



Newsletter from Software Engineers Association

Volume 11, Number 3 November, 1997

目 次

編集部から		1
ソフトウェア産業とソフトウェア研究の 沈滞状況について	玉井 哲雄	2
狩猟民族的技術者の世界	野中 哲	6
ソフトウェアデザインワークショップ報告	野呂 昌満・酒匂 寛	8
組込みソフトウェア分野のプロセス改善活動 に関する情報交換会・報告	高木 徳生・小泉 毅	18
論文募集: ソフトウェア・シンポジウム'98		38
SEA 入会案内		40



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

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

その主な活動は、機関誌 SEAMAIL の発行、支部および研究分科会の運営、セミナー/ワークショップ/シンポジウムなどのイベントの開催、および内外の関係諸団体との交流です。発足当初約 200 人にすぎなかった会員数もその後飛躍的に増加し、現在、北は北海道から南は沖縄まで、700 余名を越えるメンバーを擁するにいたりました。法人賛助会員も 30 社を数えます。支部は、東京以外に、関西、横浜、長野、名古屋、九州、広島、東北の各地区で設立されており、その他の地域でも設立準備をしています。分科会は、東京、関西、名古屋で、それぞれいくつかが活動しており、その他の支部でも、月例会やフォーラムが定期的に行われています。

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

代表幹事： 坂本啓司

常任幹事： 荒木啓二郎 高橋光裕 田中一夫 玉井哲雄 中野秀男 深瀬弘恭

幹事： 市川寛 伊藤昌夫 大場充 河村一樹 窪田芳夫 熊谷章 小林修 桜井麻里
酒匂寛 塩谷和範 篠崎直二郎 新谷勝利 杉田義明 武田淳男 中來田秀樹
布川博士 野中哲 野村行憲 野呂昌満 端山毅 平尾一浩 藤野誠治 二木厚吉
堀江進 松原友夫 山崎利治 和田喜久男

事務局長： 岸田孝一

会計監事： 辻淳二 吉村成弘

分科会世話人 環境分科会 (SIGENV)：塩谷和範 田中慎一郎 渡邊雄一
教育分科会 (SIGEDU)：君島浩 篠崎直二郎 杉田義明 中園順三
ネットワーク分科会 (SIGNET)：小林俊明 人見庸 松本理恵
プロセス分科会 (SEA-SPIN)：青山幹雄 伊藤昌夫 坂本啓司 高橋光裕 田中一夫 増井和也

支部世話人 関西支部：臼井義美 中野秀男 横山博司
横浜支部：野中哲 藤野見延 北條正顕
長野支部：市川寛 小林俊明 佐藤千明
名古屋支部：筏井美枝子 石川雅彦 角谷裕司 野呂昌満
九州支部：武田淳男 平尾一浩
広島支部：大場充 佐藤康臣 谷純一郎
東北支部：河村一樹 布川博士 野村行憲 和田勇

賛助会員会社：ジェーエムエーシステムズ 東芝アドバンスドシステム SRA PFU
東電ソフトウェア 構造計画研究所 さくらケーシーエス サンビルド印刷
富士通 ジャストシステム 新日鉄情報通信システム オムロンソフトウェア
カシオ計算機 キヤノン 中央システム 安川電機 富士通エフ・アイ・ピー
SRA東北 アスキー 新日本製鉄エレクトロニクス研究所 ダイキン工業
東北コンピュータ・サービス オムロン アイシーエス SRA中国 日本電気ソフトウェア
富士電機 プラザー工業 プロダクト・ソリューション (以上 29 社)

SEAMAIL Vol. 11, No. 3 1997年11月11日発行

編集人 岸田孝一

発行人 ソフトウェア技術者協会 (SEA)

〒160 東京都新宿区四谷3-12 丸正ビル5F

T: 03-3356-1077 F: 03-3356-1072 sea@@sea.or.jp

印刷所 サンビルド印刷株式会社 〒162 東京都新宿区築地町8番地

定価 500円 (禁無断転載)

編集部から

☆

ふたたび船便シンドロームに陥りかけている SEAMAIL Vol.11 ですが

☆☆

巻頭の玉井先生のエッセイは、ソフトウェア工学およびソフトウェア産業の現状についての問題提起です。先生が最後に書いておられるように、これを読んだ上での方々の御意見をお待ちします。e-mailで sea@sea.or.jp まで、投稿をお寄せください。

☆☆☆

続いて、新宿からシリコンヴァレイへトラバークした野中さんのエッセイも、期せずして、日米ソフトウェア・ビジネスの比較分析になっています。

☆☆☆☆

一昨年から始まったソフトウェアデザインワークショップですが、順序がちょっと逆になって、昨年の第2回のレポートが先にまとまりました。第3回はいま企画中です。実施はたぶん来年(?)。

☆☆☆☆☆

プロセス分科会(SEA-SPIN)に集まった人たちのうち、組み込みシステムを仕事としている方々が、この夏、情報・意見交換のワークショップを開催しました。それをきっかけに、組み込みソフト分科会(SIG-EMS)も生まれようとしています。

☆☆☆☆☆☆

次号は、10月末に中国・アモイで開催される ISFST-97 の論文特集を予定しています。

☆☆☆☆☆☆☆

ソフトウェア産業とソフトウェア研究の沈滞状況について

玉井 哲雄

(東京大学大学院総合文化研究科)

1. はじめに

SEAの幹事のメーリングリスト上で、幹事の新谷さんが「情報処理」にのった大須賀論文(「日本のソフトウェア問題について—現状分析と将来対策—」, 1997年8月号)を取り上げて問題提起をされた。それでしばらく前に書いて放ってあった文章を思い出した。この元は、1995年の12月に日本ソフトウェア学会のソフトウェア工学の基礎研究会の招待講演で、「何がソフトウェア工学の基礎となり得るか: ソフトウェア工学の沈滞を打開するには」と題して話した内容の前半部分である。その時はOHPを使ってしゃべっただけだが、後でそれを材料の一部を文章化してみた。その後の動きを若干捕捉してあるが、それでも時機を失っている感があるかも知れない。しかし、これまでどこにも発表していないので、SEAMAILに投稿してフィードバックを得るのも意味があるかと考えた。発表していない理由の一つは、最後に「ではどうしたらよいか」という自分なりの展望と提言を書こうと思ったからである。しかし、とりあえずこのまま出して、皆さんの意見を聞き、私の考えるところは別の機会に譲るとほぞを固めた。未完成なものをあえて皆さんの目に晒すことになるが、お許しいただきたい。

大須賀論文や新谷さんの問題提起とは必ずしも対応していないかも知れないが、議論のきっかけになれば幸いである。

2. ソフトウェア産業と研究の現状

現在、日本のソフトウェア産業にもソフトウェア研究にも、停滞感が漂っている。世上では目覚ましい製品も革新的な研究も、日本から生まれてこないといわれている。産業では、二重の意味での空洞化が進んでいる。すなわち、核となるようなソフトウェアは米国に支配され、大量の人手を投入して作られる適用型のソフトウェアは、技術力があって労働コストの安い国への生産の委託が進んでいる。企業では「作らずに買う」というのが、方針になっているという。買いに行く先は、米国やヨーロッパだったり、中国やインドだったりするわけだ。

日本のソフトウェア産業は、1990年頃まではともかくも拡大を続けてきた。ソフトウェア危機という言葉が、初めはソフトウェア工学の必要性を叫ぶための枕詞とし

て、後にはソフトウェア産業政策の正当化のための合言葉として、長らく呪文のように唱えられてきたが、それが世の中のソフトウェアに対する需要の増大に対して供給体制が追いつかないということを意味していたとすれば、ソフトウェア業界にとってはこれほど歓迎すべきことはなかったはずである。問題はプログラムの確保で、仕事を確保するための営業を必要とするような会社は、よっぽどおかしい会社だと思われていた。通産省によるソフトウェア技術者の97万人不足説(西暦2000年時点の予測)も、こうした背景で出された。しかしバブル経済ははじけ、1990年代に入ってこれまでにない長い不況が続いている。この不況はまた、コンピュータやソフトウェアという不況の影響を受けにくかった産業に、きわめて大きな打撃を与えた点でも、これまでの景気後退や不況と異なる性格のものである。

経済のバブル化現象が顕著だった時期、並行してソフトウェアのバブルも大きく膨らみ、それがやはり90年前後にはじけたのではないだろうか。つまり、ソフトウェアの需要は次のような理由により、実需よりみかけがはるかに増大していた可能性がある。

- ・ 同じようなソフトウェアを別々の組織が重複して作る。同じ業界の各社が横並びに、しかしそれぞれ別に、同じような機能の業務システムを作るようなケースである。
- ・ 同じようなソフトウェアを同じ組織が何度も繰り返し作る。業務内容、利用環境、技術の変化に応じて、ソフトウェアシステムも変化せざるをえないが、設計技術や保守・再利用技術のまずさから、むだに作りなおしているケースである。
- ・ 下請構造により全体の売り上げ高が増大する。ソフトウェア産業は非常に多くの企業を抱えるが、それらは階層的な下請構造を作っている。ソフトウェアの開発委託が、ユーザー企業からコンピュータメーカーや大手のソフトウェア会社に出されると、多くの場合、階層の上から下に何層にもわたって外注委託が流れていく。これらは経済活動としては重複して数えられ、みかけの産業規模は増大する。
- ・ これらの要因の原因でもあり結果でもあるが、日

本におけるパッケージ型のソフトウェアの流通は、個別発注型のソフトウェアの開発と比べてきわめて小さい。売上高でいえば、いまだにソフトウェア産業の15%程度を占めるに過ぎない。正確な比較データはないが、米国ではパッケージ型のソフトウェアの売り上げ比率は60%程度と言われる。

- ・ 個々のソフトウェアの規模自体が、いたずらに増大している傾向もある。冗長な作り方、使われない機能などで水腫れになっているソフトウェアが多い。

日本の大学や研究機関におけるソフトウェア研究についても、風当たりが強い。ここでも、産業界と並行するように「空洞化」現象が見られる。新たな研究分野や基本的な概念は、米国から持ち込まれる。革新的な構想や研究成果が海外に向けて発信されることは少ない。そもそも、研究の道具として用いる基本的なソフトウェアも多くが米国製である。そしてよく批判されるように、大学における研究は実践との結びつきが希薄である。日本では一般にベンチャー型の企業があまり育っていないとよく言われるが、とくに大学から生まれたベンチャービジネスはほとんどない。さらに、産業界が元気をなくしているために、肝心の人材育成の面でも、たとい質の高い教育ができていたとしてもそれを産業界が受け入れてくれるかどうかは大いなる不安があるという有様である。総じて、日本のソフトウェアの産業界も学界も、自信を喪失しているようにみえる。

3. 悲観論の蔓延

このような認識は基本的に間違っていないだろうが、あまりに悲観論に傾きすぎているようにも思われる。5-6年前を振り返ってみるとよい。当時、米国では日本のソフトウェアの信頼性、生産性の高さが一つの脅威として論じられていた。たとえば、M. Cusumano による日本の「ソフトウェア工場」についての報告は、その脅威に対して警鐘を鳴らしたものとして受け止められた。このままでは、鉄鋼、家電、自動車のように、ソフトウェアも日本にやられるという論調がよく聞かれたものである。

筆者にとって印象深いのは、1991年のテキサス州オースティンにおけるICSEである。そこで、W. Humphrey がCMMの日本と米国における調査結果を発表した。この2つの調査は、日米それぞれで開いたセミナーの参加者に、アンケートを取るという方法で行われたものである。対象者が、米国の場合は主に防衛関連を中心とした埋め込みソフトウェアの開発に携わる企業の技術者、管理者で、日本の場合はほとんどが事務処理システムに携わる企業からの参加者だったから、単純には比較できな

いということハンフリーは強調した。それにもかかわらず、比較すれば米国の方がプロセスの成熟度が高いと読める結果に対して、米国人の聴衆が「その結果はおかしい」とする反応の激しさは驚くほどだった。その頃、米国で常識化していた日本のソフトウェア開発力の強さという思い込みとは、ずいぶん異なるメッセージだったからである。その認識がまったく逆転した今となっては、この6年前の状態が信じられないくらいである。

明らかにこの時の米国側は、日本の実力を過大視した。同時に脅威論を煽ることにより産業界への補助政策や研究への予算拡大をねらう政治的な意図もあっただろう。現在では、このような認識はまったく逆転した。たとえば、1994年にスタンフォード大学のFeigenbaum等が日米のソフトウェア企業50社あまりを調査して作ったレポートでは、日本の企業が品質を重視するあまり、売れる製品を市場に出すスピードに遅れを取っているという結論を導きだしているが、全体に米国の産業界に対する自信に満ちあふれている。一方、1993年に出された棟上レポートは、やはり日米のソフトウェア産業を比較して見事な分析を行っているが、総じて日本の状況に対してきわめて悲観的である。

5年前の米国における現象と、昨今の日本における悲観的な見方とを重ねあわせてみると、現状認識が極端から極端に振れる傾向が見てとれる。確かに日本のソフトウェア産業と研究が現在大きな問題を抱えていることは、認識しなければならない。しかし、必要以上に悲観視することも、生産的ではない。

ソフトウェアが問題だという認識は、実は日本に限った現象ではない。非専門家向けに解説されるソフトウェアの現状は、その供給力や信頼性に関しておそろしく否定的である。たとえば、Scientific American 誌の1992年11月号に、「ソフトウェアのリスク」と題する解説があるが、ソフトウェアの信頼性が不確実であるという事態に対し、可能な対処法としてあげているのが次のようなものである。

- 1) ソフトウェアに対しては要求仕様を定義するのを避ける。たとえば米国連邦航空管理顧問会議の出している回状では、航空機の種々の構成要素の重大故障発生確率を飛行時間当たり 10^{-9} とすることを取り決めているが、この対象からソフトウェアは定量的なエラー評価が不可能だとして明示的に除外されているという。
- 2) システム内でソフトウェアの果たす役割をあまり重大なものでないようにする。

- 3) 現状のソフトウェアの信頼性の限界を認めて、システム全体の安全性のレベルが多少甘くなるのを許容する。

さらにその2年後のScientific American誌(1994年9月号)では、W. Gibbsという科学記者による「ソフトウェアの慢性的危機状況」という記事があり、デンバー空港事件を始めとする多くのソフトウェア開発の失敗プロジェクトを例にあげて、危機状況は慢性化しているということを、ややセンセーショナルに報じている。ソフトウェア業界で元気がいいのは、極言すれば米国でもW. Gatesだけかもしれないのである。

4. ソフトウェア工学は何をしてきたか

ここで、ソフトウェア工学の状況に目を転じてみよう。よく知られるように、ソフトウェア工学は1960年代の終わりに興り、70年代は構造化プログラミングという主導概念のもとに、理論面と実践面の相補的關係がかなりよく機能して、活発な研究、実践活動を展開した。

80年代に入ると、ソフトウェアの大規模化、複雑化に対処しなければならないという掛け声とともに、ソフトウェア工学の重心が管理技術の方に移行していった。70年代のソフトウェア工学の枠内で進められた、段階的詳細化などのプログラミング方法論、モジュール化、抽象データ型、などの分野は、次第にソフトウェア工学から外れていった。そのことで思い出すのが、1987年にカリフォルニア州モンタレイで開かれた第9回ICSEである。この時の基調講演者の一人がLee Osterweilで、その後のソフトウェアプロセス研究隆盛のきっかけとなった「ソフトウェアプロセスもソフトウェアである」という題で講演をした。この講演は、後に与えた影響の大きさから、ICSEが毎年行っている10年前に開催された同会議の論文から選ばれる最優秀論文として、1997年のICSEで表彰を受けている。さて10年前の当日の講演後、第一に質問したのがHarlan Millsで、まず会場にチューリング賞の授賞者がいたら手を挙げてくれといい、一つの手も挙がらないことを確認した後、「Dijkstra, Hoare達はどこへ行ってしまったのか。ソフトウェア工学の分野で、われわれはもう彼らから学ぶことはないのか。」と発言した。IBMの中にあっては理論派であった当時のMillsとしては、管理技術に偏向して「反知性化」したソフトウェア工学の状況に我慢がならないという感じであった。

80年代のソフトウェア工学は、プログラミング言語やプログラミング方法論に関するテーマを、すでに解決済みのものとするか、あるいは小規模ソフトウェアにのみ通用する技術で、大規模なものには役立たないとして、意識的に遠ざけた。これは発展段階にあるあらゆる分野

で起こる専門分化の現象の一つであったかもしれない。実際、たとえばACMではSIGSOFTが1976年にできて、SIGPLANとは路線を明確に分けることになった。しかし、この棲み分けのために、ソフトウェア工学は技術的な拠り所のかなり大きな部分を失ったのではないだろうか。

80年代における実践的成果には、インスペクションやプロトタイプングがあり、これらはそれなりの効果をあげた。しかし、技術的な革新性という点での魅力には欠ける。新しい技術、とくにAIとの連携はやはり80年代の大きな話題だったが、結局のところそれほど目覚ましい結果を生んでいない。

80年代の後半にはソフトウェアプロセスが多くの関心を呼んだ。続いて90年代に入るとソフトウェアアーキテクチャという概念とその研究の普及に一部の研究者が力を入れた。これらは、プログラミングレベルではなく、システムレベルで、あるいは開発管理という面で、なんとか理論的なアプローチをしようとする研究者の苦肉の策と見ることもできる。ソフトウェア工学は管理であるとしてしまうと、理論の方ではやることなくなくなってしまふ。そこでプロセスという管理面を含みながら技術的な対象となりうる分野とか、アーキテクチャといったマクロな設計対象を相手にして、そこに形式的なモデル化手法を持ち込み、理論的な取り扱いを試みようというわけである。プロセスの方は、CMMやISO9000のお陰で、同じ言葉を使いながら異なるものを想定している気味はあるものの、実務者の興味をもひくことにある程度成功した。しかし、アーキテクチャの方はその点が難しそうだ。

同じく80年代後半から、オブジェクト指向技術の分析や設計への適用が大いに関心を集めた。この動きは、形としては構造化プログラミングから構造化設計や構造化分析が生まれた70年代の流れと類似する。さらに、オブジェクト指向には、再利用・部品化というこれまたソフトウェア工学の長年の課題に、解決の糸口をつけるという期待もかけられた。70年代のソフトウェア工学が構造化プログラミングを指導原理として活況を呈したとすれば、90年代はオブジェクト指向技術を指導原理として同じような活気を取り戻せるかもしれないという期待を抱かせるに十分なものがあつたといえよう。90年代も半ばを過ぎた現在、この期待は実現したといえるであろうか。

オブジェクト指向は確かに豊かな概念であり、現在でも分散オブジェクト、設計パターンやフレームワークといった高レベルの部品化、ビジネスオブジェクトなど、活

気のあるテーマを産み出している。しかし一方で、90年代はもはや70年代のように、単一の原理ですべてをくれるほど単純ではなくなっているともいえよう。さらに学会の細分化傾向も関連する。SIGSOFTとSIGPLANの棲み分けについて述べたが、OOPSLAを主催するのはSIGPLANである。OOPSLAは相変わらず多くの参加者を集める派手な学会となっているが、SIGSOFTとしてはよい商品を先にとられたというところであろう。意地のせい、棲み分け原理が働くのか、SIGSOFT主催のオブジェクト指向をテーマにした会議は目にしないし、ICSEやFSE(ACM SIGSOFT Symposium on the Foundations of Software Engineering)でオブジェクト指向をキーワードに入れたような論文はきわめて少ない。ただし、日本やヨーロッパでは、もともとソフトウェア人口が少ないし、米国の学会の暗黙の掟(そのようなものが実際に存在しているかどうか確かではないが)にとらわれる必要もないので、ソフトウェア工学関連の研究会やシンポジウムにオブジェクト指向の話が出てくるのは珍しくないどころか、むしろ目立って多い。

