

ソフトウェアの少子高齢化が急加速 ～開発中心パラダイムに未来はあるか?～

増井 和也[†]

1. はじめに

書名に「開発」を含むソフトウェア関連図書は多数出版¹⁾され、「ソフトウェア開発」という言葉に違和感を持つ人は少ないだろう。その図書群から、「開発」作業のイメージや期待値はソフトウェアを新規に創り出す創造性のある作業との印象を強く受ける。

では、ソフトウェアに関連する現場では、新規にソフトウェアを創作する作業が大半を占めているのだろうか。実際はそうではなく、長年稼働している既存ソフトウェアに対し、法律の改正、顧客や利用組織の要件変化、基盤ソフトウェアの更新、競合相手との優位性の拡大・維持、潜在不具合や稼働後の改修ミスの発見などに伴う修正または比較的小規模な機能追加（エンハンス）を行う作業が大半を占めるとの話を聞くことがある。このような作業の費用は当該ソフトウェアの稼働開始から5年間累計で初期開発費の2倍を超えるという報告²⁾もあり、ソフトウェア技術者の多くがこのような稼働中ソフトウェアの延命作業に携わっているというのが筆者の認識である。既存ソフトウェアの延命作業が大半を占め、新規（初期）開発が極端に少ない状況が続くことを筆者は「ソフトウェアの少子高齢化」の状態と呼ぶことにする。

本稿は「ソフトウェアの少子高齢化」の状態が今後も拡大する根拠、従来の「開発中心パラダイム」が適用可能な範囲の縮小、将来に向けて既存ソフトウェアを中心にした新しい「保守中心のパラダイム」にシフトする必要性を、筆者個人の立場で論及する。

2. なぜソフトウェアは少子高齢化に向かうのか？

(1) 筆者のソフトウェアに係る経験

筆者は大学で経済学を学んだ後、1976年から（途中40～50歳台前半、管理職を兼務する時期もあったが）現在までほぼ一貫してソフトウェア技術者として従事してきている。その間、いわゆるプログラマ、SE、プロジェクトのリーダー(PL)、マネージャ(PM)、ソフトウェア現業部門の技術専門職、部門長の立場を経験した。また、OS・装置ドライバなどの基本ソフトウェアから各種業務・業種の受注アプリケーション、複数のパッケージソフトウェアまで、さまざまな種類やタイプのソフトウェアに関わってきた。

筆者の経験には、まったくのソフトウェア新規開発として各開発期間1～2年の6プロジェクト(OS～受注アプリケーション、パッケージ)がある(含むPL, PM)。ただ、経歴の大半を占める作業分野は広義のソフトウェア保守^{3), 4)}であった。その中には、他者が開発し、長年の稼働中にさまざまな修正が無秩序に加えられてきたソフトウェアの保守を引取り、他者にいつでも引継げるよう保守体制を再構築して対応したのも多く含まれる。

筆者30歳台前半、UNIXの移植プロジェクトを(PLとして)担当した。UNIXの処理理解に大量のC言語プログラムを素早く読解する力が必要となり、プログラムの読解スキルを磨くことができた。筆者はもうすぐ64歳だが、この経験からかJava等の他者開発のソースコードを読み解き、既定の保守プロセスの1タスクとして最適なソース修正を担うこともある現役の保守技術者である。

当該保守作業の依頼は今も減る気配がなく、フルタイム勤務を継続しつつ、長年の保守作業や保守組織運営の経験から得た総合的な技術や知見は、所属のソフトウェア・メンテナンス研究会の年次報告書、著作、セミナー等で発表している^{4), 5), 6), 7)}。

(2) ソフトウェア開発はますます縮小する

筆者の既報告に、稼働中ソフトウェアの価値について考察したものがある⁸⁾。そこでは、稼働中ソフトウェアの価値は障害等でそれが機能しなくなったことによる損失額（経済学でいう機会費用: **opportunity cost**）で評価すべきとした（初期開発費の高低は当該価値と無関係）。この見方から、再開発により現在の保守・運用費用を下げる投資についての損益分岐点（**break-even point**）の存在が予測できる。

