

SOFTECHSアーカイブ

日々技術が進化し、利用形態が変化し、新たな適用シーンが生まれてくるITの世界にあっては、過去を振り返るより前を見つめることのほうが大切でしょう。しかし、ITを生業とする企業や技術者が、今よりもっと重要な役割を果たすよう産業界を含む社会から求められている今日にあって、節目節目でかつての志や姿勢を確認しておくこともまた意義あることと思います。

そのような趣旨で、過去のSOFTECHSからの再録を企画しました。バックナンバーを読み進めていくと、時代背景こそ違え、かつて苦闘した課題や描いた夢が、今も姿形を変えて存在しているように思えてきます。誌面の制約から再録できたのはほんの数編ですが、読者の皆さんはどのような感想を持たれるでしょうか。

再録に当たっては、執筆者の所属・役職等も含め、基本的には掲載当時のものに手を加えていません。他の原稿と表記が違う場合などもありますが、そのあたりは勘案しながらお読みいただければと思います。

(SOFTECHS編集委員会)

〈巻頭論文〉

ソフトウェア産業の問題点と明日を考える



代表取締役社長 大久保 茂

我が国には、コンピュータのプログラムを作っている企業が、1,800社あるとも3,000社あるともいわれている。しかも、どの企業も仕事にこと欠くことなく多忙である。しかし、真にソフトウェア産業を形成するにふさわしい企業は余りにも少ない。

我が国に「ソフトウェア産業」は存在するのだろうか？

通産省の産業分類によれば、ソフトウェア業とは「コンピュータを利用するためのソフトウェアを開発することを”主たる事業”とする企業」と定義されている。しかし、ソフトウェアの開発を行っているのは、なにもソフトウェア会社に限ったことではない。むしろ、コンピュータ メーカー、ユーザ、計算センター等は、ソフトウェア専門家が生まれる以前から多くのソフトウェアの開発をしている。それでは、国際的に通用する定義のもとで、ソフトウェア産業が我が国に果して存在するかというと、その答えは残念ながら、NOと言わざるを得ない。

この分野の先進国である米国と比較すると、如何に我が国のソフトウェア会社の育ち方が、或いは歩いている道が異なっているかが明らかである。米国のソフトウェアハウスは例外なく、それも、政府関係機関、民間企業を問わず、ユーザを対象として事業を展開している。米国のソフトウェアハウスは、ユーザとの契約によりカスタムソフトウェアの開発プロジェクトに取り組み、ソフトウェアプロダクトを提供しつつ、独自のビジネス領域を開拓して、ソフトウェア業のプロフェッションを確立してきた。その結果として、ソフトウェア産業が形成されるに至ったことは歴史的事実である。コンピュータメーカーの仕事をしたり、メーカーを通じてユーザのプロジェクトを担当するソフトウェアハウスは皆無である。米国では、ソフトウェアハウスがメーカーに対して独立・中立であることが常識となっている。いわば独自の産業として、Software Industryという呼称が生まれたのである。

一方、我が国の現状はどうか。数多いソフトウェア会社のうちの80～90%が、資本関係は別としてもメーカーの傘下にそれぞれ系列化されている。メーカーに技術者を派遣する、メーカーの開発プロジェクトを引き受ける、メーカーを通じてユーザに技術

者を派遣する、メーカーを通じてユーザのプロジェクトを受注する等々、ほとんどがメーカーに依存して経営を成り立たせている。あえて、欧米流の考え方で産業分類をすれば、これ等はSoftware Industryに属するのではなくComputer Industryに分類されることになる。これが日本のソフトウェア業の特異体質である、自由経済社会に於ては、それでも良いではないか、という人達が意外と多い。私はビジネスとして、そのような道を歩くことを批判するつもりはない。どの道を歩くことも各企業の自由である。ただ、産業分類上そのような企業は電子計算機産業に分類されるのが正しいと思う。ソフトウェア・コンベンション'78に於て、当時の通産省情報処理振興課課長であった西川禎一氏もこの点について付言し、「我が国のソフトウェア産業は、まだひとつの産業としての基礎ができていないと私は断定せざるを得ない」と極めて強い調子で批判し、更にメーカーの下請からの脱皮を要請された。通産省の課長としては至極もつともな発言で、行政指導、或いは補助金の交付等による業界の育成にしても、業界の大多数がメーカーの傘下にあるなら、通産省の担当する課は電子機器課のはずである。情報処理振興課が担当するのは情報処理サービス業とソフトウェア業である。従って、現在のソフト協の80%を超える会員がメーカー系であるとするれば、正しい産業分類からするとソフトウェア産業は存在しないのだから、メーカーと同様に電子機器課の育成指導を受けるのが正しい、と私は思う。シンクタンクやコンサルタント会社は下請による仕事は一切しない。建築設計会社も建設会社の下請で設計をすることは無い。ソフトウェア会社がプロフェッションの確立と、独自の産業を形成することを目指すなら、実際に成果物を使用するユーザとの直接契約に基づき開発プロジェクトに取り組むのが筋である。そうでなければ、成果物に対する直接の責任を負うことはできないし、発注者であるユーザの正しい評価も得られない。ソフトウェアハウスはアSEMBL工業に対する部品供給の町工場ではない。

何故、我が国のソフトウェア業界のみがコンピュータメーカーに依存する体質になってしまったのだろうか。私にはその理由が理解できない。我が国のソフトウェア会社の経営者も、幾多の機会に欧米のソフトウェアハウスを訪問しているはずであ

り、国際会議にもいくたびか参加している。技術レベルの向上には目を向けても、産業の正しい位置づけには目をつむるのは何故だろうか。ユーザ市場の開拓を怠り、猫の手も借りたいメーカの傘下に集り、潤沢に仕事を流してもらおうという安易な経営にながれた結果ではないだろうか。ソフトウェア産業に、優秀な学生の応募がなくて困るということが協会でも問題となっている。社員の定着率も問題となっているところが多い。私は当然のことだと思う。一流大学の学生は、メーカの下請会社に応募するよりはメーカに入社を希望するのが当然である。優れた人材を集め、そして定着させるには、興味あるプロジェクトに責任ある立場で参加し、成果に対して正しい評価をうけられることが大切である。ユーザに提出するプロポーザルに、成果物であるドキュメンテーションに、プロジェクト チームのマネジャー、主任設計者以下全員の氏名を記し、責任を明らかにする必要がある。責任を明らかにすることにより、成果に対する正しい評価も得られる。

