

Visual Studio を知る

- 製品概要
- エディション間の機能比較
- 最適な開発環境の構築 (MSDN Subscription とは)

Visual Studio を買う

- 評価版
- お得な買い方
- 参考価格
- ライセンス情報
- オンライン販売
- 販売パートナー



Visual Studio Team System 2008 新機能の紹介

はじめに

このドキュメントでは、Visual Studio Team System 2008 で新しく追加された機能と強化された機能の概要を説明します。Visual Studio Team System 2008 はサーバー製品となる Team Foundation Server、ならびにクライアント製品となる Architecture Edition、Database Edition、Development Edition、Test Edition (および、これら 4 つの製品の機能を統合し提供する Team Suite) から構成されます。

以下では、これら各製品における Version 2005 から Version 2008 の変更点について説明します。

- **Team Foundation Server**
 1. [Team Foundation ビルド](#)
 2. [Team Foundation ソース管理](#)
 3. [Team Foundation 作業項目トラッキング](#)
 4. [Team Foundation Server の管理](#)
- **Architecture Edition**
 1. [トップダウン方式を使用したアプリケーション システムのデザイン](#)
 2. [.NET Web サービス エンドポイントの WSDL ファイルへの準拠](#)
 3. [ASP.NET Web アプリケーション プロジェクトの生成](#)
 4. [カスタム プロトタイプの保存、インポート、およびエクスポート](#)
 5. [複数のバージョンからの .NET Framework の選択](#)
 6. [複数のバージョンからの Office の選択](#)
- **Database Edition**
 1. [Transact-SQL のスタティック コード分析](#)
 2. [依存関係の表示](#)
 3. [リファクタリング](#)
 4. [データの生成](#)
 5. [MSBuild タスク](#)
 6. [その他のツール](#)
- **Development Edition**
 1. [コード分析](#)
 2. [コードメトリックス](#)
 3. [プロファイル ツール](#)
 4. [ロードテストと Web テストの Visual Studio Team Suite への統合](#)
- **Test Edition**
 1. [コードのテスト方法](#)
 2. [Web テスト](#)
 3. [ロードテスト](#)



1. Team Foundation ビルド

- **ビルド定義**
Team System 2008 の「ビルド定義」は、「ビルドの種類」に代わるものです。「ビルドの種類」とは違い、「ビルド定義」は VSTS クライアントのユーザー インターフェイスを利用して変更できます。「ビルド定義」では、バージョン管理のワークスペースもサポートされています。ローカルパスを指定し、バージョン管理で指定した任意の場所にビルド ファイルを格納することができます。
- **ビルドの継続的インテグレーション**
ビルド定義を新規に作成する場合、または既存のビルド定義を変更する場合に、ビルドのトリガを指定できます。オンデマンド ビルド、ローリング ビルド、および継続的インテグレーションを使用して、チェックインするたびにビルドを開始できます。ローリング ビルドの定義では、ビルドの間隔を定義することもできます。
- **スケジュールされたビルド**
変更がなくても、あらかじめ設定したスケジュールに沿ってビルドを実行できるようになりました。

- **ビルド エージェント**
ビルド エージェントには、ビルド コンピュータ名とは異なる名前を付けることができます。
各ビルド エージェントは、双方向通信可能なポートと、ビルドの実行に使用される既定のポートの 2 つを経由してビルド コンピュータに接続できます。
- **HTTPS と Secure Sockets Layer (SSL) を使用するビルド**
HTTPS と SSL を要求するチーム ビルドをセットアップできます。
- **Team Foundation ビルドをカスタマイズするための新しいプロパティ**
Visual Studio Team System 2008 には、ビルドをカスタマイズするための新しいプロパティが追加されました。
SkipInvalidConfigurations、CustomizableOutDir、および CustomizablePublishDir の各プロパティを使用して、C++ ビルドの動作をカスタマイズできます。
- **Team Foundation ビルドをカスタマイズするための新しいタスクとビルド ターゲット**
Team Foundation ビルドの多くの新しいビルド ターゲットは、ビルド プロセスをカスタマイズするために上書きできます。

