

ソフトウェアのテスト工数・期間を削減するためのシステムテスト自動化技術

System Test Automation Techniques for Software Development Capable of Reducing Testing Worker Hours and Shortening Testing Periods

中野 隆司 NAKANO Takashi 田中 裕大 TANAKA Yuta ダン ティ ホン イエン DANG Thi Hong Yen

製品のソフトウェア開発では、新規開発だけでなく、既存のソフトウェアを改造する派生開発も数多く行われている。派生開発は、既存のソフトウェア資産を活用できるというメリットがある一方で、品質を確保するためのテスト工数・期間は削減できていないという課題が存在する。

東芝グループは、この課題に対応する施策として、(1)システムテスト自動化の導入・推進ガイドの開発や、(2)システムテスト自動化プロセスの整備、(3)海外リソースを活用したシステムテスト自動化体制の整備、(4)テスト仕様とテストスクリプトを一元管理する環境の整備、(5)テストスクリプト技術の開発など、システムテスト自動化技術の開発と導入を推進した。今回、社会インフラ系の大規模システムに適用した結果、次機種以降におけるシステムテストのコストを30%、自動化の初期導入コストを40%、それぞれ削減可能であることを確認した。

In addition to the development of new software for installation in various products, the creation of derivation software utilizing existing software accumulated as development assets has been increasing. This approach, referred to as derivation development, is effective for the refurbishment of existing software. However, a large number of test cases need to be examined in order to assure the quality of such software for a new product, making it difficult to reduce testing worker hours and shorten testing periods.

The Toshiba Group has been implementing a broad array of measures to rectify this situation through the development and introduction of system test automation techniques, as follows: (1) preparation of guidelines for the introduction and promotion of test automation, (2) development of test automation processes, (3) establishment of a test automation organization utilizing overseas resources, (4) maintenance of a unified management environment for test specifications and test scripts, and (5) development of a technique to generate test scripts with easy maintenance. We have conducted simulation experiments applying these techniques to a large-scale social infrastructure system and confirmed that the techniques make it possible to achieve a 30% reduction in the system testing costs of subsequent systems and a 40% reduction in initial introduction costs.

1. まえがき

近年、製品の機能や価値がシステムやデジタル情報処理によって提供されるようになり、ソフトウェアの開発規模が飛躍的に増大している。ソフトウェア開発の現場では、製品出荷後の不具合発生を未然に防ぐため、デバッグやテストの重要性が一層高まっている。ソフトウェア開発における結合テストやシステムテストの期間と工数は、テスト全体の約40%にも達すると言われている¹⁾。また、IT(情報技術)の進化と多様化、ビジネスや技術革新のスピードの加速に伴って、大量のテストを短期間で行う必要性に迫られている。

製品のソフトウェア開発では、新規開発だけでなく、既存のソフトウェアを改造する派生開発も数多く行われている。派生開発におけるテスト工数の割合を図1に示す。派生開

発では、新規開発に比べて要求分析・設計・実装などに掛かるコスト、並びに要求定義から単体テストまでのテスト工数は減少する傾向がある。しかし、ソフトウェア開発の場合、ある機能への変更が別の機能にまで影響を及ぼしてしまうことが多い。また、変更の影響範囲をあらかじめ正確に把握することは難しいため、多くの場合、テスト工程において変更に関係のない機能も含めて多くのテストを実施することで、品質を確保している。このため、テスト工程に掛ける時間を大きく削減することはできず、開発全体に占めるテスト工程の割合が大きくなる傾向にある。仕様の追加や変更の規模が小さい開発ほど、テスト工程の工数が開発全体のコスト増につながるという悪循環を生じている。

