

第21回ソフトウェア工学の自動化国際会議 (ASE2006) 開催および参加報告

鷺崎 弘宜¹ 久保 淳人² 下滝 亜里³ 中川 博之⁴ 林 晋平⁵ 丸山 勝久⁶ 本位田 真一^{1,4}

¹ 国立情報学研究所 ² 早稲田大学 ³ 南山大学 ⁴ 東京大学 ⁵ 東京工業大学 ⁶ 立命館大学

2006年9月に東京にて第21回ソフトウェア工学の自動化国際会議(ASE2006)を開催および参加したので、取り上げられた主な内容を紹介する。会議の傾向として、モデル検査/記号実行に基づくプログラム解析/検証の取り組み、および、プログラム変更履歴からの特定の情報(例えばアスペクト)発掘の取り組みが多く見られ、両分野への取り組みの活発化を伺えた。会議には約220名の参加があり、国内外の研究者がソフトウェア工学自動化の最先端の取り組みについて議論し交流する良い機会となった。

Report on the 21st IEEE/ACM International Conference on Automated Software Engineering (ASE2006)

Hironori Washizaki¹ Atsuto Kubo² Asato Shimotaki³ Hiroyuki Nakagawa⁴
Shinpei Hayashi⁵ Katsuhisa Maruyama⁶ Shinichi Honiden^{1,4}

¹National Institute of Informatics ²Waseda University ³Nanzan University
⁴University of Tokyo ⁵Tokyo Institute of Technology ⁶Ritsumeikan University

This paper reports major topics of the 21st IEEE/ACM International Conference on Automated Software Engineering held at September 2006 in Tokyo. There were many presentations on program analysis/verification and pattern/aspect mining.

1 はじめに

本稿では、2006年9月に東京にて開催された第21回ソフトウェア工学の自動化国際会議(21th IEEE/ACM International Conference on Automated Software Engineering[1]: ASE2006)において取り上げられた話題と開催経緯を紹介する。我々は同会議について、開催者/運営者および参加者の各立場で関与した。それぞれの立場からの紹介を通じて、執筆時点における最先端のソフトウェア工学自動化研究の国際的傾向の一部を概観する。また同時に、ASEおよび類似の論文採択率の厳しい国際会議(例えばICSE[2]やFSE[3])への今後の活発な論文投稿と参加を促すことも目的とする。

ASEは、自動生成/検証ツールをはじめとして、ソフトウェア工学分野における自動化に関するあらゆる題材を扱い、1980年代より継続して開催され21回を数える歴史ある国際会議である。ASE2006は、ACM SIGSOFTやIEEE Computer Society他

の主催により、著者の一人である本位田実行委員長
の組織化のもと、東京・学術総合センターにて9月
18日から22日まで(本会議は20日から22日まで)
の日程で開催された。国内外から合わせて約220人
(併設ワークショップ等も含めた総数)の参加があり、
ソフトウェア工学自動化の最先端の取り組みに
ついて議論し交流する良い機会となった。

本稿では以降において、ASE2006を開催および
参加した筆者らにより、会議の開催経緯、論文採択
のための注意事項、および、本会議および併設イベ
ント等で取り上げられた話題のうちで主要なものを
紹介する。

2 ASE2006のプログラム概要

2.1 ASEの位置づけと開催経緯

ASEの特徴は上述のようにソフトウェア工学に
おける自動化のあらゆる話題を扱うことにある。ソ

ソフトウェア工学を扱う種々の国際会議の分類を以下に示す。

1. ライフサイクルプロセスの特定の工程を扱い、技術について横断的な会議。例として、要求工程を扱う RE[4]、保守工程を扱う ICSM[5]、テスト工程を扱う ISSTA[6] など。
2. 特定の技術/側面を扱い、工程について横断的な会議。例えばアスペクト指向技術を扱う AOSD[7] など。
3. ソフトウェア工学全般を扱い、工程および技術/側面の双方について横断的な会議。運営コミュニティの地域的なまとまりに応じて存在し、例えば ICSE (世界規模)、FSE (世界規模)/ESEC (ヨーロッパ) [8]、APSEC (アジア太平洋) [9] などがある。