2. Team Foundation Server のソース管理機能

- **破棄**
Team Foundation Server のソース コード管理機能では、ソース管理されたファイルを破棄、つまり完全に削除できるようになりました。
- **チェックアウト時の最新バージョンの取得**
Team Foundation Server のソース コード管理機能では、ファイルのチェックアウト時に最新バージョンのファイルを取得できるようになりました。
- **ファイルの注釈**
ソース コードのファイルに注釈を付けることができるようになりました。ソース コードの変更内容、変更者、変更日時などの情報を 1 行ごとに確認できます。
- **フォルダの比較**
ソース管理を使用して、2 つのサーバー フォルダの比較、2 つのローカル フォルダの比較、またはサーバー フォルダとローカル フォルダの比較ができるようになりました。片方のフォルダにのみ存在する項目や、項目の追加、削除、競合する変更などの相違点がかかります。

3. Team Foundation Server の作業項目トラッキング機能

- 作業項目を操作する際のパフォーマンスが大幅に向上しました。Visual Studio 2005 Team System と比較すると、スループットは約2 倍になっています。これによって作業項目を使用する操作において必要な時間が短縮されました。また、TFS データ層におけるサーバーの CPU 使用率を低減しました。すでに Visual Studio 2005 Team Foundation Server を導入いただいている場合、Visual Studio Team System 2008 に移行いただくことで同じサーバー上で、Visual Studio 2005よりも多くのユーザーの作業項目を扱うことができるようになりました。

4. Team Foundation Server の管理

- Visual Studio Team System 2008 では、多数のユーザーを TFS に追加した場合の信頼性が高まり、応答の遅延が低減しています。サポートするユーザーの総数は変わりませんが、Active Directory と Visual Studio Team System 2008 Team Foundation Server との間のユーザーの同期に必要な時間が大幅に短縮されました。

▲ Top of Page



1. トップダウン方式を使用したアプリケーション システムのデザイン

- システム デザイナを使用してアプリケーション システムをデザインする場合に、トップダウン方式を利用できるようになりました。システム デザイン ソリューションを新規で作成したり、既存のソリューションを変更したりできます。システム、アプリケーション、およびエンドポイントをメンバとしてシステム定義に直接追加できます。エンドポイントをシステム定義の境界に追加して、後でメンバに動作をデリゲートできます。メンバ名の変更と、基になっている定義名の変更を同時に行うことができます。システム定義から孤立したアプリケーション システムのメンバを修復できます。

2. .NET Web サービス エンドポイントの WSDL ファイルへの準拠

- 既存の .NET Web サービス プロバイダ エンドポイントでの操作を、Web サービス記述言語 (WSDL) ファイルに準拠させることができます。

3. ASP.NET Web アプリケーションプロジェクトの生成

- ASP.NET アプリケーションを実装するために、ASP.NET Web アプリケーション テンプレートを選択できます。このテンプレートを選択すると、ASP.NET アプリケーションに対

4. カスタム プロトタイプの保存、インポート、およびエクスポート

- 自分のみが使用するカスタム プロトタイプや、特定のコンピュータのすべてのユーザーが使用できるカスタム プロトタイプを保存したりインストールしたりできるようになりました。カスタム プロトタイプをインストールする場合、レジストリを編集しなくても、それらのプロトタイプをインポートできるようになりました。
※注：システム定義モデル (SDM) のソフトウェア開発キット (SDK) から作成されたアプリケーション プロトタイプに対応する .sdm ドキュメント ファイルをインストールする場合は、レジストリを編集する必要があります。
- カスタム プロトタイプをエクスポートして、他の人と共有できるようになりました。

5. 複数のバージョンからの .NET Framework の選択

- ASP.NET、Windows、および Office アプリケーションを作成する場合、.NET Framework のバージョンを 2.0、3.0、3.5 のいずれかから選択できます。

6. 複数のバージョンからの Office の選択

- Office アプリケーションのプロジェクト テンプレートを、Office 2003 と Office 2007 のいずれかから選択できるようになりました。

▲ Top of Page



1. Transact-SQL の静的コード分析

- Transact-SQL コードの静的コード分析を実行して、コーディングの問題点を自動的に検出したり、コードを最適化したりできます。
- プロジェクトで使用するルールを構成し、実際に静的分析において使用するルールと、そのルールに違反した場合に警告として扱うかエラーとして扱うかを決定することができます。また、警告とエラーを非表示にすることもできます。

2. Transact-SQL の静的コード分析

- プロジェクト内のスキーマ オブジェクトの依存関係チェーンをツリー ビュー形式で表示できます。オブジェクト間の参照関係 (参照されている、または参照している) が表示されません。

