

最新刊

実戦型 S E 育成の鍵

佐藤 真・牧野 勝著

A 5 判・246頁・定価2,600円(本体2,524円・税76円)¥260円

本書は、実戦的な S E を育てるための鍵を示したものである。また、人材を“育てる”といった側面からのアプローチだけでなく、S E 自身の“自己啓発”にも役立つようにとの配慮からこの面にも力を注いだ内容となっている。

〔主要目次〕 I. S E 育成の方法論 (S E プロジェクト成功の鍵/S E 育成の鍵/S E 育成の計画と実施他) II. 実戦的プロジェクト・マネージャの育成 (プロジェクト・マネージャの役割/システム分析・設計の考え方と進め方/システム開発のプロジェクト・マネジメント他)

最新刊

経営情報のための ソフトウェア製品生産工学入門

菅野文友著(東京理科大学教授)

A 5 判・272頁・定価2,800円(本体2,718円・税82円)¥310円

著者が前著『ソフトウェア・エンジニアリング』を上梓してよりすでに13年余の歳月が経過した。従って、その間の進展を整理して織り込み、あくまでも日本的なソフトウェア製品生産の実績を基調としてまとめたのが本書である。

〔主要目次〕 1. ソフトウェア製品とその生産 2. 生産管理 3. ソフトウェア生産の仕様化と設計 4. ソフトウェア製品の製造 5. ソフトウェア製品の試験 6. ソフトウェア製品の検査と品質保証 7. 信頼性・保全性設計他

最新刊

ソフトウェア再利用技術 オープン・ソフトウェアの開発パラダイム

片岡雅憲著(株日立製作所ソフトウェア開発本部副本部長)

A 5 判・512頁・定価5,500円(本体5,340円・税160円)¥310円

本書は、「ソフトウェア再利用技術」を軸としてまとめられた、ソフトウェア開発技術に関する解説書である。特にワークステーションやパソコンを中心とするコンピュータの大衆化を前提とした「オープン・ソフトウェアの開発パラダイム(開発指針)」について解説する。

〔主要目次〕 I. 再利用のためのモジュール設計技術(再利用とモデリング/モジュール設計技術他) II. ソフトウェア標準プラットフォームと部品(オープン・プラットフォーム/ソフトウェア部品他) III. 再利用ベースのソフトウェア開発環境と生産管理技術

ソフトウェア・エンジニアリング

菅野文友著 A 5・定価5,400円(税込)¥310
実務的総合管理技術としてのソフトウェア・エンジニアリングを提唱する。

ソフトウェア・モデリング

片岡雅憲著 A 5・定価5,047円(税込)¥310
ソフトウェアの「モデリング(構造化)」と「再利用」を軸とした、ソフトウェア設計技術の解説書。

ソフトウェア工学入門

石井康雄著 A 5・定価3,200円(税込)¥310
ソフトウェアの基本を学ぼうとする人のために書かれた入門書。プログラミングの初歩の知識を想定。

SE プロジェクト成功の鍵

佐藤 真・牧野 勝著 A 5・定価2,500円(税込)¥260
S E プロジェクトを成功に導くための鍵を、システム開発の流れに沿って50項目提示し解説したもの。

ソフトウェア品質管理事例集

菅野文友監修 A 5・定価28,000円(税込)¥460
ソフトウェア生産における品質管理シンポジウム10周年記念出版。87編の論文、4,000点の文献を掲載。

ソフトウェアプロセス成熟度の改善

W.S. ハンプリー著 藤野喜一監訳 A 5・定価8,000円(税込)¥360
5段階の成熟度と各段階におけるソフトウェアプロセスの改善活動について述べられている。

日科技連出版社

〒151 東京都渋谷区千駄ヶ谷5-4-2 振替 東京7-7309
電話03(5379)1238 FAX03(3356)3419〔図書目録送呈〕

ホットライオン

第11回ソフトウェア生産における品質管理シンポジウム

〔パネル討論会〕

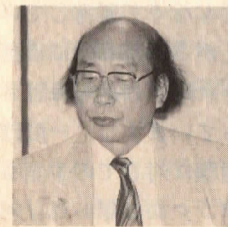
ソフトウェア技術者って、 魅力ある? それともキツイ?

— その職業生活を考える —

戸塚 優秀な学生がプログラマーの分野に進んでいくが、必ずしも定着率がよくない、優秀であればあるほど失望してやめていく人がいる、という話を理科系の先生から聞いたのはかなり前のことです。

今日の情報化社会の中で最先端の職業分野だとされながら、実際の労働と生活の実態を見ると非常に厳しいものがある。長時間労働であり、不規則な労働であり、ストレスがたまる。さらには、この道を選択しても早い時期に実質的な定年年齢がきてしまうかもしれない。そういう危惧が、若い学生諸君の中にも広がっているように見受けられます。

今日の高度情報化社会を実質的には担っているソフトウェア技術者という職業が、日常の労働と生活を通してどういう問題にぶつかっているのかをまずはっきりさせる。そして、そういう問題はどこに原因があるのか、またそれを解決するにはどういう方法、どういう選択があり得るのかを率直に議論してみたいと思います。それではまず秋月さんからお願いします。



戸塚 秀夫氏

〔出席者〕

パネルリーダー

戸塚 秀夫 ●埼玉大学教授

パネルメンバー (五十音順)

秋月 允 ●(株)アイネス取締役

風間 純一 ●日本タイムシェア(株) (当時)

中村 圭介 ●武蔵大学助教授

藤垣 裕子 ●東京大学助手

●能力に応じ、職能資格制度を導入●

秋月 私どもの会社は生命保険会社から分離独立した関係で、大量データを処理する計算センターとしてスタートしました。ソフトウェアの開発といっても、どこかにデータ処理、データプロセスということを念頭に置いています。

私どもは会社の組織を、毎年毎年少しずつ変えています。会社のスタートは基本的にはインテグレーション、あるいは公共と呼んでいるデータプロセス部隊でした。お客様からいただいたデータを処理するという過程で、自治体関係の仕事、民間の給与から始まり、今では主に VAN の関係のシステムを運営しています。この中には、計算機の運用を司る部隊もいます。情報通信のネットを張っているので、その関係の開発支援部隊は別枠でついています。私は今、システム開発本部で主

にアプリケーション中心で、実際にはこの中を幾つかの業種別に分けて対応しています。

私どものソフトウェア開発あるいはそれに対する技術者は、幾つかタイプが違います。ここでは、主にわれわれで持っている計算機、通信を利用した運用までも含めた技術があります。機器の発注関係とか機器の配置、維持運営してライフサイクルを保っていくための工程ということに立っています。