ASE は、自動化という特定の側面を扱うという意味では分類 2 に属する。しかしながら、ソフトウェア開発の効率化/高品質化を目指すうえで特定処理の自動化は自然かつ根源的な取り組みであり、ソフトウェア工学研究の多くはソフトウェア (正確にはソフトウェアの 4P: プロダクト/プロセス/ピープル/プロジェクト [10]) の定義や管理作業の自動化を直接または間接に扱うため、ASE が扱う技術と工程は分類 3 に匹敵する広さを持つこととなる。結果として ASE は、ソフトウェア工学領域において ICSE、FSE/ESEC に次ぐ権威上第三位の国際会議と認知されるに至っている。

歴史的経緯を見ると、ASE は元々 Knowledge-Based Software Engineering Conference (KBSE) と題して、人工知能をはじめとする知識処理技術のソフトウェア工学における応用を扱う会議であった。その後、話題が知識処理に限らず自動処理全般に広まったことに応じて、1997 年より現在の "ASE" に名前を変えて開催されてきた。本稿で扱う ASE2006 は、日本およびアジア周辺地域におけるソフトウェア工学自動化研究の高まりを受けて、ASE 運営コミュニティと (筆者の一人である) 本位田実行委員長を中心とした実行組織により北米/ヨーロッパ以外で初めて開催された。また、"ASE" に会議名を変更してから 10 年目にあたる記念の回もなった。

2.2 論文選考過程と論文を採択させるために

ASE2006 では論文査読と会議プログラム編成のために、2 名の委員長と 40 名の委員でプログラム委員会が組織されている (他に、13 名の査読専門委員がいる)。委員は、採録論文を決定するプロ

ラム委員会への出席が義務付けられており、各自査読を行った論文に関する判定理由や意見を必ず発言することになっている。査読報告において判定が割れた論文に関しては、査読者間で電子メール討論を行う期間がプログラム委員会会議の前に設定されており、それぞれの査読者の立場や意見を互いに確認しておくことができる。また、プログラム委員会会議に向けて、各論文ごとに討論リーダー (査読者のうち 1 名) が割り当てられる。会議当日には、リーダーは採否決定議論の冒頭で、各査読者から報告された論文の要旨、利点、欠点をまとめて述べる。その後、各査読者が判定意見を述べ、質疑応答を経て、全体の合意で採否を決定する。以上が論文の選考過程である。

以降は、筆者のひとりである丸山がプログラム委員として ASE2006 に関わった経験から、論文を採択させるための戦略に関して個人的な意見を述べる*。周知のように、ASE でのフルペーパーの採択率は非常に低く抑えられている†。採択率の低い会議での採録を目指す際に大事な点は、平均的な点数を狙うのではなく、採録を強く支持する査読者 (advocate) の存在を狙うことである。特に、ASE のようにプログラム委員の出席が義務付けられている会議の場合、当日の会議での査読者の意見が強く反映される。極端に言えば、査読報告において判定が割れていても、採録を支持する査読者が採録を支持する査読者を説得でき、会議参加者全体から反対意見がなければ、最終判定は採録となる。また、一般的に、論文の欠点を探すのは容易であるため、強い採録支持者がいなければ、その論文が採録となる可能性は高くなる。

論文が採録されるためには、提案手法に新規性が含まれ、さらにその内容が正しく記述されていることは当たり前である。その上で、単なる採録支持 (accept, but could reject) ではなく、強い採録支持を得るためには、有効性や必要性を示す例題や評価実験結果を織り混ぜ、その分野における論文の重要性、貢献や利益を明確にすることが重要である‡。また、該当分野で存在感を高め、研究の重要性や貢献を示すには、フルペーパー採録をいきなり狙うのではなく、ショートペーパー採録やポスター採録を活用することを考えるべきである。幸なことに、ASE では、ショートペーパー採録やポスター採録により、

*本来、このような記事が書ける、あるいは、この記事の内容が正しいのであれば、著者自身がかつと多くの国際会議で採録されているはずなのだが。

†2004 年は約 14%(25/183)、2005 年は約 9.6%(28/291)、2006 年は約 18%(22/121)。

‡個人的には、日本人の記述する論文には、motivating examples, contribution, benefits の明確な記述が少ないと感じている。