3. リファクタリング

- **リファクタリングの新しい種類**
リファクタリング エンジンでは、新しい種類のデータベース操作がサポートされます。既存のスキーマ間でオブジェクトを移動する "スキーマの移動" 機能、および "ワイルドカード展開" 機能を使用できるようになりました。"SELECT *" ステートメントを定義された列リストに変換できる "ワイルドカード展開" 機能は、テーブルまたはビューが別名経由で参照されている場合、その別名を使用して列リストを展開します。別名を使用して適切に列リストを展開できない場合は、完全修飾されたテーブル名またはビュー名を展開します。また、各オブジェクトの参照を完全に修飾して、クエリ ステートメントからあいまいさを排除する "完全修飾名" 機能も使用できるようになりました。
- **リファクタリングの新しいターゲット**
Visual Studio で、厳密に型指定された "DataSet" の定義を含む Visual Basic または C# プロジェクトに対して作業を行っている場合、"DataSet" で使用されているオブジェクトのリファクタリングを実行すると、Database Edition のリファクタリング エンジンは、自動的に "DataSet" の定義に含まれているオブジェクトのリファクタリングを実行します。
- **REFACTORING COMMAND GENERATOR**
新しい Refactoring Command Generator を使用すると、リファクタリング操作を、ログ ファイルではなくグラフィカルなインターフェイスで確認および追跡できます。Refactoring Command Generator ツールは、ユーザーの操作が反映された配置ステートメントを作成します。

4. データの生成

- 新しいシーケンシャル データ バウンド ジェネレータは、既存のデータ バウンド ジェネレータと同様の役割を果たします。ただし、シーケンシャル データ バウンド ジェネレータは、生成された辞書の順序に基づいて列を操作します。新しいエディタであるデータ バウンド ジェネレータ、シーケンシャル データ バウンド ジェネレータ、および RegEx String Generator を使用すると、データ ジェネレータの構成が容易になります。

5. MSBuild タスク

- スキーマやデータを比較できる、次の 2 つの MSBuild タスクをコマンド ラインから実行できるようになりました。
 - SqlSchemaCompareTask は、2 つのデータベース間でスキーマを比較します。
 - SqlDataCompareTask は、2 つのデータベースのテーブルとビューの内容を比較します。

6. その他のツール

• SCHEMA MANAGER API

Schema Manager API を使用して、スキーマ オブジェクトの追加、更新、削除、およびスキーマ オブジェクトとスキーマ ファイルの列挙を実行できます。

この API は Computer Associates (CA) の ERWin 7.2 を Visual Studio Team System 2008 Database Edition と統合するときに使用します。ERWin 上で変更を行うと、データベース プロジェクト内のスキーマ定義に変更が保存されます。ERWin の統合の詳細については、<http://ca.com/us/content/campaign.aspx?cid=144449> (英語) を参照してください。

• テーブルとインデックスのオプションの指定

テーブルとインデックスの定義で、Microsoft SQL Server 2005 で新たに追加された vardecimal ストレージ形式などのオプションを指定できるようになりました。

• SQL Server ファイル グループ内のファイルのサポート

ファイル グループ内のファイルをデータベース プロジェクトのプロパティとして定義できるようになったため、配置前ストレージ スクリプト内でファイルとファイル グループを作成する必要がなくなりました。

• データベース間の参照

サポートが強化され、データベース プロジェクト参照を使用するか、データベース メタファイル (.dbmeta) を参照することによって、別のデータベース内のオブジェクトを参照できるようになりました。このサポートによって、データベース プロジェクト内のデータベース間の参照に関する警告が減少します。

• 変数

データベースのプロパティの [new Variables] ページで "setvar" 変数を定義して、配置スクリプト内で使用できるようになりました。

▲ Top of Page



1. コード分析

コード分析ツールは、コード障害の広範囲な確認を実行します。コード障害はエラー ウィンドウに警告として表示されます。

コード分析で強化された機能は以下のとおりです。

• ルールの拡張と強化

20 種類を超える新規のルールが追加され、合計 200 種類以上のコード分析を使用することができます。

• スペル チェックでのカスタム辞書のサポート

クラス、メソッド、およびプロパティ名と同様に、リソース文字列にもスペル チェックを使用できます。カスタム辞書を使用すれば、標準的でない語句のスペル チェックも実行できます。

• よりきめ細かく制御できるようになったエラーの非表示機能

コード分析で検出された問題を、プロジェクトまたはソース内のエラー ウィンドウに表示しないようにすることができます。

• 生成されたコードのエラーを自動的に非表示にするオプション

生成されたコードのエラー メッセージを自動的に非表示にすることができます。これは、特にデザイナーから生成されたコードに役立ちます。

• コード分析ポリシーの強化

サーバーからプロジェクトに設定をコピーするときに、ローカルの設定を上書きするか、ポリシーのルールをローカルのプロジェクトのルールと結合するかを選択できます。また、ポリシーの違反について、さらに詳しい情報を得ることができます。これによって、違反の原因を特定できます。

