

ウィンターワークショップ2013・イン・那須報告

野田夏子^{†1} 岡野浩三^{†2} 早水公二^{†3} 戸田航史^{†4} 上野秀剛^{†5} 石尾隆^{†2}
林晋平^{†6} 妻木俊彦^{†7} 中村匡秀^{†8} 岸知二^{†9} 本橋正成^{†10} 鷲崎弘宜^{†9}

情報処理学会ソフトウェア工学研究会では、毎年1回参加者同士の議論を中心とした合宿形式のワークショップを開催している。2012年度は2013年1月に那須においてワークショップを開催し、例年同様活発な議論が行われた。本稿では、各テーマのセッションでの議論内容を中心に、本ワークショップについて報告する。

Report of Winter Workshop 2013 in Nasu (WWS2013)

NATSUKO NODA^{†1} KOZO OKANO^{†2} KOUJI HAYAMIZU^{†3}
KOJI TODA^{†4} HIDETAKE UWANO^{†5} TAKASHI ISHIO^{†2}
SHINPEI HAYASHI^{†6} TOSHIHIKO TSUMAKI^{†7}
MASAHIDE NAKAMURA^{†8} TOMOJI KISHI^{†9}
MASANARI MOTOHASHI^{†10} HIRONORI WASHIZAKI^{†9}

IPSJ Special Interest Group of Software Engineering (SIGSE) holds a workshop focusing on deep discussion among participants once a year. In the fiscal year of 2012, we had a workshop in Nasu, Tochigi prefecture, January 2013. We had a deep discussion about recent issues and future direction of software engineering. In this paper, we report each discussion held in each special theme session and the whole workshop.

1. はじめに

情報処理学会ソフトウェア工学研究会では、1997年より毎年冬に、参加者同士の議論を中心としたウィンターワークショップを開催している。ワークショップでは、その時々に応じて、ソフトウェア工学分野において注目を集めているテーマあるいは注力すべきテーマを取り上げ、質の高い集中的な議論が展開される。今回は、栃木県那須郡那須町での開催となり、全部で46件の論文投稿があり、参加者は約60名に上った。

今回のワークショップでは、研究会会員を中心とするソフトウェア工学の研究コミュニティからセッションテーマを募集し、7つのテーマに関するセッションを設定した。参加希望者はこれらのテーマから1つを選んでポジションペーパーを投稿し、採録された論文の著者がそれぞれのセッションに参加した。各セッションでは、それぞれのテーマの討論リーダーのもと、いま何をやるべきか、そのために解決すべき研究・技術課題は何かなどについて活発な議論が行われた。

本稿では、このワークショップについて、テーマ毎のセ

ッションでの議論内容を中心に報告する。2章から8章までは、それぞれのセッションについての報告であり、各セッションの討論リーダーが執筆を担当している。各章のタイトルが、本ワークショップにおけるテーマ名となっている。9章では簡単にワークショップ全体を振り返って報告し、10章でまとめる。

2. 形式手法 — 産官学の連携の強化を目指して—

本セッションは、岡野と早水が討論リーダーを務めた。

2.1 はじめに

モデル検査・モデル駆動型開発を始めとする形式手法は産業界でも普及の時期を迎えている。一方で、学術的にも、応用的にも、実際の開発過程への適用方法など課題も多い。とりわけ、大学・企業両サイドでの形式手法技術教育の方法や技術支援の相談窓口について関心が高まりつつある。産官学の理想的な連携のあり方の模索も必要である。そこで本セッションでは、形式手法に関連する学術分野、適用現場双方での取り組み、経験、教育実践などを採りあげ、形式手法に関する産官学多くの場での関心を持つ者の情報共有の場とすべく議論を行った。

2.2 採りあげたテーマ

今回のワークショップでは5件の論文を採録した。これらの論文のタイトルと著者一覧を以下にまとめる。形式手法に関する様々なトピックスがバランス良く採りあげられた。

†1 芝浦工業大学

†2 大阪大学

†3 フォーマルテック/産総研

†4 福岡工業大学

†5 奈良工業高等専門学校

†6 東京工業大学

†7 国立情報学研究所

†8 神戸大学

†9 早稲田大学

†10 合同会社カルチャーワークス/東京工業大学

- 段階的形式化手法による要求仕様の品質向上
 - 弓倉陽介 鷲見毅 藤本宏 和田大輝 村田由香里
- モデル検査による高レジリエンス・システムの検証
 - 早水公二 山形頼之 林伸行 大崎人士 辰野功 兼松順三 白井大
- 在庫管理プログラムに対する Alloy Analyzer を用いた検証事例
 - 森恵弥佳 岡野浩三 楠本真二
- プロセスの模倣性判定に基づく UML 設計間の整合性検証
 - 横川智教 宮崎仁 佐藤洋一郎 有本和民
- プロセスを考慮した形式手法の導入
 - 日下部茂,大森洋一,荒木啓二郎

