

システム再構築における現行機能保証に対するテストの効果的な実践

日本アイビーエム・ソリューション・サービス株式会社
黒岩 崇
佐藤 宏史

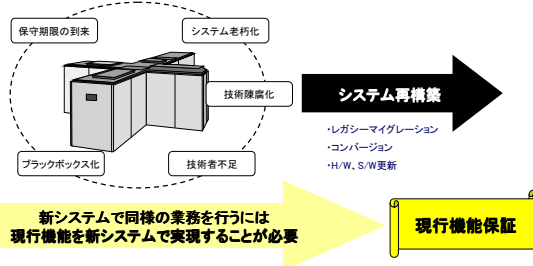
発表内容

1. はじめに
2. システム再構築における現行機能保証の検討ポイント
3. 効果的なテスト方法
4. 実践事例
5. おわりに

1. はじめに

システム再構築とは

保守期限の到来、システム老朽化やブラックボックス化などの要件から現在稼働しているシステムを新たに構築すること



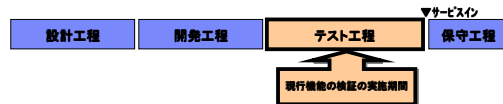
2. システム再構築における現行機能保証

システム再構築における現行機能保証とは

現行システムの機能を再構築後の新システムで同様の機能が稼働することを保証すること

つまり現行機能保証の検証を行うには、新システムの稼働させることが必要となるため、テスト工程期間中に実施する必要がある。

=> テストの効果的な実践が必要



2. システム再構築における現行機能保証

現行機能保証するために必要なテスト

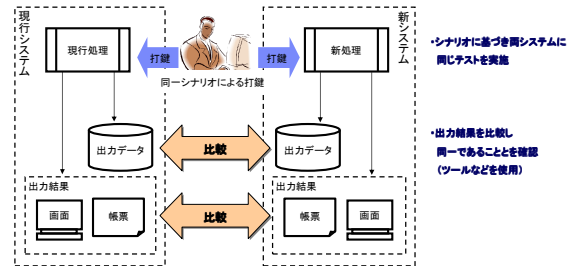
テスト工程で効果的な現行機能の保証を行うために必要となるテストを以下に挙げる

- 現新比較テスト
- 外部システムとのインターフェーステスト
- エンドユーザーによる検証

3. 効果的なテスト実施方法

現行機能保証するために必要なテスト - 現新比較テスト(1/2)

現行システムと新システムで同様のテストを行い結果同一であることを確認



3. 効果的なテスト実施方法

現行機能保証するために必要なテスト - 現新比較テスト(2/2)

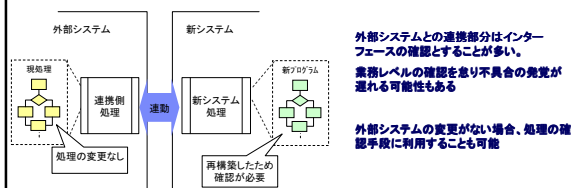
現新比較テストのポイント

| | |
|-----------|---|
| 効率的なテスト方法 | <p>テスト後の比較検証をバッチ処理化することで作業の効率化が可能</p> <p>テスト実施前にデータベース内容を含ませ同じテストケースを実施することで、データベースごとの比較が可能</p> <p>処理が限定的である場合には、事前にデータベースを含ませることなくトランザクションレベルの確認が可能</p> <p>比較対象の自動化の検討が必要 (画面の目視による確認ではミス懸念の恐れがあるため)</p> |
| 効果 | 現新システムの処理結果を比較することで新システムにおける現行機能の保証が可能 |

3. 効果的なテスト実施方法

現行機能保証するために必要なテスト - 外部システムとのインターフェーステスト

連携先の外部システムとのインターフェースおよび処理に変更がない場合のデグレードテスト



| | |
|-----------|---|
| 効率的なテスト方法 | インターフェース確認テストで業務処理のテストケースを含める (外部システムの機能を利用した現行機能保証) |
| 効果 | ドライバなどを開発することなく外部システムとの処理内容に関する現行機能保証が可能 |

