

7A-6

## ソフトウェアテストのための テストデータの自動生成

氏名 戸堀 賢治<sup>†</sup> 古澤 康一<sup>†</sup> 大橋 一史<sup>†</sup>  
所属 <sup>†</sup>三菱電機株式会社 先端技術総合研究所

近年、ソフトウェア(以下 S/W とする)の開発は、目まぐるしく変わるユーザのニーズに対応するため、開発速度の向上が求められている。S/W 開発におけるテスト作業は、設計やプログラミングと同程度か、それ以上にコストが掛かる工程であり、テスト作業の効率化によって開発期間を短縮できる。テストに必要なテストドライバ、スタブ等のモジュールの生成作業、評価作業等については、様々な自動化が検討されてきた。しかし、ブラックボックステストに使用するテストデータは、それぞれの開発対象 S/W の仕様によって異なるため、自動的に生成することは困難である。本稿では、S/W のバージョンアップ開発におけるテストデータの自動生成方法について述べる。

### 1. はじめに

S/W 開発において、テスト作業は、設計やプログラミング以上にコストが掛かる工程であり、テスト作業の効率化によって開発期間を短縮できる。そのため、テストの自動化、バグ発生時の再現の効率化等を考慮した S/W テスト環境を整備することは、非常に有効である[1]。

S/W の開発では様々なテストを実施するが、プログラムの動作の確認には、ブラックボックステストが通常用いられている[2]。現在開発中の S/W テスト環境では、このブラックボックステストを対象としており、テストの前準備、テストおよび評価の作業の自動・半自動化を行っている。本稿では、テスト対象とするアプリケーション内のモジュールをバージョンアップした場合のテストに必要なテストデータの自動生成方法について述べる。

### 2. S/W テスト環境

現在、テストに必要なテストドライバ・スタブの生成、テストの実行、評価作業、テストに必要なデータの生成を自動化する S/W テスト環境を開発している。本 S/W テスト環境では、一部のモジュールがリファクタリングやアルゴリズム変更等で新しいバージョンとなったアプリケーション(以下、新バージョンとし、バージョンアップ前のアプリケーションを旧バージョンと呼ぶ)内のバージョンアップしたモジュールのブラックボックステストを想定している(図 1)。今回、GUI モジュールはテスト対象に含まないものとする。

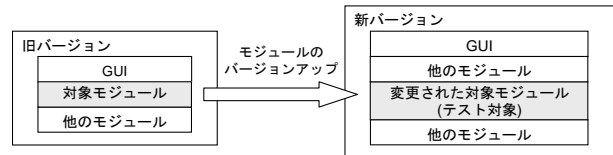


図 1. テスト対象となる事例

テスト対象となるモジュールのソースコードやリンクエラー出力からテストドライバやスタブを自動生成する。テスト対象の関数およびその引数値、実行する関数の順序等のテスト仕様を定義したデータ(以下、テスト入力データとする)を用いてテストを行う。そのテスト結果の評価は、テスト対象の関数処理後の引数値や戻り値等の関数処理結果(以下、ログデータとする)と、ログデータの期待値(以下、比較参照データとする)の比較により行う(図 2)。

S/W テスト環境で必要となるテストデータは、テスト入力データおよび比較参照データである。これらのテストデータは、テスト対象のモジュールにより異なるため、基本的にはユーザの手作業により作成される。しかし、テストの規模が大きくなるにつれ、テストデータの作成量も大きくなり、ユーザの負担および人為的ミスが増加する。

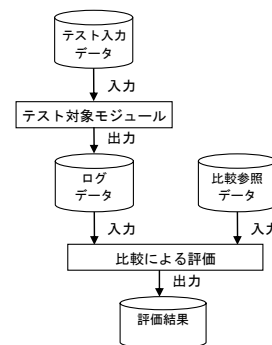


図 2. 本 S/W テスト環境によるテスト概要

Automatic Generation of Test Data for Software Testing  
<sup>†</sup>Kenji Tobori, <sup>†</sup>Koichi Furusawa, <sup>†</sup>Kazushi Ohashi  
Advanced Technology R&D Center, Mitsubishi Electric Corporation (<sup>†</sup>)

