

9th ICSE に出席して

第9回ソフトウェア工学国際会議(9th International Conference on Software Engineering, 以下 ICSE9) は、今年の3月30日ー4月2日、米国カリフォルニア州モンタレイで開催された。雨季の明けたカリフォルニアは快適な気候で、海岸線の美しいモンタレイをさらに魅力的にした。それも多少手伝ってか、参加者は1200人を数える盛況で、チュートリアルから600人ももの参加があったと、主催者側は喜んでいて、日本からの参加も多く、おそらく100人近くに達したと思われる。

今回の実行委員長(General Chairman)は W. E. Riddle、プログラム委員長は R. Balzer とわが岸田孝一氏である。プログラム委員はほかに33名の多きを数えた。うち日本からは、阪大の鳥居先生、慶大の斉藤(信)先生。さらに筆者もその末席に連なった。

基調テーマ

今回のプログラムは、基調となるテーマを明確に打ちだしたところに大きな特徴がある。そのテーマとは、「ソフトウェア・プロセスの形式化と自動化」である。技術セッションの初日(3/31)、Balzer がその趣旨を説明した。いわく、「われわれは60年代、70年代に言われたソフトウェア危機については既に解決した。すなわち、期限に合わせて定められた機能のソフトウェアを作ることは、できるようになった。」ここで会場から思わず笑いがもれる。「しかしソフトウェアの保守と進化の過程でのソフトウェア危機は、未だ解決されていない。そのために、ソフトウェアを従来のように製品(product)としてとらえるのではなく、プロセスとしてとらえることが、ますます重要になる。プログラム自身を保守するのではなく、プロセスを保守することにより、柔軟な機能拡張や再利用ができる。滝型モデルは死んだ。数学的方法論から設計的方法論へ、すなわち多人数による長期間の開発に向けた方法への、転換が行われなければならない」などとぶった。

プログラムにはさらに、次のような特徴がある。

- ① このテーマの趣旨に沿って、セッションをプロセス、形式化、自動化の3つの枠にまとめたこと。
- ② 関連する技術として、人工知能とデータベースをとりあげたこと。
- ③ 新規の論文だけでなく、すでに行われたワークショップの報告や、特定分野の最新動向といったセッションも設けたこと。
- ④ 投稿論文は、プログラム委員の少なくとも一人が推薦することという条件をつけ、厳選したこと。

基調テーマを強調する形で最初に行われたのが、L.Osterweil による「ソフトウェア・プロ

セスもまた、ソフトウェアである」(“Software Processes Are Software Too”)と題する招待講演である。筆者の知る限り、これまでの ICSE の招待講演は、外部の分野から講演者を文字通り招いて行われるのが普通で、会議の基調演説という性格のものはあまり例がないのではないかと思う。一昨年(2007)の第 8 回 ICSE では、まったくの外部ではないが、ソフトウェア工学とは分野を異にする Dana Scott が招待講演を行った。San Diego の時には(第 5 回)、N. Wirth の講演もあったが、一方でノーベル賞受賞者の生物学者による分子生物学の話もあった。

さて Osterweil の講演だが、その要点を示せば次のようになろう。ソフトウェア・プロセスにとって、プロセスの手順をなるべく厳密に記述すること、そしてその記述を個々のケースに適用して実行できるようにすることが、重要である。このためには、プログラミング技術を応用し、プロセスの記述もソフトウェアとして扱えるようにすればよい。Osterweil はこの考え方に基づいたソフトウェア開発のパラダイムを提唱し、さらにテストリングの簡単なプロセスについて、プロセス・プログラミングの例を示した。

この講演に対し、M. Lehman がコメンテータとして意見を述べた。ソフトウェア・プロセスのうちで、アルゴリズム的な記述に向くステップはごく一部である。多くのプロセスは予測しがたい性質をもち、そこに人間の工夫が要請される。プロセス・プログラミングという概念は、ソフトウェア・プロセスが決定的(deterministic) という誤解を与える、といふかなりもつともな批判である。Osterweil は、「プロセス・プログラミングの例として、現在は手続き的な記述をとっているが、これが最善と主張している訳ではない。ルールなどによる宣言的な記述に向く場合もあろう。」などと答えた。

