

Leading Edge

Smarter Documents

Microsoft Moves to the Next Level in Creating Context Sensitive Documents

By Peter O'Kelly

Microsoft Office documents haven't gotten a lot smarter during the last five to ten years, at least as far as most people are concerned. Although millions of people are now Internet-savvy, and are comfortable browsing Web pages, most people using productivity applications such as Microsoft Word, Excel, and PowerPoint probably wouldn't be able to list the most significant new features introduced between Office 95 and Office 2003.

This Office version apathy exists in no small part because a very large percentage of Microsoft customers are still using old versions of Office (Office 97 and Office 2000 in particular) precisely because they haven't perceived enough incremental value in later Office releases to warrant an upgrade. Their versions of Office seem to be good enough for their needs.

Good Enough?

On the competitive front, although Corel WordPerfect Office and IBM Lotus SmartSuite no longer challenge Office, the OpenOffice.org open source offering (<http://www.openoffice.org>) has been evaluated by many organizations seeking a lower-cost alternative for good-enough word processing, spreadsheet, and presentation applications. Sun's StarOffice application suite, based on the OpenOffice.org code base (which Sun contributed to openoffice.org), is a low-cost suite that adds a database application and traditional software product support. OpenOffice.org and StarOffice are doubly challenging to Microsoft, being low or no cost alternatives, and also supporting client platforms other than Windows (Linux, Mac OS, and Solaris, although Mac client support is limited to OpenOffice.org, which isn't a fully native Mac application).

Microsoft hopes, with Office 2003 (officially known as Microsoft Office 2003 Editions), to deliver a compelling value proposition for end users and application developers that will challenge the good-enough conventional wisdom and ultimately transform Office from a bundle of largely stand-alone applications into a tightly integrated information worker system in which users focus on their work contexts without being distracted by application boundaries. From a developer perspective, the Office System is now also an application development and deployment environment — both stand-alone and in conjunction with Visual Studio.NET.

Beyond the Competition

There are some Office 2003 scenarios that neither earlier versions of Office nor any of the good-enough candidate alternatives can handle readily. One of these is text patterns such as person names and dates within documents that are automatically recognized, and potentially useful actions, such as sending an email or instant message to a person or scheduling an appointment, that are unobtrusively presented in the user's editing context.

Within XML Schema-structured documents in the new Office, contextually appropriate actions and information, ranging from field-level help to the display of pertinent information retrieved from enterprise applications, may be presented when user focus enters a given document element. For instance, it might show the most recent orders from a given customer when the customer name is entered.

Also, at any point while working with a document, a user can request related research information from references such as dictionaries, Internet searches, enterprise applications, or any Web service-accessible resource. Actions for inserting the research results into the document context also can be provided, reducing multi-window cut/paste operations and simplifying the use of formatted and XML-tagged content.

The features behind these Office 2003 scenarios include smart tag technology, smart documents, and the Research Library, and all are new (or significantly enhanced) in Office 2003. All are also open to use and customization by application developers, producing a significant shift in the content/application boundary and making Office 2003 an attractive choice for applications that might have previously been done with more effort and deployment complexity by developers using Visual Basic, FrontPage, or other application development tools.

Smarter Than Browsers

Historically, the distinction between content and application was relatively straightforward; users created documents and application developers created applications for users. The Web irrevocably changed the game, bringing selectively active content, such as pages containing controls, links, and some local processing with ECMAScript, to mainstream users.

Appreciation has grown during the last few years for the importance of distinguishing between structure, presentation, behavior, and content when developing content-centric applications through standards initiatives such as XHTML and the myriad XML standards activities.

For example, XForms (<http://www.w3.org/TR/xforms>) is a part of the XHTML 2.0 standards activity that aims to simplify the creation of Web-oriented applications that cleanly separate structure, presentation, behavior, and content in a way that's also conducive to supporting multiple device types such as PCs and handhelds. These aims include taking advantage of devices' local processing and storage resources to minimize server round-trips, maximize application responsiveness, and facilitate off-line/disconnected applications. This is not a particularly cheery prospect for Microsoft, as it would make browser clients much more directly competitive with Office in many ways.

As such, Office 2003 is a strategic milestone for Microsoft. The company has to neutralize free/low-cost good-enough Office competitors and strongly differentiate Office from browser clients for information worker applications. Smart tag technology, smart documents, and the Research Library all have pivotal roles in the strategy.

Tagging Technology

First included with Office XP in 2001 and met with a great deal of controversy at its introduction, smart tag technology is a powerful action-in-context service layer. From a user perspective, smart tags are a natural extension of the built-in IntelliSense spelling and grammar checking tools in Office applications, but they're much more open-ended.

The dotted purple line under "Peter O'Kelly" in FIGURE 1 indicates that a smart tag recognizer has recognized a smart tag instance; in this case "Peter O'Kelly" has been recognized as a person name. Once recognized, the smart tag icon (the boxed and circled "i") above "Peter" appears when the mouse is positioned over the smart tag instance. Clicking on the smart tag icon presents an action menu, as seen on the right-hand side of FIGURE 1.