テスト工数の増大に伴う課題としては、①繰り返し行うテストによる工数の増大、②テストの抜け漏れによる後戻り工数

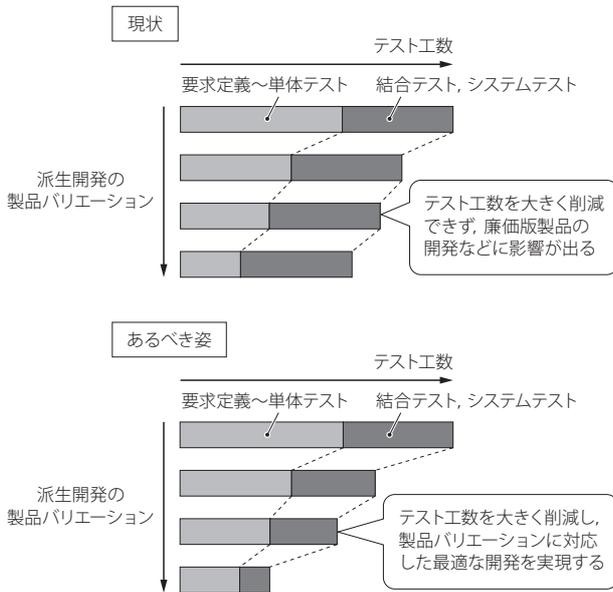


図1. 派生開発におけるテスト工数の割合

派生開発では、要求定義から単体テストまでの工数は再利用によって削減できるが、結合テストやシステムテストの工数は削減できていない。

Comparison of ratios of testing costs in current and ideal derivation development

の増大、が挙げられる。そこで東芝グループは、これらの課題を解決するために、システム全体が予定どおりの機能を満たしているかどうかを検証するシステムテスト自動化技術を開発した。ここでは、東芝グループで行ったシステムテスト自動化の取り組み、及びその適用効果について述べる。

2. システムテスト自動化の課題と施策

システムテスト自動化のプロセスを図2に示す。ステップ1では、1章で述べた①の課題を解決するために、既存のテ

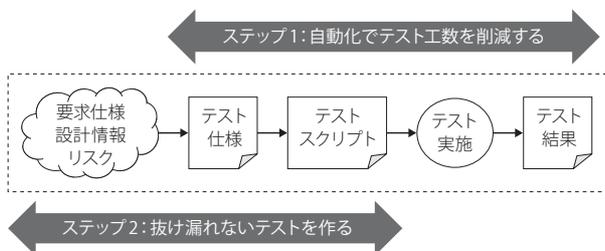


図2. システムテスト自動化のプロセス

システムテスト自動化の推進にあたり、二つのステップで取り組むこととした。

Two steps to promote system test automation

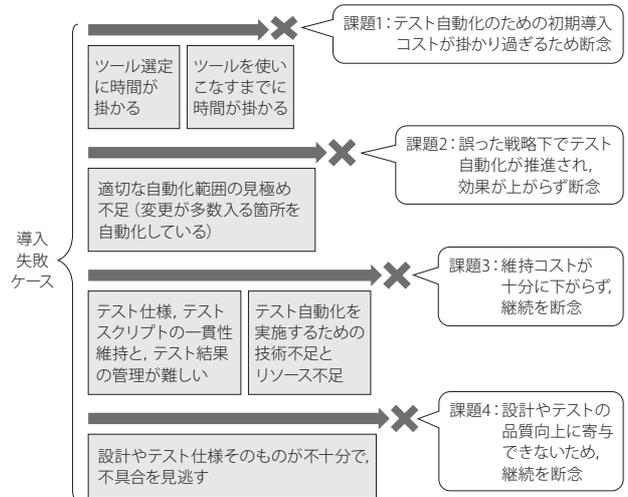


図3. システムテスト自動化の課題

東芝グループ内や社外において、システムテスト自動化の導入失敗ケースを調査した結果、四つの課題に集約された。

Issues in system test automation

スト仕様をベースにテストスクリプトを作成することでシステムテストの実施を自動化し、その後、複数の製品シリーズに展開してシステムテストの工数を大幅に削減する。また、ステップ2では、1章で述べた②の課題を解決するために、テストスクリプト仕様設計の抜け漏れチェックやテストスクリプトの自動生成によって、QCD (Quality : 品質, Cost : コスト, Delivery : 納期)の要求を満たす製品開発を実現する。

一方で、システムテスト自動化を導入・定着させる際に、多くの組織で直面している課題を図3に示す。東芝グループ内でも複数の開発部門において、いち早くシステムテスト自動化に取り組み、テスト自動化ツールの購入や試行などを行ってきた。しかし、図3に挙げた課題が顕在化し、システムテスト自動化が継続されていない状況が散見された。

ステップ1には課題1, 2, 3があり、要約すると「システムテスト自動化の導入・維持にコストが掛かり過ぎ、効果が出ない」ことが、ステップ2には課題4があり、「現状のテスト仕様抜け漏れがあり、効率的かつ効果的にテストスクリプトを作成できない」ことが明らかになった。これらをそれぞれのステップにおける阻害要因と捉えて、表1に挙げる施策を展開した。

