

Webアプリケーション開発における画面モック アップを利用した画面仕様書およびテスト仕様書 の自動生成と設計作業の効率化の提案

第26回 ソフトウェア品質管理研究会
第5分科会 ソフトウェアテスト

2011年2月25日

主査 秋山 浩一 (富士ゼロックス株式会社)
副主査 奥村 有紀子 (有限会社デバッグ工学研究所)
リーダー 坂本 一憲 (早稲田大学 情報理工学専攻)
東海 政治 (株式会社NTTデータCCS)
村上 裕子 (株式会社セゾン情報システムズ)
宮原 里枝

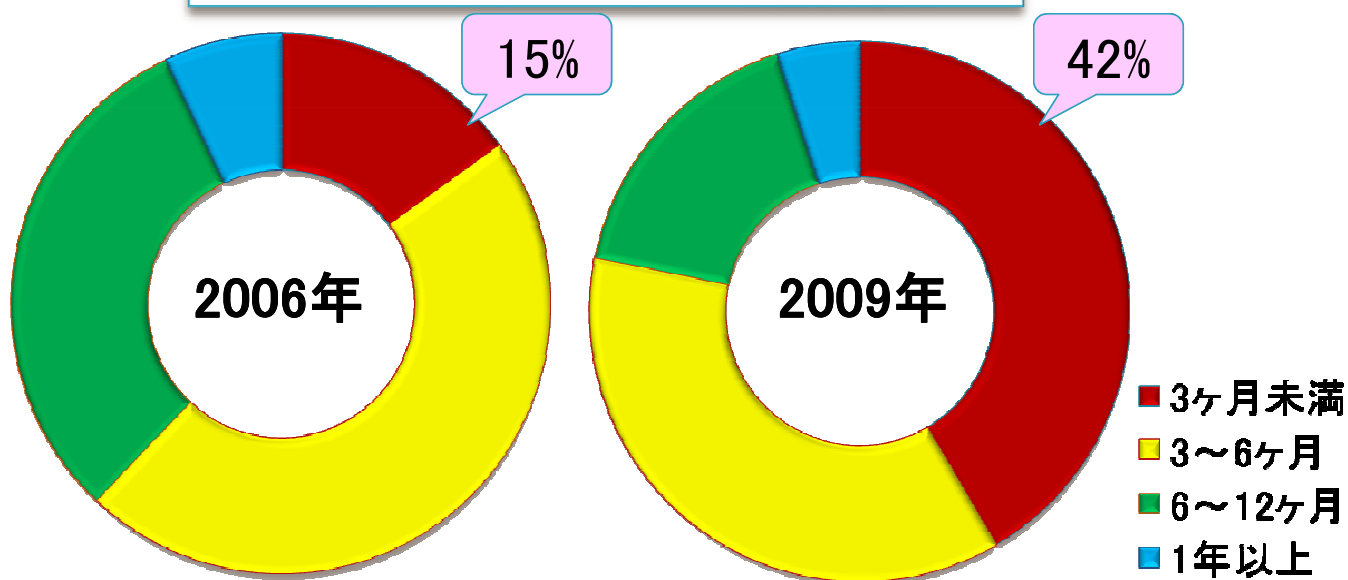
目 次

1. Web開発の現状と問題点
2. 本研究の目的
3. 提案手法
4. 検証結果
5. まとめ

1. Web開発の現状と問題点

1.1 Webアプリケーション開発の現状

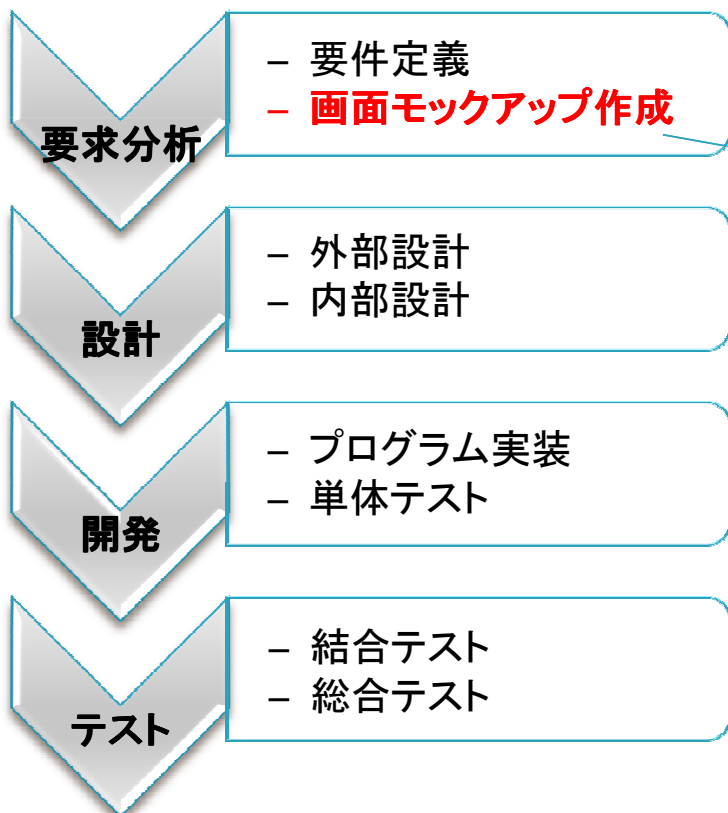
Webアプリケーションの開発期間



短期間での開発が増大
(2006年の15%→2009年には42%)

