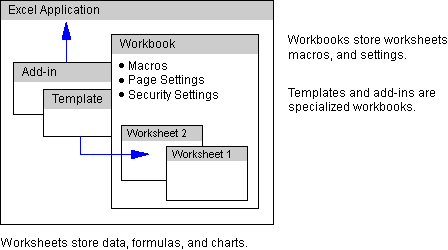
|  |  |  |
| --- | --- | --- |
| **Microsoft Excel Architecture** |  |  |

Excel documents are called workbooks. A workbook is a collection of worksheets, chart sheets, and Visual Basic for Applications modules. The worksheet is the primary location for storing data in a workbook. Each worksheet can contain cells that store data and formulas, and charts, which present data graphically. Each cell and chart item can have associated formatting, such as fonts, colours, and layout options.

You can save a workbook as a template. A template provides a pattern for creating new workbooks. You can also save workbooks as add­ins, which programmatically add custom features to Excel. The following illustration shows how Excel components fit together to create a workbook.



**Workbooks**

The workbook in Excel is analogous to the document in Microsoft Word, or the presentation in Microsoft PowerPoint. The workbook stores data in the Excel file format. Excel workbook files have the extension .xls (Windows) or the file type Excel Document (Macintosh).

The various components of Excel, such as cell data and formulas, are stored in the workbook file. Some components, such as cell data and charts, are stored on worksheets in the workbook. Other components, such as macros, book­level security settings, and page settings, are stored in the workbook file but not on worksheets.

**Cell Data**

Cell data consists of constant values stored in cells and can be either of the following:

* Numeric values, including date, time, currency, percentage, or scientific notation
* Text

The way Excel displays numeric values in a cell depends on the number format assigned to a cell. The numeric value displayed may differ from the actual value Excel stores, which is with 15 digits of accuracy. By default, Excel makes calculations based on the stored value. Such calculation is known as *full precision* calculation. However, you can have Excel calculate based on displayed values. To do this type of calculation, click **Options** (Windows) or **Preferences** (Macintosh) on the **Tools** menu, click the **Calculations** tab, and then select **Precision as displayed**.

|  |
| --- |
| **My calculations appear to be inaccurate**  If the results you get from a formula appear to be wrong, it may be due to the difference in precision between displayed and stored values. For example, if two cells each contain the value 1.007, and a formula adds them in a third cell, the result is 2.014. If all three cells are formatted to display two decimal places, Excel rounds the values. Using full precision calculation on this formula, the displayed calculation, 1.01+1.01=2.01, appears to be wrong. On the other hand, a calculation on the displayed values, 1.01+1.01=2.02, appears to be correct but results in a value that is not as precise as a full precision calculation would be. |

Calculating with precision as displayed does the following:

* Affects all worksheets in the active workbook.
* Does not affect numbers in the General format, which are always calculated with full precision.
* Slows calculation because Excel must round the numbers as it calculates.

**Cell Formulas**

Formulas use cell references when performing calculations on your data, and are part of the data that is stored in the workbook. Cell references in a formula can be relative, absolute, or mixed references in any of the following reference styles:

* A1 style
* Row­and­column (R1C1) style
* Name references

Both A1 and row­and­column reference styles refer to data by position. Using these styles, you may experience difficulty with formulas if you reposition or delete cells. One way to avoid this problem is to reference cells by name.

**Name References**

You can use a name as a reference to a cell, a group of cells, a value, or a formula. Name references can be accessible to an entire workbook or restricted to a worksheet. When a name reference is restricted to a worksheet, it can be repeated on more than one sheet so that it defines related cells on different sheets in the same workbook. A book­level name reference, on the other hand, cannot be repeated on more than one worksheet. Instead, it can be used throughout the workbook to refer to cells on one worksheet. Using book­level names eliminates the need to recreate names for each new worksheet or to type worksheet references in formulas. Sheet­level names override book­level names when used on the sheet where they are defined.

To use book­level name references, you enter the name you want to use in the name box on the formula bar. To use sheet­level name references, however, you must include the name of the sheet when you enter the name, such as Sheet1!Profit.

As an alternative to using name references in formulas, you can often use spreadsheet labels (such as category names you have added to a worksheet) instead. For example, the label of the value at the intersection of a column labelled January and a row labelled Unit Sales is January Unit Sales

**Scenarios**

The **Scenarios** command (**Tools** menu) is a tool for creating specialized formulas which pose what­if questions with your data. Scenarios can be sheet­level or book­level.