つぎに重要なことは、社員の待遇である。未だ新しい産業であり、特に形の無い商品やサービスを提供する頭脳集約型企業に対する価値は、近年、我が国でも高まりつつあるとはいえ、欧米に比べて未だ低い。したがって、我が国の業界では社員の頭数で勝負をし、質を犠牲にし、定着率の低さによる社員の平均年齢の若さを利点としている企業も数多くある。しかし、真にユーザが求めているのは、頭数による量ではなく、質である。質の高い成果物を提供することにより価値を高めることに努力し、メーカに比べ若干でも社員の給与水準を高めなければ、優れた人材は集まらないし、高い定着率は望むべくもない。ユーザが求める水準を超える、優れた技術者が育つ土壌をこそ、1日も早く、真摯な姿勢で確立する努力がなされなくてはなるまい。

ユーザ ニーズに適應できる

ソフトウェア ハウスを志向するためには

先ず、ソフトウェア ハウスにとって、ソフトウェアエンジニアリングに重点をおくか、システム エンジニアリングを優先させるかを論じる必要がある。この数年の間にユーザのソフトウェア会社の利用のしかたが顕著に変化している。最近、縦割りによる開発プロジェクトが基本設計から一括して発注されるケースが多い。かつては、2千～3千万円程度のプログラム制作フェーズのプロジェクトがほとんどであったのが、最近では2～3億円のプロジェクトが多く、なかには5～6億円のプロジェクトがフィジビリティ スタディから一括して発注されるケースもまれではない。この場合、ユーザが外注にふみきるのに2つの要因がある。先ず考えられるのは、自社内開発の体制が量的に不備である場合。システム エンジニア、プログラマの絶対数が不足で期限内にプロジェクトを完成させるに必要なマンパワーのやりくりがつかない場合である。しかも、外注先が

特にシステム設計フェーズに於て、類似システムの開発に十分な知識と経験をもってしていると判断した場合である。

第二の要因としては、自社内開発の余裕はあるが、自社のEDP部門に、開発を予定するプロジェクトについて経験がない場合で、発注先に予定をしているソフトウェア会社が、同一の問題について十分な技術と経験を有し、問題解決(Problem Solving)能力があり、エンド ユーザとの話し合いも含め、一括したプロジェクトとして委託した方がよりベターな成果が期待できると考えられる場合である。

いずれにしても、ユーザが自社内で設計フェーズを完了し、プログラム仕様書を書きあげて、プログラミング、コーディングのみを外注するケースは減少してきた。何故なら、システムを開発し、インプレメントするまでの質的・量的なワーク ロードのうち、プログラム制作は30%にも満たないからである。

ソフトウェア ハウスの システム エンジニアとプログラマ

ソフトウェア業界に於けるシステム エンジニアの定義については問題がある。再三実施されたアンケート調査の回答によれば、200名の技術社員のうちシステム エンジニアが120名と答えているケースが多い。しかし、その多くはシステム エンジニアとプログラム設計者とを混同しているのである。ソフトウェア技術者の限界は30才と言った人が過去にいたが、それはプログラマ コーダーのことではないかと思う。アプリケーション システムの開発過程に於けるシステム分析や設計には、事柄についての知識をもってすることが不可欠である。そうでなければ、エンド ユーザとの会話は通じないし、設計に必要な諸条件を聞き出すことはできない。システム エンジニアが事柄に精通するためには、専門分野について事前に、しかも十分勉強することと経験を重ねることが必要である。システム アナリスト、システム エンジニアがコンサルテーション能力を持ち、ユーザ ニーズを正しく把握し、ニーズに合った効率の良いシステムを設計するためには、やわらかい頭脳、十分な事前勉強および類似プロジェクト担当の経験を必要とするわけで、一般的には30数才になってようやく一人前になるといっても過言ではない。コンサルタントになるには、単なる基礎知識だけでは不十分である。同様に、システム エンジニアも、類似したプロジェクトをいかに数多くこなしたか、という経験がものをいう。ソフトウェア エンジニアリングもまた、ソフトウェア ハウスにとって重要な分野であることは疑問の余地はない。ソフトウェア ハウスがユーザを対象として、カスタム ソフトウェアの開発を、システムのインプレメンテーションまで一括したプロジェクトとして受託し、プログラム群を最終成果物として提供するためには、ソフトウェア エンジニアリングは重要な研究課題である。何故ならば、ソフトウェア ハウスは、システムのインプレメンテーションまで責任をもつからである。さらには、ソフ

ソフトウェア製品の利用はもとより、あらゆる言語を使い分け、ユーザー ベーシックや各種ツールの開発を必要に応じて行うからである。

ソフトウェア製品の開発・輸入・販売 —ソフトウェア製品時代を迎えて—

米国では、1980年度に於て、ソフトウェア製品の市場がカスタムソフトウェア開発を超えたという。ここ数年来のハードウェアの進歩は、まさに驚異的といえる。プライス オブ パフォーマンスは飛躍的に高まった。しかし、高性能超大型コンピュータ システムを効率よく活用するには、アプリケーションソフトウェアの開発と保守に莫大な投資と経費を必要とする。これはユーザーにとって大きな負担となり、極めて頭の痛い問題である。必然的に、すべてのプログラムを従来言語による手作りから、システムソフトウェアの利用へと、さらに将来は、アプリケーションソフトウェアの採用が論じられる時代となってきた。我が国の市場でも、ニーズに合うソフトウェア製品の

の採用を検討するユーザーが、ようやく急速にふえつつあり、近い将来にはソフトウェア製品市場は活況を呈することが予想される。米国では、ソフトウェア製品市場がすでに成熟期に入り、IBMや大手ソフトウェアハウスが自社開発にとどまらず、小規模ながらも優秀なソフトウェアハウスが開発したユニークなソフトウェアを発掘し、販売権を獲得して商品化を図り、強大な販売ネットワークによって量販をする時代に入ったという。ソフトウェア製品の良否は、ユーザー ニーズにどのようにマッチするか、パフォーマンス エフィシエンシー、サポートとメンテナンスはどうなのかによってきめられる。米国では、ユニークな発想と質的に優れた製品が、必ずしもメーカや大手ソフトウェアハウスによってのみ生み出されるとは限らないそうである。まして、ユーザー不在でベストセラーが生まれるはずはない。こうした米国のソフトウェア製品市場の傾向を我が国のソフトウェアハウスがどのようにとらえるか、1980年代後半へ向う我が国ソフトウェアハウスの成否の鍵は、どうやらこの辺りのとらえ方いかんにかかっているように私には思えてならない。

〈特集〉

CACにおける海外プロジェクトの実績と課題

1 基本的な取組み方

取締役技術本部長 菊池 毅

国際化時代を迎えて、日本の情報処理産業界も国際会議を日本で開催するだけの余裕を持つようになった。

しかし、本当に産業としての実力を身につけたかどうかは疑わしい。特にソフトウェア業界の実態を知る者は、欧米諸国の同業と比べて極めて労働集約的であり、その経営基盤がもろいことを痛感している。

