Introducing Microsoft Office InfoPath 2003

Microsoft Office InfoPath 2003 is a Microsoft Office System program that allows users to design and fill out business forms. InfoPath forms can range from simple forms that are used by members of a small team to sophisticated forms that are part of a larger business process and that are used by all employees in a company.

Among other things, you can use InfoPath to help prevent data-entry errors, keep formatting consistent, combine or export data from multiple forms, and provide different views of the data in a form. You can also establish a data connection to an external source of data, such as a database or Web service.

The forms that people fill out are saved as standard Extensible Markup Language (XML) files. This allows the information collected in the forms to be easily reused in other programs or integrated into existing business processes. A single InfoPath trip report form, for example, can provide XML data to a Customer Relationship Management (CRM) system, a petty cash system, and a travel planning system.

InfoPath includes the following features:

- ▶ The ability to fill out and design forms
- Flexibility when filling out forms
- Sample forms to get you started
- Support for developers
- Office integration
- Microsoft Windows SharePoint Services integration

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Take an animated tour

To help you get started with Microsoft Office InfoPath 2003, you can take an animated tour of the program. The tour provides an overview of InfoPath, including its key concepts and main product features. You can choose to view the entire tour or learn only about the tasks that interest you.

- 1. On the InfoPath **Help** menu, click **Microsoft Office InfoPath Tour**. The tour start page will be displayed in your Web browser.
- 2. On the tour start page, click **Start Tour**. The tour will be displayed in a new window.

Notes

For best results, viewing the tour on a computer with an 800×600 or higher screen resolution is recommended.

Macromedia Flash Player is required to view the tour. This software is included with all supported versions of Microsoft Windows.



What's new

- New and improved controls
- Better ways to work with data
- Improved support for data connections
- Visual Studio .NET integration
- Layout enhancements
- Better integration with existing business processes
- More new features

About working with forms that do not have service pack features

If you have Microsoft Office InfoPath 2003 Service Pack features enabled on your computer, and you open an InfoPath form that was designed without the service pack installed, service pack features are unavailable while you use that form. These features include filters, rules, user roles, calculations that do not require script, and certain types of controls. In addition, InfoPath Help contains information that may not apply to that form. For information about features included with the service pack, see What's new in the Microsoft Office InfoPath 2003 Service Pack.

You can upgrade the form by enabling service pack features for it. To do so, open the form in design mode, and then click **Add Service Pack Features to Form** on the **Tools** menu. If you are not sure whether you should do so, ask the form designer or your system administrator.

Note If you upgrade the form, you can only remove the service pack features from it later by exporting a copy of the form. In addition, users cannot open the upgraded form if they do not have the service pack installed and enabled on their computers.

If you do not want to upgrade the form and you want to browse InfoPath Help topics that do not discuss service pack features, ask your system administrator about disabling service pack features for your InfoPath installation. For more information, see Enable or disable service pack features.

Enable or disable service pack features

The Microsoft Office InfoPath 2003 Service Pack improves the way you work with InfoPath by allowing you to add filters, rules, and other useful features to your forms. You can also insert new types of controls, such as file attachment controls and master/detail controls, and create formulas without writing script.

To use these features when you design or fill out a form, you must enable the service pack features for your installation of InfoPath.

Note If you enable service pack features, the new forms you design can be filled out only by users who have the service pack features enabled on their computers.

- 1. On the **Tools** menu, click **Options**, and then click the **Advanced** tab.
- 2. Do one of the following:
- To enable service pack features, clear the Disable Service Pack features check box.
- To disable service pack features, select the **Disable Service Pack features** check box.

Note If you do not see the **Disable Service Pack features** check box, then you have not installed the service pack for InfoPath.

3. Restart InfoPath.

About the InfoPath workspace

Regardless of whether you are filling out or designing a form, the Microsoft Office InfoPath 2003 workspace is divided into two main areas; the form area and the task pane area. The form area appears on the left side of the screen and the task pane area appears on the right.

- ▶ The form area
- ▶ The task pane area

About forms and form templates

In Microsoft Office InfoPath 2003, you can either fill out existing forms or create your own forms in design mode. In design mode, the files you create are called form templates.

- Form templates
- ▶ Forms

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Switch between modes

Microsoft Office InfoPath 2003 has two separate modes that are designed for two distinct tasks—filling out forms and designing forms.

- Switch to filling out a form
- Switch to design mode

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

About designing new, blank forms

When designing a Microsoft Office InfoPath 2003 form, starting with a blank form provides the most flexibility and allows you to create a design that best fits your needs. While you can experiment and work the way that suits you best, following a recommended design process may save you valuable time and effort.

The **Design Tasks** task pane contains links to design tools in the recommended order of design tasks. These tasks are listed below and provide useful information on how to design your new, blank form.

Notes

If you have task panes turned off, you can show them again by pressing CTRL+F1 on your keyboard.

To access the **Design Tasks** task pane at any time, in design mode, click **Design Tasks** on the **Standard** toolbar.

1. Create the form's layout

Before you begin adding controls to a new, blank form, start by inserting layout tables. A layout table is a framework that includes rows and columns for organizing and arranging form content, including controls, sections of a form, logos, and other types of graphics. These regions provide a visual structure for the content of a form. You can drag layout tables onto your form from the **Layout** task pane. This task pane also provides easy ways to customize layout tables using familiar Microsoft Office 2003 table-editing commands, such as adding rows and columns, and splitting and merging table cells.

▶ Tip

2. Add controls to the form

After creating your form's layout, you can add functionality to your form by inserting controls. The **Controls** task pane provides access to all of the controls available in InfoPath, including any custom Microsoft ActiveX controls you've added. You can add controls to your form either by clicking them or by dragging them from the task pane to the form area. Based on the type of controls you add to your form, users filling out your form will be able to type text into text fields, make selections from option buttons and check boxes, choose entries from lists, and click command buttons to carry out commands.

3. Bind controls to the data source

Controls on a form must be bound to fields and groups in the form's data source. This allows the data that users type into the form to be saved. InfoPath automatically creates the fields and groups for you when you add controls to a new, blank form. If you're working from an existing XML Schema or data source, or if you've cleared the Automatically create data source check box in the Controls task pane, you will be prompted to manually bind each individual control you insert. In either case, you can use the Data Source task pane to view the structure of your form's data, show and hide details for each of the fields and groups, and modify their associations.

4. Create custom views

Every form that you open or create in InfoPath has a default view. You can use the **Views** task pane to design additional views, so that users can look at the information in the form in different ways. For example, in a project status form you can have a default view, which shows detailed data about the project, and another view that summarizes that data. You can also create print views that specify custom print settings for users filling out and printing the finished form. If you want to switch the view of the form based on criteria such as user roles, you can use rules to do so.

5. Publish the form

While you can save the form you are designing to your local hard drive at any time, you must publish your finished form to a shared location in order to make it available to other users. Before you publish your form, you should test it to make sure it looks and works as expected. In design mode, you can use the **Preview Form** command to view your form exactly as it will appear to users who are filling it out. After testing, you can use the Publishing Wizard to publish your finished form to a shared folder on your computer or your company's network, to your intranet or Internet Web server, or to a Microsoft Windows SharePoint Services site.

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Troubleshoot viewing the InfoPath Tour

- ▶ How can I take the InfoPath Tour?
- ▶ I want to view the entire tour; what is the intended order of the tour contents?
- ▶ When I click Start Tour on the launch page, nothing happens.
- ▶ When I start the InfoPath Tour, I get an error message that says my computer does not meet the system requirements.

About sample forms

Microsoft Office InfoPath 2003 comes with 25 sample forms that offer the following benefits:

Ready-to-use forms Predesigned business forms such as expense reports, invoices, and status reports can be used "as is" or customized in design mode to suit your needs. You can use the **Fill Out a Form** dialog box that appears whenever you open InfoPath to quickly find sample forms.

Feature examples The sample forms show useful InfoPath features in action— for example, the Asset Tracking sample form illustrates how views can be used to present similar information in different ways. Similarly, the Vendor Information sample form includes a custom task pane, which shows how adding form-specific commands and content in a task pane can help make a form easier to use. These are only two examples of useful features. You can browse through the sample forms to find many more.

Control examples You can refer to the sample forms to learn more about working with controls. For example, you can open the Expense Report sample form in design mode to better understand how a repeating table works. You can view the table's properties, see what types of controls are used in it, as well as see which fields or groups the controls are bound to. This can help you think of ways that controls can be used in your own forms.

Customizing sample forms

Although a sample form can be used as is, you can also customize it in design mode. This is useful if you like a particular form but it doesn't quite meet your needs, or if you want the sample form to more closely match your company's corporate style. For example, you might add a control, change data validation rules, change the default font or color scheme, or add a company logo. For more information about customizing sample forms, see <u>Customize a sample form</u>.

Sample forms for advanced users

If you're an advanced form designer, or if you simply want to improve your technical skills, you may find it useful to review the four developer sample forms that are included with InfoPath. Unlike the sample forms mentioned earlier, these forms are intended solely as a teaching tool for helping designers learn more advanced form-design concepts. For example, you can review the User Interface sample form to learn about customizing menus, toolbars, task panes, and other parts of the form's user interface. For more information about opening developer sample forms, see Open a sample form.

Visit the Office Developer Center on the Microsoft Developer Network Web site for the latest information about programming with InfoPath, including product news, technical articles, downloads, and samples.

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Open a sample form

Microsoft Office InfoPath 2003 includes a variety of predesigned business forms that can be used as is or customized to suit your needs. In addition, InfoPath comes with four developer sample forms that explore certain form design features in greater detail. These forms are targeted at more advanced form designers, such as those who use script in their forms.

- Open a sample form
- Open a developer sample form

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Customize a sample form

When you customize a sample form, you can work with the form in many of the same ways you would when designing a new blank form. For example, you can insert controls, insert fields and groups in the data source, add data validation and conditional formatting, and publish the form.

1. In the **Fill Out a Form** dialog box, click **Design a Form**.

Note If the **Fill Out a Form** dialog box isn't visible, open the **Design a Form** task pane by clicking **Design a Form** on the **File** menu.

- 2. Under **Design a new form** in the **Design a Form** task pane, click **Customize a Sample**.
- 3. In the **Customize a Sample** dialog box, double-click the sample form you want to customize.

Notes

When you customize a sample form, the following limitations apply:

You cannot automatically create the data source when inserting controls.

You cannot modify existing fields or groups in the data source.

Depending on the sample form, you might not be able to add fields or groups to part of the data source.

The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Getting Help on form development

This topic describes the Help resources that are available for designing forms using Microsoft Office InfoPath 2003. These resources are available in the Help systems for InfoPath and the Microsoft Script Editor (MSE), and at various Microsoft Web sites.

The InfoPath Developer's Reference

The InfoPath Developer's Reference is a set of documentation that can help you develop forms and customize sample forms.

Open the InfoPath Developer's Reference

The following table describes the sets of documentation in the InfoPath Developer's Reference.

Resource	Description
Programming Concepts	Conceptual information about programming InfoPath, including information about the InfoPath object model and form definition (.xsf) file.
Developer Sample Forms	Documentation about the developer sample forms included with InfoPath. The sample forms demonstrate some of the key form development features.
InfoPath Object Model Reference	Information about the COM-based object model for interacting with InfoPath forms and their underlying XML documents. This set of topics provides reference information for the collections, objects, properties, methods, events, and enumerations in that object model.
InfoPath XSF Reference	A complete set of reference documentation for the .xsf file, including all of its namespaces, groups, types, elements, and attributes.

The InfoPath 2003 Software Development Kit

The Microsoft Office InfoPath 2003 Software Development Kit (SDK) contains the InfoPath Developer's Reference, as well as tools and additional documentation for developing forms. You can download the SDK from the Microsoft Office Update Web site or you can browse the contents of the SDK at the Microsoft Developer Network (MSDN) Web site. The information and tools in the InfoPath SDK are updated frequently.

- Browse the InfoPath SDK
- Download the InfoPath SDK

Microsoft Script Editor Help

Reference reference and tutorial.

By adding script to a form, you can add functions that automate and extend the functionality of the form. Microsoft Script Editor (MSE) enables you to develop script for a form and supports several scripting languages, including Microsoft Visual Basic Scripting Edition (VBScript) and JScript.

Open MSE Help

The following table describes the sets of documentation in MSE Help that are relevant to developing forms.

Resource	Description
Microsoft Script Editor Help	Information about using MSE, including how to customize the program, as well as techniques for developing and debugging script.
XML	Information about using Extensible Markup Language, the XML Document Object Model (DOM), XML Schemas, XSL Transformation (XSLT), and XPath. It also includes various developer guides and language references that discuss each of these XML-based technologies.
JScript Reference	Information about using JScript, including a language reference and tutorial.
VBScript	Information about using VBScript, including a language

Resources on the Web

The following table lists additional resources that provide information about developing forms. All of these resources are available on the Web and are updated frequently.

Resource Description

InfoPath This Web site contains white papers, sample forms, tools, Developer articles, and general information about developing InfoPath Center forms. To access the InfoPath Developer Center from within InfoPath, click **Microsoft Office InfoPath Developer**Resources on the **Help** menu.

Office This Web site contains information about using programs in Developer the Microsoft Office System as a development platform. Visit Center the Office Developer Center.

Microsoft This Web site contains information about a variety of Developer Microsoft development platforms and tools. Visit the Microsoft Network Developer Network.

(MSDN)

Microsoft This Web site contains information about developing XML programs and services that use XML. Visit the Microsoft XML Developer Developer Center.

Center

Microsoft This Web site contains information about using various Scripting Microsoft scripting languages, including JScript and VBScript. Developer Visit the Microsoft Scripting Developer Center.

Center

Navigate the InfoPath Tour with the keyboard

The Microsoft Office InfoPath 2003 Tour can be navigated using the mouse or the keyboard. To navigate the tour using your keyboard, follow these steps.

- 1. In InfoPath, press ALT+H to open the **Help** menu, and then press T to launch the InfoPath Tour.
- 2. On the tour's launch page, press the TAB key until the **Start Tour** button is outlined, and then press ENTER.
- 3. When the tour window is displayed and the navigation buttons appear, do one of the following:
- To move the yellow navigation box forward to the option you want to select, press the TAB key as needed.
- To move the yellow navigation box backward to the option you want to select, press SHIFT+TAB as needed.
- 4. When the button or option you want is selected with the yellow navigation box, press the SPACEBAR or ENTER to carry out the selected command.

Keyboard shortcuts

This Help topic describes keyboard shortcuts for common tasks in Microsoft Office InfoPath 2003, such as filling out or designing forms. The keyboard shortcuts refer to the U.S. keyboard layout. Keys on other keyboard layouts may not correspond exactly to the keys on a U.S. keyboard.

For keyboard shortcuts in which you press two or more keys simultaneously, the keys are separated by a plus sign (+). For keyboard shortcuts in which you press one key immediately followed by another key, the keys are separated by a comma (,).

To print this topic, press TAB to select **Show All**, press ENTER, and then press CTRL+P.

Filling out forms

- Moving between fields
- Entering text in fields
- Selecting text in fields
- ▶ Formatting text in fields
- Working with right-to-left text
- Working with tables
- Making corrections and saving changes

Designing forms

- Developing forms
- Formatting text
- Working with tables
- Making corrections and saving changes
- Previewing a form before printing

Getting Help

- ▶ Using the **InfoPath Help** task pane
- Using the Help window
- Typing a search question

Using menus, toolbars, and task panes

- Accessing and using menu bars and toolbars
- Accessing and using task panes

Using dialog boxes

- Accessing and selecting options in dialog boxes
- Using edit boxes within dialog boxes

Customize how InfoPath starts

Microsoft Office InfoPath 2003 supports the use of a number of command-line options. These command-line options can be used to perform a variety of tasks in InfoPath when the program starts, such as opening a new form, setting the default location for saved forms, merging forms, and opening a form template in design mode.

To customize how InfoPath starts, do the following:

- 1. Open the MS-DOS prompt.
- 2. At the command prompt in the Command Prompt window, switch to the disk drive and directory that contains the InfoPath.exe file for example C:\Program Files\Microsoft Office\OFFICE11\.
- 3. At the command prompt, type **InfoPath.exe**, followed by a space and the command-line option and values that you want to use.

The following table describes each of the InfoPath command-line options.

Option	Description
Inew <form formtemplate></form formtemplate>	Opens a new form based on an existing form or form template, without automatically setting the default location for saved files. Also adds the form to the list of most recently used files. If no form or form template is specified, InfoPath starts without loading a form. The name of the form or form template can be a file name, a fully-qualified path and file name, or a Uniform Resource Name (URN). You can also use this option with the <i>Idefaultsavelocation</i> option.
/defaultsavelocation <url></url>	Sets the specified URL as the search path for the Save As dialog box the first time the form is saved. The <i>Idefaultsavelocation</i> option is disregarded unless the <i>Inew</i> option is also specified. Any necessary character encoding of the URL must be done by the caller.

laggregate <form1 form2 form3> <form formtemplate></form formtemplate></form1 form2 form3>	Merges the specified forms into a single form. Individual form names in the form list should not be enclosed by quotation marks, but the entire form list can be enclosed by quotation marks. The form names must be separated by a pipe () character. The form or form template used to perform the merge must also be specified. If a form (.xml file) is specified, InfoPath opens the form in edit mode and then merges the data into that form. If a form template (.xsn file) is specified, InfoPath begins the merge by using a new form based on the form template.
Idesign <formtemplate></formtemplate>	Opens the specified form template in design mode. If no form template is specified, InfoPath starts in design mode, and no file is opened. If a form (.xml file) is specified, the form opens in edit mode.
lembedding	Starts InfoPath without displaying a window.
/noautoformsdialog	Starts InfoPath without automatically displaying the Fill Out a Form dialog box.
/cache ClearAll	Clears the cache in which form templates are stored on the computer.

Note InfoPath also supports opening forms and form templates by using no options on the command line. If you specify a form name, the form is opened for editing; if you specify a form template name, a new form that is based on the form template is opened.

Examples

In this example, the *Inew* option is used to open a new form and set the default location for saved files:

InfoPath.exe /new "MyForm.xml" /defaultsavelocation "C:\MyForms"

In this example, the *laggregate* option is used to merge the XML data from two forms into one that is based on the specified form template:

InfoPath.exe /aggregate "MyForm1.xml|MyForm2.xml"

"MyForm.xsn"

In this example, the **/design** option is used to open a form template in design mode:

InfoPath.exe /design "MyForm.xsn"

Set InfoPath as the default program for opening and editing InfoPath forms

If Microsoft Office InfoPath 2003 is installed on your computer but does not start when you open InfoPath forms, you can set InfoPath as the default program for opening and editing InfoPath forms.

- 1. In InfoPath, click **Options** on the **Tools** menu.
- 2. Under Microsoft Office options in the Options dialog box, select the Check whether Office is the default program for opening and editing .xml files check box.

Show or hide the Fill Out a Form dialog box

- 1. On the **Tools** menu, click **Options**.
- 2. On the **General** tab in the **Options** dialog box, clear or select the **Do not automatically show the Fill Out a Form dialog box** check box.

Add a form to the Favorites form category

You can add a form to the Favorites category in the **Fill Out a Form** dialog box. This can make it easier to find frequently used forms.

- 1. On the File menu, click Fill Out a Form.
- 2. In the **Fill Out a Form** dialog box, click the form that you want, and then click **Add to Favorites** under **Form tasks**.

Create a form category

When designing forms, you can create custom categories to organize your forms. These categories appear in the **Fill Out a Form** dialog box.

- 1. In design mode, open a form that you want to assign to the category.
- 2. On the **File** menu, click **Properties**.
- 3. In the Form Properties dialog box, select the Enable custom category check box.
- 4. In the **Name** box, type the name of the category, and then click **OK**.
- 5. To assign additional forms to this category, repeat steps 1 through 4 for each form.

Add a button, menu, or command

You can rearrange and modify toolbar buttons, menus, and menu commands by using the mouse or you can use the **Rearrange**Commands dialog box which is accessible using a keyboard menu.

These tasks are identified as (keyboard accessible).

Do one or more of the following:

- Add a button, menu, or command to a toolbar
- Add a button, menu, or command to a toolbar (keyboard accessible)
- Add a command to a menu
- Add a command to a menu (keyboard accessible)

Note If you are having trouble customizing toolbars or menus, you may be attempting to add a button or command that is not applicable to the program environment you are working in. For example, if you are currently filling out a form, you cannot add a command that is available only in design mode, such as the Layout Table command, to a menu.

Find and open a form

When filling out a form, you can open new, blank forms, as well as forms that you have saved. When designing a form, you can open form templates that are installed on your computer or a shared folder on a network, as well as form templates associated with form libraries.

Open a form to fill it out

- Open a new form
- Open a saved form

Open a form in design mode

- Open a blank form
- Open a sample form
- Open a saved form template
- Open a form library's form template
- ▶ Tip

Remove a form from your computer

Do one of the following:

- Remove a form
- ▶ Remove a sample form
- Remove a custom installed form

Print more than one copy

When designing a form, you can specify how many copies of the form should print by default when a user prints the form. You can customize this setting for each view of a form. Users can adjust this setting before printing a form they have filled out.

- Specify the number of printed copies when designing a form
- ▶ Change the number of printed copies when filling out a form

Select a paper source

When designing a form, you can specify which paper source should be used when a user prints the form. You can customize this setting for each view of a form. Users can adjust this setting before printing a form they have filled out.

- Specify the paper source when designing a form
- ▶ Change the paper source when filling out a form

Select page orientation

When designing a form, you can specify default orientation settings for each view in the form. Users can adjust page orientation before printing a form they have filled out.

- Specify the page orientation when designing a form
- ▶ Change the page orientation when filling out a form

Print background colors and pictures

When you design a form, you can use options in the **View Properties** dialog box to include colors or pictures in the background of your form. By default, Microsoft Office InfoPath 2003 does not print these background colors or pictures. To print form backgrounds, use the following procedure:

- 1. On the **Tools** menu, click **Options**.
- 2. On the **General** tab in the **Options** dialog box, select the **Print** background colors and background pictures check box.
- 3. To print the form, click **Print** on the **File** menu.

Preview a form before printing

- 1. On the **Standard** toolbar, click **Print Preview** .
- 2. Use the buttons on the toolbar to review the form or make adjustments before printing.

Print a form

To print the current form, click **Print** on the **Standard** toolbar.

▶ Tip

Check the progress of your print job

Double-click the **Printer** icon in the status area of the Microsoft Windows taskbar.

When a file is printed, the icon disappears.

Note If you're printing a simple form, the printer icon may not appear in the status area of the taskbar long enough for you to click it to check printing progress.

Cancel printing

- 1. Double-click the **Printer** icon in the status area of the Microsoft Windows taskbar.
- 2. Right-click the print job you want to cancel, and then click **Cancel** on the shortcut menu.

Note If you're printing a simple form, the printer icon may not appear in the status area of the taskbar long enough for you to double-click it.

Change the page margins for a form

Page margins are the blank space around the edges of the page. Typically, nothing appears or prints in the margins except headers and footers, if your form has them. When designing a form, you can specify default margin settings for each view in a form. Users can adjust this setting before printing a form they have filled out.

- Specify default margins when designing a form
- Change page margins when filling out a form

Design a print view

When designing a form, you can create a special version of the form — called a print view— that is designed specifically for printing the form.

For example, if your form contains multiple views, you might design a print view that consolidates key information from each of those views in a single printed page. You can also specify custom print settings for a print view, such as page margins or printer, which allows you to control how and where the form is printed.

Developers can further control the way their forms print by designing and designating print views for use in Microsoft Office Word.

- Design a print view
- Design a print view specifically for Word

Insert a header or footer

When you design a form, you can specify that headers or footers appear in the top and bottom margins of the page when users print the form. If your form has multiple views, you can customize headers and footers for each distinct view.

- 1. In design mode, click **Header and Footer** on the **View** menu.
- 2. Under **Headers and footers** in the **View Properties** dialog box, do one of the following:
- To add a header to the form, click Header. In the Header
 Format dialog box, specify the text and formatting options you want.
- To add a footer to the form, click Footer. In the Footer Format dialog box, specify the text and formatting options you want.

▶ Tip

Insert a page break

- 1. In design mode, click where you want to start a new page.
- 2. On the Insert menu, click Page Break.

Note In design mode, a page break appears as a dashed line in the form area. If you preview your form, you won't see this dashed line or any other indication of the page break. To see how the page break will affect your form when it's printed, click **Print Preview** on the **Standard** toolbar.

Get form updates

Before filling out a form, you can use the following procedure to ensure that you have the most recent version of the form. It is also a good idea to check for updates before filling out a form offline.

- 1. On the File menu, click Fill Out a Form.
- 2. In the **Fill Out a Form** dialog box, click the form you want to update.
- 3. Under Form tasks, click Get Update of this Form.

▶ Tip

Export a form to Excel

You can export the data in Microsoft Office InfoPath 2003 forms to Microsoft Office Excel 2003. This allows you to analyze and work with the data you have collected in forms. You can export data from the current view of an opened form by using the Export to Excel Wizard.

Note The Export to Excel Wizard also allows you to export data from several forms at once.

1. Open the form you want to export to Office Excel 2003.

Note If the form you open contains digital signatures, you must remove the signatures before you export the form to Office Excel 2003.

- 2. On the **File** menu, point to **Export To**, and then click **Microsoft Office Excel**.
- 3. Follow the steps in the Export to Excel Wizard.
- 4. To export data from additional views of the same form, repeat the previous steps for each view.

Merge forms

Form merging must be enabled in design mode. When the forms are merged, the only parts of the form that are combined are those contained in a repeating table or repeating section or those associated with a list control or rich text box. The remaining parts of the form will not be merged.

1. Type or enter the data you want into your form, or open an existing form (.xml file).

Note After you merge forms, any non-repeating and non-rich text controls will contain data from the form you start with.

2. On the **File** menu, click **Merge Forms**.

Note If form merging has not been enabled in design mode, the **Merge Forms** command is unavailable.

3. In the **Merge Forms** dialog box, select the files you want to merge, and then click **Merge**.

About AutoComplete

Microsoft Office InfoPath 2003 uses the AutoComplete feature provided by Microsoft Internet Explorer. AutoComplete saves previous entries you've typed into fields when filling out a form. If AutoComplete is turned on and enabled for a field on the form, when you type the first few characters into that field, AutoComplete lists possible matches from entries you've typed before. You can then insert the entry you want, or if none of the suggestions match what you are typing, you can continue typing into the field.

Change AutoComplete settings

When filling out a form, you can configure AutoComplete to save and suggest only the information you want. You can choose whether to use AutoComplete for Web addresses, fields on forms, and passwords, or not to use it at all. You can also delete AutoComplete entries you've added. When designing a form, you can specify whether or not AutoComplete is enabled for a text box control on a form.

Do one of the following:

- ▶ Turn AutoComplete on or off
- ▶ Delete previous AutoComplete entries
- ▶ Enable or disable AutoComplete for a text box control

Find and replace text

You can use Microsoft Office InfoPath 2003 to quickly search for and replace text on your form.

- Find text
- Replace text

Search options

- Search by using wildcards
- Search for whole words only
- Search by matching letter case

Notes

When filling out a form, the Find and Replace features search only the text you type into the form and not text that is part of the form template, such as field labels.

Although you can search the text in a read-only text box, you won't be able to replace that text.

Check for data errors in a form

Microsoft Office InfoPath 2003 automatically checks the information you enter into a form for data errors as you type. If a problem with the information you type into a form is found, you will be presented with either a traditional dialog box alert or an inline alert.

- 1. Type into the form.
- 2. Do one of the following:
 - ▶ If a dialog box alert appears
 - If an inline alert appears
- 3. To go to the next error in the form, click **Go to Next Error** on the **Tools** menu.

Notes

If you forget to enter data into a required field, a red asterisk will appear instead of an inline alert. To make the red asterisk disappear, enter a value in the field.

Redo your last action

On the **Standard** toolbar, click **Redo** .

Note If you can't redo the last action, the **Redo** button will appear dimmed.

Undo mistakes

To undo the last action you took, click Undo on the Standard to	olbar.
Note If you decide that you didn't want to undo an action, click Re	do
on the Standard toolbar.	

Adjust line or paragraph spacing

When designing a form, you can increase or decrease the amount of space between multiple lines of text. You can also adjust the spacing before or after each paragraph of text.

- Adjust line spacing
- Adjust paragraph spacing

Notes

When filling out a form, you can use the previous procedures to adjust line or paragraph spacing in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field, and then click the **Format** menu. If the **Paragraph** command appears active instead of dimmed, the field supports line and paragraph spacing.

Remove a list

When designing a form, you can remove bulleted list formatting, numbered list formatting, or list controls from the form. When filling out a form, you can remove bulleted or numbered list formatting that you applied to text in certain types of fields.

- ▶ Remove bullets or numbering from text
- Remove a list control

Apply a heading format to text

When designing a form, you can apply a heading format to control labels or other text. Microsoft Office InfoPath 2003 includes six built-in heading formats.

- 1. Select the text whose formatting you want to change.
- 2. On the **Formatting** toolbar, click **Font** 4.
- 3. In the **Pick formatting to apply** box in the **Font** task pane, select the heading style you want to apply. InfoPath changes the text to the formatting specified for the style you applied.

▶ Tip

Note When filling out a form, you can use the previous procedure to apply a heading format to text in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Font** button on the **Formatting** toolbar appears active instead of dimmed, the field supports heading formatting.

Modify a list

When designing a form, you can format lists and list controls. When filling out a form, you can modify lists that you inserted into certain types of rich text fields.

- ▶ Change the appearance of a list
- ▶ Change the starting number of a list

Apply bold formatting

When designing a form, you can apply bold formatting to text that appears on your form. You can also make bold formatting the default format for text inside certain types of controls, such as text boxes and expression boxes.

- 1. Select the text or control that you want to change.
- 2. On the **Formatting** toolbar, click **Bold B**.

Note When filling out a form, you can use the previous procedure to make text bold in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Bold** button on the **Formatting** toolbar appears active instead of dimmed, the field supports bold text.

Apply italic formatting

When designing a form, you can apply italic formatting to text that appears on your form. You can also make italic formatting the default format for text inside certain types of controls, such as text boxes and expression boxes.

- 1. Select the text or control you want to change.
- 2. On the **Formatting** toolbar, click **Italic** .

Note When filling out a form, you can use the previous procedure to make text italic in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Italic** button on the **Formatting** toolbar appears active instead of dimmed, the field supports italic text.

Apply or remove highlighting

When designing a form, you can use the Highlight tool to emphasize text in your form or to highlight certain controls.

Do one of the following:

- Apply highlighting
- Remove highlighting

Note When filling out a form, you can use the previous procedures to apply highlighting in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Highlight** button on the **Formatting** toolbar appears active instead of dimmed, the field supports highlighting.

Apply strikethrough formatting

When designing a form, you can apply strikethrough formatting to text on your form. You can also make strikethrough formatting the default format for text inside certain types of controls, such as text boxes and expression boxes.

- 1. Select the text or control that you want to change.
- 2. On the **Formatting** toolbar, click **Font** 4.
- 3. In the **Font** task pane, select the **Strikethrough** check box.

Note When filling out a form, you can use the previous procedure to apply strikethrough formatting to text in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Font** button on the **Formatting** toolbar appears active instead of dimmed, the field supports strikethrough formatting.

Change the color of text

When designing a form, you can change the color of text that appears on the form. You can also specify a default color for text inside certain types of controls, such as text boxes and expression boxes.

- 1. Select the text or control you want to change.
- 2. Do one of the following:
- To apply the color most recently used for text, click **Font Color**A on the **Formatting** toolbar.
- To apply a different color, click the arrow next to **Font Color** A, and then click the color you want to apply.

Note When filling out a form, you can use the previous procedure to change the color of text in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Font Color** button on the **Formatting** toolbar appears active instead of dimmed, the field supports colored text.

Change the font

When designing a form, you can change the font for text that appears on the form. You can also specify a default font for text that appears inside certain types of controls, such as text boxes and expression boxes.

- 1. Select the text or control you want to change.
- 2. On the **Formatting** toolbar, click a font name in the **Font** box

Note When filling out a form, you can use the previous procedure to change the font in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Font** box on the **Formatting** toolbar appears active instead of dimmed, the field supports font changes.

Change the size of text

When designing a form, you can change the size of text that appears on the form. You can also specify a default size for text inside certain types of controls, such as text boxes and expression boxes.

- 1. Select the text or control you want to change.
- 2. On the **Formatting** toolbar, enter a point size in the **Font Size** box 10 .