会場からの質問を受付けたところ、最初に立ったのが H. Mills である。彼の質問は、いきなり、この会場にチューリング賞受賞者がいたら手を挙げてほしい、という唐突な問いから始まった。該当者がいないことを確認したうえで、Mills は、ICSE からチューリング賞クラスの知的指導者が何故か排除されてきているという。ICSE の当初は、Dijkstra, Hoare, Gries 等がいたのに、なんらかの社会的なプロセスによって、反知性(anti-intellectual) の方向に来てしまったのではないか。もう彼らから学ぶことはないのか。という強烈なパンチを、あの Mills が浴びせたのである。会場にたちまち緊張が走る。

Osterweil が多少の動揺を見せながらも、さすがにこういう議論に慣れたアメリカ人らしく、弁じた。しかし、Mills の批判は会議の運営に対して向けられたものだから、かみあはずもない。もともと Osterweil が答える必要はなかったのである。Mills も、これは主催者に言っているのだ、というようなことを付け加えた。とにかくこのやりとりから、米国のソフトウェアの学界にも、こみいった勢力争いがありそうなことが、想像された。

形式的仕様言語

続いて一般セッションに入る。まずセッション 2「形式的仕様言語」に、出た。最初の発表は、E. Berliner と P. Zave による “Experiment in Technology Transfer: PAISley Specification of Requirements for an Undersea Lightwave Cable”。Zave が開発したとくに実時間分散処理の記述に向く動作的な(operational)仕様記述言語 PAISley を、海中光ケーブル通信の制御システムの要求記述という実際的な問題に適用した、その体験報告である。筆者は去年の 10 月に今回の ICSE への投稿論文を実に 16 編査読させられたが、この論文はその中の 1 編で、なかでは一番いいと思い、ただひとつ合格点を与えた。技術移転というあまり論文にしにくいテーマであるが、プレゼンテーションがひじょうにうまい。ただし実時間システムの記述に向くという PAISley の特徴が活かされた対象でないことから(この点についても発表者は誠実に報告したが)、技術的にはやや物足りない感もあった。後で Zave に、このプロジェクトの結果から PAISley の言語仕様を見直す必要は起こらなかったかと聞いたら、問題点はあったが PAISley にはもうずいぶん長くかかずらわっているので、いいかげんに次のことをやりたいため、言語をいじるようなことはしていない、といていた。

同セッションの次の発表は、二木さんらによる “Parameterized Programming in OBJ2” である。二木さんは 1984 年に S R I に客員研究者として行かれ、J. Goguen 等と一緒に仕事をされたが、これは ETL と SRI の共同によるその後の研究成果である。代数的仕様記述システムである OBJ2 に組み込まれている、パラメータ化プログラミングの考え方を述べたもの。この考え方を、簡単な例で示せば次のようになる。ソートという概念を考えよう。OBJ2 では、これを **object** として定義できる。ソートの対象となるデータは、もっとも一般的には全順序の性質を持てばよい。全順序集合は、**theory** として定義できる。ソートに対し、全順序集合はパラメータとして働く。たとえば整数を全順序集合としてとり、ソートのパラメータをこれで実体化 (**instantiate**)すれば、整数のソート・モジュールが得られる。整数を全順序集合とみなすことを正当化する仕組みとして、OBJ2 には **view** という構成要素がある。

次は、D. E. Perry (ベル研) による “Software Interconnection Models”。これは、優秀論文として表彰された 2 編の論文のうちの一つである。内容は大きく二つの部分に分かれる。前半は、ソフトウェアの成育 (**evolution** の試し訳。「進化」ではピンとこないだろう) モデルとして、きわめて一般性の高い相互結合 (**interconnection**)モデル(IM) なるものを提唱する。既存のいくつかのモデルを、IM をさらに分類した二つのモデル、すなわち単位 (**unit**) IM と統語的 (**syntactic**) IM として、説明する。後半では、第 3 の IM として意味的 (**semantic**) IM と呼ぶものを提出し、その具体的な方法を示す。こういう概念的な整理は日本人は余り得意でないが、欧米では評価されるようだ。

ソフトウェア・プロセスの実験的研究

こんなペースで書いては長くなるばかりなので、以降はかなりはしょって紹介する。1日目のセッションで他に比較的面白かったのは、「ソフトウェア・プロセスの実験的研究」と名づけられたパネルである。座長はMCCのW. Curtis, パネリストはMCCのH. Krasner, エール大学のD. Littman, スタンフォード大学兼ゼロックスPARCのJ. Tangである。パネルといっても、各パネリストによる発表が中心で、一般のセッションとあまり変わらない。

