

# 運用者能動学習型開発に関する研究

## Study on Operators Active Learning Development

高橋良輔 当麻哲哉

Ryosuke Takahashi Tetsuya Toma

慶應義塾大学 大学院システムデザイン・マネジメント研究科  
Graduate School of System Design and Management, Keio Univ.

### 要旨

情報システムは、一度リリースされてしまうと、その複雑性や柔軟性にもよるが、機能変更・改善が容易でなくなる。そのような状況でも、運用し続けなければならないため、システムが提供する機能が不十分であればあるほど、利用者や運用者への負担は増える。一方で、開発者の視点に立つと、利用者は運用者は情報システムのライフサイクルの最初と最後でしか関わらないため、手戻りが発生するなどして、プロジェクトの遅延要因にもなりかねない。本研究では、そのような開発者と運用者の双方の負担を減らすのに効果があるモデルとして、運用者能動学習型開発 (Operators Active Learning Development) というものを提案し、それが様々なコスト削減に繋がることを実証する。

## 1. はじめに

情報システムのライフサイクルは、一般的に、構想、要求分析、設計、実装、テスト、運用、保守という工程 (フェーズ) から成り立つ。[1][2]

この内、保守フェーズは費用的に全体の約7割を占め、企業のIT関連の予算を圧迫しており、それ以外の問題として、システムの複雑性や機能不足、柔軟性不足がある。[3]

こういった問題を保守で解決できない場合、その皺寄せは運用に及び、運用者の負担が増える。

本研究では、開発者および運用者の負担を減らすための手法のモデル化に取り組み、その効果がいかに優れているかを実証する。

## 2. 提案する手法

情報システムの開発プロセスモデルとして、ウォーターフォール、反復 (スパイラル)、プロトタイプリング、アジャイルがある。[4]

この内、プロトタイプリングモデルは、開発の早い段階で試作品を作成するので、利用者・運用者と開発者との認識のずれや曖昧さを取り除くことができるメリットがある。

しかしながら、試作品の作成と製品の開発と2回の実装を行うため、開発コストが余計にかかるデメリットがある。[5]

そこで、開発者が試作品および製品を開発する手間やコストを省くために、運用者にも積極的に開発に関わってもらうモデルを提案し、その効果を測定する。

この開発モデルを、運用者能動学習型開発 (Operators Active Learning Development) と命名し、モデルの適用対象として、苗場スキースクールのWebサイト (<http://naebass.jp/>) を選ぶ。

当該サイトを選んだ理由・背景として、当該スクールがWebサイトの運用による営業活動を重視していること、既存のWebサイトの開発・保守は開発者だけしか行えないこと、開発者が運用者の要求にスムーズに応じられないため、運用者が不満を募らせて心的なストレスを抱えていることがある。

## 3. 実証検証

2010年9月に当該スクールのWebサイトの試作品を作成し、10月にその試作品を元に製品を仕上げ、11月に既存サイトを置き換えてリリースした。

WebサイトのプラットフォームとしてCMSに着目し、その中からデータベース (DB) とスクリプト

言語 PHP で構成される WordPress を採用した。

開発者は本研究の発表者である高橋だが、運用者にも開発方法を伝授して、製品の実装も一部担っていただいた。

**表1** ステークホルダーの作業役割と影響範囲

	開発者	運用者	利用者	影響範囲
テーマの変更, 編集	○			サイト全体
プラグインの有効化, 編集	○			サイト全体
ユーザーの編集	○			サイト全体
ファイルの編集	○			サイト全体
設定の管理	○	○		サイト全体
インポート	○			サイト全体
カテゴリーの管理	○	○		全ページ
リンクの管理	○	○		全ページ
フィルターなしの HTML	○	○		1 ページ
投稿された記事の編集	○	○		1 ページ
他のユーザーの記事の編集	○	○		1 ページ
ページの編集	○	○		1 ページ
ファイルアップロード	○	○		複数ページ
記事の投稿, 編集	○	○		1 ページ
閲覧	○	○	○	1 ページ