1.1 Webアプリケーション開発の現状(続き)

Webアプリケーション開発プロセス



開発プロセスの早い段階で画面モックアップを作成

※画面モックアップ(画面モック)
HTML等を用いてUIの外観を試作したプロトタイプ

1.1 Webアプリケーション開発の現状(続き)

画面モック

検証用サンプルページ (資料検索Web)

タイトル

著者名

出版社

所蔵館

出版時期 から までの間

最大表示件数 件

※会員専用のページはこちら

ユーザ名

パスワード

画面モックを作成する目的

- ユーザ関与の増大と改善
 - 早期に画面動作確認が可能
 - ユーザのフィードバックと要求を獲得
 - ユーザとの誤解と対話不足が解消
- 開発フェーズの効率化
 - 画面標準化ドキュメントの策定を支援
 - 実装に画面モックを流用可能

1.1 Webアプリケーション開発の現状(続き)

画面仕様書とテスト仕様書

- 画面仕様書

- 各UIコンポーネントの種別や特徴を記述したドキュメント
- 例) 数字入力可能かどうか、DB更新情報やDB参照元

名前	種別	初期値	Tab順	数字	英字
1 検証用サンプル	ラベル				
2 タイトル	ラベル				
3 タイトル入力テキスト	テキストボックス		1	●	●
4 著者名	ラベル				
5 著者名入力テキスト	テキストボックス		2		●
6 出版社	ラベル				

- テスト仕様書

- テストを実施する際に検証すべき内容を記述したドキュメント
- 例) タイトルの字種が全角文字形式であるか検証(12行目)

10	タイトル	種別	有効系		このアイテムはラベル形式であるか
11	タイトル	種別	無効系		
12	タイトル	字種	有効系		全角文字形式であるか
13	タイトル	字種	無効系		
14	タイトル	必須	有効系		このアイテムを入力しなくても、次の画面へ進
15	タイトル	必須	無効系		
16	タイトル	文字寄せ	有効系		このアイテムは左寄せで表示されているか
17	タイトル	文字寄せ	無効系		
18	タイトル	入出力	有効系		このアイテムは出力専用のアイテムであるか