Krasner の発表は、実際的なソフトウェア・プロセスのあり方を探るために、大規模システム開発プロジェクトの事例を集め、実験的な分析を行った結果の報告である。事例としてMCCのスポンサー会社の19のプロジェクトをとり、インタビュー調査を行った。その結果として、従来のプロセス・モデルでは重視されていないチームワーク、インフォーマルなコミュニケーションを通じての設計状況に関する共通認識、学習、交渉、顧客との意思疎通、といったプロセスの重要性が明らかになったとしている。

Tang の発表は、グループによる概念設計の過程を観察することにより、どのような要因が意思決定に寄与し、設計作業の進行につながるかについての、調査報告である。ジェスチャがよく使われていること、黒板のようなものより共通の作業機のほうが効率的であること、などの話が印象に残っている。またLittmanの話は、スペース・シャトル関係のソフトウェア開発を題材に、そこで行われた査閲 (inspection) をビデオに撮って分析した結果の報告であった。

こういうパネルをみると、米国でもグループによる意思決定プロセスに注目しているという印象を受ける。日本では、要求分析のフェーズでよく行われる合宿討議などで、こういった問題をかなり取り上げてきているところが多いのではないだろうか。Krasnerとは、翌日、水族館で開かれた招待会でいろいろ話をする機会があった。MCCはいわば寄合い所帯だから、とくにチームワークといった問題意識が強いのかもしれないが、彼の話では、米国一般にそういう点への注目が高まっているという。

形式的な方法の実践的適用について

形式主義 (formalism) を扱ったセッションがかなりあったので、それらをまとめて紹介しよう。筆者が出たものに、以下がある。

- ・ 招待講演 : D. Bjorner, 「ソフトウェア開発における形式主義の利用 : プログラムとメタプログラム (The Use of Formalisms in Software Development: Programs & Meta-Programs)」。

- ・ セッション 12 (パネル) : 「変換技法の最先端 (State-of-the-Art in Transformation Technology)」。座長 : W. Scherlis, パネリスト : D. Bjorner, M. Feather。
- ・ セッション 15 (パネル) : 「形式的仕様を日常的に用いること (Putting Formal Specifications to Everyday Use)」。座長 : P. Zave, パネリスト : D. Good, C. Jones, M. Meller-Smith, W. Scacchi, I. Sorensen。

他に、V. Stenning (Imperial Software Technology) の招待講演「環境の役割 (プログラムでは「ソフトウェア・プロセスの自動化」という題になっていたもの)」、および D. Barstow のチュートリアル的な発表の後に S. Fickas (プログラムでは C. Williams になっていたが交代) と B. Boehm がコメントするような形式のパネル「人工知能とソフトウェア工学」も、関連が深い。

Bjorner の講演や Zave のパネルから感じたことは、形式主義の実践的利用については、米国よりもヨーロッパの方が進んでいるという印象である。Bjorner や Jones のウィーン技法 (Vienna Software Development Method, 略して VDM) が当人達から紹介されたこともあり、米国では理論と実践が分離しているのに、西欧では両者が比較的融合しているという感を聴衆に与えた。VDM は、米国や日本ではまだなじみがうすいだろう。

Scherlis のパネルは、概念論に終始して面白くなかった。Scherlis とは、後に打上げパーティーでゆっくり話をする機会があった。彼と Scott の推論的プログラミング (inferencial programming) に関する論文は、日本でよく読まれていることを言い、その後進展があるのかどうか尋ねた。Scherlis によれば、大いに進んでいるという。6・7 編の論文をその後書いているから送ると言っていたが、未だに届いていない。

Barstow の話もよく整理されてはいるが、あまり新味はなかった。Boehm が、今の AI ゲームを多少皮肉って、現在はやりのシステム開発手順をフローチャートで示した。それによると、まず問題が難しいかと聞き、易しければ従来型のシステム開発を行う。難しければ、AI でやることにして LISP マシンを買ったり、AI セミナーに人をやったりしたうえで、知識工学アプローチによる開発を試みる。うまくいけばそれでよし、だめなら従来型の開発に戻るといふものである。

