

# Excel 2010: Where It Came From

## In This Chapter

- Exploring the history of spreadsheets
- Discussing Excel's evolution
- Analyzing why Excel is a good tool for developers

## A Brief History of Spreadsheets

Most people tend to take spreadsheet software for granted. In fact, it may be hard to fathom, but there really was a time when electronic spreadsheets weren't available. Back then, people relied instead on clumsy mainframes or calculators and spent hours doing what now takes minutes.

### It all started with VisiCalc

The world's first electronic spreadsheet, *VisiCalc*, was conjured up by Dan Bricklin and Bob Frankston back in 1978, when personal computers were pretty much unheard of in the office environment. VisiCalc was written for the Apple II computer, which was an interesting little machine that is something of a toy by today's standards. (But in its day, the Apple II kept me mesmerized for days at a time.) VisiCalc essentially laid the foundation for future spreadsheets, and you can still find its row-and-column-based layout and formula syntax in modern spreadsheet products. VisiCalc caught on quickly, and many forward-looking companies purchased the Apple II for the sole purpose of developing their budgets with VisiCalc. Consequently, VisiCalc is often credited for much of the Apple II's initial success.

In the meantime, another class of personal computers was evolving; these PCs ran the CP/M operating system. A company called Sorcim developed SuperCalc, which was a spreadsheet that also attracted a legion of followers.

When the IBM PC arrived on the scene in 1981, legitimizing personal computers, VisiCorp wasted no time porting VisiCalc to this new hardware environment, and Sorcim soon followed with a PC version of SuperCalc.

By current standards, both VisiCalc and SuperCalc were extremely crude. For example, text entered into a cell couldn't extend beyond the cell — a lengthy title had to be entered into multiple cells. Nevertheless, the ability to automate the budgeting tedium was enough to lure thousands of accountants from paper ledger sheets to floppy disks.



Tip

You can download a copy of the original VisiCalc from Dan Bricklin's Web site at [www.bricklin.com](http://www.bricklin.com). And yes, nearly 30 years later, this 27K program still runs on today's PCs (see Figure 1-1).



**Figure 1-1:** VisiCalc, running in a DOS window on a PC running Windows XP.

## Lotus 1-2-3

Envious of VisiCalc's success, a small group of computer freaks at a start-up company in Cambridge, Massachusetts, refined the spreadsheet concept. Headed by Mitch Kapor and Jonathan Sachs, the company designed a new product and launched the software industry's first full-fledged marketing blitz. I remember seeing a large display ad for 1-2-3 in *The Wall Street Journal*. It was the first time that I'd ever seen software advertised in a general interest publication.

Released in January 1983, Lotus Development Corporation's 1-2-3 was an instant success. Despite its \$495 price tag (which is probably close to \$1,000 in today's dollars), it quickly outsold VisiCalc, rocketing to the top of the sales charts, where it remained for many years.

### What Lotus did right

Lotus 1-2-3 improved on all the basics embodied in VisiCalc and SuperCalc and was also the first program to take advantage of the new and unique features found in the powerful 16-bit IBM PC AT. For example, 1-2-3 bypassed the slower DOS calls and wrote text directly to display memory,

giving it a snappy and responsive feel that was unusual for the time. The online help system was a breakthrough, and the ingenious “moving bar” menu style set the standard for many years.

One feature that really set 1-2-3 apart, though, was its *macro* capability — a powerful tool that enabled spreadsheet users to record their keystrokes to automate many procedures. When such a macro was “played back,” the original keystrokes were sent to the application, and it was like a super-fast typist was at the keyboard. Although a far cry from today’s macro capability, 1-2-3 macros were definitely a step in the right direction.

1-2-3 was not the first integrated package, but it was the first successful one. It combined (1) a powerful electronic spreadsheet with (2) elementary graphics and (3) some limited but handy database features. Easy as 1, 2, 3 — get it?