7

1.2 Webアプリケーション開発の問題点

画面仕様書とテスト仕様書

- 設計者と実装者の認識を合わせる大切なドキュメント
- 高品質なソフトウェア開発には不可欠
- 常に最新の情報が記述されている事

しかし、

短期間での開発が増大

- 仕様設計やテスト設計に掛かる工数を削減する事が多い
- 画面仕様書やテスト仕様書が更新されない

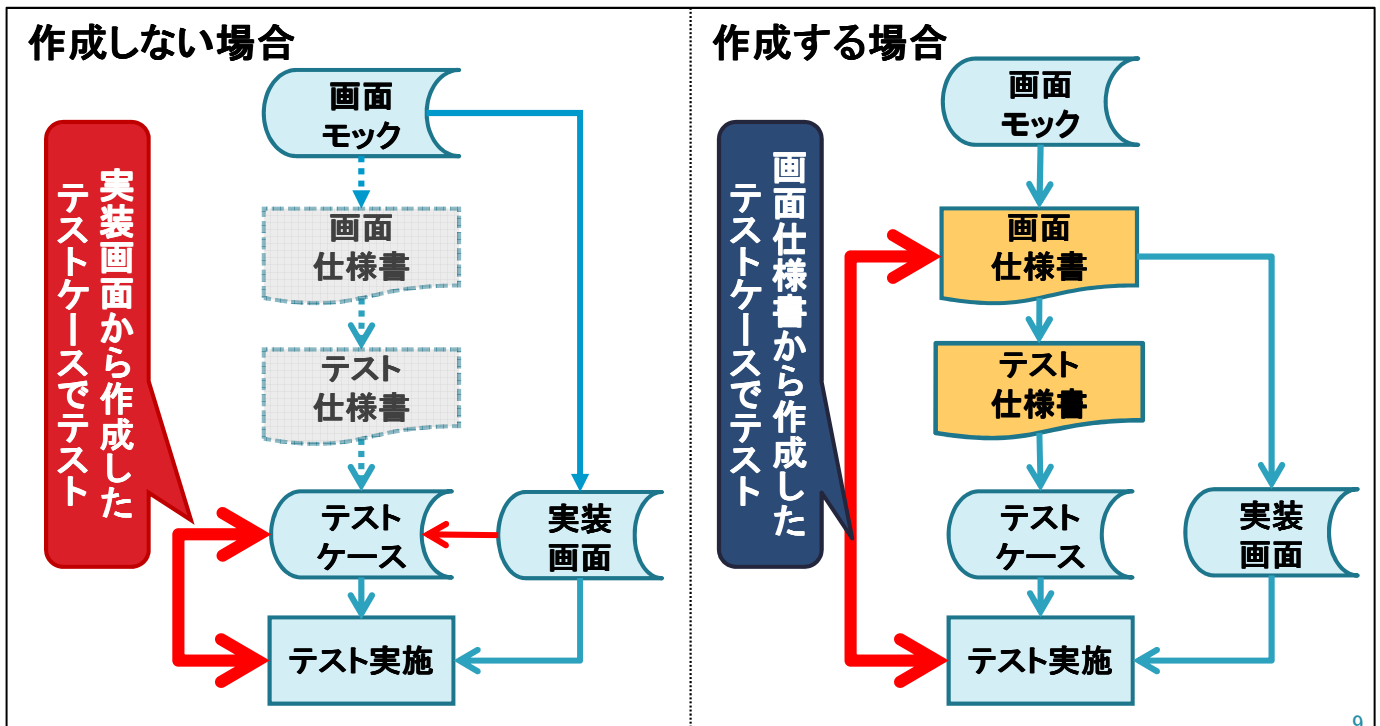


- 画面仕様書やテスト仕様書が作成されない
- 作成されたとしてもトレーサビリティが維持されない

8

1.2 Webアプリケーション開発の問題点(続き)

画面仕様書やテスト仕様書が作成されないので、
テストが仕様と実装の突合せにならない

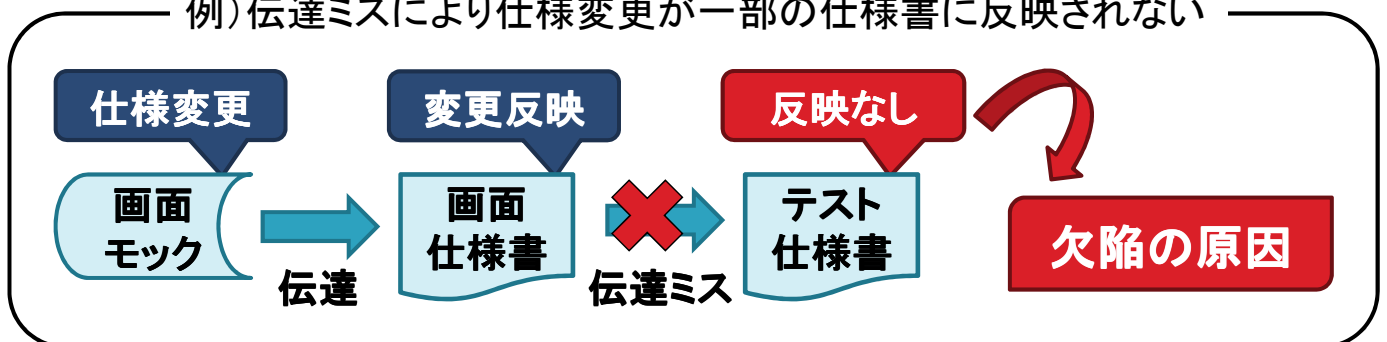


1.2 Webアプリケーション開発の問題点(続き)

画面仕様書やテスト仕様書を作成しても、
トレーサビリティが維持されない

