

**科学研究費助成事業 研究成果報告書**

平成 28 年 5 月 16 日現在

機関番号：17601

研究種目：基盤研究(C) (一般)

研究期間：2012～2015

課題番号：24500044

研究課題名(和文) 高品質なソフトウェア開発のためのテスト可視化手法の提案

研究課題名(英文) Proposal of visualization methods of software testing for high quality software development

研究代表者

片山 徹郎 (Tetsuro, Katayama)

宮崎大学・工学部・准教授

研究者番号：50283932

交付決定額(研究期間全体)：(直接経費) 4,100,000円

研究成果の概要(和文)：本研究は、「テスト実施前のプログラムの振る舞いに関する情報」および「テスト実施中のテスト実施状況に関する情報」をプログラマに提示する、ソフトウェアテストの可視化手法を提案し、評価する。その実現のため、次のことを行った。(1)結合テスト支援ツールAvisの拡張により、プログラムを可視化し、プログラム開発者にとって有用な情報を提示。(2)Javaプログラムを対象としたテスト自動実行ツールJvisの拡張により、プログラムの適用範囲の拡大。(3)テストケースを図で表現可能かどうかについて検討し、検討結果に基づいたテストケースの可視化。

研究成果の概要(英文)：This research proposes and evaluates a method to visualize software testing for programmers from information: information about behavior of a program before testing and information about status of testing during testing. The research has been achieved the followings. (1) By extending integration testing support tool Avis, it visualizes a program and shows useful information for program developers. (2) By extending automatic test execution tool for Java programs Jvis, it can be applied more complicated Java programs to. (3) A method to express test cases in a diagram is investigated, and then they can be visualized based on the result.

研究分野：ソフトウェアテスト

キーワード：ソフトウェアテスト ソフトウェア品質 テストケース テストケースの可視化 テスト実行時の可視化 UML 単体テスト

## 1. 研究開始当初の背景

### (1) ソフトウェア開発を取り巻く背景

ソフトウェアは、IT産業のみならず製造業から金融業に至るまで、あらゆる産業の付加価値創出の源泉となっており、もはや現代経済社会の基盤として欠かせない存在となっている。品質が低いソフトウェアによる、社会的損害がたびたび報告されており、その中には、社会基盤を揺るがしかねなかったり、人命が危機にさらされたりする事例も報告されている。組込みソフトウェアの分野においては、経済産業省の組込みソフトウェア産業実態調査があり、製品の不具合原因の半分近く(58.8%)をソフトウェアが占めており、ソフトウェア開発において品質向上の課題が最も重要視(64%)されている。すなわち、産業界において高品質なソフトウェア開発手法は急務である。

ソフトウェアに限らず、日本の製造業は、かねてから製品の品質を非常に重要視してきた。その理由は、日本人には、ものづくりの原点を品質の向上に置くという気質があり、ユーザとしても、品質に対して厳しい目を持つという特徴があるからである。つまり、ソフトウェア開発は、品質に対する考え方が厳しい日本が国際的に優位に立てる分野である。一方、現在のソフトウェア産業は、日本に技術はあると世界中が認めているものの、実際には、アメリカや、ヨーロッパ、インドなどが、牽引している状況がある。ソフトウェアの品質向上に関する研究を進めることは、日本のソフトウェア産業の巻き返しにつながり、我が国の経済発展のために最優先で検討すべきことである。

### (2) 研究代表者の動機

これまで研究代表者は、ソフトウェアテストに関する様々な活動を積極的に務めており、その場での議論の中で、ソフトウェアの設計分野と比較すると、テスト分野は可視化(見える化)がほとんど進んでいない、という考えに至った。そこで、ソフトウェアテスト技法および可視化手法の観点から、Javaプログラムを対象としたソフトウェアの信頼性向上手法について主に研究を進めてきた。具体的には、以下を行っていた。