Notes

When filling out a form, you can use the previous procedure to change the size of text in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Font Size** box on the **Formatting** toolbar appears active instead of dimmed, the field supports different sizes of text.

The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Copy all formatting from one object to another

Microsoft Office InfoPath 2003 includes a Format Painter tool that lets you quickly reuse the same text formatting in more than one place. This is the quickest way to copy text formatting— for example, font size and color— from one object to another.

- 1. Select the text, control, or other object with the attributes you want to copy.
- 2. On the **Standard** toolbar, click **Format Painter ✓**, and then click the object you want to copy the attributes to.

▶ Tip

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Make text superscript or subscript

When designing a form, you can use superscript or subscript text on the form. You can also apply superscript or subscript formatting to controls. Doing so makes the controls slightly higher or lower than their original position on the form.

- 1. Select the text or control you want to format as superscript or subscript.
- 2. On the **Formatting** toolbar, click **Font** 4.
- 3. In the **Font** task pane, select either the **Superscript** or **Subscript** check box.

Note When filling out a form, you can use the previous procedure to apply superscript or subscript format in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Font** button on the **Formatting** toolbar appears active instead of dimmed, the field supports superscript and subscript formatting.

Remove formatting from text

When designing a form, you can remove any character formatting you applied to text on your form. If you specified default formatting, such as bold or italic text, for a certain control, you can also use the following procedure to remove that formatting.

- 1. Select the text or control whose formatting you want to remove.
- 2. On the **Formatting** toolbar, click **Font** 4.
- 3. In the **Font** task pane, click **Clear Formatting** in the **Pick formatting to apply** box. Microsoft Office InfoPath 2003 restores the text to the default formatting for your form.

Note When filling out a form, you can use the previous procedure to remove text formatting in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Font** button on the **Formatting** toolbar appears active instead of dimmed, you can remove formatting from text in this field.

Underline text

When designing a form, you can underline text that appears on your form. You can also make underlined text the default format for text inside certain types of controls, such as text boxes and expression boxes.

- 1. Select the text or control you want to change.
- 2. On the **Formatting** toolbar, click **Underline U**.

Note When filling out a form, you can use the previous procedure to underline text in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Underline** button on the **Formatting** toolbar appears active instead of dimmed, the field supports underlined text.

Copy a hyperlink

- 1. Select the hyperlink you want to copy.
- 2. On the **Standard** toolbar, click **Copy** .
- 3. Select where you want to place a copy of the hyperlink in your form, and then click **Paste** and the **Standard** toolbar.

Note When filling out a form, if the hyperlink loses its formatting when you paste it into the form, you may be trying to insert a hyperlink where it is not allowed.

Edit a hyperlink

You can edit both the display text as well as the description of a hyperlink.

- ▶ Edit a hyperlink when filling out a form
- ▶ Edit a hyperlink when designing a form

Insert a hyperlink

Microsoft Office InfoPath 2003 automatically inserts a hyperlink for you when you type a Web site address (such as www.example.com), a file location, or an e-mail address when filling out or designing a form. However, when filling out a form you can only insert a hyperlink into fields that allow hyperlinks. You can also turn text or a picture into a hyperlink.

Note To automatically create a hyperlink for text that includes spaces, enclose the phrase in quotation marks ("). For example, you would need to type "www.example text" to have the entire phrase formatted as a hyperlink.

- ▶ Insert a hyperlink when filling out a form
- ▶ Insert a hyperlink when designing a form

Remove a hyperlink

Do one of the following:

- Remove a hyperlink completely
- Remove a hyperlink but keep the text or picture

Note When filling out a form, you can only remove hyperlinks that are not a part of the form's design.

Justify text

When designing a form, you can justify text that appears on the form. You can also specify that text should be justified inside certain types of controls, such as text boxes and expression boxes.

- Justify text on a form
- Justify text inside controls

Indent paragraphs

To indent paragraphs when designing a form, do one of the following.

- ▶ Increase or decrease the left indent of an entire paragraph
- ▶ Create a first-line or hanging indent

Notes

When filling out a form, you can use the previous procedures to indent paragraphs in certain types of rich text fields. To quickly determine whether you're in a rich text field, place your insertion point in the field. If the **Increase Indent** or **Decrease Indent** button on the **Formatting** toolbar appears active instead of dimmed, the field supports indented text.

The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Center text

When designing a form, you can center text that appears on the form. You can also specify that text should be centered inside certain types of controls, such as text boxes and expression boxes.

- Center text on a form
- Center text inside controls

Align text left or right

When designing a form, you can align text that appears on the form. You can also specify that text should align to the left or right inside certain types of controls, such as text boxes and expression boxes.

- Align text on a form
- Align text inside controls

Check spelling

By default, Microsoft Office InfoPath 2003 checks spelling automatically as you type, using wavy red underlines to indicate possible spelling problems. You can also check all spelling errors at once.

- Check spelling automatically as you type
- ▶ Check spelling all at once

Notes

When designing a form, you can specify where in the form users are allowed to check their spelling. For example, you can turn on the spelling checker for text boxes, but turn it off for lists.

When filling out a form, the spelling checker checks only text that is typed into a field that allows a spelling check and does not check text that is part of the form's design.

Set spelling preferences

- 1. On the **Tools** menu, click **Options**, and then click the **Spelling** tab.
- 2. Select the options you want.

Show or hide wavy underlines

You can show or hide the wavy red underlines that appear when you automatically check spelling.

- 1. On the **Tools** menu, click **Options**, and then click the **Spelling** tab.
- 2. Make sure that the **Check spelling as you type** check box is selected.
- 3. Select or clear the **Hide spelling errors** check box.

Turn on or off the automatic spelling checker

- 1. On the **Tools** menu, click **Options**, and then click the **Spelling** tab.
- 2. Select or clear the **Check spelling as you type** check box.

Check spelling in other languages

When you use Microsoft Office InfoPath 2003 with multiple languages enabled, InfoPath checks the spelling in your form one language at a time. If you want to check text in another language, you must use the following procedure to switch to the dictionary for that language. You can repeat this procedure for each language for which you want to check spelling.

- 1. On the **Tools** menu, click **Set Language**.
- 2. In the **Language** dialog box, select the language whose spelling you want to check, and then do one of the following:
- Check spelling automatically as you type
- Check spelling all at once

Notes

If spelling options are not available for the language you have selected, InfoPath may require files that are available in the Microsoft Office 2003 Proofing Tools.

When you fill out a form, the spelling checker checks only text that is typed into a field that allows a spelling check and does not check text that is a part of the form's design.

When designing a form, you can specify where in the form users are allowed to check their spelling.

Turn on or off German spelling rules

To check the spelling of German text, you must install the German spelling dictionary, which is available in the <u>Microsoft Office 2003</u> <u>Proofing Tools</u>.

- 1. On the **Tools** menu, click **Options**, and then click the **Spelling** tab.
- 2. Select or clear the **Use German post-reform rules** check box.

About using ink in InfoPath

If you are using a Tablet PC, you can add ink to your Microsoft Office InfoPath 2003 forms. You can write directly in fields on the form by using a tablet pen and then convert that ink to text. Depending on the design of your form, you can also write or draw inside certain controls and then save the ink along with your form.

- ▶ Fill out forms in ink
- Design forms for Tablet PC users

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Switch between typing and writing

When you run Microsoft Office InfoPath 2003 on a Tablet PC, InfoPath automatically starts in ink entry mode. This allows you to write directly on the form with a tablet pen. If you want to type on a keyboard instead, you can turn off ink entry mode.

Note You must be running InfoPath on a Tablet PC to use this feature.

On the **Ink** toolbar, do one of the following:

To type instead of write, tap **Stop Ink Entry**.

To write instead of type, tap **Start Ink Entry**.

▶ Tip

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Add handwritten notes or drawings to a form

If the form you're filling out contains a rich text field, you can insert an ink drawing and writing section inside that field and then write or draw inside that section with a tablet pen.

Note You must be running Microsoft Office InfoPath 2003 on a Tablet PC to use this feature.

- 1. Tap in the rich text field where you want to write or draw with your tablet pen.
- 2. On the Ink toolbar, tap Insert Ink Drawing and Writing .
- 3. Write or draw in the rich text field.
- 4. Do one of the following:
- To replace the ink drawing and writing section with typed text, tap **Convert Ink to Text** on the **Tools** menu.
- To copy the ink to the clipboard as typed text, tap Copy Ink as Text on the Edit menu.

Notes

Depending on the design of your form, you also may be able to write or draw directly on top of pictures on your form.

The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

About symbols

You can insert symbols that don't appear on your keyboard into your form. For example, you can insert symbols such as ¼ and ©, as well as international characters, such as Ç and ë, into your form.

You insert symbols and characters into a form by using the **Symbol** dialog box. You can also use the **Symbol** dialog box to insert Unicode characters and to find the character code for Unicode characters.

The types of symbols and characters that you can insert depend on the available fonts that you have installed on your computer. For example, some fonts may include fractions ($\frac{1}{4}$), international characters (\mathcal{C} , \mathcal{E}), and international monetary symbols (£, ¥). The Symbol font that comes with your version of Microsoft Windows includes arrows, bullets, and scientific symbols. You might also have additional symbol fonts that include decorative symbols.

▶ Tip

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Insert a symbol

You can use the **Symbol** dialog box to insert symbols that are not on your keyboard, such as ¼ and ©, into a form. You can also use the **Symbol** dialog box to insert Unicode characters into a form.

Note If you're using an expanded font, such as Arial or Times New Roman, you can use the **Subset** list to choose from an extended list of language characters, including Greek and Russian (Cyrillic), if available.

- Insert a symbol
- Insert a Unicode character

Notes

When filling out a form, you can use the procedure above to insert symbols in any field into which you can type text.

The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Insert the euro and other currency symbols

Microsoft Office InfoPath 2003 provides full support for inserting the euro € and other currency symbols in a form. You can use the **Symbol** dialog box to insert the euro and other currency symbols.

- 1. Click where you want to insert the symbol.
- 2. On the **Insert** menu, click **Symbol**.
- 3. In the **Font** box in the **Symbol** dialog box, click a font that supports the symbol you want.
- 4. In the **Subset** box, click **Currency Symbols**.

Note If the selected font doesn't support subcategories, you won't see the **Subset** box.

- 5. In the from box, click Unicode (hex).
- 6. Double-click the currency symbol you want.

The name of the selected symbol is displayed in the lower-left corner of the **Symbol** dialog box.

▶ Tip

Notes

If your printer does not have the euro currency symbol installed as one of its fonts, a box will be printed instead of the euro currency symbol. Contact your printer vendor to find out how to update your printer fonts to include the euro currency symbol.

The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

About saving and publishing forms

In Microsoft Office InfoPath 2003, there are several ways you can save and publish a form, depending on whether you are filling out a form or designing a form template.

Saving and submitting forms you are filling out

When filling out a form, you can save the form (.xml file) to your computer or to any shared location, such as your company's intranet. Depending on the form's design, you might be able to save the form to a Microsoft Windows SharePoint Services form library.

Saving a form to your computer allows you to work at your own pace. For example, if you are filling out a Purchase Order form that is located on your company's network and are unable to complete it before you need to disconnect from the network, you can save it to your hard disk and continue to fill it out while you are offline. After you finish filling out the form, you can save it back to the network or shared location. Saving a form to a shared location provides a simple way for other users to access and view the form.

In some cases, the form's design may prevent you from saving it. In those cases, it might be possible only to submit the form. Depending on the form's design, you might be able to submit the form to a database, Web service, SharePoint site, or as an e-mail message.

Saving and publishing form templates

When designing a form, you are actually creating a form template (.xsn file). You can save the form template to your computer, which enables you to design it at your own pace.

A form template is composed of several individual files, called form files. To view and work with the form template's form files, click **Extract Form Files** on the **File** menu. This saves the form files to a folder where you can manually change the individual files.

When you finish designing a form template, you can publish it to a shared folder on your computer or a network, to a form library, or to a Web server (http:// or https://). After you publish the form template, users can fill out forms that are based on it. These forms are then associated with the form template, and any changes to the form template will affect the existing forms. For example, if a field is removed from a form template, existing forms based on that form template will no longer contain that field, even if the field was previously filled in.

Note Regardless of where and how you publish a form, users must have InfoPath installed on their computers before they can fill out the form.

Unless the form template is digitally signed and operating with a security level of Full Trust, it can be difficult to move the form template after it has been published. In particular, moving a form template can prevent any existing forms that are based on that form template from opening. Therefore, it is recommended that you publish the form template to its final location before you have users fill out forms that are based on it.

Save a form you've filled out as a Web page

When filling out a form, the Export to Web feature lets you save your form as a Single File Web Page (MHTML). This format allows other users to view your form in a Web browser, but it does not allow them to edit the data in the form.

- 1. Open the form you want to save as a Web page.
- 2. If there are multiple views of the form, switch to the view you want to save.
- On the File menu, point to Export To, and then click Web.
 Note The Export To command is unavailable when designing or previewing a form.
- 4. In the **Export to Web** dialog box, select the location where you want to save the form.
- 5. In the **File name** box, type a name for the form, and then click **Export**.
- 6. Repeat the previous steps for each view of the form you want to save as a Web page.

Send a form in an e-mail message

Do one of the following:

- ▶ Send a form that you filled out
- Send a form that you designed

Save a form

When you save a form you have filled out, it is saved with an .xml file extension. When you save a form in design mode, you are actually saving a form template, which is saved with an .xsn file extension.

- ▶ Tip
- Save when filling out a form
- Save when designing a form

Submit a form

Depending on the form's design, you can submit Microsoft Office InfoPath 2003 forms that you've filled out to a Microsoft Windows SharePoint Services form library, database, Web service, or as an attachment in an e-mail message. This functionality only works if the form has been designed such that submit is enabled.

After you have finished filling out your form, click **Submit** on the **File** menu, or click the **Submit** button in the form, if it is available.

Security Before submitting a form you have filled out, make sure that the form you received is from a trusted source.

Add rows or columns to a table

When designing a form, you can add rows or columns to a layout table and columns to a repeating table. When filling out a form, you can add rows to tables that are part of the form template, or you can add rows and columns to tables that you inserted into rich text fields.

Note When designing a form, you cannot add rows to a repeating table. A row in a repeating table stores data only for fields that are part of the group that the repeating table is bound to. If you want to change the layout of controls in a repeating table by adding rows, you can insert a layout table into a cell in the repeating table and then insert controls into that layout table.

- Add rows or columns to a layout table
- Add columns to a repeating table (automatically create the data source)
- Add columns to a repeating table (bind to existing fields)

Align a table

When designing a form, you can align layout tables or repeating tables on the left side, right side, or center of a form. When filling out a form, you can align tables that you have inserted into certain types of rich text fields.

- 1. Right-click the table that you want to align, and then click **Table Properties** on the shortcut menu.
 - When filling out a form, this option may be unavailable for a table, depending on the design of the form.
- 2. On the **Table** tab in the **Table Properties** dialog box, click the option that you want under **Horizontal alignment**.

Align the contents of a table cell

You can change the horizontal alignment—left, center, or right—and the vertical alignment—top or bottom—of the contents of a table cell.

Note To align the contents of a control that is in a table cell, in design mode, double-click the control, and then select the options that you want in the *Control* **Properties** dialog box.

- ▶ Change the horizontal alignment of the contents of a table cell
- ▶ Change the vertical alignment of the contents of a table cell

Change the cell padding of table cells

By changing the cell padding of a table cell, you can adjust the space between the border and contents of that cell. When designing a form, you can change the cell padding of the cells in layout tables and repeating tables. When filling out a form, you can change the cell padding of cells in the tables that you inserted into certain types of rich text fields.

- 1. Right-click the cell that you want to change the cell padding for, and then click **Table Properties** on the shortcut menu.
 - When filling out a form, this option may be unavailable for a table, depending on the design of the form.
- 2. In the **Table Properties** dialog box, click the **Cell** tab.
- 3. Under **Cell padding**, enter the specific measurements that you want for the cell padding.

Copy or move a table or its contents

When designing a form, you can copy or move all or part of a layout table or a repeating table. When filling out a form, you can copy or move a table that you inserted into a rich text field.

Note Copying and pasting repeating tables may cause binding issues with the corresponding fields or groups.

- 1. Select the table, cells, rows, or columns that you want to copy or move.
- 2. Do one of the following:
- To copy the selection, click **Copy** on the **Standard** toolbar.
- To move the selection, click **Cut M** on the **Standard** toolbar.
- 3. Place the insertion point where you want to insert the selection, and then click **Paste** and the Standard toolbar.

Delete a table

When designing a form, you can delete layout tables and repeating tables from the form. When filling out a form, you can delete tables that you inserted into certain types of rich text fields.

Note If you delete a repeating table, or only specific parts of a table, from a form template, and existing forms (.xml files) are based on that form template, data in those forms may be lost.

- 1. Place the insertion point in the table that you want to delete.
- 2. On the **Table** menu, point to **Delete**, and then click **Table**.

Delete rows or columns from a table

When designing a form, you can delete rows and columns from layout tables and repeating tables. When filling out a form, you can delete rows or columns from tables that you inserted into certain types of rich text fields.

Note If you delete a row or column from a repeating table in a form template, and existing forms (.xml files) are based on that form template, data in those forms may be lost.

Select the row or column that you want to delete, and then press DELETE.

Resize all or part of a table

When designing a form, you can resize all or part of a layout table or a repeating table. When filling out a form, you can resize all or part of a table that you inserted into certain types of rich text fields.

Do one of the following:

- Resize an entire table
- ▶ Change the width of a column
- Change the height of a row

Split or merge table cells

When designing a form, you can split a table cell in a layout table into two or more cells. You can also merge two or more table cells into a single cell. When filling out a form, you can split or merge cells in a table that you inserted into a rich text field.

Do one of the following:

- ▶ Split a table cell into multiple cells
- Merge multiple table cells into a single cell

Add a form to a form library

You can add forms to a Microsoft Windows SharePoint Services form library by filling out a blank form based on the form template associated with that form library, or by uploading previously filled out forms.

- 1. By using a Web browser, navigate to the form library where you want to add a new form.
- 2. To add a new form, do one of the following:
 - Fill out a new, blank form
 - Upload a previously filled out form

Export data from a form library to a spreadsheet

If you work with Microsoft Office InfoPath 2003 forms in a Microsoft Windows SharePoint Services form library, you can export the data that is displayed in form library columns to a Windows SharePoint Services-compatible spreadsheet program, such as Microsoft Office Excel 2003.

- 1. By using a Web browser, navigate to the form library.
- 2. Under **Actions** on the page that displays the form library, click **Export to spreadsheet**.

Note If you do not have a Windows SharePoint Services-compatible spreadsheet program installed on your computer, you won't see the **Export to spreadsheet** link in the form library.

Merge forms in a form library

Many Microsoft Windows SharePoint Services form libraries let you merge data from multiple Microsoft Office InfoPath 2003 forms into one form.

Note This option may not be available for all form libraries. The form designer may have turned off form merging for the form, or the form library view may be customized such that it excludes the link.

- 1. By using a Web browser, navigate to the form library.
- 2. Under **Select a View** on the page that displays the form library, click **Merge Forms**.
- 3. In the **Merge** column in the **Merge Forms** view, select the check box next to each file you want to merge, and then click **Merge Forms**.
 - ▶ Tip

About digital signatures

When designing a form, you can enable digital signatures so that users filling out your form can digitally sign it. This signature confirms that the form originated from the signer and has not been altered. In addition, the signature can contain a comment from the signer. After being signed, the values in the form cannot be modified without invalidating the signature.

When adding a digital signature, the user must use a digital certificate. Digital certificates, which you can obtain through commercial certification authorities or from your internal security administrator, establish the authenticity of the signature.

In addition to enabling digital signatures so that users can sign your form, you can digitally sign your form template, which authenticates you as the author of the form template in the same way that a digital signature on a form authenticates the user who filled out the form. By doing so, you can enable scenarios that require greater trust for the form. For example, form templates that have been distributed to users in an e-mail message can be updated more efficiently if they have been digitally signed.

Note Digital signatures in Microsoft Office InfoPath 2003 make use of XML Signatures.

Digital signature options

When designing a form, you can enable digital signatures for the entire form or for specific parts of the form. If you enable digital signatures for part of the form, you must determine which data in the form should be signed. You can additionally associate that data with a section in the form. When users sign the form, the digital signature locks the data so that it cannot be changed, and stores the view with the digital signature.

In addition, you can specify whether to allow multiple digital signatures for a form, and whether those signatures should be co-signed (in which case each signature is independent of the other signatures) or counter-signed (in which case each signature signs the form, as well as the signatures that precede it).

Note You cannot enable digital signatures for a form that was designed based on an XML Schema that does not have a digital signature

namespace.

Digital certificates

When you digitally sign a form, InfoPath uses only those certificates that have a private key and a **Digital Signature** or **Both** value for the **Key Usage** attribute. In addition, the purpose of the certificate must be set as **Client Authentication** or **Code Signing**. (If you are using a certificate to digitally sign a form template, the certificate must be set as **Code Signing**.) These restrictions apply because InfoPath uses XML Signatures to digitally sign forms.

Note Because a digital certificate you create is not issued by a formal certification authority, forms signed using a certificate you created are referred to as self-signed forms. These certificates are considered unauthenticated and will generate a security warning if the form's security level is set to High or Medium. InfoPath trusts self-signed certificates only on computers that have access to the private key for that certificate. In most cases, this means that InfoPath trusts self-signed certificates only on the computer that created the certificate, unless the private key is shared with other computers.

There are two types of certification authorities, commercial certification authorities and internal certification authorities.

- Commercial certification authorities
- Internal certification authorities

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Add, remove, or review digital signatures when filling out a form

When filling out a form, you can add a digital signature to part or all of the form, depending on the form's design. You can digitally sign forms only if they have been designed to accept digital signatures. You can also remove or review digital signatures.

Once signed, the form or part of the form that was signed cannot be altered without invalidating the signature.

- Add a digital signature
- ▶ Remove a digital signature
- Review digital signatures

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Request a digital certificate

In order to digitally sign a form, you must have a digital certificate installed on your computer. You can request a digital certificate from the default certification authority on your company's intranet.

- 1. In Microsoft Windows XP or Microsoft Windows 2000, on the **Start** menu, click **Run**.
- 2. In the **Open** box in the **Run** dialog box, type **MMC**, and then click **Enter**.
- 3. On the **File** menu in the **Console** window, click **Add/Remove Snap-in**.
- 4. In the Add/Remove Snap-in dialog box, click Add.
- 5. In the **Add Standalone Snap-in** dialog box, click **Certificates**, and then click **Add**.
- 6. In the **Certificates snap-in** dialog box, click **My user account**, and then click **Finish**.
- 7. In the Add Standalone Snap-in dialog box, click Close.
- 8. In the Add/Remove Snap-in dialog box, click OK.
- 9. In the **Console** dialog box, click **Certificates Current User**, and then, under **Logical Store Name**, double-click **Personal**.
- 10. Under **Object Type**, right-click **Certificates**, point to **All Tasks**, and then click **Request New Certificate** on the shortcut menu.
- 11. Follow the steps in the Request Certificate Wizard.

Note You can also obtain a digital certificate from a certification authority such as VeriSign, Inc. or from your system administrator.

Graphics file types InfoPath can use

You can insert many popular graphics file formats into your Microsoft Office InfoPath 2003 forms, either directly or with the use of separate graphics filters. You don't need a separate graphics filter installed to insert the following file formats:

Enhanced Metafile (.emf)

Graphics Interchange Format (.gif, .gfa)

Joint Photographic Experts Group (.jpg, .jpeg, .jfif, .jpe)

Portable Network Graphics (.png)

Microsoft Windows Bitmap (.bmp, .dib, .rle, .bmz)

Windows Metafile Graphics (.wmf)

Tagged Image File Format (.tiff)

Encapsulated PostScript (.eps)

File types that require filters

You need a graphics filter installed to insert all other graphics file formats. The separate graphics filters are not supplied with the stand-alone version of Microsoft Office InfoPath 2003. To use the filters, you need to install a Microsoft Office System edition or the stand-alone version of Microsoft Office Word with your stand-alone version of InfoPath. The following list contains common file types and the corresponding graphic filters that are supported.

- ▶ Computer Graphics Metafile (.cgm)
- CorelDRAW (.cdr, .cdt, .cmx, .pat)
- Encapsulated PostScript (.eps)
- ▶ Hanako (.jsh, .jah, .jbh)
- Macintosh PICT (.pct, .pict)
- WordPerfect Graphics (.wpg)
- Other filters

Position a picture relative to text

- 1. Right-click the picture, click **Format Picture** on the shortcut menu, and then click the **Text Wrapping** tab.
- 2. Click the text wrapping style you want.

Provide alternative text for a picture

You can specify alternative text to appear instead of pictures in your form.

- 1. Right-click the picture, click **Format Picture** on the shortcut menu, and then click the **Text** tab.
- 2. In the **Alternative text** box, type the text you want. The text can be as long as you want, but some Web browsers might be able to display only a limited number of characters.

Resize a picture

You can resize pictures by using either a mouse or specific measurements.

Note When filling out a form, you cannot resize pictures that were inserted into a picture field.

- ▶ Resize a picture by using the mouse
- Resize a picture by using specific measurements

About form layout

In Microsoft Office InfoPath 2003, a form's visual structure, or layout, is created primarily through the use of layout tables. Among other things, you can use layout tables to organize controls or to arrange logos and other graphics on a form. After you insert layout tables onto a form, you can type text, such as control labels, into them and insert the controls with which your users will interact.

The **Layout** task pane provides a collection of layout tables that you can place onto a blank form. If you want to add or remove rows and columns from a layout table, click any of the options in the **Merge and split cells** list in the task pane. If the layout tables in the **Layout** task pane don't meet your needs, you can insert a custom layout table with the exact number of rows and columns that you want.

Layout tables can be formatted and modified like tables in other Microsoft Office System programs. Use the **Table** menu to format a selected table, or right-click a table to set its properties.

▶ Tip

Layout-related controls

In addition to layout tables, you can add layout-related controls, such as sections or repeating sections, to a form. When you insert a layout-related control onto your form, you are essentially inserting an empty container for storing other controls. You can use the **Controls** task pane to identify and insert controls into a layout-related control. The following provides brief descriptions of the layout-related controls that are most frequently used and why you might use them:

Optional section This type of control does not appear on a form and is not saved with a form unless a user chooses to add it. For example, although all employees in a company might use a goal setting section in their Performance Review forms, only managers might choose to add an optional section about leadership goals.

Repeating control This type of control, such as a repeating section or repeating table, lets users expand the contents of a form when it is filled out and display only the necessary number of entries in a series. For example, itemized expenses in an Expense Report form can be formatted as a repeating table. This allows the user who is filling out the form to insert additional table rows for the appropriate number of items.

Scrolling region This type of control contains other controls, retains a fixed size, and includes scroll bars so that users can scroll to see information that is out of view. Scrolling regions can be useful when a section of a form contains a lot of data, and users do not need to see all of the data at one time.

Choice group This type of control presents a set of mutually exclusive choices to users. A choice group contains choice sections, one of which appears as the default choice in a form, and those choice sections contain other controls. For example, address information in an Employee Information form can be formatted as a choice group that contains choice sections. Each choice section contains controls with the correct address syntax for a specific country or region. When employees fill out the form, they can replace the default address section with one that applies to their country or region.

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Understanding key design mode icons

Have you designed or customized a Microsoft Office InfoPath 2003 form and wondered what all of those different symbols meant? This topic will help you understand the key icons used in InfoPath design mode.

Data source icons

Understanding the different icons in the **Data Source** task pane will help you better understand how all of the different pieces of data stored in fields and groups are related to one another.

Icon Description

- Group. An element in the data source that can contain fields and other groups.
- Locked group. A group that cannot be renamed, moved, or deleted, and new fields or groups cannot be added to it.
- Repeating group. A group in the data source that can occur more than once.
- Locked repeating group. A group that cannot be renamed, moved, or deleted, and new fields or groups cannot be added to it.
- Element field. A field in the data source that stores the data entered into controls, and can contain attribute fields.
- Locked element field. A field that cannot be renamed, moved, or deleted, and new attributes cannot be added to it.
- Repeating element field. A field in the data source that can occur more than once. Controls such as lists, repeating sections, and repeating tables can be bound to repeating fields.
- Locked repeating element field. A field that cannot be renamed, moved, or deleted, and new attributes cannot be added to it.
- Attribute field. A field in the data source that stores data entered into controls and other fields cannot be added to it.
- Locked attribute field. A field that cannot be moved or deleted.
- * Required field. If a control on the form is bound to a required field, it cannot be left blank when a user fills out the form.

Binding icons

When you design a form, several icons may appear in the upper-right corner of controls. These are known as binding icons. They provide information about how a selected control relates to fields and groups in the data source. The following table describes the three types of binding icons you may see on your form.

Icon Description

- The binding is correct. This icon, along with the name of the field or group that the selected control is bound to in the data source, is displayed when a control is selected or when the mouse pointer rests on the control.
- The control may not function as expected. This icon is displayed regardless of whether the control is selected. The name of the field or group that the control is bound to, along with a short message, is displayed when the control is selected, or when the mouse pointer rests on the control. To identify the problem, right-click the control, and then click **More Details** on the shortcut menu.
- The control will not function correctly, and when a user saves the form, any data typed into the control will be lost. This icon is displayed regardless of whether the control is selected. To identify the problem, right-click the control, and then click **More Details** on the shortcut menu.

Other InfoPath icons

The following table lists a few other InfoPath icons that you should be familiar with.

Icon Description

- Repeating control, field, or group. A control, field, or group that can appear more than once. This icon appears both in design mode and when users fill out a form.
- Optional section. A control on a form that can contain other controls and does not usually appear by default. This icon appears both in design mode and when users fill out a form.
- Choice section. A control on a form that contains other controls and that is used to represent a single choice within a set of mutually exclusive choices. This icon appears both in design mode and when users fill out a form.
- InfoPath form (.xml file). A file that contains the data that users enter into a form and is created after users save their form. This icon appears in task panes and dialog boxes, such as the **Open** dialog box.
- InfoPath form template. A file or set of files that defines the data structure, appearance, and behavior of a form. This icon appears in task panes and dialog boxes, such as the **Open** dialog box.

Import a file to use as a form template

You can use importers to import files from other programs, and then use those files as form templates. This functionality depends on the particular importer. In some cases, after the form template has been imported, you may need to manually create the data source for the new form template, and then rebind the controls in the form.

- 1. In design mode, click **Import Form** on the **File** menu.
- 2. In the **Select the import method to use** list on the first page of the Import Wizard, do one of the following:
- To use a previously installed importer, click the importer, and then click Next.
- To browse for a new importer, click **Browse for a new importer**, and then click **Next**.
 - **Note** To browse for an importer, your computer must be connected to the Internet.
- 3. Follow the instructions in the Import Wizard to finish importing a file.

Add a resource file to a form template

Resource files can include HTML pages for a custom task pane or an XML document that provides the list items for a drop-down list box. To make sure that all your users have access to resource files, you can add them to your form.

- 1. In design mode, on the **Tools** menu, click **Resource Files**.
- 2. In the **Resource Files** dialog box, click **Add**.
- 3. In the **Add File** dialog box, select the file you want to add.

Show or hide a page width guide

To quickly determine whether the contents of a form will fit on a printed page, you can display a page width guide along the right side of the form in design mode. The guide is a dashed vertical line that indicates the right margin of the printed page, based on the default printer for your computer. If you have not yet set up a printer for your computer, you must do so before you can show or hide a page width guide.

In design mode, click Page Width Guide on the View menu.

Specify a custom layout width

By specifying a custom layout width for a form, you can override the default width at which layout tables and layout-related controls, such as sections, are inserted in the form area. By default, Microsoft Office InfoPath 2003 inserts these items at a width that matches that of the printed page, based on the default printer for your computer.

If you created different views of a form, you can specify a different layout width for each view.

- 1. In design mode, click **Manage Views** on the **View** menu.
- 2. In the **Views** task pane, click the view for which you want to specify a custom layout width, and then click **View Properties**.
- 3. On the **General** tab of the **View Properties** dialog box, under **Layout settings**, select the **Use a custom layout width for this form** check box.
- 4. In the **Custom layout width** box, type a value, and then select a unit of measure from the list.

▶ Tip

Enable or disable a custom task pane

A custom task pane is an .html file whose content is displayed in a Microsoft Office InfoPath 2003 task pane when a user fills out a form. Custom task panes can provide form-specific commands and Help content. You can add a custom task pane file directly to your form, so that users can work with the form and task pane while offline, or you can link the custom task pane to the form from another location.