- UIの仕様変更の頻度が高い
 - 実装して初めて分かる問題があるため
- 人的ミスによるトレーサビリティの欠如

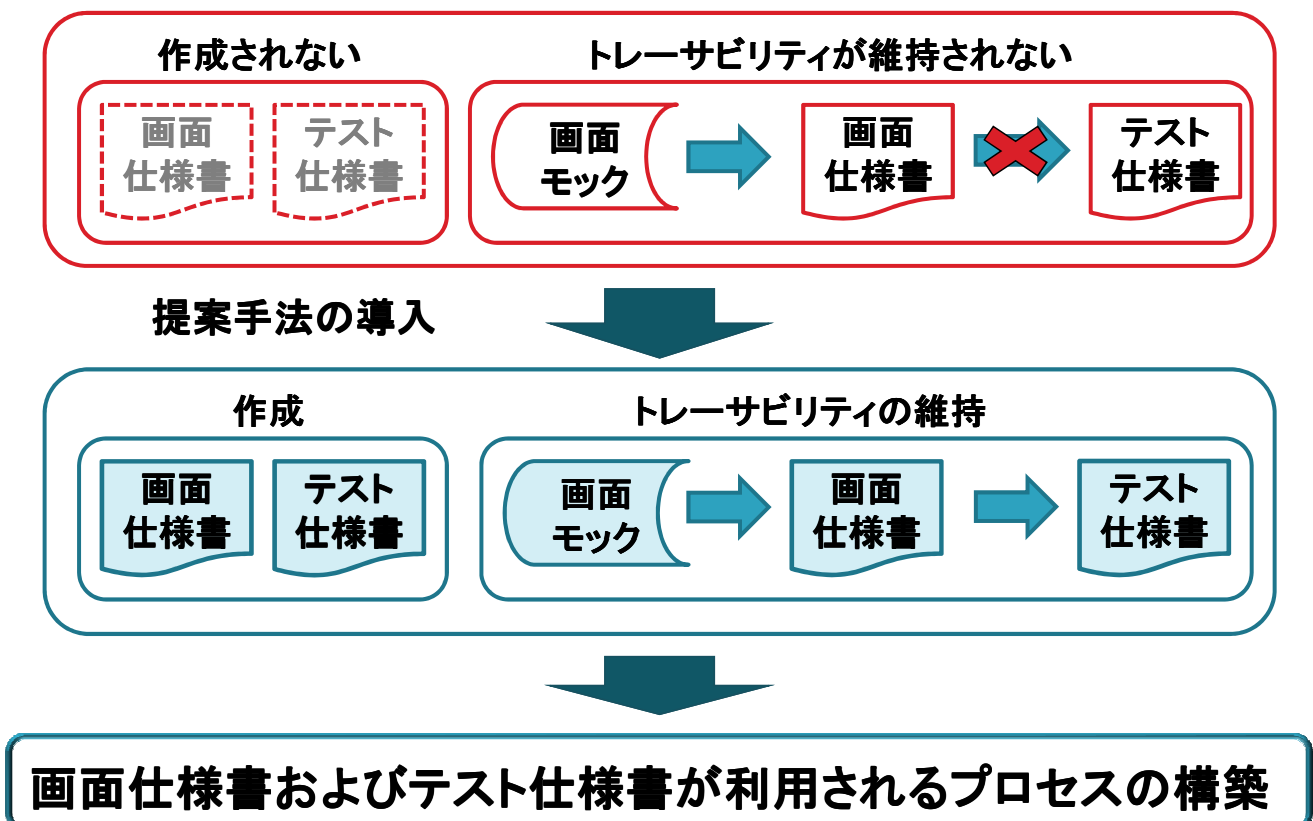
例) 伝達ミスにより仕様変更が一部の仕様書に反映されない



2. 本研究の目的

11

2.1 研究目的



12

2.2 提案手法

画面モックから画面仕様書およびテスト仕様書の自動生成

- 画面モックから画面仕様書の自動生成
- 自動生成された画面仕様書からテスト仕様書の自動生成

自動生成された画面仕様書およびテスト仕様書の更新

- 手入力による詳細情報の追記

期待効果

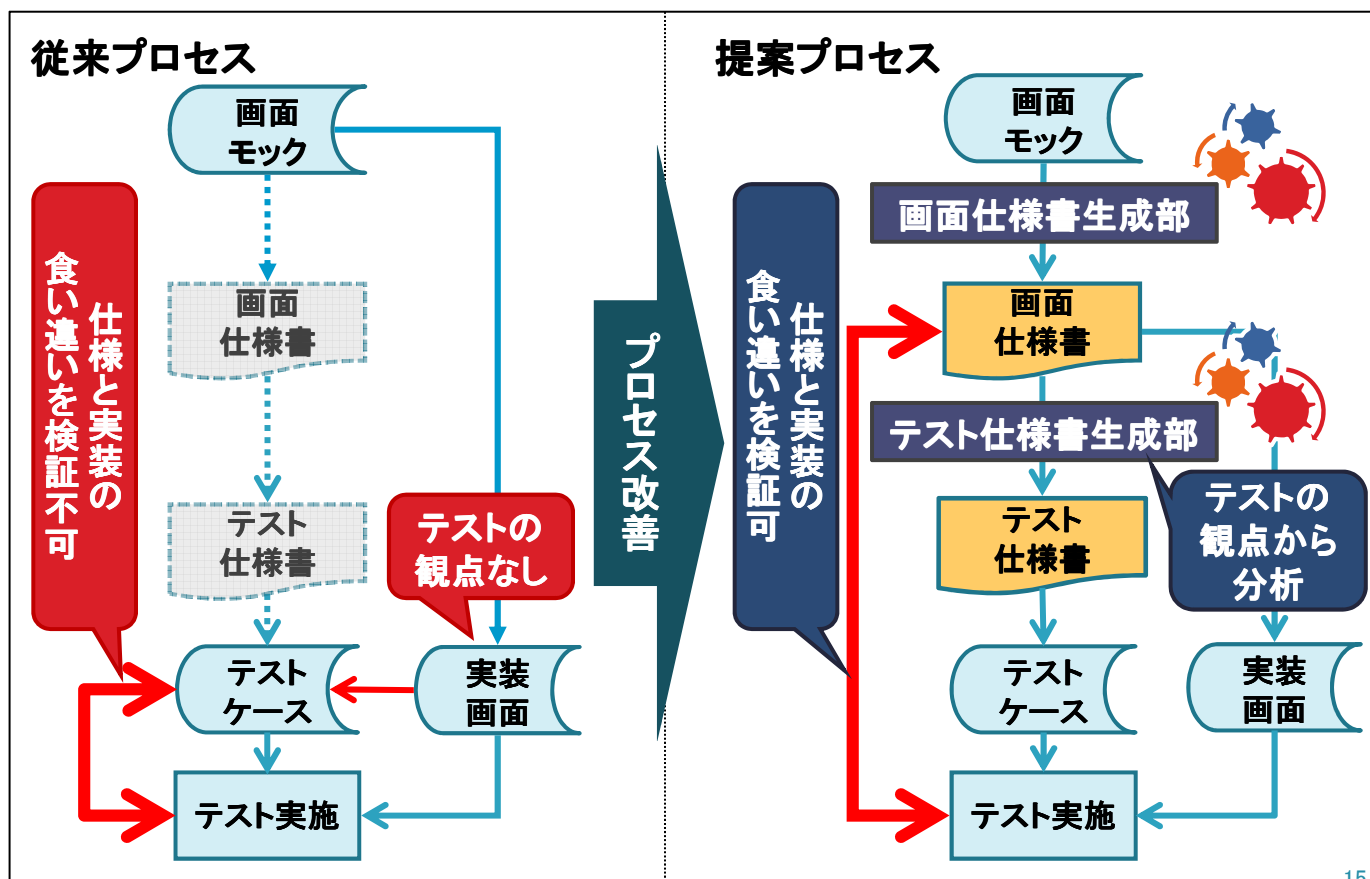
- 画面仕様書、テスト仕様書の自動生成かつ工数の削減
- 自動生成による一定の品質を保持した仕様書作成
- 詳細情報の追記による、品質の高い仕様書の作成
- 自動生成によるトレーサビリティの維持