それぞれの技術者がソフトを開発したいなという時、希望がなるべく幅広くそれぞれに見える格好にしています。私どもは職能制度といって、給与体系も含めて少し変えています。仕事は面白いけれどきついというのが、アンケートをとったりしても出ている。その魅力を彼らによくわかってもらい、技術をつけてもらうというのが、基本的にいつも願っている姿です。

職能資格制度は、7年位前から社員の能力考課等を含めて給与にまでリンクした格好で反映させています。これは、それまでの年功序列型の賃金体系から相当調整をやってきて、ここ5、6年は新人で入った時からこういう制度の下で適用されている。M職とっているグループと、S職、最初入ってやるJ職ということです。

基本的には、職務を遂行するための能力を身につけて、自らクリアしてくださいという趣旨で目標設定をし、それをクリアするとJの3から4に上がります。一般に入ったら、J3を2年、J4を2年という標準モデルが設定されています。基本的には絶対評価でやっています。それぞれの人が目標を設定し、その中で職務を遂行していくのに、自分で気付



秋月 允氏

いて努力することをベースにしています。

運用は年2回、目標管理表兼成績考課表を用いており、部門控え、人事控え、本人控えの3部コピーでワンライティングで残るようになっていきます。この用紙を各個人に配り、本人が自分で、この半期にこういう能力を身につけたい、どんな成績で、職務遂行上どんなことをやり遂げたいと書く。J3の人は皆さんができるフォローをしるとか、どうしても一般的に書いたようになっていきます。職務基準としてみると、オペレーションとか運用回りを中心にみている人もおり、必ずしもソフトの関係者だけではないので、実際には各部門で仕事の性質に合わせて大枠の目安をつくっています。

主に、1人でオンラインプログラムができるようになるとか、オンラインのデータベースを使ったシステムが設計できますということを書いています。特殊なものでは、リーダーシップをとる。もう少し部下の面倒をみるとか心配りをするというフォローを、課長と1人当たり1時間前後の面談をして目標設定をします。その時に等級、レベルとありますが、その人がJ4であっても目標とした課題が全部J4とは限らないわけで、違いを本人もマネジャーも知って、半年たった後で評価する。そんな運営方法をとっています。これを2回クリアしたという評価をすると、J4からS0に上がるようになっていきます。

先ほどモデルと言いましたが、当初はほとんどモデルの人ばかりで、逆に言うと評価がしづらい。後輩が先輩を追い抜いているケースもあります。仕事の都合上、リーダーシップをとるのが必ずしも上位等級のものということもなく運営しています。結果として、一度上がると基本的には落ちないということで、それ以上は絶対保証されており、給与にもなっていますし、一度確保すれば何とかなる。退職に対しての手当も、その時その時にこのランクに1年いれば何点と加算式にやってい

ます。退職間際にいたポストではなく、その時その時のポストの積算、それも切り替えをしたので、全社員が今自分は何点持っているという通知がいつ知らされている。その時その時で社員に努力してください、能力を上げてくだされば会社としてできるだけ保証をしていきたい、応えていきたいということです。

実際の運営では、全員が必ずしも上がりたわけではなく、いろんなタイプの人が出てきています。能力開発室が、主に研修あるいは新入社員の研修以降の全部のことを別立てでやっており、専門で8人位のスタッフがいます。それらの中で、新入社員当時の研修は全員一斉に集合研修を3ヵ月やります。チームワークづくりとか知り合う部分も相当あります。上期の研修のコースでは、基礎的な知識を主にやっています。研修カレンダーを配って、年2回位アナウンスされています。

この中で、自分で選択して取りたい、学びたいというものをマネジャーに申し出て、許可をもらって受講する。必ずしも技術者とは関係ないようなものもあります。最近では、パソコンのMS-DOSなど、実際の仕事とはあまり関係なく、わが社の中で人口を増やしたい、弱いという部分を設けています。カリキュラムの数は2、3ページ続き、このほかに情報処理の試験、工事担任者試験も含めて、達成コースというページになっています。

自分の能力を伸ばすチャンスをオープンに与え、それを習得してもらう。そこも一つの評価の対象として、次の自分のやりたい仕事をやるチャンスを自らつくっておいてもらうことをやっています。わが社が実際に運営している例としてお話をさせていただきました。

● 郊外型オフィスで社員の充足めざす ●

戸塚 続いて、日本タイムシェアの風間純一さんをお願いします。

風間 私もこの業界には長くて、卒業してからずっとコンピュータと一緒に育ってまい

りました。

今回は特に NTS の総合研究所、マシーンに関して働く技術者の労働環境、職場環境、生活環境からスポットを当ててみようと思取り組んでみました。

この総合研究所は、郊外型オフィスと位置付けています。以前ですと、お客様に隣接した場所で作業する、要するに人を集めるということでした。これは日本のやり方で、アメリカのほうも注目している部分はあるんですが、そのメリットはお客様とのインターフェースが非常にいい。各工程間の引き継ぎも非常にいい。技術力の確保ということであれば、これはメリットです。ところが、あまりにもコミュニケーションがいいので、すぐ言えば変えてくれる。かたまたまなくても、そのうち言おうということで、これは非常に大きな項目になっています。

ところが、大都市圏からの仕事の発注はどうしても下がらない。それに引き替え、地価が高騰して、ロケーション、場所と広さ、スペースの問題がクローズアップされ、オフィスのコストが開発コストの増大になってきます。ただ、これはわれわれだけでなく、発注側のお客様に関してもそうだと思います。かなりの人数を抱えて、こういうフロアを用意しておくことは、開発費という直接コスト以外に間接のコストとして付加されている。通勤問題、住宅問題も今回のテーマに則して思うのですが、技術者の生活感が変わってきます。同じようなソフト開発の仕事をするのだったら田舎へ帰ってやろう、通勤も楽だ、住宅も安い、家も買える。これは当然だと思います。しかし、出張とか、業務に携わって移動が発生してきます。これが繰り返しになる。この問題を解決する手段はたぶんあるとは思いますが、私どもの会社1社では無理です。われわれは遠隔地開発をやってみようじゃないかと発想を変えました。

それで浜松につくりました。東京、大阪と



風間 純一氏

いう大都市圏へのアクセスが便利で、温暖であり、風光明媚である。これは環境上から選びました。実際、条件として遠隔地開発をするために工程をきちんと分け、工程分離開発の体制をとりました。開発環境は、ここから発想が出ています。設備投資は必要であるという判断に立って環境も整備し、ネットワーク化をし、そこで働く人のために寮とか社宅も実現し、実行していききました。

建物は研究棟、研修センター、寮を木々よりも低い低層分散配置という配置をしています。これはデザイナーとか建築に携わる方々の意見を取り入れて、モデルは欧米型、ポストンのある研究所を参考にしたということです。