プログラム自動可視化ツール Avis を開発し、結合テストに焦点を当てたテスト支援ツールへ拡張し、結合テスト実施前にプログラムの実行経路を可視化した。

テスト前にテストコードの作成を必要としないランダムテスト法を用いたテスト自動実行ツールを試作し、テストの実施状況を可視化する機能を持たせた。

以上から、今までの研究成果や知識を社会に還元するという点において、研究代表者自身が最も貢献可能で、かつ、その効果が高い

分野は何かと考えた際に、高品質なソフトウェア開発のためのソフトウェアテスト手法の可視化に関する研究を進めることである、との着想に至った。

## 2. 研究の目的

本研究の目的は、高品質なソフトウェア開発を行うために、テスト手法の可視化を行い、高度情報化社会および安全・安心を第一とする社会における情報技術への更なる要求に、迅速に対応できる基盤を与えることにある。具体的には、「テスト実施前のプログラムの振る舞いに関する情報」および「テスト実施中のテスト実施状況に関する情報」をプログラムに提示する、ソフトウェアテストの可視化手法を提案し、評価する。

従来から、ソフトウェアの品質向上手法については様々な手法が提案されてきた。しかし、そのほとんどはソフトウェアの要求分析や設計に関しての手法である。本研究は、テスト手法を中心に研究を進める点が大きな特徴である。テスト作業は、ソフトウェアを出荷する直前の最終作業であり、ソフトウェアの品質を決定付ける最後の砦である。このテスト作業を支援することにより、高品質なソフトウェア開発を目指す。

可視化については、要求分析や設計の際に、システムへの要求や、プログラムの構造を理解させるといった、プログラムにプログラミングの際の情報を与えるための手法が多い。本研究は、テスト支援のために、テスト実施前およびテスト実施中に対して可視化を行なう点が独創的である。記述したプログラムの振る舞いに関する情報を、様々な形でプログラムに早期に提示し、テストを実施する際に役立つ情報を提供する。またテスト実施中に、テスト実施状況に関する情報を提示し、テスト作業そのものための可視化を行い、テスト作業を支援する。すなわち、ソフトウェアテストの可視化手法を提案し、テスト作業を支援することによって、高品質なソフトウェア開発を目指す。

## 3. 研究の方法

本研究では、高品質なソフトウェア開発を実現するためのテスト手法の可視化手法を提案し、評価する。その実現のため、以下の項目を実施する。

- ・結合テスト支援ツール Avis の拡張により、プログラムを可視化し、プログラム開発者にとって有用な情報を提示する。
- ・Java プログラムを対象としたテスト自動実行ツールの拡張により、プログラムの適用範囲を広げる。
- ・テストケースを図で表現可能かどうかについて検討し、検討結果に基づきテストケー

スの可視化を図る。

以下、それぞれについて述べる。

### (1) 結合テスト支援ツール Avis の拡張

Java プログラムを入力として、コントロールフローグラフ、逐次型実行経路、および、モジュール間情報を含めた実行経路を自動生成するツール Avis を開発し、結合テストに焦点を当てたテスト支援ツールへ拡張した。Avis を適用することにより、結合テスト実施前にプログラムの実行経路をプログラマに提示でき、テスト実施の指針を立てることができ、テスト実施にかかる時間を見積もることができる。

そこで、引き続き Avis の拡張を進め、プログラムの振舞いに関する情報から、プログラマ開発者にとって役立つ情報を可視化できるように Avis を改良する。また、Avis を実際に使用してもらい、その使い勝手などを調査するとともに、Avis への要望についても調査し、今後の拡張方針を決定する際に利用する。

### (2) テスト自動実行ツールの適用範囲の拡大

Java プログラムを入力として、テスト実施前にテストコードの作成を必要としないランダムテスト法を用いた自動テスト実施手法を提案するとともに、ツールを試作した。試作したツールは、ツールの実行中にテスト実施の進捗状況として、ソースコード全体のうち、どの程度のコード(行数)をテストで実行済なのかをリアルタイムに確認できるという特徴を持つ。この提案手法の有効性を示すことはできたのだが、一部の文法や変数型にしか対応しておらず適用できるプログラムに制限がある。

そこで、試作したテスト自動実行ツールの実用性を高めるために、適用範囲の拡大を目的とした拡張を行う。具体的には、現在未対応の文法や変数型に対応するとともに、拡張後のツールの有用性を評価する。

