



SEAMAIL

Monthly Newsletter from
Software Engineers Association

Volume 2, Number / January 1987

目次

編集部から		1
SEA 情報化月間記念フォーラム		
パネル討論「ソフトウェア産業の未来」	近藤・岡田・広松・佐野・大塚	2
セミナー 21 レポート-その4-		
環境構築と技術移転	岸田孝一/国布 浩	16
データ構造に基づくソフトウェア設計の一方法	白井義美/野辺良一	22
初めての海外調査	道正 一郎	26
ダメ管理者の分類とその対応	阪井 誠	29
ソフトウェア・シンポジウム '87		31
カレンダー		33

ソフトウェア技術者協会（SEA）は、ソフトウェア・エンジニアの、ソフトウェア・エンジニアによる、ソフトウェア・エンジニアのための団体であり、これまでに日本になかった新しいタイプのプロフェッショナル・ソサイエティたることを目指して、1985年12月20日に設立されました。

現在のソフトウェア技術が抱える最大の課題は、ソフトウェア・エンジニアリング研究の最前線（ステイト・オブ・アート）と、その実践状況（ステイト・オブ・プラクティス）との間に横たわる大きなギャップを埋めることだといわれています。ソフトウェア技術の特徴は、他の工学諸分野の技術にくらべて属人性がきわめて強い点にあります。したがって、そうしたテクノロジー・トランスファの成否の鍵は、研究者や技術者が、既存の社会組織の壁を越えて、相互の交流を効果的に行うためのメカニズムが確立できるか否かにかかっています。SEAは、ソフトウェア・ハウス、計算センタ、システム・ハウス、コンピュータ・メーカ、一般ユーザ、大学、研究所など、さまざまな職場で働く人々が、技術的・人間的交流を行うための自由な場であることを目指しています。

SEAの具体的な活動としては、特定のテーマに関する研究分科会（SIG）や地方支部の運営、月刊機関誌（SEAMAIL）の発行、各種のセミナー、ワークショップ、シンポジウムなどのイベントの開催、既存の学会や業界団体の活動への協力、また、さまざまな国際交流の促進等があげられます。

なおSEAは、個人参加を原則とする専門家団体です。その運営は、つねに中立かつ技術オリエンテッドな視点に立って行われ、特定の企業や組織あるいは業界の利益を代表することはありません。

代表幹事： 鈴木弘

常任幹事： 岸田孝一 長井剛一郎 盛田政敏 吉村鉄太郎

幹事： 稲田博 白井義美 大木幹雄 岡本吉晴 落水浩一郎 皆藤慎一 木村高志 久保宏志 熊谷章 斎藤信男 三枝守
正 芝原雄二 杉田義明 辻淳二 鳥居宏次 中園順三 針谷明 藤野晃延 松本崇純 松原友夫 水谷時雄 三浦信之 村井進

会計監事： 近藤秀朗 吉村成弘

常任委員長： 岸田孝一（会誌編集） 盛田政敏（企画総務） 吉村鉄太郎（技術研究） 杉田義明（セミナー・ワークショップ）

分科会世話人 環境分科会(SIGENV)：岡本吉晴 久保宏志 引地信之 松尾正敏 水谷時雄

管理分科会(SIGMAN)：岸田孝一 塩野富教 芝原雄二 鈴木信裕

教育分科会(SIGEDU)：大浦洋一 杉田義明

再利用分科会(SIGREUSE)：青島茂 阿倍正平 村井進

AI分科会(SIGAI)：安倍昭敬 坂下秀 白井豊 高田佳彦 広川昭八 野辺良一 藤野晃延 横山憲一

ネットワーク分科会(SIGNET)：鈴木弘

法的保護分科会(SIGSPL)：能登末之

CAI分科会(SIGCAI)：大木幹雄

支部世話人 関西支部：白井義美 盛田政敏

横浜支部：熊谷章 林香 藤野晃延 松下和隆

SEAMAIL編集グループ：大概亮人 岸田孝一 佐原伸 沢田寿実 芝原雄二 関崎邦夫 田中慎一郎 長井修治 野辺良一

藤野晃延 山内徹 渡辺雄一

SEAMAIL Vol. 2, No. 1 昭和62年1月1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）

〒166 東京都杉並区高円寺南1-5-4 高円寺サンハイツ404

印刷所 サンビルト印刷株式会社 〒162 東京都新宿区築地町8番地

定価 500円

編集部から

ながらくお待たせしました。表紙のデザインと色が変わったSEAMAIL-第2巻第1号をお届けします。

年の最初から船便になるのを避けるべく、編集ボランティア一同、できるだけ努力をしたのですが、すでにお手紙でみなさんに御説明したような諸般の悪条件が重なって、これまでの遅れの記録を更新するという「名誉ある」結果に終わってしまいました。

年頭にあって、編集部の某君が東京・赤坂の日枝神社に初詣にいて、SEAMAIL月内発行の願をかけてきたのですが、本人自身がその後すぐ骨折する始末。不信心者が慣れないことをしては、かえって逆効果のようです。

第2号は、ほぼ平行して編集が進んでいますから、1-2週間後には、お手元にとどくと思います。

お手紙でもお願いしたように、船便の遅れをなくすために、なんでも結構ですから、原稿をお寄せください。SEAMAILは、ふつうの学会誌などのようにフォーマルな雑誌ではなく、あくまで、SEAの会員が、相互に技術的・人間的交流を行うための気軽なメディアです。当然のことながら、論文の査読などという面倒な手続きはありません。みなさんから送られた原稿を、そのままフォーマットして掲載するだけです。忙しくてまとまったものが書けないというのであれば、「会員の声」欄への短いメッセージで結構ですから、ハガキまたはFAXで事務局あてお送りください。

情報化月間記念フォーラム

昨年10月に行われた、情報化月間記念フォーラムの最後のレポートとして、パネル討論「ソフトウェア産業の未来」のほぼ忠実な記録を巻頭に載せました。当初は半分ぐらいに要約することも考えたのですが、異色のメンバーによる示唆に富む議論なので、そのまま全体を掲載したほうが、当日出席できなかった会員の方々の参考になるだろうというので、思い切ってページをさくことにしました。

また、「未来」というテーマも、年頭の第1号にふさわしいと思います。第1巻第7号にとりあげた「ソフトウェア産業をめぐる誌上討論」と併せて読んでいただくとよいでしょう。読後の感想や御意見は、やはり、原稿かメッセージのかたちで、どうぞ。

セミナー・ウィーク特集

昨年から続いている、秋のセミナー・ウィーク・レポートの「その4」です。今回は、B1、E2の2セッションを収録しました。これも、会員の方からの要望にしたがって、少しページ数をふやし、セミナーの臨場感がつたわるよう、レポートにがんばっていただきました。

これで、全21セッションのうち、2/3の報告が終わったこととなります。3月には「春のセミナー・ウィーク」が実施されますので、あと2号で残りを全部報告したいと思っています。

会員からの投稿

道正さんに「アメリカ技術調査」のエッセイを寄せていただきました。本文中にある2つのコンファレンス(CCSW'86とPSDE)の報告を兼ねた12月フォーラムのレポートは、次号に掲載する予定です。

また、今年の3月末から4月にかけて開催される第9回ICSEには、SEAが協賛団体ということもあって、すでに別便でご案内をしたツアーには、30人以上の会員が参加されます。その報告エッセイは、あらためて何人かの方々に書いていただくつもりです。

支部からの報告

関西支部の月例会からのレポートを掲載しました。この他に、YDOCからの原稿も届いているのですが、すみません、整理が間に合わず、次号まわしになりました。最近、分科会からの成果報告が少ないようですので、それぞれの世話人の方よろしく。

幹事会報告およびSIG案内

今月は、ページ数の関係で割愛しました。近くSIG世話人の会合が開かれ、これからの活動方針などが議論される予定です。結果は、次号で報告します。なお、本誌がしばらくは船便状態を脱却できそうもないので、SIGの開催案内など時間的な制約のある情報は、これからは、別途簡単なお知らせのかたちでお送りするようになるでしょう。先日の幹事会で、4月に「売上税」の特別フォーラム、5月の総会を兼ねて「SDI」のフォーラムを開催することが決まりました。これらの案内も、いずれ別便で。

SEA 情報化月間記念フォーラム

パネル討論

ソフトウェア産業の未来

コーディネータ： 鈴木 幸一（日本アプライド・リサーチ研究所）
パネリスト： 近藤 隆彦（通商産業省）
： 岡田 康司（ソフト化経済センター）
： 広松 毅（東京大学）
： 佐野 邦雄（ジョイント・デザイン・システム）
： 大塚 隆（朝日新聞社）

1. はじめに

鈴木： 今日の討論のテーマは、「ソフトウェア産業の未来」ということになっていますが、私を含めて、この壇上に並んでいる人間はすべて、会場のみなさん方とはちがって、専門的技術知識とか業界の裏表についての情報を、あまり持ち合わせていません。それを逆手にとって、いわば岡目八目の立場から、この産業の将来について、多角的に議論できればいいのではないかと考えています。

そんな意味で、パネリストとしては、できるだけバラエティに富んだ方々にお集まりいただきました。しかし、そのために、また、逆に話が絞りにくくなるかも知れません。そのへんは、みなさんからの御質問に答えるという形で、討論の方向を何とかコントロールしていきたいと思えます。

先程、昼食をとりながら、パネリストの方々と簡単に打ち合わせをしたのですが、「ソフトウェア産業は、今の日本のなかで、最も成長率の高い産業である」というのが、われわれの共通の認識です。これは、もしかしたら誤解かも知れません。そういう目で会場を見ると、みなさんの頬がバラ色に輝いて見える（笑い）。そういった誤解を含めて、5人のパネリストの方々から、それぞれのお考えをうかがうことにしましょう。

それでは、まず最初に、国（あるいはお役所）は、コンピュータ業界やその中におけるソフトウェア業界の将来をどう見ておられるのか、通商産業省・機械情報産業局・情報処理振興課の近藤課長さんからお話いただくことにしましょう。

2. 霞が関からの眺め

近藤： お話に入る前に、通産省内部のコンピュータ関係のセクションがどんな仕事をしているか、簡単に紹介させていただきます。

2.1 MITIの役割

大きく分けて、ハードとソフトがあるわけですが、ハードに関しては電子政策課と電子機器課の2つがあります。電子政策課の主な仕事は、日本のコンピュータ産業の育成ということとして、そのために必要な技術開発の促進を担当しています。たとえば、第5世代コンピュータ・プロジェクトなどは、ここの管轄です。一方、電子機器課は、もうすこしビジネスよりの仕事、たとえば最近では、半導体の開発や輸出入に関する日米交渉などを担当しています。

ソフトウェアの関連では、2つの課があります。その1つが、私のいる情報処理振興課です。この課ができたのは約15年前で、それ以来、いわば、日本のソフトウェア産業の歴史とともに歩んできたといってもよいでしょう。業界の発展につれて、ソフトウェア関連の問題も数多く発生し、おかげで、われわれの仕事もどんどん広がっています。

開発生産性の向上や技術の汎用化といった問題に関しては、みなさんが御存じのように、IPAを通じて、いろいろな技術振興政策を実施しています。最近では、また、ソフトウェア技術者育成のための長期的な施策として、学校教育におけるコンピュータ・システムの導入を促進するために、文部省と協力して、CAIのソフト/ハードの開発もやっています。

みなさんおなじみの情報処理技術者試験も、われわれ

の課の担当ですし、コンピュータ・システムのセキュリティに関しては、安全事業所の認定も担当しています。安全問題に関しては、この他に、システム監査やコンピュータ犯罪も扱う必要があります。

また、ソフトウェアの法的保護問題に関しては、著作権法の関連で、文化庁と協同作業をしています。最近では、ガットの間でも知的所有権の問題が議論されたりしていますので、現在いろいろな角度から勉強中です。おそらく、この問題は、これからのソフトウェア産業にとって、きわめて重要な問題になるのではないかと考えられます。

ソフト関連では、もうひとつ情報処理システム開発課というセクションがあります。ここでは、特定の地域または特定の業務を対象にした公共的なシステムの構築を行っています。たとえば新聞をにぎあわせているニューメディア・コミュニティとか、一定範囲内のネットワーク化、医療情報システムなどがそうです。

2.2 産業としての複雑さ

さて、本題にはいって「ソフトウェア産業の未来」ということですが、これはとてもむずかしい問題で、一口に国の立場でといわれても困る。つまり、いろいろな人がいろいろなことをいえると思います。

まず、ソフトウェア産業だけでなく、広くそのまわりを含めた情報サービス産業全体の今後の課題を考えてみますと、そこには、実に多種多様な問題が含まれていて、それらをいちいち細かく分析してみないと、それらが一体どうからみあっているのかが掴みきれない、というむずかしさがある。

同じ情報サービスといっても、サービスの中身が違うということがあります。ソフトウェアの開発という仕事も、お客が違えばかなり違うものになってくる。つまり、どういった業種を相手にして、どんなソフトを作るかによって違います。ときには地域によっても違って来でしょう。

また、ソフトウェア企業と顧客との関係も、ひとつの重要なファクターでしょう。資本関係を抜きにしても、特定のメーカやユーザに非常に近いソフト・ハウスもあれば、中立的なところもある。さらに、仕事の形態にも請負型と派遣型という2つの形態がある。このことは、さきごろの派遣法にからんで、大いに問題になったところだ。

このようにみえますと、情報サービス産業の未来は

かくあるべしとか、こうあったほうが望ましいとかいった形で、単純に一本化しては考えられないという感じがします。

2.3 技術の汎用化と特化

次に、ここにお集まりのみなさんの御専門である技術の問題があります。いまさらいうまでもなく、ソフトウェア技術の進歩・発展はきわめて激しいわけですが、それは産業構造や企業組織に大きな影響を与えます。たとえば、技術の汎用化をめざしたツールができるとか、生産性向上のために1人1台のワークステーションが導入されるとかいったことが起これば、ソフトウェアを作っている人たちの毎日の生活にかなりの影響がでてくるわけだ。

おそらく、これから5年とか10年とかの間に、こういった技術革新によるソフトウェア産業の構造的な変化がいつそう激しくなるのではないかと予想されます。

国としても、これまで、コンピュータ関連技術の「高度化」ということを目標にいろいろな施策を行って来ましたが、これからは、何を前提にして、どこから出発するかを、もっと真剣に考えなければならぬと感じています。

ある種の技術の高度化は本当にすべきものなのかどうか、そうした高度化が行われた結果、個々の企業あるいは産業界全体がどんな影響を受けるのか、といったようなことをきめ細かく議論しないと、ただの抽象的な話で空回りしてしまい、意味のある成果が達成できません。

今後の技術開発の方向を考えると、一方には、いま述べたような汎用化・高度化の道がありますが、もう一方には、ユーザとの関係で技術を特化させて行く方向が考えられると思います。

これからのソフトウェア技術者は、ユーザを越えるような専門知識を持ち、ユーザに対していろいろアドバイスができるようにならなければ、とてもプロとしてやっていけないでしょう。ユーザ企業でも、かなりの人間を情報処理部門に展開していますし、それなりの経験を蓄積してきていますから、それを上回る知識をもってコンサルティングができるようなソフトウェア会社しか、これからの時代では、生き残っていけないのではないかと思います。

このように、技術の汎用化と特化という相互に矛盾した要求を同時に満足させてゆくのは、かなり難しいことですが、そういったことができるだけの多彩な人材

を確保できる企業とできない企業とでは、いずれ大幅に差がついてくると思います。いまは、業界全体が活況で、みんなが仕事の「量」だけに気を取られています。こうした「質」的な変化の兆候も少しずつ出てきているように思われます。

ソフトウェア産業のもうひとつの特徴は、まだ歴史が浅いということです。一番古い会社でも15～20年位で、だいたい創業者が経営していますが、これが2代目、3代目になって、経営の内容がどうなっていくか、興味深いところです。

しばらくは急激な変化は起こらないと思われませんが、これまでに話したような多くの問題があるわけですから、周囲の状況が明るいうちに、確固とした方針にもとづいて、それらの問題点を解決するための人材を確保し育成していく必要が大いにあるといえましょう。