Note Before enabling a custom task pane, you must first create an .html file by using an HTML editor such as Microsoft Office FrontPage.

- ▶ Enable a custom task pane that is part of the form
- ▶ Enable a custom task pane that is linked to the form
- Disable a custom task pane

Enable or disable the Save, Export, Print, or Send command

For each form you design in Microsoft Office InfoPath 2003, you can decide whether to enable automatic file recovery, which allows InfoPath to save the form automatically as the user works. You also control whether to enable users to save the form data, print it, send it in an email message, or export the data to Microsoft Office Excel 2003 or a Single File Web Page (.mht) file.

- 1. On the **Tools** menu, click **Form Options**, and then click the **Open and Save** tab.
- 2. Under **Enable features**, select or clear the check boxes for the features you want to enable or disable.

Set the default values for a form

The default values for a form are the initial values that the form contains. They can be standard text or formulas. Default values are stored in fields and are displayed in the controls that are bound to those fields.

Note Fields that do not appear in the data source by default are not the same as fields with blank values. If a field does not appear in the data source by default, the control that is bound to that field should be an optional section or part of an optional section.

- 1. In design mode, click **Form Options** on the **Tools** menu, and then click **Edit Default Values**.
- 2. In the **Edit Default Values** dialog box, select the field whose default value you want to set.

Note To change the default value for a field, the check box for the field must be selected and the field name itself must also be selected. A field with a cleared check box will not initially appear in the data source.

- 3. Do one of the following:
- To add a standard default value, type the default value for the field in the **Default value** box.
- To add a formula that calculates the default value, click **Insert**Formula , and then, in the **Insert Formula** dialog box, create the formula. To automatically update the value in the selected field each time the formula is calculated, select the **Update this** value when the result of the formula is recalculated check box.
- 4. If a field is a repeating field or repeating group and you want more than one instance of the field to initially appear on the form, right-click the repeating field or group, and then click Add another repeating field or group name above or Add another repeating field or group name below on the shortcut menu, depending on where you want the additional instance to appear in relation to the selected instance.

5. Repeat steps 2 through 4 for each field whose default value you want to set.

Change the default tab order for controls

When filling out a form, many people use a keyboard, instead of a mouse, to move from field to field. Therefore, it's important that controls in your form have a logical tab order.

By default, the tab order in a Microsoft Office InfoPath 2003 form is left to right, top to bottom, just like it is on a Web page. If you need to change the default tab order for your form, use the following procedure. For example, you can specify a custom tab order for a repeating table, so that users can navigate column by column rather than row by row.

- In design mode, right-click the control for which you want to change the tab order, and then click **Control Properties** on the shortcut menu.
- 2. In the *Control* Properties dialog box, click the **Advanced** tab.
- 3. In the **Tab index** box, specify something other than 0. For example, to specify that users navigate to this control the first time that they press the TAB key, enter **1** in the **Tab index** box.
- 4. To change the tab order for additional controls, repeat steps 1 through 3.

Notes

The default tab index setting for all controls in a form is 0, but the tab order starts with 1. That is, any control with 1 in the **Tab index** box will be visited first when users press the TAB key. Any control with 2 in the **Tab index** box will be visited second, and so on. Any controls with 0 in the **Tab index** box will come last in the tab order. If you want to skip controls in the tab order, enter **-1** in the **Tab index** box.

Changing the tab index setting doesn't change the order in which you tab through controls in design mode. To test the tab order in your form, click **Preview Form** on the **Standard** toolbar, and then use the TAB key to move through the form.

Protect a form's design

You can protect a form's design so that users who fill out the form cannot open or modify it in design mode.

- 1. In design mode, click **Form Options** on the **Tools** menu.
- 2. On the **General** tab of the **Form Options** dialog box, under **Protection**, select the **Enable protection** check box.

Note Selecting this setting prevents users from opening a form in design mode only when they are filling it out. The setting does not lock the form completely. Users can still open the form or modify the form's design by opening the form directly in design mode. However, they receive a message stating that the form is protected.

Use a table for layout purposes

You can design a professional-looking form by using layout tables. A layout table is a framework that includes rows and columns for organizing and arranging form content, including controls, sections of a form, logos, and other types of graphics. These regions provide a visual structure for the content of a form.

Microsoft Office InfoPath 2003 includes several predefined layout tables that can help you get started. You can insert these layout tables in a form and then customize them to meet your needs. If you prefer to use layout tables with specific dimensions, you can insert custom layout tables that have a specific number of rows and columns. You can also draw custom layout tables, which can be helpful if you want to create a more complex layout table. For example, you can create a layout table that contains cells of different heights or a varying number of columns per row.

- Insert a predefined layout table
- Insert a custom layout table with specific dimensions
- Draw a custom layout table

Notes

In design mode, the borders of layout tables appear as dashed gridlines. When users fill out a form, these gridlines do not appear on the form. If you prefer to hide the gridlines in design mode, you can turn them off by clicking **Hide Gridlines** on the **Table** menu.

Add a border around a form

You can add a border around an entire form by placing the layout and controls on the form into a single-cell layout table and then adding a border to that cell.

- 1. In design mode, press CTRL+A to select the contents of the entire form, and then press CTRL+X to cut the selection and place it on the clipboard.
- 2. On the **Insert** menu, click **Layout Table**.
- 3. In the **Insert Table** dialog box, enter **1** in the **Number of columns** and **Number of rows** boxes, and then click **OK**.
 - A single-cell layout table is inserted into the form.
- 4. Right-click the layout table, and then click **Borders and Shading** on the shortcut menu.
- 5. In the **Borders and Shading** dialog box, click the **Borders** tab.
- 6. If you want to specify formatting options for the border, do any of the following:
- To specify a custom line style, click the style that you want in the Style list.
- To specify a color, click the **Color** arrow, and then click the color that you want.
- To specify a custom width, click the width that you want in the Width list.
- 7. Under **Presets**, click **Outline**, and then click **OK**.
- 8. Place the insertion point inside the layout table, and then press CTRL+V to paste the form contents into the layout table.

Add or remove borders and shading

Borders and shading can add interest and emphasis to a form. You can add borders and shading to many parts of a form, including tables and table cells, controls, and objects such as pictures.

- Add a border
- Remove a border
- Add shading
- Remove shading

Add or remove a background color from a form

You can set a background color for an entire form. Background colors can create contrast and interest when users view a form on the screen.

- 1. On the Format menu, click Background Color.
- 2. Under **Background** in the **View Properties** dialog box, click the **Background color** arrow, and then do one of the following:
- To add a background color, click the color that you want to add.
- To remove a background color, click **Automatic color**.

Tips

About color schemes

Color schemes offer a quick and easy way to create professional-looking forms by using Microsoft Office InfoPath 2003. In design mode, when you apply a color scheme to a form, InfoPath customizes certain elements of the form layout with coordinated colors that are based on that color scheme. Depending on the design of the form, those elements can include body and heading styles, table cells, and table borders. To change the color of these elements, click any of the available color schemes in the **Color Schemes** task pane.

Apply or remove a color scheme

Applying a color scheme is one of the easiest ways to enhance the appearance of your form. Microsoft Office InfoPath 2003 provides a variety of color schemes that apply coordinated colors to certain items on your form. Because color schemes can affect different items on a form in different ways, it's a good idea to try several color schemes when you experiment with various formatting options. For example, when you select a color scheme for your form, you can apply different heading styles to text and controls to see what kinds of effects you can create.

- 1. In design mode, click **Color Schemes** on the **Format** menu.
- 2. In the **Color Schemes** task pane, do one of the following:
- To apply a color scheme, click a color scheme in the Apply a color scheme list.
- To remove the currently applied color scheme, click **None** in the **Apply a color scheme** list.

Note Color schemes affect only certain parts of a form. Depending on the design of the form, this can include body and heading styles, table cells, and table borders. If none of these items exist on your form, you may not see a visual change when you apply a color scheme.

Insert a hyperlink that opens a new e-mail message

When designing a Microsoft Office InfoPath 2003 form, you can insert a hyperlink control that opens a new e-mail message when clicked by your users. This is referred to as a mailto hyperlink. You can specify the recipient and subject when you insert the hyperlink control, or you can have the recipient and subject determined by information stored in the form's data source.

Note If you use the data source to determine the recipient and subject of the e-mail message, the hyperlink control must be bound to a specific field or fields in the data source.

- Create a mailto hyperlink (specify the recipient and subject)
- ▶ Create a mailto hyperlink (use the data source to determine the recipient and subject)

Add a background picture to your form

- 1. On the Format menu, click View Properties.
- 2. On the **General** tab in the **View Properties** dialog box, click **Use a background picture**.
- 3. In the **Add a background picture** dialog box, select the picture you want, and then click **OK**.
- 4. Select any additional options that you want.

▶ Tip

Add a picture to a form

To give a form visual interest, you can add pictures to the form. For example, you can add a corporate logo to the top of a form. You can add pictures as either embedded objects or linked objects. Embedding a picture is a good choice if you want to keep the form and its data portable. Linking to a picture is a better choice if you think you will need to update the linked image, or if you want to reduce the file size of the form. If you link to a picture, users must have access to the location of the picture source file. Otherwise, the picture will not appear on the form.

- ▶ Embed a picture
- Link to a picture
- ▶ Insert clip art

Note When designing a form, you can also insert a picture control. Like rich text controls, picture controls allow users to insert pictures into a field when they are filling out a form.

About the data source

The data source, which stores all of the data entered into a form, is made up of fields and groups. In the same way that folders on your hard drive contain and organize your files, fields contain the data in your form, and groups contain and organize those fields. For example, the fields for first name, middle name, and last name can be contained in a group called "name."

You can work with your form's fields and groups in the **Data Source** task pane.



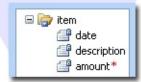
- Field
- Group

The controls on your form are bound to the fields and groups in the data source; this binding allows data entered into a control to be saved. Information entered into a bound control is saved in the field it is associated with.

Although the structure of the data source does not always exactly match the layout of your form, similarities are common, particularly for groups and fields associated with repeating tables, sections, repeating sections, and optional sections. For example, a table or section is bound to a group in the data source, and all of the controls in the table or section are bound to fields that are part of that group. In the following illustration, the Itemized Expenses repeating table consists of three columns containing a Date date picker, a Description text box, and a Cost text box. These controls are bound to a date field, a description field, and an amount field. The table itself is bound to the item group.

Itemized Expenses

Date	Description	Cost
8/26/2006 🛅	Dinner	28.45
8/26/2006 🕮	Hotel	169.95
	Subtotal	198.40
Less cash advance		0.00
Tota	l expenses	198.40



Using the Data Source task pane to insert controls

You can insert controls by dragging fields and groups from the **Data Source** task pane onto your form. Each time you drag a field or group onto your form, Microsoft Office InfoPath 2003 suggests the control that best matches that field or group— that is, the control that accepts the type and structure of data associated with the field or group. If the field has a date data type, for example, InfoPath suggests you insert a date picker control. If the group is repeating and contains three fields, InfoPath suggests a repeating table or a repeating section.

▶ Tip

Modifying fields and groups in the data source

When working directly with the data source in the **Data Source** task pane, you can add, move, reference, and delete fields and groups. In addition, you can change the following properties of fields and groups:

Name The name of the field or group. Names cannot contain spaces. They must begin with an alphabetic character or underscore ("_"), and they can contain only alphanumeric characters, underscores, hyphens ("-"), and periods (".").

Type The type of field or group. Types include element field (the standard type of field), attribute field, group, choice group, and complete XML document or Schema. Fields store data that is entered into controls; groups contain fields and cannot store data. Choice groups store data associated with choice group controls. You can use the complete XML document or Schema type to add XML documents or XML Schemas to a specific part of the data source.

Data type Defines the kinds of data that the field can store. Data types include Text, Rich Text, Whole Number, Decimal, True/False, Hyperlink, Date, Time, Date and Time, Picture, and Custom, which is for fields associated with Microsoft ActiveX controls. Only fields can have data types.

Default value The initial value that a field has when the form is first opened by a user. You can also add formulas to default values, such as the current date, a mathematical calculation, or a reference to another field.

Repeating Determines whether the field or group can occur more than once in a form. List controls, repeating sections, repeating tables, and controls that are part of a repeating section or repeating table are bound to repeating fields and repeating groups.

Cannot be blank Determines whether the field must contain a value. If the field cannot be blank, any control bound to the field that does not have a value entered into it will be marked with a red asterisk or, for some types of controls, a dashed red border.

Note If your users have already filled out forms based on your form template, the following changes can cause data loss in those forms:

Moving, deleting, or renaming a field or group

Changing a repeating field or group to a nonrepeating field or group

Changing a rich text data type to a different data type

If you want to store the same type of data in more than one place— such as a billing and shipping address in an invoice form— you can create matching fields and groups. When you reference a field, InfoPath creates a new, identical field whose properties are linked to the properties of the original field. Both fields become reference fields, and a change to one field updates the other field automatically. Reference groups, like reference fields, share the same properties; in addition, they contain the same fields and groups.

Understanding the relationship between the data source and an XML Schema

The forms that users fill out are stored as XML documents. When you create the data source for a form, you are actually creating an XML Schema, which is an XML-based file that determines the structure of those XML documents. Each group in the data source is an XML element that can contain other elements and attributes but not data, and each standard field in the data source is an XML element that can contain data. Each attribute field is an XML attribute. You can view additional schema details on the **Details** tab in the **Field or Group Properties** dialog box.

Existing XML Schemas, XML documents, databases, and Web services

When you design a form based on an existing XML Schema, InfoPath creates a data source that is based on that XML Schema. The more detailed your XML document is, the more detailed the resulting data source will be. Because using an existing XML Schemas can be more restrictive than designing a new, blank form, you cannot modify existing fields or groups in the data source. In addition, depending on the structure of the XML Schema, you might not be able to add fields or groups to part or all of the data source.

When you design a new form that is connected to a database or Web service, InfoPath builds a data structure for your form based on that database or Web service. The resulting form can be used to query or submit data to the database or Web service. Because the data source must match the database or Web service, you can only modify new fields or groups in the data source, and you can add fields or groups only to the root group in the data source.

Secondary data sources and data connections

In addition to the main data source for your form, you can create data connections to XML documents, databases, Web services, or Microsoft Windows SharePoint Services libraries and lists. These data connections result in secondary data sources, which can be used to populate list boxes, and provide the values for text boxes and other controls. With some exceptions, including specifying default values and direct editing of the fields and groups, you can work with secondary data sources in the same way that you work with the form's primary data source. In addition, you can query and submit data using data connections. For more information about data connections, see About data connections.

Design a form based on an existing XML document

When designing a new form based on an existing XML document, you can work with the form in the same way as when you design a new, blank form, including inserting and laying out controls, inserting and modifying fields and groups in the data source, adding data validation and conditional formatting, and publishing the form.

Note When you design a form based on an existing XML document, Microsoft Office InfoPath 2003 creates a data source based on the information in your .xml file. The more detailed your XML document is, the more detailed the resulting data source will be.

- 1. On the **File** menu, click **Design a Form**.
- 2. In the **Design a Form** task pane, click **New from XML Document or Schema**.
- 3. On the first page of the Data Source Wizard, click **Browse**.
- 4. In the **Open** dialog box, browse to the XML document you want to base the form on, and then click **Open**.
- 5. On the next page of the Data Source Wizard, click **Next**.
- 6. On the next page of the Data Source Wizard, do one of the following:
 - If you want to include an additional XML document or XML Schema in your form, click **Yes**, click **Next**, and then repeat steps 3 through 5.
- If you are finished adding XML documents or XML Schemas to your form, click No, and then click Finish.
- 7. In the dialog box that appears, do one of the following:
- To use the values in the XML document to create default data for your form, click **Yes**. The values stored in the XML document will become the default values for the fields in your form.
- To use only the structure of the XML document and not its values, click No.

8. To begin adding controls to the form, right-click a field or group in the **Data Source** task pane, and then on the shortcut menu, click the control you want to insert.

Design a form based on an existing XML Schema

When designing a new form based on an existing XML Schema (.xsd) file, you can work with the form in many of the same ways as when you design a new, blank form, including inserting and laying out controls, adding data validation and conditional formatting, and publishing the form.

The new form will automatically re-use the data validation of the existing XML Schema.

- 1. On the **File** menu, click **Design a Form**.
- 2. In the **Design a Form** task pane, click **New from XML Document or Schema**.
- 3. On the first page of the Data Source Wizard, click **Browse**.
- 4. In the **Open** dialog box, browse to the XML Schema you want to base the form on, and then click **Open**.
- 5. On the next page in the Data Source Wizard, click **Next**.
- 6. On the next page in the Data Source Wizard, do one of the following:
- If you want to include an additional XML document or XML Schema in your form, click **Yes**, click **Next**, and then repeat steps 3 through 5.
- If you are finished adding XML documents or XML Schemas to your form, click **No**, and then click **Finish**.
- 7. To begin adding controls to the form, right-click a field or group in the **Data Source** task pane, and then on the shortcut menu, click the control you want to insert.

Notes

Because existing XML Schemas can be more restrictive than new, blank forms, the following limitations apply:

You cannot automatically create the data source when inserting controls.

You cannot modify existing fields or groups in the data source.

Depending on the schema, you might not be able to add fields or groups to part or all of the data source.

Replace the data source with an updated XML Schema

When you design a form based on an existing XML document or XML Schema, and that XML document or XML Schema has been changed outside of Microsoft Office InfoPath 2003, you can update your form with the new XML document or XML Schema.

Note This procedure does not work for forms based on a database or Web service.

- In design mode, click Convert Main Data Source on the Tools menu.
- 2. On the first page of the Data Source Wizard, click **Browse**.
- 3. In the **Open** dialog box, browse to the updated XML document or XML Schema, and then click **Open**.
- 4. On the next page in the Data Source Wizard, click **Next**.
- 5. On the next page in the Data Source Wizard, do one of the following:
- If you want to include an additional XML document or XML Schema in your form, click **Yes**, click **Next**, and then repeat steps 2 through 4.
- If you are finished adding XML documents or XML Schemas to your form, click **No**, and then click **Finish**.
- 6. If a dialog box appears asking whether to use the values in the .xml file as the default data in your form, do one of the following:
 - To use the values in the XML document to create default data for your form, click **Yes**. The values stored in the XML document will become the default values for the fields in your form.
- To use only the structure of the XML document and not its values, click **No**.
- 7. Revise the form to match the new XML Schema. If an element has been removed in the updated XML Schema, for example, rebind or remove the controls that were previously bound to that element.

8. Test the form thoroughly.

Note Testing the form includes previewing the form, as well as opening a new form and filling it out. Make sure to test any controls bound to new or modified fields.

- 9. If there are existing forms based on this form template, upgrade those forms.
 - ▶ How?
- 10. To republish the form, click **Publish** on the **File** menu.

Add an XML Schema or XML document to the data source

When designing a form, you can import an XML Schema or the structure of an XML document. This results in a set of fields and groups (as defined by the imported XML document or XML Schema) being added to a specified group in the data source.

- 1. In design mode, if the **Data Source** task pane is not visible, click **Data Source** on the **View** menu.
- 2. In the **Data Source** task pane, right-click the group to which you want to add the XML Schema or XML document, and then click **Add** on the shortcut menu.

Note If your form uses more than one data source, click **Main** in the **Data source** list. You cannot add an XML document or XML Schema to a secondary data source.

- 3. In the Type list, click Complete XML Schema or XML document.
- 4. On the first page of the Data Source Wizard, click **Browse**.
- 5. In the **Open** dialog box, browse to the XML document or XML Schema, and then click **Open**.
- 6. In the Data Source Wizard, click Next.
- 7. In the Data Source Wizard, do one of the following:
- If you want to include an additional XML Schema or XML document in your form, click Yes, click Next, and then repeat steps 4 through 6.
- If you are finished adding XML documents or XML Schemas to your form, click **No**, and then click **Finish**.
- 8. If a dialog box appears asking whether to use the values in the .xml file as the default data in your form, do one of the following:
 - To use the values in the XML document to create default data for your form, click **Yes**. The values stored in the XML document will become the default values for the fields in your form.

• To use only the structure of the XML document and not its values, click **No**.

Set the default value for a field in the data source

A default value is the initial value that a field contains. It can be standard text or a formula.

Note If a field has a picture or file attachment (base64) data type, you cannot set a default value for it. Instead, you can set the default value for a control that is bound to the field.

- 1. In design mode, if the **Data Source** task pane is not visible, click **Data Source** on the **View** menu.
- 2. In the **Data Source** task pane, right-click the field whose default value you want to set, and then click **Properties** on the shortcut menu.
- 3. On the **Data** tab in the **Field or Group Properties** dialog box, do one of the following:
- To add a standard default value, type the default value for the field in the **Value** box.
- To add a formula that calculates the default value, click **Insert**Formula , and then, in the **Insert** Formula dialog box, create the formula. To automatically update the value in the control each time the formula is calculated, select the **Update this**value when the result of the formula is recalculated check box.

Change the data type

The values that can be entered into a control are determined by the data type of the field that is bound to the control. If a user types a value with the wrong data type into a control, an inline alert will appear.

You can change the data type of a field by using the control's properties for text boxes, list boxes, date pickers, check boxes, and option buttons. Other controls, such as sections, list controls, and picture controls, either do not have a data type or can store only one kind of data type. For fields that do not have a control bound to them, you can change their data type directly in the data source.

- ▶ Change the data type of a field using the control's properties
- ▶ Change the data type of a field directly in the data source

Notes

If your form is based on a sample form, pre-existing XML Schema, database, or Web service, you can change the data type only for fields that you have added to the data source. You will not be able to change the data type of existing fields.

Add a field or group

The fields and groups that you add to the data source define the type and structure of the data that users can enter into your form. If you want to match the properties of an existing field or group to a new field or group, you can also add a reference field or reference group to your form.

Note If the form you are designing is based on a sample form, preexisting XML Schema, database, or Web service, you might not be able to add new fields or groups to some of the existing fields and groups. In addition, you can add fields and groups only to the main data source, not to secondary data sources.

- Add a field
- Add a group
- Add a reference field
- Add a reference group

Move a field or group

If users have already filled out forms based on your form template, moving a field or group can cause data loss in those forms.

- 1. In design mode, if the **Data Source** task pane is not visible, click **Data Source** on the **View** menu.
- 2. If your form uses more than one data source, click **Main** in the **Data source** list. You cannot move fields or groups in a secondary data source.
- 3. In the **Data Source** task pane, right-click the field or group you want to move, and then do one of the following:
 - ▶ Rearrange the order of the field or group within its group
 - Move the field or group to a new field or group

Notes

If your form template is based on a sample form, pre-existing XML Schema, database, or Web service, you can move only the fields and groups that you have added to the data source.

Delete a field or group

Note Deleting a field or group will cause an error in any controls on your form that are bound to that field or group. In addition, if users have already filled out forms based on your form template, deleting a field or group can cause data loss in those forms.

- 1. In design mode, if the **Data Source** task pane is not visible, click **Data Source** on the **View** menu.
- 2. If your form uses more than one data source, click **Main** in the **Data source** list. You cannot delete fields or groups in a secondary data source.
- 3. Right-click the field or group you want to delete, and then click **Delete** on the shortcut menu.

Note If your form template is based on a sample form, preexisting XML Schema, database, or Web service, you can delete only the fields and groups that you have added to the data source.

Change the name of a field or group

If users have already filled out forms based on your form template, changing the name of a field or group can cause data loss in those forms.

In addition, if your form template is based on a sample form, pre-existing XML Schema, database, or Web service, you can only change the names of fields and groups that you have added to the data source.

- 1. In design mode, if the **Data Source** task pane is not visible, click **Data Source** on the **View** menu.
- 2. If your form uses more than one data source, click **Main** in the **Data source** list. You cannot change the names of fields or groups in a secondary data source.
- 3. Double-click the field or group whose name you want to change.
- 4. On the **Data** tab in the **Field or Group Properties** dialog box, type a new name for the field or group in the **Name** box.

Notes

Field or group names cannot contain spaces. They must begin with an alphabetic character or underscore ("_"), and they can only contain alphanumeric characters, underscores, hyphens ("-"), and periods (".").

The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

▶ Tip

View the details of a field or group

You can view the XML Schema properties of a field or group by using the **Field or Group Properties** dialog box.

- 1. In design mode, if the **Data Source** task pane is not visible, click **Data Source** on the **View** menu.
- 2. If your form uses more than one data source, click the one you want in the **Data source** list in the **Data Source** task pane.
- 3. Double-click the field or group whose details you want to view.
- 4. In the **Field or Group Properties** dialog box, click the **Details** tab.

About data connections

A data connection establishes a dynamic link between a form and an external data source, such as a Web service or database, that allows users to view and work with data from that data source. In particular, a data connection can be used to populate the choices in a list box; receive data from a database, Web service, Microsoft Windows SharePoint Services site, or XML document; submit data to a database, Web service, or Microsoft Windows SharePoint Services form library; or submit a form as an attachment in an e-mail message.

There are two types of data connections: those that receive data from an external data source and those that submit data externally. These two types of data connections function slightly differently based on whether they are primary data connections, which are established when the form is first designed, or secondary data connections, which are added after the form is designed.

Note When you design a form based on a database, the resulting primary data connection can both submit and receive data.

Primary and secondary data connections

You can create a primary data connection by designing a form that is based on a database or Web service. The main data source for the form is then defined by the structure and properties of the database or Web service. If you add a data connection that receives data to an existing form, however, the data connection results in a secondary data source, whose fields and groups contain the data that's received.

A secondary data source can be used in many of the same ways as a primary data source. However, with a secondary data source users cannot save the data they enter into the form to use offline. In addition, form designers cannot specify default values or custom, noncode-based data validation for a secondary data source.

Because any data source that is associated with a data connection must match the structure of the external data source, the following limitations apply to both primary and secondary data sources:

The data source can not be created automatically by inserting controls into the form.

The structure and properties of existing fields or groups in the data source cannot be modified. In primary data connections, however, you can add fields or groups to the root group in the data source.

Data connections that receive data (secondary data sources)

When you add a data connection to a form that receives data, Microsoft Office InfoPath 2003 creates a secondary data source for the form that matches the structure of the external data source. This external data source can include databases, Web services, Windows SharePoint Services form libraries and lists, and XML documents.

You can work with the form's secondary data source in the same way that you work with the form's primary data source. For example, you can create controls based on the secondary data source, and then apply conditional formatting to the controls. Depending on the options you specify when creating the data connection, the data from the data connection can be retrieved either automatically when the form is opened or when the user requests it.

Data connections that submit data

Unlike primary data connections or secondary data connections that receive data, data connections that submit data do not affect the form's data sources. Instead, they establish a connection between the form's existing data sources and an external data source, which enables the form's data to be submitted to a database, Web service, or a Windows SharePoint Services form library. You can also create a data connection that will submit your form as an attachment in an e-mail message.

In most cases, you will submit the entire form when using a submit data connection. In the case of Web services, however, you can choose to submit specific parts of the form. You can also define how you want the data to be submitted to the Web service— for example, whether you want to include the entire field, including tags, or just the contents of the field, or whether you want to send the data as a string to preserve white spaces.

Design a form based on a database

When you design a form that is connected to a database, Microsoft Office InfoPath 2003 builds a data structure for your form based on that database. The resulting form can then be used to submit data to and query the database. You can work with the form in many of the same ways as when you design a new, blank form— you can insert and lay out controls, add or remove data validation and conditional formatting, and publish the form.

Note Without the use of custom code, InfoPath can connect only to Microsoft SQL Server and Microsoft Access databases.

- 1. On the **File** menu, click **Design a Form**.
- 2. In the **Design a Form** task pane, click **New from Data Connection**.
- On the first page of the Data Connection Wizard, click Database (Microsoft SQL Server or Microsoft Access only), and then click Next.
- 4. On the next page of the wizard, click **Select Database**.
- 5. In the **Select Data Source** dialog box, browse to the SQL Server or Access database you want to base the form on, and then click **Open**.
 - **Note** If you are connecting to a SQL Server database, you may need to click **New Source**, next to the **File name** box in the **Select Data Source** dialog box, and then follow the steps in the wizard to create a remote data source connection to the SQL server.
- 6. If there is more than one table in your database, click a primary table for your form in the **Select Table** dialog box, and then click **OK**.
- 7. Add any additional tables.
 - ▶ How?
- 8. In the Data Connection Wizard, click Next.

- 9. On the next page of the wizard, type a name for the data connection, and then click **Finish**.
- 10. To add controls to the form, right-click a field or group in the **Data Source** task pane, and then click the control that you want to insert on the shortcut menu.

Notes

When you design a form that is connected to a database, the resulting data source consists of query fields and data fields. Query fields should be used to create controls that allow users to enter values that they can query the database with. Data fields are filled in with the query results, and also allow users to edit existing records and add new ones.

By default, InfoPath creates the **Run Query** button and the **New Record** button in the form when you connect it to a database. When clicked, the **Run Query** button queries the database based on the values in the form's query fields. The **New Record** button resets the form's data field values to their initial state.

By default, InfoPath enables the **Submit** command on the **File** menu for a form that can submit data to a database. Clicking the command allows users to submit the data in their completed form to the database.

Because the data structure for the form must match the database, you cannot automatically create the data source when inserting controls. In addition, you cannot modify existing fields or groups in the data source, and you can only add fields or groups to the root group in the data source.

Design a form based on a Web service

When you design a new form that is connected to a Web service, Microsoft Office InfoPath 2003 builds a data structure for your form based on that Web service. The resulting form can then be used to submit data to and query the Web service. You can work with the form in many of the same ways as when you design a new, blank form— for example, you can insert and lay out controls, add or remove data validation and conditional formatting, and publish the form.

You can design forms that receive data from a Web service, submit data to a Web service, or do both. If you are designing a form that does both, you can specify different Web services for receiving and submitting data.

Note InfoPath cannot connect to a Web service that uses remote procedure call (RPC) encoding. Only document literal encoding is supported.

- Design a form that submits data to a Web service
- Design a form that receives data from a Web service
- Design a form that receives data from and submits data to a Web service

Notes

When you design a form that is connected to a Web service, the resulting data source consists of query fields and data fields. Query fields should be used to create controls that allow users to enter values that they can query the Web service for. Data fields are filled in with the query results, and also allow users to edit existing data and add new data.

By default, InfoPath adds a **Run Query** button to a form that receives data from a Web service. When clicked, this button queries the Web service based on the values in the form's query fields.

By default, InfoPath enables the **Submit** command on the **File** menu for a form that can submit data to a Web service. Clicking the command allows users to submit the data in their completed form to the Web service.

Because the data structure for the form must match the data structure of

the Web service it is connected to, you cannot automatically create the data source for the form when inserting controls. In addition, you cannot modify existing fields or groups in the form's data source, and you can only add fields or groups to the root group in the form's data source.

Connect a form to a SharePoint site

You can create a data connection in your Microsoft Office InfoPath 2003 form that receives data from a Microsoft Windows SharePoint Services site or submits your form to a Windows SharePoint Services form library.

- ▶ Receive data from a Windows SharePoint Services site
- Submit data to a Windows SharePoint Services form library

Connect a form to an external XML document

When you add a data connection that receives data from an external XML document or data source to your Microsoft Office InfoPath 2003 form, Office InfoPath 2003 creates a secondary data source for your form. This data source lets you populate list boxes with values from the XML document, and bind controls to the fields and groups associated with the XML document. In general, you can work with the secondary data source in many of the same ways that you work with your form's primary data source.

Note The values associated with a secondary data source cannot be saved when a user fills out a form, although they can be submitted to a Web service using a submit data connection. In addition, you cannot edit the properties or structure of the fields and groups associated with a secondary data source, nor can you specify default values or custom, noncode-based data validation for those fields.

- 1. In design mode, click **Data Connections** on the **Tools** menu.
- 2. In the **Data Connections** dialog box, click **Add**.
- 3. In the Data Connection Wizard, click **Receive data**, and then click **Next**.
- 4. On the next page of the wizard, click **XML document**, and then click **Next**.
- 5. On the next page of the wizard, type the location of the XML document, and then click **Next**.
- 6. On the next page of the wizard, type a name for the data connection.
- 7. If you want to automatically receive data whenever the form is opened by a user, select the **Automatically retrieve data when form is opened** check box, and then click **Finish**.
 - **Note** If the form does not automatically receive data, you may want to insert a button that retrieves the data when clicked.
- 8. If a dialog box that asks if you want to add the XML document to your form opens, do one of the following:

- To include the XML document as a resource file in your form, which will always let users have access to the file, click **Yes**.
- To prevent the XML document from being added as a resource file, which will require that users have access to both the form and the location of the file, click **No**.

