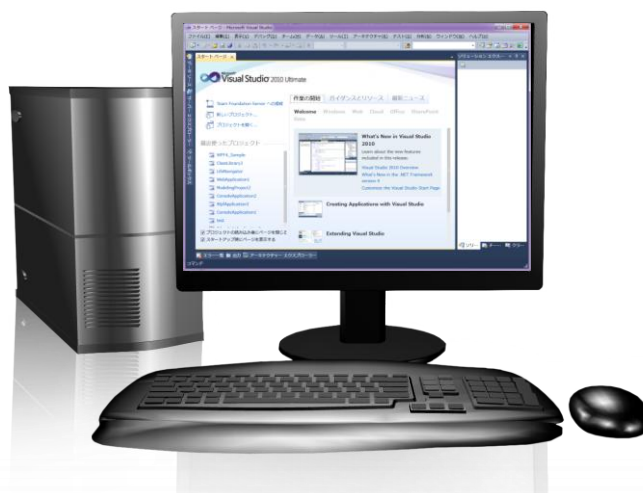


Microsoft® Visual Studio® 2010

評価ガイド

Test Manager 2010 を活用した テスト プロセスの包括的な管理



著作権

このドキュメントに記載されている情報は、このドキュメントの発行時点におけるマイクロソフトの見解を反映したものです。マイクロソフトは市場の変化に対応する必要があるため、このドキュメントの内容に関する責任を問わないものとします。また、発行日以降に発表される情報の正確性を保証できません。

このホワイトペーパーは情報提供のみを目的としています。明示、黙示、または法令に基づく規定に関わらず、これらの情報についてマイクロソフトはいかなる責任も負わないものとします。

この文書およびソフトウェアを使用する場合は、適用されるすべての著作権関連の法律に従っていただくものとします。このドキュメントのいかなる部分も、米国 Microsoft Corporation の書面による許諾を受けることなく、その目的を問わず、どのような形態であっても、複製または譲渡することは禁じられています。ここでいう形態とは、複写や記録など、電子的な、または物理的なすべての手段を含みます。ただしこれは、著作権法上のお客様の権利を制限するものではありません。

マイクロソフトは、この文書に記載されている事項に関して、特許、申請中特許、商標、著作権、および他の知的財産権を所有する場合があります。別途マイクロソフトのライセンス契約上に明示の規定のない限り、このドキュメントはこれらの特許、商標、著作権、またはその他の知的財産権に関する権利をお客様に許諾するものではありません。別途記載されていない場合、このドキュメントで使用している会社、組織、製品、ドメイン名、電子メール アドレス、ロゴ、人物、場所、出来事などの名称は架空のものです。実在する商品名、団体名、個人名などとは一切関係ありません。

© 2010 Microsoft Corporation. All rights reserved.

Microsoft、Windows、Visual Studio、Visual Studio ロゴ、は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。他のすべての商標は、それぞれの所有者の財産です。

目次

Test Manager 2010 を活用したテスト プロセスの包括的な管理	1
I. Test Manager 2010 を活用したテスト プロセスの構成例	1
1. 評価環境の構築について	2
2. 評価版の入手について	2
3. Test Manager 2010 によるテスト プロセスの流れ	2
II. テストの計画	4
1. テスト計画の作成／選択	4
2. 構成の定義	6
3. テスト環境の定義	9
4. テスト設定の定義	10
5. ビルドの定義	12
6. テスト スイートの作成	13
7. テスト ケースの作成	15
III. テストの実施	25
1. テストの状況確認	25
2. 自動テストの実施	25
3. 手動テストの実施	27
4. バグ票の作成と情報の受け渡し	31
IV. 問題の追跡	33
1. テスト実行結果の分析	33
2. ビルドの再割り当て	34
3. 再テストの実施	36
4. 要件、バグ、タスクなどの管理	38
V. まとめ	39

Test Manager 2010 を活用したテスト プロセスの包括的な管理

Microsoft Test Manager 2010 は、ソフトウェア開発における一連のテスト プロセスを包括的にカバーするテストのための専用ツールです。テストの計画、要件とテストの対応付け、テストの実施と管理、発生したバグの管理、その修正の追跡など、テスト サイクルにおける様々な作業をより効率的、効果的に実施および管理することができます。

Test Manager 2010 を活用したテスト プロセスの管理の概要については、以下の記事を参考にしてください。

<http://www.microsoft.com/japan/msdn/vstudio/2010/product/testmanager/>

I. Test Manager 2010 を活用したテスト プロセスの構成例

基本的な利用シナリオとしては、開発者は Visual Studio 2010 を利用して開発と修正作業を行い、テスト担当者は Test Manager 2010 を利用してテスト作業を行います。テストの成果物やデータは、開発基盤である Team Foundation Server 2010 で管理されます。テスト環境を仮想環境として構築する場合には、さらに Visual Studio Lab Management 2010 を構築します。

Test Manager 2010 は単体で販売はしておりません。以下の 2 つの製品にて提供しています。

- Microsoft Visual Studio 2010 Ultimate
- Microsoft Visual Studio Test Professional 2010

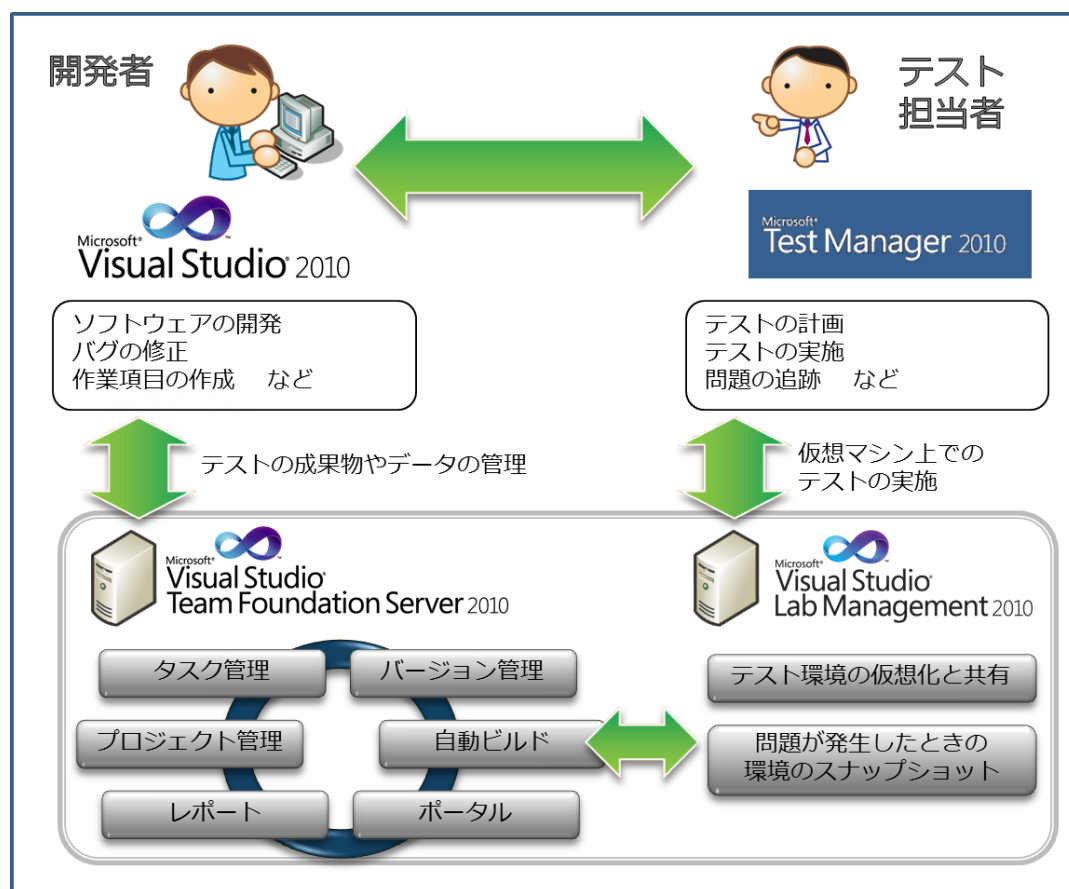


図: Test Manager 2010 を活用したテストの実施

1. 評価環境の構築について

本評価ガイドで解説する機能の多くは、1 台のコンピューター構成で確認することができます。

Team Foundation Server 2010 および Visual Studio 2010 (Test Manager 2010 を含む) は 1 台のコンピューターにインストールすることができます。さらに Team Foundation Server 2010 からインストール時に“基本構成”を選択することによって Windows Vista 以降のクライアント OS にもインストールすることができます。ただし、基本構成を選択した場合は Team Foundation Server 2010 のポータルおよびレポートの機能は利用することができません。

Lab Management 2010 を構築する場合、システム要件を満たすコンピューターであれば、Team Foundation Server 2010 と Visual Studio 2010 と同じ 1 台のコンピューターにインストールすることができます。ただし、そのコンピューターはドメインに属する必要があり、ドメイン コントローラー上にはインストールすることはできません。

Visual Studio 2010、Team Foundation Server 2010、および Lab Management 2010 のインストールについては、以下のドキュメントを参照ください。

■ Visual Studio 2010 のインストール

<http://msdn.microsoft.com/ja-jp/library/e2h7fzkw.aspx>

■ Team Foundation Server 2010 インストールガイド

<http://www.microsoft.com/downloads/details.aspx?FamilyID=2d531219-2c39-4c69-88ef-f5ae6ac18c9f&displayLang=ja>

■ Lab Management 2010 の構築

<http://msdn.microsoft.com/ja-jp/library/dd380687.aspx>

※ さらに以下の Lab Management 2010 向け更新プログラムを適用ください。

<http://support.microsoft.com/kb/983578/ja>

2. 評価版の入手について

Visual Studio 2010 Ultimate、Visual Studio Test Professional 2010 および Visual Studio Team Foundation Server 2010 の評価版は以下のサイトより無償でダウンロード入手いただけます。

<http://www.microsoft.com/japan/visualstudio/download>

構築した評価環境は、製品購入後そのまま継続利用することも可能です。Visual Studio 2010 Ultimate、Visual Studio Test Professional 2010 および Visual Studio Team Foundation Server 2010、それぞれの購入したプロダクトキーを評価版に登録することで、アンインストールすることなく、そのまま利用できます。詳しくは以下の Web サイトをご覧ください。

<http://www.microsoft.com/japan/msdn/vstudio/campaign/trial/>

3. Test Manager 2010 によるテスト プロセスの流れ

Test Manager 2010 によるテスト プロセスは大きく分けると 3 つのフェーズに分けることができます。

これらフェーズは必ずしも順番通りに実施されるものではなく、必要に応じてさかのぼって実施します。

- テストの計画
- テストの実施

● 問題の追跡

Test Manager 2010 を活用したテストを実施するには、まず Test Manager 2010 でどのようにテストが管理されるかを理解する必要があります。テストは以下のように階層化されて管理されています。

