



InstallShield 2011

リリース ノート

2010 年 8 月

はじめに

InstallShield は、ハイクオリティな Windows Installer または InstallScript ベースのインストーラ、および Microsoft App-V パッケージをオーサリングできる業界標準ツールです。InstallShield 2011 には、最新のテクノロジーを活用し、製品の品質と信頼性を高め、生産性を向上させ、他のチーム メンバーとのコラボレーションを容易にするための、多くの新しい機能と強化内容が搭載されています。

InstallShield 2011 では、InstallScript インストーラおよび InstallScript カスタム アクションにおいて、モダン言語を使った多国語インストーラの開発には欠かせない Unicode がサポートされています。Team Foundation Server をご利用の場合は、製品の開発、管理、ビルド、テスト、および配布、ならびにそのインストーラにおいて、新しい InstallShield との強い統合性を最大限にご活用いただけます。InstallShield 前提条件のカスタム ソースの場所を指定できるようになりました。この柔軟性によって、ソース コード管理システムに格納されている InstallShield 前提条件をプロジェクトに簡単に追加することができ、また複数のマシン間で共通の前提条件を共有することができます。また InstallShield を使って、Setup.exe プロパティとアイコンをカスタマイズして貴社のブランドを強化したり、ビルドの様々な段階で自動的に実行するコマンドを指定することでビルド プロセスを簡素化したり、改良されたスクリプト エディタを使って作業効率を向上したりすることもできます。その他の新しい機能を使って、64 ビット インストーラまたは App-V パッケージを容易に作成することができます。

InstallShield には、Standalone Build コマンドライン ツールが含まれていて、InstallShield のインストーラをコンパイルする部分 (および含める再配布可能ファイル) のみをビルド マシンにインストールすることができます。Standalone Build は、InstallShield Professional Edition および Premier Edition に含まれています。また、Standalone Build の追加ライセンスを別途購入することもできます。

リリース ノートのアップデートを含む InstallShield 2011 についての最新情報は、ナレッジベース記事「[Q208910](#)」をご覧ください。

新機能

InstallScript における Unicode サポート

InstallShield を使って、ランタイム文字列、ファイル、パス、レジストリ エントリ、その他のインストーラ データで Unicode を使用できる InstallScript インストーラおよび InstallScript カスタム アクションを作成できます。さらに、InstallScript コンパイラおよび InstallScript エンジンでは、今回より、スクリプト コード外部に実装された関数に Unicode 文字列へのポインタを渡すことができます。また、スクリプト コード外部に渡される構造体に Unicode 文字列を格納することもできます。そのほか、InstallScript にはモダン言語の多言語インストーラを完全にサポートするための機能が追加されています。

その結果、すべての言語は、そのサポートがインストールされているシステム上で正しく表示されます。エンド ユーザーは、システム上で Unicode 対応ではないプログラム用に使用される言語とインストーラに使用される言語とを一致させる必要がなくなりました。ただし、ターゲット システムには、フォントをインストールする必要があります。Windows の一部のバージョンでは、そのフォントがデフォルトでインストールされない言語もあります。たとえば、Windows XP 英語版システムでは、日本語

のフォントがデフォルトではインストールされません。その場合、インストーラーが日本語の文字を使用できるように、フォントをインストールする必要があります。

InstallScript および InstallScript MSI インストールにおける Unicode セットアップ起動ツール

今回より、InstallShield ではすべての Setup.exe と Update.exe ファイルが Unicode でビルドされます。以前、InstallScript および InstallScript MSI プロジェクトでは、すべての Setup.exe と Update.exe ファイルが ANSI でビルドされました。

Unicode セットアップ起動ツールは、ターゲット システムで適切なコード ページが実行されているいなくにかかわらず、セットアップ起動ツールのユーザー インターフェイスで文字を正しく表示することができます。ANSI セットアップ起動ツールは、ターゲット システムで適切なコード ページが実行されている場合のみ、セットアップ起動ツールのダイアログで文字を正しく表示します。しかし、ターゲット システムで適切なコード ページが実行されていない場合、文字化けすることがありました。

エンド ユーザーは、ターゲット システムの言語にかかわらず、Setup.exe および Update.exe ファイルを Unicode パス内部から起動することができます。たとえば、今回より、エンド ユーザーは、英語版システム上で C:\Users\日本語文字\Desktop\Setup.exe からインストールを起動することができます。以前、このインストーラーは失敗しました。

ファイル、フォルダー、レジストリ エントリ、およびサポート ファイルにおける Unicode サポート

インストーラー ランタイムの主要な部分に Unicode サポートが追加されました。これによって、ファイル名、フォルダー名、レジストリ エントリ、およびサポート ファイル名に任意の言語の文字を同時に使用できるようになりました。たとえば、ファイル名またはターゲット パスに日本語文字が含まれるファイルを、英語版システムにインストールすることが可能です。組み合わせられた言語は、ターゲット システムの現在の言語設定に関わらず正しく表示されます。

ポインター サポート

InstallScript エンジンとコンパイラーは、今回より、WPOINTER と呼ばれる新しいポインターをサポートします。対応する名前、wpointer および LPWSTR も使用可能です。たとえばパラメーターで Unicode 文字列へのポインターを受け付ける DLL 関数がある場合、この新しい種類のポインターを使うことができます。実行時にスクリプトで DLL 関数が呼び出されると、InstallScript エンジンポインターを ANSI バージョンではなく、文字列の Unicode コピーに渡します。以前、ポインターは文字列の ANSI コピーにのみ渡すことができました。

InstallShield の以前のバージョンで使用される WSTRING データ タイプを使って Unicode 文字列を渡す機能は、引き続きサポートされています。

構造体のサポート

InstallScript の構造体には、文字列、ポインター、その他の構造体をはじめとする任意の基本データ タイプを含めることができます。構造体に Unicode 文字列を含む必要があり、その構造体が外部 DLL に渡される場合、InstallScript エンジンはその構造内の文字列メンバー タイプを区別して、構造体サイズとメンバー オフセットを正しく計算します。Unicode として保存して渡す必要がある文字列メンバーは、WSTRING タイプを使って宣言できます。

以前、構造体に Unicode 文字列を含む必要があり、その構造が外部 DLL に渡される場合、InstallScript エンジンはその構造内に含まれる文字列がすべて ANSI であると想定しました。その結果、構造体のサイズとメンバーのオフセットに誤りが生じることがあり、DLL がその構造体に関連するデータの読み取りまたは書き込みを正しく行いませんでした。構造体の文字列メンバーに WSTRING を使用しても、効果がありませんでした。

既存の文字列を STRING タイプのままに残すことができます。InstallScript エンジン、スクリプト コード外部で文字列を渡すとき、引き続きこれらを ANSI 文字列として処理します。

構造体のポインター メンバーも、今回より、WPOINTER として宣言できます。これによって、構造体で Unicode 文字列へのポインターを格納できます。

文字列テーブルのサポート

InstallShield は、今回より、InstallScript プロジェクトの文字列テーブルをビルドする際に Unicode エンコードを使用します。このサポートにより、今回より、InstallScript プロジェクトの文字列テーブルに複数言語を含めることが可能となり、それらの言語がターゲット システムのコードページ設定に影響を受けることがなくなりました。さらに今回より、文字列テーブルに、ヒンディー語のようなコード ページを持たない言語の文字列を含めることができます。

インストールのユーザー インターフェイスに表示されるすべての文字列を、文字列テーブルに格納することが推奨されます。InstallScript コード内に直接文字列を書き込むこともできますが、その場合 Unicode として格納されません。したがって、正しいコード ページが設定されたシステム上でインストールが実行された場合にのみ、言語が正しく表示されます。文字列テーブルに文字列を格納して、InstallScript コードから文字列 ID を参照することで、この問題を回避することができます。

この機能によって、1-14SJA9、1-1700WL、IOA-000051919、IOA-000052767、IOA-000054177、IOB-000050617、IOB-000058234、IOC-000050725、IOC-000063585 が解決されます。

Unicode を使った InstallScript ダイアログ

InstallScript ダイアログは、Unicode サポートを含みます。これによって、たとえば InstallScript ダイアログで日本語とドイツ語、またはロシア語とポーランド語を同時に使用できます。組み合わせられた言語は、ターゲット システムの現在の言語設定に関わらず正しく表示されます。

この機能は、InstallScript および InstallScript MSI プロジェクト タイプで使用できます。

この機能により IOC-000057769 が解決されました。

InstallScript デバッガーにおける Unicode サポート

InstallScript デバッガーは、今回より Unicode をサポートします。たとえば、szMsg = @ID_MSG という InstallScript コードの行をデバッグするとき、今回より、使用中のオペレーティング システム、または文字列に使用している言語にかかわらず、szMsg の値が表示されます。ID_MSG の値に中国語文字、または複数の言語からの文字が含まれている場合、InstallScript デバッガーのあらゆる領域 (スクリプト ウィンドウ内の変数に対して表示される変数ウィンドウ、ウォッチ ウィンドウ、およびツールヒント) で、疑問符の代わりに適切な文字が表示されます。

この機能によって、IOB-000059145 が解決されます。

InstallShield キャビネット ビューアーおよびログ ファイル ビューアーにおける Unicode サポート

InstallShield キャビネット ビューアーおよび InstallShield ログ ファイル ビューアーが、新しく InstallShield キャビネット & ログ ファイル ビューアー という 1 つのツールにまとまりました。この新しいツールを使って、InstallScript キャビネット ファイル (.cab)、InstallScript ヘッダー ファイル (.hdr) および InstallScript ログ ファイル (.ilg) を開いて参照することができます。このツールには Unicode サポートが含まれているため、使用中のオペレーティング システムの言語や文字列に使用されている言語にかかわらず、各ファイル名、レジストリ キー、ショートカット、および .cab、.hdr、または .ilg ファイルに含まれるその他のデータを正しく表示することができます。以前、一部の状況において一部の文字が疑問符で表示される場合があります。

キャビネットおよびヘッダー ファイル サポートは、InstallScript プロジェクトおよび InstallScript オブジェクト プロジェクト タイプに適用します。

ログ ファイル サポートは、InstallScript プロジェクトおよび InstallScript オブジェクト プロジェクト タイプに適用します。

Team Foundation Server (TFS) との統合

InstallShield では、Team Foundation Server (TFS) 2010 との統合サポートが強化されています。

Visual Studio 2010 内から InstallShield を使用する場合、Source Control Explorer にアクセスして、InstallShield プロジェクトを Team Foundation バージョン コントロールと統合し、InstallShield プロジェクトと Visual Studio ソリューション への変更を管理することができます。

Team Foundation Build を使って、InstallShield プロジェクトと Visual Studio ソリューションを定期的、またはオンデマンドでコンパイル、テスト、およびデプロイすることもできます。 インストールはソリューションがビルドされる度に、自動的に最新のソース ファイルで更新されます。

さらに、InstallShield と Visual Studio が搭載された同じマシン上に Team Explorer をインストールした場合、Visual Studio で開かれている InstallShield プロジェクト内から Team Explorer を使用できます。 これで、次のようなタスクを行うことができます：

- InstallShield プロジェクトでの作業中にソース管理エクスプローラーを使用する。
- InstallShield プロジェクトと Visual Studio ソリューションのビルドを構成する。 新しいビルドをキューに配置。
- InstallShield プロジェクトと Visual Studio ソリューションのバグやタスクなどの作業項目を追跡する。

Microsoft SQL Server 2008 R2 サポート

InstallShield は、今回より、SSQL Server 2008 R2 上で SQL スクリプトを実行するためのサポートを含みます。 また、InstallShield の [SQL スクリプト] ビューで、製品がサポートするターゲット データベース サーバーを指定するときに選択可能な定義済みデータベース サーバーのリストには、SQL Server 2008 R2 が含まれています。

この機能は、基本の MSI、InstallScript、および InstallScript MSI プロジェクト タイプで使用できます。

この機能によって、IOA-000055382 と IOA-000055685 の問題が解決されます。

SQL Server 2008 R2 Express、SQL Server Native Client、Visual C++ 2010 用の新しい InstallShield 前提条件およびその他の再配布可能ファイル

InstallShield には、基本の MSI、InstallScript、および InstallScript MSI プロジェクトに追加することができる多くの InstallShield 前提条件が含まれています。

- Microsoft SQL Server 2008 R2 Express
- Microsoft SQL Server 2008 Native Client 10.00.2531
- Microsoft SQL Server Native Client 9.00.4035
- Microsoft SQL Server System CLR 型 10.00.2531
- Microsoft SQL Server 2008 Management Objects 10.00.2531
- Microsoft Visual C++ 2010 再配布可能パッケージ
- Microsoft Visual C++ 2008 SP1 再配布可能パッケージ
- Microsoft Visual C++ 2005 SP1 再配布可能パッケージ (x64)
- Windows Installer 3.1 - 日本語
- MSXML 6.0 SP1 - 日本語
- Microsoft .NET Framework 4.0 Client Language Pack - 日本語
- Microsoft .NET Framework 4.0 Full Language Pack - 日本語

この機能によって、IOA-000053708、IOA-000054694、IOC-000058108 が解決されます。

InstallScript コード、SQL スクリプト、VBScript カスタム アクション、および JScript カスタム アクションの作成および変更を行うスクリプト エディターの強化

InstallShield のいくつかのビューで、プロジェクトのコードを書き込むためのスクリプト エディターが強化されました。 スクリプト エディターは、以下の点で強化されています：

- **オートコンプリート機能の拡張** — スクリプト エディターに文字を入力するとき、InstallShield はその文字で始まる関数、キーワード、および定数をアルファベット順にリストするポップアップを表示します。 用語全体を手入力する代わりに、そのポップアップ リストから選択すると、InstallShield によってその用語がスクリプトに追加されます。

[InstallScript] ビューでスクリプト エディターを使用する場合、文字列定数演算子 (@) を入力すると、使用可能な文字列識別子のポップアップ リストが表示されます。 リストから適切な項文字列 ID を選択すると、InstallShield がそれをスクリプトに追加します。

ローカル変数のオートコンプリート機能も有効な場合、[InstallScript] ビューのスクリプト エディターで表示されるポップアップ リストには、ローカル変数も含まれています。

オートコンプリートを使うと、コードを手入力する手間が省けるため、作業効率が上がります。 また、コードのスペル ミスを回避することができます。

InstallShield の以前のバージョンでは、[InstallScript] ビューのスクリプト エディターでのみオートコンプリートがサポートされていました。そのため、オートコンプリート機能は SQL スクリプト、VBScript コード、または JScript コードでは使用できませんでした。 さらに、ポップアップ リストも InstallScript 関数に限られており、キーワード、定数、ローカル変数、または文字列 ID はオートコンプリートで使用できませんでした。

- **詳細な InstallScript 関数呼び出しのヒント** — 関数呼び出しのヒントが有効な場合、[InstallScript] ビューでスクリプトに関数の呼び出しを入力しているときに、InstallScript 関数呼び出しのヒント (一種のツールヒント) が表示されます。 関数呼び出しのヒントは、ビルトイン関数のパラメーター情報を表示します。 その他、ビルトイン関数の説明や、入力しているパラメーターについての説明も表示します。 詳細な呼び出しのヒントを使うと、スクリプト エディターからヘルプ ライブラリへ、そしてまたスクリプト エディターへと切り替える必要がなく、関数に関するヘルプ情報を表示できます。

InstallShield の以前のバージョンの [InstallScript] ビューでも、呼び出しのヒントをサポートしますが、その情報量が限られています。 呼び出しのヒントには、関数呼び出しとその関数のすべてのパラメーターが表示されましたが、呼び出しのヒントには、関数の説明、または入力しているパラメーターの説明は含まれませんでした。

- **構文の折りたたみ機能** — InstallShield の様々なビューで、スクリプト エディターの構文の折りたたみ機能をサポートするかどうかを指定できます。 構文の折りたたみ機能が有効な場合、スクリプトで展開可能または折りたたみ可能なブロックで始まるコードの各行の横のマージンに、プラス (+) またはマイナス (-) 記号が追加されます。 プラス記号をクリックして非表示となっているコードを展開したり、マイナス記号をクリックして表示されているコードを非表示にしたりできます。

構文の折りたたみ機能を使って、長いスクリプトを縮小することで、現在行っている作業に関連のあるコードに焦点を当てることができます。 また、スクリプトの全体的な構造を確認するのに便利です。

64 ビットの Microsoft App-V パッケージを作成できる機能

InstallShield の Microsoft App-V アシスタントには、64 ビットの Windows Installer パッケージを 64 ビットの Microsoft App-V パッケージに変換するためのサポートが搭載されています。 64 ビットの Microsoft App-V パッケージは、64 ビットの Microsoft App-V 4.6 Client を搭載した 64 ビット Windows システム上に展開することができます。 リパッケージャーに追加された新しい 64 ビット リパッケージ サポートとこの機能を組み合わせ、任意の 64 ビット インストールを 64 ビット App-V パッケージに変換することができます。

64 ビット Windows Installer パッケージを App-V パッケージに変換する場合は、64 ビット Windows システム上で行うことをお勧めします。 32 ビット システム上で変換しようとすると、64 ビット バイナリ ファイルの COM 情報抽出エラーが発生する場合があります。 また、一部の状況において、Windows Installer パッケージはパッケージそのものに含まれていない実行可能ファイルをターゲットとするショートカットを含みます。 ショートカットが、64 ビットの場所にある実行可能ファイルをターゲットとする場合、これらのショートカットは、32 ビット システム上で正しく処理されません。

