



Newsletter from Software Engineers Association

Volume 8, Number **7** December, 1993

---

## 目 次

|                                     |          |
|-------------------------------------|----------|
| 編集部から                               | 1        |
| アイスランドでのソフトウェア品質と<br>生産性国際会議に参加して   | 新森 昭宏 2  |
| ビジネス プロセス リエンジニアリングと<br>教育システム開発方法論 | 君島 浩 7   |
| SEA Seminar & Forum September '93   | 堀 真人 12  |
| SE 解放宣言                             | 臼井 義美 14 |
| 情報処理産業のこれから                         | 横山 博司 18 |
| SEA のリストラクチャリングについて                 | 坂本 啓司 20 |
| リストラとウルグアイ・ラウンド雑感                   | 大場 充 21  |
| パズルとコンピュータ                          | 勝田 祐輔 25 |

---





## ソフトウェア技術者協会 Software Engineers Association

ソフトウェア技術者協会 (SEA) は、ソフトウェアハウス、コンピュータメーカ、計算センタ、エンドユーザ、大学、研究所など、それぞれ異なった環境に置かれているソフトウェア技術者または研究者が、そうした社会組織の壁を越えて、各自の経験や技術を自由に交流しあうための「場」として、1985年12月に設立されました。

その主な活動は、機関誌 SEAMAIL の発行、支部および研究分科会の運営、セミナー／ワークショップ／シンポジウムなどのイベントの開催、および内外の関係諸団体との交流です。発足当初約 200 人にすぎなかった会員数もその後飛躍的に増加し、現在、北は北海道から南は沖縄まで、900 余名を越えるメンバーを擁するにいたりました。法人賛助会員も 50 社を数えます。支部は、東京以外に、関西、横浜、長野、名古屋、九州、東北の各地区で設立されており、その他の地域でも設立準備をしています。分科会は、東京、関西、名古屋で、それぞれいくつかが活動しており、その他の支部でも、月例会やフォーラムが定期的に開催されています。

「現在のソフトウェア界における最大の課題は、技術移転の促進である」といわれています。これまでわが国には、そのための適切な社会的メカニズムが欠けていたように思われます。SEA は、そうした欠落を補うべく、これからますます活発な活動を展開して行きたいと考えています。いままではなかったこの新しいプロフェッショナル・ソサイエティの発展のために、ぜひとも、あなたのお力を貸してください。

代表幹事： 中野秀男

常任幹事： 岸田孝一 熊谷章 玉井哲雄 深瀬弘恭 堀江進 山崎利治

幹事： 浅井美枝子 市川寛 伊藤昌夫 白井義美 大塚理恵 大場充 菊地俊彰 君島浩 窪田芳夫 小林俊明  
坂本啓司 杉田義明 武田淳男 田中一夫 鳥居宏次 中來田秀樹 中谷多哉子 西武進 野村敏次 野村行憲 平尾一浩 平山伸一 藤野見延  
二木厚吉 松原友夫 山崎朝昭 渡邊雄一

会計監事： 辻淳二 吉村成弘

分科会世話人 環境分科会 (SIGENV)：田中慎一郎 渡邊雄一  
管理分科会 (SIGMAN)：野々下幸治  
教育分科会 (SIGEDU)：杉田義明 中園順三  
ネットワーク分科会 (SIGNET)：大塚理恵 小林俊明 人見庸  
調査分科会 (SIGSURVEY)：岸田孝一 野村敏次

支部世話人 関西支部：白井義美 中野秀男 盛田政敏  
横浜支部：藤野見延 北條正顕 野中哲 松下和隆  
長野支部：市川寛 佐藤千明  
名古屋支部：浅井美枝子 鈴木智 平田淳史  
九州支部：平尾一浩  
東北支部：菊地俊彰 和田勇

賛助会員会社：NTTソフトウェア研究所 NTT九州技術開発センタ PFU SRA アスキー エイ・エス・ティ  
エスケイ・ディ オムロンソフトウェア カシオ計算機 キヤノン新川崎事業所 さくらケーシーエス  
サン・ビルド印刷 ジューエムエーシステムズ ジャストシステム  
セントラル・コンピュータ・サービス ソフトウェアコントロール ダイキン工業 テクノバ  
ニコシシステム ニッセイコンピュータ ムラタシステム リコーシステム開発  
リパティシステム 安川電機 古河インフォメーション・テクノロジー 構造計画研究所  
三菱電機セミコンダクタソフトウェア 三菱電機メカトロニクスソフトウェア 三菱電機関西コンピュータシステム  
新日鉄情報通信システム 新日本製鉄エレクトロニクス研究所 池上通信機 中央システム  
辻システム計画事務所 東芝アドバンスドシステム 東電ソフトウェア 東北コンピュータ・サービス  
SRA東北 日本NCD 日本ユニシス・ソフトウェア 日本情報システムサービス  
日本電気ソフトウェア 日立エンジニアリング 富士ゼロックス情報システム 富士写真フイルム 富士通  
富士通エフ・アイ・ピー オムロン (以上 48 社)

SEAMAIL Vol. 8, No. 7 1994年1月10日発行

編集人 岸田孝一

発行人 ソフトウェア技術者協会 (SEA)

〒160 東京都新宿区四谷3-12 丸正ビル5F

TEL: 03-3356-1077 FAX: 03-3356-1072

印刷所 サンビルド印刷株式会社 〒162 東京都新宿区築地町8番地

定価 500円 (禁無断転載)



## 編集部から

☆

この雑誌がみなさんのお手もとにとどくのは年が明けてからでしょうが、編集作業の追い込みをしているいまは、まだ12月半ば。1993年も不景気のまま終わろうとしています。

☆☆

前号の編集前記で、もう原稿の在庫が底をついた! と書いたのですが、その後たてつづけにみなさんからの投稿が e-mail & surface mail で到着し、なんとか毎月1冊ずつの編集ノルマを果たすことができました。

☆☆☆

次号には、北陸先端科学技術大学院大学 (JAIST) の落水先生の研究室から寄せられたソフトウェア分析/設計に関する2篇の力作テクニカル・レポートをお送りする予定です。御期待ください。

☆☆☆☆

ということで、いちおう1994年1月分までの編集のめどは立っているのですが、それから先は?

☆☆☆☆☆

ひきつづき、みなさんの御協力をよろしく!

☆☆☆☆☆☆

## アイスランドでのソフトウェア品質と生産性国際会議に参加して

新森昭宏

インテック・システム研究所

### 1 はじめに

思いがけず、アイスランド大学で開催された国際会議（ソフトウェア品質と生産性国際会議）でチュートリアルと講演をするようにとの招待を受けた。ことの次第の詳細は省略するが、2年前に当社からヨーロッパソフトウェア品質学会（European Conference on Software Quality）に応募した論文（インテックにおけるデザインレビューの取り組みについての論文）がアイスランド大学計算機科学科のOddur Benediktsson 教授の目にとまったということである。

会議内容について、客観的に評価しコメントすることは現在の私の手には余るが、SEAMAILの読者にとって有用な情報も含まれていると考えるのでここにサマリーを報告する。

### 2 国際会議の概要

ソフトウェア品質と生産性国際会議（Conference on Quality and Productivity in Software Development）は、アイスランド大学計算機科学科のBenediktsson 教授を中心とした実行委員会の呼びかけにより、1993年9月1日から3日間にわたってアイスランド大学で開催された。

会議の課題としては、ソフトウェアプロセスと生産性、ISO9000 認証、ソフトウェア品質管理、ソフトウェア保守における品質管理、プロセス評価と改善、オブジェクト指向技術が掲げられ、講義（チュートリアル）、講演、討論の3部から構成された。

参加者は、地元アイスランドはもとより、ノルウェー、スウェーデン、デンマーク、フィンランド、ドイツ、スイス、イタリア、カナダ、アメリカなど、16ヶ国から104名に達していた。

講演者として、ソフトウェアコンサルタントとして著名なTom Gilb氏やカーネギーメロン大学ソフトウェア工学研究所のMike Konrad氏なども招かれていた。日本からは、私の他に（株）コンピュータアプリケーションズの萩原氏が参加していた。

会議の開会時には、アイスランド産業省からの歓迎の挨拶があった他、レイキャビク市の市長による歓迎会も開催された。

### 3 アイスランドという国

アイスランドという国は、普通の日本人にとってそれほどなじみ深い国ではないであろう。「アイスランドに行く」と回りの人に言うと、「たしか、イギリスのとなりにあったよね」（それは、アイルランドだ！）などという答えが返ってきたりする。

アイスランドは、アメリカとヨーロッパの間に横たわる大西洋の北側に位置する小さな島国である。首都は、1986年にレーガン・ゴルバチョフによる米ソ首脳会談が開かれて以来有名になったレイキャビク。人口は、国全体で26万人程度であり、そのうち70%以上がレイキャビク近辺に住んでいるという。



周囲の豊かな漁場を生かして、漁業が主要産業となっている。また、火山国であることを利用した地熱発電も盛んであり、小国ながら豊かな国である。地下から汲み上げた温水を使って地域暖房のシステムが整えられている。他、屋外の温水プールや温泉もたくさん整備されている。(Benediktsson 教授によれば)「加盟するメリットがない」ので、EC(ヨーロッパ共同体)には現在のところ加盟していないが、EFTA(ヨーロッパ自由貿易連合)には加盟している。

歴史的には、北欧で勇名をはせていたバイキングの流れを引き、スウェーデン、ノルウェー、フィンランドなどのスカンジナビア諸国とルーツを同じくするノルディック民族の国である。島国であるため、バイキングの文化と伝統が純粋な形で残っているともいう。

#### 4 会議の内容

##### 4.1 私の発表

私は、“Effective Design Reviews for Software Quality”と題した講義(半日)と“Quality Control and Software Production in Japan”と題した講演(1時間)を行った。講義には、(他のセッションと並列であったため)40名程度の参加者があり、講演は全参加者を対象として行った。

講義は、私が当社の河合清博生産技術部長と共同で著した同名論文をベースとして、それにその後の成果を追加する形で行った。すなわち、当社におけるソフトウェア品質保証の基本的考え方から始まって、具体的取り組み(デザインレビューとクレーム管理、メトリクスの設定、データ収集・解析・プロセス向上サイクルなど)までを説明した。

特に、生産性指標、レビュー実施率、レビュー寄与度、内容・表現指摘率などのメトリクスの設定とデータ収集・解析・プロセス向上サイクルについては、参加者の関心を集め、いくつかの質問が寄せられた。

##### 4.2 Software Inspection についての Tom Gilb 氏の講義

Tom Gilb 氏は、ソフトウェアコンサルタントとして15年以上の実績を持ち、“Principles of Software Engineering Management”(1988年)、“Software Inspection”(近刊)、“Idea Management”(近刊)などの著書がある。Tom Gilb 氏のクライアントには、IBM、AT&T、HP、US Army などが含まれる。もともと、ロスアンジェルス生まれだそうであるが、現在はノルウェーのオスロに自宅があり、事務所はロンドンにおいて各国のクライアントを訪問するという文字どおり国際的な活動をしているそうである。品質管理の大家である W. Edwards Deming 博士や、元カーネギーメロン大学の Watts Humphrey 氏とも親交があるそうである。

Tom Gilb 氏の Software Inspection 論は、もともと IBM の Fagan 氏によって20年前に提唱されたものをベースとして、各種のコンサルテーション経験を踏まえて改良を加えてきたものであるという。

Tom Gilb 氏の唱える Software Inspection は、我々のデザインレビューに比べるとより厳密に定義されたフォーマルな品質保証技術である。すなわち、まず開発方法論をルール化し、Inspection はそのルールに基づいて厳密に行い、ルールに違反したものを欠陥(defect)またはバグとして指摘する。Inspection プロセスの開始にあたっては、開始条件をクリアすることが必要であり、その終了にあたっては、終了条件をクリアすることを求める。Inspection 時には、デザインそのものの検討はあえて回避し、対象物をルールと前工程成果物との整合性から観点からチェックし、あくまでも重大な欠陥(major defect)の発見に集中する。これにより、Inspection の効率化と開発者の生産性向上をはかることができるという。