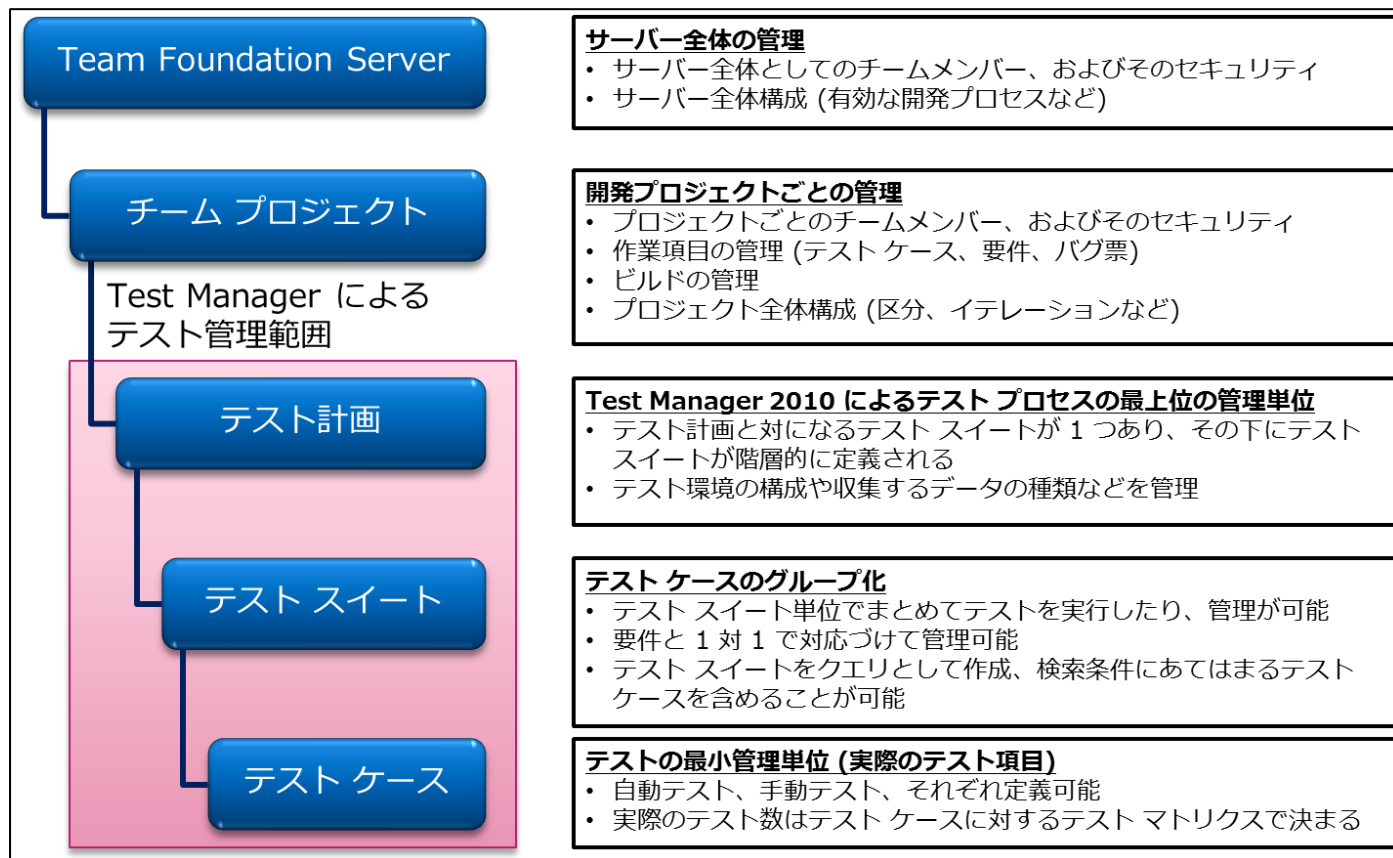


図: テストの管理の階層構造

II. テストの計画

テストの計画フェーズにおいては、テストの環境設定や、テスト項目の定義を行います。

主に以下の項目を実施します。

- テスト計画の作成／選択
- 構成の定義
- テスト環境の定義
- テスト設定の定義
- ビルドの定義
- テスト スイートの作成
- テスト ケースの作成

1. テスト計画の作成／選択

Test Manager 2010 を活用したテストの実施と管理においては、Team Foundation Server を基盤として活用します。Team Foundation Server のチーム プロジェクトごとに、テスト対象のソフトウェア、テストケース、関連する要件、バグ票、チーム メンバーなどを管理します。したがって、Test Manager 2010 でテストを実施したり、管理するためには、まず Team Foundation Server に接続して該当のチームプロジェクトを選択する必要があります。

その次に、テスト計画を作成します。テスト計画は Test Manager 2010 におけるテストを管理する最上位項目です。例えば、「Beta 1 リリース」、「受け入れテスト」など、テストの最終目的で作成するのがわかりやすいです。

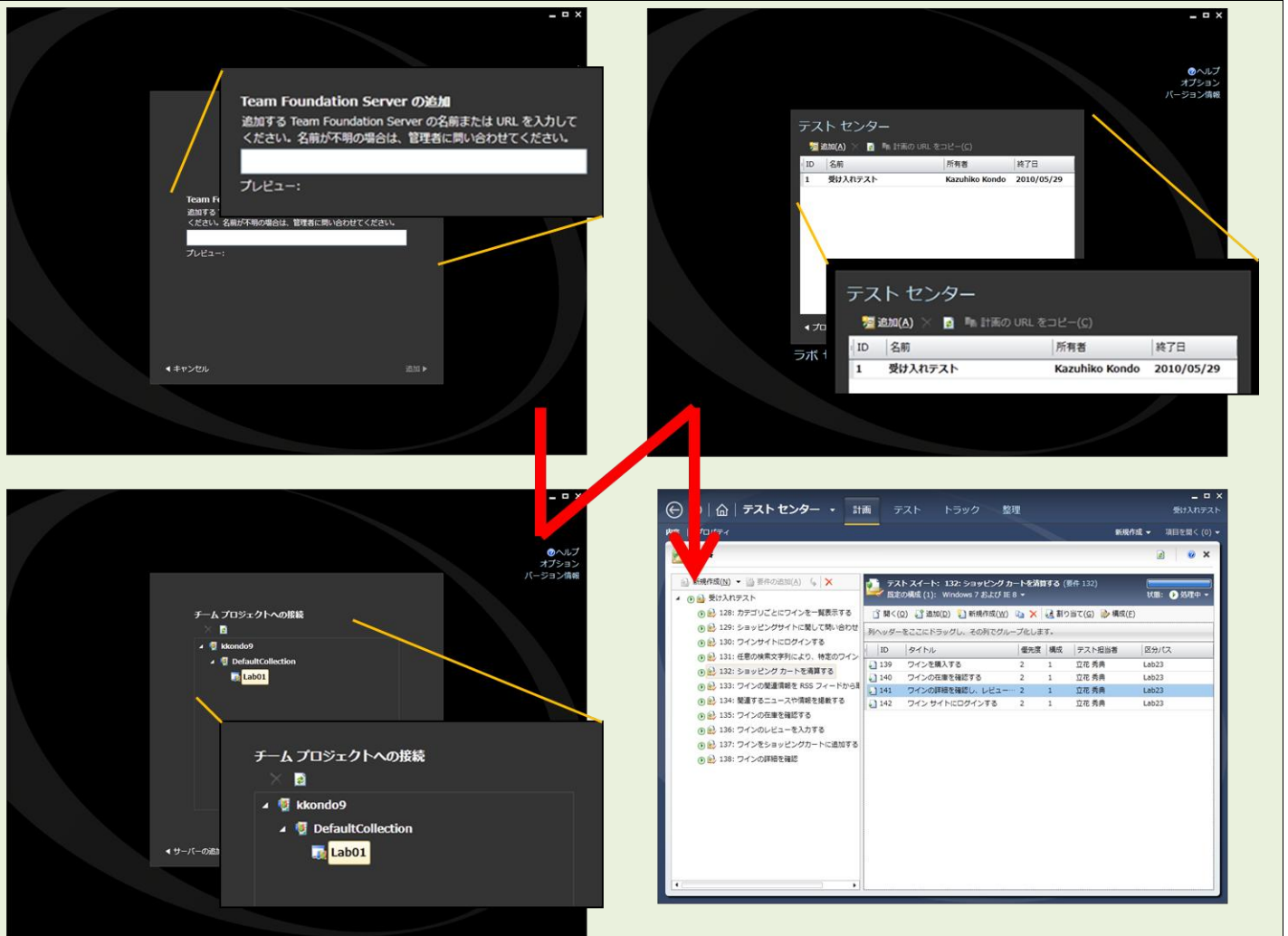
■ チームプロジェクトの選択

1. スタート メニューより [すべてのプログラム] - [Microsoft Visual Studio 2010] - [Microsoft Test Manager 2010] を選択し、Test Manager 2010 を起動します。
2. [Team Foundation Server の追加] 画面より、Team Foundation Server のサーバー名か URL を入力し、[追加] ボタンをクリックします。
3. [チーム プロジェクトの接続] 画面より、チーム プロジェクトを選択し、[今すぐ接続] をクリックします。
※ チーム プロジェクトを作成していない場合、Visual Studio 2010 のチーム エクスプローラーより、チーム プロジェクトを作成します。詳細については以下の Web ページを参考ください。

■ チームプロジェクトの作成

<http://msdn.microsoft.com/ja-jp/library/ms181477.aspx>

4. [テスト センター] 画面より、テスト計画を選択し、[計画の選択] をクリックします。既存のテスト計画が無い場合には [追加] ボタンをクリックし、テスト計画を新たに作成します。



5. テスト計画を変更する場合は [ホーム] ボタンをクリックします。



詳細については、以下の Web ページを参考にしてください。

- 方法: Microsoft テスト マネージャーからチーム プロジェクトに初めて接続する
<http://msdn.microsoft.com/ja-jp/library/dd380739.aspx>
- 方法: Microsoft テスト マネージャーから別のチーム プロジェクトを使用する
<http://msdn.microsoft.com/ja-jp/library/dd286642.aspx>

次にテスト計画の詳細を定義します。

[テスト センター] より [計画] - [プロパティ] メニューを選択します。

この段階では、テスト計画の名前、説明、所有者、状態、開始日、終了日、(必要に応じて区分パス、イテレーション) のみを定義します。それ以外の項目は、構成、テスト環境、テスト設定を定義した後、このプロパティ画面に戻って再度定義し直します。

[状態] フィールドの値の意味は以下の通りです。

状態	値
アクティブ	テスト計画でテストを計画中、または実行中
非アクティブ	テスト計画が完了

最初の段階ではまだ計画中ですので、まずは非アクティブで定義します。テスト計画の状態を“アクティブ”に設定するとテスト計画を表示、実行することができ、テスト計画を“非アクティブ”にすると表示および編集することはできなくなります。

[開始日] にテスト計画が開始される日付、[終了日] にテスト計画が完了する日付を入力します。ただし、この日付は管理のためのものであって、実際の日付より過ぎていたとしてもテスト計画の状態に影響はありません。

なお、テスト計画は削除することができます。この場合、これまで実施したテスト結果も合わせて削除されますのでご注意ください。

テスト計画の作成についての詳細は以下の Web ページを参照ください。

■ 方法: テスト計画を作成する

<http://msdn.microsoft.com/ja-jp/library/dd286583.aspx>

2. 構成の定義

エンド ユーザーの環境は統一されているとは限りません。この場合、システム要件として、ソフトウェアは複数の OS やサーバーなどをサポートする必要があります。また、特に Web アプリケーションであれば複数のブラウザーに対応する必要があることは多いでしょう。このような場合には、1 つの同じテストの手順を、サポート対象の環境構成ごとに確認する必要があります。Test Manager 2010 では、テストにおけるハードウェアやソフトウェア等のテスト条件を“構成”として定義することができます。構成を定義することによって、テスト項目の単位で必要な構成ごとに計画、実施、管理することができます。ただし、構成は

あくまでもテストの条件を決めるための文字列の情報に過ぎず、実際にテスト自体がこの構成通りに物理的に構成されているか、実施されているかについてツールはチェックしませんのでご注意ください。

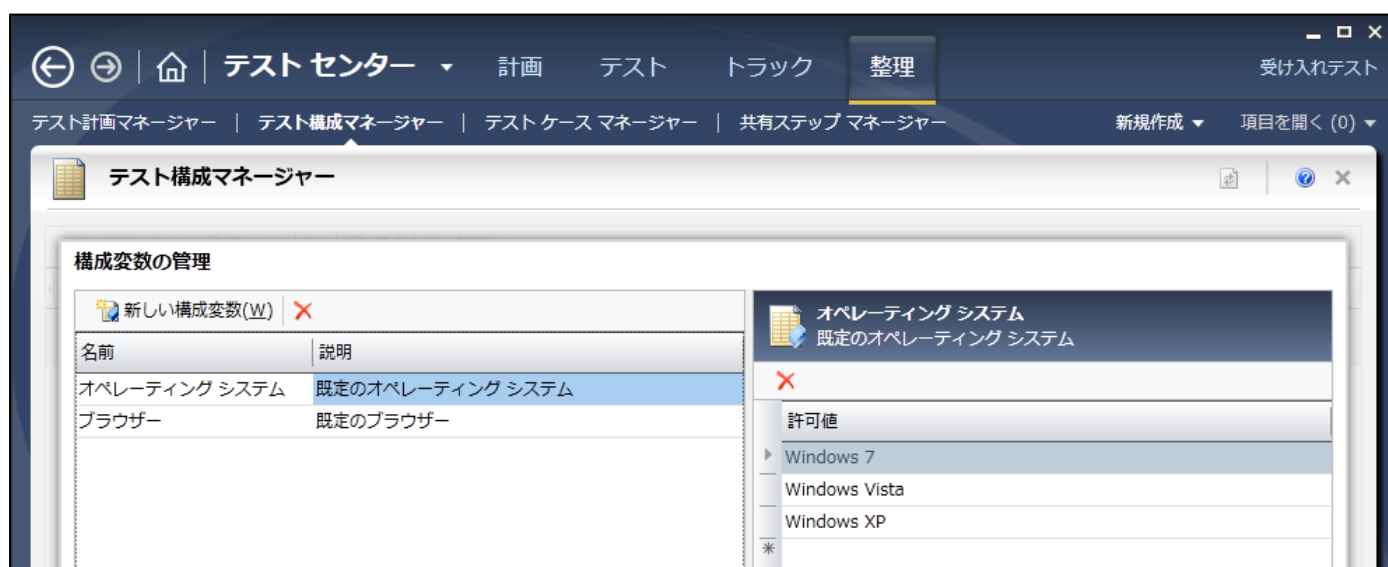
構成は次の 3 つのステップで定義します。

- A) 構成変数の定義
- B) 構成変数の組み合わせで構成を定義
- C) テストマトリクスの作成 (テストごとにどの構成でテストをするかを定義)

A) 構成変数の定義

まず、構成変数と有効な値を定義します。構成変数および値は文字列の情報であり、実際のシステム定義ではありません。

例えば、以下の例では、構成変数として“オペレーティング システム”、“ブラウザー”を定義、“オペレーティング システム”の値として“Windows 7”、“Windows Vista”、“Windows XP”が定義されています。これは、テスト対象のシステムが Windows 7, Windows Vista, Windows XP の 3 つの OS をサポート対象とするために定義しています。



B) 構成変数の組み合わせで構成を定義

次に構成を作成します。それぞれの構成変数の組み合わせと値を選択します。

以下の例では“営業端末 (現行)”という構成を作成し、“オペレーティング システム”として“Windows Vista”、“ブラウザー”として“Internet Explorer 8”が選択されています。

