



# SEAMAIL

Newsletter from Software Engineers Association

Volume 7, Number 3 1993

3

## 目次

編集部から	1
ソフトウェアシンポジウム'92 パネル討論記録	
ソフトウェアプロジェクトはなぜ失敗するか?	2
司会: 野村敏次	
パネリスト: 飯沢恒, 松村好高, 岡田正志	
分散化とエンドユーザコンピューティング	21
司会: 西条経治	
パネリスト: 岡田達郎, 橋山真人, 片貝孝夫	
LPF アピールをめぐる	新田稔 ほか 43
Call for Participations	
SEA Seminar & Forum (February '93)	44



# 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、1000 余名を越えるメンバーを擁するにいたりました。法人賛助会員も約 60 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： 熊谷章

常任幹事： 筏井美枝子 岸田孝一 玉井哲雄 中野秀男 深瀬弘恭 堀江進 山崎利治

幹事： 飯沢恒 市川寛 今別府芳暢 臼井義美 大塚理恵 落水浩一郎 片山卓也 亀田繁 君島浩 窪田芳夫 小林俊明  
杉田義明 坂本啓司 田中一夫 鳥居宏次 中來田秀樹 中谷多哉子 野村敏次 野村行憲 平尾一浩 藤野晃延 松原友夫  
盛田政敏 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)：小林俊明  
調査分科会 (SIGSURVEY)：岸田孝一 野村敏次

支部世話人 関西支部：臼井義美 中野秀男 盛田政敏  
横浜支部：藤野晃延 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：筏井美枝子 岩田康 鈴木智 平田淳史  
九州支部：平尾一浩 藤本良子

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 エスケーデー  
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 システムラボムラタ ジェ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  
ニコンシステム ニッセイコンピュータ ヒラタ・ソフトウェア・テクノロジー  
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 三菱電機北伊丹製作所  
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 日本データスキル 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 富士写真フィルム 富士通  
富士通エフ・アイ・ピー 富士通ビー・エス・シー クレオアールアンドディ SRA北海道 オムロン (以上 55 社)

SEAMAIL Vol. 7, No. 3 1993年1月25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

☆

どうした風の吹きまわしか、前号が出てまだ1カ月しか立たないのに、Seamailがまた出ました(!?)。

☆☆

去年6月長野のソフトウェア・シンポジウムにおける2つのパネル討論の記録を市川寛さん(電算)がテープから原稿に起こして下さったからです。2つ合わせてワープロで40ページを超える長編です。市川さん、どうもありがとうございました。

☆☆☆

SEAは、イベントの数だけでいえば、日本でもっともアクティブなボランティア組織だと思うのですが、記録のほうなかなか追いつきません。ぜひ、それぞれのイベントごとに、熱意あるボランティアの登場を期待します。

☆☆☆☆

ソフトウェア特許とユーザインタフェース著作権に関するLPFアピールについては、前号で回答者No.13さんからコメントをいただきましたが、この号では2人の方からのカウンター・コメントをいただきました。別にどうこういう結論がすぐに出るような問題ではありませんが、議論の盛り上がりを期待します。

☆☆☆☆☆

さて、次の号はいよいよ若手の会の報告特集です。SIGENVのみなさん、追い込み大変でしょうが、がんばってください。

☆☆☆☆☆☆

## ソフトウェア・シンポジウム'92

## &lt;パネル討論&gt;

## 「ソフトウェアプロジェクトはなぜ失敗するか?」

1992-06-10 16:00~17:30

長野県農協ビル アクティールホール

司 会: 野村敏次(日本電子計算)

パネリスト: 飯沢恒(三菱電機東部コンピュータシステム)

松村好高(SRA)

岡田正志(日本電気ソフトウェア)

## (司会)

それでは、時間になりましたので、本日最後のパネル・ディスカッション、『ソフトウェアプロジェクトはなぜ失敗するか』という誰がつけたか知りませんが、非常にストレートなテーマのパネル・ディスカッションを始めたいと思います。

ちょっとお断りしておきますが、諸般の事情がございまして、私はパネラーだったのですが司会をやらさせていただきます日本電子計算の野村でございます。よろしくお願ひします。司会をやる予定だった飯沢さんがパネラーになっております。それから、もうひとつ日本電気ソフトウェアの堀江さんがパネラーだったのですが、子供さんが怪我をなされて、今日手術ということでどうしても帰らなければいけないということで、同じ日本電気ソフトウェアの岡田さんに代わりをお願いしてあります。

最初にパネラーの方をご紹介したいと思います。皆さんに向かって一番左側が、三菱電機東部コンピュータシステム社会システム部の飯沢さんでございます。飯沢さんは、ずっと第一線で活躍しておられ、いろいろ失敗したプロジェクトがあるというお話しなので、そこら辺りをお話しいただこうと思っております。

それから、真ん中がSRAの松村さんでございます。松村さんは現在は技術本部という社内の

技術スタッフの部門におるのですが、過去10年位、実際のプロジェクトの保守、プロジェクト管理等に従事しておられました。

それから、堀江さんの代わりに岡田さんです。岡田さんは、日本電気ソフトウェアの技術管理部におられまして、日本電気の中央研究所、それからJSDの保守技術開発計画等に参加されまして、その後、ユーザー向けのシステム開発だとか品質保証とかプロジェクト管理等をやっておられ、ソフトウェア工学関係あるいはシステム構築に関する著書とか訳書を出されております。できましたら岡田さんに最後に講評でもしていただこうと思っております。

一応3名のパネラーの方々に最初15分位事例等含めましてお話しいただきまして、そのあと、会場の皆さんからの質問とか意見を交えましてパネルを進めていきたいと思っておりますのでよろしくお願ひいたします。

プロジェクト管理というのは昔から永遠の課題であって、プロジェクト管理でセミナーをやると必ず人が集まるということは、いつまで経っても解決しないということがあるからだと思うのです。プロジェクト開発においては経費節減とかいってタクシー代とかを節減しましても、プロジェクトが1ヵ月延びればそちらの経費の方が絶対にたくさん掛かっているのではないかと思います。そんな意味もありまして、

ちょうどいい機会ですからお話しされる方々の失敗あるいは成功など具体的な話を皆さんとして今後のプロジェクト開発に少しでも役立てていただければと思っております。

それでは、並んでいる順番に、飯沢さん、松村さん、岡田さんの順にお話しいただきたいと思っております。よろしくお願ひします。

### (飯沢)

飯沢です。当初、司会をやることになっていたのですが、いろいろ話してきますうちに、どうも私が今年に入ってから三つもプロジェクトを失敗して、いまだに後遺症で後始末に追われていると話したら、じゃあ、お前はぜひ司会よりもこの話をしてもらった方がよいということで、パネラーにまわりました。

先程、司会者の方から経費節減ということいろいろやるよりもプロジェクト管理をきちんとした方が経費を節減できるという話がありましたけれど、当社も、今はやりの時短ということでいろいろやっていますが、検討してみますと、プロジェクトが失敗しなければ1800時間は多分軽く達成できるのじゃないか、という結論になりました。

さて、失敗とは一体何だろうかということから入ってみたいと思います。“プロジェクト管理における失敗とは”ということで、失敗の現象ですが、これは、社外的には、納期の大幅な遅延というかたちになって現れてきます。それから当然、そういう時には品質は保証できないということで、システムの信頼性とか品質が大幅に低下してしまうといったことからお客さまからの信用を失墜させてしまうというようなことが、外には出てきます。

社内では、当然のことながら、予期せぬということから予算が大幅に超過してしまい原価割れということで、赤字になってしまいます。それだけですれば勉強代ということになるわけ

ですけれど、組織やプロジェクトはどうかというと、チームワークが崩壊してしまっていて、終わった時に、病気や精神的なダメージを受けて、会社へ出て来ないという者も出てくるということが起こります。

さらに進みますと、長い間育てたエンジニアが退職してしまうというような結果になりまして、将来の発展性が減少してしまうというような事態になります。私どものところ、三菱電機全体なんですけれども、こういうような状況を、崩れてしまったプロジェクトということで「プロジェクト崩れ」と呼んでおります。

さて、私どもの会社ですけれども、重電機だけでは発展性はない、コンピュータシステムを含めてやれば、いろいろと売り上げも伸びるだろうということから、ソフトウェアを扱うようになりました。

ソフトウェアを扱ってしばらくは、確かに売り上げがどんどん増えたのですけれども、どうもソフトウェアは作れば作るほど利益は減ってくるというような状況にトップの人が気がつきまして、グラフを作ってみますと、本当に反比例の関係で落ちてきているのです。これは一体どういうことだということ、いろいろ調べてみますと、プロジェクトの失敗コストというのですが、それが異常に大きくなっている。要するにちゃんとやっていたら特に問題はないのですけれども、かなり大きなシステムを手がけていますから、1回失敗すると何億といった損失が出るわけです。

そういったことから、ロストコストが非常に大きくなるということで、プロジェクト管理を何とかしなければいけないというようなことになってきて、永遠の課題とさきほど司会者の方が言いましたけれども、そういったことから今プロジェクト管理ということが内部でいろいろと取りざたされているのが現状のところなんです。

失敗の要因を、私なりにいろいろ分析してみたのですけれども、一つは、我々開発者サイドにいますと、受注のときからまずダメだというのが分かっているのがかなりあるのです。これはリーダーがいくらしっかりしていてもダメなのです。

どういうことかといいますと、背伸び受注ということです。無理して取る、いわゆる政治的に取ってしまうということです。到底無理な納期で受けてしまう、原価をどうみても最初から赤字と分かっているのに受注してしまいます。それから、品質面でいうと、技術的にかなり困難だと分かっているのですけれどもやらざるをえない、そういったことで背伸び受注をしてしまう。最初からある程度無理を覚悟でという受注があるわけです。

それと担当者の見積もりミス、見積もりを担当した人のミスというのがありまして、ユーザーの真のねらい、これが把握できない、つかめない、そういったことから、違った観点から見積もってしまう。それから技術的な実現可能性いわゆるヒュージビリティというのですけれども、そういったことの検討が十分なされないまま見積もりとしてしまう、そのために失敗をするという、これは受注の時の問題です。

それから、受注が決まってしばらくして打ち合わせの後、プロジェクト計画というのを立てるわけですが、計画の時点で体制確保が困難というのが割合と多いのです。

これは、予算化が最初から不足しておるので、これだけの予算だとこれだけの人しか確保できないと、上からそんなに大勢じゃ赤字になるからダメだと言われて、泣く泣くダメだと分かっているながら人が確保できないという事態がある。

それから、ノウハウの保有技術者、新しいことをいろいろやるとき、ノウハウを保有した技

術者がほしいわけですが、どうしてもそういった人がまわってこないということで、知らない者同士が集まって不安に思いながらスタートをするというような状況がある。

それから、非常に忙しい時ですと、自分のところだけでは全部できないわけで、一部外注に出すわけなんですけれども、優秀な良いソフトハウスというところがなかなか確保できない。ずーっと継続的に発注すればいいのですが、そうでない時にスポット的にいくと、なかなか発注先が見つからない。ギリギリまで探してみても今まで面識のないところへ、やむなくという形で発注をするというような状況になる。

それから、計画時点で作業量及び作業時間の見積もりというのがあるのですが、見積もり技術の不足というのですか、これは全く新しいものを見積もるときはある程度のリスクというのはやむを得ないと思うのですけれども、人によって非常に楽観的な見積もりをしてしまうという人がいて、彼の見積もりでは、いつも時間が足りなくなっていて最後にオーバーするといった個人による特徴があるかなあという感じはします。

それから、作業と実際に担当する人の能力、これがはっきりと掴めていけばいいのですけれども、このぐらいのことはできると思ったというような言い分けがきますが、作業をする人の能力を十分に掴んでいないため、時間とか作業量は見積もれたとしても、作業時間の見積もりでミスを犯してしまう、作業時間が狂えば当然できると思った時間内にできないわけですから、納期が遅れるといった事態になります。当然、原価も上がってくる。

3番目に、いよいよスタートして、実際にプロジェクトが運営されるわけですが、そういった時点でプロジェクト・リーダーの能力不足というのがひとつあります。これは、いくつもプロジェクトがあると、なんとなく不

安に思いながらも、もういないからしょうがない、アイツにやらせてみようかということプロジェクトリーダーを任命することがあるのですけれども、管理者の評価ミス、そうでなくて、彼は大丈夫だろうと思ってやったところが意外とうまくいかなくて失敗してしまう。管理者がうまくその人の能力を評価していなかったというリスクです。

それから、リーダーの意志の不足というのがあります。これは意外と多くて、実際には、大ざっぱなことは決まっていますけれども、スタートしてから仕様を詰めていくといったことが多いわけで、客先に対して主体性というのですか、自分がリードしてスペックを決めていくといった主体性がないと、お客さんが決めてくれないからしょうがないということで、なかなか仕様を決められない。そういったリーダーもいる。そういったことが、後の工程に全部シワ寄せをしていって、結果として遅れるということになるわけです。

メンバーへの動機づけが不十分ということで、リーダーとして任命されても、メンバー何人かに目的だとか意義とかそういったことをきちんと伝えていないと、協力してやろうといった意識がプロジェクトの中に持てないというようなことがあります。

今まではリーダーの問題でしたけれども、メンバーはどうかといいますと、技術の不足ということで、新しいものに対する拒否反応、これは年の若い人ほど新しいことに対する拒否反応といったものがないのですけれども、何年もやっているとなんか新しいものを受け入れることがなかなかできない。柔軟性がないということで、いやがって逃げてしまうというのですか、ギリギリまでなかなか手をつけないという拒否反応があります。

それから、いずれ進行していくかたちで、新しい言語を習得していかなければいけないとい

うことは分かっているのですけれども、自己啓発意識の不足というのですか、なかなかギリギリになるまで勉強しない、これは日本人の特徴で、受験勉強とか受験戦争とかそういうことで、ギリギリまで遊んでいていよいよいけなくなってから勉強するといった身に覚えのある方もいると思いますが、そういう傾向があります。

それから外注するときが一番気をつけなければいけないのですけれども、管理不十分というのが意外と多くて、発注してできあがってくるまであまり干渉しない、中のチェックをしてないのです。要するに相手が中の仕様をどのくらい理解しているのかその辺の確認をしてない。レビューをきちんとしていけばいいのですけれども、それをとってないために、できあがってからこんなはずではなかったということで、時には作り直すということがあります。

それから、これもそうなのですけれども、中間の現物確認というのですか、これをしておかないために、工程上では、仕様書できました、プログラムできましたとくるのですが、いよいよ動かそうとするときに、ここではは試験が終わっているというのはただ単にコンパイルが終わっている段階だというようなことがよくありまして、やはり、ためっばなしで中間でのチェックが不足していると非常に危ないということです。

それから、いくつかに分けてやりますと、全体の整合性、その辺をきちんとしておかないと、インターフェースのところでは必ず不整合がおこって実際に組み立ったときに動かなくなるということがよくあって、これが失敗の工程遅れの要因になっているという事態になります。

一方、内部的にはそうなんですけれども、開発側からみますと、お客さんというのはわがままなんで、やっぱり問題があります。基幹業務というのですか、その会社の基幹の業務ですと

その辺のところはよく分かっていますから中規模ですと問題はないのですが、大規模になると非常に危なくなってきました。というのは、いくつかの社内の関連部門の調整が非常に難しくなってくるということで、窓口も力のあるところでないとその調整ができなくて、結局は仕様がいつまでたってもお互いに整合がとれなくて決まらない。

それから、大きなシステムになりますと発注先がいくつかありまして、インターフェースの調整というのが非常に困難になってくる。いずれも窓口の技術的な能力とかあるいは責任とかの意識の問題が非常に大きくなってくるような感じがします。

当然、責任担当部署があいまいになると、このところからみな逃げてしまって、お任せになって、「動けばいいんだから、いいようにやってくれよ。」というかたちで投げられてしまう。結果的に調整をこちらがしなければならなくなって、仕様が決まらない、システム構築能力も当然不足しているということで、「本当にこれで動くの?」ということがステディーされないままスタートしてしまう。結局、仕様が決まらないのに、後の納期だけは決まっている。ですから当然納期通りにはできない。できても粗雑なものになってしまうということになります。

最近支援業務の仕事が多くなってきていますけれども、経営支援とかOA・エンジニアリング、オフィス・オートメーション、EOAといったような非定期的業務がありまして、これは又、基幹業務とは様相を異にしております。具体的なイメージというのがはっきりしていないところがあるのです。

要するに流行だからなんとなくやってみようというところがあります。これは、メーカーの営業にも責任があるかもしれません。ハードウェアを売りたいためにシステムをなんとか売

るのですけれども、何をやりたいのかお客さんとはっきり分からない。現場との関係で、仕様の変更とか改善はあって当然だという考え方で。分からないんだから、あとは改良だと。それから、当初いろいろ言うのですが、イメージがだんだん話をしていくうちに拡大してきて仕様が発散してしまう。

当然、開発予算でできなくて金額も掛かってしまう。それから、現場、実際に使うところと導入担当部門というのですか、システム部門というのがあるのですが、その意識がずれて、使わせる方は、「これでいい筈だ。」というのですが、現場ではこんなものは使えないということで、納めてからそういったことをやってるユーザーが非常に多い。

