

# Osterweil の夢は AI で開くか

田中 康<sup>††</sup>

ytanaka@kplus-solutions.com

中村 航一<sup>††</sup>

JL25377@jp.ibm.com

八木 将計<sup>†††</sup>

masakazu.yagi.zd@hitachi.com

八木 香織<sup>†</sup>

kyagi@kplus-solutions.com

泉川 大樹<sup>††</sup>

hirokiiz@jp.ibm.com

佐藤 綾美<sup>††</sup>

E35457@jp.ibm.com

山ノ内 日菜子<sup>††</sup>

HINAKO.YAMANOUCHI@ibm.com

池田 香織<sup>††</sup>

Kaori.Ikeda2@ibm.com

荒井 怜志<sup>††††</sup>

saaramobile@gmail.com

\*奈良先端科学技術大学院大学, †有限会社ケイプラス・ソリューションズ, ††日本 IBM 株式会社, †††株式会社日立製作所, ††††株式会社ワンノート

## 要旨

1987 年の Osterweil の歴史的な論文「Software Processes are Software, Too<sup>[1]</sup>」は、ソフトウェア工学のパラダイムの大きな転換を予兆するものであった。しかし、彼の先見は、彼が描いていたような成果に届かないまま現在に至っている。

我々が取り組んでいるプロセスモデルの研究に加えて、昨今目覚ましい発展を見せている AI 技術の適用が、彼の構想の実現を後押しするのではないかと考えている。本論文では、彼の構想を再理解したのち、我々の取り組みを整理して Future Presentation の議論の前段とする。

## 1. Osterweil の夢

Osterweil は 1987 年の彼の論文で、プロセスプログラムという観点を提案した。彼が提案したプロセスプログラムは、生産的活動であるソフトウェア開発プロセス自体を「ソフトウェア」と捉えて、工学的に扱おうという考えかたである。

彼は、プロセスプログラムがソフトウェアを作るためのソフトウェアであるという「二重の間接的で厄介なプロセス」であるとしながらも、人間の生得的な問題解決能力をプロセスプログラムとして取り扱うことが、ソフトウェアの開発と進化において重要であると示した。さらに彼は、この考えかたこそが、ソフトウェア工学の本質であるとも主張した。

## 2. 夢の中へ

Osterweil の探しものは何だったのか。

彼は、ソフトウェア開発プロセスそれ自体を実行可能なコードで表現するものとしてプロセスプログラムを提唱した。それを受けて、彼の提案したプロセスプログラムを記述するための適切な記述言語の研究が、例えば、手続き型言語、関数型言語、論理型、ビジュアル言語などの側面から行われた。

1987 年の論文の 10 年後、ICSE9 で行われた同タイトルの講演で彼は、10 年間の様々な研究者による取り組みの中では、技術的な課題よりも哲学的な議論が主要な関心事となりつつあること、ソフトウェアやプログラミング言語の意味論についての考察が重要であること指摘した<sup>[2]</sup>。

さらに同会議において Lehman は、プロセスプログラムに対する Osterweil の提案を受けたその後のアプローチが有用性を失っていると主張し、新たなアプローチと技法の開発が必要であると述べた<sup>[3]</sup>。しかしその後、プロセスプログラムに関する研究は大きな成果が得られないままトーンダウンして現在に至っている。

## 3. 夢を追いかけて

プロセスプログラムの考えかたがソフトウェアの開発と進化において重要であると期待した彼の論文では、一般化された解の仕様を作成し、それを実行可能な形にすることが問題解決の鍵であるという仮説が立てられた。また彼は、プロセスプログラムを、生産物の作成やタスク達成に対する活動(タスク)であるとし、タスク視点によるプロセスの記述とプロセス間の関係について整理した。その中で Osterweil は、プロセスプログラムの開発において、プロセスプログラムとプロセスプログラムクラスが実行可能なコードで表現される必要があるという前提を示していた。しかし、Lehman はこの前提自体に疑問を投げかけ、より適切なアプローチの必要性を指摘した<sup>[3]</sup>。

