

「ソフトウェアパターン」セッションテーマ紹介

鷲 崎 弘 宜†

本稿では、ソフトウェアパターンの発展の経緯と現状を概観し、今後のパターンへの取り組みの展望を得るために「ソフトウェアパターン」セッションがどのような議論を行うのかを説明する。

Topics of the Session on Software Patterns

HIRONORI WASHIZAKI†

This paper outlines the progress of software patterns and explains what discussions will take place in the session on "Software Patterns" to get insight into the directions of future software patterns efforts.

1. ソフトウェア開発とパターンランゲージ

ソフトウェアパターンとは、ソフトウェア開発における特定の文脈上で、繰り返し発生する問題と、問題に対する実証済みの解法として得られるソフトウェア構造（プロダクトや開発組織 / プロセスなどの構造）、および、解法を導いたフォースなどを包括した知識を、文書などの形式であらわしたものである。Christopher Alexander の建築学におけるパターンランゲージ¹⁾に触発されて、特にオブジェクト指向に基づく分析・設計に関するソフトウェア業界の知識が底上げされた^{2),3)}。また、ノウハウをパターン形式で記述することも一般的になりつつあるため、ソフトウェアに関する知識を書き表し、共有するためのソフトウェアパターンという方法は一定の効果を上げたと考えられる。しかしながら、ソフトウェア開発のための包括的なパターンランゲージを志向した成果は、今にいたるまで事実上一つも得られていない⁴⁾。

2. 問題意識と討論テーマ

建築におけるパターンランゲージの利用による主な効果には、顧客参加型のパターンランゲージ構築による要求獲得支援と、既存 / 発見中のパターン集合を用いた対象構造の生成的な設計支援の2つがある。両者は独立した効果ではなく、密接に相互依存していると考えられる。

要求獲得支援については、我々はウィンターワークショップ・イン・石垣島におけるパターンランゲージ体験と議論を通じて、その効果がソフトウェア開発についても同様に得られる可能性があることを確認した⁴⁾。ただし、ワークショップにおける体験はパターンランゲージの構築に至らなかったため、従来のソフトウェア開発プロセスの中でパターンランゲージが果たす役割について明確な回答を得ていない。生成的な設計支援については、上述の歴史的背景に基づくソフトウェアパターンの普及という事実から、ソフトウェア開発において同様の効果が得られる可能性があることを幾らか理解できる。しかしながら、ソフトウェア開発における真に有効なパターンランゲージがどのようなものであり、その有効性がパターンランゲージ / パターンのどのような特性 / 仕組みから得られるものであるのかについて依然として不明である。この事実は、ソフトウェアパターンの有効性に関する Ward Cunningham の以下の指摘に現れている。

“ 誰か経験のある人を雇いたいと言わなければならないとき、実際、経験を積むことによって彼らが何かを手に入れたと、我々は考えるのだろう。パターンでもって手に入れようとしているのはそれだ。つまり、理論的にどうかということは分からないが、実際に上手くいくというものなんだ。”⁵⁾

より具体的には、ソフトウェア開発という文脈において、以下の事柄が明らかにされていないため、工学的 / 科学的見地からパターンランゲージがソフトウェア開発にもたらす意義を我々は十分に把握できてい

† 国立情報学研究所
National Institute of Informatics

ない。

(1) パターンの発見・記述(マイニング)の組織的かつ体系的な方法: パターンは新たに作成されるものではなく、開発経験から発見されるものである。本ワークショップでは、全参加者が共通に経験を有する問題領域への体系的パターンマイニング手法(例えば12))の適用実験を通じて、パターンマイニングの効果的な実施方法を探る予定である。

(2) 定石の記述に最適なパターン/パターンランゲージの記述形式と、その形式がプロダクトや開発活動にもたらす影響: これまでに、Alexander形式やGoF形式、POSA形式、Coplien形式(標準形式)といった様々な形式が提案され用いられている⁶⁾。これらの形式の妥当性と開発活動への影響、ソフトウェア開発における“フォース”がどのようなものでありどのように表出できるか、および、形式の選択方法などが不明である。