** To create a scenario**

1. On the **Tools** menu, click **Scenarios**, and then click **Add**.
2. In the **Scenario name** box, type a name for the scenario.
3. In the **Changing Cells** box, enter the references for the cells where you want to store hypothetical data.
4. Under **Protection**, select the options you want, and then click **OK**.
5. In the **Scenario Values** dialog box, type the values you want in the changing cells.

For the protection options to take effect, you must activate protection for the current sheet. For information about sheet­level security, see "[Security Settings](http://informationworker.ru/ork97.en/036/036.htm#securset)" later in this chapter.

You can copy scenarios from other worksheets and other workbooks to the active worksheet. This task is known as *merging* scenarios. To merge scenarios among workbooks, all the workbooks must be open.

** To merge scenarios**

1. On the **Tools** menu, click **Scenarios**, and then click **Merge**.
2. In the **Book** box, click a workbook name.
3. In the **Sheet** box, click the names of the worksheets that contain the scenarios you want to merge.

When you merge scenarios, there may be some duplicate names. Best Case and Worst Case, for example, are common scenario names. In such instances, Excel appends additional information to the duplicate scenario names, such as creation date, creator name, or an ordinal number.

**Cell Formatting and Styles**

*Styles* are collections of format settings for cells. Styles are stored separately from the cell data, which means they can be copied between cells, changed, or deleted, without affecting the data in the cell.

The following table shows the format settings stored in styles.

Styles are saved in the workbook. If you want to reuse styles in another workbook, you can do either of the following:

* Copy the styles to another workbook.
* Save the workbook as a template.

When you copy the styles to other workbooks, you insert the styles into existing workbooks. When you save the workbook as a template, however, you automatically copy the styles to new workbooks based on this template.

** To copy styles from one workbook to another**

1. Open the source and destination workbooks for the styles you want to copy.
2. On the **Format** menu, click **Style**.
3. Click **Merge**, and select the workbook from which you want to merge styles.

All styles from the source workbook are merged into the destination workbook. If styles in the destination workbook have names that match styles being merged, you are prompted to choose whether or not to overwrite existing styles in the destination workbook.

The Normal style is likely to match the source and destination workbooks. All cells in a new workbook are initially formatted with the Normal style. You can change the settings for the Normal style, but the change does not affect new workbooks unless you save the workbook that contains the new Normal style as an auto template.

|  |  |
| --- | --- |
| **This format setting** | **Determines** |
| Number | Decimal places, separator, inclusion of dollar sign, style for displaying negative numbers, and other options for formatting different kinds of numbers such as currency, dates, fractions, and so on. |
| Alignment | Horizontal and vertical alignment, text orientation, and whether text wraps in the cell. |
| Font | Font name, style, size, special effects, and color of the text in the cell. |
| Border | Placement and style of the border of the cell. |
| Pattern | Shading and color of the cell. |
| Protection | Whether data in the cell is locked or the formula is hidden. This option does not take effect until you activate protection by clicking **Protect Sheet** (**Tools** menu, **Protection** submenu). |

**User\Excel 97\Tools\_Options\General\Font**

**Charts**

You create charts based on a range of selected cells on a worksheet. To create a chart, click **Chart** (**Insert** menu), and then follow the instructions in the Chart Wizard. In the **Chart Location** panel, the chart is stored on its sheet (called a chart sheet) if you select the **As new sheet** option; if you select the **As object in** option, the chart is embedded in a worksheet. Regardless of which type of sheet a chart is stored on, Excel stores charts in the workbook file. You can copy charts into other workbooks, and other Office applications, such as Word documents and PowerPoint presentations.

Charts are linked dynamically to data on a worksheet. This means that changes to the data are updated in the chart, and changes to a data marker on the chart are reflected in the linked data cells. Text on the chart can also be linked to text in worksheet cells. This text appears as titles, data labels, legend entries, and labels for axis tick­marks. Editing text in the worksheet cells affects the text in the charts that are linked to the cells. You can edit text directly in charts, but this breaks the link to the cells on the worksheet.

**Custom Chart Types**

Instead of formatting chart items individually, you can quickly change the look of a chart using a custom chart type. (In previous versions of Excel, these were called chart autoformats.) Each custom chart type is based on one of the 14 predefined chart types and can include a chart subtype, legend, gridline options, data labels, colour settings, patterns, and layout. Excel includes several built­in custom chart types. You can also create your custom chart types.