### 3. テストデータの自動生成

ここでは、テスト入力データおよび比較参照データの自動生成方法について述べる。テスト入力データおよび比較参照データの生成には、旧バージョンの関数名、戻り値の型、戻り値、引数の型および引数値(以下、関数情報とする)が必要である。そのため、呼び出し元の関数の処理の先頭と末尾で関数情報を取得する機能(以下、ログ出力機能とする)を用いて実現する。

#### 3.1 テストデータの生成方法

テストデータを生成するための手順は、以下のとおりである。

関数呼び出しの際に、ログ出力機能は、スタック上に格納された情報を解析することにより、呼び出し元の関数情報を取得する。

その取得した関数情報を出力することにより、テストデータを生成する。

このようなログ出力機能によるテストデータの生成には、出力対象となる関数からログ出力機能を読み出す必要がある。ログ出力機能の読み出しの実装は、自動的に、かつ、ソースコードの改修なしで、実現できるプロファイル機能を利用する。プロファイル機能とは、コンパイラが提供している機能であり、コンパイル時に対象となるモジュールの全関数の関数処理の先頭と末尾に特定の関数(以下、プロファイル関数とする)を挿入する機能である。このプロファイル関数内にログ出力機能の読み出しを実装することで、旧バージョンからログ出力機能を読み出すことが可能である。このような機能を用いて、本 S/W テスト環境では、テストデータを自動生成する。

#### 3.2 関数仕様変更への対応

ログ出力機能内で取得した関数情報は、バージョンアップの際に、関数仕様が変わらず、内部仕様のみが変わった関数については、旧バージョンと新バージョンの動作結果は全く等しくなるので、旧バージョンの関数処理前の関数情報を新バージョンのテスト入力データとして、旧バージョンの関数処理後の関数情報を新バージョンの比較参照データとして、そのまま使用することができる。

しかし、実際には、バージョンアップの際に関数仕様変更されている場合も少なくない。このような場合、旧バージョンの関数処理の前後で取得した関数情報をそのまま新バージョンのテストデータとして使用することは不可能である。そのため、新バージョンのテストのため

に、データを変換する必要がある。その場合、旧バージョンから新バージョンへの関数名、引数の並び、引数の削除、引数の追加等の関数情報の変更方法を定義したマッピングデータを用いてログ出力機能が出力する関数情報を変換する。これにより、テスト入力データおよび比較参照データを自動生成することができる(図3)。

このデータ自動生成方法を使用すれば、テストデータを作成するために、テスト環境や対象モジュールの独自の仕様や操作方法を習得する必要が少なく、ログデータを出力する関数を指定しさえすれば、旧バージョンのユーザ・インタフェースへの入力操作(例えば、GUI 操作)により、テストデータを生成できる。新バージョンのテストでは、それらのデータをそのまま使用することが可能である。

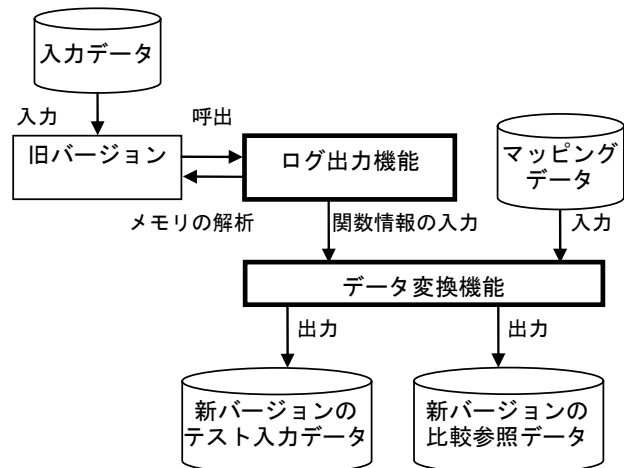


図3.テストデータ自動生成システムの概要

### 4. おわりに

本稿では、テスト作業において、ブラックボックステストのテストデータを自動生成する方法について述べた。

本稿の自動生成では、変更前の S/W を改修することなく、比較的容易な操作でテストデータを作成できるという特徴を持っており、テストの効率化、テスト品質の向上だけでなく、導入期間の短縮といった効果も期待できる。

#### 参考文献

- [1] 辻宏郷, 佐藤, 勝山, 水野, 曾我, “形式手法による通信ソフトウェア試験データの生成とその試験方法,” 情報処理学会論文誌, Vol.34, No.6, pp.1347-1361, 1993.
- [2] Myers, G.J., “*The Art of Software Testing*,” John Wiley & Sons, 1979.