



Microsoft® Office 2013 マクロ互換性について

2013 年 3 月

目次

はじめに.....	4
開発環境の変更に関する留意点	7
アプリケーションの開発タブ表示方法.....	7
開発者用リファレンスの表示方法	10
開発環境について.....	11
Office ストア	13
VBE (Visual Basic Editor) の変更点と問題点.....	14
マクロ全般に関する留意点	15
VBA 言語互換性、VBA プロジェクトについて	15
ファイル形式に関連した留意点	16
既定でインストールされなくなった機能	19
マクロの記録について	23
オペレーティング システム (OS) によって起こる問題について.....	24
64 ビット版 Microsoft Office 2013 について	25
64 ビット版 Microsoft Office 2013 のメリット.....	25
64 ビット版 Microsoft Office 2013 の留意点	26
64 ビット版 Microsoft Office 2013 の展開	31
セキュリティ機能に関連する留意点	32
Office 2013 の新しいセキュリティ機能	32
セキュリティ センターについて.....	34
メッセージ バーの表示について.....	37
保護ビュー	40
マクロのセキュリティの既定レベルについて	41
VBA プロジェクトのデジタル署名.....	43
Microsoft Backstage ビュー	43
マクロが含まれたファイルが暗号化されている場合の留意点	44
ユーザー インターフェイスへの影響.....	45
既存のアドインのサポート.....	45
リボンのカスタマイズ	46
Office UI カスタマイズについてのリソース.....	50
マクロ コードの見直しが必要な項目	51
Microsoft Office 2013 共通の変更点	51



製品別の変更点	55
Office 展開に関連するツールについて	74
Office テレメトリ ダッシュボード	74
まとめ	76
その他のトピック	77
Office Web Components (OWC) の提供	77
Office ソリューション開発環境	78
Office 技術者向けの日本語情報	81
Microsoft Office 2013 の KB	83

はじめに

Office 2010 に続く新しい Office は、ロゴも含めて一新され、“Office”と呼ぶことになりました。この新しい Office は、お客様から数多く頂戴したフィードバックをもとに、利用者がより効率的に、また質の高い作業が行えるように、新しい機能の追加や、操作性の向上など、様々な側面で大きく改善がされています。また、Office をベースに構築されるアプリケーションのさらなる強化に向けて、アプリケーションのプラットフォームとしての機能強化も行っています。

新しい Office のプラットフォームとして強化された機能として、Web の標準的な技術を利用して容易に開発ができる Office 用アプリや、Office のアプリケーションをダウンロード購入できる Office ストアなどが挙げられます。これらの新しい機能により、Office によるアプリケーションの開発や運用に関わる選択肢が増えており、Office の利用範囲がますます広がっています。

さらに、新しい Office の展開を容易にする仕組みも強化されています。クイック実行を利用すると、1 台のコンピューターの中で旧バージョンの Office と新しい Office の共存が可能となります。また、Office ファイルやアドインの利用状況と互換性の問題を可視化する Office テレメトリ ダッシュボードが標準で搭載されています。これらの機能を組み合わせて利用することにより、既存の Office を利用している環境に新しい Office を展開する作業が容易になり、また展開に必要な期間を短縮する効果も期待できます。

本ドキュメントでは混乱を避けるため、新しい Office を“Office 2013”と称しています。また、主に Microsoft Office 2003 の、Microsoft Word、Microsoft Excel®、Microsoft PowerPoint®、Microsoft Access® で構築されたマクロ、VBA によるカスタム アプリケーションを Microsoft Office 2013 上で実行する際の留意点を 5 つのポイントに分けて説明しています。マクロ コードの見直しが必要な項目についても、サンプルコードを提示しながら説明を行っています。Office 2003 以前のバージョンや、その他のアプリケーションについても関連する部分について記載を行っています。

組織の中で、Microsoft Office を利用したアプリケーションを開発、運用されている方はもちろんのこと、Office をはじめとしたクライアント端末の環境を管理されている方、セキュリティ担当されている方が、新しいデスクトップアプリケーションとして Office 2013 の導入を行う際に参考にしていただけるドキュメントです。ファイルの表示に関する互換性をまとめたホワイトペーパーと共に Office 2013 のスムーズな導入に向けた参考資料としてご利用ください。



以下に、本ドキュメント内で使用する製品の名称と略称を記述します。

- Microsoft Office (Office 2013)
- Microsoft Word 2013 (Word 2013)
- Microsoft Excel 2013 (Excel 2013)
- Microsoft PowerPoint 2013 (PowerPoint 2013)
- Microsoft Access 2013 (Access 2013)
- Microsoft InfoPath® 2013 (InfoPath 2013)
- Microsoft Outlook® 2013 (Outlook 2013)
- Microsoft Project® 2013 (Project 2013)
- Microsoft Visio® 2013 (Visio 2013)
- Microsoft SharePoint Server 2013 (SharePoint 2013)
- Microsoft Office 2010 (Office 2010)
- Microsoft SharePoint Server 2010 (SharePoint 2010)
- Microsoft Office 2007 (Office 2007)
- Microsoft Office SharePoint Server 2007 (SharePoint 2007)
- Microsoft Office 2003 Editions (Office 2003)
- Microsoft Office XP Edition (Office XP)
- Microsoft Office 2000 Editions (Office 2000)
- Microsoft Office 97 Editions (Office 97)
- Word/Excel/PowerPoint ファイル形式用 Microsoft Office 互換機能パック(Office 互換機能パック)
- Visual Basic for Application (VBA)
- Microsoft IME 2012 (IME 2012)

著作権

このドキュメントに記載されている情報は、このドキュメントの発行時点におけるマイクロソフトの見解を反映したものです。変化する市場状況に対応する必要があるため、このドキュメントは、記載された内容の実現に関するマイクロソフトの確約とはみなされないものとします。また、発行以降に発表される情報の正確性に関して、マイクロソフトはいかなる保証もいたしません。


このホワイト ペーパーに記載された内容は情報提供のみを目的としており、明示または黙示に関わらず、これらの情報についてマイクロソフトはいかなる責任も負わないものとします。

お客様ご自身の責任において、適用されるすべての著作権関連法規に従ったご使用をお願いします。このドキュメントのいかなる部分も、米国 Microsoft Corporation の書面による許諾を受けることなく、その目的を問わず、どのような形態であっても、複製または譲渡することは禁じられています。ここでいう形態とは、複写や記録など、電子的な、または物理的なすべての手段を含みます。ただしこれは、著作権法上のお客様の権利を制限するものではありません。

マイクロソフトは、このドキュメントに記載されている内容に関し、特許、特許申請、商標、著作権、またはその他の無体財産権を有する場合があります。別途マイクロソフトのライセンス契約上に明示の規定のない限り、このドキュメントはこれらの特許、商標、著作権、またはその他の無体財産権に関する権利をお客様に許諾するものではありません。

別途記載されていない場合、このドキュメントで使用している会社、組織、製品、ドメイン名、電子メール アドレス、ロゴ、人物、場所、出来事などの名称は架空のものです。実在する名称とは商品名、団体名、個人名などとは一切関係ありません。

© 2012 Microsoft Corporation. All rights reserved.

Microsoft、Office、Office 、Office 2013、Office 2010、Office 2007、Office 97、Office 2000、Office XP、Office 2003、Word、Excel、PowerPoint、Access、InfoPath、Outlook、Visio、Visual Basic、MSDN、SharePoint、IntelliSense、Windows、Windows 2000、Windows XP、Windows Vista、Windows Server は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

記載されている会社名、製品名には、各社の商標のものもあります。

開発環境の変更に関する留意点

Office 2013 では、Office 2007 で採用されたリボン インターフェイスを踏襲しており、Office 2003 以前のバージョンと比較するとユーザー インターフェイスが異なるため、開発コードへのアクセス方法や開発リファレンスへのアクセス方法も異なっています。この章では Office 2013 で開発を始めていただく際の留意点を説明します。

アプリケーションの開発タブ表示方法

リボン インターフェイスでも、VBA でのマクロ作成や記録されているマクロの実行は、Office 2003 以前のバージョンの Office と同様に可能です。マクロ コードの編集やマクロの記録などのマクロに関連する一連のコマンドは、[開発] タブに集約されています。なお、Access 2013 では、マクロに関連する一連のコマンドは、[データベース ツール] タブに表示されます。

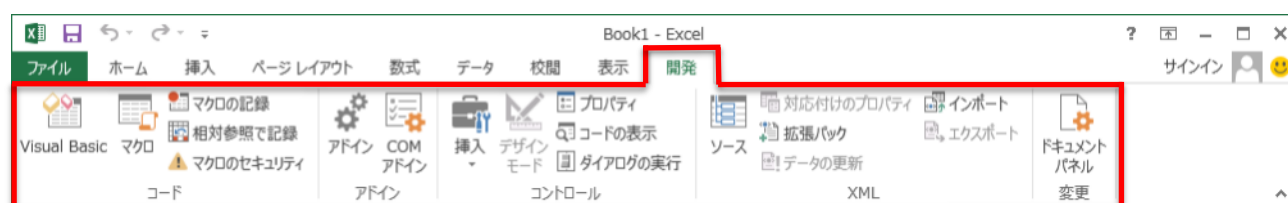


図 1: Excel における [開発] タブ

[開発] タブは、既定ではリボンに表示されていません。[開発] タブを表示するには、次のようにしてください。

➤ Word 2013、Excel 2013、PowerPoint 2013 の場合

1. [ファイル] タブをクリックし、[オプション] をクリックします。



図 2: Excel の [ファイル] タブをクリックして表示されるメニュー

2. カテゴリ ペインで [リボンのユーザー設定] をクリックし、メイン タブの一覧で、[開発] を選択します。

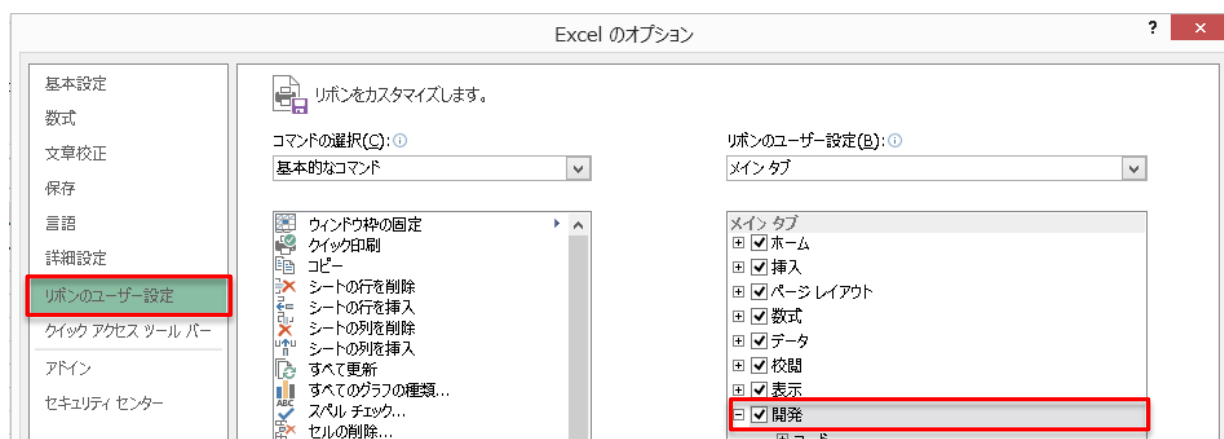


図 3: [Excel のオプション] ダイアログボックス

3. [OK] をクリックして、[オプション] ダイアログ ボックスを閉じます。

➤ Access 2013 の場合

[データベース ツール] タブは既定で表示されています。

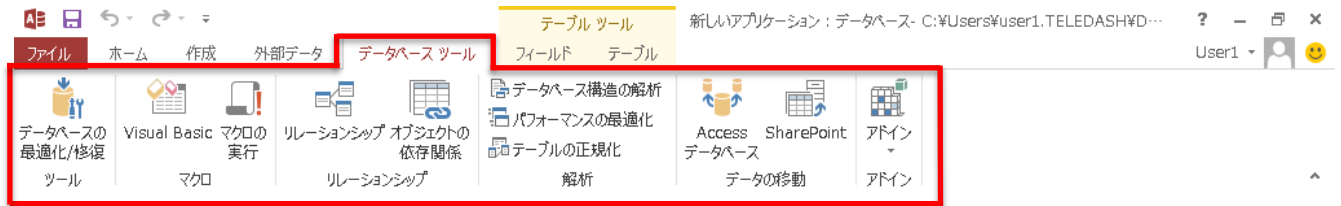


図 4 : Access における [データベース ツール] タブ

なお、Office 2007 では 1 つの Office アプリケーションで [開発] タブの表示を有効にすると、すべての Office アプリケーションで [開発] タブの表示が有効になります。Office 2010 および Office 2013 では、Office アプリケーション毎に [開発] タブの表示を有効にする必要があります。

[開発] タブの表示についての詳細はこちらをご覧ください。

➔ 「 [開発] タブを表示する」

<http://office.microsoft.com/ja-jp/excel-help/HA101819080.aspx>

➔ 「方法 : [開発] タブをリボンに表示する」

<http://msdn.microsoft.com/ja-jp/library/bb608625.aspx>

開発者用リファレンスの表示方法

Office 2013 では製品の使い方を説明するヘルプのほかに、開発者用のオンライン ヘルプを用意しています。Office 開発者用リファレンスには、Office 製品を活用したソリューションの開発に関する概念、プログラミング作業、サンプル、および利用したい機能についての Web サイトへの参照が用意されています。

また、Office 2013 の開発者用リファレンスでは、各アプリケーションで利用するオブジェクト モデルの新しい項目についての情報を提供するとともに、Office 97 以降の各バージョンの Office との間の、オブジェクト モデルの変更情報も提供しています。新しいメンバー、および更新されたメンバーの詳細については、以下の手順に従って表示される開発者用リファレンスの目次から " 新機能 " トピックをご覧ください。

➤ Visual Basic Editor (VBE) から起動

1. 起動している VBE 上で、F1 キーを押すか、または [ヘルプ] - [Microsoft Visual Basic for Applications ヘルプ] をクリックし、ヘルプ画面を起動します。

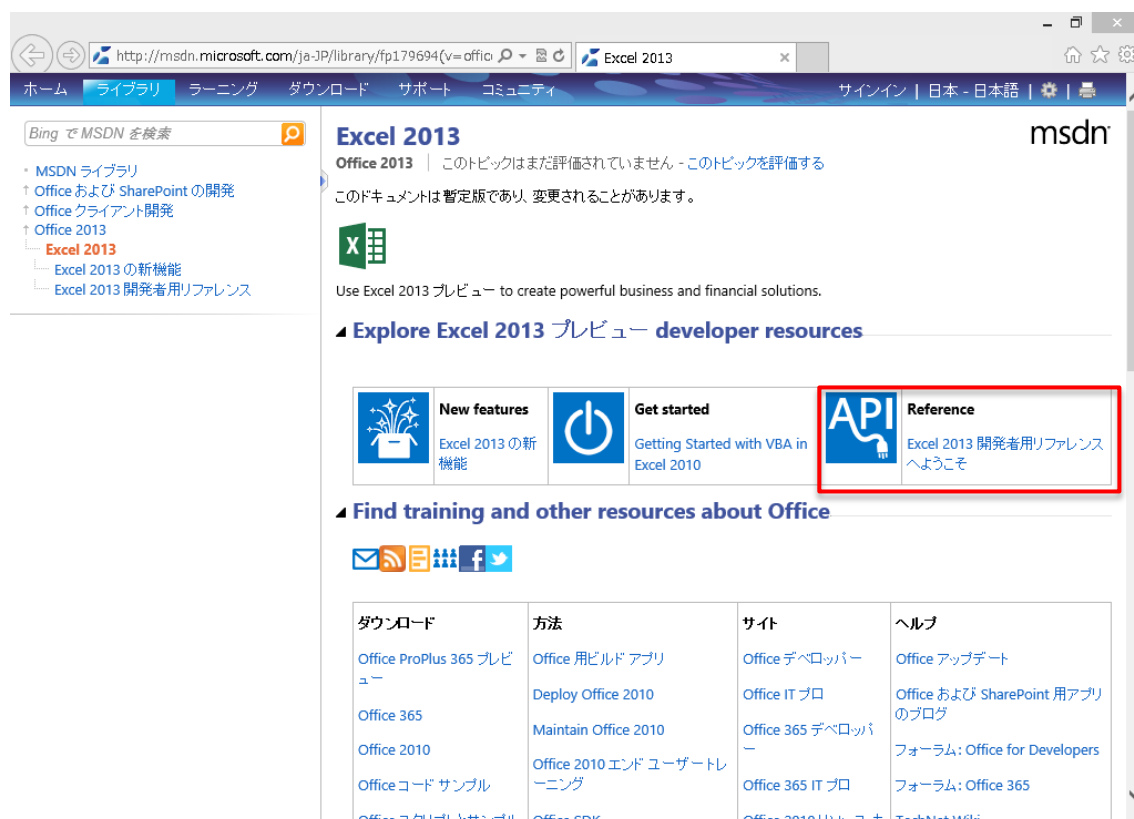


図 5 : Excel ヘルプの開発者用リファレンス画面 (Internet Explorer)

なお、以前のバージョンの Office に付属していた開発者用リファレンス ヘルプ ファイルは、セットアップから削除されました。



開発環境について

Office 2013 では、いくつかの開発環境が用意されており、要求に応じて使い分けることができます。

従来から利用されている、スクリプト コードによるカスタム アプリケーション開発や、マネージ コードによるセキュアでスケーラブルなカスタム アプリケーションの開発に加え、Office 2013 は新たに Web と統合し、Web ベースの言語 (HTML、CSS、JavaScript など) での開発が可能になりました。

Office 用アプリ (Apps for Office)

Office 用アプリは、HTML、CSS、JavaScript、REST などの標準 Web テクノロジを使用して Office クライアント アプリケーションを拡張する新しい方法を提供します。Office 用アプリにより、Office 2013 クライアント アプリケーションで新しい拡張モデルが有効になります。この新しいモデルは、Web 開発者が Office クライアント アプリケーションを拡張する Web ベースのソリューションを簡単に作成できるように作られています。Office 用アプリは基本的には Web ページであり、Office クライアント アプリケーションのドキュメント内で拡張コンテンツや機能を提供したり、クライアント アプリケーションと関連付けられた作業ウィンドウをホストしたり、電子メール メッセージでコンテキストに応じてアクティブ化したりできます。

Office 用アプリを実装するための新しい JavaScript ベースの API は、アプリケーション共通または固有のデータ型、オブジェクト、関数、イベントにより、複数の Office クライアント環境での簡単で一貫したプログラム アクセスを提供します。Office 用アプリの JavaScript API ライブラリには、Office ドキュメントの読み取りや書き込みを行い、選択変更などのユーザー イベントを処理するための、オブジェクトとメンバーが用意されています。