(3) パターンの品質特性(“無名の質(QWAN)”を含む)を確認する方法、および、品質特性がプロダクトや開発活動にもたらす影響: パターンが備えるべき基準の提案⁷⁾や、対象パターンに依存した人手によるテストの試み⁸⁾はあるものの、種々のパターン/パターンランゲージへの適用を目的とした体系的な品質特性検証の枠組みは得られてない。

(4) 相互に関連付けられたパターン集合としてのパターンカタログと、パターンランゲージの違い: パターンを語彙として、包括的に解法を作り出す言語システムとしてのパターンランゲージが、パターンカタログとどのように異なるのかについて明らかでない。

(5) 開発に用いる道具(ソフトウェアツールやプログラミング言語など)とパターン/パターンランゲージの関係: ソフトウェア開発におけるパターンが主にオブジェクト指向開発と密接に関係して発展してきた要因や、アスペクト指向⁹⁾に代表される次世代プログラム開発方法/ツールとパターンの関係について、より十分な考察が求められる。また、現在得られるパターン適用支援ツールは未成熟であり¹⁰⁾、パターンの活用を支援する実用的なツールの姿やその実現可能性について議論が必要である。

(6) パターン/パターンランゲージの活用が真に有効な工程・側面: ソフトウェア開発とは、様々な立場の人とモノが関与し、幾重もの段階を経て達成される複雑で知的な活動である。パターンランゲージが有効な適用範囲や開発工程(例えば設計)・側面(例えばプロジェクト運営や教育)を明らかにしたい。

ワークショップでは、上記の事柄について時間の許

す限りポジションペーパー発表を起点とした議論を行い、その成果を情報処理学会ソフトウェア工学研究会パターンワーキンググループ¹¹⁾などを通じて広く公開する予定である。我々は、個人や組織におけるさらなるパターン活動を通して、議論によって得られた知見が検証・発展されて、ソフトウェア開発活動および組織活動一般においてパターンおよびパターンランゲージが活用されることを期待する。

参考文献

- 1) C. Alexander, et al. *A Pattern Language*. Oxford University Press, 1977.
- 2) K. Beck, and W. Cunningham. *Using Pattern Languages for Object-Oriented Programs*. OOPSLA '87 Workshop on Specification and Design for Object-Oriented Programming, 1987. <http://c2.com/doc/oopsla87.html>
- 3) E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1995.
- 4) 松下, 平山, 青木, 廣田, 原, 権藤, 鷲崎, 細谷. ウィンターワークショップ・イン・石垣島参加報告, 情報処理学会第145回ソフトウェア工学研究会報告, 2004-SE-145, Vol.2004, No.87, 2004.
- 5) W. Cunningham. *The Patterns Movement. TheServerSide.NET: Tech Talks Library*, 2004. <http://www.theserverside.net/talks/videos/WardCunningham/interview.tss?bandwidth=dsl>
- 6) 杉本 訳. ウォード・カニンガムのインタビュー(パターンについて). [DP/ML:2755], 2004.
- 7) 鈴木, 田中, 長瀬, 松田. ソフトウェアパターン再考. 日科技連, 2000.
- 8) B. William, et al. *The Software Patterns Criteria: Proposed Definitions for Evaluating Software Pattern Quality*, 1998. <http://www.antipatterns.com/whatisapattern/>
- 9) M. Al-Sabt, et al. *Testing Software Patterns*. Microsoft Patterns & Practices, 2003. <http://msdn.microsoft.com/architecture/patterns/testingpatterns/>
- 10) 長瀬, 天野, 鷲崎, 立堀. AspectJによるアスペクト指向プログラミング入門. ソフトバンクパブリッシング, 2004.
- 11) 羽生田 監修, 金澤, 井上, 森下, 鷲崎, 佃, 細谷, 瀬戸川, 山野, 沖田 著. ソフトウェアパターン. ソフトリサーチセンタ, 2005(出版予定).
- 12) <http://patterns-wg.fuka.info.waseda.ac.jp/>
- 12) K. Lappe. *Keynote: RE Patterns Working Group*. International Workshop on Requirements Engineering Patterns, 2004.