Microsoft App-V アシスタントを使用するためには、Virtualization Pack を含む InstallShield のご購入が必要です。

InstallShield 前提条件の検索パスを指定できる機能 (前提条件ファイルのソースの場所におけるパス変数および相対パスのサポート)

InstallShield では今回より、InstallShield 前提条件ファイル (.prq ファイル) の検索フォルダを指定できます。 この機能によって、複数の開発者の間で InstallShield 前提条件を共有し、それらをソース コード管理システムで保管することが容易になります。

以前は、InstallShield が .prq ファイルを検索するフォルダーは、*InstallShield Program Files* フォルダー \SetupPrerequisites のみでした。

InstallShield でフォルダーを指定するとき、いくつかの方法があります：

- InstallShield 内部から編集またはビルドを行う場合、[ツール] メニューで [オプション] をクリックすると表示される [オプション] ボックスにある新しい [前提条件] タブを使って、マシン共通および現在のユーザーのフォルダをコンマ区切りのリストで指定できます。このタブは、マージ モジュールの検索パスを指定できる、[オプション] ダイアログ ボックスの [マージ モジュール] タブと似ています。
- ISCmdBld.exe を使って、コマンドラインからビルドする場合は、新しい -prqpath パラメーターを使ってフォルダのコンマ区切りのリストを指定します。

.ini ファイルを使って ISCmdBld.exe パラメーターを指定する場合、.ini ファイルの [Mode] セクションで新しい PrerequisitePath パラメーターを使用して、フォルダのコンマ区切りのリストを指定できます。

- MSBuild または Team Foundation Server (TFS) を使ってビルドする場合、InstallShield タスクで新しい PrerequisiteSearchPatch パラメータを使います。このパラメーターは、デフォルトのターゲット ファイルが使用されたとき、ItemGroup InstallShieldPrerequisitePath として露出されます。複数のパスを指定するには、順序指定されたパスの配列を使用します。

InstallShield 前提条件エディタの [含めるファイル] タブを使って InstallShield 前提条件にファイルを追加すると、エディターは今回より、適切な場合に <WindowsFolder> および <ISProductFolder> などの定義済みパス変数を使用します。また、追加するファイルが InstallShield 前提条件の .prq ファイルと同じフォルダ (または .prq ファイルが含まれているフォルダのサブフォルダ) に格納されている場合、InstallShield 前提条件エディターは .prq ファイル内のファイルに相対パスを使用します。[含めるファイル] タブでファイルのパスを参照したとき、InstallShield 前提条件エディターは相対パスではなく、完全パスをリストします。

InstallShield 前提条件エディターで [名前を付けて保存] コマンドを使用して .prq ファイルの場所を変更すると、InstallShield 前提条件エディターは、適切な場合に前提条件のファイルのパスを更新します。

この機能によって、IOA-000055078、IOA-000056459、IOC-000054166、IOC-000065360、IOC-000067484、IOC-000078297 が解決されます。

Windows Installer ベースのダイアログにおける新しいハイパーリンク コントロール

コントロール ツールバーに、Windows Installer ベースのダイアログで [ダイアログ] ビューに使用できる、新しいハイパーリンク コントロールが追加されました。ハイパーリンク コントロールは HTML リンクを表示します。このリンクを実行時にクリックすると、ターゲット システム上のデフォルト ブラウザでページが開きます。

Windows Installer 5 は、この新しいハイパーリンク コントロールをサポートします。このコントロールを Windows Installer の以前のバージョンを持つシステムで表示されるダイアログに使用すると、ランタイム エラー 2885 が発生して、インストールが中止されます。このため、ダイアログ上でハイパーリンク コントロールを使用するとき、インストールが Windows Installer 4.5 以前をターゲットとする場合、プロジェクトにハイパーリンク コントロールを含むバージョンと、含まないバージョンの 2 種類のダイアログを含めることをお勧めします。ダイアログに条件を追加して、ターゲット マシンの Windows Installer バージョンによってダイアログを表示または非表示とします。

この機能は、基本の MSI およびマージ モジュール プロジェクト タイプで使用できます。

Setup.exe および Update.exe においてカスタム アイコンとカスタム バージョン リソースのプロパティを指定できる機能

InstallShield では、今回より、ビルド時に作成する Setup.exe ファイルでカスタム アイコンおよびカスタムバージョン リソースのプロパティを使用できます。アイコンとバージョン リソースのプロパティは、Setup.exe の [プロパティ] ダイアログ ボックスに表示されます。この [プロパティ] ダイアログ ボックスは、エンド ユーザーが Setup.exe ファイルを右クリックして、[プロパティ] をクリックしたときに表示されます。エンド ユーザーが Windows Explorer で Setup.exe ファイルを参照したと

きにも、このアイコンが表示されます。このサポートは、基本の MSI、InstallScript、および InstallScript MSI プロジェクトで使用できます。

今回より、同じ機能 (カスタム アイコンとカスタム バージョン リソースのプロパティを指定できる機能) が、基本の MSI、InstallScript MSI、および QuickPatch プロジェクトで作成された Update.exe ファイルでも使用できます。

Setup.exe および Update.exe のカスタム アイコン

[リリース] ビューの [Setup.exe] タブには、Setup.exe セットアップ 起動ツールに使用するアイコンを指定できる "Setup.exe アイコン ファイル" 設定があります。アイコンには、.exe、.dll、または .ico ファイルを使用できます。アイコンを指定しなかった場合、InstallShield は Setup.exe ファイルのデフォルト アイコンを使用します。

以前は、ビルド中の Setup.exe ファイルが自己展開型の単一セットアップ起動ツールの場合に、InstallScript プロジェクトでカスタム Setup.exe アイコンの指定がサポートされていました。これは、基本の MSI または InstallScript MSI プロジェクトではサポートされていませんでした。また、圧縮された InstallScript インストールにおいてもサポートされていませんでした。

新しいアイコン設定を使って、Update.exe 起動ツールに使用するアイコンを指定できます。この設定は、基本の MSI および InstallScript MSI プロジェクトの [パッチのデザイン] ビュー内にあるパッチの構成の [詳細] タブで行います。また、QuickPatch プロジェクトの場合、[一般情報] ビューの [ビルドの設定] 領域にある [詳細] タブで設定できます。アイコンを指定しなかった場合、InstallShield は Update.exe ファイルにデフォルト アイコンを使用します。

以前は、InstallShield は Update.exe ファイルのカスタム アイコンの指定を一切サポートしていませんでした。

Setup.exe および Update.exe のカスタム バージョン リソースのプロパティ

InstallShield がビルド時に Setup.exe 起動ツールの以下のバージョン リソースを構成するとき、今回より、[一般情報] ビュー および [リリース] ビューで入力されたカスタム情報を使用します。

- 会社名
- 製品名
- 製品バージョン
- 著作権情報
- ファイル バージョン
- ファイルの説明

カスタム著作権情報およびカスタム ファイルの説明を使用する場合、[リリース] ビューで "カスタム バージョンのプロパティを使用する" 設定で [はい] を選択してください。

以前、InstallShield は多くの場合にカスタム情報を使用していませんでした。たとえば、InstallShield が以前に作成した InstallScript Setup.exe ファイルには、その Setup.exe ファイルをビルドした InstallShield のバージョン固有の詳細が含まれました。そのため、Setup.exe プロパティ ダイアログ ボックスに表示される製品名は、インストールする製品の名前ではなく、InstallShield と表示されました。

オートメーション インターフェイスの ISWiRelease オブジェクトには、今回より、Setup.exe のファイルの説明を設定できる LauncherFileDescription プロパティが含まれています。既存プロパティのいくつかを使って、Setup.exe のその他のバージョン リソースのプロパティを構成できます。ISWiProject オブジェクトには、CompanyName、ProductName、および ProductVersion プロパティが含まれています。ISWiRelease オブジェクトには、LauncherCopyright および UseMyVersionInfo ("カスタム バージョンのプロパティを使用する" 設定に相応する) プロパティが含まれています。

"会社名"、"製品名"、"製品バージョン"、"説明"、"著作権情報" といった新しい設定を使って、InstallShield で Update.exe ファイルをビルドするとき使用するカスタム情報を指定できます。これらの設定は、基本の MSI および InstallScript MSI プロジェクトの [パッチのデザイン] ビュー内にあるパッチの構成の [詳細] タブで行います。また、QuickPatch プロジェクトの場合、[一般情報] ビューの [ビルドの設定] 領域にある [詳細] タブで設定できます。

以前、InstallShield は Update.exe ファイルにカスタム バージョン リソース情報を使用していませんでした。

この機能によって、IOA-000031784、IOA-000035007、IOA-000043150、IOA-000052852、IOA-000056121、IOC-000051102、IOC-000055158、IOC-000070983 が解決されます。

ビルド前、ビルド中、およびビルド後に実行するコマンドを指定する機能

InstallShield Premier Edition には、ビルド プロセスの様々な段階で実行するコマンドを指定することができる、新しいリリースの設定が追加されました。これらの新しい設定は、[リリース] ビューでリリースを選択したときに、新しい [イベント] タブに表示されます。

- **ビルド前のイベント** — この設定を使って、InstallShield がリリースのビルドを開始する前に実行するコマンドを指定します。このイベントは InstallShield がリリース フォルダとログ ファイルを作成した後、リリースのビルドを開始する前に実行します。

この設定は、基本の MSI、InstallScript MSI、およびマージ モジュール プロジェクトで使用できます。

- **圧縮前のイベント** — この設定を使って、製品のデータ ファイルを .cab ファイルに格納する場合、InstallShield が .msi パッケージと .cab ファイルをビルドした後に実行するコマンドを指定します。このイベントは .cab ファイルが .msi パッケージにストリームされた後、.msi パッケージにデジタル署名が行われて Setup.exe ファイルにストリームされる前に発生します。

この設定は、基本の MSI および InstallScript MSI プロジェクトで使用できます。

- **ビルド後のイベント** — この設定を使って、InstallShield がリリースをビルドして署名を行った後に実行するコマンドを指定します。

この設定は、基本の MSI、InstallScript MSI、およびマージ モジュール プロジェクトで使用できます。

新しい [イベント] タブは、以前の [ポストビルド] タブに取って代わります。以前 [ポストビルド] タブで提供されていた設定は、新しい [イベント] タブに表示されます。また、以前マージ モジュール プロジェクトの [ビルド] タブにあったパブリッシュ関連の設定は [イベント] タブに移動しました。

オートメーション インターフェイスは、この新しいビルド イベントの設定をサポートします。ISWiRelease オブジェクトには、ビルド プロセスの様々な段階におけるコマンドを指定できる 3 つの新しいプロパティが含まれています。

- PrebuildEvent
- PrecompressionEvent
- PostbuildEvent

Setup.exe に有効期限日を設定する機能

InstallShield では、今回より、Setup.exe に有効期限日と有効期限メッセージを設定できます。エンド ユーザーが Setup.exe をプロジェクトで指定された日付以降に実行すると、有効期限メッセージが表示されて、インストールが終了します。

Setup.exe ファイルの有効期限日とメッセージの設定は、[リリース] ビューで選択されたリリースの [Setup.exe] タブに新しく追加された "有効期限日と有効期限メッセージ" 設定で行います。デフォルトで、Setup.exe に有効期限日は構成されていません。有効期限日を指定した場合、デフォルトの有効期限切れメッセージを使用するか、独自のカスタム メッセージを指定できます。

オートメーション インターフェイスには、これらの新しい設定のサポートが含まれています。ISWiRelease オブジェクトには、有効期限日とメッセージを設定できる新しいプロパティ (ExpirationDate および ExpirationMessage) が含まれています。

この機能は、基本の MSI および InstallScript MSI プロジェクト タイプで使用できます。

Visual Studio セットアップとマージ モジュール プロジェクトを既存の InstallShield プロジェクトにインポートする機能 (プロジェクト コンバーターの強化)

InstallShield では、今回より、Visual Studio セットアップ プロジェクト (.vdproj) を 基本の MSI またはマージ モジュール プロジェクト (.ism) にインポートするか、Visual Studio マージ モジュール プロジェクト (.vdproj) を InstallShield 基本の MSI またはマージ モジュール プロジェクト (.ism) にインポートすることができます。これらのタスクを行うと、Visual Studio プロジェクトに含まれている同じデータと設定を含む InstallShield インストール プロジェクトおよびマージ モジュール プロジェクトを作成できます。ウィザードを使って、プロジェクト出力、ファイル、レジストリ キー、ファイル拡張子、カスタム アクション、ターゲット システム検索、および起動条件を Visual Studio プロジェクトから既存の InstallShield プロジェクトにインポートします。

Visual Studio プロジェクトを既存の InstallShield プロジェクトにインポートするには、InstallShield の Visual Studio デプロイメント プロジェクト インポート ウィザードを使います。このウィザードでは、Visual Studio プロジェクト内の特定の設定をインポートするか、無視するかを選択することができます。

Visual Studio プロジェクトを新しい InstallShield プロジェクトに変換するための、これまでのサポートが拡張されました。Visual Studio プロジェクトに定義済み前提条件が含まれている場合、InstallShield は今回より、プロジェクトの変換中にそれらに対応する InstallShield 前提条件に変換します。これと同じ前提条件の変換機能が、Visual Studio プロジェクトを InstallShield プロジェクトにインポートするための新しいウィザードにも搭載されています。

Visual Studio プロジェクトに 1 つ以上のプロジェクト出力が含まれている場合は、変換プロセスの代わりにインポート ウィザードを使用します。InstallShield プロジェクトは、Visual Studio セットアップまたはマージ モジュール プロジェクトおよびそのプロジェクト依存関係が含まれているのと同じ Visual Studio ソリューションに含まれていなくてはなりません。インポート ウィザードを使ってプロジェクト出力を InstallShield プロジェクトにインポートするためには、InstallShield を Visual Studio 内部で開いてください。

SQL スクリプトにおける Unicode および UTF-8 サポート

InstallShield は、Unicode BOM エンコードまたは UTF-8 BOM エンコードのいずれかを使った SQL スクリプトをデザイン時および実行時にサポートします。[SQL スクリプト] ビューを使って、どちらかのエンコードを使った SQL スクリプトをプロジェクトに追加できます。これらの SQL スクリプトを、必要に応じて [SQL スクリプト] ビュー内から編集することもできます。実行時に、必要に応じてインストールおよびアンインストール中に SQL スクリプトが実行されます。

以前、InstallShield では Unicode BOM エンコード を使った SQL スクリプトが実行時にはサポートされていましたが、デザイン時にはサポートされていませんでした。したがって、このエンコードを使ったスクリプトをプロジェクトの [SQL スクリプト] ビューに追加すると、InstallShield がスクリプトを ANSI 形式に変換するかどうかを指定するためのプロンプトが表示されました。InstallShield で ANSI への変換を許可すると、それを [SQL スクリプト] ビュー内から編集できます。ただし、一部の状況において、デザイン時と実行時に文字化けが発生することがありました。InstallShield が ANSI に変換することを許可しなかった場合、そのファイルは Unicode BOM エンコードのままとなりました。このエンコードを使うと、インストールは実行時にターゲット システムのコード ページと一致しない言語の文字列を正しく使用しますが、InstallShield 内部からスクリプトを参照または編集することができませんでした。

さらに以前、InstallShield は UTF-8 BOM エンコードを使用する SQL スクリプトを適切にサポートしていませんでした。このエンコード タイプを使用する SQL スクリプトをプロジェクトに追加して、スクリプトに開発システムのコード ページと一致しない言語の文字列が含まれていた場合、[SQL スクリプト] ビューは、そのファイルを ANSI エンコードとして処理するために、文字化けが発生することがありました。また、実行時に SQL スクリプトが実行されると、文字化けが発生しました。バイト オーダー マークも文字化けして表示されました。

[SQL スクリプト] ビュー内から新しい SQL スクリプトを作成すると、InstallShield はそのファイルに Unicode BOM エンコードを使用します。[SQL スクリプト] ビューで ANSI または UTF-8 BOM エンコードを使用したい場合、別のツールを使って適切なエンコードを用いて .sql ファイルを作成してから、プロジェクトの [SQL スクリプト] ビューにそのスクリプトをインポートまたは挿入することをお勧めします。

この機能は、基本の MSI、InstallScript、および InstallScript MSI プロジェクト タイプで使用できます。

この機能により IOC-000066750 が解決されました。

SQL Server 2008 Express SP1 および Adobe Reader 9 のための定義済みシステム検索

InstallShield に新しい定義済みシステム検索が追加されました:

- SQL Server 2008 Express SP1
- Adobe Reader 9

インストールでこれらの 1 つまたは両方が必要な場合、[システム検索] ビューまたはプロジェクト アシスタントの [インストール要件] ページを使って、これらのシステム検索をプロジェクトに追加することができます。エンドユーザーがインストールを起動すると、Windows Installer はターゲット システムの要件が満たされているかどうかを確認します。要件が満たされていない場合、インストールでシステム検索用に定義されているエラー メッセージが表示されます。

この機能は、基本の MSI および InstallScript MSI プロジェクトに適用します。

リパッケージャーによる 64 ビット サポート

今回より、リパッケージャーを使って、64 ビット インストールをリパッケージできます。以前は 32 ビット インストールのみをリパッケージすることができました。

リパッケージ ウィザードは 32 ビット アプリケーションのままですが、このウィザードを 32 ビット (x86) および 64 ビット (x64) Windows システムの両方で実行することができます。