それから、仕様について責任をもっていない。要するに具体的なイメージというのをもっていないわけですから、これでいいというのは誰も言えないわけです。要するに承認をしてくれないわけです。「動いてから承認します。」というような形になって結局は仕様が決まらない。これで納期が遅れてしまい、プロジェクトは予定どおりにいかないということが非常に多くあります。

我々は、それに対してどういうふうにしようかということで考えているのですが、ここで少しなんとかうまくやりたいと思い、「早期実態把握プロジェクト」を考えました。受注以前のものではしょうがないとしても、受注した後は管理者あるいはプロジェクトリーダーの能力に負うところが非常に多いということで、そのプロジェクトを早期実態把握をして問題点を早期に解決する、リスクも事前に分かっていたら回避できるわけですから回避する、あるいは軽減すると、そういったことを考えています。

仕組みとして診断チェックリストというのを作りまして、管理者がプロジェクトを診断し、それでおかしければ指導する。プロジェク

トリイダーも同じチェックリストを使って自己診断をやる。それで診断結果としてどこか悪ければ改善をする。こういったようなサイクルで回す。そのためのチェックリストをつくらうと考えて、今、だいい進行してきています。

これは医師の健康診断というように考えてもらえばいいと思うのです。プロジェクトの健康診断というイメージで、病気になってから治療すると、やはりお金も掛かるし時間も掛かる。だから定期健康診断みたいなことをやって早めに見つかればちょっとした治療で済むのではないかということで、そのためのノウハウの蓄積、活用、それともう一つ合わせて、プロジェクトリーダーを育成する。そういったことでやったのですけれども参考になるのではないかなと思います。

「ソフトウェアの開発における問題点」ということでプロジェクトリーダーあるいは管理者に何人かピックアップしましてアンケートを記述形式で集めました。我々のところは12段階の作業工程を定義しているのですけれども、それに対してどういうところに問題があったか、問題点をリストアップしてもらいました。過去に発生した問題点、あるいは今やっているもので発生した問題点、対外組織的な面とか品質・費用・工程とかで、どこか気がついた点など、いろいろと集めました。

それをもうちょっと整理したかたちで診断チェック項目としてまとめました。最終的にプロジェクト管理として、我々のところでは10の管理を定義しているのですけれども、この診断チェック項目を、もう少し細かくそれぞれの管理項目に分けて分類し整理しました。これからチェックリストを作って診断に使っていかうかと考えています。今たとえばシステム設計の段階だったならば、こういったことについてはちゃんとしていますかというような診断の項目にしようと思っています。今は、その途中の段階です。以上です。

### (司会)

どうもありがとうございました。ご質問とかご意見とかあるかと思うのですが、最初に三人の方に発表していただきまして、その後、皆さんのご意見・質問を受けたいと思いますので、もしありましたらメモ等をちょっととっておいていただきたいと思います。

それでは、松村さんお願いいたします。

### (松村)

SRAの松村と申します。よろしくお願ひします。当パネルに参加するにあたりまして、飯沢さんの方から三つ程事例を考えてくるようにということで数点にまとめて整理はしたのですけれども、まとめ方がどちらかというとなマネージャーサイドということで分析させていただいたのと、表面的な分析しかしてありませんということです。一部生々しい事例を申し上げますので、関係する方がいらっしゃったらご勘弁ください。

プロジェクトにおける失敗とは何かを考えるというお話なのですけれども、どのような現象があらわれたかということ、一つはどのような結果で終わったかというふうに眺めてみて、ちょっと挙げさせていただきました。私の経験の範囲ですので、これ以外のことで失敗もあると思います。

一つは、何度も何度も経験するようなスケジュールの遅延とか開発コストの増大、それから実際はつくったんですけれども運用するにあたって不適確なものです。それから、極論しますと開発作業自体が途中で中断してしまったものです。最後にプロジェクトが終わった後にマネージャーが痛い思いをしますエンジニアの人たちの離脱ということで、事例と要因をあげさせていただきます。ちょっとさしつかえがありますので3年から7年前の事例で述べさせていただきます。

これは私が所属したのが事務処理アプリケー

ションシステムを開発するセクションでしたので、そちらの方のシステムの事例ですが、スケジュールの遅延です。半年ぐらいスケジュールが延びてしまいました。

これはワークステーションを使いまして、我々の作業に近いSEの方たちの作業を支援するシステムです。大きくは3つの機能を実現したわけですが、一つはデータベースと、それからマン・マシンインターフェースと、それからプログラムを解析する部分、3つの部分です。何回スケジュールの延長を行ったかと言いますと、3回です。3回にわたって約半年、当初の納期に対して半年延びてしまいました。当然スケジュールの遅延というのは、開発規模も増大するのですが、最終的なのはおよそCで20万ステップ程度のものでした。3回とも実は私が延ばしてくださいとお願いしたものです。

大体三つぐらいの要因で述べさせていただきますけれど、要因の一つは、現状分析の段階で調べた内容が調査から2週間後が変わってしまったということです。それから、もう一つは生産性が達成できなかった点です。事務分野ということで、比較的C言語とかX-WINDOWSとか未経験でしたので、そのトレーニング不足があったかと思うのです。もう一つは、この3つの機能のうちの一つだったのですけれども、設計を担当したSEの人の興味に走ってしまったので若干オーバーステップのまま作ってしまったことです。

つぎはコストの増大の事例です。3割程度でなくて非常に極端にコストが掛かった事例を申し上げます。これは、私は実際の開発時には担当しなかったのですが、見積り時に担当していましたので、事例として申し上げます。

これは汎用機上のPL/Iで、なおかつ、地方の金融機関向けの第3次オンラインシステムでして、バッチの部分とオンラインの部分に分かれていました。バッチの部分をも他の金融機関で開

発されましたソフトウェアを使って、オンラインの部分新しく作ろうといったようなもので、ちょっと間違いがあるかもしれませんが、記憶をたどってみますと、当初私が見積もったときには、13億か12億だったと思います。先程、飯沢さんの話にもありましたけれども、無理な営業で受注してしまうという傾向がありまして、いろいろ値引いていった経過は若干知っているのですが、最後にはなんと4億で受注してしまいました。いろんなエンジニアが努力をして完成はしたのですが、実態はおおよそ15億ぐらい。ですから当社の損益に対する金額は交通費とかそういった費用ではまかなえないような費用だったのです。これは過去に終わった事例なので申し上げます。

どんな要因があるかと申しますと、そもそも見積りもりのときに誤ってしまったということです。「本当にそれで受けるの?」という話はしていたのですが、SIという言葉がその当時出始めた時だったのでなんとかSIの実績を作りたいということで受注してしまったと思われます。その意味で営業の人たちには何の反省もなかったようです。このシステムの受注の過去を振り返ってみて、そういったことでした。

それから生産性の上には立っていなかったということです。その場にも実際スケジュールのキャッチアップをするために行ったのですが、何と言いますか、検証ツールが非常に不足しておりまして、オンラインツールの検証をわざとアベンドさせてコアダンプをとってそれを調べているといった、私たちの若いころやったような検証の仕方をしておりまして。

それから、思い起こせばということなんですが、当初一部開発されていた部分を使うということでスタートしたシステムですので、移植もしくはカスタマイズというところから入ろうとしたのですが、意外と、存在するものを使ってやるというケースの場合、往々にしてプログラムというか対象とするシステム自体の理

解に欠けて、テスト工程などで実際にテストデータが起こせないという痛い目にあった経験があります。

つぎは運用するにあたって実際のターゲットの環境に合わなくて手直しをしたという事例です。

手直しができたのでこれはまだ軽症かもしれませんが、どんなシステムかといいますと、PCとワークステーションをLANでつなぎまして、ローカルな業務作業を支援するシステムです。ハードウェア、LAN、RDB、それからアプリケーション開発、おのおの別会社で、当社はリレーショナルデータベースのソフトウェアとアプリケーションシステムの開発を行ったシステムです。実際のプログラム開発は、PC98の上でのMS-DOSでのCのプログラムの開発でした。およそ7万ステップ程度です。

要因の一つは開発環境の不適です。開発そのものをCでやらなくてもいいのではないのという話もあったのですけれども、実はRDBを売っている部門と、開発部門のケンカになってしましまして、RDBシステム自体を売る側はオーバーヘッドがかからないようにPC上でプログラムを開発するような形のソフトを販売をしてしまったということです。

開発者側は当初からワークステーションでアプリをのせようという意見でしたが、そういうことができなかったということです。

これは一番の大きな欠点で、PC上にプログラムをCでゴリゴリ書いていますので、680Kバイトというエリアの実効定数を大幅にオーバーフローしまして、最後にはプログラムを書き直すという、これは半分ぐらい開発したところで気付きました、それ以降には影響を与えなかったのですけれども、前半をカットオーバーする部分で実行機に移すところでもう1回書き直したということがありました。

それから、テストまで当社内で同じような環境がありましたので、行なっていたのですけれども、お客さんのハードウェアの構成および仕様・ソフトウェアの条件と当社その条件が違いまして、再三再四、再現できない不可解な現象に悩まされたといったことがありました。

つぎは、一番重いと私は思っているのですけれども、採算面というか、恥ずかしい話なのですが、開発作業を途中で中断したということもあります。

これは、やってみないかということをお願いされたシステムだったのですけれども、どういう開発かといいますと、汎用機、バブルがはじける前の年代のものですから、こういうことをやっていたのですが、汎用機上でPL/Iでおおよそ100万ステップくらいの「債券デリングシステム」を他行へカスタマイズさせるというシステムです。だいたい受注金額が2億5千万ぐらいで、当初見積もった工数が250人から300人/月です。結果はどうだったかといいますと、単体開発の途中で中断という状況になりました。

要因は何かということはいろいろ考えるのですが、そもそも営業方針、これは前の事例でもあるかもしれませんが、営業方針と開発側の方針の不一致があります。開発側というのは私だったのですけれども、私は売られる側の開発の方に従事してしまして、そのシステムの品質がまだやわらかいレベルであることを知っていました。したがって作業が大変になると合わせて、当社がすべて開発したのではなく、他の会社の方たちがバッチの部分を作っていましたので、バッチの部分が私たちには分からないということがあって、作業量をなんとか増やそうとする営業からやれと頼まれましたが、最終的には私はこれを断りました。

断った結果、どこの人たちが作業をやったかといいますと、全面的に協力会社の方へ営業が委託をしました。そちらの方で作業に着手をし

たわけです。それはいいんですが、そもそも従事された人たちがこのぐらいの規模のシステムに対して理解していないということと、そのカスタマイズしようとするフロム側のシステムがどういう状態にあるかということをよく知らない。それからカスタマイズするといっても、ユーザー要件がそんなに明確になっていないなどの条件を合わせて、作業をもう少し地道に考えればよかったんですが、先程も申しましたように、作業量が多い方ということで受けてしまったのです。

途中チェックに入ったのですけれども当社の管理者はおりませんでした。聞いてみると一括発注なんだから、あちらが悪いという言い逃ればかりがあったということ。それから、単体開発に入ったときにもチェックをしたのですけれども、その時点ですでにコストは2億ほど使ってしまったということです。

これを中断したというのは非常に勇気のいる決断だったと思います。これは私ではなく、私の上司にあたる方がそういう決断をされたんですが、あとどれくらいかかるんだろうということを知られた記憶があります。そのときに、私は約1億から1億5千万はかかりますと言ったときに、2億を使ってさらにそのくらいかかるのであれば、傷が残らないうちにやめようという決断をされました。非常にまれにみる勇断ではないかと思っております。

最後に、最近、私に一番こたえる事例を述べたいと思います。それは開発途中のエンジニアの方たちの離脱ではなくて、開発を終わった後にエンジニアの方々、社員及び協力会社の方たちが2度と付き合ってくれないという状況になると非常に悲しいということです。

これは先程申しました、PL/Iの債券のディリングシステムの後エンジニアの方たちがどうなってしまったかということをお願いしたいのですが、2年間でおおよそ60万ステップを設計

からテスト完まで非常に過酷な条件の下で行いました。その後そのチームで別のシステムを二つほど作って、3年間50万ステップ程のシステムを開発しました。さらにその後3年ほど経ったら、当初のシステムで経験してきた社員の人たちはさびしいことに三分の一になってしまっていたということです。

ここで言いたいのは、徹夜状態が数カ月も続いてしまうという作業を経験するとか、納期が非常に厳しいなかで作業をする人たちに対して何らかのリハビリをしないといけないんですけれども、それを引き継ぐことなく終わったという反省があります。

それともう一つ、うまくカットオフができたということで、過酷な条件の中で働いてきた人たちの成果でカットオフができたのですけれども、そういった形のシステム開発でなくて、別のもう少しキチッと手順を踏んだような成功体験を彼らに積ませることができなかったという反省があります。こういったことが非常に失敗であったと思っています。

成功するための課題ということでデータを集めようとか、プロジェクトの特性をキチッと把握しよう、ソフトウェア工学的なルートを開発しようとかありますけれども、やはり失敗の事例の中にノウハウがあるわけですので、それをうまく後輩の人たちに伝えるのが一番のキーポイントかなと思います。思いますがやっぱり、一番はすぐれたマネージャーをどうやって育成しようかということに行き着いてしまうのではないかと思います。

ピーター現象というのがあるんだそうですけれども、ある職種の人を別の職種に、あるいは職位と考えるべきかもしれませんけれども、変えると、前の職種では非常に優秀な人が実は凡能の人になってしまうという現象があるということは何らかの本で読んだので、きっとマネージャーにも不適確な人がいるんだろうと、

では非常に優秀ではあるけれどももと思っています。そういったことをちゃんと認識していかなければと思っています。

マネージャーとエンジニアの視点というのが違うんじゃないかと最近思い始めてまして、エンジニアの場合には非常に論理的に0と1がはっきりした視点でいいわけですが、マネージャーの視点になりますとグレーがあったゾーンのなものを見方をしないと、ちょっと問題が起きたときとか、たくさんの人と作業をするときはなかなかうまくいかないのではないかなというふうに思っています。

マネージャーになるには、やっぱりマネージャーとして非常に優秀な人を見習うのが一番早いですけれども、私はSRAに入りまして職种的にマネージャーになったところで、本当のマネージャーではないんですけれども、残念ながら、当社にはすぐれたマネージャーを見出すことができなくて、実際はお客さんのマネジメントをされている方を見習わせてもらった経験がありますので、もしも身近にいない時には、どこか別の世界の人を見習った方がよっぽど早いかなというふうに思っています。

それから最後は、やっぱり良いマネージャーとかすぐれたマネージャーになるためには、過酷な失敗ではなくて適度な失敗のトレーニングを多数重ねることが必要だと思っています。

以上で終わらせていただきますけれど、私は決して優秀なマネージャーではないと思っています。

#### (司会)

はい、どうもありがとうございました。  
では岡田さん、お願いいたします。

#### (岡田)

岡田です。先週末、出張先の方へ、失敗の経験があったらそれをひとつ披露してくれないか

と、こんな話がありまして、急遽、来たのです。とはいうものの失敗の経験というのはいくら考えてみてもあまりありません。本来、失敗しているのかもしれないけれどもあまり失敗とは認識してないのかもしれない。幸い、私の周囲にいっぱい失敗がありまして、その失敗を分析したり調整したりとか、そういう役目もしていますので、そういった経験からいろいろ今日はお話をしていきたいと思えます。

ちょっと失敗の観点が二人の方と違うかと思うんですが、最終的には失敗というのはどういうところかといいますと、最近、顧客満足度いわゆるCSという話が出てきて非常に注目されているんですが、やはりCSというのがきちんと達成されなということが失敗ではないかと考えております。これは、社会的な風潮からいって、企業主導型の社会からユーザー主導型の社会へと変わってきていることが大きなポイントではないかと思っております。

そういうことになりますと、一番重要になるのが、会社とする契約ということになってくるのですが、その契約が非常にはっきりしていないケースが多いと、そういうところでいろいろユーザーとの問題というのが入ってくるのではないかな。そういう意味で、スペック上とか見積もりとか、その責任の分担、その辺をどうするか、こういう問題がいつも最終的に問題になってくるケースが非常に多いと思えます。

そういうシステム上のトラブルというのいろいろあると思いますが、やはり、実際的にいうと、そのシステムが使われないとか、実際に使ってはみたんだけど、どうも導入効果が無いとかということがあります。それは無論、開発納期の問題とか、スケジュールオーバーとかいろいろな問題が発生するのですが、結局、システムを導入したのだけれどもうまくいかないというのが、本質的な失敗ではないかと考えております。

そこで、こういった問題からそういったことが発生してくるだろうかと、システム開発のリスク要因、そしてどういうものがあるかをきちんと分析しておいて、かつ、これを契約書上でちゃんとしておくということであれば非常に問題が少なくなるというようなことが考えられるわけです。

要求分析上でいえば、具体性が不足しているとか、実現不可能だとか、そういった問題があって発生してくる。それから、実際に開発に入ったときも、見積もりからのスケジュールオーバーみたいな話、それから、開発要因の問題とかいろいろな問題が発生してくる。そういうリスクがどういう点にあるかということ进行分析するというのが非常に重要ではないかと思えます。

