは説明していない。

海外にでていくと

この問題はいわゆるグローバル化という、不可避の傾向にかかわると、ややこみいってくる。日本企業が海外に展開する。当然にその地の企業、あるいはさまざまな国の企業との取引が急増する。いま簡単化のために製造業を念頭におこう。その事業遂行にあたって中核メーカーと部品メーカーはあたらしい部品の発注、設計、その変更にはんばんなやりとりを必須とする。そのとき共通のソフトが欠かせない。

規格品ですみ、しかも仕様の変更がなければ、やりとりは不要となろう。いわゆるモジュール型産業ならば、それでよいかもしれない。最初の要件の広示ですむだろう。だが、他の産業タイプでは、部品メーカーとの取引だけでは、市場の競争になかなか生き残れまい。顧客の嗜好やさまざまなこまかい変化への対応に、ひんばんなやりとりを必須とし、共通のソフトがないと遅れるからである。

やりとりする相手は、日本から進出する部品メーカーばかりではない。その地の部品メーカーとも大いに取引し、やりとりせざるをえない。そうするようその地の政府の規制が働く。その地の部品メーカーとのやりとりを、どのようなソフトでおこなうか、という問題である。

日本国内なら、中核メーカーが常用するソフトをもちいてCAD computer-aided designなどをこなすことができよう。ところが海外の地ではそうはいかない。その地の部品メーカーはなにもその海外日本メーカーの専属ではない。もっとも日本国内でも多くの部品メーカーは、日本国内の中核メーカーの専属ではない。たとえばトヨタをとれば、部品メーカーの多くは独立のメーカーで、トヨタ以外の複数の中核メーカーと取引している。ただ、なにしろ長期の取引がつづいてきたので、この辺は国内ではほぼ解決されてきた。ところが海外のその地の部品メーカーはそうはいかない。その部品メーカーは当然に他の国のメーカーとも大いに取引しているだろう。その際すでにある種のCADのソフトなどを用いてきたであろう。それと調整しないわけにはいかない。

もちろんそうした調整はソフト間の変換プログラムをつくれれば、なんとかなる。その事例はすでにかなり前からある。自動車関連から例をとれば、ヤザキ総業である。ヤザキはハーネスをつくる。周知のようにヤザキは、日本自動車メーカーにかぎらずGMやフォードなど世界各社と取引している。ハーネスとは自動車の車体のなかに張りめぐらされる電線であって、IT化が進んだ現代の自動車では、まるで神経系統のように貴重な働きをしている。

自動車のモデルがかわれば、そのハーネスもかわる。車の新モデルの図面は、それぞれの自動車メーカーから、それぞれ別のソフトでヤザキにおくられてくる。ヤザキはそれを変換して、その新モデルの設計図をもとに、ハーネスを設計する。それは当然に各社とのひんばんな相談を要する。たとえば、新設計ではこれほどのハーネスを通すスペースが苦しい、なんとかならないか、などである。ただし、このようにやや狭い範囲のことなら、ソフトの変換でことがすむことも多かろう。それにしてもそのソフトの変換は、ことが専門分野の技術にかかわるだけに、顧客ごとのソフトへのこまかい対応を要しよう。すなわち日本に多い受注型ソフト企業への根強い需要となる。

変化をとり入れるのが遅い

こうしたサービス型ソフト、日本風にいえば受注型ソフトの開発につき、クスマノは日本の弱点を指摘する。それは市場や顧客のニーズの変化をとり入れることが遅い、というのである。米、欧、日、インドの104プロジェクトを比較して、Cusumano [2004] はいう。なるほど日本のソフト開発は品質も生産性も米欧インドより高い。そのかわり市場の変化をとり入れるのが遅く、また少なくなる。それというのも、設計をきちんと前もってチェックして固めてしまうからだ。それで品質不具合すなわちバグがすくなく、プログラマーひとりあたりの生産性が高い。そのおなじ理由が、顧客や市場の変化をソフト開発の後期でとり入れるのをむつかしくしている、という議論を展開するのである。

そこにおそらくは自動車などのばあいと似た、欧米風の観察があるのかもしれない。自動車の最終組立は60秒のくり返し作業の連続である。検査は欧米であれば、ラインの途中ないし最終の検査ステーションがあたる。それとほぼおなじ見方で日本のソフト開発を受けとっているかにおもえる。