そういった意味からすれば、業界の中で、経営上あるいは技術上の諸問題について、自由で活発な議論をする場を持つことが大切だと思います。ソフトウェア産業の未来を切り開いていくためには、かなり強力なリーダーシップが必要であり、なんでも「まあまあ」という平穩無事的な形で摩擦を解消していくのではなく、むしろ、ときには喧嘩腰の激しい議論の中から問題をどんどん発掘し、だれかがリーダーシップを持ってそれを解決するということの繰り返しが、しばらくの間は必要ではないかと思えます。そういった意味で、今日のような場が、もっと数多くあっていいのではないかと考えています。

鈴木: どうもありがとうございました。個々の発言内容に関しての質問や意見は、あとでまとめて時間をとりますので、とりあえず、他のパネリストの方々のお話をうかがうことにしたいと思います。それでは次に、経済社会のソフト化という観点から、金融あるいは流通業を中心に、ソフト化経済センターという団体を作って精力的な活動をしておられる岡田さんをお願いします。

3. 経済社会のソフト化

岡田: まず、私どものセンターがいったい何をしているかということから、お話を始めたいと思います。といっても、それほどはっきりしたコンセプトがあるわけではありません。経済問題に関する議論をしているうちに、何がなんだかよくわからなくなると、「それがソフトノミクスだ!」ということで、自動的にうちのセンターの出番がくるという仕掛になっています(笑)。

3.1 ソフト化センターとは

コンピュータのソフトウェアは、いわゆる「ソフト」一般の中では、比較的わかりやすいもののひとつだと思います。世の中には、それ以外にも、いろいろソフトなものがあり、そして、この「ソフト」というキーワードを中心に、経済の構造変化が進んでいることはたしかだといえましょう。それではソフトって一体何だろう、と考えはじめたのは、数年前のことです。

実は、私は、それ以前から日本長期信用銀行の中で、産業動向について調査をしていたのですが、どうもソフト化・情報化・サービス化という動きが、最近の経済社会に大きな変化をもたらしている真の原因ではないか、ということに気付いたわけです。

そこで、その実態を調べてみようと考えたのですが、そうした仕事は、当然ながら個人の能力をはるかにこえている。それなら何人かの人間が集まって研究したらいいというので、いまのセンターを設立し、約200社の会社を集まていただきました。経済のソフト化ということですから、ほとんどすべての産業が関係しており、サービス業、金融、流通、鉄鉱、石油、証券等、種々雑多な企業が参加しています。

センターができてから約2年たちましたが、先程申し上げたように、どこの企業もまだ、ソフト化・サービス化・情報化の実態を正確に把握することができていないし、これからどうしたらいいかもよくわかっていない、というのが、正直な姿だといっているでしょう。

3.2 物ばなれの意味

しかし、にもかかわらず、社会のソフト化はどんどん進行しつつあります。情報ですとか、サービスですとか、実際の「物」とはちがった、われわれの考えているソフトな「物」が日本の経済の中でどれくらい力を持っているのかについて、ちょっと数字をはじいてみました。

この10年間で、こうしたソフトな物の生産高は、とうに50%をこえて6割近い水準になりつつあります。同様に、ホンモノの「物」からはなれて、ソフトの関連で働いている人たちがどれくらいいるかといえば、これも全労働人口の約6割になっています。そういった意味では、ソフト関連の産業というのは、円高だとか、デフレだとかいった因子に左右されることなく、マクロに見て非常に成長している。人もそれだけ集まっているし、生産高も上がっています。

こういった全体的な「物ばなれ」の傾向のなかで、本

当に物を作っているほうは一体どうしてくれるんだ、ということになると思いますが、この問題は、もう少し高い視野から見れば、実際にはより大きなウネリのなかに、情報化とかサービス化という現象が組み込まれているのだということがわかります。

6年ほど前の数字ですが、日本の産業界において、利益をあげるためのコストの使い方（中間投入といいますが）のなかで、鉄からはじまって流通業にいたるとの業種でも、ソフト化率（ソフトに対する費用）が高ければ高いほど、利益が上がっているという事実があります。

これは何を意味しているかといえ、要するに、経済全体のソフト化が進んでいるときには、単純に「物」の生産高を上げるだけではなく、ソフトに対して適切な投資をおこなわないと、どの産業も食べていけないという結論が、はっきり見えてきたということです。

経済のソフト化というのは嘘やハッキリではありません。今日の社会では、ソフトにお金をかけないと、利益は上がらないのです。昭和40年ごろには、こうした現象は全くありませんでした。「物」に対して設備投資をしていた会社の利益率が高かったのです。いまからでも遅くはありません。ソフト化経済センターの会員になって、ソフトにお金を使うようにすれば、あなたの会社も、必ず利益が上昇するでしょう（笑い）。

3.3 技術が職人化することの危険

さて、今日は技術者の方々のお集まりだということで、こうした経済社会のソフト化の流れの中で、コンピュータ・ソフトウェア産業がどのように位置付けられ、どんな方向に進むべきかとか、期待されるエンジニア像とは何かについて、お話ししなければいけないのですが、これはたいへんむずかしい。

そこで、何の裏付けもなしに、思いつきで申し上げると、私のような門外漢の目から見ると、特にソフトウェアのプロフェッショナル・エンジニアが、悪い意味の職人芸に凝り固まりつつあるような気がして、これはよくないことのように感じられます。

現在、ソフトウェアがひとつの産業として自立しつつあることはたしかです。たとえば、この会場には、一種異様なひとつの雰囲気がありますね（笑い）。それは、ソフトウェア業界が業界としてまとまってきて、その仕事に携わっている方々が、それぞれの企業の規模や方向はたとえ違っていても、大きな目で見ればひとつのモードに統一されていることの証拠だと思います。

そこで最初に気づく点は、雰囲気はやはり少し特殊ですね。銀行にもコンピュータ部門があって、ランチタイムになるとプログラマもお腹がすくのか（笑い）、食堂にやってくるのですが、彼らが入ってくるとなんとなくわかるんですね。御他間にもれず、銀行のコンピュータ部門もやたら残業が多くて、労使関係から見ても困るのですが、その残業の理由が、どうも自分たちの趣味に走っているように思えて仕方がない。つまり、それは、かれらがある種の職人芸にとらわれているからだと思いません。

大体、日本人が新しい技術に接するときには、どうしても職人芸の迷路に陥ってしまって、そこからはず創造的な成果は出てこない傾向がある。それは、子供のころから新しいものを作り出すための教育を受けていないからだともいえます。むしろ、どちらかといえば、新しいものをやっちゃいけない、ふつうの人間の行動の枠をこえてはいけないという形に教育されています。

こうしたことは、われわれがハードの時代から受け継いだ遺産なのです。いまの日本が世界に冠たるハード産業国家であることはまちがいありませんが、その技術的な裏付けは、ひとことではいえない、空間的には精度をあげることであり、時間的には納期を絶対に厳守することではありません。たとえば、造船業の場合を考えてみても、最近韓国などに相当追い上げられてきていますが、それでもなお世界のシェアの半分を占めているのは、仕事の精度もすぐれているうえに、納期をいつもきちんと守ってきたからだと思えます。

いままではこれでよかったのですが、これからソフトの時代になってくると、何か新しいアイデアを考え出すことが要求されるようになる。それにはどうしたらいいかといえ、やっぱりふだんから新しいことを考えるクセがついていないと、他人のアイデアを盗むしかないわけです。

発想豊かな技術者は、決していわゆる技術バカではないし、職人でもない。意識していろいろなジャンルの人間たちとつきあって、そうした交流のなかから知的な刺激を受け、新しい情報を収集することを心掛けている。自分の周りにいる同種の人間とだけ付き合っていると、どうしても馴れ合いになり、知的には一種の自家中毒に終わってしまいます。

レジャー産業の人たちなんかを見ていると、帰りにちょっと一杯というのが結構多いのですが、この場合、飲

み仲間が会社の同僚や先輩後輩だったりすることがほとんどです。プログラマの場合も多分そうなんじゃないかな、というのが、私の推測です。それが、おそらく、技術の職人化の遠い原因になっている。

ちょっとたとえが悪いかも知れませんが、私の知っている銀行のプログラマたちや、コンピュータ・メーカーの人たちを見ていると、精神的に一種の「タコ壺」のなかに入っていて、ときどき首を出してはあたりを眺めているだけのように見えます。そうした状況からいかにして脱却するかが、ソフトの時代を生きぬくためには必要なのではないでしょうか。

鈴木: どうもありがとうございます。それでは、次に東京大学の広松先生をお願いします。広松先生は、もともと計量経済学がご専門ですが、情報メディアにも関心をお持ちだということで、そうした観点からお話しただけだと思えます。

4. 市場の現状と国際化

広松: メディアに関する話題は後でご質問が出るまで一応保留しておいて、とりあえず、私の専門である経済学の立場からお話させていただきます。

手元にある資料の中から、ソフトウェア産業に関するものを探して見ると、今日のパネルのテーマに関係のありそうなアンケート調査が2つほど見つかりました。1つは、日本経済調査会が東証の一部に上場している会社を対象にして調査した結果です。みなさんの立場から見れば、ユーザ・アンケートということになりますでしょうか。もう1つは、これと性格がまったく違うもので、電気労連が加盟企業の組合員に対して行ったソフト関係労働者の実態調査です。この2つの調査を比較して見ると、おもしろい問題が指摘できるのではないかと思います。

4.1 ユーザ・ニーズの動向

日本経済調査会の調査は、かならずしもソフトウェア産業に焦点を合わせたものでなく、もっと広く、わが国の企業が全体として、産業の情報化あるいは情報の産業化に対して、現在どういう対処をしているか、将来どういう方向に進もうとしているかということ、明らかにしようとしたものです。

この調査の中に、アプリケーション・ソフトに関する項目があります。いろいろなクロス集計を行っていますが、その中のひとつに、現在それぞれの企業で、必要な

アプリケーション・ソフトを内部製作しているか、それとも外注しているか、また、将来それがどうなるかという質問項目があります。

ここには、おもしろい数字がでていますが、いくつかの業種別に分かれてはいますが、現在の段階では、どの業界でも内製の比率がきわめて高い。極端な例を上げると建設業では90%です。全体でも平均70%がアプリケーションを内製していて、残りが外注ということになります。

ところが、現状の問題点というところで指摘されているいくつかの要因の中から主要なものをピックアップして見ると、開発コストが高い、要員が足りない、開発能力が低いといったことが指摘されています。つまり内製をしてはいるものの、それぞれの社内ではこうした問題を抱えているわけです。

もう1つ、外注しようとしても、あるいは既存のパッケージ類を利用しようとしても、提供されていないとか、提供の料金が高過ぎるという、現状のソフトウェア・マーケットの条件に関する問題の指摘がありまして、この2つをあわせると80%以上の数字になっています。

こうした意味では、現在多くのコンピュータ・ユーザがそれぞれのアプリケーション・ソフトを内部で製作しているのは、どちらかといえば、すすんでやっているわけではなく、「しかたなく」やっている状況だといっているのではないかと。

その1つの証拠は、では将来どうなるかという質問の答えを見るとわかります。この将来という言葉はあいまいですが、まあ5年ぐらい先のことだと思いますが、その程度の将来には、現在80%をこえているアプリケーション・ソフトの内製が50%くらいに落ちます。つまり、約5割の企業が、将来ソフトの開発を外注したい、するつもりで考えているということになります。

こうしたことからすれば、ソフトウェア産業の市場に関する見通しは、非常に明るいということになります。少なくとも、東証一部上場企業の半分が、5年後には、アプリケーションの開発を外注しようと計画しているわけですから。

4.2 ソフトウェア労働の現実

以上は、ソフトの需要についての数字ですが、一方、反対に、ソフトを作っている人々の現状あるいは仕事の環境条件に対する感覚はどうでしょうか。

電気労連のアンケートによりますと、これは対象が組

織加盟の組合員だということで、かなり限られた調査だとは思いますが、かなり深刻な問題がいくつか指摘されていて、私としてはもっと威勢のいい、夢のある答えを期待していたのですが、結果は、どうやら逆だったようです。

労働条件、特に残業時間等に対する不満が強く出ているのと同時に、企業内における自分たちの将来に対する不安が、きわめて特徴的だと感じました。これは、逆にみなさんにお聞きしたいのですが、いわゆる「プログラマ30歳定年説」、30代や40代になってソフトウェア技術者としてやっていくことがかなりむずかしくなるという不安が、技術者自身の内部にかなりあるのではないかと、という印象を受けました。

もうひとつ大きな問題としては、これは近藤課長のほうからご指摘がありましたように、仕事の形態としての派遣のありかたがあります。派遣されるほうも派遣を受け入れるほうも、そう簡単にはいかないという問題が、この調査では指摘されています。

私自身、それほどソフトウェア産業に深くかかわってはいませんが、とりあえずこの2つの資料にもとづいて考えると、すでに他の方々からも指摘されたように、成長性ということではきわめて有望な市場ではないかと思えます。一方で、そこで働く技術者の条件というのが、そう簡単なものではないということを、あらためて実感させられました。

これは、日本国内だけの問題ではないようでして、OECDという国際協力機関の中で、情報・コンピュータ・通信関係の委員会が、1985年にソフトウェア産業に関するレポートを出しています。そこでサマリーとしてあがっていることをみますと、先程の調査にかなり似通っています。

つまり、現状認識として、ソフトウェア産業はまだ形成段階にあるということがいわれています。ソフトウェア生産の経済的地位や一国の経済の中での位置付けは確立されたわけではない。さきほど岡田さんがいわれたように、たしかに、現象としての経済のソフト化は進んでいるにかかわらず、産業としての位置付けは、いまだにはっきりしていません。さらに、ソフトウェアのマーケットもまだ成熟しておらず、未分化の段階である。以上が大まかな現状認識です。

4.3 国際化と競争

こうした現状認識にもとづいて、OECDレポートは、

いくつかの提言をしていますが、そのうち重要なものを列挙すると、第1にソフトウェアに関する研究開発ポリシーを明確にすべきこと、第2にソフト技術者の社会的地位向上を考えるべきこと、第3に各種の標準化をなすべきこと、第4にソフトウェアの法制的な面の充実をはかるべきこと、そして、第5にソフトウェアの国際的な貿易の推進があげられています。

このレポートは2年ほどまえに出ているわけですが、OECDに参加している先進20ヶ国の現状を素直に反映しているものではないかと思えます。

さて、最後に、国際的な技術競争力についてちょっと触れてみたいのですが、素人としての私の独断をいわせていただくと、日本のソフトウェア技術は、アプリケーション・ソフトは別として、OSやデータベースなどシステム・ソフトの競争力が弱いように感じられます。そこで、そうした分野に強いアメリカが、自由競争の原則を振りかざし、ソフトウェア市場の自由化を迫ってきているのが、現状ではないかと思えます。

いままでに他の産業に関しても、みなさんが御存じのように、いろいろな要求がされてきましたが、ソフトに関しても、これから同じようなことが起こると予想される。産業としてまだ形成期であるソフト産業が、そうした事態にどう対処していくかが、非常に重要な問題ではないかと思われます。これは政府の政策にも関係するわけですが、ハードのように国産重点でいくのか、あるいは自由化の方向をとるのかを、よく考えて決めなければならない。

そもそもソフトウェアという商品は、こうした国際レベルでの流通性を持つのか。アメリカ製のソフトがどこまで日本で使えるかをハードと同じレベルで論じていいのかが、まず問題になると思いますが、私自身は、日本は日本語を使っているからというだけで、簡単にすむことではないと考えています。こうしたことに関して、みなさんがどういうご意見をお持ちなのか、あとでお聞かせいただければ幸いです。

鈴木： どうもありがとうございました。次に、インダストリアル・デザイナーの佐野さんに御登場いただきます。ソフトウェアと同じようにデザインというのは設備投資がいない商売でして、そんなわけで早くから独立されて(笑い)、日本インダストリアル・デザイナー協会でも活躍されています。そうした経験にもとづいて、SEAへのアドバイスをいただければと思います。