同様に、構成変数の組み合わせと値を選択し、別のテスト対象の構成を作成していきます。



C) テストごとにどの構成でテストをするかを定義

テスト項目ごとに、どの構成をテスト対象とするかを選択します。例えば上記の例で説明すると、管理機能については、管理者端末のみで操作され、営業端末では通常実行されないため、テストする必要がないといった具合です。

これは最終的に以下のような テスト マトリクスとして定義することができます。これは既定の設定としてテスト全体で共通的に定義することもできますし、テスト ケースごとに対象の構成を選択することも可能です。

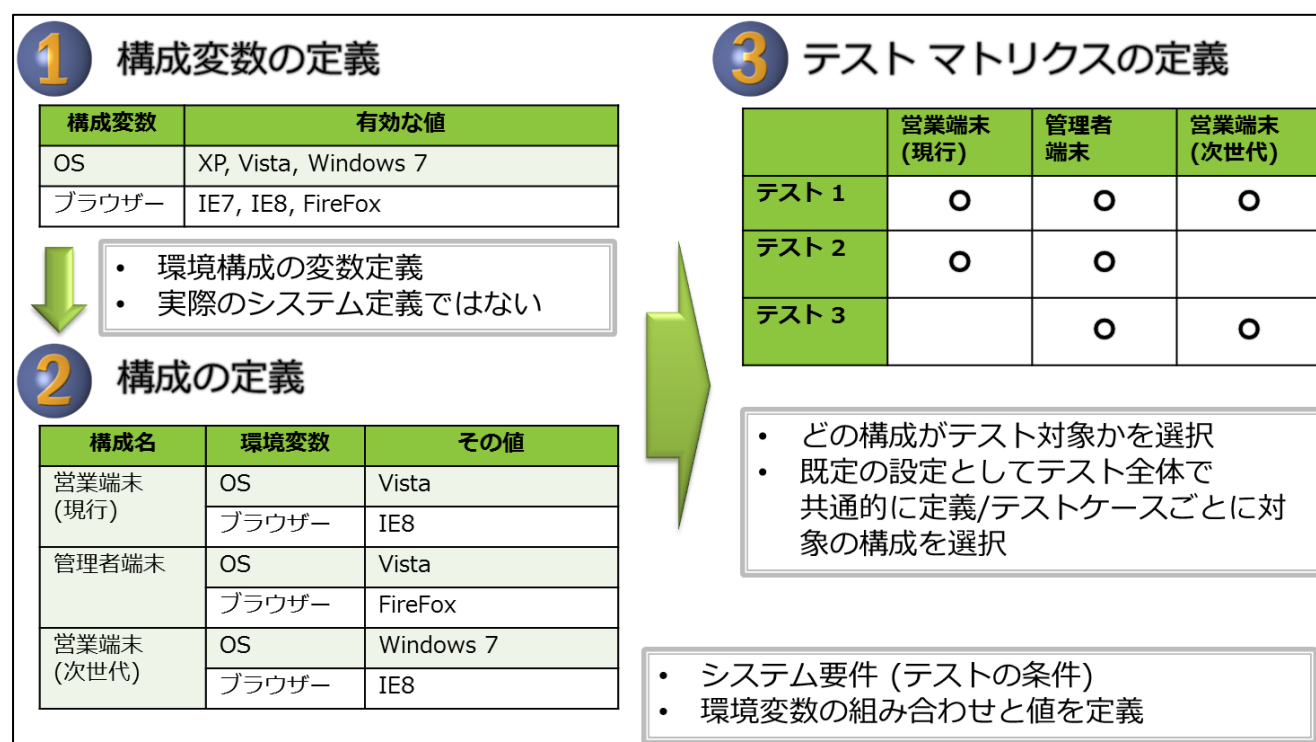


図: 構成の定義手順

上記の例では、テスト 1 は 3 つすべての構成をテスト対象としているのに対し、テスト 2 では OS が Windows Vista の 2 つの構成 (現行営業端末と管理者端末) のみを対象としています。[テスト] ビューで確認すると、テスト 1 は構成ごとに合計 3 つのテスト項目、テスト 2 は構成ごとに合計 2 つのテスト項目が確認できるでしょう。



同じ“販売管理システム”向けのテストケースが存在しています。
これはそれぞれ [構成] が“営業端末 (現行)”のテストと“営業端末 (次世代)”のものです。

実際の操作手順については、以下の URL を参照してください。

- 方法: テスト構成を作成する
<http://msdn.microsoft.com/ja-jp/library/dd286654.aspx>
- 方法: テスト構成を削除する
<http://msdn.microsoft.com/ja-jp/library/dd286658.aspx>
- 方法: 非アクティブ化するテスト構成を設定する
<http://msdn.microsoft.com/ja-jp/library/dd448714.aspx>
- 方法: テスト計画またはテスト スイートに別のテスト構成を選択する
<http://msdn.microsoft.com/ja-jp/library/dd380783.aspx>
- 方法: 特定のテスト ケースの構成を選択する
<http://msdn.microsoft.com/ja-jp/library/dd380786.aspx>
- 方法: テスト構成を更新する
<http://msdn.microsoft.com/ja-jp/library/ee805054.aspx>

3. テスト環境の定義

テスト環境の定義では、テストで使用する物理テスト環境、および仮想テスト環境を定義します。Lab Management 2010 を構築している場合、テスト環境を仮想環境として構築することができます。

ソフトウェアは、スタンドアロンで動作する場合もあれば、クライアント、サーバー型のように複数のマシンで動作することもあります。テスト環境の定義においては、テストの実行時に使用する環境に、例えば Web サーバー、データベース サーバー、デスクトップ クライアントといった“ロール”を定義します。このロールごとにテストの実行時に取得するデータを選択することができます。なお、これらロールは使用目的を示しているのみであり、実際にロールに必要な製品がインストールまたは構成されているかをチェックしたり、不足分をインストール、構成することはありません。

手動テストを実施する場合は、既定の“<ローカル>”を利用してテストを実施することができます。一方、自動テストを実施する場合は、物理環境または仮想環境を明示的に定義する必要があります。実際のテスト環境（物理および仮想）の動作には、テスト コントローラー (Visual Studio Test Controller 2010) およびテスト エージェント (Visual Studio Test Agent 2010) が必要とな

ります。テスト コントローラーおよびテスト エージェントのインストール方法については、以下の Web ページを参照ください。

■ Visual Studio Agents のインストールと構成

<http://msdn.microsoft.com/ja-jp/library/dd648127.aspx>

既定のテストの環境を選択するには、[テスト センター] より [計画] - [プロパティ] メニューを選択し、自動テスト、手動テスト、それぞれ [テストの環境] ドロップダウンリストから、テスト環境の設定を選択します。テスト環境はテストの実行時にも個別に選択することができます。

手動での実行:		自動での実行:	
テストの設定:	販売管理システム検証 (詳細データ取得) ▼ 開く	テストの設定:	<既定> ▼ 開く
テスト環境:	<ローカル> ▼ 管理	テスト環境:	なし ▼ 管理
ビルド:		構成:	営業端末
ビルドのフィルター:	基本ビルド (初期テスト (成功)) ▼	この計画の構成:	Windows 7 および IE 8 ▼

4. テスト設定の定義

テスト設定では、テストの結果として収集するデータの種類を定義したり、テスト時に環境に制限をかけてエミュレートしたりすることができます。テストの設定では、以下の項目を定義します。

- 実行するテストの種類 (手動または自動)
- テストの実行に使用する環境 (ロール)
- 各ロールに使用する各種設定 (診断データ アダプター)

ここで特に重要な設定は、診断データ アダプターです。診断データ アダプターはテスト実施時においてどのようなデータを取得するかを定義します。

設定	内容
IntelliTrace	<p>デバッグの履歴を保持する IntelliTrace で取得する診断データの詳細を定義します。</p> <p>これにより、IntelliTrace のデータを格納した IntelliTrace ファイル (.iTrace) が作成されます。Test Manager よりバグ票を作成する際に、この IntelliTrace ファイルがこのバグ票に自動的にリンクされます。この IntelliTrace ファイルから、テストを実施していない開発者は、別のコンピューターでローカル セッションを再現することができ、問題の再現性が向上します。</p> <p>詳細については、以下の Web サイトを参照してください。</p> <p>■ IntelliTrace を使用したデバッグ</p> <p>http://msdn.microsoft.com/ja-jp/library/dd264915.aspx</p> <p>※ IntelliTrace データの収集を有効にすると、コード カバレッジ データの収集は実行されませんのでご注意ください。</p> <p>※ Web クライアント ロールに IntelliTrace を使用する場合、[IntelliTrace およびテスト</p>

	の影響用の ASP.NET クライアント プロキシ] も選択する必要があります。
IntelliTrace およびテストの影響用の ASP.NET クライアント プロキシ	ASP.NET Web アプリケーションで、このプロキシを有効にすると IntelliTrace およびテスト影響度分析において、クライアントから Web サーバーへの http 呼び出しに関する情報を収集できます。
ASP.NET プロファイラー	Visual Studio 2010 から負荷テストを実施する際に、ASP.NET Web アプリケーションのパフォーマンス データを収集する ASP.NET プロファイリングに関する設定ができます。
コード カバレッジ	コード カバレッジ情報 (テストでカバーされるコードの範囲) に関する設定ができます。 ※ リモートの情報は収集されません。 ※ IntelliTrace を有効にしている場合、コード カバレッジ データの収集は実行されません
イベント ログ	収集するイベントログに関する設定ができます。
ネットワーク エミュレーション	ダイヤルアップなどの特定のネットワーク接続の速度を再現し、テストに対して疑似的なネットワーク負荷をかけ、実際の利用シナリオに近いテストを実施することができます。
システム情報	テストを実行するコンピューターのシステム情報を収集し、バグ票に記載します。
テスト影響	テストの実行時にどのメソッドが使用されたかについての情報を収集できます。これを開発者によるコードの変更と比較することにより、一連の修正作業によって影響を受けたテストを判別できます。
ビデオ レコーダー	テストの実行時にデスクトップ セッションのビデオ記録を作成できます。Test Manager 2010 よりバグ票を作成する際に、Windows Media 形式のファイル (.wmv) が添付されます。さらに、テストの手順ごとに時間のハイパーリンクがバグ票に記述され、そのハイパーリンクをクリックするとその手順を実施した時間のストリーミングが写し出されます。
操作	UI の自動テストに活用可能な UI 操作の記録に関する設定ができます。操作記録は、テスト結果と共に保存され、後で操作の記録を再生したり、実行した操作ログを表示したりできます。

テストの設定を作成したり、変更するには、[テスト センター] より [計画] - [プロパティ] メニューを選択し、自動テスト、手動テスト、それぞれ [テストの設定] ドロップダウンリストから、テスト環境の設定を選択します。

手動での実行:		自動での実行:	
テストの設定:	販売管理システム検証 (詳細データ取得) ▼ 開く	テストの設定:	<既定> ▼ 開く
テスト環境:	<既定> <新規作成...>	テスト環境:	なし ▼ 管理
ビルド:	販売管理システム検証 (詳細データ取得)	構成:	
ビルドのフィルター:	基本ビルド (初期テスト (成功)) ▼	この計画の構成:	Windows 7 および IE 8 ▼

テスト設定の定義については、以下の Web ページを参照ください。

- コンピューターの設定およびテストの設定を使用した診断情報の収集
<http://msdn.microsoft.com/ja-jp/library/dd286743.aspx>

5. ビルドの定義

Team Foundation ビルド機能を用いている場合、テスト対象のビルドを選択することができます。これによって、すべてのテスト担当者がそのビルドを使用するように指定でき、特定のビルドに実装された要件や修正されたバグを確認することができます。

ビルドを定義するには、[テスト センター] より [計画] - [プロパティ] メニューを選択し、[使用中のビルド] より [変更] リンクをクリックします。表示される [ビルドの割り当て] 画面（※ [トラック] - [ビルドの割り当て] でも表示することができます）でテスト対象のビルドを変更することができます。その際、現在のビルドと採用するビルドを比較し、ビルド間での作業項目の変更内容や、ビルドの変更にあたり推奨されるテストが表示され、テストや問題の追跡に役立ちます。詳細は「問題の追跡」で解説します。

[ビルドのフィルター] で、どのビルド定義でビルドされたものを対象とするかを絞り込むことができ、さらに [ビルドの品質評価] を指定することで、特定の状態のビルドのみを選択することができます。