64 ビット システム上でリパッケージャーを使用して、64 ビットまたは 32 ビットシステムの両方で実行可能な 32 ビット インストールをリパッケージすることができますが、誤って 64 ビット データ (64 ビット ディレクトリまたはレジストリの場所など) をキャプチャすることを避けるため、リパッケージャーを 32 ビット システム上で使用することをお勧めします。リパッケージャーが 64 ビット データをキャプチャすると、そのパッケージを 64 ビット インストールとしてフラグを含みます。その結果、パッケージは 64 ビット システムでのみ実行可能となります。

リパッケージャーは、InstallShield の Premier Edition で提供されています。

64 ビット マネージコード カスタム アクションのサポート

InstallShield は、今回より、64 ビット マネージコード カスタム アクションをサポートします。プロジェクトでマネージコード カスタム アクションを含むリリースをビルドするとき、InstallShield はカスタム アクションと関連付けられたメインの .NET アセンブリのターゲット アーキテクチャ (32 ビットまたは 64 ビット) を判別しようとします。InstallShield は、実行時にマネージコードを実行するときに .NET Framework の適切なバージョン (32 ビットまたは 64 ビット) が使用されるようにリリースを構成します。

新しいデフォルトの動作をオーバーライドするには、[ダイレクト エディター] ビューを使って、次のフィールドを含む新しいレコードを ISClrWrap テーブルに追加します。

- **Action_** — 変更するマネージコード カスタム アクションの名前を示します。
- **Name** — 次を入力します: TargetPlatform
- **Value** — 適切なアーキテクチャを指定します。 .NET Framework の 32 ビット バージョンを使用する場合、次を入力します: x86
.NET Framework の 64 ビット バージョンを使用する場合、次を入力します: x64

この機能は、基本の MSI、InstallScript MSI、およびマージ モジュール プロジェクト タイプで使用できます。

この機能により IOA-000051744 が解決されます。

64 ビット .NET Installer クラスと COM Interop のサポート

InstallShield は今回より、64 ビット .NET Installer クラスと COM Interop をサポートします。64 ビット システム上で InstallShield を使用している場合、InstallShield の [ツール] メニューにある [オプション] をクリックすると表示される [オプション] ボックスでは、今回より、.NET Framework に含まれている Regasm.exe および InstallUtilLib.dll ファイルの場所として、32 ビットと 64 ビットの 2 つのパスを指定できます。InstallShield は、.NET Installer クラスと COM Interop を含むリリースで、ビルド時に指定されたパスを使用します。

ISCmdBld.exe を使ってコマンドラインからビルドしている場合に、既存の -t パラメーターを使って .NET Framework の 32 ビットバージョンのパスを指定すると、ISCmdBld.exe は今回より、適切な場合に Regasm.exe および InstallUtilLib.dll の 64 ビットの場所を使用します。

MSBuild または Team Foundation Server (TFS) を使ってビルドしている場合に、InstallShield タスクで既存の DotNetFrameworkVersion パラメーターを使って .NET Framework の 32 ビットバージョンのパスを指定すると、ビルドは今回より、適切な場合に Regasm.exe および InstallUtilLib.dll の 64 ビットの場所を使用します。

この機能により IOA-000056407 が解決されます。

.NET Framework 4 をターゲットにする DLL で InstallScript 関数 DotNetCoCreateObject またはマネージコード カスタム アクションの使用をサポート

Visual Studio 2010 で作成した DLL が .NET Framework 4 を使用する場合に、InstallScript 関数 DotNetCoCreateObject を使って、この DLL で関数を呼び出すことができます。以前、DLL が .NET Framework の以前のバージョンを使用したときにはインストールはクラッシュしませんでした。バージョン 4 を使用したときに、インストールがクラッシュしました。この機能は InstallScript と InstallScript MSI プロジェクトで使用できます。またこの機能は、InstallScript カスタム アクションを含む基本の MSI およびマージ モジュール プロジェクトでも使用できます。

さらに、今回より、マネージコード カスタム アクションで同じような DLL ファイルを使用できます。以前は、インストールに .NET Framework のバージョン 4 を使用する DLL のマネージコード カスタム アクションが含まれている場合、クラッシュが起きました。この機能は、基本の MSI、InstallScript MSI、およびマージ モジュール プロジェクト タイプで使用できます。

この機能により IOA-00005646 が解決されます。

リリースのパス変数をオーバーライドする機能

InstallShield では、今回より、プロジェクト内の各リリースでプロジェクトのユーザー定義のパス変数、環境変数、およびレジストリ変数の値をオーバーライドすることができます。この機能を使って、ビルドする特定のリリースごとに、ビルド時にプロジェクト内の特定のファイルとフォルダーを別のファイルとフォルダーに置換することができます。

たとえば、この機能を使ってカスタム アクションのバイナリ ファイルを入れ替えることができます。個別の 32 ビットと 64 ビットのターゲット システム用に個別のリリースを作成した場合、カスタム アクションに選択された DLL を参照するパス変数をオーバーライドすることができます。これによって、InstallShield が 32 ビット リリースには 32 ビット DLL を、また 64 ビット リリースには 64 ビット DLL を含みます。インストールがインストール処理を行っているファイルを入れ替えるのにパス変数のオーバーライドを行うことはお勧めできません。これは、ファイルの 32 ビットバージョンと 64 ビットバージョンに個別のコンポーネントを使用しなくてはならないためです。

プロジェクトで 1 つ以上のパス変数をオーバーライドする場合、[リリース] ビュー内のリリースの [ビルド] タブに追加された "パス変数のオーバーライド" 設定を使います。新しい "パス変数のオーバーライド" 設定と、IsCmdBld.exe または MSBuild で -t パラメーターを使ってパス変数をオーバーライドした場合、コマンドラインまたは MSBuild 値がリリース設定で指定された値よりも優先します。

オートメーション インターフェイスには、この新しい設定のサポートが含まれています。ISWiRelease オブジェクトには、リリースに含まれるプロジェクトのユーザー定義のパス変数、環境変数、およびレジストリ変数の値をオーバーライドできる、新しい PathVariableOverrides プロパティが含まれています。

この機能は、基本の MSI、InstallScript、InstallScript MSI、InstallScript オブジェクト、およびマージ モジュール プロジェクト タイプで使用できます。

IIS Web サイト、アプリケーション、および仮想ディレクトリの MIME の種類を構成する機能

[IIS 構成] ビューに新しく追加された "MIME の種類" 設定を使って、プロジェクトに含まれる Web サイト、アプリケーション、または仮想ディレクトリの MIME の種類を構成できます。この設定を使って、ターゲット システム上の Web サーバーからブラウザまたはメール クライアントに送信できるコンテンツの種類を指定できます。

この機能は、基本の MSI、InstallScript、および InstallScript MSI プロジェクト タイプに適用します。

この機能により 1-G7VZH が解決されます。

既存の IIS アプリケーション プールを上書きする、またはそれが既存しない場合のみ作成する機能

[IIS 構成] ビューに、新しい "既存のアプリケーション プールを上書きする" 設定が追加されました。この設定は、[IIS 構成] ビューでアプリケーション プールを選択したときに、右側のペインに表示されます。この設定を使って、実行時にターゲット システム上に既に選択されたアプリケーション プールが存在する場合に起こる動作を指定できます。インストールによって、ターゲット システム上のアプリケーション プールの設定を上書きするか、アプリケーション プールをそのまま継続させることができます。この設定のデフォルト値は [はい] で、実行時に既存のアプリケーション プールが上書きされます。

この機能は、基本の MSI、InstallScript、および InstallScript MSI プロジェクト タイプに適用します。

この機能によって、IOA-000052593 と IOC-000066151 の問題が解決されます。

InstallScript および InstallScript MSI プロジェクトにおける新しいビルボード スタイル

InstallScript および InstallScript MSI プロジェクトには、進行状況ダイアログの進行状況バーの上に表示されるビルボードの新しいスタイルがサポートされています。このビルボード スタイルは、イメージ ファイル (.bmp, .gif, .jpg, and .jpeg) だけでなく、Adobe Flash アプリケーション ファイル (.swf) もサポートします。

プロジェクトに Flash ファイルとイメージ ファイルの両方が含まれているとき、ターゲット システムに Adobe Flash Player がインストールされていない場合、インストールはこれを検出して Flash ビルボードの代わりにイメージ ビルボードを表示します。

以前、InstallScript および InstallScript MSI プロジェクトで使用できる唯一のビルボード サポートでは、背景ウィンドウを使用する必要がありました。新しいビルボード スタイルで、背景ウィンドウは不要です。

新しいビルボード スタイルをプロジェクトに追加するには、[サポート ファイル/ビルボード] ビューを使って、プロジェクトにビルボード ファイルを追加します。実行時に、この新しいスタイルのビルボードを表示するには、新しい STATUSBBRD 定数を Enable 関数と一緒に使います。

新しいスタイルのビルボードは、スキン ダイアログを使用するプロジェクトでは使用できません。

基本の MSI プロジェクトにおけるイメージ ビルボードのループ サポート

基本の MSI プロジェクトの [ビルボード] ビューに新しく追加された "ビルボードのループ" 設定を使って、インストールがファイルの転送を完了するまでイメージ ビルボードを継続してループ表示してから適切な SetupComplete ダイアログを表示するかどうかを指定できます。

この設定で [はい] を選択して、ビルボードに割り当てた時間よりもファイルの転送に時間がかかった場合、インストールは最初のビルボードから再び表示します。必要な場合、ループはファイルの転送が終了するまで続きます。この設定のデフォルト値は [いいえ] です。これは、InstallShield の以前のバージョンの動作と同じです。

以前は、ビルボードに割り当てられた時間よりもファイルの転送に時間がかかった場合、インストールはファイル転送が終了するまで最後のビルボードを表示し続け、ビルボードをループしませんでした。

この機能により IOA-00005335 が解決されました。

Windows Installer 5 における Windows サービスのアクセス許可の構成サポート

InstallShield は今回より、Windows サービスのアクセス許可の構成をサポートします。このサポートは Windows Installer 5 で提供されている MsiLockPermissionsEx テーブルを使用します。Windows Installer 5 は、新しいサービスのアクセス許可の設定をサポートします。以前の Windows Installer は、これらの設定を無視します。

新しいサービスのアクセス許可設定を構成するには、[セットアップのデザイン] ビュー (インストール プロジェクトの場合) または [コンポーネント] ビューで、コンポーネントの [詳細設定] 領域にある [サービス] ノードを使用します。[サービス] ノードで、プロジェクトに新しいサービスを追加します。[サービス] ノードの下にあるサブノードを選択すると、右側のペインでそのサービスの新しいアクセス許可と関連設定を構成できます。

この機能は、基本の MSI、InstallScript MSI、マージ モジュール、MSI データベース、MSM データベース、およびトランスフォーム プロジェクト タイプで使用できます。

InstallScript プロジェクトのダイナミック ファイル リnkの構成におけるオートメーション インターフェイス サポート

InstallScript プロジェクトで、オートメーション インターフェイスの ISWiDynamicFileLinking オブジェクトおよび ISWiDynamicFileLinkings コレクションに、ダイナミック ファイル リンク サポートが追加されました。さらに今回より、InstallScript プロジェクトで ISWiComponent オブジェクトの 2 つのメソッドとプロパティを使用できます：
AddDynamicFileLinking メソッドは、コンポーネントに新しいダイナミック ファイル リンクを追加し、
RemoveDynamicFileLinking メソッドは、コンポーネントのダイナミック ファイル リンクを削除し、
ISWiDynamicFileLinkings プロパティは、ダイナミック ファイル リンクのコンポーネントのコレクションを取得します。
ISWiFile オブジェクトの DynamicFile 読み取り専用プロパティを InstallScript プロジェクトでも使用できるようになりました：このプロパティを使って、ファイルのソースがコンポーネントにダイナミックにリンクされているか、スタティックにリンクされているかを判別します。

この機能により IOC-000082900 が解決されました。

新しい FlexNet Connect 12.01 再配布可能ファイル

InstallShield はまた、基本の MSI プロジェクトおよび InstallScript MSI プロジェクトで FlexNet Connect 12.01 をサポートします。InstallShield の [アップデート通知] ビューで、2 つの FlexNet Connect 12.01 マージ モジュール (Common Software Manager が含まれているマージ モジュールと、含まれていないマージ モジュール) のどちらかを含みます。

強化機能

InstallShield の Unicode ビュー

InstallShield のいくつかのビューと領域は、すべての言語からの文字を表示および入力できるように強化されています。たとえば、今回より、英語版のコンピュータ上で [システム検索] ビューにおいてシステム検索を構成するときに、ファイル名、パス、レジストリ エントリに中国語の文字を使用できます。以前、InstallShield のこれらの領域で表示される文字は疑問符として表示されました。

InstallShield で Unicode サポートが強化された領域は、[システム検索] ビュー、および [リリース] ビュー内のリリースの [複数インスタンス] タブ、および [リリース] ビュー内のリリースに表示されるタブです。InstallShield 2010 より、その他の多くのビューで Unicode がサポートされています。

[システム検索] ビューの強化は、基本の MSI、InstallScript MSI、マージ モジュール、MSI データベース、MSM データベース、およびトランスフォーム プロジェクトに適用します。

[リリース] ビュー内のリリースのタブにおける強化は、基本の MSI、InstallScript、InstallScript MSI、InstallScript オブジェクト、およびマージ モジュール プロジェクトに適用します。

[リリース] ビューの [複数インスタンス] タブの強化は、基本の MSI プロジェクトに適用します。

InstallShield MSI Diff を使用中にログファイルを生成するためのコマンドライン サポート

InstallShield MSI Diff には、新しく /out コマンドライン パラメーターが含まれています。InstallShield MSI Diff を実行中に、コマンドラインからこの新しいパラメーターを使って、2 つの .msi、.msm、または .pcp ファイル間の差分を確認したり、トランスフォーム (.mst) またはパッチ ファイル (.msp) によって Windows Installer データベースに適用される変更点を表示するログ ファイルを生成できます。また、このツールをコマンドラインから使用して、バイナリ形式で保存されている 2 つの InstallShield プロジェクト ファイル (.ism または .ise) 間の違いを識別するログ ファイルを生成することもできます。

この強化により、IOA-000054055、IOC-000076072 が解決されます。

ランタイム ダイアログで Microsoft SQL Server のローカル インスタンスを 64 ビット システムでリストするサポート

InstallShield の [SQL スクリプト] ビューを使ってプロジェクトに SQL サポートを追加してから、そのインストールを 64 ビット ターゲット システムで実行すると、SQL 関連のビルトイン ランタイム ダイアログは、今回より、32 ビットの SQL Server のローカル インスタンス、32 ビットの SQL Server のネットワーク インスタンス、および 64 ビットの SQL Server のネットワーク インスタンスだけでなく、64 ビットの SQL Server のローカル インスタンスもリストします。以前は、インストールが 64 ビット ターゲット システムで実行された場合、SQL 関連のビルトイン ランタイム ダイアログは 64 ビットの SQL Server のローカル インスタンスをリストせず、32 ビットのローカル インスタンス、32 ビットのネットワーク インスタンス、および 64 ビットのネットワーク インスタンスのみがリストされました。

この機能は、基本の MSI、InstallScript、および InstallScript MSI プロジェクトで利用できます。

この強化によって、IOA-000054153、IOB-000056728、IOC-000053557 が解決されます。

64 ビット レジストリの変更についての InstallScript ログ記録とアンインストール

今回より、64 ビット ターゲット システム上で InstallScript または InstallScript MSI インストールが行われるときに、デフォルトで InstallScript エンジンがレジストリの 64 ビット部分に行われる変更をログ記録します。さらに、InstallScript エンジンによってログ記録された 64 ビット レジストリの変更が、今回より、アンインストール中にアンインストールされます。

以前、InstallScript エンジンがインストール中にレジストリの 64 ビット部分に変更を加えた場合、これらの変更は 64 ビット特定の変更としてログ記録されませんでした。そのためアンインストール中に削除も行われませんでした。

この強化により IOC-000046694 が解決されます。

Microsoft SQL Server Management Studio のサポート

[SQL スクリプト] ビューでは、今回より、Microsoft SQL Server Management Studio がサポートされています。プロジェクト内の SQL スクリプト ファイルを SQL Server Management Studio で開くには、[SQL スクリプト] ビューで SQL スクリプトを右クリックしてから、[Microsoft SQL Server Management Studio でスクリプトを開く] をクリックします。この新しいコマンドは、同じコンテキスト メニューにあった [Microsoft Query Analyzer でスクリプトを開く] の代わりとなります。

この機能は、基本の MSI、InstallScript、および InstallScript MSI プロジェクトで利用できます。

この強化により IOC-000080976 の問題が解決されます。

Visual Studio Web セットアップ プロジェクトからプロジェクト出力を追加するサポート

Web セットアップ プロジェクトを含む Visual Studio ソリューションと、InstallShield インストール プロジェクトを作成した場合に、Visual Studio 内部から InstallShield を使用しているとき、今回より、Web セットアップ プロジェクトのプロジェクト出力を InstallShield プロジェクトに追加することが出来ます。

この強化により IOC-000081640 が解決されます。

テキスト ファイルの変更を構成するときに Windows Installer プロパティを選択できる機能

[テキスト ファイルの変更] ビュー内の 2 つの設定について、テキスト ファイルのコンテンツにおける検索置換動作をより簡単に構成できるように強化されています。[テキスト ファイルの変更] ビューで置換ノードを選択すると右側のペインに表示される "検索する文字列" 設定と "置換後の文字列" 設定では、今回より、リストから Windows Installer プロパティを選択するか、文字列を入力することができます。このリストには、プロジェクトの [プロパティ マネージャー] ビューで使用可能なすべてのプロパティが含まれています。以前、これらの設定では手動で文字列またはプロパティを入力できましたが、プロパティのリストは表示されませんでした。