3. システムテスト自動化技術の普及・展開への取り組み

社会インフラ系の大規模システムのソフトウェア開発を対

表1. 阻害要因に対する施策とその内容

Measures to overcome hindering factors

阻害要因	施策	施策内容
ステップ1の課題： 導入・維持コストが掛かり過ぎ、効果が 出ない(課題1, 2, 3)	①導入障壁の克服	ツール選定, 自動化範囲の見極め, 自動化の計画立案と実行管理の支援
	②体制の整備	開発担当者, 支援担当者, テスト開発者 の役割とプロセスを明確化
	③環境の整備	テスト仕様, テストスクリプト, 及びテスト 実施結果を一元管理できる環境を構築
	④保守しやすい テストスクリプト の開発	テストスクリプト開発技術の応用
ステップ2の課題： テスト仕様に抜け 漏れがあり, 効率的 かつ効果的にテスト スクリプトを作成 できない(課題4)	⑤テスト設計の 品質向上のため の診断と改善	テストプロセス診断・改善技術 (テストプロセス成熟度モデルTPI NEXT® の利用)
	⑥テストスクリプト を効率良く作成	テストスクリプト自動生成技術の開発

象に, システムテスト自動化に取り組んだ。まず, 表1の施策①～④の実践を行った。施策①～④の取り組み例を以下に示す。

3.1 施策①：導入障壁の克服

図3の課題1, 2に関する導入障壁を克服するために, ツール選定や, 自動化範囲の見極め, 自動化の計画立案と実行管理の支援を行うとともに, システムテスト自動化の導入・推進ガイドを作成した。

システムテスト自動化の導入・推進ガイドは, 組織やプロジェクトでシステムテスト自動化を導入したいと思った際に, システムテスト自動化の導入を成功に導く手順として戦略策定ステップを示し, ステップごとに何をするとよいか, 何を
してはいけないかなどの参考情報を提供している。

システムテスト自動化の戦略策定ステップを図4に示す。

システムテスト自動化そのものを目的とせず, 自動化の目的(例えば, システムテストの効率向上)を明確にするこ

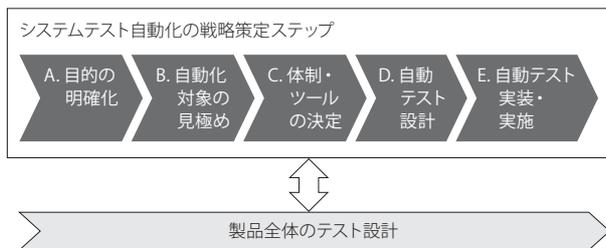


図4. システムテスト自動化の戦略策定ステップ

システムテスト自動化を失敗せずに完遂するための手順とポイントを, 五つの戦略策定ステップで整理した。

Steps to develop system test automation strategy

とや, 全てを自動化しようとせず, 自動化対象の見極めを行うことをポイントとしている。そのため, システムテスト自動化のメリット・デメリット, システムテスト自動化の向き・不向きを, プロジェクトの特徴, ソフトウェアの特徴, 及びテストの特徴, の三つの観点から分類した。

また, システムテスト自動化を担当した人がいなくなって構築した仕組みが使えなくなることを防ぐために, システムテスト自動化の活動を組織の開発プロセスの中に取り入れることや, 設計担当, テスト担当, テスト自動化担当など, 役割を意識してシステムテスト自動化を導入する必要性についても記載した。

3.2 施策②：体制の整備

システムテスト自動化導入・推進ガイドの記述に基づき, システムテスト自動化を行う体制の整備を行った。図5に各開発者と担当者の役割を示す。ベトナムのソフトウェア開発の拠点に, 約10名で構成されるテストセンターを設置し, このテストセンターにシステムテスト自動化技術を集約してシステムテスト自動化導入の立ち上がりが早くなるようにした。テストセンターのメンバーがテスト開発者となりシステムテスト自動化の実装を行うようにした。ベトナムのリソースを活用することで, テストスクリプト開発・保守のコスト削減にもつながった。また, 役割分担を明確にすることで, テスト専門家によるシステムテストの充実を図り, 品質向上の効果も狙った。