Connect an existing form to a database

When you add a data connection that receives data from a database or other type of data source to your Microsoft Office InfoPath 2003 form, you are creating a secondary data source for your form. This data source lets you populate list boxes with values from the database, and bind controls to the fields and groups associated with the database. In general, you can work with the secondary data source in many of the same ways that you work with your form's primary data source.

Notes

The values associated with a secondary data source cannot be saved when a user fills out a form, although they can be submitted to a Web service using a submit data connection. In addition, you cannot edit the properties or structure of the fields and groups associated with a secondary data source, nor can you specify default values or custom, noncode-based data validation for those fields.

To submit a form to a database, the structure of the form must match the structure of the database. To do this, you must design a form that is based on the database.

Without the use of custom code, InfoPath can connect only to Microsoft SQL Server and Microsoft Access databases.

- 1. In design mode, click **Data Connections** on the **Tools** menu.
- 2. In the **Data Connections** dialog box, click **Add**.
- 3. In the Data Connection Wizard, click **Receive data**, and then click **Next**.
- 4. On the next page of the wizard, click **Database (Microsoft SQL Server or Microsoft Office Access only)**, and then click **Next**.
- 5. On the next page of the wizard, click **Select Database**.
- 6. In the **Select Data Source** dialog box, browse to the SQL Server or Access database you want to base the form on, and then click **Open**.

Note If you are connecting to a SQL Server database, you may

need to click **New Source** next to the **File name** box in the **Select Data Source** dialog box, and then follow the steps in the wizard to create a remote data source connection to the SQL server.

- 7. If there is more than one table in your database, click a primary table for your form in the **Select Table** dialog box, and then click **OK**.
- 8. Add any additional tables.
 - ▶ How?
- 9. In the Data Connection Wizard, click **Next**.
- 10. On the next page of the wizard, type a name for the data connection.
- 11. If you want to automatically receive data whenever the form is opened by a user, select the **Automatically retrieve data when form is opened** check box, and then click **Finish**. If the form does not automatically receive data, you may want to insert a button that retrieves the data when clicked.

Connect an existing form to a Web service

You can create a data connection in your Microsoft Office InfoPath 2003 form that receives data from a Web service, or submits part or all of your form to a Web service.

Note For the primary data source to receive data from a Web service, the structure of the form must match the structure of the Web service. To do this, you must design a form based on that Web service.

- Receive data from a Web service
- Submit data to a Web service

Modify or remove a data connection

- 1. In design mode, click **Data Connections** on the **Tools** menu.
- 2. In the **Data Connections** dialog box, click the data connection you want to modify or remove, and then do one of the following:
 - To modify the data connection, click Modify, and then follow the steps in the Data Connection Wizard. For detailed instructions on how to use the Data Connection Wizard, see the appropriate topic for connecting to a database, Web service, Microsoft Windows SharePoint Services site, or XML document.
- To remove the data connection, click **Remove**.

Notes

Primary data connections that receive data (those that are established when you design a form based on a Web service or database) cannot be removed.

Update the main data connection to a database or Web service

When you design a new form that is connected to a database or Web service, Microsoft Office InfoPath 2003 builds a data structure for your form based on that database or Web service. If the database or Web service is then changed or moved, you can update your form so that it uses the updated database or Web service.

- In design mode, click Convert Main Data Source on the Tools menu.
- 2. Follow the steps in the Data Connection Wizard to update your form's data connection.
- 3. Revise the form to match the updated database or Web service. If a field has been removed in the updated database, for example, rebind or remove the controls that were previously bound to that field.
- 4. Test the form thoroughly.

Note Testing the form includes previewing the form, as well as opening a new form and filling it out. Make sure to test any controls bound to new or modified fields.

- 5. If users have already filled out forms based on this form, upgrade those existing forms.
 - ▶ How?
- 6. To republish the form, click **Publish** on the **File** menu.

Insert a button that refreshes data from a secondary data source

When you add a data connection to your Microsoft Office InfoPath 2003 form that receives data from a Web service, database, Microsoft Windows SharePoint Services site, or XML document, you can insert a button that allows users to refresh the data from that data source.

- 1. In design mode, place the insertion point where you want to insert the button.
- 2. On the **Insert** menu, click **More Controls**.
- 3. In the **Controls** task pane, click **Button**.
- 4. In the form, double-click the button, and then in the **Button Properties** dialog box, click **Refresh** in the **Action** list.
- 5. In the **Refresh** dialog box, do one of the following:
- To refresh all of the secondary data sources, click All secondary data sources.
- To refresh one secondary data source, click One secondary data source, and then click the data source you want in the Choose the secondary data source list.

Insert a control bound to a secondary data source

When you add a data connection to your Microsoft Office InfoPath 2003 form that receives data from a Web service, database, Microsoft Windows SharePoint Services site, or XML document, InfoPath creates a secondary data source in your form. You can then insert controls that are bound to the fields and groups in that secondary data source.

- 1. In design mode, place the insertion point where you want to insert the control.
- 2. On the **View** menu, click **Data Source**.
- 3. In the **Data source** list in the **Data Source** task pane, click the secondary data source you want.
- 4. Right-click the field that you want to bind the control to, and then click the control you want to insert on the shortcut menu.

Notes

The values associated with a secondary data source cannot be saved when a user fills out a form, although they can be submitted to a Web service using a submit data connection. In addition, you cannot edit the properties or structure of the fields and groups associated with a secondary data source, nor can you specify default values or custom, noncode-based data validation for those fields.

Populate a list box with values from a database, Web service, XML document, or SharePoint site

- 1. In design mode, double-click the list box you want to populate.
- 2. On the **Data** tab in the **List Box Properties** dialog box, click **Look up values in a data connection to a database, Web service, file, or SharePoint library or list**.
- 3. In the **Data Connection** list, click the data connection you want to use.
- 4. Click **Select XPath** an ext to the **Entries** box.
- 5. In the **Select a Field or Group** dialog box, click the field or group that contains the fields that provide the values you want to populate the list box with, and then click **OK**.
- 6. Click **Select XPath** an ext to the **Value** box.
- 7. In the **Select a Field or Group** dialog box, click the field that contains the values that are stored in the form when a user clicks an item in the list box, and then click **OK**.
- 8. Click **Select XPath** next to the **Display name** box.
- 9. In the **Select a Field or Group** dialog box, click the field that contains the values that are displayed in the list box, and then click **OK**.

Enable form submission

You can create a data connection in your Microsoft Office InfoPath 2003 form that allows users to submit a completed form to a Web service or Microsoft Windows SharePoint Services form library, or submit their form as an attachment in an e-mail message. In addition, you can design a form that is based on a Web service or database such that users can submit the completed form.

Note In order for users to submit their form as an e-mail message, they must have Microsoft Office Outlook 2003 installed on their computer.

- 1. In design mode, do one of the following:
- To allow users to submit the form by clicking a button on the form, double-click the button you want to use, and then in the Action list in the Button Properties dialog box, click Submit.
- To enable submit without adding a button to the form, click **Submitting Forms** on the **Tools** menu.
- 2. In the **Submitting Forms** dialog box, click **Enable submit commands and buttons**.
- 3. If you want users to submit data using the selected data connection, select the **Enable the Submit menu item on the File menu** check box.

Note If you have more than one type of submit functionality in your form, only one can use the **Submit** command on the **File** menu.

- 4. Do one of the following:
 - Allow forms to be submitted to a database
 - Allow forms to be submitted through an e-mail message, or to a Web service or Windows SharePoint Services form library
 - ▶ Allow forms to be submitted to a Web server (HTTP)
 - Allow forms to be submitted using script or managed code
 - Allow forms to be submitted using rules

Enable a form to be submitted as an e-mail message

You can design a form so that users can click **Submit** on the **File** menu to send their completed form as an attachment in an e-mail message. The e-mail addresses, subject line, and attachment file name can all be predefined— either with static values or values based on entries in the form.

Note In order for users to submit their form as an e-mail message, they must have Microsoft Office Outlook 2003 installed on their computer.

- 1. In design mode, click **Data Connections** on the **Tools** menu.
- 2. In the **Data Connections** dialog box, click **Add**.
- 3. In the Data Connection Wizard, click **Submit data**, and then click **Next**.
- 4. On the next page of the wizard, click **As an e-mail message**, and then click **Next**.
- 5. On the next page of the wizard, type the e-mail addresses you want to send the completed form to in the **To**, **Cc**, and **Bcc** boxes. To specify an address that is based on a value in the form, click **Insert Formula** next to the appropriate address box, and then use the options in the **Insert Formula** dialog box to create a formula.
- 6. In the **Subject** box, type a subject for the e-mail message. To specify a subject that is based on a value in the form, click **Insert Formula** , and then use the options in the **Insert Formula** dialog box to create a formula.
- 7. In the **Introduction** box, type the text that you want to appear in the body of the e-mail message.
- 8. In the **Attachment Name** box, type a name for the form when it is attached as a file to the e-mail message. To specify an attachment name that is based on a value in the form, click **Insert Formula** [6], and then use the options in the **Insert Formula** dialog box to create a formula.

- 9. Click Next.
- 10. On the next page of the wizard, type a name for the data connection, and then click **Finish**.

Customize options for form submission

If your form template is designed so that users can submit their forms when they are finished filling them out, you can customize the **Submit** command for the form, as well as the behavior of the form after submission.

- ▶ Customize the **Submit** command on the **File** menu
- Specify the form's behavior after it is submitted
- ▶ Customize the success or failure messages that appear when a form is submitted

Automatically submit a form by using a rule

When designing a form, you can create a rule that automatically submits a form to a database, Web service, Microsoft Windows SharePoint Services library or list, or as an e-mail message, in response to an event. This event can include a change to a particular field or group in the data source, the click of a button, the insertion of a repeating section or row in a repeating table, or the opening of a form.

▶ Tip

- 1. In design mode, do one of the following:
- To run the rule when a particular field or group in the data source is changed, open the **Data Source** task pane, double-click the field or group, and then click the **Rules** tab.
- To run the rule when the value of a control is changed, doubleclick the control, and then on the **Data** tab in the **Control** Properties dialog box, click **Rules**.
- To run the rule when the form is opened, click Form Options on the Tools menu, click the Open and Save tab, and then click Rules.
- To enable the form's default submit command to run the rule, click **Submitting Forms** on the **Tools** menu, and then click **Enable Submit commands and buttons**. In the **Submit to** list, click **Custom submit using rules**, and then click **Rules**.
- To run the rule when a button is clicked, double-click the button, click Rules and Custom Code in the Action list, and then click Rules.
- 2. In the **Rules** dialog box or on the **Rules** tab, click **Add**.
- 3. In the **Name** box, type a name for the rule.
- 4. To specify when the rule should run, click **Set Condition**, and then enter the condition. The rule will run when the event specified in step one occurs and the condition specified in this step is met.

- 5. Click Add Action.
- 6. In the **Action** list in the **Action** dialog box, click **Submit using a** data connection.
- 7. Do one of the following:
- If you have already added a data connection, click it in the **Data** connection list.
- To add a new data connection, click **Add**, and then follow the instructions in the Data Connection Wizard.
- 8. If you want to prevent any rules from running after this rule (for the current event), select the **Stop processing rules when this rule finishes** check box in the **Rule** dialog box.

About views and print views

In Microsoft Office InfoPath 2003, views are based on Extensible Stylesheet Language (XSL), which allows for the transformation and presentation of structured form data.

Each InfoPath form has a default view, which is displayed to users when they open and fill out a form.

You can also create custom views for your forms. Creating custom views can be useful in various scenarios. For example, if your form design is getting too long or too complex, you can move parts of your form into different custom views and let users choose which view best meets their needs. Or, if your form is connected to a database, you might create one view for querying data and another view for entering data.

When designing a form, you can create custom views in the **Views** task pane. Users filling out the form can access the form's views on the **View** menu. When a user switches to a different view, the form's data doesn't change; only the presentation of the form and the amount of data displayed changes.

- Designing print views
- Advanced print view options
- ▶ User roles: different views for different people

Add or delete a view

Each Microsoft Office InfoPath 2003 form has a default view, named "View 1," that users see when filling out a form. When designing a form, you can add new views to the form and delete those you no longer need.

- 1. In design mode, click **Manage Views** on the **View** menu.
- 2. In the **Views** task pane, do one of the following:
- To add a new view, click **Add a New View** under **Actions**, and then in the **Add View** dialog box, type a name for the new view.
- To delete a view, point to the name of the view you want to delete, click the arrow that appears next to the name, and then click **Delete**.

Notes

You cannot recover or restore a view after you delete it. If you delete the default view, a new default view will be created automatically.

You cannot delete print views for Microsoft Office Word by using the **Views** task pane. To delete print views for Word, on the **View** menu in design mode, click **Word Print Views**, and then select the options you want.

Rename a view

- 1. In design mode, click **Manage Views** on the **View** menu.
- 2. In the **Views** task pane, click the view you want to rename, and then click **View Properties**.
- 3. On the **General** tab in the **View Properties** dialog box, type a new name for the view in the **View name** box.

Notes

If you rename a view that is associated with a custom code, the code will no longer work with the view. To fix this problem, update the code with the new name of the view.

You cannot rename print views for Microsoft Office Word. However, you can use the Add Print View for Word Wizard to create the same view with a different name.

Modify view properties

Each view in a form has a set of properties associated with it. In design mode, you set these properties by using the **View Properties** dialog box. For example, you might specify a background color for a particular view or specify print settings, such as page orientation and print range.

- 1. In design mode, click Manage Views on the View menu.
- 2. In the **Views** task pane, click the view whose properties you want to modify, and then click **View Properties**.
- 3. In the **View Properties** dialog box, do one of the following:
- Set the background color
- Set the background picture
- Change settings for layout width
- Set text formatting options for controls
- Show or hide the view from users
- Specify print settings
- Specify page settings
- Change the designated print view

Set the default view for a form

Each Microsoft Office InfoPath 2003 form has a default view, named "View 1," that users see when filling out a form. If you create custom views for a form, you can choose which of the available views will be the default view for users.

- 1. In design mode, click Manage Views on the View menu.
- 2. In the **Views** task pane, point to the name of the view you want to designate as the default view, click the arrow that appears next to the name, and then click **Set as Default**.

Note If only one view is listed in the **Select a view** list, it is automatically designated as the default view.

Switch between views when designing a form

If you're designing a form that has two or more custom views, you can use the following procedure to switch from one view to another.

- 1. In design mode, click Manage Views on the View menu.
- 2. In the **Select a view** list in the **Views** task pane, click the view you want to switch to.

Notes

When filling out a form, users can switch between views by clicking commands on the **View** menu. For example, when filling out a form that is connected to a database, a user might click the **Query** or **Data Entry** command on the **View** menu. The active view is indicated by a check mark next to the menu item.

When designing a form, you can create a rule that automatically switches views when a user opens or submits the form.

Automatically switch views in a form

When designing a form, you can create a rule that automatically switches views when the form is opened or submitted, or when a button is clicked.

▶ Tip

- 1. In design mode, do one of the following:
- To run the rule when the form is opened, click **Form Options** on the **Tools** menu, click the **Open and Save** tab, and then click **Rules**.
- To enable the form's default submit command to run the rule, click **Submitting Forms** on the **Tools** menu, and then click **Enable Submit commands and buttons**. In the **Submit to** list, click **Custom submit using rules**, and then click **Rules**.
- To run the rule when a button is clicked, double-click the button, click **Rules and Custom Code** in the **Action** box, and then click **Rules**.
- 2. In the **Rules** dialog box or on the **Rules** tab, click **Add**.
- 3. In the **Name** box in the **Rule** dialog box, type a name for the rule.
- 4. To specify when the rule should run, click **Set Condition**, and then enter the condition. The rule will run when the event specified in step one occurs and the condition specified in this step is met.
- 5. Click Add Action.
- 6. In the **Action** list in the **Action** dialog box, click **Switch views**.
- 7. In the **View** list, click the view you want to switch to.
- 8. If you want to prevent any rules from running after this rule (for the current event), select the **Stop processing rules when this rule finishes** check box in the **Rule** dialog box.

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service

pack installed. <u>Learn more</u>

About specifying user roles for forms

In Microsoft Office InfoPath 2003, you can assign users to distinct categories, called "user roles," which are based on job title or other criterion.

User roles are often used to present customized views of a form to different types of users. For example, a form that is used to collect feedback about job applicants might have three distinct user roles associated with it— one for recruiters, one for interviewers, and one for hiring managers. For each user role, you can present users with a slightly different view of the form.

Security You should not use roles to restrict access to sensitive data in a form. Even if you make a form read-only or hide certain controls based on roles, users can potentially use a text editing program such as Microsoft Notepad to view or modify the form (.xml) file.

User role assignments

When designing a form, you can add, modify, or delete user roles by using the **Manage User Roles** dialog box.

When you create a new user role, you can assign users to it in the following ways:

By specifying user names (for example, "sales\andrew").

By specifying groups. For example, you can specify an e-mail distribution list that contains the names of all members of the marketing team.

By specifying user names that come directly from the form. For example, if your form contains a **Manager** text box, you can specify the field to which that text box is bound.

After you create a user role, you can set it as one of the following roles:

Default role Users who are not assigned to an existing user role are automatically assigned to the user role that is specified as the default role. The default role is also used for users who are part of a group but who are working offline. One user role is always set as the default.

Initiator role If you want to apply a particular user role to users who open your form for the first time, you can specify an initiator role. For example, you can define an initiator role named "Project Owner" that applies to users who fill out new project specification forms. A user who is assigned to a different user role is automatically reassigned to the project owner role when he or she opens a new project specification form. However, the next time that user opens the same form, InfoPath uses the person's assigned user role instead of the initiator role.

When a user opens a form, InfoPath determines whether that user is assigned to an existing user role. For example, if the user role assignments are determined by network name, InfoPath checks the user's name against the user's network credentials before opening the form. In the form, the name of the current user role appears on the status bar.

Working with user roles

After you define user roles for your form, you can set up a rule that automatically performs a task, such as switching views, based on events and values in the form. For example, you can create a manager user role and then create a rule to automatically switch the view when a manager opens the form. Alternatively, you can create the rule first and then define user roles as part of the process of creating the rule.

You can also vary a control's behavior based on a user role. For example, if your expense report form contains manager and employee roles, you can use conditional formatting to make the text inside a **Total Expenses** box turn red when employees accrue more than \$500 in expenses or managers accrue more than \$1,000. Similarly, you can vary the control's data validation or filter settings based on user role.

Testing user roles

If a form has multiple user roles, you can use the **Preview Form With User Role** command to test the appearance and behavior of each distinct role. For example, if you created a special summary view of your form for managers, along with a rule that switches the view based on user role, you can test that view with the manager role. In the preview window, the name of the current user role appears on the status bar.

When you test a form in the preview window, make note of the following:

Although you can preview a form to check that views work as you expect with user roles, you won't be able to simulate the process of evaluating a user against the user's given role assignment. That's because InfoPath checks user roles when you fill out the form, not when you preview it. If you want to fully test the user roles for your form, you can temporarily assign yourself to each user role that you want to test and then open the form as if you are going to fill it out.

If you click **Preview Form** on the **Standard** toolbar (or use the **Preview Form Default** command on the **File** menu) to preview your form, you'll see the view that's associated with the default user role in the preview window, regardless of your own role assignment. To preview other user roles, use the **Preview Form With User Role** command instead.

Add a user role

As part of the form design process, you can specify user roles, or specific categories of users. For example, in a human resources department, you can specify different user roles for recruiters, managers, and human resource specialists.

Security User roles allow you to present alternative views or behavior to different sets of users. They should not be used to restrict access to sensitive data in a form. Even if you make a form read-only or hide certain controls based on user roles, users can potentially use a text editing program such as Microsoft Notepad to view or modify the form (.xml) file.

- 1. On the **Tools** menu, click **User Roles**.
- 2. In the Manage User Roles dialog box, click Add.
- 3. In the **Add User Role** dialog box, type a name in the **Role** name box.
- 4. To assign users or groups of users to the role, do one or more of the following:
 - Specify one or more users who are part of a network
 - Specify one or more groups that are part of a network
 - Specify user names that come directly from the form
- 5. To set the user role as the initiator role, which is the role that is used whenever a new blank form is opened for the first time, select the **Set as initiator** check box.
 - **Note** When a user role is specified as the initiator, it takes precedence over all other role assignments.
- 6. If you customized a control's behavior based on user role, or if you're using rules to switch views based on user role, you can make sure that everything works by testing your form.
 - ▶ How?

Notes

To specify a default user role for users who aren't assigned to any other role, select the role in the **Manage User Roles** dialog box, and then click **Set as Default**.

If you assigned groups of users to a user role, users belonging to that group must be online for the user role assignment to take effect. If a user is working offline, InfoPath uses the default user role to open the form.

Delete a user role

- 1. On the **Tools** menu, click **User Roles**.
- 2. In the Roles for your form list in the Manage User Roles dialog box, click the role that you want to delete.
- 3. Click Remove.

Change the default user role

Microsoft Office InfoPath 2003 automatically specifies the first user role that is listed in the **Manage User Roles** dialog box as the default role for a form. If you defined more than one user role for a form, you can specify which user role you want as the default.

- 1. On the **Tools** menu, click **User Roles**.
- 2. In the Roles for your form list in the Manage User Roles dialog box, click the role that you want to set as the default role.
- 3. Click Set as Default.

About controls

When filling out forms, users enter information by typing in text boxes, clicking items in a list, selecting check boxes, and performing other actions. The objects that users interact with are called controls. Information that is entered into a control is stored in the form's data source. When you design a form, you use the **Controls** task pane to insert controls.

You can customize a control's appearance, including font, color, and alignment; its default value and data type; and other properties, such as the ScreenTip that appears when a user rests the mouse pointer on the control. You can also customize a control's behavior, such as its ability to filter information or respond to rules. To view the properties for a control, double-click the control on the form.

Controls and the data source

Almost all controls on a Microsoft Office InfoPath 2003 form are associated with, or bound to, the data source. Most controls, including text boxes, check boxes, and list boxes, are bound to fields. If a control is not bound, or if it is bound incorrectly, the information entered into the control cannot be saved. When you design an InfoPath form, you can choose to have the fields in the data source created automatically each time you insert a control, or you can bind controls to existing fields.

The following example illustrates the relationship between two drop-down list boxes and the fields that store their data.



Together, a control and its field determine the type of information that users can enter, as well as how they enter the information. For example, a date picker control makes it easy for users to enter dates. The field to which the date picker is bound saves only values that are dates (in other words, the field has a date data type).

Note In addition to the main data source for your form, you can create data connections to XML documents, databases, Web services, and Microsoft Windows SharePoint Services libraries and lists. These data connections result in secondary data sources that can be used to populate list boxes and provide the values for text boxes and other controls.

Inserting controls on your form

When designing a form, you can use the **Controls** task pane to insert controls. You can choose to create the data source automatically when you insert a control, in which case a field or group will be automatically created for each control that you add. You can also choose to bind the control to an existing field or group. If you choose this option, you are prompted to select a field or group from the data source when you insert the control. To switch between automatically creating the data source and binding the controls to an existing field or group, use the **Automatically create data source** check box in the **Controls** task pane.

You can also insert controls by dragging fields and groups from the **Data Source** task pane onto your form. Each time that you drag a field or group, InfoPath suggests the control that best matches the field or group. For example, if the field has a date data type, InfoPath suggests a date picker.

Types of controls

- Standard controls
- ▶ Repeating and optional controls
- ▶ File and picture controls
- Advanced controls
- Custom controls

Move a control

- 1. In design mode, select the control that you want to move, and then on the **Edit** menu, click **Cut**.
- 2. Click in the form where you want to insert the control, and then on the **Edit** menu, click **Paste**.
- 3. After you paste the control, make sure that it is still correctly bound to its field or group in the data source.
 - ▶ How?

Notes

If you move a control into or out of a repeating section, optional section, choice section, or repeating table, make sure that the data source for the control and the section or table is correctly structured. For example, if the control is moved into a repeating section, the control's field must be part of the section's group.

Delete a control

In design mode, select the control, and then press DELETE.

Note When a control is deleted from a form template, it is also deleted from all forms based on that form template that have already been filled out. Deleting a control will prevent users from seeing or working with the data that was displayed in that control. Deleting a control does not delete the field or group that the control was bound to.

Make a control required

If you want to require users to enter a value into a control, you must make the control's field required. Depending on how you want to display error alerts and the flexibility of your form, you can make the field required by displaying either a standard error message (a red asterisk or dashed red border and a ScreenTip that says "Cannot be blank") or a custom error message.

Note If your form prevents you from selecting the **Cannot be blank** check box in the **Control Properties** dialog box, you will need to display a custom error message.

- Display a standard error message when a control is left blank
- ▶ Display a custom error message when a control is left blank

Copy and paste a control

You can use the **Copy** and **Paste** commands to create a new control that matches the formatting of an existing control. However, you must change the binding of the new control so that it does not store its data in the same field or group as the original control.

▶ Tip

- 1. In design mode, select the control that you want to copy.
 - ▶ Tip
- 2. On the **Edit** menu, click **Copy**.
- 3. Place the insertion point where you want to insert the new control, and then on the **Edit** menu, click **Paste**.
- 4. Right-click the new control, click **Change Binding** on the shortcut menu, and then in the **Control Binding** dialog box, do one of the following:
- To create a new field or group to bind the control to, right-click the group to which you want to add the field or group, click **Add** on the shortcut menu, and then enter the appropriate properties in the **Add Field or Group** dialog box.
- To bind the control to an existing field or group, click the field or group that you want to bind the control to.

Notes

When you copy and paste a control with conditional formatting, the original conditions will be copied to the new control.

Because data validation is based on fields and groups, not controls, changing the field or group to which a control is bound also changes any existing data validation for that control.

Hide or disable a control based on values in a form

You can use conditional formatting to hide or disable a control based on the value of another control on the form. For example, a text box for a driver's license number can be read-only unless the user enters an appropriate age into another control on the form.

- 1. In design mode, select the control you want to hide or disable, and then click **Conditional Formatting** on the **Format** menu.
- 2. In the **Conditional Formatting** dialog box, click **Add**.
- 3. In the first box under **If this condition is true**, click **Select a field or group**.
- 4. In the **Select a Field or Group** dialog box, select the field whose value will determine whether the control is hidden or disabled, and then click **OK**.
 - **Note** Microsoft Office InfoPath 2003 verifies a condition by checking the value of the field that is bound to the control, not the control itself.
- 5. In the second and third boxes under **If this condition is true**, enter the rest of the condition, and then do one of the following:
- To disable the control, select the Read-only check box.
 Depending on the type of control, this check box may be labeled
 Don't allow users to insert or delete this control or Disable this control.
 - **Note** To disable the controls in a section, you must disable each control in the section. Disabling the actual section only prevents users from inserting or deleting the section.
- To hide the control, select the Hide this control check box.

Change a control to another control

When designing a form, you can quickly change one type of control to another type of control. For example, you can change a text box to a drop-down list box.

In design mode, right-click the control that you want to change, point to **Change To** on the shortcut menu, and then click the type of control that you want.

Notes

If you do not see the choice that you want on the shortcut menu, check the properties of the field or group to which the control is bound. The properties of a field or group determine what type of control can be bound to the field or group.

You cannot change certain controls, including buttons, hyperlink controls, or expression boxes, to a different type of control.

Change the binding of a control

- In design mode, right-click the control whose binding you want to change, and then click **Change Binding** on the shortcut menu.
- 2. If your form uses more than one data source, click the one that you want to use in the **Data source** list in the **Control Binding** dialog box.
- 3. Do one of the following:
- To create a new field or group to bind the control to, right-click the group to which you want to add the new field or group, click
 Add on the shortcut menu, and then enter the appropriate properties in the Add Field or Group dialog box.
- To bind the control to an existing control, click the field or group that you want to bind the control to.

▶ Tip

Notes

Because data validation is based on fields and groups, not controls, changing the field or group to which a control is bound also changes any existing data validation for that control.

Test the binding of a control

When designing a form, you can quickly test the binding of a control in the form by referring to the control's binding icon.

In design mode, click the control whose binding you want to test.

Text and a binding icon appear in the upper-right corner of the control. The text indicates the group or field to which the control is bound in the data source. The icon indicates whether the control is correctly bound to that group or field. The following table describes each of the binding icons.

Icon Description

- The binding is correct.
- The control may not function as expected. To learn more about the possible problem, right-click the control, and then click **More Details** on the shortcut menu.
- The control will not function correctly. When users save the form, any data they entered into the control will be lost. To learn more about the problem, right-click the control, and then click **More Details** on the shortcut menu.

Apply the same font to multiple controls

When designing a form, you can specify the fonts that appear inside the controls on a form. If you specify font and formatting options for one control on a form, you can follow these steps to quickly apply that font to some or all of the remaining controls.

- 1. Click the control that contains the font that you want to copy.
- 2. On the **Format** menu, do one of the following:
- To apply the font only to other controls of the same type, click **Apply Font to All** *ControlType* **Controls**.
- To apply the font to all of the controls on the form, click **Apply** Font to All Controls.
- 3. If you want to verify the changes by displaying sample text in the controls, click **Sample Data** on the **View** menu.

▶ Tip

Add a label to a control

If you want to name the controls in your form, you can do so by typing the label text in the form area.

- Add a label to a single control
- Add and align a group of labels and controls
- ▶ Tip

Notes

When you insert a field or group into the form from the **Data Source** task pane, the control's label is created automatically. To prevent this, on the **Tools** menu, click **Options**. In the **Options** dialog box, click the **Design** tab, and then clear the **Create labels for controls automatically when adding from data source** check box.

Insert a vertical label

You can use a vertical label to display read-only text at a 90-degree angle. For example, you can insert vertical label text into the column headings in a repeating table to save space.

Note A vertical label is an expression box control with the **Display text vertically** property selected by default.

- 1. In design mode, place the insertion point where you want to insert the vertical label.
- 2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
- 3. Under Insert controls, click Vertical Label.
- 4. In the **Text** box in the **Insert Vertical Label** dialog box, type the text that you want.
- 5. To test the vertical text, click **Preview Form** on the **Standard** toolbar.

Assign a keyboard shortcut to a control

When you assign a keyboard shortcut to a control, users can immediately navigate to that control by pressing a combination of keystrokes. Using the keyboard to work with a form is often faster than using the mouse. In addition, these shortcuts can be helpful for people who have motor disabilities that prevent them from using a mouse.

Note This option is available for the following controls: text box, rich text box, drop-down list box, list box, date picker, check box, option button, picture, file attachment, button, and ink picture.

- In design mode, right-click the control for which you want to set a keyboard shortcut, and then click Control Properties on the shortcut menu.
- 2. In the *Control* Properties dialog box, click the **Advanced** tab.
- 3. In the **Access key** box, type a character.

Note If you choose to use keyboard shortcuts in your form, you need to communicate to users that the shortcuts exist. For example, you might type "Name (ALT+N)" for the label of a text box so that users know there's a keyboard shortcut associated with the text box.

Notes

An "access key" is a keyboard shortcut that uses the ALT key as part of the shortcut.

Certain keyboard shortcuts are already in use by Microsoft Office InfoPath 2003 and other Microsoft Office System programs. For example, if you press ALT+F in any Microsoft Office program, the **File** menu will open. Although you are free to reassign existing keyboard shortcuts to the controls in your form, doing so may confuse users who are accustomed to these Office shortcuts.

Change text alignment in a control

When designing a form, you can specify the way text aligns inside certain types of controls, such as text boxes and expression boxes.

- 1. In design mode, right-click the control for which you want to set the text alignment, and then click *Control* Properties on the shortcut menu.
- 2. In the *Control* Properties dialog box, click the **Display** tab.
- 3. In the **Alignment** list, click the text alignment option that you want.

Align text in a control with its label

On occasion, the text inside a control may appear to be slightly offset from the surrounding label text. This is especially evident when you remove the border from a control. To make your form look more professional, you can automatically align the text in a control with its label.

Note This option is available for text box, rich text box, expression box, and date picker controls.

- 1. In design mode, right-click the control that you want to align, and then click *Control* Properties on the shortcut menu.
- 2. In the **Control Properties** dialog box, click the **Size** tab.
- 3. Under **Align control with label**, click **Align**, and then click **Apply**.

Notes

When you click **Align**, Microsoft Office InfoPath 2003 automatically adjusts the control's height, bottom padding, and bottom margin settings as necessary. If you change any of these settings later, you may need to realign the control.