それで、実際に、プロジェクトをいろいろみているんですけども、非常に多くのプロジェクトが、なんとなくスタートしている。これは、ユーザーの方のシステムについての考え方がはっきりしていないとか、いろいろな要因があるとは思いますが、プロジェクトが不明確な状態でスタートしてしまっている例が非常に多い。それで、計画とかそういうものをきちんと作ってない。しかも、そのプロジェクトにどういうリスクがあるかをきちんと分析したり管理をしてない。

私どもの会社ではある金額以上はプロジェクト・レビューという形で、きちんとチェックをするようにしているんですけども、実際的に小さなプロジェクトですと、こういうことをしないケースが非常に多いということで、何となくスタートしてしまっている。

それから、いろいろな問題が発生するのですけれども、そういった問題を、自分の責任という観点でとらえるということが非常に少ない。いろいろ話を聞いてみますと、結局、これはこういうふうユーザーが言ったからこうなったん

だとか、いろいろな問題に帰着している。じゃあ、どうしてそのへんをきちんと話し合わなかったのかという自分の責任という観点でモノをとらえるということが非常に少ないのではないかという気がします。

プロジェクトをスタートするというに関して、初歩的な問題、そういうものをきちんと把握しているかということ、その辺を把握せずに未解決のままスタートしてしまっている。それから、具体的施策をきちんと作っていないということです。これは、非常に抽象的にこういうことをしていこうというだけで作業をスタートしてしまっている。

それから一番大きな問題は、やはりヒューマン・ファクターの問題だと思うんですけども、ヒューマン・ファクター問題というのは、多発しております、こういうのも、私どもの中では若干いろいろチームを組んで、ヒューマン・ファクター問題を検討するというようなこともしております。

その前に、ヒューマン・ファクター問題のポイントみたいなことを簡単に述べさせていただきます。

基本的に、感情的となった問題はなかなかうまく收拾することが不可能だということです。それから、自分流の管理方式をきちんと見出ししていくスタンスが必要で、やはり、上辺の管理をやるだけというのはどうもうまくいかない。

それから、その人の体調とか、その辺まで気を配る。それからストレスの問題です。自分の尺度、価値感というものを人に押しつけているケースというのが非常に多い。その場その場で毅然としてやることをやっていくと、そういうことも一方では必要になってくる。

これは非常によく言われるんですけども、ま

ず、うまくいかないと思ったことはうまくいかないということ。

リーダーの悪いところ、その辺は敏感に周囲の人に伝わる。それから、一旦信用を失いますと、そういったものを復旧するのは非常に難しいとか、いろんな人が集まって仕事をする、そういう中で、若干いろいろな問題を起こす人もあるわけですが、そういう人たちに対してどう接するかというあたりが非常に大きな問題ではないかと思えます。

若干、例をあげますと、たとえば、非常に経験の少ない若年層の人が集まったプロジェクト、これは某プロジェクトと言っておきますけれど、その例を分析したものなのですから、プロジェクト要員が非常に不安定だということ。これは、その人が未経験であり知識が少ない、それから、責任の重大性というものをちゃんと把握してないという状況にある、それから安定感がないとか、非常に地方でやる仕事が多い、長期出張があるということで、その辺で不安定さが増強されている。不安とかプレッシャーからの焦りというものがあるというトラブルが発生しやすかったというようなプロジェクトだったのです。

そこで、リーダーはどのような解決法をしたかということですが、やはりリーダーとしてきちんとした姿勢をとっていかうということで、一貫したプロジェクト方針をもって接するというようなやり方を取って行った。それから、メンバーの育成、その辺にもストーリー性をもって対応した。一方ではそれぞれの人の個性というものを尊重すると、そんなようなことをして失敗しそうなプロジェクトが問題なく終了しました。

もう一つ、これはいろんな会社の人が集まってやったプロジェクトの例で、この中で人と人との軋轢が非常に大きな問題になった。しかも、いろんな会社の人が集まっておりますの

で、そういった意味で統一的な管理をしていくのが非常に難しいというケースでございませう。それがために、個人的なプレーが非常に増強されてしまった。従って、ヒューマン・ファクター問題に非常に踏み込んでいくという姿勢がとれないというような環境であったのです。

そこで、どういうことをしていったかということ、作業分担を非常に詳細に決めていった。それから、仕事に対して、一つの仕事をやるにも、一つずつ説得をしながら進めていった。それから、基本的には平等になるように、そういう対応をしていった。そういったようなことをやりながらなんとか終了させたというものです。

その他にもいろいろ用意してきましたけれども、時間がございませぬので、ちょっととばしまして、基本的にはプロジェクトを成功させるためには、二つ大きなポイントがあると思えます。

まず、リスクマネジメントをきちんとするということです。これは、きちんとプロジェクトを分析する、そういうスタンスがいる。それからもう一つは、目標というものを作ってその達成のマネジメントをきちんとやっていくことが非常に大きなポイントじゃないかと思えます。それをもう少しかみくだいて言いますと、いわば、マネージャーとしていわゆる支援型の管理ではなくて、自分が積極的にマネジメントをしていくという姿勢をとるということ。やはりプロジェクトごとに勘所みたいなのが当然あるので、その辺をきちんと把握しながらやっていく。あくまでも、現場志向でモノを考えていく。それから最良の選択、カスタマイズとか、いろいろあると思えますが、基本的にはマネージャーがきちんとした方針をもってやっていくことが非常に重要ではないかと思っております。

同じことをもう少しかみくだいて言うと、基

本的には人の問題です。仕事をきちんとやることと、きちんとした分析をしながらやることが重要だろうというように思われます。

#### (司会)

どうもありがとうございました。

飯沢さんと松村さんにはご自身の体験等を含めましてプロジェクトの失敗の要因、それから、その解決策的な話をしていただきまして、失敗のない岡田さんには、まわりの話と、ヒューマン・ファクターの問題、それから、解決策についてのお話をいただきました。

多分いろいろご質問があるんじゃないかなあと思うのですが、会場の皆さんで、もしご質問・ご意見等がありましたら、遠慮なく手を上げていただいて、言っていただきたいと思えます。ございますでしょうか。はい、どうぞ。今マイクが行きますので、申し訳ありません。最初に会社名と名前をお願いいたします。

#### (坂本)

オムロンの坂本です。よろしく願いいたします。いろいろおもしろい話をお聞きしたのですが、お三方ともお話の中で、私が非常に気になっているところがお話に出なかったので、そういうところはないのかと、そういうご意見をお聞きしたいと思えます。

プロジェクトが失敗する時には、失敗するときのストーリーというか、時代劇の定型版みたいなストーリーがあって、その中の一つに、実は中味は失敗しているんだけど、それが表面化していなくて、くすぶっている状態というのがずっとあって、かなり最後の納期に近づいたところでポッと火を吹く、いわゆる狼男に変身するという、そういう瞬間がほとんどのものにあると思うんです。そういうことがあるのが定型版のストーリーだということはどう思われるかということと、もしそうだと思うれば、なぜくすぶり続けるんだろうかという

のが一つ。

それと、普通の人間が狼男になった瞬間に、実は他のところから見て、「アッ、狼男や」と言ったときの、そのときの狼男の認識のレベルが、少し経過すると変わってくるのが多いんです。最初は、ちょっと困った奴やなあというくらいの狼男で済んでいると、しばらくして手当てしているうちに、本物の狼男になって、もうクタクタになるようなことになる。いわゆる何か問題をもった時に対する対処の仕方のパターンが二段構えというか、一発で済まない。一発かましたけれども、その次に本当の山があったということが多くあるんじゃないか、その辺についてのご意見をお聞きしたいと思えます。

#### (司会)

はい、分かりました。今二点質問がございましたが、その点に関してどなたからでも結構ですのでお願いしたいと思います。もう一度まとめますと、実際にプロジェクトの失敗が表面化する前に長い間くすぶり続けている、それはなぜだろうという話が一点だと思います。それから、もう一つは、実際に火を吹いた時に、担当者というか、当事者の方が、最初、それから時の変化とともに変わってくるのではないかということですね。それについて、どなたからでも結構ですのでお願いします。

#### (飯沢)

最初の質問のくすぶっている状態ですが、これはやはり私も経験があるのですが、二つ問題があると思うのです。一つは管理者の問題だと思うのですが、自分の気にいらぬ方法というのをいやがる傾向があります。そうすると、下の人は隠してしまうわけです。どうせ怒られるのだったら最後に1回おこられようということになりますと、途中で報告して、あれ出せこれ出せとか、あるいは、なんでそうなったとか責任を追及されて実際の仕事にリストアされちゃうということのを嫌いまして、実

際、話によく聞くのですけれども、「もうあの人に言えばいつも怒られるんだから、怒られるんだったら最後に1回、納期の間際におこられる方が楽でいい。」と、そういうリーダーがいるわけです。

もう一つ、リーダー自身の性格として、なんとかリーダー自身の力でなんとかしよう最後まで頑張ってしまうわけです。そして、いよいよどうしようもなくなった時に、万歳してしまう。そういう二つの傾向があるのです。なんとかしようと思っているリーダーがいるのはいいと思うのですが、問題なのは自分で見通しがないうちもあるわけですから、上の人がよく見ていなければいけないということです。要するに問題が二つあるのです。管理者の問題とリーダーの問題です。その辺を見ているのは管理者の責任ではないかと思えます。

#### (松村)

なぜ、くすぶり続けるのかという話なんですけれども、飯沢さんの意見に賛成です。力を入れているプロジェクトというのは、ちゃんとチェックしますので、くすぶりというか、すぐ気付いて早急な対処をしますので、意外と早く鎮火するんじゃないかと私は思うのです。「いや、まあいいや、いつかいつか。」というのは、どうしても後手後手の方にまわってくるケースが多いと思います。当事者でなくても、私もこの例であげたシステムに対して「早く手を打たないと。」と言ったこともあるんですけれども、やっぱり管轄担当の者がいて、ある程度それ以上のことは言いにくい。それで後で非常に大変だったというケースがあります。

やっぱり、管理者だけじゃなくて金額レベルにおいて、経営者がちゃんとチェックするべきではないかと私は思っているんです。徴候は当然ありますので、先程の十何億になってしまったというのは特にそうだと思います。

それから狼男になることの方に関しまして

は、私が思いつく例は、多人数で、なおかつ協力会社の方たちと作業をしているケースなんですけれども、やっぱり、当人も悪いと思っている時点があります。しかし、やっぱり、リーダーの方ですとか、マネージャーでもいいのですけれども、だんだん状況がもっと悪くなると、最終的には、その人の批判のところにいきまして、批判されるくらいだったら話しやすいメンバーとしかコミュニケーションしないという傾向も、私が見たようなケースでありますので、その時一歩踏み込んで、たとえ非難されたとしても申し訳なかったということで、言ってもらえれば軽症で済むかと思うのですけれども、往々にして、非難する側の人を避けて人は通っていくようなケースの方が多いのではないかと考えています。そういった時に手を打つというのは、そこまでは、第三者的な人を、そのリーダー、マネージャーの人をはずすというのは僕は賛成できないので、第三者というか、今まで利害関係の無いような人を入れて、その人を支援するような形で作業を進めると、うまくはいかないでしょうけれども、少しは良くなるのではないかと考えています。

#### (司会)

岡田さん、いかがですか。

#### (岡田)

私自身は、くすぶり続けたという経験はもっていないです。大体、私の方ですと、十名くらいから数十名のプロジェクトを担当したことがあるのですけれども、その程度ですと、大体、全員把握できますので、やっていることをその周囲へ行行って、「どうだ。」とかいろいろやって、ちょっと気になると、「そのプログラムをちょっと見せろ。」と。それで、どうだといって大体ざっとチェックすると、大体本人が申告しているとおりにきているか、もう全然話にならないのか、大体判断ができます。そうすると、もうその時点で、担当を減らして他にまわすとか対応をしてきましたので、直接くすぶり続けてきた例というのはあまり経験していません。

ん。

よく、くすぶり続けている例というのは、それをマネージしている人が、直接やっていることの中に踏み込めない場合だと思います。だから、ある程度、進捗管理表が何かでパッパッとチェックだけして、実際にその人がどういふことをやっているかまで踏み込まないと、どうも最終的にそういう方向に行っちゃうじゃないかとそんな気がしています。

(司会)

よろしいでしょうか。他に何かご質問ご意見等ありましたらご遠慮なく。はい、どうぞ。

(君島)

富士通の君島です。教育担当をしております。教育について、気になることがあるのですが、要するに、プロジェクトリーダーに任命する時の教育の手続きが確立していないのではないかという点で、パネラーの会社の手続きを聞きたいのですが、富士通の例で言いますと、「アンタ、来月からリーダーをやってもらよ。」というので、それだけでおしまいです。

一応、プロジェクトリーダー講座はやってはいますが、それは希望者が受講するだけです。来月からリーダーになる人が受けているかどうかというのは、そういうことは全然関知していません。多分、何にも教育を受けないでリーダーになっている人が大勢いると思います。富士通は、そういう恥ずかしい状態なのですが、パネラーの皆さんのところのリーダーの任命と教育が連動しているかどうかを教えてください。

(司会)

いかがでしょう。

(飯沢)

非常に耳の痛い話なんですけれども、実は私、プロジェクト管理の講座長をやっている人

ですけれども、実際には、現場ではプロジェクト管理の教育とリーダーの任命は全然無関係で、要するに任命してから、あっ、こういうことができなかった、しまった、やらせておけばよかった、そういう事態がよくあります。というのは、リーダーの下について仕事をしているときに、非常に優秀でよく仕事をやっていた。これだったらリーダーができそう、次はリーダーにしてあげようかと感性で任命する型式が多い。やらせてみると、意外と管理の基本を知らないことが多い。こういうことについて気をつけなければいけないと、その辺については基本的なことなので教育でずいぶんやっているはずなのですが、それだけでもやっておけば、ずいぶん違ったのではないかと最近反省が多いのが実態です。

(司会)

松村さん、いかがですか。

(松村)

ちゃんとした教育をやってからリーダーにするかといいますと、逆に、リーダーという職位になってから教育をしているケースの方が多いのかなという気がしています。私も中年になりましたので申し上げますけれども、意外と、机と机を並べて人を育てる時間というのは、サラリーマンに与えられている時間というのはOJTの方なんですけれども意外と少なくて、人を育てるといふか、こちらが育てるといふ気がないのかもしれないけれども、サラリーマン生活をやっている、実際、ある人と同じ時間というのか同じ作業で同じことを考えている時間というのは、私が振り返るとせいぜい2,3年の間に2回ぐらいかなと、意外と短いものだなという気がしている最中です。

(司会)

岡田さんは。

(岡田)

当社の場合は、全くリンクしていません。基

本的には必要な教育というのは定められていますので、そのコースを順次受けてきた人ということなんですけども、じゃあ、そういうものを受けてきた人をちゃんとリーダーにしていくかということ、そうではありません。

逆に、私自身は、あまり教育を信用していないということ、ちょっと失礼になるのですがけれども、教育を受けてきて、その人にいろいろ聞いてみて、「じゃあ、おまえ、これについてどうなんだ。」と聞いても、どうも反応がパツとしない。そういうケースが結構あるのです。やっぱり教育を受けてきたというハンコだけは頂戴してくるのですけれども、一体どういうことを聞いてきたのか。あるいは、最近の教育ですと、どちらかということ、モノを聞いてくるという受動的な教育というのが非常に盛んなのですけれども、もうちょっと積極的に自分から何かをやるという教育をやっていないと、リーダーの選任とリンクはしていかないのではないかと、そんな感じがしています。

### (司会)

はい、ありがとうございます。他にありますか。でしょうか。

ちょっと非常にいじわるな質問をしたいのですが、今、三人のパネラーの方々、経験をもとにこういうことで失敗をしました、で、失敗の要因等をみますと、なるほどなあというのはたいがい全部あがっていたのではないかと私自身は思っています。それに対する対応策もいろいろな話が全部あがっています。それだけ分かっているにもかかわらず失敗する。やはりさきほど坂本さんがおっしゃっていた“くすぶる”という話があるんですけれども、おそらくそれは表面化した時には、多分失敗した時として認識されるのではないかと思います。だから、くすぶり続けている時期のうちに早く発見して、それに対して何らかの手を打つということが、失敗してもいかにリカバリーできるかということになると思います。

たとえば、人の投入でも、何かの技術導入でも、こういったことがあるのではないかということ、その辺をどういうふうにお考えになるかというのが一点と。

もう一点は今、この時代ですから技法とか方法論とかツールとか何かを使ってプロジェクトが失敗しないような試みがなされたかという二点についてお聞きしたいのですが。

### (飯沢)

対策ということなんですけれども、実は、プロジェクト管理というのはず一つ前からやってきていまして、対策という対策はすべて打ってきたような気がするんです。

一つには、さきほどの教育という問題もあるんですけれども、最初は無計画というのがあったのです。計画が全然できていない、したがって失敗する。じゃあ、計画をちゃんとやろうということで、そういった教育もしましたし、OJTもしました。最近では、プロジェクト計画書というのをプロジェクトがスタートする前にやりましてレビューをちゃんとする。今年に入ってから、二つ大きな失敗というか、大きな崩れがあったんですけれども、どちらもプロジェクト計画書がものすごく、今までに例がないくらい立派にできて、レビューとしてこれなら大丈夫というふうなことでもってスタートしたのですけれども、結果としては、それだけに、あれだけ立派なものができているのだから失敗するはずがないということで、逆に発見が遅れたのかもしれない。