だが、日本の自動車の最終組立でも他の工程でも、立ち入って観察すると、異なる情景がみえる。日本では生産ラインの労働者もライン作業のなかで品質不具合を検出し、ときにそれを手直す。それが他国との品質や生産性の差の、真の基盤であった。つまり、日本の方がより後まで調整しているのだ。それとやや似たことがソフトでも見られないのだろうか。その探求の途として、日本が世界をリードするゲームソフトの世界をつぎにみよう。

5. ゲームソフトの世界

みるべき研究論文

なるほどゲームソフト、とりわけ家庭用の分野では、日本企業の活躍は海外でも目立っているようだ。任天堂、セガなどめざましい企業がある。したがって、それらを調べた、みるべき文献が少なくない。なかでも立本 [2003, a, b]、小橋 [1998] [1997]、砂川 [1998] など

が丹念である。いずれも複数の事例に立ち入って、ベテランの話を聞いている。

とりわけ立本「2003 b」は、20社を尋ね、開発の仕事进行深入し、しかもその説明がまことに明晰である。とくにそれが詳しくとりあげた、ふたつの個別プロジェクトの事例分析はすばらしい(立本 [2003 b] pp. 264 - 273)。開発のプロセスをその最末端の労働組織までおり、仕事プロセスの推移のみならず、その職種別の人数、そのプロジェクトの段階をおっての人数の変動まできちんと記す。しかもなお、残念ながら人材形成への観察がとぼしい。人材形成に長期を要するか、それとも短期でよいか、それこそがこのシリーズの関心事なのに、残念ながらその点への吟味が見られない。

もうすこし具体的に説明しよう。人材形成の根幹、個別労働者の仕事経歴すなわちキャリアを観察していないのである。もしまったくの中途採用、経験者尊重ならば、その旨の明記があるとありがたい。だが、案外に内部育成の傾向を中心としているようだ。それならば、関連の深い分野でしだいにより高度な仕事へと移動していく道筋、すなわちキャリアの分析こそ肝要なのに、その吟味がない。そしてその点は、他のていねいな論文にもみられない。

小橋 [1998] は9社の話を聞いている。そして立本 [2003 a, b] とおなじく人材「管理」をあつかう1項をたてながら、人材の形成には観察がおよばない。キャリアをみようとする視点がないのである。だが、この論文はゲームソフト開発の重要な特性を指摘している。それは立本 [2003]、砂川「1998」、またゲームソフトではなくソフト開発一般を対象とした妹尾 [2001] にもほぼ共通するがゆえに、ここでとくに考察しておきたい。

終わりの段階でも修正を

その特性とは、設計の修正あるいは変化を、そのプロジェクトの最終段階に近いところでも、とり入れようとしていることである。それは思いのほか高度なことなのだ。その意味を知るには、ゲームソフト開発にとどまらず、一般のソフト開発の方式をふりかえる必要があ

る。

ごく初期はいわば「天才」的なプログラマーが、すべての仕事を個人技でこなしてきたようだ。だが、しだいに組織化する。そうせざるを得ない。売れゆきがのび企業化すれば、どの国でも当然の帰結である。その仕事の組織化をまず描いた理論モデルが、周知の滝 waterfall 型であった。

Waterfall 型とは、滝の遡上がむつかしいように、修正や変化を遡らせない、という趣旨の命名であろう。ソフト開発で設計をなるべく初期の段階で固定し、変更を少なくする方式をいう。とはいえ、ソフトの開発に変化やトラブルは避けがたい。そこで、ソフト開発の工程を、調査分析、基本設計、詳細設計、プログラミング、テスト、メンテナンスなど数個の段階にわけ、トラブルがおこった直前の段階までさかのぼってのトラブル対策や変更は認める。しかし、それ以上は遡上させない方式をいう。

だが、それでは顧客のニーズや市場の変化などにすばやく対応できない。初期の設計段階以外、市場の変化をとり入れるのはむつかしい。これにたいし、かなりあとの段階まで修正や変化を認める方式が、日本のゲームソフト開発に認められた。それをこれらの論文は強調する。

ゲームソフト開発では、そもそも仕事の段階区分が緩くなっている。基本設計あるいはそのまえの調査分析にあたる段階は、ゲームソフトでは「実験」などとよばれ、2—4人ていどの中核メンバーがアイデアを練る。そしてその後半には小人数のプログラマーなどもいれ、試作品をつくる。この実験段階に時間をかける。半年から1年などといわれる(小橋[1997] p. 87)。