6. テスト スイートの作成

テスト スイートは、テスト ケースをグループ化するテストの管理項目です。

Test Manager 2010 では、テスト ケースだけではなく、テスト スイートごとにテストを実施したり、管理することができます。

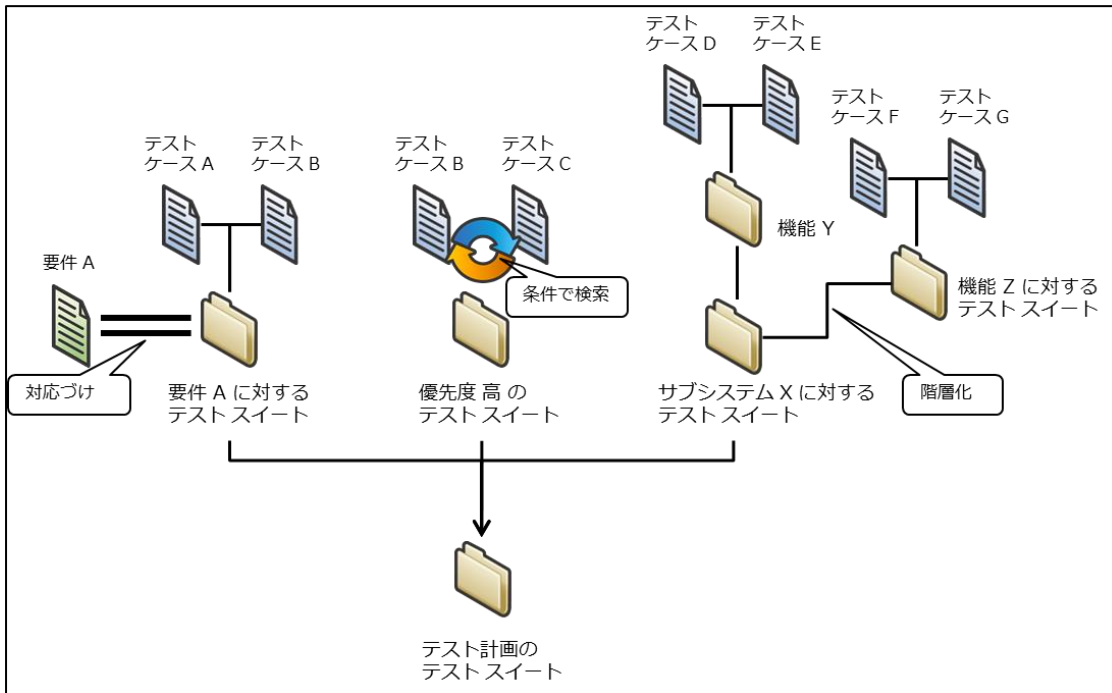


図: テストスイートとテスト ケースの関係

テスト スイートには以下の 3 種類があります。

- **通常のテスト スイート**

任意のテスト スイートを作成することができます。通常のテスト スイートは階層化する（子スイートを持つ）ことが可能です。

- **要件と対応づけたテスト スイート**

Team Foundation Server の“要件”カテゴリの作業項目と 1 対 1 で関連付けが可能です。これによって、要件ごとにテストの重要度や実装の優先度を判断することができます。要件カテゴリの作業項目は、チーム プロジェクトに適用されている開発プロセスによって異なります。MSF for Agile Software Development v5.0 の場合は“ユーザー ストーリー”、MSF for CMMI Process Improvement v.5.0 の場合は“必要条件”です。既に作成したテスト スイートに要件を紐づけることはできません。要件と関連付けると、テスト スイートのアイコンに赤いチェックマークがつきます。



- **クエリ ベースのスイート**

テスト スイートは決まったテスト ケースをあらかじめ設定することもできますが、クエリ ベースのスイートとして特定条件にあてはまるテスト ケースを動的に追加することも可能です。例えば、「優先度が高いテスト ケース」というクエリーベースのテスト スイートを作成すれば、プロジェクトの途中で優先度が高いテスト ケースが作成された場合に、条件にあてはまるテスト ケースをテスト スイートに容易に追加することができます。クエリについては「**エラー! 参照元が見つかりません。**」を参照ください。

テスト スイートの下にはさらに子テスト スイートを作成することができ、テスト スイートをグループ化して、階層構造を持つこ

とができます。ただし、クエリ ベースのテスト スイート、および要件から作成したテスト スイートには、子テスト スイートを作成することはできません。

■ テスト スイートの作成

1. Test Manager スタート メニューより [すべてのプログラム] – [Microsoft Visual Studio 2010] – [Microsoft Test Manager 2010] を選択し、Test Manager 2010 を起動します。
2. [テスト センター] ビューより、[計画] メニューをクリックします。

■ 通常のテスト スイートの作成

3. テスト計画のトップ テスト スイートを右クリックし、[新しいスイート] を選択し、名前を入力します。

■ 要件と対応づけたテスト スイートの作成

4. テスト計画のトップ テスト スイートを右クリックし、[要件を計画に追加] を選択します。
クエリ画面が表示されますので、必要に応じてクエリを修正し、[実行] ボタンをクリックします。
Team Foundation Server に格納されている要件が一覧されますので、該当の要件を選択し、[要件を計画に追加] ボタンをクリックします。

■ クエリ ベースのテスト スイートの作成

5. テスト計画のトップ テスト スイートを右クリックし、[新しいクエリ ベースのスイート] を選択します。
クエリ画面が表示されますので、テスト スイートの名前をつけ、クエリを修正し、最後に [テストスイートの作成] ボタンをクリックします。
どのようなテスト ケースが返るか確認したい場合には、クエリ修正後 [実行] ボタンをクリックしてください。



A) テスト スイートの状態管理

作成したテスト スイートの状態を変更する場合、画面右上に表示されている [状態] の矢印をクリックして、以下の 3 種類から

選択します。なお、“処理中” 状態のテスト スイートのみ [テスト] ビューに表示され、テストを実行することができます。



計画中	現在要件や実装が途中であり、テストがまだ準備状態で実施できない
処理中	テストが準備状態にあり、いつでも実施することができる
完了	テストが完了し、期待される品質レベルを満たしている

作成したテスト スイートがすでに準備段階にある場合には [処理中] のままに、要件が確定していなかったり、実装が途中の場合においては [計画中] に値を変更して、不用意にテストが実施されないように変更ください。

7. テスト ケースの作成

Test Manager 2010 で実施するテストは自動テストと手動テストに分けることができます。自動テストは、テスト実施手順があらかじめプログラム化されたものであり、手動テストはエンド ユーザーによるシステムの操作を手順化したもので、開発プロジェクトの最終工程である総合テストや、システム化案件の発注側による受け入れテストなどで活用ができます。

A) 自動テストの作成

自動テストは、テスト手順があらかじめプログラム化されたものです。

自動テスト ケースを作成したり、テスト ケースに自動テストを関連付けるには Visual Studio 2010 が必要です。Visual Studio 2010 を用いて以下のテストを実施することができます。

- コード化された UI テスト (UI の自動テスト)
- 単体テスト
- データベース単体テスト
- 汎用テスト

※ ロードテストはテストを実行しながら、結果を表示することができないためお勧めできません。

自動テストを実施するには、以下の 2 つの定義が必須です。

- 物理環境、または仮想環境
「テスト設定の定義」および「テスト設定の定義」を参照ください。
- 実行対象のビルド

「ビルドの定義」を参照ください。

■ テスト ケース (自動) の新規作成

※ 本手順は、Visual Studio 2010 による各種テスト、および Team Foundation Server の作業項目の作成の知識が必要です。詳細については各 MSDN Library の記事を参照ください。

■ 作業項目の作成

<http://msdn.microsoft.com/ja-jp/library/ms181316.aspx>

■ 単体テストを使用したコードの検証

<http://msdn.microsoft.com/ja-jp/library/dd264975.aspx>

1. Visual Studio 2010 を起動します。
2. チーム プロジェクトのバージョン管理に格納されているテスト対象のプログラムを開きます。
3. 自動テストが作成されていなければテストを作成します。
4. [テスト ビュー] または [テスト リスト エディタ] ウィンドウを開きます。これらのウィンドウは、メニューより [テスト] - [ウィンドウ] で開くことができます。
5. 該当のテストを右クリックし、[テストからテスト ケースを作成] を選択します。
6. [オートメーションの状態] フィールドが “自動” になっていることを確認し、さらに、作成しているテスト ケース作業項目の [関連付けられたオートメーション] タブをクリックして、必要項目が入力されていることを確認します。
7. 作業項目に必要な事項を入力します。この手順は Team Foundation Server の基本操作と同じです。

詳細については、以下の Web ページを参照ください。

■ 自動テストの作成

<http://msdn.microsoft.com/ja-jp/library/dd380755.aspx>

B) 手動テストの作成

手動テスト ケースは、Visual Studio 2010 の [チーム エクスプローラー] から、Test Manager 2010 から作成することができます。また既に作成済のテスト ケースを登録することもできます。

新しいテスト ケース 2*: ショッピングカートに商品を追加

タイトル(I): ショッピングカートに商品を追加

状態

担当者(G): Kazuhiko Kondo

状態(S): デザイン

優先度(P): 2

オートメーションの状態(U): 自動化なし

分類

区分(A): Lab24

イテレーション(R): Lab24

ステップ | 概要 | テストされた要件 | すべてのリンク | 添付ファイル | 関連付けられたオートメーション

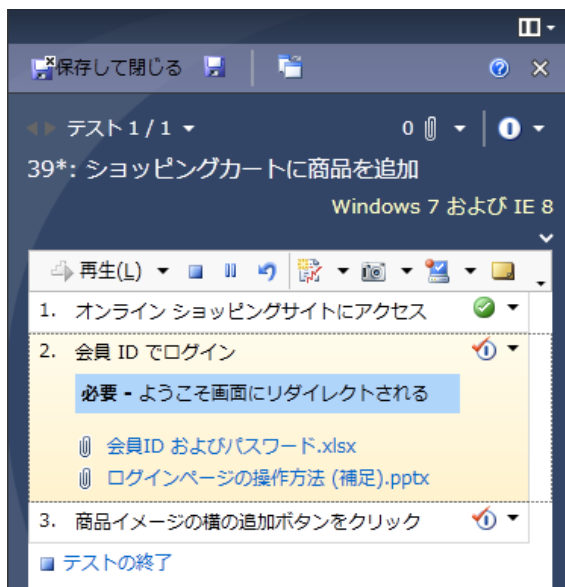
ステップの挿入 | 共有ステップの挿入 | パラメーターの挿入

操作	必要とされる結果
1. オンライン ショッピングサイトにアクセス	
2. 会員 ID でログイン	ようこそ画面にリダイレクトされる
商品イメージの横の追加ボタンをクリック	当該商品がカートに追加されることを確認

[ステップ] タブに実際にテストを行う手順を記入します。

[操作] 列には、テスト担当者が行う操作を、[必要とされる結果] 列には、操作の実行によって、成功とみなす結果を記述します。手順の番号の横に赤いチェックがついているものは検証が必要なステップを示しています。もし単なる操作の実行だけでいい場合には、[必要とされる結果] 列の記載は省略することができます。テスト ケースの入力時においては、Excel やタブ区切りのテキストをコピー & ペーストで貼り付けすると入力が容易になることもあります。

“クリップ” マークのボタンは、その手順に関連する補足資料として、ファイルを添付し関連付けることができます。例えばログインに関するステップであれば、ID やパスワードを記載したファイル、例えば商品入力であれば、商品コードやカテゴリを記載したファイルを添付して関連付けることによって、実際のテストの実行時に、その手順になると補足資料が表示されます。



テスト ステップ以外のテスト ケースにおける主な入力項目については、以下をご覧ください。

表: 主な入力項目の説明

フィールド または タブ名	説明								
タイトル	このテスト項目の名前です。テスト センター上に表示され、必要な場合にはテスト ケースの検索にも利用できるため一定のルールで名前付けすることをお勧めします。								
担当者	このテスト ケースを管理するオーナーです。実際のテストの実施担当者とは異なります。								
状態	<p>テスト ケースの現在の状態を示します。状態は以下の 3 つから選択します。</p> <table> <tr> <th>値</th><th>意味</th></tr> <tr> <td>デザイン</td><td>テスト ケースは設計中であり、レビューおよび承認はまだ行われていません。</td></tr> <tr> <td>準備完了</td><td>テスト ケースはレビューおよび承認されており、実行準備ができています。</td></tr> <tr> <td>終了</td><td>テストが終了したか、または不要になったことを示します。</td></tr> </table>	値	意味	デザイン	テスト ケースは設計中であり、レビューおよび承認はまだ行われていません。	準備完了	テスト ケースはレビューおよび承認されており、実行準備ができています。	終了	テストが終了したか、または不要になったことを示します。
値	意味								
デザイン	テスト ケースは設計中であり、レビューおよび承認はまだ行われていません。								
準備完了	テスト ケースはレビューおよび承認されており、実行準備ができています。								
終了	テストが終了したか、または不要になったことを示します。								
優先度	このテスト ケースの優先度を選択します。テスト センター上で並び替えやフィルターに利用できたり、クエリ ベースのテスト スイートの条件などにも利用できるため、一定のルールをあらかじめ決めておくことをお勧めします。								
オートメーションの 状態	<p>テストが手動テストか、自動テストかを定義します。手動テストの場合、“自動化なし” を選択します。</p> <table> <tr> <th>値</th><th>意味</th></tr> <tr> <td>自動化なし</td><td>手動テストのテスト ケースを示しています</td></tr> <tr> <td>計画済み</td><td>テスト ケースに自動テストを後で追加することを示します</td></tr> <tr> <td>自動</td><td>自動テストのテスト ケースを示しています。自動テストを関連付けると、この値が自動的に設定されます。</td></tr> </table>	値	意味	自動化なし	手動テストのテスト ケースを示しています	計画済み	テスト ケースに自動テストを後で追加することを示します	自動	自動テストのテスト ケースを示しています。自動テストを関連付けると、この値が自動的に設定されます。
値	意味								
自動化なし	手動テストのテスト ケースを示しています								
計画済み	テスト ケースに自動テストを後で追加することを示します								
自動	自動テストのテスト ケースを示しています。自動テストを関連付けると、この値が自動的に設定されます。								
[添付ファイル] タブ	<p>ここにはドキュメントなどを添付することができます。テストの詳細が書かれた仕様書や UI 操作手順書などの関連ドキュメントを添付することができます。</p> <p>添付ファイルは Test Runner によるテスト実施時において、クリップ型のアイコンから開くことができます。</p>								



