

ゲーミフィケーションを用いたバグパターンによる 欠陥除去を促進する手法の提案

新井 慧† 坂本 一憲† 鷲崎 弘宜† 深澤 良彰†

早稲田大学 基幹理工学研究科 情報理工学専攻†

1. はじめに

バグパターンとは、様々なプログラムで繰り返し現れる、欠陥を引き起こす可能性のあるソースコード断片をパターン化したものである。ソースコードからバグパターンを検出するツールを利用することで、欠陥の早期発見が可能となり、開発時の欠陥除去コストを削減できる。

欠陥検出ツールによって欠陥を検出したにも関わらず、欠陥を修正せずに放置する場合がある。その原因の一つとして、欠陥の誤検出が挙げられる。現在広く用いられている欠陥検出ツールには、FindBugs[1]や PMD[2]などがあるが、いずれのツールでもある程度の誤検出が発生する。そのため、開発者はツールの検出結果を誤検出として認めてしまいやすくなり、正しい結果を見落としてしまう[3]。また、重大な欠陥に直結しない欠陥についても、修正する必要性が低いことから無視されてしまう。しかし、ソフトウェアの品質を低下させ、長期的に見て欠陥を引き起こす可能性があるため、早期に修正されることが望ましい。

上述した問題を解決するため、欠陥報告の際にゲーミフィケーションを導入した、開発者の欠陥除去に対するモチベーションを向上させる手法を提案する。ゲーミフィケーションとは、ゲームを構成する要素をゲーム以外のものに適用し、利用者のモチベーション向上を図る手法である[4]。欠陥検出ツールによる報告データをもとに、ゲーミフィケーションを利用した、バグパターン解析ツールを提案する。

本論文では、以下の二点を研究課題とする。

RQ1 欠陥検出ツールによって検出された欠陥の情報と修正者を特定できるか？

RQ2 検出された欠陥は、修正情報を提示することで、提示がない場合よりも除去されやすいか？

なお、本論文では、欠陥検出ツールによって示

された「欠陥を引き起こす可能性のある箇所」についても「欠陥」と呼ぶものとする。

2. 提案手法

2.1 提案ツールの構成

提案ツールで利用する欠陥検出は、Java 言語に対応している FindBugs を利用する。FindBugs は、バグパターンを利用した欠陥検出ツールとして代表的である。また、ソースコードのバージョン履歴を利用するため、Git を併用し管理する。Git とは、バージョン管理システムの一つであり、管理対象フォルダにファイル更新(commit)による変更履歴・差分(diff)の情報が保存される。図1にツールの全体像を示す。

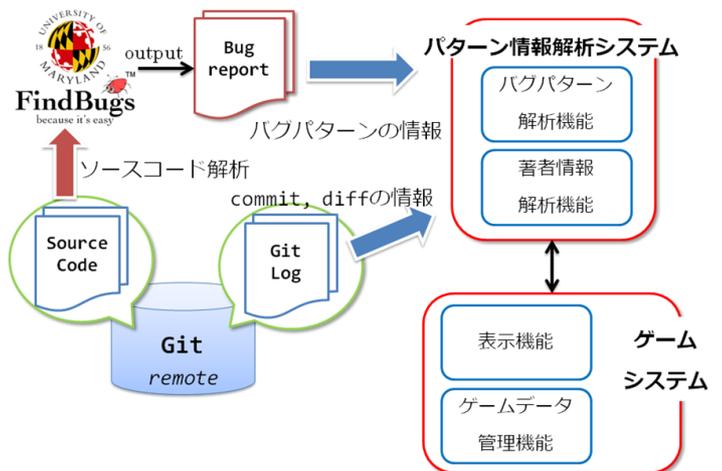


図1 提案ツールの全体像

提案ツールは、バグパターンと著者の情報を集約するバグパターン情報解析システムと、その情報をゲーム化するゲームシステムによって構成されている。

2.2 バグパターン情報解析システム

提案ツールでは、FindBugs によって出力されたデータと FindBugsAPI を利用して、バグパターンの名称、パターンの原因(以下カテゴリ)、優先度、ランクを抽出し、Git log の修正情報から当該コード行の著者、修正者を特定する。

2.3 ゲームシステム

集約されたバグパターンの情報をもとに、パターン報告を行う GUI ベースのアプリケーション

"Facilitate defect removal based on bug pattern using Gamification"

†Satoshi Arai, Kazunori Sakamoto, Hironori Washizaki, Yoshiaki Fukazawa, Waseda Univ.