Lotus followed up the original 1-2-3 Release 1 with Release 1A in April 1983. This product enjoyed tremendous success and put Lotus in the enviable position of virtually owning the spreadsheet market. In September 1985, Release 1A was replaced by Release 2, which was a major upgrade that was superseded by the bug-fixed Release 2.01 the following July. Release 2 introduced *add-ins*, which are special-purpose programs that can be attached to give an application new features and extend the application’s useful life. Release 2 also had improved memory management, more functions, 8,192 rows (four times as many as its predecessor), and added support for a math coprocessor. Release 2 also included some significant enhancements to the macro language.

Not surprisingly, the success of 1-2-3 spawned many *clones* — work-alike products that usually offered a few additional features and sold at a much lower price. Among the more notable were Paperback Software’s VP Planner series and Mosaic Software’s Twin. Lotus eventually took legal action against Paperback Software for copyright infringement (for copying the “look and feel” of 1-2-3); the successful suit essentially put Paperback out of business.

In the summer of 1989, Lotus shipped DOS and OS/2 versions of the long-delayed 1-2-3 Release 3. This product literally added a dimension to the familiar row-and-column-based spreadsheet: It extended the paradigm by adding multiple spreadsheet pages. The idea wasn’t really new, however; a relatively obscure product called Boeing Calc originated the 3-D spreadsheet concept, and SuperCalc 5 and CubeCalc also incorporated it.

1-2-3 Release 3 offered features that users wanted — features that ultimately became standard fare: multilayered worksheets, the capability to work with multiple files simultaneously, file linking, improved graphics, and direct access to external database files. But it still lacked an important feature that users were begging for: a way to produce high-quality printed output.

Release 3 began life with a reduced market potential because it required an 80286-based PC and a minimum of 1MB of RAM — fairly hefty requirements in 1989. But Lotus had an ace up its corporate sleeve. Concurrent with the shipping of Release 3, the company surprised nearly everyone by announcing an upgrade of Release 2.01. (The product materialized a few months later as 1-2-3 Release 2.2.) Release 3 was *not* a replacement for Release 2, as most analysts had expected. Rather, Lotus made the brilliant move of splitting the spreadsheet market into two segments: those with high-end hardware and those with more mundane equipment.

## Too little, too late

1-2-3 Release 2.2 wasn't a panacea for spreadsheet buffs, but it was a significant improvement. The most important Release 2.2 feature was *Allways*, an add-in that gave users the ability to churn out attractive reports, complete with multiple typefaces, borders, and shading. In addition, users could view the results on-screen in a WYSIWYG (What You See Is What You Get) manner. Allways didn't, however, let users issue any worksheet commands while they viewed and formatted their work in WYSIWYG mode. Despite this rather severe limitation, many 1-2-3 users were overjoyed with this new capability because they could finally produce near-typeset-quality output.

In May 1990, Microsoft released Windows 3.0. As you probably know, Windows changed the way that people used personal computers. Apparently, the decision-makers at Lotus weren't convinced that Windows was a significant product, and the company was slow getting out of the gate with its first Windows spreadsheet, 1-2-3 for Windows, which wasn't introduced until late 1991. Worse, this product was, in short, a dud. It didn't really capitalize on the Windows environment and disappointed many users. It also disappointed at least one book author. My very first book was titled *PC World 1-2-3 For Windows Complete Handbook* (Wiley). I think it sold fewer than 1,000 copies.

Serious competition from Lotus never materialized. Consequently, Excel, which had already established itself as the premier Windows spreadsheet, became the overwhelming Windows spreadsheet market leader and has never left that position. Lotus came back with 1-2-3 Release 4 for Windows in June 1993, which was a vast improvement over the original. Release 5 for Windows appeared in mid-1994.

Also in mid-1994, Lotus unveiled 1-2-3 Release 4.0 for DOS. Many analysts (including myself) expected a product more compatible with the Windows product. But we were wrong; DOS Release 4.0 was simply an upgraded version of Release 3.4. Because of the widespread acceptance of Windows, that was the last DOS version of 1-2-3 to see the light of day.

Over the years, spreadsheets became less important to Lotus. In mid-1995, IBM purchased Lotus Development Corporation. Additional versions of 1-2-3 became available, but it seems to be a case of too little, too late. The current version is Release 9.8. Excel clearly dominates the spreadsheet market, and 1-2-3 users are an increasingly rare breed.