東芝のソフトウェア技術センターが支援担当者となり, 活動の全体設計や運営支援を行った。

テストセンターのメンバーには, 開発プロジェクトにおけるシステムテスト自動化の目的や, 目標, 社内におけるシステ

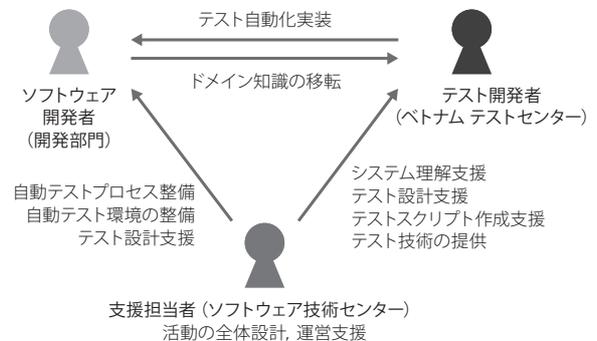


図5. 各開発者と担当者の役割

システムテスト自動化の専門家であるテストセンターがテスト開発者となり, システムテスト自動化を進める。

Roles of workers in charge of each section

ムテスト自動化推進の将来像などを説明し、システムテスト自動化が期待されている技術であることを共有した。

毎週、開発担当者と支援担当者、支援担当者とテスト開発者の二つの定例会議を開催した。定例会議での主要な確認・検討項目は、以下のとおりである。

- (1) テストスクリプト開発状況の確認
- (2) システムテスト自動化範囲の見極め
- (3) テストスクリプト開発や実行に関するリスクの先取り
- (4) 自動化されたシステムテストの実施結果の確認
- (5) 不具合と思われる現象の確認

3.3 施策③：環境の整備

図3の課題3に関しては、システムテスト自動化の環境整備を行った。システムテスト自動化を安定運用するには、整然としたシステムテスト管理が前提と考えている。テスト仕様とテストスクリプトの対応が取れていない場合は、システムテスト結果の管理が難しくなる。また、仕様変更や派生開発時に既存のテストスクリプトの修正に時間が掛かったり、最悪の場合には作り直しが発生したりする。東芝グループでは、テスト仕様とテスト結果を一元管理できるWebシステムであるテスト管理ツールTETRAPLUSを広く活用している⁽²⁾。このツールは、テストの実施状況を様々なビューで確認・分析できるとともに、テスト資産（テスト仕様・結果）を再利用するための仕組みを提供している。

テストセンターと連携してテストスクリプトとテスト仕様の一貫性を維持するために、図6に示すテスト管理ツール

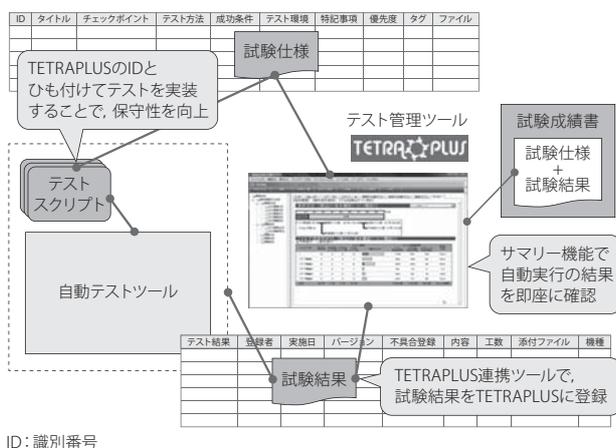


図6. システムテスト自動化環境の整備

テスト管理ツールTETRAPLUSとテスト自動化ツールを連携させて、テスト仕様とテストスクリプトの一元管理を実現した。

Established system test automation environment

TETRAPLUSを軸としたシステムテスト自動化環境を整備した。

市販やフリーソフトウェアのテスト自動化ツールで作成したテストスクリプトとTETRAPLUSに登録されたテスト仕様をひも付けし、開発した連携ツールを用いて自動実行したテストの結果をTETRAPLUSに保存した。これにより、仕様変更時などには、テスト仕様から修正が必要なテストスクリプトがたどれるようになり、テストスクリプトの一元管理が実現できた。また、テスト自動化ツールでテスト実行時に取得した画面キャプチャーをTETRAPLUS上に登録することで、システムテスト結果のエビデンス管理を行えるようにした。更に、システムテストの合否を自動的に判断することが難しいテストに関しては、TETRAPLUS上にキャプチャーした画面をテスターが目視することで、システムテストの合否を判断できるように工夫した。