5. デザイン産業の経験から

佐野： いま鈴木さんがいわれたように、私どもの業界はソフトウェア業よりいっくらか長い歴史を持っていますから、いいこと悪いこと両方を含めて経験をお話します。他山の石としてお聞きください。

インダストリアル・デザインという職業が日本に入ってきたのは、戦後の1950年頃のことです。そして、1952年に日本インダストリアル・デザイン協会が設立されました。その設立パーティーのときに、千葉大の小池先生という方が、インダストリアル・デザインの本質は工業を人間化することだという趣旨のご挨拶をされています。その後30何年たつて、デザインの潮流はいろいろ変わってきたわけですが、結局われわれの仕事の本質は、この小池先生のことばにまた戻ってしまうんじゃないかと、感じています。今日は、産業としての観点から議論するということですので、われわれの商売の実態を手短にお話します。

5.1 業としてのデザイン

まず、日本にインダストリアル・デザイナーが何人位いるかですが、大体1万5千から2万人位だと思います。そのなかで、エンプロイー・デザイナーつまり企業に雇われている人間が過半数です。これは、日本のデザインの質に関する大きな特徴だといえます。外国では、もちろんそんなに多くありません。

次に、私のところのような独立のデザイン事務所がどれくらいあるかというデータを調べてみると、私の個人的な予測を上回って、全体で260社ほどあるようです。それらの事務所の年間売上総額は、少ないのでびっくりされると思いますが、59年度で42億ちょっとです。

一方、企業内のデザイナーがどれだけ稼いでいるかはほとんどわからないのですが、コストに換算すれば、1万人かける400万として約400億になります。ビール業界の年間売上げ総額の数字をどこかで見たことがあるのですが、ちょうどビールの泡の分くらいが、デザイナーの稼ぎだということでしょうか(笑い)。

さて、デザイナーの平均年齢はというと、先程プログラム30歳定年説という話がありましたが、私たちの協会のメンバの平均年齢は47歳です。たまたま私自身はその平均値ぴったりなのですが、全国平均だともう少し若く、35~6才位だと思います。

教育機関は、大学が29、専門学校が7つあります。毎年7~800人の卒業生が出ていますが、やはり大会

社に入りたいという人が多いようです。教育の内容ですが、本来だと芸術的な内容の基礎をしっかりとやらないといけないのですが、産学共同に近い実学的な内容がほとんどで、これは将来何らかの問題を生むのではないかと思います。

5.2 ライフ・スタイルの変化

先程、プログラマの集団には特定の雰囲気があるというお話がありましたが、デザイナーの場合も同じでして、企業内に取り込むと、他の人間とちょっと人種が違うような感じで扱いにくいという面がある。そのせいかもしれませんが、デザイン・センタを会社本体から切り離して独立させるという動きもあるようです。

契約についてですが、以前は、きちんとした契約なしに仕事に入ってしまうことがよくあったのですが、最近では過半数の人が、契約を結んでから仕事をするというスタイルが変わってきています。契約をしない限り線一本引かないという人もいて、だんだん日本も契約社会になりつつあるようです。

ところでいったい、デザイン料というものが、どうやっていくらか支払われるのかは、みなさんも関心をお持ちだと思いますが、通常は、デザイン・プロセスを20いくつかの項目に分けて、それぞれに時間単価を掛けて計算されます。しかし、これでは新しいアイデアを生み出すための考案料が抜けていて、ただの労働報酬じゃないか、できればロイヤルティー方式にもっていききたいということは、以前から話題になっていまして、そういうケースも増えてきてはいるようです。

仕事の評価についても、これまではずっと、イタリアだと北欧だとからの情報を受信するだけだったのですが、ちかごろは日本もデザイン情報の1つの発信地に変わりつつあります。

デザインという職業の未来に関する話題を探すと、企業の経営者に対するアンケートで、製品におけるデザインの役割が増すか否かという質問の回答が、いっそう重要になる89.9%、重要になる8%、現状と大差ない2%という結果でした。経営者の目から見て、インダストリアル・デザインの重要性は、今後ますます増加するということのようです。これも、岡田さんのいわれる社会のソフト化のあらわれでしょうか。

それから、デザイン事務所の経営者への質問で、将来必要とされる能力は何かという問いに対する回答で、一番多かったのは企画・計画力で70%、国際感覚が23.

4%, コーディネート能力が22.5%でした。そして、将来あまり必要でないという項目のうち、デザイン感覚がなんと17.3%にも達している(笑い)、これはちょっと考えなくては、と感じています。ソフトウェアの方ではどうなのでしょう。

5.3 方法論上の課題

1950年代には、われわれ工業デザイナーの仕事というと、大量生産される商品の原型(プロトタイプ)を作ることが主で、ほとんどが個人的な作業でしたが、今は仕事の内容もやり方もかなり変わってきて、企業内ではほとんどグループ・ワークになりました。きわめて日本的な創造的集団あるいは集団的創造というかたちができあがっています。

日本では、ふつう、工業製品を誰がデザインしたかということは問題にされません。たとえばここにマイクがありますが、デザイナーの名前はどこにも書いてありません。こうしたデザインの無名性は、日本という社会のひとつの特徴だといえます。

昔は、こういう形を作ってくれという依頼を受けて仕事が始まったのですが、今はそれ以前の段階で、どんな商品を作ったらいいかという相談から始まる仕事が多く、ここ10年くらいのあいだに、急に増える傾向にあります。デザイナーの守備範囲も、商品企画から集まって、設計、生産、流通、ユーザと、全般に広がってきています。つまり、単に物をつくるということではなく、その物を使う環境を含めて統一的なストーリーを作って、それにもとづいて物作りを始めるわけです。

デザイン・プロセスは、大体において、リニアに計画から具体化へと進んでいくのですが、これはソフトの場合もそうでしょうけれど、分析またはコンセプト作りのフェイズがきわめて重要な意味を持っていて、われわれはその部分をデザイン・ソフトと呼んでいます。一方、手を動かす仕事はデザイン・ハードと呼ばれます。以前は、前者をプレーン・ワーク、後者をハンド・ワークという呼び方をしていました。

このように、どんな形を作るかの前に、何を作るかを考えなければいけないとすれば、そうした分類に従って、専門化も進んできています。コンセプトの確立を主体に、消費者の傾向まで含めた全体的視点を持ったジェネラリストと、形を専門にする造形スペシャリストに分かれる傾向があり、教育にさいしても、両者を区別しなければ、ということになってきています。

もう1つの問題点としては、デザインという感覚を取り扱うというための方法論を、われわれは過去何十年かかけて体系化してきたわけですが、その方法論自体に問題があるのではないかという問題意識が、最近取り上げられつつあります。

デザインは、ことばなどに比べると、目にみえるという点が特徴で、わかりやすいという性格が非常に強い。ですから、アイデアを目にみえるものに変換して、それをプレゼンテーションするという、一種のシュミレーションみたいなことを何回もやるわけです。いまでは、当然コンピュータを導入してやっています。こうして、シュミレーションの回数を増やしてゆくと、仕事のやり方が固定化されてしまう。

それが日本のデザインの質と非常に係わりがあるのですが、ネームプレートをとってしまうと、どこのだれのデザインかわからないという結果になっている。これは、視覚化ということだけを重要視した方法論の固定化に原因がある。したがって、そういった袋小路を抜け出すには、もっと方法論を多様化していかなければならない。

そのことにはもう、多くの人が気付いているのですが、今までの方法論が長い時間をかけてやっと作りあげてきたものだから、それを自分たちで壊す勇気がなかなか持てないというのが現状のようです。

鈴木: どうもありがとうございました。いろいろ示唆に富んだアナロジーだったと思います。それでは、最後に、ジャーナリストという第三者の立場から、朝日新聞科学部の大塚さんをお願いします。

6. ジャーナリストの目から

大塚: 新聞社の科学部というところは、ソフトウェア産業にそれほど関係の深い部所ではありません。そのなかでも、私は医学関係の分野を担当していて、最近では男女生み分け技術についての記事を手がけたりしています。ですから、まったくの第三者という感じで、何で自分がここに座っているのか疑問なのですけども、実は、新聞記者になる前に、一度みなさんと同じ商売を経験したことがあるのです。

最初は、事務機のメーカーで、小型のオフコンのSEをやっていました。その後、2年ほど、半分アルバイトみたいなかたちで、あるソフトハウスにお世話になり、IBMのユーザに3年ぐらいいて、今の商売に代わったという経歴です。ですから、ソフトの話というと、自分自

身の10年前のつたない経験を思い出しながらお話することになりますので、そのへんはご勘弁ください。

6.1 技術と産業

私自身がいたころのソフトウェア商売は、非常に労働集約的な産業で、なんでこんなに人がたくさん必要なのかという感じが、いつもしていました。しかも、それらの人たちがみんな忙しく働いている。私もずいぶん徹夜をしました。そのうちに、そういう生活からドロップ・アウトしてしまったわけですが、当時のコンピュータ産業には勢いがあったと思いますし、最近の勢いはそれにもましてすごいという感じがします。

私のいる新聞社は築地の癌センターと魚市場にはさまれたすばらしい環境に位置しているのですが(笑い)、その8階の食堂から海のほうを見ると、大きな23階建ての赤いビルが建ったんですね。そのビルを日本IBMが全館借り切って、ソフト開発部隊が入っているということです。知り合いに聞いてみたら、IBMというのは非常にセキュリティが厳しくて、ああいふ独立ビルを借り切らないと秘密が守れないのだそうです。ちょうどわが社のビルを見下ろすように立っていて、同じ情報産業でも斜陽になった新聞業界と、日の出の勢いのソフトウェア業との関係を象徴しているような気がします(笑い)。

ところで、ソフトウェア産業ということばの定義が、私にはどうもよくわかりません。今日ここには、主にソフトハウスの方々が集まっておられるのですが、プログラマやシステム・エンジニアは、ユーザにも、メーカーにも、また大学や研究所にもたくさんいるわけで、総数は多分何十万人というオーダーになるでしょう。で、われわれ一般人がソフトウェア産業ということばを漠然と聞いたときに、正直いってイメージがわかないわけです。一時そこに身を置いた私でも、ぴんと来ないところがありますね。

そこで、わが社の記事データベースのキーワード検索を使って、ちょっと調べてみました。ただし、対象の記事は過去2年分しか入っていません。

「ソフトウェア産業」という単語でひくと、出てきた記事は5件でした。これが多いか少ないかですが、比較のために「自動車産業」で検索すると、144件です。次に、「ソフトウェア技術者」でひいてみると、16件の記事が見つかりました。つまり、コンピュータ関係の専門紙ではどうだか知りませんが、一般には、「ソフト

ウェア産業」ということばよりは、「ソフトウェア技術者」のほうなじみが深いのではないかと思います。

6.2 一般的イメージ

さて、具体的にどんな内容の記事がのっているかですが、たとえば、文部省がソフトウェア技術者の養成のために、情報工学関係の定員を増やすという記事がありました。それから、都心のビルがどんどんソフトウェア部隊に占拠されているというのもありますね。さらに、ソフトウェア技術者の人集めが難しいとも書いてあります。それから、これは1昨年の暮れに出た記事ですが、「急増するソフト需要、生産自動化へ動く」という見出しで、通産省がやっておられる例のシグマ・プロジェクトの紹介があるわけですね。

こういった記事を通じて、いわば新聞を1つのフィルターとして、一般の人々のソフトウェア産業に対するイメージが形成されることになるわけです。つまり、産業規模は拡大しつつあり、人手が足りない状況だけれども、なかなか労働条件がきびしらしく、人が集めにくい、といったところでしょうか。

これは、別に新聞が特に悪意をもって、世論操作をしているというのではなく(笑い)、一般的にそういう産業動向があって、時として、それが強められたり弱められたり、ということだと思います。シグマの前提条件として、通産省が出したソフトウェア技術者の需給ギャップの予測がありますが、私が10年前にプログラマだったころにも、まったく同じようなグラフを見た記憶しています。それが変わっていないということですね。

これからは、そうした産業構造の成り立ちとその原因または改善の方向について、技術者自身の口から、外部の人々への説明を積極的に(もちろんマスコミも利用して)行うべきだと思います。そのことは、必ずよい結果をもたらすでしょう。

6.3 技術者に期待すること

最後に、産業ではなく、ソフトウェア技術者という存在を、世間一般がどう見ているかということについて、私個人の意見を述べさせていただきます。

私は、科学部に来る前に科学朝日という雑誌にいました。コンピュータ・サイエンス関係の取材でアメリカに行ったことがあります。いま、人工知能とくにエキスパート・システムがもてはやされていますが、私自身が気になっているのは、AIとソフトウェア技術の関係が一体どうなるのだろうか、ということです。

アメリカで、AIの権威者の何人かと会う機会がありました。たとえば、MITのミンスキー教授ですが、かれは、応用技術よりも基礎研究の方に関心があって、人間の知能をいかにしてコンピュータに移植するかという研究をやっています。そのミンスキー先生に、AIはいまどれくらいのレベルにあるのかと質問したら、物理学の世界でいえば、ガリレオを経てニュートンが出てくる前の状態だそうです。ニュートンからアインシュタインまで、あと200年ある(笑)。

また、ノーベル賞をもらったカーネギーのサイモン教授。いま、根本的な現象を入れて、そこからいろいろな法則を演えきするようなプログラムを作っています。聞いてびっくりしたんですが、ケプラーの第3の法則を発見したというんですね。これは御存じのように、惑星の太陽から平均距離と公転周期が一定の比になっているという、古典天文学の法則なんですが、ケプラーの観測結果だけ入れてやって、プログラムが自動的にそういう法則を発見したということなんです。

それから、人工知能ということばを発明したスタンフォードのマッカーシー教授。この人の話は非常にむずかしくてよくわからなかったのですが、別れぎわに、いまどうしているのかと聞いたら、飛行機の座席予約プログラムをいかにうまく保守するか、考えているというんですね。たとえば、同じ飛行機で、イラン人とイラク人を隣り合わせに座らせないようにするには、どうしたらいいか(笑)。こういったことが必要になったら、すぐにプログラムが変更できる。まあ、一種の自動プログラミングだと思えます。

それが可能であるかどうかは別にして、そういったことをかなり掘り下げて、根源的なところからさかのぼって考えている人たちがいるということが、きわめて印象的でした。

もちろん、日本にもそういう方はおられるでしょうが、さきほど広松先生がおっしゃったように、日本でソフトウェア技術というと、どうしても応用的な側面が強いと私も思うのです。ソフトウェア・クライシスが10数年前からずっと続いていて、それに対して文部省が大学の定員を増やしたり、通産省が情報処理技術者試験の回数を増やしたりするというのは、それはそれで結構とは思いますが、技術者の方々には、もう少し別の角度からものを考えていただきたいというのが、私の願いです。

ソフトウェアに関する需給ギャップが今後も依然とし

て開き続けるとすれば、逆に将来、どこからか突然黒船(あるいはUFO)があらわれる可能性があります。それは当分先のことでしょうし、現実性は薄いかも知れませんが、ソフトウェア産業の未来を支えていくみなさんには、どこか心のすみに止めておいて欲しいのです。

アメリカやヨーロッパでは、ソフトウェアの研究者や技術者たちが、電子ネットワーク上でいろいろなコミュニケーションを行なう環境が整っています。そういった社会機構は侮りがたいパワーを秘めているんじゃないでしょうか。ソフトウェアが、いま、産業としてはハッピーだとかがっているだけに、その間に、できればそういった社会的なメカニズムの確立までを含めて、お考えいただければと思います。

7. 質疑応答

鈴木: さて、以上5人の方々にお話をうかがったのですが、司会をしている私自身、ソフトウェア産業論を過去こういう形で聞いたことがありませんし、経験もないので、焦点を絞ることはむずかしいのですが、成長率だけは周囲から見ているとすごい、しかし、先行きの中身に関しては、いろいろ問題が出てきていて、これから大きな変化があるだろうというのが、みなさんの御意見であったようです。