2.3 おわりに

上記テーマでの発表ならびに、それらの発表に対する議論を行った。個々の論文の発表の後では全員で議論を行った。その中でメーリングリスト立ち上げ、産学官のプロジェクトに関する議論などがなされた。

3. 開発マネジメントにおける産学の問題共有と連携強化

本セッションは、戸田と上野が討論リーダーを務めた。

3.1 はじめに

ソフトウェアの利用環境の拡大に伴い、開発されるソフトウェアの数も増加傾向にある。しかしながら、ソフトウェア開発プロジェクトの40%ではなんらかの問題が発生しているとの報告があり、プロジェクトの失敗の回避がソフトウェア開発における大きな課題となっている。これらの問題解決のために多数の研究成果が報告されてきた。

しかしながら、このような学術界の研究成果が産業界の実務の場において十分に活用されているとは言いがたい。その原因としては、アカデミックなコミュニティでの議論の前提が実務を反映していない、研究の方針がそもそも現場のニーズを適切に反映できていないといったものが考えられ、このような産学間での齟齬についてはこれまでも多くの指摘がなされてきた。

これまでウィンターワークショップでは研究者、実務者の双方が参加し、開発マネジメントに関する話題について様々な観点から議論を行ってきた。今回もこれを踏襲するが、それに加え「産学の問題共有と連携強化」について議論も行った。

3.2 議論のテーマ

今回のワークショップでは12件の論文が採録となった。これらの論文が扱っているトピックを紹介する。

3.2.1 工数見積もり技法

ソフトウェア開発工数の見積もりは、開発リソースの決定において非常に重要な位置を占めることから、古くは

Boehm のCOCOMO が、最近では新たな数学的モデルの提案や、複数の数学的モデルを大量のデータに対して適用することによる統一的な精度評価の試みや、データの処理方法や見積もり結果の補正等の見積もりモデルの前処理、後処理についての研究が進められている。

ワークショップでは上で述べた開発工数見積もりモデルを現場に適用する際の問題点を中心に議論を行った。

3.2.2 ソフトウェアテスト・レビュー

ソフトウェアテスト・レビューは開発するソフトウェアの規模に関わらず必要であり、かつソフトウェアの品質はテストやレビューに大きく依存する。このためソフトウェアテスト・レビューは、工数見積もりと同様に古くから数多くの研究がなされてきた。

ワークショップでは、ソフトウェアテストについてはテストの効率化や実施結果を定量的に管理するための指標についての議論を、レビューについては、リソース割り当ての最適化に代表される効率化のための施策についての議論を行った。

3.2.3 プロセス改善

ソフトウェア開発プロセスの改善やそれに伴う効率化は個々のプロジェクトだけではなくソフトウェア開発組織にとって重要な要因の一つである。ソフトウェア開発プロセスの改善の施策は大きく2つに分けることができ、1つはプロジェクト、組織といった大規模な要因を対象とするもの、もう1つは開発者のような小規模な要因を対象とするものである。前者の例としてはアジャイル型開発やオフショア型開発の導入、後者の例としてはPSP(Personal Software Process)が挙げられる。近年ではより実務に即した形の開発手法の提案や改善のためのデータ収集を行うツールや収集する指標の提案がなされている。

ワークショップではプロセス計測や評価、改善に関する取り組みや、計測に用いる指標について議論を行った。

3.2.4 データマイニング

Subversion や git 等の版管理システムや Trac, Red-mine 等のバグ管理システムに代表されるプロジェクト管理ツールの普及を背景として、開発リポジトリデータを対象としたデータマイニング技術が注目されてきている。特にこれらのツールが利用されているOSS(Open Source Software)プロジェクトからは開発データの取得が容易であることから、OSSを対象とした研究が盛んになってきている。しかしながら、OSSのデータから得られた知見をそのまま企業の現場で用いることができるとは限らないため、OSSと企業の開発データの双方からのアプローチが重要視されつつある。

ワークショップではデータマイニングを実行するに当たって直面する問題やその解決策、またOSSから得られた知見の開発現場での適用について議論を行った。

3.3 おわりに

セッションは実務で直面する問題の側からの発表と研究で用いる技法の側からの発表に大別され、各発表について異なる側からの多くの意見が出された。そして発表とそれに伴う意見交換の結果、産学間で問題意識は共有されていることは確認できた。また全体の議論では、技法の適用時に用いるデータについて、実務を踏まえた粒度の設定が必要である、メトリクスにはその意味を含めた取り扱いが必要である、等の意見が出された。本セッションでの議論が研究の進展や現場での問題解決のきっかけ、さらには産学での問題のさらなる共有と連携強化のきっかけとなれば幸いである。