ソフトウェア工学が、ソフトウェア生産の現実に追いついていない面もあるかもしれない。大規模化、複雑化という合言葉が、ソフトウェア工学の存在意義を説明するのに使われ続けてきたが、現実には、小規模多品種のソフトウェアの生産など、さまざま異なる性質の要求があるだろう。また、ソフトウェア工学が想定していた対象システムは、米国ならたとえば防衛関連の巨大システム、日本ならたとえば銀行システムのような民間の巨大システムで、いずれにせよ特定の顧客向けの受注型システムであった。たとえばシュリンクラップ型のソフトウェアをどう作ったらよいかという問題は、旧来のソフトウェア工学が真正面から取り組んでこなかったものいえるだろう。

その点で、マイクロソフトのお膝元たるシアトルで開かれた1995年のICSEはある意味では象徴的であった。当初はビル・ゲイツを基調講演者にという話もあったが、最終的にはマイクロソフトの関係者の発表や展示はほとんどなかった。そのかわり基調講演の一人がM. Cusumanoで、その題目がなんとマイクロソフトであった。5年前は日本のソフトウェア工場に目をつけ、今度はマイクロソフトというのはビジネススクールの教師らしく目端がきくというべきか。しかし、マイクロソフトほど一般のマスコミで取材されている対象に対して、新たな情報や知見を披瀝するのはクスmanoにしても荷が重い。話としてはつまらなかった。聴衆の反応も冷ややかだったが、印象に残った質問が一つある。質問者は大学の人らしく、「クスmano氏の話によれば、マイクロソフ

トでは大学で一生懸命やってはいけないと教えていることを常にやっているのではないか」というものである。聴衆が少し湧いたが、マイクロソフトの技術や商売のやり方を批判したい気持ちはよく分かる。とくに情報として目新しいものはないが、マイクロソフトを持ち上げる内容の講演の後であってみれば。しかし一方で、筆者には変な質問のようにも思えた。そもそもクスmanoはほとんど技術の話をしていないので、ソフトウェア工学上好きからざることをマイクロソフトがやっているかどうか判断できるような内容はなかったと思う。だから、この質問は講演内容によらず質問者が先入観念として持っているものに違いない、というのがおかしく感じた理由の一つである。同調者も同様の発想であろう。また、この質問者の意図は、大学で正しく教えているソフトウェア工学からすると、マイクロソフトの開発プロセスは劣悪であるという主張であるが、皮肉にとれば、大学で教えているようなやり方ではソフトウェア企業としては成功しない、ともいえないことはない。質問者とその同調者がそれに気づかないのもおかしく思った。

5. これからのソフトウェア工学

ここを書く予定であった。実際書こうとしているのだが、理由があって少し先になる。そこで、最初にも述べたように、狡猾な戦略に出ることにした。このままでSEAMAILに出し、幸い何らかのフィードバックが得られれば、これから書くものに参考にしようというのである。というわけで尻切れとんぼのようだが、ご意見を賜りたい。

狩猟民族的技術者の世界

野中 哲

(Apple Computer)

米国のシリコンバレーに本社を置く A 社の日本法人に入社して6年半が経過した。この程度の経験で、米国のように多様な国についてなにか普遍的なことを述べられるとは到底思えないが、それでもシリコンバレーと呼ばれる地域のソフトウェアエンジニアの生態は多少理解したつもりである。それらの一部を紹介してみようと思う。

ほんの6年で半数以上の社員は私より後の入社になり、私は古参の部類に属するようになってしまった。米国の本社では近々、私にとって4人目の社長を迎えることになる。慣れたとはいえ、この人の流動性の高さには、驚くことが多い。

3年程前、まだ A 社の経営危機が現在ほど表面化していなかったころ、A 社は米国内で、数百人単位の大規模な求人広告をとある雑誌に掲載した。この求人には「基本ソフト第8版という次世代のオペレーティングシステムの開発のための募集」というお題目が掲げられていた。当時それを見た私は非常に驚いたものである。驚いた第1の理由は、A 社はそれを遡ることほんの2年程前に、大規模なレイオフを実行していたばかりだったから。第2の理由は、これだけ大量の人間を採用したとしても、次世代オペレーティングシステムがみごと完成した暁には、また大量に技術者が余り、大規模なレイオフが繰り返されるのが既定の事実のように思えたからだ。

このような状況で、最初にはたして技術者は応募してくるのやら、私は不審に思った。しかし、暫くしてこれは私の見当違いであることがわかる。どうやら、かれらは遠い未来のことはあまり考えないらしいのだ。採用する側はとにかく、目のプロジェクトを成功させることが先決で、その先のことを今あれこれ考えても意味がない、と思っており、また、職を求めてやってくる人間の側も、取り敢えずこの数年の間に、どのような経験を積み、それをどう次へつなげるか、ということのみを考えているようだ。

会社の将来性とか安定性などまったく眼中にない。未来のことはあまり考えない”と書いたが、これはやや不正確な書き方である。みんな無鉄砲ではないから、未来のこと、将来の人生設計は優秀な技術者ほどちゃんと考えている。考えていないのは、「将来自分の会社がどうなるか」ということについてである。自分の将来は考えても、会社の将来は考えない。

この状況は、私に昔見た西部劇を思い起こさせた。腕利きのガンマンが、西部のとある町に流れ着く。一宿一飯の恩義を受けたかれは、その町の悪者を退治して、最後にまたよその町へ流れていく、といったパターンのものだ。少々格好が

よすぎる気もするが、このガンマンの姿が、プロジェクトごとに会社を転々とする、技術者たちの姿と頭の中で2重写しになったのである。

教育という点に関していえば、会社が人材を育てるというケースはほとんどない。必要なスキルセットを持っている人を、どこから探してくるのがふつうである。人材の募集に関しては、必要な知識、技術、経験、担当する仕事に関して、かなり詳細な要求仕様が示されている。ただし、これは会社で教育を受ける機会が少ないということではなく、むしろ日本に比べるとかなり充実しているといってもよいのではないかと思う。社内のみならず、社外の講習会や、大学の短期的なコースに参加するプログラムが各種用意されている。しかし、上司の勧めや、命令で講習会に参加するということはまずなく、個人の自発的な意志で参加するケースが大半である。本人に本当にやる気があるかどうかを確認する手段として、たとえばコースの終了試験に合格した場合のみ、会社が授業料を負担するといった条件がついているものもある。会社がこのようなプログラムを用意するのは、もちろん、社員の技術水準を高めるといった目的もあろうが、第1の理由は有能な人材を会社に繋ぎとめるための、いわば福利厚生面の面が強いように感ぜられる。教育機会を充実させるのは、日本でいえば会社が社員のための保養所を建てるようなものではないだろうか。

人の異動が激しいので必然的に安定した組織が発達しない。組織変更は日常茶飯事だ。日本に比べて、会社の組織が平坦かといえば、必ずしもそうではない。平社員から CEO までの階層の数は、日米ともそれほどかわらないように思える。ただ、直属の上司の裁量の範囲は、比較にならないほど広い。極端に言えば「ボスが OK といえればなんでもあり」である。

プロジェクトにおける責任の所在は明確である。だから、なにか失敗があれば、それは特定の個人の責任になる。組織の責任という曖昧なものは存在しない。あるのはあくまでも個人の功績と失敗である。この個人主義は米国にかなり、普遍的であるようだ。医者でも、床屋でも、旅行代理店でも、客は担当者を名指しで指名する。その人のサービスがよければ客とチップはふえ、悪ければへる。したがって会社の方針というのは、実は社長の方針である。社長がかわれば、会社も180度変わる。

転職のうちおよそ半分は、よりよい待遇を求めて移る人々だったり、家族の事情などによるもので、これらは、いわば前向きの転職といえるだろう。しかし、あとの半分は与えられた責任を成し遂げることができず、夜逃げのようにやめて

行く人間たちである。製品のリリースが近くなったにもかかわらず、機能が実現できていない、あるいは山ほどバグがあり、とても動きそうもない、といったケース。その問題が火を噴く直前に辞めてしまうのだ。もちろん、そこはみなさん大人であるから「おめでとう、次の職場でも頑張る」と声をかけるが、同僚はたまったものではない。担当者が去ると「責任約束」もかれらとともにどこかへ行ってしまふ。新しい担当者が来て「それは前任者が勝手に約束したことです。私は知らない」というのは、よく聞かれる台詞である。口約束ほど当てにならないものはない。契約書が発達するのも当然だろう。

「今夜は残業してバグを直す」か「今夜は家族と夕食をとる」のどちらを選択するかも完全に個人の自由な選択にまかせられる。この選択に関して同僚や上司が干渉するのは、見たことがない。つきあいの残業など想像もつかない。ただ「米国人はみんな9時に来て5時に帰ってしまう」というのは嘘である。リリース直前は、やはりみんな必死でデバッグする。土日出勤、連日徹夜というのも日本人と同じ。ただそのあとかれらは、大胆に(一ヶ月とか)休暇をとる。これは大きな違いだろう。

従業員を採用するときは、かれらの言葉では'offerを出す'という。それを受けるかどうか、最終的な決定を下すのは個人だ。よって「あなたはめでたく採用の運びとなりました」ではなく「あなたがofferを受けてくれて、とてもうれしい」といういい方がなされる。これは「会社と個人がまったく対等な立場にある」という事実を象徴していると思う。

準備から収穫まで、最低でも数ヶ月を必要とする農耕に比較すると、狩猟での仕事の周期は、はるかに短い。何ごとを行うにしても結果がでるのが非常に早いのである。長期的な利益よりも、短期的な利益が優先される。台北の故宮博物館へ行くと親子三代かけて作った工芸品が展示されている。ヨーロッパへ行けば150年かけて建築された教会がある。狩猟民族にはこういうものは作れないだろうと思う。ものごとを考える時間の単位が、根本的に異なっているのだ。経営の目標の設定と評価のサイクルは3ヶ月が基本である。目標が達成されなければ、直ちに軌道修正のアクションがとられる。一年経っても改善されなければ、社長が首になる。まるでプロ野球の外人選手のようなだ。

ソフトウェアの開発でも、このような文化的特徴が反映されている。プロセスは重視されない。とにかく現物が尊重される。いわゆる概要設計書であるとか、詳細設計書であるといったドキュメントは、ほとんど書かれない。設計だけしてコーディングしないエンジニアなどはいないから、そのようなドキュメントの必要性はないのだ。設計はかれらの頭の中と、ノートの上に存在する。とにかく実際に動作する現物が尊重され、開発が決まれば、すぐにコーディングがスタートする。数ヶ月以内に何らかの形で動作するプロトタイプができ上がらないと、プロジェクトそのものがキャンセルされて

しまうからだ。プログラムの内容を解説するドキュメント、たとえばモジュール構成図のようなドキュメントが書かれることもまずない。開発方法論やCASEといったものについても、各技術者が個人的興味で使うケースを除いて、ほとんど使用されていない。組織的にCASEツールを採用して、開発効率を向上させようなどという話は、CASEブームのまっただ中のころでさえ、耳にすることはなかった。もっと、導入効果が速効的にあらわれるものでないと、狩猟民族の嗜好には合わないであろう。どちらかといえば「何を作るか?」を必死に考えている。「どうやって作るか」はあまり気にかけていないようだ。

一方、ソースコード以外のドキュメントが一切書かれないかということ、そのようなことはなく、大量に書かれ、かつ品質もよく、読みやすく、きちんと保守もされるドキュメントがある。それらは、主に外部から見える仕様に関連したドキュメント類である。これは、利用目的が非常に明確だからである。第一に企画書、提案書としての性格を持っているもの。これは外部仕様がきちんと書かれているかどうかで、そのプロジェクトにゴーサインがでるかどうかの判断材料となるからである。第二には、そのソフトウェアがライブラリであったり、API(application program interface)を持っていたりする場合、当然そのインターフェースを利用してくれるソフトウェアがふえてくれなければ、そのプロジェクトは成功とは呼べないので、そうなってくれるように、外部仕様書はきちんと書かれる。

企画書や提案書のたぐいは、雨後の竹の子のごとくに大量に書かれている。ふつうのエンジニアたちが自分のアイデアをどんどん提案書、企画書というかたちにして、周囲や上層部にどんどん売り込む姿には、当初はずいぶん驚かされたものだ。ちょっとでもチャンスがあれば、どんどんスピニングアウトして行く。みんな一応サラリーマンとはいえ、なにが個人商店の集まりのようにさえ思える。

さて以上を乱暴にまとめてみると

- ・定住しない。
- ・超個人主義の集団
- ・目先利益追求型

といったところだろうか。このような特徴が、変化の激しい、パソコンやインターネットの世界で強味を発揮した結果、現在のところ、シリコンバレーに生息する人間たちが、この業界をリードしている原因のような気がする。はたして、かれらのリードはいつまでつづくのであろうか?

最後に余談ではあるが、最初に紹介した'基本ソフト第8版'は、ついに完成することなく終わった。最近発売された'基本ソフト第8版'は名前こそ受け継いでいるが、内容は当初計画されていたものとは、まったく異なるものとなった。

ソフトウェアデザインワークショップ報告

— SD96 Workshop Report —

野呂 昌満 & 酒匂 寛
(SD96 プログラム・実行委員長)

第2回ソフトウェアデザインワークショップは、"仕様、設計から実装への流れを実践的に議論する"との副題のもとに、1996年11月28日(木)から30日(土)まで株式会社オムロンの保養施設であるLaLa GOTENBAを会場として開催された。本稿は、当該ワークショップに参加されなかった会員に内容の一端を紹介することを目的とする。

1. 開催までの経緯とワークショップの趣旨

第2回ソフトウェアデザインワークショップ(以下SD96と略す)開催の趣旨は、参加者募集に以下のように述べられている。

このワークショップは、とかく目先の実装技術(オブジェクト指向、イントラネット、etc)一辺倒に偏りがちな昨今のソフトウェア技術論の現状を見据えつつ、仕様(やりたいこと)と実装(できたもの)の間をきちんと埋めるべき「ソフトウェアの設計」を議論する場として企画されました。

オブジェクト指向万能のようにいわれている昨今の状況の中で、多くの技術者はソフトウェアの設計とはいかなる行為であり、どうすればよいかをじっくり考察することもままなりません。教科書に書かれた設計方法論を読みこなすにも、時間と個人的努力が必要です。

昨年度の本ワークショップでは"ソフトウェアの設計とはいかなる行為か"というテーマで議論し、広く深いソフトウェア設計の世界についての概要を垣間見ようを試みました。今年度は、さらにソフトウェア設計について理解を深めるために、議論主体ではなく、積極的に手を動かしてみようと考えています。すなわち具体的な題材を用いて、仕様の定義を行い、その仕様に基づくソフトウェアの設計と実装のプロセスを体験・理解・議論することを目標にしたいと思えます。

このような目的のためには、参加者が同じ土俵に上らなければなりません。それはとりもなおさず、簡潔でかつ内容的にもきちんとした記述方法をベースに議論する必要があります。現在の予定では、プログラミング言語MLを共通記述として、それによって仕様を定義し、さらに実装への橋渡しの過程を体験することで理解しようと

考えています。

なおMLに関する事前の知識は必要ありません！ワークショップの最初に簡単な講義を行います。これはあくまでも同じ土俵でソフトウェア設計の議論をするための手段に過ぎません。

多くの方々の積極的な参加をお待ちしています。

ここには、第1回ワークショップ(SD95)の後を受けて、さらに具体的に設計とはいかなる行為かを理解するために、MLを使つての設計を考えてみようと思われている。SD95は、"設計とはいかなる行為か"ということに的を絞った議論中心のワークショップであった。横道にそれるが、SD95の報告はそのプログラム委員長であった伊藤昌夫氏から近々このSEAMAIL誌上でなされると思うので、詳細の紹介はここでは控える。今回の実行・プログラム委員長であった酒匂・野呂2人ともSD95に参加し、それが終わった時点で、SD96は議論中心ではなく設計コンテスト等の"手を動かす"企画を含むものにしてはどうかとの提案が、酒匂さんからなされた。すなわち:

- CASE ツールを使って設計をしてみる
- Smalltalk-80 で実装まで考えてみる
- Java でプログラミングする

等々可能性のあるものは広く考慮の対象とした。

その結果、

- 1) 高水準の計算モデルを持つ
- 2) 詳細設計仕様記述に向く
- 3) 抽象データの記述が可能である
- 4) ポータブルで無償の処理系が存在する

などの理由により、MLを設計について議論する共通の基盤として選ぶことになった。

2. なぜMLだったか？

SD96の目的は、実際にソフトウェアの設計作業を疑似体験して見ることを通じて、"設計とはいかなる行為か？"ということを理解しようということであった。

企画時点での、酒匂・野呂の打合せでは、"SD96 で大きな視点から (top-down に) 設計について議論をしたことを捕捉する意味で、実装 (プログラミング) の視点から (bottom-up に) 設計について考えよう、"

これに基づいて、実習 (手を動かす作業) を行なうならば、以下の機能を満たすソフトウェア ツールが必要になると考えられた:

- 1) プログラムの仕様記述
- 2) データ構造, アルゴリズムの設計・記述
- 3) モジュール (抽象データモジュール) の定義

さらに、絵に書いた餅で終らせないためには:

- 4) 設計結果が容易に実装に結びつけられる

ことが要求される。

以上の条件を満たすものとして ML が選ばれたのである。

ML は関数型プログラミング言語である。その計算モデルは参照の透明性が保証され、副作用がないことを前提に定義されている。すなわち、同じ記号に束縛される値はあるプログラム部分の中では同じものである。われわれの多くが慣れ親しんでいる、フォンノイマン型コンピュータのハードウェアを素直に抽象化した命令型の計算モデルは、副作用を前提としている。ここで副作用とは、ある変数に値を代入しその値を後から使うことを指すが、命令型の計算モデルではそれは意図した行為なので、副作用と呼ぶのはふさわしくないかも知れない。命令型の計算モデルは、フォンノイマン型コンピュータから見れば自然なものであるが、人間から見れば関数型の計算モデルの方が自然である。

アルゴリズムの記述に関しても、関数型の計算モデルでは、再帰を用いて繰り返しに関する計算手順を記述する。命令型の計算モデルにおける繰り返しは、再帰式の不動点を繰り返し内の不変式として記述したものになる。命令型プログラミング言語で繰り返しを記述するさいに現れる繰り返しの制御変数や、繰り返し脱出制御のためのスイッチ変数などの、計算 (問題解決アルゴリズム) の本質とは関係ないものは、関数型プログラミング言語では必要がない。

抽象データモジュールが定義できることも、ML の大きな特徴の1つである。SD96 では、与えられた問題を抽象データ設計法またはオブジェクト指向設計法に基づき設計・実装することも、視野に入れていた。その抽象データ定義機能を用いて問題の解を記述することも念頭に置いて、ML が選ばれた。

3. 設計課題

SD96 では、2つの設計課題を出し、それらを各グループが選択し解くという形式で、会議が進められた。課題の1つは、前代表幹事の山崎利治さん作の"酒屋の在庫問題"、もう1つは"グラフ上で最短経路を求める問題"であった。

前者は、その原典が山崎さんの手で、参考文献 [1,2] に書かれている。ここでその内容をそのまま書くことは、著作権の関係上好ましくないので、その要約だけを載せる。詳細は原典を参照されたい。