この強化は、基本の MSI および InstallScript MSI プロジェクトで使用できます。

この強化により IOA-000051122 の問題が解決されます。

IISROOTFOLDER で環境変数パスを解決する InstallScript テキスト置換の強化

InstallScript インストールには、IIS PathWWWRoot レジストリ値に環境変数が含まれている場合に、IIS データを IISROOTFOLDER にインストールするためのサポートが追加されました。以前、IIS サポートを含む InstallScript インストールで IISROOTFOLDER 変数のテキスト置換が行われた場合、Windows Server 208 以降のシステム上では、IISROOTFOLDER の環境変数部分が解決されませんでした。コンポーネントが IISROOTFOLDER にインストールされるように構成されている場合、未解決のパスが無効であるためにインストールが失敗しました。

この強化により IOA-000051120 の問題が解決されます。

Update.exe マニフェストに必要実行レベルを指定する機能

"必要実行レベル" 設定は、基本の MSI および InstallScript MSI プロジェクトの [パッチのデザイン] ビューにあるパッチの構成の [詳細] タブで設定できます。また、QuickPatch プロジェクトの場合、[一般情報] ビューの [ビルドの設定] 領域にある [詳細] タブで設定できます。

これらの新しい設定を使って、Windows Vista 以降のプラットフォーム上で Update.exe ファイルがアップグレードを実行するために必要とする最低実行レベルを指定します。InstallShield が、必要レベルを指定するためのマニフェストを追加します。デフォルトで、InstallShield は以前のセットアップ起動ツールのマニフェストで構成されたレベルを使用します。

以前、"必要実行レベル" 設定は Setup.exe セットアップ起動ツールでのみ使用できました。Update.exe パッチを作成する場合、InstallShield は以前のセットアップ起動ツールのマニフェストで構成された必要実行レベルと同じレベルを使います。

この強化により、IOA-000055255、IOC-000079301 が解決されます。

非圧縮 .cab ファイルの作成機能

[リリース] ビューで選択されたリリースの [ビルド] タブに、新しい "Cab の最適化タイプ" 設定が追加されました。"圧縮" 設定で [圧縮] またはカスタム オプションの 1 つを選択した場合、"Cab 最適化タイプ" 設定を使って、リリースの .cab ファイルをビルドするときに InstallShield が使用する圧縮の種類を指定します。使用可能なオプションには、[LZX 圧縮]、[MSZIP 圧縮]、または [非圧縮] があります。

"Cab の最適化タイプ" 設定は、これまでの "サイズの最適化" 設定に取って代わります。"サイズの最適化" 設定は、LZX 圧縮と MSZIP 圧縮のみをサポートし、非圧縮オプションは提供されていませんでした。

オートメーション インターフェイスには、この新しい設定のサポートが含まれています。ISWiRelease オブジェクトには、新しく CabCompressionType プロパティが追加されました。このプロパティを使って、オートメーション インターフェイスを通してリリースをビルドするときに、3 つの圧縮オプションの中から 1 つを指定できます。

この強化は、基本の MSI および InstallScript MSI プロジェクトで使用できます。

この強化により IOA-000053420 が解決されました。

InstallShield 前提条件エディターにおける定義済みオペレーティング システム条件のリスト拡張

InstallShield 前提条件エディターを使って InstallShield 前提条件の設定を追加または変更するときに表示される [前提条件設定] ダイアログ ボックスに、Windows Server 2008 R2 用の定義済みオペレーティング システム条件が追加されました。一部の Windows 7 オプションは、ワークステーションのバージョン、サーバーのバージョン、またはいずれかのバージョンをチェックするため、どの Windows バージョンが選択されているかをより明確に示すように、これらのオプションの名前が変更されました。[前提条件設定] ダイアログ ボックスには、Windows Server 2008 R2 ではなく Windows 7 のみをチェックする新しいオプションも追加されました。これらの変更によって、任意の InstallShield 前提条件について、よりターゲットを絞ったオペレーティング システム条件を素早く定義することができます。

InstallShield 前提条件は、基本の MSI、InstallScript、および InstallScript MSI プロジェクトに追加できます。

この強化により IOA-000054066 の問題が解決されます。

[再配布可能ファイル] ビューと [前提条件] ビューの新しい [更新] ボタン

[再配布可能ファイル] ビュー (基本の MSI、InstallScript MSI、およびトランスフォーム プロジェクト) および [前提条件] ビュー (InstallScript プロジェクト) に表示される再配布可能ファイルのリストを更新するための、新しい [更新] ボタンが追加されました。以前は、InstallShield でいずれかのビューが開かれている状態でコンピュータに再配布可能ファイルを追加したとき、更新済みのリストを表示するためには、プロジェクトを一度閉じてから再び開く必要がありました。

この強化は、基本の MSI、InstallScript、InstallScript MSI、およびトランスフォーム プロジェクトで使用できます。

この強化により IOA-000051535 が解決されます。

IUSR ユーザー アカウントのアクセス許可設定におけるサポート

InstallShield は、よく知られるユーザー アカウント IUSR のファイル、フォルダー、およびレジストリ キーの保護をサポートします。

プロジェクトでアクセス許可の構成にカスタム InstallShield 処理を使用する場合、[アクセス許可] ダイアログ ボックスの [ユーザー] リストに新しい [IUSR] オプションが追加されています。カスタム InstallShield 処理を使う場合、[一般情報] ビューの "ロックダウンの設定方法" 設定でこのオプションを選択します。この機能は、基本の MSI、InstallScript MSI、マージ モジュール、およびトランスフォーム プロジェクト タイプで使用できます。

InstallScript 関数 SetObjectPermissions を使ってアクセス許可を設定する場合、今回より、szUser パラメーターで IUSR を渡すことができます。この関数は、InstallScript プロジェクトおよび InstallScript MSI プロジェクトの InstallScript イベントで使用できます。基本の MSI、InstallScript MSI、およびマージ モジュールプロジェクトでは、InstallScript カスタム アクションを使って、この関数を使用することができます。

この強化により IOC-000080797 が解決されます。

SetObjectPermissions 関数の新しい IS_PERMISSIONS_OPTION_ALLOW_ACCESS 定数

InstallScript 関数 SetObjectPermissions と使用できる、IS_PERMISSIONS_OPTION_ALLOW_ACCESS 定数が追加されました。SetObjectPermissions の nOptions パラメーターで、数値 0 の代わりにこの定数を渡して、設定されている許可が特定のファイル、フォルダー、またはレジストリ キーへのアクセスを可能とすることを示します。

この関数は、InstallScript プロジェクトおよび InstallScript MSI プロジェクトの InstallScript イベントで使用できます。基本の MSI、InstallScript MSI、およびマージ モジュールプロジェクトでは、InstallScript カスタム アクションを使って、この関数を使用することができます。

この強化により IOC-000081824 が解決されます。

DialogSetInfo 関数の新しい定数

nInfoType パラメーターに DLG_INFO_ALTIMAGE を使用するとき、InstallScript 関数 DialogSetInfo の nParameter パラメーターで 2 つの定数を利用できます:

- **DLG_INFO_ALTIMAGE_VERIFY_BMP** Ñ この定数は、szInfoString で示されたビットマップを、次に続くダイアログに使用することを指定します。このビットマップを使用する前に、インストールはそのビットマップの存在を確認します。そのビットマップが存在しない場合、関数はビットマップが見つからなかったことを示すエラーを返します。
- **DLG_INFO_ALTIMAGE_REVERT_IMAGE** Ñ この定数は、そのダイアログが、デフォルトのビットマップを表示することを指定します。この定数は、値 -1 の代替として使用できます。この定数を使用すると、インストールはビットマップの存在を確認しません。

nParameter パラメーターで TRUE を渡すと、インストールはビットマップの存在を確認しません。この状態でビットマップが存在しない場合、関数はエラーを返します。

この関数は、InstallScript プロジェクトおよび InstallScript MSI プロジェクトの InstallScript イベントで使用できます。

この強化により IOA-000054733 の問題が解決されます。

.NET Framework 4.0 サポートにおける InstallScript 言語の強化

新しい FOLDER_DOTNET_40 InstallScript 変数が追加されました。この変数は、.NET Framework 4.0 ファイルのパスを格納します。Is 関数と使用するための 2 つの新しい定数が追加されました:

- REGDB_KEYPATH_DOTNET_40_CLIENT
- REGDB_KEYPATH_DOTNET_40_FULL

プロジェクトの DIM リファレンスにおいて、ファイルとレジストリ セットに関する詳細を表示する機能

[DIM リファレンス] ビューで DIM リファレンスを選択すると、新しい 2 つのタブが表示されます:

- **ファイル セット** Ñ このタブには、選択された DIM が参照するすべてのファイル セットに関する読み取り専用の情報が表示されます。これには、スタティック ファイル、ダイナミック リンク ファイル、および空白のディレクトリが含まれます。
- **レジストリ セット** Ñ このタブには、選択された DIM が参照するレジストリ セットに関する読み取り専用の情報が表示されます。

これらの強化機能は、基本の MSI およびマージ モジュール プロジェクト タイプで使用できます。

DIM リファレンスにおけるパス変数のサポート

プロジェクトに DIM リファレンスを追加するとき、今回より InstallShield では .dim ファイルのパスに絶対パスの代わりにパス変数を使用できます。このサポートによって、異なるソース ファイルの場所を持つビルドおよび開発マシンでリリースをビルドしやすくなります。[DIM リファレンス] ビューの [全般] タブには、選択された .dim ファイルの適切なパス変数を含むパスがリストされます。

このサポートは、開発システム上で InstallShield がパス変数を使用するように構成されている場合に活用できます。(つまり、[オプション] ダイアログ ボックスの [パス変数] タブで、パス変数のサポートを許可するように設定する必要があります。) InstallShield で絶対パスを使用するように設定した場合、[DIM リファレンス] ビューの [全般] タブには .dim ファイルへの絶対パスが表示されます。

この強化は、基本の MSI およびマージ モジュール プロジェクト タイプで使用できます。

この強化により IOA-000053026 の問題が解決されます。

各パッチ構成の出力場所とキャッシュにおけるパス変数のサポート

[パッチのデザイン] ビューで選択されたパッチの構成の [共通] タブには、適切なフォルダを参照できる 2 つの設定 ("パッチ出力の場所" および "パッチ作成キャッシュ") があります。InstallShield では、フォルダに絶対パスの代わりにパス変数を使用でき

るようになりました。このサポートによって、異なるディレクトリ構造を使用するビルドおよび開発マシンでリリースをビルドし易くなります。

このサポートは、開発システム上で InstallShield がパス変数を使用するように構成されている場合に活用できます。(つまり、[オプション] ダイアログ ボックスの [パス変数] タブで、パス変数のサポートを許可するように設定する必要があります。) InstallShield で絶対パスを使用するように構成した場合、[パッチのデザイン] ビューにはフォルダへの絶対パスが表示されます。

この強化は、基本の MSI および InstallScript MSI プロジェクトで使用できます。

この強化により IOA-000054041 の問題が解決されます。

InstallScript 関数 ServiceExistsService および ServiceGetServiceState は昇格された権限を必要としません

今後、InstallScript 関数 ServiceExistsService および ServiceGetServiceState は昇格された権限を必要としません。したがって、基本の MSI インストールの [インストール] - UI シーケンスなどで、エンドユーザーが制限された権限を持つ場合でも、インストールでこれらの関数を呼び出すことができます。

この強化により IOC-000059857 が解決されました。

重要な情報

InstallShield の評価

InstallShield のライセンスを購入していなくても、InstallShield をインストールしてアクティベーションを行わず、またはライセンス サーバーに接続せずに一定の期間使用することができます。アクティベーションを行わず、またはライセンス サーバーに接続せずに使用すると、InstallShield は一部の機能が制限された評価モードで起動します。詳細については、ナレッジベース記事「Q200900」を参照してください。評価版の制限は、InstallShield がアクティベートされたとき、またはライセンス サーバーに接続して、そのライセンスがチェックアウトされたときに解除されます。

InstallShield Standalone Build を取得する

Standalone Build のインストールは、単一の圧縮された実行可能ファイルで、以下のいずれかの方法で取得できます：

- InstallShield DVD をお持ちの場合、ファイルは DVD に含まれているので DVD ブラウザーを使って見つけることができます。
- InstallShield をダウンロードされた場合、Standalone Build ライセンス情報の説明に従って、Standalone Build インストール ファイルをダウンロードすることができます。

Standalone Build のライセンス構成に関する詳細については、Standalone Build ライセンス情報を参照してください。

InstallShield の複数エディションをインストールする

InstallShield 2011 の Premier、Professional、または Express は、同時に同じシステム上に 1 つのエディションのみをインストールできます。

InstallShield の複数バージョンをインストールする

InstallShield 2011 は、同じマシン上で別のバージョンの InstallShield と共存することができます。

InstallShield 2011 Standalone Build は、同じマシン上で別のバージョンの Standalone Build と共存することができます。ほとんどの場合、InstallShield がインストールされているマシン上に Standalone Build がインストールされることはありません。この両方を同じマシン上にインストールして、オートメーション インターフェイスを使用する場合は、InstallShield ヘルプ ライブラリの「Standalone Build と InstallShield を同一マシン上にインストールする」ヘルプ トピックに記載されている、特別な登録とアンインストールの考慮について参照してください。

InstallShield と Visual Studio との統合

Microsoft Visual Studio の統合は 1 回につき InstallShield の 1 バージョンとのみ可能です。システムで最後にインストールまたは修復された InstallShield のバージョンが Visual Studio の統合に使用されます。

Visual Studio 2003 以前の統合サポートの終了

Visual Studio 内部から直接 InstallShield プロジェクトを作成、編集、およびビルドするには、Visual Studio 2005 以降が必要です。InstallShield を Visual Studio 2003 以前と統合することは出来なくなりました。

MSBuild の統合

MSBuild を使って、InstallShield プロジェクトと Visual Studio ソリューションをビルドするには、.NET Framework 3.5 以降が必要です。

InstallScript オブジェクトが非推奨となりました

InstallScript オブジェクトに代わって、InstallShield 前提条件が推奨されます。将来的なリリースで、InstallShield では InstallScript オブジェクトを作成または使用することはできません。また、定義済み InstallScript オブジェクトも提供されません。さらに、マージ モジュール ホルダー オブジェクトも使用できなくなります。InstallScript オブジェクトの代わりに、InstallShield 前提条件が推奨されます。InstallShield 前提条件エディターを使って独自の InstallShield 前提条件を作成して、InstallScript オブジェクト テクノロジーが使用できなくなるときに備えてください。これらの InstallShield 前提条件は、InstallScript、InstallScript MSI、および基本の MSI プロジェクトで共有することができます。

プロジェクトのアップグレードに関するアラート

以下は、InstallShield 2010 以前のバージョンで作成されたプロジェクトを InstallShield 2011 にアップグレードする際に発生する可能性がある問題についての情報です。また、新しい InstallShield 2011 プロジェクトと InstallShield 2010 以前のバージョンから InstallShield 2011 にアップグレードされたプロジェクト間の潜在的な動作の違いについてもアラートします。最新情報はナレッジベース記事「[Q208911](#)」を参照してください。

InstallShield の以前のバージョンで作成されたプロジェクトのアップグレードに関する一般情報

InstallShield 2011 を使って以前のバージョンで作成されたプロジェクトを開くと、プロジェクトを新しいバージョンに変換するかどうかを質問するメッセージ ボックスが表示されます。[変換する] を選択すると、変換が行われる前に、例えば 0.770 というファイル拡張子が付加されたプロジェクトのバックアップ コピーが作成されます。以前のバージョンの InstallShield でこのプロジェクトを再度開く場合、元のプロジェクトのファイル名から 0.770 を取り除いてください。InstallShield 2011 プロジェクトを以前のバージョンの InstallShield で開くことはできませんので、ご注意ください。

InstallShield 2010 以前、InstallShield 12 以前、InstallShield DevStudio、InstallShield Professional 7 以前、および InstallShield Developer 8 以前のバージョンの InstallShield で作成された既存プロジェクトを InstallShield 2011 にアップグレードできます。InstallShield MultiPlatform または InstallShield Universal で作成されたプロジェクトは InstallShield 2011 にアップグレードすることはできませんので、ご注意ください。

InstallShield は Setup.exe および Update.exe の Unicode バージョンのみをビルド (今回より、ANSI バージョンは作成できません)

今回より、InstallShield のすべてのプロジェクト タイプで、Setup.exe および Update.exe ファイルが Unicode でビルドされます。これは、InstallShield 2011 で作成される新しい基本の MSI、InstallScript、InstallScript MSI、および QuickPatch プロジェクトすべてに適用します。また、これは InstallShield の以前のバージョンから InstallShield 2011 にアップグレードされたすべてのプロジェクトにも適用します。このため、セットアップ起動ツールを Unicode と ANSI のいずれのバージョンでビルドするかを指定するための設定がなくなりました。