When aligning a rich text box control, InfoPath changes the existing value in the **Height** box to **auto**. This reduces the height of the rich text box so that the text inside is better aligned with the surrounding text. If this is not what you want, close the **Rich Text Box Properties** dialog box, and then press CTRL+Z to undo the change.

Set the default value for a control

A default value is the initial value that a control contains. It can be standard text or a formula.

Default values are stored in fields and are displayed in the controls that are bound to those fields. If you want to set the default value for all controls that are bound to a specific field, you can set a default value for the field itself, instead of setting the default value for each control.

Note To set the default value for a section or a repeating table, use the following procedures for the individual controls in the section or table.

- ▶ Set the default value for a text box, rich text box, date picker, or list control
- ▶ Set the default value for a list box or drop-down list box
- Set the default value and default state for a check box
- ▶ Set the default value and default state for a group of option buttons
- Set a default picture for a picture control
- Set a default file for a file attachment control

Add placeholder text to a control

You can add placeholder text to the controls in your form that instruct users on how to fill out your form. An example of placeholder text might be "Click here and type."

Note This option is available only for text box and rich text box controls.

- 1. In design mode, right-click the control for which you want to add placeholder text, and then click *Control* Properties on the shortcut menu.
- 2. In the *Control* Properties dialog box, click the **Display** tab.
- 3. In the **Placeholder** box, type the text that you want to display in the control.

Change the size of a control

When designing a form, you can change the size of a single control by changing its properties. You can also change the size of multiple controls at once.

- ▶ Change the size of a single control
- ▶ Change the size of multiple controls at once

Insert a scrolling region

A scrolling region control contains other controls and can display vertical and horizontal scroll bars. Scrolling regions are useful when you want to include a fixed region on a form that will scroll to accommodate large amounts of data. For example, if you use a repeating table to display numerous records from a database, you can insert the repeating table inside of a scrolling region to display just a few records at a time. Because scrolling regions are used exclusively for layout purposes, they are not bound to fields or groups in the data source.

- ▶ Insert a scrolling region
- Customize a scrolling region

Change the margins for a control

After inserting a control in your form, you can specify the amount of blank space that appears between that control and the text, tables, and other controls that surround it. For example, to add extra space between label text and a text box, you can increase the margin setting for the text box.

- 1. In design mode, right-click the control whose margins you want to change, and then click *Control* Properties on the shortcut menu.
- 2. In the **Control** Properties dialog box, click the **Size** tab.
- 3. Under **Margins** in the **Top**, **Bottom**, **Left**, and **Right** boxes, type the amount of space that you want between the control and the surrounding text, tables, or other controls, and then select a unit of measurement from the list for each amount.

▶ Tip

Change the padding in a control

After inserting a control in your form, you can change the amount of space, or "padding," between a control's border and its contents. For example, if you want to add extra white space at the bottom of a text box, you can change the bottom margin setting for the text box.

▶ Tip

- 1. In design mode, right-click the control whose padding you want to change, and then click *Control* Properties on the shortcut menu.
- 2. In the **Control** Properties dialog box, click the Size tab.
- 3. Under **Padding** in the **Top**, **Bottom**, **Left**, and **Right** boxes, type the amount of space that you want to appear between the control's border and the text inside it, and then select a unit of measure from the list for each amount.

▶ Tip

Change the properties for multiple controls at once

- In design mode, click one control, and then hold down CTRL and click each additional control whose size you want to change.
- 2. On the **Format** menu, click **Properties**.
- 3. In the **Properties (Multiple Selection)** dialog box, do one of the following:
- To specify the height for the selected controls, under **Size** in the **Height** box, type the height that you want, and then select a unit of measure from the list.
- To specify the width for the selected controls, under Size in the Width box, type the width that you want, and then select a unit of measure from the list.
- To adjust the amount of blank space between the borders of the selected controls and their contents, under **Padding** in the **Top**,
 Bottom, **Left**, and **Right** boxes, type a number, and then select a unit of measure from the list.
- To adjust the amount of blank space between the selected controls and the surrounding text, tables, or other controls in the form, under Margins in the Top, Bottom, Left, and Right boxes, type a number, and then select a unit of measure from the list.
- To line up the selected controls with their labels, under **Align** control with label, click **Align**.

Notes

Depending on the types of controls that you select, options on the **Size** tab may be unavailable.

Set a control as read-only

If you do not want users to change the contents of a control, you can make it read-only. For example, if the department code in an expense report form is the same for everyone who uses your form, you can use the department code as a default value and make the text box read-only.

Note This option is available only for text box, rich text box, bulleted list, numbered list, and plain list controls.

- In design mode, right-click the control that you want to mark as read-only, and then click Control Properties on the shortcut menu.
- 2. In the *Control* Properties dialog box, click the **Display** tab, and then select the **Read-only** check box.

Note Making a control read-only prevents users from typing in the control, but it does not affect the control's appearance in any way.

Notes

You can also use conditional formatting to make a control read-only unless a specific value is entered into another control. For example, a text box for a driver's license number can be read-only unless the user enters an appropriate age into another control on the form. To learn more about this option, see Hide or disable a control based on values in a form.

Show or hide sample data

When designing a form, you can use sample data to quickly check the text formatting of controls. Sample data shows how text will appear on your form when users fill it out. Sample data is not visible when you preview a form or when a user fills it out.

In design mode, click **Sample Data** on the **View** menu.

Enable or disable the spelling checker for a control

You can specify whether Microsoft Office InfoPath 2003 checks the spelling of the data that users type into a text box, rich text box, or list control. By default, the spelling checker is enabled.

- 1. In design mode, right-click the control for which you want to enable or disable the spelling checker, and then click *Control* **Properties** on the shortcut menu.
- 2. In the *Control* Properties dialog box, click the **Display** tab, and then do one of the following:
- To enable the spelling checker for the control, select the **Enable** spelling checker check box.
- To disable the spelling checker for the control, clear the **Enable** spelling checker check box.

About using ActiveX controls on forms

In Microsoft Office InfoPath 2003, you can make Microsoft ActiveX controls available in design mode and then use those controls on the forms that you design. This is useful when you want to provide a user interface element that isn't available in Microsoft Office InfoPath 2003, such as a slider control that comes with Microsoft Windows. You or your company can also create custom ActiveX controls for a specific business use. For example, a hospital might develop an ActiveX control that allows medical imagery to be displayed on a form.

Making ActiveX controls available in design mode

Before you can add ActiveX controls to your forms, you must make them available in the **Controls** task pane. InfoPath includes an Add Custom Control Wizard that guides you through the process of selecting the ActiveX control, including an installation cabinet (.cab) file with your form, if necessary, and specifying binding options and other properties for the ActiveX control. You can open this wizard by clicking **Add or Remove Custom Controls** at the bottom of the **Controls** task pane.

After the ActiveX control appears in the task pane, you can insert it onto your form. If you double-click the ActiveX control, any property pages that are specific to that control will appear alongside the standard InfoPath property pages for ActiveX controls.

If a number of form designers in your organization need to use the same ActiveX control on their forms, and if that control is not already installed on their computers, you can create a custom setup program, such as a Microsoft Windows Installer Package (.msi) file. This custom setup program can quickly install and register the ActiveX control for the form designers.

Making ActiveX controls available to users

When you publish a form that contains an ActiveX control, that control must be installed and registered on your users' computers in order for the form to open. Some ActiveX controls, like those that are included with Windows, may already be installed and registered on the users' computers. Other ActiveX controls, such as custom controls that you develop, may not be installed and registered. In this case, you must take measures to install and register the controls for your users.

If you plan to install your form with a custom installation program, you can install and register the control at that point. Alternatively, you can specify an existing .cab file in the Add Custom Control Wizard.

When you use the Add Custom Control Wizard to make an ActiveX control available, the wizard asks you whether you want to package an existing .cab file along with your form template. This file allows the control to be installed and registered on your users' computers. If you choose to include a .cab file with your form and your users do not already have the ActiveX control installed, InfoPath prompts them to install the control when they open the form for the first time.

InfoPath automatically installs and registers the control for users if both of the following requirements are met:

The .cab file is signed. A digital certificate is required to sign a .cab file; this creates a digital signature on the file.

The digital signature is from a form developer or other "trusted" publisher. If the .cab file is signed but the publisher is not yet trusted on the user's computer, InfoPath displays a security alert and requires the user to enable trust for the publisher.

Security issues

When designing forms that include ActiveX controls, you should keep the following security issues in mind:

Use caution when you add ActiveX controls to your form. ActiveX controls may be designed in such a way that their use could pose a security risk. It is recommended that you use only controls from trusted sources.

ActiveX controls cannot be installed by or run on forms that have a restricted level of trust (as specified in the **Form Options** dialog box in design mode).

For information about designing more secure ActiveX controls, visit the Microsoft Developer Network (MSDN) Web site.

Add or remove an ActiveX control from the Controls task pane

Before you can use a Microsoft ActiveX control on your form, you must make it available in the **Controls** task pane. Any controls that you add to Microsoft Office InfoPath 2003 appear at the bottom of the **Insert controls** list, under the **Custom** category.

- Add an ActiveX control to the Controls task pane
- ▶ Remove an ActiveX control from the **Controls** task pane

Notes

If you develop your own ActiveX control for use in InfoPath, you must identify that control as both safe for initialization and safe for scripting. Otherwise, you won't be able to add your control to the **Controls** task pane. For specific information about these options and about designing ActiveX controls in general, visit the Microsoft Developer Network (MSDN) Web site.

Insert an ActiveX control

When you insert a Microsoft ActiveX control onto your form, Microsoft Office InfoPath 2003 inserts a custom control that is bound to either a group or field in the data source, depending on the options that you specified in the Add Custom Control Wizard.

Note To make ActiveX controls available in design mode, use the Add Custom Control Wizard to add them to the list of available controls in the **Controls** task pane.

- ▶ Insert an ActiveX control (automatically create the data source)
- Insert an ActiveX control (bind to an existing field)

Insert a button

You can create a button on your form to make it easy for users to perform common tasks, such as submitting a form to a Web service or querying a database. You can also associate a button with rules or custom code that runs when a user clicks the button.

- 1. In design mode, place the insertion point where you want to insert the button.
- 2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
- 3. In the **Insert controls** list in the **Controls** task pane, click **Button**.
- 4. Double-click the button icon that was inserted onto your form, and then, in the **Action** list in the **Button Properties** dialog box, click the action that you want the button to perform.
 - Available actions
- 5. In the **Label** box, type the text that you want to appear on the button.

Insert a date picker

You can add a date picker to your form so that users can easily enter dates. Date pickers are bound to fields.

- ▶ Insert a date picker (automatically create the data source)
- ▶ Insert a date picker (bind to an existing field)

Insert a check box

You can create a check box on your form so that users can quickly enter true or false values. Check boxes are bound to fields.

- ▶ Insert a check box (automatically create the data source)
- ▶ Insert a check box (bind to an existing field)

Change the default state of a check box

- 1. In design mode, right-click the check box, and then click **Check Box Properties** on the shortcut menu.
- 2. On the **Data** tab in the **Check Box Properties** dialog box, do one of the following:
- To make the default state of the check box cleared, click **Cleared**.
- To make the default state of the check box selected, click **Checked**.

Note The default state of the check box will determine the value that is stored in the data source. For example, if the default state of the check box is selected, the value in the data source will match the value that is selected in the **Value when checked** list.

Insert a group of option buttons

You can insert a group of option buttons on your form so that your users can quickly enter specific values. Option buttons are used to offer users a single choice within a set of mutually exclusive choices.

A group of option buttons is bound to a single field in the data source. When a user clicks an option button, the value associated with that option button is stored in the field.

- ▶ Insert a group of option buttons (automatically create the data source)
- ▶ Insert a group of option buttons (bind to an existing field)

Insert a choice group

A choice group is a control on a form that is used to present a set of mutually exclusive choices to users. By default, a choice group contains two choice sections, one of which appears to the user as the default choice on the form.

When users move their mouse pointer over the default choice on the form, a shortcut menu button appears in the upper-left corner of the choice group. Users can click this button to see the options for removing or replacing the default choice.

For example, if you want to collect addresses from employees who are located around the world, you can design choice sections that account for the varied address requirements of different countries and regions. When filling out the form, employees can replace the default address section with one that applies to their own country or region.

- ▶ Insert a choice group (automatically create the data source)
- ▶ Insert a choice group (bind to an existing group)

Insert a repeating choice group

You can insert a repeating choice group on your form so that users can add multiple choice groups when filling it out.

When users move their mouse pointer over the default choice on the form, a shortcut menu button appears in the upper-left corner of the repeating choice group. Users can click this button to see options for inserting, removing, or replacing choices.

For example, on a travel form, you can use a repeating choice group to allow users to enter information about their business trips. For each transportation segment, users can select one of three choices— flight, car, or train. Each choice displays a different set of fields.

- ▶ Insert a repeating choice group (automatically create the data source)
- ▶ Insert a repeating choice group (bind to an existing group)

Customize commands for a choice section

You can customize the menu commands that are used to insert, remove, or replace choice sections when users fill out a form. To quickly rename all the default menu commands that are associated with a choice section, change the name of the group to which the choice section is bound. If you want more control over specific commands, or if you do not want to change the group name, you can customize each command separately.

- ▶ Rename all the menu commands for a choice section
- ▶ Add or customize a single menu or toolbar command name for a choice section

Insert an ink picture control

You can create an ink picture control on your form that lets people who are using Tablet PCs draw images when filling out the form. Ink picture controls are bound to fields.

- ▶ Insert an ink picture control (automatically create the data source)
- ▶ Insert an ink picture control (bind to an existing field)

Security Ink picture controls are not a secure method of collecting and storing user signatures. If you want to collect and store signatures on your form, it is recommended that you use digital signatures.

Insert a picture control

If you create a picture control on your form, users can choose to insert pictures into that control when they fill out the form. Picture controls are bound to fields.

- ▶ Insert a picture control (automatically create the data source)
- ▶ Insert a picture control (bind to an existing field)

Notes

In addition to picture controls, users can add clip art, photos, and other pictures to rich text fields that support full rich text formatting.

Insert a file attachment control

If you insert a file attachment control onto your form, users can add file attachments when they fill out the form. File attachment controls are bound to fields.

- ▶ Insert a file attachment control (automatically create the data source)
- ▶ Insert a file attachment control (bind to an existing field)

Notes

InfoPath blocks users from attaching potentially unsafe files (such as .bat, .exe, .vbs, and .js files) that may contain viruses.

Insert a list box

You can create a list box or drop-down list box on your form so that users can enter predefined values. List boxes are bound to fields and can look up values in a fixed list, in other parts of the form, or through a data connection to an XML document, a Microsoft Access database, a Microsoft SQL Server database, a Web service, or a Microsoft Windows SharePoint Services library or list.

- 1. In design mode, place the insertion point where you want to insert the list box.
- 2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
- 3. In the **Controls** task pane, do one of the following:
- To automatically create a field in the data source that is bound to the list box, select the **Automatically create data source** check box.
- To bind the list box to an existing field, clear the **Automatically** create data source check box.
 - **Note** If the check box is unavailable, the form's design does not allow the data source to be automatically created.
- 4. Under Insert controls, click List Box or Drop-Down List Box.
- 5. If you cleared the **Automatically create data source** check box in step 3, select a field in the **Control Binding** dialog box to which you want to bind the list box.
- 6. Double-click the list box that was inserted on the form.
- 7. In the **List Box Properties** dialog box, do one of the following:
 - Insert a list box that looks up values in a fixed list
 - Insert a list box that looks up values in another part of the form
 - ▶ Insert a list box that looks up values by using a data connection

Note The information in this topic may not apply if you are working with a

form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Insert a list control

When designing a form, you can insert list controls, which automatically format text into lists as users type into them.

- ▶ Insert a list control (automatically create the data source)
- ▶ Insert a list control (bind to an existing field)

About master/detail controls

If you expect users to work with large amounts of data on your form— for example, with multiple records in a database— you can use a master/detail control to make it easier for them to manage that data. When a user selects an item in a master control, related information about that item appears in a detail control.

For example, you can use a master/detail control when designing a human resources form that is connected to a database of employee records. The master control can display employee name and job title, and the details control can display employee location, hire date, date of birth, and so on. In database terms, this is the equivalent of a one-to-one relationship.

Similarly, you can use the master control to display one set of records (for example, customer names) and the details control to display related records (for example, customer orders). In database terms, this is the equivalent of a one-to-many relationship.

The master/detail relationship

A master/detail control is actually a set of two related controls. One of these controls is the designated master control; the other is the designated detail control. The master control is always a repeating table. The detail control can be either a repeating table or a repeating section. You can create the master/detail relationship by inserting the appropriate repeating controls on your form and then changing the properties of each control. Or, you can insert a predefined master/detail control from the **Insert controls** list in the **Controls** task pane.

When you insert a master/detail control from the **Controls** task pane, Microsoft Office InfoPath 2003 establishes a one-to-one relationship between the master control and the detail control. This means that each selected row in the master control has only a single matching result in the detail control. In a typical one-to-one relationship, both master and detail controls are bound to the same repeating group in the data source.

If you prefer, you can use key fields to establish a one-to-many relationship for a master/detail control. In this type of relationship, each selection in the master control can result in multiple matches in the detail control. For example, if users select a supplier record in a master control, they can view the products that are associated with that supplier in the detail control. In a typical one-to-many relationship, master and detail controls are bound to separate repeating groups in the data source

To establish a one-to-many relationship, you must use options on the **Master/Detail** tab to specify the appropriate key fields.

Insert a master/detail control

You can use the master/detail control to link one control to another, so that the selection in the first control determines what appears in the other.

A master/detail control is actually a set of two related controls. One control is the designated master control, and the other is the detail. The master control is always a repeating table. The detail control can be either a repeating table or a repeating section.

Microsoft Office InfoPath 2003 provides a predefined master/detail control that you can insert by using the **Controls** task pane. You can also create a master/detail control by inserting a repeating table and repeating section (or two repeating tables) and then changing the properties for each control.

- ▶ Insert a master/detail control (automatically create the data source)
- ▶ Insert a master/detail control (bind to an existing group)

Change the default master/detail relationship

When you insert a master/detail control onto your form by using the **Controls** task pane, Microsoft Office InfoPath 2003 establishes a one-to-one relationship between the master control and the detail control. This means that for each selection in the master control, a single matching result appears in the detail control. You can change the master/detail relationship from the default one-to-one relationship to a one-to-many relationship. In other words, for each selection in the master control, multiple matching results can appear in the detail control.

When you insert a master/detail control, InfoPath inserts a repeating table and a repeating section by default. Both of these controls are bound to the same repeating group in the data source. By default, both controls contain other controls, which are bound to fields in that repeating group. In a one-to-many relationship, however, master and detail controls are typically bound to separate repeating groups. If you've inserted a master/detail control from the **Controls** task pane, and you want to establish a one-to-many relationship instead of a one-to-one relationship, you can delete the detail control that InfoPath creates and then create a new one that better suits your needs, as outlined in the following procedure.

Tip

- 1. In design mode, select the repeating section that is designated as the detail control, and then press DELETE.
- 2. Insert a new repeating section onto the form.
 - ▶ How?
- 3. Insert the controls you want into the repeating section.
 - ▶ How?
- 4. Right-click the repeating section, and then click **Repeating Section Properties** on the shortcut menu.
- In the Repeating Section Properties dialog box, click the Master/Detail tab.
- 6. Under Master/detail settings, click Set as detail.

- 7. In the **Link to master ID** list, click the name of the master control.
- 8. Under Master/detail relationship, click By key field.
- 9. Click **Select XPath** next to the **Key field (master)** box, and then, in the **Select a Field or Group** dialog box, click the key field for the master control, and then click **OK**.
- 10. Click Select XPath next to the Key field (detail) box, and then, in the Select a Field or Group dialog box, click the key field for the detail control, and then click OK.

Insert a section

You can use a section to organize controls on your form for your users. A section contains other controls and is bound to a group.

When you insert a section onto your form, you are essentially inserting an empty container. For the section control to be useful, you must insert other controls inside it.

- Insert a section (automatically create the data source)
- ▶ Insert a section (bind to an existing group)

Notes

InfoPath offers other "container" controls that are similar to sections but that offer additional functionality. To create a section that users can insert or delete, you can use an optional section. To insert a section that can be replaced with another section, use a choice group. To create a section that users can insert multiple times (like a record in a database), use a repeating control, such as a repeating section.

Insert an optional section

You can create an optional section on your form that users can choose to insert and remove when they fill out the form. By default, an optional section does not appear on the form until a user inserts it. For example, on a travel request form, you can include an optional "Car Rental" section. Users who need to rent a car can insert and fill out this optional section; those who don't need to rent a car can leave the section hidden.

Optional sections are bound to groups. Any control in an optional section must be bound to a field that is part of the optional section's group. Unlike conditional formatting, which can make a section invisible but does not remove it from the data source, an optional section is not part of the form's data source until a user inserts it.

- ▶ Insert an optional section (automatically create the data source)
- Insert an optional section (bind to an existing group)

Notes

If a user removes an optional section after filling it out, the data in that section is lost, not hidden.

Customize commands for an optional section

You can customize menu commands that are used to add or remove an optional section when users fill out a form. To quickly rename all of the default menu commands associated with an optional section, change the name of the group to which the optional section is bound. If you want more control over specific commands, or if you do not want to change the group name, you can customize each command separately. You can also customize the hint text that users click to insert the section.

- ▶ Rename all the menu commands for an optional section
- ▶ Add or customize a single menu or toolbar command name for an optional section
- Customize the hint text for an optional section

Insert a repeating section

You can create a repeating section on your form so that users can add additional sections when filling it out. For example, you can use a repeating section to present a series of career objectives on a performance review form. The person filling out the form can add new objective sections as necessary.

When users move their mouse pointer over the repeating section, a shortcut menu button appears in the upper-left corner. Users can click this button to see options for inserting or removing the section.

Repeating sections are bound to groups. Any control in a repeating section must be bound to a field that is part of the repeating section's group.

- ▶ Insert a repeating section (automatically create the data source)
- ▶ Insert a repeating section (bind to an existing group)

Customize commands for a repeating section

You can customize menu commands that are used to add and remove a repeating section when users fill out a form. To quickly rename all the default menu commands that are associated with a repeating section, change the section name. If you want more control over specific commands, or if you do not want to change the section name, you can customize each command name separately. You can also customize the hint text that appears below the section on the form, which users click to insert a new section.

- ▶ Rename all the menu commands for a repeating section
- Add or customize a single menu or toolbar command name for a repeating section
- Customize the hint text for a repeating section

Notes

You can also use these procedures to customize commands for a recursive repeating section, which is a type of repeating section.

Insert a repeating recursive section

A repeating recursive section contains other controls, can be inserted within itself, and is bound to nested reference groups. You can use repeating recursive sections to create hierarchical content, such as an outline.

- ▶ Insert a repeating recursive section (automatically create the data source)
- ▶ Insert a repeating recursive section (bind to an existing repeating group)

Insert a repeating table

If you want users to be able to enter information in a table, you can add a repeating table control to your form. Repeating tables display just one row by default. When filling out a form, users can insert additional rows as they need them. For example, you can use a repeating table on an expense report form to collect itemized expenses from employees. The person filling out the form can add only as many expense rows as he or she requires. If your form includes a data connection to an external data source, such as a database, you can use a repeating table to automatically display multiple records.

When users move their mouse pointer over the rows in the repeating table, a shortcut menu button appears to the left of each row. Users can click this button to see options for inserting or removing rows. Users can also quickly insert rows in the table by clicking **Insert item** below the table. You can customize this hint text or remove it from the form.

Repeating tables are bound to repeating groups. Each column in a repeating table contains a control, and each control is bound to a field that is part of the repeating table's group.

- Insert a repeating table (automatically create the data source)
- Insert a repeating table (bind to an existing group)

Rearrange columns in a repeating table

Like any other table, a repeating table typically contains a number of vertical columns. However, you shouldn't attempt to copy a column and paste it elsewhere in the table, because this can cause binding errors. To rearrange columns, you must follow the steps in the Repeating Table Binding Wizard.

- 1. In design mode, double-click the **Repeating Table** label below the table.
- 2. On the **Data** tab in the **Repeating Table Properties** dialog box, click **Change Binding**.
- 3. On the first page of the Repeating Table Binding Wizard, do one of the following:
- If your form uses a single data source, click **Next**.
- If your form uses more than one data source, click the one that you want to use in the **Data source** list, and then click **Next**.
- 4. In the **Columns in table** list on the second page of the wizard, select the column that you want to move, and then do one of the following:
- To move a column to the left, click **Move Up**.
- To move a column to the right, click **Move Down**.
- 5. Repeat step 4 for each column that you want to move.

Note When you rearrange columns, existing text formatting or column sizes may be lost.

Customize commands for a repeating table

You can customize the menu commands that are used to add or remove rows in a repeating table when users fill out a form. To quickly rename all the default menu commands that are associated with a repeating table, change the name of the repeating group to which the table is bound. If you want more control over specific commands, or if you do not want to change the group name, you can customize each command name separately. You can also customize the hint text that appears below the repeating table on the form, which users click to insert a new row.

- ▶ Rename all the menu commands for a repeating table
- Add or customize a single menu or toolbar command name for a repeating table
- Customize the hint text for a repeating table

Insert a text box

You can create a text box on your form that users can enter data into. Text boxes are bound to fields.

- ▶ Insert a text box (automatically create the data source)
- ▶ Insert a text box (bind to an existing field)

Insert a rich text box

You can create a rich text box on your form that lets users enter formatted text, paragraph breaks, tables, lists, and images when they fill out the form. Rich text boxes are bound to fields.

- ▶ Insert a rich text box (automatically create the data source)
- ▶ Insert a rich text box (bind to an existing field)

Insert an expression box

You can use an expression box to display read-only text, display the value of another control on the form, and create formulas that are based on XPath expressions.

Note An expression box only displays data and does not store it, so you should use an expression box only if you do not want to save its value or reference it in another formula.

- 1. In design mode, place the insertion point where you want to insert the expression box.
- 2. If the **Controls** task pane is not visible, click **More Controls** on the **Insert** menu.
- 3. Under Insert controls, click Expression Box.
- 4. In the **Insert Expression Box** dialog box, do one of the following:
- To type text or an XPath expression directly into the expression box, type in the **XPath** box.
- To insert a formula, click **Edit Formula £**.
- 5. To test the expression box, click **Preview Form** on the **Standard** toolbar.

Enable or disable character formatting in a control

You can enable character formatting in rich text box controls. This lets users who are typing information into the control format their text. For example, they can apply a color to the text or make it bold.

Note To enable character formatting for a bulleted list, numbered list, or plain list control, the control must be bound to a field with a rich text XHTML data type.

- 1. In design mode, right-click the control for which you want to enable or disable character formatting, and then click *Control* **Properties** on the shortcut menu.
- 2. In the *Control* Properties dialog box, click the **Display** tab.
- 3. To enable or disable character formatting for the selected control, select or clear the **Character formatting** check box.

Enable or disable full rich text formatting in a control

You can enable full rich text formatting in rich text box controls. This lets users entering information into the control format the text in a number of ways. For example, they can switch to another font or insert a picture into the control.

- 1. In design mode, right-click the rich text box control for which you want to enable or disable full rich text formatting, and then click **Rich Text Box Properties** on the shortcut menu.
- 2. In the **Rich Text Box Properties** dialog box, click the **Display** tab.
- 3. To enable or disable full rich text formatting for the selected control, select or clear the **Full Rich Text (images, tables, etc.)** check box.

Enable or disable paragraph breaks in a control

Enabling paragraph breaks for a text box or rich text box control lets users create new paragraphs of text when they are typing information into the control.

- 1. In design mode, right-click the control for which you want to enable or disable paragraph breaks, and then click *Control* **Properties** on the shortcut menu.
- 2. In the *Control* Properties dialog box, click the **Display** tab.
- 3. To enable or disable paragraph breaks, select or clear the **Paragraph breaks** check box.

Enable or disable text wrapping for a control

You can enable text wrapping for a number of controls in Microsoft Office InfoPath 2003, including text boxes, rich text boxes, and expression boxes. When this option is enabled, text will automatically wrap inside controls when users type text into them. Without this option enabled, any text that exceeds the width of the control will be hidden. You can also choose a text wrapping style for picture controls.

- ▶ Enable or disable text wrapping for a text box, rich text box, or expression box control
- ▶ Enable or disable text wrapping for a picture control

Specify the character limit for a text box

You can limit the amount of text that users can type in a text box. For example, in a survey form, you can use this feature to prevent users from typing more than 500 characters in a comments box.

- 1. In design mode, right-click the text box for which you want to limit characters, and then click **Text Box Properties** on the shortcut menu.
- 2. In the **Text Box Properties** dialog box, click the **Display** tab.
- 3. Under **Options**, select the **Limit text box to** check box, and then specify the number of characters that you want.

Notes

If you enable paragraph breaks for a text box, the character-limit option will be unavailable.

The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Manage text scrolling in a control

If paragraph breaks are enabled for a control, or if you specify that the text should wrap inside the control, you can specify the scrolling options for that control. For example, you can make scroll bars appear on a text box when the user types more text than the text box is able to display by default.

Note Scrolling options are available for text box, rich text box, and expression box controls. By default, a text box scrolls automatically as the user types, and a rich text box and an expression box expand to show all of the text.

- 1. In design mode, right-click the control for which you want to set a text-scrolling option, and then click *Control* Properties on the shortcut menu.
- 2. In the *Control* Properties dialog box, click the **Display** tab.
- 3. In the **Scrolling** list, click the text-scrolling option that you want.

▶ Tip

Add or remove a currency symbol

- 1. In design mode, right-click the text box or expression box whose data you want to format, and then click *Control* Properties.
- 2. On the **Data** tab or the **General** tab in the **Control Properties** dialog box, click **Format**.

Note You can add currency symbols only for text boxes that have a decimal or whole number data type, or for expression boxes that are formatted as a decimal or whole number.

- 3. In the **Format** dialog box, do one of the following:
- To add a currency symbol, click **Currency**, and then in the **Currency** list, select the type of currency you want to display.
- To remove a currency symbol, click **Number**.

Note Adding or removing a currency symbol affects only the displayed value, not the value that is saved in the data source.

Add or remove a thousands separator in a number

- 1. In design mode, right-click the text box or expression box whose data you want to format, and then click *Control* Properties.
- 2. On the **Data** tab or the **General** tab in the **Control Properties** dialog box, click **Format**.

Note You can add a thousands separator only for text boxes that have a decimal or whole number data type, or for expression boxes that are formatted as a decimal or whole number.

- 3. In the **Format** dialog box, do one of the following:
- To add a thousands separator to the number, select the Use a digit grouping symbol check box under Other options.
- To remove a thousands separator from the number, clear the Use a digit grouping symbol check box under Other options.

Notes

Depending on locale, the separator may separate more or less than thousands.

Adding or removing a thousands separator affects only the displayed value, not the value that is saved in the data source.

Add or remove percent formatting for a number

You can format a text box or expression box so that numbers that users type into the box are displayed as percentages. For example, a value of 0.2 would be displayed as 20. This formatting affects both how values are displayed and how values are saved. This means that any value typed into a control with percent formatting will be treated as a percent value. So if a user types **30** into a text box that has a percent data format, the saved value would be 0.30. If a user types 0.30, the saved value would be 0.0030.

- 1. In design mode, right-click the text box or expression box whose data you want to format, and then click *Control* Properties.
- 2. On the **Data** tab or the **General** tab in the **Control Properties** dialog box, click **Format**.

Note You can add percent formatting only for text boxes that have a decimal data type or for expression boxes that are formatted as a decimal.

- 3. In the **Decimal Format** dialog box, do one of the following:
- To format the number as a percentage, click **Percentage**.
- To remove percent formatting, click **Number**.

Note Microsoft Office InfoPath 2003 does not add percent symbols to numbers that are formatted as percentages. To add a percent symbol to a control's label, click where you want the percent symbol to appear, and then type %.

Change how a date or time is displayed

- 1. In design mode, right-click the date picker, text box, or expression box whose data you want to format, and then click *Control* Properties.
- 2. On the **Data** tab or the **General** tab in the **Control Properties** dialog box, click **Format**.

Note You can specify how dates and times are displayed only for date pickers and text boxes that have a date, time, or date and time data type, or for expression boxes that are formatted as one of those data types.

- 3. Depending on what you have selected to format, in the **Format** dialog box, do one or both of the following:
 - To format the date, click the display style you want in the Display the date like this list.
- To format the time, click the display style you want in the **Display the time like this** list.