それでは、会場のほうから御質問をいただいて、個々の問題を少し掘り下げてみたいと思います。

7.1 他業種からの進出

Q: 最近、いろいろな企業の情報産業への進出が目につきます。たとえば、不況業種といわれる企業が、その情報処理部門を別会社にしたりしています。パネリストの方々は、こういう現象について、どのように感じておられるのでしょうか、また、どんな期待をお持ちでしょうか。

鈴木: では、近藤さん。

近藤: おっしゃる通り、各産業の情報化という現象が、たいへんな勢いで進んでいると思います。構造不況といわれている業界で、そうした動きがかなり活発であることも事実です。

ここでの1つの問題点は、いろいろな産業がそれ自身の情報化を進めてゆくときに、そのために必要な人材を採用でカバーできるかといえば、決してそうではないということです。

岡田さんがさきほど指摘されたように、コストにソフトの占める率がたいへん高まっていますから、それをいかに効率的に、かついいものを作っていくかということが必要なわけで、各産業の情報化は、必然の成り行きだと思いますが、それを支える人材面で、たいへんなさん苦勞されている。これをなんとかしたい、なんとかならないかというのが、企業からの強い希望ですし、行政的にも何かの手を打つ必要があると考えています。

一方、各産業の情報化や情報産業への進出を、ソフトウェア市場への新たな競争の導入と考えてみると、これは、市場の活性化をもたらすという意味では、たいへん歓迎すべきことであり、結果として、技術力の高い企業を育てるようなインパクトをもたらすものだと思います。ただ、ソフトウェア産業自体がまだ弱い体質であることは、先程から多くの方が指摘されておられますので、そうした競争の激化にともなって、いろいろな問題が出てくるでしょう。

総体的には、ソフトウェアに対する需要が供給を上回っている現状を考えれば、質問者が指摘されたような方向に進むのは、きわめて自然ではなからうかと思えます。

岡田： 私も、近藤さんとほぼ同じような意見です。私どもの考え方は、産業というのをもう少し幅広くとらえようということです。かならずしもソフトウェアに限らず、多角化戦略ということで、多くの企業がいろいろな分野に手を出しているのが現状だと思います。自分の専門分野だけで食べていこうという会社は、非常に少ないですね。

特に不況だからとか、この業種が若いからということだけでなく、一般的になるべく商売のチャンネルを広げていこうという戦略がある種の流行になっているようです。それは、かつての高度成長期によく見られた、何でもいから手を広げようという姿勢とは、少し様相を異にしています。

1つの大きな違いは、業種的に見て、コンピュータ・ソフトウェアも含めてソフト的な仕事への進出が目立つことですね。たとえば、レストランをやったり、学習塾を手がけてみたり、何かもうかりそうな仕事には必ず手を出してみようということだと思いますが、つまり、一般に、そういった「ソフト」な産業は、もうかると思われているわけです。

ふつう、そうしたニュー・ビジネスの可能性は、新規事業開発室とか企画室とかで検討されるのですが、その

さいの候補のなかに、この業界つまりソフトウェア・ハウスが入っている、ということになります。

他業種からソフトウェア業への進出には、2通りのケースが考えられます。

1つは、収益を目標にする場合です。新しいお金もうけの口はないかということで、社内のソフトウェア部門を分離したり、新しい会社を作ったりしているわけです。

もう1つは、労務対策です。コンピュータ関係の仕事は、多くの場合、もともと会社本来の業務とは性質が異なり、勤務形態も大分ちがいますので、それを同じ事業体の中で管理するのは、かなりむずかしい。銀行の例でいいますと、以前はコンピュータ部門は本店のなかにあった。それが、だんだん人が増えてくると、ちょっと離れた場所にコンピュータ専門のオフィスを作る。こうした動きがもう一步進むと、分離・独立させて別の法人にするというかたちになります。

私は、ソフトウェア産業は、成熟してはいないまでも、十分青年期の産業に育ってきていると思えますが、そういう時期における特徴は、有能な人材が全体として不足することです。教育機関はあまりあてになりませんから（笑い）、どうしても内部で人間を養成していかなければならない。人数をそろえるために専門外の人間でも採用するわけですが、上に立つ人間がたりない。

そこへ新規参入の企業が増えれば、リーダー・クラスの人間を対象としたスカウトが激しくなる。どこの企業も、新しい利益を出そうと一生懸命なわけですから、そのために、いろいろなことをやる。資本の論理がここにも働いていると思えます。

鈴木： それでは、この辺で次の質問に移りましょう。

7.2 外国との比較・その他

Q： 今日のテーマ「ソフトウェア産業の未来」については、私なりに考えているのですが、どちらかといえば、かなり悲観的です。

その理由は、情報化の未来という視点で、先行している欧米を観察すると、情報化の進展によって社会が動いていくスピードの差を感じるからです。単に、技術のレベルがちがうということじゃなくて、ある一定の期間（5年でも10年でもいいのですが）を切ってみたとき、日本の社会はそんなに変わっていないのに、外国では急速に変わっているということがあがると思うのです。

たとえば、さっき大塚さんがいわれたネットワーク・

コミュニティにしても、日本ではまだ電話線や光ファイバーがつながっているだけの状態ですが、欧米ではすでに、たくさんの技術者が日常の情報交換に利用していて、そのことが社会に対して一種のインパクトを与えているという事実がある。そういった、個々の技術の比較でなく、むしろ技術の社会的インパクトという面で、なんとなく日本は遅れをとっているような気がするのですが、このへんについて、パネリストの方々の御意見をお聞かせください。

鈴木： 国際的な事情に詳しい広松先生、いかがでしょうか。

広松： たしかに、現象として、いま例にあげられたネットワークの問題をはじめとして、日本と欧米ではかなりの差があるように思います。

それは、広くいえば、言語のちがいが、文化のちがいが等にまでさかのぼることになると思いますが、そこまで手を広げないことにして、私なりに考えている大きな原因は、日本における情報化が、まず産業レベルから入ってきて、家庭の中にはまだ入りきらないでいる状況にあるからだと思います。

家庭に入るとということに関していえば、これは極端ないいかたかもしませんが、新しい技術が入りやすいのは、オモチャからではないかと思えます。その意味で、いま話題になっているファミコンなんか、ひとつの足がかりになるのではないのでしょうか。

それと、さきほどの質問でちょっと気になっていることがあるので、少しお話してみたいのですが、いま私がいったような意味で、日本のソフトウェア会社の場合、まだ業種別のしがらみがあって、たしかに情報処理部門を別会社として分離・独立させようとしたとき、もとの業界内での知識の蓄積はありますが、新会社が純粋にソフトウェア・ハウスとして生きていくためには、しがらみをもったままいくのか、それとも積極的に自立の方向をめざすのか、1つの分かれ道になるのではないかと思えます。

ちょうど10年くらい前に、シンクタンク・ブームという現象がありました。いろいろな企業がそれぞれの企画部門を分離させて、シンクタンクをめざしたのですが、結局、日本ではうまく根づかなかった。そのことの大きな原因は、やはり企業内の一部所としての役割と、独立した会社としての機能が、根本的に違うからではないか、と思えます。

鈴木： よろしいでしょうか。では、次の御質問をどうぞ。

Q： 今の質問に関連して佐野さんにお聞きしたいのですが、インダストリアル・デザインの分野で日本と欧米とではどう違うのかを、参考までにお聞かせください。たしか、佐野さん御自身も企業内デザイナーから出発して独立されたとお聞きしているのですが、外国の場合はどうなのでしょう。

佐野： 日本の特色として、企業内デザイナーが非常に多く、それが日本のデザインの質を左右しているということはさきほど申し上げたのですが、わかりやすくいってしまえば、企業に勤めているデザイナーは、デザイナーである前に企業人だということです。これは、企業の中で教育されるからということもありますし、そういうふうに自分から思っている人が多いですね（笑い）。ですから、個人的な気持ちなどは、なかなか表に出さない。

そのことが、欧米のデザイナーからみると驚きのようです。日本では、マーケティング・サーベイにもとづいて、そのデータをもとにデザインをしたりするんですが、イタリア人あたりにいわせると、とても信じられないといえますね。かれらの場合、まず自分のデザイン・アイデアがあって、それを企業に売り込むというかたちで仕事をしている。ところが、日本のデザインは企業主導型で、自分たちの方法とまったくちがう。にもかかわらず、これだけ質が高くなってきたのは、どういうわけだろう。不思議としかいいようがない（笑い）。そういう基本的な違いがありますね。

それでは、日本のフリーの人たちが、いったい何をやっているのかといえば、私を含めて、ほとんど企業の中にいる人と変わらないことしかやっていないんじゃないかと思えますね（笑い）。そのへんの基本的な意識を変える方向で、もっとがんばらなくては、と考えています。

同じ「物づくり」の世界でも、日本では変に許されるところがあるわけです。私の個人的な感想ですが、日本ほど企業とユーザが仲良くやっているところはないんじゃないでしょうか。一種の癒着といってもいい。ですから、あまり創造性が高いデザインは、拒絶こそされませんが、認められないという状況が、これまでずっと続いている。そういう中で、物のかたちに関しても、これはこうあるべきだという頑固な固定観念が、ここ何十年かかってできあがっていて、単なるバリエーションなら許されるが、しかし基本的な変更は拒否されてしまう。さ

きほど、物にかかわるパラダイムの見直しが必要だといったのは、そういった意味です。

例えば、わかりやすい例をあげると、35ミリカメラがあります。あれは今ゴツゴツした感じですね。機械工学の粋みたいな製品の概念的イメージがあって、それがそのまま外観を形づくっているわけですが、そこをいじるのは非常にむずかしい。とくに、産業界で実際に物を作っている人たちが、一番こだわりますね。デザインが外部から、もうちょっと人間工学的にフィットするものを作たらどうかと提案しても、最後はやはり、あのゴツゴツがいいんだという結論になってしまいます(笑)。そういったことにチャレンジしていくのが、これからのデザインの課題じゃないかと思うわけです。

Q: 何かの本で読んだのですが、MEのひとつにCTスキャンという装置がある。いまでは、ちょっとした病院なら入っているようです。それで、この種の機械の開発プロセスですが、アメリカの場合は、メーカーが勝手にそれぞれのアイデアで作っていて、そのあいだ医者はあまりかかわらない。で、完成した機械を医者を使って、もし何か問題が起きたとすれば、その責任は機械を作ったメーカーがとるようになっている。

一方、日本では、技術的には十分そうした機械を開発できる状況であっても、最後にどっちが責任をとるかという問題に関して、企業サイドでは責任がとれない、というより、とりたくない。そんなことでもたもたしているうちに、結局、CTスキャンの開発では、アメリカに遅れをとってしまった。そういった事実があるそうです。

こうした問題について、御意見をお聞きしたいのですが。

鈴木: これは、大塚さんをお願いします。

大塚: 非常にむずかしい御質問で、うまく答えられる自信がないんですが、私なりの考えを述べさせていただきます。

医療分野へのエレクトロニクス機器の進出は、最近めざましいものがありまして、国内のメーカーも競って開発を進めていますし、いまあげられたCTスキャンの例でいえば、すでに日本のメーカーのほうがいい製品を開発しているというのが、現在の状況だと思います。

こういった新しい医療技術とか、医療機器などに対する受け止め方は、国によってずいぶん差がありまして、私は国内の状況しか知らないのですが、メーカーの方からお話をうかがうと、アメリカなどでは、新しい機器を持

っていくと、お医者さんが積極的に受け入れてくれる。しかし、日本では、お医者さんたちは一步高くかまえていまして、我々なんか取材にいても、なかなかお話を聞かせてもらえないのですが、メーカー側からの接触もなかなかうまくいかないそうです。

そんなわけで、せっかくいい製品が開発されても、実際に医療の現場で生かされるのは、欧米のほうが先になるということが多いようです。

これには、文化的な背景もありますし、あるいは法律的な制約など、社会制度のちがいもありますから、アメリカの場合といちがいに比較することはできないと思いますが、それぞれの業種に特有の社会的な問題については、ソフトウェアをおやりになるかたも、これから十分に注意していかなければいけないと思います。

いま、若手のお医者さんで診断用のエキスパート・システムに関心を持っている人は多いようですし、実験的なシステムを作っている人もいます。そうした人たちの協力によって、うまくすれば、向こう側の慣習を変えることも、できるかも知れません。

鈴木: 時間もなくなってきましたので、そろそろ最後ということにしたいのですが、他に御質問は。

7.3 技術文化論

Q: さきほど、岡田さんが「タコ壺」というおもしろい表現を使われましたが、われわれの会社も、まさに「タコ壺文化花ざかり」といえるくらいの状況にあります。

そこで何がいけないかといいますと、1つには、生産性があがらない。お互いにそれぞれの小宇宙の中で、似たようなものを重複して作っているということがあります。もう1つは、広松先生があげられた調査で、残業の問題や将来の不安ということがありましたが、そのことも、実はタコ壺と関係しているのではないかと思います。

佐野さんのお話をうかがうと、デザイナーの世界でも同じような状況があるようなのですが、どうしてタコ壺になってしまうのか。私ども管理者側は、とくにタコ壺の好きな人を採用しているつもりはありません(笑)、その辺について、御意見をお聞かせ願えればと思います。

岡田: なぜかといわれても、私は自分がタコではないので、よくわかりませんが(笑)、私自身の経験にもとづいて、現象論的に分析してみましよう。

私どものソフト化経済センターは、参加各社からの出向の方がたくさんいるわけです。で、最初に何が起きた

かといいますと、何しろ「ソフト」な企業が多いものですから、われわれ銀行屋と服装が違うんですね。ネクタイをしていなかったりとか、上下の色が違う洋服を着ていたりとかで、まったく色とりどりでした。

ところが、いまでもそうかといえば、決してそんなことにはなっていません。いつのまにか、みんな同じような三つ揃いのスタイルになっている。問いただすと、「いやあ、ここに来たからにはと思って、背広を買っちゃったんですよ」（笑い）というわけです。

タコ壺というのは、別の見方をすれば、金太郎飴につながる。日本の会社では、社長から守衛にいたるまで、同じように会社の方針を唱えることができるわけですね。そうした文化を作ってきたのは高度成長期のように思えるのです。

そして、それによってもたらされたのは、さきほども例に出した造船業のような企業のスタイルです。これは、今までは悪くなかった。つまり、そうした会社がこれまでうまく機能してきたのは、規格品を作っていればよかったからではないかと思えます。

ところが最近では、マーケットの要求がバラバラになってきていて、いろいろな人の種々雑多な要求を全部満足させなければならない。こうなると、これまでの規格品の世界から抜けだして、企画品というアイデアが必要になってくる。しかし、タコ壺からは決して企画品は出てこない、ということです。

そういった意味で、今日のようにいろいろな種類の人が集まって、意見や情報を交換すると機会を持つというのは、非常にいいことだと思いますし、また、これからもそうしていかなければダメだと思います。

Q: ソフト化経済センターに、100万以上の出資金も出して200社も集まるというのは、「ソフト」ということばが、かなりいいイメージを持っている証拠だと思います。

しかし、そのソフト化を支えているわれわれプログラマの仕事のイメージはというと、それほど明るくない。いまのタコ壺文化ということもありますし、新聞記事でも5件しか扱われてなかったりする。そうしたイメージ・ギャップの原因と対処の仕方について、何か御意見はないでしょうか。

鈴木: ソフト化の潮流の中で、ソフトウェア技術者の位置付けが決して高くないということでしょうかね（笑い）。では、イメージ・アップのためのサジェスション

をひとこと。

岡田: 暗い明るいというのは、これはもう価値観の問題ではないかと思えますね。このあいだ、ソフトウェア業界のかたにお会いして、工場みたいに暗くやっていますよ、といわれて現場を見せていただいたのですが、予想に反して（笑い）、けっこう明るくやっているようでしたから、やはり価値観のせいでしょう。私どもは決して暗いとは思っていません。

たとえ、ソフトウェア業界の現状が自分たちで思い込んでおられるように暗かったとしても、バラ色の未来が待っていますし、世の中にはもっと暗い業界があって、こうしたフォーラムを開けないところだってたくさんあるわけですから（笑い）、自ら暗いといって落ち込む必要はないと思えます。