論文の発表機会を減らさないように工夫している[§]。採択率の低さに奮えて採択をあきらめずに、ぜひ投稿に挑戦してほしい。

2.3 本会議の構成

本会議 3 日間にそれぞれ行われる 3 件の基調講演を除いて、2 並列トラックにより構成され、それぞれに扱う論文の種類や採択基準/採択率が異なる。本会議の構成トラック/セッションを以下に示す。

(1) 研究論文: 上述の選考過程に従って厳選された 22 件のフルペーパー発表、および、17 件のショートペーパー発表。121 件の投稿があり、フルペーパーの採択率は 18% と厳しいものであった。

(2) ツールデモ: 「動く」「見える」最新研究成果としての 9 件のデモンストレーション発表。投稿 9 件、採択率 100%。

(3) パネル討論: "ASE" に名前を変更して 10 年目を記念して、パネル討論 ASE Retrospective が開催された。ASE のこれまでの開催やコミュニティ発展の経緯を振り返った後に、今後の方向性や発展について議論が交わされた。

(4) ミニチュートリアル: モデル検査フレームワーク Boger、および、ソフトウェア工学における評価手法に関するそれぞれ 90 分単位の小講義。

2.4 併設イベント他

ソフトウェア工学自動化技術の進化および多様化に対応して、多数のワークショップやチュートリアル、シンポジウムが本会議の前に開催された。以下の併設イベントは博士課程シンポジウムを除いて、それぞれ特定の比較的萌芽的な分野を深く取り上げて、同分野における研究の発展や多様な分野からの参加を促すことを目的とする。

(5) ワークショップ: 2 件の国際ワークショップおよび 2 件の日本語ワークショップが開催され、それぞれ本会議とは独立に論文募集、査読、採択、論文/ポジションペーパー発表が行われた。各ワークショップは、ソフトウェア工学の自動化に関して最近の注目される 1 つの話題 (ホットトピック) に特化して、比較的少人数による集中的議論によりその分野における研究の発展とコミュニティ育成を促す役割を担っている。そのため各ワークショップでは、本会議の各トラックと比較して未完成ながらも非常に斬新な手法や着眼点が披露されることも多く、ソフト

[§]2004 年は 26 編、2005 年は 35 編、2006 年は 17 編が採録されている。

ウェア工学自動化および周辺領域の今後の方向性を捉える上で貴重な情報源である。

(6) チュートリアル: 全 4 件の半日チュートリアルが本会議の前に開催された。

(7) 博士課程シンポジウム: 博士課程の学生による研究テーマや途中成果に関する 6 件の発表。研究の方向性や内容、手順などについて他の学生や経験をつんだ研究者/教員と議論しアドバイスを受ける貴重な機会として、毎年開催されている。

2.5 ソーシャルイベント、表彰その他

ソーシャルイベントとして、本会議前日の夜にレセプション、本会議 2 日目にバンケットが開催され、それぞれ多数の参加者が交流し歓談する良い機会となった。バンケットでは開催地域における文化的な催し物を楽しむことが慣例であり、ASE2006 では殺陣や和太鼓が演じられて好評を博した。その様子を図 1 に示す。

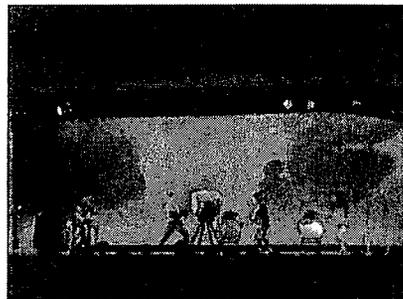


図 1: バンケット風景

バンケットでは以下の表彰も行われた。まず、ASE2006 における最優秀論文に授与される ASE Best Paper Award は以下に送られた。

- P. Weißgerber and S. Diehl, "Identifying Refactorings from Source-Code Changes"

また、優秀な論文に授与される ACM SIGSOFG Distinguished Paper Award は以下に贈られた。各論文の概要は後述する。

- N. Volanschi, "A Portable Compiler Integrated Approach to Permanent Checking"
- L. Murta, C. Werner and A.V.D. Hoek, "ArchTrace: Policy-Based Support for Managing Evolving Architecture to Implementation Traceability Links"

3 本会議の話題

3.1 基調講演

基調講演は、大学よりソフトウェア形式仕様/検証の研究者、産業界よりソフトウェア自動解析の研究者、そして、周辺領域として組込み/ロボティクス分野の研究者の計3名が招かれ、バランスの良い内容であった。基調講演の様子を図2に示す。