4. ソースコードと機能の対応関係を特定する技術

本セッションは、石尾と林が討論リーダーを務めた。

ソースコードと機能の対応関係の特定は、ソフトウェアの保守やリエンジニアリング活動の重要な要素であり、国外はもとより国内でも様々な研究が行われている。これらの研究は、Concept Location や Feature Location, Impact Analysis, Concern Location, Traceability Recovery など、様々な技術に関連づけられて発表されている。その一方で、ソフトウェアの「機能」のような曖昧なものを扱わなくてはならないのに、様々な用語や手法の分類について研究者間で「一貫した」認識がなく、議論に支障が出る場合があった。本セッションは、これらの分野に関わる研究者の間で、技術の現状や課題に対する共通の理解を作ることを主目的として開催した。本セッションの具体的な議題は次の通りである。

- 「機能」とはどう定義されているのか。
- ソースコードと機能の対応関係として、何を抽出したら正解になるのか。
- 抽出したものはどう使用されるのか。
- 「真の正解」はどのように検証できるのか。
- どのような実験を行うことが適切なのか。
- 開発現場で実際に適用されたという話に乏しいが、現場では受け入れられないのか。
- 開発者は手法の「有用性」は納得できるか。
- 手法およびツールへの「信頼」はどこから来るのか。

上記テーマでポジションペーパーを募集したところ、以下の全7件（うち大学3件、企業4件）のポジションペーパーが投稿され、9名の参加者による議論が行われた。

- 城間祐輝, 風戸広史 (NTT) : 用途に応じたトレーサビリティ復元とその有効性の評価に向けて
- 岡田譲二 (NTT データ), 鹿島悠 (阪大), 坂田祐司 (NTT データ) : Feature Location 技術の開発現場での活用に向けて
- 風戸広史, 大島剛志 (NTT) : Feature Location を活用

したソフトウェア機能の文書化に向けて

- 久米出 (奈良先端大), 中村匡秀 (神大), 新田直也 (甲南大), 柴山悦哉 (東大) : 動的解析によるフレームワーク学習に向けて
- 林晋平 (東工大) : Feature Location ベンチマークの現状と課題
- 長谷川勇 (zigsow) : システムの分類による impact analysis 導入の課題整理
- 土屋良介, 鷺崎弘宜, 深澤良彰 (早大) : 派生プロダクト群における要求・実装間のトレーサビリティリンク抽出

セッションでは、各参加者から、現在取り組んでいる技術と、そのとらえ方について多数の意見が表明され、活発な議論を行うことができた。以下にその結果をまとめる。

まず、これまで訳語が存在していなかった Feature Location 技術について、参加者が持ち寄った候補に基づき訳語を定めた。その結果、Feature Location に対しては「(ソフトウェア) 機能検索」、同様に Concept Location に対して「概念検索」、Concern Location に対して「関心事検索」の訳語を充てた。以降、国内での研究発表においては、上記の訳語を積極的に利用していただくことを期待する。

また、機能検索の有用性を高めるためには、機能検索ツールの要件として、以下の2点が重要であることが挙げられた。

- 因果関係が明確であること。機能検索手法の利用者は、得られたモジュールの候補が「なぜ候補として挙げられたか」「何が候補に入らないか」が理解できなければ、得られた候補に漏れがある可能性への不安から逃れられない。その結果、他のモジュールを確認するための工数を削減することにつながる。因果関係が明確なツールの例として、grep が挙げられた。
- 結果の確認・共有が可能なこと。大規模なプログラムから機能検索を行う場合、検索結果のモジュール一覧も膨大なものとなる。それを人手で精査するには長時間を要するため、その工程も含めて支援されることが望ましいという意見が出た。例として、古典的な手法ではあるが、検索結果を Excel 等のスプレッドシートにまとめる方法が挙げられた。

さらに、技術の評価・ベンチマークの作成についても議論を行った。その結果、用途によって正解が異なること、開発現場で観測できるソフトウェアへの変更は必ずしも機能と対応しないこと、客観的な正解を適切に作る方式があればそれ自体が新しい機能検索の手法となってしまう可能性があることなど、非常に悩ましい問題であり、継続的な議論が必要であることがわかった。今後行われる研究発表においても、完全な評価は困難であろうが、何をもって正解とし、どのような基準で既存研究と比較していくか、分