そのあと商品化の作品の製作にとりかかる。当然にかかわる職種や人数もふえ、10—50人くらいの単位となる。その最終段階に近いところでも、些細な変化にとどまらず、かなりの大きな変化や修正までとりこむ。そうした指摘はこれらの論文に、強弱の差はあれ、ほとんど共通する。

柔軟な組織

最終段階に近いところでの修正を入れる理由は、これもまた大半の論文に共通するのだが、ふたつあろう。小橋 [1998] [1997] がもっともよくその点を展開しており、以下おもにそれにしたがって説明していく。

理由の第一は、ゲームソフトの性質である。小橋はいう。家庭用ゲームソフトは顧客と商品の関係が「双方向性」、つまりインターラクティブであるため、顧客の反応を予想しながら修正する。たとえば顧客にとってあまりにむつかしすぎても困る。そうした予想からおもいついた修正(ところによっては「調整」ともいう)をほどこす。

理由の第二は、組織すなわちプロジェクトチームの性質にある。うえにのべた修正は、リーダーが提示しきめる、とはかぎらない。プロジェクトチームのだれでも提案できるし、実際に提案もする。しかもチームはさまざまな職種からなる。音の専門者、絵のデザイナー、もちろんプログラマーもいる。そこで隔意なく相談する。あるいは相談の範囲はもっと広く、社内の別のプロジェクトのスタッフにみてもらい、その意見を聞き修正したりする。こうしたいわば柔軟な組織ゆえに、最終段階に近いところでも修正をいれることができる、というのである。

小橋 [1998] はこの組織の柔軟さを、日本企業の長期の雇用保障と、「年功的」な報酬、すなわち長期の業績におうじた報酬の払い方、という慣行による点が大きいともいう。長期ゆえにおたがいのことをよく知りあう。さらに、あるプロジェクトの売上が大きいからといって、それで報酬の多くがきまるのではない。成功報酬の割合はむしろ小さい。それゆえに隔意ない討議がしやすい、というのである。だが、長期の雇用保障といっても、解雇がないわけではない。企業が衰退し他社に移動せざるを得ないことも当然にある。市場経済のなかにあるのだから、解雇がないわけがない。

それにしても、ここまでの説明はそれなりに納得的である。そしてなにも日本企業にかぎらず、第5章でみるかの米の投資銀行の代表例、ゴールドマンサックスのしくみにも似ている。

そこはかなり長期の雇用慣行があった。そして仕事組織はプロジェクトチームを組み、チームワークがまことに重視されている。

映画の小津方式か、長期方式か

のころは人材形成方式である。ゲームソフト業界の人材形成の実際を解明した研究文献をまだ見出していない。ただし、うえにみた製作グループは、絵、音、ソフトという異種の専門者たちの小集団である。異種の構成チームを自由に動かすには、おそらくつぎのふたつの方策、そしてそれぞれに応じた人材形成方式が考えられる。そのいずれなのか、それともそれぞれに存在し競っているのかどうか。ひとつは映画の小津方式であり、他はうえにみた受注型ソフトの内部昇進方式である。いったいどちらがゲームソフト開発により適しているのだろうか。

映画の小津方式を説明しておく。小津安二郎の映画につき書かれた文献はそれこそ汗牛充棟だが、その製作方式は心して読まない、なかなかわからない。記されてはいるのだ。そしてゲームソフトのベテランも映画界を意識している。映画も異種の専門者たちが一本の映画をつくるためにチームを構成するという点では、ゲームソフト開発とかわらない。そして映画とゲームの違いは、ソフト会社のベテランによれば、最終段階での手入れの大きさの差異にある、という。映画はほんの数%変えるにすぎないのにたいし、ゲームは大きく変えることもできる、というのである（小橋 [1997] p. 92）。

映画の最終の段階に近いところでの修正、変化とは、映画ではおそらく「編集」である。たとえばフィルムのつなぎである。ほんのわずかなコマの切り落としなどなら、まさにゲームソフトのベテランの言うとおりでであろう。だが、あるいは場面を切りだすこともあろう。ゲームソフトほどの大きな変化といえないまでも、かならずしも小さな変化ではない。

そして小津方式とは、その最終調整をふくめ小津安二郎監督すなわちリーダーがほとんど決定した。⁴⁾ チームの自由な議論にゆだねるのではない。そうでないと、音、絵、機械、シナリオという異質の専門者たちをうまく束ね、よい