- 基本の MSI プロジェクトの [リリース] ビューにあるリリースの [Setup.exe] タブにあった "セットアップ ランチャの種類" 設定が削除されました。
- 基本の MSI、InstallScript MSI、および QuickPatch プロジェクトから "アップデート ランチャの種類" 設定が削除されました。この設定は、基本の MSI および InstallScript MSI プロジェクトの [パッチのデザイン] ビュー内にあるパッチの構成の [詳細] タブで使用できました。QuickPatch プロジェクトの場合、この設定は [一般情報] ビューの [ビルド設定] 領域にある [詳細] タブで使用できました。

Win32 API の定義変更

InstallScript エンジンに Unicode サポートが追加されたため、InstallScript ヘッダー ファイル ISRTWindows.h でプロトタイプ化される Win32 API 関数が更新されました。適切な場合、既存の ANSI (A) 定義のほかに、API プロトタイプのワイド (W) バージョンが追加されています。一部のプロトタイプでは、A または W バージョンが指定されていません。その場合、エンジンは W バージョンを使用しようとします。以前は A バージョンが使用されました。

InstallShield 2010 以前のプロジェクトを InstallShield 2011 にアップグレードする場合、これらの新しいプロトタイプが、同じ API のユーザー定義のプロトタイプと競合する可能性があります。可能な限り、InstallScript で提供されているプロトタイプを使用することが推奨されます。ただし、これらの Windows API に InstallScript 用に新しく追加されたプロトタイプではなく、独自のプロトタイプを使用する場合、プリプロセス定義のリストに ISINCLUDE_NO_WINAPI_H を追加します。その場合、[ビルド] メニューから [設定] を選択します。[コンパイル/リンク] タブにある [プリプロセス定義] ボックスに ISINCLUDE_NO_WINAPI_H と入力します。そうしなければ、コンパイル エラーが発生する可能性があります。

InstallScript コードにおける Unicode サポートを確認

InstallScript コードでユーザー定義の Win32 API その他の外部 DLL プロトタイプを使用する場合、API が Unicode 文字列入力をサポートするバージョンを持つとき、プロトタイプが BYVAL/BYREF WSTRING または WPOINTER を使用するように更新します。また、構造体を入力として受け取るコード内の API 関数を確認して、文字列または文字列ポインター メンバーを含む API 関数が必要な場合に Unicode として適切に宣言されていることを確認します。スクリプトで、現在 Win32 の A バージョンを呼び出す場合、W バージョンに更新します。

InstallScript プロジェクトにおける差分リリースのサポート

InstallShield 2011 以降の InstallScript プロジェクトで作成された差分リリースは、その InstallScript インストールも InstallShield 2011 以降で作成されている場合にのみ製品をアップデートできます。InstallShield 2011 以降を使って、以前の InstallScript インストールが InstallShield 2010 以前を使って作成されている製品のアップデートを作成する場合、差分リリースではなく、完全リリースを作成してください。

これは、Unicode サポートが InstallShield 2011 の InstallScript エンジンに追加された新しい機能であるためです。差分リリースを使って、ターゲット システム上で製品の以前のバージョンをアップデートした後、以前のバージョンのインストールおよび以前のランタイム エンジンを使って、メンテナンス処理を行います。InstallScript エンジンの InstallShield 2010 以前のバージョンは、Unicode 格納形式を読み取ることができないため、インストールは失敗します。

InstallScript エンジンによってインストールされた 64 ビット レジストリ エントリのアンインストール

今回より、64 ビット ターゲット システム上で InstallScript または InstallScript MSI インストールが行われるときに、デフォルトで InstallScript エンジンがレジストリの 64 ビット部分に行われる変更をログ記録します。さらに、InstallScript エンジンによってログ記録された 64 ビット レジストリの変更が、今回より、アンインストール中にアンインストールされます。

InstallShield 2010 以前を使って、64 ビット レジストリ データを作成するインストールを作成した場合、InstallShield 2011 を使って製品のアップグレードを作成するとき、ベース インストールによってログ記録された既存の 64 ビット レジストリ エントリは、製品がターゲット システムからアンインストールされるときに削除されません。この制限を回避する唯一の方法は、アンインストール中にレジストリ データを手動で削除する方法のみです。

スクリプト エディターの変更

[InstallScript スクリプト] ビューのスクリプト エディター ペイン内で、InstallShield 2010 以前のバージョンでは CTRL+SPACEBAR キーを押すとオートコンプリート機能としてビルトイン InstallScript 関数のリストが表示されましたが、今後は表示されません。今回より、オートコンプリート機能によるビルトイン InstallScript 関数のポップアップ リストを参照するには、その関数の最初のいくつかの文字を入力するだけです。ポップアップ リストには、その他の InstallScript キーワードも含まれています。

スクリプト エディター ペイン内で右クリックすると表示されるコンテキスト メニューは、InstallShield 2010 以前のバージョンとは異なります。

- コンテキスト メニューから [空白を表示] コマンドが削除されました。スクリプト エディターで空白文字や記号を表示または非表示にするには、[スクリプト エディターのプロパティ] ダイアログ ボックスを使用します。このダイアログを使って、フォント、構文の色、行番号などのスクリプト エディターのその他の設定も変更できます。このダイアログ ボックスにアクセスするには、スクリプト エディター内を右クリックしてから、[プロパティ] を選択します。
- コンテキスト メニューから、[大文字に変換する] コマンド、または [小文字に変換する] コマンドが削除されました。スクリプト エディター内のテキストを大文字に変換するには、そのテキストを選択して CTRL+SHIFT+U を押します。小文字に変換するには CTRL+U を押します。
- コンテキスト メニューから [検索] コマンドまたは [置換] コマンドが削除されました。スクリプト内で検索を行うには、CTRL+F を押してから、検索したいテキストを指定します。文字列を検索してから、それを別の文字に置換するには、CTRL+H を押してから、適切な情報を指定します。別の方法として、[編集] メニューにある [検索/置換] コマンドを使用することもできます。

改訂されたスクリプト エディターを含むビューは、[InstallScript] ビュー、[SQL スクリプト] ビュー、および [カスタム アクションとシーケンス] ビュー (このビューで VBScript または JScript ファイルを参照する場合) です。

InstallScript デバッガーの変更

InstallScript コードをデバッグするために InstallScript デバッガー (ISDbg.exe) をインストール開発マシンからデバッグを行うマシンにコピーする場合、必ずコピー先マシン上の同じフォルダに SciLexer.dll と呼ばれるファイルもコピーしてください。SciLexer.dll は、インストール開発マシン上の ISDbg.exe ファイルと同じフォルダにあります (*InstallShield Program Files フォルダ\System*)。

DPI の変更と、その InstallScript ダイアログにおける影響

1 つ以上の編集済み InstallScript ダイアログを含む、InstallShield 2010 以前で作成した InstallScript または InstallScript MSI プロジェクトを InstallShield 2011 以降にアップグレードする場合、使用中のコンピュータが InstallScript ダイアログを編集したときに選択されていたのと同じ DPI 値を使用することを確認してください。そうでない場合、実行時にダイアログのサイズに問題が生じることがあります。

プロジェクトをアップグレードした後、必要に応じて使用中のコンピュータの DPI 値を変更することが出来ます。そうすると、実行時にダイアログのサイズが正しく表示されます。1 つ以上の編集済み InstallScript ダイアログを含む InstallShield 2011 以降のプロジェクトを InstallShield の将来のバージョンにアップグレードするとき、アップグレード中に同じ DPI 値を使用する必要はありません。

既存のプロジェクトにおける既存の InstallShield 前提条件のデザイン時およびビルド時の場所変更

今回より、InstallShield が InstallShield 前提条件ファイル (.prq ファイル)、その関連データ ファイル、および依存関係を検索するフォルダーを指定できるようになりました。以前は、InstallShield が .prq ファイルを検索するフォルダーは、*InstallShield Program Files フォルダ\SetupPrerequisites* のみでした。

任意の InstallShield 前提条件を *InstallShield Program Files フォルダ\SetupPrerequisites* フォルダから、[オプション] ダイアログ ボックスの [プロパティ] タブで定義した新しいカスタム場所 (または検索パスを定義することができるその他の任意の場所) に移動させた場合、InstallShield 2010 以前のプロジェクトを InstallShield 2011 にアップグレードするときに以下の手順が必要な場合があります：

1. [再配布可能ファイル] ビューまたは [前提条件] ビューで、プロジェクトに含まれているが、カスタム場所に存在する各 InstallShield 前提条件のチェック ボックスをクリアします。 そのデータ ファイルまたは依存関係がデフォルトの場所からカスタム場所に移動されている各 InstallShield 前提条件のチェック ボックスもクリアします。
2. 新しい [更新] ボタンをクリックします。
3. 手順 1 で、プロジェクトから削除した各 InstallShield 前提条件のチェック ボックスを選択します。

InstallShield がプロジェクトの ISSetupPrerequisites テーブルから前提条件のパスを削除します。 InstallShield 2010 以前のプロジェクトでは、このテーブルに完全パスが格納されました。 前提条件のチェック ボックスをクリアしてから、[更新] ボタンをクリックしないで再び選択した場合、InstallShield は ISSetupPrerequisites テーブルにファイル名のみではなく、引き続き完全パスを使用します。

InstallShield 2010 以前のプロジェクトを InstallShield 2011 にアップグレードした場合で、InstallShield 前提条件の場所を変更してから、プロジェクトにその前提条件を追加した場合、この [更新] 手順を行う必要はありません。 また、InstallShield 2011 を使って新しいプロジェクトを作成した場合も、[更新] 手順は不要です。 両方の状況において、InstallShield はプロジェクトの ISSetupPrerequisites テーブルにパスを含みません。 このため、デフォルト パスではなく、カスタム検索パスを利用することができます。

InstallScript MSI インストールによる同製品の将来のメジャーバージョンの上書きを防ぐ

新しい InstallScript MSI プロジェクトの [アップグレード] ビューには、ISPreventDowngrade というメジャー アップグレード項目が含まれています。 この項目は、エンド ユーザーが製品の現在のバージョンで同じ製品の将来のメジャーバージョンを上書きできないように防ぎます。 InstallScript MSI プロジェクトを InstallShield 2010 以前から InstallShield 2011 にアップグレードした場合、ISPreventDowngrade 項目は自動的に追加されません。 必要に応じて、手動で ISPreventDowngrade 項目を追加することができます。 手順については、InstallShield ヘルプ ライブラリの「現在のインストールによる同製品の将来のメジャーバージョンの上書きを防ぐ」を参照してください。

Trialware サポート

Trialware ビューを含むのは InstallShield Premier Edition のみです。 このエディションでは、Try and Die タイプの Trialware を作成できます。 InstallShield は今後、Try and Buy/プロダクト アクティベーション タイプの Trialware を作成するためのサポートを含みません。

既存の InstallShield Activation Service アカウントをお持ちの場合、InstallShield 2011 で引き続き Try and Buy/プロダクト アクティベーション タイプの Trialware を作成することができます。 具体的な手順については、ナレッジベースの記事「Q200884」を参照してください。

バグ修正

1-12UC1P (InstallScript)

実行時に言語ダイアログが表示される場合に、上矢印と下矢印キーを押したとき、言語リスト内を移動することができます。

1-KBTG5

InstallShield インストール中に表示される CustomerInformation ダイアログの "シリアル番号" 設定には、単一のフィールドが表示されます。これによって、シリアル番号全体をクリップボードにコピーしてから、このフィールドに貼り付けることが可能になりました。以前、このフィールドは複数に分かれていました。そのため、コピーして貼り付けを行う場合、シリアル番号の各部分を個別にコピーおよび貼り付ける必要がありました。

IOA-000030302 (QuickPatch)

読み取り専用ファイルをアップデートする累積 QuickPatch プロジェクトをビルドしたときに、ビルド エラー -1007 が生成される問題が解決されました。

IOA-000033311 (基本の MSI、InstallScript MSI)

InstallScript 変数 DISK1SETUPEXENAME を即時モードのカスタム アクションで使用した場合に、エンド ユーザーが Setup.exe セットアップ起動ツールを起動してインストールを実行したとき、DISK1SETUPEXENAME 変数は今回より、セットアップ起動ツールのファイル名とファイル拡張子を含みます。以前、この変数は空白でした。

IOA-000044450

InstallScript 関数 PathDelete は、今回より、パスに含まれる小数点を正しく処理します。以前一部の状況において、この関数を使ってパス バッファに含まれる特定のディレクトリを削除するとき、パスに小数点が含まれている場合に誤ってパスが削除されることがありました。

IOA-000047710

プロジェクトに含まれるダイアログで、編集フィールド コントロールの Text 設定を変更した後に、プロジェクトから対応する文字列 ID と値を削除した場合、InstallShield は NEW_STRING## (## は一意の数値) など一意の文字列 ID を生成して、その文字列 ID とコントロールとを関連付けます。必要に応じて、文字列 ID とその値を変更することができます。以前この状況下では、InstallShield は ***IS_STRING_NOT_DEFINED*** の値をコントロールに割り当て、これをプロジェクトから削除する唯一の方法は、[ダイレクト エディタ] ビューを使って Control テーブルから削除する方法のみでした。

IOA-000048528 (MSI データベース)

MSI データベース プロジェクトで、ファイルを追加した後に保存すると InstallShield がクラッシュする問題が解決されました。以前、一部の状況において InstallShield がクラッシュしました。

IOA-000048747 (InstallScript)

メジャー バージョン 番号が 127 以上の InstallScript インストールを使って、メジャー バージョン 番号が 127 以下の製品をアップグレードできるようになりました。

IOA-000049470 (InstallScript)

フォントを含むコンポーネントの "アンインストール" 設定に [いいえ] が選択されている場合、そのフォントがターゲット システムに永久に残るようになりました。以前、そのコンポーネントを含む製品または機能をアンインストールすると、そのフォントもアンインストールされました。

IOA-000049854 (InstallScript)

InstallShield 日本語版を使って InstallScript One-Click Install インストールをビルドした場合に、InstallShield がデフォルトで生成する Web ページ (Setup.htm) は、今回より日本語システム上で正しく表示されます。以前、このページでは文字化けが発生しました。

IOA-000050400 (QuickPatch)

"新規 UpgradedImage フォルダを作成する" 設定に [はい] が選択されている QuickPatch パッケージをビルドするとき、パッチおよびアップグレードの検証が有効化されている場合に InstallShield がクラッシュする問題が解決されました。

IOA-000050926 (Standalone Build)

同一の Standalone Build インスタンスを使って 2 つの個別のプロジェクトからビルドをリリースしようとするとき、Standalone Build が致命的なエラー -1013 が発生する問題が解決されました。

IOA-000051118、IOB-000059480 (InstallScript)

ビルトイン InstallScript オブジェクトは、今回より、英語以外の言語を含む複数言語 InstallScript インストールが実行時に正しく機能するために必要な言語リソースを含みます。

IOA-000051393 (InstallScript、InstallScript MSI)

SdFeatureTree ダイアログで使用される 2 つの文字列エントリおよび終了セットアップ メッセージ ボックスで使用される 1 つの文字列エントリの韓国語翻訳が修正されました。

IOA-000051828 (基本の MSI、Merge Module)

Visual Studio プロジェクト (.vdproj) を InstallShield プロジェクト (.ism) に変換すると、言語情報が正しく変換されます。以前一部の状況において、言語情報が正しく変換されませんでした。

IOA-000051894 (基本の MSI、InstallScript MSI)

.reg ファイルをプロジェクトにインポートすると、今回より、レジストリ データにはアスタリスク フラグ (*) が付きます。アスタリスク フラグは、ターゲット システム上にそのレジストリ データが存在しない場合にはそれがインストールされ、存在する場合にはそれがアンインストールされることを示します。これは、キーまたは値を持たないインポートされたレジストリ キーに適用します。以前一部の状況下で、一部のレジストリ データにはそのフラグが付きましたが、一部のフラグにはプラス フラグ (+) が付きました。プラス フラグは、そのレジストリ データがターゲット システムにインストールされることを示します。

IOA-000051938

「データ型および定義済み構造」ヘルプ トピックの VARIANT の説明に、VARIANT データ型をデータ構造内で定義することができないことが説明されています。以前、この情報は含まれていませんでした。

IOA-000052023 (マージ モジュール)

DIM リファレンスをマージ モジュール プロジェクトに追加するとき、DIM に依存関係が含まれている場合に InstallShield がクラッシュする問題が解決されました。

IOA-000052103 (InstallScript)

Setup.gif スプラッシュ スクリーンを含む InstallScript インストールを作成したとき、今回より、実行時にスプラッシュ スクリーンが表示されます。以前は、.gif ファイルでスプラッシュ スクリーンは表示されませんでした。.bmp ファイルのみ使用できました。

IOA-000052186 (基本の MSI、InstallScript MSI)

InstallScript カスタム アクションで VarSave または VarRestore (InstallScript 関数) が呼び出されたとき、INSTALLDIR の値は nType パラメーターで SRCINSTALLDIR、VAR_SRCTARGETDIR、または SRCTARGETDIR 定数が渡された場合のみ変更されます。以前一部の状況下で、これらの定数の 1 つが渡されたかどうかにかかわらず、INSTALLDIR の値が変更されました。

IOA-000052187 (基本の MSI、Merge Module)

"Installer クラスを含む" 設定に [True] が設定されている .NET アセンブリを含む DIM ファイルへのリファレンスを追加したとき、今回より、Installer クラスの関数を呼び出すカスタム アクションが、ビルド時に基本の MSI または マージ モジュール リリースに追加されます。以前、カスタム アクションは追加されませんでした。

IOA-000052226 (InstallScript)