2. コード メトリックス

- コード メトリックスとは、開発者が、開発中のコードの状況をより適切に理解できるようにする、ソフトウェアの測定基準です。開発者はコード メトリックスを使用することで、どの型やメソッドを書き直したり、さらにテストしたりする必要があるのかを理解できます。さらに開発チームは、ソフトウェアの開発中に潜在的なリスクを識別し、プロジェクトの現在の状態を理解し、進捗状況を管理することができます。

3. プロファイル ツール

開発者は、Visual Studio Team System 2008 Development Edition のプロファイル ツールを使用し

てコード内のパフォーマンスに関連する問題を測定、評価、および特定できます。
プロファイル ツールには、以下の機能が追加されました。

- **64 ビット サポート**

プロファイラでは、64ビット オペレーティング システムおよびハードウェア上で動作する 64ビット アプリケーションと 32 ビット アプリケーションの両方がサポートされるようになりました。

- **割り当ての完全な呼び出し履歴**プロファイラでは割り当ての完全な呼び出し履歴が提供されます。これは、ユーザー コード以外のコードの中で、ユーザーの操作によって間接的に発生した割り当てを把握する際に役立ちます。完全な呼び出し履歴を使用することによって、コードのどの部分によって間接的な割り当てが発生したかを正確に把握することができます。

パフォーマンス セッションのプロパティ ページで設定を構成することによって、割り当てデータを収集できます。結果を表示するには、パフォーマンス レポートで割り当てビューを使用します。

- **行レベルのデータのサンプリング**

プロファイル ツールのパフォーマンス レポートでは、命令ポイントと行ビューが提供されるようになりました。また、モジュール ビューに行情報が含まれるようになりました。

- **レポート上の不要項目の非表示**

不要項目を非表示にするようにパフォーマンス レポートを構成できます。これによって、呼び出しツリー ビューと割り当てビューに表示されるデータの量を制限できます。不要項目を非表示にすると、パフォーマンスに関する問題点がより明確になるため、パフォーマンス レポートを分析する際に役立ちます。

- **ランタイム コントロール**

プロファイル ツールにはランタイム コントロールが含まれています。ランタイム コントロールはプロファイラと連動して自動的に起動します。ランタイム コントロールを使用して、パフォーマンス データのログ記録を一時停止したり、再開したりできます。また、アプリケーションの起動と同時にログ記録を一時停止することもできます。これを利用すると、アプリケーションの起動時にデータの収集をスキップできます。ランタイム コントロールを使用すると、アプリケーションの実行中に興味深い動作が発生したときに、パフォーマンス データ内に注釈を手動で挿入できます。後から注釈が付いたデータをフィルタ選択できます。

- **フィルタを使用した分析**

パフォーマンス レポートにタイムスタンプ、プロセス、スレッド、およびマークでフィルタをかけることができます。フィルタをかけた分析結果を表示するには、[show query] ボタンを使用します。コマンド ラインの場合は、/summaryfile オプションを使用します。

- **レポートの比較**

プロファイラではレポートの比較がサポートされるようになりました。パフォーマンス エクスプローラを使用するか、コマンド ラインで /diff オプションを指定することによって、レポートの比較を実行できます。

- **チップ カウンタのサポートの強化**

プロファイル ツールでは、以前よりもわかりやすいチップ カウンタ名 ("L2 Misses"、"ITLB Misses"、"Mispredicted Branches" など) が提供されます。XML ファイルを変更して、特定のアーキテクチャ向けのカウンタを追加で構成することができます。

- **Windows カウンタのサポート**

プロファイラは、Windows カウンタ ("% Processor Time"、"% Disk Time"、"Disk Bytes/sec"、"Page Faults/sec" など) を収集するようになりました。パフォーマンス セッションのプロパティ ページにある Windows カウンタ ノードを使用するか、コマンド ラインで /wincounter オプションを指定します。マーク ビューにカウンタが表示されます。カウンタをフィルタのエンドポイントとして使用できます。

- **圧縮されたレポート ファイル**

プロファイル ツールでは、すばやく開くことができる、小さいサイズに圧縮されたレポート ファイルを生成できます。このファイルは、完全なレポートから作成された分析済みのファイルです。パフォーマンス エクスプローラでレポートを右クリックして [分析されたものを保存] を選択するか、コマンド ラインで /summaryfile オプションを指定します。