作品ができない、というのであろう。そしてそれは、すくなくとも小津作品に関しては、だれも否定できない見事な実績の裏付けがあった。

映画方式では職人型が

小津方式を例外とみるわけにはいくまい。映画の全盛時代をとっても、なるほど溝口健二、木下恵介は小津にくらべれば、リーダーの専制のていどは下がるだろう。それでもスタッフと自由に相談して、という対応にはみえない。当時の映画の監督たちがプロジェクトメンバーの自由な発言、提案に大いに耳を傾けた、とはおもえない。小津を例外ときめつけるわけにはいくまい。

この映画の世界では、人材形成はキャリア方式ではない。それぞれの狭い専門分野で、一人前とその弟子—いまなら「アシスタント」とよぶのだろう—の2段階である。照明なら照明屋、音響なら音響屋、カメラマンならカメラマンの分野でおわる。いわゆる職人方式であろう。例外はもちろんある。たとえば小津は、当初は短い間ながらカメラマンであった。助監督のポストがなく、カメラマンから出発した、といわれる。一般にはそこからリーダーはでない。少数ながら案外にみられるのは、脚本の世界から監督へのややほそい途であろうか。

この方式、いわば職人方式でよい作品ができるのは（もちろん駄作もあるが）、小津の例をみるまでもなく否定できない。だが、それなら小橋のいう長期雇用方式とは相いれない。照明も音響もカメラも、会社勤めもあれば独立の稼業もあろう。小津のころはわりと会社勤めが多かったであろうが、当今は独立稼業が多かろう。しかし会社勤めとはいえ、はたして事実上の長期保障があったかどうか。監督もすくなくからず移動した。というより契約制で短期が主流であった。小津はもちろん松竹で通したが、それは小津の名声、実績によるところが大きかった。その小津ですら「契約が切れる」、すなわち解雇のうわさがないではなかった。

他方、長期の慣行があるとしよう。それならば、さきの受注用ソフト開発の観察でみたように、とにかくキャリアを形成する、という方式

が可能となる。ゲームソフト開発業界は実際にはどちらの方式をとったのであろうか。それを解明した研究論文を寡聞にして知らない。

変化を終わり近くにでも入れる

それにしても修正や変化をプロジェクトの最終段階に近いときまで入れるには、長期重視の慣行が有利にはたらくかにおもわれる。プロジェクト仲間の意思疎通がよりスムーズになり、いわば阿吽の呼吸が異職種間のひとたちにも認められよう。それゆえに自由な提案、討議が行われやすい。

ただし、そうした性質、特徴はゲームソフト開発などといういわば創造的な分野にかぎられるのであろうか。その特徴は、話が一挙にとぶかにみえようが、非芸術的、非創造的とみなされている分野にも、ていどの差はあれ認められよう。第4章でみる自動車製造業をとってみる。職場にすぐれた中堅層が厚く育つ。その人たちが製造の最終段階でも品質不具合などを検出し直している。最初の設計にも発言をしているのみならず、それにつづく生産ラインの設計にも発言する。さらに終わりに近い段階でも、ことは小さくとも大いに参加し、手直ししている。最初の設計さえ固めれば、あとは機械の作業にまかせる、あるいは人の作業も機械的に、という考えではそうはいかない。

その点はわたくしのささやかな観察からもいえる。ある自動車製造職場での聞きとりであった。わりと頻繁におこなわれる問題解決の「会議」の模様を知りたかった。それを話し手につたえると、今日4時からそれがある。見学したらどうかと誘われた。もちろん訪ねた。生産ラインのすぐそばで、すなわち会議室ではまったくなく、20名ほどが立ったまま議論していた。前回指摘された問題点への対策処置、その結果、さらに要請される対策の説明などがつづく。そこには技術者はもちろん生産労働者のベテランも保全も、多くの関係者が参加していた。まことに手短ながら自由な討議であった。

なるほど製造業では変化を後の段階でとり入れるのは、ゲームソフトにくらべはるかに小さく、かつ少ないであろう。だが、同じ製造業、

同じ自動車産業をとる。そして日米でくらべてみる。そうしたら、多分、日本企業の方が米企業より、より後の段階でも変化や修正をより多くとり入れているのではないだろうか。