ということは、よく考えてみると、そうやって教育をするということは、逆に言えばこういうふうにすれば上をだまし易いということはないのでしょうか、通し易いという、そういったような学習効果ができてしまって、要するに、形式的にはきちんとしているけれど、実際やらしてみると、中味が伴っていないというような傾向になっているのではないかと思います。

す。そうすると、この先、次にどういう対策を立てようかというのが、今の私の課題というか悩みなのです。

技法というのはどうかということなのですが、これはというのが見つからなくて、やはり、自分たちに合ったものを工夫しながら探していくというのが、今の現状です。たしかに管理ツールを使っているというものはあるのですが、それはどちらかというと、計画時点での計算ツールだとか、あるいは品質管理の手法とか、そういったたぐいのもので、何かトラブルが起こった時にどのように対処するかといったノウハウを集めたものはありません。

(松村)

事が起こってから、それを回復するためにどうすることをすべきか検討するのに私も組まれて入ったことがあるのですが、やはりそれはやっていて前向きな検討事項ではないような気がします。次の作業に移ったときにどういう事象が起こるのだろうかということをちゃんと予測し対応を立てて、最悪何か起こったときには納期をどこまでギリギリ延ばしても大丈夫かというようなことを再三再四にわたって検討すれば、多少防げると思っています。

私も、事例にあがったシステムですけれども、実際に設計期間中4ヵ月ぐらい、毎週2回程、それこそ11時になるまで、普通の勤務時間帯は設計とか客先との交渉をして、その終業後の時間をさいて、どこまで延ばしておいても大丈夫かということと、次の作業に移った時にどういうルールをすべきかとか、どういうチェックをしておくべきかとか、あるいはどういうツールを準備しておけば多少は救われるかということを検討したこともありました。設計期間中も大変でしたけれども、実際の単体テストとか結合テストに入ったときに非常に徹夜徹夜のガンバリズムで押し通したのですけれども、多少とも事前の検討のおかげで、その時間、その時のテストケースに移ったときの徹夜はそんな

に我慢しきれないものではなかったという記憶があります。

それから、技法と方法論に関しては、自分たちのプロジェクトに合ったものにカスタマイズして、よりフィットする形で適用するのが一番望ましいと思っています。

(岡田)

私は、さきほどもちょっと言ったのですが、いくら形式を作っても、おそらく、うまくいかないものはうまくいかないだろうと思っています。やはり、リーダーなり管理者になった人が実際にプロジェクトの相当中味の方に踏み込むということまでやらないと問題が発生してしまうのではないかとということで、私のところでは少なくともリーダーとか主任クラスになった人に対して、もっと踏み込みなさいということを一応いっています。やはりそういうことが重要ではないかと思えます。

それから、ツールとか技法については、これも前から使えと言っているんですけども、基本的には使わないケースが非常に多いということです。それはどうしてかということ、実際使うためにはそれなりの工数も必要ですから、きちんとそういう工数と費用の面でちゃんと保障するというようなことをしていくと、それから、やはり、そういうツールを使うにあたっては相当の支援体制というものをつくっていかないといけないだろうということで、通常のプロジェクトの中で、非常に厳しい損益計算をしているなかで、ツールを使えというのはなかなか難しいのではないかと。最低限の標準というのはつくって実行していますが、それ以上強制するというのは難しい。そういう意味で、支援体制を強化していこうという方向で今やっています。

(司会)

はい、ありがとうございました。時間も迫ってきたんですが、会場の方からご意見がござい

ましたら。

はい、平岡さん、どうぞ。

(平岡)

長野大学の平岡です。今、プロジェクトの失敗に関して、中年のおじさん方の悩みということで、何か失敗をすると上の方が人が責任をとられると思うのですけれど、実際には、末端というか若いのが、あと1週間早くこれがうまくいかないのを教えてくれればよかったというのがあるのではないかと。私はあまり経験がないのですが、そういうことを思ったことがあるのです。本当、若い人には不慣れが多いので失敗しているのではないかなと思うのです。そういう若い人を送り出している立場なんであまり言えないのですけれども、そういう隠れたものはあるのでしょうか。

(司会)

簡単に、どうでしょうか。たとえば、若い方々がもう少し早く報告してくれていたなら、もっと早くこれに気付いていてリカバリーできたのとか、問題にならなかったことがあるのではないかとということなんですけど、そういうことはございますでしょうか。

(飯沢)

非常にジェネレーションギャップというのがあるのかもしれませんが、我々はすごく焦ってしまうのです。要するに、納期が迫っているのに、なんでアイツ定時に帰ってしまうのかとか、そういうのがあるのです。きちんと計画させてきちんとそのとおりにやってもらえば、別に何も問題はないのですが、やはり、プロジェクト全体をやった時には、ある程度せっぱつまった時にはプライベートなことは犠牲にしてほしいというのは管理者の勝手な言い分かもしれませんが、そういうのがある。

個人的な行動をとってしまうと、プロジェクト全体になんとなくいやなムードというかそう

いったものが発生してしまう、というようなことで、なかなか若い人たちも思ったことを口にできないような状況にある。で、そういう雰囲気させているのがどうも我々自身の反省しなければいけないことなんですけれど、もっと率直にいろんな意見が出るようにすれば、早くいろいろな問題に気がついて運営がうまくいくのではないかと。ヒューマン・ファクターということも岡田さんが言っていましたけれども、会社を超えたコミュニケーションの問題をクリアすれば割合とうまくいくプロジェクトが多いと思っています。

(岡田)

私はずいぶん前にそういうように感じていました。なんでアイツこんな時にそんなことするのだとか、なんでこんな時に休暇を三日も取るのだとか、そういう感覚がずいぶんありました。ただ、最近はもう割り切りまして、そういうものだと思ってやればスッキリします。

(司会)

玉井先生、手をあげられていますね。どうぞ。

(玉井)

筑波大学の玉井です。ちょっとやじ馬的な興味かもしれませんが、プロジェクトが失敗した後どうなるかということなんですけれども、特に大きな失敗プロジェクトの後、そのリーダーというのは、やはり左遷とか降格ということで、もう一回プロジェクトの責任をとるといようなチャンスはあまりこないのか。それとも、もう一回チャンスが与えられて、そのときに今度は成功するというのが多いのか。やっぱり失敗する人は失敗しがちなのかという点です。それと、そうではなくて、あまり、そういうのは個人ベースに責任が還元されないで、ある意味では、そういう後の意識というのは薄いのか、その辺をちょっと伺いたい。

(司会)

どうでしょうか。生々しいですけれども。

(飯沢)

リーダーの質によると思うのですけれども、何度も同じ失敗を繰り返すリーダーもいます。すると、この人はこういうところで弱点があると、それはなんとなく見ていて分かりますから、だんだんとリーダーに任命する機会は減ってきます。そういう意味では、ある意味では格下げというような感じになると思うのです。

もう一つは、やはり失敗しても自分でなんとかしようという責任意識を持っている人は、次でも、リーダーをやらしてみようとか、やはりあれだけ頑張ってアイツができなかったんだからしょうがないとか、あると思うのです。そういったときには、もっと困難なプロジェクトでも再度やらしてみる。その時のリーダーの態度にもよると思うのです。逃げようとしているリーダーについては、もうこれはダメだというように見切りをつけますけれども、それでも挑戦しようとしているリーダーには何度でも挑戦させてあげようと思っています。

(司会)

松村さん、いかがですか。

(松村)

飯沢さんの意見に賛成なのですけれども、私どもの会社では、意外とパニッシュメントを与えられない。どのデータとはいいませんが、このなかの当事者の人が役員に昇格しているくらいですから、当事者への罰則はないのではないかと私は思っています。

(岡田)

当社の場合も、基本的には人の問題ではなくて、部とか事業部の問題というスタンスです。ただ、心の奥底には、あの人がリーダーだと危ないなあという感じはすると思うのです。だから、それが最終的に何らかの形でね返ってくるということは当然あると思いますが、少なくとも

も表面的にその人をどうこうという問題にはなりません。

(司会)

はい、どうもありがとうございました。

文責 市川(電算)

ソフトウェア・シンポジウム'92

&lt;パネル討論&gt;

## 「分散化とエンドユーザ・コンピューティング」

1992-06-11 16:00~17:30

長野県農協ビル アクティーホール

司 会: 西条経治(長野県中小企業団体中央会)

パネリスト: 岡田達郎(SRA)

橋山真人(花王)

片貝孝夫(片貝システム研究所)

## (司会)

皆さん3日間ないし2日間大変ご苦労さまでした。一番最後のクロージング・パネルになりました。ここでは『分散化とエンドユーザ・コンピューティング』ということで、いままでのソフトウェア・シンポジウムの中でも、ある意味でちょっと変わった異色のテーマかと思えます。

申し遅れましたが、私、司会を勤めさせていただきます、ご当地長野におります西条と申します。よろしくお願いいたします。

それではパネリストの方のご紹介をさせていただきます。最初は岡田さんでござります。昭和29年愛知県生まれで55年にSRAに入社されました。現在は環境支援部の課長をなさっております。いままで受託アプリケーションの開発を手掛けられ、現在はエンドユーザ・コンピューティングEUC指向の社内システムの開発をご担当なさっております。今日のキーワードは「分散協調」ということです。

真ん中の方は橋山真人さん。昭和19年神奈川県生まれ、42年当時の花王石鹸に入社されまして、その後花王システム工科学校の校長先生等を経られまして、現在は花王のシステム開発部情報教育担当首席部員兼花王ソフト開発株式会社の常務取締役をなさっております。なお、花王は昭和59年の第1回のOA賞を日本OA協会

から授与されておられます。担当分野は教育、著書には「花王の社内パソコン革命」というものがあります。今日のキーワードは「情報リテラシー」ということです。

最後の方は、片貝孝夫さんです。21年群馬県生まれ、58年に片貝システム研究所を設立されまして、現在代表取締役ということ。昭和59年より成蹊大学の講師もなさっております。業務経歴といたしましてはパソコン用の各種アプリケーションシステムを開発されたり、簡易言語の企画をされたり、種々のコンサルテーションをなさっております。著書もたくさんあるのですが、最近のものは「パソコン入門」「パソコン脅威の10年史」「データベース事始め」「パソコン活用の手引」ということだそうです。片貝さんのキーワードは「ユースウェア」ということです。

なお今回は情報サービス産業に関係のある方が多いということで、一つはユーザ企業でエンドユーザ・コンピューティングをどう進めるのか、またそのポイントは何かという観点、もう一つは、ユーザ企業のエンドユーザ・コンピューティングを情報サービス産業がどう支援できるか、そういう二つの観点からお話いただくことになっております。

タイムチャートは、この後お三人の方からそれぞれのEUCに関するポジションを約10分位

ずつでお話しいただきます。その後、ユーザ企業が情報サービス産業に何を期待するかということ、引き続きまして、逆に情報サービス産業がユーザ企業のエンドユーザ・コンピューティングをどう支援できるかという三つのフェーズを進めさせていただきたいと思います。

それでは、最初に岡田さんの方からお話をお願いしたいと思います。

(岡田)

SRAの岡田です。よろしくお願ひいたします。私は紹介の通り、現在SRAで社内システムの改革を行っております。今回、ペーパをご紹介させていただきましたけれど、ペーパは社内システムでの改革をするにあたっての私の青年の主張であります、青年といっているのは私だけなのですけれども。このパネルでは、この主張をEUCという観点からお話したいと思ひます。

社内のシステム改革が始まったのはユーザ要求の多様化に答えるためです。社内では基幹システム、それは販売管理システム、生産管理システム、会計システム等ですが、従来の方で動いています。

しかし、現在は多様化と即時性を求められる時代です。多くの企業でもそうだと思うのですが、現場レベルでも細かな分析、判断および意志決定が重要であるということになり、管理会計システムとか戦略情報システムといったシステム構築の要求が出てきました。

SRAのシステムが機能として必要としたものは、特に近年プロジェクトが大型化してきましたので、現場レベルでのいち速い採算見込みです。採算見込みといひますのは、あといくら予算が使えるのといった管理です。また、プロジェクト自身も多様化しましたので、それぞれのプロジェクトに合った管理方法を実現する必要がありました。

このような状況というのはEUCということを考えておられるその他のユーザ企業にも言えることではないかと思ひておひます。

現在の社内システムは中央集権化したシステムであります。単一のマシン上に全ての機能を登載して、中央の管理部門によって企画から開発、管理、運用をすることを想定して構築されたもので、エンドユーザにとって物足りないものになっています。

現場では中央管理部門での分析・判断・意志決定を待てない状態にいるからです。といひますのは、採算を管理する場合に月次処理を待つて今後いくら使えるのというような管理を、もっと速く確認したいとか、他にもっといろいろな情報がほしいという要求があるからです。

しかし、これに対して現在のシステムはあまりにも大規模化複雑化しているため、このような要求に答えるには柔軟性がありません。情報システム部門は、限られたシステム資源上に多様な要求を盛り込もうとがなばっているわけですが、あちらが立てばこちらが立たずということで要求の調整をするのが精一杯で、バックログをため込んでいる状態になっています。

このような状況から多様化と即時性に耐えられるシステムにするには、現在のような中央集権的なシステムではダメであると考えました。今後のシステムは分散協調型でEUCのシステムをめざすことにしました。

今、ダウンサイジングが話題になっていますけれど、非常によい方向に動いていると思ひます。ハードウェアの高性能化、コンピュータの利用技術の向上というようなシステムを取り巻く状況はユーザにとっては願ってもないことではないでしょうか。

分散協調型と言ひましたけれど、これはどのようなことかといひますと、情報は必要な部門

に配置し、各部門間で必要なデータはお互いに交換する。またデータ交換をスムーズに行うために中央に管理部門を置くという形を考えています。

それから、あと重要なのはエンドユーザ・コンピューティングへということで、利用部門が自ら企画し開発していける社内のシステムであること、そして利用部門による運営を前提としました。

ただし、このようにするには、いくつかの整備が必要だと思っています。利用部門がやりたいようなシステムを作って拡張していけるだけの権限と、そのシステムが全社のシステムとして協調的に動けるようなシステムでなければなりません。

当社の社内システムは、このように考えましたけれども、これは他のユーザ企業でも同じような方向になっていくのではないかと考えています。それはハードソフトともユーザのもとに分散していき、そこでユーザが拡張を自由にでき、かつ運用責任を持たなければならないのではないかと考えています。

そのようにシステム環境が変わっていけば、私たち情報システム部門は、これからは利用部門へのコンサルティングのような機能、そして先程のデータ交換の通信あたりの管理をしていかなければいけないと思います。

(司会)

ありがとうございました。引き続きまして、橋山さん、お願いいたします。

(橋山)

花王の橋山です。私自身は昭和42年に花王に入ったのですが、大学時代もコンピュータをやっていました。使う方の勉強をしていたのですが、花王に入ってすぐやらされたのが業界初の受注オンラインシステムを作ることでし

た。

当時ユニバックさんの機械を2億5千万で花王が買い取って、8000ステップで全部システムを作れと言われました。システムといってもアプリケーションではなくて、当時のOSは4Kしかなかったので自分でコードの変換から、ディスクといっても当時はドラムでしたけれど、リードライトの命令を自分で作り出すわけです。エラーが返ってくると自分でそのエラーを解析して処置を行うというソフトを組んでいました。

なぜ、こういう話をするのかというと、当時2億5千万円の機械のメモリが20Kバイトしかなかった、しかもスピードもいまのファミコン以下の4マイクロ秒でした。いまのファミコンは16ビットということもありますが、ナノ秒に入っています。ということは、わずかこの25年間で2億5千万円の機械が2万5千円で買えるという時代になったのです。

私がおもうに、我々アセンブラでそういう機械をさわった人たちがパソコンの出現のときにいち早く気がついたわけです。我々よりちょっと後のコンパイラから入った人たちは全部落ちこぼれたわけです。これを「疲労コンパイラの落ちこぼれ」といっているのですけれど。

それから10年間位そういう仕事をして、昭和54年にちょうどPC8000が出たときに私どもはびっくりしました。まさか、コンピュータが自分で買える時代が来るとは思わなかった。コンピュータが好きで学校へ入った私でさえ、という言い方はおかしいですけど、私でさえコンピュータを自分で持てる時代が、IBMブランドのコンピュータが自分で買える時代が来るとは思わなかった。

ちょうどその頃、大型計算機を使っていたシステム開発部というものが非常に不思議な存在に感じられ、自分でジレンマに落ちいていた

時期でした。なぜこれだけ残業をやっているのに人には評価されないか、我々のコンピュータの使い方は間違っているのではないか、というジレンマを感じていました。経理部だって100時間も残業をやっていれば理解してくれるのに、計算機部門だけが残業をやればやるほど、専門バカだとか言われて、非常に疑問を感じていたところに、パソコンが出てきたので、こっちの方がおもしろいということで、今思えばエンドユーザ・コンピューティングを進めていたわけです。

当時は村八分でした。情報システム部の中でパソコンをやったなぞは非常にめずらしい存在で、当時のトップが新しいもの好きでカバーしてくれなければ当然落ちこぼれていたでしょう。