## Quattro Pro

The other significant player in the spreadsheet world is (or, I should say, *was*) Borland International. Borland started in spreadsheets in 1987 with a product called *Quattro*. Word has it that the internal code name was *Buddha* because the program was intended to "assume the Lotus position" in the market (that is, #1). Essentially a clone of 1-2-3, Quattro offered a few additional features and an arguably better menu system at a much lower price. Importantly, users could opt for a 1-2-3-like menu system that let them use familiar commands and also ensured compatibility with 1-2-3 macros.

In the fall of 1989, Borland began shipping Quattro Pro, which was a more powerful product that built upon the original Quattro and trumped 1-2-3 in just about every area. For example, the first Quattro Pro let you work with multiple worksheets in movable and resizable windows — although

it did *not* have a graphical user interface (GUI). More trivia: Quattro Pro was based on an obscure product called Surpass, which Borland acquired.

Released in late 1990, Quattro Pro Version 2.0 added 3-D graphs and a link to Borland's Paradox database. A mere six months later — much to the chagrin of Quattro Pro book authors — Version 3.0 appeared, featuring an optional graphical user interface and a slide show feature. In the spring of 1992, Version 4 appeared with customizable SpeedBars and an innovative analytical graphics feature. Version 5, which came out in 1994, had only one significant new feature: worksheet notebooks (that is, 3-D worksheets).

Like Lotus, Borland was slow to jump on the Windows bandwagon. When Quattro Pro for Windows finally shipped in the fall of 1992, however, it provided some tough competition for the other two Windows spreadsheets, Excel 4.0 and 1-2-3 Release 1.1 for Windows. Importantly, Quattro Pro for Windows had an innovative feature, known as the *UI Builder*, that let developers and advanced users easily create custom user interfaces.

Also worth noting was a lawsuit between Lotus and Borland. Lotus won the suit, forcing Borland to remove the 1-2-3 macro compatibility and 1-2-3 menu option from Quattro Pro. This ruling was eventually overturned in late 1994, however, and Quattro Pro can now include 1-2-3 compatibility features (as if anyone really cares). Both sides spent millions of dollars on this lengthy legal fight, and when the dust cleared, no real winner emerged.

Borland followed up the original Quattro Pro for Windows with Version 5. In 1994, Novell purchased WordPerfect International and Borland's entire spreadsheet business, and Version 6 was released.

In 1996, WordPerfect and Quattro Pro were both purchased by Corel Corporation. As I write, the current version of Quattro Pro is Version 14, which is part of WordPerfect Office X4.

There was a time when Quattro Pro seemed the ultimate solution for spreadsheet developers. But then Excel 5 arrived.

## Microsoft Excel

And now on to the good stuff.

Most people don't realize that Microsoft's experience with spreadsheets extends back to the early '80s. Over the years, Microsoft's spreadsheet offerings have come a long way, from the barely adequate MultiPlan to the powerful Excel 2010.