ルーマニア語インストールの SdSetupType2 ダイアログでは、今回より、デフォルトで [完了] ラジオ ボタンが選択されています。以前、このダイアログのラジオボタンは、いずれもデフォルトで選択されていませんでした。

IOA-000052229 (基本の MSI、InstallScript MSI、スマート デバイス)

モバイル デバイス インストールに Microsoft SQLCE 3.1 再配布可能ファイルを含めた場合、実行時にアプリケーション マネージャーがモバイル デバイスにアプリケーションをインストールできないことを通知するエラーが表示される問題が解決されました。

IOA-000052233 (マージ モジュール)

InstallShield は、マージ モジュールプロジェクトに含まれる DIM リファレンスの名前が 65 文字を超える場合に切り詰めて表示します。(追加の 7 文字は、一意の Windows Installer キー用に予約されています。)以前は、名前が 28 文字を越えるときに切り詰めて表示されました。

IOA-000052246 (基本の MSI、InstallScript MSI、マージ モジュール)

Reg-Free COM ウィザードを使うとき、対応する COM サーバーに適切でない名前を持つマニフェストを指定したときに、InstallShield がクラッシュする問題は解決されました。

IOA-000052360 (基本の MSI、InstallScript、InstallScript MSI)

[IIS 構成] ビューの "ASP スクリプト タイムアウト(秒)" 設定に値を入力すると、今回より、インストールは実行時にタイムアウト値を適切に設定します。

IOA-000052365 (マージ モジュール)

DIM リファレンスをマージ モジュール プロジェクトに追加して、その DIM にダイナミック ファイル セットが含まれている場合、今回より、ダイナミック リンク ファイルがリリースに含まれます。

IOA-000052410 (InstallScript カスタム アクションを含む基本の MSI、InstallScript、InstallScript MSI)

InstallScript プロジェクトからビルドされたインストールが、セットアップの初期化中にエラー -5009 を生成して失敗する問題が解決されました。また、InstallScript MSI プロジェクトまたは InstallScript カスタム アクションを含む基本の MSI プロジェクトからビルドされたインストールが Windows 2000 システム上でエラー 1628 を生成して失敗する問題も解決されました。以前、これらの状況下では、インストールが Windows 2000 システム上で実行されたときに、これらのランタイム エラーが発生しました。

IOA-000052500 (InstallScript、InstallScript MSI)

特定の種類のダイアログ コントロール (たとえば SdFeatureTree ダイアログの選択ツリー) が InstallScript ダイアログ上で無効である場合に、インストール時に Windows 2000 システム上で誤ったランタイム エラー 0x80040707 が発生する問題が解決されました。

IOA-000052533 (InstallScript、InstallScript MSI)

コンピュータ上のドフォルト コード ページで使用できない文字を名前に持つ言語を含むリリースをビルドするときに、プロジェクトで任意のダイアログを編集している場合、ビルド エラー -6118 が発生する問題が解決されました。

IOA-000052639 (基本の MSI)

多言語の複数インスタンス圧縮インストールの 2 番目のインスタンスが実行されたとき、今回より、コマンドプロンプト ウィンドウが表示されません。以前は、.msi パッケージが空白スペースを含む場所で圧縮された場合に、コマンドプロンプト ウィンドウが開きました。

IOA-000052661 (基本の MSI)

実行時にログ ファイルが作成されて、インストールが CostFinalize アクションの前に終了した場合、今回より SetAllUsers.dll ファイルの ShowMsiLog エントリ ポイントを使ってログ ファイルを表示できます。これは、Windows Installer ログ記録を使って複雑な起動条件や、コストイングが完了する前に失敗するその他のロジックをデバッグするとき、通常はログファイルが表示されない場合に特に便利です。

デフォルトで、ShowMsiLog カスタム アクションは、ログ ファイルを開くための Notepad を起動する実行可能ファイル カスタム アクションです。このデフォルトの実装を使って、CostFinalize アクションの前の UI シーケンスでインストールを停止すると、ランタイム エラー -2732 (「ディレクトリ マネージャーが初期化されていません」) が発生します。このランタイム エラーを

回避するためには、カスタム アクション ウィザードを使って ShowMsiLog カスタム アクションを編集して、以下の設定を構成してください:

- 種類 — Windows Installer のダイナミック リンク ライブラリの関数を呼び出す
- 場所 — Binary テーブルに保存する
- ソース — <ISProductFolder>\redist\language independent\i386\SetAllUsers.dll
- ターゲット — ShowMsiLog

ただし、この代替の実装を行うと、ログ ファイルを表示する Notepad ウィンドウが他のウィンドウの背景に開く場合があります。

IOA-000052853 (InstallScript MSI)

[テキスト ファイルの変更] ビューを使って長いテキスト文字列のテキスト置換を構成したときに、そのインストールがクラッシュする問題が解決しました。

IOA-000052856

[検証設定のカスタマイズ] ダイアログボックスは、今回より、ICE101 から ICE105 までの適切な説明を表示します。(このダイアログ ボックスにアクセスするには、[ツール] メニューから [オプション] を選択して、[検証] タブで [カスタマイズ] ボタンをクリックします。) 以前、ダイアログ ボックスは「ユーザー定義の ICE 検証規則」をリストしました。

IOA-000052895

InstallShield ヘルプ ライブラリの SRCDIR インデックス エントリは、今回より、「SRCDIR」ヘルプ トピックをポイントします。以前は、誤ったヘルプ トピックにポイントしていました。

IOA-000052897 (InstallScript)

公式のデジタル証明書 (テスト証明書ではない) を使ってデジタル署名が行われた InstallScript One-Click Install インストールを作成したときに、エンド ユーザーがセキュリティ警告に遭遇する問題が解決されました。

IOA-000052924 (InstallScript)

InstallShield 前提条件を含む InstallScript インストールがサイレントで実行されたとき、今回より、その InstallShield 前提条件のインストールが必要な場合にインストールされて、前提条件のユーザー インターフェイスは表示されません。以前、InstallShield 前提条件がインストールされませんでした。

IOA-000052944 (基本の MSI、InstallScript MSI)

製品名にアンパサンドが含まれている場合、今回より、Setup.exe が表示する各実行時のダイアログのタイトル バーに、余分なアンパサンドは表示されません。これは、セットアップ前提条件ダイアログやセットアップ初期化ダイアログなどに適用します。

IOA-000052988

InstallShield ヘルプ ライブラリの「Setup.exe および Update.exe コマンドライン パラメーター」トピックには、/hide_usd コマンドライン パラメータについて正しい説明が記載されています。このコマンドライン オプションが使用されているときに、アップデート可能なインストールが複数の以前のインスタンスを検出すると、インストールは新しいインスタンスを作成することが説明されています。

IOA-000053015

MSBuild を使って InstallShield プロジェクトを含むソリューションをビルドしたときに、InstallShield.targets ファイルが見つからないために発生する MSBuild エラーが解決されました。以前、同じシステム上に InstallShield の複数のバージョンがインストールされている場合に、このエラーが発生することがありました。

IOA-000053060

InstallShield ヘルプ ライブラリの「アップデートを確認するショートカットを作成する」ヘルプ トピックが更新されました。以前、このヘルプ トピックには、「ターゲット」および「引数」設定に古い値がリストされていました。

IOA-000053125 (基本の MSI)

基本の MSI インストールをデバッグ モードで起動中に ISSCRIPTDEBUGPATH プロパティ値をコマンドラインを使って設定すると、今回より、InstallScript デバッガーが開いて、インストールで即時実行 InstallScript カスタム アクションが起動します。以前は、InstallScript デバッガーは起動されませんでした。InstallScript デバッガーには、昇格された権限が必要です。このため、コマンドプロンプト ウィンドウを昇格された権限を使って起動してから、そのコマンドプロンプト ウィンドウ内からインストールをデバッグ モードで起動することが推奨されます。

遅延、コミット、またはロールバック InstallScript カスタム アクションをデバッグするとき、コマンドラインを使って ISSCRIPTDEBUGPATH プロパティ値に指定されたパスは、これらのカスタム アクションにアクセスすることができません。このため、デバッグを行っているシステムのレジストリに Setup.dbg ファイルのパスを追加する必要があります。そのためには、次のレジストリ キーを作成してください:

```
HKCU\Software\InstallShield\ISEngineMajor.Minor
```

Major は、InstallShield のメジャー バージョン番号で、*Minor* は、InstallShield のマイナー バージョン番号です。

たとえば、InstallShield 2011 の場合、次のレジストリ キーが既存しない場合は、これを作成します:

```
HKCU\Software\InstallShield\ISEngine17.0
```

また、レジストリ キーに **DebugPath** という文字列値を追加して、その値データに Setup.dbg ファイルへのパスを設定します。

IOA-000053147

InstallShield 前提条件エディターを使って InstallShield 前提条件の依存関係を指定するとき、InstallShield 前提条件エディターで依存関係のファイル名が切り詰めて表示される問題が解決されました。以前、InstallShield 前提条件エディターでは、.prq 依存関係の名前が 62 文字を超える場合に、それが切り詰めて表示されました。この問題は、[開く] ダイアログ ボックスを使って .prq 依存関係を参照および選択した場合に発生しました。インストール プロジェクトに依存関係を含む前提条件を追加した場合、切り詰められたファイル名がビルド エラーの原因となりました。

IOA-000053160 (基本の MSI、InstallScript MSI)

Windows Mobile ウィザードの [デスクトップの設定] パネルで [デスクトップ セットアップのアンインストール中に、デスクトップからデバイス メディアを削除する] チェック ボックスをクリアすると、デスクトップ製品がアンインストールされるときに、Windows Mobile インストールがデスクトップからアンインストールされません。以前、Windows Mobile インストールは、[デスクトップの設定] パネルのチェック ボックスでの選択にかかわらずアンインストールされました。

IOA-000053172 (基本の MSI、InstallScript MSI)

今回より、ISWiProject オブジェクトの BuildPCPFile メソッドを使って、パッチ作成プロパティ ファイル (.pcp) の設定に基づいたパッチ パッケージ (.msp) をビルドできます。以前は、パッチのビルドがエラー -1072803327 (ERROR_PCW_INTERNAL_ERROR) を生成して失敗しました。

IOA-000053189、IOA-000053383 (基本の MSI、InstallScript MSI)

デフォルトの Setup.exe 文字列は、Setup.exe ファイルが多言語リリース用にビルドされている場合、再び .msi パッケージに格納されます。これによって、言語関連の様々なランタイム問題が解決します。たとえば、Setup.exe ファイルを含む InstallScript MSI リリースをビルドする場合に、デフォルト言語が英語以外で、エンド ユーザーが .msi パッケージをトランスフォームなしで直接起動した場合、今回より、インストールを Setup.exe から実行する必要があることを通知するエラー メッセージが表示されます。InstallShield 2010 でビルドされたインストールでは、エラー メッセージの代わりに ##StringID## などの文字列が表示されました。その他の例として、多言語を含む基本の MSI プロジェクトのマイナー アップグレードがあります。マイナー アップグレードで、新しいショートカットが ##ID_STRING_1## などの名前でインストールされることがなくなりました。

IOA-000053205

プロジェクトと関連ファイルがネットワーク上に格納されていて、InstallShield が英語以外のシステム上にインストールされている場合に、[InstallScript] ビューの中央ペインに表示される [関数] リストから InstallScript 関数が消える問題が解決されました。

IOA-000053272 (基本の MSI、InstallScript MSI、Merge Module)

サブフォルダを含むダイナミック リンク フォルダを含んでいて、2 つのサブフォルダの名前にシステム コード ページでサポートされていない同じ数の複数バイト文字が含まれているとき、今回より、それらのサブフォルダとそこに含まれるすべてのファイルは、実行時に予定通りターゲット システムにインストールされます。以前、InstallShield が 2 つのサブフォルダに一意のコンポーネントを適切に作成しなかったために、実行時にサブフォルダが 1 つだけしか作成されず、これら両方のサブフォルダに含まれるすべてのファイルがそのサブフォルダにインストールされました。

IOA-000053336 (Standalone Build)

InstallShield.Interop.Msi.dll および InstallShield.Interop.Stg.dll ファイルは、今回より、Standalone Build インストールの実行時にインストールされます。以前、これらのファイルがインストールされなかったため、このアセンブリを参照するリリースをビルドすると、ビルド エラーが発生しました。

IOA-000053374 (基本の MSI、InstallScript MSI)

[レジストリ] ビューで、DWORD 値の値データを [MYPROPERTY] などのプロパティに設定してから、値をダブルクリックして値データを編集すると、今回より、プロパティが保持されます。以前、値をダブルクリックすると、値データが数値 0 に変更されました。

IOA-000053402 (基本の MSI、InstallScript MSI)

実行時に SetupError ダイアログが表示される時、このダイアログのエラー メッセージには、今回より、終止符の後、エラー文字列の前に空白スペースが入ります。たとえば、SetupError ダイアログでは、「エラー 1327。ドライブが正しくありません [1]」の代わりに、「エラー 1327。ドライブが正しくありません [1]」と文字列が表示されます。

IOA-000053573

InstallShield ヘルプ ライブラリの「ConfigFileLoad」ヘルプ トピックに、正確な情報が含まれています。InstallScript 関数 ConfigFileLoad を使って新しい構成ファイルを作成することができないことが説明されています。以前、このトピックでは、この関数を使って新しい構成ファイルを作成することができるという、誤った情報が掲載されていました。このヘルプ トピックにはまた、新しい構成ファイルを作成するために、CreateFile および CloseFile を使って空のファイルを作成してから、ConfigFileLoad その他の関数を使って必要に応じてファイルをロードおよび変更することができるという説明が追加されています。

IOA-000053590

InstallShield ヘルプ ライブラリ「AddFolderIcon」ヘルプ トピックでは、szWorkingDir パラメーターの説明が改訂されています。AddFolderIcon は、ショートカットの [プロパティ] ダイアログ ボックスの [ショートカット] タブにある [作業フォルダー] ボックスに szWorkingDir ディレクトリを書き込むことが説明されています。ヌル文字列 ("") をこのパラメーターに渡すと、関数はこの [作業フォルダー] ボックスを空白のままにして、[リンク先] ボックスのパスが使用されます。このヘルプ トピックには、以前、このパラメータについて誤解を生む情報が含まれていました。

IOA-000053731 (InstallScript MSI)

新しい InstallScript MSI プロジェクトの [アップグレード] ビューには、ISPreventDowngrade というメジャー アップグレード項目が含まれています。この項目は、エンド ユーザーが製品の現在のバージョンで同じ製品の将来のメジャー バージョンを上書きできないように防ぎます。

IOA-000053800 (トランスフォーム)

[カスタム アクションおよびシーケンス] ビューの [スクリプト] タブを使って、トランス フォーム プロジェクトの Binary テーブルに格納されている VBScript または JScript カスタム アクションのコードを編集すると、次回プロジェクトを開いて [スクリプト] タブでスクリプトを参照したときに、コードが正しく表示されます。以前は、コードが正しく表示されませんでした。

IOA-000053840 (仮想化)

InstallShield がショートカットのアイコン リソースを取得できなかった場合、App-V パッケージをビルドしたときに無効な .ico リファレンスを作成する問題が解決されました。以前、App-V マニフェストと .osd ファイルに .ico ファイルへの無効なリファレンスが含まれました。その結果、Microsoft System Center Configuration Manager コンソールを使ってパッケージがパブリッシュされたときに、"ファイルが見つかりません" エラーが発生し、SMSAdminUI.log ファイルにログが記録されました。

IOA-000053950、IOB-000059050 (基本の MSI、InstallScript MSI)

[IIS 構成] ビューで Web サイト、アプリケーション、または仮想ディレクトリの "コンテンツのソース パス (ローカルまたは UNC)" 設定に物理パスを手動で入力すると、今回より InstallShield は、InstallShield 2009 と同様に新しい対応 Directory テーブル エントリを作成します。InstallShield 2010 では、Directory テーブル エントリが作成されなかったために、ランタイムエラーが発生しました。

IOA-000053978 (スマート デバイス)

Microsoft Smart Device CAB プロジェクトおよび InstallShield スマート デバイス プロジェクトを含む Visual Studio ソリューション内から InstallShield を使用しているときに、Visual Studio がクラッシュする問題が解決されました。

IOA-000054021 (基本の MSI、InstallScript MSI、マージ モジュール)

今回より、CustomActionData プロパティがマネージコード カスタム アクションで使用可能なインプット プロパティの 1 つとして含まれます。このため、このプロパティを手動で入力せずに選択することができるようになりました。

IOA-000054040 (基本の MSI、InstallScript、InstallScript MSI、マージ モジュール)

InstallScript 関数 SetObjectPermissions への呼び出しに nPermissions パラメーターで以下の 1 つ以上の定数を渡したときに発生するコンパイラー エラー C8025 の問題が解決されました: STANDARD_RIGHTS_ALL、STANDARD_RIGHTS_READ、STANDARD_RIGHTS_WRITE、または STANDARD_RIGHTS_EXECUTE。以前、これらの定数が ISRTWindows.h で定義されていないためにコンパイラー エラーが発生しました。

IOA-000054270

"MSI パッケージ ファイル名" 設定の説明が訂正されました。この設定に入力された値が .cab ファイルの名前に使用されるという説明が削除されています。

IOA-000054310 (基本の MSI、InstallScript、InstallScript MSI)

[IIS 構成] ビューを使って IIS 7 アプリケーションのハンドラー マッピングを構成するプロジェクトを作成したとき、実行時にターゲット システム上で新しいハンドラー マッピングだけが構成されます。以前、すべての新しいハンドラー マッピングが構成されましたが、既存のハンドラー マッピングも構成されました。