日本からの発表

二木さんの発表についてはすでに述べたが、他にも日本人の発表がいくつかあった。寺本さん (日本電気) は、セッション 8 「信頼できるソフトウェアの作成プロセス」というパネルで、「ソフトウェア開発の各プロセスの担当者が、そこで発生の予想されるエラーの数

を見積り、前のプロセスから持越していると想定されるエラーの数を加え、そこから実際にそのプロセスで発見したエラーの数を差引いて、それを次のプロセスに渡す。最終結果と見積りのずれを各担当者にフィードバックして、見積りの精度を上げるとともに、エラー発見の率を高める。」というきわめて実戦的な品質管理の方法と実践結果を発表した。このパネルの他の二人（N. Levenson と P. Neuman）が、安全なソフトウェアを作ることがいかに難しいかという観念論ばかり述べて、聴衆を僻易させたのとは対照的で、よかった。

論文としては、宮崎、村上両氏（富士通）の「誤差に基づくソフトウェア計量（Software Metrics Using Deviation Value）」、野村さん（日本電子計算）の「日本におけるソフトウェア工学ツールの利用状況（Use of Software Engineering Tools in Japan）」があった。パネルには、「既存の環境のサンプリング」に松本さん（東芝）が、「将来の環境の動向」に横井さん（ICOT）が、「ソフトウェア・プロセスの改良アプローチ」に岸田さん（SRA）が、参加された。

最終パネル

会議をしめくくるパネルは、Balzer も言ったようにテーマとして異質で全体のプログラムからはみだした、知的所有権保護の問題を扱うものであった。演題を、「個人および国際的所有権：問題はいかに深刻か」という。個人と国際をならべたところが不自然であるが、実は前半と後半に分れていて、前半で3人のパネリストが個人の所有権の問題を論じ、後半でやはり3人のパネリストが国際問題を論じた。前半の座長を勤めた女性弁護士(Pamela Samuelson)は、なかなかものの分った人のようだった。「現在、ソフトウェアの著作権に関連した裁判例は多いが、裁判官がソフトウェアについてよく知らないため、ソフトウェア業を保護するつもりで、かえってソフトウェアの流通や標準化の障害となり業界に不利益をもたらすような判例が、出されている。今のうちに、ソフトウェア業界が正しい主張をしておく必要がある。」といった指摘をしていた。また、聴衆からの質問のなかに、「東南アジアで著作権を保護していない国があるが、そういうところの海賊版封じにうまい対策があるか？」というのがあったが、それに答える前に、19世紀の米国は著作権を認めず、ヨーロッパのあらゆる本を複製して利用していたという歴史的事実に注意を喚起したことも、印象的であった。かなりリベラルな人らしい。

後半の座長は、最近までSRIの幹部だったD. Brandinである。ACMの会長を勤めたこともあり、日米の先端技術比較のレポートをACMから出しているが、それを読まれた人も多いただろう。現在は、Strategic Technologies, Inc.という会社を作って率いているらしい。後半の議論は、時節がら、ややJapan Problemの色彩を帯びた。聴衆からヨーロッパの人が一人立って、議論がアメリカ側に偏していると穏やかに抗議したが、日本人は自分も含めとくに発言しなかったのは、やはり反省すべきであろう。

ツール・フェア

ツールおよび本の展示が併設されたばかりでなく、新しい試みとして展示中のツールの説明をするセッションが、一般セッションと同時に開催された。

商用ツールに目立つ傾向は、ソフトウェア開発CADと称するような、図式を用いた設計開発支援システムの数が多いことである。たとえば、Cadre Technologies の Teamwork, Chen & Associates の ER-Designer, I.G.L. の SPECIF-X, Interactive Development Environments の Software through Pictures, Reasoning Systems の Refine など。

日本からは、慶応大学、静岡大学、SRA が参加。いずれもソニーのワークステーションを使って、UNIX 上のツールを見せた。

まとめ

ICSEに限らずソフトウェア工学の会議に出るたびに思うことは、ソフトウェア工学は難しいということである。計算機科学とソフトウェア開発の現場の間にあって、技術的にも面白く実践上も役に立つ仕事をするには、なかなか大変である。今回も、全体に概念的な話が多く、もう少し技術的な問題に特化するか、あるいは実際的な話題に徹するかした方がよいのではないかと、思うことが多かった。