広松: 根本的な答えにはならないと思いますが、むかしはプログラマの人数が少なくて、それなりの存在が認められていたと思うのです。ところが、人数が増えてくると、当然まわりの扱いがちがってきますね。たとえば、生活水準でも、全員が中流ということになってくれば、ことばの矛盾になるので、新しい呼び名を考え出さなければならない。

そういった意味で、ソフトウェア産業やコンピュータ業界に一端の責任があるとすれば、門外漢からみて、この世界はことばのインフレですね。次々に、カタカナの新しいことばを出しすぎると思えます。最近、もしプログラマという呼び名が暗いイメージになっているとすれば、それは、ある種の自業自得ではないかという気がします。

鈴木: まだまだ、議論はつきないと思いますが、予定の時間がきてしまいましたので、この辺で今日のパネル討論を終りにしたいと思います。また、いつかの機会にこうした場を持つことができれば、ソフトウェア業界の人たちだけでなく、われわれのような専門外の者にとっても、きわめて有益だと思います。パネリストのみなさん、御苦労さまでした。会場のみなさんも、御協力ありがとうございました（拍手）。

[文責・編集部]

SEA 秋のセミナー・ウィーク
セッション B1

環境構築と技術移転

講師： 岸田 孝一
報告者： 国布 浩

1. はじめに

このレポートでは、セッション前半の講演の記録を割愛して、後半の自由討論の部分をなるべく忠実に再現することにしました。

講演の内容については、講師が使われたOHPを以下に箇条書きで要約しておきますので、参考にしてください。

1. 環境の概念

開発プロセスとプロセス・モデル
開発アプローチと開発支援環境
開発技法と開発支援ツール

2. 技術動向

(1) これからの環境変革

社会的背景
ニーズの拡大
パラダイムの崩壊
経済的背景
ワークステーションの進歩
ネットワークの発展
技術的背景
ソフトウェア工学の成熟
AI技術の実用化

(2) 技術要素とその動向

ツール
分散化
データベース支援
拡張性
AI

(3) 環境開発アプローチ

進化論的な開発
計画的進化 + ボトムアップ
シナリオの必要性
環境の機能的階層化
プロトタイピング

実用性とR&D

3. UNIXの経験

環境スタッフ

In-House Tool Factory

Advanced Technology Group

スタッフの分散

教育と応用

Testbedの成功と失敗

中堅管理者教育の失敗

新人教育の長期的効果

日本語化とドキュメント

4. 国家プロジェクトの場合

(1) JSDプロジェクト

PPDSの場合

技術的リーダーシップの弱さ

ウォータフォールの弊害

WGの効用

SMEFの場合

強力な技術的リーダーシップ

WGを活用した技術移転

プロトタイピングの可視性

(2) SIGMAをめぐる問題点

開発アプローチ

原思想からの離反

ウォータフォール再び!

技術課題への政治的選択

運営体制

官僚的集権化

技術的閉鎖主義

R&Dの軽視

Right Staff(?)

5. 環境革命の意義

(1) 新技術のインパクト

プロダクトの変貌

開発パラダイムの変化

環境の分散と統合

開発組織の形態

(2) 環境革命とは何か?

幕末との時代的アナロジー

UNIX黒船論

藩政改革か文明開化か?

ネットワークと草の根の志士

討論では、とくに結論らしい結論は出ていません。環境構築や技術移転に関して、これが正しいアプローチだといえるものは、いまのところまだないと思います。会社ごとに、単純には解決できない問題が数多くあります。そのあたりの事情は、討論にも随所に出ています。読者のなかにも、参加者と同じような悩みを抱えている方も多いのではないのでしょうか。このレポートが、そうした問題を1歩でも解決に近づけるヒントになれば幸いです。

2. 環境のイメージ

岸田： まず最初に、「環境」に対するイメージを整理しておきたいと思います。

実は、最近4~5人で、環境やツールについて何回か討論を重ねているのですが、その中で、おたがい「環境」というひとつのことばで違うことをいっているのではないか、ということになりました。「何のための環境か」という観点で単純化すると、ソフトウェア「工場」とソフトウェア「工房」という2つの対極的なイメージに分かれます。

ソフトウェア工場とは、設計と製造の分離という概念を土台として、ウォーターフォール型の流れ作業で、つまりテレビや自動車を作るのと同じスタイルでソフトウェアを作っていくという考えです。そこでは、ほぼ仕様の決まったソフトウェア、しかも作り方もよくわかっているような製品について、効率よく生産するにはどうしたらよいか、ワークステーション上でどんなツールをどんなふうに組み合わせ、全体のデータベースをどんなふうに管理するのかということが問題です。

それに対して、工房とは、どこかの会社で働いていたけれど、アイデアがあふれすぎたためにスピン・アウトして、どこかのマンションの1室を借りて、気のあった仲間数人で、わりと技術的におもしろい仕事だけを引き受けてやっているところのイメージです。何を作るんだ

か漠然としているような注文をうけて、ユーザと直接対話したり、ネットワークでいろいろな人の意見を聞いたりしながら、仕様と設計とプログラミングをほぼ同時平行的にすすめてゆくという、ある意味では非常に職人的な環境になります。

そこで、まずみなさんに自己紹介をかねて、いま仕事をしている環境と、ことばに対する主観的なイメージをお話いただきたいと思います。

[結果]

工場派——10人

工房派——10人

ありがとうございました。それでは、討論に入りたいと思います。どなたからでも、どうぞ。

3. 環境をどう導入し、ひろめるか

O： 私の会社は、マイクロプロセッサのソフトを作っていて、従来はインテル系のスタンダードアロン型デバッグを使用していましたが、今年4月にUNIXマシンが入ってきて、そちらに移行している最中です。

いまのところ、まだUNIXの機能を使いこなしてはいません。平均的プログラマにとっては、別にいろいろなツールが使えなくてもよいわけで、すぐに、コンパイルを何本か流したりする。そうすると、UNIXの専門スタッフから、そんなことをするとマシンの負荷があがって、エディタが動かなくなる、と叱られてしまう。一体、われわれ「一般人」はどう使えばいいのか(笑)。

岸田： ハードウェア設備と利用者の人数は?

O： マシン自体は、VAX750以上の性能があるといわれているものが、3台あります。1台あたり常時10人程度で使っていますが、ただそのマシン本来のリアルタイムOSの下でUNIXが動いているせいか、とまかくレスポンスが遅いんですね。

岸田： 従来環境と異なる新しい機械が入ってきて、いきなり無制限にみんなに開放したわけではないと思うのですが、導入時の教育とか、使い方とかについて、具体的な問題をあげてみていただけますか。

O： 最初は、使いたい人が使うということで、主に、物好きの多いわれわれのグループだけが使っていました。ところが、他のマシンを使っていたグループが、そこから追い出されて、大量にこちらに移動してきた。そこで、マシンの運用をどうするかとか、いままであまり気にしていなかったパフォーマンスの問題が出てきたわけです。

岸田： Oさん自身としては、そうした問題をどうしたら解決できるとお考えですか。

O： メーカー系のソフト会社ですから、他社のマシンを入れるのはむずかしいので（笑い）、プロジェクトごとにワークステーションを導入する方向で、負荷分散を考えています。

N： 私の会社でも、最近UNIXマシンを入れましたが、一番の問題は、汎用大型マシンのTSSを使っている人たちが、なかなかUNIXを使ってくれないことです。

岸田： われわれの場合もそうでした。6年前にUNIXを入れて、いまは2人に1台の端末があります。受託開発作業の5～6割が大型の事務処理（バンキング・オンラインなど）ですが、そちらの部門は、どちらかといえばUNIXにそっぽをむいて、あいかわらず汎用TSSにしがみついている。

一方、Oさんのところのように、プロセス制御系の受託開発部門は、客先の環境が劣悪で、MDSの前に待ち行列を作っていたり、片道1時間半かけてデバッグに出掛けたらマシンがダウンしていたとか（笑い）という状況で、こちらのチームは、すぐにUNIXにとびつきましたね。

現在では、制御関係の仕事の9割以上は、最初からUNIXを使うという前提条件で受託しています。その結果、数年前まで赤字だった仕事も黒字になってきた。これは、まあ、機械化したんだから、当然といえば当然の話です。一方、バンキング・システムはどうかといえば、こちらは機械化以前の問題で、スケジュールがめちゃくちゃにハードですから、どのプロジェクトも火をふいている（笑い）。

N： 知らない人は、何かよいものがあれば、すんなり受け入れるということですか。

岸田： 現状が、紙と鉛筆しかない悲惨な状況だと、UNIXでもパソコンでもMDSでも、はいりやすいでしょう。それに、ゼロまたはマイナスの状態から、ともかくプラスになるのだから、効果も大きい。

しかし、汎用TSSの環境には、一応それなりのツールがそろっているので、たとえば他にもっといい環境があると聞いても、慣れの問題もあるし、なかなか、これまでの住み慣れた世界から抜け切れないということはあるでしょうね。

T： 私の会社でも、スタッフの人たちはUNIXを使

っていますが、私たちはラインの人間は、触ったこともありません。

私自身は客先で、汎用機のCOBOLやFORTRANを使って仕事をしていますが、いまあるツールにちょっと手を加えて、使いやすくする程度で、あまり環境という概念は意識はしていません。

ツールはいっぱいありますが、ゴミ箱というか、オモチャ箱というか、どこに何があって、どう使ったらいいのか、わからないというのが現状です。岸田さんの話に、ネットワークでツールをばらまくということが出てきましたが、そうした情報交換はほとんどありませんね。

S： うちでは、半年程前から開発にUNIXを使うようになりました。マシンはVAXで、私のセクションには、2人に1台くらいの割で端末があります。他のところは、セクションごとに1台くらいしかなく、マネージャたちが生産管理などに使っているようです。

O： 私のところでは、生産管理は大型のマシンでやっています。

G： 私の所には、かなり大きなホスト・マシンがあって、開発はTSSでやっています。UNIXについては、個人的な興味はあるのですが、どういうものかよくわからない。

そうした「工場」的な環境で開発をしていて、まず第1に考えるのが納期ですね。期限までに仕事を終わらせるには、どんな環境があれば役立つのかということです。現状は、納期間際になると、1日24時間マシンを使っている、とにかく端末の待ち行列が多くてしかたがない。ところが、空いているときは誰も使わないわけです。これは、環境の問題というより、作業工程の問題かと思いますが。

岸田： 端末の台数はどうですか。

G： 5～6人に1台くらいだと思いますが、ホストは開発専用ではなく、定例業務もやっています。全国を結んだオンライン・サービスもやっていますし、バッチジョブも流しています。そこに、TSSで開発用端末をつないでいるので、レスポンスが遅いというネックがあります。それをどう解決するのも問題です。

岸田： その場合、端末を使うのは、プログラミング以降の工程ですか。

G： コンパイルから、結合テストまでですね。設計をしているときは端末は遊んでいます。

岸田： ドキュメントの機械化についてはどうお考えで

すか、設計段階から機械を使うというのは、

G: 最近やっとみんなの関心がそちらに向きはじめてところで、具体的には考えていません。正直に言えば、そのためにどういったツールや環境があるのかということすら、よくわかっていない状況です。

O: 私のところでは、ドキュメントの機械化を、これからUNIXマシンでやろうとしているところです。現在はワープロを使っています。

S: それは何かヒナ型みたいなものを用意して、エディタで個別の情報を入れてゆくやり方ですか。

O: そうですね、モジュールの仕様記述を一定の様式に統一していますので、既存のドキュメントをエディタで修正するかたちで、仕事はずいぶん楽になりました。他のフェイズとの連結は、まだ何も考えていません。

F: 私のところでは、J-Starを導入してドキュメントの機械化を試みているのですが、やはり最初の定型を作るのに時間がかかります。マシンは10台あるのですが、全員で使えるだけの数ではありませんから、効果がどれだけあがっているのかは、よくわかりませんが、

4. 環境の自動化について

岸田: いま、環境とかツールとかいった場合に、プログラミングやデバッグ用のツールが、主に使われていると思います。しかし、これは10何年も前からいわれていることですが、ソフトウェア開発で一番問題が多いのは、それ以前の工程です。

アメリカのTRWという会社で、UNIXを導入を計画したときに、作業分析をしてみたところ、1つのプロジェクトに参加している人間(マネージャ、設計者、プログラマ、セクレタリすべてを含めて)が使っている時間のうち、直接ソース・コードと関係のある仕事は全体の2割程度しかなく、残りの8割は、ドキュメントを書いたり、打ち合わせをしたりという、いわゆるOAとか文書処理のカテゴリーに属することに費やしていることがわかりました。

TRWは、要求定義のSREMその他のソフトウェア・エンジニアリング・ツールを開発したことで有名な会社ですが、UNIX導入の最初の時期は、すでに別のマシンで動いているそれらのツールはほうっておいて、むしろドキュメンテーションとかプロジェクト・ミーティングの効率化にUNIXをどう使うかを重点的に考えた。その結果、生産性を2倍くらいあげています。

多分、これからわれわれが自分の周りの環境の自動化を計画するときに、このあたりが一番のキーポイントになるのではないかと、思います。その場合に、メインフレーム上の環境では、どうもうまくいかないのではなからうかと、感じています。

B: そういう意味では、むしろ、パソコンやマイコンのほうが、ツールの蓄積はるかに多いのではないですか。たとえばMacなんか。

岸田: 私も半年前ですが、Macを買いました。日本語機能がいまひとつなので、まだOHP作りにはUNIXを使っていますが、アメリカの同業者を見ると、Macの愛用者が多いですね。

W: これからワークステーションの時代だといわれていますが、プログラミング以前の、アイデアの整理や、ドキュメント作りの仕事で、どんな支援をコンピュータに期待できるのか。

そのときに、できれば、自分のだけじゃなく、世の同業者のノウハウや、成果を何らかのきっかけでデータベース化して再利用できれば、一番コスト・パフォーマンスがいい。しかし問題は、Macを1台買ってくればすむかということで(笑い)。決して、そういう話ではないですね。

岸田: 私の会社にも、いつのまにかMacが何台かごろごろしていますが、正式にワークステーションとして採用したわけではなく、少数のMacフリークたちに、好き勝手にいじらせている段階です。問題は、かれらが達成したの成果を社内にどうやって移転していくかでしょう。

会社を運営してゆくためには、工場的な色彩で、日常のオペレーティング・コストを稼いだ部門は当然必要ですが、ただそれだけは、これだけ変化の激しい世の中で、5年後、10年後に飯を食っていけるかどうかの保証はない。

新しい技術動向をウォッチし、それを取り込むための部門をどんなかたちでつくり、維持してゆくかが、ソフトハウスみたいな商売には、大切だと思います。いわば、工場の中の工房といったらよいでしょうか、しかし、その両者の接点はむずかしいですね。これで正解というやり方は、ないんじゃないかという気がします。

5. 技術格差と技術移転

N: いわれることの意味はわかります。UNIXでも

Macでも、頭のいい人間はほっておいてもどんどん勉強していく。ところが、そうでない普通の人や、初級者とか問題のある人に対しては、どう教育していったらいいんでしょうかね。

H: 私は、UNIX環境のまったくなかったところに、UNIXを導入して、教育やら周囲の説得やらで、ずいぶん苦労した経験があるのですが、最近では、なんにもしないのが一番いいのではないかと、思ったりしています(笑い)。

UNIXというのはひとつの文化みたいなものだから、なんにもせずに、人々を新しい文化に直接ぶつけて、ある種のショックを与えるのが、まず第1段階として必要ではないでしょうか。

岸田: 技術移転というのは、技術に格差があるから移転が必要であり、それが仕事になり、商売になるわけですね。ある人が情報をたくさんもっている。一方、こっちには情報を欲しがっている人または必要としているだろう人がいる。そうしたら、2人の間に何らかのリンクを張るしかない。

たとえば、会社の若手で有望な人間がソフト工場の片隅でCOBOLに押し潰されて死にかけている(笑い)、そういう状況を見て、管理者として、何とかしてやろうと思ったら、かれに刺激を与えるための情報リンクをどうやって張るか、です。そういう場合、具体的なツールやワークステーションが、抽象論でなく、外部世界への具体的な情報ゲートウェイとして役立つのではないかと思います。