一般化すれば、柔らかな組織がより自由な討議を促し広げる。そして、そういう意味では、多分日本職場の方が民意、といってももちろん技能の高い人たちの民意だが、それをいかす度合いが、他国よりもやや大きいかもしれない。なんの統計資料もないけれど、そう考えると、説明できる事柄が広がる。ソフトウェア業の分析は、案外に多くの産業に共通する発見を確かめた。

注：

- 1) 中村 [1990]、小池 [1991] などである。ほかに1990年代末から2000年はじめにかけ、いくつかの研究があらわれた。とりわけ妹尾 [2001] がある。その唱道する「状況論的リーダーシップ」ははなはだ魅力的である。だが、その人材形成については、「生活世界」を広げるという抽象的ないい方にとどまり、その意味がなおはっきりしない。残念ながらキャリアの探求はまことにとぼしい。いわゆる創造的な仕事では、「生活世界」やそれに類する言葉が用いられるが、技術なしには無理であろう。それならば技術の形成方式の解明がのぞまれる。プラスアルファとしての「生活世界」であろうが、重視するならば説明がほしい。
- 2) これは小池 [1991] には明記していないが、労働省職業能力開発局委託の仕事であった。当時、労働省が職業能力開発を、それまでの生産労働者中心から大卒ホワイトカラーに広げようと考えていた。そのために企てられた調査委員会であった。わたくしはそのころ中央職業能力開発審議会の委員であり、この調査委員会の主査をつとめた。調査委員会には当時のさまざまな産業の大手企業の人事スタッフが参加し、委員の各社に大卒10年者と20年者それぞれ20人ほどの人事情報の提示をもとめた。当時の霞が関の威力か、多くの企業がこれに答えた。この人事情報をもとに、調査担当の研究者たちが事例の聞きとりもおこなった。わたくしも若い研究者たちと手分けし、数十社をたずねた。わたくしはおもに営業中心の事例を担当したが、ソフトウェア企業も3社まわった。
- 3) こうした仕事の流れはかなり一般的であった。わたくしのみるかぎり、当時の実際は、中村圭介 [1990] のていねいな観察結果とほとんど一致する

(pp. 20 - 24, 50 - 60)。なお、ソフトの仕事の内容については、管見のかぎりでは、断然立本 [2003 a] がくわしい。詳細設計など各段階の仕事内容をこれほど明確に記した文章を他に知らない。ただし、ゲーム産業の分析が主眼であるために、最初の段階、「調査分析」のところが、ふつうの受注ソフト業界の「要求分析」の説明とは大分異なる。「要求分析」は、まさに注文主の企業の仕事の流れの解析こそが肝要であり、それに注文主の、本人もまだはっきりしない意向を鮮明にする、という提案作業である。それがゲーム業界ではおもに内製であるために鮮明にでてこない。

- 4) すべてを小津がきめた、とはかぎらない。有名な例は「東京物語」の風景実写の編集の話である。そのひとコマを編集担当者、浜村が小津に聞かずにひとコマを切った。小津がその試写をみて、ひとコマ切ったね、と浜村に聞いた。いけませんか、と浜村が答えると、いやかえってよくなったから、と小津が答えたという。これは風景実写でもひとコマの切り落としを見逃さない、小津の厳密さの証として引かれるが、参加者たちの多少とも自由な貢献があったことをもしめそう (高橋 [1982] pp. 50 - 51)。

文献：

- 生稲史彦 [2003] “ソフト開発の内製・外製とパフォーマンス”、新宅純二郎、田中辰雄、柳川範之「ゲーム産業の経済分析」東洋経済、pp. 207 - 233
- 板倉 稔 [1993] 「スーパーSE」日科技連、224p.
- 経済産業省 [2012] 「平成22年特定サービス産業実態調査報告書 ソフトウエア業、情報・提供サービス業、インターネット付随サービス業編」253p. 経済産業省
- 経済産業省 [2004] 「我が国情報処理の現状 平成14年情報処理実態調査」289p. 経済産業省
- 小池和男編著 [1991] 「大卒ホワイトカラーの人材開発」東洋経済、312p.
- 小池和男 [1993] 「アメリカのホワイトカラー」東洋経済、181p.
- 小橋麗華 [1998 a] “日本のゲームソフト会社の人材マネジメント”、「国際研究論叢」大阪国際大学、12 - 4, pp. 1 - 22
- 同 [1998 b] “ソフトのイノベーションー任天堂のデファクトスタンダード形成とソフト開発”、伊丹敬之他「日本企業の経営活動」第3巻、有斐閣、所収、pp. 334 - 360
- 同 [1997] “日本における家庭用テレビゲームソフトウエアの開発”、「国際研究論叢」10 - 3/4, pp. 81 - 107