### It started with MultiPlan

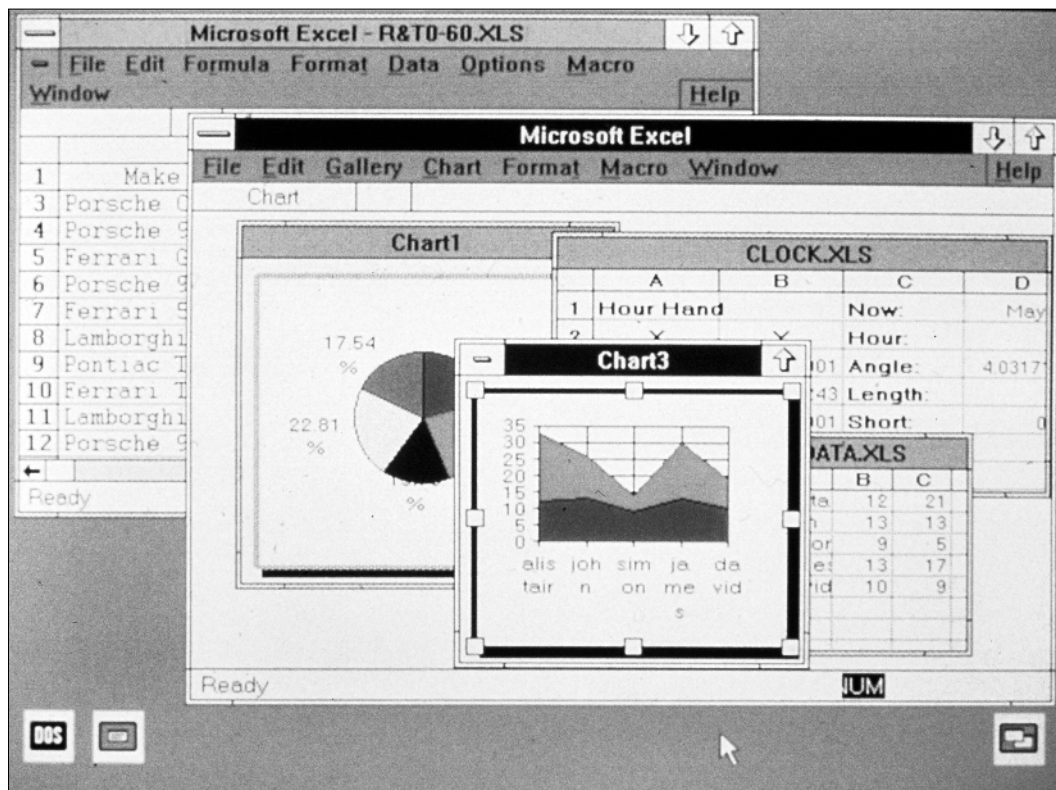
In 1982, Microsoft released its first spreadsheet, *MultiPlan*. Designed for computers running the CP/M operating system, the product was subsequently ported to several other platforms, including Apple II, Apple III, XENIX, and MS-DOS.

MultiPlan essentially ignored existing software user-interface standards. Difficult to learn and use, it never earned much of a following in the United States. Not surprisingly, Lotus 1-2-3 pretty much left MultiPlan in the dust.

## Excel arrives

Excel sort of evolved from MultiPlan, first surfacing in 1985 on the Macintosh. Like all Mac applications, Excel was a graphics-based program (unlike the character-based MultiPlan). In November 1987, Microsoft released the first version of Excel for Windows (labeled Excel 2.0 to correspond with the Macintosh version). Because Windows wasn't in widespread use at the time, this version included a runtime version of Windows — a special version that had just enough features to run Excel and nothing else. Less than a year later, Microsoft released Excel Version 2.1. In July 1990, Microsoft released a minor upgrade (2.1d) that was compatible with Windows 3.0. Although these 2.x versions were quite rudimentary by current standards (see Figure 1-2) and didn't have the attractive, sculpted look of later versions, they attracted a small but loyal group of supporters and provided an excellent foundation for future development.

Excel's first macro language also appeared in Version 2. The XLM macro language consisted of functions that were evaluated in sequence. It was quite powerful, but very difficult to learn and use. The XLM macro language was replaced by Visual Basic for Applications (VBA), which is the topic of this book. However, Excel 2010 still supports XLM macros.