ソフトウェアハウスの経営基盤は、その持てる技術力の強さによってある程度推し測ることができる。当社は、技術力強化のため、積極的に海外との交流を進めてきた。海外プロジェクトは、市場の拡大に寄与するだけでなく、社内の技術者のスケールアップの場としても、大きな意味をもっている。海外プロジェクトの背景、国内との差異および当社の取組み方を紹介し、今後の展開に役立てたい。

1. 海外プロジェクトの背景

ここでいう「海外プロジェクト」とは、外国にある事業所のために、ソフトウェアを開発すること、現地にインストールしユーザーに引渡すこと、ユーザーを教育することおよび保守のため現地に駐在することなどを指す。要するに、納入先が外国であることおよびユーザーの主体が外国人であることの2つを満たすプロジェクトをいう。

自由世界第2位の経済大国となった日本への外圧は極めて強い。先進国や発展途上国からは市場開放や技術移転を、低所得国からは経済援助を要求されている。理の通った要求には、積極的に対応してゆくことが望ましい。我々の海外プロジェクトは、技術移転の変型と見なすことができ、国際社会に日本の立場を理解させる一助ともなる。

昭和50年以降、プラント輸出が盛んになるにつれ、プロセス制御のためのコンピュータ システム(以下プロコンという)も輸出されるようになってきた。プラントの大半は、石油、石油化学および鉄鋼関連であり、国内で培ったプロコンのソフトウェア開発技術を持つ当社は、幸いにもかなりの規模のソフトウェア開発を一括で請負う機会に恵まれたわけである。

一方、昭和56年頃より金融の国際化が始まり、海外支店や現地法人の業務の機械化を進める動きが出て来る。ここでも

長年の金融業務システム開発の技術蓄積がものをいい、ユーザーニーズにマッチしたソフトウェアを提供して来た。また昨今、国際ネットワークの拡がりにつれて、国内の金融システムも海外を意識せざるを得なくなり、経済の世界は一体化が進みつつあるといっても過言ではない。

この他、製造業に関しても、失業の輸出と非難されている製品輸出に代えて、現地生産が盛んになり、現地工場の業務の機械化のニーズも幅広く出ている。業務のシステム化には、金融も含めて現地の法律、習慣および物の考え方のちがいを十分認識する必要がある。また、いわゆるカントリーリスクを心得ておかないと、経営基盤の強化どころではなくなることを、肝に命じたい。

2. 海外マーケット

当社の主力業務であるアプリケーション ソフトウェアの開発という観点から海外を眺めると、ターゲットとすべき市場はおのずから定まって来る。

かつて、昭和50年～51年にかけて、市場調査をかねて東欧諸国からの引合いを数多くフォローした事があったが、そのいずれもが当時の我々の技術力のみでは対応できない代物であった。たとえば、生産管理用端末のライセンス契約、パイプライン コントロール システムと楽器のパーター、中近東向製油所システムのジョイント ベンチャーなどである。これらの案件に対応するには、商社機能および総合エンジニアリング技術を持つ企業でなければならない。当時の企業としての実力を冷静に判断すれば、単独で東欧市場を開拓することは不可能に近い事であった。

10年後の今日、事情は変わったであろうか。変わってないとも言えるし、大いに変わったとも言える。海外市場を単独で開拓するには、相当の無理がある点では変わっていない。しかし、アプリケーション システムの構築技術に関しては、かなりの国際競争力(コスト、納期、品質)を持つに至ったと考えてよい。すなわち、他の企業と組んで国際入札に参加しても、十分に戦える力がついた点では大いに変わったのである。

さて、我々の市場は、これを地域別および適用業務別に分

けて考えることができる。地域といっても地理上の地域ではなく、経済面からの分類で先進国、中進国、発展途上国および産油国である。共産国は一応、中進国と見なしてよいであろう。適用業務は特に限定する必要はないが、やはり国内で多くの経験を積んだアプリケーションが基本となろう。

先進国におけるニーズは、日本企業の海外支店、現地法人または合弁会社のシステム構築が多い。さらに、これらの企業はワールドワイドな通信ネットワークで結ばれるので、国際ネットワーク構築技術が要求される。当面は、日本企業に関連したアプリケーションソフトの開発が続くにしても、いずれは国情に合わせたアプリケーションパッケージの輸出が実現するであろう。

中進国はそのほとんどが、先進国の優れた生産技術の移転を望んでいる。日本が欧米から先進技術を導入し、今日の地位を築いたのは周知の事実であり、後を追う彼等にとっては当然の要請である。しかし、鉄鋼の例を見るまでもなく、ブーメラン現象を恐れて、現在の最先端技術を簡単に開示する日本企業は少ない。一方、現代の生産技術は、コンピュータの高度利用を前提としており、導入後も日々改善・改良していかざるを得ない性質のものである。したがって、彼等が技術を手に入れたとたん、我国と同じ立場に立てるわけのものでもない。こう考えると、中進国への技術移転は、必ずしも強大な競争相手を育成することにはならず、むしろ好ましい国際分業を助長することになる。中進国におけるコンピュータ利用技術の普及は、我々のもつ技術を提供する機会が広がることにつながる。

3番目のカテゴリーである発展途上国に対して、我々は何をなしうるのであろうか。一般論としていえることは、技術協力の一環としてのコンピュータ教育であろうか。南北問題と総称される途上国のかかえる諸問題は、あまりにも複雑で、情報格差ひとつとっても簡単に解決できそうにない。今後の研究課題である。

最後に産油国であるが、多少テンポは落ちたものの、いぜんとしてプラント建設は盛んであり、単なる原油輸出国から製品輸出国になる日も近い。概してコンピューターリゼーションに熱心であり、労働人口も少ないため、外国技術者への依存度が高い。コンピュータの利用は、生産プロセスだけでなく、いわゆる企業経営にも急速に広がっている。魅力あるマーケットである。

以上、大まかに海外マーケットを展望したが、どの国でも自国の産業を保護・育成する政策を持っており、ソフトウェア産業も例外ではない。一昨年、ジャカルタのソフトウェアハウスを訪問する機会を得たが、当地では仕事の大部分が官公庁であり、発注は地元企業優先で外国企業が応札する場合は、地元企業とジョイントすることが条件になっているとのことであった。