また、Office 用アプリは SharePoint Online や SharePoint、ファイル共有を利用した集中管理が可能なため、マクロやアドインと比較すると、配布やアップグレードを容易に行うことができます。そのため、ユーザーへの展開およびアップグレードの計画と実施にかかるコストを削減でき、さらに運用の簡素化が可能です。また、Office ストアで提供されているサードパーティ製の Office 用アプリも同様に容易に組織へ展開できます。ユーザーの Office ストアの利用は、グループ ポリシーにより制限することもできるため、組織における統制を行うこともできます。

Office 用アプリをサポートする Office 2013 クライアント アプリケーション

- Excel 2013
- Excel Web App
- Word 2013
- Outlook 2013
- Outlook Web App
- Project Professional 2013



Office 用アプリについての詳細はこちらをご覧ください。

➔ 「 Office 2013 用アプリの概要 」

<http://technet.microsoft.com/ja-jp/library/jj219429.aspx>

➔ 「 Office 用アプリの作成 」

<http://msdn.microsoft.com/ja-jp/library/jj220060.aspx>

Visual Basic for Application (VBA)

VBA は繰り返し作業の自動化や、操作の効率化のために利用できる Microsoft Office 向けのプログラミング言語です。プログラミング環境として Visual Basic Editor (VBE) を利用し、簡単にカスタム アプリケーションを開発できます。VBA は個別の Office 2013 アプリケーション開発用として、引き続き利用できます。

Microsoft Office 開発ツール (VSTO)

Microsoft Office 開発ツールは、以前のバージョンより、Visual Studio Tools for Office (VSTO) として Visual Studio と共に提供されています。最新の Visual Studio である Visual Studio 2012 では、Office 2007、Office 2010 および、Office 2013 用の Office ソリューション開発が可能です。Visual Studio 2012 には、Office 2013 を対象とするソリューションの作成用に新しいプロジェクト テンプレートが用意されています。さらに、Office 2013 の 32 ビット版および 64 ビット版のソリューションを作成できます。

また、リボン デザイナーを使用して、Office 2013 の各アプリケーションのリボンをカスタマイズできます。リボンをカスタマイズできるアプリケーションには、Word 2013、Excel 2013、PowerPoint 2013 の主要アプリケーションの他、InfoPath 2013、Project 2013、Visio 2013 などもあります。リボン デザイナーでは Outlook 2013 のエクスプローラー ウィンドウのリボンもカスタマイズできます。

➔ 「Visual Studio 2012 の新機能」

<http://msdn.microsoft.com/ja-jp/library/vstudio/bb386063.aspx>

➔ 「Visual Studio を使用した Office 用アプリの作成」

<http://msdn.microsoft.com/ja-jp/library/vstudio/jj620920.aspx>

➔ 「VBA ソリューションと Visual Studio Tools for Office ソリューションの比較」

<http://msdn.microsoft.com/ja-jp/library/ss11825b.aspx>

ソース コード管理 (Source Code Control) アドインの削除

Developer Source Code Control は、Access 2013 のアドインとして使用できなくなりました。



Office ストア

Office 2013 には、Microsoft Office ソリューション用の Office ストアが用意されています。Office ストアを使用して、組織で使用するソリューションを購入できます。また、アプリ カタログと呼ばれる組織用の内部ストアを利用することもできます。アプリ カタログでは、社内開発または社外開発の区別なく、すべての企業ソリューションが一元的に提供されます。これにより、IT 部門はソリューションを集中的に管理および監視でき、ユーザーはソリューションを簡単に見つけられるようになります。

Office ストアを使用すると、IT 部門とエンド ユーザーに複数のメリットがあります。

- **管理の簡素化**：すべてのソリューションを一元的に管理できます
- **セキュリティの強化**：アプリとセキュリティの脅威を 1 箇所で監視できます
- **集中構成**：ソリューションの可視性と可用性を管理し、集中的な構成を使用して悪質なコードを無効にできます
- **検出可能性**：ソリューションを簡単に見つけてダウンロードできます
- **リモート実行**：ユーザーのコンピューターにインストールしなくてもソリューションを利用できます

VBE (Visual Basic Editor) の変更点と問題点

VBA の開発環境である VBE には、基本的には変更はありません。ここでは一部の変更点と、既知の問題点を説明します。

変更点

➤ コード ウィンドウのホイール マウス対応

Office 2007 以降から、Visual Basic Editor (VBE) コード ウィンドウでホイール マウスのスクロールをサポートしました。

問題点

➤ IME の予測入力の表示画面の位置がおかしい

本ドキュメントのリリース時点で、VBE 上にて予測入力を使用して入力しようとする場合、入力場所とは異なる場所（主に直前に入力した場所）に、「x x（TAB キーを押すと予測候補を表示します）」という表示が出力されます。（xx は、予測候補）

現象

1. VBE 上で、“大きな栗の木” とタイプし入力します。
2. 改行を複数入れ、“おお” とタイプした際、「大きな栗の木（TAB キーを押すと予測候補を表示します）」のコメント画面が 1. の上の位置に出力されます。

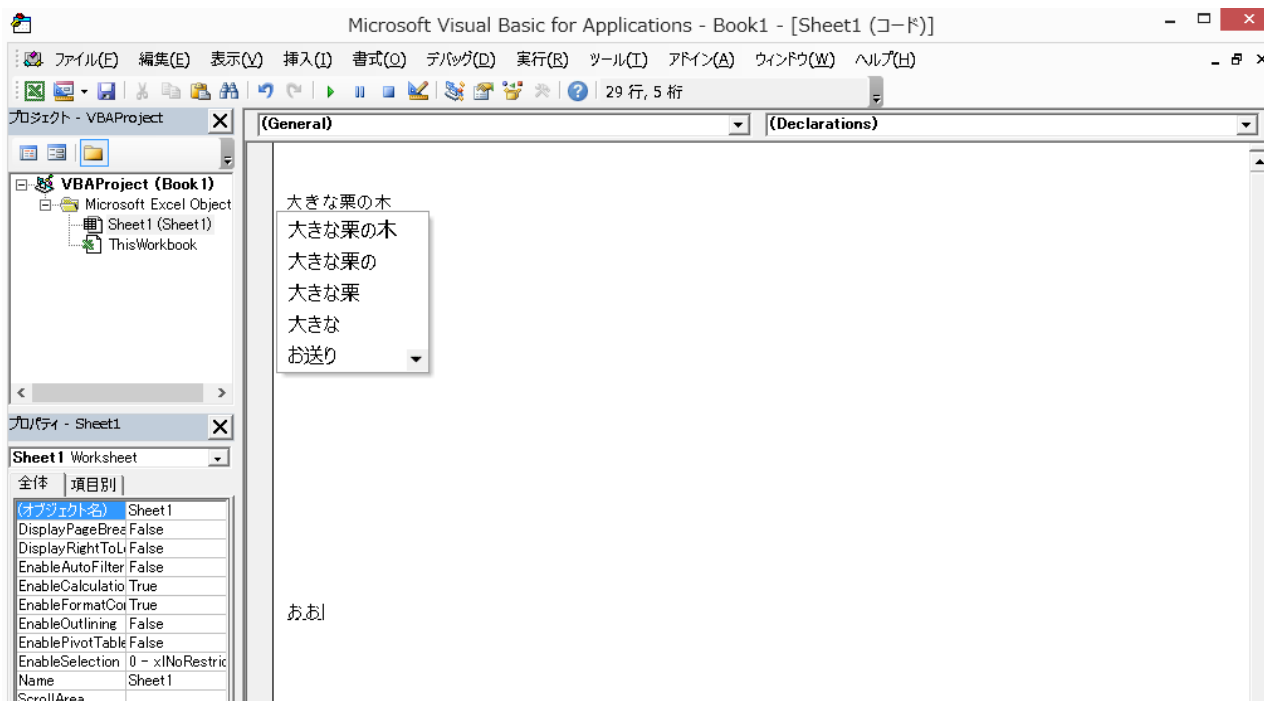


図 6：予測入力画面

マクロ全般に関する留意点

Office 2013 では、Office 2003 以前と同様に、VBA によるカスタム アプリケーションを利用できますが、いくつか留意すべき点があります。ファイル形式によるマクロの利用可否に加え、既定のインストールで提供されなくなったコンポーネントや、特定のオペレーティング システム (OS) によって起こる問題などを説明します。マクロ コードについての留意点は本ドキュメントの「[マクロ コードの見直しが必要な項目](#)」で説明します。また 64 ビット版 Office 2013 のマクロ コードについての留意点は本ドキュメントの「[64 ビット版 Microsoft Office 2013 の留意点](#)」で説明します。

VBA 言語互換性、VBA プロジェクトについて

Office 2013 では、Office 2010 の 32 ビット版、および 64 ビット版で新たに採用された VBA 7 が引き続き採用されています。VBA 7 は、Office 2007 以前に採用されていた VBA 6 と互換性があります。ただし、32 ビット環境と 64 ビット環境での動作について、マクロ コードの修正が必要になる場合があります。また、Office 2013 のオブジェクト モデルに対しても、追加、見直しが行われていますので、それに伴ったマクロ コードの修正が必要になる場合があります。

Office バージョン	VBA バージョン
Office 2013	7.1 (32 ビット版)
	7.1 (64 ビット版)
Office 2010	7.0 (32 ビット版)
	7.0 (64 ビット版)
Office 2007 SP 2	6.5
Office 2003 SP 3	6.5

表 1: Office バージョンと VBA バージョン

ファイル形式に関連した留意点

Office 2013 では、Office 2007 や Office 2010 と同様に、Open XML 形式のファイル フォーマットが採用され、Office 2003 以前のバージョンの Office で使われていた拡張子に加えて、新しい拡張子が用意されています。新しい拡張子ではマクロが有効なファイル形式と無効なファイル形式を拡張子で区別しているため、エクスプローラーやデスクトップなどのアイコンで、マクロが有効なファイルかどうかを判断することができるようになりました。

マクロが無効な Open XML 形式である、Word 2013 の拡張子 .docx 、Excel 2013 の拡張子 .xlsx や、PowerPoint 2013 の拡張子 .pptx では、マクロを含めてファイルを保存することができません。これらのファイルを編集、VBE を起動してマクロ コードを作成することができますが、ファイルの保存時に以下のメッセージが出力されます。

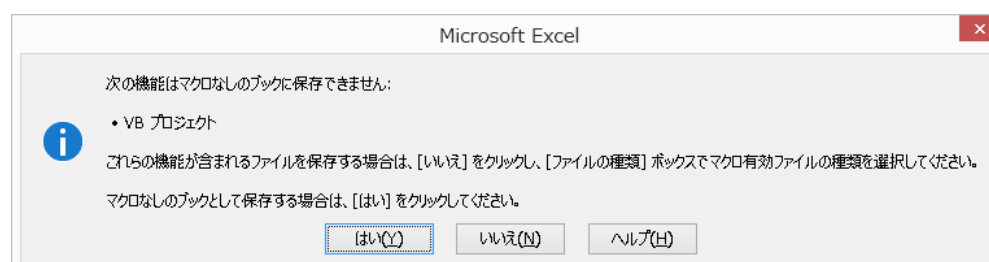


図 7: Excel でマクロなしで保存するかどうかのメッセージ


Office アプリケーション	通常ファイルの 拡張子・アイコン		マクロ有効ファイルの 拡張子・アイコン	
Word 2013	docx		docm	
Excel 2013	xlsx		xlsm	
PowerPoint 2013	pptx		pptm	

表 2: 通常ファイルの拡張子とマクロ有効ファイルの拡張子

Office 97 - 2003 互換形式のファイルについては、以前と同じようにマクロを含んだファイルを保存することができます。

次の表は各製品の主な拡張子ごとにマクロの有効かどうかを示しています。

アプリケーション名	拡張子	マクロの有効化	説明
Word 2013	.docx	不可	Word 文書
	.docm	可	Word マクロ有効文書
	.dotx	不可	Word テンプレート
	.dotm	可	Word マクロ有効テンプレート
	.doc	可	Word 97 - 2003 文書
	.dot	可	Word 97 - 2003 テンプレート
Excel 2013	.xlsx	不可	Excel ブック
	.xlsm	可	Excel マクロ有効ブック
	.xlsb	可	Excel バイナリ ブック
	.xltx	不可	Excel テンプレート
	.xltm	可	Excel マクロ有効テンプレート
	.xlam	可	Excel アドイン
	.xls	可	Excel 97 - 2003 ブック
	.xla	可	Excel 97 - 2003 アドイン
	.xlt	可	Excel 97 - 2003 テンプレート
PowerPoint 2013	.pptx	不可	PowerPoint プレゼンテーション
	.pptm	可	PowerPoint マクロ有効プレゼンテーション
	.potx	不可	PowerPoint テンプレート
	.potm	可	PowerPoint マクロ有効テンプレート
	.ppam	可	PowerPoint アドイン
	.ppsx	不可	PowerPoint スライドショー
	.ppsm	可	PowerPoint マクロ有効スライドショー
	.ppt	可	PowerPoint 97 - 2003 プレゼンテーション
	.ppa	可	PowerPoint 97 - 2003 アドイン
	.pot	可	PowerPoint 97 - 2003 テンプレート

表 3：ファイル拡張子及びそのファイルのマクロの有効化

➔ 「Office 2010 でサポートされるファイル形式」

<http://technet.microsoft.com/ja-jp/library/dd797428.aspx>

➔ 「Office 2010 の XML ファイル名拡張子」

<http://technet.microsoft.com/ja-jp/library/cc179191.aspx>

次にファイル形式に関する互換性の注意事項を記載します。

➤ Office 2013 非対応のファイル形式

Office 2013 では「97-2003 および 5.0/95 ブック形式」の保存をサポートしていませんので、マクロ内で「xlExcel9795」のファイル フォーマットを指定した構文がある場合などにエラーが発生することがあります。

たとえば、下記の構文が含まれるブックを Excel 2013 で開いても正常に動作させることができません。

`ThisWorkbook.SaveAs FileFormat:=xlExcel9795`

また、Visual Basic for Applications (VBA) のモジュール シートを含む Microsoft Excel 5.0/95 ブック ファイル (.xls) を開こうとした場合、または Excel 2013 で作成した VBA プロジェクトを含むブックを Microsoft Excel 5.0/95 ブック (.xls) のファイル形式で保存しようとした場合、「これらのファイルを開いたり保存したりするには、現在インストールされていないコンポーネントが必要です」、というエラー メッセージが表示されます。これらのエラー メッセージが表示された場合は、Office Web サイトで「VBA コンバーター」を検索することをお勧めします。

これらのエラーを解決する方法の詳細については、こちらをご覧ください。

➔ 「Excel でワークブックを開くか保存しようとする Microsoft Office オンラインで "VBA コンバーター" を検索することを要求されます。」

<http://support.microsoft.com/default.aspx?scid=kb;ja-jp;926430>

その他、Access 2013 では、Access データ プロジェクト (ADP) ファイル、および Access 97 データベース ファイルを開く機能が削除されています。

➤ RTF ファイル形式

RTF ファイル形式は Word 2010 以降の新しい機能に対応していません。Word 2010 以降のバージョンの新しい機能は、RTF 形式で保存すると失われます。また、Word 2013 は Open XML に基づく新しいコンバーター インターフェイスをサポートしています。

Word 2010 での変更点の詳細についてはこちらをご覧ください。

➔ 「Word 2010 での変更点 - 変更された機能」(Office 2010 時点の情報)

[http://technet.microsoft.com/ja-jp/library/cc179199\(office.14\).aspx#BKMK_Changed](http://technet.microsoft.com/ja-jp/library/cc179199(office.14).aspx#BKMK_Changed)

既定でインストールされなくなった機能

Office 2013 では、以前のバージョンの Office に既定でインストールされていた一部のコンポーネントと機能がインストールされなくなりました。ここではそれらの機能のうち、マクロを利用したカスタム アプリケーションでの影響が考えられる機能を説明します。

共通コンポーネント

➤ Office Web コンポーネント (OWC)

OWC は Office 2007 以降では提供されなくなりました。詳細は本ドキュメントのトピック「[Office Web Components \(OWC\) の提供](#)」をご覧ください。

➤ 日本語音声合成エンジン

日本語音声合成エンジンは Office 2007 以降では提供されなくなりました。

➤ Office Startup Assistant (OSA)

Office Startup Assistant (OSA) は、Office 2007 では既定で無効になっていましたが、Office 2010 から提供されなくなりました。代わりに Office 2010 から、各アプリケーションの Backstage ビューの [新規作成] タブが改善されました。

➤ Microsoft Office Document Imaging (MODI)

Microsoft Office Document Imaging (MODI) は、Office 2010 から提供されなくなりました。以下の 4 機能については、代替方法があります。

- Microsoft Office Document Scanning
- Microsoft Document Image Writer のプリンター ドライバー
- .tiff ファイル、.mdi ファイルの表示
- OCR (Optical Character Recognition)

詳細はこちらをご覧ください。

- ➔ 「Microsoft Office Document Imaging (MODI) の機能を回復する代替の方法」 (Office 2010 時点の情報)
<http://support.microsoft.com/kb/982760/ja>

➤ アドインデザイナー

Office 2013 には Msaddndr.dll が含まれません。従って Msaddndr.dll を使用して作成されたアドインは動作しません。

詳細はこちらをご覧ください。

- ➔ 「A custom add-in that uses interfaces in the Msaddndr.dll file does not work in Office 2013」
<http://support.microsoft.com/kb/2792179/ja-jp>

Word 2013

➤ OASYS コンバーター

OASYS コンバーターは、Word 2010 より提供されなくなりました。そのため、ワープロ専用機 OASYS のデータファイルを Word のデータとして読み込むことができなくなりました。

➤ 一太郎コンバーター

一太郎 コンバーターは、Word 2010 より提供されなくなりました。そのため、一太郎のデータファイルを Word のデータとして読み込むことができなくなりました。

➤ カスタム XML マークアップ

カスタム XML マークアップ機能は、Word 2010 より提供されなくなりました。そのため、カスタム XML マークアップを含むファイルを開いた場合、ファイルを保存すると、XML マークアップが削除されます。

詳細はこちらをご覧ください。

➔ 「Word 2013 で文書を開くときにカスタム XML マークアップが削除されます。」

<http://support.microsoft.com/kb/2761189/ja>

Excel 2013

➤ Speak メソッドで日本語音声合成の使用ができない

日本語音声合成エンジンが Office 2007 以降では提供されなくなったため、Range.Speak および Application.Speech.Speak メソッドが使用できません。使用するためには音声合成エンジンがインストールされている必要があります。