#### 4. 夢をあきらめないで

1987年のOsterweilの論文では、人が、プロセスの特定を通じて問題を解決するという能力そのものが重要であると考察されている。同様に、Pressmanも、技術において重要なのはプロセスであり、目に見えてわかりやすいツールではなく、プロセスを重視すべきだと彼のソフトウェア工学のレイヤーモデルを使って述べている<sup>[4]</sup>。

我々は、図1に示すように、Osterweilが提起したプロセスプログラムの有効性の鍵が、プロセスプログラムクラスの定義方法、要求に対して整合の取れたプロセスプログラムクラスの選定方法、選定したプロセスプログラムクラスからプロセスプログラムを定義する方法、そして定義されたプロセスプログラムを実行プロセスとして制御する方法として定義できるのではないかと考えている。さらに、これらのプロセスプログラムクラス定義とその活用が、問題領域を構造的に定義することができるプロセスモデル手法に基づいてのみ実現可能ではないかという仮説を立てている。

#### 5. 夢の途中

PRePモデルは、業務プロセスを概念実体として定義し、それらの関係を構造的に記述するプロセスモデリング手法である<sup>[5]</sup>。我々は、プロセスのクラスをパターンとして定義する部分、および、定義したプロセスパターンをプロセスプログラムとしてインスタンス化する部分にPRePモデルを適用する。さらに、PRePモデルで構造的に記述したプロセスから、処理や活動の観点に展開するための方法としてBPMNに基づいたBlueworks Liveを適用する。

さらに、図1に示す各抽象度からのインスタンス化を支援する仕組みとして、IBMのWatsonおよび、watsonx OrchestrateのAIプラットフォームを活用する研究を進めている。これは、あらかじめ固定化されたプロセスプログラムではなく、解決すべき問題自体を特定し、特定した問題に即したプロセスプログラムを再構成することに多くの経験と知識が求められると考えられているためである。

ソフトウェアプロセスの構造的なクラスパターン定義方法と、AIによる実行時の再利用支援のアプローチは、Osterweilの構想を有用な問題解決手段として実現できるのではないかと考えている。その中で、プロセスパターンをクラスとして定義する基準や、プロセスプログラムクラスから適切なパターンを抽出し構成する方法は今後の研究課題である。

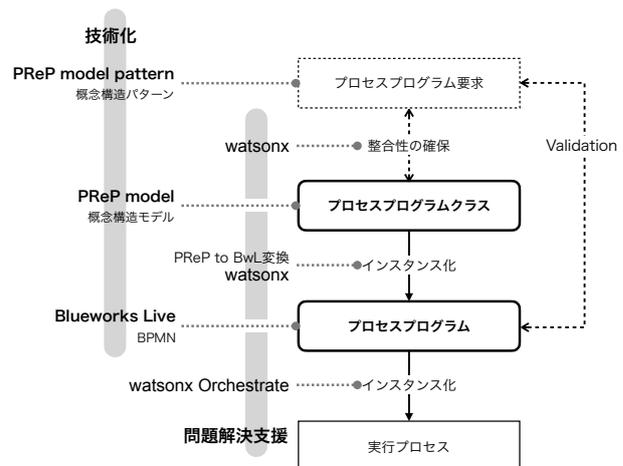


図1 プロセスプログラムの抽象度とその利用モデル。

#### 参考文献

- [1] Leon J. Osterweil, Software Processes are Software Too, Proceedings of the 9th international conference on Software Engineering, ICSE 1987
- [2] Leon J. Osterweil, Software Processes are Software Too, Revised, An Invited Talk on the Most Influential Paper of ICSE 9, 1997
- [3] M.M.Lehman, Process Modeling - Where Next, An Invited Talk on the Most Influential Paper of ICSE 9, 1997
- [4] Roger S. Pressman, Software Engineering a practitioner's approach, Mc Graw Hill, 2001
- [5] 田中 康, PReP model - 現実世界をデザインする - PReP モデルによる業務レベル設計, POTASSIUM PRESS, 2022