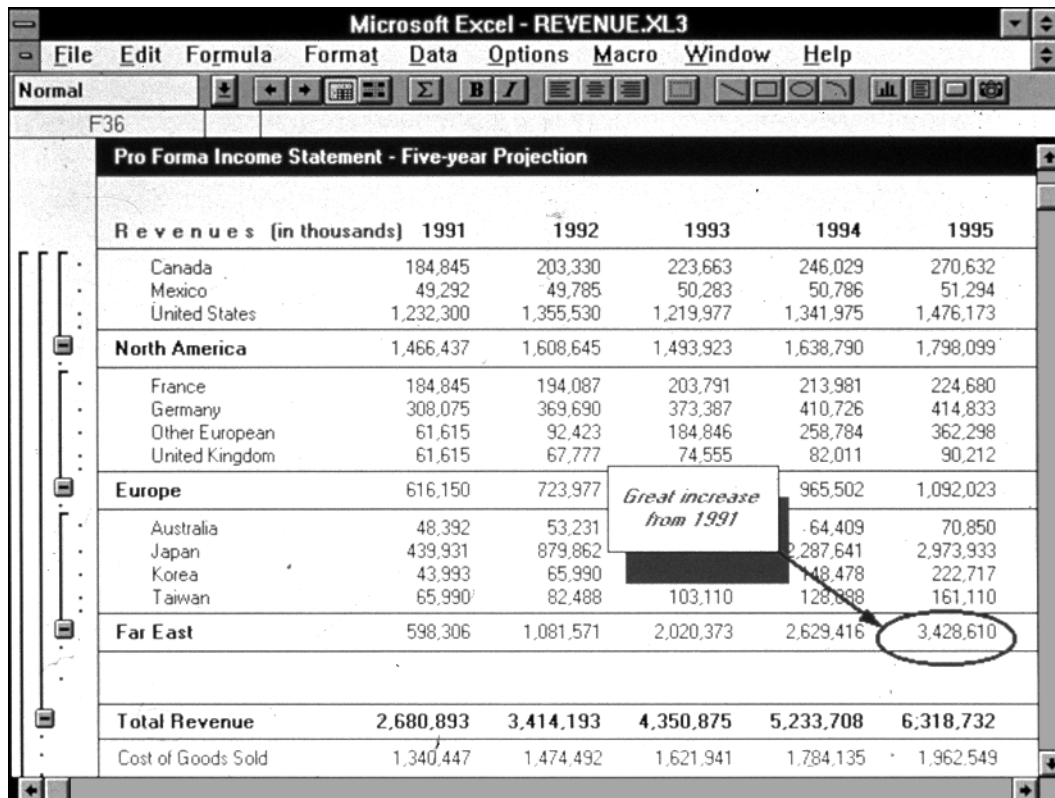


**Figure 1-2:** The original Excel 2.1 for Windows. This product has come a long way.

(Photo courtesy of Microsoft)

Meanwhile, Microsoft developed a version of Excel (numbered 2.20) for OS/2 Presentation Manager, released in September 1989 and upgraded to Version 2.21 about 10 months later. OS/2 never quite caught on, despite continued efforts by IBM.

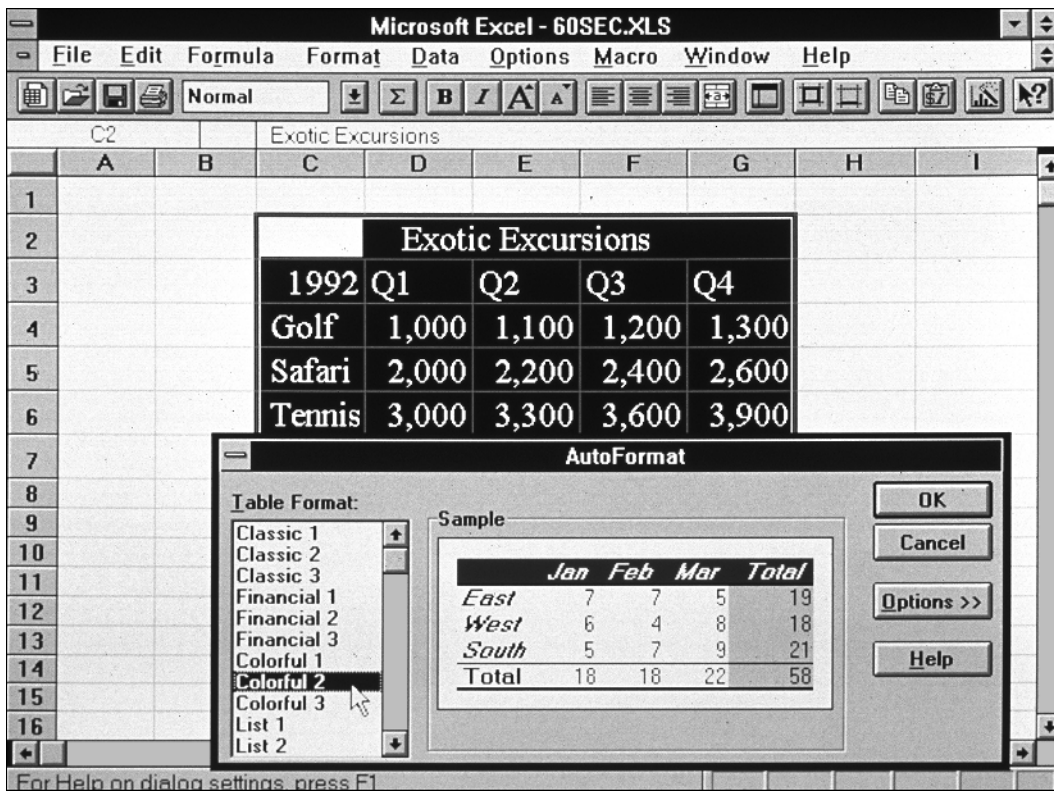
In December 1990, Microsoft released Excel 3 for Windows, which boasted a significant improvement in both appearance and features (see Figure 1-3). The upgrade included a toolbar, drawing capabilities, a powerful optimization feature (Solver), add-in support, Object Linking and Embedding (OLE) support, 3-D charts, macro buttons, simplified file consolidation, workgroup editing, and the ability to wrap text in a cell. Excel 3 also had the capability to work with external databases (via the Q+E program). The OS/2 version upgrade appeared five months later.