それが今10年たって、ひっくり返ったわけです。パソコンが主役です。これは一昨年、富士通さんが日経新聞にMESSAGE 90sを発表したときに、日経の記者がはっきり情報処理の主役はパソコンだと打ち出していました、まだまだその辺りを気が付いていない方が多いようです。

エンドユーザ・コンピューティングということは、さきほどこの4人の中で話が出たのですけれど、非常に生意気な言葉です。考えてみれば、まだまだホストが上であって、その下にずっとエンドユーザがいるという発想ですから、我々はファーストユーザ・コンピューティングというべきで、EUCという言葉が出てくること事態が、だめなのではないかと思っています。

その背景は、私どもSIS戦略情報システムということでは、かなり評判になった企業ですし、コンセプトは進んでいると思うのですが、そのコンセプトを推進するときに手段が大型寄りに片寄りすぎたと言えなくもありません。

たとえば、こういう事態がおきていた。全国

4ブロックに某社のものすごいコンピュータを導入して各支店に数千台の端末を導入してあらゆる情報が支店まで出るようにした。

ところが現場ではどういうことが起きたかということセールスマンの直行直帰勤務が始まったのです。自宅からテリトリへ行ってしまふ。支店に寄らないわけです。そうすると、どんないい情報が支店に出ても、どうしようもない。

そうすると、その数千人のセールスマンが何を要求し始めたかということ、自分たちが16ビットのパソコンたとえばダイナブックみたいなものを買って、ワープロで帳表を作る。たとえば、あるAというセールスマンは月別の売上げとか、Bというセールスマンは同じことでも縦横が違う表を作っている。その表をホストに送るからデータを入れて返してくれという要求に、従来の情報システムがお手上げになっている。

実は後から気がついたのですけれど、完全なクライアントサーバ方式です。クライアント側の方でアプリケーションを動かしてサーバいわゆるホストに対して情報を要求する。これに従来の情報システムがいっさいお手上げになってしまった。これははこの企業にも起きていることだと思います。

これを完全に切り換えていくことを花王では昨年からは始めているのです。再構築です。これは大変なことをやらなければいけません。これは従来のデータベースもソフトも全部切り換えていかなければいけません。

当然エンドユーザ・コンピューティングですから、クライアントサーバになっていな

ければいけません。それから各部門にはPC LANです。ダウンサイジングというのはパソコンに切り換えるだけでなくLANが当然なければならぬ、それが第2世代のLANでネットワークOSがないと当然だめです。ですからホストコンピュータは全部サーバに成り下がっていただこうと思っています。それではちょっとという場合はエンタープライズサーバとか、何とかハブでもいいのですけれど。そういうことが背景にあるわけです。それから環境としては、当然さきほどからいっているPC LANとか、リレーショナルデータベースとか、クライアントサーバ型に変えていかないとエンドユーザ・コンピューティングに進まない。

そうになっていったときに情報システム部門はどうなっていくか。私は10年前にパソコンが出現したときから情報システム部門不要論を唱えています。なぜ情報システムがあるのだろうか。あるから悪くいわれて専門バカと言われるのです。それならば情報システム部門は必要ないと思うのです。そういうと結論は極端になるので、役割は変わるのだと思います。コンサルタント、さきほど岡田さんの話にもあったコンサルテーションとか情報サービスセンターとして、またはデータベースセンターとして機能を果たす。

データベースとは一元管理をやらなければデータベースではありませんし、商品コードとか得意先コードのコード管理も全部一元管理しなければいけません。ただ一元管理というのはホストでやるということではありません。ロジカルデータベースにすればよいので、実際のデータベースは分散型のデータベースでよいのです。ここら辺がアカデミックなことをやっている人には分かっていただけなのですけれど、データベースはロジカルでいいわけです。フィジカルには分散していてかまわない、そういう製品があれば一番いいのですけれど。

そういうエンドユーザ・コンピューティング

にしていくときに、何が重要になってくるかという、そういう仕組みを変えていくことだと思います。PC LANを導入してサーバを入れてクライアントサーバに切り換えて、当然ソフトはSQL対応型にして、データベースもSQL型にすると、それで言語はCとか第4世代言語とか、Excel、Wordみたいなパッケージを利用していく。こういう切り換えが大事です。しかし、もっとも重要なのはエンドユーザ教育です。

花王では情報リテラシー教育ということでコンピュータリテラシーとビジネスリテラシーに分けています。

従来のOA機器教育は全部行き詰まっているのです。どこの企業でも、うちでも同じことです。オペレーションを全部教えたけれども、コンピュータリテラシーを教えただけで、車でいうと、ハンドルを回すと曲がるとか、ブレーキを踏むと止まるとか、そういう教育をただだけ、どこへ行きたいかという人の満足は満たさなかった。全部ペーパードライバに終わってしまうだけです。

その反省を込めて、我々は、今日NTTの方が講演なさったそうですけれど、やはりCSCWというか、Computer-Supported Cooperative Work コンピュータと対話するのではなくてコンピュータの向こう側にいる人間と対話するためにコンピュータを使うわけですから、オペレーションは当然必要なのですが、それ以外にコミュニケーション技術を教えなければいけない。コミュニケーションとかプレゼンテーションとかドキュメンテーションとかファイリングとかディベートです。そういう技術を身に付けない人がコンピュータを普及させていったらどうなるであろうか。そういうことでビジネスリテラシーです、これを必須で教えていこう、新入時から10何年経った人にも教えようということで情報リテラシーというのを位置づけています。

これは、皆さんの方がお詳しいでしょうけれども、慶応大学の環境情報学部長の相磯さんが言い始めている情報リテラシーに企業側の要請としてプラスαしたものを位置づけています。

この教育が終わったときにどうなるかという、情報システム部門は要らなくなるであろう。これはシステム指向ワークデザインの考え方でいうとSEも要らなくなるのです。これには10数年20年かかるでしょうけれども、各部門の管理者がSE的センスを身に付けるように教育をもっていけばSEはいらない。もともといらぬものを一生懸命作っていたのではないかというのが我々の反省です。エンドユーザ・コンピューティングとはほんとうに進んでいくと、情報システム部門はいらないという問題がありますから言い直しますますけれど、結論はエンドユーザ・コンピューティングを進めるあるいは進んだ場合は情報システム部門の役割はそうとう変わるであろう。

花王では今400人ぐらい情報システム関連でいますけれど、これが十分の一位ですむのではないかと想定しております。エンドユーザ・コンピューティングとはそういうことだと思います。以上です。

#### (司会)

ありがとうございました。引き続きまして片貝さんをお願いいたします。

#### (片貝)

片貝でございます。橋山さんのセンセーショナルな話を聴いたので毒氣にあてられております。

私も当初は大型コンピュータのシステムの方をやっておりました、大型コンピュータの仕事をやっているときに思ったのは、いくら徹夜してやってもなかなか評価されない。評価されないのに加えてシステムが出来上がっても、誰も喜んでくれない。喜んでるのは誰なのかと

思いますと、法人という顔のない人なのです。これは非常に喜んでるのです。しかし、そこで働いている個人個人はありがたいと言って涙を流して喜ぶとか、感謝された覚えはないということで、どうも苦勞の割に実入りの少ない商売だと思っていました。

そこへパソコンが出てきたのです。それに飛びつきました。これだと思ったわけです。パソコンで人のよろこぶ顔がみたいと思ひまして、パソコンの世界に飛び込みました。結果として会社を作るはめになりまして、今に至っているわけです。とにかくコンピュータを使いこなすことが一人ひとりできるようになると、すごくいいのではないかと。使わせられるのではなくて、使うということです。それが大事だと思っていました。

今、こういう時代になってきてまして、非常によろこばしいと思っておりますけれど、しかし、まだまだ不足していると思っております。たとえば、日本のコンピュータ業界のリーダの中で自分でキーボードを叩いている人は何人いるのだろうかと思ひますと、私は10人のうち9人は叩いていないと思ひます。使わずしてコンピュータを論じているのではないかと思ひます。これはとんでもない話で、キーボードをさわってみれば、どんなに使いづらいかというのは分かると思うのですけれども、そういうことすら分からないでコンピュータの仕事をやっているのではないかというような感じがすごくいたします。

私も実は5年くらい前までキーボードを叩かなかったのです。特に昔から大型コンピュータをやっていた人間というのは、キーボードを叩くというのは、下賤の仕事みたいな気持ちがあつて、どうしても、せっぱ詰まってキーパンチマシンの前に座っていると考へてしまう。紙とエンピツできれいに終わるのがSEの仕事だと思ひていたので、何となく触らなかつたのです。

ところが回り中が触っていてワープロなど使わないと文章が書けないとか、いろいろなことが始まって、そんなことで攻め込まれていったのですが、あるとき名刺一枚自分で入れなければいけないはめになって、パソコンに入れていったのですが20分もかかったのです。20分かかったのではコンピュータなんか使えない、人に使わせて、その結果を腕組みして見ているしかできないということになった。

これでは絶対にだめだ、とにかくブラインドタッチに近いものを覚えようと思いました。今は手で書くよりも早く日本語を入れることができます。そうしてみると、コンピュータがよくわかってくる。コンピュータというものは本当に道具になるのだということが実感で分かってきました。とにかく理屈の前にキーボードをちゃんと教えなければならぬと思っているところです。

日本語を入れるのにマウスで入れるわけにはいきませんし、それから手で書いてということも速くならないですからなかなか難しい。霊長類の中で一番進んだ能力を人間の指は持っているのですから、やっぱりキーボードを使って手で書くより早く日本語を入れる訓練が一番大切です、そのあとの話は全部それができてからの話だという感じがしています。

我々の会社は今50人位なのですけれど、アルバイトから61才のおじいさんまで、全員がブラインドタッチです。それができないと、ほとんど仕事にならない。もちろん全員がコンピュータの専門家ではなくて事務をやっている人もいるわけですが、全員が叩けます。アルバイトで入ったときでも、皆が叩けるから、叩けるのが当たり前と思ってとにかく一生懸命やるのです。一週間もするうちに速くなってしまいうという環境になっています。

どうしてそういう環境になっているかといいますと、たぶん我々の会社は日本で一番進んだ

ユーザだと思っているからで、そのようにやってきました。販売管理システムとか経営何とかシステムとかそういうのがあるという意味でなくて、コンピュータを道具として使いこなしているユーザとしては、たぶん日本一だと思っています。一人一台のコンピュータ環境はありますし、ネットワークOSとかUNIXだとか、そういうのがすべて環境として整っておりまして、まさにエンドユーザ・コンピューティングを自分たちでやっているという状況です。

ふつうはワープロで使うことが多いです。それから経理の方でしたら販売管理とか給与計算とかそういうのはパソコンのパッケージでやってしまいますし、それからグループコンピューティングでは名刺の管理とかスケジュールであるとか会議室の予約とか、諸々の資料がどこにあるとか、誰か電話を受けとったときは伝言をいちいちメールに打つのは大変だから決まりきったパターンの画面に入れるとパカッと送られるとかいった伝言メモの機能とか、10いくつかのグループコンピューティングの機能のソフトを自分たちで作りました。

教育をやっているセクションでは研修センターを持っていますので、研修センターの予約からお金の請求とか自分たちで全部infomixというデータベースを中心に使って作っています。

ユーザサポートをやっている部門はユーザからばんばん電話がかかってくるのですが、電話がかかってくると聞きながら質問を打ち込んでしまいます。そして、その質問に自分がどう答えたかも打ちこんでしまいます。

そうしますと、たとえば3時間後に同じ人から電話がかかってきたときにユーザのIDさえ言ってくれば、誰が電話をとっても番号をポンと叩くと、過去の問合せ履歴がポンと出てくるのです。誰がどういう答えをしたのかということがわかり、また、お客さんのレベルも分か

りますから、リターンキーというのがどこにあるのか分からないという人には、一番大きなやつですよというような言い方をするとか、コンピュータのプロの人だったらもっと省略した言い方をするとか、どんな言い方をしたらよいかも分かってしまう。そんなようなシステムを自分たちで考えるわけです。

もしユーザサポートシステムがダウンしたら、もうほとんどパニックです。紙と電話だけでは、とても仕事はできないという環境にまでいつてしまっているわけです。

このように自分たちの仕事は自分たちで考えて設計して作っています。もし作れなければ、最終的には開発をやっている部門の人間に頼んでやらせる。そしてああでもないこうでもない文句を言ってどんどん直させる。つまり、設計はどんなことがあってもユーザ部門がやるのです。そういうのを見ていると、以前から情報システムの仕事をやってまいりましたけれど、本当にお客さんに申し訳けないことをしたと、私は思っています。

通常システムというのは、1回作って納めておしまいになってしまいます。我々が自分たちのためにシステムを作ってみたら、最初に考えたシステムはとても使い物にならない。こんな使いにくいものを誰が使うかといって、すぐ文句をいう。どんどん直しに入りまして、1年くらい経つと、もう似ても似つかないものになっている。システムというものは、このくらい変えていかないと、使い易いものにならないということがつくづく分かりまして、過去のお客さんに謝ってまわりたいくらいの私の心境なのです。

ところが、この間、ある会社へ行きましたら、15年前に作ったオンラインシステムが健在でまだ動いておりまして、本当に何といたらよいか分からなかったですけども、本当に間違いなく使いにくいだろうな思うのです。です

から、システムというのは、ユーザが自分で考えたものでなければ、使い易くは絶対にはないと思っています。

第三者の仕事が分からない人間が、いくら考えたって絶対にだめだということ、だから、私なんか、今めざしているところというのは、さきほど「ユースウェア」とキーワードでいただきましたけれども、「ユースウェア」という言葉を考えて何とか普及させようと思っているのです。

ハードウェアとソフトウェアという二つのウェアがある。しかし、ハードとソフトがあれば誰でも使えるのかというと、そうではない。やっぱ環境を整えたり、何か事があつたらすぐにバックアップしたり、こういうようなことをする、いわゆるサービスというものがなければいけないだろう。そういうことがいらないパワーユーザはハードとソフトを持っていれば自分でやっていける。

ところが誰でも使わなければいけない、当社でしたら新入社員が入るとハードとソフトを全部買わなければいけませんから一人に80万くらいかかるわけです。とにかく10人入ると当社はコンピュータだけで800万かかるのです。

そういうような環境になっている場合は、ユースウェアサービスも一緒に考えてやっていると、どこかでネックが起こってできない人が出てくるわけです。出来る人は自分でやればいいし、出来ない人はユースウェアサービスを受けながら使えばよいというようになってくると思うのです。

しかし、考えるのは自分です。自分ですよということです。自分で考えてもらって、それを技術的にバックアップするのはユースウェアであると、それで少しづつ前へ進んでいただくということです。そういうサービスがあつて初めて、三つのウェアの三位一体ということになる

のですけれども、それがあって誰もがコンピュータを気楽に自分のために使えるのではないかと考えています。

エンドユーザ・コンピューティングという言葉は、橋山さんも述べられておりましたが、確かに気に入らない。どういう言葉がよいか、よく分からないのですが、鉄道総合技術研究所の尾関さんという方がパーソナルコンピューティングというようなことをいっていましたが、何とかエンドという言葉は払拭しないとまずいのではないかと思います。本当はユーザによるユーザのためのコンピューティングだということで、自分でやるということ、要は、自分でやる、それをささえるのが、情報システム部門であり情報サービス産業であるだろうと思います。

#### (司会)

はい、ありがとうございます。三人のお話には利用部門が企画、開発運用をして自分のためにシステムを作り上げていくのだという共通点がありました。エンドユーザという言葉はファーストユーザあるいはエンドユーザ・コンピューティングをファーストユーザ・コンピューティングとかパーソナル・コンピューティング、何か別の言葉に変えた方がいいというお話と、情報システム部門につきましては橋山さんから、過激には不要論ちょっと言い直すと役割が変わるというお話があったかと思えます。この役割が変わるということは岡田さんからのご指摘にもありました。

続きまして、情報システム部門というものを情報サービス産業に置き換えて、ユーザ企業あるいはユーザの立場で何を期待するか、こちらのコメントをいただきたいと思えます。これにつきましては、橋山さんと片貝さんをお願いしたいと思っております。よろしいでしょうか。では、橋山さん先をお願いします。

#### (橋山)

はい、さきほどの片貝さんのユースウェアと私どもの情報リテラシーの中のビジネスリテラシーというのは完全に一致する考え方だと思います。

情報サービス産業に何を期待するかというと二つあると思います。一つはビジネスリテラシーというのを教える講師がいないのです。コンピュータも自分で使っていて電子メールも当然打てる、それからExcelも当然使える、それから社内データベースからも自由にデータがとれる、NIFTY-Serveも当然入っている、などというインフラを持ちながらプレゼンも教えることができる、それから、ドキュメンテーション、文書化も教えることができる、ディベートの技術も教えることができる、ハイブリッドな講師というのが世の中にいないのです。

ディベートを教えることができる、いわゆるコミュニケーション技術としてディベート、ドキュメンテーション、プレゼンテーションを教える人は沢山います。いわゆるアナログ的な人間です。