手動テスト ケースの作成に関する詳細は以下の Web ページを参照ください。

- 方法: 手動テスト ケースを作成する

<http://msdn.microsoft.com/ja-jp/library/dd286659.aspx>

C) 共有ステップの作成

複数のテストで共通の一連のテスト手順は、共有ステップとして定義することができます。

例えば、アプリケーションへのログインは多くのケースにおいて共通の作業です。これに対し、すべてのテスト ケースで、「1. ユーザー ID を入力する」「2. パスワードを入力する」「3. [ログイン] ボタンをクリックする」という冗長なステップを入力するのは手間です。このような共通の処理は共通ステップとして定義し、複数のテスト ケースで共通的に利用することができます。この結果、テストの手順や内容が変更となった際に、変更にかかる労力を最低限に抑えることができ、テスト ケースの保守が容易となります。

テストの実施時においては、共有ステップ名の横の三角形をクリックすることで、共有ステップの実際のステップが展開されます。

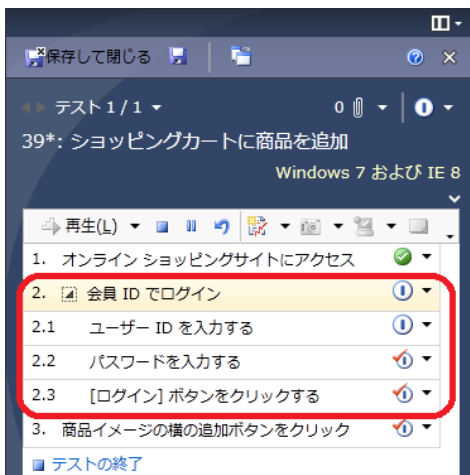


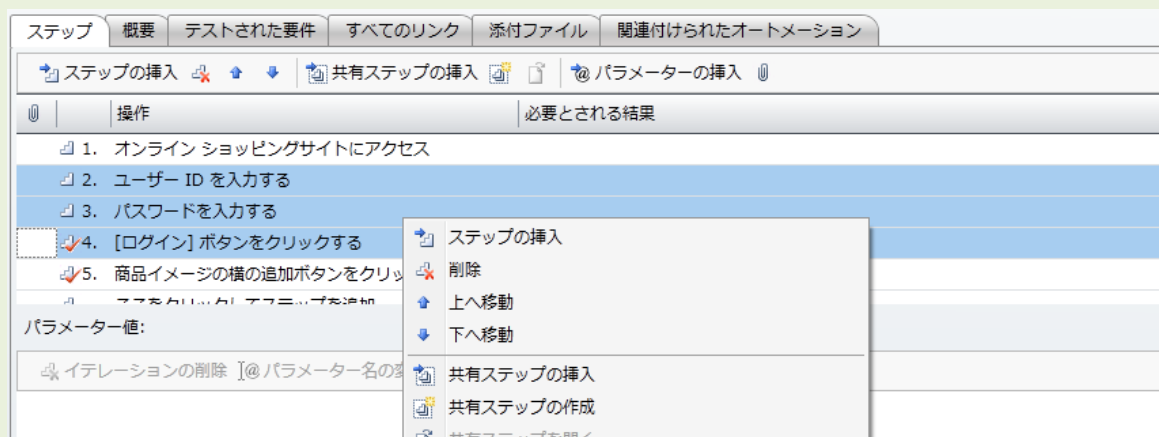
図: 共有ステップの実行 (赤い四角内は共有ステップ)

■ 共有ステップの新規作成

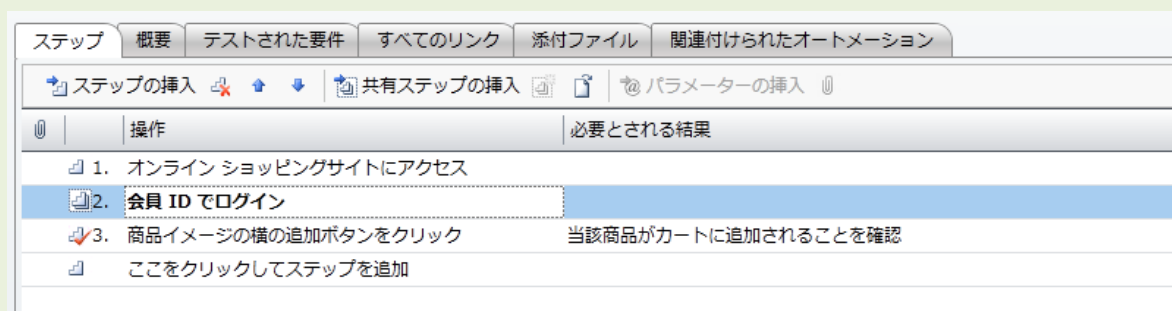
1. Test Manager 2010 を起動します。
2. [テスト センター] ビューより、[整理] - [共有ステップ マネージャー] を選択します。
3. [新規作成] ボタンをクリックします。
4. 共有ステップを定義します。ステップの定義自体は、通常のテスト ケースの作成と同じです。

■ 既存のステップを共有ステップに変換する

1. Test Manager 2010 を起動します。
2. 共有化するステップを含む、テスト ケースを開きます。
3. 共有化するステップを選択 ([CTRL] ボタンで複数選択することが可能)、右クリックして [共有ステップの作成] を選択します。



4. [共有ステップの作成] ダイアログに、ステップの名前を入力します。
5. 共有ステップが作成されたことを確認します。



6. 共有ステップの内容を確認したり修正するには、共有ステップで右クリックし [共有ステップを開く] を選択するか、[共有ステップ マネージャー] より、該当の共有ステップを探して開きます。

D) 異なるデータで複数回実行するテストの作成

テスト ケースによっては、同じテスト手順であっても、異なる条件やデータで複数回実施する必要があることもあります。

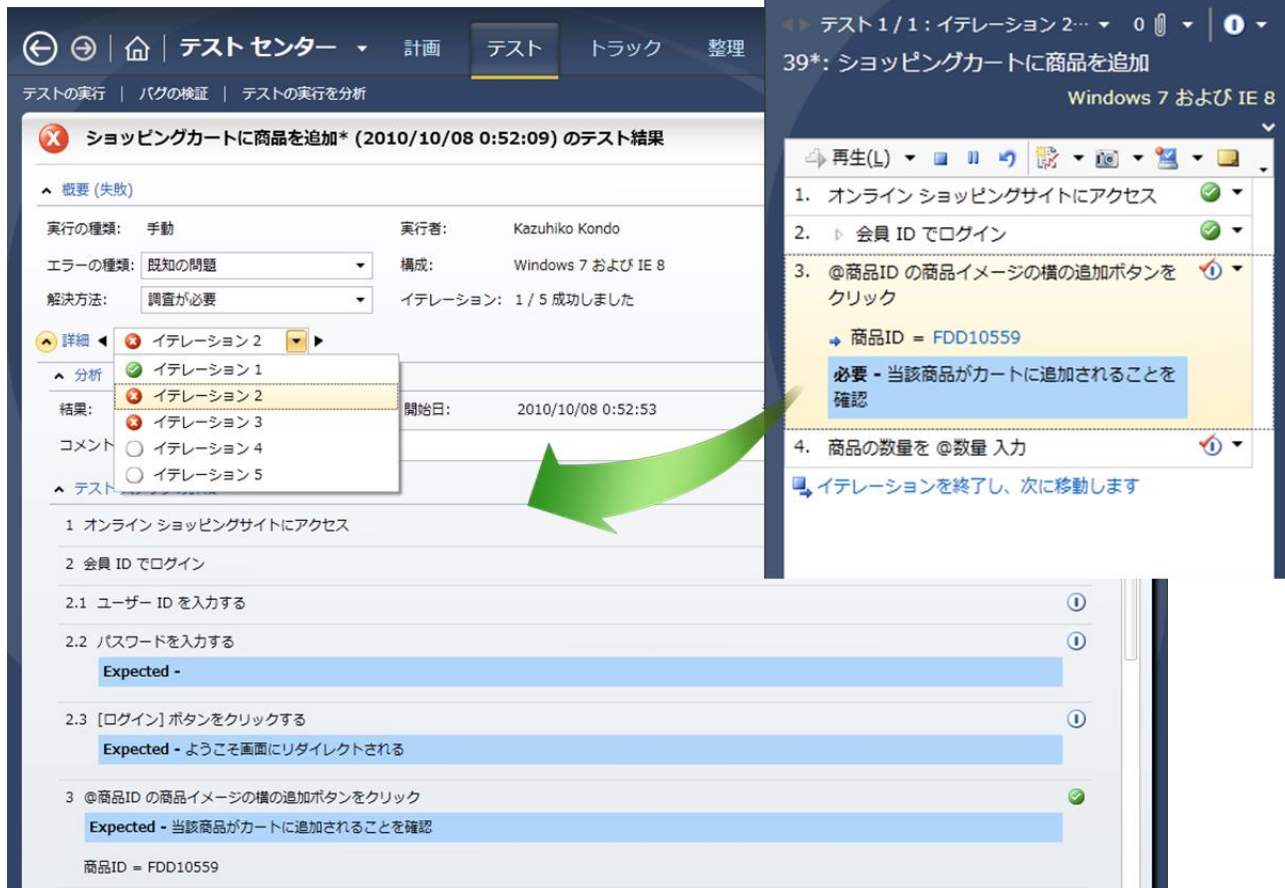
Test Manager 2010 では、条件をパラメーターとして与えることによって、異なるデータで複数回実施するテストを作成することが可能です。

ステップ	概要	テストされた要件	すべてのリンク	添付ファイル	関連付けられたオートメーション
<div> ステップの挿入 共有ステップの挿入 @パラメーターの挿入 </div>					
	操作	必要とされる結果			
1.	オンライン ショッピングサイトにアクセス				
2.	会員 ID でログイン				
3.	@商品ID の商品イメージの横の追加ボタンをクリック	当該商品がカートに追加されることを確認			
4.	商品の数量を @数量 入力	@数量 分、商品が追加されていることを確認			
	ここをクリックしてステップを追加				
パラメーター値:					
イテレーションの削除 [@パラメーター名の変更 ✕ @パラメーターの削除					
商品ID	数量				
PDG00112	4				
FDD10559	12				
GHK00368	5				

テスト ステップの定義時に [パラメーターの挿入] ボタンをクリックします。[パラメーター] の挿入ダイアログが表示されますので、パラメーター名を入力すると「@パラメーター名」という定義がテスト ステップに記述されるのが確認できます。同時に下部にパラメーター値を入力するセルが表示されます。各行は 1 回のテストで使用する値です。5 パターンのテストが必要な場合には、5 行分テスト データを定義します。

なお、共有ステップにもパラメーターを設定することができます。共有ステップのパラメーターの値の設定は、共有ステップのテスト ケースではなく、共有ステップを挿入した各テスト ケース側で定義します。ただし、共有ステップには値の組み合わせを 1 つだけ定義することができます。これは各テスト ステップ側で値を指定しなかった場合に、デフォルト値として利用されます。

パラメーターを持つテスト ケースは、イテレーションという形で繰り返し実行されます。5 行パラメーター値を定義した場合には、5 回テストを実施します。最終的に各結果もイテレーションごとに表示され、テストの全体結果としてはいずれかのイテレーションが失敗した場合には、“失敗” と表示されます。



E) 作成済のテストステップの追加

Test Manager 2010 および Visual Studio 2010 で既に作成済のテスト ケースをテスト ケースに追加することも可能です。

■ 既存のテスト ケースをテスト スイートに追加する

1. Test Manager 2010 を起動します。
2. [テスト センター] ビューより、[計画] メニューをクリックします。
3. 任意のテスト スイートを選択し、画面右側の [追加] ボタンをクリックします。
4. 検索クエリが表示されますので、検索条件を設定し（通常は既定のままで問題ありません）[実行] ボタンをクリックします。
5. 表示されたテスト ケースを選択（[CTRL] ボタンで複数選択することも可能です）し、[テスト ケースの追加] ボタンをクリックします。