野全体での経験の蓄積が進むことが望まれる。

セッションとしては、大学および企業からの参加者のバランスが良く、目指したい目標、経験に基づく現状分析について、幅広い議論を行うことができた。従来、ウィンターワークショップでは「プログラム解析」というセッション名で幅広い解析技術を取り扱ってきたが、本年はあえて技術の種類を絞ることで、踏み込んだ議論を行った。今後もプログラム解析分野に関しては、目的別、技術別など、研究者の問題意識の変化に合わせて、多様なセッションが開催されれば良いと考えている。

5. 要求の責務

本セッションは、妻木が討論リーダーを務めた。

要求工学グループのセッションでは、異なった環境や背景を持った参加者の議論を、議論のための議論に終わらせないために、取り上げたテーマに関する手法を構築するという最終目標を掲げて議論を行うことにした。

議論のテーマについては、個々の参加者から希望を出してもらい、その中から2つのテーマを選択し、2つの検討チームを作った。どちらのチームに参加するかについては、参加者自身の意思にまかせた。

最初のチームが取り上げたテーマは、市販製品に対する一般利用者の評価を次期製品に反映するための仕組みに関するもので、評価情報の収集方法と次期製品への反映方法が大きな課題となる。後者の課題については製品開発会社の体制に係わるものであることから、評価情報の収集のための仕組み作りに関することが専ら議論の中心となった。利用者に負担を掛けない仕組みということで、本チームからは、FBなどで採用されている評価ボタンを成否に組み込み、インターネットを通して該情報を収集する方法が提案された。

2つ目の検討チームでは、セキュリティ問題について検討が行われた。短時間で手法の考案にまで持っていくには問題をさらに絞り込む必要があるため、本チームでは入力情報の内容から危険性を判断するための方法を検討することにした。その結果、フィーチャモデルとゴールモデルを使って入力情報の危険性を予測するための手法を考案することができた。すなわち、ゴールモデルを使って入力情報の隠された意図とフィーチャモデルによって明らかにされたシステムの機能の組み合わせから危険の可能性を予測するものである。

当初、手法の構築という目標はかなり挑戦的なアプローチかと思われたが、それぞれのテーマの中で参加者たちが抱えている課題がある程度明らかになるとともに、課題の解決に対するより建設的な議論が可能であったし、技術的にもそれなりの手法を考案することができたと感じている。

6. サービス・クラウドがもたらすシステム開発パラダイムの変化

本セッションは、中村が討論リーダーを務めた。

6.1 テーマ設定の経緯

様々な計算資源や情報がネットワークのどこかで管理され、ユーザは必要に応じてそれらを「サービス」として利用するクラウドコンピューティングの考え方は、これまでのITシステムに大きなパラダイムシフトをもたらした。クラウドがもたらす仮想マシン基盤や、並列分散処理、ビッグデータ活用、ソフトウェアサービス等は、実現可能なシステムの品質やバリエーションを大きく広げるものである。

こうしたクラウド・サービスは、プロダクトとしてのシステムの実現範囲を広げるだけでなく、システム開発(エンジニアリング)のあり方さえ、大きく変化させるインパクトを持っている。その一方で、クラウドを活用したシステムを、「いかに上手くつくるか」は体系的な議論が無く、それぞれの開発組織が独自に模索している状態である。ソフトウェア工学分野においても、クラウド・サービスを活用した最新のシステム開発には追従できておらず、工学的な開発手法の体系化は、これからの大きな課題である。

WWS2013のサービス指向セッションでは、「サービスやクラウドの導入によって生じるシステム開発の変化」に焦点をあて、工学的な議論を行うべく、論文募集を行った。

ディスカッションにおいては、システム開発における様々な局面において、クラウド・サービス技術が、何をどのように変えうるのか、様々な視点から議論することを狙った。

6.2 採択論文

- マス・カスタマイズ開発に向けたモデルチェーンの構築と利用：細野 繁，下村芳樹(NEC)
Webサービスの要件定義から運用までの各工程のモデルを束ね、モデルチェーンとして資産化する方法と、これらを利用したカスタマイズ開発方法を述べている。
- クラウドソフトウェアの継続的デリバリについて：横山重俊，吉岡信和(NII)
アジャイル開発のプラクティスである継続的インテグレーションやデリバリを、クラウドアプリケーションに対して適用する支援ツール `dodai` を開発している。
- 連続的サービスサプライネットワークアーキテクチャ：青山幹雄(南山大)
リアルタイムなサービス連携を含むソフトウェアビジネスの多層階層構造をモデル化するアーキテクチャ「継続的サービスサプライネットワーク(CSSN)」を提案している。
- サービス/クラウドにおける価格付け：田仲正弘，村上陽平(NICT)