今後は、単品のプロジェクト受注だけでなく、その国の同業

者との提携により、国情に合ったアプリケーションソフトウェアを提供して行く方向で、海外マーケットを検討する必要があるだろう。

3. 当社の取組み方

海外プロジェクトを、ビジネスとして成功させるためには、日頃から前記のマーケットを念頭においてそれなりの準備をしておく必要がある。当社では、昭和58年を初年度とする五カ年事業計画を遂行中であるが、計画の一環として、海外向の仕事の拡大するために、次の四項目に重点をおいて体制を整備しつつある。

- (1) 海外専門SEのグループ化と若手技術者の育成
- (2) 英文ドキュメンテーションの機械化
- (3) 語学教育の強化
- (4) 開発および保守のための環境整備

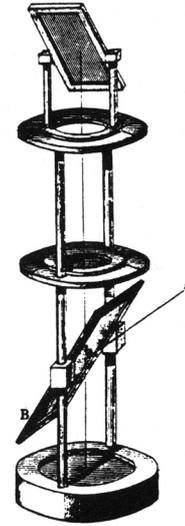
海外プロジェクトの成否は、仕様確定までのテクニカルネゴにある。短期間の現地折衝で仕様を固めるには、かなりの手腕を必要とする。経験10年以上の優秀なプロジェクトマネージャーの下に、海外派遣を経験したSEを集め、特定の業界の業務をOJTを通して勉強させる。つまりアプリケーションごとに特化したSEを養成し、海外経験を継続して積ませることにより、テクニカルネゴのできる人材を育成していく。

英文ドキュメントを日本語から翻訳して作成することは、コストおよび納期の点で問題がある。初期の頃は、翻訳専門家に外注したが、特定の業務およびコンピュータシステムの両方に堪能な翻訳者を得ることは難しく、誤訳に悩まされて止めてしまった。現在では、標準化を徹底させSEがはじめから英文でドキュメントを作成する方式に改めている。またワードプロセッサが普及してきたので、英文ワープロを設置しパターン化および専任オペレータ制により、比較的低コストで完成図書を作れるようになった。今後はワープロの台数を増やし、技術者自身が操作する方式にもって行く。仕事量が増大すると、欧米人の雇用も考えねばならない。

業務で海外へ出張する技術者には、事前に最低50時間の英会話レッスンを受ける事を奨励している。即効性は望めないが、いわゆるコンプレックスをとり除く効果はある。語学教育は、英会話だけに限らないが当面は英語に限定し、英文仕様書の作成方法と英会話の教育に重点をおく。欧米諸国では、長期出張には家族帯同が常識である。当社に来るエンドユーザーの技術者も例外ではなく、伴侶をつれてくる。英語に堪能な場合は、頼みこんでグループレッスンを社内で実施するケースもあり、ビジネスを離れて相互理解に役立つ。この他、一般研修の一科目として希望する者はいつでも外部の語学スクールへかよふことができるコースがあり、語学力の全体としての底上げをはかっている。

海外プロジェクトのターゲットマシンは、国産機ないし、IBMかDECが多い。国産機を輸出する場合は開発時に問題はないが、外国機では、再輸出よりも現地購入が多く、開発用にマシンを臨時に調達しなければならない。当社ではIBM4341およびVAX780を保有しており、マシン調達で苦勞することはない。開発支援ソフトウェアの整備も、生産技術研究室の手で着々と進んでおり、さらに本年度からは、保守技術の研究に着手する。近い将来、海外向リモートメンテナンス体制も可能となろう。

いずれにしろ、海外プロジェクトは総合力の勝負であり、安易に取り組むべきではない。これからも地道に実績を積み重ねていきたい。



〈社内研究会レポート〉

保守業務用エキスパート・システム研究会

2 ニューラル・ネット による工数見積

開発第一部 技術員 小林 清二

概要

保守プロジェクト用の作業計画立案支援システムのサブシステムとして、人工知能(AI)技術の1つであるニューロ技術を応用した作業工数の見積りシステムを開発した。

本システムは、対象プログラムのステップ数と、生産性に対する変動要因のパターンを入力すると、プログラムのモジュール設計およびプログラミングの作業工数を見積もる。システムには、ニューラル・ネットの学習アルゴリズムとしてバック・プロパゲーションを採用。生産性変動要因のパターンと、それに対する生産性のカテゴリを学習させることによって、作業工数を計算する。

1. はじめに

一般に、保守プロジェクトでは、限られた要員によって大小さまざまな案件を随時処理しながら、トラブルの発生に緊急に対処していく必要がある。保守プロジェクトのマネージャーやリーダーは、対象となるシステムを熟知し、メンバーや顧客の状況を的確に判断しながら、作業計画を作成し、また随時、計画を調整していく必要がある。

このように、プロジェクトの作業計画の作成と調整は、担当者の経験と技術を必要とする作業といえる。このような作業に対しては、エキスパート・システムまたは人工知能(AI)技術の応用による支援が望まれるところであり、また、計画の客観的な最適化と評価のためには、システム工学的アプローチも必要であると思われる。

そこで、当研究会の活動として、保守プロジェクト用の作業計画立案支援システム(会話型)の構築をめざした。ただし、工数および作業環境の制約があるため、今回は、そのサブシステムとして、モジュール設計とプログラミングの作業工数を見積もるシステムだけを開発した。開発にはVAX8600とPC-9801を使用し、プログラムはC言語で記述した。本システムは、AI技術の1つであるニューロ技術を応用しているところに特徴がある。

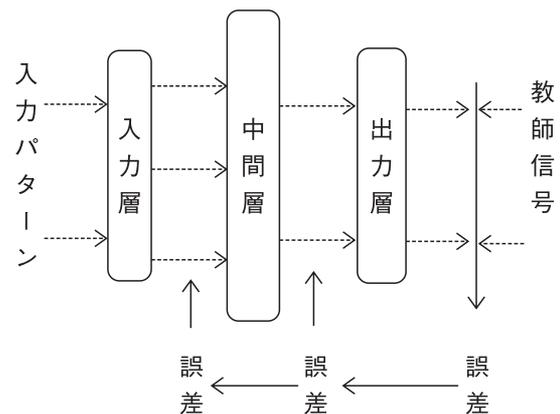
2. バックプロパゲーションについて

本システムは、ニューロ技術のPDP(Parallel Distributed Processing)モデルの代表的アルゴリズムであるバック・プロパゲーション(Back Propagation)を採用しているので、ここで簡単に解説する。

2.1 概説

バック・プロパゲーションのネットワークは、入力層、1つまたは複数の中間層、および出力層の3種類の層から構成される(図1)、さらに、各層は1つまたは複数のユニット(形式ニューロン)から構成される。各層のユニットは隣の層のすべてのユニットと結合し、この結合の強さを変えることによって学習が行われる。結合の強さは、〈結合荷重〉と〈しきい値〉で表わされる。入力層へ入力されたデータは、中間層、出力層へと前向きに伝搬される。そして、出力層からの出力値と入力層に対応した教師信号との誤差の2乗和が最小になるように、出力層から中間層、入力層へと、各層間の結合の強さを修正する。