その上で、教育とかセミナーとかは、そうした刺激を与えたあとでおこなうのが、一層効果があるのではないのでしょうか。まず、具体的にたくさん情報の入っている(または入ってくる)箱みたいなものを、みんなの目につくところに置いておくのがいい。

A: いまのお話ですと、いろいろな人間がいて、どこかへこんだところがあれば、そのへこみをならすために、外部から機械なりなんなりを与えて、救ってやろうということですか。

岸田: いや、むしろ、もっと前向きに、もし自分自身が他人よりある点で進んでいると思ったら、より多くの情報を得て、さらに前進する工夫が必要だと思います。

また、後ろからついてくる人間に何かを与えるだけではおもしろくありませんから、別のジャンルで、自分より技術的に強い人間を見つけて、かれの知識を吸収する。

ときには、そういう人間に新しいゲートウェイを与えて、さらに伸ばし、自分にとってのアンテナとしても役立てるといえるでしょう。

N: 刺激という話題に関してですが、私のところでもSUNワークステーションがあって、たしかに興味を持った人間は触るのですが、全然触ろうともしない人たちがいる。汎用TSSの世界で生きている人たちですね。何とかしなくちゃ思って、研究会みたいな場を作って、UNIXの勉強を始めたのですが、それでも動いてくれないので、困っているのですが。

岸田: それは段階があるのではないかと思います。どういう勉強会をやられたのか知りませんが、一番最初の段階は、底上げということよりも、誰が潜在的に可能性を持っているのかを選び出すために、勉強会の場を利用すべきではないかと思うのです。

で、何人かセレクトして、その人たちを1歩も2歩も前進させる。そうすると他の人たちは、自然とついてくる。それが第2段階です。そして、ある程度みんなのレベルが上がった時点で、本当に重点的な教育をするというステップです。

たとえば、私の会社で、ずっとUNIXを嫌っていた大型事務処理の受託部門が、いまUNIX化の方向へいっせいに動き出していますが、それは、工房派の連中が、UNIXの文書処理機能やグラフィクス・ツールを使って出力した日本語のソフトウェア仕様書を目にして、これは自分たちにも使えると思いはじめたからです。

そこにいきつくまでは、じっとガマンですね。ですから教育というのは、全員の底上げ教育でなくて、最初は、誰が潜在的リーダーなのかという選択をする手段として考える必要があると思います。ソフトウェア技術は、もともときわめて人間的ですから、そういう宿命を持っているのではないのでしょうか。

6. ネットワークのインパクト

U: 情報ゲートウェイという観点からいうと、ワークステーションとネットワークのもたらすインパクトは大きいですね。日本UNIXユーザ会のネットワークjunetに、最近うちもつながったのですが、外国へのチャンネルがあって、ARPAやUSENETのニュースが自分の端末で読めるというのは、ものすごい刺激です。

岸田: 国際電子メールは、いまや常識になりつつあると思います。

いま、私が関係しているはICSEのプログラム委員会でも、プログラムの構成についての議論や論文の審査は、実際に委員会が開かれる前に、ほとんど電子メールですませてしまうかたちになっています。委員会は11月にロッキーの山の中で開く予定なのですが、この分だと、十分スキーを楽しむ余裕がある（笑い）。

それから、いま発行が予定されている環境関係の新しい雑誌では、原稿集めから審査、編集まですべてネットワークの上でやろうとしています。

E: そういう意味では、シグマのネットワークに大いに期待を持っているのですが。

岸田: 私たちが2年前、JISA/STCでシグマ・プロジェクトの最初のコンセプトをみんなで議論したときに期待したことは、極端に言えば、それだけです。

ただ、実際のプロジェクトの進め方をみていると、いささか不安になってきました。いまのところ、東京にメインフレームのマシンを何台か入れた巨大なセンターを作ることにしか決まっていなくて、そんなものを作るくらいなら、その予算の半分以下のお金で、全国各地に、ネットワーク・ノードを設置することができます。われわれのオリジナル・コンセプトでは、まず、そうした小さなサブセンターを日本中に作るべきだと主張したのですが、残念ながら受け入れられませんでした。

1つの大型センターを中心に、そこにみんながつながるというのは、1時代前のTSSの発想ですね。ネットワークという以上は、あちこちに、それぞれ地域の特徴を持った、しかし権利としては平等のノードが存在しなくてはなりません。それらのノードは、また、シグマにおける環境開発・技術移転の地域センターとしての機能を自動的に果たし、有用なツールがあちこちから芽を出して、成長していくでしょう。

しかし、こうした草の根民主主義的な考え方は、どうやらお役所の方々に嫌われたようです（笑い）。

X: 地方のサブセンターについては、作ってほしいという要望がシグマ開発本部に寄せられていて、これから検討中だとのことですから、あきらめるのは早いと思うのですが。

岸田: もちろんあきらめてはいません。ただ、当初の予定からすると、すでにシグマのネットワークは動いてくなくてはいけません。

A: 予定は予定で、そうすぐには動かない（笑い）。

岸田: ボランティア・ベースのjunetが、現在、

ほぼ全国的な規模で動いているのに、シグマのネットワークはなぜ動かないかが問題だと思うのです。

A: シグマでは、標準ワークステーション上でネットワークを動かそうとしていますから、そうすぐには無理でしょう。現在、秋葉原の開発本部を中心に、プロジェクトの開発を担当する会社同士のネットワークを張ろうと計画しています。

岸田: それは、あくまでシグマの出資会社や、開発参加会社だけの閉じられたネットワークですね。そうでなくて、シグマに興味を持った人ならだれでも入れるような開かれたネットワークがまず必要だと、私はいつているのです。そうしたネットワークを通じて、いろいろな人々の意見を吸収し、フレキシブルに計画の方向を修正できるようなプロジェクトの構造を、オリジナル・コンセプトの中で私たちは考えていたのです。

いまのやり方でも、何年かさきには最終的には同じネットワークができるかも知れません。しかし、私は、開発の成果もそうですが、そこにいきつくまでのプロセスが、より大切だと思います。そのプロセスを通じて、みんながいろいろなアイデアや考え方をもち寄ってこそ、日本のソフトウェア開発のカルチャーを変えることができる。これがオリジナルなシグマへの提案でした。

今日のセミナーのテーマに関連づけていけば、最初に何かのポリシーや先入観を持って環境を設計するより、技術移転の促進という観点に立って、ノンポリシーではあるが、非常にインパクトの強いワークステーションなりネットワークなりツールなりを用意して、みんながそれに対して、どう反応するかを観察し、その反応を正しい方向へ導き、加速することによって、人々のものの考え方や仕事のやり方を変えていくことが大切だと思います。

7. 終わりに

こうした討論の例にもれず、議論が燃えかかったところで時間切れになって、ネットワークと技術移転の問題が論じつくされなかったのは、残念でした。なお、講師のシグマに対する批判的意見は、米国 UNIX Review 誌 87年2月号のインタビュー記事に、よりストレートなかたちで載っているのだから、興味のある方はお読みになれるとよいでしょう。

SEA 秋のセミナー・ウィーク

セッション E2

データ構造に基づくソフトウェア設計の一方法

講師：白井 義美

報告者：野辺 良一

1. はじめに

本セッションは、前半を講師自らが実際の仕事で使用している、データ構造に基づく設計方法の説明があり、後半は前半をひとつの例とする設計方法に関する自由討論が行われた。

ソフトウェア設計の技法や方法論の紹介は、これまで多くの文献やセミナーでも行われているが、ソフトウェア開発の現場で日々仕事をしているエンジニアがどのような現状にあり、またなにが問題になっている、それをどう解決していこうとしているのか、といった現場に密着した場を提供しようということが、講師の主旨であった。

2. SP-FLOWとは

講師自ら使用しているこの方法を実現化したきっかけを、講師はこう話している。

「現在もソフトウェアの商品をつくらしているが、その商品をつくりながら、どういう設計をして行ったらいいかを考えたが、目的としては、いかに手をかけずにまともなのができるかということです。一生懸命に書けばいい設計技法はあるのですが、コストにあうようにするには、また、いかに効率よく行かせるかを主眼にし、かつできあがったシステムがちゃんと動くということまで、現在あるようなものになっています。

この方法を、SP-FLOWと呼んでいますが、Structured programming のではじめの頃から使っていて、少し古典的なイメージがするかもしれません。しかし、現在ソフトウェア開発の現場で使われている設計技法の多くは、60年代後半から70年代に提案されたものであり、そうした方法（技法や方法論）の機械化を含めて、どう使っていくかというのが、ソフトウェア開発環境の現在の問題点ではないかと思えます。

ソフトウェア開発には、開発方法のモデルがいくつかあって、古典的なものではウォーター・フォール・モデルといったものがありますが、多くのものは設計工程を

いくつかに分けています。SP-FLOWでは、この各設計工程を、つまり、要求定義からプログラム設計までを、同一の様式でもっておこないたいというところからスタートしています。つまり、各工程で関係する人達が、共通に理解できて、かつ保守にもそのまま利用できるように、ということです（図1参照）。

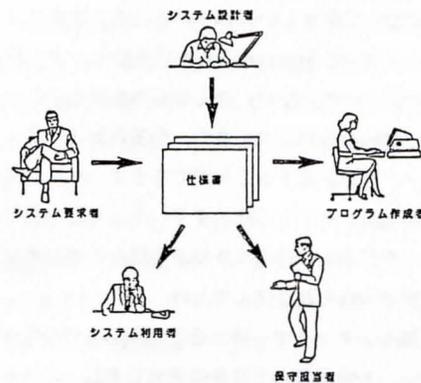


図1 SP-FLOW利用概念図

そのために、記述の形式性と自然言語的な読みやすさという相反する特性を、制御フローと処理記述とを組み合わせることによって、合理的に解決しようとしたものです。」

3. SP-FLOWの基本的概念

(1) SP-FLOW用紙

この方法では、設計の際の自由な分析を標準化された表現で記述できるように、専用の用紙（SP-FLOW用紙）を用いている。この用紙は、時間の流れが縦軸に設定されており、全体で5つの記入エリアが用意されている（図2参照）。

図2の左から順に、システム外のデータの動き、ファイルの入出力、システム内のデータの動き、制御の流れ、処理の内容の記述が行われ、それらをお互に関連づけ

でき記入できるようになっている。

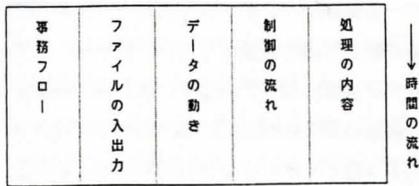


図2 SP-FLOW用紙

(2) 記述法

処理するシステムの内容を表現するために、次の3つの要素を明確する必要がある。

- ・制御の流れ
- ・データの流れ
- ・処理の内容

SP-FLOWではこの3つの要素を、専用用紙に次のように記述している。

- ・制御の流れは、縦方向に時間軸を設定し、それに沿って構造化された制御線を書くことであらわす。
- ・データの動きは、特定のデータエリアを用紙上に設定し、そのエリア間の動きをデータフローとしてあらわす。
- ・処理の内容は、制御の流れの一時点を指定し、データエリア上に示したデータの変換処理を文章によって記述する

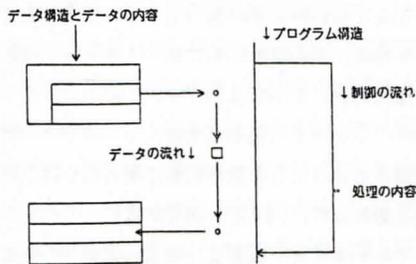


図3 SP-FLOW記述法

(3) データ構造の記述

SP-FLOWで使用するデータ構造図は、データ・フォーマットの記述としても利用できるように、各項目の左側に構造化レベルを示す枠を付加したものである(図4参照)。

(4) その他

これまでに記述した以外に、

- ・プログラム構造の導出方法
- ・モジュール化の方法
- ・文書化
- ・最適化
- ・今後の展望

等が話されたが、紙面の都合上省略させていただいた。

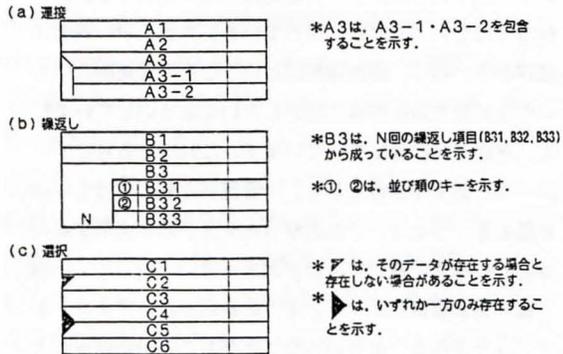


図4 データ構造図例

4. 自由討議

(1) 設計方法の現状

A: わたしのところでは、主にNSチャートとモジュール構成図を、制御系とデータ系の両方で使っています。機械化がされているわけではなく、社内に設計の基準があってそれに基づいてやっています。

B: PADを使っています。それで先ほどのSP-FLOWの話を書きいて、SP-FLOWでは縦軸を時間にしていますが、PADですと、詳細化をしていくと右方向に深くなってきて、時間的経過が解りにくいですね。それと、専用のテンプレートがあるのですが、枠組を書くのがひじょうに面倒です。

C: HIPOで概念設計をして、それからPADで残りの細かいところを落すようにしています。

D: わたしたちのところでは、フローチャートを使用していません。ユーザと一緒に概念設計をして、これには自然言語の手書きのものを使用していますが、そこで細かい点まで決めてしまい、それからはプロトタイプングしながら作っています。

E: 我々はフローチャートを使っていないのですが、フローチャートを書けばプログラムが書けるかという、そうじゃないと思います。言語にはその言語に適したフ

ローというのがあると思う。例えば、F o t r u n 向きのフローというのがあったほうがいいのではないかと思う。つまり、言語特性を表現しやすいフローチャートというのは、ディテール・フローの世界では、一番重要ではないかと思う。もっと上の工程、つまりユーザが要求してくる要求定義を図式化するのとは、区別しないといけないのではないか。

F: 決った形式でデータを記述していくような設計の仕方ではないのですが、作っているシステムが、画面の遷移が多いので、画面遷移図ということで各画面のいろいろな特徴を画面の横に記述していくようにして、それが設計書になっています。これはある意味で、ユーザと話す叩き台として、一番便利なものではないかと思える。つまり、できあがったシステムを使用する人達とのQ&Aを通して設計が固まっていくからです。例えば、ある画面があって、そこに表示されるアイコンをクリックすると、どういったことがおこるかといったような機能の決定には、ひじょうに役だっています。

(2) ユーザにとっての判りやすさ

白井: かつて私はNSチャートをずいぶん使ったことがありまして、修正を支援してくれるようなワークステーションでもあれば違うかもしれませんが、とにかく書き直しがたいへんでしたね。先ほどHIPOとPADを組み合わせて使っているという方がいましたが、HIPOというのはご存じのように、入力と出力と処理がありますが、それを分けて書けるというか、分けて書くのですが、それがエンド・ユーザにとって読みやすいように思えます。しかし、制御の流れを具体的に示せないところに、大きな問題があるように思えます。ですから、細部になると、他の方法を使うようになる。

G: わたしどもではPADを使っていますが、標準化ということではなくて、会社の部署によって開発言語が違ってまして、私の所ではCを使っていますので、それにあわせてPADを使っています。要求仕様とかという設計の上流ではPADは使っていません。

白井: そういう意味でいいますと、採用するプログラミング言語によって、随分違うことはありますね。HIPOというのはCOBOLを意識して作られたと思うのですが、自然言語で文章をかなり書きますから、エンド・ユーザ向けとはいえますね。

H: エンド・ユーザとの関係ですが、COBOLの世界でエンド・ユーザが関係するというのは、先ほどもち

よっと似た話がありましたけど、ほとんどが画面と帳票で、これが一番重要なインタフェースになります。ですから、画面の動きや構造を見たりとか、帳票のレイアウトを決めるのが重要になっています。あとSEが手を加えるのは、画面の要求に対して、個々のサブプログラムをどういうふうなデータでリンクしていくかということとを設計して行くので、7割くらいは自動的にできる。

