

エージェント指向を用いた PSP 演習支援システムのエラー型詳細設計

Detailed Design of Defect Types PSP Practice Support System with Agent-Oriented

山口 大輔¹, ○塩谷 啓¹, 新美 礼彦², 高橋 宗雄¹
 Daisuke Yamaguchi¹, ○Hiromu Shioya¹, Ayahiko Niimi², Muneo Takahashi¹
¹ 桐蔭横浜大学
¹Toin University of Yokohama
² 公立はこだて未来大学
²Future University-Hakodate

Abstract: In this paper, we propose the Personal Software Process (PSP) Practice Support System efficiently to transmit only necessary programming capability and improvement for detailed design of Defect Types. This system can transmit programming to specific human among many software processes using a Agent technology. The system is also synthesized to do parallel and cooperative proposing internally. Applying the proposed method to a personal process-removing task, a flexible programming for quality of software. Agent studies the best transmission programming according to the makes planning in the personal process.

1. はじめに

近年、情報化社会の進展とともにソフトウェアは、重要性がますます高くなっており、より高度で複雑なものになっている。さらに、より速く、より良い品質で生産することが求められている。ソフトウェア技術者は、今まで以上に品質と生産性を向上させることが必要となってくる。

ソフトウェアの品質と生産性は、その開発のプロセスに依存している。従来、ソフトウェア技術者の開発プロセスは、個人の経験や直感的な開発手法に頼ったものとなっている。ソフトウェア技術者個人の不確実な予測や癖が、ソフトウェアの品質や生産性を低下させる原因となる場合がある。しかし、ソフトウェア技術者個人が規範のある、一貫したプロセスを実践することができれば、ソフトウェアの品質や生産性を向上させることができる。さらに、開発するソフトウェアの規模を、より正確に見積もることができる。

こうした膨大な連続データから、様々な情報を取得し、そのデータを解析する場合にはエージェント技術が用いられていることが多い[1]。情報処理技術の進展に伴い、我々は膨大なデータを手にする事が可能になった。エージェントは、周囲の状況を認識し、最終目標に向かい、自律的に動作するものであり、マルチエージェントは、複数のエージェント間で情報を共有し、それぞれが独立的に動作するので、共通意識に基づく目標達成に適している。この技術を適用することによって、コミュニケーションツールに必要な情報の配信と共有を効率化することが期待できる。

本研究では、パーソナルソフトウェアプロセス (PSP) 支援システムをエージェント技術と組み合わせることにより、よりソフトウェア開発を効率的に行うためのエラー型の詳細設計の検討を行う。この

システムでは、PSP 支援システムを組み込んだエージェントにより、個人のソフトウェア開発におけるプログラミング能力を向上するために注意しなくてはならないエラー型を策定する。個人のソフトウェア開発におけるプログラミング作業時のプロセス改善を、通常の PSP 支援システムだけでは発見することが出来ないコンパイルやテスト時の作業における検出されたエラーを適切に判断できる支援を行う。これにより個人のプログラミング能力が改善され、ソフトウェア開発を計画的に行い、より品質の高いプログラムを作成可能とする、個人に特化したシステムの構築を行う。原稿を執筆してください[1].

2. エージェント

エージェントは、本来与えられた仕事をこなす「代理人」を意味する。本論文では、動的な環境の中に置かれた計算機システムで、自律的に環境を知覚して行動することで、自分に与えられたゴールを達成するものをエージェントと呼ぶ[2]。また、エージェントが協調や交渉などの相互作用により個々の機能を生かして、新しい機能を創出し、さらに集合体として疑似的な知性を映し出すエージェントの集合をマルチエージェントと呼ぶ[3]。

3. パーソナルソフトウェアプロセス (PSP)

パーソナルソフトウェアプロセス (PSP) とチームソフトウェアプロセス (TSP) は、ソフトウェア開発作業を管理して、効率的に高品質なものを開発するという、個人とチームのソフトウェア開発能力を改善する手法である[7][8]。

PSP 並びに TSP を演習する際には、主に以下の作業が必要となる。

- ・ PSP 並びに TSP の作業工程の開始時から終了時

- までの時間を記録する
- ・ 開発作業で欠陥数等のデータを記録する
 - ・ 事後分析工程で今までに蓄積したデータを整理、加工し帳票を作成する
- PSP の枠組みは、図 1 に示す。

