

## 不確かさを抱擁するソフトウェア開発

深町拓也<sup>†1</sup> 鷗林尚靖<sup>†1</sup>  
細合晋太郎<sup>†1</sup> 亀井靖高<sup>†1</sup>

本稿では、不確かさを抱擁するソフトウェア開発の重要性について議論すると共に今後の研究の方向性を示す。

### Embracing Uncertainty in Software Development

TAKUYA FUKAMACHI,<sup>†1</sup> NAOYASU UBAYASHI,<sup>†1</sup> SHINTARO HOSOAI<sup>†1</sup>  
and YASUTAKA KAMEI<sup>†1</sup>

This paper discusses on the importance of embracing uncertainty in software development and shows the future research directions.

#### 1. はじめに

ソフトウェア開発は「不確かさ」(Uncertainty)との闘いである。技術者は、「要求仕様の曖昧さは早い段階で解消すること」「設計には曖昧さを含まないこと」などを開発の拠り所としている。しかし、これらを完璧に実施することは現実的には難しい。

従来のソフトウェア工学は「不確かさ」を避けるべきものとし、技術的にきちんと扱って来なかった。現在、多くのシステム開発は従来以上に「不確かさ」が生まれやすくなっている。SNSに代表される昨今のシステムは多様な価値創造を目的としており、システムの利用が新たなニーズを生む。システムに対する要求仕様や設計方針をある時点で決定したとしても、すぐにその前提が成り立たなくなる。方針決定もいくつかの選択肢の中から選ぶことが多く、決定までは「不確かさ」として存在することになる。従来のソフトウェア工学では、「不確かさ」に対するサポートはリスク管理などの方法論に留まっており、設計や実装・テスト技術の中では明示的に扱われて来なかった。

今後は、「不確かさ」を抱擁(あるいは包容)するモデル駆動開発支援系、プログラミング言語処理系、検証系、欠陥予測モデルなどを提供していくことがソフトウェア工学の研究上重要になっていくものと思われる。

#### 2. DevOps時代のソフトウェア開発

筆者らが、「不確かさ」に関する研究に興味を抱いたきっかけは、Twitter社の方から同社におけるソフトウェアリリース方法の話聞いたことである。Twitter社では新幹線の運行モデルに近い方法をとっている。定刻になったら新幹線が出発するように、毎日、定刻になったらシステムリリースする。その定刻までに完成した機能のうちレビューの投票を通じてOKとなったもののみがリリースの対象となる。この話は衝撃的であり、従来のソフトウェア工学的な品質保証を覆すものである。Twitterのようなシステムは停止することはできないし、保守や機能改善も随時実施せざるを得ない。従来のシステム開発(リリースが年や月単位)が離散的であるとする、これからのシステム開発は連続的(毎日)になる。このようなソフトウェア開発形態は一般的にDevOpsと呼ばれ、開発(Development)と運用(Operation)が一体化している。今後、より多くのソフトウェア開発がこのような形態に移行していくものと予想される。

DevOps時代のソフトウェア開発は、図1に示すように「不確かさ」を常に抱えた上で開発し、「不確かさ」が解消した部分から連続的にリリースする時代になると予想される。そのためには「不確かさ」を開発支援の中に明示的に位置づける必要がある。

<sup>†1</sup> 九州大学  
Kyushu University

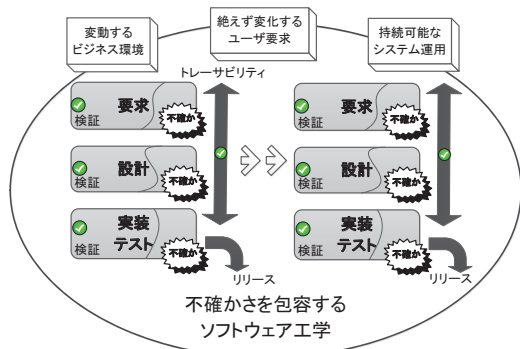


図 1 DevOps 時代のソフトウェア開発

表 1 不確かさの分類

不確かさ	要求	解	ソフトウェア工学研究としてのチャレンジ事項
レベル 1	○	△	不確かさが存在しても要求分析・設計・実装が継続できるモデル駆動開発支援機構 (不確かな設計・実装の着脱支援, 不確かさを含む設計モデルの整合性検証, 不確かさを考慮した設計とコードのトレーサビリティ検証, 不確かさの着脱を考慮したモデルからのコード生成, 不確かさの影響分析の実現など)
レベル 2	○	×	同上
レベル 3	△	×	要求仕様の不確かさを解消する支援機構 (実行トレースからの非機能要求抽出の実現など)
レベル 4	×	×	要求変化に対応してシステムを自己適応させる機構