13

3. 提案手法

14

3.1 従来プロセスと提案プロセスの比較



15

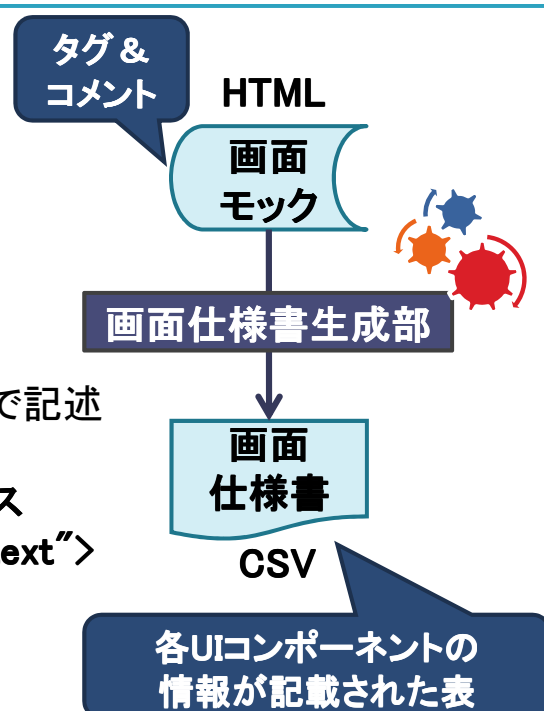
3.2 画面仕様書生成部

1. 画面モック(HTMLファイル)の解析

- タグの解析: UIコンポーネントの種類
 - テキストボックス、ラベル、リンクなど
 - タグの種類やtype属性の値で判別
- コメントの解析: UIコンポーネントの特徴
 - タグの直前に定義した構文のコメントで記述
- 例: 最大文字数が5文字のテキストボックス
`<!-- 最大文字数 = 5 --><INPUT type="text">`