F) テスト構成を個別に変更

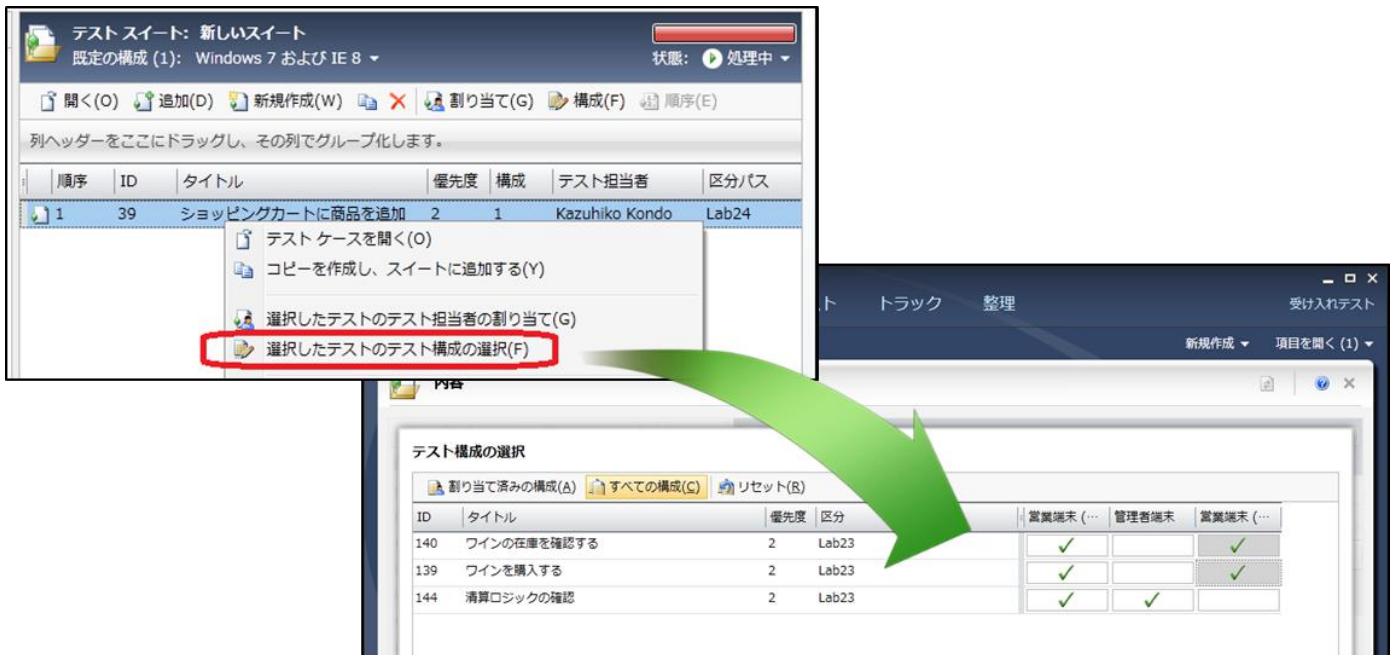
テスト ケースによっては、既定の構成とテスト実施の対象が異なる場合があります。この場合、テスト ケースごとにテスト構成を変更することができます。

テスト構成を変更するには、以下の2つの方法があります。

- テスト ケースを選択（[CTRL] ボタンで複数選択することも可能です）し、右クリックから [選択したテストのテスト構成の

選択] を選択。または画面右側の [構成] ボタンをクリックします。

- テスト スイートを選択し、右クリックから [すべてのテストのテスト構成の選択] を選択します。この場合、テスト スイート内のすべてテスト ケースが表示され、個別にテスト構成を変更することが可能です。



詳細については以下の Web ページを参照ください。

- 方法: テスト計画を作成する - 特定のテスト ケースの構成を更新

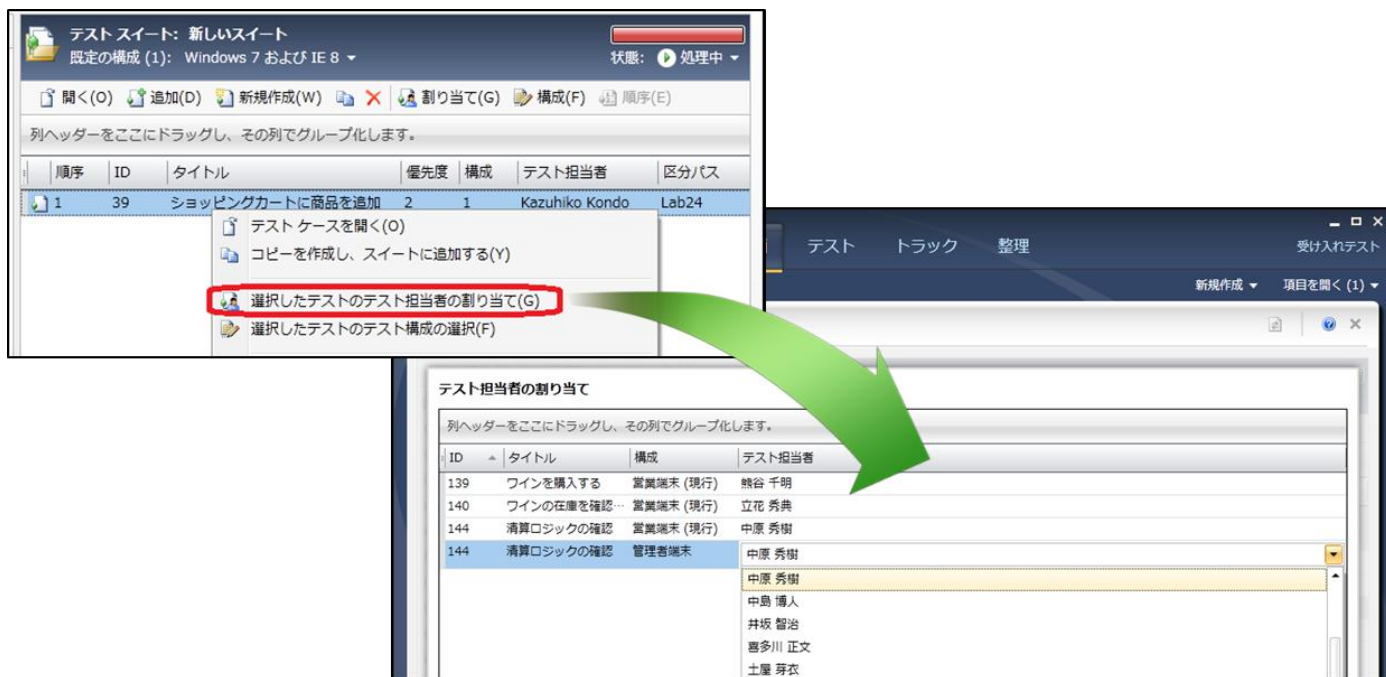
<http://msdn.microsoft.com/ja-jp/library/dd286583.aspx#UpdateConfigurations>

G) テスト担当者を変更

テストの担当者を変更します。テスト担当者を割り当てると、テストの実施時や分析時にテスト担当者でフィルターをかけて管理することができます。テスト担当者の割り当てを変更するには、以下の 2 つの方法があります。

- テスト ケースを選択 ([CTRL] ボタンで複数選択することも可能です) し、右クリックから [選択したテストのテスト担当者の割り当て] を選択。または画面右側の [割り当て] ボタンをクリックします。
- テスト スイートを選択し、右クリックから [すべてのテストのテスト担当者の割り当て] を選択します。この場合、テスト スイート内のすべてテスト ケースが表示され、個別にテスト担当者の割り当てを変更することが可能です。

テストの担当者と、テスト ケースの所有者は異なります。この手順ではテスト ケースの所有者 (作業項目上の “担当者” フィールド) は変更されません。



詳細については以下の Web ページを参照ください。

- 方法: テスト計画を作成する - テストをテスト担当者に割り当て

<http://msdn.microsoft.com/ja-jp/library/dd286583.aspx#AssignTests>

H) テスト ケースの優先度を設定

テスト ケースの優先度として、テスト スイート内の順序を指定します。テスト スイートごとテストを実施した際には、ここで指定した順番でテストが実施されます。

順序を変更するには該当のテスト スイートを右クリックし [テストの順序指定] を選択します。要件からテスト スイートを作成した場合には、順序は変更できません。



詳細については以下の Web ページを参照ください。

- 方法: テスト計画を作成する - テスト スイート内のテスト ケースの整理
<http://msdn.microsoft.com/ja-jp/library/dd286583.aspx#OrderTests>

III. テストの実施

テストの実施フェーズにおいては、実際に作成した手動テスト ケース、自動テスト ケースを実際に実行します。

1. テストの状況確認

テストスイートごとのテストの実施状況は、[テスト] ビューの画面右上に表示されるバーで把握することができます。マウス ポインターをバーの上に置くとそれぞれの状態におけるテスト項目数が表示されます。



また、テスト一覧は状態別にグループ化されて表示されます。各状態の意味は以下の通りです。

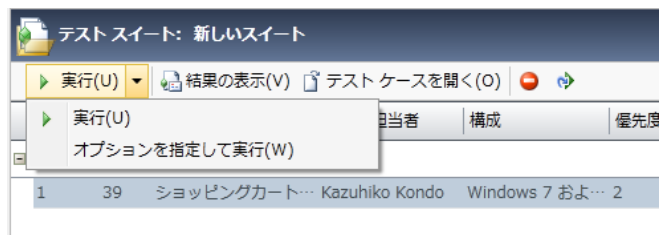
状態	アイコン	意味
成功	✓	テストは成功しました。
失敗	✗	テストは失敗しました。
ブロック	⏸	テストは一時的に無効化されています。
アクティブ	ⓘ	実施準備ができています。一度も実施されていないか、ブロックが解除されたか、問題が修正され再実施の必要性があるテストです。
処理中	▶	テストは現在実施中です。

2. 自動テストの実施

テスト計画で作成した自動テスト ケースを実施します。

[テスト] メニューより該当のテスト スイートを選択すると、テスト計画で作成したテスト ケースが画面右側に一覧化されます。
[構成の定義] で定義したテスト マトリクスに従い、構成ごとにテスト ケースが複製されているのがご覧いただけます。
自動テストを実施するには、テスト ケースを選択し、上部の [実行] ボタンをクリックします。テスト環境やテスト設定を変更す

る場合は、[実行] ボタン右の下矢印をクリックし、[オプションを指定して実行] を選択します。



テストを実行すると [テストの実行] ページが表示されます。テストはリアルタイムに実行されているので、[最新の情報に更新] ボタンをクリックし、経過を取得ください。



テストの実行結果は、いったんウィンドウを閉じて、後で参照することもできます。またテストの結果を分析し、エラーの種類やその後の対応を追記することもできます。手順については「テスト実行結果の分析」を参照ください。

自動テストの詳細については以下の Web ページを参照ください。

■ 自動テストの実行

<http://msdn.microsoft.com/ja-jp/library/dd286580.aspx>

3. 手動テストの実施

テスト計画で作成した手動テスト ケースを実施します。

実行の手順は自動テストと同じです。[テスト] メニューより該当のテスト スイートを選択すると、テスト計画で作成したテスト ケースがテスト マトリクス数分作成されています。手動テストを実施するには、テスト ケースを選択し、上部の [実行] ボタンをクリックします。テスト環境やテスト設定を変更する場合は、[実行] ボタン右の下矢印をクリックし、[オプションを指定して実行] を選択します。

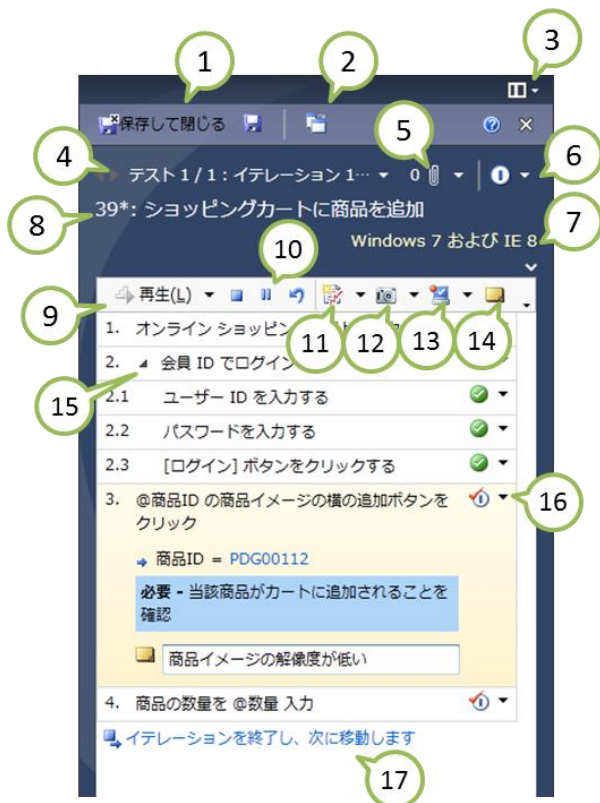
自動テストと異なるのは次の手順からです。


ツールが長細くなり、画面左側に表示されるのがご覧いただけます。

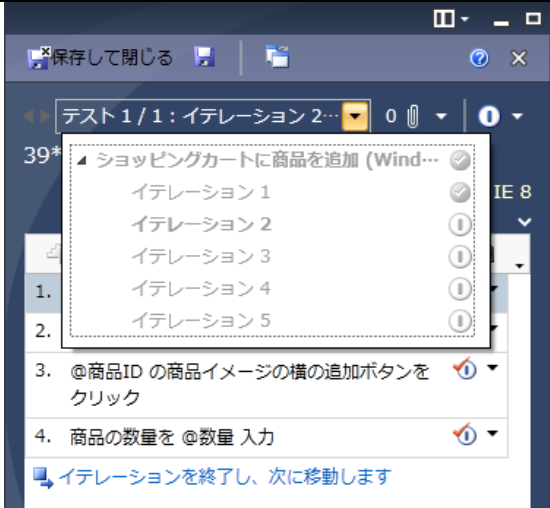
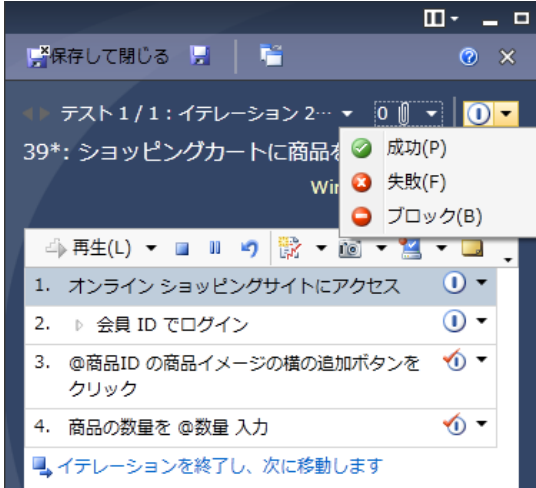
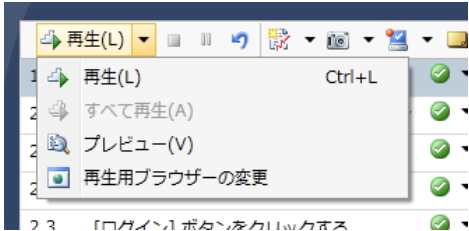