3.4 施策④：保守しやすいテストスクリプトの開発

システムテストを効率良く実行するための前処理や、後処理、共通処理などを切り出した。また、システムテスト環境やシステムテストデータの変更に対応しやすいデータ駆動スクリプトの構造を採用した。

4. 適用効果の例

社会インフラ系の大規模システムの開発を対象に、システムテスト自動化に取り組んだ効果の例を示す。

自動化の向き・不向きや、再利用を考慮したシステムテスト自動化の対象範囲を特定し、既存のテスト仕様の50%を自動化できれば有効と判断してテストスクリプトの開発を行った。今回の適用例では、30%のシステムテスト自動化が完了し、手動テストのコストをほぼ30%削減できた。今回は、システムテスト自動化の初期導入コストが掛かったが、次機種以降では、この30%に相当するシステムテストのコストが削減できると考えられる。

図7にシステムテストのコスト推移予測を示す。テストセンターの活用により、自部門だけでシステムテスト自動化を進める場合に比べて、初期導入コストを40%削減できた(自部門だけでシステムテスト自動化を実現した場合の初期導入コストは予測値)。これにより、前機種に比べてほぼ同等のコストでシステムテスト自動化部分を含めたテストが実施できるので、今回の開発で自動化の初期導入コストがほぼ回収できることが分かる。また、コスト推移予測では、システムテスト自動化を進めることで、次機種以降のシステムテストのコストは更に削減可能であり、今後大幅なコスト削減が期待できる。

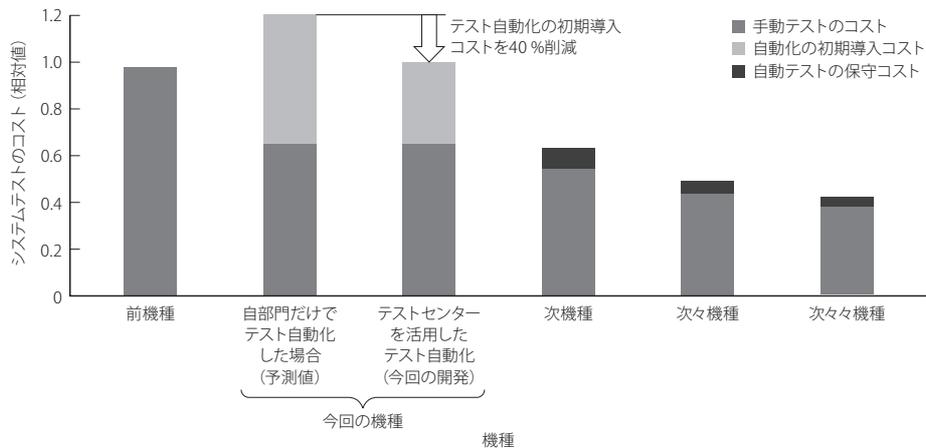


図7. システムテストのコスト推移予測

今回の開発では自動化の初期導入コストが掛かっているが、次機種以降の開発では大幅なコスト削減が期待できる。

Changes in estimated system testing costs

5. あとがき

テスト工数増大の課題に対して、システムテスト自動化の普及・展開に向けた取り組みを実施し、コスト削減効果を確認できた。ベトナムのテストセンターを活用したシステムテスト自動化が、東芝グループ内で広がっている。

現在、表1の施策⑤、⑥の取り組みを進めており、品質面でもシステムテスト自動化の効果を上げられるようにしていくとともに、今後もシステムテスト自動化の普及・展開を推進していく。

文献

- (1) 独立行政法人 情報処理推進機構 (IPA) 技術本部 ソフトウェア高信頼化センター (SEC) 監修. ソフトウェア開発データ白書2016-2017. IPA, 2016, 377p.
- (2) 河村 透, 小笠原秀人. ソフトウェアのテスト管理システム. 東芝レビュー. 2011, 66, 1, p.32-36.

・ TPI NEXTは、Sogeti Nederland B.V. の商標。



中野 隆司 NAKANO Takashi
研究開発本部 ソフトウェア技術センター
ソフトウェアエンジニアリング技術部
日本品質管理学会会員
Software Engineering Technology Dept.



田中 裕大 TANAKA Yuta
東芝エレベータ(株)
技術本部 ソフトウェアテクニカルセンター
Toshiba Elevator and Building Systems Corp.



ダン ティ ホン イエン DANG Thi Hong Yen
東芝ソフトウェア開発ベトナム社
Toshiba Software Development (Vietnam) Co., Ltd.