サービスやクラウドの値段に関する議論、特に、資源に対する価格付け、ユーザ要件に基づく拡張への投資、複数事業者による連携サービスの利益配分について述べている。

- 大学間連携によるクラウドサービスの共同開発・共同運用に向けた挑戦：梶田将司(京大)

アカデミッククラウドサービスを、複数大学が共同で開発・運用する際の課題を整理している。アプローチとして、共通仕様・実装への対応、多様性への対応を挙げている。

6.3 ディスカッション

セッション初日は、各自自己紹介と近況報告を行い、セッションペーパーの発表を行った。発表は持ち時間 30 分で、比較的余裕を持った質疑応答や議論が行えた。

セッション2日目は、初日の発表内容を踏まえて、全体ディスカッションを行った。まず、テーマである「サービス・クラウドがもたらすシステム開発パラダイムの変化」について、議論を行った。サービスやクラウドが開発局面もたらす変化の中で最も顕著なのは「**運用を考えたシステム開発**」が重要になってくるということである。このことは昨今話題の DevOps というキーワードにも見られる。

サービス・クラウド技術は、オンデマンド・セルフサービスや柔軟な伸張性、動的バインドやマルチテナンシといった、多様なシステム運用オプションを提供する。運用の選択肢が多様であるからこそ、開発するシステムがどのように運用されるべきかを、開発者がしっかり考えておく必要がある。また、状況に応じて最適な運用ポリシーが自動適用されるように、システムの中に自らの状態を外部から計測できる API を備えておくべきだという議論も行われた。

また、エンタープライズソフトウェアの階層アーキテクチャに「運用層」を追加するべきだという意見や、現在のクラウドのサービスモデル(IaaS, PaaS, SaaS)を時間軸方向に拡張し、それぞれのサービスモデルの運用モデルを考慮すべきという意見も出た。

これまでのソフトウェア工学は、開発をいかにうまくやるかを追求する研究がほとんどである。サービスやクラウドという新しい技術の登場によって、運用がより一層重要となってきた。これからのソフトウェア工学では、開発と運用をシームレスにつなぐライフサイクルをどう実現するかを研究すべきであり、そこに新たな研究シーズがあるのではないかと、という結論で締めくくられた。

セッションの最後では、国内のサービス・クラウドコミュニティの発展に向けて、どのように活動していくべきかも話し合われた。特に学生や若手研究者へのプレゼンスをあげるために、最新のサービス・クラウド技術を試し、学ぶ「場」を与えることが重要であるという意見が出た。次のウィンターワークショップでも引き続き議論したい。

7. ソフトウェア工学研究の共通問題

本セッションは、岸が討論リーダーを務めた。

本セッションは、ウィンターワークショップ 2010・イン・倉敷で行った「ソフトウェア工学研究の評価」[1]を一回目として、ウィンターワークショップやソフトウェア・エンジニアリング・シンポジウムなどで継続して行ってきたセッションである。このセッションは、そもそもソフトウェア工学研究は評価が難しく論文が採録されづらいという研究者の素朴な悩み(愚痴)から始まった。もちろんこうした議論はこのワークショップ参加者だけがしているものではなく、国内外のソフトウェア工学コミュニティにある程度共通したものであり、ソフトウェア工学研究評価に関する議論はいろいろな機会になされてきた。

本ワークショップでは、実務での実証的な評価ができる段階に到達する前の技術やアイデアを、どうやって評価するかという議論を中心に進めてきた。これはソフトウェア工学研究では、ともすれば実規模の問題に適用できるかどうか、つまりスケラブルな技術であることを強く求められたり、あるいは企業ではそれがどれだけ役に立ちビジネスに貢献するかを問われたりするが、この段階の技術やアイデアは、十分にスケールし実問題に適用できるまで成熟していない場合が多いからである。そしてこの段階の研究評価には、共通問題の適用が有用であろうとの議論へと発展してきた。

共通問題としては例えば国内では酒屋問題[2]が有名である。これはひとつの問題に様々な設計技法、あるいはプログラミングパラダイムを適用することで、その技法やパラダイムを比較し、それぞれの特性をくっきりとさせるために使われてきた。ソフトウェア工学の研究は特定の評価軸だけで評価することが難しいものも多く、そうした中で共通問題が活用されてきた。共通問題には大きく、以下のようなものがあり、いずれも、ソフトウェア工学研究の評価に活用することができると考えられる。

- ベンチマーク型(分析比較型): 複数の解法の存在する問題を共通問題として提示し、それらの解法の特性を比較するものである。例えば情報処理誌に掲載された酒屋問題などがそれに該当する。共通の問題を解くことにより、提案する解法がどういう特徴を持つかを、他の解法との比較の中で相対的に評価することができる。