- **ホットパス**

プロファイラでは、パフォーマンス レポートのコール ツリーと割り当てビューで、パフォーマンスに最も大きい影響を与えるコード パスを自動的に展開する機能が提供されます。

- **レポート ビューのデータを HTML 形式でコピーする機能**

プロファイラでは、書式情報を含むレポート データをクリップボードで保持する機能がサポートされます。パフォーマンス レポートから、書式情報を含むデータ (見出しと値が含まれた表) をコピーして貼り付けることができます。

- **Windows Communications Foundation のサポート**

プロファイル ツールでは、Windows Communications Foundation (WCF) がサポートされるようになりました。

4. ロードテストとWeb テストの Visual Studio Team System 2008 Team Suite への統合

- テスト ビューとテスト結果から、ロードテストとWeb テスト用のパフォーマンス セッションを作成できます。



1. コードのテスト方法

より多くの製品コードに対応した単体テストを、より迅速かつ容易に作成および実行できるようになりました。

• Visual Studio 2008 Professional Edition の単体テストの使用

Visual Studio 2008 Professional Edition を使用する開発者は、単体テストと順序指定テストという 2 種類のテストを作成および実行できるようになりました。単体テストは、特定の製品コードのメソッドが正しく動作するかどうかの検証、リグレッション テスト、またはパディ テストやスモーク テストを実行するために使用します。順序指定テストは、他のテストを指定された順序で実行します。

• 単体テストの容易な実行

単体テストの開発者は、新しくなったメニューとキーの組み合わせを使用して、より迅速にテストの実行を開始したり、実行するテストを選択したりできます。また、製品のソースコードにアクセスすることなく、バイナリ ファイルからテストを生成できます。戻り値とメソッド パラメータとして使用される汎用データ型のテストを生成できます。

• テスト クラス間の継承の使用

テスト クラスは、他のテスト クラスからメンバを継承できるようになりました。これによって開発者は、基本テスト クラス内で初期化またはテストを作成し、他のすべての派生テスト クラスに継承させることができます。この機能によって、テスト コードの重複を回避できます。また、開発者に単体テストをより正しくカスタマイズするための選択肢が提供されます。

• デバイス上での単体テストの実行

Visual Studio では、C と Visual Basic で記述されたスマート デバイス アプリケーションをテストするための一連のツールが提供されます。これらのツールは、Test Edition で提供される機能のサブセットです。

• ホスト アダプタの作成

通常は、Team System のテスト ツールが提供する既定の環境でテストを実行します。異なる環境でテストを実行するには、ホスト アダプタを使用します。新しいホスト アダプタを作成するには、Visual Studio SDK を使用します。Visual Studio SDK は、[アフィリエイトサイト](#) (英語) からダウンロードできます。

• 単体テストのデータ バインドの強化

ウィザードを使用して、単体テストを CSV ファイルと XML ファイルが含まれたデータソースと容易にバインドできるようになりました。

2. Web テスト

Visual Studio Team System 2008 では、Web テストを作成するためのコントロールが追加されました。

• 別の Web テストの呼び出し

Web テストには、別の Web テストの呼び出しを挿入することができます。

• Web テストのデータ バインドの強化

Visual Studio Team System 2008 Test Edition には、CSV ファイルと XML ファイルのサポートが組み込まれています。新しいウィザードを使用すると、容易にデータ バインド プロセスを完了できます。プロセスを完了する前に、データをプレビューすることもできます。

• Web テスト機能の強化

Visual Studio Team System 2008 Test Edition では、テスト レベルの検証ルールがサポートされます。検証ルールは、テスト レベルで作成できます。テストの個々の要求に対して、作成したルールを適用できます。テスト中にエラーが発生した場合、Web テストを中止することができます。また、予期した HTTP ステータス コードが返されるかどうかを検証することもできます。Visual Studio Team System 2008 Test Edition では、Web テストから要求を取り出して、別の Web テストを新しく作成できます。また、他の Web テストの呼び出しを挿入することもできます。つまり、Web テスト コンポーネントを作成し、Web テストと Web 要求を再利用できます。

3. ロード テスト

より実的なロード モデリングのオプションを使用して、ロード テストを実行できます。また、返されたデータを、豊富かつ柔軟な方法で整理することもできます。

• ロード モデリングの制御

ロード テストで提供されるロード モデリングのオプションが追加されました。これらのオプションを使用すると、想定しているアプリケーションや Web サイトの実際の使用状況がより適切にモデル化されたロード テストを作成できます。テストの実行回数、各テストの所要時間、またはユーザーがテストを実行するペースに基づいて、使用状況をモデル化