酒屋の在庫問題要約: ある酒屋では、コンテナで複数銘柄のお酒を搬入し、単一銘柄の出庫依頼を受けつける。毎日数個のコンテナの搬入、数十件の出庫依頼がある。コンテナはその番号、搬入日時、銘柄とその数量の組の集まりを記述した積荷表とともに搬入される。出庫依頼は銘柄、数量、および送り先名を添えて行われる。受付係と倉庫係がいて、受付係は、在庫なしの連絡、出庫指示書作成および在庫不足リスト作成を行なう。倉庫係は、搬入のさいには積荷表を受付係に渡し、お酒を倉庫に搬入し、受付係からの出庫依頼に従ってお酒を倉庫から搬出する。出庫依頼は、出庫指示書または電話による。受付係の仕事を自動化するシステムを作る。

グラフ上で最短経路を求める問題に関しては、弧に重みのついた有向グラフの単一出発点からすべての節点 (node) までの最短距離を求めるアルゴリズムが Dijkstra によって提案されている [3,4]。このアルゴリズムを使って最短距離を求めるプログラムを作ることを課題とした。

アルゴリズムの要約はつぎの通りである:

1. 出発点節点を注目する節点とする。
2. すべての節点を走査するまで以下を繰り返す。
 - 2.1 注目する節点を走査済みとする。
 - 2.2 注目する節点からすべての隣接節点までの距離を求める。
 - 2.3 未走査節点から最短節点を抜きだし注目する節点とする。

詳しくは参考文献 [3,4] を参照されたい。

4. SD96 進行概略

SD96 で行なったことの概略を時間を追って示せば、以下の通りである:

11月28日

13:00 - 18:00 ML の講義, 講師佐原伸 (SRA)

抽象データモジュールの定義方法を中心に ML による関数仕様の記述方法,

文法とその意味について例を用いて説明。

11月29日

9:00 - 18:00 グループに分かれて ML で設計

複数グループに分かれて上記2課題の内1つを選んで設計・実装。

19:30 - 22:00 成果発表・議論

各グループの設計過程および結果について発表内容について議論。

11月30日

9:00 - 12:00 SD96を振り返って

各グループから代表者を出し、SD96に参加しての感想と意見を含め設計について議論した。

5. 設計をしてみよう

5つのグループに分けて設計・実装を行なった。グループ毎にもプログラム委員から責任者を決めて指導的役割を割り当てた。グループの構成員については付録2に示す。

3つのグループがグラフの最短距離問題を1つが酒屋の在庫問題を設計・実装した。もう1つのグループは全体の監督役ということで実際に課題を解くことは行なわなかったが、作業中に助言を与えたり、まとめの発表を行なったりとの役割を果たした。

グラフの最短距離問題においては、グラフを表現するデータ構造をどのようにするかが鍵であった。グループ1は節点に番号をつけそれを参照 (reference) にする形式でグラフを表現した。グループ2は節点を、隣接点をたどる等のメソッドを持つオブジェクトと考え抽象データ型として実装した。グループ4はグラフの弧を出発点と到達点距離の3つ組で表現する方法をとった。関数型言語でのプログラミングという観点からはグループ1とグループ4のデータ構造表現方法が適切であり、実装も素直に行なえたようである。一方グループ2のアプローチは汎用性を狙い過ぎたあまり問題が大きくなり過ぎ結局実装には至らなかった。

グループ3は唯一酒屋の在庫問題を解いた。課題文を読んでオブジェクトを抜き出すという、オブジェクト指向設計のアプローチに基づき設計を試みた。しかし、オブジェクトの同定が不完全なままコード記述に移り、力任せに完成しようとしたことがたまたま、実装は未完成、設計は手続き指向に基づいたものとなってしまった。

6. まとめ

以上述べたように、一見、よく定義された問題を設計・実装してみただけのワークショップであったかのように考えられるが、日頃開発現場で直面するようないくつかの問題が

SD96でも起こったようである。以下例を挙げる:

- 1) 設計が不完全なまま実装に移ろうとすると多くの歪みがコード上に現れる。それを力任せにねじ伏せようとするとさらに悪い結果になる。
- 2) 問題の大きさに応じた解決法をとらないと、結果として不要な人月を必要とすることになる。

一方で計算の本質的なところだけを考える習慣をつけ、それを自分の本能 (second nature) にしてしまえば、実装を通してよい設計が見えてくるのではないかという期待も持てた気がする。ワークショップでも実務に ML を使えるか否かの議論があった。しかし、そういうことよりも、ML のような"よい"関数型プログラミング言語でのよい実装方法を自分の技術とすることで、プログラミング言語に捕らわれない問題解決・設計技術を持つ素養が身につくと考えられる。

SD96では実装に力点が置かれ、設計に関する議論ができなかったとの意見もあったが、昨今の"分析・設計が重要であり実装は付け足しの人足仕事である"というような風潮に一石を投じることができたのではないかと考えている。プログラミング技術の軽視、ソフトウェア工学研究におけるプログラミング言語研究が軽んじられる、という風潮があるが、これは危険である。実はプログラミング言語がその背景に持つ計算モデルは問題の解決方法に少なからず影響を与える。また、計算モデルに基づく言語機能がよい解決方法は何かというソフトウェア開発の根幹に関わる問題の答えを示唆しているのである。

最後に、このような報告書はワークショップ終了直後に書くものであるということが身に染みて解った。"設計してみよう"の一部の詳細として ML による最短経路問題の解決に関する解説論文を佐原さんが近々 SEAMAIL に寄稿される予定である。技術の詳細の一端はそちらを読みたい。SD97を含め、今後のソフトウェアデザインワークショップの発展を祈りたい。

参考文献

1. 山崎利治他, "共通問題によるプログラム設計技法解説 (その1)", 情報処理第25巻9号。
2. 山崎利治他, "共通問題によるプログラム設計技法解説 (その2)", 情報処理第25巻11号。
3. E.W. Dijkstra, "A Note on Two Problems in Connection with Graphs", Numerische Mathematic 1:269-271, 1959.
4. 島内剛一他編, アルゴリズム辞典, pp.455 Dijkstra 法 (最短経路) の項, 共立出版, 1994.

付録-1 参加者の主張 (position paper)

赤間智朗 (ネクストファウンデーション)

(1) これまでに開発してきた主なソフトウェア・システムは?

HDLC 通信制御
ATM 交換機 (NTT 試作版) 制御
OpenLook GUI ビルダー

ページ記述言語開発

(2) ソフトウェア設計とは何をすることか? また、実際どのような行為を行なっているか?

ソフトウェア設計とは抽象化されていたもの (企画, 構想) をより具体化させるための工程であり, 通常はこの工程で必要となる仕様書は全て作成されるべきであり, この後の工程 (実装) で円滑に作業を行なわせる上で非常に重要なものと認識しています。

(3) その他, ソフトウェア設計に関して考えていること (何でも自由にお書きください)

現在, 各種の支援ツールや言語等が出回っていますが, このようなツールや言語等を使用する前に, まずどのようなことをやろう (設計しよう) としているのかを明確にする必要があります。当然といえば当然なのですが, ツールや言語仕様の理解ばかりに時間をとられ, 設計自体がおろそかになるケースも少なくないのです。

あと, 各自のコミュニケーションの少なさという問題もあります。よほど, 簡単なプロジェクトでない限り, 設計作業は複数人数 (例外もありますが) で行なわれます。作業を円滑に進める上でもコミュニケーションをとることは重要であると思います。

昨年のワークショップでは, さまざまな現場担当者からの活発な意見がだされ, 1年たった今, どのような改善がなされているかが非常に興味のある所です。

朝倉鉄也 (リコー)

(1) これまでに開発してきた主なソフトウェア・システムは?

ページプリンタの組み込みソフトウェア

(2) ソフトウェア設計とは何をすることか? また、実際どのような行為を行なっているか?

実装されるハードウェア, OS 等の制約を考慮して, いかに要求仕様を満たすかを決定する工程。それにより, ソフトウェアの構造, 使用するアルゴリズム等を選択する。

(3) その他, ソフトウェア設計に関して考えていること (何でも自由にお書きください)

現場では常に納期に追われていて, 分析, 設計の工程を省略しがちである。管理者も開発者も, 開発期間が短いと, すぐにコーディングに入りたがる。そのため, ソフトウェア構造が悪くなり, テスト工程でバグが収束していかないという結果に落ちている。しかし, そういった開発工程はすでに定着してしまっていて, 変えるにはかなりの労力が必要であり, 簡単ではないと思う。

上田鉄雄 (中日電子)

(1) これまでに開発してきた主なソフトウェア・システムは?

組み込みシステムのソフトウェア

負荷心電計 (ストレステストシステム) のデータ構造設計, まとめ, 保守, サポート。1部のモジュールの作成。4機種ほど (現在も作成中)。

麻酔 (血圧) プロッタのファームウェア。血圧計との通信部, 全体構成のまとめ。1機種のみ

68000, 68030, 68031.

SH7043 (現在進行中)。

C およびアセンブラ。

その昔は, 8085, 8086, ASM80, ASM86, PL/M, PL/M86, S/PLH も使っていました。

日の当たることのなかったいくつかの開発品のファームウェア

市販 AD ボード + 専用 AMP + 制御 BOX + DOS/IV 機による 300ch 心電収録機 (単品)

市販 AD ボード + PC-98 によるアナログデータ収録機 (単品)

いづれもハード制御と簡単な収録 & 表示のみ

MS-C 6.0, VC++ 1.0

(2) ソフトウェア設計とは何をすることか? また、実際どのような行為を行なっているか?

顧客とのあいだで要求仕様 (ハード, ソフト込みの仕様) をはっきりさせること。どのような, ハード, ソフトとしたら, 要求仕様を満足させられるか考えること。

ハードウェアの設計段階から, 口出しして, どうやったら, 性能がでるか, 考えること, 制限のなかで, できるだけたくさんの CPU パワー, メモリを使えるようにしてもらうこと。

機能のモジュール分け, 階層分け。

開発環境の整備: 232c デバッグ, クロスコンパイル環境。

ハードウェアテストプログラムの作成。

モジュール間 I/F の設計。

各モジュールの設計 (担当者ごと)。

モジュールを組み合わせてのデバッグ。

評価。

顧客評価。

ドキュメント： トップダウン & ボトムアップで作成しています。

- (3) その他、ソフトウェア設計に関して考えていること
(何でも自由にお書きください)

現在の課題は： 自社におけるソフトウェアの正しい見積りの仕方、自社用のソフトウェア部品化(組み込みシステム用)、ISO9001 対応。

組み込みソフト開発環境、もっと低価格にならないものか。市販 RTM のライセンス料、少量生産に向いていない。

毎回専用ハードに作り込みなので、工数がすごくかかっている。実際には、ハードウェアの設計工数の数倍に達している。これは、正しいことか？

上田秀樹(オムロン)

- (1) これまでに開発してきた主なソフトウェア・システムは？

ATM のソフトウェア移植: アーキテクチャが違うコンピュータへの移植のため、自動変換ツールを C++ を用いて開発している。

- (2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行なっているか？

まだ私の中では漠然としていて抽象的ですが、考えていることは、与えられた仕様に対して、それをモデル化すること。また、そのモデルに対して、論理的構造に組み立て上げること。

- (3) その他、ソフトウェア設計に関して考えていること
(何でも自由にお書きください)

ソフトウェアの開発能力は人によって違いますから、当然理解力も異なるでしょう。そのギャップをうめるのは、個人の努力でしかないのでしょうか？ センスが絡むのでしたら、個人の能力になってくるので、どうしようもないのかとも思っています。

また、動けばいいソフト開発というのは、結局避けられないのでしょうか。もちろん納期の関係で妥協しなければならない点があるのは理解できます。ただ、後々のことを考えると、後から利用できるようなことを埋め込む必要もあるのではないかと、とも思います。

とりとめがありませんが、他の人がどのように考えているのかも知りたいです。

小田朋宏(SRA)

- (1) これまでに開発してきた主なソフトウェア・システムは？

流体の熱伝達のシミュレータ。

- (2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行なっているか？

システムの仕様記述を基にしてシステムの構成要素とそれぞれの構成要素の責任を明確にすることによって、そのシステムがどのように問題を解決し、処理するかを定義すること。

- (3) その他、ソフトウェア設計に関して考えていること
(何でも自由にお書きください)

実行可能仕様の構成要素延長でシステムの構成要素を大きく変更しながら実際に動作させることができるような設計環境が望ましいと思う。

加藤隆夫(エドモンドビジネス)

- (1) これまでに開発してきた主なソフトウェア・システムは？

経理・財務。

資材管理。

人事・給与。

- (2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行なっているか？

業務ソフトウェアを前提として、“設計”をシステム設計とソフトウェア設計とプログラム設計という分け方にした場合。ソフトウェア設計とは、コンピュータという機械と特定の業務(システム)を結びつける橋渡しの役割をするものを品質とコストを意識してデザインすることである。

- (3) その他、ソフトウェア設計に関して考えていること
(何でも自由にお書きください)

1) 組織向けから個人向けへ

2) 部品開発と組立製造の分離思想の導入

3) ネットワーク市場に乗せられるような設計

栗原茂(PFU)

- (1) これまでに開発してきた主なソフトウェア・システムは？

日本語入力ソフト

Lisp 処理系の移植開発

構造化言語の開発ソフト(YPS の UNIX 版)

- (2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行なっているか？

再利用性、一貫性、簡易性、効率性を考慮し、product

の青写真をきちんと定義し、下流工程に繋げることだと思っています。

最近では設計を行っていませんが、再利用を前提としたオブジェクト指向技術の導入が必要であると思っています。

(3) その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

ライフサイクルにおいて、設計フェーズは、つきつめると最もコスト的に重要であると認識しています。また、設計のよし悪しは、設計者の能力やさじ加減により大きく左右されると思っています。つまり、設計フェーズでは、いまだに人間の頭に頼っている部分が大きいと思っています。この知的作業を支援する、ある程度の自動化を含む smart な設計支援ツールや、設計の善し悪しを定量的に測定できるメトリクスが特に重要だと思っています。

繁田雅信(富士電機)

(1) これまでに開発してきた主なソフトウェア・システムは？

- 1) 飲料自動販売機制御ソフトウェア
- 2) 自動調理装置制御ソフト
- 3) 自販機設計支援・合理化システム
(図面作成合理化)
- 4) 組み込み機器用 OS
- 5) 自販機ソフト開発支援システム

(2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行っているか？

ソフトウェア設計とは： 機能に対する要求(多くの場合情報が少ない)を的確に分析・把握し、最適な実現手段を表現すること。

具体的には、

- (1) 要求(課題)の発掘(モデル化)
- (2) アーキテクチャ、アルゴリズムの考案と決定(表現)
- (3) 実現のプロセスの設計
- (4) 実生産であるプログラミング工程に(1)(2)のプロダクトを的確におくこと

実際の行為： オブジェクト指向中心の開発をおこなっている。(下記番号は上記番号に対応)。

- (1) OOA(主な手法としては Booch, OMT)による要求のモデル化
- (2) 上記の延長で OODによるアーキテクチャ、アルゴリズムのモデル化
- (3) OOにより開発における情報モデルを作成し、整流、合理化視点で再構築
- (4) 抽象データによるプログラミングを励行し、OODの結果をできるかぎり忠実にプログラムに反映させ

ている。

(3) その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

元来ソフトウェアは目にみえないことを理由に実行されて初めて判断される傾向にあると考える。現在、オブジェクト指向中心で開発をおこない、モデル化で見えるようにしようとしているが、思ったように効果がでない。原因としては、

- 1) モデル、設計者、レビューの質がなかなか向上しない。
- 2) 企業が望む(管理のしやすさ、実は錯覚)決められた工程間(ウォータフォール)でチェックしようとする。と抽象化されているはずのモデルが情報量過多となり、判断しづらくなる。

今後の方向として、

- 1) モデル、設計者、レビューにたいする教育の方法論または支援
- 2) 開発プロセスのビッグウォータフォール→多回転スパイラル化を進めていきたい。

嶋中徹(PFU)

(1) これまでに開発してきた主なソフトウェア・システムは？

インストーラ(オフコン)
現在は画像処理屋(修行中)

(2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行なっているか？

目的の集合Aとその具現化たるソフトウェアの集合Bという2つの集合を考えた場合に、集合Aから集合Bへの1対多写像から、拘束条件(金、人、環境、保守、etc..)下において最適(何が最適なのだから…)となるような写像を1つ選択する作業。

実際は目的自体の変更がしばしばであるため、或る写像を設計段階で決定することは困難！！

(3) その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

考えたことがない。

関口直利(SRA)

(1) これまでに開発してきた主なソフトウェア・システムは？

電気メーカー部品管理システム
衛星データ地上利用システム
ガス会社自動検針システム

(2) ソフトウェア設計とは何をすることか？また、実際どの

ような行為を行なっているか？

- ・こういうプログラムを作るよ。いいですか？と確認するもの。
- ・お客様の要求と作成するシステムの機能差を説明／合意するもの。
- ・システム実装の最適化を考える作業。

(3) その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

設計方法や手法は、開発する環境(主に人)によって様々であるので、いろいろな方法の共通点とは何かを考えてみたい。それを見つけることによって方法が変わっても最適な設計ができると思う。

また、設計の重要性や必要性の認識についてエンジニアとお客様の間で異なることがあるが、予算や技術の未熟さなどによって必ずしも合意できて仕事をするのは稀である。その際の事実の説明力と説得力を見つけたと思う。

高橋正一 (SRA)

(1) これまでに開発してきた主なソフトウェア・システムは？

UNIX をリレーショナル・データベースのサーバとし、Windows パソコンをクライアントとする、業務向けクライアント・サーバシステム。

(2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行なっているか？

ユーザの雑多な要求を取りまとめ一旦抽象化し、目的と手段を明確に分離する。さらに、情報の流れと機能、人の役割を明確にし、物理・論理の両面で実現可能なレベルにまで詳細にアーキテクチャを定めること。

実際には、順に、概念定義、要求定義、基本設計、詳細設計を行なう。各作業で実現可能性を検証し、問題があれば前作業を再度行なう。

(3) その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

設計に関する理論や手法は巷に数多くありますが、実際に、開発現場で採用されることは少ないです。多くの場合、忙しさに紛れて、一部の人の経験にたよって、スケジュール優先で進められるのが現状です。この状況が、少しでも改善されればと考えています。

私の担当プロジェクトですが、基本設計から始めて半年間やってきたことが、実現不可能ということで、全て白紙に戻して、要求定義からやり直しになりました。上流工程の重要性を、しみじみと実感する毎日です。

出野徹 (オムロン)

(1) これまでに開発してきた主なソフトウェア・システムは？

機器組み込み型ソフトウェアの開発です。より具体的には、金融機関向け自動入出金機(ATM)のドライバソフト・アプリケーションソフトを開発しました。

(2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行なっているか？

私共の会社・部門においては、「設計」とは機能設計～プログラミング完了までを意味します。また「設計者」は普通、設計したソフトのデバッグまでを担当します。

「設計」とは与えられた要求仕様を最適な品質(プログラム容量・処理速度・変更容易性・納期等々)で実現するためにソフトウェア全体を分解し、配置し、組み立てる行為すべてと考えます。

(3) その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

設計者は要求仕様の分析者やプロジェクトの管理者より低レベルの仕事とみられており、入社後数年の経験を経てようやく「よい設計」が可能なレベルに到達するやいなや分析・管理の仕事に担当が変わります。有能な設計者にそれなりのキャリアパスを用意することと早く一定レベル以上の設計ができるような方法論を見出すことの両面がよい設計によって作られたソフトを蓄積するために必要と考えています。

友枝敦 (SRA)

(1) これまでに開発してきた主なソフトウェア・システムは？

ラスタグラフィックスレイトレーシング、テキストチャーマッピング

アナログLSIレイアウトCAD回路図→LSIレイアウト図展開

ソフトウェア自動合成システムフレームによる内部表現、Reader/Printer

Smalltalk 動的メトリクス計測ツール

(2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行っているか？

問題を具体的に解決するためのアーキテクチャを考案するところだと考えます。で、「解決」ってえところが肝です。様々な状況というか制約というかを考慮に入れながらやんなきゃなので、一筋縄ではないですね。さらに解決する対象や解決された後の望まれる姿も曖昧だったりしますし、解決のための資源時間は限られてるし、そりゃもうてんやわんやですわ。

で、僕としては有史以前の仕様確定を前提とする「設計」はナンセンスだと考えます。なんたって、新規開発に

は公然の秘密：

(<http://www.sra.co.jp/people/tomoeda/reuse/witchcraft/productivity/secret-work.html>)

がありますからね。

だからコンビシステムは例外として,"設計"は変更を前提とする設計を行うことが肝要と考えます。この考えを押し進めていくと、仕様の変更に対する強さ免震性や再利用に向かうのは自然の成り行きです。従ってこれからの時代の"設計"は免震構造や再利用構造(!?)を考案するところと捉えます。

で、僕としてはできるだけ問題(領域)のヘソを捉える様に意識/努力していると。その具体的内容は... う〜んどうしてんだろ。よくわかりません。^;ただかようなことをやっていると当然のことですが、立ち上がり時にコストがかかることがネックですね。でもまあ私は同じ蕎麦でも、駅前の立ち食いよりも評判のいい手打ちを食べたいし、世間もそうあって欲しいと思う今日この頃なのであります。

(3)その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

#さ~再利用するぞ!