- コンテスト型(結果比較型): ひとつの目的を達成するための解法を、その目的の達成度合いによって比較するものである。ベンチマーク型が特性の分析的な比較や特性間トレードオフの議論が重視されるのに対して、最終的な目的の達成度、達成への総合力、バランス力といった工学的観点を重視して評価する点に特徴がある。例えば MDD ロボットチャレンジ[3]などが該当する。

- グランドチャレンジ型(課題達成型あるいは裾野効果型): 解決されることにより、その分野にとって大きな進歩につながるような、一定の意義と難易度を持った問題を提示し、解を求めるものである。解法の比較ができるという

点では上記の二つの型と類似した側面も持つが、むしろ提示された課題を解くことそのものが大きなチャレンジであり、またそれを解くためには様々な副課題や関連課題の解決が必要となることで、研究の裾野が広がるという効果を持つものである。形式手法のグランドチャレンジ[4]などが該当する。

共通問題が有用であるためには、その共通問題が現在のソフトウェア工学研究に適したものでなければならない。酒屋問題ができて約30年が経過しているが、その間にソフトウェア工学をとりまく技術環境やビジネス環境は大きく変わっており、ソフトウェア工学が扱う内容そのものも拡大・変化している。前回のワークショップでは、現在のソフトウェア工学研究に即した新しい共通問題の必要であるので、そうした共通問題をコミュニティ内で募ってはどうかという議論を行った。その一環として、ワークショップでの議論を整理してコミュニティ内外の人にソフトウェア工学研究と共通問題について幅広く知っていただくことが有益であるとの議論に達した。

そこで今回のワークショップでは、その具体的な内容についてさらに検討を深めた。その結果、学会誌「情報処理」の特集号として、共通問題を取り上げてもらう方向で、その内容の議論を行った。その結果、以下のような内容について企画を進めることが決まった。

- これまでのワークショップ議論の説明。
- ソフトウェア工学研究の段階とそれに応じた研究評価の在り方についての議論の紹介。
- 現在までに提案され活用されてきた国内外の様々な共通問題の紹介
- 酒屋問題の再点検。30年たった現在の視点で、現在における共通問題への要件の提言。
- 酒屋問題以降、ソフトウェア工学研究の広がりの中で必要となってきた新しい研究領域での共通問題の必要性の提言。コンテキストウェアシステムやサイバーフィジカルシステムなどにおける共通問題など。
- 関連した話題として教育における共通問題の活用。

こうした経緯と議論を踏まえ、ワークショップ参加者を中心に特集号(2013年9月号を予定)の準備を行っている状況である。本セッションの成果として、ぜひ読んでいただきたい。

8. モデリング、アジャイルとパターン

本セッションは、本橋と鷺崎が討論リーダーを務めた。本セッションでは、最初に、以下の各ポジションペーパーの発表と質疑を通じて、パターンランゲージやプロジェクトランゲージに対する取り組み、さらにはパターンランゲージを一つのルーツとするアジャイル開発や、モデリン

グについても合わせて実践上の現状や展望を概観した。

- パターンの構造化に基づくパターン・ランゲージの拡充: 中野 聡之 (早稲田大学)
- パタンによる対立解消の試み: 本橋 正成 (合同会社カルチャーワークス/東京工業大学)
- パターンランゲージとデザイン思考: W プロセス 2.0: 羽生田 栄一 (株式会社豆蔵)
- アジャイルモデルの習得におけるワークショップの有効性の分析: 鈴木 翔大 (早稲田大学)
- Systematic Mapping を用いたアジャイルワークショップの分析: 鷺崎 弘宜 (早稲田大学)
- リバースモデリングを用いた組込みソフトウェア向けシステム設計メソッドロジー: 三上 徹也 (日本アイ・ビー・エム株式会社)

議論を通じて、パターンランゲージや関連する概念の定義がしばしば不明瞭であり成果共有や研究発展の妨げとなっていること、および、その一例として、属人性を低減させた形で既存パターンランゲージについてパターンを拡充する手法が未整備であることが問題として識別された。

そこで、パターンランゲージや関連する概念を、一意に解釈可能な形でメタモデルとしてモデル化した。図1にパターンランゲージおよび関連する概念間の関係を整理した結果を、UML クラス図により示す。

