

修士論文要旨

開放環境科学 専攻	学籍番号 80227265	氏名 山崎 誠也	やまざき せいや 山崎 誠也
(論文題目)			
AspectJ Programs を対象とした静的解析によるデバッグ支援			
(内容の要旨)			
<p>ソフトウェア開発においてバグの発生を防止することはできない。そこで、そのバグを少なくするために、トレース(特定の関数が呼び出されたり、特定の文が実行されたりする様子の監視)やブレークポイント(処理を一時停止するポイント)の設定という機能を持つデバッガが登場している。しかし、新技術に対してデバッガが対応するまでには、時間がかかる。</p> <p>例えば、AspectJ の有用性については数多く報告されているが、同時に、aspect(様々な部分に適用される処理)は新たな複雑さをプログラムに導入している。aspect はオブジェクトと独立して記述されるため、開発者がオブジェクトのすべての振る舞いを把握することが難しい。なぜなら、AspectJ は、実行時にプログラムの順番が決定されるため、トレースやブレークポイントの設定をしても AspectJ のソースコードとの対応付けにくい。そこで、AspectJ 独自のデバッガが必要になる。特に Aspect 指向固有のバグとして、aspect から core(本質的な Java 処理)に与える影響がある。これは、本来 core と aspect は、独立性・保守性・再利用性を高くしておくのが理想である。しかし、aspect 側から core の private 変数の値を変更してしまうと、aspect と core の独立性が低くなり、切り離すことができなくなる。だから、aspect と core の独立性を保つために、いつでも aspect 側から core の private 変数の値を変更してしまう個所を発見することが必要である。</p> <p>そこで、本研究では、core に影響を与える個所として</p> <ol style="list-style-type: none">1. privileged aspect による private へのアクセス2. aspect による core の private 変数を変更するメソッド実行 <p>の二つを取り上げる。1.において privileged aspect は、本来アクセスできないはずである他クラスの private 変数にアクセスし、変更可能である。これは、カプセル化を破壊する行為で後にバグになる可能性が高いのでユーザーに提示した。また、2.において、直接 private 変数を変更するメソッドを呼び出す時は、明らかに private 変数を変更している。さらに、他のクラスにある他のメソッドを呼び出してもその呼び出されたメソッドが、また他のメソッドを呼び出し、最終的に private 変数を変更している場合も private 変数を変更している。だから、本研究では、この場合も対応し、呼び出し関係を追跡して、途中経過もユーザーに提示した。</p> <p>本研究によって、aspect が core に与える影響を提示することでユーザーの負担を軽減できることが示された。</p>			

慶應義塾大学大学院理工学研究科 前期博士課程
(内容の要旨は約 25 行程度で記入のこと)