(2)に関連して,,,オブジェクト指向に期待しています。現状のオブジェクト指向がこの問題に対する解を与えているとは考えないが、最も近似解であると思われるので。

なぜなら、再利用のためには"ソフトウェアに関する知識"を表現する必要がある。関数などに比しオブジェクト(フレーム)はソフト知識の表現形式として優れている。

私は、オブジェクト指向は再利用(ソフト資産化)へと向かうものと捉えている。つまりオブジェクト指向は次のような進化を遂げていくと考える。

第一世代：再利用に適したソフト表現(いわゆるモデル化)

第二世代：ボトムアップ、汎用的(水平)部品利用

第三世代：トップダウン、専用的(垂直)部品利用

第四世代：これらに加えプロセスの知識が統合、組織化

第五世代：第四世代に形式技術A I技術が融合した開発環境

大雑把に見て現状は、OOA,OODが第一世代、OOPが第二世代、デザインパターンが第二、五世代(抽象部品化)フレームワークが第三世代に入らんとするところ、といったところであろうか。

個人的には、第三世代に移行する際に重要になるのが、クラスからアーキテクチャへの視点の変化だと考えてい

る。なお、ここでいうアーキテクチャとはオブジェクトモデルに代表されるような各種モデルやコードなどの実現の仕掛け、記述といったものである。

第三世代では第二世代での抽象クラスに対し、抽象アーキテクチャがある意味で重要な関心事、ということになる。これに伴い継承も、属性やサービスという一次的なものからアーキテクチャという二次元的なものへと移行、進化する必要が有る。

この時、アプリケーション(のアーキテクチャ)がそのドメインという方向に抽象化された抽象アーキテクチャが、ドメインモデル、いわゆる俗にいうフレームワークということになろう(個人的にはこのフレームワークの定義は気に入らないが...).

当然第四世代では第三世代ドメインモデルに、ドメイン固有の開発のやり方やノウハウ、方法論(!?)が加味されることになる。

友田稔(ダイキン工業)

(1) これまでに開発してきた主なソフトウェア・システムは?

UNIX(SVR4 など), X-Window のポーティング
UNIX の OS の改造, 各種デバイスドライバ開発
CAD システムの開発 (WindowsNT/95 対応)
社内ツールの開発 (WindowsNT/95 対応)

(2) ソフトウェア設計とは何をすることか? また、実際どのような行為を行なっているか?

ソフトウェア設計とは何か? (エンドユーザが触れるアプリケーション開発の場合):

対象を理解し、コンピュータ化する物の抽出(理解分析)

ソフトウェア構造の検討(プログラミングのための分析設計)

プラットフォーム、開発環境の利用方法の検討

実際どのような行為を行なっているか

オブジェクト指向(OMT)手法で、対象を分析

プラットフォーム(開発環境)の利用方法の検討(工夫)

最近行なっているソフトウェア設計の傾向として、Windows 専用のアプリであるにもかかわらず、プラットフォームに依存する部分と非依存部分を完全に分離し、その掛け渡しとして仮想の機能(インターフェース)を設定して、プラットフォームを意識しない設計(ただし、言語には多少依存)とプラットフォームの利用を検討する設計に分けて作業している。

(3) その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

ソフトウェア設計とは、要求(対象)と計算機をどう結び付けるかで、現在、OOなど対象を扱うことがもてはやされているが、それと同じレベルで、プラットフォームの利用技術や、言語に関する知識、開発環境などが重要であると考えられる。実際、OOが分かっているでもこれらが役に立たない。さらに、着眼点や、アイデア(利用技術)の絞り出し方も重要で、これらの技術の育成などがむずかしい。

現状、仕事の関係で、Windowsの利用技術を含めた設計をOOを用いて行っているが、以前の仕事のデバイスドライバ開発や、UNIXのポーティングなどでは、対象に対するソフトウェア設計というより、与えられた環境(ハードやソフトウェアのAPI:インターフェース)の利用の検討(利用技術の追求)がほとんどであった。

ソフトウェア設計の技術を語る時、一般に暗黙のうちにビジネス系ソフト(OSの上で動作する、最もユーザーに近いアプリケーション)をベースに語られているが、インフラ側のOSそのものやミドルウェアなど、ジャンルによって、手法や、考え方、実際の作業方法はかなり異なっているように思う。これらジャンルでのソフトウェア設計も議論してみたい。

方学芬(SRA)

(1) これまでに開発してきた主なソフトウェア・システムは？

言語解析(COBOL, JCLなど)システム
リバースツールのGUIシステム

(2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行なっているか？

ソフトウェア設計はモデリングとモデル構築の問題だと思います。個人として図を書いたり、メモをとったりして、モデルをはっきり見えるまで、繰り返します。明確できたモデルは正式に図式にしています。その人間精神能力を半分放棄していると思いつつも、適切な記述法もなし、重視されていないので、そうしかないとあきらめています。コードさえできれば、どう成り立てたのは開発時点で誰も関心がありません。

(3) その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

- 有名な設計方法はいくつか提唱され、そのソフトウェア設計方法の選択を取り巻いている宗教的な熱狂がありますが、本当に万全でしょうか。万全でなければ、設計支援ツールの有用性に疑問があります。
- 設計の目的は、複雑なことを簡単なものにしたいですが、簡単になれるはずがないために、簡単に見える仕組みもつけるのでしょうか？

3. いろんな設計方法を検証して、その方法に適用な対象領域を分類して、これらの方法はお互いに補い合い、増強し合うこともできます事例を会社の財産として蓄積しているところがありますか。
4. モデルを記述手段として、メモや図式なんでも、いいです、自由にシステム構造のモデルを簡単に描くことができ、それは設計の起源だと思います。それを支援するツールをご存知ですか。

堀茂樹(富士電機)

(1) これまでに開発してきた主なソフトウェア・システムは？

自動販売機の制御, OS-マルチタスクリアルタイム OS, アプリアセンブラ. 8ビット CPU, ステップ数約6万. 処理内容は、通信を用いたマスタースレーブ制御・ダイナミック I/O 制御のシーケンス制御・LCD/キーボードの MMI.

自動販売機の制御, OS—マルチタスクリアルタイム OS, アプリC言語, 16ビット CPU, ステップ数約15万. 処理内容は、通信を用いたスレーブ制御・I/O 制御・データベース制御.

自動販売機の制御ボードのテスト装置, PC98を用いたテスト装置, OSはMS-DOS, アプリはC言語, ステップ数は約3千, 処理内容は回線制御・I/Oのシーケンス制御.

(2) ソフトウェア設計とは何をすることか？また、実際どのような行為を行なっているか？

[1] 位置付け

顧客と営業の仕様打合せ -->
制御仕様書作成(担当業務) -->
ベースソフトに新規・変更分を組み込み -->
テスト機でブラックボックステスト -->
出荷

自販機の機種と仕様の相違

↓仕様 機種→	機種_1	...	機種_50
仕様_1	AをBする		AをCする
:			
仕様_n			

[2] 何をすることか？

— ソフトウェア設計の1側面はお客様の要求を仕事に変換すること → カップ自動販売機(カップに飲料が入って出てくるタイプ)の場合、「粉原料をどう混ぜて溶かすか」がノウハウで顧客により要求が変化します。例えばコーヒーとクリームを同時に混ぜたいお客様もいれば、クリームは後から混ぜたいお客様

もいる。これらの要求をまともに聞くと顧客毎にプログラムを作る必要がある。これをもとにもやると作る労力・管理・試験の労力で利益がなくなる。この要求とやることのバランスをとるための仕様を作成するのがソフト設計と考える。

そこで、リソース(ベースソフト)を見て要求の内容をまとめプレゼンし、(相手は営業)OK となって制御仕様を作成する。また、実験が必要なものは実験計画の中でブラッシュアップしている。問題としては、お客からの仕様変更・我々の理解不足・我々のリソースがどうなっているのかが理解・整頓・分析されていなくて、もので確認する段階で変更に変更を重ねている。

- ソフトウェア設計の2番目の側面は機械のクセを織り込むこと → 自販機で飲料を調理する時いろいろな原料を混ぜる。シロップ飲料(例えばコーラなど)では炭酸水・冷水・シロップ・氷が混ぜ合わされるが、この順番・タイミングが自販機の種類によって違って来る。このクセを顧客から隠し、制御の中に閉じ込め異なる機械でも同じ味になるように仕様を作るのがソフト設計と考える。

これは前述の内容と矛盾しており、この調整は個人のスキルによってやり方が変わる。この個人のスキルを均一にすることがおおきな課題と考えている。

- 3番目には過去の資産(ベースソフト)を生かしながら新しい機能を付加することを考えること → 自販機のソフトは「ベースソフト+新規仕様-時代遅れ仕様」からなり、ベースのソフトは今までのノウハウの積み重ねの結果であり、大事にする必要がもちろんあるが、新しい機構・機能にも対応する必要があり、このバランスをとることが必要になる。そこでまずやっていることは、ベースソフトの仕組みをソースから追ひ、その上で制御仕様に逆展開している。制御仕様の作成者、それを受けてプログラミングする方の両方の担当は固定しておらず、最も信頼できるドキュメントから新しい制御仕様を作る。

- (3) その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

現在、自販機のソフトドキュメントで大きなものに「制御仕様書」「プログラム設計書」「チェックリスト」の大きく3つある。

これらは各々独自の視点(あってないようなもの)で作成されており、ドキュメント上は何の因果関係もないがそれぞれがそれぞれを参照して作られている。そのため、プログラムのあるモジュールは複数の「制御仕様書」をみなければ作成できず混乱状態にあるといえる。

少なくとも「制御仕様書」「プログラム」「チェックリスト」の構成を同一化し、連携の取れる形にし、仕様書の変更がプログラムに反映されるようにしていきたい。

松島弘幸(リコー)

- (1) これまでに開発してきた主なソフトウェア・システムは?

入社2年目なので、まだシステム開発の経験はありません。ソフトウェアの開発支援に関する研究とプロトタイプ開発を行っております。

学生時代は、ロボットビジョンのシステムを開発しました。

- (2) ソフトウェア設計とは何をするのか? また、実際どのような行為を行なっているか?

現在、オブジェクト指向言語を使用しておりますので、ソフトウェア設計=クラス設計ということになりますが、開発したクラスをフレームワークとして提供できるよう、再利用性、モジュール性を重視しております。また、クラスの役割を明確にした設計も重要であると考えております。これに関してはデザインパターンの考え方を参考にして、取り入れております。

- (3) その他、ソフトウェア設計に関して考えていること(何でも自由にお書きください)

オブジェクト指向においては、分析→設計→実装という開発フェイズをシームレスに行うことができるのが利点であるといわれますが、実際には各フェイズの間の溝は大きいように思えます。また、ソフトウェアの再利用性を考えた場合、よい分析を行えばそれで再利用が得られるかということこれも疑問で、私はよいフレームワーク設計がキーポイントになると考えております。

付録-2 SD96 グループ構成表

- グループ1: 佐原伸(責任者)
赤間智朗 加藤隆夫 関口直利 友田稔
- グループ2: 小田朋宏(責任者)
朝倉鉄也 栗原茂 繁田雅信 方学芬
- グループ3: 野呂昌満(責任者)
上田鉄雄 高橋正一 出野徹 松島弘幸
- グループ4: 伊藤昌夫(責任者)
上田秀樹 久保田益則 友枝敦 堀茂樹
- PCグループ: 山崎利治(責任者)
岸田孝一 熊谷章

「組込みソフトウェア分野のプロセス改善活動に関する情報交換会」報告

高木 徳生 (*1)、 小泉 毅 (*2)

(*1) オムロン株式会社 ソーシャル事業グループ 開発・生産センタ ソフトウェア開発部

(*2) キヤノン株式会社 ソフトウェア技術開発部

1. はじめに

去る1997年8月29日から30日の2日間に渡り、主に組み込みソフトウェア分野の会社8社計23名が集まってそれぞれの会社でのプロセス改善活動に関わる情報交換会が静岡県御殿場 LaLaGOTENBA (オムロン (株) 保養所) で行われた。この集まりは、特に組み込みソフトウェア分野の会社でプロセス改善活動に従事している人たちがプロセス改善の現場でどのような活動が行われているのか、あるいは活動を進める上で実際に抱えている問題などについてじっくりと議論をしたいという声最近増えてくるなか、関係している会社が一同に集まって意見交換をする場を設けたらどうかという話が持ち上がり開催するに至った。

本会の目的は、単なる情報の交換の場だけではなく、今後の現場でのプロセス改善活動に活かすことができる知識を参加者が持ち帰ることができることとし、当日の進め方なども参加者間で意見交換しながら準備を進めてきた。当初は興味あるテーマを決めてグループディスカッション中心の会とする方向で検討を進めてきたが、参加者の興味分野が広くテーマを絞ることが難しいことなどから参加各社での取り組み紹介を行って意見交換をするという進め方をすることになった。また個別に議論したいテーマがある人が発表する場も一部設けた。本稿では、情報交換会の内容を紹介するために各社からの発表概要と参加者のポジションペーパー及び参加後の感想などをもとに報告する。また、会終了後も本メンバーの意見交換用に開設しているメーリングリスト上で議論が続いているため、その内容についても紹介する。

2. 情報交換会スケジュール

とき：1997年8月29日13:00～30日12:00

ところ：オムロン (株) LaLa 御殿場

静岡県御殿場市深沢字長尾2571

スケジュール：

8/29 (金)

13:00 ~ 13:15 オープニング

13:15 ~ 13:30 富士フィルム (株)、大同信号 (株)

自己紹介

13:30 ~ 14:30 (株) リコー 発表&ディスカッション

「組込み型ソフトウェアのプロセス改善の概要

(92年度からの活動概要)」

14:30 ~ 15:30 山一情報システム (株)

発表&ディスカッション

「ISO9001をベースとしたプロセス改善」

(グループウェアを使ったプロジェクト管理事例も含めて)

15:30 ~ 15:45 <Coffee Break>

15:45 ~ 16:45 オムロン (株) 発表&ディスカッション

「オムロンにおけるプロセス改善活動5年目の

ターニングポイント」

16:45 ~ 17:45 富士ゼロックス (株)

発表&ディスカッション

「富士ゼロックスにおける Assessment の実際」

17:45 ~ 18:30 休憩

18:30 ~ 20:00 懇親会

20:00 ~ 22:00 個別テーマ発表&ディスカッション

／マネジメントグループディスカッション

22:00 ~ フリーディスカッション

8/30 (土)

9:00 ~ 10:00 ダイキン工業 (株)

発表&ディスカッション

「構造化手法を中心としたソフト生産性向上の取り組みと

技術移転について」+プログラム自動生成ツールのデモ

10:00 ~ 11:00 キヤノン (株) 発表&ディスカッション

「キヤノンにおけるソフトウェア生産性改善活動の展開」

11:00 ~ 11:15 <Coffee Break>

11:15 ~ 11:45 全体ディスカッション

11:45 ~ 12:00 今後の進め方、クロージング

3. ポジションペーパー

本章では、本情報交換会参加者を中心にポジション・ステートメントと参加した感想について紹介する。

紹介メンバー(敬称略、順不同):

8社24名 (括弧内は、情報交換会へは不参加の方)

ダイキン工業(株):尾本林貞、新田武史、友田稔、
(小幡健司)

富士写真フィルム(株):鈴木圭一、菅原耕一

(株)リコー:松瀬健司、小田利彦、近藤満、山村隆

富士ゼロックス(株):中村淳

キヤノン(株):小泉毅、山田潤二

大同信号(株):菊池実

山一情報システム(株):田淵秀之、小笠原正人

オムロン(株):坂本啓司、板嶋敏男、丹羽徹、朝見昇、
唐島めぐみ、稲垣勝巳、新原直樹、高木徳生

以下、上記名簿順に紹介する。(敬称略)

■尾本 林貞(おもと しげさだ)

会社名&所属:ダイキン工業株式会社 電子技術研究所

(1) 自己紹介

電子技術研究所のソフトウェア技術グループに所属しています。4年ほど前から「マイコンソフトの生産性・信頼性の向上」を目指した取り組みを行っていますが、「プロセス改善」という切り口から言うと直接その役目を担った体制があるわけではなく、プロセス改善ということを念頭に置きながら、具体的には「開発手法による標準化」、「自社開発ツールによる自動化」を推進して、結果としてソフト開発の効率化、ひいてはプロセスの改善に結びつけようという取り組みになっています。我々は研究部門ですがここ1~2年は、より設計部門への寄与度を高めることと上記手法やツールの布教活動も兼ねて、設計部門との製品共同開発への参画度合が高まって来ています。また、我グループを構成する技術者が以前、グラフィックスターミナルのファームウェア開発やグラフィックスワークステーション

の基本ソフト移植研究に携わっていたことから、マイコンソフトからWS/PCアプリまでソフト開発力という面から空調設計部門のソフト技術者に比べ一日の長がありますし、逆に空調機制御と言う面からはほとんど素人です。このことが、現場への技術移転、空調製品の共同開発場面でいろいろな問題を生みます。その問題の整理も含め、今は手探り、試行錯誤の状態、現場への本格的な定着までまだまだ道は険しいといったところです。ちなみに上記技術を活かして、空調部門のマイコンソフト開発は構造化手法をベースに行っていますが、我々自身の自動化ツールそのものはオブジェクト指向で開発しています。

(2) 参加した感想

今回参加するまではどんな情報交換が出来るのか若干の不安と大きな期待がありましたが、御陰様で有意義な時間が過ごせたと有り難く思っています。感想を一言で言うと、情報交換会で皆さんのプロセス改善の取り組みを聞かせて頂いて、さらにその後の私の疑問にメールで答えて頂いたお話を伺って、プロセス改善に対する自分の不勉強さを痛感した次第です。(何せ、坂本さんがオムロンのメンバーの方達に「CMMの勉強不足だ!」とおっしゃってたくらいですから、我々はなおさらです。)

■新田 武史(にした たけし)

会社名&所属:ダイキン工業株式会社 電子技術研究所

(1) 自己紹介

2年ほど前から社内の組み込みソフト開発の支援をしています。昨年は大型空調機の、今年は小型空調機の組み込みソフトを対象としています。それまでは空調機とは関係の無い仕事をしていました(尾本の紹介にもありましたが入社以来グラフィックス関連一筋でした)。当初はソフトウェア作り、という観点で開発プロセスを見て、その部分の改善に注目していました。ところが我々が対象とするドメインではソフトウェア開発は製品開発のほんの1部であり、この部分の工数を最少にして最高の品質にするだけでは片手落ち、との考えを持つに至りました。最近ではソフトウェアを含む物づくり(軸足はソフトウェアですが)という観点で開発プロセスを見るようにしていこうと考えています(まだまだ現状調査段階ですが...)。もちろん現実に