Note Display styles that have an asterisk will update based on the user's system settings.

4. To change the locale date and time settings, click the region you want in the **Locale** list.

Note Changing how a date or time is displayed affects only the displayed value, not the value that is saved in the data source.

Change how negative numbers are displayed

- 1. In design mode, right-click the text box or expression box whose data you want to format, and then click *Control* Properties.
- 2. On the **Data** tab or the **General** tab in the **Control Properties** dialog box, click **Format**.

Note You can specify how negative numbers are displayed only for text boxes that have a decimal or whole number data type, or for expression boxes that are formatted as a decimal or whole number.

3. Under **Other options** in the **Format** dialog box, click the display style you want in the **Display negative numbers like this** list.

Remove formatting from a date, time, or number

- 1. In design mode, right-click the date picker, text box, or expression box whose formatting you want to remove, and then click *Control* Properties.
- 2. On the **Data** tab or the **General** tab in the **Control Properties** dialog box, click **Format**.

Note You can remove the formatting from dates, times, and numbers only for date pickers and text boxes that have a date, time, date and time, whole number, or decimal data type, or for expression boxes that are formatted as one of those data types.

3. In the **Format** dialog box, click **None (display XML value)**.

Notes

When you remove the formatting from a date, time, or number, the displayed value will exactly match the value that is saved in the data source.

The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Specify the number of decimal places displayed

- 1. In design mode, right-click the text box or expression box whose data you want to format, and then click *Control* Properties.
- 2. On the **Data** tab or the **General** tab in the **Control Properties** dialog box, click **Format**.

Note You can specify the number of decimal places only for text boxes that have a decimal data type or for expression boxes that are formatted as a decimal.

- 3. In the **Decimal places** list in the **Decimal Format** dialog box, do one of the following:
- To always display a fixed number of decimal places, click the number of decimal places you want.
- To show the number of decimal places that the user types into the control, click **Auto**.

Note Changing the number of decimal places affects only the displayed value, not the value that is saved in the data source.

About conditional formatting

Conditional formatting allows you to change the appearance of a control based on values users enter into your Microsoft Office InfoPath 2003 form. You can set conditions that determine the formatting for each control, such as font style, text color, and background color. You can also hide or disable a control. Controls can have multiple conditions, which means that a control's appearance can change in different ways depending on the values entered into it. A control's appearance can also change based on the values entered into other controls on the form, or on whether the data in a control has been digitally signed.

You can use conditions to determine whether the value of a control is blank, is within a specified range, equals the value of another control, or starts with or contains certain characters. You can then associate formatting with that condition to highlight important data and provide a more interactive experience for users when they fill out the form. For example, conditional formatting can be set to:

Hide a control unless a specific check box is selected.

Disable a **Look Up** button unless the user enters a name into a specific text box.

Make a text box read-only unless a specific value is entered into another text box.

Change the font color and style for all expenses that require a receipt.

Change row colors in a repeating table based on the value of a text box in the table.

Mark financial entries with red text when they are below zero, and green text when they are equal to or higher than zero.

▶ Tip

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Add conditional formatting

You can use conditional formatting to change the appearance of a control based on values users enter into your form.

- In design mode, select the control to which you want to add conditional formatting, and then click Conditional Formatting on the Format menu.
- 2. In the **Conditional Formatting** dialog box, click **Add**.
- 3. In the **If this condition is true** boxes, enter the condition.

Note Microsoft Office InfoPath 2003 verifies a condition by checking the value of the field that is bound to the control, not the control itself.

- 4. To add additional conditions, click **And**, and then in the rightmost box under **If this condition is true**, do one of the following:
- To apply the formatting only when both conditions are true, click and.
- To apply the formatting when one or both of the conditions are true, click **or**.
- 5. Under **Then apply this formatting**, select the formatting you want to apply for your conditions, and then click **OK**.
- 6. If you have specified multiple conditional formats for a control and want to change the order in which they are evaluated, select a conditional format in the **Conditional Formatting** dialog box, and then click **Move Up** or **Move Down**.

Note If more than one specified condition is true, InfoPath applies only the formatting of the first condition in the list that is true.

Remove conditional formatting

- 1. In design mode, select the control you want, and then click **Conditional Formatting** on the **Format** menu.
- 2. In the **Conditional Formatting** dialog box, select the conditional format you want to remove, and then click **Remove**.

About formulas and functions

Formulas are made up of functions, references to other fields or groups, static values (such as a number), and mathematical operators. You can use formulas to calculate mathematical values, display dates and times, reference fields in a form, or perform string operations such as combining multiple fields into one field. A formula will always return a single result.

Formulas are stored in fields and are displayed in the controls that are bound to those fields. You can create a formula by using the control's properties (for text boxes, rich text boxes, and date pickers) or the field's properties.

Functions, such as **sum** and **avg**, are a crucial component of formulas, and you can use functions wherever you can insert a formula. Most functions require arguments to operate on, and some functions require more than one argument. All functions return a value. For example, in a function that sums the values in a column, the argument is the repeating field to which the controls in the column are bound. The function adds and returns the values of all of those fields. Depending on the function, an argument can be a static value or a field, or a static value or field that is dynamically selected at run time based on another value.

The following is an example of a formula that calculates the average of the values in a repeating field, then calculates the sum of the values in a different repeating field, and then adds the sum and the average together:

avg(field3) + sum(field4)

In this example, there are two functions, **avg** and **sum**, each of which requires one argument (field3 and field4, respectively). The formula returns the results of the functions added together.

When you work with formulas, keep the following in mind:

If you display a formula in a text box, rich text box, or date picker, make the control read-only to prevent users from typing over the result of the formula.

In the control's or field's properties, define whether to update the results of a formula whenever the values in the formula change or whether to show only the initial result of a formula.

When arguments are used in functions, all of the values in a repeating field will be evaluated. For example, to sum a column in a repeating table, sum the field to which the column is bound: **sum(field1)**.

Functions can be nested within other functions, in which case the result of one function provides the argument for another function.

Because mathematical formulas usually depend on integer or decimal values as arguments, make sure that the **Treat blank values as zero** check box is selected on the **General** tab in the **Form Options** dialog box.

Formulas in Microsoft Office InfoPath 2003 are based on XPath expressions. If you are familiar with XPath, you can edit the XPath expression directly.

Formulas refer to fields and groups in the form, from the context of where the fields and groups are located. A field referencing itself uses a period (.) instead of its own name. For example, a formula that uses the **Name** function to return its own field name looks like this: **name(.)**

The following tables list the functions that are available in InfoPath.

Date and time functions

Function Description		Example
now	Returns the current date and time. No argument is required.	now()
today	Returns the current date. No argument is required.	today()

Field functions

Function	Description	Example
name	Returns the name of the specified field or group, as well as its namespace prefix. If the argument is a set of fields or groups, it returns the name and prefix of the first field or group.	name(field1)
position	Returns the index (or position) of the field or group with respect to the other fields and groups in the same group.	position()
count	Counts the number of instances of the specified field or group.	count (field1)
id	Returns the fields or groups associated with the specified ID. This function is not available in the Insert Function dialog box; it must be typed in the Formula box in the Insert Formula dialog box.	id(field1)
last	Returns the value of the last occurrence of the field in the current context.	field1[last()]
local-name	Returns the name of the specified field or group. If the argument is a set of fields or groups, it returns the name of the first field or group.	local- name(field1)
namespace- uri	Returns the full namespace of the specified field or group. If the argument is a set of fields or groups, it returns the namespace of the first field or group.	namespace- uri(field1)

Math functions

Function	Description	Example
avg	Calculates the average value of all of the numerical values in the argument.	avg(field1)
Boolean	Returns true if the argument is not null, is a positive or negative number, or is a non-empty string. Otherwise, returns false. Returns false if the argument is null, is zero or an NaN value, or is an empty string.	, ,
ceiling	Returns the smallest integer that is not less than the argument (rounding up).	ceiling(field1)
eval	Returns the values of a set of fields. The first argument defines the set of fields; the second argument defines the expression to calculate for the set of fields. Usually, the eval function is nested within a function that operates on a set of fields, such as sum or avg .	sum(eval(group2, "field1 * field2"))
false	Returns false.	false()
floor	Returns the largest integer that is not greater than the argument (rounding down).	floor(field1)
min	Returns the smallest number in the argument.	min(field1)
not	If the argument is false or null, returns true. If the argument is true or not null, returns false.	not(field1)
number	Converts the argument into a number.	number(field1)
nz	Returns the value of the specified field. If the field is blank, it returns zero (0).	nz(field1)
round	Rounds the argument to the nearest integer.	round(field1)
sum	Calculates the sum of the arguments.	sum(field1)

true	Returns true.	true()
max	Returns the largest number in the argument.	max(field1)

Text functions

Function	Description	Example
contains	If the first argument contains the second argument, returns true. Otherwise, it returns false.	contains(field1, "city")
concat	Combines the value of fields with other fields or text strings.	concat(field1," of ",field2)
normalize- space	Removes the white space from the argument.	normalize- space(field1)
starts- with	If the first argument starts with the second argument, returns true. Otherwise, it returns false.	starts- with(field1, "B")
string	Converts the argument to a text string.	string(field1)
string- length	Returns the number of characters in the argument.	string- length(field1)
substring	Returns a specific part of a text string. The first argument defines the text string, the second argument defines the starting position, and the third argument defines how many characters to include.	substring(field1, 3,5)
substring- after	Returns a specific part of a text string. The first argument defines the text string, and the second argument defines what to look for in the text string. The function returns the text that follows the first occurrence of the second argument.	substring(field1, "country")
substring- before	Returns a specific part of a text string. The first argument defines the text string, and the second argument defines what to look for in the text string. The function returns the text that precedes the first occurrence of the second argument.	substring- before(field1, "the end")
translate	Returns a converted text string. The first	translate(field1,

argument defines the text string, the second argument defines the characters to change in the text string, and the third argument defines what to change the characters to. The characters in the second argument are converted to the characters in the third argument's matching position.

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Insert a formula

Formulas are stored in fields and are displayed in the controls that are bound to those fields. When designing a form, you can create formulas in text boxes, rich text boxes, date pickers, and expression boxes.

Note You should use an expression box only if you do not want to use the result of the formula in another formula or save it as part of a form, because expression boxes only display data and do not store it.

- 1. In design mode, double-click the control where you want to create the formula.
- 2. On the **Data** tab in the **Control Properties** dialog box, click **Insert Formula** .

Note To create the formula by using the field's properties, right-click the field in the **Data Source** task pane, and then click **Properties**.

- 3. In the **Insert Formula** dialog box, do one or more of the following:
- To insert a field or group in the formula, click **Insert Field or Group**, and then click the field or group in the **Select a Field or Group** dialog box.
- To insert a function in the formula, click **Insert Function**, and then select the function in the **Insert Function** dialog box. If the function requires an argument, double-click the function in the **Formula** box, and then click a field or group.
- To insert a value or mathematical operator in the formula, type the value or symbol for the mathematical operation in the Formula box. The symbols appear in the following table (note that you should include spaces around the division symbol).

Operation	Symbol
Add	+
Subtract	_
Multiply	*

Divide /

▶ Tip

Add, subtract, multiply, and divide values

When designing a form, you can create a formula that calculates values in text boxes, rich text boxes, date pickers, and expression boxes. Because expression boxes only display data and do not store it, you should use an expression box only if you do not want to use the calculated value in another calculation or save it as part of your form.

Formulas are stored in fields and are displayed in the controls that are bound to those fields.

- 1. In design mode, double-click the control where you want to create the formula.
- 2. On the **Data** tab of the **Control Properties** dialog box, click **Insert Formula** .

Note To create the formula by using the field's properties, rightclick the field in the **Data Source** task pane, and then click **Properties** on the shortcut menu.

- 3. In the **Insert Formula** dialog box, click **Insert Field or Group**.
- 4. In the **Select a Field or Group** dialog box, click the field whose value you want to include in the formula, and then click **OK**.
- 5. After the field in the **Formula** box in the **Insert Formula** dialog box, type the symbol for the mathematical operation that you want. The symbols are shown in the following table (note that you should include spaces around the division symbol).

Operation	Symbol
Add	+
Subtract	-
Multiply	*
Divide	1

- 6. To add the second value to your formula, click **Insert Field or Group**.
- 7. In the **Select a Field or Group** dialog box, click the field whose

value you want to include.

Notes

To create more complicated formulas, you can repeat steps 3 and 4, include functions, and use parentheses.

To add the values in a repeating field, use the **sum** function.

Calculate the average for a column of numbers

Formulas are stored in fields and are displayed in the controls that are bound to those fields. When designing a form, you can create a formula that calculates the average for a column of numbers in a repeating table by using a text box, rich text box, or expression box.

Note You should use an expression box only if you do not want to use the calculated average in another calculation or save it as part of your form, because expression boxes only display data and do not store it.

 In design mode, double-click the control where you want to create the average formula. The control should not be part of the repeating table.

Note To create the formula by using the field's properties, right-click the field in the **Data Source** task pane, and then click **Properties**.

- 2. On the **Data** tab in the **Control Properties** dialog box, click **Insert Formula** .
- 3. In the **Insert Formula** dialog box, click **Insert Function**.
- 4. In the **Categories** list in the **Insert Function** dialog box, click **Math**.
- 5. In the **Functions** list, click **avg**, and then click **OK**.
- 6. In the **Formula** box in the **Insert Formula** dialog box, double-click the text in parentheses.
- 7. In the **Data source** list in the **Select a Field or Group** dialog box, click the field whose values you want to average. The column must be bound to the selected field, and the field itself must be a repeating field or part of a repeating group.

Display the contents of multiple fields in one control

When designing a form, you can create a formula that combines and displays the contents of two or more fields into a single control by using a text box, rich text box, or expression box.

Note You should use an expression box only if you do not want to use the value in another calculation or save it as part of your form, because expression boxes only display data and do not store it.

- 1. In design mode, double-click the control where you want to display the combined values.
- 2. On the **Data** tab in the **Control Properties** dialog box, click **Insert Formula** .
- 3. In the Insert Formula dialog box, click Insert Function.
- 4. In the **Categories** list in the **Insert Function** dialog box, click **Text**.
- 5. In the **Functions** list, click **concat**, and then click **OK**.
- 6. In the **Formula** box in the **Insert Formula** dialog box, double-click the text in parentheses.
- 7. In the **Data source** list in the **Select a Field or Group** dialog box, click the field whose value you want to combine, and then click **OK**.
- 8. Repeat steps 6 and 7 for every field whose value you want to combine.
- 9. In the **Formula** box, type the text you want to display between the fields when they are combined. For example:
- To display a space between two fields, type," ", (for example, concat(field1, " ", field2)).
- To display a comma between two fields, type,",", (for example, concat(field1, ",", field2)).
- To display two fields as run together, make sure the fields are separated by a comma (for example, **concat(field1, field2)**).

Display the current date

When designing a form, you can use a text box, rich text box, or date picker to display the date that the form was first created. You can also use an expression box to show the current date, although this value will be updated with the current date each time the form is opened.

Note You should use an expression box only if you do not want to use the current date in another calculation or save it as part of the form, because expression boxes only display data and do not store it.

- In design mode, double-click the control where you want to display the current date.
- 2. On the **Data** tab in the **Control Properties** dialog box, click **Insert Formula** .
- 3. In the **Insert Formula** dialog box, click **Insert Function**.
- 4. In the **Categories** list in the **Insert Function** dialog box, click **Date and Time**.
- 5. In the **Functions** list, click **today**, and then click **OK** twice.
- 6. On the **Data** tab in the **Control Properties** dialog box, clear the **Update this value when the result of the formula is** recalculated check box.

▶ Tip

Display the current date and time

When designing a form, you can use a text box, rich text box, or date picker to display the date and time that the form was first created. You can also use an expression box to show the current date and time, although this value will be updated each time the form is opened.

Note You should use an expression box only if you do not want to use the current date and time in another calculation or save the current date and time as part of the form, because expression boxes only display data and do not store it.

- 1. In design mode, double-click the control where you want to display the current date and time.
- 2. On the **Data** tab in the **Control Properties** dialog box, click **Insert Formula** .
- 3. In the **Insert Formula** dialog box, click **Insert Function**.
- 4. In the **Categories** list in the **Insert Function** dialog box, click **Date and Time**.
- 5. In the **Functions** list, click **now**, and then click **OK** twice.
- 6. On the **Data** tab in the **Control Properties** dialog box, clear the **Update this value when the result of the formula is** recalculated check box.

▶ Tip

Round a number in a control

Formulas are stored in fields and are displayed in the controls that are bound to those fields. When designing a form, you can use a formula in a text box, rich text box, or expression box to round to the nearest integer a value that users type into a control on the form.

Note You should use an expression box only if you do not want to use the rounded value in another calculation or save it as part of the form, because expression boxes only display data and do not store it.

- 1. In design mode, double-click the control where you want to create the rounding formula.
- 2. On the **Data** tab in the **Control Properties** dialog box, click **Insert Formula** .

Note To create the formula by using the field's properties, right-click the field in the **Data Source** task pane, and then click **Properties**.

- 3. In the **Insert Formula** dialog box, click **Insert Function**.
- 4. In the **Categories** list in the **Insert Function** dialog box, click **Math**.
- 5. In the **Functions** list, click **round**, and then click **OK**.
 - ▶ Tip
- 6. In the **Formula** box, double-click the text in parentheses.
- 7. In the **Data source** list in the **Select a Field or Group** dialog box, click the field whose value you want to round.

Sum a column of numbers

Formulas are stored in fields and are displayed in the controls that are bound to those fields. When designing a form, you can create a formula that sums a column of numbers in a repeating table by using a text box, rich text box, or expression box.

Note You should use an expression box only if you do not want to use the calculated sum in another calculation or save it as part of your form, because expression boxes only display data and do not store it.

1. In design mode, double-click the control where you want to create the sum formula. The control must not be part of the repeating table.

Note To create the formula by using the field's properties, right-click the field in the **Data Source** task pane, and then click **Properties**.

- 2. On the **Data** tab in the **Control Properties** dialog box, click **Insert Formula** .
- 3. In the **Insert Formula** dialog box, click **Insert Function**.
- 4. In the **Categories** list in the **Insert Function** dialog box, click **Math**.
- 5. In the **Functions** list, click **sum**, and then click **OK**.
- 6. In the **Formula** box in the **Insert Formula** dialog box, double-click the text in parentheses.
- 7. In the **Data source** list in the **Select a Field or Group** dialog box, click the field whose values you want to sum. The field must be bound to the column and it must be a repeating field or part of a repeating group.

Edit the XPath expression for a formula

Formulas in Microsoft Office InfoPath 2003 are based on XPath expressions. If you are familiar with XPath, you can edit the XPath expression directly in the **Insert Formula** dialog box. You can also edit the XPath expression by using the control's properties (for text boxes, rich text boxes, and date pickers) or the field's properties.

- 1. In design mode, double-click the control that contains the expression you want to edit.
- 2. On the **Data** tab in the **Control Properties** dialog box, click **Insert Formula** .

Note To edit the XPath expression by using the field's properties, right-click the field in the **Data Source** task pane, and then click **Properties**.

- 3. In the **Insert Formula** dialog box, select the **Edit XPath** (advanced) check box.
- 4. In the **Formula** box, edit the XPath expression for the formula. To include fields, groups, and functions in the expression, click **Insert Field or Group** or **Insert Function**.

About data validation

Data validation can be used to display error alerts when users enter incorrect values into a form. Rather than checking for errors after a form is completed, data validation verifies values as the form is being filled out.

Working with data validation when filling out a form

Microsoft Office InfoPath 2003 automatically checks the data you type into a form and displays an error alert if the value does not match the condition specified by the data validation rule. If a value you enter does not match the specified condition, an error alert provides you with immediate feedback.

Error alerts can be either dialog box alerts or inline alerts. You can correct errors immediately, or you can use the commands on the **Tools** menu to navigate through and correct all the errors in a form at once. Uncorrected errors are marked with a dashed red border. Required entries are marked with a red asterisk or, for some types of controls, a dashed red border. To see details about the error, you can right-click the control.

If you save a form without fixing the errors, you will be reminded about the errors with a dialog box alert. You can save forms with validation errors, but you cannot submit them.

Working with data validation when designing a form

Data validation is always associated with a specific control and occurs when a user fills out a form. Note that InfoPath verifies a condition by checking the value of the field that is bound to the control, not the control itself.

- Types of data validation
- Types of error alerts
- Schema-based validation

To test a data validation constraint, including the conditions and error alerts, preview your form and enter valid and invalid data into the control you want to test. Error alerts will appear in the preview exactly as they appear to your users.

Add data validation

Data validation can be used to display error alerts when users enter incorrect values into a control. You can also write code to perform advanced validation. Note that Microsoft Office InfoPath 2003 verifies a condition by checking the value of the field that is bound to a control, not the control itself.

- Require a specific type of data
- Require a specific value or range of values
- Require a specific pattern, such as a phone number
- Add code-based validation

Remove data validation

To prevent unwanted data validation alerts from appearing when users fill out a form, you can remove data validation conditions, change the data type, or change the control from required to optional.

- Remove data validation conditions
- Change the data type
- ▶ Change a control from required to optional

Notes

If a form is based on a pre-existing XML Schema or a sample form, the schema may contain validation conditions that you cannot modify using InfoPath.

Require a specific data entry pattern in a control

You can create data entry patterns that define how users must enter numbers and text into a control. For example, you can require that users enter a phone number with parentheses around the first three digits. Data entry patterns display error alerts when users type an incorrect pattern into a control.

Note Patterns do not automatically format data entry into a particular pattern.

- 1. In design mode, select the control for which you want the validation to occur, and then click **Data Validation** on the **Format** menu.
- 2. In the **Data Validation** dialog box, click **Add**.
- 3. Make sure the first box under **If this condition is true** contains the control's field.
- 4. In the second box, click **does not match pattern**.
- 5. In the third box, click **Select a pattern**.
- 6. In the **Data Entry Pattern** dialog box, do one of the following:
 - In the **Standard patterns** box, select one of the patterns.
 - In the **Custom pattern** box, enter a pattern. You can type directly in the box. To insert special characters in the pattern, click the character in the **Insert special character** box.
 - Tip
- 7. Click **OK**.
- 8. In the **Error alert type** box, click the type of error alert you want to display to users.
- 9. In the **ScreenTip** box, type the text you want to display when a user points to the control or right-clicks the control.
- 10. In the **Message** box, type the text you want to display in the dialog box either immediately (if you chose a dialog box alert) or

when the user requests more details (if you chose an inline alert).

Add a rule

When designing a form, you can use rules to automatically display a dialog box, set a field's value, query or submit to a data connection, switch views, or open or close a form in response to certain events and conditions. The events can include a change to a particular field or group in the data source, the click of a button, the insertion of a repeating section or row in a repeating table, or the opening or submission of a form. The conditions can include whether the value of a field is blank, is within a specified range, equals the value of another field, or starts with or contains certain characters.

You can also add multiple actions for each rule. For example, you can automatically open a new form and display a dialog box message when a field's value exceeds 100.

- In design mode, do one of the following:
- To run the rule when a particular field or group in the data source is changed, open the **Data Source** task pane, double-click the field or group, and then click the **Rules** tab.
- To run the rule when the value of a control is changed, doubleclick the control, and then on the **Data** tab in the **Control** Properties dialog box, click **Rules**.
- To run the rule when the form is opened, click **Form Options** on the **Tools** menu, click the **Open and Save** tab, and then click **Rules**.
- To run the rule when a button is clicked, double-click the button, click Rules and Custom Code in the Action list, and then click Rules.
- To enable the form's default submit command to run the rule, click **Submitting Forms** on the **Tools** menu, and then click **Enable Submit commands and buttons**. In the **Submit to** list, click **Custom submit using rules**, and then click **Rules**.
- 2. In the **Rules** dialog box or on the **Rules** tab, click **Add**.
- 3. In the **Name** box in the **Rule** dialog box, type a name for the

rule.

- 4. To specify a condition for when the rule should run, click **Set Condition**, and then enter the condition. The rule will run when the event specified in step one occurs and the condition specified in this step is met.
- 5. Click Add Action.
- 6. In the **Action** list in the **Action** dialog box, click the action you want, and then enter the options for that action.
- 7. Repeat steps 5 and 6 for each action you want to run for this rule.
- 8. If you want to prevent any rules from running after this rule (for the current event), select the **Stop processing rules when this rule finishes** check box.

Automatically close a form by using a rule

When you design a form, you can create a rule that automatically closes a form when the user clicks a button.

▶ Tip

- 1. In design mode, double-click the button you want to create the rule for.
- 2. In the **Action** box on the **General** tab, click **Rules and Custom Code**, and then click **Rules**.
- 3. In the **Rules** dialog box, click **Add**.
- 4. In the **Name** box in the **Rule** dialog box, type a name for the rule.
- 5. To specify when the rule should run, click **Set Condition**, and then enter the condition. The rule will run when the button in step one is clicked and the condition specified in this step is met.
- 6. Click Add Action.
- 7. In the **Action** box in the **Action** dialog box, click **Close the** form.
- 8. If you want to prompt the user to save their changes before the form closes, select the **If changes have not been saved, prompt the user to save** check box, and then click **OK**.
- 9. If you want to prevent any rules from running after this rule (for the current event), select the **Stop processing rules when this rule finishes** check box in the **Rule** dialog box.

Automatically open a form by using a rule

When you design a form, you can create a rule that automatically opens another form in response to an event. The event can include the following: a change to a particular field or group in the data source, clicking a button, or the insertion of a repeating section or row in a repeating table.

▶ Tip

- 1. In design mode, do one of the following:
- To run the rule when a particular field or group in the data source is changed, open the **Data Source** task pane, double-click the field or group, and then click the **Rules** tab.
- To run the rule when the value of a control is changed, doubleclick the control, and then on the **Data** tab in the **Control** Properties dialog box, click **Rules**.
- To run the rule when a button is clicked, double-click the button, click Rules and Custom Code in the Action list, and then click Rules.
- 2. In the Rules dialog box or on the Rules tab, click Add.
- 3. In the **Name** box in the **Rule** dialog box, type a name for the rule.
- 4. To specify when the rule should run, click **Set Condition**, and then enter the condition. The rule will run when the event specified in step one occurs and the condition specified in this step is met.
- 5. Click Add Action.
- 6. In the **Action** list in the **Action** dialog box, click **Open a new** form to fill out.
- 7. In the **Form ID or Location** box, click the form you want to open, and then click **OK**.
 - **Note** If the form you want is not listed, cancel the rule, open and close the form you want to automatically open, and then try

again.

8. If you want to prevent any rules from running after this rule (for the current event), select the **Stop processing rules when this rule finishes** check box.

Automatically query by using a data connection

When designing a form, you can create a rule that automatically queries an XML document, database, Web service, or Microsoft Windows SharePoint Services library or list in response to an event. This event can include a change to a particular field or group in the data source, the click of a button, the insertion of a repeating section or row in a repeating table, or the opening or submission of a form.

▶ Tip

- 1. In design mode, do one of the following:
- To run the rule when a particular field or group in the data source is changed, open the **Data Source** task pane, double-click the field or group, and then click the **Rules** tab.
- To run the rule when the value of a control is changed, doubleclick the control, and then on the **Data** tab in the **Control** Properties dialog box, click **Rules**.
- To run the rule when the form is opened, click Form Options on the Tools menu, click the Open and Save tab, and then click Rules.
- To enable the form's default submit command to run the rule, click **Submitting Forms** on the **Tools** menu, and then click **Enable Submit commands and buttons**. In the **Submit to** list, click **Custom submit using rules**, and then click **Rules**.
- To run the rule when a button is clicked, double-click the button, click **Rules and Custom Code** in the **Action** list, and then click **Rules**.
- 2. In the **Rules** dialog box or on the **Rules** tab, click **Add**.
- 3. In the **Name** box in the **Rule** dialog box, type a name for the rule.
- 4. To specify when the rule should run, click **Set Condition**, and then enter the condition. The rule will run when the event specified in step one occurs and the condition specified in this step is met.

- 5. Click Add Action.
- 6. In the **Action** list in the **Action** dialog box, click **Query using a** data connection.
- 7. Do one of the following:
- If you have already added a data connection that can query data, click it in the **Data connection** list.
- To add a new data connection, click **Add**, and then follow the instructions in the Data Connection Wizard.
- 8. If you want to prevent any rules from running after this rule (for the current event), select the **Stop processing rules when this rule finishes** check box in the **Rules** dialog box.

Automatically set a field's value

When designing a form, you can create a rule that automatically sets a field's value in response to an event. This event can include a change to a particular field or group in the data source, clicking a button, the insertion of a repeating section or row in a repeating table, or the opening or submission of a form.

▶ Tip

- 1. In design mode, do one of the following:
- To run the rule when a particular field or group in the data source is changed, open the **Data Source** task pane, doubleclick the field or group, and then click the **Rules** tab.
- To run the rule when the value of a control is changed, doubleclick the control, and then on the **Data** tab in the **Control** Properties dialog box, click **Rules**.
- To run the rule when the form is opened, click Form Options on the Tools menu, click the Open and Save tab, and then click Rules.
- To enable the form's default submit command to run the rule, click Submitting Forms on the Tools menu, and then click Enable Submit commands and buttons. In the Submit to list, click Custom submit using rules, and then click Rules.
- To run the rule when a button is clicked, double-click the button, click Rules and Custom Code in the Action list, and then click Rules.
- 2. In the **Rules** dialog box or on the **Rules** tab, click **Add**.
- 3. In the **Name** box in the **Rule** dialog box, type a name for the rule.
- 4. To specify when the rule should run, click **Set Condition**, and then enter the condition. The rule will run when the event specified in step one occurs and the condition specified in this step is met.

- 5. Click **Add Action**.
- 6. In the **Action** list in the **Action** dialog box, click **Set a field's** value.
- 7. In the **Field** box, click , and then in the **Select a Field or Group** dialog box, select the field you want to update.

Note If your form uses more than one data source, click the one you want in the **Data Source** list. If your form only uses one data source, the **Data Source** box is unavailable.

- 8. Do one of the following:
- To update the field with a static value, type the value in the Value box.
- To update the field based on a formula, click **Insert Formula f**, and then enter the formula in the **Insert Formula** dialog box.
- 9. If you want to prevent any rules from running after this rule (for the current event), select the **Stop processing rules when this rule finishes** check box in the **Rule** dialog box.

Automatically show a dialog box

When designing a form, you can create a rule that automatically shows a dialog box in response to an event. This event can be a change to a particular field or group in the data source, the click of a button, the insertion of a repeating section or row in a repeating table, or the opening or submission of a form.

▶ Tip

- 1. In design mode, do one of the following:
- To run the rule when a particular field or group in the data source is changed, open the **Data Source** task pane, double-click the field or group, and then click the **Rules** tab.
- To run the rule when the value of a control is changed, doubleclick the control, and then on the **Data** tab in the **Control** Properties dialog box, click **Rules**.
- To run the rule when the form is opened, click Form Options on the Tools menu, click the Open and Save tab in the Form Options dialog box, and then click Rules.
- To enable the form's default submit command to run the rule, click **Submitting Forms** on the **Tools** menu, and then click **Enable Submit commands and buttons**. In the **Submit to** list, click **Custom submit using rules**, and then click **Rules**.
- 2. To run the rule when a button is clicked, double-click the button, click **Rules and Custom Code** in the **Action** list, and then click **Rules**.
- 3. In the Rules dialog box or on the Rules tab, click Add.
- 4. In the **Name** box, type a name for the rule.
- 5. To specify when the rule should run, click **Set Condition**, and then enter the condition. The rule will run when the event specified in step one occurs and the condition specified in this step is met.
- Click Add Action.

- 7. In the **Action** list in the **Action** dialog box, do one of the following:
- To show a static message, click **Show a dialog box message**, and then type the message in the **Message** box.
- To show a dynamic message based on a formula, click **Show a dialog box expression**, click **Insert Formula**, and then enter the formula in the **Insert Formula** dialog box.
 - ▶ Tip
- 8. If you want to prevent any rules from running after this rule (for the current event), select the **Stop processing rules when this rule finishes** check box in the **Rule** dialog box.

About filtering data on a form

By using list box controls with repeating tables or repeating sections, you can create filters that allow users to show or hide data on their form. These filters display only the rows or sections that meet specific conditions. Users typically apply a filter by selecting entries in another control. For example, if you are designing a form that allows users to work with customer data in a repeating table, you can include a list box that allows them to filter customers by city.