➤ Lookup ウィザード

Lookup ウィザードは、Excel 2010 より提供されなくなりましたが、INDEX 関数と MATCH 関数を組み合わせた式を作成することで、列と行の交点の値を検索する数式を作成することができます。

➤ インターネット アシスタント VBA

インターネット アシスタント VBA は、Excel 2010 より提供されなくなりました。開発者がインターネット アシスタントの構文を使用して、Excel のデータを Web ページに発行することはできなくなりました。

➤ 条件付き合計式ウィザード

条件付き合計式ウィザードは、Excel 2010 より提供されなくなりましたが、SUMIF 関数や SUMIFS 関数を利用した式を作成することで、指定した条件が満たされる場合に、セル範囲のデータを合計する数式を作成することができます。

Access 2013

➤ Microsoft Office Snapshot Viewer

Snapshot Viewer は Access 2007 以降では提供されなくなりました。Access 2007 以降で使用する場合は、ダウンロード センターから入手してご使用ください。

➤ 「Microsoft Access Snapshot Viewer」 (Office 2007 時点の情報)

<http://www.microsoft.com/downloads/details.aspx?displaylang=ja&FamilyID=b73df33f-6d74-423d-8274-8b7e6313edfb>

➤ データ アクセス ページ (DAP)

Access 2007 以降ではデータ アクセス ページ (DAP) 機能は提供されなくなりました。DAP で保存されたページは Internet Explorer で表示することはできますが、Access 2007、Access 2010 および Access 2013 で編集することはできません。

データ アクセス ページを使用する代替方法として、Web データベースを作成し、Access Services を使用して SharePoint サイトに発行する方法があります。

詳細については、こちらをご覧ください。

➔ 「SharePoint Server 2010 (における Access Services の新機能) (Office 2010 時点の情報)

<http://go.microsoft.com/fwlink/?linkid=182566&clcid=0x411>

➤ カレンダー コントロール

Access 2010 から、Microsoft カレンダー コントロール (mscal.ocx) は提供されなくなりました。代わりに、Access 2010 および Access 2013 では日付選択コントロールを使用できます。以前のバージョンの Access でカレンダー コントロールを使用していたアプリケーションを Access 2010 もしくは Access 2013 で開くと、エラー メッセージが表示され、カレンダー コントロールは表示されません。

この問題を解決するための回避策として、Access 2007 または以前のバージョンの Access を取得し、その中に含まれるカレンダー コントロール (mscal.ocx) を使用してください。

➔ 「Access 2007 ダウンロード : Access Runtime 」

<http://www.microsoft.com/downloads/details.aspx?FamilyID=d9ae78d9-9dc6-4b38-9fa6-2c745a175aed&displaylang=ja>

➤ dBASE サポート

Access 2013 から dBASE サポートが削除されたので、ユーザーは外部データ dBASE データベースに接続できなくなります。Access データ プロジェクト ファイルを開く機能が削除されました。



➤ Access の 3 段階の状態管理ワークフロー

Access 2013 からワークフローのエントリ ポイントは使用できなくなります。UI マクロでは、ワークフロー コマンドを使用できません。新しいワークフローの開始またはワークフロー タスクを持つ UI マクロがある既存の Access 2010 データベースを開くと、Access によって警告が表示されます。

InterConnect

InterConnect は Office 2010 から提供されなくなりました。Microsoft Office InterConnect 2007 は、今後のバージョンの Microsoft Office でも、更新プログラムとして提供されるサービス パックを適用することで引き続き利用できます。

SharePoint Workspace

SharePoint Workspace が Office 2013 製品ベースから削除されています。代替機能として Office 2013 に付属する Foler Sync をリリースします。この機能により、SharePoint のドキュメント ライブラリをオフラインにして同期できます。

Microsoft Office Picture Manager

Picture Manager は Office 2003 でリリースされました。Office 2013 からこの機能は基本的に Windows Live フォト ギャラリーに置き換えられています。

Microsoft クリップ オーガナイザー

Office 2013 ではクリップ オーガナイザー機能が削除されています。この機能は [メディアの挿入] ダイアログ ボックスに置き換えられています。この機能を使用すると、Office.com のクリップ アート コレクションやその他のオンライン ソースからコンテンツを見つけて挿入することができます。

Office 2013 の変更点の詳細についてはこちらをご覧ください。

➔ 「Office 2013 での変更点」

[http://technet.microsoft.com/ja-jp/library/cc178954\(v=office.15\)](http://technet.microsoft.com/ja-jp/library/cc178954(v=office.15))

マクロの記録について

Office 2013 では、以前のバージョンに引き続きマクロの記録を利用できます。マクロの記録により自動生成されたソース コードを利用することで、開発者の生産性を高めることができます。但し、一部製品のマクロ記録機能の廃止や、オブジェクト モデルの変更などがあるため注意が必要となります。

Excel の図形オブジェクトの操作

Excel 2007 では図形オブジェクトの操作はマクロに記録されなくなりましたが、Excel 2010 以降では、図形オブジェクトの操作をマクロに記録することができるようになりました。

PowerPoint の [マクロ記録] 機能

PowerPoint 2007 以降では、[マクロ記録] 機能がなくなりました。

詳細はこちらをご覧ください。

➔ 「マクロの記録について」 (Office 2010 時点の情報)

<http://office.microsoft.com/ja-jp/powerpoint-help/HA010338328.aspx?CTT=1>

オブジェクトの操作

リボン インターフェイスの導入や、オブジェクト モデルの変更、オブジェクト既定名の変更などにより、以前のバージョンで生成されたマクロ コードが期待どおりに動作しない場合があります。CommandBar オブジェクトや図形 (Shape) オブジェクトを操作するマクロ コードを見直し、必要に応じて修正を行ってください。

詳細は本ドキュメントの「[Excel 2013 の変更](#)」をご覧ください。



オペレーティング システム (OS) によって起こる問題について

Office 2003 以前のバージョンの Office で開発したカスタム アプリケーションを Office 2013 へ移行する際には、実行する OS によっても配慮すべき点があります。以下に、製品別に既知の問題点を挙げておきます。

Access 2013

Access 2013 で、全角のアラビア数字で始まる 1 文字以上のフィールド名が含まれるクエリーを実行すると、クエリー式のエラーが表示されます。

詳しくは以下を参照してください。

- ➔ 「Access 2007 または Access 2003 でクエリーを実行すると、全角のアラビア数字が原因でエラー メッセージが表示される」 (Office 2007 時点の情報)
<http://support.microsoft.com/kb/932994/ja>

Windows Internet Explorer 7 以降のバージョン

Internet Explorer 7 以降で Office 2013 のドキュメントを表示すると、新しいウィンドウが開くことがあります。レジストリの変更をすることにより、従来の Internet Explorer の様に同じウィンドウ内でドキュメントを開くことができます。

詳しくは以下を参照してください。

- ➔ 「Windows Internet Explorer 7 または Internet Explorer 8 で Office 2007 のドキュメントを表示すると新しいウィンドウが開く」 (Office 2007 時点の情報)
<http://support.microsoft.com/kb/927009/ja>

64 ビット版 Microsoft Office 2013 について

Office 2013 では 従来の 32 ビット版に加え、64 ビット版が提供されます。以下では、64 ビット版 Office のメリットと留意すべき点について説明します。また、64 ビット版 Office 2013 の展開についても説明します。

64 ビット版 Microsoft Office 2013 のメリット

大容量のメモリが使用可能

32 ビット版では、使用できるメモリの上限が 4 ギガバイト (GB) でしたが、64 ビット版では、4 ギガバイト (GB) を超えた大容量のメモリを使用することができます。

- Excel で 2 ギガバイト (GB) を超える大きさの Excel スプレッドシートを使用する
- Project で大規模なプロジェクトを構成する多数のサブ プロジェクトを扱う

特に、上記のような場合に大容量のメモリを利用できることによるメリットが大きく発揮されます。

ハードウェアによるデータ実行防止 (DEP) により、既定でセキュリティが強化される

DEP は、意図しないコードが実行されないようにすることでコンピューターを安全に保つことができる機能です。Office 2013 の 32 ビット版では、必要に応じて DEP をアプリケーション レベルで構成しますが、Office 2013 の 64 ビット版の場合は、Office アプリケーションに常に適用されるため、セキュリティが強化されます。

詳細についてはこちらをご覧ください。

➔ 「Office 2013 の 64 ビット版」

[http://technet.microsoft.com/ja-jp/library/ee681792\(v=office.15\).aspx](http://technet.microsoft.com/ja-jp/library/ee681792(v=office.15).aspx)

64 ビット版 Microsoft Office 2013 の留意点

ActiveX コントロールと COM アドイン

64 ビットに対応したオペレーティング システムが動作しているコンピューターには 64 ビット版と 32 ビット版の ActiveX コントロールや COM アドインをインストールすることができますが、32 ビット版 Office 用に作成された ActiveX コントロールとアドイン (COM) DLL (ダイナミック リンク ライブラリ) は、64 ビット版のプロセスで動作しません。64 ビット版には、64 ビット版のコントロールのみを実行することができますので、64 ビット版のコントロールとアドイン (COM) が必要となります。

詳細はこちらをご覧ください。

- ➔ 「Office 2010 の 32 ビット バージョンと 64 ビット バージョンとの互換性」 (Office 2010 時点の情報)
[http://msdn.microsoft.com/ja-jp/library/ee691831\(office.14\).aspx](http://msdn.microsoft.com/ja-jp/library/ee691831(office.14).aspx)

Visual Basic for Applications (VBA)

➤ VBA 7

VBA 7 は VBA 6 に代わる新しい VBA エンジンです。VBA 7 は Office 2010 の 64 ビット版に対応するためにリリースされました。Office 2013 の 32 ビット版および 64 ビット版の両方で使用できます。

➤ コンパイラ定数

“VBA 7” と “Win 64” という 2 つの条件付きコンパイラ定数が用意されています。VBA 7 定数を使用すると、アプリケーションで VBA 7 と VBA 6 のどちらを使用しているのかを判別することによって、コードの下位互換性を確保できます。Win 64 定数は、コードが 32 ビット版として実行されるのか、64 ビット版として実行されるのかを判別するために使用します。

次のコードは実行環境によって結果が異なります。

```
#If VBA7 And Win64 Then
    Const C_VERSION As String = "64 ビット版 VBA7"
#ElseIf VBA7 Then
    Const C_VERSION As String = "32 ビット版 VBA7"
#Else
    Const C_VERSION As String = "VBA7 以外のバージョン"
#End If

Sub VBA_version()
    MsgBox C_VERSION
End Sub
```

64 ビット環境でのコンパイラ定数は以下の通りです。

定数	値	説明
Vba6	True/False	実行環境が Visual Basic for Applications Version 6.x であるかどうかを示します。
Vba7	True/False	実行環境が Visual Basic for Applications Version 7.x であるかどうかを示します。
Win32	True/False	実行環境が 32 ビットであるかどうかを示します。
Win64	True/False	実行環境が 64 ビットであるかどうかを示します。

表 4: 64 ビット環境でのコンパイラ定数

➤ 64 ビット版 VBA

VBA に関しては、32 ビット版で利用していたコードを、基本的には変更なしで 64 ビット版 Office 2013 で利用できます。しかし、一部のコードについては、64 ビット版用のコードを使用する必要がありますので、そのコードに特定のキーワードと条件付きコンパイラ定数を含めることで、以前のバージョンの Office との下位互換性が確保されるように処置する必要があります。

従来の VBA コードを 64 ビット版 Office で実行する場合に問題となるのは、64 ビットの情報を 32 ビットのデータ型に読み込もうとした結果、情報の一部が切り捨てられてしまうことです。その結果、メモリのオーバーランやコードの予想外の動作、アプリケーション エラーが発生する可能性があります。

➤ 外部関数の利用

VBA ライブラリとタイプ ライブラリを組み合わせれば、Microsoft Office アプリケーションの作成にさまざまな機能を使用できるようになります。ただし、コンピューターのオペレーティング システムや他のコンポーネントとの直接のやりとりが必要になることもあります。たとえば、メモリやプロセスを管理するとか、ウィンドウ、コントロールなどのユーザー インターフェイスを操作するとか、Windows レジストリに変更を加える場合などが、これに当たります。このような状況では、ダイナミック リンク ライブラリ (DLL) ファイルに組み込まれた外部関数を使用するのが最善策です。VBA でこれを行うには、Declare ステートメントを使用して API 呼び出しを行います。

Declare ステートメントを使用して Windows アプリケーション プログラミング インターフェイス (API) またはその他の DLL エントリ ポイントにアクセスする VBA コードは、32 ビット版と 64 ビット版で異なります。API への入力および出力を確認して更新した後で、Declare ステートメントに PtrSafe 属性を追加して更新する必要があります。Declare ステートメントは、PtrSafe 属性がないと 64 ビット版 VBA 7 で動作しません。64 ビット版 VBA 7 には、新しいデータ型の LongLong、LongPtr と新しいキーワードの PtrSafe が追加されました。

次のコードは VBA 6 サンプル コードです。

```
Declare Function SHBrowseForFolder Lib "shell32.dll" _
    Alias "SHBrowseForFolderA" (lpBrowseInfo As BROWSEINFO) As Long

Public Type BROWSEINFO
    hOwner As Long
    pidlRoot As Long
    pszDisplayName As String
    lpszTitle As String
    ulFlags As Long
    lpfn As Long
    lParam As Long
    iImage As Long
End Type

Sub TestSHBrowseForFolder ()
    Dim bInfo As BROWSEINFO
    Dim pidList As Long
    bInfo.pidlRoot = 0&
    bInfo.ulFlags = &H1
    pidList = SHBrowseForFolder(bInfo)
End Sub
```

次のコードは VBA 7 のサンプル コードです。32 ビット コードと 64 ビット コードを組み合わせで使用する場合には、正しいコードが実行されるように処置する必要があります。

```
#if VBA7 then      ' VBA7
  Declare PtrSafe Function SHBrowseForFolder Lib "shell32.dll" _
  Alias "SHBrowseForFolderA" (lpBrowseInfo As BROWSEINFO) As Long
Public Type BROWSEINFO
  hOwner As LongPtr
  pidlRoot As Long
  pszDisplayName As String
  lpszTitle As String
  ulFlags As Long
  lpfn As LongPtr
  lParam As LongPtr
  iImage As Long
End Type

#else      ' Downlevel when using previous version of VBA7
  Declare Function SHBrowseForFolder Lib "shell32.dll" _
  Alias "SHBrowseForFolderA" (lpBrowseInfo As BROWSEINFO) As Long
Public Type BROWSEINFO
  hOwner As Long
  pidlRoot As Long
  pszDisplayName As String
  lpszTitle As String
  ulFlags As Long
  lpfn As Long
  lParam As Long
  iImage As Long
End Type

#end if
Sub TestSHBrowseForFolder ()
  Dim bInfo As BROWSEINFO
  Dim pidList As Long

  bInfo.pidlRoot = 0&
  bInfo.ulFlags = &H1
  pidList = SHBrowseForFolder(bInfo)
End Sub
```

VBA の詳細については、Office アプリケーションの Microsoft Visual Basic for Applications ヘルプの「64 ビット版 Visual Basic for Applications の概要」および「Declare ステートメントの使用例」を参照してください。

VBA の互換性についてはこちらをご覧ください。

➔ 「Office 2010 の 32 ビット バージョンと 64 ビット バージョンとの互換性」 (Office 2010 時点の情報)
[http://msdn.microsoft.com/ja-jp/library/ee691831\(office.14\).aspx](http://msdn.microsoft.com/ja-jp/library/ee691831(office.14).aspx)

描画エンジン

グラフィカル デバイス インターフェイス (GDI) の 32 ビット版と 64 ビット版には違いがあります。64 ビット版では MMX がサポートされないため、パフォーマンスに影響が生じる可能性があります。

Word

- Microsoft Office Document Imaging (MODI)
Microsoft Office Document Imaging (MODI) は、Office 2013 で提供されなくなりました。また、MODI に関連するすべてのコンポーネントは、Office 2013 の 32 ビット版と 64 ビット版の両方で非推奨となっています。
- 数式エディター
レガシーの数式エディターは 64 ビット版 Office 2013 ではサポートされませんが、32 ビット版 Office 2013 インストール (WOW64) ではサポートされます。

詳細については、こちらをご覧ください。

➔ 「Office 2013 の 64 ビット版 – 全般的な機能後退」
[http://technet.microsoft.com/ja-jp/library/ee681792\(v=office.15\).aspx#BKMK_FeatureDeprecations](http://technet.microsoft.com/ja-jp/library/ee681792(v=office.15).aspx#BKMK_FeatureDeprecations)

Publisher

Microsoft Works データベース コンバーター (wdbimp.dll) は、Office 2010 の 32 ビット版および 64 ビット版の両方のインストールから削除されました。Office 2010 よりも前のバージョンでは、このコンバーターは、Microsoft Works で作成されたデータ ソースへ接続するために差し込み印刷機能で使用されていました。

64 ビット版 Microsoft Office 2013 の展開

Office 2013 では、64 ビット版の Windows オペレーティング システム上で実行される 64 ビット版の Office 2013 アプリケーションはもちろんのこと、32 ビット版 Office 2013 アプリケーションの動作もサポートしています。64 ビット版のオペレーティング システムで動作する 32 ビット版 Office 2013 アプリケーションをサポートすることにより、コントロール、アドイン、および VBA との互換性が向上します。

インストールする Office 2013 のエディションを選択するときの推奨事項は次の通りです。

32 ビット版 Office 2013 は 64 ビット版と同じ機能を提供し、32 ビット版のアドインとも互換性があります。Office に対する既存の拡張機能に依存している場合は、32 ビット版および 64 ビット版のサポートされる Windows オペレーティング システムを実行しているコンピューターに 32 ビット版 Office 2013 をインストールすることをお勧めします。

メリットにも書かれているような、大きい Excel スプレッドシートを使用している Excel の熟練ユーザーが組織内にいる場合は、64 ビット版 Office 2013 が効果を発揮します。また、社内のソリューション開発者がいる場合は、64 ビット版 Office 2013 で社内ソリューションをテストおよび更新できるように、64 ビット版 Office 2013 のインストールをお勧めします。

なお、Office 2013 のインストールを行うにあたり、1 台の PC に 64 ビット版と 32 ビット版を共存させることはできません。

➔ 「Office 2013 のシステム要件」

[http://technet.microsoft.com/ja-jp/library/ee624351\(v=office.15\).aspx](http://technet.microsoft.com/ja-jp/library/ee624351(v=office.15).aspx)