はその第一歩として、開発現場に対してソフト開発効率化／信頼性向上を目指して、主に開発手法／ツールの導入を進めています。その中でいろいろと得たこともあります、まだまだ悩み事も絶えず、手探りの状態を脱しません。

(2) 参加した感想

・他社におけるいろいろな取り組みについてのお話を聞け、更に他社の技術者の方々との交流も深めることができ、私としては大変刺激的な満足のいく2日間でした。

・今後の私たちの取り組みのヒントのようなものをたくさん得たように思います。

・「プロセス改善」と言えば、最初は「管理技術」が中心となる為か、我々が主に取組んでいる「開発技術」はある種毛色が違うという感じもありました。

・他社では全社的に「ソフトウェアのプロセス改善」に取り組んでおられるようで、当社においてもその必要性を感じる反面、

1) 他社ではソフトの規模（も開発している陣容）も我々より大きく、ソフト開発のコストに与えるインパクトも大きい。

2) そのため、開発手法以前のルールづくり等の管理の充実がより大切。

という事情があり我々と取り組みのスタンスが異なっている、とも感じました。

・ちなみに当社の生産本部では、ISO9001への取り組み（工場として取得済み）により、製品開発全体での管理面ではそれなりに充実してきておりますが、特にソフトの開発工程だけで見ましてもまだまだ課題は多いと思っています。

・最後に事務局の皆様はじめ参加、サポートの皆さまには色々お世話になりました。どうもありがとうございました。今後ともよろしくお祈りします。

■友田 稔（ともだ みのもる）

会社名&所属：ダイキン工業株式会社 電子技術研究所

ソフトウェア技術グループ

(1) 自己紹介

電子技術研究所のソフトウェア技術グループに所属し、2年ほど前から「マイコンソフトの生産性・信頼性の向上」を目指した取り組みを行っています。それまでは、グラフ

イクワークステーションのOS部分の移植/開発や、Windowsベースのアプリケーションを開発していました。このような背景もあり、私自身は組み込みマイコンのソフト開発技術者と言うより、アプリケーションソフト開発の技術者です。現在の取り組みでは、マイコンソフト開発を支援する形で、今回デモを予定している自動化ツールなどの開発、サポートを担当しています。今回の技術交流会では、開発したツールのデモと共に、手法やツールの導入に関する議論ができればと思っています。

※ 現場の開発者が自ら手法やツールを導入する場合は、比較的スムーズに導入できると思いますが、ある種、第3者的立場から手法やツールの導入を推進すると、それが良いものであっても何らかの抵抗を受けるように思います。私のようにツールを開発し導入しようとしている立場としては、この抵抗を和らげスムーズに導入するノウハウを蓄積しなければならないと考えています。

(2) 参加した感想

同じ立場の技術者の方とインフォーマルな形で情報交換できた事は大変有意義でした。また、その後のメーリングリスト上での議論も、大変参考になっています。印象に残った事を1つ2つ...

「プロセス改善」は理解されにくいという事について、「見えにくい」のではなく、「見えるようにしていないだけ」と言う意見は、当たり前といえばそれまでですが、自分に当てはめてみると「理解されにくい」事を前提に周りに理解を求めていたようで、もう少し考え方をシフトしなければと感じました。また、同じ組み込み系のソフトでも急激に変化している情報産業に用いられている製品と、当社のように関係が浅い製品では、ソフト開発に求められるものも異なり、それに合ったアプローチが必要で、我々のアプローチである技術面からの取り組みは、当社にとって有効な手段の1つであると再認識できました。

今後、日常の環境から離れてリフレッシュし、発想の転換や、きっかけの作れる場となる事を期待します。

■小幡 健司（おばた けんじ）

会社名&所属：ダイキン工業株式会社 電子技術研究所

ソフトウェア技術グループ

(1) 自己紹介

4年ほど前から「マイコンソフトの生産性・信頼性の向上」をテーマにしています。取り組んできた内容は、

- ・開発手法（リアルタイム構造化手法）の勉強、評価
- ・製品のソフトを試作しながら手法のカスタマイズ
- ・手法で作成したモデルからソースコードへの変換法検討
- ・手法、自動化ツールを実際の製品開発へ適用（開発現場に入っの共同開発）

などです。現在は、適用範囲の拡大に取り組んでいます。プロセス改善とも関係すると思いますが、新しいことを適用した効果を定量的に評価することに苦労しています。収集した開発データから人的な要因を排除することができずにいますので、正しく評価できているとは思えないまま、手法やツールの導入効果の数字をたたき出しております。

■鈴木 圭一（すずき けいいち）

会社名&所属：富士写真フイルム株式会社

宮台技術開発センター

(1) 自己紹介

・業務内容

現在の職場では組込み系のソフトウェアの設計・開発をするグループに所属しています。また、職場内の品質技術グループにも所属し、主に ISO9001 認証取得に向けて職場内の事務局員として活動しています。（事業所で認証取得を考えています）

・参加のきっかけ

弊社、菅原の紹介で参加させていただくことになりました。そもそも菅原からの誘いにOKしたのは「CMM」がキーワードになっています。昨年1年間CMMのレベル2のKPAについて社外の方と勉強会をしていたこともあり、菅原君の昨年のシンポジウムの参加報告を目にして、声を掛けたのがきっかけだったと思います。CMMについてはレベル2のKPAの意図くらいしか知識がありませんが、よろしくお願ひします。

・参加目的

現場でのプロセス改善活動に活かすことができる知識を持ち帰るといふより、知識が深まることを期待していますが、プロセス改善活動といふような位置付けで現在は活動

（仕事）をしていないので、皆様方からの情報が設計・開発サイドの立場で参考になる（活かせる）ことを楽しみにしています。弊社には組織としてプロセス改善を推進する部署がありません。尾本さんのメールで言われている点（ソフト開発に対してほとんど理解できてない経営トップに、どのようにしたらプロセス改善の重要度を実感として理解させることができるのか、成功された先人の苦労話、ノウハウ等聞き、それを我が社（何も着手出来ていない）に当てはめながら、意見交換をしたい。）について興味があります。しかし、経営トップ云々はトップダウンで会社が進めていく場合であって、現状を考えると担当者個々が組織的にプロセス改善をする必要性の認識をもつようになることがまず必要だと思っています。またそのためには、どのような取り組みが有効か参考になる事例紹介が聞けたらと思っています。また、現状の開発スタイルを大きく変えることなくプロセス改善できる（していける）開発ツールの紹介が聞けたらと思っています。

(2) 参加した感想

2日間、各社の事例紹介を聴講して、弊社の組織としてのプロセス改善への取組みが非常に遅れていることを痛感しました。まずは、機会ある毎に上位の人に刺激を与え、少しでも組織としてプロセス改善に取り組んでいけるよう意識づくりを推し進めるとともに、個人的には更なる学習を進めていくつもりです。私はプロセス改善・プロセス管理屋という点では素人なので、みなさんへのインパクトある意見がなかなか言えませんが、プロセス改善活動はこれから益々重要になると思っている人間なので、これからよろしくお願ひします。

■菅原 耕一（すがはら こういち）

会社名&所属：富士写真フイルム株式会社

宮台技術開発センター

(1) 自己紹介

宮台技術開発センターの中の、感材機器開発グループのソフトウェア開発グループに属しております。

プロセス改善については、我々の所属部隊で特に何かを行っているわけではありませんし、またSEPGのようなサポート部隊があるわけでもありません。典型的なCMMレ

ベル1の組織であります。プロセス改善については、全く個人的な興味で勉強しているような状態であり、組織の成熟度を上げて自分たちの仕事をやりやすくしたいと「個人的に」思っている状況です。しかし、「組み込み系」においては、ソフト屋だけが「プロセス改善だ」と叫んでも、結局はハード屋のわがままに振り回されていたりで、ソフト屋だけのプロセス改善を考えていても限界があるように思えます。「ハードとソフトを含めた組織全体に対するプロセス改善」の第一歩を踏み出すためには、何が重要なのかについて、ヒントが得られたらいいなと考えています。ソフトウェアの生産性向上の取り組みとしては、5～6年くらい前に、構造化手法にチャレンジし、Teamwork という市販ソフトを用いてDFD等を書きましたが、それがうまく設計やコーディングの工程につながらず、不満を持った経験があります。

DFDやDDをうまく記述してやれば、そのままソースコードに変換できるはずと思い、しばらくすればそんなツールは現れるだろうと思っていました。ところが、その後、構造化手法はすたれてしまい、オブジェクト指向ばかりが脚光を浴び、どんどんと我々のような組み込み系のソフト環境は置き去りにされていった感があります。

(2) 参加した感想

今回の情報交換会に参加して、皆さんのさまざまな取り組みの話聞き、たいへん参考になりました。と同時に、何も取り組んでいない我々の現状が、たいへんはかしく、また危機感を持ちました。これからも、皆さんの経験を聞かせていただいて、失敗するメカニズムを理解し、成功への作戦を充分練った上で、何かに取り組んでいきたいと思っています。その際は、ぜひ皆さんの助言をいただきたいと思っています。

■松瀬 健司(まつせ けんじ)

会社名&所属：株式会社リコー IPP事業部 技術統括室

(1) 自己紹介

92年度から組み込み型ソフトウェア分野のプロセス改善推進を実施しています。当初は3名のメンバーでプロセス改善をスタートし、徐々に成果を出しながら現在、メンバー6人とともにソフトウェアの生産性と品質を向上する活

動を実施しています。96年にソフトウェアの生産性を向上するための全社的な組織としてP-P-Tが発足し、全社的な展開活動も実施しています。CMMをベースに改善を行っていますが、CMMのレベル評価は重視せず、具体的な改善活動とメトリクス活動を中心に実施しています。

(2) 参加した感想

他社さんの取り組みをいろいろ認識することができ、非常に感謝しております。また、企業の中でのソフトウェアプロセス改善への取り組みは、必ずビジョンと執念を持った1人、2人のキーマンにより実施されていることが伺えました。現場での問題意識から改善活動をされているケース、スタッフのポジションで幅広く改善されているケース、各社さんともそれぞれ企業の問題意識にマッチした、活動ができてい事も感じました。

ベストプラクティスを同じ仕事にたずさわっている方々から学ぶこと、そして、それを参考にプロセス改善を進めていこうと考えていました。しかし、ビジョンと信念を持って、長期間活動されてきた成果はなかなかまねのできないものです。ソフトウェアプロセス改善への取り組み方法が、いかに多種多様であり、どれが正解か、ということもなく、しかも、企業の中のいろいろな圧力に負けない馬力が必要である、ということも改めて認識できました。ソフトウェアのプロセス改善の仕事においては、一企業の中で情報交換したり、課題認識を確認したり、するにはメンバーの数など非常に無理なことが言えます。その意味では、他社さんとの数々の情報交換の場は重要な機会であるとおもえます。今回の場は非常に有意義な場であったと感じております。

■小田 利彦(おだ としひこ)

会社名&所属：株式会社リコー 研究開発本部

ソフトウェア研究所&システムソリューション事業本部

(1) 自己紹介

かつては石油探査のための地層解析システムをLispで開発することから、故障診断エキスパートシステムの開発を経て、数年前よりメトリクス、ドメインエンジニアリング、オブジェクト指向開発等に関する研究開発や社内展開に携わってきた。現在は組み込みソフトウェアにおけるソフ

トウェア再利用の促進を Systematic に進めていく活動に加わっている。世の中の技術の動きに対して、組込み型システムの開発はいつまでも旧態依然でよいのかという疑問がある。また、職人的な集約的であるがゆえの強みも技術レベルが低ければ、今後は他国に容易に取って代わられる技術(生産?)分野になる恐れがあり、組込みソフトウェアの技術インフラを確立・蓄積・パッケージ化することが重要と考えている。

(2) 参加した感想

社内に比較対象の少ない同業者同士の集まりは、得るものが多く貴重な機会でした。今後ますます、情報交換の場、啓蒙を受け与える場として発展することを願います。

■近藤 満 (こんどう みつる)

会社名&所属：株式会社リコー 画像事業統括センター
兼 IPP 事業部 兼 S.S. 事業部 P-PT

(1) 自己紹介

組織変更の多い弊社で、1年弱前から現在の所属となりました。現在は特に所定の部署に所属することなく、幾つかの事業部を取りまとめたセンターに、そしてその事業部の一つである IPP 事業部に兼務で所属しています。

特にソフトウェアのプロセスを専門としている訳ではありませんが、4年～5年程前から組み込み型システムのソフトウェア品質問題の多発、評価期間の肥大化、類似ソフトウェアの繰り返し開発、各事業部での重複開発、等に注目し改善活動に参加してきました。1年程前にソフトウェアの生産性に注目したプロジェクトの発足に漕ぎ着け、その中で特に組込み型のソフトウェアの生産性は如何にしたら上がるのか、その実践、また如何にしたら設計の現場に新プロセスを抵抗少なく浸透させられるのかと格闘しています。

(2) 参加した感想

当社では、設計現場に近い所のプロセスや成果発表に対して保守的であり、社外発表は限られてきました。情報交換は基本的にギブ・アンド・テイクが礼儀との思いと、この保守的な社内文化故に他社との交流に消極的な姿勢を採って来たと言えます。個人的には情報交換により学び、お互いを刺激しあうことで、つまらない遠回りをするこ

く、ベストのアプローチが出来るのだと思っていますが、未だ会社レベルのコンセンサスを得るまでに至ってません。

今回の情報交換会では、多くの会社の、必ずしも組織形態に拘らない改善活動に対する取り組み、実態、現場の情熱を知る事が出来、当社が学ぶこと大でした。頂いた情報に学び、私どもが成長した結果を発表させて頂きお返しが出来るようにしていかななくてはと思っています。まずは個人レベルでの情報発信から開始出来ればと思っています。#白熱した議論をメールで実施するのはなかなか大変そうです。1～2回/年の、課題を絞った直接の交換会も良いですね。

■山村 隆 (やまむら たかし)

会社名&所属：株式会社リコー システムソリューション
事業本部 P-P T (Pプロジェクトチーム)

(1) 自己紹介

現在の仕事に就く前は、PC用のビジネスアプリケーションソフトの開発現場にいました。そのころからCMMやソフトウェア工学には興味があり、勉強会などをやっていたのですが、昨年全社的な規模で、ソフトウェアの生産性を向上させようという気運が高まり、現在のPプロジェクトチームが発足し、この仕事を始めました(PプロジェクトのPはProductivityのPです)。私の主な担当はPC/WS向けの業務ソフトですが、プロジェクトとしては、全社を対象としており、組み込みを含め、様々なソフトウェアを対象としています(とはいってもそれほどパワーはなく、絞り込まざるを得ない状態です)。

CMMをベースに活動していますが、改善活動の指針として利用しようと考えています。開発現場にいかに関問題意識を醸成するかが現在の関心事です。

(2) 参加した感想

思いを同じくする方たちが一堂に会し、本音で話し合えたという事は、大変すばらしいことだと思いました。今まで、例えばCMMについても自分の経験を元に自己流に解釈していた部分がありましたが、今回皆さんのご意見をうかがっている内に、そういう解釈をするのかという新鮮な発見がたくさんありました。なかなか成果の現れにくい地味な仕事ではありますが、横の連携を強くし、相互に刺激

しあう機会が今後も継続することを期待します。

■中村 淳 (なかむら あつし)

会社名&所属：富士ゼロックス株式会社 開発計画部
CMM 推進 G

(1) 自己紹介：

80年代の終わり頃米国 Xerox と共同で400人くらいの Project に参加し、Project 管理の具体的な進め方を学んだ。これをきっかけとして、90年代前半はソフトウェア開発部門長のもとで Project 管理ならびに品質/生産性向上のための改善活動を推進した。結果としては、トップの巻き込みが弱く思うような改善はできず、95年になって担当役員が変わり、本格的に品質/生産性向上のための改善活動を推進することとなった。改善のためのモデル選定から始まり、95年10月から CMM による改善活動を開始した。現在 Formal な Level-2 の認定取得に向け活動推進中 (Pilot Project による標準手順の試行中)。現在の課題としては、どのようにして関係者を巻き込むか。

(2) 参加した感想

各社ともこの改善領域に対して意識の高い方々が出席されており、活発な意見交換がされたことは成果であったと思う。この種の改善活動に対してのアプローチはいろいろとあり、各社各様で大変興味深く聞くことができた。CMM では、アセスメントを改善活動のトリガーとして位置づけており、その意味で、弊社のアセスメントをベースにした活動内容はフォーマルなアプローチの一例として参考になったのではないかと考えている。

この種の改善活動はなかなか先が見とおせない。計画を立てても、他の突発的な作業が発生したりするため、なかなか計画が守れない。従って、コミットをすることが難しい。これは、自身のグループといった限定された範囲の改善活動ではなく、組織としての改善活動であるため、対象が不特定で範囲も広く、また進捗管理が非常に難しいといった点に起因しているものと思われる。SEI も自身のセルフアセスメントの結果は、Level-2 前後とっていることから推測できる。

改善活動は、会社にとって非常に大事な活動として位置づけられているが、このような対象があまりはつきりせず、

かつ広範囲におよぶような改善活動の推進は、時間がかるし結果もなかなか見えにくい。そういった意味で、気が短くなった TOP の風当たりも強いものになりがちである。

(今回集まれた方々は、上記のような環境下で活動を推進されているというのが私の認識です。焦らず、しかし確実に活動を継続されるよう期待しています。私も微力ながら皆さんがたの仲間という認識です。)

■小泉 毅 (こいずみ つよし)

会社名&所属：キヤノン株式会社 ソフトウェア技術開発部

(1) 自己紹介

私どもの場合は、事業部制の中、本社部門の中にプロセス改善支援/技術導入/教育/人材育成などの機能があるわけですが参加される各社のみなさまは、同種の機能が会社の中でどのように位置づけられているのかおのおの位置づけでどのような利点/難点があるのかなど大枠の話を、一度全体でしたいと思います。

弊社よりご紹介できる内容という点でみると、

- 社内の標準化活動、CMMへの取り組みなどの経緯
 - 現在取り組んでいる改善活動(プロセス標準化に先行)
 - ・メトリクスの導入
 - ・ハード開発プロセスから分離のためのシミュレータ開発
 - ・ソフトエンジニアの育成制度
- など、弊社での経験をまとめてお話するということになります。

(2) 参加した感想

同じ機器組み込み分野のかたがたが集まり、各社でどのような取り組みがなされているかを知ることができて非常に有意義な時間を過ごすことができました。今回は計8社が集まったわけですが、これだけのさまざまな分野から集まってなおかつ、問題意識を共有した密度の濃い議論ができたということは機器組み込み系開発という特殊性がある、企業の中での、SEPG、技術支援部門という位置づけに共通する「何か」があるのだと感じました。

SIG-EMS の発足にあたり、弊社から継続的な情報発信ができるかどうかの不安もありますが、まずは、その「何か」を発見し、知恵を出し合えるようなコミュニティになることを期待しています。

■山田 潤二 (やまだ じゅんじ)

会社名&所属：キャノン株式会社 ソフトウェア技術開発部

(1) 自己紹介

対象業務分野： 組み込み系中心

自身の業務内容： 社内各事業における、OOA/OOD 導入コンサルタント的な仕事分析/設計技術教育担当など

(2) 参加した感想

皆さんのお話を伺っている中で、自分にとっては、とても新鮮な視点といくつも出会えて、有意義な時間を過ごせました。どうもありがとうございました。今後ともよろしく願いいたします。