### (3) テストケースの可視化に関する検討

テスト技法に関する研究は、近年多く進められつつあるが、可視化(見える化)はほとんど行われていない。ソフトウェア開発プロセスにおける要求分析や設計においては、例えば UML(Unified Modeling Language)を用いたシステムの要求や構造、振舞いを表現する方法が提案され、その普及が進んでいるが、テストを可視化する試みは少ない。

そこで、まずテストケースの可視化について検討を行う。具体的には、従来自然言語で記述されるテストケースについて、図を使った表現が可能かどうか、また、図を使うこと

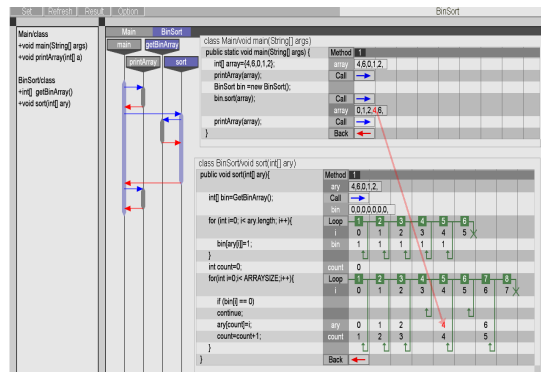


図 1. TFVIS の外観

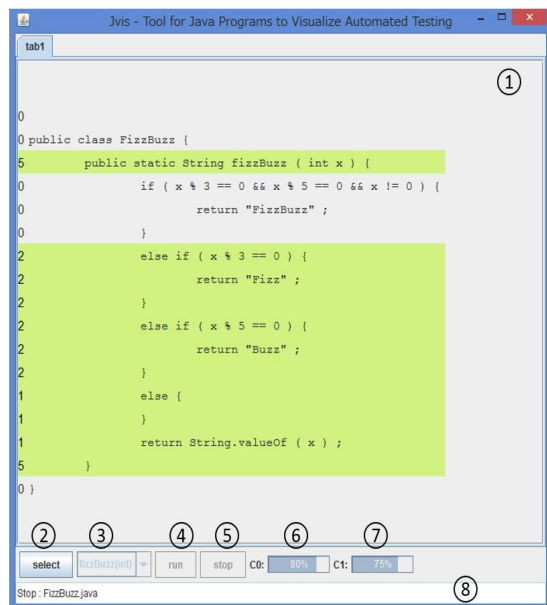


図 2. Jvis の外観

によって全体を俯瞰することが可能かどうかを検討する。検討結果に基づき、テストケースの可視化手法を具体化する。

## 4. 研究成果

### (1) 研究の主な成果

#### 結合テスト支援ツール Avis の拡張

結合テスト支援ツール Avis の機能拡張として、Avis にデバッグ機能を実装した TFVIS(transitions and flow visualization)を開発した。図 1 に、TFVIS の外観を示す。TFVIS は、プログラム実行時の各変数の状態や更新といったデータ遷移を可視化することによって、プログラムの挙動の把握を支援し、デバッグ作業を容易にする。TFVIS により、プログラマはバグを含んだプログラムの特異な挙動を一目で判断することが可能となり、Java プログラムでのバグの原因特定に要する時間を 46%減らすことができた。