お客様が来られて、私は二度ほど往復しましたが、息が切れます。何でこんなに遠くしたんだと言うんですが、ソフトウェア技術者は運動不足ですので、こういうのも少し考えたほうがいいんじゃないか。ここではガラスを多く使っています。ガラスを多く使ったというのは別に意味があって、都会では得られない環境です。朝、出勤して昼になる。夕暮れがきれいに見えます。星が出ます。夜でも昼でもマシン室の中でやっている、やっぱり疲れるんです。だから、人間のリズム、生体のリズムを感じながら仕事をしてもらおうというねらいも入っています。

遠隔地開発をどのように実現したか。PMS規定というのがあります。これはプロジェクト管理規定で、十数年来ずっと続けており、年度年度で改訂してきています。これを全社で標準化することです。その中にSOPがありますが、この規定のプロジェクト管理の手順

の部分システム化しよう。これはどこでも使えるようにということで、EWSとネットワークを組んで、UNIX上で実現しています。ですから、ある人が大阪でプロジェクト管理のデータを入力すると、どこへ出張に行ってもドロワの中に入っているの、簡単に開けられる。ドキュメントもすべてそういう感じで運用しようということの試みの一つです。これによって、進捗とか作業時間が定量的に把握できます。私は生産管理部でプロジェクト管理の元締めになっているので、全社のオーダープロジェクトが全部見えるようになっております。

もう一つのシステムとして、技術見積り、スキルズ・インベントリーのシステムを提供しています。先ほどの仕様変更の多さとともに、発足段階の見積りの甘さというのは大きなテーマになっています。われわれは勘と経験以外に、コストモデルでものを提供したいということです。2年間データをとって、帰分析で一番いいところを取っており、これからもこれを続けていこうと考えております。これらを実現するために、ネットワークとEWS、サードパーティのツールも含めて、わが社で開発している管理ツールとか設計支援ツールがあります。お客様から受注するときは、東京とか大阪の事業部が主体となります。上流工程を事業部でやって、SOPを見たら全部管理でわかりますので、分離受注をします。これがもどってきてテストとなって、納品されます。この仕組みをうまく動かすと、札幌でも大阪でも似たようなことができると考えています。

すでに、1年半になりますが、新卒者の確保は当初から予定していた効果です。われわれが予想もしなかった効果も出てきました。地域の活性化に貢献ということです。そこだけが活性化するのではなく、人的な交流も出てきます。周辺の大きな企業の人たちと密接にコンタクトできるようになり、われわれも

活性化になっています。また、離職率の低下と社員の満足感があります。社員の満足度というのは個人それぞれに基準があり、自分は満員電車とぐちゃぐちゃとしたリストがいっぱい積んである机がいいという人にはあまり向きません。

効果が出た代わりに、課題は当然ついて回ります。出張者がだんだん増えてきた。漸増です。今の仕事の仕方ですと、お客様と共同でやらなければならない。テストのときも一緒にいてくれ。旅費がだんだん増えてきました。それから、生産性の計画実績差異が出てきました。かなり落ち込んだということではありません。われわれが想定していたよりも、実績と計画が5、6%位の差で変動する。これには開発環境を移設しなければならない。また新人が多く、経験者と新人の比率がほかの事業所に比べて非常に多くなったことで、不慣れ、経験不足の面で若干われわれの想定よりも変動が出ています。ただ感覚的に変動が出ているということではなく、先ほどのようなツールを用いて、かなりの確に計測できています。

私はシステムズオペレーションの調査団でテキサスへ行った時、EDSを見ることができました。そうすると、環境、設備はきちんとしています。それと同時に、技術者、働く人への意識づけの高さ、意識の高さがある。こういうものを見てくると、そこで働く人がいて、きちんとした設備があつて、だからお客様は安心してアウトソーシングができるのだろう。こうなってくれば、われわれもきちんきちんとしていこうと考えています。

●仕事のきつさと管理体制●

戸塚 次に、中村圭介さんをお願いします。

中村 ソフトウェア技術者は当然魅力のある職業なんですが、きつさというのを考えてみた場合、二つあると思います。一つはストレスです。それについては、藤垣さんが詳細な分析を報告されると思います。もう一つは

キャリアの不透明さです。いったいどのゴールに向かって、どういう訓練を受けさせてもらうのかわからず、よくわからない。これがきつさの原因というか、二つの現象だと考えています。それに対して、秋月さんと風間さんが企業の方として何とか解決したいとおっしゃられた。秋月さんの場合は能力向上と教育訓練に力点を置かれ、風間さんの場合は経営管理の合理化を強調された。それによって、二つのきつさが何とかクリアできるんじゃないかというお話だと思います。

私はこの数年間この業界のことを調べていて、幾つかの興味深い事実を発見しました。ここで話するのは、そのうちの三つです。

一つは大手のメーカーは別としても、ソフトウェア業界は中小企業が多く、経営管理体制が整っていない企業が多い。たとえば、教育訓練を担当したり処遇制度をちゃんと考える人事労務のセクションとか、ソフトウェア生産を管理するセクションがそもそもない。

そうはいっても、誰かが管理しなければいけない。誰が管理しているのかと技術者の人などにお話を伺っていくと、プロジェクトのリーダーと呼ばれる上級の技術者の人たちでした。そのリーダーは生産管理業務だけではなく、与えられた個々のメンバーへの仕事を割り振り、残業のことから派遣技術者の管理まで、要するに労務管理業務も同じように担っている。と同時に、リーダーも技術者ですから、技術者として上流工程からいろんな仕事を行う。プロジェクトのリーダーは、第一線の管理者でありながら、同時に技術者という姿が見えてきた。これは大きなところでもおそろくそうです。



中村 圭介氏

経営管理体制そのものが整っていない未熟さを、プロジェクトリーダーが代わりにやっている間接管理だといってよいと思います。マネジメントがいろんな管理をするのではなく、マネジメントがいて一般の働く人たちがいて、その間にリーダーが入って、このリーダーを媒介としたいろんな管理がなされている。それが一つの興味深い事実でした。

二つ目はリーダーを媒体とした間接管理の体制の中で、合理的、効率的にソフトウェアを開発していくために、いろんな管理技法とか手法、プロジェクト管理ツールを大手の企業も考えて適用しようとしています。どうも私の見る範囲では、そんなに成功しているとは思えない。

メンバーの自己申告による不確かな情報をリーダーが把握して、その不確かな情報を基に進捗状況を報告する。そうすると、進捗管理そのものが根底からうまくいかない。

あるいは、さまざまな標準化をされています。ドキュメントとか設計書とか、いろいろお伺いしました。現場でやっておられる方は必ず効果があると実感しているんでしょうが、ごく大ざっぱな数字で見ると、そんなに全体としては上がっていない。