セキュリティ機能に関連する留意点

Office 2007 以降では、セキュリティ機能の見直しが行われており、既定のセキュリティ設定や警告メッセージ等のユーザー インターフェイスが変更されています。また、Office 2013 では、ユーザー設定のローミングをはじめ、新しく機能が追加されています。既定で、高いセキュリティの設定となっているため、カスタム アプリケーションを実行する環境を見直し、利用者へのアナウンス、インストール時のカスタマイズ等での対応が必要になることが考えられます。またマクロへの電子署名など、開発者が留意する点があります。

Office 2013 の新しいセキュリティ機能

Office 2013 で追加または強化された 6 つの機能について説明します。新しい 6 つの機能は、クラウド 環境との連携や利便性の向上、悪用の防止を狙いとしたセキュリティ対策を提供します。

クラウド コンテンツ への シングル サインイン

ユーザーは、単純なサインイン操作により、Office からクラウドのコンテンツに簡単にアクセスできます。プロファイルを作成して 1 回サインインするだけで、ローカルとクラウドの Office ファイルにシームレスにアクセスして作業できます。認証をやり直す必要はありません。サインインには、マネージ OrgID かフェデレーション OrgID を使用します。Office 2013 の使用中にそれらの ID を簡単に切り替えることもできます。

ユーザー設定のローミング

ユーザー設定（基本設定および最近使ったドキュメント）のローミングや各 ID で使用できるサービスを追跡する API が提供されます。これにより、ユーザーのワークフローが改善され、IT 管理者による認証の管理も容易になります。ユーザーは、使用する ID プロバイダー（Microsoft アカウントまたは Office 365 へのアクセスに使用するユーザー ID）やアプリで使用されている認証プロトコルに関係なく、セッションごとに 1 回認証を行うだけで、SkyDrive やすべての Office アプリを使用できます。認証と ID プロビジョニングは、完全にクラウドで行うことも、Active Directory、Forefront Identity Manager (FIM)、または Active Directory フェデレーション サービス (ADFS) を使用して行うこともできます。

新しいファイル暗号化方式

パスワードを忘れて問題になることもなくなりました。新しい Office 2013 のファイル暗号化方式では、パスワードで保護された Word、Excel、PowerPoint のファイルを、IT 管理者が秘密エスクロー キーを使用して解読できます。パスワードの適用時に公開キーと証明書のペアがファイルのヘッダーに格納されるようになりました。パスワードをなくしたり忘れたりした場合は、IT 管理者がコマンド ライン ツール（DocRecrypt）を使用して（もちろん、アクセス許可と、証明書と秘密キーへのアクセスも必要です）、パスワードを解除するか、ファイルを新しいパスワードで保護することができます。

詳細はこちらをご覧ください。

- 「Remove or reset file passwords in Office 2013」 (英語)

<http://technet.microsoft.com/en-us/library/jj923033.aspx>

- 「Office 2013 で、Word、Excel、PowerPoint ファイルに設定したパスワードの変更や削除が可能になりました!」

http://blogs.technet.com/b/microsoft_office_/archive/2013/02/12/office-2013-word-excel-powerpoint.aspx

デジタル署名の強化

デジタル署名も強化されています。Open Document Format (ODF v1.2) ファイル形式がサポートされるようになり、サードパーティ製の他のアプリケーションなどで署名された ODF ドキュメントを Office 2013 で開いた場合にデジタル署名を確認できるようになりました。さらに、Office 2013 では、ドキュメントが XAdES (XML Advanced Electronic Signatures) を使用して署名されている場合に、ドキュメントが署名された場所の住所、署名者の役割または役職、署名の目的などの詳細情報を含めることもできます。

新しい UI の IRM クライアント

Information Rights Management (IRM) も簡単になりました。Office 2013 には、新しい UI を搭載した IRM クライアントが含まれています。これにより、たとえば、取引先と共同作業していて非公開のドキュメントを共有するユーザーが簡単に ID を選択できます。Rights Management Services (RMS) サーバーのサービスの自動検索もサポートされています。さらに、Microsoft Office Web Apps で読み取り専用の IRM がサポートされているため、SharePoint ライブラリ内の IRM で保護されているドキュメントや Outlook のメッセージに添付されている IRM で保護されているドキュメントを Office Web Apps で表示できます。

保護ビューの強化

保護ビューは、制限された環境 (lowbox と呼ばれます) でファイルを開くことによってコンピューターの悪用を防止します。これにより、ファイルを Excel、PowerPoint、または Word で編集用を開く前に調べることができます。Office 2013 では、プロセスの分離が強化されているほか、ファイルがユーザーによって安全と見なされるまで lowbox からのネットワーク アクセスがブロックされます。

詳細はこちらをご覧ください。

- ➔ 「Office 用アプリのプライバシーとセキュリティ」

[http://msdn.microsoft.com/ja-jp/library/office/fp161165\(v=office.15\).aspx](http://msdn.microsoft.com/ja-jp/library/office/fp161165(v=office.15).aspx)

- ➔ 「Office 2013 のセキュリティの概要」

[http://technet.microsoft.com/ja-jp/library/cc179050\(v=office.15\).aspx](http://technet.microsoft.com/ja-jp/library/cc179050(v=office.15).aspx)

セキュリティ センターについて

Office 2007 以降では、セキュリティとプライバシーに関する設定を、セキュリティ センターで一括して設定できるようになりました。また、Office 2013 では、Office 2003 以前のバージョンで使用されていた [最高]、[高]、[中]、[低] のセキュリティ レベルが簡素化され、高いセキュリティとユーザーの利便性を両立した設定になりました。

セキュリティ センターを表示するには、[ファイル] - [オプション] - [セキュリティ センター] の順にクリックします。



図 8: 左ペインの [セキュリティ センター] をクリックして表示されるダイアログ ボックス

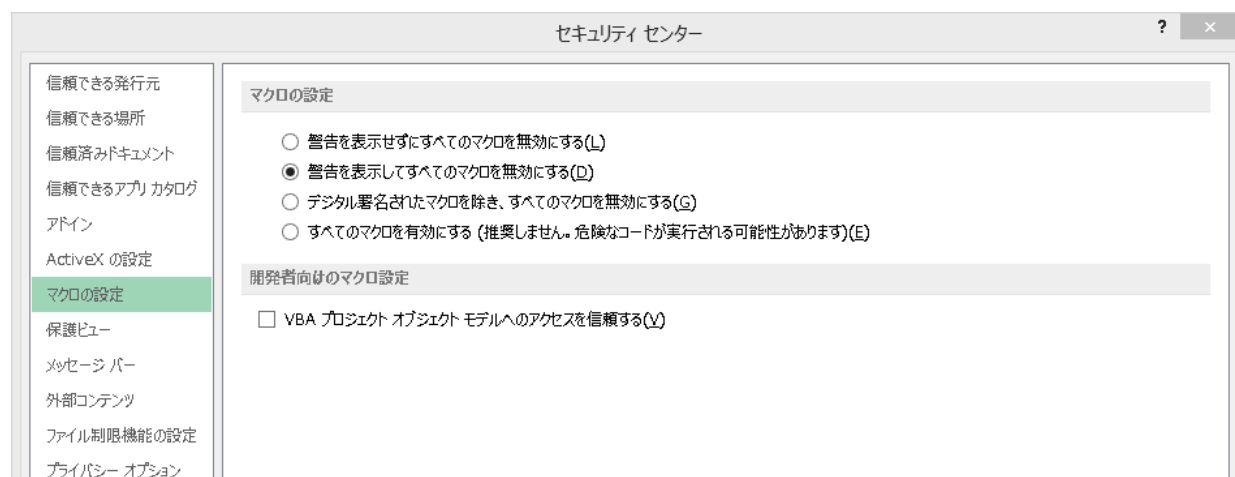


図 9: [セキュリティ センターの設定] ボタンをクリックして表示されるダイアログ ボックス

セキュリティ センターでは、左側のメニューにより下記の設定を行うことができます。またアプリケーションによって設定できる項目が異なっています。

メニュー	設定内容
信頼できる発行元	マクロに添付されるデジタル署名で利用される証明書の、信頼できる発行元の管理
信頼できる場所	信頼できるマクロを保存しているフォルダーの管理
信頼済みドキュメント	マクロが含まれているファイル等を常に有効にするかどうかの設定
アドイン	アドイン形式のマクロを有効にするかどうかの設定
信頼されているアプリケーション カタログ	Office 用アプリのアプリケーション カタログの管理
ActiveX の設定	ActiveX コントロールを有効にするかどうかの設定
マクロの設定	マクロを有効にするかどうかの設定
保護ビュー	サンドボックス環境でプレビューするかどうかの設定
メッセージ バー	マクロを無効にしたときメッセージを表示するかどうかの設定
外部コンテンツ (Excel のみ)	外部のデータへのリンクを有効にするかどうかの設定
ファイル制限機能の設定	有効なファイル形式の定義に適合しないファイルを識別するかどうかの設定
プライバシー オプション	Microsoft に対するプライバシーについての設定

表 5: セキュリティ センターで設定可能な項目

Office 2007 では、『信頼できる場所』、『ファイル制限機能の設定』という機能が追加され、Office 2010 では、『信頼済みドキュメント』、『保護ビュー』という機能が追加され、『ファイル制限機能の設定』の拡張と強化が行われました。さらに Office 2013 では、『信頼されているアプリケーション カタログ』という機能が追加されました。

➤ 信頼できる場所

ローカル フォルダーやネットワーク共有のフォルダーを、信頼できる場所として指定することで、そのフォルダー内のファイルは信頼できるファイルとして扱われ、未署名のマクロ、ActiveX コントロール、外部データ ソースへのリンク等が警告されません。

詳細はこちらをご覧ください。

➔ 「Office 2010 の信頼できる場所の設定を計画する」

<http://technet.microsoft.com/ja-jp/library/cc179039.aspx>

➤ 信頼済みドキュメント

マクロが含まれているファイル等を開いた場合、セキュリティ上の危険性が存在する場合があります。メッセージ バーにセキュリティの警告が表示されます。信頼済みドキュメントでは、ファイルのマクロや ActiveX コントロールなどを有効にした場合、そのことが記憶され、次にそのファイルを開いた時にファイルのマクロや ActiveX コントロールなどが有効になっており、セキュリティの警告にわずらわされることがなくなります。またマクロや ActiveX コントロールなどを追加または修正を行った場合もセキュリティの警告は表示されません。ただし、ファイルを信頼した後にファイルを移動した場合は表示されます。ドキュメントを信頼すると、ドキュメントは保護ビューで開かれなくなります。

➤ 保護ビュー

保護ビューでは、セキュリティに関するメッセージを表示せずに、危険性のあるファイルを制限モードで開くかどうかを設定できます。詳細については、「[保護ビュー](#)」をご覧ください。

➤ ファイル制限機能の設定

ファイル制限機能を使用して、古い形式のファイルが、保護ビュー以外で開かれなくすることができます。さらに、特定の種類のファイルを保護ビューで開くかどうかや、編集を許可するかどうかなど、ファイルを開いた時の動作を指定することができます。

詳細はこちらをご覧ください。

➔ 「Office 2010 のファイル制限機能の設定を計画する」

<http://technet.microsoft.com/ja-jp/library/cc179230.aspx>

➤ 信頼されているアプリケーション カタログ

信頼されているアプリケーション カタログ機能を使用して、Office 用アプリの起動を許可するかどうかや、信頼できるアプリケーション カタログのアドレスを登録し、メニューに表示させるかどうかなどを設定できます。

Office 2013 でのセキュリティについてはこちらをご覧ください。

➔ 「Office 2013 でのセキュリティ」

<http://technet.microsoft.com/ja-jp/library/cc179171.aspx>

メッセージ バーの表示について

Office 2003 以前のバージョンの Office では、ダイアログ ボックスでセキュリティ警告を表示していましたが、Office 2007 以降ではメッセージ バーに警告を表示します。警告に気付かず、マクロが動作しなかったり、データが期待どおりに更新されなかったりする場合がありますので、エンド ユーザーにメッセージ バーについて正しく認知してもらう必要があります。

セキュリティ センターで設定した [信頼された場所] からドキュメントを開いた際は、セキュリティ警告のメッセージ バーは表示されません。組織全体で利用するようなマクロを含んだドキュメントに関しては、格納場所を決め、その格納場所をグループ ポリシー等で、セキュリティ センターの「信頼された場所」に設定することを推奨します。

マクロが含まれているファイル等のメッセージ バー

マクロが含まれているファイル等の新しいファイルを開くと、メッセージ バーが表示されます。これは、マクロが含まれているファイル等には、ウイルスが含まれていたり、セキュリティ上の危険性が存在したりする場合があります、コンピューターや組織のネットワークが損害を受ける可能性があるためです。通常は、黄色いバーが表示されますが、赤いバーが表示される場合があります。これは、ファイルの内容が一部欠落しているなどでファイルが壊れている場合や、署名またはハッシュ等を改ざんした場合等に起こる可能性があります。

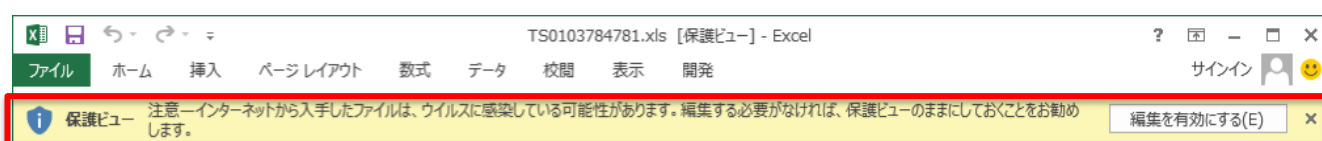


図 10: メッセージ バー (黄色)

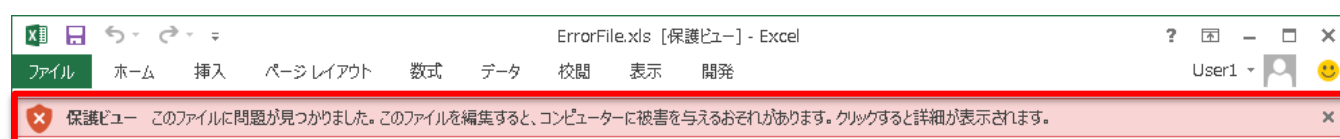


図 11: メッセージ バー (赤色)

次に、信頼されている場所以外からダウンロードされたファイル、および Office ファイル検証で無効なファイル形式が検出された場合について説明します。

➤ 信頼されている場所以外からダウンロードされたファイル

信頼されている場所以外からダウンロードされたファイルとは、インターネット一時フォルダー、Temp フォルダーなど、安全でない場所から開いたファイル、システム管理者が、ネットワーク上の場所またはすべての場所に対して、ドキュメントを信頼する機能を無効にしている場合に開くファイル、または、信頼しようとしているファイルが、".dotx"、".dotm" などの拡張子が付いたテンプレート ファイルであるものです。

マクロの警告について

Office 2003 以前のバージョンの Office ではマクロや ActiveX コントロールを含むドキュメントを開く際に、セキュリティの警告ダイアログ ボックスが表示され、コンテンツを有効化するか、無効化するかをエンド ユーザーが選択できました (但し、Office アプリケーションのセキュリティ レベルの設定によります)。Office 2007 以降の規定ではマクロが無効化され、セキュリティの警告がメッセージ バーに表示されます。メッセージ バーの [コンテンツの有効化] ボタンをクリックして、有効化することができます。一度コンテンツの有効化を行った後、次にそのファイルを開いた場合、セキュリティの警告は表示されず、マクロが有効化されています。

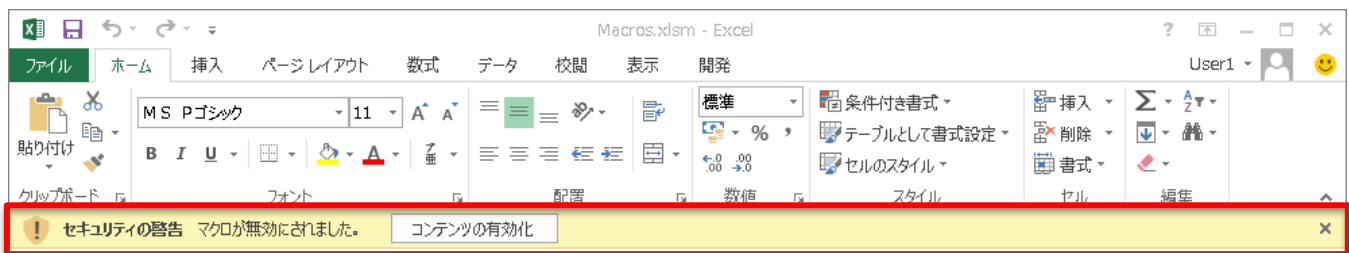


図 12: マクロが無効化された場合にメッセージ バーに表示される [セキュリティの警告]

セキュリティ センターで既定のセキュリティ レベルを下げずに、セキュリティの警告を表示しないようにするためには、有効なデジタル署名でマクロに署名し、その発行元を明示的に信頼するか、そのドキュメントを信頼できる場所へ移動します。

➔ 「Office ドキュメントのマクロを有効または無効にする」

<http://office.microsoft.com/ja-jp/infopath-help/HA010354316.aspx>

データ リンクの警告について

Office 2003 以前のバージョンの Office ではドキュメントを開く際、または明示的にデータ更新を行う際にセキュリティ警告ダイアログ ボックスが表示されます。Office 2007 以降の規定ではすべての接続は無効になり、警告ダイアログ ボックスは表示されません。メッセージ バーに「セキュリティの警告 - データ接続が無効にされました」、または「セキュリティの警告 - リンクの自動更新が無効にされました」というメッセージが表示されます。

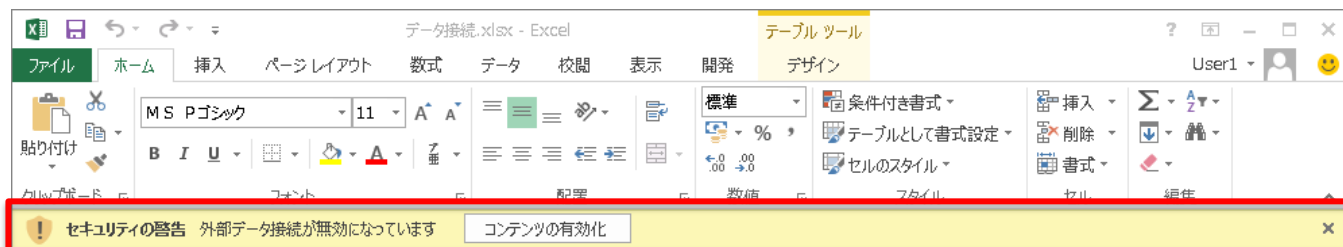


図 13: データ接続が無効化された場合にメッセージ バーに表示される [セキュリティの警告]