片や電子メールを使って自由に情報交換はやってはいますが、特に情報システムにいるような人はコミュニケーションが下手です。デジタルな世界に生きていますから。

両方の技術を持ったハイブリッドな講師というのは非常に少なく、我々、私の下にいる人が教育が好きだということで技術を身に付けてやっていますけれども、情報サービス産業さんに、特に情報サービス会社さんをお願いしたいのは、情報というのはコンピュータだけではないと、たぶん、ここにいらっしゃる皆さんは情報と聞くとすぐにコンピュータが頭に浮かぶと思うのですけれども、コミュニケーションといたったときにコンピュータを思い浮かべる人は、C&C、Computer and Communicationですから日電さんだけでいいと思うのです。現在は日電さんもC&Cという言葉はあまり使っていま

せんが。

実は、コミュニケーションといった場合に3割ぐらいしかコンピュータに乗らない。それを10割だと思っているのがシステム開発部門とか情報サービス産業の人たちだということをよく言われるのです。これをよくわきまえて人を育てていけないといけないと思います。エンドユーザ・コンピューティングが進むと、いままでは情報システム部門に対しサービスすればよかったのが、情報システム部門の役割が変わったり、なくなったりしているのですから、情報サービス産業はどこをどうサービスするかという、エンドユーザなのです。そうすると当然コミュニケーション技術というのが必要になりますから、そういう人になっていかないとだめだと思うのです。

それから、もう一つは、もっと技術的なことをいうと、エンドユーザ・コンピューティングというのはやっぱり時間がかかると思うのです。そうすると、ダウンサイジングやクライアントサーバの用語の前までMMLマイクロ・メインフレーム・リンクという用語が流行ったのですが、ここが大事だと思うのです。

エンドユーザにPCが入った、サーバが入った、そしてまだホストコンピュータに基幹系のデータがある。それをどうやって落とすのだと。単にファイル転送ではだめです。これはエンドユーザがExcelとかLotus 1-2-3とか使って、パラメータを出せば自由にデータベースから引っ張ってくるという仕組みが必要なわけです。その仕組みは情報システム部門とかエンドユーザではできませんから、こちら辺はソフトウェアハウスさんとか情報サービス産業さんをお願いをしたいと思っています。

ですから二つ、もう一度いいますと、情報リテラシー教育とは何かということで、その辺りをサポートできるようなことを要求するのと、マイクロ・メインフレーム・リンクの辺りをサ

ポートしてくれるようなソフトを開発してほしいと思っています。

最後に、もうちょっと2、3分あるでしょうか。わたしも7年前ちょっと遅かったのですが、UNIX気遣いでした。花王でVAXを一番先に導入しました。UNIXを使いたいためにVAXを導入したわけですが、いまでもVAXは入ってはいますが、いまは私はUNIXにあまり興味がなくなっています。

なぜかという、これは当たるかどうかわかりませんが、UIとかOSFとかX/Openなどということをやっているうちに、相手は1社ですから全部マイクロソフトにやられるのではないかと思います。しかもアップルとの訴訟問題で勝ってしまったわけですから。ハードを持っていませんし。下手をすると全部Windows NTにやられるのではないかとということをしてUIの大会で話したことがあります。早くあんな争いはやめて一緒になりなさい、敵は1社ですよ。

たぶん、マイクロソフトさんはUNIXはあまり問題にしていらないと思うのです。やはりGUIが問題です。今日の新聞にマッキントッシュが非常に一人勝ちで売れているという記事が日経に出ていました。他のメーカーが、これはまずい、と色めきだっているそうですが、当然の話で、マッキントッシュは私が自宅でも使っていますが、五つの子供が自由に遊びます。電源を入れると使えるからです。98とかIBMのパソコンなんか五つの子供では絶対に遊ばない。あそこら辺に気がつかないとダメです。

一番何を言いたいかという、マッキントッシュを使っている人が裏で、UNIXが動いているか何が動いているか全然知らないでしょう。ウィンドウが動いていることすら知りません。アイコンが出てくるだけです。それでいいわけです。そのアップルとIBMが手を結んだわけですから、他のメーカーは何もしないと、IBMにやられてしまうのではないかと私は思ってい

ます。

UNIXに興味をなくしたのは、インフラというのはバックグラウンドでいいと、ブラックボックスでいいのではないかということをおもっているからです。このあたりには異論がある方がいるのではないかと思うので、時間があつたら、後で討論をさせていただきたいと思います。

### (司会)

はい、ありがとうございました。今回来られている方はどちらかと言いますとUNIX指向の方が多いいと思いますので、今おっしゃったことについては、後で質問等をしていただければと思つています。

それでは、片貝さん、続きましてお願いします。

### (片貝)

今の橋山さんの話を受けて、まさにその通りだと思つたのです。私はUNIXが統一仕様になってくれれば一番いいと思つたのですけれど、メモリもたくさん取れますし、自由にいろいろなことができます。しかし、標準的なものになってこない限り、DOSの世界、Windows NTの世界というのはどんどん今進んでいる。アプリケーションもものすごくある。

ですから、私はどちらかというところ、UNIXと思つているのですけれども、しかし現実問題、うちの社内にマッキントッシュがいくつか入つてきてグループウェアのシステムに対してWINGZからクライアントサーバができるものですから、入つてしまうのです。WINGZから入つてグループウェアのスケジュールデータあるとかいろいろなデータをWINGZのソフト環境で回して、そちらで使つている。確かにものすごくいいわけです。

社内ではもうひとつExcelからSQLを発行できるものを作りました。エクセリンクという名

前をつけて売り出そうとしているのですが、それはExcelの環境ですべてUNIXのデータベースが使えてしまいます。こうなると、もうUNIXはバックエンジンという感じになってまいりまして、橋山さんがおっしゃったような環境にどうもなりそうだと思います。

ただ、UNIXはスケラビリティがありますから、机の上におけるものからホストコンピュータと同じくらいの能力か、またはそれ以上の能力のものまで、たとえばNキューブなんていうものはものすごいスピードを持っていますし、ホストコンピュータで実現しえないようなスピードでトランザクション処理をやりまから。そうしますと、UNIXはバックエンジンとしてはすごい。ただし、サンとか一生懸命がんばっていますけれど、バックエンジンにしかかなり得ない。やっぱりユーザがつかねば、どうにもならないという話です。そこが残念なところなんです。

みなさん、もしUNIXを中心に考えられるのであれば、ともかく、ソフトウェアがポンポンと乗るようにしていただかないと、ユーザは使えないです。また、儲からないからパッケージを作る会社が出てきません。まあ、そんなところがあります。

それから、エンドユーザ・コンピューティングというテーマですので、まず情報サービス産業がこれからユーザが情報サービス産業に期待できるかということなのですけれども、今の状態で期待できる部分というのはピークカットです。それしかないのではないかと思います。

本来ならば、ユーザは自分でやりたいわけです。ソフトウェアハウスは仕事のことをよく知りません。ユーザ企業の仕事のことをユーザほど知らないですから、できることなら自分でやりたいと思つている。しかし、大規模システムの開発となると、開発要員が足りないから情報サービス産業に頼むというのが現実なわけで

す。冷静に考えてみれば、それでしかありえないわけです。

ではそうした中で、どうやったら魅力のある情報サービス産業になれるかということです。大学の先生もいらっしゃるということですが、もし企業の情報サービス産業の方がいらっしゃれば、私もその一人なのですけれど、まず全社員の机の上に専用のコンピュータがあるかどうか。それを自問自答してみただけませんかでしょうか。

もし、それが無いようでしたら、結局ユーザが何を考えて何を困っているかわからないわけです。自分がまず困って見なければいけないわけです。パソコンというものは少なくとも自分の道具でも十分あるわけです。見積りを書いたりするときがあります。それと同時に計算もするし、それからネットワークで結んで相互にプロジェクトの進捗管理をしたり技術的な情報を連絡し合ったりとか、それぞれ必要なわけです。そういうことに自分でコンピュータを使ってみなければ、たぶん絶対にユーザをサポートすることはできないと思います。

会社に帰ったら机しかなくて、お客さんのところへ行ったら、コンピュータを使ってサポートしろといっても何をサポートしていいのか分からない、言われたことしかできない。言われたことしかできないのであればユーザは困ります。

ユーザはコンピュータのことは分からないのですから、とにかく前へ前へ先導してくれるようなサポートがほしいわけです。だから新しいソフトが世の中に出てきたら「こういうのが出てきたから、あなたにはこういうのが向いているのではないのでしょうか」とか、いろいろなものを先回りして環境を整えるようなことをしてくれるのがうれしい。つまり、先生と呼びたいわけです。ユーザの人は、情報サービス産業の人を。先生と呼ばしてくれるようなところまで

いってくれば、お金を払ってぜひきていただきたいというようになると思うのです。

そこを業務のことを知ってというのは無理だと私は思います。その会社の社員になって、ずっといる人にくらべて、たまにしか行かないで1年間仕事をしてきたからといって仕事のこととはわかりません。それは無理なので、そうではなくて、コンピュータのサイトからしかできないのですけれども、自分はコンピュータのプロですから、自分のためにコンピュータを使い続けるということです。それを受けてユーザのところへ行って少なくともコンピュータのサイトからみてどういう環境があればユーザは仕事がし易いかということは分かりますから、それを提供してやることできればユーザはすごく喜ぶと思います。

少なくとも私の会社はそうやっています。自分たちがコンピュータを使い切ると。その上で、お客さんのところへ「これは便利ですよ。」ということ、熱っぽく語りに行くわけです。「とにかくLANを張りなさい。LANを張ればいろいろなことが始まる、絶対おもしろいですよ。」といって、自分の会社へ引っ張ってきて会社の中を隈なく見せます。「こんなことをやっているのですよ。」ということを見せる。そうすると、お客さんも目から鱗が落ちたような気分になって、それから「じゃあ、社内LANを張ろう。」ということで始まることが多いのです。

この不況にもかかわらず、あふれかえっているくらい仕事はきていますし、受けられない仕事も沢山あります。あまり、教えてしまうとまずいのですけれども、商売の敵のようなものですから。けれども、そのくらい我々にの対するユーザーの期待は大きいんです。ですから、是非、自分が使うということ、をまず覚えてほしい。それが第一点です。

あとは、技術的にいろんなことを知らなければ

ばいけない。UNIXだけでシステムを作ったら、今の環境では今までの情報システムとほとんど変わらないシステムしか作れません。UNIXをホストにしてパソコンを端末にしてシステムを作るといったら、ほとんど変わらないと思います。そうではないシステムをユーザーは要求しているのです。

もちろんいままでのようなシステムがあってもいいですけども、我々もそういうシステムを作ってますが、ユーザーはもっと自分で勝手にできる環境を欲しがっているわけですから、PC LANだとかネットワークOS、それからリレーショナルデータベースは前提です。それにユーザーに開放するデータなわけなのですから、そういうものをちゃんと知ることです。

ちょっと話はそれますがリレーショナルデータベースを設計する時に、私が社員に口をすっぱくして言うことは、テーブル名とかフィールド名を決める時にユーザーと相談しなさい、絶対に勝手に決めてはいけない、データはユーザーのもので、ということです。

いままでの情報システム部門というのはデータは自分で抱え込んでいる。自分がわざと抱え込んでいるわけではないのですが、データはユーザーに見せる必要はないと始めから信じ切っているのです。だから、勝手に、例えばF01とかF02とかコーディングが簡単に済むような名前を付ける。それでは、ユーザーがそれを見にいこうとした時に何が何だか全然わからないわけです。

だから、名前を決める時は、ユーザーに名前を決めていただく。ただし勝手に決めてもらうと困るでしょうから、話し合いの上で決めるということです。売上げファイルというのはSalesという名前を付けなければいけないのかどうなのか、製品台帳はProductという名前にするのか、顧客台帳はCostという名前がつくとか、あ

る程度進むと会社では話をする時にフィールド名で呼びあっているわけです。ユーザーも情報システム部門の人もフィールド名で呼びあう。しかも、ある程度集約されたテーブルなどは、ユーザーにそのまま開放することが多いわけですから、テーブルの形とか、全部ユーザーに渡しておくとのパワーユーザーでしたら、SQLを全部自分で叩いて検索してもらえばいいわけです。そうでなかったら、いくつかのパターンを作ってあげることになるわけです。少なくとも、データはユーザーのものだと思います。

たとえば、営業課長が取ってきた受注をとりあえずキャビネットに入れたとします。キャビネットを開ければ出てくるのは当たり前です。ところが、受注データをキーボードから入れた瞬間に吸取紙でとられるように情報システムに取られて、あとは「出してくれ。」と言ったら、「この形でしか見ることができません。」とか、「どこにあるのか教えろ。」と言っても、「それは何に使うんですか。」とか倉庫業者が何を言っているのだというのがユーザーからの立場です。倉庫業者は「入れといてくれ。」と言ったら「はい。」と言って入れておけばいいので、「出せ。」と言ったら出せばいいのでそういう全てはユーザーの意志の下に動けるような形を作り上げなければいけないと思うのです。

RDBは大事ですし、それから当然、抜き出したデータはLotus 1-2-3で使いたいとかExcelで使いたいとか、又はWINGZで使いたいということは当然あるわけです。又は、ワープロに一太郎に取り込みたいとか、そういうことは、もうしょっちゅうですし、それから、多分windowsも知らなければいけません、エンジニアとしては、DOSももちろん知らなければいけないし、windowsも知らなければいけない、そういう環境、いろんなことを知らなければいけないわけです。

そのためにはただ、勉強しただけではだめで、ユーザーが自分で自分のために最高の環境を作ってみるということです。そして、その中で色々と勉強して分かることが多いと思うのです。その環境をユーザーに提供していけばいい。そうすれば、ものすごく魅力のある会社になれると思います。又、魅力のある人になれるのではないかと思います。とにかく、今のようなままでは、単なるピークカットの助っ人にならないことだろうと思います。

(司会)

ありがとうございました。

情報サービス産業は優れたユーザーであれということにつきますのでしょうか。そういう環境を作りなさいと、そうすると、すごい先生にもなれるかもしれないし、橋山さんがおっしゃいましたビジネスリテラシーも教えることができる。そういうようなことかと思えます。あと2,3、情報サービス産業の方が、仕事の飯の種にするようなヒントもコメントしていただいたかと思えます。

それでは、三番目に入りたいと思います。次は、情報サービス産業の広場から、ユーザー企業のエンドユーザーコンピューティングをどのように支援できるか、ということにつきまして、お話、コメント頂きたいと思えます。

まずは、岡田さんの方からお願い致します。

(岡田)

企業のEUCをどのように支援するかということ色々と考えてきたのですけれども、今日この場になりまして、花王の橋本さんとか片貝さんから、私が考えていたことよりもかなり激しい情報システム部門不要論、それからSE不要論を聞いて、ちょっとびっくりしているところです。やはりこれからのユーザーの姿、私たちのお客さんの姿がそうなるのかなと思っています。

しかし又、いろいろ期待されていることも伺いできて良かったと思います。そこでまず、ソフトハウスにいるものとして、「これからも活用してください、益々お役に立てます。」とお答えしておきたいと思えます。

と言いますのは、かなり色々なことを期待していることを、お話しして頂けたかと思うのですが、まだまだ、コンピュータを利用する環境というのが不便なものではないかと思えます。それで、今回の色々なセッションにおいても、テーマはこれからどういうふうに使えやすくしていこうか、そういうことが、全てのテーマではないかというように思っています。

私たちも、今までは今の社内システムをつくり変えていく上でも、まだまだ現状として不満な点が一杯ありましたので、それを、今現在でできることで工夫しながら作った点があります。まだまだ、これからいろいろな基礎的な技術が発展していったらいいと考えておりますし、自分たちのところでも、そのようにしていかなければいけないかというように思っております。

それで、それだけではなく、どんどん普及していくことも大事なことだと改めて感じています。普及というのは、たとえば、こういうところでいろいろな事を主張することだと思えます。こういうところをこうしたらどうかというような議論が情報システムの発展につながっていくのではないかと改めて感じています。そういう意味で、一つは、これからの情報サービス産業がどういうふうに関与して行けるかということでは、もっとより良い利用環境、これを作っていくことではないかと思えます。その点についてのいろいろな示唆というものを、片貝さんなどから頂けたかと思っております。

しかし、今のソフトハウスは、受託という開発がかなりを占めているところがあります。これは、エンドユーザーのシステムの利用ということでは、まだこれから重要な部分を占めてい

くのではないかと思います。たとえば、これからどんどん自分たちのものを作りたい。ただしそれが大きくなった場合、自分たちのパワーだけでできるか。足りなくなった場合は、やはり私たちがお助けすることができるのではないかと考えております。

それから、システムのコンサルティングが重要になってくるのではないかと考えております。今のお話の中にもありましたけれども、エンドユーザーはコンピュータそのものは知りたくないと思えるからです。自分たちのやりたいことのために使うのであって、そのための道具のことについてはいちいち知りたくないと思うのです。したがって、ユーザーは「こういうふうに使いたいんだよ。こんなことできるのなの。」と質問するのではないのでしょうか。これに答えていかなければいけないのではないかと思います。「こういうことをするには、こっちのマシンよりもこっちのマシンの方がいいですよ。ソフトはこんなものが、今出ていますよ。」ということをユーザーの方々にコンサルティングなどを通して伝えていかなければいけないと思います。