■菊池 実 (きくち みのる)

会社名&所属：大同信号株式会社 第二技術部

(1) 自己紹介

担当業務：システムおよびソフトウェアの設計・製造
(組込およびDOSアプリケーション)

現在は、上記の製品開発業務のほか、業務改善(ソフトウェア開発プロセスの改善)活動に携わっています。

(2) 参加した感想

今回参加した情報交換会のような場に初めて参加したが、非常に有益な情報が得られたと感じています。また今後もこのような情報交換会に積極的に参加していきたいと考えています。

・各社発表について

各社の事例(プロセス改善の過程)を聞くと、例外なく SEPG (ソフトウェア技術グループなど各社名称は違う)が専任として発足され、そのグループが中心となり推進し、“3年~5年で効果が現れ、定着してきた”とのことであった。また非常に大変であったことも実感として伝わってきた。

(確認したところ、専任グループは3名~6名であり、各フェーズ、重要ポイントでは、各プロジェクトのキーマンも参加していたとのことであった。また、社内の状況調査・分析段階より、社外からの専任アドバイザを置いていた会社もあった。)我が社では、今の体制でより良い改善および定着化というものが行っているか非常に不安になった。

・グループディスカッションについて

各社発表においては、効果が現れ、定着しつつあるとして

いる会社であっても、新技術の導入・技術移転となると、個人のスキル不足が大きな問題となり、なかなかうまくいかないとの話が聞け、少しほっとした。当社と同じ(?)状況でもプロセス改善を行い、効果を上げ定着しつつあるのである。しかし、個人のスキルアップは早急に取り組まなければならない当社の問題でもある。

・フリーディスカッションについて

各自立場をはなれ、本音(?)の会話ができ、非常に有意義であった。どの会社でも同じような(顧客からの要求仕様の分析・表現方法、仕様が不明確なままでの見積もり、不確定要素を含みながらの工程計画、メトリクスの設定・評価基準など)問題を抱え、悩み苦しんでいるという話を聞け、安心した(安心している場合ではないのだが...)。このようなディスカッションを行っている中で、自分はまだCMMの要求を満足させるだけの活動(規則類の改定等)を行ってきたのではないかと考えさせられ、「新たな気持ちで取り組んでいかなければならないのでは」という気持ちにさせられた。

■田淵 秀之 (たぶち ひでゆき)

会社名&所属：山一情報システム株式会社 システム統括部
品質保証グループ

(1) 自己紹介

現在、品質管理グループで、ISO9000の推進を行っています。今年2月に認証取得しました。品質関係の経験としては、ISO9000の関係で2年ほど行っており、その前は、システム監査を1年と汎用系基幹システムの開発を6年ほど行っていました。ISO9000、システム監査という世界は、プロセス改善の世界ではどうも嫌われているようで、先のシンポジウムでも、「CMMがISO9000のようになってはいけない」とのお話もありました。ISO9000についてはいろいろ言われているようですが、私はプロセス改善のアプローチ方法の1つとして、有効な手段と考えています。このあたりについて、情報交換会でお話したいと思っています。また、情報交換会で期待していることは、高木@オムロンさんと同様に、プロセス改善の推進者として、プロセスをどのように定着、改善させるのかについて、皆さんと議論したいと思っています。他に、品質に良く効くプロセ

ス(アクティビティかな?)についても議論したいですね。

(2) 参加した感想

品質を良くするために社内で何かを推進しているという同じ立場にいる多くの人々のお話が聞け、大変有意義な時を過ごすことができたと感じております。私は、この手の仕事は、何か信念や信じるものがなければやっていけないものと感じています。ところが、いろいろ推進を行っていると、この信念とか信じるものが疑問に思える時があります。また、時々、自分のやっていることが良いことなのか悪いことなのか客観的に見つめたくくなります。こんな時に、他社の取組みや同じ立場にいる人を聞きたくなるのですが、今回の情報交換会+その後のメールでの議論は、私にとって大変有意義でした。

特に収穫としては、私の信念はある意味・範囲では共感してもらえたことと、ISO9000では保守的なり改善に向かないかも知れないというお話しが聞けたことです。

■小笠原 正人(おがさわら まさひと)

会社名&所属:山一情報システム株式会社

システムソリューション第一部

(1) 自己紹介

現在、新規事業関連のSI業務を担当する部署に所属しており、提案・導入・保守窓口の統括、およびC/S系プロジェクトでのプロジェクト管理、品質管理、といったマネジメント系の作業を主に行っています。

プロセス改善については、これといった取組みをしている訳ではありませんが、弊社田淵の話にあったように、ISO9001の認証取得に向けての取組みに参加し、そこでの成果を部門内の他のプロジェクトに伝播させるという活動を進めており、その中でプロセス改善の必要性を感じ、模索しているといったところ。今時点の関心事としては、プロジェクト活動における作業プロセスや管理プロセスの改善ということで、特にプロジェクトをコントロールしていく上でのルールやメカニズム、それらのインフラ整備といったテーマにあります。

プロジェクト管理におけるグループウェア(LotusNotes)等の管理ツールの活用と利用推進といった点で何か情報の発信が出来ればと思っておりますが、まずは先にのべた課

題に対するヒントが得られないかといった期待をもって参加させていただきますので、よろしくお願いします。

(2) 参加した感想

SEPGという立場、役割についてよく理解しないまま参加したこともあり、今一つ話についていけないところがあったというのが正直な感想でした。しかしながら、日頃の取組みや情報交換の場での皆さんの前向きで積極的な姿勢には大いに感動させられました。また情報交換会を終えた後、改めてディスカッションの内容を振り返ってみることで、自分なりにプロセス改善の取組み方、考え方に関してのヒントが掴めた様に感じ始めており、これをもとに今後の活動に活かしていけるのではと考えています。その意味で自分なりに大きな成果が得られたと感じている次第です。このような素晴らしい会に参加させて頂き、本当に有り難うございました。

■坂本 啓司(さかもと けいし)

会社名&所属:オムロン株式会社 ソーシャル事業グループ

開発・生産センタ ソフトウェア開発部

(1) 自己紹介

私の仕事はプロセス改善以外に実際の開発プロジェクトも受け持っており、昨年はこれまでの私の経験の中で最大のプロジェクトをまですまず成功と言うレベルで完成させることが出来ました。この成功の原因は、これまでのプロセスに関する知識が役立ったこと、SEAで培った人脈から多くの方に助けていただいたこと、そして最後に運が良かったこと、であると思っています。このプロジェクトは全社的な注目を受けたテーマで、本社筋からは「順調に開発中と報告があるがそんなに順調なのはおかしい。何か隠しているのではないか?」と疑われたりしたテーマでした。

ところで、私が今回のワークショップの仕掛け人です。いくつかの会社の方とそのうち情報交換会をやりましょうかという言いながら長らく不義理をしていたのですが、事務局の高木君のおかげでやっと実現する運びになりました。最初はオムロンとどちらかの会社と個別に情報交換会をと思っていたのですが、ついでなら全部いっぺんにということになり盛大なワークショップとなりました。プロセス改善には10年ほど関わっています。

振り返って見ますとこの10年間は前半の5年間と後半の5年間に別れ、さらに後半の5年間も踊り場にきているように思います。それぞれの5年間はそれなりに成果があったが成果が飽和状態になり新しい取り組みが求められていると言う気がしています。現在の状況変化のポイントは二つあると思っています。

一つは我々のプロセス改善の取り組み方法の限界というものと、もう一つは経営課題解決とプロセス改善との関係づけをますます強く求められてきているということです。一つ目の取り組み方法のことですが、我々はCMMベースのプロセス改善を5年以上やっていますが、我流の限界という行き詰まり感があります。(CMMで言うところのInstitutionalizationが不十分でレベル2からのずり落ちを防ぐのに必死になってしがみついていると言う感じ)この閉塞感を打破するためにもう少しCMMの基本的なやり方に立ち戻った方がいいのではないかと最近思っています。つまりもう少しアセスメントベースのプロセス改善にシフトすべきではないかと。そのためにはSEPGがCMMをもう一度しっかりと勉強し直すことと、SEPG以外にもプロセス改善に関わる人達に幅広くCMMを勉強してもらう必要があります。それにはCMMの教科書の日本語化が必須。ということで、今、SEAの有志の方々とCMM日本語化の活動をしています。(いろいろと大変ですが・・・)

もう一つの経営課題とのリンクですが、今年我々のところではすごい金額のソフトウェア設計費用の削減プロジェクトが進んでいます。これはツールを入れて作業が楽になったとか設計効率が何ステップ良くなったとか計画精度が良くなったと言った話しでは全く相手にしてもらえない、会社の利益がいくら上がったかという話しです。ソフトを担当している人から見ると、これをソフトのプロセス改善でやれと言う経営トップの指示は全く無茶苦茶のものと思わずですが、一般的に各社ともこんなことは良くあると思います。そこで今我々がもっとも力を注いでいるところは、ソフトウェア設計費用が決定される要素を整理し、どの部分が本当にソフトウェア設計部隊の責任範囲であるかを明確にし、その部分の改善成果をキチッと計測することと、例えば営業、企画、SEといった設計部隊以外の部門

の責任範囲を明確にするということをしています。ソフトウェアというのはなかなかやっている人以外に理解してもらえず、一見訳のわからないような要求が突き付けられますが、結局企業の中でプロセス改善をするというのはマクロで見ると経営課題を解決するために実行するわけで、経営課題解決のストーリーを説明できないプロセス改善は支持してもらえないのは当然です。そこで現在の私の最大の関心事は、トップダウンの経営課題とボトムアップの現場問題解決型プロセス改善をいかにして繋いでいくかということです。

(2) 参加した感想

ワークショップの進め方をグループディスカッションにしようか全体議論にしようかといろいろ迷ったのですが、その心は「せつかく多くの方が集まっていたのでそれぞれの方々に満足してかえっていただきたい。それぞれの人に今回のワークショップのゴールを明確にさせていただいてそれを達成して帰っていただきたい。漫然と集まったというだけのワークショップにしたくない。そこで興味の対象が違うかもしれないのでグループディスカッションと言うのはどうか?」というものでした。結論は多くの参加者の「広く各社の活動を知りたい」という意見を入れて全体会議としました。この種のワークショップとしては異例のまじめな議論がされ、おまけに全くの異例なことに時間進行もびったり予定通りというものでした。参加後のみなさんの感想からも満足してもらえたようです。私自身もいろいろなお話が聞けて大変勉強になりました。それだけではなくワークショップを機会に、ここの所もやもやしていたプロセス改善に対する閉塞感について考えをまとめる事が出来ました。これを元に先日、経営トップへのプロセス改善への取り組み体制再構築を提言し、これから具体策を検討することになりました。大変いいタイミングのワークショップとなりました。

これからはSEA-SPINのサブグループとしてのSIGEMSで継続して技術交流をしていきたいと思っています。

■板嶋 敏男 (いたじま としお)

会社名&所属：オムロン株式会社 技術本部 IT研究所

ソフトウェアラボ

(1) 自己紹介

過去15年以上、オムロン全社のソフトウェア生産性・品質の向上に取り組んできました。ライン（実際の商品開発を担当）ではなくスタッフ（本社）です。他社でも同じと思いますが、オムロンも、ソフトに関する技術及び取り組みを統括する組織があり、そこに所属しています。過去、15年の取り組みを振り返ってみると、ターニングポイントが3つあります。

1) 82年スタートのASプロジェクト

・抜本的なソフトウェアの生産性・品質向上の取り組み

（IBMのIPTをベース）

技法、ツール、OSの3本柱（UNIX、C言語、社内開発のリアルタイムモニタなど）--->他社に追いつき、追い越す

2) 87年からのWSを中心とする分散開発環境

・UNIX、C言語による開発が本格化

・先進の技術導入

（ITRON、構造化技法、CASEツール）

・ソフトの規模が増大（64KBの制限なし）

3) 92年スタートのSPIプロジェクト

・CMMベースに開発管理技術向上（ソフトの巨大化）

1)、2)に関しては、生産技術が中心で、品質、管理技術については、注力していなかったように思います（本音でいえば、技術、ノウハウなどはそれほど必要ない。やることをきっちりやれば問題はないはず）。この年代は、本社の力（金と技術）が使えたことと新しい技術への取り組みなどができ、良い経験をさせてもらいました。

しかし、管理技術の取り組みが薄かったというつげは、87年から92年にかけて、次々と大きなPJで失敗するという形で現れました。ソフトの巨大化に対して、従来技術では、対応できなくなったのです。これに対して、経営者から打ち手を考えるように指示があり、有識者も含めて立案し取り組んだのが、SPI-PJでした。ですから、プロセス改善には、かれこれ5年間くらい取り組んでいます。2年間は、リソースを投入して取り組んだので（パイロット事業部と共同）、SPIという知名度もあがり、また効果もあったと感じています。また、95年度からは、OOTの導入に注力しています。

現在は、5年間継続してきた、SPIの取り組みも、曲がり角にきており、プロセス成熟度レベルならびにプロダクトのQCD向上ができるように、打ち手が必要な時期と思っています。まだまだ、書きたいのですが、本番に取っておきます。

(2) 参加した感想

今回、このような形で他社との交流会に参加したのは、初めてですが、大変良かったと思っています。この会合で、私は、他社の取り組み状況をしることにより、自社の取り組みを客観的に評価（我々の取り組みについて、ベースの考え方は妥当かまた我々のレベルは他社に比べてどうか）したいと思っていました。進め方については、最初ということで、まじめでかつ内容にも良かったと思っています。各社ともに特長があり、また参考となる情報をえることができました。特に、我々と同じ組み込み分野でプロセス改善に取り組んでいる人とフォーマル、インフォーマルで情報交換ができて、大変有意義でした。

今後は、さらに突っ込んだ議論ができる場となり、その成果を持ち帰り、各社のプロセス改善に活用できるようになればと考えています。よろしくお願いします。

■丹羽 徹（にわ とおる）

会社名&所属：オムロン株式会社 技術本部 IT研究所

ソフトウェアアラ

(1) 自己紹介

学生時代と入社後しばらくはPC/WS上のソフトウェアの開発に携わっておりました。根はプログラマーですが、その後、上流工程の設計品質向上の取組みや、オブジェクト指向などの技術にも関わりながら、所属部署のミッションでもある、マネジメントを中心としたプロセス改善活動に携わって来ました。

(2) 参加した感想

組込み開発を抱える他社様のプロセス改善活動事例をこれだけ一度に伺う事ができて、非常に参考になりました。特に各社様ともソフトウェア開発者に対してある一定の比率のSEPGの部門（あるいは担当者）が存在したこと、また、各社が改善活動の方向性がそれぞれ異なっていることが印象的でした。その中で、各社の改善活動の方向性は、

各社がおかれている事業環境が大きく影響していることを再確認することができました。その異なる事業環境の中で、いかにソフトウェアの改善活動を経営が認識している課題と結び付けるかが成否の分かれ目になると感じました。当社の改善活動もまだまだ成熟しているわけではないので、いかに現状の課題を正しく認識して、トップから改善活動を実践していくことが重要かを再認識しました。今回学んだことを今後の自分への戒めとして日々の業務に役立てていきたいと考えます。どうもありがとうございました。

■朝見 昇 (あさみ のぼる)

会社名&所属：オムロン株式会社 インダストリアル事業
グループ 開発センタ 第一開発部

(1) 自己紹介

4年ほど前からSEPGとしてソフト開発の標準化を行ってきました。現場が困っていないとスタッフの言うことは聞いてくれないので、(困ってないから聞く必要は無いのは当然と思いますが)SEPGとしての限界(自分の能力の限界かもしれません)を感じていました。今はSQAとしてあるテーマに張り付いて、ソフト開発の品質を見ています。

SEPGの立場で思うのは、

- ・現場の人間は毎日の業務をこなすのに精一杯。
- ・問題や疑問を感じても、今までのやり方から抜け出すための勇気やそのための時間が無い。
- ・SEPGは問題の指摘や改善の方向の検討はできても、金や時間を与えるスポンサーにはなれない。十分な動機付けもできない。
- ・改善のトリガーや、改善のための金や時間を与えることができるはマネージャー。

マネージャをいかに動かすかがSEPGの能力の一つ。けれども、感性のないマネージャは動かない。いつも最後のところでデッドロックしています。

(2) 参加した感想

各社で様々なプロセス改善の取り組みを実施されている状況を伺って、最近少々マンネリ気味であった私は良い刺激になりました。組織に合わせた取り組みは狭い範囲でしか物を見れていなかった私には非常に参考になりました。

今後のプロセス改善の取り組みの参考にします。今後このような機会が継続して持たれることを期待します。

■廣島 めぐみ (からしま めぐみ)

会社名&所属：オムロン株式会社 インダストリアル事業
グループ技術統括センタ 技術開発センタ

(1) 自己紹介

コントローラなどの組み込み系のソフトウェア開発、WS上でのC/Sアプリのソフトウェア開発を担当して、3年ほど前から所属事業部門を対象とした専任のSEPGとなりました。当初から、特定の組織の改善活動に(主にプロジェクト管理的観点から)取り組んでいるSEPGを支援するSEPGというスタッフ的な役割と、ソフトウェア設計技術の導入・ソフトウェア開発ノウハウの普及という役割をもって取り組んできました。特に、1年ほど前からは後者に力点を移し、パイロットプロジェクトの開発に参加しながらテスト技術やC/Sアプリの開発プロセスの技術検証を進めているところです。この成果をどのようにまとめ、これをどのようにして他のプロジェクトへ移管していくかが、現在の課題です。

(2) 参加した感想

ちょうど今後の自部門のプロセス改善活動の進め方に思い悩んでいた時期で、もやもやしたものを抱えながら参加したのですが、それを思い切るヒントになるようなものを持って帰ることができました。それと同時に、中途半端なパワーのかけ方や取り組み方ではプロセス改善を進めることは難しい、と改めて思いました。個別の話では、自由度をもたせたくみ(開発手順など)が開発者の創意工夫を引き出してうまく回っている(自分なりに解釈した言葉にしている)ので直接の発言と違うかもしれないが)というのが印象に残っています。今回は、対象を絞った議論もやりたいです。今後とも、よろしくお願いします。

■福垣 勝巳 (いながき かつみ)

会社名&所属：オムロン株式会社 ソーシャル事業グループ
開発・生産センタ ソフトウェア開発部

(1) 自己紹介

入社から約5年間、駅務機器（券売機、精算機、改札機等の間）、銀行システム等の通信ソフトウェアの開発を担当していました。その後、SEPG担当者となり、現在に至っています。最近はソフトウェアの開発からはすっかり遠ざかっています。

(2) 参加した感想

同じ分野のソフトウェア開発に携わる人の集まりということで、それぞれの環境も比較的似ており、効率的で相互に理解しやすい議論ができたと思います。また、私の想像より多くの会社で専任のプロセス改善メンバがいたこと、CMMを十分に理解して取り組みのベースにしている（自分の理解の不足を痛感した）こと、発表させていただいた内容について、鋭い指摘をいただいたことなど、とても刺激になりました。今後ともよろしく願いいたします。

■新原 直樹（にいばら なおき）

会社名&所属：オムロン株式会社 ソーシャル事業グループ
開発・生産センタ ソフトウェア開発部

(1) 自己紹介

入社後 POS や CAT（クレジットカード認証端末）の開発に携わり、93年にSEPGの専任となりました。以来4年間はメトリクスや計画の充実、プロセス改善などに取り組んで現在に至っています。私共の部門では過去にもプロセス改善に取り組んだことから、私も昨年は特定プロジェクトのプロセス改善にべったりと入ったのですが、これがなかなかうまく行きません。なぜ今回は思うように進まないのかをはっきりさせることが、今の私達の課題の一つです。