このツールは Test Runner と呼ばれ、手動テストを操作するものです。Test Runner には手動テスト ケースで定義したステップが表示されますので、テスト担当者はその手順に従い、テストを実行します。

Test Manager の操作方法について説明します。



番号	説明										
1	Test Runner を閉じます。テストが途中の場合は、テスト結果は“失敗”となります。										
2	<p>Test Runner から、Test Manager 2010 に切り替えます。この場合、テストは継続状態となります。</p> <p>Test Manager 2010 から Test Runner に戻るには、メニュー下部の [テスト ランナーに戻る] ボタンをクリックします。</p> 										
3	<p>Test Runner の画面の表示位置を指定します。既定は [左にドッキング] です。</p> <table border="1"> <thead> <tr> <th>値</th><th>意味</th></tr> </thead> <tbody> <tr> <td>左にドッキング</td><td>Test Runner を画面左に固定します。</td></tr> <tr> <td>右にドッキング</td><td>Test Runner を画面右に固定します。</td></tr> <tr> <td>フローティング</td><td>Test Runner を固定せず、ウィンドウの位置を自由に変えることができます。</td></tr> <tr> <td>常に手前に表示</td><td>Test Runner をフローティングにした際、他のウィンドウがアクティブになっても、Test Runner が常に最前面に表示されるようにします。</td></tr> </tbody> </table>	値	意味	左にドッキング	Test Runner を画面左に固定します。	右にドッキング	Test Runner を画面右に固定します。	フローティング	Test Runner を固定せず、ウィンドウの位置を自由に変えることができます。	常に手前に表示	Test Runner をフローティングにした際、他のウィンドウがアクティブになっても、Test Runner が常に最前面に表示されるようにします。
値	意味										
左にドッキング	Test Runner を画面左に固定します。										
右にドッキング	Test Runner を画面右に固定します。										
フローティング	Test Runner を固定せず、ウィンドウの位置を自由に変えることができます。										
常に手前に表示	Test Runner をフローティングにした際、他のウィンドウがアクティブになっても、Test Runner が常に最前面に表示されるようにします。										
4	<p>現在実行しているテストを表示しています。</p> <p>[異なるデータで複数回実行するテストの作成] を定義した場合、テスト パターンがイテレーションとして表示されます。</p>										

									
5	テストに関連付けられているファイルを表示したり、追加することができます。								
6	<p>テスト ケース全体の結果を指定します。</p> 								
7	現在テストしている、テストの構成を表示します。								
8	現在のテスト ケース名を表示します。								
9	<p>[再生] ボタンは、既に記録済の UI 操作を再現します。UI 操作を記録する場合、テスト設定の定義で “操作” が有効になっている必要があります。</p>  <table border="1"> <thead> <tr> <th>操作</th><th>意味</th></tr> </thead> <tbody> <tr> <td>再生</td><td>特定のステップに関連付けられている UI 操作を再現します。</td></tr> <tr> <td>すべて再生</td><td>記録されている UI 操作をすべて再生します。</td></tr> <tr> <td>プレビュー</td><td>再生する手順を表示します。これを選択すると [クリップのプレビュー] ダイアログ</td></tr> </tbody> </table>	操作	意味	再生	特定のステップに関連付けられている UI 操作を再現します。	すべて再生	記録されている UI 操作をすべて再生します。	プレビュー	再生する手順を表示します。これを選択すると [クリップのプレビュー] ダイアログ
操作	意味								
再生	特定のステップに関連付けられている UI 操作を再現します。								
すべて再生	記録されている UI 操作をすべて再生します。								
プレビュー	再生する手順を表示します。これを選択すると [クリップのプレビュー] ダイアログ								

		グ ボックスにテキスト形式で手順が表示され、これからどのような操作が再生されるか事前に確認することができます。								
	再生用ブラウザの変更	UI の操作を再現するブラウザを指定します。サポートされるブラウザについては、以下の Web ページを参照ください。 http://msdn.microsoft.com/ja-jp/library/dd380742.aspx								
10	テストを操作します。 <table><tr><th>アイコン</th><th>意味</th></tr><tr><td></td><td>テストを終了します。</td></tr><tr><td></td><td>テストを一時停止します。</td></tr><tr><td></td><td>テストをリセットし、やり直します。</td></tr></table>		アイコン	意味		テストを終了します。		テストを一時停止します。		テストをリセットし、やり直します。
アイコン	意味									
	テストを終了します。									
	テストを一時停止します。									
	テストをリセットし、やり直します。									
11	バグ票を作成します。 <table><tr><th>操作</th><th>意味</th></tr><tr><td>バグの作成</td><td>Team Foundation Server の作業項目である “バグ” を作成します</td></tr><tr><td>調査用バグの作成</td><td>問題の解析を詳細に行うためのバグ票です。バグに含める操作の時間範囲を指定できます。操作は操作ログに記録され、このデータを手動テスト ケースを作成するために使用できます。調査用バグでは、時間範囲を指定することで、バグに含まれる診断データ アダプターのデータの量を減らすことができます。</td></tr><tr><td>既存バグの更新</td><td>既存のバグ票を更新します。この操作を選択すると、クエリが表示されますので、該当のバグ票を検索し、選択します。 クエリの詳細は「要件、バグ、タスクなどの管理」を参照ください。</td></tr></table>		操作	意味	バグの作成	Team Foundation Server の作業項目である “バグ” を作成します	調査用バグの作成	問題の解析を詳細に行うためのバグ票です。バグに含める操作の時間範囲を指定できます。操作は操作ログに記録され、このデータを手動テスト ケースを作成するために使用できます。調査用バグでは、時間範囲を指定することで、バグに含まれる診断データ アダプターのデータの量を減らすことができます。	既存バグの更新	既存のバグ票を更新します。この操作を選択すると、クエリが表示されますので、該当のバグ票を検索し、選択します。 クエリの詳細は「要件、バグ、タスクなどの管理」を参照ください。
操作	意味									
バグの作成	Team Foundation Server の作業項目である “バグ” を作成します									
調査用バグの作成	問題の解析を詳細に行うためのバグ票です。バグに含める操作の時間範囲を指定できます。操作は操作ログに記録され、このデータを手動テスト ケースを作成するために使用できます。調査用バグでは、時間範囲を指定することで、バグに含まれる診断データ アダプターのデータの量を減らすことができます。									
既存バグの更新	既存のバグ票を更新します。この操作を選択すると、クエリが表示されますので、該当のバグ票を検索し、選択します。 クエリの詳細は「要件、バグ、タスクなどの管理」を参照ください。									
12	画面キャプチャを取得し、そのステップに関連付けます。 画面キャプチャは PNG 形式でファイルとして保存され、実際の画面キャプチャで問題の状況を視覚的に補足したり、アプリケーションが前後でどう修正されたか、修正の履歴管理を行うことができます。 <table><tr><th>操作</th><th>意味</th></tr><tr><td>四角形キャプチャのスクリーンショット</td><td>任意の領域の画面の画面キャプチャを取得します。 この操作を選択すると、画面がグレーになりますので、マウスをドラッグしながら領域を指定します。</td></tr><tr><td>全画面キャプチャのスクリーンショット</td><td>画面全体の画面キャプチャを取得します。</td></tr><tr><td>ウィンドウ キャプチャのスクリーンショット</td><td>特定ウィンドウの画面キャプチャを取得します。</td></tr></table>		操作	意味	四角形キャプチャのスクリーンショット	任意の領域の画面の画面キャプチャを取得します。 この操作を選択すると、画面がグレーになりますので、マウスをドラッグしながら領域を指定します。	全画面キャプチャのスクリーンショット	画面全体の画面キャプチャを取得します。	ウィンドウ キャプチャのスクリーンショット	特定ウィンドウの画面キャプチャを取得します。
操作	意味									
四角形キャプチャのスクリーンショット	任意の領域の画面の画面キャプチャを取得します。 この操作を選択すると、画面がグレーになりますので、マウスをドラッグしながら領域を指定します。									
全画面キャプチャのスクリーンショット	画面全体の画面キャプチャを取得します。									
ウィンドウ キャプチャのスクリーンショット	特定ウィンドウの画面キャプチャを取得します。									
13	現在のテスト環境の状態をスナップショットとして取得します。 Lab Management 2010 を利用し、テスト環境を仮想環境として利用している際に利用できます。									
14	各ステップにコメントを記述することができます。バグ票を作成して問題を報告する際に、テスト担当者が各ステップの補足を記載する際に利用します。									
15	テスト手順です。テスト ケースの作成時に定義したものが表示されます。 共有ステップの場合、三角の印をクリックすると、共有ステップの詳細手順が展開されます。									
16	各ステップのテスト結果を指定します。									

- テスト担当者による実際の画面の操作。これを取得するには「テスト設定の定義」で、[ビデオ レコーダー] が有効になっている必要があります。画面の操作はビデオ ストリーミング (Windows Media 形式) で記録されており、そのストリーミング ファイルをバグ票に添付することができます。テストの手順ごとに時間のハイパーリンクがバグ票に記述され、そのハイパーリンクをクリックするとその手順を実施した時間のストリーミングが写し出されます。
- 任意のタイミングで画面キャプチャを取得することができます。「手動テストの実施」の No.12 を参照ください。実際の画面キャプチャで問題の状況を視覚的に補足したり、アプリケーションが前後でどう修正されたか、修正の履歴管理を行うことができます。
- テストを実施しているシステムの情報を記録することができます。これを取得するには「テスト設定の定義」で [システム情報] が有効になっている必要があります。そのテスト環境の、OS の種類や、搭載メモリ容量、CPU の規格など、問題の分析に有用なシステムの情報を渡すことができます。
- Visual Studio 2010 の新機能 IntelliTrace のデータとして、デバッグの履歴をファイルとして受け渡すことができます。これを取得するには「テスト設定の定義」で [IntelliTrace] が有効になっている必要があります。IntelliTrace はデバッグの履歴を保持することができる機能であり、テストを実施した際、どのルート (呼び出し) が行われ、その際にどのような処理 (ファイル IO、データベース アクセス等) が行われたかをデバッグ情報として取得することができます。これまでこのような情報を必要とする場合、ダンプ ファイルを取得していました。ダンプ ファイルはその瞬間の情報しか保持しないのに対し、IntelliTrace はどのようなコードが実行されたかを見ることができるため、実際に処理を実行していない開発者がコードを追跡する際に強力な情報となります。
- Visual Studio Lab Management 2010 で仮想環境を構築した場合、問題が発生した瞬間のスナップショットを担当の開発者に引き渡すことができます。「手動テストの実施」の No.13 を参照ください。

IV. 問題の追跡

1. テスト実行結果の分析

テスト ケースを右クリックし、[結果の表示] を選択すると、直近のテスト結果が表示されます。

実施したテストの結果すべてを確認する場合には、[テスト センター] ビューより [テスト] - [テストの実行の分析] を選択します。手動テストの実行結果を表示する場合には、[ビュー] で [手動実行の表示] を選択状態にしてください。



該当のテストを選択し、[開く] ボタンで詳細を確認します。



[テスト] カテゴリの中で、該当のテストを探します。テスト結果の詳細を確認する場合、[結果の表示] ボタンをクリックすると、テストのより詳細な結果が表示されます。ここからテストの実施時に取得した各種診断データ ファイルにアクセスすることができます。

[エラーの種類] ボタンをクリックし、エラーの原因を選択します。既定値は“なし”、“回帰”、“新しい問題”、“既知の問題”、“不明” から選択します。その割合は、[結果の概要] カテゴリの [理由別の失敗したテスト] に示されます。同様に、[解決策] ボタンをクリックし、解決のための対策を“なし”、“調査が必要”、“テストの懸案事項”、“製品の懸案事項”、“構成の懸案事項” から選択します。その割合は、[結果の概要] カテゴリの [分析別の失敗したテスト] に示されます。

分析の結果、最終的にバグであることが判明した際には、[バグの作成] ボタンをクリックすると、ここからバグ票を作成することができます。

詳細については以下の Web ページを参照ください。

■ 方法: Microsoft テスト マネージャーを使用してテストの実行を分析する

<http://msdn.microsoft.com/ja-jp/library/dd553274.aspx>

2. ビルドの再割り当て

問題を特定した後、開発者は、バグ票とともに引き継がれたさまざまな情報にもとづき、アプリケーションを修正します。アプリケーションの修正理由はバグの修正だけではなく、次に優先度の高い要件の実装や、仕様変更に基づく修正など様々な理由があります。テストに新しいビルドを割り当てるには、[テスト センター] ビューより、[計画] - [プロパティ] を選択、テスト計画のプロパティより、[使用中のビルド] の [変更] ボタンをクリックします。「ビルドの定義」も参考にしてください。