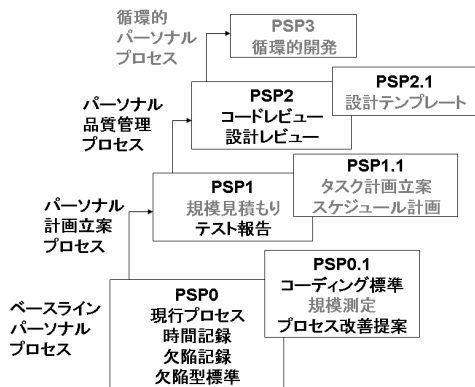


図 1 PSP の枠組み

これらの作業をサポートするソフトウェアが必要であり、幾つかの開発が行われているが、有効とされる支援システムが確立されていない。ソフトウェア開発が大規模な場合、チーム開発が非常に重要となってくる。図 1 で太字になっている部分は、個人プロセス改善中心である。それ以外の部分についてチーム特性が強くなり、個人プロセス改善だけでは、ソフトウェアを効率的・高品質に開発することは出来ない。

4. 支援システム構成

支援システムでは、前述により定められたエージェントが PSP の枠組みを個人別に記録する。記録すべき帳票内容は多数あるが、その中の 1 部であるベースラインパーソナルプロセスのプログラム規模 (LOC) を表 1 に示す。PSP では、開発プロセスによって記録すべき内容が異なり、パーソナル品質管理プロセスになると、帳票への記録内容が非常に多い。PSP 経験者やソフトウェア開発に長年従事してきている人は、帳票記録作業を然程苦にしないが、それでも作業に追われてしまうと記録が曖昧に行われることが増える。正確に記録を行うためには、自動的に行う必要性があり、作業の軽減や、プロセス測定に重要である。

記録内容が欠落すると、その先の計画を正確に行えない。そこで、支援システムは、全ての作業を記録し、その記録情報を共有し、エージェントは、そこまでの記録情報が他者との違いによって、個人の特徴を示す内容か検討を行う。図 2 に記録の流れを表す。

これはプロセス改善するために必要な情報を提供する機能を支援する重要な情報源となる。しかしながら、支援システムに全てのプロセス改善内容を提示することが委託されることは無い。共有情報自体に、プロセス改善を検討している作業員それぞれの意志が働いている。本支援システムの機能には、作業員の意志までもエージェントが読み取る自律機能を含まないものとする。このシステムの重要な点は、システムとして持つ情報を共有

する事によって、個人の特徴を早期に発見する処理が行われることにある。

表 1 プログラム規模 (LOC)

	計画値	実績値	累積値
ベース	/		
削除			
修正			
追加			
再利用			
新規・変更 LOC			
総 LOC	/		
将来の再利用 LOC			

表 2 欠陥型のテーブル

型番号	型名	説明
10	文書	コメント、メッセージ
20	構文	スペリング、区切、誤字、命令形式
30	ビルド、パッケージ	変更管理、版管理、ライブラリ
40	アサインメント	名前の重複、宣言、範囲、制限
50	インターフェース	入出力、ユーザー書式、手続き呼出と参照
60	チェックング	エラーメッセージ、チェック不良
70	データ	構造、内容
80	機能	論理、再帰、計算、ポインタ、ループ、機能欠陥
90	システム	構成、メモリ、タイミング
100	環境	テスト、設計、コンパイル、他の支援システムの問題

本研究のエラー検出能力について支援システムは、欠陥記録について、欠陥型として判別し、型は表 2 に示すテーブルによる分類分けが行なわれる。そのため、エージェント一つに対して、1 人の作業記録を行う構成が、処理の並列および分散効率に適していると考えている。そのため、エージェントの集団による支援システムの構成上、マルチエージェント技術を用いることが本支援に適した構成となっている。

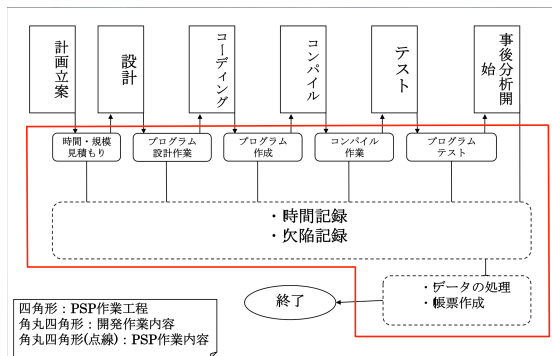


図2 PSPの記録の流れ

5. エージェント指向による支援システム

以上のように、支援システムは、マルチエージェントにより構成されるので、エージェントが記録情報の取り扱いをどのように支援に用いるかを決定することが個別にできる。そのためシステムは、情報を問わず、特定の役割を設定できるので、一つのエージェントが複数のマルチエージェントに属することで組織の人間関係を表現可能である。