なぜだろうと考えたとき、一つは標準化を進めている工程そのものは全体のごく一部で、問題は上流工程にあって、その工程では標準化がなかなかできていない。したがって、一部でやっても、全体に影響を及ぼさない。少なくとも、そんなに効果が上がっているようには思えません。

要するに、ソフトウェア開発に適合的な管理技法はまだ発見できていないようだということがわかってきたんです。これが二つ目の興味深い発見です。

三つ目は、労務管理にかかわることです。OJTというのは、やさしい仕事から始めて、意識的にだんだん難しい仕事を与えていく方法です。また off-JT を採用している企業も

多く、この業界は中小企業が多いにもかかわらず、職場外の教育訓練はかなり充実しているとみていい。能力を向上させようという積極的な意思、方針、行動はたしかに先ほどの標準化とは別に役に立っている、プラスに働いていることがわかる。

では、問題はないのか。先ほどの間接管理と関係してきますが、間接管理と同時にもう一つの特徴は、プロジェクトが流動的だということを見つけたんです。固定的ではなく、常に変わる。それぞれの技術者に仕事を割り当てるのはリーダーの役割です。リーダーは個々のメンバーの力量を把握して、公平に教育訓練を施させているのだろうか。リーダーは常に忙しく、しかも実質的に教育訓練を与える決定権を持っている。この人たちがメンバー全員に公平に、かつそれぞれに適切な訓練、つまり OJT と off-JT の機会を与えているのだろうか。論理的に疑問を持ちました。

もう二つほど、労務管理について言うと、人事考課は行われているんですが、ソフトウェア開発というのはそもそも集団作業です。にもかかわらず、集団間の比較なしに個人の直接の評価が行われている。それと同時に、技術者の働きをよく知っているリーダーが第一査定者になっていない。また、ソフトウェア業界というのは専門職の集まりである。にもかかわらず、専門職制度がものすごく多い。ある意味で、専門職というのは企業の中で地位を与えるということです。それによって技術者を会社側が直接つかまえようとしているとも解釈できますが、職制と一体化して、その機能を十分に発揮できていない。

要するに、調査によって発見できた興味深い三つの事実が意味するのは、生産管理、労務管理がいずれも、リーダーを媒介とした間接管理を前提としたものになっていないようだ。どこかでちぐはぐな面が起きている。これを何とか解決しなければ、ソフトウェア技術者の職業のきつさというのは解決できない

んじゃないか。その時、道は二つあって、一つは、間接管理そのものの体制を変えるという方法です。定量的生産管理体制でもプロジェクト管理ツールでも構いませんが、管理部門をきちっとして個々の技術者あるいは技術者集団の仕事ぶりを押さえるというやり方です。もう一つは間接管理なんだ、リーダーを中心とした自立的な技術集団がいるんだということ前提にして、ここに決定権、裁量権をかなり明白に与える。勝手にやってください。その代わり、作業環境とか職場外教育訓練のメニューとかを管理部門として用意しましょう。差し当たりは、この二つの方向がある。どちらの方向に向かうだろうかということ念頭に、いろいろな合理化、工夫をしていかないと、ちぐはぐさはいつまでたってもなくなるんじゃないかというのが、現段階の感想です。

●ソフト技術者のストレス度とその理由●

戸塚 それでは最後に、藤垣さんにお話しいただきます。

藤垣 職業性ストレスの研究をソフトウェア技術者を対象としてやってきましたので、ストレス研究から見たソフトウェア技術者の職業生活というものを考えてみたいと思います。まず最初に、ソフトウェア技術者のストレスというと、テクノストレスという言葉が浮かぶと思いますが、これはアメリカの臨床心理学者のクレーク・ブロードという人が1981年に書いた『テクノストレス』という書名の中で導入した言葉です。邦訳が1983年頃に出ています。この本の中で扱っているストレスの内容は非常に多岐にわたっています。①コンピュータを扱う作業者の知的作業負荷やマ

シンのレスポンスタイムが短くなることによって作業密度が上がる話から、②長時間労働の疲労の話、それから③コンピュータの論理に自分を適応させることから起こる人格変容の話もあります。人格変容とは、具体的にテクノ依存症やテクノ不安症が挙げられています。テクノ依存症は、コンピュータの相手をしているほうがずっと楽で、人間関係が粗になってしまう状況です。テクノ不安症は、コンピュータの前に座ると不安になる状況です。上記の①②③のように大きく三つに分けられますが、これらを全部まとめてテクノストレスという名前で呼んでいるので、定義があいまいです。使う人によって言葉の内容が異なるので、テクノストレスという言葉は使わないでお話したいと思います。

ソフトウェア技術者の作業負担に関する先行研究として、大きく分けて二つの流れがあります。経営学サイドからと、医学保健学サイドからの研究です。経営学サイドからの研究は、情報処理技術者のストレスを列挙して、その主観評定を行うとか、技術的な難しさとストレスの関係を探る。性格によるストレスの違い(非常にアクティブな方で心疾患になりやすいタイプの性格特性があり、ソフトウェア技術者の中にもタイプAの方とそうでないタイプの方がある。それでストレスがどう変わるかという研究)や、職種の違いによるストレスの差、ハードウェア技術者とのストレスの違いを観察したのがあります。これらの研究は組織関連のマクロストレス、給与、命令体系、学習機会の問題などの分析は細かいのですが、作業内容に立ち入った分析が粗いために、職場でストレス要因となる環境条件を洗い出すための基礎資料があまり得られない傾向があります。ただ、業界全体として参考になることはいろいろあります。

医学保健学サイドからの研究は、ソフトウェア技術者の心身の健康状態の分析がなされ



藤垣 裕子氏

ています。他業種でもやられている一般健康調査をやってみたり、メンタルヘルスの面の調査をやってみたり、生活パターンがどのように乱れているのかを調査したのがあります。ただ、これらの医学保健学サイドの研究は、特に精神衛生面の研究や一般傾向調査のもので、テクノ依存症の枠にとらわれて、その枠に合うもののみ拾い出そうとする傾向が多々ある。また、健康調査や意識調査の結果が、ソフトウェア開発の作業内容のどのような特質や特性に起因するのかについての考察が希薄な点に問題があります。

このような先行研究の問題点を踏まえると、ストレス反応が作業のどのような性質や特性に起因するのかを分析して、現場の職場における対策の基礎資料を得られるかたちでストレス研究をするべきではないかと考えられます。