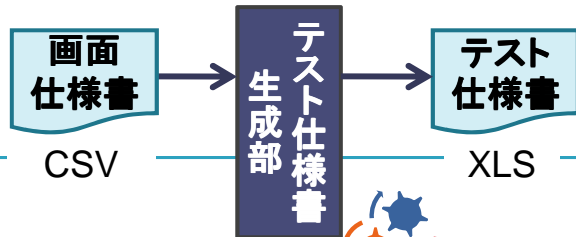
2. 解析結果から画面仕様書を自動生成

→ **作成コストの削減、人的ミスの防止**



16

3.3 テスト仕様書生成部



テスト仕様書生成ツール

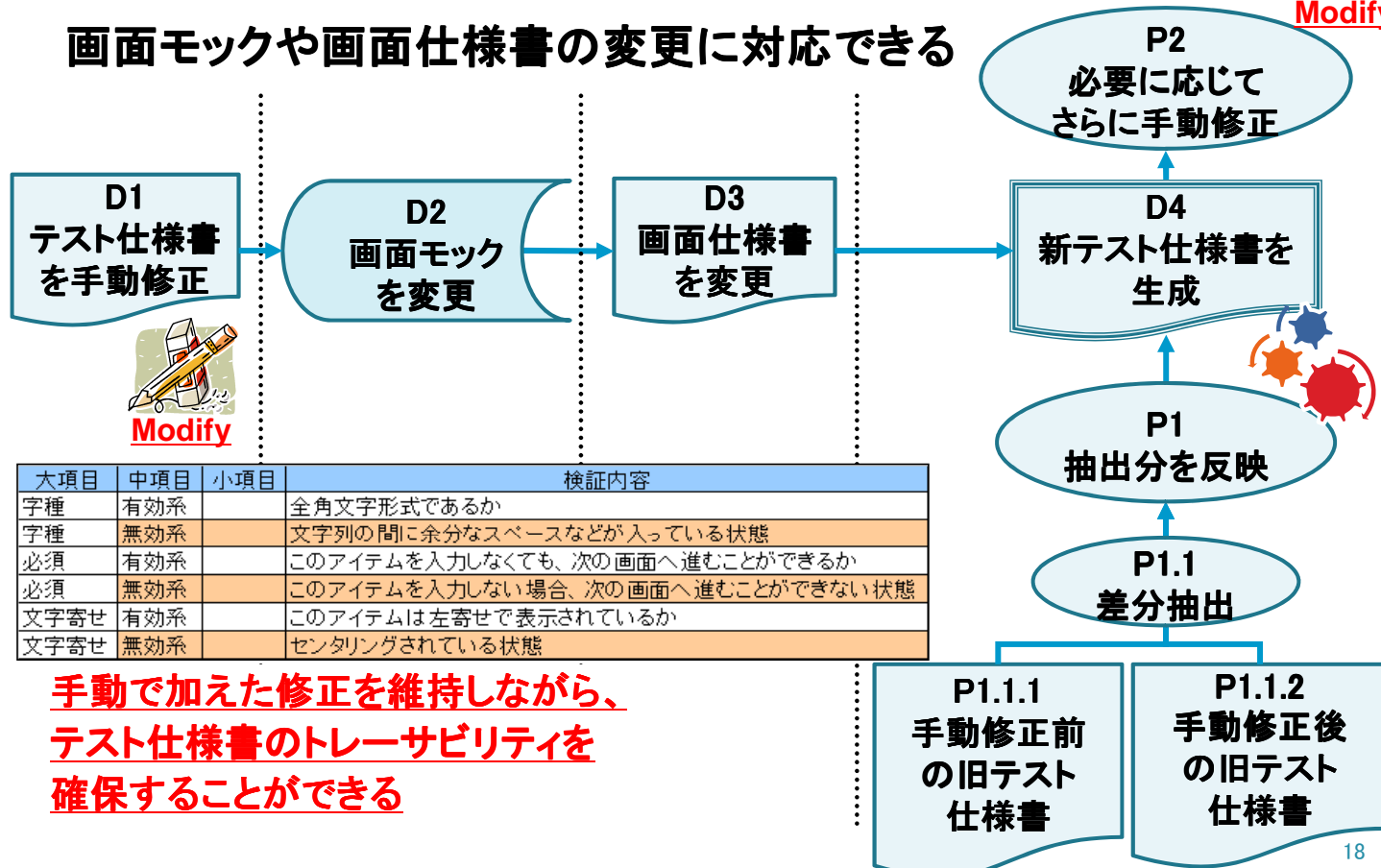
- 画面仕様書を解析し、テスト仕様書を生成する
 - 画面仕様書から情報を読み取り、有効系の検証内容を生成
 - 画面モックに無い無効系の動作を画面仕様書に手動修正で追記し、それを元にテスト仕様書を完成

No.	名前	種別	初期値	Tab順	数字	英字	記号	カギ	全角
1	検証用サンプルページ(資料検索Web)	ラベル							●
2	タイトル	ラベル							●
3	タイトル入力テキスト	テキストボックス		1	●	●	●	●	●
4	著者名	ラベル							●
5	著者名入力テキスト	テキストボックス		2		●		●	●
6	出版社	ラベル							●

項番	アイテム名	大項目	中項目	小項目	検証内容
1	検証用サンプルページ(資料検索Web)	種別	有効系		このアイテムはラベル形式であるか
2	検証用サンプルページ(資料検索Web)	種別	無効系		ラベル形式ではない状態
3	検証用サンプルページ(資料検索Web)	字種	有効系		全角文字形式であるか
4	検証用サンプルページ(資料検索Web)	字種	無効系		文字列の間に余分なスペースなどが入っている状態
5	検証用サンプルページ(資料検索Web)	必須	有効系		このアイテムを入力しなくても、次の画面へ進むことができるか
6	検証用サンプルページ(資料検索Web)	必須	無効系		このアイテムを入力しない場合、次の画面へ進むことができない状態
7	検証用サンプルページ(資料検索Web)	文字寄せ	有効系		このアイテムは左寄せで表示されているか
8	検証用サンプルページ(資料検索Web)	文字寄せ	無効系		センタリングされている状態
9	検証用サンプルページ(資料検索Web)	入出力	有効系		このアイテムは出力専用のアイテムであるか
10	検証用サンプルページ(資料検索Web)	入出力	無効系		入力専用アイテムの状態
11	タイトル	種別	有効系		このアイテムはラベル形式であるか
12	タイトル	種別	無効系		ラベル形式ではない状態
13	タイトル	字種	有効系		全角文字形式であるか
14	タイトル	字種	無効系		文字列の間に余分なスペースなどが入っている状態
15	タイトル	必須	有効系		このアイテムを入力しなくても、次の画面へ進むことができるか