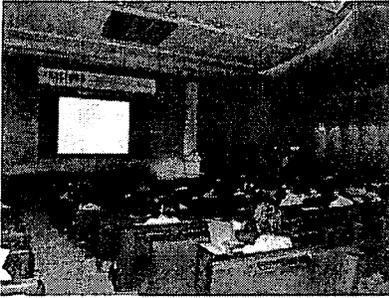


図 2: 基調講演の様子

(1) "Verifying Specifications with Proof Scores in CafeOBJ", K. Futatsugi

北陸先端大学院大学の二木教授より、代数仕様言語 CafeOBJ[11] における検証譜 (Proof Scores) による仕様検証に関する講演があった。CafeOBJ は、OBJ 言語を発展させて、講演者のグループが 10 年以上の活動の中で開発し提供してきた代数ベースの実行可能な形式仕様言語およびその処理環境である。講演では、仕様検証の基本的な話題に始まり、対象システムの要求や仕様の抽象化に基づく形式モデルの作成、形式仕様言語による仕様の作成、記号実行による仕様の性質チェック、および、項の簡約や定理証明による仕様特性の証明等を CafeOBJ 環境で実施する方法と利点および応用例が説明された。

近年、ソフトウェアシステムに対する信頼性要求の高まりに伴い、仕様を厳密に定義し検証する実用的技術が世界的に強く求められており、講演はその要請に答えるものであった。

(2) "Winning the DARPA Grand Challenge", S. Thrun

DARPA Grand Challenge[12] は、米国防総省国防高等研究事業局 (DARPA) が主催する無人ロボットカーレースである。講演では、Stanford 大学の Thrun 教授による DARPA Grand Challenge への参加報告と技術解説が行われた。Thrun 教授がリーダーを務める Stanford Racing Team は同レースに参加し、人間による介入なしに約 132 マイルを走

破して優勝した。同チームが使用したロボットカーには、センサ入力の解釈と各部制御に機械学習とマルコフ過程に基づく確率モデルが利用されており、講演では、上述の機械学習アルゴリズムについて解説が行われた。またシステム全体のアーキテクチャは、多層 (Layers)、コントロールループ (Control Loop)、および、パイプライン (Pipes and Filters) を組み合わせたものであり、未知の状況/環境変化にリアクティブ対応可能な組込みシステムとして一般的なものと考えられる。講演では、チャレンジ達成のために採用した策として、以下などが挙げられた。いずれも信頼性の高い組込みシステム開発において従来から指摘される重要な実践項目であり、それらへの堅実な取り組みと積み重ねが重要であると感じられた。

- システム統合に最初から注力
- 幅広いテスト
- ツールによるバグの自動サーチ
- 厳格なコードガイドライン適用
- リポジトリの活用
- 学習による障害物対応
- 要素技術の洗練 (カメラ画像の範囲拡大、画像認識の向上など)

なお、コンテストとしてみた場合に類似の試みとして日本では、主に組込みソフトウェア技術者への教育機会提供を目的として、LEGO ブロックで組み立てられた自律走行ロボットの制御ソフトウェア開発コンテスト (ET ロボットコンテスト [13]) や、小規模な飛行船制御システムの開発を参加者間で競うコンテスト (MDD ロボットチャレンジ [14]) が開催されてきている。しかし DARPA Grand Challenge は、国の全面的な支援の下で現実の車と広大な土地そのものを扱うものであり、国をあげての研究開発および技術発展支援のスケールの違いを痛感した。

(3) "Automatic property checking for software: past, present and future", S. K. Rajamani

Microsoft Research India の Rahamani 博士から、ソフトウェアのモデル検査についての講演が行われた。Rajamani 博士は、モデル検査、定理証明、および、反例駆動型詳細化の各種技術を用いた C 言語プログラム静的解析ツール SLAM[15] の主開発者の一人であり、同 SLAM の具体例を伴う解説が行われた。講演では、モデル検査に関する最新の研究動向が報告された。また、今後の展望として、各種の統合開発環境に (構文誤りと同様に) 意味誤りを指摘するようなモデル検査ツールが統合されるようになるだろう、との見通しが示された。