** To create a custom chart type**

1. Select a chart that you want to save as a custom chart type.
2. On the **Chart** menu, click **Chart Type**.
3. On the **Custom Types** tab, click **User­defined**.
4. Click **Add**.
5. In the **Name** box, type a name.

If you want to add a description, type it in the **Description** box.

Excel stores galleries of built­in and user­defined chart types on separate chart sheets in workbooks with reserved file names, as shown in the following tables. Note that the user­defined chart gallery file is created only after a user­defined chart type is created.

To create a unified look, you can build a gallery of custom chart types for your workgroup. For example, you can create a series of custom chart types with a consistent layout and colour scheme, perhaps designed to be integrated into a PowerPoint presentation.

** To distribute custom chart types**

* Copy the user­defined chart gallery file Xlusrgal.xls (Windows) or Excel Chart User Gallery (Macintosh) to each user's computer in the location shown in the following table.

|  |  |
| --- | --- |
| **Operating system** | **Folder name and location** |
| Windows | Program Files\Microsoft Office\Office |
| Macintosh | System Folder: Preferences |

|  |  |
| --- | --- |
| **Windows chart type** | **Filename and location** |
| Built­in chart gallery | Program Files\Microsoft Office\Office\Xl8galry.xls |
| User­defined chart gallery | Program Files\Microsoft Office\Office\Xlusrgal.xls |
|  |  |
| **Macintosh chart type** | **File name and location** |
| Built­in chart gallery | System Folder:Preferences:Excel Chart Gallery (8) |
| User­defined chart gallery | System Folder:Preferences:Excel Chart User Gallery |
|  |  |
|  | **Tip**In Windows 95 and Windows NT Workstation 4.0, you can use a system policy to specify the path to a standard user­defined chart gallery on a network drive for all Excel users in your workgroup. In the System Policy Editor, set the following policy: |

**User\Excel 97\Miscellaneous\Chart Gallery**

**Macros**

Macros are stored in Visual Basic modules in workbooks. Unlike previous versions of Excel, Visual Basic modules are not stored on a module or macro sheet. Instead, you view Visual Basic code with the **Visual Basic Editor** command (**Tools** menu, **Macro** submenu). You can distribute macros in the following ways:

* As workbooks

You can store macros in a workbook. To make the macros available each time Excel starts, store the workbook in the startup or alternate startup folder.

* As add­ins

You can distribute the macros as a standalone, customized version of Excel by saving a workbook as an add­in. To automatically open the add­in each time Excel starts, store the add­in in the startup or alternate startup folder.

**Security Settings**

You can help protect specific sheets or entire workbooks. To configure security settings, including passwords, point to **Protection** (**Tools** menu), and then click a command. Security settings are stored in the workbook file. The only way to change them is to open the workbook, modify the security settings, then save the file.

**Page Settings**

Page settings for the layout of the workbook, such as page orientation, margins, and headers and footers, are based on the auto template. The page settings in the auto template are initially defined by Excel; however, you can modify these settings in the workbook or save them in a template. Once a workbook is created, its page settings are saved in the workbook file; they are not automatically saved back to the template. You modify page settings with the **Page Setup** command (**File** menu).

**Templates**

A *template* is a special workbook used as a pattern to create new workbooks. Templates have the file extension .xlt (Windows) or the file type Excel Document (Macintosh).

To maintain consistency among workbooks, you can create templates for a workgroup. For example, you can create a sales report workbook, save it as a template, and distribute it to a workgroup. When users in the workgroup create weekly sales reports based on the template, Excel has a consistent set of menus, macros, and toolbars, and the workbooks have a consistent format.

Settings saved in a template determine the following characteristics of new workbooks based on that template:

* Cell formats
* Custom menus, macros, and toolbars
* Number and type of sheets in a workbook
* Page formats
* Row and column styles
* Text, dates, numbers, formulas, and graphics, such as a company name and logo

When you open a template, Excel opens an untitled, unsaved copy of the template that contains all data, formatting, formulas, macros, styles, scenarios, and so forth that are contained in the template. The original template file remains unchanged.

**To create a template**



1. Create a workbook that includes the text, formats, and formulas you want to have in the template.
2. On the **File** menu, click **Save As**.
3. In the **File name** box (Windows) or **Save as** box (Macintosh), enter a file name.
4. In the **Save in** box, select the folder in which to store the template.
5. In the **Save As type** box (Windows) or **Save File as Type** box (Macintosh), select **Template**.