3.3 テスト仕様書生成部

画面モックや画面仕様書の変更に対応できる



大項目	中項目	小項目	検証内容
字種	有効系		全角文字形式であるか
字種	無効系		文字列の間に余分なスペースなどが入っている状態
必須	有効系		このアイテムを入力しなくても、次の画面へ進むことができるか
必須	無効系		このアイテムを入力しない場合、次の画面へ進むことができない状態
文字寄せ	有効系		このアイテムは左寄せで表示されているか
文字寄せ	無効系		センタリングされている状態

手動で加えた修正を維持しながら、テスト仕様書のトレーサビリティを確保することができる

4. 検証結果

19

4.1 従来手法と提案手法の比較実験

従来手法

画面仕様書
テスト仕様書を
手作業で作成

- **新規**で**手入力**作成した時の所要時間
- **保守**で**変更箇所**を手入力で再作成した時の所要時間
- **変更箇所の影響数**確認

比較実験

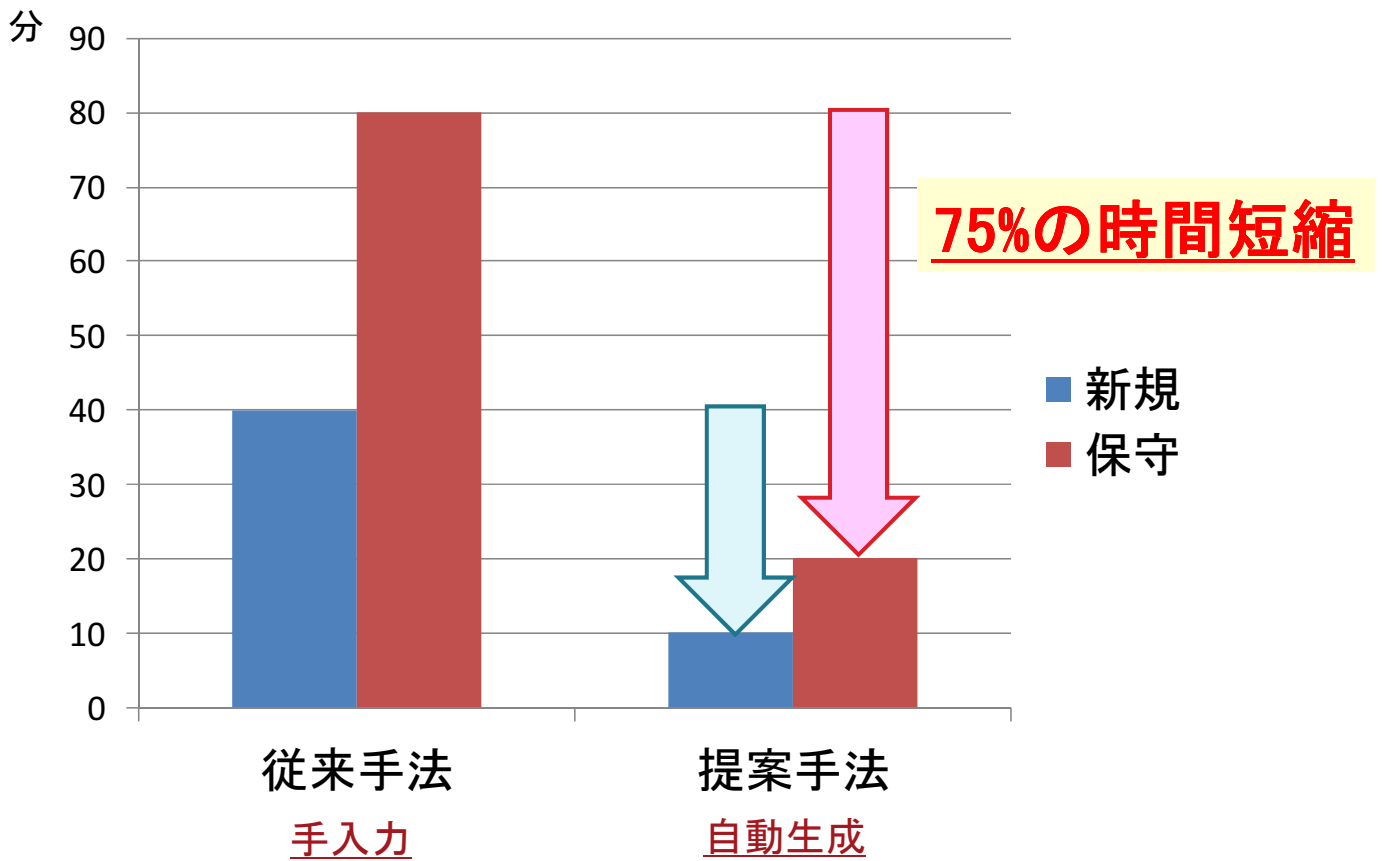
提案手法

画面仕様書
テスト仕様書を
画面モックから自動生成

- **新規**で**自動生成**した時の所要時間
- **保守**で**変更箇所**を自動で再生成した時の所要時間
- **変更箇所の影響数**確認