3.2 研究論文

採択された研究論文フルペーパーの分野内訳を表1に示す。前年 ASE2005 との主な違いは、アスペクト指向技術のみを単独で扱うセッションがなくなったこと、および、自動的な解析/検証系やマイニング/リバースエンジニアリング系のセッションが多数設置されたことである。特に後者の両分野について、モデル検査/記号実行に基づくプログラム解析/検証の取り組み、および、プログラム変更履歴を活用した特定情報(例えばリファクタリングやアスペクト)発掘の取り組みが多く見られ、近年これらへの取り組みが国際的に活発化していると考えられる。

また、ASE における一般的な傾向として、各研究論文の成果は(当然ながら)全て何らかの自動化された実装を伴い、純粋に理論のみで閉じた成果はほぼ存在しえず、近い将来においてツール等の形式により実用化を高く期待できるものが多い。

扱われる対象領域は、小規模システムから分散/DBシステムまで多岐にわたり、自動化技術適用の広がりや領域ごとの堅実な積み重ねを見ることができた。研究論文の発表の様子を図3に示す。

表 1: 研究論文フルペーパーの傾向

分野	件数
プログラム解析/検証	5
マイニング/リバースエンジニアリング	4
モデリング/合成	3
アーキテクチャ	2
テスト	2
影響解析	2
追跡性	2



図 3: 研究論文発表の様子

以降において、筆者らが聴講した論文発表の幾つかを取り上げる。

(1) "Identifying Refactorings from Source-Code Changes", P. Weißgerber and S. Diehl

バージョン管理システム(CVSリポジトリ)から取得できる対象プログラムの開発履歴上で「メソッドの移動」といった原始的リファクタリング群および複合リファクタリングが行われた時点/箇所を特定する手法とその実装が提案された。提案手法は、各リファクタリングについて実施前後にプログラムが満たすべきシグネチャベースの条件を事前定義し、条件が変更トランザクションにおいて満たされる場合に、リファクタリングの必要条件(外部に対する振る舞いが実施前後で保たれていること)を満たすかどうかをコードクローン解析により近似的に判定し、最終的にリファクタリング候補群を順序付けて提示する。論文では、Tomcat等の現実かつ大規模なプログラムに対して提案手法が有効に機能することを実験的に確かめている。

該当論文は ASE Best Paper Award を受賞しており、提案手法の独創性や、有効性を緻密に検証済みであることなどが高く評価されたと考えられる。

(2) "Mining Aspects from Version History", S. Breu and T. Zimmermann

プログラムにおける横断的な関心事対応部分が、時として、最初から存在するのではなく、開発の途中における何らかのプログラム拡張(あるいは修正)をきっかけとして追加されることに着目し、バージョン管理システムから得られる変更履歴情報より、そのような「横断的な変更」(cross-cutting changes)群をアスペクトの候補として自動的に特定する手法とその実装が提案された。特定にあたり、プログラム所有上の近接性(変更トランザクション実行者の同一性)や時間的な近接性(トランザクション実行時間の近さ)に基づいて、メソッド呼び出しに関する横断的な変更の断片群を、有用な大きさにまとめあげる手法を導入している。さらに、Eclipse等の現実かつ大規模なプログラムに対して提案手法が有効に機能することを実験的に確かめている。

(3) "A Portable Compiler-Integrated Approach", N. Volanschi

ソースコードの静的チェック機構を組み込んだ gcc (GNU Compiler Collection) 拡張、mygcc が提案された。現在世の中に存在する多くの解析ツールはどれも広く利用されるに至っていないことを指摘し、より実用的なツールの必要性が語られた。提案手法では、コードパターンにより指定された statement 間で成り立つべき状況を表現した Constrained Reachability Queries (CRQ) と呼ばれるクエリを記述し、gccでのコンパイル時にチェックを行う。これにより、「メモリ確保失敗時のチェック漏れ」や「close後のファイルの読み込み」等の状況を検出することができる。gccへはソースコードにして1000

行強の非常に小さな修正・追加で解析機能を組み込んでいるにも係わらず、既存のツールに劣らない解析能力を持つことが示された。

該当論文は、ACM Distinguished Paper Awardを受賞した。明快に示された解析戦略の選択理由と実装方法、著名なコンパイラへの組み込みによる利用のしやすさ等が評価されたと考える。