(3) ディテール・フロー・チャートは必要か

I: ディテール・フロー・チャートは必要ないのでは、という問題提起したい。というのは、従来からフロー・チャートは何のために書くのかという議論がありまして、プログラムを作るためというのと、メンテナンスのためにリストでは追えないからフロー・チャートが必要なんだということがありました。

で、後者のほうですが、最近、プログラム・ソースからフロー・チャートを作成するツールが出てきていたりして、メンテナンスの面からはフロー・チャートはいらなくなってきている、といえるのではないかと思います。次に、コーディングのためにフロー・チャートが必要だということですが、事務処理と技術計算を一緒にレベルでは扱えませんが、少なくとも事務処理でいいますと、まあCOBOLが大半ですが、そのCOBOLの世界では、ある程度の仕様を与えてやると、COBOLのソースを作成してくれるツールというの出はじめていますが、実際われわれはそういったツールを使っているのですが、それを使っていると、プログラムは、制御構造の意識をしなくてもいいようになってきています。ですから、プログラミングのためにもフロー・チャートはいらないのではないか、ということです。

J: フロー・チャートの是非ですが、プログラム・サイズとか複雑さということ抜きにして考えないほうがいいのではと思います。例えば、典型的なCOBOLのプログラムでいいますと、帳票とか画面の設計がちゃんとできているものでは、大きくて1,500くらいで、まあ、500~800くらいで済んでしまいます。そういうものと、複雑なデータアクセスを要求されるような万を超えるようなプログラムとを、同等なレベルでは議論できないのではと思います。で、短いプログラムや単純なプログラムに関しては、ドキュメントは残す必要がないと思います。

K: 今の議論で疑問なんですけど、機能分割が上手に行われていて、つまりモジュール化さえしっかりしていれば

ば、概要は必要だが、内部のディテールに関する説明は
いらぬ、ということなんでしょうか。

白井： 私はいらぬのではないかという意見ですね。
何が大切かといいますと、プログラムというのはプログ
ラム・リストにのっていることしかやっていないんでし
て、プログラムが何をしたいかということリストから
探すのはいへんですから、概要みたいなものを残し、
後はコメントなりをつけて中をみたりすればいいのでは
ないか、ということです。

L： システム設計とプログラム設計は別ものだと思
います。要求仕様とかシステム設計をおこなうときの
情報というのは、後、つまりメンテナンスでは必要だと
思います。また、プログラムに分割した後でも、そのア
プログラムがどういうものかを示すものというのは必要だ
と思います。で、私がいいたいのは、その後プログラ
ムの機能が決った後、その中をどう展開して行くかとい
うようなプログラム構造に関しては、フロー・チャートの
ような細かなものはいらぬと思います。

(4) 保守に役立つ方法

M： 私は主に人の作ったプログラムの保守をしてい
るのですが、保守をするときに、データの流りがはっきり
判っていないと、ひじょうにメンテナンスしづらいです
ね。というのは、わたしどもが保守をしているシステム
というのは、外部環境ですごく変わるんですね。です
からデータの流りというのを明確にする必要があります。
メンテナンスのときに、詳細なデータ・フローはいら
ないんですが、データの流りは必要です。

白井： HIPOにしろPADにしろ、設計段階で使用
されていると思うのですが、同じものが保守にも役だっ
ているのか、という疑問がありますね。で、私の経験か
らすると、保守に一番役立つのドキュメントは、コメン
トではないかと思っています。つまり、ある処理をや
っている前に、そのやっている内容が書いてあるのが一番
役だっていました。で、現在はどんな方法でデータの流
れを押えていますか。

M： はっきりいって、今メンテナンスをやっているシ
ステムには、データの流りを示すものはありません。で
すから、逆に仕事の最初は、データ・フローをつくる
ところからはじめています。つまり、プログラム・
ソースをみて、データの流りを明確にしてから、とい
うことです。それを行うための技法とか方法は特に決め
られていませんので、用紙とフォーマットが決っている

だけで、その中にどう書くかということは設計者自身に
任されています。

(5) 可逆性による設計の検証

N： 前半の講演にあったのですが、要求からプログ
ラムの間を一貫してうまくつなげる、ということが重要で
はというお話があったと思うのですが、一番最初に要
求定義があつて、それを実現するためにいくつかの部品
に分けていくということを設計でやると思うのですが、
その要求定義からできた部品構成の根拠が明らかにな
ればいいのではないかと思います。つまり、可逆性とい
うことができればいいのではないかと。それによって、設
計の検証も可能になると思うんですね。それで、開発と
保守の両方に役立つということにもなります。

白井： おっしゃる通りだと思います。ただ、それをやる
ためには人間だけでは、つまり手作業では無理でしょう
ね。そういったツールは当然必要ですが、そういったア
イデアを簡単に試してみ、今日みたいにみんなでワイ
ワイやって、それでどうしたらいいかという、ハード
ウェアを含んだ環境というのが、ともかく必要ですね。

今日参加しているみなさんの環境というのが、ともか
くいろいろあるようで、短い時間のなかでいまひとつハ
ッキリしていないのですが、今後のソフトウェアの設計
において、言語と人間の間をフォローするツールとい
うのが絶対に必要ではないかと思っています。できれば、ワ
ークショップのように何日か泊り込んで、議論をするとい
うような機会をもって、この続きをやりたいと思います。
今日はこのへんでおわりにしたいと思います（拍手）。

5. おわりにかえて

講師の白井さんが最後にいっていたように、なんら
かの結論を出すには、少し議論の時間が短すぎるのと、参
加者のお互いの環境の理解が充分でないと、設計の方法
論の是非までは議論が発展しないようです。

これはソフトウェア・エンジニアの特徴なのか、それ
とも人間本来の性格なのか、自分の普段やっていること
をやめて、新しいことを取り入れるには、なかなか抵抗
があるようです。特に年寄りといわれるようになってか
らこの傾向が出てくるようですが、世の中で30代を年
寄りとはいわれないし、まして思考の柔軟性がなくなるな
どとは考えられない。ソフトウェア開発の世界が、いつ
まで「紺屋の白袴」でいるかというのは、われわれエン
ジニア自身が思考方法を転換していかなくてはならない
のではないのでしょうか。

初めての海外調査

道正一郎

協同システム開発

1. はじめに

12月1日から約2週間、協同システム開発（JSD）で進行中のソフトウェア環境統合化技術開発計画プロジェクトの一環として、海外調査を行った。このプロジェクトは、次期シグマ計画ともいわれており、通産省・情報処理振興事業協会からの予算で行っている国家プロジェクトであり、プログラム自動生成を含んだ、要求定義・設計から保守までを一貫して支援する環境を構築する技術開発をおこなっており、昭和60年を初年度とする5ヶ年計画のものである。今回行った調査は、プロジェクトの主旨に基づいて、国際会議等の出席を中心に、ソフトウェア開発の最新技術動向を探り、それをプロジェクトの計画に反映させることがおこなった目的である。

私自身にとって生まれて初めての海外出張であり、貧困な英会話と貧困なソフトウェア技術の知識のため、十分な新技術の取得は楽ではなかったが、いくつかの重要と思える先進的な技術内容を垣間見ることができたのが、幸運な体験であった。

調査の前半は、テキサス州オースチンで開かれたCS CWのコンファレンスへの参加、後半はカリフォルニア州パロアルトで開かれた第2回PSDEのシンポジウムの参加が中心であったが、そのほかにもいくつかの会社を技術訪問できた。

この2つのイベントに関しては、わたし自身も報告者として参加したSAE主催のフォーラムが、昨年12月に開催されている。そのレポートが近々SEAMAILに載るそうなので、わたしのこの報告は、技術的なものは少なくして、経験談として読んでいただければ幸いである。

2. 東京からロサンゼルスへ

12月1日、20:35成田発のJAL 64便の機上のひとになって約9時間、ロサンゼルス空港に到着した。17時間の時差のため、到着時刻は同日午後1時近くであった。気温は26°Cと予想していた以上に暖かく、最初は冬着ばかり用意して失敗したと思ったが、昼夜の温度差は激しく、むしろ夜は日本以上に寒さを感じた。

ロサンゼルス空港が私にとって異国への第1歩であり、ここでの入国審査が最初の手続きである。同じ日本人でも、目的が観光とビジネスでは、入国審査の際の質問内容がかなり異なるようである。観光の場合は、ほとんどがフリーパスであるがビジネスとなると、どのような目的で、どれくらい滞在して、アメリカのどのあたりを回るのか等、かなりしつこく聞かれる。私に立ち会った黒人女性の審査員は、大柄でもあり、その質問を受けた自分は、まさにヘビににらまれたカエルを思わせた（私自身は日本では大きい方だが、アメリカ人の大きさは別格という感がする）。

ともかくにも、なんとかアメリカでの第1関門は通過できたのだが、ただ、いずれにせよホテルでのチェックイン・チェックアウト、トラベラーズ・チェックから現金への交換、国際電話、食事等、フォーマルな会話部分（ある一定のパターンがあるもの）は、なんとか理解できても、インフォーマルな部分になると皆目対応できない。そんなことから、3度の食事を例にとっても、メニューをみて一応セレクトはするのが、出てくるものがイメージに反することが多々あったりして、また、お酒についてはメニュー等なく、予め銘柄を知っていなくてはならない（どこにでもあると思っていたジンライムを頼んだが、それはないと断られてしまった）。このように、日常的会話もままならず（質問すれば増々判らなくなってしまう）、更に固有名詞も乏しいため、ともかく旅行中必須とされる手続きは、終始緊張の連続であった。

3. オースチンへ

翌日朝、コンチネンタル航空にてコンファレンス会場になっているテキサス州オースチンへ向かった。ロッキー山脈を越えて約3時間、前日ロスで合わせた時計は12:30を指しているが、ロスとの時差で14:30に修正しなければならない。アメリカ本土は、4つのタイムゾーンに分割されている。アメリカ国内を駆け巡る営業マンは、客先とのアポイントメント等のタイム・スケジュールを、どのように管理しているのだろうか。また、ネットワークで張られた各々のコンピュータの時刻をど

のように管理しているのだろうかと考えさせられる。

オースチンは、北にダラス、東にヒューストンを臨むテキサス州の中央に位置し、情報産業の広がりから成長したところである。

(1) CSCW' 86の参加

Computer-Supported Cooperative Workと題するこのコンファレンスは、技術的・社会的・組織的・認知的な作業領域の視点から、コンピュータ支援による共同作業において、相互規律の観点を中心とした研究成果の発表であった。応募約80編の投稿論文のうち30編が採用され、それらは、

- ・共同作業の技術的(産業及び研究機関)側面からの経験
- ・共同/チームワークの経験的研究
- ・グループ行動、組織構造及び実践的作業の上でのコンピュータ技術のインパクト
- ・技術サポートの現状(例えば、データベース、構造化ドキュメント・システム、ハイパーテキスト)
- ・グループ作業を分析するための理論的モデル
- ・上記に関連する、マルチ・メディアを使った会議、グループの意志決定支援、ヒューマン・コンピュータ・インタフェース

といったトピックに焦点をあてている。

その中で、ブレンストーミングやKJ法によって、黒板、大きな模造紙、カード等を使用して行われてきた従来のグループ討議にかわり、XeroxのCOLABシステムのようなニューメディアの出現から、ワークステーション等を利用することによってWYSIWIS(What You See Is What I See)の原則で、共同作業のあり方を考え直す、いわゆる共同作業のパラダイムのシフトについて論述されたセッションが印象深かった。

(2) MCCへの訪問

コンファレンスの期間中、ご多忙にもかかわらずL.A. Belady氏自から、上記CSCW' 86の主催者であるMCC(Micro electronics and Computer Technology Corp)を案内していただくこととなった。MCCは、DEC、ハネウェル等のコンピュータ・メーカ、および、モトローラ、ナショナル・セミコンダクタ等の半導体メーカとの共同出資(約20社のシェア・ホルダ)により設立された共同体組織である。設立は1983年で歴史

はまだ浅いが、現在約400名のテクニカルスタッフにより、

- ・ Semiconductor Packaging/Interconnect
- ・ Software Technology
- ・ VLSI/CAD
- ・ Parallel Processing(次世代言語開発、資源分散システム)
- ・ AI/KBS(大規模知識ベース、自然言語処理等)
- ・ Database(知識ベース、オブジェクト指向データベース)
- ・ Human Interface(マルチモード・マルチメディアの統合等)

の7つの研究プログラムが走っている。

これらは、1990年代に向けての長期プログラムであり、いいかえると、日本のICOT、ESPRIT、およびイギリスを中心とするAlvey等のライバル組織とも考えられる。

L.A. Belady氏は、上記のSoftware Technologyグループの責任者であり、Leonardo(for Low-cost Exploration Offered by Network Approach to Requirements and Design Optimization)と称する、大規模な分散コンカレント・リアルタイム・システムに向けて、グラフィック指向ソフトウェア開発環境の構築に携わっておられる。名の由来は、"レオナルド・ダ・ビンチ以来の改革"からとられたものである。これは、ソフトウェアのフルオートメーションを旨とするのではなく、むしろ、ソフトウェア設計チームに対して、最良のコンピュータ支援による環境の構築に焦点があてられている。この目的は、現在我々がすすめているプロジェクトの目的と一致するもので、今後も技術交流が行えればと思っており、今年の2月に来日するそうで、そのときは我々の環境をみてもらうことにしている。

4. バロアルトへ

同行して下さったSRAの松尾、栗原さんの助けもあって、CSCWが無事終了し、12月6日に、次のシンポジウムの開催地であるカリフォルニア州バロアルトに向かった。これで全行程の約半分を消化したことになり、時差ボケ(jet lag)もなんとか解消されたが、連日の肉食や慣れない英語による緊張で、多少疲労を感じてきた。しかし、サンフランシスコでSEA海外研修団と合流し、バロアルトへは貸し切りのミニバスで日本語

ガイド付ということもあって、一時的だが、観光気分を味わうことができた。

パロアルトは、シリコンバレーの生誕地であり、スタンフォード大学、Xerox PARC、アップルコンピュータ、サン・コーポレーション等がある、アメリカの高度技術地区のひとつである。

(1) PSDEの参加

Practical Software Development Environments のシンポジウムは、ACM SIGSOFT/SIGPLANの sponsorship により、ソフトウェア開発環境の実践的な側面から討議されたもので、今回が第2回であり、論文発表とデモを組み合わせたスタイルで行われた。

応募総数 175 編の投稿論文のうち 25 編が採用されたという狭き門のシンポジウムである。日本からの論文は 1 篇もなかったが、昨年 7 月に行われたプログラム委員会の選考で全滅したそうで、このへんの事情は、PSDE のプログラム委員である岸田さんからお聞きした。その話を聞いたときの感想であるが、海外にペーパーを応募するときには心構えがあるようで、そのへんを是非まとめて、SEAMAIL に載せていただけたら、これから海外に（英語で）ペーパーを応募しようとする人には有益ではないかと思う。

シンポジウムは 7 つのカテゴリに分類される。

- ・環境アーキテクチャ（プロジェクト・マスタ・データベース等）
- ・環境を利用した経験（プログラミング環境）
- ・環境活用のケース・スタディ（W/S の使用等）
- ・環境データの獲得
- ・プロトタイピング環境
- ・グラフィック環境/ユーザ・インタフェース
- ・先進的構造化エディタ

それらの詳細は、近々の SEAMAIL を参考にしていただくとして、各論文発表の後の質疑応答についてひとこと感想を述べてみる。

日本においては質問が少ない場合、チェアマンやコーディネータが質問し、それでもなお時間が余る光景を目にするが、アメリカでは”遠慮する”など到底ありえず、質問マイクに行列ができる。ただし、日本と同様にクエスチョンではなく、ステートメントめいたものがあるのは変わらないようだ。しかし、一見活発に見えるが、英語のよく判る人にあとで聞いてみたら、質のよい質問は

あまりないとのこと。また極端な例ではあるが、”ソフトウェア・プロセスとフットボールゲームとのアナロジー”についての質問があったが、こうした冗談めいた質問に対して、アメリカ気質だろうか冗談で答えを返すようである。