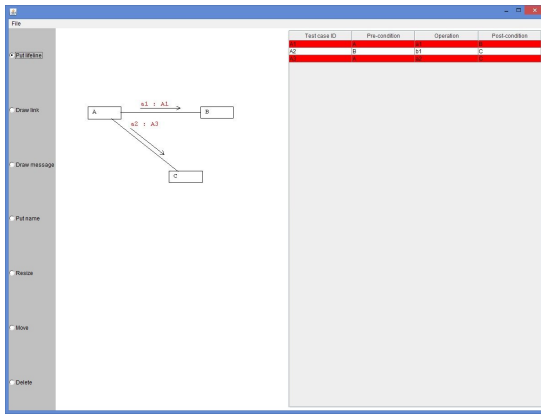


図 3. テスト用コミュニケーション図作成支援ツールの外観

また、Avis に実装した可視化のアイデアを、分散システムやマルチスレッドプログラムについて適用可能かを検討し、その検討結果に基づいて、適用手法を提案した。

#### テスト自動実行ツールの適用範囲の拡大

既に試作済みの、Java プログラムを対象とした、テスト実施前にテストコードの作成を必要としないランダムテスト法を用いた単体テスト実行ツールを、Jvis(Tool for Java programs to visualize testing)と名付けた。図 2 に、Jvis の外観を示す。この Jvis の適用範囲の拡大として、具体的に以下のことを行った。

- ・ユーザからの入力が必要な読み込みメソッドが存在するプログラムへの対応。
- ・(実装途中などの理由で)未完成なメソッドが存在するプログラムへの対応。
- ・テスト対象メソッドの引数がオブジェクト型であるプログラムへの対応。

以上について、ツール Jvis に実装して、その有用性を評価した。

また、Jvis を Web アプリケーションとして実現することについて検討し、一部を実装した。

#### テストケースの可視化に関する検討

まず、従来自然言語で記述されるテストケースについて、図を使った表現が可能かどうかについて検討した。次に、検討結果に基づいて、ソフトウェア開発プロセスにおいて、要求分析や設計で用いられる UML(Unified Modeling Language)をベースとした、テスト用コミュニケーション図とテスト用状態マシン図を新たに提案した。これらの図を用いることによって、テストケースの全体像を俯瞰でき、かつ、テストケースにおける漏れや不備、重複などを防止できることを示した。

また、テスト用コミュニケーション図については、作成支援ツールを試作し、その有用

性について評価した。図 3 に、試作したツールの外観を示す。

#### (2) 得られた成果の国内外における位置づけとインパクト

本研究課題は、下記の 3 つの項目を設けた上で、それぞれの項目に取り組んできた。

- ・結合テスト支援ツール Avis の拡張
- ・テスト自動実行ツール(Jvis)の適用範囲の拡大
- ・テストケースの可視化に関する検討

現時点ですべての項目において、ほぼ当初の計画通りに研究が進んだ。研究成果については、都度とりまとめ、査読付き雑誌論文への掲載や、国際会議および学会での発表を行っており、社会的に認知され評価されたと判断できる。

また、今回の研究実績を含めて、ソフトウェアのテストから見たソフトウェアの品質保証のあり方について、各種セミナーや講演会にて招待講演を行っている。

#### (3) 今後の展望

高品質なソフトウェア開発を実現するために、本研究課題にて実施した各種テスト可視化手法を引き続き改良することによって、テスト工程の支援を行う。また同時に、手法の応用として、ソフトウェア開発プロセスの他の工程に適用し、安全安心を第一とする社会における情報技術への更なる要求に、迅速に対応できる基盤を与える。

研究代表者は、ソフトウェアテストに関する様々な活動を積極的に務めている。その場での議論の中で、この研究内容についても議論しており、テスト分野での可視化が進めば、その技術は要求分析や設計、デバッグや運用保守にも効果が期待できることを確信している。

#### 5 . 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

〔雑誌論文〕(計 20 件)

T. Katayama, H. Nakamura, Y. Kita, H. Yamaba, K. Aburada, and N. Okazaki: ``TFVIS: a Supporting Debugging Tool for Java Programs by Visualizing Data Transitions and Execution Flows,`` Journal of Robotics, Networking and Artificial Life (JRNAL), Vol.2, issue 3, pp.177-181, doi:10.2991/jrnal.2015.2.3.9, 2015 (査読有)。

M. C. Saputra, T. Katayama, H. Suyono, and A. Basuki: ``A Web-Based Software

Testing Tool with Visualization for Java Programs,' ' 宮崎大学工学部紀要, No.44, pp.209-220, 2015 (査読無).

M. C. Saputra and T. Katayama: ``Code Coverage Visualization on Web-Based Testing Tool for Java Programs,' ' Journal of Robotics, Networking and Artificial Life (JRNAL), Vol.2, issue 2, pp.89-93, doi:10.2991/jrnal.2015.2.2.5, 2015 (査読有).