表1にあるように、運用者は影響範囲が主に1ページ内にとどまる作業が大半であり、サイト全体に及ぶものは殆どない。

これは、サイト全体に影響が及ぶような作業に関する専門的知識を運用者が有していないことと、そのような作業を行って、万が一、異常が発生した時に対応できないリスクを回避するためである。

一方で、ページの編集など影響範囲がページ内にとどまる作業を運用者に委譲することにより、運用者は開発の初期段階で WordPress の操作方法を学習することにもなった。

これは学習コストの大半が運用・保守フェーズではなく開発フェーズの実装・テスト段階で発生することを意味しており、かつ開発者の作業負担を減らす要因にもなった。

具体的には、通常であれば開発者が作成するはずのページ（2010年11月5日時点で66枚）を全て運用者に作成していただいた。

その代わり運用者の負担は増えたけれども、以前はページの作成・編集を全て開発者に依頼しており、出来上がるまで約1週間かかっていたところ、今回は自分自身でページを作成するので、その場で作業成果を確認することができるようになり、その分の心的なストレスは解消された。（図1参照。）



図1 新旧 Web サイトの比較

最後に、運用者から得られたフィードバックを列挙する。

- ・ WordPress という新しいツールを学ぶことができた。
- ・ 今まで、原本が個人持ちされており、その個人しか更新することができなかった。それが、Web上で原本を共有する仕組みに変わり、複数人間が更新できるようになった。
- ・ 原本が個人持ちでないため、どの環境（自宅、スクール、その他）でもログオンさえすれば更新ができるようになった。
- ・ ページの追加や更新が容易である。また、構成（どのページ配下にする、順番をどうする等）が簡単に換えられるようになった。
- ・ 昔の考えでは、公の場に間違いやとんでもないことを露出するのは、信頼やモラルに反する事のように言われていたけど、Twitter 等のようなものが日常化してきて、その辺が崩れてきていると思う。それは、印刷物と違い後に残らないものであり、変更が直ぐに簡単に誰でもできるようになり、あまり問題視されなくなったからだと思う。

#### 4. 考察

開発者は製品を開発するための労力を減らすことができた。

一方、運用者に開発初期から能動的に開発に関わってもらうことによって、以前のサイトと比べ、運用者の心的負担を減らすことに繋がった。

更に、運用者自身が開発に関わり、何度もページを作成・編集を繰り返す内に、サイトの操作手順も自然と覚えていき、その結果、開発者が運用者に操作手順を教える手間も省くことができた。

よって、運用者能動学習型開発は、開発者と運用者にとってお互いに利益を享受できるモデルであると言える。

#### 5. おわりに

今後の課題として、まず運用者は専門的な知識を殆ど有していないため、たとえ能動的に学習しても、できる範囲は開発者と比べると限りがある。そのため、例えば、開発者が運用者に「砂場」の様なものを提供して、その中で自由に開発してもらう環境を整えたりして、運用者が楽しく開発できるように導くことが重要である。

また、運用者が開発したサイトはユーザビリティなどの標準から外れていることも少なくないので、そこは開発者がフォローに回って修正を手助けしてやるなど、運用者をサポートすることも重要である  
と考える。

### 参考文献

- [1] C. P. Barr, E. D. Callender, “Life-cycle Management & Documentation Concepts for The Space Station Program”, IEEE, 1988, pp.33.
- [2] 原田俊彦, “企業 IT 動向調査 2008”, 日本情報システム・ユーザー協会, 2008, pp.159.
- [3] Joe Tucci, “Keynotes - EMC World 2010”, EMC Corporation, 2010.
- [4] Craig Larman, Victor R. Basili, “Iterative and Incremental Development: A Brief History”, IEEE, 2003.
- [5] Ellis Horowitz, John B. Munson, “An Expansive View of Reusable Software”, IEEE Transactions on Software Engineering, VOL. SE-10, NO. 5, 1984, pp.477-487.