パターンランゲージによる知識共有

友野 晶夫

Japan PLoP 代表

mtomono@acm.org

概要

筆者は、2000 年 7 月より「ポータルシステム構築のためのパターンランゲージ」（Pattern Language for Constructing Portal System、以下PLfCPS と略する）を記述し始め、それをいくつかのプロジェクトに適用してきた。本論分では、このパターンランゲージを通じて実際に起きた知識共有を考察する。

1. 用語定義

(1) パターン

くり返し発生する一定の出来事[1]、それを生成する空間的形態。そうした空間的形態を生み出すパターンもあり、再帰的に存在する。

(2) パターンランゲージ

パターン群とその適用順序を示したもの。「文章」としての対象を生み出す「言語」としての機能を有する。

(3) パターンマイニング

現存するプラクティスやシステムからパターンを抽出すること。

(4) ポータルシステム

複数のユーザがアクセスし、それぞれの目的のために利用するシステム。

(5) コンテキスト

ユーザが一まとまりとして認知するシステムとの対話。

2. 背景

PLfCPS を記述し始めた当時、Web を介してサービスを提供するシステム形態が盛んに求められるようになりつつあった。このシステム形態は、サービス提供範囲の拡大やソフトウェアの配置に関わる運用コストの低減に有効である。一方で、Web は、そもそも一方的な情報の配信を目的として生まれた。アプリケーション構築を可能とするためのさまざまな機構は、後から逐次的に追加されたものである。今日においても、そうしたアプリケーションのための開発環境は十分に整備されておらず、それら機構の統合に関してかなりの部分がアプリケーション開発者の負担となっている。

また、Web 関連の案件以外にも、分散環境におけるコンテキストの実現やオブジェクトラッピングへの対処が問題となっているプロジェクトが数多く存在した。

3. 「ポータルシステム構築のためのパターンランゲージ」

本パターンランゲージの中心的な課題は、各コンテキストを独立したものとして実現することにある。技術的な側面から見ると、それ以外にもシステムの並列性の確保、および、アプリケーション領域とシステム領域の分離（データベースラッピングを含む）の実現を課題として取り上げている。

本パターンランゲージは、既に 4 つのプロジェクトに適用された。それらの規模は 17-66KSLOC である。そのうち 3 つは既に構築段階を終えている。

また、本パターンランゲージの記述は完成していない。適用を行うごとに得られた新たな知識がパターンランゲージにフィードバックされ、それをパターンランゲージに取り込むということを継続的に行っているためである。技術的な環境は変化しつづけているため、今後とも完成することはないであろう。これは、オレゴン大学において 1970 年より適用されているパターンランゲージ[2, 3]を見てもわかるように、パターンランゲージにおいては一般的なことといえる。

4. PLfCPS を通じた知識共有

PLfCPS の利用者には一切の義務を課さなかったが、本節で述べるような知識共有が自然発生的に行われた。したがって、この節に述べる知識共有は、すべてインフォーマルなプロセスを通じて行われた。

次の各小節のようなことが見られた。

4.1 パターン内容の伝達

筆者がアーキテクトとして参加したプロジェクト以外では、筆者はプロジェクトに対して PLfCPS についての指導を行っていない。しかし、現在では PLfCPS の適用経験者は既に 10 名を超えている。PLfCPS は、主に、PLfCPS の適用経験者により広く伝えられた。

また、適用経験者がプロジェクト間で移動することにより、PLfCPS の伝達範囲も広がった。

4.2 パターンの問題点の共有

PLfCPS をそれぞれのプロジェクトに適用した者から数多くの質問が筆者に寄せられた。また、記述上の誤りや改善点の指摘も受けた。これらの問題点は、さまざまなプロジェクトの下で複眼視的に明らかにされたものである。また、多くの場合、問題点とともに次の小節で述べるような解決策も報告された。

4.3 パターンの改善

適用途上のプロジェクトから新たな解決策が数多く報告された。また、上述の問題点に対応するため筆者が新たな解決策を用意する場合もあった。また、パターンランゲージを適用したプロジェクトに対して適用後にパターンマイニングを行い、その知見を集積している。