この場合、[コンテンツの有効化] ボタンをクリックしなければ、最新の外部データ、または外部データが更新された際の変更はドキュメントに反映されません。[コンテンツの有効化] でコンテンツを有効化すると、データ接続が有効になり更新が反映されます。一度コンテンツの有効化を行った後、次にそのファイルを開いた場合、セキュリティの警告は表示されず、データ接続が有効になっており、更新が反映されています。

データ リンクについての詳細はこちらをご覧ください。

➔ 「メッセージ バーのセキュリティの警告を有効または無効にする」

<http://office.microsoft.com/ja-jp/excel-help/HA010354318.aspx>

➔ 「信頼できる発行元を追加、削除、表示する」

<http://office.microsoft.com/ja-jp/excel-help/HA010354309.aspx>

保護ビュー

保護ビューは、ドキュメント、プレゼンテーション、およびブックをサンドボックス環境で開くことにより、さまざまな種類の問題を引き起こす可能性のある行為を未然に防ぐことができます。サンドボックスは、特定のオペレーティング システム コンポーネントやアプリケーションから分離された、コンピューター メモリの一部または特定のコンピューター プロセスです。このように分離していることで、サンドボックス環境で実行されるプログラムやプロセスは、危険が少ないと見なされます。サンドボックス環境は、多くの場合、コンピューターの状態が不安定になったり異常終了したりする可能性がある新しいアプリケーションやサービスをテストするのに使用されます。サンドボックス環境はまた、アプリケーションやプロセスがコンピューターに悪影響を及ぼすのを防ぐためにも使用されます。

保護ビューでファイルを開くと、そのコンテンツは表示できますが、編集、保存、または印刷することはできません。ActiveX コントロール、アドイン、データベース接続、Visual Basic for Applications (VBA) マクロなどのコンテンツも使用できません。ファイルのコンテンツをコピーして、別のドキュメントに貼り付けることはできます。また、保護ビューでは、ドキュメント、プレゼンテーション、またはブックの署名に使用されているデジタル署名の詳細を表示できません。

Office 2013 の既定で制限され、保護ビューで開かれるファイルの例は、次のとおりです。

- Word 95 のドキュメントおよびテンプレート
- Word 6.0 のドキュメントおよびテンプレート
- Word 5.0 以前のドキュメントおよびテンプレート
- Excel 4.0 のブックおよびスプレッドシート
- Excel 3.0 および 2.0 のスプレッドシート
- Excel 4.0、3.0、および 2.0 のマクロ シートおよびアドイン ファイル

➔ 「Office 2010 アプリケーション セキュリティ」 (Office 2010 時点の情報)

http://blogs.technet.com/office2010_jp/archive/2009/11/12/ms-office-2010.aspx

➔ 「Office 2010 の保護されたビューの設定を計画する」

<http://technet.microsoft.com/ja-jp/library/ee857087.aspx>

マクロのセキュリティの既定レベルについて

マクロを使用すると、日常的に使用しているアプリケーションの機能を拡張でき、反復的な作業の自動化や、面倒な作業を効率化するなど大変便利です。但し、マクロによる処理は自由度が高いため、セキュリティへの配慮が必要です。たとえば、悪意を持って作成されたマクロには、ユーザーの文書やシステムに危害を加えるマクロ コードが含まれている場合があります。

Office クライアント アプリケーションは、悪意のあるマクロから、システムとファイルを保護するためにセキュリティ レベルが設定されています。作成元が不明なマクロを無効にすることもできますし、必要に応じてマクロを有効にすることもできます。

次の表は、バージョンごとのマクロのセキュリティ レベルの既定の設定をまとめたものです。

バージョン	既定のセキュリティ レベル
Office 2013	警告を表示し、すべてのマクロが無効
Office 2010	警告を表示し、すべてのマクロが無効
Office 2007	警告を表示し、すべてのマクロが無効
Office 2003	高：信頼できる作成ものからの署名付きマクロだけが有効

表 6: バージョンごとの既定のセキュリティ レベル

Office 2013 の既定の設定ではマクロは無効にされますが、メッセージ バーにセキュリティの警告が表示され、必要に応じて有効化できます。Office 2003 以前のバージョンから上位のバージョンへ移行する場合は、セキュリティの既定のレベルを確認してお使いください。

マクロを含むドキュメントの格納場所は、セキュリティ センターの [信頼できる場所] で設定することを推奨します。セキュリティ センターの [信頼できる場所] で設定された場所からドキュメントを開く場合は、セキュリティ警告は表示されず、マクロは有効化されます。



Office 2013 のマクロのセキュリティに関する詳細はこちらをご覧ください。

➔ 「セキュリティ センターのオプションと設定を表示する」

<http://office.microsoft.com/ja-jp/word-help/HA010354326.aspx>

➔ 「Office 2010 の VBA マクロのセキュリティ設定を計画する」

<http://technet.microsoft.com/ja-jp/library/ee857085.aspx>

また、管理用テンプレートや Office カスタマイズ ツール (OCT) を利用してユーザーのセキュリティ設定を管理者が一元管理できます。詳細はこちらをご覧ください。

➔ 「Office 2010 のセキュリティを構成する」

<http://technet.microsoft.com/ja-jp/library/ff400327.aspx>

➔ 「Office 2010 で ActiveX コントロールのセキュリティ設定を計画する」

<http://technet.microsoft.com/ja-jp/library/cc179076.aspx>

➔ 「Office 2010 のアドインのセキュリティ設定を計画する」

<http://technet.microsoft.com/ja-jp/library/ee857086.aspx>

➔ 「Office 2013 でのセキュリティ」

<http://technet.microsoft.com/ja-jp/library/cc179171.aspx>

VBA プロジェクトのデジタル署名

デジタル署名とは、安全性を高めるため電子的に暗号化した署名です。Office 2007 以降から VBA プロジェクト全体に対してデジタル署名を追加することができます。デジタル署名によって署名された VBA プロジェクトに対して、1箇所でも項目に変更が行われると、署名済みのデジタル署名は破棄されます。署名後になんらかの変更が加えられていた場合、そのマクロを含むドキュメントを Office アプリケーションで開くと、セキュリティ警告が表示されるため、マクロの改ざんを検出することができます。

マクロのデジタル署名の詳細はこちらをご覧ください。

➔ 「マクロ プロジェクトにデジタル署名を追加する」

<http://office.microsoft.com/ja-jp/word-help/HA010354312.aspx>

Microsoft Backstage ビュー

Microsoft Backstage ビューのセキュリティについて説明します。Microsoft Backstage ビューの [情報] タブには、ファイルへのアクセス許可をはじめ、セキュリティに関する情報が含まれており、設定することができます。たとえば、ファイルのアクセス許可では、ブックまたはシートの保護、ユーザー アクセスの制限、および デジタル署名の追加を設定することができます。また、[オプション] - [セキュリティ センター] では、セキュリティに関する情報を一括で設定することができます。



図 14 : Excel 2013 の Backstage ビュー [情報] - [ブックの保護]

詳細はこちらをご覧ください。

➔ 「Office 2010 の Office ユーザー インターフェイスの概要」 (Office 2010 時点の情報)

[http://technet.microsoft.com/ja-jp/library/ff461922\(office.14\).aspx](http://technet.microsoft.com/ja-jp/library/ff461922(office.14).aspx)

➔ 「Microsoft Office Backstage (パート 3 - [情報] タブ) 」 (Office 2010 時点の情報)

http://blogs.technet.com/b/office2010_jp/archive/2009/10/12/microsoft-office-backstage-3.aspx

セキュリティ以外の詳細については、別途提供されているホワイトペーパー 「Microsoft Office 2013 ファイル フォーマットおよびドキュメント レイアウトの互換性について」を参照してください。

マクロが含まれたファイルが暗号化されている場合の留意点

Office 2007 以降で、マクロが含まれたファイルを Open XML 形式の暗号化ファイルにした場合、マクロ コードも暗号化されます。安全性を保つため、ファイルを開く際にマクロの部分は自動的にウイルス チェックされますが、チェックを行うアンチウイルス製品が Microsoft AntiVirus API に対応していない場合、暗号化されたマクロ コードのチェックを行うことができず、マクロが無効化されます。これは、Office 互換機能パックを通して、Office 2003 などを開く場合も同様です。

Microsoft AntiVirus API に対応したアンチウイルス製品がない場合には、Office 97 - 2003 互換形式で暗号化されたマクロ入りドキュメントを扱うようにしてください。

詳細はこちらをご覧ください。

➔ 「2007 Office プログラムで、暗号化された Office オープン XML ファイルに含まれているマクロが実行されない」 (Office 2007 時点の情報)

<http://support.microsoft.com/?scid=kb%3Bja%3B927150>

ユーザー インターフェイスへの影響

Office 2007 で採用されたユーザー インターフェイスのリボンは Office 2013 でも採用されており、XML を使用して柔軟にカスタマイズが可能となっています。一方でリボンを採用する以前のバージョンの Office で開発した、ユーザー インターフェイスのカスタマイズを含むカスタム アプリケーションを実行する際には注意が必要です。

既存のアドインのサポート

Office 2003 以前のバージョンで開発されたカスタム アプリケーションや Office と統合したソリューションの多くは、ユーザー インターフェイスとして CommandBar オブジェクトを使用しています。Office 2007 以降バージョンでもこのようなマクロは正常に動作します。さらに、ほとんどの場合マクロ コードの変更が不要です。ただし、CommandBar オブジェクトで作成したメニューやツール バーは、すべてリボンの [アドイン] タブ内に表示されます。たとえば、カスタマイズにより以前のメニュー構造に項目が追加される場合、リボン インターフェイスでは [アドイン] タブ内に [メニュー コマンド] グループが作成されます。マクロにより、組み込みツール バーに項目が追加される場合はそのコントロールを含む [ユーザー設定のツール バー] グループが作成されます。アドインにより追加されるツール バーは、[カスタム ツール バー] グループにまとめて表示されます。



図 15 : Excel 2013 の [アドイン] タブ

[アドイン] タブでマウス クリックなどのイベントが発生すると、イベントに対応するマクロ コードが実行されます。Office 2003 以前のバージョンでメニュー項目やツール バー項目の削除や再編成を行っていたアドインは動作しません。

リボンのカスタマイズ

リボンを採用していない以前のバージョンの Office やアプリケーションでは、メニューやツール バーにコマンドを追加して、それらのイベントに対応するコールバック関数を発行していました。リボンでは XML によるカスタマイズや、Visual Studio 2012 を利用したマネージ コードによるカスタマイズ方法以外にも、Office 2010 以降、オプション画面よりカスタマイズする方法が新たに追加されています。

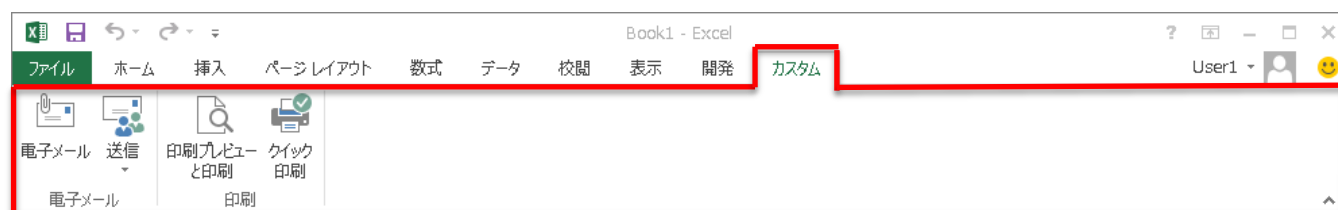


図 16 : Excel 2013 のリボン をカスタマイズしてタブを追加

オプション画面の「リボンのユーザー設定」によるリボンのカスタマイズについては、別途提供されているホワイトペーパー「Microsoft Office 2012 ファイル フォーマットおよびドキュメント レイアウトの互換性について」を参照してください。

カスタム アプリケーションおよびアドインに対する影響や、カスタム アプリケーションにおいて開発者が新しいユーザー インターフェイスの要素を利用したり拡張したりする方法は、こちらをご覧ください。

➔ 「Office 2010 で Office ユーザー インターフェイスをカスタマイズする」 (Office 2010 時点の情報)

<http://technet.microsoft.com/ja-jp/library/ff468686.aspx>

XML によるカスタマイズ

タブやグループに配置するコマンド群を Open XML 形式で定義することで追加できます。任意のテキスト エディターを利用して XML マークアップを作成することにより、特別な開発環境を用意せずにユーザー インターフェイスの拡張が可能です。

```
<?xml version="1.0" encoding="utf-8"?>
<customUI xmlns="http://schemas.microsoft.com/office/2009/07/customui">
  <ribbon startFromScratch="false">
    <tabs>
      <tab id="tab2010" label="ホーム">
        <group id="grp2010" label="ブログ">
          <button id="btn2010" label="ブログホーム"
            imageMso="BlogHomePage" size="large"
            onAction="Macro2010" />
        </group>
      </tab>
    </tabs>
  </ribbon>
</customUI>
```



図 17: XML によるリボンのカスタマイズ実行結果

詳細については、次項「[Office UI カスタマイズについてのリソース](#)」をご覧ください。

Microsoft Office 開発ツール (VSTO) によるカスタマイズ

Visual Studio 2012 を利用して、独自のリボン タブやリボン グループをアプリケーション レベルのアドインに追加することができます。

詳細については、次項「[Office UI カスタマイズについてのリソース](#)」をご覧ください。また、マネージ コードによる開発については、本ドキュメントの「[Office ソリューション開発環境の選択](#)」をご覧ください。

Backstage ビューのカスタマイズ

Office 2007 でのリボン導入に合わせて、RibbonX のスキーマ (Office の ユーザー インターフェイスをプログラムでカスタマイズする際に利用) が公開されました。Office 2010 以降、RibbonX の機能が拡張され、リボンのカスタマイズと同様の方法で、Backstage ビューのカスタマイズが可能です。

```
<?xml version="1.0" encoding="utf-8"?>
<customUI xmlns="http://schemas.microsoft.com/office/2009/07/customui">
  <backstage>
    <tab id="tab2010"
      label="Web">
      <firstColumn>
        <taskGroup id="tskGrp1">
          <category id="category1"
            label="ホーム">
            <task id="task1"
              label="ブログホーム"
              imageMso="BlogHomePage"
              onAction="Macro2010_1" />
          </category>
        </taskGroup>
      </firstColumn>
      <secondColumn>
        <group id="grp1"
          label="ブログ">
          <topItems>
            <button id="button1"
              label="リンク"
              onAction="Macro2010_2" />
          </topItems>
        </group>
      </secondColumn>
    </tab>
  </backstage>
</customUI>
```




図 18: XML による Backstage ビューのカスタマイズ実行結果

カスタム アプリケーションにおいて開発者が新しいユーザー インターフェイスの要素を利用したり拡張したりする方法は、こちらをご覧ください。

➔ 「Office 2010 の Backstage ビューについて (開発者向け)」 (Office 2010 時点の情報)

<http://msdn.microsoft.com/ja-jp/library/ee691833.aspx>



Office UI カスタマイズについてのリソース

Microsoft Developer Network (MSDN) に、リボン拡張機能について詳細に説明しているブログ、および技術に関する記事があります。リボンの包括的な説明については、こちらをご覧ください。

➔ 「Jensen Harris : Office ユーザー インターフェイスのブログ」 (英語、Office 2010 時点の情報)

<http://blogs.msdn.com/jensenh/default.aspx>

タブ、グループ、およびコントロールのリボン拡張機能に関する開発者向け情報については、こちらをご覧ください。

➔ 「Visual Studio 2010 リボンの概要」 (Visual Studio 2010 時点の情報)

[http://msdn.microsoft.com/ja-jp/library/bb386097\(VS.100\).aspx](http://msdn.microsoft.com/ja-jp/library/bb386097(VS.100).aspx)

➔ 「Visual Studio 2010 Office UI のカスタマイズ」

<http://msdn.microsoft.com/ja-jp/library/bf08984t.aspx>

➔ 「Office 2010 で Office ユーザー インターフェイスをカスタマイズする」 (Office 2010 時点の情報)

<http://technet.microsoft.com/ja-jp/library/ff468686.aspx>

マクロ コードの見直しが必要な項目

Office 2003 以前のバージョンで開発したカスタム アプリケーションを Office 2007 以降のバージョンで利用する際に、マクロ コードを見直す必要があるかもしれません。このトピックでは製品機能の変更や追加、それに伴うオブジェクト モデルの変更など、Office 2007 以降のバージョンへマクロ コードを移行する際に影響がある事項を説明します。

Microsoft Office 2013 共通の変更点

Office 2007 以降のバージョンではリボン インターフェイスや新しいオブジェクトなど、新規に追加された機能がある一方で、Office 2003 以前のバージョンから提供されていた機能で、提供されなくなったり、機能制限されたりするものもあります。

図表を SmartArt へ変換すると名前が変わる

Office 2003 以前のバージョンで作成したドキュメントに含まれる、組織図などの図表オブジェクトを、SmartArt に変換することができます。但し、SmartArt に変換した場合は図表オブジェクトの名前が変わるため、それらのオブジェクトの名前を指定して操作するマクロは期待通りに動かなくなります。マクロから操作する可能性がある場合は、これらのオブジェクトを SmartArt に変換しないようにしてください。またデフォルトで名づけられた図表オブジェクトをコピーした場合、図表オブジェクトの名称の後ろに付随している数値のみが変更されます。



図 19: 図表の変換の確認ダイアログ ボックス

次のコードのように名前を指定して図表を操作するマクロは、対象の図表オブジェクトが SmartArt に変換されると、名前が変わるため正常に動作しません。変換した場合は、変換後の図表オブジェクトの名前を調べて指定し直してください。

```
Sub moveDiagram()  
    ActiveSheet.Shapes("Organization Chart 2").IncrementLeft 20  
End Sub
```



図 20: サンプル コードを実行したときのエラー メッセージ

Application.FileSearch の廃止

Application.FileSearch メソッドは Office 2003 以前のバージョンで提供していたファイル検索機能と共に削除されたため、Office 2007 以降のバージョンではこのメソッドの使用はできません。このメソッドを使用しているマクロは実行時にエラーとなります。

次のコードは、Office 2007 以降のバージョンでは失敗します。

```
Sub SearchExcelBook()
    With Application.FileSearch
        .LookIn = "C:\Document"
        .FileType = msoFileTypeExcelWorkbooks
        .Execute
        MsgBox .FoundFiles.Count
    End With
End Sub
```



図 21: サンプル コードを実行したときのエラー メッセージ

回避策など詳細な情報はこちらの技術情報を参考にしてください。

➔ 「Access 2007 で FileSearch 関数を実行しようとする、エラー メッセージ "指定した式に、FileSearch プロパティに対する正しくない参照が含まれます" が表示される」 (Office 2007 時点の情報)