T. Katayama, S. Urata, Y. Ogata, Y. Kita, H. Yamaba, K. Aburada, and N. Okazaki: ``Prototype of a Supporting Tool to Generate Testing Communication Diagram,' ' Journal of Robotics, Networking and Artificial Life (JRNAL), Vol.2, issue 1, pp.9-12, doi:10.2991/jrnal.2015.2.1.3, 2015 (査読有).

T. Katayama, S. Kitano, Y. Kita, H. Yamaba, K. Aburada, and N. Okazaki: ``Proposal of a Testing Method Using Similarity of Interleaving for Java Multi-threaded Programs,' ' Journal of Robotics, Networking and Artificial Life (JRNAL), Vol.1, issue 4, pp.299-302, doi:10.2991/jrnal.2015.1.4.10, 2015 (査読有).

T. Katayama, S. Kitano, Y. Kita, H. Yamaba, and N. Okazaki: ``Proposal of a Supporting Method for Debugging to Reproduce Java Multi-threaded Programs by Petri-net,' ' Journal of Robotics, Networking and Artificial Life (JRNAL), Vol.1, issue 3, pp.207-211, doi:10.2991/jrnal.2014.1.3.8, 2014 (査読有).

北野翔一郎, 片山徹郎: ``ペトリネットを用いた Java マルチスレッドプログラムの実行を再現することによるデバッグ支援手法の提案,' ' 宮崎大学工学部紀要, No.43, pp.243-248, 2014 (査読無).

中村紘人, 片山徹郎: ``データ遷移の可視化手法によるバグの原因特定支援について,' ' 宮崎大学工学部紀要, No.43, pp.249-256, 2014 (査読無).

中村紘人, 片山徹郎, 喜多義弘, 山場久昭, 岡崎直宣: ``Java プログラム実行時のデータ遷移可視化によるデバッグ支援,' ' 情報処理学会 ソフトウェアエンジニアリングシンポジウム 2014 (SES2014), pp.125-130, 2014 (査読有).

T. Katayama, H. Nakamura, Y. Kita, H. Yamaba, and N. Okazaki: ``Proposal of a Visualizing Method of Data

Transitions to Support Debugging for Java Programs,' ' Journal of Robotics, Networking and Artificial Life (JRNAL), Vol.1, issue 2, pp.111-115, doi:10.2991/jrnal.2014.1.2.3, 2014 (査読有).

T. Katayama, Z. Zhao, Y. Kita, H. Yamaba, and N. Okazaki: ``Proposal of a Method to Build Markov Chain Usage Model from UML Diagrams for Communication Delay Testing in Distributed Systems,' ' Journal of Robotics, Networking and Artificial Life (JRNAL), Vol.1, issue 2, pp.120-124, doi:10.2991/jrnal.2014.1.2.5, 2014 (査読有).

片山徹郎: ``単体ランダムテスト実行/可視化ツール“Jvis”の適用事例,' ' IPA 独立行政法人情報処理推進機構, 先進的な設計・検証技術の適用事例報告書 2013年度版, PART III 検証事例, pp.111-63 - 111-82. 2014 (査読無).

浦田聖也, 片山徹郎: ``テストケースの可視化を目的としたテスト用ダイアグラムの提案,' ' 宮崎大学工学部紀要, No.42, pp.263-269, 2013 (査読無).

浦田聖也, 片山徹郎: ``テストケースの可視化を実現するテスト用ダイアグラムの提案,' ' 情報処理学会 ソフトウェアエンジニアリングシンポジウム 2013 (SES2013) (電子データ配布), 2013 (査読有).

S. Urata and T. Katayama: ``Proposal of Testing Diagrams for Visualizing Test Cases,' ' Proc. 6th Int'l Conf. on Software Testing, Verification and Validation (ICST2013) (電子データ配布), 2013 (査読有).

Y. Nishi, T. Katayama, and S. Yoshizawa: ``Combinatorial Test Architecture Design Using Viewpoint Diagram,' ' Proc. 2nd Int'l Works. on Combinatorial Testing (IWCT 2013) (電子データ配布), 2013 (査読有).