(2) 参加した感想

色々な会社の中で、一見同じような業務に携わっているように見えるものの、皆さんの話を聞いていると、取り組みや抱える問題、立場、環境、仕事への思いなど実にさまざまな人達の集まりであると感じました。自分自身の疑問に対して直接的に答を得られた人もおられるかもしれません。私はといいますと、皆さんの知識の深さを感じる反面、ただただ自分の知識の貧弱さ（それすらないかもしれませんが）にかなり打ちのめされています。しかし、これが今回の私の最大の収穫であると思っていますし、大変勉強になったことは言うまでもありません。（皆さん本当に知識の

バックグラウンドが広いですね。）

次の機会がありましたら、もっと多くの方の話をお聞きしたいと共に、さらに的を絞り突っ込んだ話もできればという欲張った思いを持っております。

■高木 徳生（たかぎ やすなり）

会社名&所属：オムロン（株）ソーシャル事業グループ
開発・生産センタ ソフトウェア開発部

(1) 自己紹介

弊社ソーシャル事業グループのソフトウェア開発に対するプロセス改善の推進グループ（SEPG）に所属しております。3年ほどまえからこの分野に関わり始めたのですが、最近社内/外ともにプロセス改善の重要性がますます盛んに問われ出している雰囲気を感じています。今回、多くの会社の方々にご賛同いただけたことはとてもうれしく感じています。

さて、プロセス改善を進めている立場から、プロセス改善を組織の中に根づかせるためには、プロセス改善自体の技術移転が必要であると思っております。プロセス改善といっても、各現場でいろいろな手法があると思いますが、効率的にプロセス改善を推進するやりかたは何か？そこでSEPGの役割とは何か？また、組み込み系会社の特徴をプロセス改善推進に活かしていけないか？などということに関し議論できればと思っております。

(2) 参加した感想

本会の事務局を担当させていただいた私としては、今回急な設定であった本会に多くの企業から多数の参加をしていただいたことを大変うれしく思っております。当日の議論も（一部ではちょっと雰囲気が堅いという声もありましたが）非常に熱のこもった議論ができ、とても有意義な会になったのではないかとおもっております。この場を借りて皆さんにお礼を申し上げたいと思います。本会で感じましたのは、組み込み系ソフトという分野の企業でのプロセス改善の取り組みがだいたい弊社が取り組み始めた時期と同じ時期に取り組みされてること、また取り組み内容は各社さまざまですが課題と感じている点は同じであるということです。つまり、プロセス改善を企業ですすめていく場合の現場や経営にどのようにその有効性を理解してもらうか

その為にプロセス改善推進者はどのように考え行動をするべきかということが共通の課題であると思います。これに対してはこうすればいいというものではなく、その時代の環境や状況、また企業の風土や事業分野などによって異なる、即ち常にプロセス改善の取り組みプロセスも改善していく必要があるのだということを感じました。これまで SPIN のワークショップにも参加させていただいておりますが、組み込みソフト系という分野を絞った企業での同じような役割の人たちとの交流はまた違った新鮮な雰囲気を感じました。是非、今後ともこのネットワークが継続しさらに広がっていくことを期待します。

4. 各発表概要

4.1 各社発表概要

(1) 「組み込み型ソフトウェアのプロセス改善の概要

(92年度からの活動概要)」

発表者：(株)リコー松瀬 健司

弊社におきましては92年後半より、コピー、マルチファンクションコピー(FAXやプリンター機能が搭載された商品)に組み込まれるソフトウェアの開発プロセス改善活動を、CMM(カーネギーメロン大学ソフトウェア工学研究所によるソフトウェアプロセス成熟度モデル)に基づいて積極的に展開してまいりました。この活動を始めるきっかけとなったことには、商品機能が多機能に発展するに当たってソフトウェアの開発量が急激に増大し、従来のソフトウェア開発のスタイルではQCDの達成が困難となってきたこと、弊社におきまして92年度後半にISO9001を受信すること、などがあげられます。ISO9001の対応整備、並びにソフトウェア開発のQCDの精度向上を目標に、92年度にCMMにおいて提案されているソフトウェアプロセス改善実施組織としてSEPG (Software Engineering Process Group) を発足しプロセス活動を開始しました。

SEPG組織を中心に、ソフト改善PG、構造改革委員会などの委員会活動を通じてソフトウェア開発プロセスの定義付け(標準化)活動や設計ツールの導入検討などを実施してきました(1次標準化、2次標準化、3次標準化・・・)。また、大規模ソフトウェアの急場の品質向

上策としましては、従来から実施しておりました品質保証テスト(QAテスト)とは別に設計内テストのプロセスと体制を構築しました(総合評価体制確立)。さらには、今後のソフトウェアプロセスを構築(改善)していくにあたって、ソフトウェアマネージャーの意識改革が重要であり、同時、品質管理、メトリクス、開発管理、構造化分析手法、オブジェクト指向に代表されるソフトウェア設計手法・管理技術の変革などの教育・啓蒙活動を展開してきました。

これらの活動が効果的に働き、コピー等に組み込まれるソフトウェアの品質を大幅に向上することができました。94年度以後は、ファクシミリ分野、プリンター分野など弊社を代表する商品群全体に活動範囲を広げ、現在に至っております。

●発表に関する感想

今回は、初めての8社合同の情報交換会ともあって各社のプロセス改善に対する取り組みを理解することから始めなくては詳細な議論に発展することは難しいと思いました。そんなこともあって、事務局の方に、「まず、各社のプロセス改善への取り組みを発表してもらい、お互いの状況を確認するところから始めた方がよい」と、提案させていただきました。結果的に、各社の会社事情とそれに適した取り組み内容の大筋を理解することができ、非常によかったかと思っております。また、自分たちが詳しく知りたい内容や詳細な改善ポイント、苦勞話の数々について、細かく情報交換をできる相手もわかりました。弊社から発表させてもらった内容は、ソフトウェアプロセス改善の全体的な取り組みに関する内容でしたが、多くの意見や質問があり、やはり関心が高い内容であったかと・・・解釈させてもらっています。いろいろな会社の方々からの意見を、1人で聞いていると一見、話が発散し、ポイントが絞れないような感想を抱きました。しかし、関心事が会社毎(担当者毎)に異なるからだと理解しております。会社内でそれほどたくさんの人材を抱えられない、ソフトウェアプロセス改善の分野で、各企業の方々とのローカルな交流会は、プロセス改善エンジニアの育成の場、自社の成果の客観的な評価の場、プロセス改善方法の情報源の場として活用することが重要なことだと感じています。

(2) 「ISO9001をベースとしたプロセス改善」

発表者：山一情報システム（株）田淵 秀之

弊社では、95年からISO9000をベースにしたプロセス改善の検討を開始したが当時、社内ではISO9000よりCMMでやるべきだとの意見もあり、ISO派とCMM派の2グループで熱い議論を交わした。論点は、運用（手順の実行）徹底方法、認証取得のタイミングであった。CMM派の意見は、意識改革により運用を徹底方法させるべきであり、認証取得はレベル2に達成してから（当社はレベル1であった）目指すべきであるというものに対して、ISO派は、運用徹底には外部審査が必要であり、先ず認証取得が必要との意見であった。最終的には経営判断によりISO9000を選択し、97年2月に認証を取得した。

上記経緯の通り、ルールを作っても守られないという課題に対しISO9000という強制力でもって対処するという方法を選択した。ISO9000の弊害？（必ず手順に準拠しなければならない。これに対してCMMは必ずしも手順通りである必要はなくゴールを満たしてれば良いことになっている）については、手順作りそのものをプロジェクトの裁量にまかせる部分を多くする、という工夫を行い、ソフトウェア開発でもISO9000を適用しやすい環境を作った。

これまでの活動効果はまだこれといって言えるものはないが、プロセス改善という観点からは、プロジェクトの裁量にまかせた仕組みがプロジェクトの創意工夫を生み出し、良い結果を出している。具体的には、すばらしい手順が作られたり、前向きに（同じやるなら良い方法でやろう！と積極的に）取組むプロジェクトが多く発生するといった結果を出している。

●主な質疑応答

Q：ISO9000は規定を守ることが重要であり、改善つまり規定を変更するという考えとは相反するのでは？

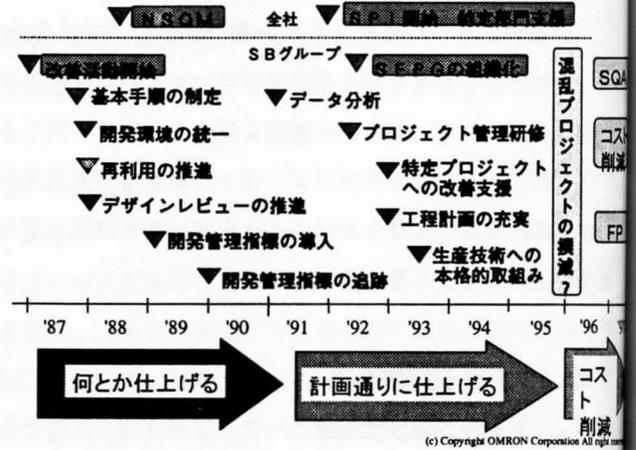
A：言われて見れば、その懸念はある。しかし、認証取得の目的が、純粋に品質向上であり、そのためには規定の見直し（プロセス改善）が必要であり、ISO9000でもプロセス改善は可能と考えている。審査の観点も、是正（品質システムの見直し）に重点を置いている。

(3) 「オムロンにおけるプロセス改善活動5年目の

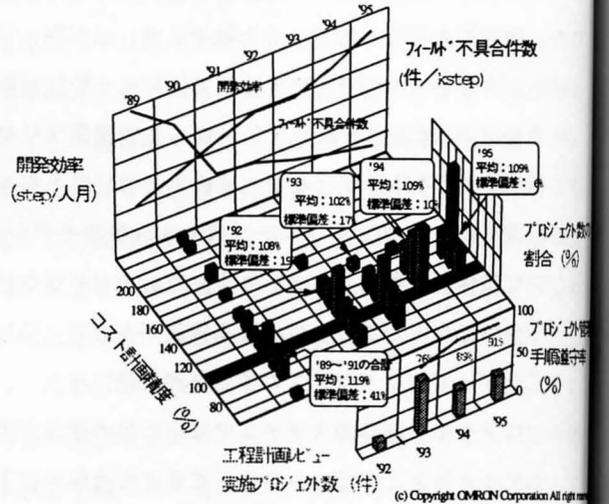
ターニングポイント」

発表者：オムロン（株）坂本 啓司

プロセス改善活動の経過



開発計画予実精度向上



プロセス成熟度の現状認識

95年まででほぼ混乱プロジェクトはなくなり計画予実精度、生産性、品質が向上し、プロセス成熟度レベル2をほぼ達成。

プロセス成熟度レベル3に向けての課題

- レベル2の歯止め
成熟度モデルの発流カスタマイズに負債事項あり。特に大きなものとしてソフトウェア品質保証機能。
- 大規模開発の計画どおりの実行
従来の開発に対し一桁近く大きな開発が出てきている。計画の充実と、より厳密なプロジェクト管理、リスク管理。
- 客先対応ソフトの生産性向上
作業工程の定務で効率化。工程単位で改善効果測定。プラットフォーム再構築で都成部の再利用向上。

どんな変化があったか？

- 取組み内容の限界
 - ・特定の人たち(SEPG)のインタープリットの限界
 - ・人が変わればまた一から説明が必要
 - ・改善成果の歯止めが難しい
 - ・実はCMMのことをあまり勉強していない
- 経営課題の変化
 - ・約束が守れるようになるとそれは当たり前になる
 - ・次は開発コストの問題がクローズアップ
 - ・しかし開発コストは問題議論のベースが不明確
 - ・しかも開発コストは開発部門だけの問題ではない

どう取組むか？

- 取組み内容の限界
 - ・Institutionalization(制度化)
 - ・レベル3達成を会社方針としてもらう
 - ・CMMの再勉強
 - ・CMM日本語化に積極参加
- 経営課題の変化
 - ・プロセス成熟度の改善成果をアピール
 - ・ビジネスプロセス全体を整理
開発以外の問題を明確化
 - ・ソフトウェア機能の開発量計測、FPAの取組み
 - ・ソフトウェア設計力の強化

(4) 「富士ゼロックスにおける Assessment の実際」

発表者：富士ゼロックス (株) 中村 淳

CMM の Formal Assessment を実施した企業として、その Assessment Process について具体的に説明した。弊社が CMM コンサルタントを呼びフォーマルなアプローチを採用した理由は、主に次の3点。

- 1) 改善活動を推進する核となるメンバーの、ソフトウェアエンジニアリング技術に対しての知識ベースを高める必要があった。そのためには実績のあるモデルならびにコンサルタントを採用することが手っ取り早いと考えた。
- 2) TOP を説得していくには、それなりの経験を積んだコンサルタントの立場を有効活用すべきと考えた。SEPG で検討した結果をコンサルタントに説明し、コンサルタントから TOP に伝えてもらう。CMM には Professional Judgment という都合のいい言葉がある。
- 3) 改善活動を最初からカスタマイズしていく自信がなかった。

(5) 「構造化手法を中心としたソフト生産性向上の

取り組みと技術移転について」

& 「プログラム自動生成ツールのデモ」

発表者：ダイキン工業 (株) 新田 武史、友田 稔

1) 当社技術開発ロードマップ

マイコンソフトの生産性、信頼性向上を目指したソフト開発の標準化(ダイキン版リアルタイム構造化手法)、自動化(ダイキン版マイコンソフト開発支援環境)に取り組んでいる。

2) 開発現場への技術導入の流れ

'92からの取り組みにより、製品への適用が徐々に進んできている。

3) 手法、ツール適用により解決された従来のソフト開発における問題点

- ・ソフトウェアの設計仕様書では、全体像および各機能の関連が分かり難い
- ・手作業によるコーディングのため、ミスが発生しがち。
- ・後工程重視(とにかく作って試験で確認)の為、手戻りも多い
- ・コードと設計仕様書の不一致が発生しがち。

4) 技術移転上の苦労話

- ・開発現場は忙しく、製品開発業務に新手法勉強の時間がつぶされる
- ・開発期間、コストが厳しい
- ・ソフトウェアに対する現場の認識が低い
- ・設計ノウハウが仕様書として残っていない
- ・これまでのやり方へのこだわり
- ・推進する側の製品に関する知識不足
- ・改善が進むと現場の人間は仕事を失う?

5) 今後の課題(技術的及び導入推進上課題について説明)

6) プログラム自動生成ツールの紹介

(モデル記述方法、生成コード等)

7) 当日の質疑応答

- ・自動生成されたコードに手を入れる必要はあるのか
→ない
- ・なぜ汎用のツールを使わないのか。
→汎用のツールも評価したが、我々のニーズに合ったものがなかった。

(6)「キヤノンにおけるソフトウェア生産性改善活動**の展開」****発表者：キヤノン（株）小泉 毅**

●取り組み

< Phase I (1990~1992) >

- ・電気技術者からソフトウェア技術者への技術教育
 - ・WS 1台/人体制、高級言語とリアルタイム OS 導入
 - ・ウォータフォール型開発工程の導入/CMM への接近
- 開発領域の問題—非常に多彩なドメイン、工程技術が未成熟（設計/製造/テスト）、（改善推進側の）要員不足などの理由により開発工程導入は失敗/CMMからの撤退。

< Phase II (1993~1995) >

- ・ソフトウェア技術者教育体系の確立
 - ・ソフトウェアエンジニア育成制度（*）の開始
- （*）ソフトエンジニア育成制度：情報系以外を中心とした新入社員を2年間、ソフト技術開発部（本社組織）に配属し、体系的技術者教育/事業部からの依頼テーマによる実習を行う。その後、事業部へ再配属となる。
- ・アプリケーション製品の普及、シミュレータによるハードとの並行開発
 - ・ソフトウェア開発の専任組織化

< Phase III (1996~) >

- ・オブジェクト指向設計/製造技法の導入
- 部品化によるビルディングブロック式開発、ネットワーク製品対応
- ・ソフトウェアメトリクスの導入
- 開発状況可視化と計画精度向上、新技術導入の効果計測

●今後の課題

- 1) 製品技術領域の激しい変化に対応することが危急の課題（Windows/インターネット、分散オブジェクト技術、ソフト開発技術：分析/設計/実装/テスト/評価技術などへの対応）
- 2) 人材育成
 - ・製品開発部隊には新技術導入の情報源も時間もない
 - ・最新の技術を取り込み使いこなすセンスを身につけた人材の供給が必要
- 3) 標準プロセスの確立—CMM ベースの改善はとりあえず

Pending

- ・製品領域の変化が激しくプロセスの再現性を求めても Pay しない
- ・マクロなデータ把握と大きなプロセス分割(ex. 開発 vs 評価)がまず先決

4. 2 個別テーマ発表**(1)「ソフトウェア技術導入について思うこと—パイロット PJ へのテストチーム制導入事例をベースとして」****発表者：オムロン（株）唐島 めぐみ**

●発表内容（議論のポイント）

現在、パイロット PJ（PC 上のアプリケーションソフト開発）にて、開発メンバーの一員となり、インクリメンタル開発・テストチーム制など開発プロセスを改善するための技術を検証しています。今後、検証結果をベースに技術移転する際に、どのように効果をアピールするか、どのような形で移転するか、を悩んでいます。

●感想

取り組みの目的・目標や期待効果が不明確とのご指摘を受けました。将来困るであろう問題として設定して取り組んでいることに甘えて、あいまいにしてきた部分を指摘されてしまったと思いました。質疑応答の中で、技術移転については各社それぞれの考え・方法で取り組んでおられる印象を受けましたので、次回はぜひ具体的な内容（事例）について議論させていただきたいです。

(2)「全社的なプロセス改善活動をいかに継続するか？ 本社スタッフのプロセス改善活動のありかたについて」**発表者：オムロン（株）丹羽 徹**

●発表内容（議論のポイント）

- ・ひとたび行き渡った改善活動を維持するには相当のパワーが必要
- ・しかし経営層からは改善活動を継続させることはあまり評価されない
- ・いかに現場に改善活動を根づかせるか？

●感想

私の今回の発表は、当社だけの悩みのようでした。ただ、改善活動が一通り浸透した何社かは、次に直面する自分た

ちの問題として真剣に捉えて頂けたようです。トップと現場に対する改善活動のバランスよくアピールしていくことが重要であることを認識しました。

(3)「ソフトウェア開発プロジェクトにおける開発計画の分析～品質、生産性との関連性～」

発表者：オムロン（株）稲垣 勝巳

●発表内容（議論のポイント）

ソフトウェアプロジェクトの開発計画の充実度が、コスト予測精度（見積もりコストに対する実績コスト）に影響を与えること、コスト予測精度がソフトウェアの生産性および品質（製品出荷後の市場品質）との関係をデータ分析により紹介した。

●感想

紹介させていただいた内容について、分析が不十分な点に関して率直なご意見をいただけたことが最も大きな成果でした。また、データ分析の着眼点についても新たな観点を教えていただくことができ、有意義な場となりました。紹介させていただいた分析では、当然と考えていることを実データを用いて証明しようとしたものであり、よく考えると十分な分析ができていないままに、それらしい傾向が出ていればそこで結論に結び付けてしまい中途半端に終わっていた感がありました。本当の問題は何か、その問題はどうすれば解決できるのかという分析の目的を達成できるもに変えていきたいと思えます。今後の分析を進めるにあたって、たいへん参考となりました。また、今回の参加者の方々は組み込み機器の開発に従事しておられる点、方法は違うと言えども大きく捉えるとほぼ同じ活動をなさっている方がほとんどであったため、仮説の意味や背景を説明するまでもなく直感的な議論できた点は非常に効率的でした。最後に、次の機会にはまた何らかの紹介をさせていただき、ぜひ皆様のご意見を伺いたしたいと思います。今後ともよろしくお願いいたします。