(4) "ArchTrace: Policy-Based Support for Managing Evolving Architecture to Implementation Traceability Links", L. Murta, C. Werner and A.V.D. Hoek

ソフトウェアの進化に応じて、アーキテクチャ記述内の各要素のバージョンと、要素を実現するコード群のバージョン群がそれぞれ更新されていく状況において、それぞれのバージョン更新時に追跡リンクを更新（あるいは更新しない）といった幾つかの代表的なポリシーを準備し、その事前選択によって半自動的に、アーキテクチャ記述とコード間の適切な対応関係を保つ手法とその実装が提案された。また、実際のソフトウェア開発プロジェクトにおける履歴に照らし合わせた結果、提案手法が有効に機能することを確認している。

該当論文は ACM Distinguished Paper Award を受賞しており、提案手法の着目点や緻密な有効性検証の結果などが高く評価されたと考えられる。

3.3 デモンストレーションとポスター展示

本セッションは、各研究論文セッションにおけるショートペーパー全17件のポスター発表と、ツールデモセッション全9件のデモンストレーションにより構成される。会議参加者が積極的な意見交換が出来ることを目的として開設されたセッションである。セッションの様子を図4に示す。

開催に当たっては、事前の各研究発表セッションにおいてショートペーパー著者それぞれに5分のアピール時間が与えられるといった運営者側の配慮も見られた。当日はセッションの2時間を通して常に各ポスターの前で活発な議論がなされ、運営者がセッション終了を促すといった異例の盛況振りであった。

本セッションではフルペーパー発表者がポスター発表者の研究に関してコメントしたり、異分野間での研究内容の相互紹介などが見られ、発表者・参加者とも自身の研究に対する新しいアイデアとモチベーションを得るための絶好の機会になったと思われる。



図4: ポスター展示の様子

4 ワークショップ

ワークショップテーマは以下の4件であった。これらのうち、筆者らが参加した2件について詳しく報告する。

- アスペクト指向ソフトウェア開発 (2回目): アスペクト指向開発の特に自動化に関する議論
- ソフトウェア開発における知識協調 (コラボレーション) 支援 (2回目): ソフトウェア開発に必要な広範囲に渡る知識の協調方法と自動的な支援に関する議論
- Web2.0時代のソフトウェア開発環境 (初回, 日本語)
- 要求工学ツール (初回, 日本語)

(1) Web2.0時代のソフトウェア開発環境 (Web-SDE: Japanese Workshop on Leveraging Web2.0 Technologies in Software Development Environments)

本ワークショップでは、近年盛んに取り上げられる、新しいWebという意味合いを持った「Web 2.0」の風潮に触発されたソフトウェア開発環境に焦点が当てられ、その優位性と問題点が議論された。このワークショップでは、Webや開発環境に関心のある日本の研究者や実践者を集め、広く意見交換を行うことも目的としており、発表時間とは別に十分な議論時間が割り当てられた。

今回のワークショップでは2つのセッションが立ち、計6件の発表が行われた。午前のセッションでは「ソフトウェア開発支援」と題し、主にウェブ上に構築されたサービス等を利用したソフトウェア開発環境が扱われた。また、午後のセッションでは「ソフトウェア理解」と題し、主に開発者の持つ集合知を利用・共有する手法が議論された。参加者は十数名で、セッションでの議論は大変活発で濃密なものとなった。

(2) 要求工学ツールワークショップ (Japanese Workshop on Requirements Engineering Tools; JWRET)

本ワークショップでは、招待講演が1件、要求工学支援ツール(以下ツール)に関するデモあるいは構想の発表が計5件行われた。招待講演では、国立情報学研究所の田原により、TopSEプロジェクトにおける要求工学教育の取り組みと成果について報告がなされた。ツールデモおよびツール構想の発表におけるトピックは次の通りであった: ステークホルダやドメイン知識を考慮した要求分析、ドメインオントロジを用いた要求獲得、要求工学におけるパターン、アーキテクチャ設計を導く非機能要求記述。