<http://support.microsoft.com/default.aspx?scid=kb;ja;935402>

Chart オブジェクトの変更

Office 2007 より、Chart オブジェクトは OfficeArt オブジェクトを基にして作成されます。これによりマクロから Chart.Pictures コレクションを利用できない場合があるので、Chart.Shapes コレクションを利用するようにしてください。

次のコードは Chart.Shapes コレクションを利用したサンプル コードです。

```
Sub Chart_Object()  
    ActiveSheet.ChartObjects("グラフ 2").Chart.Shapes.AddPicture _  
        Application.GetOpenFilename(), msoFalse, msoCTrue, 10, 10, 50, 50  
End Sub
```

また、オブジェクト モデルの変更以外にも凡例表示やレイアウトに関する動作結果が変更になっているため、Office 2003 以前のバージョンでマクロを動作させた場合と、見た目が異なる結果になる可能性があります。詳細はこちらをご覧ください。

➔ 「グラフ作成での変更点」 (Office 2007 時点の情報)

<http://technet.microsoft.com/ja-jp/library/cc179160.aspx>

マクロのコメント

Office 2003 以前のバージョンでは、コメントと宣言文以外に何も記述されていない VBA コードでも、コメントと宣言文は保持されファイルに保存されました。Office 2007 以降のバージョンでは、コメントと宣言文以外に何も記述されていない VBA コードは保存されません。サブルーチン、または関数を追加することで保存されます。

TextBoxes オブジェクトについて

Office 2007 以降のバージョンの VBA コードでは TextBoxes を含んだコードを実行すると、エラーが発生する場合があります。Office 2007 以降のバージョンの VBA コードでテキスト ボックスにアクセスする場合には、Shapes オブジェクトを利用してください。

次のコードは、Textboxes を利用したサンプル コードです。

```
Sub text_boxes()  
    'Textboxes を使用  
    Sheets(1).TextBoxes.Add 10, 10, 10, 10  
    'Shapes を使用  
    Sheets(1).Shapes.AddTextbox(msoTextOrientationHorizontal, 20, 20, 20,  
20).Select  
End Sub
```

次の機能は別途提供されているホワイトペーパー「Microsoft Office 2013 ファイル フォーマットおよびドキュメント レイアウトの互換性について」を参照してください。

- 特殊記号入力アドイン
- PDF/XPS および ODF フォーマットのサポート

Office 2013 の共通の変更点の詳細については、こちらをご覧ください。

➔ 「Office 2013 での変更点」

[http://technet.microsoft.com/ja-jp/library/cc178954\(v=office.15\).aspx](http://technet.microsoft.com/ja-jp/library/cc178954(v=office.15).aspx)

完全 Open XML 形式への対応

➤ 定数の追加

Office 2013 は、完全 Open XML 形式でのファイル保存に対応しました。それに合わせて、各製品の VBA に完全 OpenXML のファイル形式を表す定数が用意されています。

アプリケーション	定数
Word 2013	wdFormatStrictOpenXMLDocument
Excel 2013	xlOpenXMLStrictWorkbook
PowerPoint 2013	ppSaveAsStrictOpenXMLPresentation

表 7: 完全 Open XML 形式を表すファイル形式の定数

製品別の変更点

Office 2013 では、新しいシェイプやグラフの採用のほかにも、機能の拡張や廃止などにより、オブジェクト モデルに変更点があります。開発者リファレンスの「新機能」の項に、新しく追加されたオブジェクトやメンバーの詳細な説明が記載されています。また、Office 97 以降の各バージョンと、Office 2013 との間の、オブジェクト モデルの比較変更情報も記述されていますので、Office 2013 へ移行する際にお役立てください。

開発者用リファレンスの表示方法は、本ドキュメントのトピック「[開発者用リファレンスの表示方法](#)」をご覧ください。

Word 2010 の変更

➤ Word 2013 のオブジェクト モデル変更

Word 2013 のオブジェクト モデルが変更され、以下のメソッドおよび定数が削除されました。

バージョン	オブジェクト	メソッド/定数	名称
2007	ContentControl	メソッド	ToggleEditPlaceholderText
	WdCalendarType	定数	wdCalendarChina
2003	WdCalendarType	定数	wdCalendarChina
	WdLanguageID	定数	wdBrazilianPortuguese
		定数	wdDzongkhaBhutan
		定数	wdFarsi
		定数	wdFrenchZaire
		定数	wdMacedonian
		定数	wdMacedonian
2002	WdCalendarType	定数	wdCalendarChina
	WdLanguageID	定数	wdBrazilianPortuguese
		定数	wdDzongkhaBhutan
		定数	wdFarsi
		定数	wdFrenchZaire
		定数	wdMacedonian
		定数	wdMacedonian
2000	WdCalendarType	定数	wdCalendarChina
	WdLanguageID	定数	wdBrazilianPortuguese
		定数	wdChineseHongKong
		定数	wdChineseMacao
		定数	wdEnglishTrinidad
		定数	wdFarsi
		定数	wdFrenchZaire
		定数	wdMacedonian
	WdWordDialogTab	定数	wdDialogFilePageSetupTabPaperSize
		定数	wdDialogFilePageSetupTabPaperSource

表 8 : Word 2013 で削除されたオブジェクト モデル

次の機能は別途提供されているホワイトペーパー「Microsoft Office 2013 ファイル フォーマットおよびドキュメント レイアウトの互換性について」を参照してください。

- 個人情報スマート タグの削除
- 自動要約の廃止
- [リサーチ] 作業ウィンドウの削除

Word 2013 の変更点についての詳細は、こちらをご覧ください。

→ 「Word 2013 の新機能」

<http://office.microsoft.com/ja-jp/support/HA102809597.aspx>

Excel 2013 の変更

➤ シングル ドキュメント インターフェイス (SDI) への変更

Excel 2013 より、ウィンドウがマルチ ドキュメント インターフェイス (MDI) から、シングル ドキュメント インターフェイス (SDI) に変更されました。これは、プログラミングに影響します。SDI によって、各ブックは独自のトップレベル アプリ ウィンドウおよび独自の対応するリボンを持ち、同一のプロセス インスタンスに存在することになります。

例として、次のコードを 2 つのブックで順番に実行すると、Office 2010 以前と Office 2013 では各ウィンドウのステータス バーの状態が異なります。

```
Private Sub Workbook_Open()  
  
    'システム日時をステータスバーに表示する  
    Application.StatusBar = CStr(Now())  
  
End Sub
```

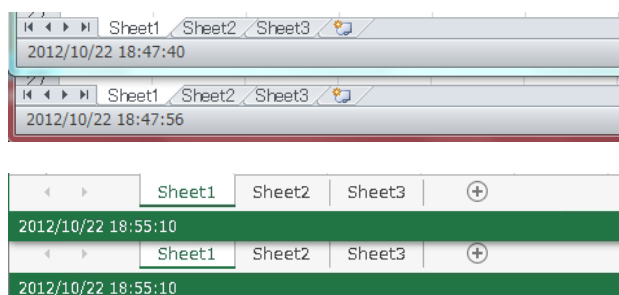


図 22: 同じインスタンスのすべてのブックのステータス バーが更新されるため、2 つのブックで同じ時刻となる

SDI への変更に関連して、オブジェクト モデルに以下の考慮事項があります。

機能	説明	SDI 化の影響
Application.Visible	オブジェクトを表示するか、非表示にするかを決定します。	すべてのウィンドウが非表示になった場合、このプロパティは False になります。 このプロパティを True に設定すると、すべての非表示のウィンドウが表示されます。 シェルからドキュメントを開いた場合、そのウィンドウのみが表示され、このプロパティは True となります。 このプロパティは、ドキュメントの階層に関わらず、すべてを表示/非表示します。
Application.ShowWindowsInTaskbar	True を設定すると、開かれているブックごとに別々の Windows タスクバー ボタンを表示します。 既定値は True です。	この設定は非推奨となりました。 既存のコードから呼ばれた場合、何もしません。
Application.Caption	Excel のウィンドウのタイトル バーに表示される名前です。名前が省略されるか、または Empty 値が設定されていると、"Microsoft Excel" という文字列を取得します。	現在のインスタンスのすべてのウィンドウで更新されます。
Application.Hwnd	Microsoft Excel ウィンドウの最上位レベルのウィンドウ ハンドルを示す長整数型 (Long) の値を取得します。	アクティブウィンドウのハンドルが返されます。
Application.FormulaBarHeight	アクティブ ウィンドウに関連付けられた数式バーの高さを読み書きします。	現在のインスタンスのすべてのウィンドウではなく、現在アクティブなブックのウィンドウに作用します。
Application.DisplayFormulaBar	True を設定すると、数式バーを表示します。	現在のインスタンスのすべてのウィンドウ上で作用します。
VBA StatusBar Updates	ステータス バーを VBA により更新します。	同じインスタンスのすべてのブックのステータスバーが更新されます。
Workbook.Protect (Password, Structure, Windows)	ワークブックを変更できないよう保護します。	保護対象の「ウィンドウ」について、値の設定内容に関わらず保護は有効になりません。 True を設定しても、何も起こりません。

表 9 : SDI 化の影響を受けるオブジェクト モデル

➤ 図形のオブジェクトの既定名の変更

Office 2003 以前のバージョンの Excel、および PowerPoint で図形オブジェクトをマクロ コードから操作する際に、各オブジェクトに明示的に名前を付けていない場合、英語名 (内部名) と日本語名 (表示名) の両方で操作ができました。たとえば、AutoShape の「四角形」をドキュメントに挿入すると、オブジェクトの名前が “四角形 1” と表示され、マクロから “Rectangle 1” と “四角形 1” のどちらでもそのオブジェクトを操作できました。

Excel 2013 では「四角形」の日本語名 (表示名) が「正方形/長方形 1」に変更されています。Excel 2013 で新規に図形を挿入した場合、既定の名前を使ってオブジェクトを操作するには “Rectangle 1” あるいは “正方形/長方形 1” のどちらかを指定する必要があります。どちらの場合でも、明示的にオブジェクトに名前を付けている場合は問題ありません。

また、オブジェクトの名前「Rectangle 1」などの、数値の採番方法に関して、Excel 2003 以前のバージョンの場合は図形を消しても、これまでの最大値に 1 を加えたものが名前となります。Excel 2007 以降のバージョンの場合は、図形を消して保存すると 1 から始まります。

次のコードは Excel 2007 以降のバージョンで実行すると、赤枠の箇所です失敗します。

```
Sub putNewRect1()  
    ' 四角形オブジェクトの追加  
    ActiveSheet.Shapes.AddShape msoShapeRectangle, 10, 10, 50, 50  
    ' 図形オブジェクトの内部名を指定して操作  
    ActiveSheet.Shapes("四角形 1").Top = 20  
    ' 図形オブジェクトの名前を指定して操作  
    ActiveSheet.Shapes("Rectangle 1").Left = 20  
End Sub
```



図 23: 図形オブジェクトの規定の名前が変わったことによって起こるエラー メッセージ

次のコードのように、明示的にオブジェクト名を命名することを推奨します。

```
Sub putNewRect2()  
    ' 四角形オブジェクトの追加  
    Dim NewRect As Shape  
    Set NewRect _  
        = ActiveSheet.Shapes.AddShape(msoShapeRectangle, 10, 10, 50, 50)  
    ' 追加した図形オブジェクトに名前を設定  
    NewRect.Name = "NewRect"  
    ' 図形オブジェクトの名前を指定して操作  
    ActiveSheet.Shapes("NewRect").Top = 20  
    ActiveSheet.Shapes("NewRect").Left = 20  
End Sub
```

- Excel 2007 以降のバージョンのシートでセル数をカウントするとオーバーフローになる
Excel 2007 以降のバージョンのワークシートでは、取り扱えるセルの数が大幅に増えました。1,048,576 行、16,384 列に拡張されたため、Long 型の範囲を超える個数のセルを含む Range オブジェクトで Count メソッドを使用した場合、エラーになります

以下のコードはオーバーフローを起こします。

```
Sub countCell()  
    MsgBox ActiveSheet.Cells.Count  
End Sub
```



図 24: サンプル コードを実行したときのエラー メッセージ

Excel 2007 以降のバージョンで拡張されたセル数を扱うには次のコードのように Variant 型の Range.CountLarge の使用をご検討ください。

```
Sub countCell()  
    MsgBox ActiveSheet.Cells.CountLarge  
End Sub
```

また、直接 Range.Count を使わない場合でも、全行や全列に対して処理を行う場合には注意が必要です。次のコードはオーバーフローします。

```
Sub countCells()
    Dim cnt As Long
    ' 全行、全列に対して処理を行う
    For Each r In ActiveSheet.Rows
        For Each c In ActiveSheet.Columns
            cnt = cnt + 1
        Next
    Next
    MsgBox cnt
End Sub
```

オーバーフローの問題以外にも、全行や全列を対象とした処理は以前のバージョンの Office で実行する場合よりも長時間かかる可能性がありますので、留意する必要があります。

- Excel 2007 以降のバージョンのワークシートを、Excel 2003 以前のバージョンのワークブックへ移動、またはコピーをすると、実行時エラーが起こる

マクロにより新規にワークブックを作成すると、Excel 形式のワークブック (.xlsx/.xlsb/.xlsm) が作成されます。また、Excel 2007 以降のバージョンで Excel 97-2003 形式のファイル (.xls) を開いた場合、互換モードで開かれます。マクロにより新規にワークブックを作成し、新規ワークブックのワークシートを互換モードのワークブックに移動、またはコピーするとエラーになります。

互換モードのワークブックでは次のコードはエラーになります。

```
Sub insNewSheet()
    Dim shBase As Workbook, shNew As Workbook
    ' 互換モードで Workbook を開き、新規 Workbook を開く
    Set shBase =
        Workbooks.Open(Filename:="C:\Excel2k3book.xls")
    Set shNew = Workbooks.Add
    ' 互換モード Workbook に新規 Workbook のシートを移動
    shNew.Sheets.Move before:=shBase.Sheets(1)
End Sub
```



図 25: サンプル コードを実行したときのエラー メッセージ

このような場合は、移動先の Excel 97-2003 形式 (.xls) を Excel 形式 (.xlsx) に変更し、移動元のマクロを含むワークブックを Excel マクロ有効ワークブック形式 (.xlsb や .xlsm) に変換してから実行してください。

- Excel 97-2003 形式 (.xls) のセルの範囲を超えるコピーで、警告メッセージが表示される
Excel 形式 (.xlsx) のファイルから行や列をコピーして、Excel 97-2003 形式 (.xls) のファイルに貼り付ける際に、警告メッセージが表示されることがあります。これは、Excel 97-2003 形式 (.xls) の Excel で扱える範囲外にあるセルにデータが存在することで起こります。コピー元である Excel 形式 (.xlsx) のファイルでは、シート サイズは 16,384 列 x 1,048,576 行ですが、コピー先となる Excel 97-2003 形式 (.xls) のファイルでは、256 列 x 65,536 行までしか扱えません。そのため、動作が制御され、警告メッセージを表示しています。また、この現象は、コピーの対象でない行や列で、扱える範囲外のセルにデータが存在している場合でも発生します。行や列単位の選択ではなく、セル単位で選択してからコピー、貼り付けします。コピー先ファイルを一旦 Excel 形式 (.xlsx) で保存した上で、Excel 97-2003 形式 (.xls) に保存し直すようにしてください。

- AutoFilterMode プロパティについて
Excel 2003 では、AutoFilterMode プロパティを確認するマクロを作成して、リスト (Excel 2007 以降ではテーブルと呼ばれる) 内に選択があった場合に、そのリストでオートフィルターがオンになっているかどうかを判断できました。Excel 2007 以降では、AutoFilterMode プロパティは、ワークシートのオートフィルターでのみ使用でき、テーブルの構成要素であるオートフィルターでは使用できません。これは、Excel 2007 以降のプロパティが各テーブルに独自の AutoFilter オブジェクトを指定し、そのオブジェクトがテーブルの使用を通じてワークシートごとに複数のオートフィルターを有効にするためです。

リスト (テーブル) の AutoFilterMode プロパティを確認するマクロを含む Excel 2003 ブックを開いた場合、AutoFilterMode プロパティは正しく動作しない可能性があります。

次のコードは、Excel 2007 以降のバージョンで実行した場合と、Excel 2003 で実行した場合で、異なる結果を表示することがあります。

```
Sub insNewSheet()  
    MsgBox "AutoFilterMode is " & ActiveSheet.AutoFilterMode  
End Sub
```