When users move their mouse pointer over a filtered row or section on their forms, a filter icon appears on the shortcut menu button next to each row or section. This icon indicates that a filter has been applied. Users can click the shortcut menu button to see options for inserting or removing rows or sections, and also for refreshing the filtered data, if necessary. If you don't want users to see this icon, clear the **Show filter icon to indicate filtered items** check box in the properties dialog box for the repeating table or repeating section.

You can also use a filter to create cascading list boxes. In this scenario, a selection in the first list directly affects what appears in the second list. For example, you can create a **Manager** list and an **Employee** list on a form. When a user clicks a name in the **Manager** list, the **Employee** list displays only those employees who report to that manager.

Add a filter

When designing a form, you can allow users to filter data in a repeating section or repeating table, based on values that the user enters or selects elsewhere on the form. For example, if you use a repeating table to collect information about current projects, you can design a list box that allows users to filter projects by status.

Note To complete this procedure, your form must include both a control that filters data (typically a list box control) and a control that displays the filtered data (a repeating section or repeating table).

- Right-click the Repeating Table or Repeating Section label below the table or section on the form, and then click Control Properties on the shortcut menu.
- 2. On the **Display** tab in the **Control Properties** dialog box, click **Filter Data**.
- 3. In the **Filter Data** dialog box, click **Add**.

Note If you want to add a condition to an existing filter, click the filter that you want, and then click **Modify**.

4. In the **Specify Filter Conditions** dialog box, enter the filter condition.

Example

Note Microsoft Office InfoPath 2003 verifies a condition by checking the value of the field that is bound to the control inside the repeating table or repeating section, not the control itself.

- 5. To add additional filter conditions, click **And**, and then in the rightmost box, do one of the following:
- To apply the filter only when both conditions are true, click **and**.
- To apply the filter when one or both of the conditions are true, click **or**.

Notes

When users move their mouse pointer over rows or sections that have

been filtered, a filter icon appears on the shortcut menu button for each row or section. This provides users with a hint that their data is filtered and also allows them to reapply the filter, if necessary. If you don't want users to see the filter-specific shortcut menu button, clear the **Show filter icon to indicate filtered items** check box in the properties dialog box for the repeating table or repeating section.

Remove a filter

- 1. Right-click the **Repeating Table** or **Repeating Section** label below the table or section that has the filter that you want to remove, and then click **Control Properties** on the shortcut menu.
- 2. On the **Display** tab in the **Control Properties** dialog box, click **Filter Data**.
- 3. In the **Existing filters** list in the **Filter Data** dialog box, click the filter that you want to remove, and then click **Remove**.

Note If you want to remove a condition from an existing filter, rather than the filter itself, click the filter in the **Existing filters** list, and then click **Modify**. In the **Specify Filter Conditions** dialog box, click **Delete** next to the condition that you want to remove.

About previewing and testing forms

When designing a form, you can use the Preview Form feature to test the functionality and appearance of your form before you publish it. Previewing and testing your form allows you to see and work with the form from your user's perspective.

Note Unlike the Print Preview feature, which only provides a static image of the form, the Preview Form feature is interactive and provides a simple way for you to test your form.

When you click **Preview Form** on the **Standard** toolbar, your form opens in a new window. You can then test your form by entering data into controls to check various features, such as text formatting, conditional formatting, rules, formulas, and data validation. This allows you to identify mistakes in the form's design, and then quickly switch to the design window where you can correct them. To help you identify which window you are in when you are previewing a form, **Preview** or **Design** appears in the title bar of each window.

In addition to previewing a form, you can also use sample data to quickly test the appearance of your form. Sample data is placeholder text that appears in controls on your form, and provides an example of how text will appear in the control when a user fills it out. Sample data can only be seen in design mode, and is not visible when you preview or fill out a form.

You can also test your form by filling it out just as your users will do.

Preview a form

When designing a form, you can preview it to test its appearance and functionality, and to ensure that it will work as expected when users fill it out.

Note Before you publish your form, it is recommended that you preview and test its appearance and functionality.

On the **Standard** toolbar, click **Preview Form**. The form will open in a new window.

▶ Tips

Publish a form template

After designing a form template, you can make it available for others to use by publishing it to a shared folder on your computer or a network, to a Microsoft Windows SharePoint Services form library, or to a Web server.

Note Regardless of where or how you publish your form, users who want to fill out the form must have Microsoft Office InfoPath 2003 installed on their computers.

- 1. In design mode, click **Publish** on the **File** menu.
- 2. On the first page of the Publishing Wizard, click **Next**, and then do one of the following:
 - Publish to a shared folder
 - Publish to a SharePoint form library
 - Publish to a Web server

Save a form template's form files to a folder

When designing a form, you can view and work with a form template's form files by saving them to a folder. The Extract Form Files feature saves the form files to a folder, which allows you to manually make changes to individual files. For example, by extracting form files, you can manually make changes to the form definition (.xsf) file.

- 1. In design mode, open the form template whose form files you want to save to a project folder.
- 2. On the File menu, click Extract Form Files.
- 3. In the **Browse For Folder** dialog box, select a location in which to save the form files.

Save a form template without service pack features

When you design a form that uses the Microsoft Office InfoPath 2003 Service Pack features, you can export a version of that form for users who are unable to install the service pack. This will enable them to open the form.

For example, the service pack enables you to add a rule to your form that opens a dialog box when the user opens the form. If you export a version of the form without service pack features enabled, the rule isn't included and the user won't see the dialog box.

- 1. On the **Tools** menu, click **Form Options**, and then click the **Advanced** tab.
- 2. Click **Export**.
- 3. In the **Export** dialog box, browse to the folder where you want to save the form.
- 4. In the **File name** box, type a name for the form, and then click **Save**.

Run script when a user saves a form

- 1. On the **Tools** menu, click **Form Options**, and then click the **Open and Save** tab.
- 2. Under **Save behavior**, select the **Save using custom code** check box, and then click **Edit**.
- 3. In the Microsoft Script Editor (MSE), locate the following comments in the code, and replace them with the code you want to run before and after the user saves the form:
 - // Write the code to be run before saving here.
 - // Write the code to be run after saving here.

Specify a form upgrade method

When you modify a form template in design mode, the existing forms (.xml files) that are based on it may need to be upgraded so that they are compatible with the new version of the form template. This ensures that users can continue to work with their existing forms. Microsoft Office InfoPath 2003 allows you to choose how the existing forms should be upgraded.

- 1. In design mode, click Form Options on the Tools menu.
- 2. In the **Forms Options** dialog box, click the **Advanced** tab.
- 3. Under **Form template version** in the **On version upgrade** list, click the version upgrade method that you want to use.

Add service pack features to a form template

If you open a form template that was created prior to the installation of the Microsoft Office InfoPath 2003 Service Pack, you will need to choose whether to enable service pack features for the form template.

- 1. Open the form template.
- 2. When prompted, click **Yes** if you want to enable service pack features.
- 3. Save the form template to apply the new features.

Note To use forms that have service pack features enabled, users must have the InfoPath 2003 Service Pack installed on their computers.

Modify an existing form template

When you change a form template, including controls and fields or groups in the data source for the form template, you affect all of the existing forms (.xml files) that are based on that form template. Some changes to the data source can permanently delete data from existing forms. For example, if a user enters data that is stored in a field that you later remove, that data is permanently deleted from the data source and no longer appears on the form.

The following changes to a form template can cause data loss:

Renaming, moving, or removing a field or group from the data source

Changing a field or group from repeating to nonrepeating

Changing the data type of a rich text field

If you are certain that your changes to a form template will not cause data loss, you can work directly with the published form template. Otherwise, create a copy of the form template and store that copy in a staging area, such as your local hard disk. You can then change and test only the copy of the template, which helps ensure that data is not lost in forms that are based on that form template. After you finish changing and testing the copy of the form template, you can overwrite the published form template by publishing your copy.

- Modify a copy of a form template
- Modify a published form template

Move a published form template to a new location

If you publish a form template and then want to move it to a new location or change the Uniform Resource Locator (URL) for it, existing forms that are based on that form template may stop opening. If the form template has a Full Trust security level, you can move it without affecting existing forms that are based on it. If you want to move the form template from one intranet site to another, or to a new location within the same domain — for example, from http://www.example.com/form to http://www.example.com/expensereport— you must maintain the same form ID for the form template.

To maintain the form ID for a form template, follow these steps before you move the form template:

- 1. Open the published form template in design mode.
- 2. On the File menu, click Properties.
- 3. In the **Form Properties** dialog box, copy the form ID that appears in the **Form ID** box.
- 4. Publish the form template to the new location.
- 5. In design mode, open the form template from the new location.
- 6. On the File menu, click Properties.
- 7. In the **Form ID** box, replace the existing form ID with the one that you copied from the published form template.

Note If the form template has a security level of Domain and and you are moving it across different domains on an Internet site, none of its existing forms will open unless they are edited in Microsoft Notepad or edited by using an SDK tool. For more information, see the Microsoft Office InfoPath 2003 Software Development Kit (SDK).

About sharing forms on a SharePoint site

Many teams use Microsoft Windows SharePoint Services to collect information and share files. If this is the case in your organization, you can publish Microsoft Office InfoPath 2003 forms to form libraries, which are designed specifically for sharing collections of related forms. For example, a sales team might use a form library to fill out, save, and view sales report forms.

Form libraries act as a file server, where users can fill out and store forms based on the same form template. One form template is associated with each form library, so that every form created from that form library will have the same structure and appearance.

Although you can publish forms to any network location, publishing them to form libraries offers additional benefits for users, including the ability to merge multiple forms and export form library data to a spreadsheet.

Prerequisites for working with form libraries

To publish a form to a form library you must have access to a Windows SharePoint Services site. You must also have permissions to create a form library on the site.

To fill out forms in a form library users must have a network or Internet connection to the Windows SharePoint Services site, as well as permissions to add completed forms to the form library.

Designing forms for form libraries

While in design mode in InfoPath, you can do the following:

Specify form library columns In design mode, you can specify parts of the form you want to highlight in form library columns. Form library columns are useful when you want to look at important information in a form without having to open each form individually. For example, if you use a form library to store expense report forms, you could automatically create form library columns that show each employee's name and the total expenses claimed.

Link controls to SharePoint lists and libraries You can establish a connection between controls in your form and a Windows SharePoint Services library or list. For example, you might specify that items in a drop-down list box come directly from a Windows SharePoint Services list. Whenever users open the form, the contents of the list box update to reflect the latest changes to the Windows SharePoint Services list.

Publish a form to a form library You can use the **Publish** command on the **File** menu to open a wizard that guides you through the process of publishing a form to a form library. This wizard can also speed up the process of publishing a new version of an existing form.

Gain access to form libraries The Open from SharePoint Site command on the File menu offers immediate access to existing form libraries. This option helps you quickly find and modify form templates from within InfoPath.

Working with forms in form libraries

From the Windows SharePoint Services site itself, you can do the following:

Fill out forms In a form library, you can click the **Fill Out This Form** button in the default view to fill out a form in InfoPath. When you click **Save** in InfoPath, the form is automatically saved as an item in the form library.

Merge forms If form merging has been enabled, the form library will include a **Merge Forms** link that lets you select the forms you want to merge. This is particularly useful if you need to summarize information. For example, if you manage a team that fills out sales call reports, you might want to merge the names and phone numbers of potential sales leads into a single form, so that you have all of the data you need in one place.

Export data Depending on the form template and the way the form library is set up, form library columns may be able to display both data stored within the form and properties of the form. You can export the data in form library columns to a Windows SharePoint Services-compatible spreadsheet program, such as Microsoft Office Excel 2003.

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Create a form library on a SharePoint site

If your organization uses Microsoft Windows SharePoint Services to share information, you can publish a form template to a form library. Users can then fill out forms that are stored in that form library.

To publish a form to a form library, you must have access to a Windows SharePoint Services site, as well as permissions to create a form library on that site.

- 1. In design mode, open the form template you want to set as the default for all forms created in the form library.
- 2. On the File menu, click Publish.
- 3. On the first page of the Publishing Wizard, click **Next**.
- 4. Under Where do you want to publish this form, click To a SharePoint form library, and then click Next.
- 5. Under **Do you want to**, click **Create a new form library** (recommended), and then click **Next**.
- 6. Complete the remaining steps in the Publishing Wizard.
 - ▶ How?

Add, remove, or modify form library columns on a SharePoint site

As part of designing a Microsoft Office InfoPath 2003 form, you can specify whether or not you want form library columns to appear in the default view of a Microsoft Windows SharePoint Services form library. You can do this when you design your form, or you can wait until you publish the form to the form library.

- 1. In design mode, open the form template for which you want to specify form library columns.
- 2. Do one of the following:
- Specify form library columns when designing a form
- Specify form library columns when publishing a form

Note Depending on the form and the way that the form library is set up, form library columns on the Windows SharePoint Services site may display both data stored within the form and properties of the form itself.

Modify a form template from a form library

If you have published your form to a form library on a Microsoft Windows SharePoint Services site, you can access the form template directly from that form library and edit it in Microsoft Office InfoPath 2003.

Note Some changes made to a form template can result in data loss from existing forms that are based on that form template. For example, if you remove a control in a form template that collected specific data, the control will no longer appear in existing forms that are based on that form template, and the data will be lost. To prevent compatibility issues with existing forms, republish the new form template to a different location.

- 1. By using a Web browser, navigate to the form library.
- 2. Under **Actions** on the page that displays the form library, click **Modify settings and columns**.
- 3. In the **General Settings** section on the **Customize** page, click **Edit Template** in the **Template** field. The form template will open in InfoPath.
- 4. Make the changes you want, and then click **Save** on the **Standard** toolbar.

▶ Tip

Enable or disable form merging

- 1. In design mode, click **Form Options** on the **Tools** menu.
- 2. On the **General** tab in the **Form Options** dialog box, do one of the following:
- To enable merging, select the **Enable form merging** check box.
- To disable merging, clear the **Enable form merging** check box.

Design a form for merging

You can design a form template so that your users can merge the data from multiple forms into one form. When the forms are merged, the only parts of the form that are actually combined are those contained in a repeating table or repeating section, or associated with a list control or rich text box. The remaining parts of the form will not be merged.

- 1. In design mode, click **Form Options** on the **Tools** menu.
- 2. On the **General** tab in the **Form Options** dialog box, select the **Enable form merging** check box.

How InfoPath uses XML technologies

Microsoft Office InfoPath 2003 makes use of a variety of XML technologies, including XSLT, XML Schemas, XHTML, XPath, the DOM, XML Signatures, and an XML processor that works with all of these technologies. These technologies are governed by the World Wide Web Consortium (W3C), which recommends a specific set of rules, guidelines, and conventions for producing files that are easy to generate and process by different computers and applications. In addition to the XML technologies that InfoPath supports, InfoPath makes use of special processing instructions and namespaces and has a number of methods for accessing the DOM through the InfoPath object model.

InfoPath is a Microsoft Windows—based application that is used to create forms based on XML and its associated technologies. Although it is not necessary to understand all of the details of these XML technologies, the following table will help you understand how these technologies are used in InfoPath.

Name	Description
XML	XML is the output format that is produced by an InfoPath form. XML is also used to contain the sample data that is displayed in the fields of a form before a user fills it out.
XSLT	XSLT is the format of the view files that are produced when a form is designed. InfoPath does not support the use of XSL Formatting Objects (XSL-FO) for the presentation of XML data, nor does it support the importing of arbitrary .xsl files.
XML Schema	XML Schemas are used as the primary means of data validation in a form, and they provide the underlying structure of the XML document that is produced by the form. In addition, an XML Schema is used to define the structure of the form definition (.xsf) file that is generated by InfoPath when a form is designed. InfoPath does not support the use of XML-Data Reduced (XDR) or Document Type Definition (DTD) for defining the schema of XML documents.
XHTML	XHTML is well-formed HTML and is primarily used when developing rich text areas.
XPath	XPath expressions are used for binding controls to a form, as

	well as for data validation and conditional formatting expressions.
DOM	The DOM is used to programmatically access the contents of the source XML document that a form produces. It is also used to provide read-only access to the contents of the .xsf file.
XML Signature	XML Signatures are used to digitally sign forms that were created by InfoPath. Each InfoPath form can contain one or more digital signatures. InfoPath supports digital signing of portions of a form, as well as the form as a whole.
XML processor	The XML processor is used to load the source XML of a document into memory, validate it using the XML Schema, and produce the view of the document by using XSLT. The XML processor supported by InfoPath is Microsoft XML Core Services (MSXML) 5.0. Earlier versions of this processor are not supported.

About XML

Extensible Markup Language (XML) is a markup language defined by the XML Working Group of the World Wide Web Consortium (W3C). XML is similar to Hypertext Markup Language (HTML) in that XML is a tag-based language specifically designed for delivering information on the Web. However, XML is different from HTML in that the tags that XML uses are not predefined. Instead, the W3C XML recommendation specifies a set of rules that must be followed so that you can create your own meaningful set of tags.

- XML rules
- Structure of XML documents

About XSLT

Extensible Stylesheet Language Transformation (XSLT) is a declarative, XML-based language that is used to present or transform XML data. The transformation of XML data involves taking the source document, considered to be a tree of nodes, and creating a second XML document from it in another format. The XSLT style sheet provides the rules and format that specify how the output document is built. If the output is in an appropriate format, such as HTML, it can be used for presentation of the source document.

Using XPath expressions, XSLT matches components (or nodes) of an XML document with XSLT templates and then applies transformation and formatting rules to the matched components. XSLT templates are structures that specify the components of the XML document to be transformed, as well as the final format into which the matching components should be transformed. Because XSLT is implemented in XML, XSLT defines a special set of elements and attributes that can be used to create the transformations. XSLT can be used to transform any type of XML document, including XML data, XML Schemas, or other XSLT style sheets.

From a programming perspective, XSLT supports a number of general methods for processing XML data:

Data types A small set of data types, including Boolean, number, string, and node-set.

Operations Various operations that can be performed on the XML data, including **template**, **apply-template**, **sort**, and **output**.

Flow control Programming statements that are used to control the flow of the transformation, including **if**, **for-each**, and **choose**.

About XML Schemas

An XML Schema is an XML file with an .xsd file name extension. An XML Schema is used to define the structure and type of data that an XML document can contain. It specifies the elements, attributes, and data types that can be used in an XML document, along with the structure that must be followed in order for the XML document to be valid for that particular XML Schema.

Schema files have a root element named **schema**, and all of the element, attribute, and data type definitions are nested within the schema.

The XML Schema can also be used to validate the data contained in the elements and attributes. The following is a list of some of the types of data validation that can be performed by an XML Schema:

Data types Control the types of data that an element or attribute can contain. Examples include string, Boolean, float, integer, date, and time.

Constraining facets Place limits on the value space of data types. Examples include length, pattern, enumeration, minimum and maximum ranges, and precision.

Cardinality Controls the number of permitted occurrences. Examples include one, zero or one, zero or more, and one or more.

Choice Limits values to those contained in a list of values.

Sequence Defines the order in which elements may be used.

Defaults Provide values that are used when no other value is specified.

The namespace for XML Schemas is: xmlns:xsd="http://www.w3.org/2001/XMLSchema"

About XPath

XML Path Language (XPath) is a query language that is used to search for and retrieve information contained within the nodes of an XML document. In Microsoft Office InfoPath 2003, these nodes equate to fields or controls on a form. XPath queries are formed as expressions. These expressions are used to address the various parts of an XML document, to manipulate strings, numbers, and Boolean values, and to match a set of nodes within the document.

XPath views an XML document as a tree of different node types. XPath expressions identify these nodes in the XML document based on their type, name, and values, as well as the relationship of a node to other nodes in the document. An XPath expression yields one of the following basic objects:

Node set A collection of nodes that match the expression. Node types include root (document), element, attribute, namespace, processing instruction, comment, and text.

Boolean A **true** or **false** value.

Number A numerical value.

String A text value.

XPath expressions are formed by using patterns that specify the location path of a node or set of nodes, and that are relative to the context node. The context node is the section of the XML document from which you are currently positioned. It is represented in XPath as a period (.). XPath also supports the use of patterns that have absolute locations based on the document root. The document root is represented in XPath as a forward slash (/).

In addition to location paths, XPath has a number of functions that can be used to retrieve data from an XML document. The following is a list of the types of functions that XPath supports:

Node Used for working with the various node types. Examples of node functions include **name**, **node**, **processing-instruction**, **comment**, and **text**.

Positional Used for nodes that belong to a node set. These functions determine or match the position of the nodes. Examples of positional functions include **name**, **id**, **last**, and **count**.

Numeric Used to return numeric values. Examples of numeric functions include **ceiling**, **floor**, **number**, **round**, and **sum**.

Boolean Used to evaluate XPath expressions to **true** or **false**. Examples of Boolean functions include **boolean**, **false**, **not**, and **true**.

String Used to manipulate and parse string text. Examples of string functions include **concat**, **contains**, **starts-with**, **string**, **substring**, and **translate**.

About namespaces

XML namespaces are used to uniquely identify elements and attributes within an XML document to avoid naming conflicts. Namespaces are declared as a Uniform Resource Identifier (URI) and are typically located in the start tag of the root element of an XML document. However, they can also be defined at the node level, where they are valid for that node and all of its descendants. Associating a URI with a namespace ensures that elements with the same name remain distinct.

The Namespaces in XML recommendation, developed by the World Wide Web Consortium (W3C), provides the **xmlns** attribute to uniquely define a namespace for an XML document. The following is an example of a namespace that is used to define the vocabulary for Extensible Stylesheet Language Transformation (XSLT):

xmlns:xsl="http://www.w3.org/1999/XSL/Transform"

A namespace is divided into three sections:

xmIns keyword The keyword defined by the W3C for identifying XML namespaces. This section is defined first and is separated from the namespace prefix by a colon (:).

Prefix The abbreviated name of the namespace that is used in declaring all elements and attributes that belong to it. The prefix can be any name other than xmlns or xml, which are reserved words.

Definition The URI that uniquely defines the namespace. Can also be defined as a URN or a URL.

After a namespace is declared for an XML document, any element or attribute that belongs to that namespace uses the namespace prefix in its declaration. For example, .xslt files use the xsl namespace prefix, as shown in the following example:

...

</xsl:template> </xsl:stylesheet>

About processing instructions

Processing instructions (PIs) are used to provide information in an XML document that the XML parser passes on to the calling application. PIs are primarily used to tell an application how to handle the data contained within an XML document. PIs must begin with an identifier called a *target*, which follows rules similar to those for element and attribute names. Targets are case-sensitive and must start with a letter or underscore. The rest of the target can contain letters, numbers, hyphens, underscores, periods, and colons. Any valid XML textual characters can appear after the target. The following is an example of a processing instruction that is used to indicate that the XML document should be displayed by using an XSL style sheet:

<?xml-stylesheet type="text/xsl" href="MyStylesheet.xsl"?>

PIs can be placed anywhere in an XML document outside of other markup. They can be placed in the prolog, following the root element, or within the value of an element. They are usually placed in an XML document's prolog.

About the XML DOM

The XML Document Object Model (DOM) is used to access and update the content, style, and structure of XML documents through a programmatic interface. This interface takes the form of an object model made up of objects, properties, methods, and events that can represent and manipulate the components of an XML document. The DOM stores the data of an XML document in a hierarchical, tree-like structure that mimics the structure of the document itself.

The DOM can be used to access any component of an XML document, including its elements, attributes, processing instructions, comments, and entity declarations. Any XML document can be loaded into the DOM. When an XML document is loaded into the DOM, it is read from start to finish and stored in the DOM as a logical model of nodes.

The DOM programming interface enables applications to traverse the tree and manipulate its nodes. Each node is defined as a specific node type according to the XML DOM enumerated constants, which also define valid parent and child nodes for each node type. For most XML documents, the most common node types are element, attribute, and text. Attributes occupy a special place in the object model because they are not considered child nodes of a parent; instead, they are treated as properties of elements.

The following table lists some of the DOM objects that can be used to work with an XML document, along with the XML node types that they represent.

DOM object	XML node type
XMLDOMDocument	Represents the XML document as a whole. This object exposes properties and methods that allow you to navigate, query, and modify the content and structure of an XML document.
XMLDOMNode	Represents a single node in the document tree. This object is the base object for accessing data in the XML DOM, and it includes support for data types, namespaces, and XML Schemas.
XMLDOMNodeList	Represents a collection of nodes. This object

	enables iteration and indexed access operations on the live collection of IXMLDOMNode.	
XMLDOMElement	Represents an element of the XML document.	
XMLDOMAttribute	Represents an attribute of the XML document.	

About XML processors

XML processors are software programs that are used to parse XML documents and provide access to the documents' content and structure. XML processors can also validate the structure of XML documents and transform the documents into a variety of formats.

XML documents are text files that contain data that is formatted as a series of elements and attributes. Elements within an XML document can also be thought of as nodes of the document. The root element of the document is the root node, and all of the other elements are child nodes (or descendants) of the root node. Attributes are additional data associated with an element and are considered to be properties of the element. When an XML processor first loads an XML document, it parses the nodes of the document and places them in random access memory (RAM) in the form of a Document Object Model (DOM).

After an XML document has been parsed and loaded into the DOM, you can use the programmatic interfaces of the DOM to access and manipulate the XML document that the DOM contains. If an XSL Transformation (XSLT) is associated with an XML document, the XML processor can be used to apply the XSLT to the document to produce the specified output format.

About XML Signatures

XML Signatures are digital signatures that are designed for use in XML-based transactions. They are defined through a joint effort of the World Wide Web Consortium (W3C) and the Internet Engineering Task Force (IETF). XML Signatures contain a series of XML elements and attributes that define various components that are needed to sign an XML document more securely. This includes information about the location of the XML document to be signed, the algorithm used for signing, and information about the key used for signing.

An important feature of XML Signatures is the ability to have multiple digital signatures contained within one XML document. This allows multiple users to work on the same XML document, with each user providing a new digital signature on top of signatures that are added by other users.

Adding a digital signature to an XML document helps ensure that the data contained in the document is more secure, and that the identity of the person who signed the XML document can be verified.

About security features of InfoPath

The security features of Microsoft Office InfoPath 2003 are specific to designing, filling out, and deploying forms, and they are related to security zone and level settings in Microsoft Internet Explorer. These security features are designed to help protect data in your InfoPath forms and on your computer from unsafe operations, such as accessing or sending data to a source that you do not trust.

Security levels for forms

InfoPath provides three security levels for forms: Restricted; Domain; and Full Trust. The security levels determine whether a form can access data on other domains or access files and settings on your computer. The security levels also affect the features on a form when users fill it out. For example, if the form contains a list box that displays data from a Microsoft Office Access 2003 database, the security level for the form can determine whether the form opens and whether the form accesses the database or displays a security message without accessing the database.

When you design a form, InfoPath automatically selects the appropriate security level for the form, based on the form's features. The selected setting is as restrictive as possible. If you prefer a different security level for a form that you are designing, you can override the default setting by manually choosing a different security level for the form. If a form requires a security level other than the one that you or a user gives it, the form either does not open or does not work correctly. For example, if you design a form and specify that it requires a Full Trust security level, the user must grant full trust to that form when filling it out. Otherwise, the form does not open.

Forms run in one of three security levels, depending on where they are located, how they are installed, and whether they are digitally signed. The security levels are as follows.

Restricted

When running at the Restricted security level, a form can access only content that is stored in the form itself. This means that the following features do not work correctly:

Custom task panes

Data connections, except submission through an e-mail message

Microsoft ActiveX controls and custom controls

Managed code and script

Roles based on locations in an Active Directory directory service

Rules associated with opening forms

Print views for Microsoft Office Word 2003

Custom dialog boxes

Linked pictures

Human workflow services

Domain

When running at the Domain security level, a form can access content that is stored in the form itself and content that is stored in any of the following locations:

Same domain as the form

Domains that are included in the Trusted sites zone in Internet Explorer

Content in the Local computer zone in Internet Explorer, although a security message may appear before the content is accessed

Content in the Local intranet zone in Internet Explorer, although a security message may appear before the content is accessed

When a form accesses content in a zone, it does so according to the security levels specified for that zone in Internet Explorer. By default, a form in the Restricted sites zone in Internet Explorer cannot open because scripting is disabled for all content in that zone. A form in the Internet zone in Internet Explorer can open, but it cannot access content that is stored in a different domain.

Full Trust

When running at the Full Trust security level, a form can access content that is stored in the form itself and content from any of the following locations:

Same domain as the form

All other domains, without first displaying a security message about

accessing the content

Files and settings on the computer; all of the same resources that the person who is filling out the form can access on that computer

The Full Trust security level can be applied only if the form template is digitally signed with a trusted root certificate or if the form was installed on the computer by using an installation program such as Microsoft Windows Installer (.msi file).

Additional security features for forms

InfoPath provides additional features that can help you control the security of your forms. These features include:

Form design protection If you design a form, you can use this feature to prevent users from opening the form in design mode when they are filling it out. Note that the setting does not lock the form completely. Users can still open or modify the form's design by opening the form directly in design mode. In such cases, however, users receive a message stating that the form is protected.

Digital signatures When filling out a form, you can digitally sign a form or specific parts of the form. Doing so helps authenticate you as the person who filled out the form and helps ensure that the contents of the form are not altered.

In addition, you can digitally sign a form template that you design and then set the security level for that form template to Full Trust. Such forms, when deployed using an e-mail program, can be more efficiently updated using e-mail.

Customization for save, print, send, and export If you design a form, you can use these settings to turn on or off specific commands and options within InfoPath that allow users to save, print, send, or export a form that they filled out.

Trusted publishers and trusted forms These settings enable you to manage the list of form developers and publishers that you trust, and specify whether trusted forms can access files and settings on your computer when you fill them out. Trusted forms are forms that are installed on your computer, or forms that are digitally signed with a trusted root certificate and have a security level of Full Trust.

When you install a form on your computer, you automatically enable the form to access the files and settings on your computer. A digitally signed form that you did not install on your computer, however, cannot automatically access the files and settings on your computer. To enable such access for a digitally signed form, you can use the trusted form setting to change the security level for digitally signed forms.

Merge If you design a form template, you can specify whether users can import data from forms that are based on that form template into a single form. If you disable form merging, the **Merge Forms** command on the **File** menu is unavailable. This can help prevent users from seeing sensitive data.

Internet Explorer security zones and levels

In Internet Explorer, security zones and levels enable you to specify whether a Web site can access the files and settings on your computer and how much access those sites can have. InfoPath uses some of these settings to determine whether a form can access the files and settings on your computer and how much access that form can have. InfoPath also uses some of these settings to determine whether a form can access content that is stored in domains other than the domain in which the form is stored. For information about how security zones and levels affect security levels for InfoPath forms, see the preceding section, "Security levels for forms."

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Set the required security level for a form

When you design a form, Microsoft Office InfoPath 2003 automatically selects the appropriate security level for the form, based on the features of the form. The selected setting is as restrictive as possible. You can override the setting and manually choose a different security level for a form.

- 1. In design mode, click **Form Options** on the **Tools** menu.
- 2. In the **Form Options** dialog box, click the **Security** tab.
- 3. Under Security Level, clear the Automatically determine security level based on form's design (recommended) check box.
- 4. Click the security level that you want.

Notes

If the form requires a Full Trust security level, the form must be either installed on the user's computer or digitally signed with a trusted root certificate. Otherwise, the user will not be able to grant full trust to the form.

The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. Learn more

Allow trusted forms to access files and settings on your computer

You can specify whether you want to allow trusted forms to access files and settings on your computer. Trusted forms are forms that are either installed on your computer or digitally signed with a trusted root certificate, and have a security level of Full Trust.

When you install a form on your computer, you automatically enable the form to access files and settings on your computer. A digitally signed form that you did not install on your computer, however, cannot automatically access files and settings on your computer. To enable such access for a digitally signed form, you must change the security setting for all digitally signed forms.

Security To help protect the security of your computer and data, use caution when you install forms or allow forms to access files and settings on your computer. You should install or open forms only from trusted sources, and you should question messages, attachments, or files that you receive from unexpected sources. If a file has an unusual file name, time stamp, file name extension, file size, or other property, do not open the file.

- 1. On the **Tools** menu, click **Options**.
- 2. Under **Security options** in the **Options** dialog box, select or clear the **Allow fully trusted forms to have access to files** and settings on my computer check box.

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Mark a site as trusted to disable security messages

If a form accesses data from a Web site outside its own domain, you might receive security messages when you open the form. If you are certain that the form is accessing data from a trusted source, you can prevent the messages from appearing by adding the site to your Trusted sites zone in Microsoft Internet Explorer.