個別発表後の全体討論では、要求工学におけるツールの役割について議論が行われた。現状のツールにおける支援の実現手段を列挙し、将来のツールに求められる特徴が議論された。議論の結果、要求工学ツールに求められる特徴として次の点が指摘された: 機械的作業の自動化、スケーラビリティの克服、教育との連携、ツール教育コストと成果とのトレードオフ、方法論の適用促進、フェーズ(工程)間連携の促進、ツール適用の効果の明確化。フェーズ間連携については、企業におけるツール導入において重要な考慮事項であるとの指摘があった。

5 チュートリアル

筆者らが参加した1件について詳しく報告する。

(1) ソフトウェアの構造、振る舞いおよび進化の可視化 (Visualizing the Structure, Behavior and Evolution of Software, Stephan Diehl)

ソフトウェアにおける構造、振る舞い、進化の可視化に関する内容であった。可視化の定義から始まり、アルゴリズムアニメーション(アルゴリズムの振る舞いの可視化)、ビジュアルデバッグ、ソフトウェアアーキテクチャの可視化、ソフトウェア進化とメトリクスなどについて、多数の可視化ツールが紹介された。たとえば、オブジェクト指向システムの進化の可視化手法として、LanzaのEvolution Matrixが解説された。最後に、可視化パイプライン(データ獲得、分析、可視化)の観点からまとめが行われた。本分野における概観を知れるだけでなく、面白さも感じさせる良いチュートリアルであった。

近年では、SOFTVISやVISSOFTといった、ソフトウェア可視化に関するシンポジウム/ワークショップが定期的に開催されているようであり、本分野への期待が伺える。なお、Diehlによる『Software Visualization』がSpringerから出版予定となっている。

6 所感

鷺崎: 運営側の立場として、特に、プログラム編成およびワークショップ選定/開催に従事した。今回の開催を契機に、国内において産業界を含めソフトウェア工学自動化研究/開発が活発となり、より多くの実用化を意識した実装を伴う研究成果が国内外に向けて発信されることを期待したい。ソフトウェア開発の効率化/高品質化を目指すうえで特定処理の自動化は自然かつ根源的な取り組みであり、人を含めたシステム系全体の中でソフトウェアの比重が増大していく中、ASEの取り組みは今後重要性を増していくと予想される。発信に向けた自身の課題として、成果の完成度を高める最後の詰めや実ソフトウェアによる緻密な有効性実証の重要性を痛感した。

久保: 全体としてモデル検査に関する発表が多く、同分野の活気が強く感じられた。私個人としてはソフトウェアリポジトリなど既存資産の分析に興味があり、Mining Software Repositories セッションの各発表や、Analysis Iセッションのデザインパターン抽出に関する発表が印象に残った。

下滝: 非常に爽りある体験だった。日本の同年代の研究者の方々との交流が出来たのが良かった。しかし、一方で、著名な研究者、特に自分が注目している研究者に質問等ができる機会があったのにも関わらず、その機会を十分にいかせなかったのが非常に悔やまれた。今回参加してみて、自分の姿勢を見直す必要性を強く感じただけでなく、今後自分に何が必要なのかを実感できた。積極的にコミュニケーションを取り、研究者ネットワークを構築していけるよう努力したい。

中川: 今回のASEでは初めて運営者側の立場で参加することとなったが、国際会議のメカニズムと会議開催の裏にある多大な労力を知ることができ、研究とその発表ができる機会が持てることの有り難さを身をもって知ることができた。

本会議では各研究発表を聴講する時間を十分に取れなかったが、幾つかのセッションに接して、トップレベルの国際会議における論文採録には、自身の研究と実世界の問題とを関連付けて重要性を示すことが特に必要であると感じた。この点においては小生の研究にも改善を要する箇所が多く、今後の研究に対する良い刺激となった。

最後に、ASE, ICSEなどで感じるのは当分野における日本からの論文発表が少ない点である。最初から本会議での発表は難しいかも知れないが、併設ワークショップ等を利用してまずは本会議に参加し、国際的に活躍している研究者や最新の研究トピック

に肌で接することがモチベーションを向上させ自身の研究を洗練化させる最も良い方法であろう。その意味でも本会議は小生にとって貴重な経験となった。林: ASEには初めての参加であったが、大変有意義な時間を過ごした。特に、質疑応答やワークショップでの議論など、普段論文集を読むだけでは得られない内容を聴き、さらにソーシャルイベント等も含め、他の研究者達と会話しネットワークを広げることができたことは大きな収穫であった。