(2) PSDEでのデモ

PSDEでのデモは、同会場（ホテル）の一角で行われ、約 7~8 社から出品されていた。しかし、会場の狭さ（アメリカの巨人群にとって不釣り合い）やデモ時間の短さから、充分に見れなかったのは残念である。また、日本と違いパンフレットを出しているところは少なく、何しろ予備知識もないため、何が何だかわからないというところが正直な感想である。

共通にいえることは、例えば IDE (A.I. Wasserman によるプロトタイピング・サポートシステム RAPID/USE の拡張版) をはじめとして、ほとんどがワークステーション上で作成されたシステムで、それらの大半が SUN, Symbolics, フランスの Bull のマシンであった。日本ではまだままだの感があるが、アメリカではワークステーションが全盛であることを実感した。

そんなせいか、IBM の PC でおこなっていたブースには、余り人だかりはなく、たまたま自分がそこを通りかかった時、説明員から Just looking? と声をかけられ、つい Yes と答えてしまい、あまり役に立たない説明を受けるはめになってしまった。

5. おわりに

ほんの 2 週間の海外調査であったが（私にとって非常に長く感じたが）、コンピュータ技術の成長のエネルギーの源は、やはりアメリカであったと痛感させられた。併せて、今後大いに新しい発想（たとえ現在 Toy System といわれても）を含めて、日進月歩の技術革新に目を向けてゆく必要があるとつくづくと思い知らされた。

蛇足ながら、我々が現在行っているプロジェクト (FASET - Formal Approach to Software Environment Technology) に関してのお問い合わせは、03-503-4981 にどうぞ。

ダメ管理者の分類とその対策

阪井 誠

ソフトウェア・リサーチ・アソシエイツ

1. はじめに

SEA関西支部で行っている月例会で、8月から3回にわたって討論された内容を、少々私見をまじえながら報告します。

討論の内容は、一見ふまじめのようですが、実はかなりシビアなものです。プロジェクト管理で、よく要員の質とかがありますが、プロジェクトを一番ダメにするのは、管理者のダメによることがおおく、それをエンジニアが苦勞しながらおぎなっている、というのはいいすぎでしょうか。ともかくも、管理、マネジメントの質の低さはみのがされ勝ちのように思えます。

評価というと、どうも上から下という図式ができあがっていますが、下のものが上を評価するということがあってもいいわけです。年功序列という日本の社会体制は、そう簡単にはならず、ダメ管理者がある日突然優秀になるということはなく、そのためにも、まずダメ管理者を分類し、次にその対策を考えよう、というのが討論の主旨でした。

SEAMAILの一番最後のページのカレンダーをみますと、SEAの会員の一番おおい年齢層は、30代ということですから、この業界の歴史の浅さからすると、管理者、マネージャのかたがおおそうです。意見、反論等があればSEAMAILに投稿してください。また、研究会に出席されなかった会員のかたの、とくに東京の会員の意見を待っています。

2. ダメ管理者の分類

「分類するといっても、うちにはダメ管理者がいないから、難しいですね」という冗談をまじえながら討論は始まった。

はじめのうちは、お互いに自社の恥をさらすようで、抽象的な意見しかでていなかったが、やはりSE同士、話しが具体的になるにしたがい、実感のこもった意見が増え、非常に興味深かった。

まず、管理者を分類しようということから始めた。

- ・SE上がりの管理者

- ・プログラマ上がり

- ・SE経験者

- ・システムに素人の管理者

この分類では、ダメ管理者を定義づけるものではなく、単に自分たちの周りの管理者の分類をしたものである。大きくSE経験者とそうでないのに分かれるが、ここにあげたどの種別に属する管理者にも、ダメな管理者がいることがわかった。つまり、経歴だけでは管理能力は判断できない、ということになる。

研究会に参加していた会員が所属している会社によって、管理者の経歴が違っていたが、総じていえるのは、メーカー系はシステムに素人の管理者がおおく、ソフトウェア・ハウスにはSE上がりがおおい傾向があるようだ。

次に分類を行ったのは、管理者の行動パターンによるものである。

- ・攻撃型

- ・保守型

- ・革新型

- ・守備型

- ・保守型

- ・革新型

この分類でもダメ管理者を典型化することはできなかった。

その次におこなわれたのが、じゃあ、どおいったときに、この管理者はダメだと判断するかを議論して、それに名称をつけたらということで、できあがったのが、表1のダメ管理者の分類である。

3. ダメ管理者への対策

分類ができると、それではその対策ということになった。ダメ管理者への対策も、まず、一般論から始まり、次に具体論に入った。

まず一般論では、ピータの法則やランチェスターの法

則が話題になった。特に、ピータの法則は、会社組織では、無能と評価されるまで昇進し続けるので、会社が安定期に入ったときには、その立場では無能な人々の集まりになってしまうという法則である。これに対しては、法則が書かれていた本に述べられているように、最適な地位に留める努力が必要と考えられる。

具体的な対策ということで、分類した個々についての対策をだすということにして、かなりつこんだ討論もしたのだが、なぜか全をうめることができなかった。うめた対策方法は、表1にまとめてあるが、どなたか足りないところを補足していただけたらとおもいます。

4. 悪妻との関係

突如として悪妻との比較をしてみたいとおもう。それはこんな逸話によるものである。

ギリシャの哲学者ソクラテスの妻は、歴史に残る悪妻であるが（夫のソクラテスに問題があったという人もいるが）、ある日、結婚すべきか悩んでいる人がソクラテスに相談したとき、ソクラテスはこう答えた、「よい妻をもらおうと幸せになれるし、そうでないときは哲学者になれる」と。妻と管理者を置き換えて考えてみてください。

もうひとつ悪妻の話。モーツアルトの妻のコンスタンツエもなかなかの悪妻で通っているが、モーツアルト自身は全くそうは思っていなかった。モーツアルトに人類の宝ともいべき名曲の数々を作曲させたのは、妻の遊びと浪費癖によるものという逆説をいう人もいる。つまり、モーツアルトは、最愛の妻の浪費をまかなうために、せせせと作曲に励んで、それを苦痛と思っていなかった。愛される管理者というのがありますね。

悪妻の場合は離婚ということは考えられるが、管理者の場合は首にすることはできないし、ほかのプロジェクトにいけば、こんどはそのメンバーが困るということになって、真剣に対策を考えなくてはならないのだが、いまのところ決定的なものはないようです。

5. まとめ

討論は3回（5時間）と、けっこうな時間をかけられたが、結論めいたことをだすにはいたらなかった。

管理者の分類、対策はいろいろあるが、力（技術力、コミュニケーション力）や教養のある管理者、見本となるうしろ姿の絵になる（長島のような？）管理者である

べきだ、という本題から少々離れたあるいはあきらめもた。

しかし、仕事の内容がちがったり、会社の体質があたり、当然人の性格がさまざまであるのに、ダメな管理者の類似ということがわかったのは、大きな収穫であった。ともすると愚痴になりやすいテーマであるが、身近な問題だけに、これからも真剣に考えていかななくてはともおもう。

分類	対策
デストロイヤー型	?
猪突猛進型	おだてる
敵前逃亡型	敵の方向をごまかす
優柔不断型	?
責任回避型	提案を受け取らないと責任のかかるような提案をする
敵視型	飽きるまで提案する
夢想型	?
部下不信型	反対の提案をする
孤立無援型	?
完全主義型	あえて欠点をつくった提案をする
無為無策型	おぜん立てをしておく
情緒不安型	パターンをつかむ
自信過剰型	?
ハイブリット型 (上記組み合わせ)	?

表1. ダメ管理者の分類とその対策

ソフトウェア・シンポジウム '87

1987年6月4日(木) 5日(金) 東京農林年金会館 (東京・虎ノ門)

6/4		6/5	
9:30	オープニング		
10:00	招待講演	チュートリアル	ツールパネル1
12:30	昼休	昼休	
13:30	A1: AI指向	A3: 言語	B3: 管理・定量化
15:00	B1: 教育		
15:30	A2: ツール・環境	パネル討論	ツールパネル2
17:00	B2: アプリケーション開発		
18:00	情報交換パーティ		
20:00			

情報サービス産業協会(JISA)
ソフトウェア技術者協会(SEA)
共催

招待講演
Software Process Programming
コロラド大学 教授 Leon Osterweil

司会: 日本電子計算 野村敏次

チュートリアル
TRON
東京大学 講師 坂村 健

司会: パナファコム 熊谷 章

A1: AI指向
チェアマン: 富士ゼロックス情報システム 藤野晃延

モデル記法に基づいた事務アプリケーション開発支援システムの試作
日本電気ソフトウェア 大瀧陽悦, 伊藤充
設計仕様書の再利用の試み
日本電子計算 永井光俊
OPSS による AI システム構築法に関する一考察
花王 八木光重

A3: 言語
チェアマン: 野村コンピュータシステム 佐原 伸

Ada による通信プロトコルの実現
構造計画研究所 久木野誠, 前田美恵子, 金子清美
国際電信電話 堀内浩規, 長谷川享, 加藤聡彦
既存言語のオブジェクト指向化
日本電気ソフトウェア 大形和宏
Smalltalk インパクト
ソフトウェア・リサーチ・アソシエイツ 川又行雄

B1: 教育
チェアマン: 日本電子専門学校 河村一樹

SE導入研修はどうあるべきか 一効果的な導入研修の方法—
菱言システム 高橋基之, 服部知子, 吉川謙司, 大塚理恵, 吉野道子
K E 育成のための基本的考え方とその具体的アプローチ
CSK 熊谷正夫
学校教育における CAI への一考察
東京システム開発 杉本尚子

B3: 管理・定量化
チェアマン: 沖ソフトウェア 芝原雄二

規模見積りにおけるファンクション・ポイント法の実用化に関する一考察
日本ビジネスコンサルタント 横寺俊二, 高橋淳
あるテストツールの進化プロセス
ソフトウェア・リサーチ・アソシエイツ 石原千秋, 岸田孝一
テスト支援ツールの実践的応用 (各種エラー発生成長モデルの適応)
東芝エンジニアリング 小室豊

A2: ツール・環境
チェアマン: 構造計画研究所 新美論

事務処理システムにおける分散開発環境
野村コンピュータシステム 佐藤圭
ウィンドウを活用したプログラミング支援環境
ソフトウェア・リサーチ・アソシエイツ 栗原正利
DMC (概念による設計法) によるプログラム設計支援環境
管理工学研究所 大林正晴

パネル討論
広域ネットワーク

コーディネータ: 東京工業大学 村井純
パネラー: アスキー 深瀬弘恭
日本デジタルイクイップメント 小幡広昭
シグマシステム開発本部
他

B2: アプリケーション開発
チェアマン: 東京システム技研 浅川新一

フォールト・トレラント・システムの開発事例
三菱電機東部コンピュータ・システム 野田政博
上下水道用情報処理設備ソフトウェア生産の機械化
富士ファコム制御 亀山素和, 土井崇, 斎藤文弘
DRACO を利用したアプリケーション開発事例
協同システム開発 赤坂仁志, 野辺良一

ツールパネル

ツール展示者の皆さんによるパネル討論を計画しています。

ツール展示

下記の大学, 研究所からの商品でない先進的なツールの展示, 発表を行います。

慶応大学
大阪大学
東京工業大学
静岡大学
協同システム開発 FASET プロジェクト
他

問い合わせ先: ソフトウェア技術者協会(SEA) tel.03-312-3256

情報サービス産業協会(JISA) tel.03-436-3938

参加申込要領については裏面をご覧ください

ソフトウェア・シンポジウム'87 実行委員会

実行委員長

日本電気ソフトウェア 岡田正志

プログラム委員

パナファコム 熊谷章

日本電子計算 野村敏次

東京システム技研 浅川新一

日本ソフトウェア開発 木間教大

日本コンピューター・システム 村井進

神戸コンピューターサービス 盛田政敏

富士ゼロックス情報システム 藤野晃延

ジェーエムエーシステムズ 別役高明

野村コンピュータシステム 佐原伸

日本電子専門学校 河村一樹

協同システム開発 川崎好文

沖ソフトウェア 芝原雄二

構造計画研究所 新美論

日本電子計算 臼井義美

プログラム委員長

日本電子計算 高田佳彦

ソフトウェア・リサーチ・アソシエイツ 林 香

ツール展示実行委員長

ソフトウェア・リサーチ・アソシエイツ 岸田孝一

参加申込

申込の受付は、ソフトウェア技術者協会(SEA)事務局が担当しております。申込用紙に必要事項を記入の上、SEA事務局までお送り下さい。到着しだい参加券と請求書をお送りします。

参加費(1名)

4月20日迄に申込の場合 22,000円 (JISA 正/賛助会員会社, SEA 正/賛助会員)

25,000円 (会員外)

4月21日以降に申込の場合 25,000円 (JISA 正/賛助会員会社, SEA 正/賛助会員)

28,000円 (会員外)

尚、5月28日以降は、当日会場にてお申し込み下さい。

..... 切り取り線

ソフトウェア・シンポジウム'87 申込用紙

申込者

(氏名) _____

(会社名) _____

(部署) _____

(会社住所) 〒 _____

(TEL) _____

請求書送付先 〒 _____
(申込者と異なる場合のみ記入)

参加費

22,000円 X _____人 = _____円

25,000円 X _____人 = _____円

28,000円 X _____人 = _____円

合 計 _____人 _____円

申込先: 〒166 東京都杉並区高円寺南1-5-4 高円寺サンハイツ404

ソフトウェア技術者協会 事務局

ソフトウェア・シンポジウム係

TEL: (03) 312-3256 FAX: (03) 318-3909

SEAこれからの活動予定

-1987-

SEA会員状況

◆3月(March)

17(火)~19日(木)

春のセミナー・ウィーク(機械振興会館)

30(月)~4月2日(木)

第9回ICSE(米国 モントレー)

◆4月(April)

8日(水)

管理分科会月例会(協同システム開発)

9日(木)

第7回AI分科会月例会(機械振興会館)

15日(木)

環境分科会月例会(機械振興会館)

17日(金)

特別フォーラム:「売り上げ税を考える」

(東京:青年会議所)

27日(月)

月例フォーラム:

「ICSE&長岡ワークショップ報告」

(機械振興会館ホール)

◆5月(May)

19日(火)

月例フォーラム:「SDIをめぐる」

(機械振興会館ホール)

SEA定期総会(機械振興会館:研修1)

◆6月(Jun)

4(木)~5日(金)

ソフトウェア・シンポジウム'87

(東京:農林年金会館)

17(水)~18日(木)

第1回技術交流シンポジウム[企画中]

(浜松:静岡大学)

19(金)~20日(土)

第4回ソフトウェア信頼性シンポジウム

(大阪:ガーデンパレス)

25日(木)

月例フォーラム:テーマ未定

(機械振興会館:研修2)

昭和62年1月24日現在の入会会員状況は以下の通りです。

正会員--710(先号比38名増)

賛助会員--22社(先号比1社増)

<地域分布>

<男女分布>

勤務 居住

男 = 670

女 = 40

北海道 = 1 1

岩手 = 4 4

新潟 = 5 5

栃木 = 4 3

茨城 = 4 6

埼玉 = 10 55

千葉 = 13 53

東京 = 465 279

神奈川 = 44 139

長野 = 9 9

富山 = 1 1

石川 = 2 2

静岡 = 6 6

岐阜 = 1 3

愛知 = 8 6

滋賀 = 1 2

京都 = 4 9

大阪 = 65 50

奈良 = 1 3

兵庫 = 20 28

福岡 = 5 5

熊本 = 6 6

長崎 = 1 1

鹿児島 = 3 3

<年齢分布>

20-24 = 39

25-29 = 159

30-34 = 190

35-39 = 185

40-44 = 67

45-49 = 34

50-54 = 11

55-59 = 7

60以上 = 5

会員の名簿管理は、パソコンを使用しておこなっています。住所、会社部署等の変更があったときは、なるべく早く事務局までお知らせください。



ソフトウェア技術者協会

〒166 東京都杉並区高円寺南1-5-4 高円寺サンハイツ404
TEL.03-312-3256 FAX.03-318-3909