Tom Gilb 氏は、Inspection のための実施プラン、議事録、各種のルールなどのサンプルを提示した。また、



実際に Software Inspection によって生産性向上を達成した IBM や HP などの事例を紹介した。それによれば、Software Inspection の実施により、設計、開発までの段階で高信頼性を達成することができ、単体テスト、結合テストを省略できるようになるとのことである。

Tom Gilb 氏の Software Inspection は、協調を重んじる日本的なデザインレビューとは対照的なフォーマルな方法論であるが、Tom Gilb 氏は、Inspection を開発者にとって楽しいものにできるし、そうしなければならないと述べていた。また、チェックリストの使用については、「チェックリストは単なるルールの裏返しであってはならない」、「チェックリストは英知を集めたものでなければならない」との警句を発していた。

#### 4.3 成熟モデルについての Mike Konrad 氏の講演

カーネギーメロン大学ソフトウェア工学研究所(CMU SEI: Carnegie Mellon University, Software Engineering Institute) は、ソフトウェア工学の研究推進とソフトウェアを利用したシステムの品質向上を目的としてアメリカ政府によって1984年に設立された研究機関である。成熟モデル(CMM: Capability Maturity Model)とは、ソフトウェア開発組織のプロセス評価を行うために、1987年にSEIのWatts Humphrey氏らによって提唱されたモデルである。

Konrad氏は現在、CMMに基づいたコンサルテーションとCMMの改良・拡張作業を進めている。今回の講演では、CMMの概要説明とその改良・拡張作業の現状、ソフトウェアプロセス評価の国際標準化プロジェクトSPICE(後述)との関わりなどについて説明があった。

CMMは、プロセス管理と品質管理の考え方をベースとして、ソフトウェア開発組織の成熟度を評価するもので、その段階に応じて5段階の尺度を設定している。5段階とは、レベル1(初期状態:initial)、レベル2(継続可能状態:repeatable)、レベル3(定義状態:defined)、レベル4(管理状態:managed)、レベル5(最適化状態:optimized)である。5段階のそれぞれの段階において、より高位の段階へ到達するための具体的目標と方策が定義されている。

CMMに基づくアメリカ国内での評価結果によれば、評価対象となった組織(国防関係の受注業者など)の74%が初期状態のレベルにあるとのことである。

CMMに対しては、大規模組織を対象としている、アメリカ国防省の受注業者評価という色彩が強いなどの批判があるが、BOOTSTRAPやTRILLUMなど(後述)他のソフトウェアプロセス評価モデルのベースとなっている。現在も、小規模組織の評価や評価手法の具体化などの観点からモデルの改良・拡張が進められており、本年2月にCMM V1.1がリリースされ、さ来年以降にはCMM V2がリリースされる予定であるという。

また、ISO9000認証との関連については、一部の項目がレベル3に相当することを除いて、ISO9000認証はほぼレベル2に相当するとのことである。<sup>1</sup>

#### 4.4 ソフトウェアプロセス評価の国際標準化プロジェクト SPICE について

ダブリン市立大学のJoc Sanders氏から、ソフトウェアプロセス評価の国際標準化プロジェクトSPICEについての説明があった。

SPICE(Software Process Improvement and Capability dEtermination)とは、CMMをはじめとしたいくつかのソフトウェアプロセス評価モデル構築の経験をベースとして新しい国際標準をつくるためのプロジェクトである。ベースとなっているソフトウェアプロセス評価モデルは、以下のようなものである。

<sup>1</sup>CMMとISO9000との関係については、最近のIEEE Computer誌にも比較論が掲載されている[2]。



- CMU SEI による CMM
- HP 社のソフトウェア品質・生産性分析
- Compita 社 (スコットランド) の STD (Software Technology Diagnostic)
- British Telecom のソフトウェア評価方法論
- ESPRIT II プロジェクトにおける BOOTSTRAP
- Bell Canada, BNR, Northern Telecom による TRILLIUM

CMM やその他のモデルでの経験を踏まえて、5 段階ではなく、6 段階のモデルとすることが検討されている。また、ISO9000 シリーズ、特に ISO9001, ISO9000-3, ISO9004 との整合性にも留意しているとのことである。

SPICE プロジェクトは、ISO/IEC JTC1/SC7/WG10 と連携を取っており、1994 年 6 月にドラフト作成を行い、1996 年 9 月に JTC1 の Technical Report とし、それ以降に国際標準とすることを目標としているとのことである。作成中のドキュメント類は、ネットワークでオンライン入手できるように (具体的には、Internet の anonymous FTP サーバから入手できるように) セットアップされているとのことである。

#### 4.5 Bell Canada, BNR, Northern Telecom における通信ソフトウェア開発プロセス評価モデル Trillium について

Bell Canada の Jean-Normand Drouin 氏が、通信ソフトウェア開発プロセスモデル Trillium について講演した。Trillium とは、Bell Canada, BNR (Bell Northern Research), Northern Telecom の 3 社が、通信ソフトウェアの調達業者のプロセス評価のために CMM をベースとして開発した評価モデルである。Trillium には、そのほか ISO9000 や、アメリカにおける品質管理賞である Malcom Baldrige 賞の審査項目、IEC300, IEEE 標準, Bellcore 内部標準なども取り込まれている。

Trillium の特徴は、品質管理、人的資源、開発環境などの項目ごとに評価を行うことであり、それによって評価対象である調達業者にとって次に克服すべき問題点・達成すべき目標を明らかにすることであるという。

#### 4.6 Esprit プロジェクトの一環として開発された BOOTSTRAP 評価方法論について

Oulu 大学 (フィンランド) の Pasi Kuvaja 氏と ETNOTEAM (イタリア) の Adriana Bicego 氏が、Esprit プロジェクトの一環として開発された BOOTSTRAP 評価方法論について講演した。BOOTSTRAP も、CMM をベースとした方法論であり、Trillium と同様に ISO9000 を取り込んでいる。

BOOTSTRAP については、IEEE Software 誌の 1993 年 5 月号にも紹介記事が掲載されている [1]。

#### 4.7 オブジェクトモデル技術によるソフトウェアプロセスのモデル化について

アイスランド大学の Benediktsson 教授が、Rumbaugh らによって提唱されたオブジェクトモデル技術 (OMT: Object Modelling Technique) とそれによる ISO9000-3 のソフトウェア保守プロセスの記述実験の紹介を行った。

OMT は、クラス間の関係を表すオブジェクトモデル、クラスの時間的振る舞いを表すダイナミックモデル、データの流れを表すファンクショナルモデルからなる強力なモデリング技術である。Benediktsson 教授は、この OMT を使って ISO9000-3 に記述されたソフトウェア保守の概念の記述実験を行った。現時点では、まだ実験レベルであるが、より詳細な検討を行うことにより、ISO9000-3 自身の分析、モデル記述に使用できるのではないかとのことである。



## 5 会議全般について

今回の会議は、ワークショップといってもいいほど小規模のものであり、ICSE(International Conference on Software Engineering)などの国際会議には規模という面ではとうてい及ばないものであった。それでも、参加者同士が十分に交流できるという小規模会議の良さを感じさせてくれた会議であった。実際、私は個人的にパーティの場で Tom Gilb 氏と Software Inspection やデザインレビューについてじっくり話すこともできた。

SEA でも最近ホットな話題となっている ISO9000 については、EC 統合という背景から生まれた標準ということもあってか、ヨーロッパにおける関心の高さを感じた。会議の場でも、認証取得のメリットや効果、標準自体の妥当性などをめぐって、いろいろの意見が交換された。たとえば、20 人とか 30 人とかいった小規模のソフトウェア会社が多い地元アイスランドからの参加者からは、経済性の観点から ISO9000 認証取得の困難さが語られていた。既に ISO9000 認証を取得した IBM Germany の Wolfgang Dette 氏<sup>2</sup>からは、品質保証という観点からみて、ISO9000 に準拠するための努力を行ったことはそれほど効果を持たなかった、むしろ TQM(Total Quality Management) の仕組みを整えてきたことに意義があったとの指摘があった。

## 参考文献

- [1] Bootstrap Project Team: "Bootstrap: Europe's Assessment Method," IEEE Software, May 1993
- [2] R.C.Bamford and W.J.Deibler: "Comparing, contrasting ISO9001 and the SEI capability maturity model," IEEE Computer, October 1993

---

<sup>2</sup>IBM Germany の取り組みについては、SEAMAIL の Vol.8, No.4 (September, 1993) に掲載されていた John Harauz 氏の論文でも引用されている

# ビジネス プロセス リエンジニアリングと 教育システム開発方法論

君島 浩

(富士通)

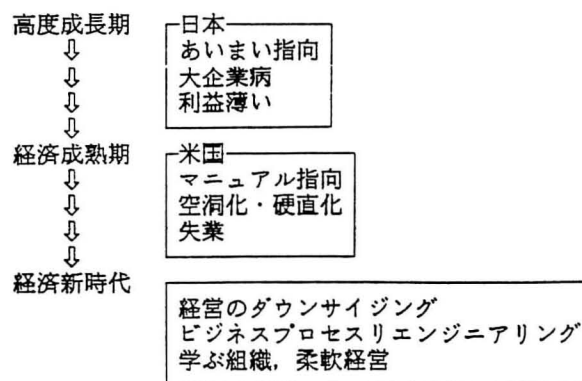
ビジネスプロセスリエンジニアリング(BPR)を解説するある資料に、教育システム開発方法論(ISD)が登場していた。BPRもISDも、一部の人間が関心を示しているだけで、経営者も、ソフトウェア技術者も、教育担当者もあまり注目していない。本稿では、BPRとISDとの関係を、自分なりに整理してみる。

## 1. 経営の動向

日本の経営学は、高度成長期にのみ通用した時代遅れの不完全なものである。財務は借金を前提にした利益の薄いものである。売上高は増大しても、社員の給料は上がらない。仕事の指令や評価はいい加減であり、特にホワイトカラーの仕事は生産性が低い。仕事のあいまいさは、経営を柔軟にするのには役立ったが、大企業を意識的にコントロールすることは難しい。硬直化とは別の意味での大企業病をもたらした。社員の責任があいまいなために、やるべきことをやらなくても済むという大企業病である。

米国の経営学は、早くから経済成熟期に入ったため、日本よりも進んでいる。仕事の命令、評価が明確であり、IBM、ディズニールランド、マクドナルドハンバーガーなどの優良企業が生まれた。しかし、財務重視のための空洞化、職務分掌による硬直化、社会の階層化による失業などを生んだ。社会の階層化とは、人種差別を極度に回避するために、教育程度を下げたことであり、標準英語を読み書きできない階層が、失業に苦しんでいる。例えば、米国では製品取扱説明書を書くテクニカルライタは、標準英語を読み書きできるアングロサクソンだけである。

米国は経済新時代に突入しようとして、経営のダウンサイジング、ビジネスプロセスリエンジニアリング、学ぶ組織、柔軟経営などの新方式を導入しつつある。米国より遅れている日本は、米国の経済成熟期の経営学とともに、経済新時代の経営学を急速に導入する必要がある。しかし、その企業変革への取り組み意欲は、極めて低い。



いまは経済成熟期を迎えて不況気味になっている。しかし、これからの経済新時代は暗いものではない。市場にも社内にも、まだまだ未開拓部分があるからである。この部分を開拓すれば、高度成長期よりもさらに幸福な時代を迎えることが可能である。日本の流通業を指導してきた渥美俊一は、これを「商品の真空地帯」、「作業の真空地帯」と名づけている。