今回はワークショップでの発表を行い、議論を楽しむことができたが、本会議での巧みな発表に奮起されました。特に、日本での開催にも係わらず、発表は国外勢に圧倒されていたことが印象に強い。今後の日本からの更なる研究発表を期待するとともに、それに貢献できるよう努力したい。

丸山: ASEを含め多くの国際会議で存在感を示すことが重要であることを改めて感じた。

本位田: ASEに関して、特に2点を強調したい。海外のアクティブな研究者は、投稿先の国際会議を明確に絞っており、その論文投稿締切日に向けた研究遂行スケジュールが確立している。ソフトウェア工学の分野では、ICSE, FSE/ESEC, REとASEがそのターゲットとなっている。要求工学をテーマとするREとソフトウェアツールをテーマとするASEのコミュニティが被っているところも注目すべきところではある。日本からの採択件数を増やすためには、まず、各自が、きちんと投稿先を明確にした上で、長期的な視野での研究活動を実施しないといけない。2点目は、30代の人が、ASEの中心的な役割を果たしている事実である。ちなみに今年のプログラム委員長の1人は、数年前に学位を取得した新進気鋭の若手である。私の立場としては意識して日本からも若手をこのコミュニティに送り込んでいかなければならない。

最後に、多くの海外からの参加者から非常に良く組織され、過去最高の会議であったとのコメントをいただいた。これはひとえに本国際会議にご賛同いただいた多くの企業、1年半かけて準備に忙殺していただいたカンファレンス委員会のみなさん、学生ボランティアの諸君のご尽力の賜物であると確信しています。この場を借りてお礼申し上げます。

7 おわりに

冒頭で述べたように、ASE2006は多数の参加者を集めて成功であったといえる。各種併設イベントの実施も功を奏し、次代の自動化アプローチに向けて国内外の実務家/研究者間で活発に議論と交流が行われたと考えられる。また、会議で取り上げられ

た話題は、プログラム解析/モデル検査といった伝統的なものから、リファクタリング/アスペクトマインディングといった比較的最近のものまで多岐にわたり、ソフトウェア工学における自動化の広がりや深化をあらためて認識することができた。

今後、2007年は11月にアメリカ・アトランタで開催される予定である。ASE2007について、本稿の研究集会における発表時点では、研究論文の投稿が間に合う。日本からの投稿と発表、参加が増えることを期待したい。

謝辞 ASE2006の開催に協力/たずさわられた方々、参加された方々に感謝いたします。また、ASE2006実行委員会/運営事務局がWeb掲載用に提供されている画像を、本稿で利用させていただきました。

参考文献

- [1] 21st IEEE/ACM International Conference on Software Engineering, <http://www.ase-conference.jp/>
- [2] 28th International Conference on Software Engineering, <http://www.isr.uci.edu/icse-06/>
- [3] 14th ACM SIGSOFT Symposium on the Foundations of Software Engineering, <http://www.cs.uoregon.edu/fse-14/>
- [4] International Requirements Engineering Conference (RE), <http://www.requirements-engineering.org/>
- [5] International Conference on Software Maintenance (ICSM), <http://conferences.computer.org/icsm/>
- [6] International Symposium on Software Testing and Analysis (ISSTA), <http://www.cis.udel.edu/issta06/>
- [7] Aspect-Oriented Software Development Community & Conference :: AOSD, <http://aosd.net/>
- [8] European Software Engineering Conference (ESEC), <http://www.infosys.tuwien.ac.at/esec/>
- [9] Asia-Pacific Software Engineering Conference (APSEC), <http://www.cse.iitk.ac.in/apsec06/>
- [10] Eric J. Braude: Software Engineering: An Object-Oriented Perspective, Wiley, 2001.
- [11] CafeOBJ official homepage, <http://www.1dl.jaist.ac.jp/cafeobj/>
- [12] DARPA Grand Challenge, <http://www.darpa.mil/grandchallenge/>
- [13] ET ロボコン, <http://www.etrobo.jp/>
- [14] 二上貴夫 編: MDD ロボットチャレンジ 2005 産学連携によるモデルベース組込み開発の実践, 情報処理学会, 2006.
- [15] SLAM Project, <http://research.microsoft.com/slam/>