IOA-000054312 (InstallScript)

[SQL スクリプト] ビューで SQL サポートを使用する InstallScript プロジェクトからインストールを作成すると、今回より、インストールは空白名を持つサーバーをターゲットとする SQL サーバー接続をスキップします。この動作は、InstallShield 2009 以前のプロジェクトでの動作と同じです。InstallShield 2010 では、インストールがそのようなサーバーへの接続を試みて、関連 SQL スクリプトを実行しました。

IOA-000054375 (基本の MSI、InstallScript、InstallScript MSI、マージ モジュール、トランスフォーム)

ショートカットをプロジェクトに追加した場合で、そのショートカットが多くのアイコンを含む実行可能ファイルにあるアイコンを使用したときに、[ショートカット] ビューで起こる表示の問題が解決されました。

IOA-000054479 (基本の MSI、InstallScript、InstallScript MSI、マージ モジュール)

一定の条件下でビルド エラー -6213 が発生する問題が解決されました。以前、コンポーネントの "ビルド時の .NET スキャン" 設定にデフォルト値の [依存関係およびプロパティ] が選択されているとき、コンポーネントの .NET アセンブリに多くの依存関係が含まれている場合に、このビルド エラーが発生することがありました。

IOA-000054485 (基本の MSI、InstallScript、InstallScript MSI)

インストールを InstallShield プロジェクト内部から起動したときに、[InstallScript] ビューで最後に行われた変更が最新のビルドに含まれていない場合、InstallShield は今回より、インストールを実行する前にスクリプトをコンパイルします。以前、この状況でインストールを起動しようとしても、何も起こりませんでした。

IOA-000054509 (基本の MSI、InstallScript MSI、マージ モジュール、トランスフォーム)

名前に 1 つ以上の空白スペースが含まれているサービスをコンポーネントに追加できるようになりました。以前は、サービス名に空白スペースを使用しようとすると、InstallShield がエラーを表示しました。

IOA-000054648 (InstallScript、InstallScript MSI)

一定の条件下で、ダイアログ コントロールのランタイム文字列が切り詰めて表示される問題が解決されました。以前一部の状況下で、ダイアログ コントロールの幅を変更するなどして、ダイアログを編集した場合に、そのダイアログ コントロールの文字列が誤って切り詰められることがありました。

IOA-000054693 (InstallScript)

InstallShield と共にインストールされる Settings.xml ファイルで構成されるタイムスタンプ サーバーをオーバーライドしたとき、今回より、InstallShield はビルド時に InstallScript リリースにデジタル署名するために入力された値を使用します。デフォルト XML には以下が含まれます:

```
<DigitalSignature Timestamp="http://timestamp.verisign.com/scripts/timestamp.dll"/>
```

異なるタイムスタンプ サーバーを使用するには、必要に応じて Timestamp 属性の値を変更します。タイムスタンプを無効にするには、Timestamp 属性に空白値 ("") を使用します。この新しい設定は、マシン全体に反映されますので、ご注意ください。

以前は、タイムスタンプのオーバーライドは基本の MSI および InstallScript MSI プロジェクトで利用できましたが、InstallScript プロジェクトでは利用できませんでした。

IOA-000054785 (基本の MSI、InstallScript、InstallScript MSI)

今回より、新しい言語ウィザードに高い言語 ID 番号を持つ言語のサポートが追加されました。以前、新しい言語ウィザードを使って特定の言語のサポートをプロジェクトに追加しようとすると、ウィザードでプロジェクトに言語を追加できないことを通知するエラーが表示されました。

また、このウィザードを使って新しいプロジェクトでこれらの言語のサポートを有効にして、新しいプロジェクトにその言語を追加すると、InstallShield にそれを追加することができます。以前、この状況下では、InstallShield がクラッシュしました。

IOA-000054845 (基本の MSI)

[ダイアログ] ビューに表示されるダイアログで編集フィールド コントロールを選択すると、このコントロールのその他の設定と共に Property Is Integer 設定が右側のペインに表示されます。以前は、この設定が編集フィールド コントロールにリストされませんでした。

IOA-000054872 (基本の MSI、InstallScript、InstallScript MSI、InstallScript オブジェクト、マージ モジュール)

[リリース] ビューで選択されたリリースの [署名] タブでデジタル証明書のパスワードを入力したとき、今回より、入力したパスワードがこの設定で保持されます。以前は、1 つのリリースにパスワードを入力してから別のリリースを選択して、最初のリリースに戻ると、InstallShield プロジェクト (.ism) を保存していない限り、パスワード設定が空白になりました。

IOA-000055016 (基本の MSI)

今回より、複数インスタンス インストール用にコマンドラインで Windows Installer プロパティ TRANSFORMS を設定するとき (たとえば、Setup.exe /instance=1 /v"TRANSFORMS=myTransform.mst")、Setup.exe で /instance コマンドラインパラメーターを使用できます。以前は、/instance パラメーターのみを使用できました。TRANSFORMS プロパティは無視されました。

IOA-000055352 (基本の MSI、InstallScript MSI)

リリース フラグを使ってビルド時にリリースから機能を除外する場合、その機能にダイナミック リンク ファイルを含むコンポーネントが含まれている場合、InstallShield がビルド時にそのコンポーネントのダイナミック ファイル リンクをスキャンすることがなくなりました。以前、ビルド時に InstallShield がダイナミック ファイル リンクのスキャンを行いました。このとき、ダイナミック リンク フォルダーが存在しない場合にビルド エラーが発生しました。

IOA-000055353

InstallShield ヘルプ ライブラリの「Standalone Build をビルド マシンにインストールする」ヘルプ トピックには、今回より、ISAppServices.tlb および ISUpgrade.tlb ファイルへの正しいパスがリストされています。両方とも Program Files フォルダー \Common Files\InstallShield\Shared にインストールされます。以前は、誤ったパスが記載されていました。

IOA-000055379 (基本 MSI、InstallScript、InstallScript MSI)

今回より、[IIS 構成] ビューで選択された Web サイト、アプリケーション、または仮想ディレクトリで、.NET Framework 4 が "ASP.NET バージョン" 設定および "ASP.NET プラットフォーム" 設定にリストされています。

IOA-000055894

[リリース] ビューのサブノードを選択してからリリースをビルドすると、今回より、フォーカスが選択したサブノードに残ります。以前は、一部のプロジェクトの種類で、フォーカスがルート ノードに移動しました。その他のプロジェクトの種類では、右側のペインがインライン ヘルプ切り替わり、フォーカスが選択したサブノードに維持されませんでした。

IOA-000055986 (基本の MSI)

InstallShield Express プロジェクトを Premier または Professional エディションにアップグレードすると、今回より、[ファイルとフォルダー] ビューの [インストール先コンピューターのフォルダー] ペインで表示/非表示に切り替えることができる定義済みフォルダーのリストに 64 ビット フォルダーの場所が追加されます。定義済みフォルダーのリストは、[インストール先コンピューターのフォルダー] ペインでノードを右クリックしてから [定義済みフォルダーの表示] をクリックすると表示されます。

IOA-000056024 (基本の MSI)

[ビルボード] ビューを使って、インストールにビルボードを含めた場合、進行状況バーは、進行状況ダイアログに表示されるとすぐに動き始めます。以前、進行状況バーは、インストールがターゲット システムにファイルをコピーし始めるまで動きませんでした。

IOA-000056027 (基本の MSI、InstallScript MSI)

プロジェクトの Directory テーブルに新しいエントリを追加して、DefaultDir フィールドに値を入力したとき、今回より、InstallShield が新しい文字列 ID を作成して、それを DefaultDir フィールドと関連付けることがなくなりました。この動作は、InstallShield 2009 以前での動作と同じです。

IOA-000056034 (基本の MSI、InstallScript、InstallScript MSI、InstallScript オブジェクト、マージ モジュール、トランスフォーム)

以下の領域で、再び CTRL+C および CTRL+V を使って、1 つのコンポーネントから別のコンポーネントにファイルをコピーおよび貼り付けできるようになりました。

- 基本の MSI、InstallScript MSI、マージ モジュール、トランスフォーム プロジェクトの場合: [コンポーネント] および [セットアップのデザイン] ビュー内の [ファイル] サブビュー
- InstallScript および InstallScript オブジェクト プロジェクトの場合: [コンポーネント] および [セットアップのデザイン] ビュー内の [スタティック ファイル リンク] サブビュー

InstallShield 2010 のこれらの領域で CTRL+C および CTRL+V を使っても、効果が得られませんでした。

IOA-000056062 (基本の MSI、InstallScript MSI、マージ モジュール)

マネージコード カスタム アクションのクローンを作成すると、今回より、元のカスタム アクションに構成された "スクリプト内実行" 設定がクローンされたカスタム アクションにも使用されます。以前は、クローンされたカスタム アクションの "スクリプト内実行" 設定には、元のカスタム アクションの値にかかわらず常に [即時実行] が選択されました。

IOA-000056205 (基本の MSI、InstallScript MSI)

リリース ウィザードの [InstallShield 前提条件] パネルに、[個々の選択に従う] オプションが追加されました。つまり、[リリース] ビューでリリースを選択したときに [Setup.exe] タブに表示される "InstallShield 前提条件の場所" 設定から使用できるオプションと同じオプションを、このパネルでも選択することができます。

IOA-000056256 (基本の MSI、InstallScript、InstallScript MSI、InstallScript オブジェクト、マージ モジュール)

プロジェクト ファイル (.ism) に XML ファイル形式を使用する場合で、[ダイレクト エディタ] ビューを使って、Binary テーブルに新しいエントリを手動で追加したとき、InstallShield では今回より、プロジェクト ファイルを再び開くことができます。以前、このビューを使って Binary テーブル エントリを追加してからプロジェクトを保存し、閉じた後に、そのプロジェクトを再び開こうとすると、InstallShield で .ism ファイルを開くことができないことを通知するメッセージが表示されました。

IOA-000056376 (InstallScript デバッガー)

今回より、デバッグ マシンに元の .dbg ファイルを作成したドライブ名が存在しないときに、InstallScript デバッガーを使って InstallScript をデバッグすることができます。

IOA-000056406

既存のリリースを再ビルドするとき、InstallShield が以前のビルド中に作成されたビルド ログ ファイルとレポート ファイルの更新日を変更することがなくなりました。

IOA-000056424

InstallShield ヘルプ ライブラリの「GetAndAddAllFilesCost」ヘルプ トピックには、今回より、InstallScript 関数 GetAndAddAllFilesCost の正しい構文が記述されています:

```
GetAndAddAllFilesCost (szSrcDir, szWildcard, szTargetDir, nClusterSize, nvInstallCostHigh, nvInstallCostLow, nvUninstallCost);
```

以前、リストされているパラメーターの一部の順番に誤りがありました。

IOA-000056470 (InstallScript, InstallScript MSI)

日本語のようこそダイアログ内のテキストは、このダイアログの英語版の現在のデフォルト テキストを反映して更新されました。

IOA-000056534 (InstallScript MSI)

InstallScript エンジン埋め込み UI ハンドラーとして使用する InstallScript MSI インストールを作成したとき (つまり、[一般情報] ビューの "InstallScript ユーザー インターフェイスの種類" 設定で [新しいスタイル (Windows Installer 4.5 が必要)] オプションを設定)、今回より、製品を Windows Vista 以降のシステムで [プログラムの追加と削除] を使って削除することができます。以前、これらのシステムから製品を削除する唯一の方法は、Setup.exe を再度実行して [削除] オプションを選択する方法のみでした。[プログラムの追加と削除] を使って製品を削除しようとしても、製品はアンインストールされませんでした。

IOA-000056536 (InstallScript)

コンポーネントの "インストール先" 設定における以下の例のように、再びスクリプト定義の場所を入力することができるようになりました。

```
<FOLDER_COMMON_APP>\<IFX_COMPANY_NAME>\<IFX_PRODUCT_NAME>
```

以前は、この設定に前述のようなパスを入力すると、InstallShield でエントリに無効な文字が含まれていることを通知するエラーが表示されました。

IOA-000056568 (基本の MSI、InstallScript、InstallScript MSI)

基本の MSI および InstallScript MSI プロジェクトで、[IIS 構成] ビューでアプリケーション プールの "制限間隔 (分)" 設定で値を指定したときに、IIS 7 以降が搭載されたシステム上でランタイム エラーが発生する問題が解決されました。InstallScript プロジェクトのこの設定で値を指定すると、実行時にアプリケーション プールがインストールされます。以前は、アプリケーション プールがインストールされず、エラーも表示されませんでした。

IOA-000056624

InstallShield ヘルプ ライブラリの「TCP ポート番号とサイト番号の構成」ヘルプ トピック内のステップ 1c が改訂されました。[プロパティ マネージャー] ビューで括弧は使用されていないため、プロパティ名は角括弧で囲まれていません。

IOA-000056730 (InstallScript)

マシンの C:\Reports に 1 つ以上のファイルが既に存在する場合に InstallScript インストールをビルドすると発生するビルド エラー -5046 が解決されました。

IOA-000056769

InstallShield ヘルプ ライブラリの「Oracle のインスタンスに接続して SQL スクリプトを実行する」ヘルプ トピックが改訂され、InstallShield 前提条件を Oracle 11g Instant Client のバージョン 11.1.0.7 用に準備する方法について追加されています。このヘルプ トピックでは、Oracle の 11.1.0.7 より後のバージョンを使用する場合についても説明されています。その場合、InstallShield 前提条件エディターで Oracle 11g Instant Client 11.1.0.7 前提条件を開いてから、その前提条件の名前を適切なバージョン番号を反映するように変更する必要があります。さらに、前提条件の条件を更新して、適切なバージョン番号を反映させることも必要です。

IOB-000050440

MSBuild を使って InstallShield プロジェクトを含む Visual Studio ソリューションをビルドするときに、InstallShield プロジェクトにプライマリ出力が含まれている場合、一部の状況において発生するビルド警告 -6248 が解決されました。以前、この警告が一部の状況において、誤って発生することがありました。

IOB-000053545 (基本 MSI、InstallScript MSI)

InstallShield プロジェクトを含む Visual Studio ソリューションにおいて、InstallShield は今回より、出力グループに複数のフォルダーが含まれるときに、プライマリ出力の依存関係をリストします。以前 InstallShield では、出力グループに複数のフォルダーが含まれるときに、依存関係の表示をサポートしませんでした。

IOB-000053816

InstallShield ヘルプ ライブラリの「DialogSetInfo」ヘルプ トピックには、今回より、InstallScript 関数 DialogSetInfo のサポートを含むプロジェクトの種類がリストされています。

- InstallScript
- InstallScript MSI (イベント ドリブン型の InstallScript、InstallScript カスタム アクションを除く)

以前は、このトピックにプロジェクト固有の情報はリストされていませんでした。

IOB-000056986

InstallShield ヘルプ ライブラリの「ダイアログ スキンの指定」および「ダイアログ スキン」ヘルプ トピックには、InstallScript または InstallScript MSI プロジェクトにおいて [ダイアログ エディタ] でダイアログを編集したあとに標準ダイアログにダイアログ スキンを適用、または標準ダイアログからスキンを削除することができないことが説明されています。このため、編集する予定のダイアログにスキンを指定するときは、まずスキンを指定してから、編集してください。

IOB-000058418 (InstallScript)

InstallScript リリースのアップグレード中、またはメンテナンス モード中に、InstallShield 前提条件を起動する必要がある場合、これが起動されるようになりました。以前は、初回インストール中に前提条件がターゲット システムにインストールされる必要がある場合のみ、InstallShield 前提条件が起動しました。

IOB-000058716

Windows 7 システム上で、InstallShield の オフライン アクティベーション プロセスが、正常に動作します。以前一部の状況において、アクティベーションが完了する前に (要求コードが保存されて、応答コードが入力される前)、それが成功したことを通知するアクティベーション ダイアログが表示され、続けて InstallShield が起動されたときに、アクティベーション ダイアログが再び表示されました。

IOB-000058755 (InstallScript)

1 つ以上のマージ モジュールを含む InstallScript プロジェクトで [IIS 構成] ビューを使って IIS 仮想ディレクトリを構成したとき、実行時に仮想ディレクトリとマージ モジュールがインストールされます。以前、実行時にマージ モジュールはインストールされましたが、仮想ディレクトリはインストールされませんでした。

IOB-000058888

[ダイレクト エディタ] ビューの検索置換機能で、[単語単位で探す] チェック ボックス を選択すると、今回より、InstallShield は 単語単位で一致した項目のみ置換します。以前一部の状況下で、部分的に一致する単語が置換されました。

IOB-000059291 (基本の MSI、InstallScript MSI)

[IIS 構成] ビューで構成された IIS Web サイトについて、実行時に Windows Installer プロパティを使って TCP ポート番号を設定すると、IIS Web サイトのコンポーネントがアンインストールされるときに、Web サイトのコンポーネントがパーマネントでない限り、Web サイトがアンインストールされます。以前この状況下では、Web サイトがアンインストールされませんでした。

IOB-000059388 (基本の MSI、InstallScript、InstallScript MSI)

[IIS 構成] ビューで仮想ディレクトリの設定の一部を構成して、"ディレクトリの参照" 設定をデフォルト値の [いいえ] のままにした場合、今回より、その結果として出来たインストールがターゲット システム上の仮想ディレクトリの参照を有効にすることがなくなりました。

IOB-000059398 (InstallScript、InstallScript MSI)