できるようになりました。

● **ロード テスト アナライザの表示の強化**

ロード テスト アナライザに追加されたサマリ ビューでは、キー インジケータと結果が 1 ページに表示され、そのページを印刷およびエクスポートできます。また、重要な情報を表示する 4 つの組み込みのグラフが追加されました。最大 4 つのグラフを同時に表示できます。この機能強化によって、最大 4 つの表を同時に表示できるようになりました。

● **強化されたロード テストの結果レポジトリの管理**

Visual Studio Team System 2008 Test Edition では、新しいレポジトリ管理ダイアログ ボックスが提供されます。このダイアログ ボックスから、ロード テストの結果レポジトリに直接アクセスできます。これにより、ロード テスト結果の参照、インポート、エクスポート、および削除が容易になりました。

● **XML ファイルでのスキーマの公開**

Visual Studio Team System 2008 Test Edition を使用した場合、データは XML ファイル形式で作成および格納されます。作成されるファイルを以下に示します。

テスト メタデータ ファイル。このファイルの種類の拡張子は .vsmdi です。テスト メタデータ ファイルには、ソリューションのテストに関する情報が格納されます。

テスト結果ファイル。このファイルの種類の拡張子は .trx です。テストを実行すると、Visual Studio は自動的にテスト結果を .trx ファイルに保存します。

テキスト形式の手動テスト。このファイルの種類の拡張子は .mtx です。手動テストをテキスト形式で作成すると、この種類のファイルとしてテスト プロジェクトに保存されます。Visual Studio Team System 2008 Test Edition では使用する XML ファイルはすべて、TestTypes.xsd という名前の新しい XSD によって定義されます。これらのファイルを (手作業またはプログラムで) 編集する場合は、変更内容をこの XSD で定義されたスキーマに準拠させる必要があります。同様に、上記の拡張子を付けて作成したファイルはすべて、この XSD で定義されたスキーマに準拠している必要があります。スキーマに準拠していない場合、Visual Studio Team System 2008 Test Edition はそれらのファイルを使用できません。

Visual Studio 2005 で作成されたテスト プロジェクトには、XML ファイルが含まれています。Visual Studio 2005 のテスト プロジェクトを開くと、Visual Studio 2008 のプロジェクト アップグレード ウィザードが起動し、ファイルを新しい形式に変換するかどうかを確認されます。Visual Studio Team System 2008 Test Edition でこれらのファイルを使用するには、ファイルを変換する必要があります。いずれかのファイルを変換、つまりアップグレードしなかった場合、そのテスト プロジェクトを Visual Studio で開くことはできません。同様に、古い形式の既存のファイルをテスト プロジェクトに追加すると、ファイル形式をアップグレードするかどうかを確認されます。[はい] を選択すると、ファイルが変換され、テスト プロジェクトの一部として使用できるようになります。[いいえ] を選択すると、ファイルをプロジェクトに追加する要求は取り消されます。

メリット :

- Web テストの検証ルールの強化。検証ルールをより柔軟に適用し、検証結果を Web テストのプログラム フローの管理に使用できます。
- ロード モデリングの制御の強化。実行するロード テストのロード モデリングをより柔軟に制御できるようになります。
- ロード テスト アナライザの表示の強化。新しい組み込みのグラフと表示機能によって、ロード テストの結果を理解しやすくなります。
- ロード テストの結果レポジトリの管理の強化。ロード テストの結果レポジトリに容易にアクセスできるようになります。
- テスト結果の XML ファイルのスキーマ化。自動的に XML 形式で .trx (テスト結果) ファイルに格納されたテスト結果を、プログラムで処理できるようになります。

おわりに

このドキュメントでは、Visual Studio Team System 2008 で新しく追加された機能と強化された機能の概要を説明しました。Visual Studio Team System 2008 は、Visual Studio 2005 から提供を開始した Team System 製品ラインにとって初のメジャー アップグレードであり、さまざまな機能追加はもちろん、既存機能に関しても大幅な強化を行っています。このドキュメントが皆様の Visual Studio Team System 2008 の評価、導入の参考になれば幸いです。

参考情報

- [Visual Studio Team System 最新情報](#)
- [Visual Studio 最新情報](#)
- [MSDN ライブラリ](#)

▲ Top of Page