たとえば、日本における高価な霜降り牛肉は生産者が勝手に「市場ニーズ」を捏造したものに過ぎない。米国の顧客は、多少固くてもよいから赤身で旨みのある安い牛肉を享受している。それを日本ではだれも商売にしようとはしない。乗用車もどのメーカーも似たりよったりであり、実用本位の超経済車や1800cc以下の高級乗用車はどこも作らない。

渥美氏は流通業におけるパート、店員、主任、店長、本社スタッフの仕事を、明確な業務マニュアルと命令体系によって、少しもムダのないものにした。それに比べると、日本の一流企業の仕事はムダだらけである。いくらでも生産性を向上する余地がある。

ダイエーやセゾングループの経営者を指導してきた渥美氏が経営学界で注目されないのは不思議である。かれは今後の新経済時代にも通用する。すぐれた経営モデルを提示する人物として尊重されるべきだろう。



## 2. 経済新時代へ向けて

経済新時代へ向けての施策は明確である。

### 1) 米国の経済成熟期に追いつく

業務マニュアル

明確な命令 2) 米国の欠点を回避する

硬直化に対する柔軟な作業

失業を生み出す階層化の回避。弱者の就職推進。

### 3) 米国と共に今後の方向を決める

経営のダウンサイジング (空洞化の回避)

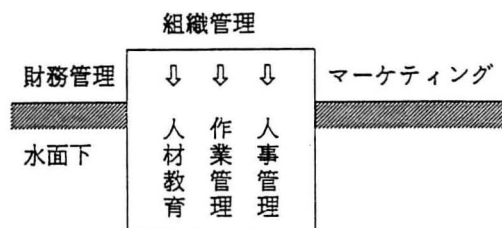
BPR

学ぶ組織。

### 2.1. 経営のダウンサイジング

日本でも米国でも、経営者は事業展開に集中して、そのためのマーケティング、組織管理、財務管理に重点を置いてきた。これが仕事の第一線から視点が離れ、社員の幸福を忘れた原因である。会社をブラックボックスと見立てて、株主と変わらないようなコントロールをしてきたのである。

今後の経営者は、会社の実質的規模、あるいは心理的規模を小さくして、仕事の第一線をコントロールしなければならない。渥美氏は流通業の経営において、作業管理、人事管理、人材教育を柱から外さないようにしてきた。「経営の基本はカネではなく、仕事である」。



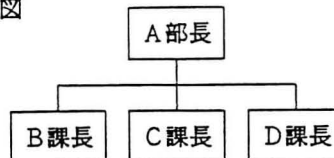
作業管理は、製造業の生産管理で多大な研究・実績の成果があっただけで、その後の研究が進んでいない。これからは作業管理の研究・実践に力を入れなければならない。

### 2.2. ビジネスプロセスリエンジニアリング

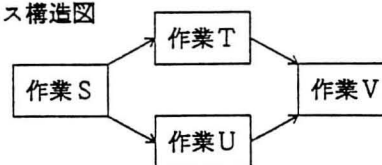
仕事の構造は、階層構造とネットワーク構造(プロセス構造)とでとらえることができる。日本で組織図を描くと、階層構造図を描く。階層構造を変えても、

管理職の担当範囲が変わるだけで、第一線の仕事は変わらない。第一線の仕事における入力・出力という結線を変えようというのが、プロセス構造の変革である。プロセス構造が硬直化してムダがある場合、階層構造をいくら変えても意味はないということである。

階層構造図



プロセス構造図



### 2.3. 階層化の回避

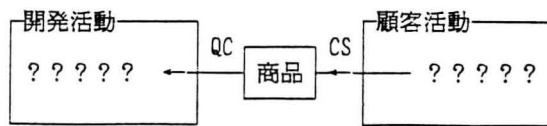
米国も欧州もいろいろな形で階層社会を生み、生まれながらの差別・失業が発生している。日本は比較的平等な社会であるが、富めるものが受験投資をして有名大学に進学する。会社では OJT が主体であるために、生まれつき飲み込みのよい社員だけが昇格昇給し、そうでない人は脱落していく。このことは本人のためにも、会社の収益のためにも、国力のためにも避けなければならない。

年功序列制が崩壊して実力給制に移行しつつある。実力給制を実施するには、教育の機会均等を厳守する必要がある。OJT 主体のあいまいな教育ではいけない。後で述べるような作業管理と ISD の推進は、明確な命令と明確な業務マニュアルを確立するので、教育の機会均等の意味で弱者を救うことになるだろう。底上げがなされることは、トップ層の能力発揮にもよい影響を与えるはずである。

### 3. プロダクト指向からプロセス指向へ

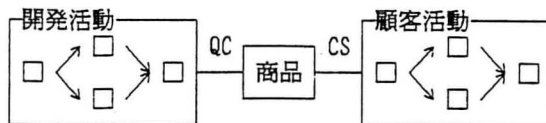
ソフトウェアの世界では、顧客満足度 (CS) やオブジェクト指向が注目を浴びている。しかし、これらの静物指向の考え方と並行して、プロセス指向の考え方を開拓する必要がある。品質向上が低迷しているのは、開発プロセスへのメスの入れ方が不足しているからである。

## プロダクト指向



従来は顧客の活動プロセスを分析しないで CS 設計をし、開発の活動プロセスを分析しないで QC 改善をしたのである。

## プロセス指向



プロセス指向においては、顧客の活動プロセスをきちんと把握して、商品の真空地帯をなくす。開発の活動プロセスをきちんと把握して、作業の真空地帯をなくし、納期や価格を改善する。QC や CS もプロセス分析と連携することにより、もっと強力な推進ができる。

## 4. プロセス指向の諸動向

新しい動きが注目されていないとはいっても、よく見るとプロセス指向の動きは結構ある。

## 1) プロセスプログラミングの研究

ソフトウェア工学の世界ではプロセスプログラミングが研究され、一般業種にも適用を拡大しようとしている。

## 2) 新情報化人材育成カリキュラム

新情報化人材育成カリキュラムでは、計算機科学の知識だけでなく、ソフトウェア工学の開発プロセスに沿った教育項目を重視している。

## 3) 開発作業の共通フレームの策定

同じく新情報化人材育成施策や ISO では、ソフトウェア開発工程の標準制定に取り組んでいる。

## 4) 製品マニュアルの工学化

製品マニュアル作りは勘と経験の状態から、開発工程を追ったプロセス指向になった。

## 5) 教育の工学化

教育も勘と経験の状態から、開発・運用工程を

追ったプロセス指向になりつつある。

## 6) 作業管理の取組

F. テイラーの職務研究以来低迷していた作業管理の分野で、小林忠嗣の DIPS 運動、伊桜淑親の仕事研究運動などが登場してきた。

## 5. 学ぶ組織と柔軟経営

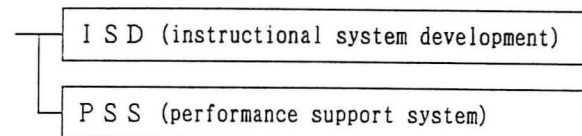
BPR とは、けっして一回限りの大構造改革ではなく、常にリエンジニアリングを行うところに意義がある。そうでないと米国が失敗した硬直化にすぐに到達する。

今後の経営は時間との勝負である。商品の真空地帯を短期に次々と開拓していき、作業の真空地帯を短期に次々と開拓していくことである。そのためにプロセスは頻繁に変えることになり、これが「柔軟経営」を実現することになる。

社員は新商品・新作業を短期間にマスターする必要がある。それが「学ぶ組織」である。そのためには、教材開発の期間短縮と、教育実施の省力化が必要になる。社員の年齢構成が釣鐘型になること、転社やパート採用が増えること、総労働時間が短縮されることも、教育のスピードアップを必要とする。

## 6. 教育活動のスピードアップ

IBM は 20 万人の社員に対して 6 千人の教育要員を抱えていた。経営の合理化においては、教育の合理化も重要課題になった。そこで「IBM の 21 世紀教育」を立案したのである。その骨子は ISD と PSS である。



ISD とは米国国防総省が開発した教育システム開発方法論であり、教材開発作業自体に、作業プロセスと要員教育という発想を打ち立てたものである。作業プロセスを明らかにしてコントロールすることにより、教材開発の期間を短縮できるし、経験者でなくても短期間で教材開発者になることができる。また教材の構造は、受講者の職務プロセスを分析して、その構造に合わせたものにする。これによって受講者の受講期間・適用期間を短縮できる。教材開発自身も専門的な



作業の一種であり、作業管理によるプロセス指向の改善の対象になる。

### ISD



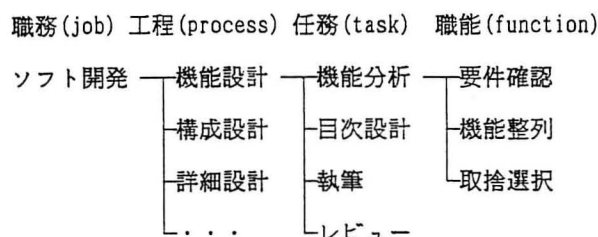
PSS とは職場の WS にヘルプ・DB・CAI・知識ベースを盛り込んだものであり、講師工数と教室スペースを節約することをもたらす。これによって「必要な時に必要な教育を受ける」ということが可能になる。

IBM は膨大な教育資源を削減するために、ISD と PSS を採用することになった。しかし、日本の多くの企業のように教育資源の少ないところでも、少ない資源で教育効果を上げるという意味で、ISD や PSS を追求するという手段は同じでよいはずである。

以上によって、BPR を推進するには、ISD が不可欠であるという結論を導き出すことができた。

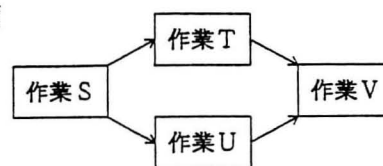
### 7. 作業管理と ISD の要点

作業管理や ISD においては、作業というものをきちんと把握する。まず、職務を階層構造に分解する。



これを作業分解構造 (WBS) という。次にこれらの作業項目を、最も性能 (納期, 生産性, 稼働率など) がよいように配列する。これがプロセスの設計であり、作業設計 (work programming) という。作業が工程階層の場合にはこれが有名な工程設計 (process programming) になる。設計の結果を作業計画 (work program) という。

### 作業計画

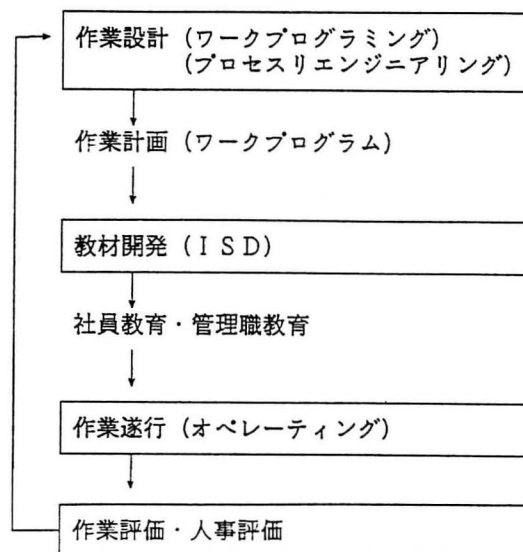


工程設計は一回限りではなく、QC 運動のように常に反省して、頻繁に改善していく。

ISD では、この作業分解構造や作業計画を確認する。もし不明確な場合には、教材開発者が作業分析を行って、それによってプロセスの構造に合致した構造を持つ教材開発するのである。

### 8. BPR と ISD のまとめ

ビジネスプロセスリエンジニアリングと ISD とは、次のような PDCA サイクルを構成する。



なお作業管理の用語であるプログラミング、プログラム、オペレーティングなどが、コンピュータ用語に転用されたことは知っておくべきことだろう。作業管理の用語がコンピュータ用語に転用されたのは、コンピュータが作業管理における合理化の代表的手段の一つだからである。情報システム開発の最初に要件分析をするのは、顧客の作業計画 (プログラム) を洗い出して、それをコンピュータプログラムのモデルにするのにはほかならない。したがって、ここで述べた作業管理の概念は、ソフトウェア開発のモデルとして、新入社