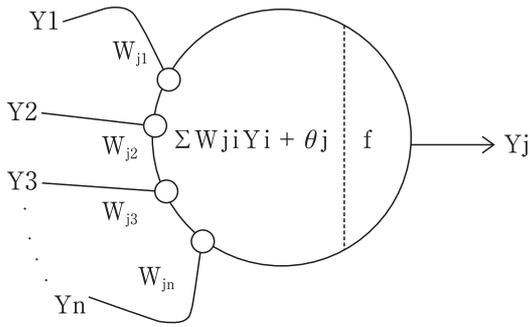
図1 バックプロパゲーションのネット構成



2.2 学習アルゴリズム

ユニットは実際のニューロンを模して、その働きを数学的に単純化している(図2)。

図2 ユニットj



すなわち、ユニットjへの入力 X_j は、他のユニットからの出力を Y_i 、それらとの結合荷重を W_{ji} 、ユニットjのしきい値を θ_j とすると、

$$X_j = \sum_i Y_i \cdot W_{ji} + \theta_j \quad (2.1)$$

と表わされ、さらに、ユニットjの出力 Y_j は応答関数 f によって、

$$Y_j = f(X_j) \quad (2.2)$$

と表わされる。関数 f として、システムでは次式のようなシグモイド (Sigmoid) 関数とよばれる関数を用いた。

$$f(x) = 1 / \{1 + \exp(-\mu x)\} \quad (2.3)$$

ここで、 μ は定数である、シグモイド関数は、出力範囲 $[0, 1]$ をとる非線形の増加関数である。

本システムでは、基本的に、結合荷重 $\{W_{ji}\}$ としきい値 $\{\theta_j\}$ の修正を次のような手順で行った。

- ① 入力層への入力パターンに対し、各層のユニットの変換によって、出力層から出力 $\{Y_k\}$ が求められる。
- ② ①で求めた出力 $\{Y_k\}$ と入力パターンに対する教師信号 $\{T_k\}$ とから、次式によって出力層から中間層への学習信号 $\{\delta_k\}$ を求める。

$$\delta_k = (T_k - Y_k) f'(X_k) \quad (2.4)$$

ここで、 X_k は式(2.1)によって求められる出力層のユニット k の値である。

- ③ 中間層間および中間層から入力層への学習信号 δ_i は次式によって求める。

$$\delta_i = f'(X_i) \sum_j \delta_j W_{ji} \quad (2.5)$$

ここで、 X_i は式(2.1)によって求められる各層のユニット i の値である。

- ④ 式(2.4)と式(2.5)で求めた δ から、結合荷重としきい値のそれぞれの修正量 ΔW と $\Delta \theta$ を、次式により求める。

$$\Delta W_{ji}(t) = \eta (\delta_j Y_i) + \alpha \Delta W_{ji}(t-1) \quad (2.6)$$

$$\Delta \theta_j(t) = \beta \delta_j \quad (2.7)$$

ここで、 η 、 α 、および β は定数。また、 (t) は今回を、 $(t-1)$ は前回は表わす。

- ⑤ 次の学習パターンがあるとき①～④を繰り返す。
- ⑥ 全パターンの学習がひと通り終了したとき、各パターンの誤差の絶対値の累計が、あらかじめ定めた限界誤差 E_0 以下ならば学習を終了する。

2.3 学習アルゴリズムの評価

アルゴリズムの性能評価のために、排他的論理和の4パターンを学習を行った。ネットは3層で、入力層、中間層、および出力層のユニット数はそれぞれ2、2、1とした。結合荷重としきい値は、乱数によりそれぞれ $[-0.2, 0.2]$ と $[-0.1, 0.1]$ の範囲で初期化し、限界誤差 E_0 は0.1とした。各定数 μ 、 η 、 α 、および β を恣意的に変えて学習回数を調べたところ、だいたい300回から3,000回で学習を終了した。学習は初期値に大きく依存し、初期値によっては学習が終了しないことがあることはわかったが、実験数が少ないためか、各定数の値に対し文献に示されるような顕著な傾向はみられなかった。

バック・プロパゲーションを現実問題に適用する場合、中間層の数とそのユニット数、各定数の値、アルゴリズム自体など理論的に決定できないものが多く、また、入出力をどのように数値化して与えるかなど問題の表わし方も難しく、適用には担当者の十分な経験が必要であるといえる。

3. システムの解説と評価

3.1 システムの概要

作業工数の見積りは、単に、各フェーズではほぼ決まっている生産性とステップ数の積で求められることが多い。しかし、生産性は作業担当者や対象によってかなり異なる。そこで、モジュール設計とプログラミングにおける生産性の変動要因とそのパターンに対する生産性の代表値(ステップ数/人月)をネットに学習させることにした。

生産性変動要因として、今回は、表1に示すような項目を選んだ。入力層の1つのユニットに対して1つの項目の情報を与えることにした。入力層の各ユニットへは、入力値として、生産性を向上させるとき1を、生産性を低下させるとき0を与える。

出力層の各ユニットは、それぞれ生産性の代表値のカテゴリを示す。各入力パターンに対し、生産性の代表値カテゴリを選び、教師信号として、選んだカテゴリに対するユニットへ1を与え、他のユニットへは0を与えることでネットの学習を行う。

システムは、学習した生産性変動要因のパターンに対して生産性の代表値カテゴリを示す。そして、システムに入力したプログラムのステップ数をその代表値で除することによって工

表1 モジュール設計とプログラミングにおける生産性変動要因

項目	入力値	状態	内容	対応ユニットの番号
難易度	1	低	シンプルで易しい。前工程の完成度もよい。	0
	0	高	複雑で難しい。前工程の完成度が悪い。	
経験年数	1	高	あらゆる面で経験し不安材料も少ない。	1
	0	低	システム全体の理解度が低く経験が少ない。	
類似性	1	あり	類似性のある前例がある。	2
	0	なし	まったく新規である。	
前工程からの継続作業か	1	YES	前工程に引き続き作業する。	3
	0	NO	前工程の作業をしない。	
品質要求	1	中	特に明示された要求がない。	4
	0	高	厳しい品質が要求される。	

数を入力する。

なお、学習パターンの更新は、そのつどネットを初期化して、再度、全パターンの学習を行うことになる。

3.2 実行例

表2に入力データの1例を示す。この例は、筆者の属するプロジェクトの技師2名に生産性変動要因の各パターンに対する生産性を回答していただき、これを基に筆者が作成した。この例の対象フェーズは、プログラミングである。当然、他のプロジェクトやフェーズでは、異なるデータを作成することになる。