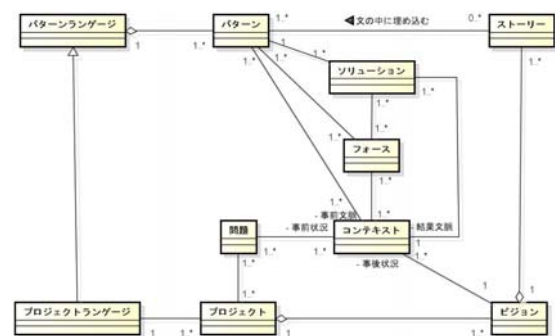


図1: パターンランゲージのメタモデル

図1における各要素の定義を以下に示す。

- 問題とは、ビジネスなど実社会における関係者にとって暗黙的及び明示的にかかわらず障害のこと。その問題は、過去、現在、そしてビジョンの中の未来という時間的な流れの中で変遷する。
- プロジェクトとは、その有している問題を解決することを目的とした計画及びその実体のこと。実際には、プロジェクトが次のプロジェクトを生み出しながら継続して行く意味では「コミュニティ」という言い方もできる。

- コンテキストとは、その問題を内包する文脈のことで、その文脈に基づくことによって、ある事象が問題として認識されることになる。
- フォースは、制約事項のこと。コンテキストとフォースが組合わさることで問題が立ち上がる。
- ソリューションとは、コンテキストにおけるフォースを消滅し、問題を解消させるもの。
- パターンとは、関連するコンテキスト、フォースとソリューションを内包するひとつの単位。多くの場合、単語として機能する。パターン間の自己参照はフォースで説明されている。なお、ソリューションが欠落したパターンを「パターンの種」と呼ぶ。
- パターンランゲージとは、パターンの集合と、それによって立ち現れる構造のこと。複数のパターンが集まっただけではランゲージとして成立せず、何世代かにわたり生成と評価を経て、ある世界観（ドメイン）を記述し得ると判断されたときにパターンランゲージと呼ばれる。パターンランゲージは、プロジェクトに比べて汎用的なものでプロジェクトに関係しないものも含まれる。
- プロジェクトランゲージとは、パターンランゲージを、ある特定のプロジェクトにフォーカスしたもの。プロジェクトに合わせてパターンの追加や編集もされ、パターンランゲージへと昇華される。パターンランゲージの構造が再定義される場合が多い。
- ビジョンとは、そのプロジェクトで実現したい状況を「ありあり」「いきいき」と記述したストーリーの集合。預言者の描くリアルなイメージが本来の意味。
- ストーリーは、パターンを単語として、用い、その関連性を文章としてビジョンを記述したもの。ストーリーは、その利用者の体験として語られることも多い。

また、図1における要素間の関係を以下に示す。

- ある組織やコミュニティが「問題」（通常、複数）をもっており、それを改善・解決した場合にそれに関わるステークホルダーを構成員とする「プロジェクト」を立ち上げる。プロジェクトでは、現場をフィールドワークして問題の置かれた「コンテキスト」（通常、複数）を理解し、そのコンテキストのもつフォースを探りながら、問題そのものや解決に直接・間接に関連した「パターン」を見出していく。通常、この部分では KJ 法に代表される質的研究手法が適用される。
- その際、既存の「パターンランゲージ」からそ

のプロジェクトに適用可能なパターンを識別するが、適切なパターンがない場合、その現場ドメインのために新たなパターンをそのプロジェクト固有のパターンとして創出することもできる。それらは合わせて既存のパターンランゲージを拡張した「プロジェクトランゲージ」を定義したものと考えられる。

- 「パターン」とは、繰り返し現れるコンテキストの中で「問題」「フォース」「ソリューション」をセットにして定義したもので、問題を含む事前文脈としての「コンテキスト」をそのパターンを適用することで事後文脈としての「コンテキスト」に変換する働きをもつ。したがって、そのプロジェクトで対象とする状況全体をカバーするように関連するパターン群を識別しておくことは、そのプロジェクトにとって問題解決を促すだけでなく、そもそも問題が解決した後の全体状況に対する各ステークホルダーの夢や希望を想起させるものでもある。

- そこで、そのプロジェクトに対するパターン群が「プロジェクトランゲージ」用の語彙として識別された次のステップでは、そもそも各ステークホルダーがそのプロジェクト（や後続するプロジェクト）の中で何がしたいのか・どうなっていて欲しいのかを「ビジョン」として、そのプロジェクトランゲージの語彙を使って、「ストーリー」つまり物語の形で描きだす・語り下ろすことである。プロジェクトは複数のビジョンをもつことができるし、個々のビジョンも1つ以上複数のストーリーの集合として構成される。

- こうして書き下ろされたストーリーは、そのプロジェクトの目指す目標となるので、その重要性・実現可能性と制約・思い入れ等をもとに優先順位を付け、アジャイルなスタイルで順次、実現・改変・消滅させていくことになる。