員教育などで教育すべきことだと考える。

<参考文献>

- 1) 渥美俊一,「チェーンストア経営の目的と現状」, 実務教育出版, 1986.
- 2) 渥美俊一,「チェーンストア能力開発の原則」, 実務教育出版, 1987.
- 3) 伊桜淑親,「研究発想で仕事を動かす」, 実務教育出版, 1988.
- 4) G.J.Gery, 「Electronic Performance Support Systems」, Weingarten Publications, 1991.
- 5) 清水康敬編著,「情報通信時代の教育」, コロナ社, 1992.
- 6) 君島浩,「企業内教育システムハンドブック」, ソフトリサーチセンター, 1992.
- 7) W.E.Riddle, 「ソフトウェア開発工程管理の改善を目指して」, SRA, 1992.
- 8) 通産省,「新情報革命を支える人材像」, コンピュータエージ社, 1993.
- 9) 中央情報教育研究所,「新情報化人材育成カリキュラム」, 1993.
- 10) 飯塚他編著,「ソフトウェアちょっといい話'93」, 日科技連, 1993.
- 11) 君島浩,「社内教育システム開発方法論」, 日科技連, 1993.
- 12) 小林忠嗣, DIPS 運動の著作各種



## SEA Seminar &amp; Forum September '93

堀 真人

(SONY)

mho@saskg.semicon.sony.co.jp

しばらく前になるが、SEA 主催の9月のセミナー & フォーラムに参加した。レポートをしばらく捏ねくり回しているうちに時間がたっしまい、状況はかなり変わってしまった部分もあると思うが、参加されなかった方々にとって何かの参考になればと思い、投稿することにした。

日時：9月14日9:30～16:00

会場：文祥堂ホール(東京・銀座)

内容は以下のとおり(プログラムより抜粋)。

Seminar (9:30～12:30)

with Demonstration

ネットワーク新時代—これからのビジネス基盤としてのインタネット

講師：深瀬弘恭，吉村伸，浅羽登志也(IJ)

Forum (13:30～16:30)

インタネットを利用した分散ビジネス/分散開発

コーディネータ：中野秀男(大阪大学)

パネリスト：Seminar 講師各氏，

杉田義明(日本NCD)，西武進(S&amp;S)，

佐野晋(NEC)

IJが商用インタネットのサービスを開始し、すでに数十社の企業が契約している。まず、IJの深瀬社長が実際の商用サービスを通して日本の問題点に関して説明された。

● 現在 IP(Internet Protocol) で接続されているのは、世界120ヶ国にのぼる。国際的な通信は2国間協定が必要であるが、日本が2国間協定を持つのは19ヶ国(米国は対象外)に過ぎない。他の国々とも当然通信しているが、これはアンダーグラウンドである。国際VAN業者が黙ってパケットを流しているというのが実情である。各国のIPネットワーク関係者は、それぞれの国内でこれらの協定を結ぶよう活動している。

● 日本を海外から見ると、情報に関しては流入量が流出量を大きく越えている。たとえば、アメリカ公文書館はIP接続されており、その膨大なarchiveが日本からもアクセスできる。これはもちろん無料である。たとえば、予算やクリントンのスピーチなど

は即座にアクセスできる環境にある。一方、日本では官報すらオンライン化されていない。このような状況は海外から非常に問題視されている。今後、日本が発信側に回ることが必要だと考えている。

● また、国際回線が非常に高価なために、アジアの情報センターはホンコンやシンガポールに作られているのが現状である。たとえば、シンガポールから東京への回線よりも、シンガポールからアメリカへの回線の方が安いからだ。これにより、日本はアジアにおいても情報ネットワークのleaf nodeになりつつある。こうした現状を行政がまったく認識していないのは、非常に問題である。

次に、吉村さんと浅羽さんが、IPネットワークに関する技術的説明とデモを行った。主なタイトルを挙げる。

- IP ネットワークの歴史と現状
- IP ネットワークのアクセスポリシー
- IP プロトコルとは
- IP 上のアプリケーション
- セキュリティ

特に、セキュリティについては、Dual Homed host の話から始まって、Cisco のルータで内側からの接続を許可し、外からの接続を拒否するなどの細かい設定までが説明された。個人的にはこの話が収穫の1つだった。

デモは、ISDN 回線経由で Internet にアクセスして、IP 上のアプリケーションであるarchie, gopher, WAIS を、X 端末で実際に動作させるというものだった。最近ではさまざまな雑誌にこれらのツールの解説が載っているため、以下には簡単な記述にとどめる。

## ○archie

欲しいfileがさまざまなftpサイトのどこにあるか探すツール。

## ○gopher

IPnetwork上のさまざまなサーバの情報にアクセスするためのツール。menu driven なインタフェースで、どの地域(たとえば、Pacific)のどのサーバのどの情報をアクセスするか、といった感じでどんど

ん木構造をたどって必要な情報を探す。画像も取れる。現在日本で実験的に、天気図を gopher で提供するサービスもある。

#### ○ WAIS(Wide Area Information Server)

さまざまな情報のデータベースを検索するという意味では gopher と同じだが、データベースは IP 上に分散しており、キーワードで検索してファイルをアクセスするという点が gopher とは異なる。このソフトは PDS だったが、これを製作した会社が商品として販売を開始した。

デモの形態は次の通りであった：

[ホール]

III --- ISDN 回線 ----- | -- cisco -- X 端末

このために ISDN 回線を引いてデモをした。この時点で筆者の環境は社外への IP アクセスができなかったためデモは非常におもしろかった。

午後のフォーラムは、話題が非常に多岐にわたり、IP とは直接関係のない話題も多かったので、気になったところだけを抜き出すと：

#### ● ネットワークの広がりや衰退

大阪大学でもキャンパスネットワークを張っているが、端の方は腐っている。NEC などでも同様な現象が見られる。会社というなら、ホワイトカラーの取り込みは必須である。たとえば、大学では、事務・生協・図書館が IP 上になれば便利にならないと思うが、いずれも問題がある。

#### ● ユーザインタフェース

アメリカではコンピュサーブに入ると無料でアクセス用インタフェースがついてきて、これを使うとキーボードを打つことはない。こういったユーザインタフェースが日本でまったくないのはおかしい。ネットワークがもう一段広がるためには、こういったソフトが必要だ。逆に考えれば、日本ではそのための必然性がないのではないか？

#### ● 専用線 vs III

どちらが完全に優位ということはない。しかし、多数の相手全部に専用線をはるのはコストも時間もかかるから見合わない。IP であれば、相手が IP に乗っていればだれとでも通信できる。また、III のコスト 64Kbps が月額 50 万円で専用線はせいぜい東京-岡山間。それを考えれば、全世界が見える IP は高価ではない。

#### ● 社内ネットワーク

NEC ではメールに対する課金を行っている。ま

た、松下でもそういった話が出ている。

#### ● IP でのビジネスチャンス

たとえば、ソフトをダウンロードしてもらって、ある程度使用したところで気に入ったら買ってください、という shareware 的なビジネスができるのでは？

#### [感想]

筆者はこのようなミーティングに出ると、いつも刺激を受け、一時的に脳が活性化する。特に、今回はデモを見たことで、社内での新しい開発アイテムをいくつか思いついたりした。

日本で IP 上での情報が輸入超過になっているのは、非常に問題だと思う。ソニーも近く III(あるいは Spin?) と契約するはずだが、たとえば、IR(investors relation) など不特定多数を相手にする業務は、IP 上でも行うというような体制を組めば、おもしろいかもしれない。

また、たとえば、現在、筆者の属する部署では、VMS でのソフトウェア開発が必要であり、dialup で Decnet 接続を行っているが、これなども相手が IP Reachable であれば、両端で同じソフトを使用し Decnet を encapsulation 通信することもできる。そうすれば、複数のソフトウェアハウスを相手にしても、こちら側は IP のアクセスポイントさえ持っていればいいのだから、非常に楽なのだ。

[注] Internet 関係の情報は書かれた瞬間に古くなるという部分があるので、SEA に限らずさまざまな情報源に接する必要がある。このレポートも 93 年 9 月時点での情報がかなり含まれているのでご注意を。

## SE 解放宣言

臼井 義美

(日本電子計算)

### 1. 情報民主主義の時代

ソフトウェア技術者は長年、大型コンピュータを中心としたピラミッド型管理体制の下で、機械的な言語を強要され、非人間的な作業に取り組んできた。ところが、近年ようやく自分自身の管理下で仕事ができるワークステーションと、自分の好みの仕様で設計できるCASEツール、それに快適なユーザインターフェースを手に入れた。

すなわち、これまでの全体主義的支配のもとで、決められた作業を繰り返すマニュアル・レイバーから、個人が自らの価値観をベースに独自の仕事を発見して行くヒューマン・ワーカーへと変身する環境が整ったのである。したがって従来のような一律型マネジメントは効力を失い、個人の資質や能力・個性の差によって大きく成果が左右されるようになってきた。

さて、このような状況はソフトウェア技術者にとって手放して喜べるのであろうか？ はたしてワークステーションはソフトウェア技術者を開放してくれるのか？ 待ちのぞんでいたCASEツールはわれわれソフトウェア技術者の救世主なのか？ すばらしいユーザインターフェースはわれわれの苦痛を和らげてくれるのだろうか？

情報社会の到来ともてはやされている情報産業の実体は、情報とは本質的に無関係な単なるソフトウェア製造業なのだ。ほんとうに情報を使いこなしているのは、コンピュータはもとよりソフトウェアともまったく縁のない人たちなのである。これまでは技術を制するものが世界を制する時代であったが、モノ中心の社会がコト中心の社会に移行すると、情報を制するものが世界を制することになる。

ところで、世界のリーダーは、産業革命以来、石炭・石油・金融・鉄工などの資源を元手に世界を支配していた巨大多国籍企業から、一部の革新的な技術エリートに移行した。また、巨大コンピュータのメーカーも凋落し、ワークステーションやパソコンなど、より民主的な情報機器のメーカーが繁栄している。

いまや実質的に世界の情報を支配しているテレビ業界でさえ、民主主義の嵐が吹き荒れている。従来チャネルを独占していた地上波によるテレビ局に代わって、衛星放送やケーブルテレビが物凄い勢いで伸びてきている。そこでは、放送と通信の実質的な区別は消滅し、ペイ・パー・ビュー（視聴した量に応じて料金を支払う）や、双方向テレビなどが、雨後の竹の子のように発生し、これまで特定の人間や組織に独占されていた情報を、視聴者側が自主的に選択できるようになってきた。

そういう意味において、コンピュータの世界の情報流通の手段であるパソコン通信は、非常に民主的な組織であり、現在のところ、発信者の生の情報が何者にも編集されずに受信者に届く数少ないメディアである。このように、情報を取り巻くあらゆる環境は民主化に向かっている。その中であって、われわれソフトウェア技術者は、どのように自らの民主化に取り組もうとしているのだろうか？

### 2. コンピュータ・ハーレムの崩壊

さて、おりからの不況に慌てふためいた経営者は、いままで近づこうとしなかったコンピュータ周辺の実態に目を向けはじめた。コンピュータの黎明期から聖域扱いをされて、温度や湿度・埃・火災などに対して最適にコントロールされ、管理者さえ寄せつけない奥の院に、悠然と鎮座している巨大なコンピュータが、実はまったくの役立たずであることに気がついたのだ。

このウドの大木は、電気は喰いまくるし、紙はトラック単位で食べ尽くす、大量のゴミは吐き出すという贅沢極まりない物体で、おまけにその周辺をたくさん男女が取り巻いて、1秒たりとも機嫌を損ねないように気をつけている。知らない間に奥の院はコンピュータ・ハーレムと化していたのだ。