**Figure 1-3:** Excel 3 was a vast improvement over the original release.

(Photo courtesy of Microsoft)

Version 4, released in the spring of 1992, not only was easier to use but also had more power and sophistication for advanced users (see Figure 1-4). Excel 4 took top honors in virtually every spreadsheet product comparison published in the trade magazines. In the meantime, the relationship between Microsoft and IBM became increasingly strained, and Microsoft stopped making versions of Excel for OS/2.



**Figure 1-4:** Excel 4 was another significant step forward, although still far from Excel 5.

(Photo courtesy of Microsoft)

## VBA is born

Excel 5 hit the streets in early 1994 and immediately earned rave reviews. Like its predecessor, it finished at the top of every spreadsheet comparison published in the leading trade magazines. Despite stiff competition from 1-2-3 Release 5 for Windows and Quattro Pro for Windows 5 — both were fine products that could handle just about any spreadsheet task thrown their way — Excel 5 continued to rule the roost. This version, by the way, was the first to feature VBA.

Excel 95 (also known as Excel 7) was released concurrently with Microsoft Windows 95. (Microsoft skipped over Version 6 to make the version numbers consistent across its Office products.) On the surface, Excel 95 didn't appear to be much different from Excel 5. Much of the core code was rewritten, however, and speed improvements were apparent in many areas. Importantly, Excel 95 used the same file format as Excel 5, which is the first time that an Excel upgrade didn't use a new file format. This compatibility wasn't perfect, however, because Excel 95 included a few enhancements in the VBA language. Consequently, it was possible to develop an application using Excel 95 that would load but not run properly in Excel 5.



In early 1997, Microsoft released Office 97, which included Excel 97. Excel 97 is also known as Excel 8. This version included dozens of general enhancements plus a completely new interface for developing VBA-based applications. In addition, the product offered a new way of developing custom dialog boxes (called UserForms rather than dialog sheets). Microsoft tried to make Excel 97 compatible with previous versions, but the compatibility was far from perfect. Many applications that were developed using Excel 5 or Excel 95 required some tweaking before they would work with Excel 97 or later versions.



Cross-Ref

**I discuss compatibility issues in Chapter 26.**

Excel 2000 was released in early 1999 and was also sold as part of Office 2000. The enhancements in Excel 2000 dealt primarily with Internet capabilities, although a few significant changes were apparent in the area of programming.

Excel 2002 (sometimes known as Excel XP) hit the market in mid-2001. Like its predecessor, it didn't offer many significant new features. Rather, it incorporated a number of minor new features and several refinements of existing features. Perhaps the most compelling new feature was the ability to repair damaged files and save your work when Excel crashed.

Excel 2003 (released in fall 2003) was perhaps the most disappointing upgrade ever. This version had very few new features. Microsoft touted the ability to import and export eXtensible Markup Language (XML) files and map the data to specific cells in a worksheet — but very few users actually needed such a feature. In addition, Microsoft introduced some “rights management” features that let you place restrictions on various parts of a workbook (for example, allow only certain users to view a particular worksheet). In addition, Excel 2003 had a new Help system (which put the Help contents in the task pane) and a new “research” feature that lets you look up a variety of information in the task pane. (Some of these required a fee-based account.)