- 砂川和範 [1998] “日本ゲーム産業に見る企業者活動の継起と技術戦略ーセガとナムコにおけるソフトウェア開発組織の形成”、「経営史学」32 - 4, pp. 1 - 22
- 妹尾 大 [2001] “ソフトウェア開発の新潮流ー状況論的リーダーシップの胎動”、「組織科学」35 - 2, pp. 65 - 80
- 高橋治 [1982] 「絢爛たる影絵ー小津安二郎」文芸春秋、317p.
- 立本博文 [2003 a] “ゲームソフトとソフトウェア開発プロセス”、新宅純二郎、田中辰雄、柳川範之「ゲーム産業の経済分析」東洋経済、所収、pp. 235 - 257.
- 同 [2003 b] “製品タイプと開発プロセスの適合性”、同上書所収、pp. 259 - 288
- 中島聡 [2011] 「エンジニアとしての生き方 IT技術者たちよ、世界へでよう」インプレスジャパン、271p.
- 中村圭介 [1990] “ソフトウェア産業の概要と分業構造”、“ソフトウェアの開発と生産管理”、戸塚秀夫、中村圭介、梅沢隆「日本のソフトウェア産業」東京大学出版会、pp. 11 - 89、所収.
- 日本情報システム・ユーザー協会 J U A S、経済産業省情報処理振興課「2009年度版 企業IT動向調査2010」298p. 日本情報システム・ユーザー協会
- 同、経済産業省情報処理振興課 [2011] 「ユーザー企業 ソフトウエアメトリックス調査2011」278p. 日本情報システム・ユーザー協会
- 同 [2011] 「IS人材のためのキャリア形成のヒント」第1巻「傾向調査編」33p.、第2巻「プロフェッショナル紹介編」、第3巻、39p. 「ISプロフェッショナルからのメッセージ編」、第4巻「組織戦略編」91p.
- 延岡健太郎 [2006] 「MOT “技術経営” 入門」日本経済新聞出版社、325p.
- 藤田英樹、生稲史彦 [2010] “ソフトウェア産業における経営スタイルの革新ーカスタム・システム開発を支える人事システム”、尾高煌之助、松島茂、連合総研「イノベーションの創出ーもの造りを支える人材と組織」有斐閣、所収、pp. 205 - 233.
- 三輪卓己 [2001] 「ソフトウェア技術者のキャリア・デベロップメントー成長プロセスの学習と行動」中央経済社、294p.
- 山田隆夫 [2009] 「SEの基本」日本実業出版社、267p.
- 加藤忠宏 [1999] 「提案型システムコンサルタント養成講座」同友館、246p.
- Cusumano, Michael A. [2004] *The Business of*

Software: What Every Manager, Programmer, and Entrepreneur Must Know to Thrive and Survive in Good Times and Bad, Free Press, サイコムインターナショナル監訳「ソフトウェア企業の競争戦略」ダイヤモンド社、2004年

Cusumano, Michael, & David Yoffie, [1998] *Competing on Internet Time*, Free Press, 松浦秀明訳「食うか食われるか ネットスケープ vs. マクロソフト」毎日新聞社、1999年、271p.

Cusumano, Michael A. & Richard Selby [1995] *Microsoft Secretes : How the World's Most Powerful Company Creates Technology, Shapes Markets, and Manages People*, the Free Press, 512p. 山岡洋一訳「マイクロソフト・シークレット」上 360p. 下 306p. 1996、日経、

McConnel, Steve[2004] *Professional Software Development: Shorter Schedule, Higher Quality Products, More Successful Projects, Enhanced Careers, Persons Education*, 松原友夫、山浦恒央訳「ソフトウェア開発プロフェッショナル」日経B P社、275p.

Perry, Paul & Keith Ermel, and James Shields, [1994] *Insider Guide to Software Development, Que Corporation*, 藤木祐二訳「ソフトウェア開発インサイダーズ ガイド」海文堂、1995、276p.

Zackary, Pascal G. [1994] *Showstopper !*, Free Press, 312p. 山岡洋一訳「闘うプログラマー」日経B P社、上、下

以上