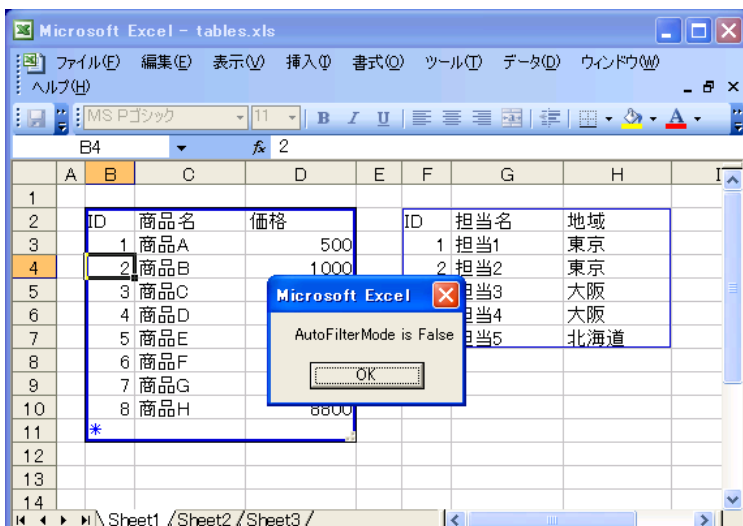


図 26 : Excel 2003 で、オートフィルターが指定されていないテーブルを選択して、マクロを実行した結果

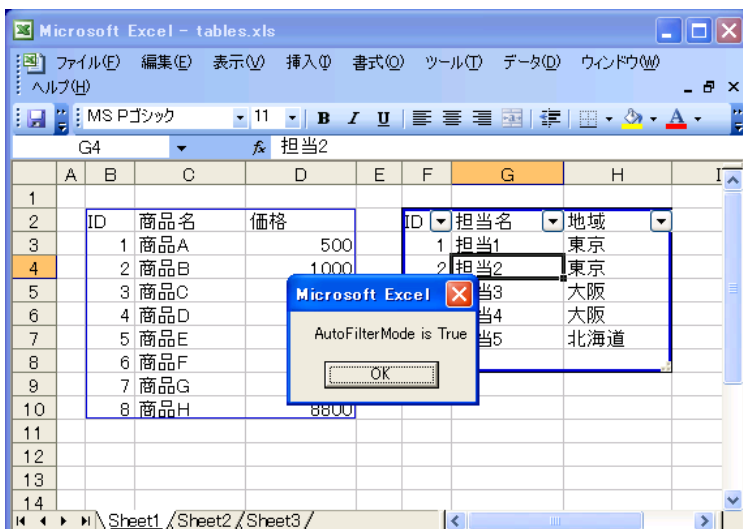


図 27 : Excel 2003 で、オートフィルターが指定されているテーブルを選択して、マクロを実行した結果

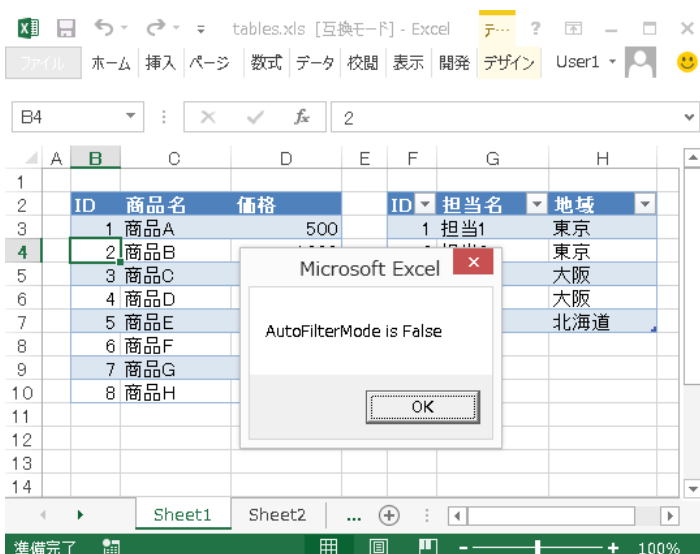


図 28 : Excel 2013 で、オートフィルターが指定されていないテーブルを選択して、マクロを実行した結果

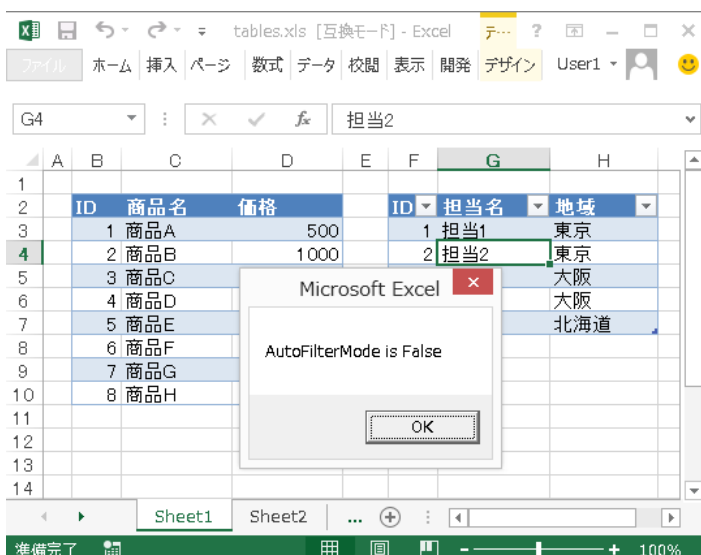


図 29 : Excel 2013 で、オートフィルターが指定されているテーブルを選択して、マクロを実行した結果

➤ ListObject.InsertRowRange (について)

Excel 2003 では、リストの最下行に新しいレコードを追加するための特殊な行が存在しました。Excel 2007 以降では、レコードの追加方式が変わったため特殊な行は存在しません。このため、マクロで

ListObject.InsertRowRange を使用して行を追加している場合、Excel 2003 では戻り値として挿入された行の範囲オブジェクトが返されますが、Excel 2007 以降の場合は Nothing が返されます。但し、リストのデータをすべて削除した場合などは、Nothing にならないことがありますので注意してください。

データのあるリストを選択して、次のコードを実行した場合、Excel 2007 以降ではエラーが発生します。

```
Sub getNewRow()
    Dim objNewRange As Range
    Set objNewRange = ActiveSheet.ListObjects(1).InsertRowRange
    objNewRange.Cells(1).Value = "It's a new row."
End Sub
```

➤ オブジェクト モデル変更

Excel 2007 以降オブジェクト モデルが変更され、以下のプロパティおよび定数が削除されました。

バージョン	オブジェクト	プロパティ/定数	名称
2007	AboveAverage	プロパティ	FormatRow
	ColorScale	プロパティ	FormatRow
	FormatCondition	プロパティ	FormatRow
	WdLanguageID	プロパティ	FormatRow
	IconSetCondition	プロパティ	FormatRow
	Top10	プロパティ	FormatRow
	UniqueValues	プロパティ	FormatRow
	XlRobustSave	定数	xlMinimalSave
			xlNormalSave
			xlSafeSave
2013	Application	プロパティ	EnableAnimations (無効化)

表 10 : Excel 2007 以降削除されたオブジェクトモデル

➤ Excel 5.0/95 形式のマクロについて

Excel 2013 で Excel 5.0/95 形式のマクロを含むブックを開いた際に、次のメッセージが表示されることがあります。

「このファイルの VBA プロジェクトを開くためには、現在インストールされていないコンポーネントが必要です。詳細については、office.com で「VBA コンバーター」を検索してください。」

これは、Excel 2013 では Excel 5.0/95 形式のマクロをサポートしていないためです。サポートセンターより VBA コンバーターパックを入手し、PC に適用することで Excel 5.0/95 形式のマクロを実行することが可能です。

➔ 「Error message when you try to open an Excel 5.0/95 workbook that contains a VBA module sheet in Excel 2007: "Opening the VBA project in this file requires a component that is not currently installed"」 (英語) (Office 2007 時点の情報)

<http://support.microsoft.com/?scid=kb%3Ben-us%3B926430&x=6&y=16>

➤ ワークシート関数の変更

ワークシート関数は確率分布の値を返す統計関数や配列に関わる関数などを中心に追加されています。

詳細はこちらをご覧ください。

➔ ワークシート関数一覧 (機能別) (Office 2010 時点の情報)

<http://office.microsoft.com/ja-jp/excel-help/HP010342656.aspx>

Excel 2013 の変更点についての詳細は、こちらをご覧ください。

「Excel 2013 の新機能」

[http://msdn.microsoft.com/ja-jp/library/office/ff837594\(v=office.15\).aspx](http://msdn.microsoft.com/ja-jp/library/office/ff837594(v=office.15).aspx)

PowerPoint 2013 の変更

➤ 図形のオブジェクトの既定名の変更

PowerPoint 2003 以前のバージョンでは図形オブジェクトをマクロ コードから操作する際に、AutoShape の「四角形」をドキュメントに挿入すると、オブジェクトの名前が "Rectangle 1" と表示されますが、マクロから "Rectangle 1" と "四角形 1" のどちらでもそのオブジェクトを操作できました。

PowerPoint 2013 では図形オブジェクトをマクロ コードから操作する際に、AutoShape の「四角形」をドキュメントに挿入すると、オブジェクトの名前が "Rectangle 1" と表示されますが、マクロから "Rectangle 1" と "正方形/長方形 1" のどちらでもそのオブジェクトを操作できます。

次のコードは、PowerPoint 2007 以降のバージョンで実行すると、赤枠の箇所ですべて失敗します。

```
Sub putNewRect1()  
    ' 四角形オブジェクトの追加  
    ActiveWindow.Selection.SlideRange.Shapes.AddShape _  
        (msoShapeRectangle, 10, 10, 50, 50).Select  
    ' 図形オブジェクトの内部名を指定して操作  
    ActiveWindow.Selection.SlideRange.Shapes("四角形 1").Top = 20  
    ' 図形オブジェクトの名前を指定して操作  
    ActiveWindow.Selection.SlideRange.Shapes("Rectangle 1").Top = 20  
End Sub
```



図 30: 図形オブジェクトの規定の名前が変わったことによって起こるエラー メッセージ

次のコードのように、明示的にオブジェクト名を命名することを推奨します。

```
Sub putNewRect2()  
    ' 四角形オブジェクトの追加  
    Dim NewRect As Shape  
    Set NewRect = ActivePresentation.Slides(1).Shapes.AddShape _  
        (msoShapeRectangle, 10, 10, 50, 50)  
    ' 追加した図形オブジェクトに名前を設定  
    NewRect.Name = "NewRect"  
    ' 図形オブジェクトの名前を指定して操作  
    ActivePresentation.Slides(1).Shapes("NewRect").Top = 20  
    ActivePresentation.Slides(1).Shapes("NewRect").Left = 20  
End Sub
```

➤ 最大サウンド ファイル サイズの廃止

最大サウンド ファイル サイズの指定機能については別途提供されているホワイトペーパー「Microsoft Office 2013 ファイル フォーマットおよびドキュメント レイアウトの互換性について」を参照してください。

PowerPoint 2013 の変更点についての詳細は、こちらをご覧ください。

➔ 「PowerPoint 2013 の新機能」

<http://office.microsoft.com/ja-jp/powerpoint-help/HA102809628.aspx>

Access 2013 の変更

➤ データ中心アプリケーション

Access 2013 には、データ中心 Web アプリケーションを作成できるようにすることだけを目的として設計された新しい機能が組み込まれています。Access Services は、Microsoft SQL Server 2008 を使用してデータを格納します。SQL Server を使用してデータを格納すると、管理性と拡張性が向上します。

➤ 他のテクノロジーとの統合

標準のテクノロジーを使用することにより、Access 2013 では Access アプリケーションのカスタマイズに使用できる開発者向けツールの数が大幅に増えています。SQL Server をデータ ストアとして使用したため、Access 2013 では Access アプリケーションの管理性と拡張性が飛躍的に向上しています。Office 365 および SQL Azure との互換性により、Access アプリケーションの範囲が大きく広がっています。

➤ RunCommand について

Access 2007 以降のバージョンでは DoCmd.RunCommand acCmdImport オブジェクト モデルの呼び出しと RunCommand マクロのインポートはサポートされなくなりました。この呼び出しを利用すると、エラーが発生します。

次のコードは、Access 2007 以降のバージョンでは失敗します。

```
Sub RunCommand ()
    DoCmd.RunCommand acCmdImport
End Sub
```

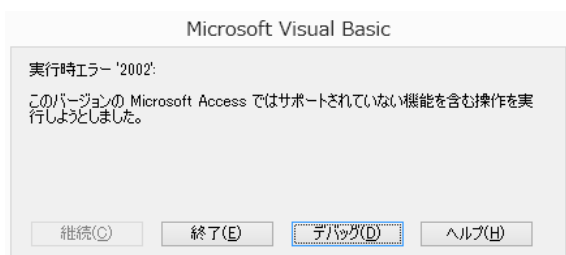


図 31: サンプル コードを実行したときのエラー メッセージ

acCmdImport の代わりに、ImportAttach コマンドを利用するようにしてください。(表参照)

Name	Value
acCmdImportAttachAccess	544
acCmdImportAttachdBase	552
acCmdImportAttachExcel	545
acCmdImportAttachHTML	550
acCmdImportAttachLotus	554
acCmdImportAttachODBC	549
acCmdImportAttachOutlook	551
acCmdImportAttachParadox	553
acCmdImportAttachSharePointList	547
acCmdImportAttachText	546
acCmdImportAttachXML	548

表 11: ImportAttach コマンド一覧

➤ サンドボックス モードについて

Access 2003 から追加されたサンドボックス機能が有効な場合、安全と見なされないマクロやクエリーの実行がブロックされます。Access 2013 では既定でサンドボックス モードが有効なため、以前のバージョンの Access で作成されたデータベースを Access 2013 で開いた場合に、マクロの一部が動作しない場合があります。サンドボックス モードは、レジストリの設定を変更することで、無効にすることができます。

サンドボックス モードの説明や無効にする方法については、こちらをご覧ください。

➔ 「Access 2010 でサンドボックス モードを使用する」(Office 2010 時点の情報)

<http://office.microsoft.com/ja-jp/access-help/HA010342092.aspx>

➤ [フィールドの追加] 作業ウィンドウについて

Access 2010 からは、[フィールドの追加] 作業ウィンドウがなくなり、代わりに [Data Type] ギャラリーが登場しました。[Data Type] ギャラリーのエントリ ポイントは [フィールドの追加] 作業ウィンドウに似ていますが、よりわかりやすくなっています。[フィールドの追加] 作業ウィンドウでは単一のフィールド テンプレートしか選択できませんでしたが、[Data Type] ギャラリーでは複数のフィールド テンプレートを選択できるため、機能性がさらに高まりました。[フィールドの追加] 作業ウィンドウを呼び出すマクロは失敗し、その際に通知メッセージが表示されることもありません。

➤ [オートフォーマット] について

[オートフォーマット] は、Access 2010 以降のフォームおよびレポート レイアウト ビューのリボン ユーザー インターフェイスから削除され、代わりに Office テーマが使用されるようになりました。[オートフォーマット] コマンドは従来と同じツール バー コントロール ID (TCID) で引き続き使用できるため、クイックアクセス ツール バーやカスタム リボンに追加できます。Office テーマはカスタマイズや拡張ができ、Office.com からのダウンロードや他のユーザーと電子メールで共有することもできるので、フォームおよびレポートの書式オプションが従来よりも充実します。

Access 2013 の変更点についての詳細は、こちらをご覧ください。

➔ 「Access 2013 の新機能」

[http://msdn.microsoft.com/ja-jp/library/fp179914\(v=office.15\).aspx](http://msdn.microsoft.com/ja-jp/library/fp179914(v=office.15).aspx)

Outlook 2013 の変更

➤ オブジェクト モデル拡張

Outlook 2013 から オブジェクト モデルが拡張され、以下のメンバーが追加されました。

オブジェクト	プロパティ/メソッド/イベント	名称
AppointmentItem	イベント	ReadComplete
ContactItem	メソッド	ShowCheckAddressDialog
		ShowCheckFullNameDialog
		ReadComplete
DistListItem	イベント	ReadComplete
DocumentItem	イベント	ReadComplete
Explorer	プロパティ	ActiveInlineResponse
		ActiveInlineResponseWordEditor
	イベント	InlineResponse
		InlineResponseClose
JournalItem	イベント	ReadComplete

オブジェクト	プロパティ/メソッド/イベント	名称
MailItem	イベント	ReadComplete
MeetingItem	イベント	ReadComplete
PostItem	イベント	ReadComplete
RemoteItem	イベント	ReadComplete
ReportItem	イベント	ReadComplete
SharingItem	イベント	ReadComplete
TaskItem	イベント	ReadComplete
TaskRequestAcceptItem	イベント	ReadComplete
TaskRequestDeclineItem	イベント	ReadComplete
TaskRequestItem	イベント	ReadComplete
TaskRequestUpdateItem	イベント	ReadComplete

表 12 : Outlook 2013 から拡張されたオブジェクトモデル

➤ オブジェクト モデル追加

Outlook 2013 から、以下のオブジェクトが追加されました。

オブジェクト	プロパティ/メソッド/イベント	名称
PeopleView	プロパティ	Application
	プロパティ	Class
	プロパティ	Filter
	プロパティ	Language
	プロパティ	LockUserChanges
	プロパティ	Name
	プロパティ	Parent
	プロパティ	SaveOption
	プロパティ	Session
	プロパティ	SortFields
	プロパティ	Standard
	プロパティ	ViewType
	プロパティ	XML
	メソッド	Apply
	メソッド	Copy
	メソッド	Delete
	メソッド	GoToDate
	メソッド	Reset
	メソッド	Save

表 13 : Outlook 2013 から追加されたオブジェクト

➤ オブジェクト モデル変更

Outlook 2013 から、オブジェクト モデルが変更され、以下のメンバーが削除されました。

オブジェクト	プロパティ/メソッド/イベント	名称
AppointmentItem	プロパティ	Links
ContactItem	プロパティ	Links
DistListItem	プロパティ	Links
DocumentItem	プロパティ	Links
Exception	プロパティ	Links
JournalItem	プロパティ	Links
Link	プロパティ	ApplicationClass
		Item
		Name
		Parent
		Session
		Type
Links	プロパティ	ApplicationClass
		Count
		Parent
		Session
	メソッド	AddItem
		Remove
MailItem	プロパティ	Links
MeetingItem	プロパティ	Links
MobileItem	プロパティ	Actions
		Application
		Attachments
		BillingInformation
		Body
		Categories
		Class
		Companies
		ConversationIndex
		ConversationTopic
		CreationTime
		Count
		EntryID
		FormDescription
		GetInspector
		HTMLBody
		Importance
		ItemProperties
		LastModificationTime
		MessageClass
		Mileage

オブジェクト	プロパティ/メソッド/イベント	名称
		MobileFormat
		NoAging
		OutlookInternalVersion
		OutlookVersion
		Parent
		PropertyAccessor
		ReceivedByEntryID
		ReceivedByName
		ReceivedTime
		Recipients
		ReplyRecipientNames
		ReplyRecipients
		Saved
		SenderEmailAddress
		SenderEmailType
		SenderName
		SendUsingAccount
		Sensitivity
		Sent
		SentOn
		Session
		Size
		SMILBody
		Subject
		Submitted
		To
		UnRead
		UserProperties
	メソッド	Close
		Copy
		Delete
		Display
		Forward
		Move
		Reply
		ReplyAll
SaveSaveAs		
Send		
イベント	AttachmentAdd	
	AttachmentReadAttachmentRemove	
	BeforeAttachmentAdd	
	BeforeAttachmentPreview	
	BeforeAttachmentRead	
	BeforeAttachmentSave	
	BeforeAttachmentWriteToTempFile	

オブジェクト	プロパティ/メソッド/イベント	名称
		BeforeAutoSave
		BeforeCheckNames
		BeforeDelete
		Close
		CustomAction
		CustomPropertyChange event
		Forward
		Open
		PropertyChange
		Read
		Reply
		ReplyAll
		Send
		UnloadWrite
NotelItem	プロパティ	Links
PostItem	プロパティ	Links
RemotelItem	プロパティ	Links
ReportItem	プロパティ	Links
TaskItem	プロパティ	Links
TaskRequestAcceptItem	プロパティ	Links
TaskRequestDeclineItem	プロパティ	Links
TaskRequestItem	プロパティ	Links
TaskRequestUpdateItem	プロパティ	Links