ソフトウェア技術者が、他職種の人に比べてハイストレスかという話題に移ります。これはソフトウェア技術者協会で行った、男性713人のデータの結果(図1参照)です。CFSIという蓄積的疲労兆候調査を使っています。これは心身の違和感、疲れなどの感じが何日も停滞している状況、すなわち蓄積的疲労を尋ねる評定法です。ソフトウェア技術者の場合には数カ月間、数年にわたる開発期間を持

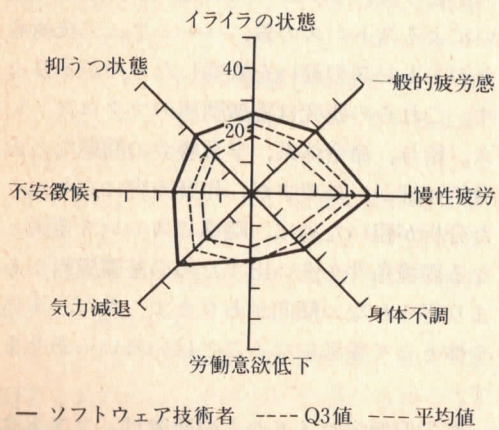


図1

っているため、蓄積的疲労が重要であるかと思われ、この調査票を使いました。抑うつ状態というものも入っており、メンタルヘルスも含めた心理アセスメント的な要素を持っています。

一人ひとりの技術者に、81個の質問に対して○か×かで答えていただいています。81個の質問は、過去の調査における1万2,000人のデータの因子分析によって、8つの特性群、すなわち不安兆候、抑うつ状態、イライラの状態、一般的疲労感、慢性疲労、身体不調、労働意欲低下、気力減退というパターンになっています。81個への○か×かで、各特性項目群に対する項目への平均訴え率を求めて、それを図示しています。一番内側の点線は電機労連で行ったもので、職種はソフトウェア技術者に限りません。生産現場の方も事務の方、ハードウェアの方もいらっしゃいます。その全体を含めた1万2,000人のデータの平均値です。2番目の点線はQ3値といいます。75%の職場が含まれるラインです。この1万2,000人のデータは全部で男性95職場、女性34職場あったんですが、これは男性のパターンですから、95職場のうち75%の職場がこのQ3値の点線の内側に入ると考えてください。このQ3値のラインの外に出るということは、かなり高得点です。そしてソフトウェア技術者協会全体で行った男性の平均は、そのQ3値を上回っているため、この調査票の結果を見る限り、ローストレスとは言えないと考えています。電機労連における多くの職種の含まれた基本パターンに比べて、かなり高レベルだということです。

次に、ソフトウェア技術者にとって何がストレス要因であるかを考えてみます。調査のうえで、主観的なストレス要因評定を行いました。項目は、「納期の時間的切迫」、「毎日の仕事量の多さ」、「プロジェクトチーム内の人間関係」、「ユーザーとの人間関係」、「納品後のトラブル」、「仕様があいまいである」など

表1

1	A.	納期の時間的切迫	5.2
2	B.	仕事量の多さ	4.8
2	K.	仕様のあいまいさ	4.8
4	C.	休日不足	4.5
4	D.	残業・夜勤の多さ	4.5
4	O.	要員の質的不足	4.5
7	G.	雑用の多さ	4.4
7	J.	納品後のトラブル	4.4
9	E.	勉強時間の不足	4.3
9	S.	報酬への不満足	4.3
9	T.	人材育成機会の不足	4.3
12	H.	チーム内人間関係	4.2
12	P.	開発環境の悪さ	4.2
14	F.	仕事の難度	4.1
14	M.	能力発揮機会の不足	4.1
14	Q.	開発場所の悪さ	4.1
17	I.	ユーザー人間関係	4.0
17	L.	自由裁量権の少なさ	4.0
17	N.	仕事割当の不平等	4.0
17	U.	技術の変化	4.0
21	R.	昇進の機会の不足	3.4

といった22項目、これは調査に先立つ面接調査で洗い出していたものですが、それに対して7段階評価で答えてもらいました。1番は全くストレス要因ではない、7番が非常に大きなストレス要因であると思う、というような7段階判定です。このように主観的なものですが、これでストレス要因の中での評定順位を調べてみたわけです(表1参照)。

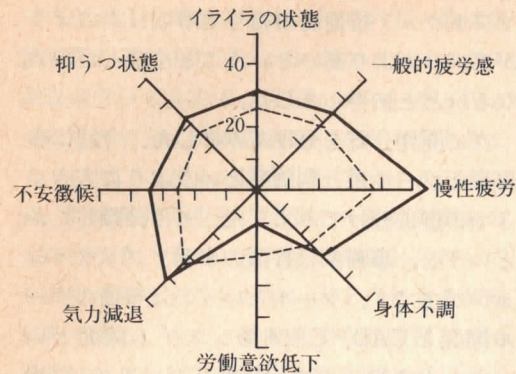
主にソフトウェア技術者協会での約800人のデータを基にしていますが、男子も女子も、「納期の時間的切迫」、「仕事量の多さ」、「仕様のあいまいさ」のストレス要因の評定が上位の三つを占めています。報酬への不満は9位です。経営学サイドからの情報処理技術者のストレス要因の列挙と主観評定という研究は、これと同じように7段階評定で行っているんですが、アメリカの研究では報酬への不満が必ず1位か2位を占めます。日本の技術者はアメリカの技術者に比べて、報酬への不満よりも仕事量に関するストレス要因を高く評定する。これはなぜなんだろうと考えてみたのですが、社会学者の書いた本によると、

ソフトウェア技術者の給与水準は日本のほうがアメリカより高いということを読んだので、なるほどと納得しました。

次に開発分野を分けてみました。今回は事務系とツール系と制御系というように大きく3つの群に分けてみました。どう分類したかという、事務系は普通の事務アプリケーションが主です。ツールというのは普通のツール開発とCAD/CAMのシステム開発といったものを指します。制御系というのはOSの開発とファクトリーオートメーションと通信などをまとめました。

まず、「納期」に関しては、制御系が一番評定が高い。これは、3群の間で有意差があるんですが、なぜかという理由に関しては、制御系というのはメーカー系のソフトウェアハウスあるいはメーカー系列の真下にいるということもあるのかもしれませんが、「ユーザーとの人間関係」とか「仕様のあいまいさ」に関しては、確実に事務系がほかの群を抜いて評定が高くなっています。有意に差が出ていますが、ユーザーとの対応が事務アプリケーションでは負担要因になるのではないかと解釈しています。作業環境は、「開発環境の悪さ」というストレス要因であらわされますが、ツール系に比べて事務系が非常に高い、納得できる結果が出ています。以上のストレス要因判定の結果が「何がストレス要因か」という問いに対する分析です。