(4)「プロセス改善推進のポイント

—プロセス改善プロセスの分析から—

発表者：オムロン（株）高木 徳生

●発表内容（議論のポイント）

- ・我々の部門では、メトリクス評価分析フィードバックやマネージャへの技術啓蒙などの組織全体的な取り組みと共に特定プロジェクトのプロセスを分析し改善するという取り組みを行っている。
- ・特定プロジェクトのプロセス改善の為には、現状を把握し（AsIsMap）、問題を抽出し、改善されたプロセスを定義し（ShouldBeMap）実行、評価するという改善サイクルを実行する。ここで現状をいかに把握するかというのが非常に重要なポイントである。
- ・ソフトウェア開発プロセスの現状を把握するためには、開発工数や不具合数といったデータやプロダクト（ドキュメント、ソースコードなど）と実設計者への作業のヒヤリングなどを通じて行うことになる。ここでの問題は、プロセス粒度や現場設計者の協力をいかに引き出すかである。
- ・過去の経験から、このようなプロセス改善を行う場合には以下のポイントに注意する必要があると考える。

- 1) 目に見える取り組み…ゴールを明確にする。プロセス改善計画を立てる。ゴールと計画を関係者全員で了解する。アウトプットは図表で残す。など
- 2) 改善サイクルは早くまわす…手をつけられる部分から分析、改善をすすめる。時間をかければいいというものではない。
- 3) 役割分担を明確にする…必ずしもSEPGが主導権をにぎってもうまく進むとは限らない。ベストプラクティスは現場から。
- 4) 情報の整理と活用…因果関係に着目した情報整理に注意する。

●感想

- ・プロセス改善活動といっても、具体的には違う取り組みのことを指している場合が多い。今回説明した弊社での取り組みはCMMなどを活用して組織的にというのとは違った（どちらかといえばTQC的）取り組みである。あまりこのような取り組み事例を他では耳にしたことがないため、今回こちらからの発表に対してどう思われるかを伺えるきっかけとして良い場であった。
- ・今回の改善ポイントについてはまだ自分でも実証できたものではなく半分アイデアレベルのものである。このような悩みを持っているのは自分だけかという少し不安もあった

が、面白い分析観点とのご意見も頂けうれしかった。今後より具体的なプロセス改善プロセスの改善ノウハウを掴めるよう取り組んでいこうと思っている。

5. ポスト・情報交換会

「管理指向か技術指向か－メーリングリストの議論から－」

今回のワークショップでは、参加した8社のうち6社から、何らかの形でCMMを参照し、あるいはISO9000認証取得を契機として、組織的改善の取り組みが発表された。これには、プロセス改善をテーマとしたワークショップであったことと、最初の会合でもあり、企業としての取り組みの全体像の紹介が期待されていたことなど、特殊な条件もあるのだが、それを差し引いても、管理指向の改善活動が広く行われていることを印象付けることになった。

さて、このワークショップの後、メーリングリストでは、さかんな議論が展開されている。テーマを要約するとすれば、「管理指向か技術指向か」ということになるだろう。その発端は以下のA氏の発言である。

<A氏>

－ snip －

我々は「プロセス改善」という切り口から言うと直接その役目を担った体制があるわけではなく、プロセス改善ということを念頭に置きながら、具体的には「開発手法による標準化」、「自社開発ツールによる自動化」を推進して、結果としてソフト開発の効率化、ひいてはプロセスの改善に結びつけようという取り組みをしているわけですが、皆さんの取り組みと順序が逆というか、切り口が違っているように感じた点です。

確かに、ハンフリーさんの本にも「技術的に高度なツールや手法がソフトウェア危機に対し魔術的な効果を発揮するだろう考えるのは、間違いだけでなく危険であり、プロセス改善から注意を逸らせてしまう」と書かれていますし、坂本さん達が監訳された「ソフトウェアプロセスのトレンド」の中で紹介されているオムロンさんの取り組みも「初期の成熟度の組織は、たとえばCASEツールを用いたSA/SD技法のように高度な設計技法の推進というより、むしろ混乱プロジェクトを取り除くという初歩的なことから行う必

要がある」と述べられています。そういうことから言うと、我々はそれに逆らったアプローチを取っているとも言えるわけですが、しかし私には、「人と組織に起因するソフトウェアプロセスの問題を、人と組織の改善によって解決するというプロセス改善の取り組みは、解き方自体に無理がある」のではないかと思える部分があります。

－ snip －

私としては、人間が介在することによって生じるソフト開発上の問題は、人間が介在する限り解消しないと思いますので、人間の介在を無くす(＝自動化)か、あるいは人間の介在を事実上なくしてしまうための機械的支援による自由度の排除(＝標準化)を目指す技術的アプローチが重要だと思っています。

－ snip －

そしてそれは、事務処理系の多種多様なドメインを対象とする場合はさておき、対象分野、製品が限定できる組み込み系の場合にその実現の可能性が高いと思っています。ということで、組み込み系とそうでない場合でプロセス改善のアプローチに自ずと違いはあると思いますが、早い話「自動化によって問題のあるプロセスそのものを無くしてしまうことがプロセス改善の最高の解」だという乱暴な見方について皆さんはどう思われるでしょうか？

－ snip －

この発言をきっかけとして、CMMとは何か、SEPGの仕事とは何かという問題が、改めて議論されることになった。以下、その要旨を追ってみよう。

<B氏>

- ・ソフトウェア開発には、技術導入や自動化に還元できない、人間そのものの問題がある。それは、本来、工学的な(再現性のある)解決策のないものだ。
- ・CMMのLEVEL2へ至るプロセスは、人間そのものの問題を、うまく工学的な枠組みに取り込んで、人間そのものの問題を解決する糸口、つまり、説得の材料を提供している。
- ・CMMが改善の順序として、管理的側面を第一にあげているのは、人間そのものの問題を、最低限、改善の障害に

ならないようにクリアするためであって、技術的アプローチを否定しているわけではないのではないか。

- ・人間そのものの問題をどう扱うかによって、改善活動のスタンスが変わってくる。

<C氏>

- ・混乱プロジェクトを取り除くという初歩的なことから始めなければならない組織は、現に混乱プロジェクトを抱えている組織にかぎられる。初歩的問題を抱えていない組織では、技術指向のアプローチをとることは当然だ。それぞれの組織の問題分析に基づいて戦略を考えるべきだ。
- ・CMMの最大のポイントは「ソフトウェアを人間系の難しい問題として改善の第一歩を踏み出すべきである」といっているところだと思います。レベル3以上の内容は別にCMMでなくてもいろいろな改善の取組みパラダイムはある。

<D氏>

- ・目的は、ソフトウェアの生産性と品質の向上であり、それを実施するにあたっては、各社の事情による。各社がもっともやりやすい体制、組織の課題解決にもっとも短期間で到達できる策をとればよい。
- ・生産性と品質を向上するためにツールは必要である。しかし、一般にツールから入ることは危険である。ツールは手段であって、導入する際に目的に置き換わっているようであれば、ツールをいれても何も効果が得られない。
- ・改善対象が大きく、一気に改善を行うためには組織の位置付けも考えなくてはならないこともある。なんらかの機能で改善ツール（電子化ツールに限らず改善の種という意味）をもてることと、それを定着させるために必要ならば人への教育と組織の位置付け、組織への働きかけ、も考えなくてはならないこともある。SEPGはこれらすべてが仕事となる。

<E氏>

- ・全ての組織に対し、技術指向アプローチでないとか、管理指向アプローチでないとか、だめだとかというこ

とではない。CMMは、一般的に下位レベル（つまり管理の問題）のKPAから取り組んだ方がリスクが少ないといっていると解釈できないか。

- ・管理にも技術がある。計画を立てる、進捗管理するということは、過去からの研究や経験の積み重ねで得られた知識（技術）である。
- ・ソフトウェア開発は、コンピュータと人間の共同作業で行うものである。つまり、ソフトウェア開発を改善する（プロセスを改善する）場合には人間とコンピュータ両面から考える必要がある。

この議論に明確な結論が出たわけではない。しかし、「ソフトウェアの品質、生産性を向上するという目的は同じでも、改善のアプローチには、各企業の事情によってさまざまなのが考えられる」ということには、一応の共通理解が成立しているようである。この議論は現在も、組織を構成する個人の問題、管理者のモラル、技術の問題へと移り、さらに続いている。

ここで議論されている問題は、一見して、機器組み込み型という世領域に限られたことではない。しかし、その背景には、共通の領域で類似の問題意識を共有しているという実感があり、だからこそ、活発な議論ができるのかも知れない。

6. おわりに

本情報交換会は、7月中旬に話が持ち上がるという急な設定にも関わらず23名もの参加者があったということは各社ともプロセス改善が重要でかつ現実的な取り組みになっていることを実感させるものである。今後本会への参加メンバーを中心に組み込み系ソフトにおけるプロセス改善活動を議論する場として、本年10月からSEA-SPINのSIG(SIG-EMS)を発足し活動を継続していくことになっている。今回の情報交換会での熱のこもった議論のように、参加者全員が何かを与え、得られるような議論が今後も継続して行っていくことを期待したい。

最後に、お忙しい中、本稿編集の為に多大なご協力いただいた参加者各位に感謝する。

論文/報告/ツール展示 募集

ソフトウェア・シンポジウム'98

1998年6月17日(水)～19日(金) 於: 大阪市立大学(大阪市・住吉区)



主催: ソフトウェア技術者協会(SEA)

協賛(予定): 日本ソフトウェア科学会 情報処理学会 情報サービス産業協会 電子情報通信学会

今度で18回目を迎えるこのソフトウェア・シンポジウムは、さまざまな場で活躍している技術者や研究者、管理者などが一堂に会し、ソフトウェア技術に関する多面的な経験や知識を交流するユニークで貴重な場を、これまで提供し続けてきました。今回は、3年ぶり関西エリアに戻り、元SEA代表幹事の中野秀男先生の本拠地である大阪市立大学・学術情報総合センターを会場として、来年6月中旬に開催の予定です。

新しい世紀の開幕を目前に控え、インターネットの急激な発展に伴って、コンピュータの応用領域は大きく広がっています。それは、一方で新しい市場やビジネスの機会を創り出すとともに、他方では、社会構造や生活パラダイムの変革を促す波となって、われわれの日常に押し寄せてきつつあります。ソフトウェア技術に対する期待はますます大きく、また責任もますます重くなってきたように思われます。

そこで、今回のシンポジウムでは、「共鳴による発展」を基調テーマとし、それによって、産・官・学その他さまざまなコミュニティに属する人びとの間でのコミュニケーションの拡大を図り、技術やビジネスの新たな発展のきっかけを作ることを目指しています。

前回に引き続き、産業界での地道な研究・改善活動を広くご発表いただく場として、通常の研究論文・経験論文に加えて、事例報告のカテゴリも募集します。事例報告は発表を主体とするカテゴリで、OHP原稿で審査しますので、論文を執筆する必要はありません。応募論文/報告のテーマは、ソフトウェア関連技術全般にわたります。たとえば

・分析/設計技法	・ツール	・開発環境	・ネットワーク	・教育	・メトリクス
・プロジェクト管理	・CSCW	・品質管理	・プロセス改善	・保守	・標準化
・人間の要因	・構成管理	・新パラダイム	・アーキテクチャ	・マルチメディア	セキュリティ

など、ソフトウェアに関連する領域のテーマであれば何でも構いません。先進的な研究テーマはもちろん、開発現場での苦労・努力をまとめた経験論文、事例報告も大歓迎です。奮ってご応募ください。

なお、論文/報告発表セッションは、今回も2トラック構成で、各カテゴリからそれぞれ10編程度を採録する予定です。シンポジウム最終日には、採録された論文の中から最優秀論文賞を選定し、表彰します(賞金20万円、ただし、事例報告は選考の対象になりません)。

【シンポジウムスタッフ】

実行委員長: 中野秀男(大阪市立大学)、田中一夫(山一情報システム)

プログラム委員長: 落水浩一郎(北陸先端科学技術大学院大学)、伊藤昌夫(ニル・ソフトウェア)

プログラム委員: 現在選定中です。

ローカルアレンジメント: 小林修(ダグ)、阪井誠(SRA)

【応募要領】

応募論文/報告は未発表のものに限ります。他への重複投稿もご遠慮ください。論文/OHPは、下記の要領で作成し所定のカバーシートを添付したものを7部、ツール展示についてはカバーシートとツールの内容を紹介する資料を3部、所定の送付先宛に1998年1月31日(必着)までに郵送にてお送りください。プログラム委員会による採否審査結果は、3月下旬に応募者全員に通知します。

<<研究/経験論文>>: 論文にて審査します。論文の書式は自由ですが、用紙はA4サイズに限ります。分量は5～10ページ程度を目安としてください。なお、論文の冒頭に必ず要旨を書いて下さい。

<<事例報告>>: OHP原稿(紙に白黒印刷して下さい)にて審査します。発表時間は15分を予定していますので、OHPは10枚程度が目安となるでしょう。発表要旨(A4用紙で1ページ以内)を添付してください。OHP原稿に要点などを注記していただければ、査読者の理解が深まると思われれます。

なお、論文はできるだけ日本語で執筆して下さい。英文の論文も受け付けますが、OHPおよび発表は日本語に限ります。その他不明な点は、プログラム委員長までお問い合わせください。

主要スケジュール: 1998年1月31日 論文/報告/ツール展示応募締切

1998年3月31日 論文/報告採否通知

1998年4月30日 カメラレディ原稿締切

***** 応募論文カバーシートは次ページにあります *****

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



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

その主な活動は、機関誌SEAMAILの発行、支部および研究分科会の運営、セミナー/ワークショップ/シンポジウムなどのイベントの開催、および内外の関係諸団体との交流です。発足当初約200人にすぎなかった会員数もその後飛躍的に増加し、現在、北は北海道から南は沖縄まで、700余名を越えるメンバーを擁するにいたりました。法人賛助会員も30社を数えます。支部は、東京以外に、関西、横浜、長野、名古屋、九州、東北、広島の各地区で設立されており、その他の地域でも設立準備をしています。分科会は、東京、関西、名古屋で、それぞれいくつかが活動しており、その他の支部でも、月例会やフォーラムが定期的に開催されています。

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

なお、会費は次の通りです：個人正会員一年間8,000円(入会金3,000円)。法人賛助会員一年間1口あたり100,000円(入会金なし、賛助会員会社の社員は各種イベントに正会員に準ずる割引価格で参加できます)。

代表幹事： 坂本啓司

常任幹事： 荒木啓二郎 高橋光裕 田中一夫 玉井哲雄 中野秀男 深瀬弘恭

幹事： 市川寛 伊藤昌夫 大場充 河村一樹 窪田芳夫 熊谷章 小林修 桜井麻里 酒匂寛
塩谷和範 篠崎直二郎 新谷勝利 杉田義明 武田淳男 中來田秀樹 布川博士 野中哲 野村行憲
野呂昌満 端山毅 平尾一浩 藤野誠治 二本厚吉 堀江進 松原友夫 山崎利治 和田喜久男

事務局長： 岸田孝一

会計監事： 辻淳二 吉村成弘

分科会世話人 環境分科会(SIGENV)：田中慎一郎 渡邊雄一
教育分科会(SIGEDU)：杉田義明 中園順三 篠崎直二郎 君島浩
ネットワーク分科会(SIGNET)：松本理恵 小林俊明 人見庸
プロセス分科会(SEA-SPIN)：青山幹雄 伊藤昌夫 坂本啓司 高橋光裕 田中一夫 増井和也

支部世話人 関西支部：白井義美 中野秀男 横山博司
横浜支部：野中哲 藤野晃延 北條正顕
長野支部：市川寛 佐藤千明 小林俊明
名古屋支部：筏井美枝子 石川雅彦 角谷裕司 野呂昌満
九州支部：武田淳男 平尾一浩
東北支部：河村一樹 布川博士 野村行憲
広島支部：大場充 佐藤康臣 谷純一郎

賛助会員会社：ジェーエムエーシステムズ 東芝アドバンスドシステム SRA
PFU 東電ソフトウェア 構造計画研究所 さくらケーシーエス
サン・ビルド印刷 富士通 ジャストシステム 新日鉄情報通信システム
オムロンソフトウェア カシオ計算機 キヤノン 中央システム 安川電機
富士通エフ・アイ・ピー SRA東北 アスキー 新日本製鉄エレクトロニクス研究所
ダイキン工業 東北コンピュータ・サービス オムロン アイシーエス
SRA中国 日本電気ソフトウェア 富士電機 プラザー工業 プロダクト・ソリューション(以上29社)

事務局： 〒160 東京都新宿区四谷3-12 丸正ビル5F

TEL: 03-3356-1077 FAX: 03-3356-1072 E-mail: sea@sea.or.jp

SEA 1996年～1998年の主なイベント（実績と予定）

1996	9/17～18	Workshop at MCC	米国・Austin
	10/9～11	Int'l Symposium on Future Software Technology '96	中国・西安
	10/24～26	第10回教育ワークショップ	八丈島
	11/28～30	第2回ソフトウェア・デザイン・ワークショップ	御殿場
	11/29	分散オブジェクト指向 Forum	神戸
	12/16～17	Int'l Symposium on Science of Computing & Technology of Software	東京
1997	1/22～25	第13回若手技術者ワークショップ	静岡
	1/24	Forum (モバイル時代のサイバーラーニング)	東京
	2/27	Forum (オブジェクト指向の新しい設計方法論)	東京
	3/14	Forum (ソフトウェア・プロセスと開発支援環境)	東京
	4/24～26	第9回テクニカル・マネジメントワークショップ	隠岐
	5/15	Forum (これからの Software Enterprise Management)	東京
	6/18～20	ソフトウェア・シンポジウム'97	博多
	6/20～21	開発ツール & 環境ワークショップ	博多
	6/21	SIGENV ワークショップ'97	熊本
	6/24	Forum (ソフトウェア・アーキテクチャ)	東京
	9/25	Forum (インターネット時代のセキュリティを考える)	東京
	10/16～18	第11回教育ワークショップ	神戸
	10/27	Forum (PGP Nights: 11/10 & 18 も)	東京
	10/29～31	Int'l Symposium on Future Software Technology '97	中国・廈門

	11/12～14	International Conference on Formal Engineering Methods	広島
	11/25	Forum (ファンクションポイント法の導入と活用)	東京
	12/1	Asia-Pacific SPIN Workshop	香港
1998	6/17～19	ソフトウェア・シンポジウム'98	大阪

入会申し込み先

〒160 東京都新宿区四谷3-12丸正ビル5F
ソフトウェア技術者協会 (TEL 03-3356-1077, FAX 03-3356-1072)

SEA 入会申込書（正会員：入会金 3,000 円，年会費 8,000 円） 97-11

氏名： _____ (ふりがな： _____)

生年月日： 19____年____月____日 性別（男 女）血液型（A O B AB）

勤務先名： _____

所属・役職： _____

勤務先住所（〒 _____） _____

勤務先 TEL： _____ - _____ - _____ (内線 _____) FAX： _____ - _____ - _____

自宅住所：（〒 _____） _____

自宅 TEL： _____ - _____ - _____ E-mail： _____

資料送付先 & 連絡先（どちらかにチェック） 勤務先 自宅

SEA 入会申込書（賛助会員：年会費 1口 100,000 円，何口でも可） 97-11

会社・団体名： _____

代表者氏名： _____ (ふりがな： _____)

連絡担当者： _____ (ふりがな： _____)

所属・役職： _____

住所：（〒 _____） _____

TEL： _____ - _____ - _____ (内線 _____) FAX： _____ - _____ - _____

申込口数： _____ 口



ソフトウェア技術者協会

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