一般業務システムの例であるが、仮に新開発技術により再開発が従来想定より大幅に低い費用で可能と見積もられたとしよう。だが、再開発に伴う導入側付帯費用として、要件定義時の関連部門対応、並行稼働（当該期間は二重の設備要）、移行、各種運用マニュアル等再作成、再教育訓練、初期不良対応、開発と並行稼働期間中の既存システム保守・運用（熟練技術者が再開発にシフトすれば、代替費用や障害拡大リスク高となる）などの費用が開発作業自体とは別に発生する。これら付帯費用は、再開発費の大小よりも業務影響範囲に大きく依存する。再開発版稼働後の保守・運用費の削減がある程度期待できた場合でも、付帯費用も含む全再開発費の回収に10年以上の期間が必要な投資計画となる場合も珍しくない。上記の再開発損益分岐点は開発中心指向の考え方を持つ人が期待するほど低くないと筆者は評価した。

筆者が大手ITベンダで（事業部組織だったが）技術担当の経営管理部門を兼務した際、半期、年度、中期（3年）の期間予算達成が第一目標の状況では、各期間予算に影響が少ない費用で納まる保証がないかぎり、再開発の選択は困難ということも経験した。

反対に、筆者が考える既存ソフトウェアの価値モデルでは、保守や運用の重要性が再評価でき、当該価値の維持・向上を目的とした保守やシステム運用の適正費用がイメージできる。この見方では、今以上に保守や運用に費用を掛け、稼働中のソフトウェアの延命を図っても経済的または経営的に無駄な費用増とは言えないことになる（図1⁷⁾）。

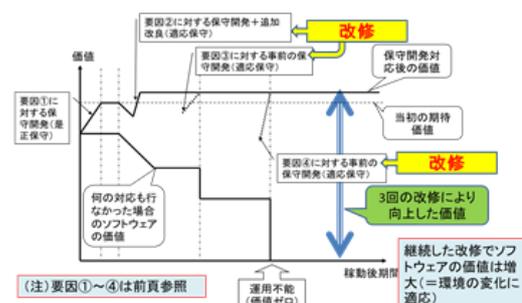


図1 適切な保守対応後の稼働中ソフトウェアの価値変化

[†]SEA 正会員、ソフトウェア・メンテナンス研究会（略称：SERC、<http://serc-j.jp/>）代表幹事

(3)ソフトウェア保守作業は隠されるが無くならない

ソフトウェア開発費が低いという甘言に乗り再開発してみたが、予算を大幅に超過し、保守や運用の費用も期待するほど下がらない場合がある。この場合、再開発投資回収が想定通りでない責任を少しでも逃れるため、実際の保守費用を別費目（例：次の開発費）で処理して保守作業が発生していないように見せる、本来対応すべき保守作業をせずに保守費用を下げた見返りとしてシステム障害リスクや障害対応の不利リスクを高めてしまっている事例に、筆者は今まで複数遭遇している。

3. 経営者の零線に触れない開発中心パラダイム

「開発」に愛着を感じる人びとの中には、スマートフォンアプリや一部の新しい分野の組込ソフトウェアで、現在も大量の新規開発が行われていることに注目しているかもしれない。それらは学会発表、情報技術関連雑誌などの媒体で大きく取り上げられることが多いこともあるだろう。逆に取り上げられることが少ない既存ソフトウェアの少子高齢化対応が実際にはおおくあったとしても、一部の停滞した分野や再開発が必要な悪い状態の事例として紹介されることもある。

ただ、脚光を浴び、新しい分野で開発されたソフトウェアも、開発後はただちに既存ソフトウェアとなり、延命対象となるのである。新規開発の生産性・品質のさらなる向上が、実は保守されるべき既存ソフトウェアを急増させ、大量の高齢化ソフトウェアを残す方向に拍車をかける。

株主から期間損益の確保を徹底して求められる厳しい経営環境の中で、経営者、システムの所有者、利用者の意思として、既に稼働実績があり、安定していて高い価値を持つ既存ソフトウェアを（たとえそれが旧式のものであっても）延命を図る意思決定は今後も一般的になされていくものと筆者は経験から予測する。