一体この巨大なオバケは何なんだ？ われわれはこのオバケを養うために一所懸命働いていたのではなかったのか？



いま、多くの経営者はこの事実を知って怒り狂っている。「えーい、お化けコンピュータを奥の院から引きずりだして現場でこき使え!」。ダウンサイジングという言葉は、ワークステーションを礼賛した言葉ではなく、巨大なお化けに騙されていたことへの復讐の言葉なのだ。このお化け退治とともに、その周辺に寄生していたソフトウェア技術者たちも、いまやゾロゾロと奥の院から日向に這い出してきつつある。

大型コンピュータの周辺に巣食っていたソフトウェア技術者は、COBOLというナベ・カマ言語を捨て、Cというさらに非生産的なネジ・クギ言語の使用を強要されている。ナベ・カマだと一般の技術者でも扱えるが、ネジ・クギから作り上げようというのはもはや職人以外には対処しきれない。にもかかわらず専用のワークステーションを提供しようなどという甘い言葉に釣られて、取り返しのつかない泥沼に陥っているのだ。

ワークステーション、うーん、いいネーミングだ。技術者を縛り付けておくのにこの上ない表現だ。分散環境という美しい名の仕事場に縛り付けられた技術者は、ワークステーションを取り上げられると茫然自失、思考停止となり、麻薬の切れた患者のような不完全人間となってしまう。さらにその縛りを強固なものとするために、ユーザインターフェースと称する接着材をも提供し、ソフトウェア技術者とワークステーションを一体化してしまった。その上、分散化することで管理不能に陥ることを恐れた経営者は、グループウェアなどと称して、ネットワークでソフトウェア技術者の集団拘束を画策し始めているのである。CASEも使いようによっては、ソフトウェア技術者に対する作業管理の強化のための手段にさえなりかねない。

困ったことに、こうした環境は、ソフトウェア技術者自身が自ら望んで作り上げているのだ。分散環境とは、ネットワーク・ケーブルを引き回してその中を流れるデータを取り出し、決められた手順でできるだけ効率よく物作りをするをを目指しているのだ。経営者がネットワーク管理者を取り込んで、その作業状態を常時監視するようなことにでもなれば、まるで個室に入れられて流れてくる餌をついばみ、1日1個の卵を生んでいるプロイラーと同じ姿ではないか?

### 3. 製造業から創造業へ

現在のソフトウェア技術者の多くは、ソフトウェアの製造業者である。しかし、CASEの出現は、同類のソフトウェアを大量生産するためには、大勢のソフトウェア技術者を必要としなくなるということを意味する。ソフトウェアは単に情報を取り扱う手段に過ぎないのだから、われわれはその製造だけに係わってはいけいないのだ。そのために、われわれは単なるソフトウェアの製造業から脱皮して、一刻も早く情報の創造業に移行すべきなのである。

情報の創造者は、ユーザから与えられた問題をどう解決するかという立場でなく、解決すべき問題は何かを見つけださねばならない。ユーザ要求を正しく理解して正確にシステムを構築することではなく、ユーザの立場を理解して、その活動をコンピュータによって最大限に支援する環境を提供するのだ。すなわち、ユーザの活動に有効な情報を、必要な時に最適な形で取り出し、発信できるシステムが必要なのだ。過去のデータを利用して正確な集計表を作成するなどといった偏狭な世界に閉じこもってはいけいない。いままで、ユーザ要求の取りまとめやコンピュータのお守り程度しかできなかったのは、ソフトウェア技術者が情報を読み取る目を持っていなかったからである。

ほんとうにユーザが望んでいるのは、情報を活用することによって企業自体の売上が上がること、コストが削減できること、社会に貢献できること、より快適な作業環境を構築することなどである。したがって、その本来システムを利用することによる効果そのものがソフトの価値であって、開発コストがいくらかかったかは関係がない。従来のように開発コストを積算して価格をはじき出すような製造業的企业体質では、CASEを利用することによるコスト削減がさらなる企業間競争を招くだけである。われわれが一所懸命頑張ってソフトウェアを製造し、少しでも効率よく作業できるようにと努力した結果が、自らの豊かさにまったくリンクしていないのだ。

まず、システムを開発している目的が、企業の競争力強化という企業サイドからの論理にしたがって、ただ黙々と働いている半機械の人間と化したソフトウェア技術者にとって、ユーザ本位のシステムなどいつまでたってもできはしないのだ。もし、コンピュータを自分自身のために使いこなしており、自らもユーザな

のだということが理解できれば、UNIX なんぞという非人間的な開発環境に耐えられるはずがないではないか。自らを機械の奴隷と自認しているからこそ、改善も革新もなくただ与えられただけのアホな OS のもとで、自逆的な環境にじっと我慢ができるのだ。だから、数字をいっぱい書き込んだ販売一覧表や原価計算書・収支一覧表などという、単なるデータのダンプリストみたいな紙屑を臆面もなくユーザの前に差し出すのだ。そんなものは、データベースの中に放り込んでおけばよいのだ。忙しい経営者や管理者に、何の目的もなくプリントした紙屑の中から、必要とする情報を自分で選び出せというのか？ それなら、その間コンピュータは一体何をしているのだ？ そのような紙屑をのんびりながめているバブル経営者こそ、社会から抹殺されるべきであり、紙屑を作って得意になっている技術者も、オバケもろとも葬り去られるべきなのだ。

本来、豊かさとは金を儲けることでなく、金を使うことでなく、欲しいものを手に入れ、やりたいことをやることなのである。企業と一緒に危機感にさいなまれたり、過剰なガンバリズムに巻き込まれて自分を見失ってはならない。もともとコンピュータは、人間の不得手な部分をまかせて本来やりたいこと、すなわち自己実現を支援してくれる魅力的なしもべとなるはずだったのだから。

#### 4. CASE は救世主か？

これまでのように、ソフトウェアの製造屋として CASE を利用すると、バカの暇潰しにしかならない紙屑を放出するだけの、大量の類似ソフトウェアが市場に流れ出し、社会は膨大な屑ソフトウェアと不要データのゴミ箱となりかねない。われわれは、CASE を社会にソフトウェア公害をもたらすゴミ製造機にしてはならない。

ところで、日本に一体どれくらいの会計システムが存在しているのだろうか？ この分野などは、目的が明確で、また法的規制が多く、自由度の少ない領域であるのに、自力でシステムの開発ができる企業のほとんどが、独自の会計システムを持っているのではないだろうか？ そして、一旦税率の変更などが発生すれば、日本中の会計システムの変更に膨大な電力と紙を消費し、何千何万という SE が、身勝手なスケジュールで労働を押しつけられた上に、かなりの数のシステムがバグという名の後遺症に悩まされるのである。わ

れわれはもう何十年も同じ愚行を繰り返している。ソフトウェア技術者自身が目覚めなければ、分散環境に変わったからといって、このような問題は決して解決されることはないだろう。

極論すれば、会計システムは日本に数種類あれば十分である。商法や会計規則に定められた仕様で仕訳処理を行ない、期間集計を行なって所定の会計帳票を作るだけである。このようなレベルで企業の独自性が意味を持つわけがないので、数多くのオプションやカスタマイズ可能なモジュールが用意されていれば十分である。それなら、意欲のある会計専門のソフトハウスが、その全知全能を振り絞って最高のシステムを構築すればよい。それでこそクリエイタとしての独自性を発揮できる。

その会計システムにユーザ企業の独自性を付加したいのなら、データの集合から、あるタイミングでその経営者独自の判断で、最も適切な情報加工ができるような環境を提供することである。いつ利用するのかわからないような集計表を、紙にジャカスカ印刷するような愚は犯してはならない。

ソフトウェアの生産性向上の最有力手段として、いつも引合に出される技術に部品化と再利用がある。自動車や電気製品などのハードウェアの製造工程で部品化が成功したのは、業界標準をいち早く設定し、その規格に合わせて多くの部品メーカーが標準部品を供給したからである。ソフトウェア業界では、共通に利用できる部品がまったく提供されないで、わずかに社内標準などを決めて、細々と無益な再利用のまねごとをしているだけだ。

われわれは、広く共有できる部品の規格を決定し、優秀な部品メーカーを育てる必要がある。現在、この業界は先端産業だと自負しているようだが、実態は産業革命以前の工場制手工業となんら変わっていない。ソフトウェア技術者自らが、口ではオープンシステムなどといっているくせに、ソフトウェアの閉鎖的世界をどのようにオープン化するかというビジョンには、ついぞお目にかかったことがない。だいたいコンピュータ同士を接続するために、なぜ人間が介入しなければならないのか？ プログラム同士でデータ交換するのに、なぜ人間がそのつど駆り出されなければならないのか？ こうした状況は、ソフトウェア技術者がコンピュータの奴隷であることの証左ではないか？

このような欠陥に満ちた環境の中で、ソフトウェア技術者は、自らの意志と無関係なシステム開発に黙々と取り組んでいる。人手不足の時代には、女性の社会進出などと企業と癒着したマスコミに踊らされ、若い女性たちまでもが、このくそ面白くない非人間的な作業に大量動員されている。まるで野麦峠にでてくる女工哀史の再来である。自らの意志で創造する温かくて幸せな家庭作りを放棄して、現在の情報支配者であるマスコミに洗脳されたとはいえ、組織に従属しコンピュータの奴隷として生きるのはあまりにも惨めではないか？

#### 5. 目覚めよ、ソフトウェア技術者！

さて、この不孝な現場に残されたソフトウェア技術者諸君！ 馬鹿な仕事からきっぱりと手を引こう。何のためにコンピュータが存在するのだ？ われわれは、コンピュータが機嫌よく働いてくれるように世話をする介護人ではない。コンピュータのバカさ加減を補うために配置された奴隷でもない。経営者が見えをきって導入したコンピュータの空き時間を埋めるために、無理矢理仕事を捜しだし、その仕事をこなすためにソフトウェア技術者がかき集められ、人件費を稼ぐためにさらに仕事を作られ、仕事の量に耐えられなくなったコンピュータをグレードアップするという悪循環の中で、ドサクサまざれに現場に押し込められただけなのだ。

ここでも日本人の農耕民族としての性格が遺憾なく発揮されている。マスコミを天の声として崇めることによって、拝金主義に犯されたソフトウェア技術者は、自らの人間としての権利を主張することなく、金のためひたすら自分の身を削って、コンピュータの横暴にただじっと耐えているのだ。このような状況では、オリジナルな情報の創造はもとより、それを支援するソフトウェアさえ、日本人自らが構築することは困難であろう。いつまでたっても他の文化圏の価値基準で作られたソフトウェアを買い漁り、不本意ながらもその環境に自分を合わせることによってのみ、自己の存在を確認するしかない哀れな存在なのだ。

ソフトウェア技術者諸君、目を覚まそう！ ワークステーションの登場は、われわれの民主化を達成できる千載一遇のチャンスなのだ。自らの内面から湧き出る感性和シンクロナイズできるようなユーザインターフェースは、われわれの創造性をかき立ててくれる快

適作業空間を提供できるだろう。また、CASEこそは、われわれが従来こき使われてきた作業をコンピュータに担当させることによって、コンピュータを利用する側につけるのだ。われわれは自らの責任において、コンピュータは人間の苦勞を軽減し人間の心の欲望を満たすために作られたのだという本来の姿を、内外に徹底的に知らしめてやらねばならない。いまこそソフトウェア技術者は、コンピュータの奴隷から、コンピュータの使用人に劇的に変革するのだ。そのためには、まず、自分たちが開放されるべき存在であることを自覚することから始めなければならない。