Storing workbook templates in the startup or alternate startup folder automatically makes the template available when you click **New** (**File** menu). However, this slows the Excel startup time. Alternatively, save templates in the Microsoft Office\Templates folder.

**Note**Although templates in the startup or alternate startup folder are automatically available for creating new workbooks, only the template with the reserved name Book.xlt (Windows) or Workbook (Macintosh) stored in the startup or alternate startup folder is an autotemplate.

|  |  |
| --- | --- |
| **Helping protect this item** | **Helps protect it in this way** |
| Sheet contents | Prevents editing or deleting cells on worksheets, as well as items in chart sheets |
| Sheet objects | Prevents moving, editing, resizing, or deleting graphic objects on worksheets and embedded charts |
| Scenarios | Prevents changing the definitions of scenarios |
|  |  |
| **Helping protect this item** | **Helps protect it in this way** |
| Structure | Prevents deleting, moving, hiding, unhiding, renaming, or adding sheets to the workbook |
| Windows | Prevents moving, resizing, hiding, unhiding, or closing windows in the workbook |
|  |  |
| **How Are Word Templates and Microsoft Excel Templates Different?**  Unlike Word, Excel does not maintain an attachment between workbooks and the templates on which they are based. In both Word and Excel, the template works like a read­only document, but in Word, when you create a document, it has at least one template attached to it, and potentially more. In Word, changing a template (redefining a style, for example) can affect all documents to which the template is attached. This situation is not true in Excel. Once a workbook is created from a template, there is no persistent link between the two, and changing one does not affect the other. |  |

**Add­ins**

You can save workbooks as add­ins. Add­ins compiled from Excel workbooks have the file extension .xla (Windows) or the file type Excel Document (Macintosh). Once created, add­ins are protected because they cannot be edited.

You can create add­ins to assemble and distribute custom features that, from the user's point of view, act as if they are built into Excel. For information about creating, maintaining, and distributing add­ins, see the *Microsoft Office 97/Visual Basic Programmer's Guide*, published by Microsoft Press and available wherever computer books are sold.

Several add­ins are included with Excel. Before you can use an add­in, you must install it by rerunning the Office Setup program, and then load it into Excel by clicking the **Add­ins** command (**Tools** menu). The add­in's functionality is then available to all open workbooks and remains loaded in Excel until you unload it through the **Add­ins** command (**Tools** menu). You can also load an add­in for just the current Excel session by opening the add­in through the **Open** command (**File** menu).

If you choose a Typical installation during Setup, some add­ins are installed and loaded automatically, whereas others are installed only, and must be loaded in Excel manually. For information about the components installed for each type of installation, see Appendix D.Except where otherwise indicated, all add­ins are installed in the locations shown in the following table.

The following table describes the add­ins included with Excel. Some add­ins require a dynamic-link library (DLL) or compiled C add­in in addition to the add­in file, as indicated in the table.

To use Query with Excel, you must install the Query application, the Query add­in, the drivers for the types of data you want to retrieve, and the necessary ODBC files. To install these files on a computer running Windows, rerun Setup and click **Add/Remove**; then select the **Data Access** option. To use Query on a Macintosh computer, copy the necessary files from the compact disc that came with the Value Pack for Office 98.

|  |  |
| --- | --- |
| **Operating system** | **Add­in location** |
| Windows | Program Files\Microsoft Office\Office\Library |
| Macintosh | Microsoft Office 98:Office:Excel Add-ins |
|  |  |
| **Add­in** | **Windows file name** |
| Access Links Add­in | Acclink.xla |
| Analysis ToolPak | Analys32.xll (Office\Library\ Analysis folder) |
| Analysis ToolPak ­ VBA | Atpvbaen32.xla |
| AutoSave | Autosave.xla |
| Conditional Sum Wizard | Sumif.xla |
| File Conversion Wizard | Fileconv.xls |
| Internet Assistant Wizard | Html.xla |
| Lookup Wizard | Lookup.xla |
| Microsoft Query | Xlquery.xla |
| ODBC | Xlodbc.xla Xlodbc32.dll Xlquery.xla (Office\Library\ Msquery folder) |
| Report Manager | Reports.xla |
| Solver Add­in | Solver.xla Solver32.dll Solvsamp.xls (Office\Library\ Solver folder) |
| Template Utilities | Template Numbering.xls |
| Template Wizard with Data Tracking | Wztemplt.xla |
| Update Add­in Links | Updtlink.xla |
| Web Form Wizard | Webform.xla |
|  |  |
|  |  |