表 14 : Outlook 2013 から削除されたオブジェクト モデル

➤ Office モバイル サービスのサポート中止

Outlook 2013 では Office モバイル サービス (OMS) はサポートされなくなります。SharePoint では引き続きサポートされます。

➤ ユーザー データグラム プロトコル (UDP) の機能削除

Outlook 2010 以前で新しいメールの通知やフォルダーの更新で使用されていた UDP は削除され、非同期通知方式に置き換えられました。

➤ アドインの無効化

Outlook 2010 のアドイン弾力性が拡張され、Outlook 2013 ではアドインの起動、シャットダウン、フォルダー切り替え、アイテムのオープン、反復タイミングなどのアドイン パフォーマンス カウンターが監視されます。たとえば、平均起動時間が指定されている時間を超えると、Outlook はアドインを無効にして、アドインが無効にされたことをユーザーに対して通知します。ユーザーはアドインを無効化させないよう設定することができ、その場合 Outlook はアドインがパフォーマンスしきい値を超えてもアドインを無効にしません。システム管理者の管理レベルも新しく拡張され、グループ ポリシーを使用して無効にするアドインを指定できます。

アドインを無効化する既定の基準時間（ミリ秒）

- 起動：1,000
- 終了：500
- フォルダーの切り替え：500
- アイテムのオープン：500
- 呼び出し頻度：1,000

Outlook 2013 の変更点についての詳細は、こちらをご覧ください。

➔ 「Outlook 2013 の新機能」

[http://technet.microsoft.com/ja-jp/library/cc179110\(v=office.15\).aspx](http://technet.microsoft.com/ja-jp/library/cc179110(v=office.15).aspx)

Office Web Apps の変更

➤ クラシック認証モードからクレームベース認証モードへ

Office Web Apps の表示および編集の機能は、クラシック認証モードを使用する SharePoint Web アプリケーションでは動作しません。クラシック認証モードを使用する SharePoint 2010 Web アプリケーションを SharePoint 2013 へ移行する場合、Office Web Apps が動作するように構成するには、移行した SharePoint Web アプリケーションをクレームベース認証に変換する必要があります。

また、既存の SharePoint 2010 にてカスタム ソリューションを構築している場合は、SharePoint 2013 への移行にあたって、ソリューションがクレームベース認証モードで動作するよう十分に検証を行うことを推奨します。



Office 展開に関連するツールについて

Office テレメトリ ダッシュボード

Office 2013 では、展開時の互換性検証を支援するツールとして、Office テレメトリ ダッシュボード が新たに提供されます。このツールは、Office 2010 の互換性ツールである Office Migration Planning Manager (OMPM)、Office Code Compatibility Inspector (OCCI)、および Office Environment Assessment Tool (OEAT) に代わるものです。

Office テレメトリ ダッシュボードを使用すると、移行計画および互換性評価に要する時間の削減が可能になり、Office 展開全体の時間を短縮できます。また、移行後もモニタリングを続けることにより、将来にわたって IT 運用コストの削減、その次の Office への移行費用の削減も期待できます。

Office テレメトリ ダッシュボードについての詳細は、こちらをご覧ください。

➔ 「Office テレメトリ ダッシュボードを展開する」

[http://technet.microsoft.com/ja-jp/library/jj219431\(v=office.15\).aspx](http://technet.microsoft.com/ja-jp/library/jj219431(v=office.15).aspx)

Office テレメトリ ダッシュボードによる互換性検証支援

Office テレメトリダッシュボードは、Office ファイルやアドインの利用状況と互換性の問題を可視化するためのツールです。このツールを利用することで、以前のバージョンで動作していたファイルやアドインが最新の Office に移行した場合に問題なく動作するか検証することができます。また Office テレメトリ ダッシュボードは Office 2013 に標準で含まれているため、実際の Office の利用実態に基づいた検証を進めることができます。

さらに、本ツールの開発は日本に拠点を持つ開発チームが担当しており、日本のお客様のフィードバックを強く反映したツールとなっています。

クイック実行による新旧バージョンの共存

Office 2013 では、クイック実行と呼ばれる新しい展開方法が登場します。クイック実行は、Office 365 上で仮想化された Office をストリーミング配信し、Office のフル機能をインストールする方法です。クイック実行では、現在使用している Office をアンインストールせず、最新の Office をインストールでき、1 台のローカルコンピュータ内で新旧 Office バージョンを共存利用することができます。これにより、以前のバージョンの Office でしか動作しないアドインを含む Office ファイルがある場合は、以前のバージョンで利用し、それ以外の作業は、最新バージョンの Office を利用し、快適な環境で作業を進めることができます。

このことから、Office テレメトリ ダッシュボードの互換性検証支援機能を併用し、必要最小限のファイルの互換性検証を完了させた時点で、Office 2013 の展開が開始できます。



図 32: クイック実行の仕組み

まとめ

本ドキュメントでは、Office 2010 から採用された VBA7 や、Office 2013 で改善されたセキュリティ機能などの影響による互換性の注意事項、また、Office 2007 以降のバージョンで同様に注意いただきたい ユーザー インターフェイス、グラフィックエンジン、ファイルフォーマットに関するポイントも説明してきました。

また、現在判明しているマクロの互換性の問題点や、それに関連する Office 2013 の機能変更点についても説明しています。

一部コードの修正が必要な注意事項があるものの、環境面を工夫することで、運用が楽になる面もございます。以前のバージョンで開発したカスタム アプリケーションを Office 2013 で利用する際などに本書をご活用いただき、洗練された多彩な描画表現、編集作業の効率化、堅牢なセキュリティなどの Office 2013 の新機能を、ビジネスにおいて十分にご活用ください。

その他のトピック

Office Web Components (OWC) の提供

OWC は、Office 2007 のクライアント スイートからは削除され、Microsoft Office Project Server 2007 のみに付属となりました。Office 2007 では、Microsoft Office SharePoint Server 2007 に付属する Excel Services コンポーネントにより OWC で提供していた一部の機能が置き換えられました。これにより、OWC で実現されていたスプレッドシート、グラフ、ピボットテーブル、またはデータ ソースの各コントロールの機能を Web ページ内で使用するソリューションや、この ActiveX コントロールをコンピューターにインストールする必要のあるクライアント アプリケーションが影響を受けます。

Excel 2003、Access 2003、または Microsoft Office FrontPage 2003 Edition を使用して作成された OWC が必要な Web ページや OWC を利用したカスタム ソリューションを使用している場合は、OWC を Microsoft ダウンロード センターより入手してください。

なお、Access 2013 では、OWC のサポートが終了しました。そのため、以前のバージョンの Access で作成した OWC のピボットテーブル、ピボットグラフはデータシート ビュー、フォーム ビュー、レイアウト ビュー、デザインビューでは開くことができますが、ピボット ドロップ エリアやコンポーネントは表示されません。ピボットテーブルやピボットグラフでのみ開くよう設定されていた場合は、標準フォームで開きます。また、Access 2013 の VBA では OWC を認識せず、OWCXX.dll を参照している場合は、エラーが発生します。

OWC はダウンロード センターより入手できます。詳細はこちらをご覧ください。

➔ 「Office 2003 アドイン : Office Web Components」

<http://www.microsoft.com/downloads/details.aspx?FamilyID=7287252c-402e-4f72-97a5-e0fd290d4b76&DisplayLang=ja>

また、Office XP 用および Office 2007 用の OWC サービスパックがリリースされています。詳細はこちらをご覧ください。

➔ 「Office XP ツール : Web Components」

<http://www.microsoft.com/downloads/details.aspx?FamilyID=982b0359-0a86-4fb2-a7ee-5f3a499515dd&displaylang=ja>

➔ 「2007 Office system 用 Office 2003 Web Components Service Pack 1 (について)」

<http://support.microsoft.com/kb/937961/ja>

➔ 「2007 Microsoft Office system 用 Microsoft Office 2003 Web Components Service Pack 1 (SP1)」

<http://www.microsoft.com/downloads/details.aspx?displaylang=ja&FamilyID=c815dffa-d5f3-4b71-bf46-13721bd44682>

Office ソリューション開発環境

長い間、Microsoft Office を利用したソリューションは、VBA により構築されてきました。そのため VBA は、個人ユーザーや企業ユーザーから信頼され、かつ有用な開発環境として評価されてきました。しかし、最近になって多くの開発者から、ソリューションの諸要求を満たすには VBA ではやや力不足との意見も聞かれるようになりました。そのような要求を満たすため、VBA から、より高機能な開発環境である Visual Studio 2012 への移行を、1つの選択肢としてお勧めします。

Visual Studio 2012 では、Office 2013 の各種アプリケーション上で動作するアセンブリを作成するためのテンプレートを提供しています。これによってドキュメント ベースのソリューションだけでなく、テンプレート ベースのソリューションや、アプリケーション レベルのソリューションとして展開できるテンプレート (グローバル テンプレート、またはアドイン) を作成できます。その際、VB .NET や、C# を使用して、共通言語ランタイム (CLR: Common Language Runtime) の下で実行するアセンブリを作成できます。この CLR 上で実行されるコードは、マネージ コードと呼ばれます。

CLR は、ユーザーの作成したプログラムが、領域外のメモリにアクセスするなどの不正な操作を行わないようメモリを管理します。また、CLR は、Microsoft .NET Framework のクラス ライブラリや、Office 2013 が公開しているオブジェクト モデルで提供される充実したライブラリへのアクセスを提供します。

それでは、どのような場合に VBA が適していて、どのような場合にマネージ コードが適しているのかを比較してみます。

➤ VBA が適している場合

- ドキュメントとコードを分離できない場合
- エンド ユーザーのコンピューター上に Microsoft .NET Framework がインストールされていない場合
- Excel のカスタム関数が必要な場合
- Word 固有のメソッド呼び出しを使用する場合

➤ マネージ コードが適する場合

- Visual Studio 2012 の開発環境の利点を活用したい場合
サーバー エクスプローラーや IntelliSense といったツールを利用できます。
- Microsoft .NET Framework の充実したライブラリを利用したい場合
たとえば、「 Web サービスの利用」や、「 XML の解析」などの煩雑な処理を、Microsoft .NET Framework が提供するクラス ライブラリを利用して行えます。
- コードの保守管理を容易にしたい場合
コードはドキュメントと分離されるので、1 つのアセンブリを複数のドキュメントで共用することができます。

- ドキュメント配布に関して柔軟な対応が必要な場合
アセンブリをネットワーク共有フォルダーに配置しておけば、Office 2013 アプリケーションでそのドキュメントを開くたびに、常に最新のアセンブリを利用できます。
- より柔軟なセキュリティ モデルを実装する場合

次に、セキュリティに関して留意すべき点を挙げます。

➤ VBA を使用する場合

マクロを動作させるためには、エンド ユーザーのコンピュータで、セキュリティ センターの [マクロの設定] を、次のいずれかを選択する必要があります。

[警告を表示して全てのマクロを無効にする] に設定し、マクロのセキュリティ警告の判断を、エンド ユーザーに判断させます。

[デジタル署名されたマクロを除き、全てのマクロを無効にする] に設定し、デジタル証明書をマクロ コードに署名します。

[すべてのマクロを有効にする] に設定し、すべてのコードを実行可能にします。但し、この場合は悪意のあるコードも実行可能になりますので、この選択肢は使用すべきではありません。

➤ マネージ コードを使用する場合

マクロを動作させるためのデジタル署名は、必要ありません。厳密な名前など Microsoft .NET Framework にはいくつかのセキュリティ評価の証拠となるものがあります。

管理者権限をもつユーザーにより、セキュリティ ポリシーを設定するツールを使用して、アセンブリを完全に信頼することができます。また、コードの実行を不可に設定することもできます。

エンド ユーザーにより、Office 2013 アプリケーションから、セキュリティ設定の変更をすることはできません。たとえば、万が一エンド ユーザーが危険性の存在するコードを含むドキュメントを開いても、そのコードは実行されません。

以上のように、VBA を使用するか、マネージ コードへ移行するかを判断するには、ドキュメントの配布形態、セキュリティやコードの保守のしやすさなどの観点から判断することができます。今後、Microsoft Office ソリューションを構築する際には、VBA に加えて Visual Studio 2012 も選択肢に加えてご検討ください。

Visual Studio については、こちらをご覧ください。

➔ 「Visual Studio Tools for Office デベロッパー ポータル」 (Office 2010 時点の情報)

<http://msdn.microsoft.com/ja-jp/office/aa905533.aspx>

➔ 「Visual Studio 2010 VBA ソリューションと Visual Studio Tools for Office ソリューションの比較」

<http://msdn.microsoft.com/ja-jp/library/ss11825b.aspx>

新しい選択肢「Office 用アプリ」

Microsoft Office ソリューション開発の新しい選択肢として、Office 2013 から Office 用アプリが導入されました。

Office 用アプリは、Web とユーザーが慣れ親しんだ Office アプリケーションを統合します。これにより、開発の焦点は固有の言語 (VBA や VSTO) から Web ベースの言語 (HTML、CSS、JavaScript など) に移り、Web 開発スキルを Office 開発に利用できるようになります。Office 用アプリにより、Web ベースの機能を Office アプリケーションに追加できます。



図 33 : Web を Office アプリケーションに組み込む

Office 用アプリについては、こちらをご覧ください。

➔ 「 Overview of apps for Office 」

[http://msdn.microsoft.com/en-us/library/jj220082\(v=office.15\).aspx](http://msdn.microsoft.com/en-us/library/jj220082(v=office.15).aspx)

Office 技術者向けの日本語情報

Office 製品に付属する開発者リファレンスの他にもそれぞれの製品及び技術に関する、ニュース、概要、学習コンテンツ、技術資料などをオンラインで提供しております。

- ➔ 「 Microsoft Office 製品情報」
<http://office.microsoft.com/ja-jp/default.aspx>
- ➔ 「 Microsoft Office 2010、2007 - ダウンロード、IT サポート」
<http://technet.microsoft.com/ja-jp/office/default.aspx>
- ➔ 「 Office デベロッパー センター - Office 開発者向け技術情報」
<http://msdn.microsoft.com/ja-jp/office/default.aspx>
- ➔ 「 Word 技術資料一覧」 (Office 2007 時点の情報)
<http://msdn.microsoft.com/ja-jp/office/cc811467.aspx>
- ➔ 「 Excel 技術資料一覧」 (Office 2007 時点の情報)
<http://msdn.microsoft.com/ja-jp/office/cc998644.aspx>
- ➔ 「 PowerPoint 技術資料一覧」 (Office 2007 時点の情報)
<http://msdn.microsoft.com/ja-jp/office/cc998646.aspx>
- ➔ 「 Access 技術資料一覧」 (Office 2007 時点の情報)
<http://msdn.microsoft.com/ja-jp/office/cc998645.aspx>
- ➔ 「 InfoPath 技術資料一覧」 (Office 2007 時点の情報)
<http://msdn.microsoft.com/ja-jp/office/cc998643.aspx>
- ➔ 「 Visio 技術資料一覧」 (Office 2007 時点の情報)
<http://msdn.microsoft.com/ja-jp/office/cc807027.aspx>
- ➔ 「 Microsoft サポートオンライン - 高度な検索」
<http://support.microsoft.com/search/?adv=1>
- ➔ 「 Microsoft Office 2010 スイート」
<http://office.microsoft.com/ja-jp/suites/>
- ➔ 「 Microsoft Office サポート」
<http://office.microsoft.com/ja-jp/support/?CTT=97>
- ➔ 「マイクロソフト ユーモール Office 2010 製品情報」
<http://www.microsoft.com/japan/athome/umall/office2010/default.aspx>
- ➔ 「 Office 2010 リソースキット」
<http://technet.microsoft.com/ja-jp/library/cc303401.aspx>



- ➔ 「 Microsoft Office 2010 」
[http://msdn.microsoft.com/ja-jp/library/cc313152\(v=office.12\).aspx](http://msdn.microsoft.com/ja-jp/library/cc313152(v=office.12).aspx)
- ➔ 「 Office 2013 リソースキット」
<http://technet.microsoft.com/ja-jp/library/cc303401.aspx>
- ➔ 「 Office トレーニング センター」
http://blogs.technet.com/b/microsoft_office_/archive/2013/01/18/office-aa.aspx


Microsoft Office 2013 の KB

Office 2013 のマクロに関する KB の一覧を以下に記載します。

KB 番号	KB タイトル	概要
2210978	戻り値の型は、VBA マクロは、64 ビット環境で実行を明示的に宣言する必要があります。	64 ビットの VBA アプリケーション内で戻り値の型を明示的に宣言しなければ、エラーが発生する場合がある。
983043	32 ビット版の Office 2010 プログラムで VBA マクロを編集すると、" コンパイル エラー : 64 ビット版のシステムで使用するには、このプロジェクトのコードを更新する必要があります " というエラー メッセージが表示される	VBA 6.0 で実行されるように設計された宣言ステートメントの構文を使用すると、エラーメッセージが表示される。
2188032	Word 2007 以降のバージョンで、コンテンツ コントロールの追加や削除をしていないにもかかわらず ContentControlBeforeDelete および ContentControlAfterAdd イベントが発生する場合があります	Word 2007 以降のバージョンで、ContentControlBeforeDelete および ContentControlAfterAdd イベントが、コンテンツ コントロールの追加や削除をしていないにもかかわらず、コンテンツ コントロール内で、半角英数以外の文字を入力するなどの、特定の操作を行った後、[元に戻す] または [やり直す] を実行すると発生します。
983044	[5 月 27 日に公開] 32 ビット版の Office 2010 プログラムで VBA マクロを実行しようとする、" コンパイル エラー : プロシージャが大きすぎます " というエラー メッセージが表示される	32 ビット版の Microsoft Office 2010 プログラムで Visual Basic for Applications (VBA) マクロを実行しようとする、エラー メッセージが表示されます。
2075994	アクセス 2010 年の修正プログラム パッケージ (アクセス ・ x ・ none.msp、エース ・ x ・ none.msp): 2010 年 6 月 29 日	BeforeChange イベントまたは afterInsert イベントを使って、データの更新および、サブフォームのレコードを更新し、接続モードで新しいレコードを作成する。フォームデータを編集後、サブフォームを選択すると、フォームのデータがコミットされ、誤ったサブフォームのデータが表示される。

このドキュメントに記載されている情報は、このドキュメントの発行時点におけるマイクロソフトの見解を反映したものです。変化する市場状況に対応する必要があるため、このドキュメントは、記載された内容の実現に関するマイクロソフトの確約とはみなされないものとします。また、発行以降に発表される情報の正確性に関して、マイクロソフトはいかなる保証もいたしません。このドキュメントに記載されている情報は、このドキュメントの発行時点における製品を表したもので、計画のためにのみ使用してください。情報は、将来予告なしに変更することがあります。

© 2012 Microsoft Corporation. All rights reserved.

Microsoft、Office、Office 、Office 2013、Office 2010、Office 2007、Office 97、Office 2000、Office XP、Office 2003、Word、Excel、PowerPoint、Access、InfoPath、Outlook、Visio、Visual Basic、MSDN、SharePoint、IntelliSense、Windows、Windows 2000、Windows XP、Windows Vista、Windows Server は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。