長い目で見れば、開発は **discrete** であり、保守は **continuous** といえる。開発時点のみに着目する開発中心のパラダイムでは、長年稼働しているソフトウェアの価値を正當に評価はできず、システムのオーナーである経営層に対し、「稼働中のソフトウェアは老朽化しているので新たに再開発が必要」というアピールは共鳴を得ることのないパラダイムとなってきていると筆者は分析する。

4. 今高齢化ソフトウェアのケアが求められている

高齢化（老朽化）したソフトウェアが多数残存する状態を、「あるべき姿でない」と指摘するだけでは現状の解決にはならない。住む人の大半が高齢者の町に、大きな投資をして産婦人科の大医院を設立したとしてもビジネス的に失敗することが容易に想像できる。その失敗の原因は自分には無く、町の高齢化にあるのだとの主張に傾聴する人は少ないだろう。

世の中に既存ソフトウェアが大量に存在し、その寿命が延びている（高齢化している）現在、新規開発を想定した高生産性や高品質を求めるだけの技術的アプローチをいくら洗練し、研究し、また教育しても、それを活かせる場がますます少なくなっていく現状があることを正視すべき時にきている。

高齢化社会では老人をケアするためのグッズや医療サービスが必要となるように、高齢化した既存ソフトウェアをケアするサービス（保守サービス）がもっとも重要視される必要がある。

このような保守サービスを（既存ソフトウェアを残して）どのような技術で効率化し、高品質に対応していくか、高齢化ソフトウェアをケア（保守）する技術者が何を生きがいとして従事することができるか、今後のソフトウェアの分野の人々が自らで解決すべき大きな課題となるであろう。

その対応策として、あるべきソフトウェア保守プロセス実装には医療分野のプロセスをベンチマークし、実装していくことが有効⁵⁾であると筆者は考えている。「保守は悪」という考え方を捨て、将来のためにソフトウェア保守を中心としたパラダイム（稼働中ソフトウェアが受診者、保守者が医師となるようなソフトウェア医療プロセスパラダイム）を早急に築き上げる必要性を提言したい。

5. まとめ

ソフトウェアの少子高齢化状態が急速に進む。その対応には従来の開発中心アプローチでなく、（いわゆる老朽化したものを含む）既存ソフトウェアを前提に、広い意味のケア（保守）を中心としたアプローチが必要である。

なお、「ソフトウェアの開発が無ければ保守もない」ということは明白な事実である。また、文献³⁾のソフトウェア保守プロセスにおける「修正の実施」アクティビティからは、いわゆる「開発プロセス」を呼び出すというタスクが記されている。その点から、ソフトウェアの保守作業と開発作業には、引継ぎなどの連携すべき部分と実施上に共通なタスクも少なからず存在する。

ただ、従来は開発中心のプロセスを保守作業に無理に適用する方法が採られてきたが、ソフトウェアの少子高齢化状態では、保守プロセスから始める（付随する開発タスクも含めた）パラダイムへのシフトの是非について、今後の議論活性化を望みたい。

参考資料等

- 1) 国立国会図書館サーチ:16.2.19 時点、書名に「ソフトウェア開発」を含む10~16年出版の書籍検索ヒットは734件
- 2) ユーザ企業ソフトウェアメトリックス調査2014 第7章 保守調査 分析結果 図表7-71 保守費用分析
- 3) JIS X0161:2008 ソフトウェア技術—ソフトウェアライフサイクルプロセス—保守 (ISO/IEC 14764:2006) で定義される是正保守、予防保守、適応保守、完全化保守のすべて
- 4) ~ISO14764による~ ソフトウェア保守開発 増井和也、馬場辰男他 2007 ソフト・リサーチ・センター
- 5) ソフトウェア・メンテナンス研究会 第24年次報告書 Dグループ報告 個人研究 「ソフトウェア保守で何をすべきか医療プロセスから考える (増井和也)」
- 6) ソフトウェア・メンテナンス研究会 第15年次報告書 Dグループ報告 個人研究 「ソフトウェア保守の経済-I (増井和也)」
- 7) 2015年2月 「ソフトウェアの価値をさらに高めるために「保守・改修」で攻める!!」増井和也 日本情報システムユーザ協会オープンセミナー
- 8) ソフトウェア・メンテナンス研究会 第18年次報告書 Dグループ報告 個人研究 「ソフトウェア保守の経済-IV (増井和也)」