次に、ソフトウェア技術者の意欲はどうか、考えてみます。意欲とストレスというのは非常に関係しており、労働意欲が低下すれば心身の不調に対する訴え率も上がるので、そこを考えてみたいと思います。CFSIパターンの中で、この軸は労働意欲の低下ですから(図1参照)、この訴え率が高いほど意欲は低下している、これが低いほど意欲はあると解釈できます。事実、単調作業においては、イライラの状態と労働意欲低下が非常に高く、ほかの訴え率は低いという縦長のパターンに



— 職場H ---ソフトウェア技術者
図2

なります。ところがソフトウェア技術者のパターンを見てみると、全部で28職場位とったのですが、必ずここが引っ込んでおり、労働意欲低下の訴え率は低い。つまり労働意欲は非常に高い。それなのに、ほかの7特性の訴え率が高い。この図(図2)はユーザー系ソフトウェアハウスの一つの職場Hのパターンですが、このようなパターンがソフトウェア技術者に特徴的である。これは一つの解釈では、意欲は非常にあるのにストレスは高い。「魅力はあるけれど、かつきつい」という状況にあるのではないかと思います。

さて、それではソフトウェア技術者のストレス要因はなぜ発生するのだろうかということ、過去の研究のデータと自由意見のまとめから出してみたいと思います。調査をするうえで、こちらが設定した設問に対する答えだけでなく、自由意見欄を設けたところ、多彩な返答がかえってきました。

たくさんあるので、大きく二つに分けてみました。一つは業界の特徴に規定されるもの、ハードウェアの開発のように歴史が長いのに比べてソフトはまだ数十年の歴史ですから、業界の未成熟みたいなものから出てくる問題として考えられるものと、もう一つは歴史がどうかということよりも作業内容それ自体の特徴によるもの、この二つに分けられるので

はないかと思えます。

業界の特徴に規定されるものとしては、①技術者の慢性的不足、それは教育不足や仕事量の多さを招きます。②請け負い構造であることは、メーカー主導の開発納期の融通のなさを招きます。③派遣技術者の問題とか、④経営法の特異性、技術者と経営者の未分離は先ほどのお話にもありましたが、産業としての未熟さがある。⑤受注管理の未確立というのは、会社が抱えている技術者の能力査定があるとしたら、この人には次にはこういう仕事をさせて技術を上げたいというものと、会社としてどういうものを今後受注していききたいという両方のものです。技術者の能力アップと、会社として次にどういうものを受注するかが、両方で計画的に一致しているのかということが挙げられます。

作業内容の特徴に規定されるものとしては、納期見積りの難しさとか仕様のあいまいさの問題です。上流工程の作業が非常に難しいという意見なども挙げられます。

最後に、これは私のヒアリングと調査を基にして考えたものですから皆さんからの意見を聞きたいのですが、なぜ作業内容の特徴に規定されるストレス要因が発生するのかを考えました。

建築やハードウェアの設計とソフトウェアの設計を比較した場合、建築やハードウェアの設計では部品となる規格品や仕様を記述する標準形式や用語があります。ソフトに関しては規格品がない。ですから、ぼんやりとしたユーザーのイメージを具体化する作業は、ほとんど技術者個人個人の裁量に任されている。プロの技術者は仕様書に書かれていない潜在的な問題を洗い出すことが期待されており、その能力は過去の経験に負っているということがたくさんある。そんなことが作業内容の特徴に規定される負荷要因の一つの背景としてあるのではないかと。

もう一つは、対ユーザーコミュニケーション

におけるルールや標準形式を確立しにくい。何かトラブルが発生した時、こういうルールになっておりますということがまだできていない。あるいは、こちらも対ユーザーとのコミュニケーションの中で発生する問題ですが、製造のほうに入った場合でも、ソフトウェア作成における規格品や部品があまり存在していないということが挙げられるのではないかと。これに関しては、私が限られた調査やヒアリングから得たものです。

このような規格品や標準仕様の欠陥は、ソフトウェア設計における一つの大きな特性だと思います。よくソフトウェア技術者のストレス要因として、孤独な一人作業である、慢性的な忙しさがある、顧客対応が難しいという3点が挙げられますが、これは別にソフトの開発に限ったことではありません。ハードウェアの開発にも建築の場合にも、この3つに関しては十分挙げられます。ですから、これよりも標準仕様の欠如のほうが、本質的には問題なのではないかと考えています。

規格品や標準仕様の欠如は、ソフトウェア技術者の開発技術の未熟さに由来していて、歴史的にまだ浅いから今後発展するものなのか、それともソフトウェア開発作業の特徴で、これから歴史が深まっていった場合にもこの問題点は残るのかは、一つの話題になるかと思えます。

最後に、技術者の意見を一つ挙げておきます。「ソフトウェアの開発とは製造プロセスではなく、学習とコミュニケーションのプロセスであって、ユーザーの要求は永久に確定しない」という意見です。それからソフトウェア工学者の意見として、「ユーザーと技術者はシステムの要求仕様をつくる過程で互いに学習している。要求仕様というのは作成した時点での両者の学習の成果に過ぎない」という意見です。どちらの意見も、ソフトウェア開発プロセスは、いわゆるハードウェアとか建築といった作成プロセスとは本質的にプロセ

スが異なるのではないかと意見を提示していると思われます。これは一つの考え、意見でしかありませんから、このようなことを皆さんで議論していただけたらと思います。

＊

戸塚 風間さんの日本タイムシェアの実験は素晴らしいと思います。しかし、同時に作業環境、労働環境が素晴らしいものになる反面、進捗管理、作業時間の管理を定量的に把握していく方向へ自分たちは進んでいくようになった場合、ここで問題にしているソフトウェア技術者にとってのストレスという問題とどう関係してくるのだろうか。そこを立ち上げてお話しただけなら、おそらく中村さんとの間の考え方の若干のギャップ、コンフリクトの意味がもっと見えてくるのではないかと思います。

藤垣さんのお話を承って、前提があると思います。それはストレスというものをどう理解するかによりますが、ストレスは生産性とか品質というものを考えていく場合にマイナス要因であるという前提で議論を進めていくやり方があります。ただ、私はソフトウェア技術者のことを勉強し始めてから、ソフトウェア技術者のやられている多くのことは、研究者がぶつかっている問題と似ているなと思っていますが、研究者が論文を書くときは、ストレスがないと書かないんです(笑)。差し迫って首をつるような雰囲気になって初めてものを書くということがあって、ストレスという問題の生産性をどう考えていくのか。どこになるとハイスレスで、プロダクティビティー自体をだめにしていくと考えるのか。しごく常識的な前提ですが、そこがはっきりしないと議論が先へ進まないという感じがしております。