しかし、救世主として現われた CASE は、やがてわれわれの同胞を叩きのめす日が来るであろう。CASE が普及しユーザインターフェースが知的になれば、エンドユーザは自分のやりたいことをコンピュータの支援の下で実現するだろう。そうなれば、製造業者としてのソフトウェア技術者は、もはや無用の存在と成り果て、大勢の同胞が飢えに苦しみ、ソフトウェア難民となって幾多の他業種へと漂流していくことになるのだ。だから、コンピュータの奴隷から開放され、コンピュータを意のままに動かすためには、一刻も早くソフトウェア・エンジニアから脱皮し、自分独自の視点から情報の価値を読み取り利用できるインテリジェンス・クリエイターへ移行するのだ。その鍵は、ハードウェアでもソフトウェアでもなく、情報の中に隠されている。世間に流布している数々の情報の意味を鋭く嗅ぎ分け、自分の目でもう一度世界を見つめ直すそうではないか？

[注] 以上は、日本情報教の狂信者のたわごとですので、関係者の皆様、気にしないで下さい。

---

臼井 義美 (Yosimi Usui)  
日本電子計算 (株) 大阪支店  
社会システム営業部  
usui@ss.osaka.jip.co.jp

---



## 情報処理産業のこれから

横山 博司

(松下電器産業)

いまは、経済全体が、縮小の傾向にあります。いわゆる経営マインドの冷え込みです。円高も手伝って、自動車や電機に代表される職種が減少傾向にあります。そのため、不況下でも合理化のための投資として、“不況に強い”といわれていたはずのコンピュータ業界、なかでもソフトウェア業界がもろに波をかぶっています。また、業界内の流れでも、情報システム部門のダウンサイジングや、EUC(エンドユーザコンピューティング)に代表されるような動きにも、大きな影響を受けています。

業界トップのF社も状況がおかしくなっていますし、低価格DOS/Vに押されまいとするN社も大変な労力を払っています。ただ、PC-98に代表されるパソコンに強いN社は、汎用コンピュータから、オフコン、パソコンにいたる全領域で、一人勝の様相を呈しています。これは、N自動車が、セドリックなどの大型車で強みを見せていた時期に、T自動車は、小型車のカラーで巻き返しを図り、順次、ブルーバード、スカイラインと大型車を駆逐していった構図と似た感があります。しかし、“付加価値”と称して、セルシオなど大型化・高級化に走ったことが、買い替えサイクルの長期化、部品点数の増加という形となってあらわれ、重荷になってきています。それは、実用性を重視したマーチが根強い人気ということが証明しているでしょう。

ところで、各企業ではいま、経費削減の号令により、接待費はもとより、設備費、外注費、教育費から、蛍光灯、鉛筆、消しゴムに至るまで、ことこまかく実施されてきています。その構成員であるサラリーマンが影響を受けないわけがありません。個人は、今年春の辛口な昇給、夏・冬のボーナスからもわかるように、収入も増えない。財布の紐が固くなって必要なもの以外は買わない、いい方を変えれば、本物が試されているということがいえるわけです。

さて、ここで、ソフト業界に目を向けてみましょう。いままでの業績は、はたして本物だったのかということです。答えは、いうまでもなく、Noです。情報化投資が残したものは、“動かないコンピュータ”の氾濫です。“ダウンサイジング”という名のもとに、やみくもに買った数百万のワークステーションは、いまどうしていますか？ メールマシンですか？ お絵かき道具ですか？ 1日中電源が入れられないまま、あるいは、キーボードが触れられないまま放置され、“何に使おうか？”とって担当者

は、右往左往しています。せっかくの高性能のマシンが埃をかぶっているのです。

あるいは、パソコンについても、つい昨年買ったばかりなのに、お店には、もう新しいのが並んでいる。性能は倍で値段は半額ということも、珍しくありません。Windowsが動かないから、一太郎の新バージョンが動かないからといってはまた購入。古いマシンは、片隅においやられています。それでも触ろうものなら、回りから「いまごろそんなもので何してるの？ 遅れてルー！」という非難の目が向けられるでしょう。

新しいワークステーションは使い切れずに放置され、ちょっと型の古くなったパソコンは片隅においやられています。つまり、顧客は、大きな投資をしてきたのです。それに対して、情報処理産業界は、とても本当に胸を張って応えてきたとはいえません。

SEの原点は、ユーザーニーズの把握にあるわけですが、提供した品物は、“ハード”あるいは、“パッケージソフト”であり、システムではなかったのです。コンピュータメーカーの営業は、ハードの値段を顧客に示し、ひどいときには、ソフトを1円で落札したのです。1円でどんなシステムが組めるというのですか？ システム開発をすればハードの値段以上にコストがかかる。そんなことを正面から提示しようものなら、高すぎて契約は取れません。

そのことを示す事例を少しお話をしましょう。以前、システム開発を受託するにあたって、私は、十分に検討した上での見積り価格を提示しました。すると「倍見積りだ」という非難をよく受けたものです。それも外からだけでなく内からのものです。ちゃんとした信頼性の高い、メンテナンス性のよいものを提供するために、正規に見積もると世間相場の倍になるのです。このようなことでは、失注は明らかです。

つまり、顧客には、本来の価格を提示できずにいるわけです。だからよくて半値で開始したシステムの開発を進めていくと次から次へと追加変更という費用が発生していきます。もちろん、完成後のメンテナンス費用も相当なものです。費用は雪だるま式に増えて行き、顧客は、“こんなはずじゃなかった！”と嘆くことになります。

視点を変えて、システム開発の内幕を少し追ってみましょう。受注したシステムを誰が開発するのかといえば、

SEと称して、入社したてのいわゆるアマチュアがプログラムを開発します。これは、「ソフトウェアクライシス」として騒がれた人材不足を背景に、やむなく許されてきました。逆に、幸運にも数少ないスキルの高いプログラマーにあたっただとしても、多くの仕事をかけ持ちしているかれらにとって、仕事に見合った報酬が与えられるわけではなく、給料など安く押さえられているのが現状です。

技術の向上などを図る機会すら与えられないこともあるでしょうし、生産性の名のもとに、同じ時間、同じステップ数をこなした人が、信頼性などの品質とは無関係に同じぐらいの金額を得ることのできる状況では、意欲までも失ってしまうことになります。

そんな中で、ちょっと気の利いたSEは、いや、技術力の向上をあきらめた人たちは、自分で開発することをやめ、人を動かして楽して稼ぐ味を覚え、その「術」に長けようとするわけです。これらの手配師をマネージャーと呼んで、現場から離れ、人を替え、派遣先を替え次々と配員して行くことで、自分の食いぶちを稼がせるようなことをすることになります。

そんなことから、いざシステムに問題が起こったときに、顧客が、マネージャーたる人物にクレームを出しても、かれが解決してくれるわけはありません。現場の担当者が呼びつけられ、即時対応を迫られます。これは、とても企業活動と呼べるようなものではなく、個人に責任をかぶせているだけなのです。

もう一度、見積り時点まで話を戻しましょう。その時点で、どんなシステムができあがるのか、できあがり動かして見せられますか？ できあがってみなければわからない。特に、古典的なウォーターフォール・モデルにしがみついて（それすら採用していない組織は論外！）いるのが実情で、顧客としては、どんなものができあがるかわからないようなシステムに、恐ろしくて費用は出せません。そこで、派遣社員として「管理のしやすさ」の名のもとに、目の届くところで常駐させてシステムを開発するわけですが、これがとても最初の見積りに合わない。管理面でも、技術面でもスキルの低さにより、スケジュールはどんどん遅れて、「完成した」ハズのシステムを動かしてみると、ほろほろとバグレポートの山となってしまう。

結局これらをまとめてみますと....

顧客は、何億もの大金を払ってきたが、満足のいくシステムが手に入らない。SEは、よい仕事をしようが、技術の向上を図ろうが、ステップ単価・時間単価であり、給与に反映されない、高い報酬が得られない。ソフト会社

は、顧客から高い収入を得られないし、人のやり繰りだけに終われ、そのうちにぼろが出て仕事にあぶれる。メーカーは、ハードの価格下落により収益がでない。こういったことの悪循環になってしまうわけです。また、これらは、現場での開発手法の未確立、ソフトウェア技術者の社会的認知の低さとなってあらわれています。

その対策としては、まずコンピュータシステムとは何なのか、利用技術とは何であるかを熟知して、顧客との一番近い接点に位置しているソフトウェアハウスが、以下のことを実践し、真に顧客に社会に貢献することです。

#### 1. 顧客の絞りこみと信頼の熟成

- (1) ハード、パッケージという素材提供でなく、システムの提供によって顧客の満足を得る。
- (2) 技術スキル向上の奨励による高度な技術の提供。

#### 2. 若手の起用

- (1) 次世代情報システムに向けてのプロジェクトの見直しと採用。
- (2) 個人向けの汎用システム（情報家電と呼ばれる分野）と法人向け専用システム開発の2局化への対応。
- (3) 貢献度に応じた十分な待遇。

#### 3. 管理の複雑さと重複部門の見直し

- (1) 収益に貢献していない部署、部門を削減する。→ マネージャーの整理統合
- (2) 実行の伴わない評論家の排除

#### 4. 支出の削減

収益につながらない不要不急業務の廃止

#### 5. 長期投資の見直しと確立

ソフトウェア工学の確立の為の先行投資。

-----  
<あとがき>

書き始めると止まらないな。もっと話は出てきそうですが、今回は、これでおしまい。多分に偏った意見かもしれないので、反論など、ご意見のある方はどうぞ（応えられないかもしれませんが）。趣旨としては、ソフトウェアエンジニアリングの発展にエールを送っているつもりなのですけど...

横山博司 (e-mail:hiroshi@str.cpd.mei.co.jp)

(略歴)

1979年: (株)SRA 入社

1991年: 松下電器産業 (株) 情報機器本部コンピュータ事業部入社、在職中。

1993年: SEA 関西システム技術分科会世話役

## SEA のリストラクチャリングについて

坂本 啓司

(オムロン)

最近の SEAMAIL に、IBM の大場さんが「SEA のリストラクチャリング」というタイトルで連続投稿されており、最新の Vol.8, No.6 では、かなり思い切った大胆な提案をされています。これを受けて会員の意見を問うアンケートが実施され、また先日の幹事会（だったかその後の忘年会だったか）でも、1つ1つの提案についての議論がされました。アンケートの結果や議論の内容については、いずれ公式の報告があると思いますので、そちらに任すとして、ここではメールによるレスポンスを返すような軽いノリで、私の意見を述べたいと思います。

私は去年の秋から幹事をさせていただいていますが、以前の SEA の状況についてはあまりよく知りません。幹事になってから SEA の内情が分かり、財政のフローが毎月赤字でストックを食い潰していることと、会員数が以前聞いていたものより減少していることが気になっていました。そこに大場さんの連続投稿と提案が出されて、これに刺激され、私からも活性化の提言をしてみようと思ったわけです。

さて、皆さんは SEA の会員になるにあたって、それぞれの思いや目的をもっておられたはずですが、自分の意志で入会されているからには、「会の発展を願う」のは共通の思いのはずです。会の発展にはいろいろな形があると思いますが、象徴的なものは会員数の増加だと思っています。

そこで、このあたりで「会員を増やそうキャンペーン」をやってみてはどうでしょうか？ というのが私の提案です。

キャンペーンの具体的提案の前に、なぜ会員が増えないのかを考えてみました。

その1 SEA を知らない。

その2 わざわざ会員にならなくても会員と同じメリットが享受できる。

その3 メリットがないと思っている。

その4 SEA が嫌い。

SEA が嫌いな人は放っておくとして、前の3つの原因を取り除くことについて提案を見てみたいと思います。

### (1) 有名人は SEA の宣伝を

本を書く・講演をする・セミナーの講師をする等の有名人が、幹事を中心に多数おられますが、この方々に SEA の宣伝をしていただく。本の著者欄に「SEA 会員」と書いてあったのは、私の知るかぎりでは富士通の君島さん