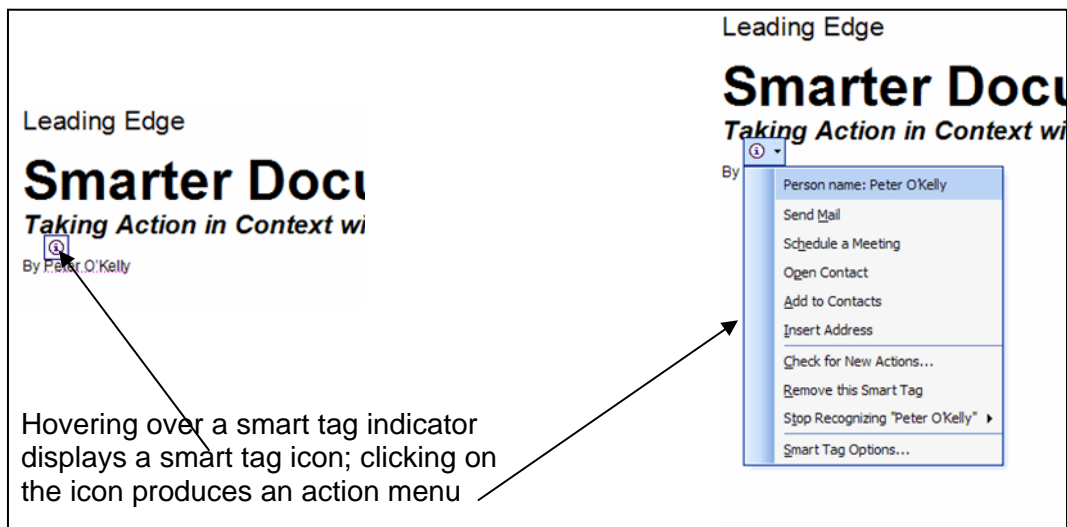


FIGURE 1: A smart tag recognizer detects the “Peter O’Kelly” is a name and displays an icon when the cursor passes over it.

There’s also a Smart Tags dialog that provides some user options for smart tag technology in Word 2003 such as which recognizers to use and the ability to remove recognizers or add more smart tag recognizers and actions from Microsoft and third-parties.

More Than Meets the Eye

Although straightforward from a user’s perspective, the smart tag architecture is actually quite elaborate, and the smart tag software developer kit (SDK) is designed for professional developers. Smart tag recognizers and action handlers can be either COM components or XML, and most often execute on the user’s PC rather than on a server. Developers can use Primary Interop Assemblies (PIAs) to incorporate .NET managed code as well.

Additionally, recognizers and action handlers are loosely-coupled and user-selectable. I might elect to use the default Microsoft person name action handler shown in FIGURE 1, for example, while you could opt for a person name action handler that offers actions based on information in your company’s PeopleSoft implementation.

Action menus can be quite sophisticated, with cascading and dynamically-defined action lists, something new in Office 2003. Also, developers and application administrators have a variety of options available to them for deploying and maintaining smart tag recognizers and action handlers, facilitating smart tag applications that are seamless for users and also secure and automatically updated for application developers and administrators.

Smart tag technology has also been highly productive for information-oriented service providers such as LexisNexis and WestLaw, for tracking contexts such as FedEx shipping, and for application integration. Microsoft offers a suite of smart tag recognizers and actions for its Microsoft Business Solution offerings, Great Plains for example, that greatly simplify the use of business application-managed information from within Office.

Early Controversy

Smart tag technology had a controversial introduction because several popular reporters and Weblog authors were alarmed by Microsoft’s plan to include the technology in Internet Explorer (IE) as well as Word and Excel. Briefly, the IE smart tag technology naysayers asserted that Microsoft would inevitably use smart tag technology in IE to monitor user activity, something that anyone who understands the loosely-coupled smart tag architecture realizes is nonsensical.

Microsoft had planned to include smart tag technology in IE to make smart tags more useful for organizations publishing their Office documents to intranet or Internet sites, but removed the technology from the default configuration of IE in order to minimize the controversy.

Office 2003 brings many enhancements for smart tag technology, addressing performance, features, security, and deployment services. Smart tag support in Access and PowerPoint is also new in Office 2003; see the Microsoft document at <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnofftalk/html/office01022003.asp> for a detailed overview. Smart tag technology also will be available for use in IE, with Office 2003 (via a new smart tag SDK for Office 2003, to be precise), but it won't be enabled by default. Additionally, browser-based SharePoint clients will use the smart tag model so that hovering over a person's name in a SharePoint page will present smart tag-style actions in context.

Smart Documents

Office 2003 includes a new Document Actions task pane that expands on the smart tag model. The new smart document features enable developers to associate actions with XML Schema elements within a document, rather than smart tag types, and the actions appear in the Document Actions task pane (initially available only in Word, Excel, and Outlook if Word is used as the message editor).