もう一つだけ藤垣さんにうかがいたいのは、今日お話しになった中村さんのように、管理の仕方というのは直接管理か間接管理かというところがかかなり大きいとおっしゃった時に、

管理の仕方の差異とハイストレスの問題はどういう相関があると考えられるのか。ここは研究者相互お二人の議論を詰めていくと、もっと詰められた議論ができるかもしれないと思いがらうかがっていました。

中村さんとの関連で言えば、分野あるいは製品の性格によって直接管理が可能であり、トヨタのプロダクションシステムで開発された日本の製造業の生産手法が、ある分野については結構広がり得るのではないかという議論にまで立ち入ろうとしています。今日はもっぱらソフトウェアハウスのソフトウェア技術者の問題だと思えますが、そこをどうお考えになるか、お考えを伺えればと思います。

●技術の中身の見直しが必要●

秋月 私どもは20年位前には、ご多分に漏れず納期が守れない、誰かが作ったプログラムがいつ終わるかわからない。だから、いつ納品できるかわからないということの繰り返しでした。ともかく今日納品しようとした。間に合わない。じゃあ単純な話、少なくとも次の日に納品できれば、1日の遅れで済む。一つの仕事の単位を1日でできる仕事量に置き直せばいい。私どもの会社で、一つのプログラムを小さな300ステップ位の単位に細かく分けました。誰かができなくても、必ずリカバリーできる単位を設定する。そうすることによって、基本的には納品を守る、あるいは生産の質であり量を把握していくということを徹底してやりました。

しかし、オンライン化とかユーザーのニーズの変化、多様化ということでは、片方では標準化をわが社の中でも進めています。それだけではないかと思っております。いいものと言われるもの、技術力がある人と言ってしまうと、そういうものが現実にはあります。その辺りがあまり評価されないんじゃないか。特技というのとは違うんですが、技術が評価されていないんじゃないか。もう一回見直しをやってみようと思っているところです。通

信の技術者は知識だけではなく、経験や勘も含めた技術がないと、エラーの発生の場所がわかりにくいということがあります。実際にやられている仕事そのものの中身の見直しと、技術者というのはどんなものかという見直しをきちっとやっけていかなければいけないのではないか。総論の技術と個々の技術者の技術とは違うのではないかということも、検討していかなければいけないと思っております。

戸塚 それは先ほどの職能資格制度のような構想と結びついているんですか。

秋月 一つには、一般的なノーマルなものだけが評価され、何年かたった時、SEからリーダーなり何なりというプロセスをイメージすると、それと評価とは必ずしも一致しないのではないかと考えております。

戸塚 ありがとうございます。それでは風間さん、お願いします。

●早めの問題発見・対応を…●

風間 今のままの状況でソフトウェア業界がどんどん進んでいって、果たしてどこでよくなるかということです。躊躇していたら、このまますぐ10年はたってしまうだろう。環境のよさについては、建築会社に頼めばつくってくれますから、定量的把握についてお話ししておきたいと思っております。

今、ソフトウェアの開発方法論は多種多様で建築手法のように画一化はされていない。われわれのようにソフト開発をやっていると、各メーカーとかユーザー独特の手法、ドキュメントを要求されます。これは一概に悪いとは言えません。これを早くきちんとするのが第一義だと思います。

支援ツールを当てがったからそれでいいという感覚はありません。むしろ今まできつと言ったのは、納期間際になって問題が出る。プログラムができていない。そこからきつきが始まるんです。それまでの時間を把握すると、やっぱり低い。きつきを求めるのではなく、事前に問題の早期発見のために使えとい

うことです。ですから数字で現われるもの、期間とか時間は取れるわけです。

ただし、今われわれが提供しているものに関しては、エキスパートシステムがついていないので、見る人がその数字をどうとらえるか。われわれが改善していかなければならないことです。たしかに小規模プロジェクト、1カ月の間に2週間で納期を迎えるようなオフコン、パソコンには向いていないこともわかっています。こういうことを事前に把握して、もし製造要員が必要ならば納期間際で手配するのではなく、危ないと思ったらやっておきなさいというためのもので、何もそれががんじがらめにするつもりはありません。

戸塚 中村さん、よろしく。

●状況に応じて管理スタイル確立●

中村 プロジェクト管理のための支援ツールで管理しようとするのではなく、問題の早期発見に役立てるとおっしゃられましたが、もし問題の早期発見をしようという意思があるならば、こういうツールがなくてもしているんじゃないか。つまり、ツールの有無ではなく、そういうことをすることがプロジェクト全体、チーム全体の何らかの規範にふれるからできないのではないかという気がします。

能力開発でも、管理がきつくなるだけじゃないかとおっしゃる方がいらっしゃいますが、この点は違った考えを持っています。大学で働いていると、大学の講義の品質とか商品の性質、量についてチェックする消費者は学生だけです。ところが、普通に商品を生産して顧客に納める場合、きちとしたルールを守らなければいけない。その時、当然ながら責任というものができる。責任を遂行するために、管理は必ず必要になるだろう。その管理のスタイルが問題だと思います。管理を取り払えと言っているわけではありません。

秋月さんはおっしゃられなかったのですが、アイネスの場合は地方自治体の税務処理の受注計算をやられていた。これは法律である程

度決まっていますから、部品の標準化がかなり進む。そのように、製品とか分野によって管理のスタイルはおそらく違ってくると思います。

戸塚 それでは藤垣さん、最後をお願いします。

●開発環境や管理の仕方とストレス●

藤垣 ソフトを作る楽しさと労働意欲の低下がイコールではないのではないかというお話と、ソフトウェア技術者の意欲と他の業種の人の意欲は違うのではないかというご指摘がありました。面白い指摘なんです。具体的にソフトウェア技術者の意欲と他の業種の人の意欲がどんなふう違うのか示していただけたらと思います。

辞めた人の調査に関してですが、魅力がないから辞めるのであって、残った人は魅力があつてきついに決まっているということは、確かにそうです。今回は残って魅力があると思っている人の調査の結果です。調査集団がソフトウェア技術者協会、SEAという団体で、意欲的に高いところでもありますから、そういう偏向もかかっているかと思うんですが、参考にさせていただきます。

開発支援ツールがあれば、それでストレスが軽減されるといいのだろうかという指摘も、興味深いものです。調査の中で、開発環境が非常に整っていると思っている群と、あまり整っていないと思っている群でストレスの状態を比較してみると、開発環境が整っている群は訴え率がある特性群に対しては下がるんですが、抑うつ状態の訴え率は下がらない。つまり負荷が下がっているとはいえない。あのパターンのバランスでいえば、精神的な負荷がかかるパターンになっているんです。