そのため、システムを改善するためには、エージェント単体だけではなく、エージェント間の利用者に対する支援の方法を改善することが求められる。それによりエージェントが関係するマルチエージェント全てに影響を及ぼす。その事が、共通する問題点に関して、改善の効果を高め、システムの運用に最適である。しかしながら、情報共有と配信の関係から見た場合には、マルチエージェントの枠組みが抱える問題点全てを、必ず改善できるわけではないが、内部における相互作用により、改善することが期待できる。

運用に際しては、使用者の作業記録を行う。その記録行為は、キー操作を含むものとしている。その上で、実際の作業時間、プログラムのコンパイル、実行までを、問題に従って記録する。

問題は、15問を1サイクルとして行い、その繰返しによる作業工程から、プロセス改善を導き出すものとしている。しかし、この15問は、全行程の作成に関して10問、行程として記録が残っているものから、改善点を検討する問題が5問設定されている。今回は、エラーの検出に関する能力改善のため、まず10問の全行程作業から、改善を行う。

問題10問は、それぞれに製作者側の意図が含まれている。数問を解いた時点で、問題作成者の意図が見えてくる場合もあるが、基本的には、解答中にそうしたことを意識せずに行う。

支援システムの運用においては、問題作成が困難を極めるものである。理由として、システムの利用者が全て同じ能力とは限らない為である。本支援システムは、個人のエラー検出能力改善を目的としている為、コーディング中のタイプミスなどをなくすなどの注意喚起が多くを占めていた。欠陥型テーブルへの移行も自動で行うためには、テーブルの定義が曖昧すぎることも問題に含まれている。そのため、支援システムの出している内容が具体的にどの程度有効な支援であったかを判断

することは、非常に難しい。

また、エラー検出能力改善されたとしても、プロセス改善につながり、その効果としてプログラミング能力が必ずしも向上するとは限らない。その理由としては、作業環境の一定状況に置かれていないことにある。プログラミング能力改善に、エラー検出能力改善は、必要な能力ではあるが、全体像としてプロセス改善がさらに必要となる。エラー検出能力改善は、必ずしもプログラム開発を専門に行う人間とは限らない。そのため、外的要因が非常に多く、疲労からくる不注意などは、本システムでは品質の高いプログラムを作成可能とすることに適していても、個人の特徴抽出の判断材料にすることがふさわしいのかと言うエージェントの自律性を定義する部分においての問題点を産み出した。

どこからどこまでを個人の特徴として判断し、問題点が何かを明確にすることは、コンパイルのエラーや実行時のテストにおいて、明確に解ることはあるが、それ以外は、エージェントが他のエージェントと情報共有することによってある程度の予測ができたとしても、その理由が必ずしも適切であるかと言う判断が、現状では出来てはいないという、運用上の状態が判明している。

6. エラー型詳細設計

PSP演習において、上記支援システムを利用していても、エラーとして型の認識は、正確に欠陥型へ置き換えられていない可能性がある。

表3 拡張された欠陥型のテーブル

型番号	型名	説明
21	誤字	スペリング, 句読点
22	命令フォーマット	一般的なフォーマット問題
23	Begin-end	処理を適切に区切っていない
24	構文	命令形式
41	名前づけ	宣言、重複
42	スコープ	
43	初期化とクローズ	変数、オブジェクト、クラスなど
44	範囲	変数の制約、配列の範囲
51	内部	プロシージャ呼出しとインターフェース
52	I/O	ファイル、表示、プリンタ、通信
53	ユーザ	フォーマット、内容
81	ポインタ	ポインタ、文字列
82	ループ	1つずれ、インクリメント、再帰呼出し
83	アプリケーション	計算、アルゴリズム

現在、PSP では表 3 の内容までエラー型が拡張されている。[9]

拡張された欠陥型は、主にレビューの結果から、もっとも記述頻度の多いエラーの型に対する説明を抽出し、分類化されている。そのため、作成時に発見されたものなのか、見逃しによって出現したものかの差は明らかとなっていない。