それから、エンドユーザーの方々にも、本当にやりたいことをするためには、それなりの技術も知識もいると思います。そんな時に、橋山さんの方からありましたけれども、セミナーとかワークショップ等の教育的な事業もサービスさせていたかなければいけないと思います。そういうものをトータルすると、これは小規模な形のSIになると思うのです。まず、大規模なものというよりは、小規模なSIの形で、いろいろな企業にサービスをしていくような体制を取ることが大事だと思っております。以上です。

#### (司会)

はい、ありがとうございます。

引き続きまして、同じテーマで片貝さんをお願いしたいと思います。

#### (片貝)

ユーザーの側からも情報産業の側からも、両方喋れということ、又ご指名なんですけれども、今度は情報サービス産業がどうしたらやっいていけるか、ということを考えてみたいと思います。

今年の2月の中頃から末にかけてアメリカにダウンサイジングというテーマで視察に行ってきました。その時に国際石油資本のテキサコに行ったのですが、そちらで情報システム部門の方と色々話をして参りまして、あそこでは10年前からエンドユーザーコンピューティングを強力に進めているということで、社員3万人の内、いわゆる情報システム部門としている人は800人ということです。ところが、「コンピュータを使ってシステムを作る人が何人いるんですか。」と聞いたら「さあ、それはどれくらいでしょう。よく分かりません。」要するに各事業部にいっぱい、それぞれやっいるから分からないということです。

情報システム部門の役割というものを話してくれたのですが、10年前からエンドユーザーコンピューティングというようなことを進めたことによって、予算は各事業部が勝手に使いますから、どこでどういうシステムが作られているか全然分からないというのです。だから、コンピュータシステムのデパートみたいな感じで全てがあるのです。マシンも全ての種類がありまして、ネットワークも全ての種類があったり、とにかく全部あるという環境になってしまっていて、どこかでコントロールしようかと思ったんだけど、全然コントロールできない。「じゃあ、どうするんですか。」と聞いたら、「これからはもう、コントロールしません。」ということをはっきり言っていました。そこには、情報システム部門が管理していく時代は完全に過ぎたという感じがしました。

それから、ネットワークをシームレスにするんだということを盛んに言っていました。つま

り、いろんなネットワークができあがっているのですけれども、そのネットワークをユーザーから見て、シームレスにする。こういう絵を描いていたんですけれども、ワークステーションを1つポツンと書くのです。それから線をパーツと横に引いて、そこからいろんなものをぶら下げてあるのです。ユーザーにとって、何ができるかという環境をどれだけぶら下げるか、どれだけ用意できるか、これが我々の仕事だと言って、そここのところになると、すごく熱中して話してくれました。

そこに書いてあった絵はワークステーションがあって、そこにワールドパーフェクトとビジネスグラフがありました。それから、パソコンのところで使える Lotus 1-2-3 がありました。そのための PC LAN のサーバーがあるわけです。ダウンロードして使っているというわけですが、それ以外に、そこでは、プロフェスという IBM の Profesional Face System という、ホストで動くグループウェアがありますがそれが使えるとか、それから SAS が使えるとか、いろいろなものが使えます。それから大学の研究室に行けるとか、販売管理システムに入れるとか。

つまり、システムからユーザー、端末を見ているのではなく、つまり、ワークステーションから個人のユーザーという視点からその人が何ができるかという環境整備、それを考えているわけです。電子メールもいろんなメールが走っているわけですが、電子メール自体もシームレスにしていかなければいけないので、電子メールデータをコンバートして違うメールシステムにオーダーしてやるとか、そんな事をいろいろやっているのです。これはアメリカへ行った時のユーザーの話なのですけれども、ちょっと話しておきたかったのです。

あと、コンサルティング会社であるアーサーアンダーセンコンサルティングという会社がありまして、シカゴなんですけど、そこに行ってきた。

ました。アンダーセンコンサルティングは、今ではコンサルティングから入っていったシステムまで作るという会社ですから、ソフトウェア会社と言ってもいいと思います。すばらしいコンサルティングをする会社ではあるのですが、実はだんだんだんだん仕事が、ある会社に取りられていってしまう。どこに取りられるかといったら EDS に取られてしまうわけです。ダラスにある EDS にです。

EDS は単なる計算センターでした。保険関係の計算センターで大きくなったのです。どうしてダラスにあるかと聞いたら、どこからでも線を引っ張ると一番近いので通信コストが一番安いからダラスにしたと言っていました。ダラスのゴルフ場を買い取ってその中にボンとビルがあります。そんな会社だったのですが、実はファシリティマネジメントのサービスをしながらだんだん企業から仕事を取っていったわけです。

たとえば、アンダーセンコンサルティングがシステム開発をして納めた。納めたら引き上げてしまいますから、引き上げた後、運用サービスを全部 EDS が引き受けるわけです。引き受けていくうちに、だんだん業務に精通してくる。そして、次のバージョンアップであるとか、次の開発の時に、アンダーセンコンサルティングの人が偉そうに行くわけです。「そろそろ、あれですから、次期のシステム……」と言って行くと、「うちは、EDS に頼むからいいです。」という話になって、仕事を取られてしまうのです。

つまり、EDS の社員が仕事にずーっと精通してくるわけですから、アンダーセンコンサルティングが偉そうに來ても、もう状況が分からない、手法をいくら駆使しても、もうそういう問題ではなくなっているのです。このように取られるのでアンダーセンコンサルティングでは何を始めたかという、いわゆる EDS と同じことを始めました。

とにかく、開発が終わったらメンテナンスを全部引き受けるということです。これはアウトソーシングという言葉に今なっていると思うのですけれども、それに繋がってくるという感じですか。

だから、情報サービス産業というのは、きれいごとを言ってシステムカットして、おしまいにするのでは、もうユーザーは絶対に満足しないということです。最後のサービスまで引き受けるという覚悟でやっていかないと、ますます要らない存在になってしまうおそれがある。しかし、それに気がつけばそれにがっちり食い込んでいけばいい。

これからは企業の分捕り合戦になるのではないかなと思います。とにかく、自分のお客さんというのは一回おさえたら、逃さないようにしていかなければいけない。そうして、その中で、どんどんいいサービスを提供してその会社が発展していくようにやっていくと、そうすればお客さんも逃げないでいくでしょうし、サービスしていく側も仕事にどんどん精通していきますから、仕事自体が面白くなってきます。

そして何社か持っていれば、内部で動かせばその中で渡り歩くこともできます。そうすれば、1つの会社だけでなく、幾つかできるというのが情報サービス産業のメリットでもあるわけですし、楽しみでもあるわけです。そういうことが享受できるわけです。

一社の社員になってしまうと、その会社のことしかできませんけれども、外部にいと、いろいろな会社を見られるという良さがある。会社としては何社かをガッチリおさえこんでサービスしていくというやり方が取れると、情報サービス産業も非常に強みが出てくるのではないかと思います。

しかし、最大の敵は、もしかしたらIBMです。IBMはコダックの情報システム部門を買い

取って、アウトソーシング事業を始めています。これからそういうことを、どんどんやろうとしていますから、IBMはコンピュータメーカーではなくてアウトソーシングの事業で、私に言わせればユースウェア事業者ということで一括りになってしまうんですけども、そんなことになってきますから、最大の敵はIBMであり、つまりコンピュータメーカーが敵になってくると思います。それに打ち勝って、お客様を確保していくことが大事ではないかなと思います。以上。

#### (司会)

ソフトを受託したその後のサービスまで全部やりなさいと、そういうご指摘かと思います。

あと、橋山さん、何か関連でコメントありますでしょうか。

#### (橋山)

私は、最後にEDSの話をしようと思っていたのです。もちろん、皆さんEDSという会社はロス・ペローで有名になって日本EDSにも取材が相次いでいるそうですけれど。実は、私も数年前EDSの存在を知って、よくうちの若い人に言っていたのです。「情報システム部門ごとEDSに売っ払っちゃったっていいんだぜ。」さっき出たコダックのような企業が米国にはあるわけですから。「だから、僕の言っていた情報システム部門不要論が当たったじゃないか。」と言っているわけです。

日本でそこまでドラマティックにできる経営者はいないと思います。米国でも、実は引き取ってもらったあと1年しか契約がなくて、EDSを首になって訴訟問題を起こしていますから、そういう問題があることはあるんですけども、もはや情報システム部門がどうしても必要かということを真剣に考えないと、経営者から考えれば売り払ってしまったでもいいわけです、これはもう車と同じです。

私が最後にひと言、言いたかったのは、私は中年暴走族と言われるくらい車が好きで、今日も実は車でスッ飛ばしてきたんですけども、車も、昔を考えると、どこの企業にも車輻部というのがあったのです。40年、50年前は、車の免許を持っているだけで、イバっていたのです。各部門から運転していってくれとか、メシ、今晚一杯おごるからとか、特に新聞社に多かったそうです。今、そんな部門はないでしょう。なぜか。皆が車の免許を取ってマイカーになってきたからです。

だからマイコンっていうのも、マイは私のコンピュータというように考えれば、車の後を追いかけていて、車輻部と同じ運命をたどるのではないのでしょうか。頼んで頼まれて、「じゃあ、しょうがない作ってやるか。」なんて言っていると、情報システム部はなくなった方がいいと皆から言われる。

これはデータプロセッシング協会から、名前は変わりましたが、10年前に有名な論文とかエッセイが出たのです。これを最後のお話にしたいのですけれども『DP部門不安論・DP部門沈没』という有名なエッセイが出たんです、これは三菱油化の情報システム部の課長の方が書いた論文で非常に話題を呼んだのです。ちょうどあの頃、小松左京の『日本沈没』がベストセラーになった時代でした。

『DP部門沈没』という有名なエッセイがDP協会の『データマネジメント』という本に出たので、バックナンバーを調べるとあるみたいですが、経理部と人事部長が帰りの電車の中で一緒になって、「ときに、あの評判の悪かったDP部門はどうなったんですかね。マイコンの技術にも遅れて、パソコンも強力に反対していたし、電子メールも反対して全て、大型計算機でやろうと、メーカーの手のひらで動いていた部門はどうなったんですか。」「あんな部門は皆がなくなった方がいいと言っていたんで、なくなったんじゃないですか。」という書き出

しで始まって、非常に話題を呼んで、その後『続 DP部門沈没』が出たくらいですから、ぜひこれをお読みいただくと、10年後の今を完全に見通しているのです。我々はこういう部門になってはいけないのではないかと、10年前に当時の部長と話した記憶が残っています。

皆さん、ぜひ、興味のある方はそれを読んでいただきたいと思います。

### (司会)

かなり過激な話が出たと思います。

あと10分少々時間がありますので、ご質問のある方お願いします。

### (船戸)

名古屋市工業研究所の船戸です。エンドユーザーコンピューティングのエンドユーザーがおかしいというのは同感です。ホストはバックエンドですからユーザーが一番中心だと思います。

けれどもユースウェアという観点で、特に日本の場合は日本語処理というか、漢字の処理はやっぱり、まだまだではないかと思うのです。それは入力段階でもそうですし、あるいは出力の段階でも、さきほどちょっとOHPで誤字じゃないかと思うのがあったのですけれども、シンタクスチェッカーというのは日本語だとない。文章のチェックはなかなかできないといった形で、文書処理もこれからだし、そういった意味でのユースウェアという部門も、まだまだこれからではないかと思います。

それから、片貝さんの方は、ペン入力をあまりご評価なされていないように聞き取ったのですが、橋山さんの中で直行直帰される営業の方ってお話がありましたけれども、やはり直行直帰される営業の方というのはペン入力のマシンが欲しいのではないかと思うのです。ですか

ら、ペン入力のマシンっていうのがこれからエンドユーザーではなくてユースウェアという観点でどんな影響を与えるか、ということについてご意見をいただきたい。

もう一点はさきほど、片貝さんの方から、自分の環境として完璧なものになるようになってから、お客さんに勧めなさいというお話があったのですが、多分SEAに参加している方はそれなりに自分の環境を作っていて、むしろ開発のための環境としては結構ガッチガッチに作っているのですけれども、それが、ユーザーにとって本当に必要な環境かどうかというところが、必ずしも上手くいってないのではないかと思うのです。そのへんについて何かコメントがあればいただきたいと思います。

#### (司会)

最初のペン入力とユースウェア、これは片貝さんの方から。

#### (片貝)

確かにおっしゃるとおり、漢字の問題はとにかく、私はローマ字日本語変換からやっているし、カナから入るのは、いろんな人の話を聞くと、まあ専門家以外は不可能で、ローマ字から入るのですが、しかし、スペルチェックは全面的にできないわけで、非常にまだ、プアーだと思います。この部分はぜひ、専門の方にもっと研究していただきたいと思います。どういうふうになるのが一番素晴らしいのか私にはよく分かりませんが、とにかく、AI的な機能は絶対に必要になってくるわけで、誰もが文脈を見れば分かるのに、とんでもない字が出てくるようなカナ-漢字、ローマ字-漢字変換しかできないので、これは非常にかったるいです。よく見ないと字を間違えてしまう。ワープロで間違えた字っていうのはすぐ分かります。そういうのが出てきてしまう。これはぜひやらなければいけないことです。

ただ、私が思うのは、ジャストシステム、私

はジャストではないかなと思うのですが、とにかく、FEPを切り出してくれたことです。これをものすごく評価していいと思うのです。つまり、過去においては、今まで日本語入力の部分を何かアプリケーションで作る人は、自分で日本語入力を作らなければいけなかったわけです。それを切り離してくれたので、どこへもくっつけられるようになった。2万円か3万円で買ってくれば、そこに入れられるという、これは”国民栄誉賞”ものだと思うのです。ただ、それで止まっているから、もう少し何とか進めてほしいというのがあります。

それから、ペン入力について、私は実は否定しているわけではないのです。とにかく、キーボードを先にやってくださいという事なので、その上で、入力方法なんていっくらでもあった方がいいと思います。しかし、キーボードをちゃんと覚えなくて、ペン入力だなんていうことだけから入るとするのは損ですよと言いたいです。

とにかく、キーボードから。電子メールを打つに、マウスでもペンでもなかなか打てないわけです。電子メールっていうのはしょっちゅう、やる必要がありますから、もうバタバタと打てる方がいいわけです。だから、とにかく、普通の人にはキーボードをちゃんと覚えてほしい。その上で、TPOに応じペン入力もあるだろうし、それ以外の音声入力もあるだろうし、いろんなものがあると思うのです。ただ基本はキーボードであるというのが私の考えです。

#### (橋山)

ご質問にあったように、今、セールスマンに、実はインプットはPOTパーソナル端末、アウトプットはファックスを自宅にということをやっていたのですけれども、ファックスですから出張から帰ってきて1週間いないと6メートル位出ていて、何が最優先で見なければいけないのか分からない。これはもう4,5年前から始めたので、まだ、パソコン通信が使える環境にな

かったということで、これから、そういうものはパソコン通信に繋がってくると思います。

今度は、セールスマン全員数千人にノートブックよりも、もっと小さいものを持たせようと、パームトップとノートブックのあいのこぐらいです。これは、強引に主張して、将来は32ビットのwindowsが動くものにしてくれ、そのCPUは486を積んでくれと言っているわけですが、実はペン入力になっています。ただし、あれが使えるかどうかは、我々が決めるのではなくて、セールスマンが決めるわけです。いくら僕らがびっくりするほど認識率が高くなっていると感じても、我々が決めるのではなくて、使う方が使わなかったら終わりですから、果たしてペン入力を使ってくれるかどうか。

現場ではスピードが要求されますから、やはり後はタッチです。ある程度、パターン化しておいてタッチ、これと併用になると思います。現場で担当ですぐインプットしなければいけないものはパターン化できますから、そういうものはタッチで入れて、帰ってきて何か処理をするときにペン入力を使うという併用になると思うのです。

我々はやはり、エンドユーザーコンピューティングを進めていくときに、エンドユーザー、私もエンドユーザーという言葉が嫌いだと言いながら使っていますけれども、やっぱり、普及した方が勝ちなのです。パソコンも、私、10年間やっていて、イヤッていうほど分かったのが、いいものが普及するのではなくて、普及したものがいいものということです。

ですから、10年前から私はビジネスでは98を使おうと思っています。普及しているものはこれだけです。しかし、自分で買うのであればMcCintoshを買うとよく言っています。ビジネスと趣味とは違いますから。ビジネスではベータとVHSを見て分かるように、普及して

いくものを採用していかないと、企業としてはまずいということになってきます。これはハードだけではなくてソフトにも全部そうです。

これから何が普及していくのか、それこそ片貝さんもさっきおっしゃったようにWindows NTかOS/2かUNIXかサーバーOSか、それはもう分からない。しかし、我々情報システム部門のひとつの役割は、この辺をウォッチしてこういうものが普及していくんだからこれを使えとガイドラインを出していくわけです。そのガイドラインを出せなくなったら情報システム部門は本当にいらなくなると思います。

たとえば、うちではどういうガイドラインを出したか。去年の5月に、ネットワークOSはNetWareでいく、それからMS-windowsでいく、いわゆるソフトの標準を図ったわけです。ハードは98でも何でもいい、IBMでもいいし、富士通でもいい。クライアントOSはMS-windows、ネットワークOSはNetWare、それでいくんだと。それを守ってくればアプリケーションは何でもよろしいと。