Note The Trusted sites zone specifies security settings for sites that you trust— sites that you can download or run files from without worrying about damage to your computer or data. The default security level for the Trusted sites zone is Low. Therefore, Internet Explorer allows all cookies from Web sites in this zone to be saved to your computer and read by the Web site that created them. In addition, depending on the security level of the form, Microsoft Office InfoPath 2003 allows some forms to access data from trusted sites without displaying a security message.

- 1. In InfoPath, click **Options** on the **Tools** menu.
- 2. On the **General** tab in the **Options** dialog box, click **Internet Options**.
- 3. In the Internet Properties dialog box, click the Security tab.
- 4. In the Select a Web content zone to specify its security settings list, click Trusted sites.
- 5. Click Sites.
- 6. In the **Add this Web site to the zone** box in the **Trusted sites** dialog box, type the address of the Web site that you want to add, and then click **Add**.

Note If the Web site address does not start with **https://**, clear the **Require server verification (https:) for all sites in this zone** check box. This stops InfoPath and Internet Explorer from verifying that the server for each Web site in the zone is secure before connecting to it.

Enable or disable digital signatures

When designing a form, you can specify whether or not users are able to use digital signatures when they fill out the form. Once signed, the form or part of the form that was signed cannot be altered without invalidating the signature.

Note If a form was designed based on an XML Schema, you can enable digital signatures for the form only if the XML Schema has a digital signature namespace.

- ▶ Enable digital signatures for the entire form
- ▶ Enable digital signatures for part of the form
- Disable digital signatures

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Digitally sign a form template

If you digitally sign a form template with a trusted root certificate, you can set the security level for the form template to Full Trust, which means the form can access files and settings on the user's computer or on a different domain. In addition, you can deploy and update that form template by sending it and updated versions of it to others by using an email program.

- 1. In design mode, click **Form Options** on the **Tools** menu.
- 2. In the **Form Options** dialog box, click the **Security** tab.
- 3. Under Form Signing, select the Sign this form check box.
- 4. Click Select Certificate.
- 5. In the **Select Certificate** dialog box, click the certificate that you want to digitally sign the form with.

Note If a certificate is not trusted on a user's computer, Microsoft Office InfoPath 2003 displays a security message and requires the user to enable trust for the certificate issuer before the user can open the form.

Note The information in this topic may not apply if you are working with a form designed using Microsoft Office InfoPath 2003 without the service pack installed. <u>Learn more</u>

Search Japanese text

The feature described in this Help topic is available only if support for Japanese is enabled through Microsoft Office Language Settings.

- 1. On the **Tools** menu, click **Options**, and then click the **Japanese Find** tab.
- 2. Under **Treat as equal**, select the check box for one or both of the following:
- Match minus/dash/cho-on Searches without distinguishing among minus signs, dashes, and long vowel sounds.
- Match cho-on used for vowels Searches without distinguishing between characters with long vowel sounds and plain characters.

Adjust line breaks for East Asian text

The feature or some of the options described in this Help topic are only available if support for Japanese, Simplified Chinese, Traditional Chinese, or Korean is enabled through Microsoft Office Language Settings.

- 1. Place the insertion point in the text where you want to apply a line break.
- 2. On the **Format** menu, click **Asian Typography**.
- 3. Under Line break in the Asian Typography task pane, click Normal or Strict.
- 4. If you are designing a form and you want to apply a default line break setting to all new forms that you design, click **Options** in the **Asian Typography** task pane. Under **Asian typography line break** in the **Options** dialog box, click the line break option you want, and then click **Set as default for new forms**.

Note When filling out a form, you can adjust line breaks for East Asian text in rich text boxes, but you cannot apply a default line break setting for new forms.

Automatically adjust character spacing for mixed East Asian and Latin text

The feature or some of the options described in this Help topic are only available if support for Japanese, Simplified Chinese, Traditional Chinese, or Korean is enabled through Microsoft Office Language Settings.

When you type East Asian characters and numerals together, or East Asian text in the same paragraph with Latin alphabet characters or numerals, the spacing between certain characters or punctuation may become irregular and make the text difficult to read. In such cases, Microsoft Office InfoPath 2003 allows you to adjust the character spacing automatically.

- 1. On the **Format** menu, click **Asian Typography**.
- 2. In the **Asian Typography** task pane, do one of the following:
- To adjust the character spacing between East Asian and Latin alphabet characters, select the Automatically adjust space between Asian and Latin text check box.
- To adjust the character spacing between East Asian characters and numerals, select the **Automatically adjust space between Asian text and numbers** check box.

InfoPath Glossary

ABCDEFGHIJKLM NOPRSTUVWX

Α

Back to top

absolute URL

The full address of a page or other resource on the Internet. An absolute URL includes a protocol, such as "http," a network location, and an optional path and file name. For example, http://www.example.com/ is an absolute URL.

action

A part of a rule that is used in conjunction with conditions to automatically display dialog boxes, set values, query and submit data, and open forms.

active field

The field on a form in which the cursor or pointer is positioned.

ActiveX

A set of technologies that enables software components to interact with one another, regardless of the language in which the components were created.

ActiveX control

A reusable software component that incorporates Microsoft ActiveX technology. ActiveX controls can be inserted onto forms to produce interactive content. An ActiveX control can be written in a variety of programming languages, including C, C++, Java, and Visual Basic.

attribute field

A field in the data source that can contain data and that is an attribute, instead of an element. Attribute fields cannot contain other fields.

Back to top

bind

To connect a control to a field or group in the data source so that data entered into the control is saved to the corresponding field or group. When a control is unbound, it is not connected to a field or group, and data entered into the control is not saved.

C

Back to top

cascading list box

Two or more list boxes that are connected to each other in such a way that selecting a value in one changes the set of choices in the other.

cascading style sheet (CSS)

One or more data formatting declarations that determine the appearance of specified elements in an .html or .xml file. CSS can be embedded in a Web page or .xsl file, or stored in a separate .css file linked to a Web page or .xsl file.

child element

In an Extensible Markup Language (XML) tree structure, the element contained by another element, which is called the parent element.

choice group

A control that is used to present a set of mutually exclusive choices. By default, a choice group contains two choice sections, one of which appears to the user as the default choice on the form.

choice section

A control that contains other controls and is used to represent a single choice within a set of mutually exclusive choices. Users can replace one choice section with another when filling out a form.

closed content model

A set of Extensible Markup Language (XML) document rules stating that an element in an XML document cannot include any information

that is not declared in the schema that the document references.

COM

See Component Object Model

Component Object Model (COM)

A specification developed by Microsoft for building software components that can be assembled into programs or can add functionality to existing programs that are running on Microsoft Windows operating systems.

conditional formatting

A set of conditions that change the appearance or content of a control based on values entered into the form.

control

An object, such as a text box, check box, or command button, that enables users to enter or display data, display data choices, or perform an action. Some controls, such as sections, can also make a form easier to read.

CSS

See cascading style sheet

CSS selector

In a cascading style sheet, a name identifying a user-defined style. Depending on how it is defined, a CSS selector can be used with a single type of element or any element in a document.

custom installed form

A form that is installed with a custom setup program, such as Microsoft Windows Installer, and that usually has access to files and settings on the computer.

custom task pane

An area within the Microsoft Office InfoPath 2003 program window that displays a custom .htm file. Custom task panes can provide form-specific commands and Help content.

D

Back to top

data adapter

An object used to submit data to and retrieve data from databases, Web services, and Extensible Markup Language (XML) files.

data connection

The connection between a Microsoft Office InfoPath 2003 form and an external data source, such as a database, Web service, Microsoft Windows SharePoint Services library, or Extensible Markup Language (XML) file. Data connections are used to query and submit data from a form.

data source

The collection of fields and groups that define and store the data for a Microsoft Office InfoPath 2003 form. The main data source contains Extensible Markup Language (XML) data that is saved with the form. A secondary data source contains XML data retrieved through a data connection, from an external location, such as a database. Controls on a form are bound to the fields and groups in a data source.

data type

A property of a field that defines the kinds of data the field can store. Data types include Text (string), Rich Text, Whole Number, Decimal, True/False (Boolean), Hyperlink, Date, Time, Date and Time, and Picture.

data validation

The process of testing the accuracy of data; a set of rules you can apply to a control to specify the type and range of data that users can enter into that control.

declarative programming

Rule-based programming in which each rule is independent and the order in which rules are declared or executed is not significant.

default role

A setting associated with a particular user role. Users who are not assigned to an existing role are automatically assigned to the role that is specified as the default.

design mode

The Microsoft Office InfoPath 2003 design environment in which you

create or modify a form template.

DHTML

See Dynamic HTML

dialog box alert

A data validation message that opens a dialog box that displays a custom error message when invalid data is entered into a control.

digital signature

An electronic, encryption-based stamp of authentication on a form or a section of a form. This signature confirms that the form or section originated from the signer and has not been altered.

Document Object Model (DOM)

A World Wide Web Consortium (W3C) specification that describes the structure of dynamic Hypertext Markup Language (HTML) and Extensible Markup Language (XML) documents in a way that allows them to be manipulated through a Web browser.

Document Type Definition (DTD)

The predecessor of a schema definition (.dtd) file that specifies rules for elements, attributes, and relationships in an Extensible Markup Language (XML) document.

DOM

See Document Object Model

DTD

See Document Type Definition

Dynamic HTML (DHTML)

An extension of Hypertext Markup Language (HTML) that allows you to add interactivity to content. DHTML supports multimedia and database access, and it provides an object model. Content that uses DHTML can change and update dynamically in response to user actions, and programs can change the styles and attributes of page elements or objects in that content as well as replace the existing elements or objects.

embedded object

An object that is created in a source file and inserted into a destination file, without maintaining a connection between the two files. After it is embedded, the object becomes part of the destination file and changes that you make to the embedded object are reflected only in the destination file.

event

An action, such as a mouse click or key press, that is recognized by an object and for which you can define a response. An event can be caused by a user action, script or programming code, or the operating system.

event handler

The script function code in a form that responds to a form event or data validation event.

expression box

A read-only control on a form that displays the result of an XPath expression, which retrieves or calculates the result from other data in the form.

Extensible Hypertext Markup Language (XHTML)

A markup language that extends Hypertext Markup Language (HTML) and reformulates it as Extensible Markup Language (XML).

Extensible Markup Language (XML)

A subset of Standard Generalized Markup Language (SGML), a language for creating customized data structures that allow for the definition, transmission, validation, and interpretation of data between programs, servers, and organizations.

Extensible Stylesheet Language (XSL)

An Extensible Markup Language (XML) vocabulary that is used to transform XML data into another form, such as a Hypertext Markup Language (HTML) document, and that defines the format or presentation of the data in the resulting document.

F

Back to top

field

An element or attribute in the data source that can store data entered into controls on a form. If the field is an element, it can contain attribute fields.

file attachment control

A control that allows users to attach files to a form when they fill it out. A form designer can limit the types of files that users can attach.

filter

A set of conditions that is applied to data in order to display a subset of the data.

form

A document with a set of controls into which users can enter information. Microsoft Office InfoPath 2003 forms can contain features such as rich text boxes, date pickers, optional and repeating sections, data validation, and conditional formatting.

form area

The area of the Microsoft Office InfoPath 2003 workspace that displays the form you are working with.

form definition file

An Extensible Markup Language (XML) file that contains information about all other files and components within a form, including user interface customization, XML Schemas, views, business logic, events, and deployment settings. Form definition files have an .xsf file name extension.

form files

A collection of files that are used to implement a form. File types can include .html, .xml, .xsd, .xslt, script, and other file types that are necessary to support the functionality of the form.

form library

A folder on a Microsoft Windows SharePoint Services site that stores a collection of forms that are based on the same form template. Each form in a form library is associated with user-defined information that is displayed in the content listing for that library.

form library columns

The vertical arrangement of data into one or more side-by-side sections of a form library on a Microsoft Windows SharePoint Services site. For every form in the library, the columns display information entered into controls that the form designer associated with those columns.

form template

A file or set of files that defines the data structure, appearance, and behavior of a form.

formula

An XPath expression that consists of values, fields or groups, functions, and operators. Formulas can be used to calculate mathematical values, display dates and times, and reference fields.

function

A predefined XPath expression that returns a value based on the results of a calculation.

G

Back to top

GIF

See Graphics Interchange Format

Graphics Interchange Format (GIF)

A graphics file format frequently used to display indexed-color graphics on the World Wide Web. GIF is a compressed format that is designed to minimize file transfer time over the Internet, supports up to 256 colors, and uses lossless compression, which means that no image data is lost when the file is compressed. GIF-format files have a .gif file name extension.

group

An element in the data source that can contain fields and other groups. Controls that contain other controls, such as repeating tables and sections, are bound to groups.

Back to top

HREF

See hypertext reference

HTML

See Hypertext Markup Language

hyperlink

A pointer from text, a graphic, or another page element to a Web page or file. On the World Wide Web, hyperlinks are the primary way to browse between Web pages and Web sites. A hyperlink is also called a link.

Hypertext Markup Language (HTML)

The standard markup language for documents on the World Wide Web. HTML is a subset of Standard Generalized Markup Language (SGML). It uses tags to indicate how Web browsers should display page elements such as text and graphics, and how Web browsers should respond to actions such as the activation of a link by a key press or mouse click.

hypertext reference (HREF)

An attribute in a Hypertext Markup Language (HTML) document that defines a link to another location within the document, another document, or an e-mail address.

Back to top

initiator role

An optional setting associated with a specific user role. A user who opens a new form is automatically assigned to this role, which overrides any other role assignments for that user.

ink

The writing or drawing strokes made with a tablet pen.

ink entry mode

The environment for filling out a form by writing in ink. On a Tablet PC, Microsoft Office InfoPath 2003 automatically opens in ink entry

mode.

ink guide

A shaded area that increases in size when data is written in a form field with a tablet pen.

ink section

An area inside a rich text box where users can draw or write with a tablet pen. Microsoft Office InfoPath 2003 saves handwritten notes or drawings inside an ink section as ink.

inline alert

A data validation message that adds a dashed red border to controls that contain invalid data. Users can right-click a control with an inline alert to display an error message that explains the validation error.

Internet

The worldwide collection of computers, networks, and gateways that use TCP/IP protocols to communicate with one another.

intranet

A network within an organization that uses Internet technologies and protocols but is available only to certain people, such as employees of a company. An intranet is also called a private network.

J

Back to top

Joint Photographic Experts Group (JPEG)

A graphics file format used to display high-resolution color graphics on the World Wide Web. JPEG graphics apply a user-specified compression scheme that can significantly reduce the file sizes of photo-realistic color graphics. A higher level of compression results in lower quality, whereas a lower level of compression results in higher quality. JPEG-format files have a .jpg file name extension.

JPEG

See Joint Photographic Experts Group

JScript

An interpreted, object-based scripting language that borrows from C,

C++, and Java. It is the Microsoft implementation of ECMA script, as defined by the specification of the European Computer Manufacturers Association.

K

Back to top

key field

A field in the data source that links related data in master and detail controls.

L

Back to top

layout table

A framework that includes rows and columns for organizing and arranging content on a form, including controls, control labels, sections of a form, logos, and other types of graphics.

linked object

An object that is created in a source file and inserted into a destination file, while maintaining a connection between the two files. The linked object in the destination file is updated automatically when the source file is updated.

list control

A control on a form that can be formatted as a bulleted, numbered, or plain list. A user can enter text into a list control, which repeats as needed.

M

Back to top

master/detail control

Two related controls in which selection in one control (the master) determines what is displayed in the other control (the detail). When a user selects data in a master control, additional information about

that data appears in a detail control.

metadata

Data that describes other data. For example, the words in a document are data; the number of words in the document is an example of metadata for that document.

method

A procedure similar to a statement or function that operates on specific objects. For example, in the Microsoft Visual Basic Editor, you can apply the **Print** method to the **Debug** object to transfer printed output to the Immediate window.

Microsoft Script Editor (MSE)

A programming environment used to create, edit, and debug Microsoft JScript or Microsoft Visual Basic Scripting Edition code in a Microsoft Office InfoPath 2003 form.

Microsoft Windows Installer

A tool that is used to create installation packages for software that is to be deployed on Windows operating systems.

MSDN

A Microsoft Web site and subscription program that provides developers with the latest tools, information, code samples, and training materials for Microsoft development tools (http://msdn.microsoft.com).

MSE

See Microsoft Script Editor

Ν

Back to top

namespace

A mechanism that uniquely qualifies element names and relationships to avoid name collisions on elements that have the same name but come from multiple sources.

object

A table, chart, graphic, equation, or other form of information. Objects that are created in one program, such as spreadsheets, and linked or embedded in a document for another program are OLE objects.

object browser

In the Microsoft Visual Basic Editor, a window that displays information about objects, properties, methods, and constants in the current project and in referenced object libraries. You can use the window to search for an element, get help on it, or paste it into a code module.

object model

A hierarchical library of objects and collections that represents the content and functionality of a program. Objects and collections have properties, methods, and events that can be used to manipulate and interact with a program.

object model diagram

A graphical representation of the objects and collections within a specified object model.

offline

In reference to one or more computers, not being connected to a network or the Internet.

open content model

A set of Extensible Markup Language (XML) document rules stating that an element in an XML document can contain child elements and attributes that are not declared in the schema that the document references.

optional section

A control on a form that contains other controls and that usually does not appear by default. Users can insert and remove optional sections when filling out the form.

Back to top

parameter

A value that is assigned to a variable at the beginning of an operation or before an expression is evaluated by a program. A parameter can be text, a number, or an argument name assigned to another value.

parent element

In an Extensible Markup Language (XML) document, the element that contains another given element, which is called the child element.

PNG

See Portable Network Graphics

Portable Network Graphics (PNG)

A file format for compressed bitmap graphics, similar to the GIF format. PNG supports variable transparency of images (alpha channels) and control of image brightness on different computers (gamma correction). PNG-format files have a .png file name extension.

processing instructions

Information stored in the prologue of an Extensible Markup Language (XML) document. This information is passed through the XML parser to any application that uses the XML document.

property

A named attribute of a control, a field, or an object that defines a characteristic of an object, such as size, color, or screen location, or an aspect of its behavior, such as whether the object is hidden.

publish

The process of making a form available for others to fill out. A published form template is saved to a shared location such as a Web server or a file server.

R

Back to top

recursive section

A control that contains other controls, can be inserted within itself, and is bound to nested reference fields.

reference field

A field that is associated with another field so that their properties always match. If the properties in one field are changed, the properties in the other field are updated automatically.

reference group

A group that is associated with another group so that their properties and the properties of the groups and fields they contain always match. If the properties of one group are changed, the other group is updated automatically.

relative URL

The address of a document or other resource in relation to the active document or resource. Unlike an absolute URL, a relative URL does not include the domain of or the full path to the resource. It might include only the file name and file name extension, and possibly part of the path to the resource.

repeating field

A field in the data source that can occur more than once. Controls such as repeating sections, repeating tables, and bulleted, numbered, and plain lists can be bound to repeating fields.

repeating group

A group in the data source that can occur more than once. Controls such as repeating sections and repeating tables are bound to repeating groups.

repeating section

A control on a form that contains other controls and that repeats as needed. Users can insert multiple sections when filling out a form that contains repeating sections.

repeating table

A control on a form that contains other controls in a table format and that repeats as needed. Users can insert multiple rows when filling out a form that contains repeating tables.

Resource Manager

The interface for adding and removing resource files that support the functionality of a form.

rich text field

A field that can show formatting and graphics, such as an embedded object, in addition to text.

root element

The element in an Extensible Markup Language (XML) document that contains all other elements in that document. It is the top-level element of an XML document and must be the first element in the document.

rule

One or more conditions and actions that automatically perform tasks based on events and values in the form.

S

Back to top

script

A type of computer code that sends a set of instructions to a program, such as a Web browser. A script is run directly by a program that can interpret the language in which the script is written. On the World Wide Web, script is frequently used to customize or add interactivity to a Web page. In Microsoft Office InfoPath 2003, script is used to add functionality to a form.

scrolling region

A control that contains other controls and that can display vertical or horizontal scroll bars. Scrolling regions are used for layout purposes only and are not bound to fields or groups in the data source.

section

A control on a form that contains other controls.

server

On a network, a computer that provides access to files and other resources that are also part of the network. On the World Wide Web, a computer that hosts Web pages and responds to requests from

Web browsers. Also referred to as a host.

shortcut key

A function key or key combination, such as F5 or CTRL+A, that you can press to perform a command. In contrast, an access key is a key combination, such as ALT+F, that you can press to move the focus to a menu, command, or control.

shortcut menu

A menu that shows a list of commands relevant to a particular item. To display a shortcut menu, right-click an item or press SHIFT+F10.

sizing handle

One of the small circles or squares that appear at the corners and sides of a selected object. You drag these handles to change the size of the object.

structural editing

Editing that occurs within a structured document, such as a well-formed Extensible Markup Language (XML) document.

subscript

Text that is slightly lower than other text on a line. Subscripts are often used in scientific formulas.

superscript

Text that is slightly higher than other text on a line, such as a footnote reference mark.

Т

Back to top

table

One or more rows of cells frequently used to display numbers and other items for quick reference and analysis. Items in a table are organized into rows and columns.

Tablet PC

A computer that runs Microsoft Windows XP Tablet PC Edition. With a Tablet PC, you can write directly on the screen by using a tablet pen and use the pen to perform mouse functions. Also referred to as

a tablet computer.

tablet pen

A pen-like device that comes with a Tablet PC and is used to interact with the tablet screen.

task pane

An area within a Microsoft Office program window that contains commonly used commands. Its location and small size allow you to use these commands while viewing and working on files that also appear in the program window.

text box

A control into which a user can enter text. Rich text boxes allow data entry with text formatting choices, whereas standard text boxes do not.

U

Back to top

Unicode

A character encoding standard developed by the Unicode Consortium. By using more than one byte to represent each character, Unicode enables almost all of the written languages in the world to be represented by using a single character set.

Uniform Resource Identifier (URI)

A character string that identifies a resource on the Internet by type and location.

Uniform Resource Locator (URL)

A unique address that identifies the location of a page, file, or other type of resource on the Internet or an intranet. It typically includes four elements: the protocol for accessing the resource, such as http://; the server type; the server name, which is often the name of the organization that maintains the resource; and a suffix, which typically identifies the type of organization that maintains the resource.

Uniform Resource Name (URN)

A scheme for uniquely identifying resources that may be available on

the Internet by name, without regard to where they are located.

URI

See Uniform Resource Identifier

URL

See Uniform Resource Locator

URN

See Uniform Resource Name

user role

A predefined category that can be assigned to form users based on job title, network domain, or another criterion. Roles are typically used to present customized versions of a form to different types of users.

V

Back to top

valid

Error free or acceptable according to an established set of rules or standards. For example, data entered into a form is valid if it meets the form's data validation rules.

valid XML

A well-formed Extensible Markup Language (XML) document that conforms to a specific set of constraints, usually defined in an XML Schema.

VBScript

See Visual Basic Scripting Edition

view

A form-specific display setting that can be saved with a form template and applied to form data when the form is being filled out. Users can switch between views to choose the amount of data shown in the form.

Visual Basic Scripting Edition (VBScript)

An interpreted, object-based scripting language that is a subset of the Microsoft Visual Basic programming language.

W

Back to top

Web browser

A program that interprets Hypertext Markup Language (HTML) files, formats them into Web pages, and displays them. You can use a Web browser to go from one Web page to another by following hyperlinks, to download files from the Internet to your computer, and to play audio or video files that are embedded in a Web page.

well-formed XML

An Extensible Markup Language (XML) document in which there is only one root element, and all of the elements nest properly within each other and are syntactically correct.

X

Back to top

XHTML

See Extensible Hypertext Markup Language

XML

See Extensible Markup Language

XML attribute

An Extensible Markup Language (XML) structural construct. A namevalue pair, separated by an equal sign and included in a tagged element, that modifies features of an element. All attribute values are text strings and must be enclosed in quotation marks.

XML data

Data that is saved in an Extensible Markup Language (XML) format. Microsoft Office InfoPath 2003 forms are saved as XML data files.

XML element

An Extensible Markup Language (XML) structure that consists of a start tag, an end tag, and the information between the tags. Elements can have attributes and can contain other elements.

XML parser

A software component used to process Extensible Markup Language (XML) documents by validating their structure and producing views or transformations of their content.

XML Path Language (XPath)

A language used to address parts of an Extensible Markup Language (XML) document. XPath also provides basic facilities for manipulation of strings, numbers, and Boolean values.

XML Schema

A formal specification, written in Extensible Markup Language (XML), that defines the structure of an XML document, including element names and data types, which elements can appear in combination, and which attributes are available for each element.

XML Signature

An Extensible Markup Language (XML)—based digital signature that can be used to help secure the data contained in an XML document. XML Signatures are a standard governed by the World Wide Web Consortium (W3C).

XML template file

An Extensible Markup Language (XML) file that contains the sample data that appears in the fields of a form before a user fills it out.

XPath

See XML Path Language

XSL

See Extensible Stylesheet Language

XSL Transformation (XSLT)

A language that is used to transform Extensible Markup Language (XML) documents into other types of documents, such as Hypertext Markup Language (HTML) or XML. It is a subset of Extensible Stylesheet Language (XSL).

XSLT

See XSL Transformation



New and improved controls

What's new home

The Microsoft Office InfoPath 2003 Service Pack includes new controls for collecting data and improves the things that you can do with existing controls.

Master/Detail control This control allows users to efficiently work with large amounts of data. For example, in an employee information form, a master control can display employee names and job titles, and a detail control can display employee locations, hire dates, emergency contact names, and other details.

File attachment control This control allows users to attach files to a form they are filling out.

ActiveX control This control type, based on Microsoft ActiveX technologies, allows form designers to insert an ActiveX control into their forms, such as a Microsoft Windows slider control or a custom control designed specifically for InfoPath.

Choice group This control lets users replace one option with another option when they are filling out a form. For example, users can replace a default shipping address with an alternate address.

Scrolling region This control allows users to scroll through text inside fields on their form. Scrolling regions are a good space-saving option when you have a lot of data to display in a form.

Repeating recursive section This control can be inserted within itself, is bound to nested reference fields, and repeats. This is useful when you want to create hierarchical content, such as an organization chart or outline.

Vertical label This control allows you to add text that appears at a 90-degree angle in the form, such as a vertical column heading.

Control enhancements In addition to providing new types of controls, the service pack improves the way you work with existing controls. Among other things, you can now:

Edit the properties of multiple controls simultaneously.

Align the text in a control with the surrounding text.

Limit the number of characters or enable paragraph breaks in text boxes.

Use values from the form to populate list boxes. For example, the names in a list box can come directly from a list of employees in a repeating table in the same form. If an additional employee is added to the table, the list box automatically updates to reflect the addition.



Better ways to work with data

What's new home

The Microsoft Office InfoPath 2003 Service Pack adds or enhances support for the following:

Formulas and functions Calculate mathematical values, display dates and times, reference other fields in a form, and perform string operations such as combining multiple fields into one field— all without script.

Filters You can now filter data in certain controls like repeating tables. You can also use filters to create cascading list boxes, where the selection in one list box determines the subset of choices in the second list box.

Data validation Use pattern matching to define data validation requirements for controls. Data entry patterns restrict how users enter numbers and text into a control. For example, you can require that users enter a phone number with parentheses around the first three digits. In addition, you can associate multiple data validation alerts with a single field.

Conditional formatting Hide or disable a control based on the value of another control on the form. For example, a text box for a driver's license number can be read-only unless the user enters an appropriate age into another control on the form. In addition, you can now specify conditional formatting for a wider variety of controls.

Enhanced XML Schema support InfoPath now supports the following schema structures: abstract types; abstract elements and substitution groups; required wildcard (<any>); repeating or optional model groups (sequence, choice, group and all with minOccurs or maxOccurs other than 1); and inline schemas.



Improved support for data connections

What's new home

By connecting your form to an external source of data, such as a database of employee information, you can include sophisticated functionality in your forms, including dynamic list boxes and offline data stores. The Microsoft Office InfoPath 2003 Service Pack offers the following new types of data connections and improvements to existing data connections:

Secondary data sources In addition to populating the choices for list boxes, you can now bind controls to secondary data sources. You can then use these controls to submit data to and query the secondary data source.

Web services The Web services data connection now allows you to easily connect to Web services that use ADO.NET DataSets to both receive and submit data.

Windows SharePoint Services libraries and lists The SharePoint library or list data connection allows you to receive data from a Microsoft Windows SharePoint Services form library or list in your InfoPath form. For example, the entries shown in a list box can come directly from items in a form library. In addition, you can create data connections that allow forms to be submitted to a Windows SharePoint Services form library.

Submit forms using an e-mail program Completed forms can now be submitted as an attachment in an e-mail message, with predefined recipients or recipients based on the content of the form.

Other submit options You can now submit a specific group or field in the form, just the contents of a group or field in the form, or the entire InfoPath form (including the processing instruction) to a Web service. In addition, you can submit the form's data as a string, which enables you to submit digitally signed data.



Visual Studio .NET integration

What's new home

With the Microsoft Office InfoPath 2003 Toolkit for Visual Studio .NET, which can be downloaded from the Microsoft Developer Network (MSDN) Web site, you can use Microsoft Visual Studio .NET 2003 to create, debug, and build Microsoft Office InfoPath 2003 projects that use Microsoft Visual C# or Microsoft Visual Basic .NET managed code. By using a combination of InfoPath to design forms and Visual Studio .NET to write code, you can build your project in debug or release mode. You can create new forms or add managed code to existing ones.



Layout enhancements

What's new home

The Microsoft Office InfoPath 2003 Service Pack enhances layout and formatting features, and also improves support for printing forms.

Font and paragraph formatting You can now type values into the **Font Size** box, and define spacing and indentation options for paragraphs.

Headers and footers The service pack improves the way you add and format headers and footers.

Printing When designing forms, you can define additional printing options for your users, including number of copies, collation, print range, printer name, margins, paper size, paper source, and page breaks. You can also display a page width guide that allows you to see whether the contents of a form will fit on a printed page.

Additional formatting tools InfoPath now includes the Format Painter tool, which lets you quickly copy formatting from text and other controls. You can also use the **Symbol** dialog box to insert symbols or the table drawing tool to draw a complex table— for example, one that contains cells of different heights or a varying number of columns per row.



Better integration with existing business processes

What's new home

The act of filling out a form is often one step in a larger business process that involves other tools, processes, or people. The Microsoft Office InfoPath 2003 Service Pack improves support for forms that are part of a business process by including or improving the following features:

Digital signatures Users can now sign specific sets of data in a form, insert and see a digital signature in a form, and view a form as it looked when it was signed. In addition, users can add comments to their digital signature.

User roles When designing a form, you can create categories, called user roles, and then assign users to those roles. This is helpful when you want to change the view of a form or the behavior of certain controls based on a particular user role. For example, in an expense report form, managers and employees might see different versions of the same form.

Human Workflow Services Human Workflow Services (HWS) is an information worker service that comes with Microsoft BizTalk Server 2004. HWS lets companies create business processes that users can interact with in programs like InfoPath. When filling out a form in InfoPath, the Workflow task pane lets users activate an HWS action and respond to HWS tasks. For example, a user can assign a form to his or her manager for review by clicking a button on the Workflow task pane.



More new features

What's new home

The Microsoft Office InfoPath 2003 Service Pack includes support for the following features:

Rules You can specify conditions and actions, called rules, that automatically perform tasks based on events and values in the form. You can use rules to replace tasks you would otherwise accomplish by writing script. For example, you can use rules to automatically open or close a form, fill in a value in a field, switch views, show a dialog box, or query or submit to a data connection.

Tablet PC support If you use a tablet pen to fill out forms you can write directly in fields on a form, and then convert that ink to text. Depending on the design of your form, you can also add handwritten notes or drawings to certain fields, and then save the ink along with your form. If you design forms for Tablet PC users, you can now specify background pictures for ink picture controls.

Publishing options After designing your form, you can send it to others as an attachment in a Microsoft Office Outlook 2003 e-mail message. In addition, you can now design and publish forms in design mode at the Full Trust security level.

Form management The new **Fill Out a Form** dialog box appears whenever you open InfoPath. The options in this dialog box allow you to easily organize, locate, and update your forms. You can also open and design a form from this dialog box.

Feature access You can prevent users from gaining access to certain toolbar buttons and menu commands on their forms, including the **Save**, **Save As**, **Export to**, and **Print** options. You might do this, for example, when you want users to only be able to submit a form.

Enhanced language support You can now create and edit forms that contain text in Indic, South Asian, and right-to-left languages.