だけです。本を出版される方は、ぜひ「SEA 会員」と入れてください。SEA 以外の集まりで講演・講師をされる方は、必ず SEA の宣伝をお願いします。

### (2) 会員と非会員の差をもっと明確に

会費は個人持ち、参加費は会社持ちがほとんどのため、セミナー等の参加費の差別はあまり効果がないと思います。半分冗談ですが、セミナーで非会員は後ろの方にしか座れないとか、ワークショップ等の合宿で泊まる部屋に差を付けるとか、個人レベルでメリットの差を感じるものにしたらどうでしょうか？

### (3) 宣伝するための情報を集める

会員を辞めた人、SEA のイベントには参加するが会員にならない人の意見を集めて、対策を考える。また、皆さんの感じておられるメリット、会員になった理由について SEAMAIL に投稿していただく。これらの情報をもとに、各会員はそれぞれ1人以上の新規会員を勧誘する。ということで、さっそく、私自身が感じている SEA 会員としてのメリットを述べてみたいと思います。

歴史的経過（ちょっと大げさですが）でいいますと、3段階に分かれます。まず入会当時は会社の枠を越えて情報入手ができることに、大変な新鮮さを感じました。しかも、その内容はメンバーのキャラクターを反映し、本音の、とても面白いものばかりです。しばらくは give のほとんどない take ばかりでしたが、お世話になるばかりでなく何かお役に立つことは？ とボランティア精神が出てきたのが次の段階です。この段階で会社には、「世間のお役に立つことをやっているのだから」と出歩くことを大目に見てもらうようになりました。そして、最後の段階ですが、現在は SEA の集まりに出ていろいろな人といろいろな話ができること、すごい方々ばかりでこんないい方は失礼かとは思いますが、お友だち・人脈づくりが最大の楽しみです。

それにしても、この種の議論はメールでやり取りするのが一番いいように思うのですが、SEA の会員がすべてメールで繋がっていない現状では、無理な望みですね。すでに2回ほど特集がありましたが、YDOC や seakansai といったメーリングリストでの議論の一端を報告してもらおうとか、または、ネットに繋がっていない方は、今回の私のように軽いノリで、直接 SEAMAIL に投稿してみませんか？



## リストラとウルグアイ・ラウンド雑感

大場 充

(日本 IBM)

IBMの大場です。前号までの連載、『SEAのリストラ』を終えて、ほっと一息。肩の荷がおりたような感じです。ところが最近、『日経』紙を読んでいて、ある記事が目にとまりました。そんなことがきっかけで、この連載後記を書くことにしました。

その記事というのは、『リエンジニアリング』に関する解説記事でした。リエンジニアリングといっても、ソフトウェア工学の話ではありません。『ちゃんとした』経営の話のようです。その記事では、『企業のリストラ』と『リエンジニアリング』とはまったく異なるものであると主張していました。そうかも知れません。

その記事の著者によれば、『リストラは『受動的』であるのに対し、リエンジニアリングは『能動的』である』そうです。つまり、リストラの『きっかけ』が『組織の外』から与えられるものであるのに対して、リエンジニアリングは、『組織の中』から、主として経営のトップの意志にもとづいて始まるものだそうです。

もう一つ、『リストラは『利益を生み出さない』のに対し、リエンジニアリングは『組織に大きな利益をもたらす』ものである』そうです。リストラでは、企業内での余剰人員を減らし、組織をスリムにすることを要求される例が多いことは、事実のようです。記事の著者によれば、リエンジニアリングでは、『従来のビジネス・プロセスを根本から見直し、プロセスを最適化する。このとき、余剰な資源が見つければ、そのような資源を別の目的で有効利用することができる』そうです。

ところでこの著者は、『リエンジニアリングとは何か』の定義についてまったく記述せずに、記事を終わっています。こんなことは、『ソフトウェア工学の世界では許されない』ことです。私の推察では、『リエンジニアリングとは、従来のビジネス・プロセスを、ある『明確な組織の目標』との関連から見直し、その目標を最も効率よく達成するようプロセスを作り替えること』です。ここで重要なことは、『組織の目標』を再確認し、必要ならば明確に再定義することによ

う。

もしこの私の理解が正しければ、そのようなビジネス・プロセス・リエンジニアリングは、全社的な品質管理(TQC)でいわれている『プロセスの改善』と同じものです。どうもそれが、従来からいわれているプロセス改善とは少し違うもののようです。では、『何が違うのか?』。私の解釈では、『やはり『リストラ』ではないか?』ということです。つまり、『従来型の『インクリメンタルな改善』ではどうにもならない状況に追い込まれている』ということではないでしょうか?

組織の最も基本的な目標にもとづいて、組織の現状を検討し、原点に立ち返って組織を作り直す。これが、組織の再構築、すなわち『リストラクチャリング』です。もう一步踏み込んでいえば、『組織の再生』です。そのためには、既存の組織構造を壊し、既存のプロセスを壊し、新しい状況(時代)に適合した組織構造・プロセスを作り上げなければなりません。組織のイノベーションです。既存の枠組みを根本から組み替えるための『創造的破壊』が必要です。

『リストラ』とは、そのような『既存組織の創造的破壊と再生の行為』につけられた名前です。つまり、『現象』につけられた名前です。ここでは、その行為の意図と周囲へのインパクトだけが問題にされているのです。『どうやるか?』の方法については、問題にしません。『リエンジニアリング』は、まさにこの『方法』を問題にしているのです。『どうしたらうまく、効果的な組織の再生ができるか?』の質問に答えようとしているのが、リエンジニアリングだと思います。

リエンジニアリングは、リストラクチャリングの手段です。ここに1つの仮定があります。それは、『組織の目的は、不変である』という仮定です。つまり、『個々の組織には、組織固有の目標があり、その目標は変わらない』とする原理です。この原理にもとづけば、『組織は一過性のものであり、目標が状況に適合しなくなれば、その存在する意味はなくなる』のが当然です。理論的には、そうかも知れません。少なくとも

も米国では、それが真実のように見えます。米国では、一部の超有名企業を別にすれば、「30年以上、ある企業が常に一流企業としてとどまることはできない」といわれています。わがIBM社が、この「経験則」に挑戦していることは衆知の事実です。

よきにせよ悪しきにせよ、アメリカ社会は「組織が技術革新する」動的な社会です。しかし、日本の社会は「組織が環境に合わせて変態(メタモルフォーズ)する」静的な社会であるように見えます。少なくとも、過去何百年間はそうであったといえます。私は、これが「日本社会の強さの原点」だと思っています。もちろん、変態の過程で滅んでしまう組織はあります。しかし、「組織が変態できる」ことは、うまく行けば「ほとんど永遠に組織が存続し得る」可能性を暗示していることになります。「天皇制」、『家』など、日本古来のシステムは、そのようにして現代でも存続し続けているのです。

「システムすなわち組織」ではありません。『家』がよい例ですが、そのようなシステム(ドラッカーの言葉を使えば、『コミュニティ』)では、『存続自体が目的』なのです。これに対して、『組織』は、「社会に特定の影響を与える」という具体的な目標をもっています。ソフトウェア産業に属している企業組織についていえば、『社会に与える特定な影響』とは、特定ユーザ向けソフトウェアの開発サービス、ソフトウェア保守サービスであったり、汎用ソフトウェア製品の開発・販売です。

そのようなサービスやソフトウェア製品を、他の企業と競争できる(より高い)生産性で生産でき、市場で競争できる(より安い)価格で供給できるとき、企業はそのような生産を「経済活動」として実施できるようになります。つまり、社会に需要が存在し、他の企業よりも高い生産性で、よりよい品質の製品またはサービスを生産できるとき、はじめて企業は組織として社会に存在しうるわけです。

社会が進歩すると、それまで社会全体と調和していた組織の目標が、社会と調和しなくなったり、反社会的とさえ見られるようになることがあります。冷戦が終わってしまうと、平時にも莫大な投資を必要とする「軍需産業」の目的は、社会との調和を失い、組織の存続の危機に直面します。これが組織の運命です。そのような組織を、もはやアメリカ社会は必要としないし、そのような莫大なコストを国家財政から支出

すること自体、いまのアメリカには不可能なことになっています。

冷戦という世界状況が終わったいま、米国の軍需産業は存続の危機に瀕しています。多分、存続は不可能でしょう。つまり、社会の状況が変わり、そのような組織を必要としなくなりつつあるのです。米国のコンピュータ産業も、それに近い状況にあります。もはやコンピュータ産業は、資本集約的でも、知識集約的でもなく、自動車産業と同じような単なる「労働集約型産業」になりつつあります。そのような理由で、労働コストの高い米国でのコンピュータ生産は、社会の実情に合わなくなりつつあります。

コンピュータ産業を取り巻く情勢は、わが国でも同じです。われわれの生活実感がどうであろうと、われわれ日本人「被雇用者」の労働コストは、世界一高いのです。日本人労働者の生産性が、その賃金に見合うだけ高くなければ、日本でのコンピュータの生産は不経済以外の何物でもありません。結局、国産コンピュータ・メーカーの世界市場における競争力をそぎ落とし、組織としての存続を不可能にするだけに終わるでしょう。

わが国コンピュータ・メーカーの「リストラ」も、わがIBM社の「リストラ」も、突き詰めれば、それらの企業の「メタモフォシス」(Metamorphosis)を可能にし、組織としての寿命を長くするための「自己調整」であるかのように見えます。別の言葉でいえば、組織の「延命手術」です。そのこと自体が反社会的な行為かどうかの問題は別として、少なくとも「日本人的な感覚」では、組織も「家」などのシステムと同じように、「存続を重要な目的の一つにしている」とする観察が成り立つように思えます。

まさに同じことが、ソフトウェア産業についてもいえます。インドのソフトウェア技術者の8倍以上の生産性、米国のソフトウェア技術者の1.3倍以上の生産性で、ソフトウェアを開発・保守できなければ、同じソフトウェアをわが国のソフトウェア技術者が開発・保守する意味はありません。『米の関税化』で世間を騒がせている『ガット・ウルグアイ・ラウンド』のほんとうの意図はここにあります。『サービスの自由化』と『相互認証』です。最近、ソフトウェア業界をにぎわせている「ISO 9000」も、この世界的な流れに沿ったものだそうです。

「サービスの自由化」は、いうまでもなくすべてのサービス業務に関して、国境の障壁を無くそうとする考え方です。今日まで、サービス業務は「非関税障壁」が最も顕著で、自国の企業が他国の企業に対して、きわめて有利な立場でビジネスを展開できる業種でした。「サービス先進国」の米国は、この点について従来から関心をもっていました。それは、米国は将来、「もの」を大量に輸入し、付加価値の高い「サービス」を輸出する「サービス立国」となることが予測されているからです。

「サービス立国」としてサービスを輸出するためには、輸出先の国（輸入国）における政府の業務に対する許可や認可は、輸出の障壁となります。そのような将来の輸出障壁を、米国はいまのうちに取り除いておこうとしているのです。それはそれでよいと思います。わが国の産業も、その時までには高付加価値サービス生産の競争力をつけて、競争すればよいからです。わが国からの輸出も可能です。サービスの自由化とはそういうことです。10年後、いま農業従事者がやっていることを、われわれソフトウェア技術者がもう一度やるのでは情けない。

「相互認証」とは、米国系の企業や米国籍の個人が米国内で受けた業務許可や認可、資格などを、基本的に日本国内でも認めることをいいます。逆のケースもあります。わが国の企業や個人が米国で、日本で受けた免許や資格の範囲で同等の業務を営むことが許されます。簡単な例は、大学卒業（学士）の資格です。日本の国内であれば、どの大学で教育を受けたかは、資格としては問題になりません。ところが、たとえ米国の有名大学を卒業していたとしても、（文部省の管轄下にある日本の）「大学卒業」とは正式には認められません。こんな不合理なことはなくなります。