### 3. 不確かさの分類

筆者らは、表 1 に示すように、「不確かさ」にはレベル 1 から 4 までであると考え (不確かな部分が存在しない場合は便宜上レベル 0 とし、数字が大きくなるほど「不確かさ」が高くなる)。レベル 1 と 2 は設計・実装の不確かさである。レベル 1 は、要求仕様に基づき一応の解 (設計・実装) を与えることできるが、それを正式に採用できるか否かが不確かな場合である。レベル 2 は、解を与えることができない状況である。すなわち、将来問題となりそうなことを要求仕様として抽出できているが、それをどう実現して良いのかが不明な箇所が存在する。レベル 3 は、将来問題となりそうな事項が不確かなながらも想定できている場合である。しかし、正式な要求仕様として与えることができない。レベル 4 は将来的問題が全く想定できていない状況で、リリース後に問題が顕在化した場合はシステムを新たな要求に適応させる必要がある。

### 4. 不確かさに関わる研究と今後の展望

ソフトウェア工学研究のコミュニティでも「不確かさ」に関する研究は重要視されつつある。実際、ここ数年、「不確かさ」を取り扱った論文が数多く発表されている<sup>1)-8)</sup>。これらの研究は、要求や設計モデリングにおける「不確かさ」、テストにおける「不確かさ」、「不確かさ」が表現可能なモジュール化機構など多岐

表 2 不確かさに関わる研究テーマ

区分	研究トピック	不確かさのレベル
現在の主な研究	1. 不確かさを含むモデリングとその検証 <sup>2)</sup>	レベル 1
	2. 要求分析段階の不確かさ <sup>6)</sup>	レベル 1
	3. 不確かさに起因する変更波及 <sup>7)</sup>	レベル 1
	4. 不確かさを含むモデル変換 <sup>3)</sup>	レベル 1
	5. 不確かさを許容するテスト技術 <sup>1)</sup>	レベル 1, 2, 3
	6. 不確かさを許容するモジュール化機構 <sup>4)</sup>	レベル 1
	7. 不確かさを許容する自己適応システム <sup>8)</sup>	レベル 4
今後の研究課題	1. 不確かさを許容するリファクタリング	-
	2. 不確かさの自動修復	-
	3. 不確かな箇所の特定化技術	-
	4. 不確かさの履歴管理とその可視化	-
	5. 不確かさに関するリポジトリマイニング	-

に渡っている。表 2 に現時点での主な研究トピックと今後の研究課題を示す。取り扱っている「不確かさ」のレベルも付記する。現時点の研究の多くはレベル 1 の「不確かさ」にとどまっている。

アジャイル手法は「変化を擁護」した (Embrace Change)。これからのソフトウェア開発は「不確かさを擁護」しなければならない (Embrace Uncertainty)。「不確かさ」に関する研究はまだ始まったばかりであり、今後の発展が期待される。

謝辞: 本研究は、文部科学省科学研究補助費 基盤研究 (A) (課題番号 26240007) による助成を受けた。

### 参考文献

- 1) Elbaum, S. and Rosenblum, D. S.: Known Unknowns: Testing in the Presence of Uncertainty, FSE 2014, pp.833-836, 2014.
- 2) Famelis, M., Salay, R., Chechik, M.: Partial Models: Towards Modeling and Reasoning with Uncertainty, ICSE 2012, pp.573-583, 2012.
- 3) Famelis, M., Salay, R., Sandro, A., Chechik, M.: Transformation of Models Containing Uncertainty, MODELS 2013, pp.673-689, 2013.
- 4) Takuya Fukamachi, Naoyasu Ubayashi, Di Ai, Peiyuan Li, Yu Ning Li, Shintaro Hosoi, and Yasutaka Kamei: Uncertainty-aware Architectural Interface, AOAsia/Pacific 2014 (Workshop at FSE 2014), pp.4-6, 2014.
- 5) Raccoon and Dog: Unknownness, ACM SIGSOFT Software Engineering Notes, Volume 38 Issue 5, pp.8-17, 2013.
- 6) Salay, R., Chechik, M., Horkoff, J., and Sandro, A. D.: Managing Requirements Uncertainty with Partial Models, Requirements Engineering, Volume 18, Issue 2, pp.107-128, 2013.
- 7) Salay, R., Gorzny, J., and Chechik, M.: Change Propagation Due to Uncertainty Change, FASE 2013, pp.21-36, 2013.
- 8) Yang, W., Xu, C., Liu, Y., Cao, C., Ma, X., Lu, J.: Verifying Self-Adaptive Applications Suffering Uncertainty, ASE 2014, pp.199-210, 2014.