Note

**For some reason, Microsoft chose to offer two sub-versions of Excel 2003. The XML and rights management features are available only in the stand-alone version of Excel and in the version of Excel that's included with the Professional version of Office 2003.**

**Because of this, Excel developers may now need to deal with compatibility issues within a particular version!**

## A new user interface

Excel 2007 (Version 12) became available in late 2006 and was part of the Microsoft 2007 Office System. In terms of user interface, this upgrade was clearly the most significant ever. A new Ribbon UI replaced menus and toolbars. In addition, the Excel 2007 grid size is 1,000 times larger than in previous versions, and the product uses a new open XML file format. Other improvements include improved tables, conditional formatting enhancements, major cosmetic enhancements for charts, and document themes.

Reaction to the new UI was mixed. Some users loved it, others hated it. Several companies even created add-ins that allowed Excel 2007 users to revert to the old menu system. Clearly, Excel 2007 is easier for beginners, but long-time users may spend a lot of time wondering where to find their old commands.

The current version, Excel 2010, is part of Microsoft 2010 Office System. Apparently, the decision-makers at Microsoft are a bit superstitious. They skipped Version 13, and went straight to Version 14.

Excel 2010 features enhancements in pivot tables, conditional formatting, and image editing. The product now supports in-cell charts called *sparklines* and the ability to preview pasting before committing to it. A new *backstage* feature is devoted to document-related tasks, such as saving and printing. In addition, end users can now customize the Ribbon. And finally, dozens of new worksheet functions are available — mostly highly specialized functions that replace old functions that had some accuracy problems.

## Current Competition

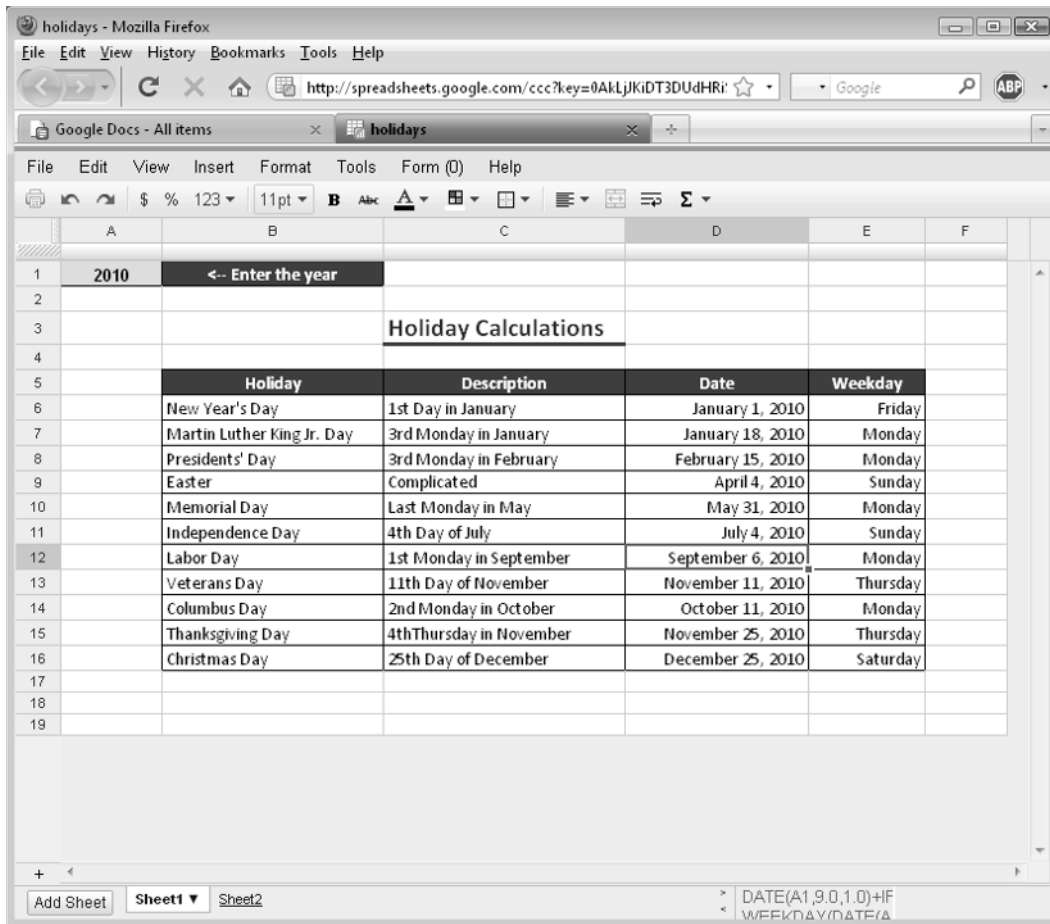
So there you have it: More than three decades of spreadsheet history condensed into a few pages. It has been an interesting ride, and I've been fortunate enough to have been involved with spreadsheets the entire time.