具体的に、ソフトウェア産業について考えてみましょう。米国のソフトウェア・ハウスがイギリスの認証機関から受けた ISO 9000 の認証は、日本国内でも有効です。逆に、日本のソフトウェア・ハウスが日本の認証機関から取得する認証は、米国やヨーロッパ各国においても有効になります。これは、非常に重要なことです。ISO 9000 の認証を取得したソフトウェア会社は、実質的には世界中どこでもビジネスを展開できるからです。認証を取得した外国のソフトウェア会社と、そうでないわが国の企業（ソフトウェア会社）が、わが国の企業からの受注で競争した場合、認証を取得

している外国の企業が有利になるからです。

この問題について説明しましょう。もし上述のような状況で、発注者であるわが国の企業が、発注したソフトウェアを部分（部品）とした製品を販売するためには、ISO の品質規格に関する認証を受けた（この場合は外国の）企業に発注せざるを得ません。そうしなければ、その製品の販売が困難になるからです。特に、外国への輸出は不可能になります。また、発注者である企業が、発注したソフトウェアを社内利用する場合でも状況は変わりません。その企業が製品やサービスを輸出する場合、認証を取得している企業へ発注しておいた方が有利です。輸出の意図がない場合でも、「公正な取引」を第三者に対して実証し易い利点があります。

ガットのウルグアイ・ラウンドと日本企業（特にソフトウェア業界）のリストラは、以上のように「サービスの自由化」と「ISO 9000」という2つのキーワードで深く結び付いています。付加価値の高い、ほんとうの意味で「日本人の口に合った米」を作れなければ、わが国の「米農業」は世界的な「米農業競争」に勝ち残ってゆくことはできません。同じように、「日本のユーザがほんとうに必要としているソフトウェア」を開発し、低価格で提供できる力がなければ、わが国の「ソフトウェア業」は国際競争に負け、消滅して行くことになるでしょう。

そのような国際的な「ソフトウェア競争」に勝ち残るために、わが国のソフトウェア業界は再構築されなければなりません。これは、単に技術の問題ではありません。技術を組織的に利用し、生産活動に組み込むマネジメントの問題です。日米の自動車競争でわれわれが学んだことは、「競争力の根源が単なる技術にあるのではなく、日本的品質管理に代表される組織化にある」ことでした。米国のビッグスリーはそのことを学び、積極的にリストラに取組み、今度はさらに「組織化」に取組み始めています。ソフトウェアの分野でも似たようなことが1991年ごろ米国で起こりました。「ソフトウェア工場に学べ!」がそれです。

ソフトウェア業界のリストラで達成すべきことは、ビジネス面での「専門化」（ドメイン指向）による国際競争力の強化と、技術面での「プロセス再構築」（ソフトウェア・プロセスのリエンジニアリング）による生産力（高い生産性で、品質の高いソフトウェアを開発できる能力）の増強であると思います。この両者とも、



「組織のノウハウ」が基本です。つまり、「解決すべき問題に関するノウハウ」と「解決法についてのノウハウ」です。各企業は、この2つの軸で、自社の「知的資産」を再評価し、専門化すべきドメインを選択し、そのドメインの問題に焦点を絞って技術を再評価し、プロセスを再構築して行かなければなりません。

注意しなければならないことは、ここでいう「技術」とは、単なるプログラミング言語の知識などをいっているのではなく、問題を解決するための知識をいっていることです。「COBOL プログラマは、もう必要ない」といういい方は、まちがいです。「COBOL か C か C++ プラカ?」が問題ではなく、「どんな問題を解いた経験があるか?」が重要なのです。同様に、「JSP か SA か OMT か?」が問題ではなく、「どの方法でどんなシステムを設計した経験があるか?」が重要なのです。重要なことは、「現実の問題を解くことができるかどうか?」であって、「何を知っているか?」ではないということなのです。

そのような状況で、いま SEA に求められていることは何でしょう? いうまでもなく、SEA はソフトウェア技術者が自分の意志で参加している団体、「コミュニティ」です。それは、企業の壁を超えた、ある意味でグローバルな、ソフトウェア技術者自身によるソフトウェア技術者のためのコミュニティです。このコミュニティに参画することによって、われわれは「何かを学ぶ」と同時に「何かを教える」のです。それぞれのコミュニティには、それぞれ固有の文化があります。SEA にも特有な文化が育っています。その文化からも、われわれは影響を受けています。

それでは、具体的にわれわれは何を学び、何を学ぼうとしているのでしょうか? 1つは、技術に関する「ローカルな知識」です。特定の技術の事前評価や事後評価などがそうです。たとえば、「CMM は役に立つのか?」という問題について、いろいろな人から意見を聞くことができます。もちろん、自分の意見もいうでしょう。自分の意見を表明することによって、われわれは合意の形成に参画しているのです。合意ができれば、それは SEA の文化に組み入れられます。そのような、いろいろな問題に関する暗黙の合意の積み重ねが、今日の SEA の文化の基盤になっています。この「文化の形成に参画し、結果としての合意を知識として受け取ろう」とするのが目的の一つでしょう。

これからの技術者は、技術者として世に出てから、

技術者であることを断念するまで、「たえまなく」学習することを要求されます。それが、知識社会を生き抜くただ一つの方法です。いままでの技術者は、一度ある技術に習熟してしまえば、一生その技術を基礎に「価値を再生産する」ことが可能でした。ところが最近では、技術は急速に陳腐化し、1つの技術だけで価値を再生産してゆくことは不可能になりました。アセンブラから C へ、そして C から C++ へと、技術をどんどん更新してゆかなければ、プログラミングができなくなっています。このような状況では、「次に何を学ぶか?」が問題になります。「適切な技術を適切なタイミングで取り入れて行くこと」の判断が技術者に求められます。

コミュニティとしての SEA は、個々のソフトウェア技術者に対して、「次に何を学ぶべきか?」を議論する「場」を提供します。もちろん、情報も提供します。しかしこの場合、「場」の提供機能が重要なのです。情報は、商業紙などで十分に収集できるでしょう。しかし、「場」の提供は他のメディアや組織では不可能です。制約のできるだけ少ない条件で、技術に関する情報や意見の交換を可能にする「場」は、SEA のようなボランティアによって運営されるコミュニティだけが提供できるのです。

SEA のもう一つの重要な機能は、われわれソフトウェア技術者に、われわれの「まったくの自由意志による貢献」の機会を提供してくれていることです。それは、義務感からの貢献でもなければ、「ギブ・アンド・テーク」を成立させるための貢献でもありません。単に、われわれの「何か貢献したい」という自然な意志から、「私は、このことなら何かいえる」と思っていることについて、自分の経験・意見・知識を提供する。つまり、会議に参加するのです。「受け取る」のではなく、「与える」のです。誰かが話せば、ほかの人達は聞きます。その話は、聞いている人たちの思考に必ず影響を「与える」のです。これが、われわれの世界(知的なコミュニティ)における「貢献」なのだと思います。

## パズルとコンピュータ

勝田 祐輔

(日本ユニシス)

SEAMAIL Vol.8 No.4 に、玉井先生がお書きになった「あるパズル」に刺激を受けて、過去にいくつか関わりをもったパズルの中から1つを紹介してみようと思いました。実は、SEAMAIL にこの種の問題があってもよいのではと、以前から感じていました。それに、昔、子どもに買い与えた「穴あき迷路の本」の訳者が岸田孝一さんであったと知ったのは、そのずっと後のことでした。そうだ、編集者もパズルが好きに違いない。

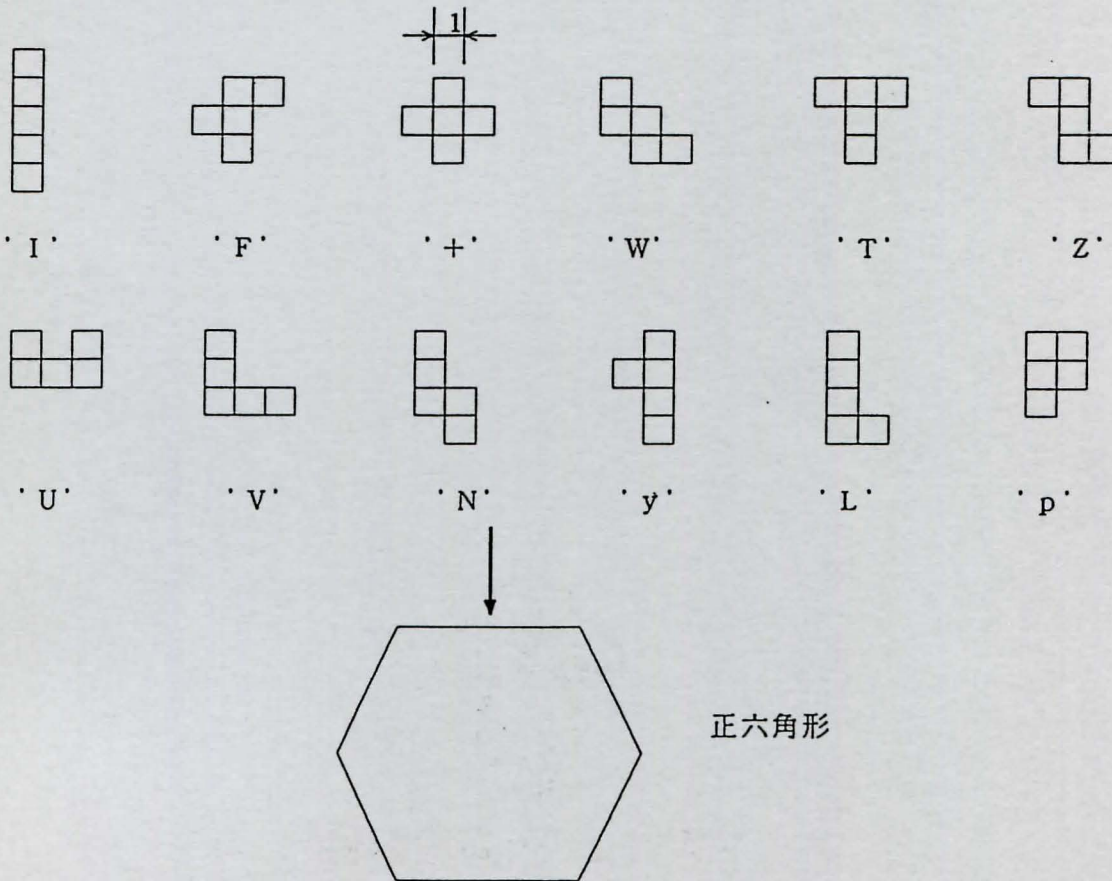
さて、技術力向上の一環として、毎年社内でプログラミング・コンクールが開催されてきて10年余りになりますが、SEA 常任幹事の山崎利治さんは、永い間コンクール課題の出題者でありました。筆者も過去回答を作って応募し、アルゴリズムや計算量について議論し、大いに啓発されたものでした。そして、自分でも何回か出題したわけですが、ここでその中から最近のものを紹介したいと思います。

**課題** 単位長1の正方形5個よりできる、いわゆるペントミノの12種類の駒(下図)を、適当な大きさの正六角形の中に、駒が互いに重ならないように収納したい。そこで、なるべく小さな正六角形の1辺の長さを求め、合わせて、駒の詰め込み例を示すプログラムを作成してください。

結果は、後日報告したいと思います。SEA 会員諸氏で、ご興味のある方はトライしてみてください。

[注1] 'I' ..... 'P' はそれぞれの駒の名称。

[注2] 十分に大きい正六角形への駒の詰め込み方は、無限にあります。







**ソフトウェア技術者協会**

〒160 東京都新宿区四谷3-12 丸正ビル5F  
TEL.03-3356-1077 FAX.03-3356-1072