- なお、こうしたプロセスの中で効果の高かったプロジェクトランゲージ内の新規パターンは、後続プロジェクトや他プロジェクトでも利用されるようになり、結果として、特定プロジェクトを超えて、広く汎用的なパターンランゲージの語彙として認知されていく。その意味で、プロジェクトランゲージとパターンランゲージとは相対的な違いであり、特定のプロジェクトでのパターン適用活動に照準を当てた場合の説明上の区別ということができる。

さらにメタモデルを得たうえで、既知のパターンを構造化（構造に関するモデル化）し、構造を変化させることにより、要求に対して解決策の異なるパターンを導出する手法を考案した。セッションにおいては、考案したメタモデルおよびパターン導出手法を含めた論文ドラフトの執筆を完了し、メタモデルを参照する形で各参加者が以下の論文

を論文誌や国際会議へ投稿した。

- 中野 聡之, 角谷 将司, 鈴木 翔太, 鷺崎 弘宜, 深澤 良彰, 羽生田 栄一, 本橋 正成, 三上 徹也, “パターンの構造化を利用したパターンランゲージの拡充” (投稿中)
- Masanari Motohashi, “Pattern Language to Project Language” (under review)

今後の課題として, 提案するモデルをメタモデルとして具体的にパターンランゲージを記述および拡張することを支援するツール環境の整備, ならびに, 実プロジェクトへの適用を通じた提案モデル・拡張手法の妥当性および有効性の実証が挙げられる。

9. ワークショップ全体を振り返って

ワークショップは例年通り 1 泊 2 日で行われ, 1 日目の昼過ぎに開始され, 2 日目の昼過ぎに終了した。プログラムも例年通りで, まず全体セッションで参加者全員が顔を合わせた後, 各テーマに分かれてテーマ別のセッションを行った。夕食後にはナイトセッションを開催し, テーマにこだわらず参加者同士が自由に集まり様々な話題について肩のこらない議論を行った。昼間のテーマ別セッションに比べて, ナイトセッションはカジュアルな議論の場であるが, 様々な情報を交換し, また交流を広げる良いチャンスになったようである。テーマ別セッションに加え, ナイトセッションでの参加者同士の交流を楽しみにしている参加者も多いようで, このような交流の場の提供も研究会の重要な役割かもしれない。2 日目は朝から昼前まで各テーマセッションでの議論を続け, その後全体セッションで各セッションの議論内容を報告した後, 閉会した。



図 2 全体セッションの様子

前回のウィンターワークショップに比べると, 投稿数,

参加者数が若干少なく, 正直なところ, 開催当日までは少々寂しいワークショップになるのではとの危惧もあった。しかし, ふたを開けてみれば, 今回も例年通り各テーマともに非常に議論が盛り上がり, 全体としても活気のあるワークショップとなった。各テーマともに進行方法を討論リーダが工夫し, 単にポジションペーパーを発表するだけに留まらない深い議論が行われた。また, 議論から生まれたアイデアを論文化したり, 学会誌特集号の企画としてまとめたりなど, 具体的ないくつかのアウトプットが生まれたことも今回の特徴であった。

セッションテーマは, より議論のポイントを絞ったものになっているものが例年より多かった。このようなテーマにすることにより, より深い議論を行うことができたと思われる。一方で, テーマを絞ることにより, ワorkshopそのものには参加したいが議論したいテーマがないために参加を見送ったということもあるかもしれない。今年の参加者数の減少の一因として, この辺りの事情も考えられる。今後のワークショップにおいては, ソフトウェア工学に関係する人が広く興味を持ち議論できるテーマと, ある程度技術ポイントを絞り込んだテーマとを, バランスよく設定するなどの考慮も必要かもしれない。

10. おわりに

本稿では, 本年 1 月に開催されたウィンターワークショップ 2013・イン・那須について報告した。

本ワークショップは, 多くの参加者のご協力により大変有意義なものになった。ウィンターワークショップは, 参加者との議論により自身の研究やソフトウェア工学実践に役立つ情報が得られるだけでなく, 様々な参加者と出会い交流を広げる良い機会でもある。次回以降もぜひ多くの方にご参加いただきたい。

参考文献

- 1) ウィンターワークショップ・イン・倉敷 2010 論文集, IPJS, Symposium Series, Vol.2010, No.3.
- 2) 山崎利治: 共通問題によるプログラム設計技法解説, 情報処理, Vol.25, No.9, p934, Sep.1984.
- 3) <http://www.ertl.jp/ESS/2010/index.php>, 2010 年の開催案内.
- 4) Tony Hoare: The Verifying Compiler: A Grand Challenge for Computing Research, In Journal of the ACM, Vol. 50, No. 1, pp.63—69, 2003.