Things have changed. Microsoft not only dominates the spreadsheet market, it virtually owns it. What little competition exists is primarily in the form of free open-source products, such as OpenOffice and StarOffice. Increasingly, you hear about Web-based spreadsheets, such as Google Spreadsheets (see Figure 1-5). Microsoft has responded, and now has its own Web-based version of Excel and other Office 2010 applications.

In the final analysis, Microsoft's biggest competitor is probably itself. Users tend to settle on a particular version of Excel, and if things are working well, they have very little motivation to upgrade. Convincing users to upgrade to a new version that provides only a few advantages is one of Microsoft's biggest challenges.

## Why Excel Is Great for Developers

Excel is a highly programmable product, and it's easily the best choice for developing spreadsheet-based applications.



**Figure 1-5:** A Web-based spreadsheet from Google.

For developers, Excel's key features include the following:

- **File structure:** The multisheet orientation makes it easy to organize an application's elements and store them in a single file. For example, a single workbook file can hold any number of worksheets and chart sheets. UserForms and VBA modules are stored with a workbook but are invisible to the end user.
- **Visual Basic for Applications:** This macro language lets you create structured programs directly in Excel. This book focuses on using VBA, which, as you'll discover, is extremely powerful and relatively easy to learn.

- **Easy access to controls:** Excel makes it very easy to add controls, such as buttons, list boxes, and option buttons, to a worksheet. Implementing these controls often requires little or no macro programming.
- **Custom dialog boxes:** You can easily create professional-looking dialog boxes by creating UserForms.
- **Custom worksheet functions:** With VBA, you can create custom worksheet functions to simplify formulas and calculations.
- **Customizable user interface:** Developers have lots of control over the user interface. In previous versions, changing the interface involved creating custom menus and toolbars. Beginning with Excel 2007, it involves modifying the Ribbon. Changing the Ribbon interface is not as easy as it was in previous versions, but you can still do it.
- **Customizable shortcut menus:** Using VBA, you can customize the right-click, context-sensitive shortcut menus.
- **Powerful data analysis options:** Excel's PivotTable feature makes it easy to summarize large amounts of data with very little effort. The data can reside in a worksheet or in an external database.
- **Microsoft Query:** You can access important data directly from the spreadsheet environment. Data sources include standard database file formats, text files, and Web pages.
- **Extensive protection options:** Your applications can be kept confidential and protected from changes by casual users.
- **Ability to create add-ins:** With a single command, you can create add-in files that bring new features to Excel.
- **Support for automation:** With VBA, you can control other applications that support automation. For example, your VBA macro can generate a report in Microsoft Word.
- **Ability to create Web pages:** You can easily create a HyperText Markup Language (HTML) document from an Excel workbook. The HTML is very bloated, but it's readable by Web browsers.

## Excel's Role in Microsoft's Strategy

Currently, most copies of Excel are sold as part of Microsoft Office — a suite of products that includes a variety of other programs. (The exact programs that you get depend on which version of Office you buy.) Obviously, it helps if the programs can communicate well with each other. Microsoft is at the forefront of this trend. All the Office products have extremely similar user interfaces, and all support VBA.

Therefore, after you hone your VBA skills in Excel, you'll be able to put them to good use in other applications — you just need to learn the object model for the other applications.