それはなぜだろうか。これは仮説ですが、支援ツールが整い開発環境が整うにつれて、作業密度が増加するということがあるのではないか。たとえば、支援ツールができること

によっていろんな効果があると思いますが、マシンのレスポンスタイムが短くなるというのも、開発環境がよくなるということの一つだと思います。今までは一つの作業が終わってマシンのレスポンスが返ってくるまでの間、暇な時間としてコーヒーでも飲んで待っていたけれども、マシンのレスポンスタイムが短くなるとそういう時間がなくて、常に意識を集中してコンセントレーションしている時間が長くなる。テンションレベルがいつでも高く、それを下げる時間がない。マシンの機能が増えてきてツールができてくると、単純なキーインのようなルーチンワークが減ってくる。ルーチンワークが減ってくるとどうということが起こるかという、先ほどと同じですが、重要な思考をする思考時間が増えてくるので、質的な作業密度も上がるのではないか。ですから、支援ツールあるいは開発環境が整うことイコールストレスの軽減とはいえないと考えています。

戸塚先生からのストレスは生産性を上げるうえでマイナスであるという前提があるのではないかという質問に対しては、古くからこういうグラフができています。横軸にストレスを取って、縦軸に生産性みたいなものを取ったとすると、必ず逆U字型グラフになります。今までの調査の結果でいくと、どう考えてもソフトウェア技術者はこっちのほうに含まれるのではないかという印象があります。

最後に、管理の仕方とストレスの関係ですが、面白い視点かと思えます。調査の中で、私は職場別の集計50~60職場持っていますが、職場によって随分違います。その経年変化を調べたこともあります。職場別のストレスの違いがいったい何によって発生するのかが興味深い問題です。戸塚先生がご指摘のように、管理の仕方によって職場別の差が現れる場合もあるし、作業分野がまったく違うとか、ユーザーのあり方が全然違うとか、いろんな要因があると思います。ストレスが高

いとすぐ管理の仕方と結び付くわけではありませんが、他の要因の作業内容、分野、ユーザーのあり方といったものを一定にして、管理のあり方だけを変えた時にどう違うのかということをしなくてはいいませんが、一つの研究対象だと思います。

—*—

戸塚 今日、風間さんが強調なさいましたように、今のままのソフトウェア技術者の作業、生産の進め方というのは今日ぶつかっている難しい問題を解決できないだろうという認識は、60年代の末から70年代にかけてアメリカで随分議論され、また実践もされてきた、革新的なマネジャーたちの考え方もあったと思います。今の日本でも、やはり意欲的なマネジャーたちがそういう方向を模索していらっしゃるということに改めて伺って、大変興味深く思いました。また、果たして最も適切な管理方式とは何であるかということについては、国際的にもまだ定着していないのではないかと思います。それは風間さんがおっしゃったように、一つの実験、模索としておやりになっていることで、その結果がどうなるかは意味があると思います。

それから秋月さんが率直におっしゃったことも、示唆的であるように思えます。以前、秋月さんが発言しておられるものを読んだことがあるのですが、日本のものを作るという点でのプロダクションシステムとして国際的に風靡したトヨタのプロダクションシステムと似たようなものが、日本のソフトウェア業界に広がって行って、やがてアメリカのソフトウェア業界に対する重大な挑戦になるのではないかという議論をなさったように記憶しています。ただ、その場合でも、幾つかの周知な限定をつけていらっしゃいまして、そう単純な議論にはなっていない。その辺りのやや留保なさっている問題が今後どう詰められていくかが、研究者にとっても、業界の方々にとっても問題ではないかと思えます。

二つ申し上げたいことがあります。一つは日本の自動車の労使関係のことも同時にこの時期調査していたんですが、テーラーの思想の延長線上でスーパーテーラリズムという形で日本のトヨタシステムが出てきているという解釈と、必ずしもそうでないという解釈と二つあります。私などは、日本におけるテーラリズムというのはそう簡単に根を下ろさなかったという説のほうが当たっているという感じがします。トヨタシステムといわれるものもそれなりの修正をしているところが多分にあり、そこをどう考えるかということ抜きに、テーラリズムの思想を適用すればいい、日本の工業分野での成功はそこにあるというつかまえ方は、やや単純ではなからうか。それに加えて、最近のトヨタならトヨタの労使関係を見ると、従来の管理様式を若干修正するというのをマネジャーたちも真剣に考えていらっしゃるようですから、そういう全体の配置の中で考えなければならないと思っています。

もう一つのポイントは、どういう管理のシステムを開発するにしても、現在のソフトウェア技術者、今後入ってくるソフトウェア技術者の合意をどう勝ち取っていくか、その合

意がどう形成されるかという仕組みが重要だろう。そういう意味で、たとえば労働組合にはなじまない業界で、組織率が大変低いということを知っていますが、ソフトウェア技術者の横のアソシエーションのようなものができた場合、その方々の合意できるような生産管理、作業管理のシステムがこの業界にどう広がっていくかということ抜きにして、マネジメントの側からだけ議論ができるものかどうか。日本の製造業における「優れたシステム」ができるプロセスでは、労使関係の中のさまざまな苦勞をマネジャーはなさっている。そのことを抜きにして、これこそが最も合理的であるという管理システムは、突如天才的なマネジャーの発想から広がってくるものでもあるまいとみています。そういう点で、ソフトウェア技術者たちのアソシエーションといった問題をどう考えるかということが、本日議論されなかった領域としてあるのではないかと思っております。

限られた時間でしたが、ご登場願いました4人のパネリストの方々から、それぞれ今までの蓄積を前提にして大変興味深いお話を承りました。改めて御礼申し上げたいと思います。どうもありがとうございました。◎

「QCサークル」誌'92年6月号

——QCサークルの基本が楽しく・やさしく読める雑誌——

職場の第一線の方々、正しく、効果的なQCサークル活動、QC活動を行うために、またより一層のリーダーシップを身につけていただくために、必要な内容をわかりやすく掲載してあります。そして管理者の方々にも、QCサークル活動の推進、助言・指導の仕方で大変参考になります。

【6月号内容】

- ・特集 QCサークル活動101回目のハードル
- ・体験談 3件
- ・QC手法基礎コース [18] 管理図 (その2)
- ・(いき)²サークルあの手この手

- ・QCサロン
 - ・体験談推薦状況一覧
- そのほか、ズームアップインタビュー、ホットライン、トピックス、茶の間、みんなの広場のページなどです。

- ★ A 5判、各月約104~112ページ
- ★ 1冊340円、年間(12月号)購読料3,420円 国内送料サービス
- ★ お申込みは (株)日科技連出版社営業部へ 〒151 東京都渋谷区千駄ヶ谷5-4-2 電話 (03) 5379-1237~9