喜多義弘, 片山徹郎, 富田重幸: ``Java プログラム読解支援のためのプログラム可視化ツール Avis の実装と評価,' ' 電子情報通信学会論文誌, Vol. J95-D, No.4, pp.855-869, 2012 (査読有).

松岡慎吾, 喜多義弘, 片山徹郎: ``Java プログラムを対象とした単体テスト自動実行および可視化ツール“Jvis”の開発,' ' 宮崎大学工学部紀要, No.41, pp.261-268, 2012 (査読無).

浦田聖也, 片山徹郎: ``テストケースの可視化を目的としたテスト用状態マシン図作成手法の提案,' ' ソフトウェアテストシンポジウム 2012九州 (JaSST'12 Kyushu), pp.30-33, 2012 (査読有).

松岡慎吾, 喜多義弘, 片山徹郎: ``Java プログラムを対象とした単体テスト可視化ツール`Jvis`の開発,'`情報処理学会 ソフトウェアエンジニアリング シンポジウム 2012(SES2012), IPSJ-SES2012017 (電子データ配布), 2012 (査読有).

[学会発表](計8件)

片山徹郎, 日下部茂, 栗田太郎: ``形式仕様記述からテスト設計してみた,'`ソフトウェアテストシンポジウム 2015 東京(JaSST'15 Tokyo), p.84, 2015年2月20日, 東洋大学(東京都・文京区).  
安達賢二, Stuart Reid, 片山徹郎, 佐々木方規, 中野直樹, 西康晴: ``テストエンジニアの育成による組織力・チーム力の向上 ~現場が幸せになる育成とは?また、エンジニア自身が成長するためには~,``ソフトウェアテストシンポジウム 2014 東京(JaSST'14 Tokyo), パネルディスカッション, 2014年3月8日, 東洋大学(東京都・文京区).

北野翔一郎, 片山徹郎: ``Java マルチスレッドプログラム向けの拡張ペトリネットを用いた実行の再現を利用したデバッグ支援ツールの試作,'`情報処理学会 SE 研究会, 情報処理学会研究報告, Vol.2014-SE-185, No.23, pp.1-8, 2014年7月9日, 富良野文化会館(北海道・富良野市).

趙治家, 片山徹郎: ``分散システムの通信遅延をテストするためのUMLモデルからMarkov連鎖使用モデルの構築手法,'`平成25年度電気関係学会九州支部連合大会, 06-2A-11 (p.379), 2013年9月25日, 熊本大学(熊本県・熊本市).

中村紘人, 片山徹郎: ``Javaプログラムのデバッグ支援を目的とした実行時のデータ遷移の可視化について,'`平成25年度電気関係学会九州支部連合大会, 06-2A-12 (p.380), 2013年9月25日, 熊本大学(熊本県・熊本市).

松岡慎吾, 片山徹郎: ``テスト実施状況のリアルタイムな可視化を行う単体テスト可視化ツール`Jvis`の開発,'`電子情報通信学会ソフトウェアサイエンス研究会, 信学技報 Vol.112, No.373, SS2012-52, pp.37-42, 2013年1月10日, 石垣市民会館(沖縄県・石垣市).

浦田聖也, 片山徹郎: ``ソフトウェアの信頼性向上を目的としたテスト用コミュニケーション図作成手法の提案,'`平成24年度電気関係学会九州支部連合大会, 05-2P-01 (p.534), 2012年9月25日, 長崎大学(長崎県・長崎市).

松岡慎吾, 喜多義弘, 片山徹郎: ``リアルタイムな可視化を実現する単体テスト可視化ツール`Jvis`の開発,'`情報処理学会九州支部 若手の会セミナー

2012, pp.7-12, 2012年9月7日, 国民宿舎虹の松原ホテル(佐賀県・唐津市).

6. 研究組織

(1) 研究代表者

片山 徹郎 (KATAYAMA, Tetsuro)

宮崎大学・工学部・准教授

研究者番号: 50283932