InstallScript 関数 FeatureFileInfo は、今回より、ターゲット システム上で指定のファイルが見つからない場合にエラーを返します。以前、関数が成功しなかった場合でも、常に成功を示す 0 を返しました。

IOC-000040911 (基本の MSI、InstallScript MSI、スマート デバイス)

Windows Mobile ウィザードまたはスマート デバイス セットアップ ウィザードの [アプリケーション情報] パネルで "アプリケーション名" 設定または "会社名" 設定にコンマを入力してから [次へ] をクリックすると、ウィザードではコンマを使用できないことを説明するエラー メッセージが表示されます。以前、これらの設定でコンマが無効な文字であるにもかかわらず、このような検証は行われませんでした。そのため、これらの設定にコンマを使用した場合にはランタイム エラーが発生しました。ランタイム エラーは、セットアップ ファイルが無効であるためにアプリケーションをインストールできなかったことを通知しました。

IOC-000045115

InstallShield ヘルプ ライブラリの「OnComponentError」ヘルプ トピックには、これまでに記述されていなかったパラメーター ErrorInfo.FileError.Description、ErrorInfo.FileError.File、および ErrorInfo.FileGroup についての情報が掲載されていません。

IOC-000050558

InstallShield ヘルプ ライブラリの「SdOptionsButtons」ヘルプ トピックには、SdOptionsButtons ダイアログ上のプッシュ ボタン コントロールのコントロール ID を変更してはならないこと、およびこれらのプッシュ ボタン コントロールをデフォルト コントロールとして設定することができないという重要な情報が追加されています。このヘルプ トピックでは、コントロール ID を変更する、またはこれらのコントロールの 1 つをダイアログのデフォルト コントロールとして設定すると、SdOptionsButtons ダイアログが予定通りに動作しなくなることも説明されています。

IOC-000055567 (基本の MSI、InstallScript MSI)

プロジェクトで InstallValidate および InstallInitialize アクションを連続してシーケンスして、その間に使用されていないシーケンス番号が存在しない場合に、[アップグレード] ビューで [セットアップをインストールし、不要なファイルを削除する] オプションから [新規セットアップをインストールする前に、以前のセットアップを完全にアンインストールする] オプションに切り替えたとき、今回より、InstallShield によって RemoveExistingProducts アクションが正しくシーケンスされます。以前この状況下で、InstallShield は RemoveExistingProducts アクションをシーケンスし直すことが出来ませんでした。

IOC-000056288 (基本の MSI)

プロジェクトの InstallUISequence テーブルに含まれているダイアログの名前を変更すると、InstallShield は今回より、InstallUISequence テーブル内のそのダイアログへのリファレンスを適切に変更します。

IOC-000056960

InstallShield ヘルプ ライブラリの「DoInstall の例」ヘルプ トピックにおいて、サンプル コードが訂正されました。以前、古い戻り値が記載されていました。

IOC-000060340 (InstallScript)

インストールで共有コンポーネントからのファイルをターゲット システムにインストールする場合、そのファイルの元のセキュリティ設定がターゲット システム上で変更される問題が解決しました。以前、ファイルがロックされている場合、ターゲット システム上でそのファイルのセキュリティ設定が変更されました。

IOC-000061176

InstallShield ヘルプ ライブラリの「応答ファイルのサイレント ヘッダー」ヘルプ トピックが訂正されました。このヘルプ トピックでは、応答ファイルを作成するとき、サイレント インストールの実行に使用される応答ファイルを作成する InstallShield と同じバージョンを使用することが説明されています。

IOC-000066525

InstallShield ヘルプ ライブラリの「アンインストール用にログされた InstallScript 関数」ヘルプ トピックが訂正されました。

IOC-000071368 (基本の MSI、InstallScript MSI、QuickPatch)

[パッチのデザイン] ビューにあるパッチ構成の "ターゲット製品名" 設定では、アポストロフィを使用できるようになりました。これは、基本の MSI および InstallScript MSI プロジェクトに適用します。

[一般情報] ビューの [ビルドの設定] 領域の [識別] タグにある "ターゲット製品名" 設定でも、アポストロフィを使用できます。これは、QuickPatch プロジェクトに適用します。

以前、Microsoft ツール Patchwiz.dll の以前バージョンではアポストロフィの使用がサポートされていなかったため、これを使用することはできませんでした。

IOC-000074259 (InstallScript)

プロジェクトに Visual Basic Runtime オブジェクトが含まれていて、インストールが Visual Basic Runtime を日本語システムにインストールする場合に、ステータス テキストが文字化けする問題が解決されました。

IOC-000075959 (基本の MSI、InstallScript MSI、マージ モジュール、トランスフォーム)

CTRL+C と CTRL+V を使って、あるコンポーネントから別のコンポーネントにショートカットをコピー アンド ペーストしたとき、プロジェクトを保存したときに変更が保持されます。以前、ショートカットが、コンポーネントから別のコンポーネントに移動しませんでした。

IOC-000076034 (QuickPatch)

QuickPatch プロジェクトで [一般情報] ビューの [カスタム アクション] 領域にあるカスタム アクションのチェックボックスをクリアした場合に、ベース インストール プロジェクトのカスタム アクションに OR 演算子を使った条件が含まれているとき、ターゲットシステムで QuickPatch パッケージが実行されるときにカスタム アクションが実行する問題が解決されました。以前、一部の状況下でカスタム アクションが実行しました。

IOC-000077506

InstallShield ヘルプ ライブラリの「InstallShield Professional 6.x からの移行」ヘルプ トピックでは、InstallShield Professional 6 プロジェクトを InstallShield 2011 にアップグレードするときに、「インストールされているアプリケーションのバージョンを判別できません。セットアップを終了します。」のようなメッセージが実行時に表示された場合の対処方法が説明されています。

IOC-000078276 (基本の MSI)

ラジオ ボタン コントロールを含むダイアログのクローンを作成してから、そのダイアログでラジオ ボタン グループ コントロールに関連付けられているプロパティを変更した場合、InstallShield が元のダイアログ上のラジオ ボタン コントロールを削除する問題が解決されました。

IOC-000079459 (基本の MSI、InstallScript MSI)

アップグレードの作成時にパッチの最適化機能を使用すると、今回より、COM 抽出を使用するコンポーネント内のレジストリ キーが InstallShield によって 同期されます。以前は、これらのレジストリ データが同期されませんでした。

IOC-000079640

InstallShield ヘルプ ライブラリに、新しいヘルプ トピック「セットアップ起動ツールのファイルのプロパティをカスタマイズする」が追加されました。このヘルプ トピックでは、InstallShield のどの設定を使って、Setup.exe セットアップ起動ツールのバージョン リソースを構成するかが説明されています。"カスタム バージョンのプロパティを使用する" 設定と、"起動ツールの著作権" 設定などの様々な設定をクリックしたときに、[リリース] ビューに表示されるインライン ヘルプには、この新しいヘルプ ライブラリ トピックへのハイパーリンクが含まれています。

IOC-000079814 (基本の MSI、InstallScript MSI)

インストールに Oracle Instant Client 前提条件が含まれていて、それが実行時に起動される場合で、インストールが TNS サービス名を使って Oracle データベース サーバーに接続を試みるとき、インストールは今回より、Oracle Instant Client 前提条件がインストールされた後に再起動を行わずにサーバーへ接続することが出来ます。

IOC-000080799 (基本の MSI、InstallScript、InstallScript MSI)

InstallShield プロジェクトの ISSQLDbMetaData 内の QueryDatabasesCmd フィールドを編集して、ターゲットシステム上の複数のレコード セットを返すカスタム SQL クエリを作成した場合、エンド ユーザーが SQLLogin ダイアログからデータベースを参照しようとしたときに、すべての適切なデータベースがリストされます。以前は、最初のレコード セットの結果のみがリストされました。

IOC-000080835 (基本の MSI、InstallScript、InstallScript MSI)

ログオン情報機能を使ったインストールを作成して（「既存のユーザー アカウントを作成または設定する機能を追加する」ヘルプトピックを参照）、エンド ユーザーが日本語システム上でインストールを実行した場合に、特定のパスワードを入力したときに発生する問題が解決されました。以前一部の状況下で、エンド ユーザーが特定のパスワードを入力すると、実行時にユーザー アカウントが作成されませんでした。

IOC-000080886 (基本の MSI、InstallScript MSI)

ロールバック中の実行がスケジュールされている SQL スクリプトに条件ステートメントを指定した場合、条件が True 評価された場合のみ、ロールバック中に SQL スクリプトが実行します。以前は、条件が True 評価されたかどうかにかかわらずロールバック中に SQL スクリプトが実行しました。

IOC-000080964

[再配布可能ファイル] ビューおよび [オブジェクト] ビューで、完全な Visual C++ 9.0 マージ モジュールが Visual C++ 9.0 ポリシー マージ モジュールとして表示されることがなくなりました。また、ポリシー マージ モジュールが、完全なマージ モジュールとして表示されることもなくなりました。

IOC-000081031

InstallShield プロジェクト ファイル (.ism) を保存したとき、[ファイルをアーカイブ可能にする] チェック ボックスがデフォルトで選択されています。このチェック ボックスは、.ism ファイルの [プロパティ] ダイアログ ボックスに表示される詳細属性です。

IOC-000081136 (基本の MSI、InstallScript MSI、マージ モジュール、トランスフォーム)

REG ファイルのインポート ウィザードを使って角かっこ ([]) または波かっこ ({ }) が含まれている .reg ファイルをインポートするとき、InstallShield は今回より、対応する Registry テーブル エントリにこれらの特殊文字を含めます。インポートする .reg ファイルには、特殊文字にエスケープ文字を含むことができません。以前、InstallShield は特殊文字を含むレジストリ データをインポートしませんでした。

IOC-000081641 (InstallScript)

InstallScript プロジェクトの [一般情報] ビューで、"アイコン インデックスの表示" 設定に値を指定すると、今回より、インストールは [プログラムの追加と削除] の製品エントリに適切なアイコン リソースを使用します。以前は、インストールが常に指定されたアイコン ファイルの最初のアイコン リソースを使用しました。

IOC-000081714 (基本の MSI)

InstallShield Express 2 で作成されたプロジェクトを InstallShield 2011 にアップグレードしてから、[テキスト ファイルの変更] ビューを開くと、Windows Installer エラー 2228 が発生する問題が解決されました。

IOC-000081718 (InstallScript、InstallScript MSI)

InstallScript ダイアログを編集してから、使用中のコンピュータの DPI 値を変更して、その編集されたダイアログを含むインストールをビルドした場合、実行時にダイアログが正しいサイズで表示されます。以前、実行時に編集されたダイアログが、編集されていないダイアログと異なるサイズで表示されました。

IOC-000081856 (InstallScript、InstallScript MSI)

InstallScript イベント OnCustomizeUninstInfo および InstallScript SdCustomerInformation* 関数のデフォルト コードは、今回より、Windows 9x- 固有のコードを含みません。

IOC-000082083

"デフォルトの場所" 設定に Windows Installer プロパティを使用する Visual Studio セットアップ プロジェクトを、InstallShield を使って InstallShield 基本の MSI プロジェクトに変換できます。以前は、この種類の Visual Studio セットアップ プロジェクトを変換しようとする、致命的なエラー -9021 が発生しました。

IOC-000082134 (InstallScript)

InstallScript オブジェクトを含むプロジェクトのリリースをビルドしてから、[ビルド] メニューにある [リフレッシュ ビルド] コマンドを使用して再ビルドを行うと、エラー -9008 が発生する問題が解決されました。

IOC-000082208 (基本の MSI、InstallScript MSI、QuickPatch)

InstallShield では、今回より、ファイル数が 32,767 を超える標準のパッチ ([パッチのデザイン] ビューを使用する) または QuickPatch パッケージを作成できます。以前、この状況ではビルド エラー -6415 が発生しました。

IOC-000082424 (基本の MSI)

すべてのサポート対象言語の言語 .ini ファイルで、「アラビア語」と「ヘブライ語」の言語名が翻訳されました。これらの翻訳された言語名は、言語の選択ダイアログで使用されます。

IOC-000082530 (InstallScript)

マシンの C:\LogFiles に 1 つ以上のファイルが既に存在する場合に InstallScript インストールをビルドすると発生するビルド エラー -5045 が解決されました。

IOC-000082646

InstallShield ヘルプ ライブラリの「InstallShield カスタム アクション リファレンス」ヘルプ トピックに、ISSQLServerInitialize および ISSQLQueryDatabases カスタム アクションについての詳細が追加されました。

IOC-000084101 (基本の MSI)

エンド ユーザーが製品の新しいバージョンの上に古いバージョンをインストールしようとしたときに、実行時に表示されるメッセージ ボックスに含まれる IDS_PREVENT_DOWNGRADE_EXIT 文字列において、今回より、メッセージ ボックス上に表示される [OK] ボタンと同じ言語で文字列内の「OK」も翻訳されています。

IOC-000084807 (基本の MSI、InstallScript、InstallScript MSI)

プロジェクトで INSTALLSHIELD_IIS_NEXT_NEW_SITE_NUMBER を使って、次に使用可能な新しいサイト番号に IIS Web サイトをインストールする場合で、ターゲット システム上に既にサイト番号 9 と 10 が存在するとき、今回より、インストールは Web サイトをサイト番号 11 にインストールします。以前、IIS 7 以降が搭載されたシステム上では、インストールが失敗しました。

システム要件

このセクションでは、InstallShield で作成されたインストールを実行するターゲット システム (ランタイム環境) の要件、ならびに InstallShield を実行するために必要なシステム (オーサリング環境) の要件が説明されています。

InstallShield を実行するシステムの要件

プロセッサ

Pentium III クラスの PC (500 MHz 以上を推奨)

RAM

256 MB の RAM (512 MB 推奨)

ハードディスク

500 MB 空き領域

ディスプレイ

1024 x 768 (XGA) 以上の解像度

オペレーティング システム

Windows XP
Windows Server 2003
Windows Vista
Windows Server 2008
Windows 7
Windows Server 2008 R2

ブラウザ

Microsoft Internet Explorer 6

権限

システムの管理者権限

マウス

Microsoft IntelliMouse、またはその他の互換性があるポインティング デバイス

ターゲット システム (デスクトップ コンピューター) の要件

ターゲット システムは、次のオペレーティング システムの最低要件を満たさなくてはなりません:

Windows 2000
Windows XP
Windows Server 2003
Windows Vista
Windows Server 2008
Windows 7
Windows Server 2008 R2

ターゲット システム (モバイル デバイス) の要件

InstallShield に、Microsoft Windows Mobile Device Center または Microsoft ActiveSync を使用するデスクトップ インストールにモバイル デバイス インストール追加してファイルをモバイル デバイスに転送することができるサポートが追加されました。

InstallShield では、Windows Mobile Device Center や ActiveSync、または他のデスクトップ コンポーネントを使用しないデバイスを直接ターゲットするインストールもサポートされています。

InstallShield がサポートする各オプションの概要については、InstallShield ヘルプ ライブラリの「モバイル デバイスにインストールを作成する」を参照してください。

Windows Mobile デバイスの要件

InstallShield では、多数の Windows Mobile プラットフォームおよびプロセッサがサポートされています。以下は、Windows Mobile プラットフォームの一覧です。

- Windows Mobile 6.x Professional と Classic
- Windows Mobile 6.x Standard
- Windows Embedded CE 6.x
- Windows Mobile 5.0 for Pocket PC
- Windows Mobile 5.0 for Smartphone
- Windows CE .NET 5.0
- Windows CE .NET 4.x
- Pocket PC 2003
- Pocket PC 2002
- Pocket PC
- Palm-size PC 2.11
- Palm-size PC 2.01
- Handheld PC 2000
- Handheld PC Pro
- Handheld PC 2.0
- Smartphone 2003
- Smartphone 2002

InstallShield は、この一覧にないプラットフォームもサポートしますが、デフォルトで条件設定ができないという点で上記のプラットフォームと異なります。追加のプラットフォームのサポートを追加する場合、または、特定のプラットフォームをターゲットするときの条件を変更する場合、InstallShield と共にインストールされている Settings.xml を変更することができます。詳しい情報については、InstallShield ヘルプ ライブラリの「使用可能な Windows Mobile プラットフォームの一覧またはその詳細設定を更新する」を参照してください。

InstallShield では、次の Windows Mobile プロセッサがサポートされています。

- ARM920
- ARM820
- ARM720
- Common Executable Format (CEF)

- Hitachi SH4
- Hitachi SH3E
- Hitachi SH3
- i686
- i586
- i486
- MIPS R4000
- MIPS R3000
- MIPS R2000
- SHx SH4
- SHx SH3
- StrongARM-XScale

Palm OS デバイスの要件

InstallShield では、Palm OS 3.5 以降がサポートされています。

Windows Mobile デバイス インストールのデスクトップ要件

Windows Mobile デバイスにアプリケーションをインストールするために使用されるデスクトップ コンピューターの要件は以下のとおりです。

- Windows XP 上にある Microsoft ActiveSync 3.x 以降 (Windows Mobile 5.x 以降のデバイスには ActiveSync 4.x が必要です)
- Windows Vista 上の Microsoft Windows Mobile Device Center
- 管理者権限

Palm OS デバイス インストールのデスクトップ要件

Palm HotSync は、Palm OS デバイスにアプリケーションをインストールするために使用されるデスクトップ コンピューターに必要です。

既知の問題

既知の問題の一覧は、ナレッジベース記事「[Q208912](#)」を参照してください。