ネットの構成は、入力層、中間層、および出力層の3層で、それぞれのユニット数は5、9、および9である(図3)。結合荷重としきい値の初期値の範囲は、それぞれ[-0.2, 0.2]と[-0.1, 0.1]とし、各定数 μ 、 η 、 α 、および β は、それぞれ4、0.5、0.9、および0.3とした。

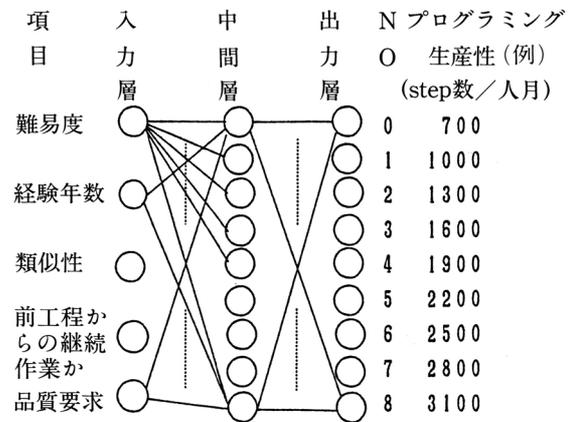
このような条件下で学習を行ったところ、限界誤差 E_0 を2.0としたとき、学習は451回で終了した。実行時間は約3分であった(さらに、中間層のユニット数を5にして実行したとき、2,154回で学習を終了した)。

表2 入力データ(例)

パターン	入力値 01234	代表値カテゴリ (step数/人月)	代表値カテゴリの番号
1	00000	1000	1
2	10000	1300	2
3	01000	1900	4
4	00100	1300	2
5	00010	1300	2
6	00001	1600	3
7	11111	2800	7
9	01111	2200	5
10	10111	1900	4
11	11011	2500	6
12	11101	2500	6
13	11110	2500	6

注:表中項目欄の01234はそれぞれ入力層のユニットの番号を指す。

図3 ネットの構成(例)



さて、この例のように、生産性変動要因の全パターンを学習していないときは、未学習パターンの扱いが問題となる。そこで、表3に示すような未学習データをシステムに入力した。その結果を表4に示す。ここで、出力層のユニットkの出力層を Y_k 、各ユニットに対するカテゴリの代表値を P_k とすると、次式により生産性 P^* が得られる。

$$P^* = \sum Y_k P_k / \sum Y_k \quad (3.1)$$

表4には各パターンに対する生産性の計算結果も示してある。

実験数が少なく、また、理論的解析もしていないので、未学習パターンに対する生産性の値を式(3.1)によって求めてよいか、現段階では判断しかねる。

表3 未学習パターン

パターン	入力値				
	0	1	2	3	4
14	1	0	0	1	0
15	1	0	1	0	1
16	1	0	0	0	1

注:表中の01234はそれぞれ入力層のユニットの番号を指す。

表4 未学習パターンに対する出力結果と生産性P*

パターン	出力層ユニットの値								生産性P* (step数/人月)	
	0	1	2	3	4	5	6	7		8
14	0	0	0.72	0	0	0.34	0	0	0	1589
15	0	0	0.14	0.02	0.82	0	0	0	0	1808
16	0	0	0.04	0.48	0.15	0	0	0	0	1649

注:出力層のユニットの値は少数点第3位以下四捨五入とする。

生産性P*は少数点以下四捨五入とする。

表中の1~8はそれぞれ代表値カテゴリ(出力層ユニット)の番号を指す。

3.3 評価と今後の課題

本システムと同様のシステムを、ニューラル・ネットではなく、通常のプログラムで実現しようとした場合を考えてみよう。

統計学的手法(多変量解析)によって変動要因と生産性の関係を表わすことができるが、そのためには正規化された相当数のデータの収集が必要となる。また、その関係を十分表わしきれない可能性がある(問題が非線形である場合など)。したがって、システム化は相当困難な作業になるはずであり、本システムの優位性が認められる。

本システムは、現段階でも、独立したシステムとして使用できるが、さらに改良を重ねる必要がある。今後のおもな検討課題を下記に示す。

- ① 入力データの作成方法
- ② 生産性変動要因の項目数
- ③ 入力値のアナログ化
- ④ ステップ数算出のシステム化
- ⑤ 未学習データの扱い

今後、できればこれらの課題に取り組みたい。

4. おわりに

現在、ニューロは、非常に多くの応用事例が発表されており、第2次ブームをむかえているようだ。また、多くの分野で応用が期待されており、世界モデルの構築なども検討されている。

工数の見積りにニューロ技術を採用した動機は、その開発の簡単さと話題性を狙ったという面が多分にあるが、ニューロ技術の応用の広さと将来性には大きな注目をよせるべきである。また今後、我々の仕事にも大きくかかわってくるであろう。

参考文献

- 1) 中野馨ほか『ニューロコンピュータ』技術評論社(1989)
- 2) 麻生英樹『ニューラルネットワーク情報処理』産業図書(1988)
- 3) 宮沢文夫『実践!ニューラルネット』ASCII, vol. 12, # 11, pp. 233-240(1988)
- 4) 『プロジェクト見積基準(暫定版)』(株)コンピュータアプリケーションズ内部資料(1986)

〈技術レポート〉

これからのシステム保守について

国際金融システム部 主任技師 嶋田 圭吾

1. はじめに

第3次オンラインを始めとする大手企業の大規模開発が一巡した上、経営環境の先行きが不透明な現状を考えると、大規模開発が今後激減することは容易に想像ができる。最近の急激なダウンサイジング現象なども、経営者がシステム投資の効率化を真剣に考えだした結果と捉えることもできよう。さらに、現在保有している膨大なソフトウェア資産を、より効率的に活用しようとするれば、今後脚光を浴びてくる分野がシステム保守サービスであることは疑う余地もない。

しかし、最近の保守のやり方を見ても、10年前と大差なく、理想的なシステム保守サービスと言うには、まだまだ程遠いのが現状である。

そこで、ここでは保守の現状を踏まえた上で“理想とするシステム保守”とは何かという点と、保守に関して現在活動中である社内委員会「新保守21」や社外研究会「SMSG*」の状況報告などを交えて述べていきたい。

2. 保守の抱える問題について

現状の保守について考えたとき、何の問題もないと思っている人は皆無であろう。それどころか、昔から何一つ改善されていないというのが現実なのではないだろうか。そこで、代表的な保守の問題について、以下に私の考えを述べる。