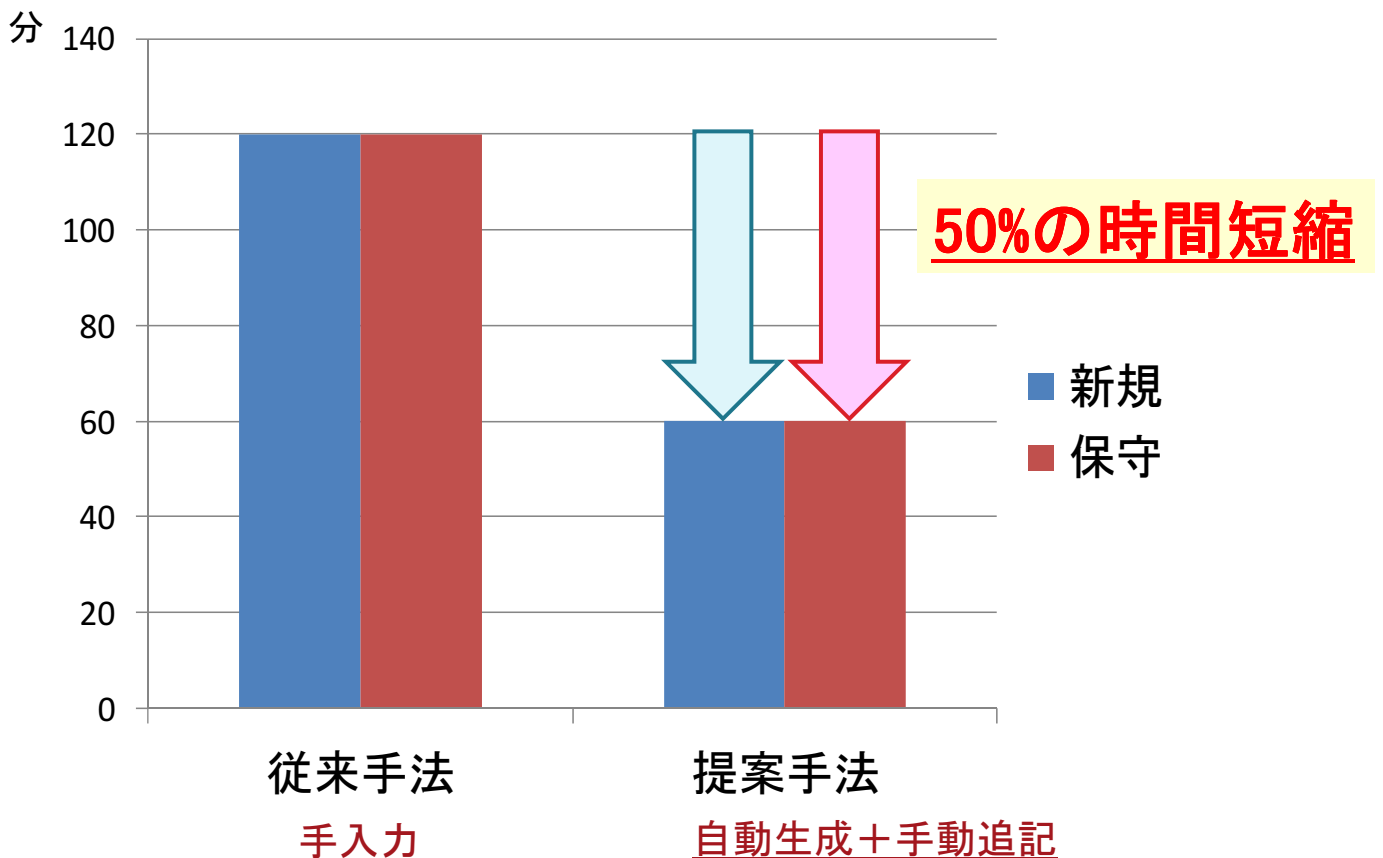
20

4.2 画面仕様書生成 - 所要時間比較



21

4.3 テスト仕様書生成 - 所要時間比較



22

5. まとめ

23

5.1 問題解決

問題1 画面仕様書とテスト仕様書が作成されない

- 提案手法により、画面仕様書生成において 75% の時間短縮
- 提案手法により、テスト仕様書生成において50% の時間短縮



**大幅な時間短縮が可能となったので
容易に仕様書が作成できるようになった**

24

5.1 問題解決

問題2 画面仕様書やテスト仕様書を作成しても、
トレーサビリティが維持されない

提案手法では人手で加えた修正を維持した状態で再生成され
影響も見られなかった

保守コストを低減したうえで
トレーサビリティが確保できるようになった

25

5.2 残課題

実業務への適用

- 業務プロセス改善
- ツールのブラッシュアップ

ツールの実業務への
導入と効果の検証

仕様書の自動生成実業務への適用

- 画面モックから画面仕様書を自動生成
- 画面仕様書からテスト仕様書の自動生成

自動生成可能な
仕様書の現場導入

今後は、業務プロセスへの導入と現場規模による検証で
有効性を図りたい

26

ご清聴いただきありがとうございました