[ビルドの割り当て] ページが表示されますので、[使用可能なビルド] より該当のビルドを選択し、[計画に割り当てる] ボタンをクリックします。



新しいビルドを適用することによって様々な影響が考えられますが、Test Manager 2010 では影響度を把握するための機能を搭載しています。

A) ビルドの修正理由の把握

発行したバグ票はもちろん、関連する要件やタスクなどを Test Manager 2010 で管理することができます。したがって、発生した問題が現在どのような状況にあるかを一元管理し、把握することができます。

しかし、開発者はバグ票とともに引き継がれた様々な情報をもとに、問題を解析し、アプリケーションに必要な修正を行います。その結果、最新のビルドが次々とできあがりますが、Test Manager 2010 から問題を一元管理することができるとはいえ、その最新ビルドがどの問題が修正された結果なのかを追って探すのは一苦労です。特にバグの検出とその修正のサイクルは頻繁に行われる上、1 つのビルドに 1 つのバグのみが解決されているとは限りません。さらに場合によっては新しい要件による実装がビルドに組み込まれていることも考えられます。

Test Manager 2010 は、その最新のビルドに関連付けられた要件、バグ、タスクなどを洗い出し、提示することができます。これによってそのビルドがどのような理由により修正されたものかを、異なる複数の理由が組み合わさったものでも判別するのが容易となります。



詳細については、以下の Web ページを参照ください。

■ バグ修正、新機能、要件のあるビルドの確認

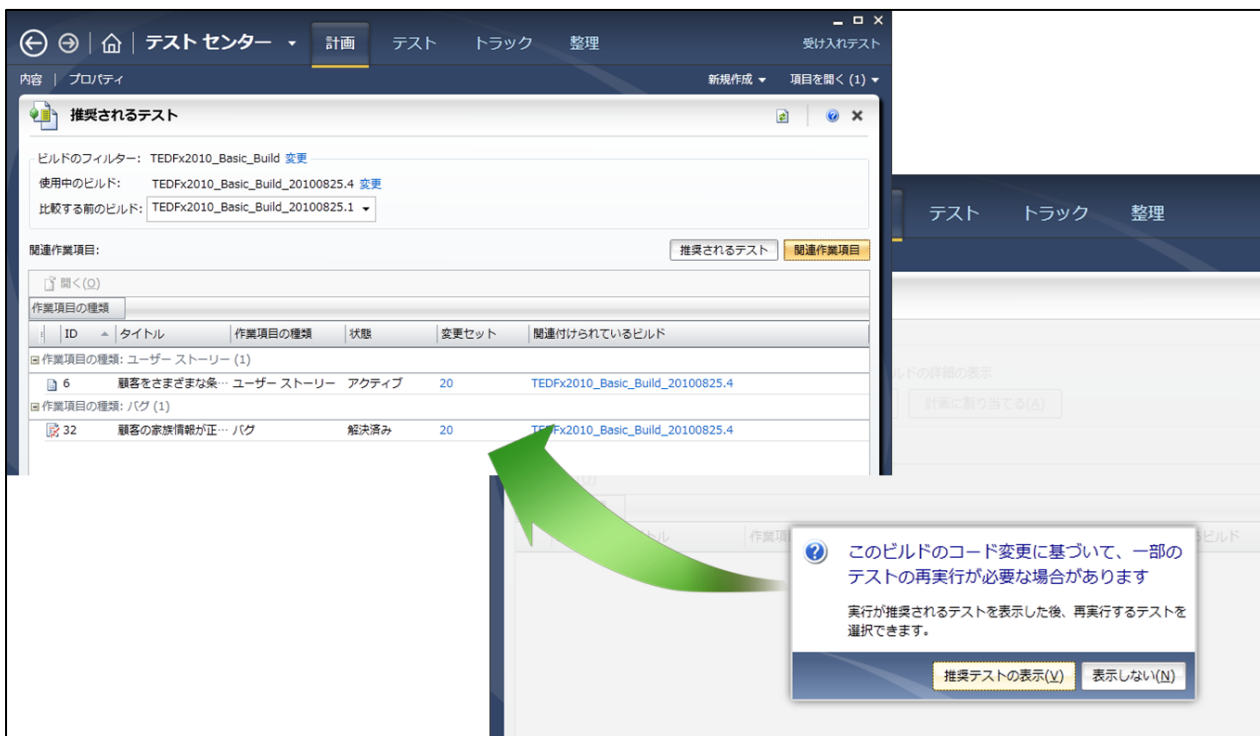
<http://msdn.microsoft.com/ja-jp/library/dd286586.aspx>

B) 推奨されるテストの実施

既に実施済で問題がなかったテストがソフトウェアの修正後も引き続き期待通りに動くとは限りません。そのため回帰テストが必要となりますが、数多くのテストをすべて再度実行するのは非効率です。“推奨されるテスト” はビルドの変更により、テストで実施したコードの動作に影響があるかもしれない場合に、そのテストをテスト担当者に提示します。

推奨されるテストを特定するには、あらかじめ Test Runner を使用して手動テストを実行するか、自動テストを実行する必要があります。また、これらのテストを実行するときは、[テスト設定の定義] で“テスト影響データ”を収集するように指定しておく必要があります。このテスト影響データには、テストの実行時にどのメソッドが呼び出されるかについての情報が含まれています。このメソッドとテスト ケースの関連付けは、メソッドに加えられた変更に基づいて、どのテスト ケースを再度実行する必要があるのかを判断するために保存されます。

また、テスト影響データは、テストが実行されて、成功したときにのみ収集されます。この条件が満たされた場合にのみ、テスト全体が実行されたときに使用されるメソッドについて正確なデータを収集できます。テストが失敗した場合は、テストの実行に使用されるメソッドのデータのうち一部しか収集されず、部分的なデータであるため正確ではありません。



詳細については、以下の Web ページを参照ください。

■ コード変更により影響を受けたテストの実行の推奨

<http://msdn.microsoft.com/ja-jp/library/dd286589.aspx>

3. 再テストの実施

最新のビルドが再割り当てされ、報告していたバグの問題が修正された後は、問題が解決しているか再テストをします。Test Manager 2010 では、報告していたバグの再検証が容易となっています。

[テスト センター] ビューより、[テスト] - [バグの検証] を選択すると、発行されたバグを一覧で確認することができます。

自分が担当となっているバグ票は [現在のユーザーに割り当て済み] より、自分が発行したバグ票は [現在のユーザーが作成] で確認することができます。



該当のバグを選択し [検証] ボタンをクリックすると、該当のテストを再実行することができます。

テストが終了し、Test Runner を閉じると最終的にバグ票のステータスを解決済みにするかを選択する画面が表示されます。再テスト実施後に作業項目を異なるツールで開くといった手間が省けます。



4. 要件、バグ、タスクなどの管理

Test Manager 2010 では、要件やテストを追加したり、バグ票やタスクを検索する際に、作業項目のクエリを利用します。
[テスト センター] ビューより、[トラック] - [クエリ] を選択すると、要件、バグ、タスク、リスクといった様々な作業項目を、特定の条件に絞り込んで検索し、状況を確認したり、最新情報を書き込んだりすることができます

The screenshot shows the 'クエリ: 担当しているタスク' (Query: Tasks I am responsible for) window in Test Manager 2010. The window has a dark blue header with navigation buttons and tabs. The main area is divided into two panes. The left pane shows a tree view of queries, with 'マイ クエリ' (My Queries) selected. The right pane shows a query builder and a list of results.

Query Builder:

おおよび/または	フィールド	演算子	値
	チーム プロ...	=	@Project
AND	作業項目の種類	=	テスト ケース
AND	担当者	=	@Me
AND	優先度	<	3

* 句を追加するにはここをクリックします

Actions: 実行(R), 列のオプション(O), 開く(Q), コピーの作成(C)

列ヘッダーをここにドラッグし、その列でグループ化します。

Results Table:

ID	タイトル
4	夜間バッチ - 営業マスター更新
3	夜間バッチ - 販売マスター更新
2	夜間バッチ - 契約マスター更新
1	顧客の基本情報からターゲット顧客を絞り込む

クエリ結果: 4 個の結果が見つかりました。

Test Manager 2010 は Team Foundation Server のチーム プロジェクトで管理されている作業項目のクエリを利用しています。同じものが Visual Studio 2010 のチーム エクスプローラーで確認することができます。したがって、Visual Studio 2010 で独自に作成されたクエリを利用することもできますし、Test Manager 2010 側でクエリを作成することもできます。クエリの設定方法は、Visual Studio 2010 のチーム エクスプローラーで作成する方法と同じです。

クエリは SQL 分の WHERE 句に近いです。(厳密には異なりますが) イメージとしては、作業項目が 1 つのテーブルに格納され、各列の条件を指定することで、該当の作業項目を抽出することができます。

以下の図は、同じく厳密には正確ではありませんが、Team Foundation Server のクエリを理解するうえでわかりやすいので参考にしてください。

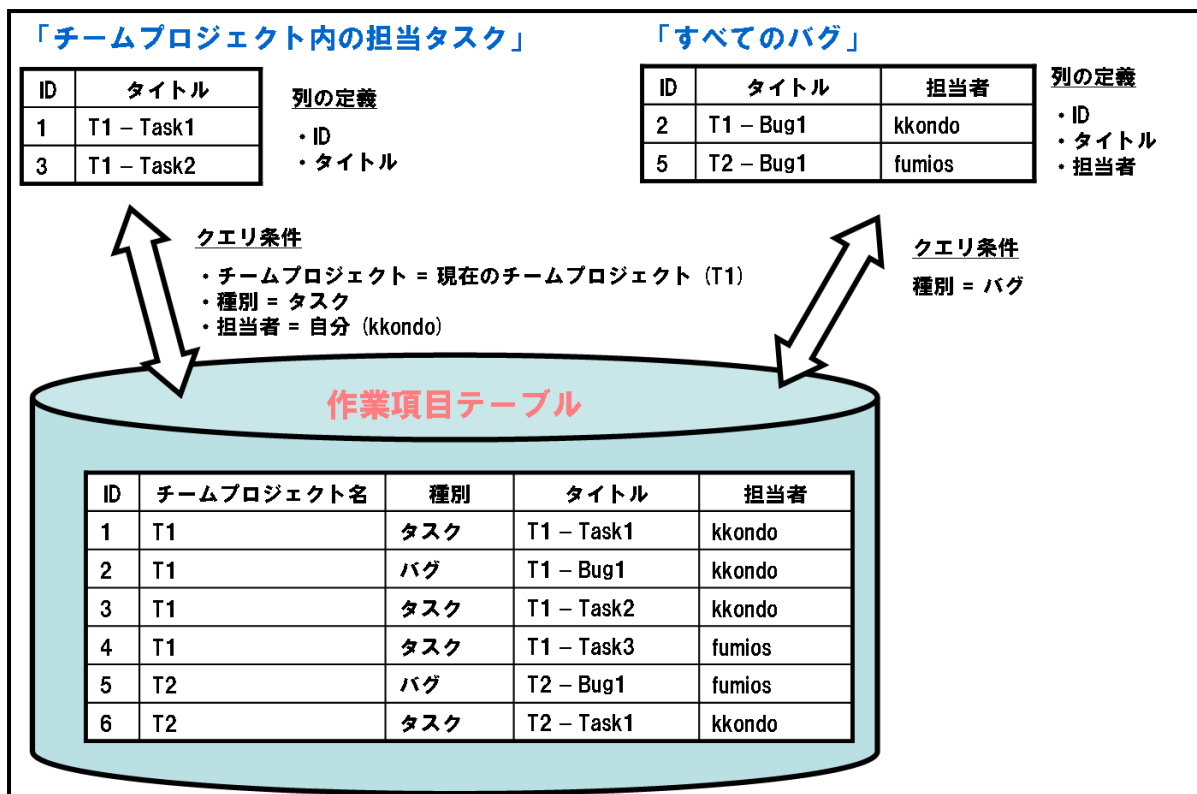


図: クエリのイメージ図

クエリには、チーム全体で共有する [チーム クエリ] と、担当者のみが利用する [マイ クエリ] の 2 種類があります。必要に応じてどちらのクエリを作成するかを決めてください。

クエリを定義する際には、「クエリ条件の定義」と「結果を表示する列の定義」の 2 つに分けることができます。

クエリ条件は、「フィールド」、「演算子」、「値」の組み合わせで定義します。2 つ以上の条件からなる場合は、「および/または」、で排他的条件なのか、そうではないのか (AND / OR) を指定します。

また、フィールドに応じて、以下の変数が利用可能です。

- @Me: 現在のユーザー アカウント
- @Project: 現在のチームプロジェクト
- @Today: 現在日時

列の定義では、表示する列、表示順番、列の幅が指定可能です。クエリ結果を右クリックし [列のオプション] を選択します。

V. まとめ

ソフトウェア開発におけるテスト担当者と開発者間の効率的な情報伝達は、最終的なソフトウェアの品質を左右する重要な要素です。Test Manager 2010 は、テストの計画、実施、問題の追跡にわたる広範囲なテスト プロセスをカバーする専用ツールです。特に受け入れテストや、開発プロジェクトの最終工程である総合テストで活用することでテストの実施と発生した問題の解決を効率化し、最終的なソフトウェアの品質向上に貢献することができます。

Microsoft[®]

www.microsoft.com/japan/msdn/vstudio