2.1 やりがい

保守は開発に比べて“やりがい”がないとよく言われる。確かにシステムの開発段階から参加している人にとって、保守の仕事に新鮮味を感じないのは事実であろう。しかし、保守から参加した人が、やりがいがないと言えるほど保守の仕事は簡単なものではないと思う。他人の作成したシステムに触れて、そこから設計思想を学びとるという行為の中には、非常に高度な技術的要素が含まれているからだ。実は、保守にやりがいがないのではなく、やりがいを発見できない人が多いだ

けではないだろうか。

2.2 ローテーション

保守と言うと必ず話題に上るのがローテーション問題である。現在の保守技術では仕事が属人的にならざるを得ないため、アサイン期間がどうしても長期化してしまうという大きな問題がある。この問題を考えるとき非常に難しいのは、長期アサインの是非について、まだ結論が出ていないという点にある。それは、ある特定の人を長期アサインすることによって長期・安定的な受注が得られるという現実があるからではないだろうか。ただ私は、全員を長期アサインする必要はないと思う。キャリア・パス等により計画的にローテーションする人と、長期的にアサインする人とを、始めから明確に分離して要員計画を立てていけばよいと思っている。ただし、長期アサイン者がローテーションを切実に希望した場合に、1年位のうちには実現できるような体制を常に考慮しておく必要はあるだろう。

2.3 評価

3番目に評価の問題について取り上げてみたいと思う。開発の第一線で活躍していた優秀なSEを、ある日から保守プロジェクトにアサインした場合、その人が一人前の保守SEになるためには、どの位の時間が必要であろうか。システム規模にもよるが最低でも3~6ヶ月程度はかかるであろう。さらに、そのプロジェクトのリーダー・クラスになるためには、1年は必要であろう。問題となるのは、その期間、その人をどのように評価すべきかという点である。おそらく、実績ではなく理解速度に応じて評価を与えるような仕組みが必要となるであろう。

また逆のケースで、アサイン期間が長く、生産性が普通の人の10倍位ある場合、その人に対する評価はどのようにしたら良いのだろうか。やはり、多少割り引いて評価すべきなのであろう。

これらのことから、保守の評価制度は開発とは独立して設定しておくべきと考える。

* SRA他32社が参加しているソフトウェア・メンテナンス研究会(Software Maintenance Study Group)の略称。

2.4 技術

保守プロジェクトにいと新しい技術が学べないという話を良く聞く。開発から保守までを、ずっとやってきている人にとっては、確かにその通りかもしれない。私は、この点を考えても、開発から参加している人を保守で長期アサインすることは良くないと思っている。

ただ、だからといって保守には新しい技術が何もないと言っている訳ではない。それは、他人の作ったシステムに触れたとき、必ず自分の持っていなかった技術を発見することができるからだ。さらに最近では、WS(ワークステーション)やAI(人工知能)技術を利用した新しい保守が検討され始めるなど、今後の保守では、ますます幅広い技術力が要求されるようになってくるであろう。

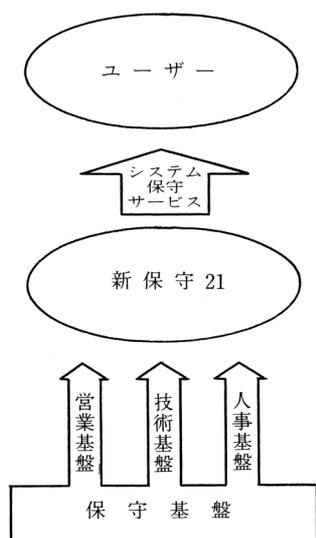
3. 保守に関する研究・推進活動

現在、社内社外を問わず保守に関する研究・推進活動が非常に活発に行われている。その中から私が参加している2つの委員会について、その活動内容を簡単に紹介する。

3.1 新保守21

“新保守21”は「CAC長期経営計画」に基づいた積極的なシステム保守サービス事業を実現するために、1990年2月27日に発足した社内委員会である。“新保守21”の基本構想としては技術・人事・営業の3つの保守基盤を徹底的に整備することにより、21世紀に向けて今までにないシステム保守サービスを実現することにある(図1参照)。

図1 「新保守21」のイメージ



3.1.1 21世紀における保守とは

“新保守21”では、21世紀におけるシステム保守サービスについて、次のようなイメージを描いている。

- (1) 戦略的パートナーとして、顧客に密着したサービスを提供する。
- (2) 他社開発システムの保守をも含めた積極的な保守サービスを提供する。
- (3) コンサルティングからシステムの運用まで含めた幅広いサービスを提供する。
- (4) 新しい方法論、整備された支援ツール群、保守ネットワークによるリソースの有効利用など高度技術によって体系化された均一な保守サービスを提供する。

3.1.2 保守基盤の整備

“新保守21”では、上記のような保守サービスを実現するために、当面、保守基盤の整備を進めなければならないと考えた。具体的には、以下のとおりである。

- (1) 技術基盤
 - ・保守メソッドロジーの確立
 - ・最新技術の利用
 - ・保守支援ツールの導入・開発
 - ・保守用社内ネットワークの構築 など
- (2) 人事基盤
 - ・ローテーション・ルールの確立
 - ・保守技術者用、教育研修制度の確立
 - ・保守技術者の評価基準、待遇基準の確率 など
- (3) 営業基盤
 - ・保守サービス品目の充実
 - ・料金体系の確立
 - ・受注条件の設定 など

3.1.3 現在の活動状況

上述のような基盤整備に向け、“新保守21”では現在、次のような活動を進めている。

- (1) 新保守21推進委員会
 - “新保守21”の目標達成に向け全般的な推進活動を実施。現在、以下の5つの分科会を発足させ、その取りまとめを中心に活動している。
- (2) 保守環境に関する分科会
 - 保守作業環境のダウンサイジング化を最終目標とし、理想環境の構想を提案していくための分科会。現状のプロジェクト環境から理想環境へ移行していく上での問題点やその効果についても検討する。
- (3) 支援ツールに関する分科会
 - 現状あるツールの整理と流通、および、開発中ツールの作成推進活動を行う分科会、さらに、新しいツールの構想や研究活動も行う予定。
- (4) 作業標準に関する分科会
 - 保守作業手順、品質管理方法、作業体制、トラブル時の対応方法等の標準化を検討する分科会。

将来的には、標準化資料の1つである「システム保守標準」の全面改定も検討する。

(5) 保守ドキュメントに関する分科会

保守用ドキュメント体系の確立を第一目標とし、同時に開発時に作成すべき保守のためのドキュメントを提案していく分科会。ドキュメントの電子化による文書検索等についても検討対象とする。