それから、ホスト系は基幹系は再構築を行なう。エンドユーザーコンピューティングとかエンドユーザー系のデータベースは再構築するから、それにさっき言ったExcelに取り出すとかそういう仕組を我々と外部のソフトウェアハウスさんと提携して作り上げる。もちろんIBMさんに協力してもらいましょう。そういう仕組を作って、それから教育していくということで進めようとしています。

とにかく、この世界は普及したものがいいものだけということ、イヤッていうほどこの10年間で我々は分かったので、さっきから出ている漢字入力のところもVJE-γがいいのかATOKがいいのかというのも、それぞれ好き嫌いですからそこも規定をしません。

今MS-windowsの上で動いているOASYSが試

作版として、うちに入ってきています。MS-windowsの上でOASYSが動かないと使えませんよとはっきり富士通に突きつけていますので、多分OASYSのwindow版が出てくることは間違いない。

またゼロックスにもJ-Starは非常にいい、ただ、あの機械でしか動かせないのではPIPSの二の舞になるぞ、ソードの二の舞をやる気かと、だからwindow版のJ-Starを作ってくださいと言っています。

ジャストシステムにはwindow版の一太郎を作ってください。Just-windowsもいいけれど、やっぱりMS-windowsが標準になってきたら、そういうものを作っていかなければ駄目だと。

これはダイナウェアにも同じことを言っています。DYNA-windowsが普及するのではなくて、やっぱりマイクロソフト・ウィンドウが普及するのだから対応をしてくれ、それ以外はうちは使わないと。

普及したものがいいものなのです。いいものが普及するというのは、いいものが普及するのではなく、いいものを普及させたいという技術者の驕りなのです。いいですか、これは技術者の驕りです。自分の作ったものを普及させたいというのはよく分かります。しかし、それはそうなるとは限らないということです。普及したものがいいものなのです。我々のエンドユーザーコンピューティング戦略はそれなのです。普及したものを使うということ、もしくは、これから普及すると思われるものを見極めて、間違えることはあるかもしれませんが、そういう戦略でやろうと思っています。

(司会)

もう1点、環境の問題があったのですが。

(片貝)

それだけちょっと簡単によろしいですか。

多分、電子メールとかそういう関係はもう困っていないと思うのですが、プロックスというIBMのプロフェッショナル・オフィス・システム、これはIBMのSEが自分達のために作ったもので、つまり、グループウェアです、社内の電話帳とか、いろいろな情報の共有化とかやっているの、そういうものを作ってみたらいいと思います。作るか、または私どもが売っていますので買っていただくとか。そういうものを使って見ると、お互いに何人かの人で仕事をしているわけですから、その間の技術情報の交換とか、いろんなことは情報の共有化ということですぐにできると思われま。

とにかく、10人の人がいちいち話をするためには $10 \times 9 \div 2$ なのです。だから、45のパス、コミュニケーションパスがあるわけで、リーグ戦と同じわけですから、一カ所に情報を書いておけばすぐに共有化できてしまうということはかなり便利だと思います。そんなことをやってみると、そのままユーザーに繋がっていく話ではないかと思ひます。

(司会)

では、もうお一方と思うんですが。どうぞ。

(小林)

コンピュータ・コンサルティングの小林と申します。

エンドユーザーコンピューティングで、これから重要になっていくのはトラブル対応だと思います。LANを張って、電子メールでやり取りして、データをシェアして、いろいろなツールが使える、というような環境になっていくと思うのですが、おそらくトラブルが絶対に起こるはずで、サーバーがクラッシュしてしまったとか、自分のマシンがクラッシュしてしまったというときに、本当にエンドユーザー自身がそれを全部回復できるのかということです。

エンドユーザーというのはやはり業務のプロ

であって、その道具としてコンピュータを使っているわけですから、環境を修復するというのはなかなか難しいと思います。確かに、EDP部門不要論とかいうのがあって、その通りだと思おうのですが、おそらく、これからは社内のEDP部門が、環境という空気のようなものを面倒をみていかなければいけないのではないかと。外側にいる情報サービス産業の人間も、そういうことを面倒をみてやれるようにならなければいけないのではないかと、私は感じているのです、これからエンドユーザーコンピューティングがどんどん進んでいくと、逆に新しいビジネスチャンスが生まれてくるのではないかと考えているのですが、その辺いかがでしょうか。

(岡田)

私の方から、今のトラブル関係のフォローについてお答えします。実は、社内システムをやる時も、このへんが非常に難しかったところだと思うのです。ただし、うちの場合は、そばに専門家のいる会社でしたので、ある程度の割り切ったところがあります。しかし、一般のユーザですと、いまの質問の通りの状況があると思います。そのためにはトラブルの起こりにくいようなハードとソフトを提供する。たとえばデータベースでしたら最近ミラー化されているものも出ていますので、そういったものとか、ハードが落ちたときは自動リカバリ機能を付けるとかが必要かと考えています。その上で環境に対応するサービスをしていくことが最善であり、ご指摘のとおりだと思います。

(司会)

それでは、時間がちょっと過ぎましたがこれでクローズイングパネルを終わりにさせていただきます。と思います。

最後にお三人の方に拍手をお願いしたいと思います。

文責 市川(電算)

## LPF アピールをめぐって (続き)

前号掲載の「LPF アンケートを読んで」に対して2人の方からコメントが寄せられました。他の回答者の方々も、またその他の方々も、ぜひ論争に加わってください。

### 「LPF アンケートを読んで」を読んで 新田 稔 (SRA)

回答者 No.13 さんは、LPF からの提案で特許を反対しようとしている「ソフトウェア」について全く勘違いをしている。それは No.13 さんの文面に「一太郎...」の例が登場することから最もよくわかる。LPF が特許に反対しようとしているのは、そういう個々のプログラムではなく、もっと基本的事柄な、たとえばものの考え方とかアルゴリズムである。

No.13 さんは、一太郎が売られているのは現行の特許法のおかげだと主張されているが、全くその通りである。そしてそれは、現行の特許制度が、LPF が反対しているソフトウェアの特許をほとんど認めていないからだ。もしこのソフトウェアの特許が認められていれば、ワープロソフトなどは特許にがんじがらめにされ、個人が買えるような価格では到底販売できなかつただろう。

No.13 さんはハードウェアに関する特許の経験から、基本的な特許ほど良い特許であるとおっしゃっているが、それはハードウェアの特許には、あまりにも基本的なものは誰にでも既知・自明であるとわかる(したがって特許としては認められない)という安全弁があるからだ。

残念ながらソフトウェアでは、ソフトウェア技術者にとっては自明であることが特許の審議を行なう者にとっては自明でないということが少なくない。いま私たちがソフトウェアの特許を阻止しておかなければ、ソフトウェア産業の将来に健全な競争と発展は望めない。

最後に蛇足ながら。ソフトウェアに関する知識の欠如をハードウェアからの類推で補って、ソフトウェアもそうあるべきだ、あるいはそうに違いない、と主張するのはもうやめにしませんか。ソフトウェアは本質的にハードウェアとは異なるということ認識し、ソフトウェアについてわからないことはわからないとして、みんなで考えていきましょう。

### An Objection 回答者 No.24

ソフトウェアの実体は、そこに内包された「モデル」であると考えられる。何のモデルかといえば、それはコンピュータ化の対象として考えられたデータ処理プロセスのモデルである。

すべてのモデルは、意味と表現を持つ。

あるプログラミング言語によるモデルの「表現」を著作権によって保護しようというのが、現在のプログラム保護の立場であろう。ソフトウェアの最終形態が何らかの言語を用いた表現になる以上、この考え方は一見正しいように思われる。しかし、有名な「ソフトウェア進化論」のパラダイムが指摘したように、モデル化対象のプロセスは、それがこの社会の中で生きている人間の営みに密接に関連している限りは、絶えず変化する。したがって、その表現としての

プログラムは、著作権によって保護されているふつうの著作物とちがって、つねにだれかの(ふつうは原作者とは別人の)手で改変されなければ、実際の役に立たない!(ソフトウェアの保守がハードウェアのそれとまったく異なるのはそのためである)

もちろん、中には例外的に、変化のスピードがきわめて遅い(または永久に変化しない)プロセスもありうる。たとえば、連立方程式の解法 etc である。ソフトウェア進化論でいわれる S-type プログラムに属するような、そうした例外的プロセス・アルゴリズムについては、特定の言語を用いたその表現を著作権によって保護することが、もし著作者であるプログラマ自身が望み、またそこに、余人には真似のできない高度なコーディング技法が用いられているならば、妥当であろう。

ワープロやゲームソフトなども、比較的变化の遅いプロセス・モデルの表現であり、特定の言語またはハードウェアを前提とするその表現は、もし作者がそれを望むなら、不法コピーなどの権利侵害に対して(何らかの法的メカニズムを通じて)守られるべきだと思う。

OS やコンパイラ、あるいはエディタといった基本ソフトはコンピュータ文化を支えるインフラストラクチャであり、無料ですべての人々に配られるべきだという FSF の主張は、やや別の次元に属し、それに賛同するか否かは個人の自由であろう。ただ、現実に FSF がこれまで行って来た努力には敬意を表せざるを得ない。GNU Emacs, Gnu CC などのフリーソフトのユーザ数はすでに日本でもかなりの人数に達しており、それらを利用するがビジネス上の利益につながると考えているユーザたちは、FSF に対して有形/無形の支援をこれまでもしてきたし、これからも続けるであろう。

さて、ソフトウェアの特許は、いったい何をどのような形で保護しようというのであろうか?

もし、ソフトウェアの実体である「モデル」をその表現形態のいかんにかかわらず保護しようというのであれば、それはナンセンス以外の何者でもないであろう。なぜなら、その場合、哲学あるいは数学の分野に属するほとんどすべての成果がその保護対象になってしまうからである。LPF アピールに挙げられたいくつかの例題(XOR をもちいた bit の反転 etc)は、そうしたカテゴリに属する。もともと人類共通の知的財産であるものを、だれかが勝手に申請し登録するだけで権利を主張するというのは、暴挙というほかはない。

モデルはモデルでしかない。それに意味を与えるのは、ユーザである。モデル作りの専門家である数学者や哲学者はそのことをよく心得ており、特許申請などを決して考えたりはしない。われわれソフトウェア技術者もその謙虚さを学ぶべきであろう。

# SEA Seminar & Forum (February 1993)

## 参加者募集



新しいスタイルに変わって2回目のSEA月例Seminar & Forumでは、しばらく長引きそうな構造不況をちょっと忘れて、ソフトウェア分野における国際交流または国際分業の今後の展望について討論することを考えました。東西冷戦構造が崩壊した後の世界において、日本のソフトウェア技術あるいはソフトウェア業界は、どのような立場からどんな貢献をなしうるのか、また、なすべきなのか？

幸いわがSEAには、そうした国際活動に豊かな経験をお持ちのメンバーが大勢いらっしゃるのです、それぞれの立場からの多角的な御意見がうかがえると思います。ふるって御参加ください。

### 開催および申込要領

- 日時： 1993年2月16日(火) 14:00～20:30
- 場所： 機械振興会館 地下3F 研修室 (東京・港区芝公園・東京タワー前)
- プログラム： Seminarは14:00～17:30, Forumは18:30～20:30.

**Seminar: 「ソフトウェア開発における東洋的アプローチ」** 講師： 熊谷章 (PFU 上海) & 松原友夫 (Office Peopleware)

これまでのソフトウェア開発方法論は、西欧的な思想(科学哲学)の枠組みにもとづくものがほとんどであったといつてよいでしょう。その結果、今日のような"ソフトウェア危機"がもたらされ、しかもその解決の見通しは依然として立っていません。明治の革命以後、独特なやり方で東西文化のミックスに成功をおさめてきた日本としては、このあたりで何らかの国際貢献をなすべきであろうと思われます。このセミナーでは、新たに上海で日中合作のソフト開発事業に取り組まれる熊谷さんと、これまでISOやIEEEなどの舞台で多彩な活動をされてきた松原さんのお2人を講師にお迎えして、中国哲学やインド思想のソフトウェアへの応用について語っていただきます。これからの開発パラダイムの転換を暗示するようなユニークなアイデアの数々がうかがえるものと期待されます。

**Forum: 「Multi-National Software Development: その現状と将来」** コーディネータ： 杉田義明(日本NCD)

パネリスト(予定)： 片山禎昭(東芝アドバンスシステム) 岸田孝一(SRA) 熊谷章(PFU 上海)

篠原令(アスキー) 松原友夫(Office Peopleware) 他

これまで、国際的な場におけるソフトウェアの協同研究や協同開発は、欧米を相手にした場合には、テクノロジーギャップの存在を前提とした向うからこちらへの技術移転、アジアを相手にした場合は、経済各差を利用した下請け外注がほとんどでしたが、いつまでもこうした状況をそのままにしておいてよいとは思えません。このパネルでは、上記のセミナー講師お2人に加えて、さまざまな国際活動の経験をお持ちの会員各氏をパネリストに、これからの国際間の協同開発/研究のあるべき姿、またそこで日本が果たすべき役割を探ってみたいと思います。

### 4. 参加費：

Session	SEA 正会員	SEA 賛助会員	一般
Seminar(午後)	10,000円	15,000円	20,000円
Forum(夜)	2,000円	4,000円	6,000円

- 申し込み方法： 下の申込用紙に必要事項を御記入の上、SEA事務局までFax、郵便またはE-Mailでお申込みください。折り返し受講票をお送りします。SeminarまたはForumのみの受講も可能ですし、もちろん両方続けての参加も歓迎します。なお、参加費は当日会場受付にてお支払いください(領収書を差し上げます)。申込受付後のキャンセルは原則としてお断りします。

申込み宛先： 〒160 東京都新宿区四谷3-12 丸正ビル5F ソフトウェア技術者協会(SEA)

Tel: 03-3356-1077 Fax: 03-3356-1072 E-Mail: sea@sea.or.jp

### SEA Seminar & Forum (Feb. 1993) 参加申込票

氏名(ふりがな)： \_\_\_\_\_ ( \_\_\_\_\_ )

会社名： \_\_\_\_\_

部門・役職： \_\_\_\_\_

住所：(〒 \_\_\_\_\_ ) \_\_\_\_\_

Tel: \_\_\_\_\_ Fax: \_\_\_\_\_

種別(該当欄にチェック)：  正会員 (No. \_\_\_\_\_ )  賛助会員 (No. \_\_\_\_\_ )  一般

参加Session(該当欄にチェック)：  Seminar  Forum 参加費： \_\_\_\_\_ 円

# SEA これからのイベント開催予定

## 1993年2～6月

2月12日(金): **SEA Special Forum in Shanghai** @ 花園飯店(中国・上海)

今年の国際CASE Symposium (in 山東省) の企画打合せを兼ねた日中技術交流 Forum.  
参加希望者はSEA事務局まで御連絡下されば詳細をお知らせします。

2月16日(火): **SEA Seminar & Forum (February)** @ 機械振興会館(東京)

詳細は裏面参照。

2月18日(木): **TISA Seminar** @ 宮城第一ホテル(仙台)

主催: 東北地域情報サービス産業懇談会(TISA) 協賛: SEA

プログラム(予定)

ダウンサイジング時代のソフトウェア技術者に求められるもの: 堀江進(NES)

エンドユーザにおける分散型アプリケーション開発: 講師交渉中

Unixを用いた社内情報システムの分散化: 岸田孝一(SRA)

2月26日(金): **SEA 全国縦断 Forum in 金沢** @ (会場未定)

テーマは"フォーマルアプローチ: ドライか, ウェットか?" 詳細は現在企画中。

3月4日(木): **SEA Special Tutorial on Object Orientation** @ 青年会議所会館(東京)

"オブジェクト指向方法論 OMT" の著者 Dr.W.Premarlani(GE) を講師に迎えての特別チュートリアル。  
逐次通訳は同書の監訳者・羽生田栄一氏(FXIS)。

3月15日(月): **SEA Seminar & Forum (March)** @ 機械振興会館(東京)

テーマは"フォーマルアプローチ: ドライか, ウェットか?"

詳細は現在企画中。

4月9日(月): **SEA Seminar & Forum (April)** @ 機械振興会館(東京)

企画はこれから。

4月21日(水): **SEA 全国縦断 Forum in 仙台** @ (会場未定)

企画はこれから。

4月22日(木)～24日(土): **第5回 テクニカル マネジメント ワークショップ** @ 山形(会場未定)

テーマは"業界リストラクチャリングとテクニカル マネジメント".

実行委員長は野村敏次(JIP)& 山崎朝昭(ENICOM)のお2人。詳細は現在企画中。

5月28日(金): **SEA Seminar & Forum (May)** および年次総会 @ 機械振興会館(東京)

企画はこれから。

6月9日(水)～11日(金): **Software Symposium '93** @ 戦災復興記念館(仙台)

現在, 論文募集中。

\*\*\* 事務局からのお知らせ \*\*\*

今年からSEA事務局は毎週水曜日お休みとさせていただきます。

Fax(03-3356-1072)または留守電(03-3356-1077)またはE-Mail(sea@sea.or.jp)を御活用下さい。



**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