を構築する。プロジェクト全体や利用するメンバーがそれぞれ修正したバグパターンの数や情報をもとに、ゲーム要素を応用した表現で示し、メンバーの欠陥修正に対するモチベーション向上を図る。

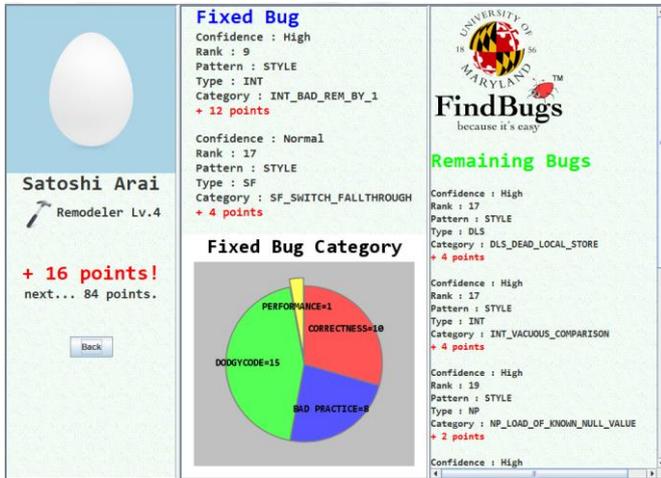


図2 バグパターン報告表示例

2.4 ゲームメカニクスの利用

ゲームメカニクスとは、ゲーミフィケーションによって導入されるゲームの構成要素のことである。例えば、ゲームでよく見られるレベル制度やボーナス点などが該当する。

提案ツールが導入するゲームメカニクスは、Profile completion progression と、それに関する Reward continuous play である[5]。ここでは、ポイント制度とステータスの形で導入する。これらは、利用者内で獲得点数や肩書きでの競争を促し、モチベーションを高める効果が期待できる。まずポイント制度に用いて、欠陥を修正したメンバーに、修正した欠陥の重要度に応じた得点を与える。図2では、各バグパターンの点数と自身が獲得した点数を表示している。ここで、FindBugs のバグパターンにそれぞれ定義されている Rank の情報を利用する。Rank は 0 から 20 まで定義されており、0 に近いほど重要なバグパターンである。獲得点数は $21 - \text{Rank}$ とし、重要な欠陥ほど得点を高く設定する。

次に、ステータスの要素を用いて、修正したバグパターンの種類に応じた肩書きを各メンバーに定義する。前述の通り、FindBugs による報告にはバグパターンの名称、カテゴリ、優先度、ランクの情報がそれぞれ付加してある。獲得したポイントが一定まで溜まった場合、過去に修正したバグパターンのカテゴリに従って、レベルアップと同時に新たな肩書きへ変化するような仕組みを作る。図2では、名前の下に肩書きとレベルを表示している。メンバーそれぞれに違う肩書きが持た

される環境となるため、プロジェクトの管理者からは各メンバーの修正できる欠陥の傾向が読み取れるようになり、メンバーの能力が把握しやすくなる効果も期待できる。また、プロジェクト全体の埋め込み数及び除去欠陥数やカテゴリなどの傾向も表示すれば、管理者は開発過程で特に注視すべき欠陥を想定しやすくなる。また、その想定を考慮することで、開発チームはバグパターンへの意識が高まるため、欠陥の埋め込みを未然に防ぐ効果も期待できる。

3. まとめと展望

本論文では、ゲーミフィケーションを用いたバグパターンによる欠陥検出手法を提案した。FindBugs と Git からバグパターンの情報、ソースコードの変更履歴を抽出し、情報を集約した上で、アプリケーションから情報を表示する。その過程で、ゲーミフィケーションを利用して利用者となる開発メンバーにより多くの欠陥を除去させる仕組みを提案した。

RQ1 本ツールのバグパターン情報解析システムにより、検出された各欠陥の情報と、その著者及び修正者を特定できた。

RQ2 ゲームシステムでゲーミフィケーションの要素を用いて修正情報を提示することで、欠陥の除去促進効果が期待できる。

今後の展望として、欠陥報告の過程で、より適切で理解しやすい数値を用いて評価できないか検討したい。その上で、提案ツールを用いた評価実験を行い、欠陥除去に対するゲーミフィケーションの有用性を示したい。また、その他のゲームメカニクスを取り入れることができるか、またどのように取り入れるべきかを考え、欠陥除去工程とゲーミフィケーションの関係性についても考察したい。

4. 参考文献

- [1] D. Hovemeyer et al., FindBugs™ <http://findbugs.sourceforge.net/>.
- [2] T. Copeland et al., PMD, <http://pmd.sourceforge.net/>.
- [3] S. Deterding, D. Dixon, et al., "From Game Design Elements to Gamefulness: Defining 'Gamification'", In Proceedings of the 15th International Academic MindTrek Conference (MindTrek'11), pp. 9-15, 2011.
- [4] M. G. Nanda, et al., "Making Defect-Finding Tools Work for You", 32nd International Conference on Software Engineering (ICSE '10), pp. 99-108, 2010.
- [5] J. Hamari, A. Järvinen, "Building Customer Relationship through Game Mechanics in Social Games", IGI Global, 2011.