(6) RE-3テクノロジー研究分科会

保守技術の高度化を目指す上で、今後有望と見られている3つの技術、リストラクチャリング、リエンジニアリング、リバース・エンジニアリングを研究し、実際の保守に役立てていくための分科会。ベンダー・ツールの評価、実際のツール導入、プロトタイプ作成なども検討する予定。

3.2 SMSG

SMSGは、現在の保守のやり方に問題意識を持つ企業が参加して、保守に関する勉強をしながら、解決策について話し合う社外研究会活動である。活動は1年単位で行われるが、初年度の参加企業を見てみると、保守の問題が非常に幅広い業種に及んでいることにまず驚かされる。さらに、各企業とも非常に熱心であり、保守の問題がいかに深刻であるかがうかがえる。

活動は月1回のペースで、勉強会と小グループによる研究活動とが交互に行われている。CACからは技研の秋田室長と私の2名が、それぞれ「保守技術に関する研究グループ」と「ドキュメントに関する研究グループ」に参加している。前者については、秋田室長が時々「技研月報」で紹介しているので省略し、ここでは私が参加している後者の活動について、その内容を簡単に紹介する。

3.2.1 ドキュメントに関する研究内容

- (1) 保守に関する必要情報を整理する。
- (2) 必要情報の体系化を行う。
- (3) 保守ドキュメント体系案を作成する。
- (4) ドキュメントごとに必要項目を定める。
- (5) 開発で作成できるドキュメントを選別する。

3.2.2 メンバー構成

メンバーは、以下の9社から構成されている。

- ・電力会社 4社
- ・都市銀行 1行
- ・証券会社 1社
- ・生命保険会社 1社
- ・ソフトウェア・ハウス 2社

3.2.3 活動の経緯と現状

これまでに、5回の研究活動を行ったが、その経緯と現状に

ついては以下のとおりである。

- (1) 各社の開発・保守ドキュメントを調査。
- (2) 各社の現状ドキュメントにおける問題点を分析。
- (3) ドキュメント問題に関する対策を検討。
- (4) 保守における必要情報を整理。
- (5) 現在、保守情報の体系化案を検討中。今年度中には体系化案を完成させる予定。

4. 保守プロジェクトへの適用

今まで述べてきたことについて、どれも実現は未来の話であり、現状の保守プロジェクトに、今すぐ適用するのは難しいと思っている人も多いかもしれない。しかし、実は本気になりさえすれば、実際の保守プロジェクトにすぐにでも適用できるような考え方が数多く含まれているのである。

以下に、これまで私がSBプロジェクトにおいて実践してきた保守改善策の具体例とその効果について述べてみたい。

4.1 小規模プロジェクトの統合

SBプロジェクトで保守がスタートした頃は、10人程度のプロジェクトが2つあり、全く別々に作業を行っていた。しかし、この程度の小規模プロジェクトで将来を見越した要員計画（ローテーション計画）を立てることは非常に困難であったため、プロジェクトを統合し、20人規模のプロジェクトにすることで問題を解決しようと試みた。当初はメンバー間の交流も少なく、期待していたような効果は表われなかったが、3年たった今ではメンバーの配置替えも可能となり、多少は柔軟な要員計画が可能となってきた。

やはり、保守を長期間継続していくことを考えると、ある程度余裕の持てるプロジェクト規模は、絶対に必要であろう。

4.2 大幅なメンバー入れ替えの実施

SBプロジェクトにおいて、開発システムがカットオーバーする直前1年間くらいは機能改善の仕事が多く、作業内容は極めて保守に近いものであった。カットオーバー後、すぐに保守プロジェクトがスタートしたが、当初は全員開発時のメンバーであったため、プロジェクトが発足した直後からローテーションを希望する人が多かった。顧客はローテーションに難色を示していたが、約1年半の間にメンバーの8割をローテーションすることができた。その結果、総合力では低下したかも知れないが、前向きな姿勢で保守に取り組もうという人だけが残ったため、プロジェクトの活性化という観点では非常に効果があったと思っている。ただ、これができる背景にはプロジェクト規模の拡大があったことは事実であろう。

4.3 新人3年教育制度の実施

ローテーションを実施していく上で1つだけ、どうしても解決

できない問題があった。それは、中堅技術者が思うようにアサインできないという点であった。それで、とりあえず新人をアサインすることにしたのだが、そのとき思いついたのが“新人3年間教育制度”であった。この制度は、開発プロジェクトより保守プロジェクトの方が新人教育に適しているという持論から、まず新入社員を保守プロジェクトにアサインし、約1年間のOJTを実施する。2年目は、1人前のメンバーとして仕事をこなしながら、技術力の向上を目指す。3年目には、これまでの経験を生かしながら、指導員として新人教育を担当する。その後は、本人の希望と教育的見地によって、ローテーションしていくという制度である。

SBプロジェクトでは、毎年2名の新人を受け入れて、この制度を実践しているが、2年目からは非常に大きな戦力となり、期待以上の成果をあげている。

4.4 WS・PCの導入によるダウンサイジング化

大型汎用機システムを対象とした保守の場合、現状ではホストマシン以外を使うことは全くないといっても良い。しかし、世の中の情勢は確実にダウンサイジングの方向にあり、保守技術者といえどもWSやPCの知識は、今後、必要不可欠なものとなる。

そこで、SBプロジェクトでは、WS1台とPC14台を導入し、保守の機械化に利用することでノウハウを蓄積しようと試みた。導入後、約1年が経過し、関係モジュールの検索がWS上で可能になるなど、一応の成果は上がっているが、ノウハウの蓄積という観点からは、まだまだ十分とは言えないのが実情である。しかし、ダウンサイジングにたいする保守技術者の不安

を和らげる効果は、十分にあったと考えている。

5. これからのシステム保守

最後に、これからのシステム保守が、どうなるかという点について、少し私の考えを述べてみたい。

今までの保守技術は、常に、開発の延長として考えられてきた。その背景としては、従来の保守が開発者を必ず何人か残して行っていたために、保守独自の方法論など、あまり必要としなかったからではないだろうか。

しかし、これからの保守を考えたとき、他人の開発したシステムを保守するケースが増えてくるはずである。そのときに必要なのは、調査・分析・情報管理などの技術であり、今までの開発技術とは、方法論からして全く異なったものでなければならない。近い将来、必ず、この保守技術は確立されるであろう。そのときこそ、保守は、本当の意味での新時代を迎えるのではないだろうか。

6. おわりに

「これからのシステム保守について」というテーマであったため、保守のことばかり書いてきたが、保守を考えるに当たっては、開発についても見直さなければいけない点が数多くある。今後、保守の重要性が高まるのは確かであろうが、開発がなくなる訳ではない。やはり、これからの保守について考えるときには、開発まで含めた、総合的な見地からの考察が必要であろう。