現段階では、こうした新たな解決策を筆者がパターンランゲージに反映している。

4.4 コミュニティの醸成

PLfCPS を介して、プロジェクトとは独立に緩やかなコミュニティが形成された。このコミュニティにおいて現在も PLfCPS の課題についての継続的な議論が持たれている。また、リライトを伴わない簡単な質疑応答に関しては、筆者を介さずに適用経験者間で行われている。

5. 考察

5.1 学習方法の伝達

パターンには、解決策が有効であることを示すリーズニングが示されている。これにより解決策の正当性とスコープが利用者に伝えられる。また、パターンランゲージには、内包する各パターンの適用のプロセスが示されている。このようにパターンランゲージには学習方法が内在している。この点が適用経験者間で PLfCPS の伝達を行いやすいものとしたと思われる。

これは、従来のフレームワークがもっていた急峻な学習曲線の問題を解消するのに有効であろう。

5.2 適用経験者による知識媒介

パターンランゲージを一般の設計ドキュメントと比較したときに、暗黙知への依存度に差異が見られる。設計ドキュメントでは、プロジェクトにおける合意に立脚しているため、暗黙知に大きく依存している。一方でパターンランゲージの記述においては、リーズニングを通じて暗黙知への依存度が低減されている。

この点が適用経験者にプロジェクトをまたいでパターンランゲージを媒介することを促したと思われる。

5.3 言語を基盤としたコミュニティ

PLfCPS 自身が議論の立脚する「言語」として機能している。各プロジェクトでは暗黙のうちにこのような「言語」が形成されているが、それはプロジェクトの終了とともに雲散霧消してしまう。これをプロジェクトのように期間が限定されたアクティビティから独立させ、明示することにより、継続的なコミュニティが形成されたと思われる。

5.4 自律的な適応

継続的なコミュニティでの議論を通じ、前節で述べたような問題点の共有や改善がコミュニティのメンバにより自発的に行われることで、パターンランゲージは新たな技術環境に自立的に適応していくものと思われる。これは、「言語」という機能に内在するプロセスと考えられる。

6. 課題

パターンランゲージの有効な点をより広い範囲で活かし、また、コミュニティの存続を磐石なものとするために次のような課題が挙げられる。

6.1 新たなパターンランゲージの育成

PLfCPS の有効性は明らかになりつつある。そこで、PLfCPS の領域以外にもパターンランゲージを配置することで同様の効果を期待したい。

現在、ある部署では CRM に注力することを決定している。CRM に関するパターンランゲージを起こし、PLfCPS の辿ったプロセスを再現することに取り組んでいる。

6.2 パターンランゲージの存在の通知

PLfCPS が有効であっても、その存在が知られていなければ効果を生むことはできない。今後、パターンランゲージを増やしていくことを考えると、パターンランゲージごとに広報活動を行うのが得策とは思えない。この問題を解決するために、JPLoP[4] のようなパターンコミュニティへと各パターンランゲージのコミュニティを統合していくことを考えている。

6.3 ライタの育成

多くの場合、問題点を指摘する者は解決策も示してくれている。現時点では、PLfCPS の改善点の反映を筆者が行っているが、パターンライティングの技法さえ伝えれば、Open Source の開発スタイルのように各自がパターンランゲージを発展させていくことができるようになる。

これが実現した時点で、PLfCPS は本当の意味で筆者の手を離れ、自律した言語として存在するようになる。

6.4 記述形式の洗練

パターンランゲージの改善を自律的に行うようになったときの記述の質を保つためにも記述の評価を継続的に行っていく必要がある。このための仕掛けとして、Writers' Workshop[5] に準ずるものを導入する必要があるだろう。

参考文献

[1] Alexander, C. The Timeless Way of Building, Oxford University Press, 1979.

[2] Alexander, C. The Oregon Experiment, Oxford University Press, 1975.

[3] University of Oregon Planning Office : http://www.uoregon.edu/~uplan/

[4] JapanPLoP : http://www.kame-net.com/jplop/

[5] Coplien, J. O., Bobby Woolf. A Pattern Language for Writers' Workshop : Pattern Language of Programming Design 4, Addison-Wesley, 2000.