3. 効果的なテスト実施方法

現行機能保証するために必要なテスト — エンドユーザーによる検証
最終的に毎日利用しているユーザーが利用することによる検証テスト



使ってもらうことで「何かおかしい」という気づきがある
・数値がちょっとおかしいなど

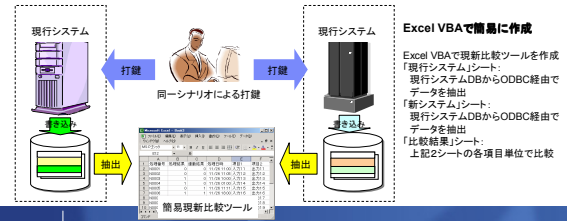
通常はユーザー受入テスト(UAT)で実施(ST終了後)

| | |
|-----------|--------------------------------------|
| 効率的なテスト方法 | 受入テストのタイミングや、暫定的なシステム開放でエンドユーザに利用させる |
| 効果 | 現新比較テストなどで見つけることができないような些細な不具合の検知が可能 |

4. 実践事例

現新比較テスト実践事例(1)

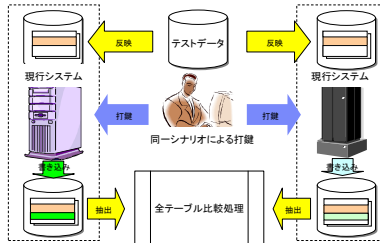
- ✓システム規模: 某金融機関 Webアプリケーション(約200画面)
 - ✓テーブルアクセス範囲が限定的であるため、トランザクション単位でテーブルの現新比較を実施
 - ✓検証に時間がかからず効率化が可能となった
 - 検証ツール効率化: Excel VBAを利用し、簡単に自動比較が可能
 - テスト効率化: トランザクション単位の比較で効率よく検証可能
- ⇒当日中に検証、不具合の修正も可能に



4. 実践事例

現新比較テスト実践事例(2)

- ✓システム規模: 某金融機関 基幹系システム
- ✓開始前に現新で同様のデータ内容に設定、同一シナリオによる打鍵が完了した後、全テーブルの現新比較を実施。(項目レベルによる比較)
- ✓全データの比較が可能となり業務的な内容を理解せず比較検証が可能



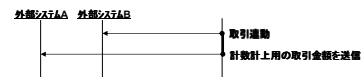
4. 実践事例

外部連携テスト実践事例

- ✓システム規模: 某金融機関 Webアプリケーション(約200画面)
 - ✓外部インターフェーステスト時に業務的な観点でテストケースを設定し実施
 - ✓1取引でnシステムと連携するため、同時に複数の外部システムとテストを行い効率化を図った(オンライン、バッチ合わせて22の外部システムと連携)
- ⇒短時間で外部システムのインターフェーステストが可能、サービス後の不具合なし。



- ・インターフェーステストに業務的なバリエーションを含めテストケースを作成
- ・1取引で複数システムと連携する場合、業務的な内容による複数システム同時の連携テストを実施



5. おわりに

システム再構築を行う場合のその他検討事項

- ✓ 非機能（パフォーマンスなど）は別テストで実施
 - パフォーマンステストによる検証
 - 経験者やテスト実施者の感覚（画面遷移が遅いなど）
- ✓ 新規機能・機能変更に対する考慮
 - 追加機能（現行システムにない機能の新規追加）または機能変更（機能を改修）に対してテスト対象としないよう対応を入れることが必要
 - ex) 言語コンバージョンが終了した後に現行機能保証を実施し、その後改修実施
- ✓ 変更差分の追いつき対応
 - 再構築期間が長期に及ぶ場合に、現行稼働システムに変更が入る場合、後追いで開発中の新システムに対して適用する場合の現行機能保証方法
 - ex) 後追い対応のみ実施、影響のある機能も含め実施