現在の拡張されたエラー型では、主にこれまでの研究成果のレビューを基に設計されたものであるが、括弧のようにどの型と判断すれば良いのか不明瞭な項目も依然として残っている。

詳細にするべきは、括弧の取り扱いについてである。括弧はどの方と判断するかが非常に難しい。エージェントによる自律的支援を行う場合、自動補完などの機能を第一に検討する。そうした場合、型の 20、40、80 では、その意味合いが大きく変化してくる。単純に判断した場合、型 20 であり、拡張した表現の場合、型 21 を選択するべきある。しかし、拡張された中から選択した場合に作業意図から、その欠陥は型 44 や型 82 と判断してしまう場合も起こりうるのである。

そのため今回は、詳細設計に当たり、括弧による起因であったとしても、それは全て型 21 になるよう、現在の型の設計に、詳細な事例を追記し、その欠陥型判断の思考段階において、エージェントは自律的に型の修正ができる支援を行う必要がある。

7. まとめ

今回のエラー型の詳細設計は、エージェントの情報共有により支援システムから提供される新たな詳細型が、より正確に個人の欠陥型に関する理解能力の改善につながった。

エージェント指向により、演習問題理解や事後分析での変更や修正が、短時間で対応することが可能になり、欠陥型判断の誤りの検出に支障をきたすことが無く、正しい帳票作成に大きな効果をあげることが期待出来ると言える。しかし、正しい帳票作成がエージェント指向における支援システム利用者の情報共有設定のルールに登録されるログデータの質の変化をさらに生む可能性がある。登録されるログデータの質に違いがありすぎると、特徴抽出を行った結果がエージェント学習に有効なデータとして即時利用可能なデータではなくなるという問題点も考慮しなくてはならない。登録されるデータの活用方法が、ソースコードの作成内容別の適合システムに依存していることが問題ではないかとも考えられるためである。

今後、欠陥型の詳細設計が進むことで、エラー検出能力改善という利用者の本質的な特徴抽出を行える。詳細型設計によって得られる帳票データは、エージェントの強化学習が行いやすい項目群や目標設定になるため、より現実世界の秘書に近い高機能面を含めた様々な点から検討を行なっていく。

今後、こうした問題をエージェント間の自律性によって自動的に解決出来るようになれば、マルチエージェントによる創発性の一つの解が見えるかも知

れない。

参考文献

- [1] 住田一男、三池誠司：知的情報検索の動向, 人工知能学会誌, Vol.11, No.1, pp.10-15, 1995
- [2] 金淵培：Agent 技術の現状と実用化, 人工知能学会誌, Vol.12, No.6, pp.850-860, 1997
- [3] Daisuke Yamaguchi, Kenneth J. Mackin, Eiichiro Tazaki and Muneo Takahashi, Emergence in Waste Incinerator Emission Prediction using Probabilistically Optimal Ensemble of Intelligent Multi-Agents, Global Engineering, Science, and Technology Society International Transactions on Computer Science and Engineering, Vol.31, No.01, pp.9-18, 2006
- [4] 山口 大輔、ケネス J. マッキン、小田川 美伯、片山 富美代、高橋 宗雄：マルチエージェント技術を用いた情報共有と配信の支援システム, 日本ファジィ学会, 第 23 回ファジィシステムシンポジウム講演論文集, pp. 301-304, 2007
- [5] 大沢英一：マルチエージェント環境における交渉のモデル, 人工知能学会誌, Vol.10, No.5, pp.690-696, 1995
- [6] 山村雅幸、宮崎和光、小林重信：エージェントの学習, 人工知能学会誌, Vol.10, No.5, pp.683-689, 1995
- [7] Watts S. Humphrey, ソフトウェア品質研究会訳：“パーソナルソフトウェアプロセス技法”, 共立出版, 1999.
- [8] Watts S. Humphrey: PSP ネットワーク訳”パーソナルソフトウェアプロセス入門”, 共立出版, 2001
- [9] Watts S. Humphrey、秋山義博監訳：JASPIC TSP 研究会訳”PSP ガイドブック ソフトウェアエンジニア自己改善”, 株式会社翔泳社, 2007

連絡先

〒225-8502 神奈川県横浜市青葉区鉄町 1614
桐蔭横浜大学 工学部 電子情報工学科
山口 大輔
TEL: 045-974-5070 FAX: 045-974-5070
E-mail:yamaguti@intl.toin.ac.jp