The Word example in FIGURE 2 shows how a smart document can provide supplemental information in context. In this example, business intelligence information is displayed and made available for importing. There's also an action button for submitting the report when it's complete and a control for selecting among chart display formats. As this example suggests, the Document Actions task pane can incorporate a variety of control and component types, including COM controls and, through PIAs, .NET managed code components.

The screenshot displays a Microsoft Word document titled "financial report test.xml". The main content area features a 3D bar chart showing financial data for four quarters (Q3 2001, Q3 2002, Q4 2001, Q4 2002) across four categories: Cost of Revenues, Research and Development, Sales and Marketing, and General and Administrative. Below the chart is a table titled "Fourth Coffee Income Statement" with the following data:

	Three Months Ended March 31		Three Months Ended March 31	
	2001	2002	2001	2002
Revenue	\$6,403	\$0	\$18,719	\$0
Operating expenses				
Cost of Revenues	899	1,395	2,588	3,823
Research & Development	1,069	1,066	3,015	3,123

The Document Actions task pane on the right includes a coffee cup icon, a "Year" dropdown menu (1998-2002), "Import Financial Data" and "Submit Report" buttons, and a "Chart" section with three chart format icons. A "Remove Smart Document Information" button is located at the bottom of the task pane.

FIGURE 2: The Document Actions pane on the right-hand side of this Word 2003 smart document are associated with XML schema elements within the document.

As with smart tag technology, smart documents are straightforward for users and powerful for developers. Developers working with the smart tag/document SDK can create a wide range of applications that would have otherwise required the use of custom applications and most likely taken user focus out of context.

Windows SharePoint Services are an ideal companion for smart documents. SharePoint users create new instances of smart documents from portal-based templates; Office and SharePoint together handle the seamless and secure downloading and caching of related resources as necessary. SharePoint-managed lists also are usable within smart documents, and SharePoint provides for workflow/business process automation needs as well. It's also possible for users to directly add smart tag resources (using an ungainly-named "XML Expansion Pack" feature), but the most common scenario will be Office 2003 and SharePoint working together.

Smart tag technology and smart documents are also highly complementary. For example, a developer could use smart tag recognizers to automatically XML-tag content, with the tags in turn driving smart document Document Actions.

At Your Fingertips

Another new feature in Office 2003, the Research Library (also known as the Research Task Pane) is a facility for making research materials available to users through the Office task pane. Right-clicking on a word or phrase presents a menu with a new "Look Up..." action, which, if you select it, pops up the Research Task pane and presents the results of the look-up action. Users can select among multiple reference types, ranging from built-in dictionary and thesaurus references to Web services-accessible external references such as MSN and subscription-based offerings from companies such as Factiva and Gale that provide news searches and company profiles, respectively.

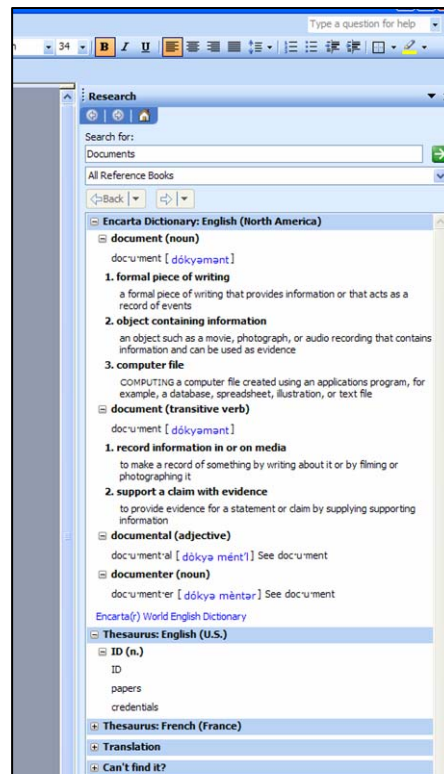
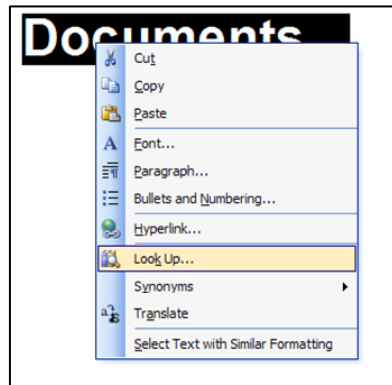


FIGURE 3: The new Research Task Pane is a potentially huge time saver by providing reference material within a document including one displayed in IE.

The Research Library is available within Office 2003 in Word, Excel, PowerPoint, Outlook, Publisher, and OneNote. Some references are available as part of the local Office application; others require an Internet connection. New research resources can be added using a Web services interface, and the result task pane can include actions, so the Research Library can be used in ways similar to smart documents, offering a different invocation model — user-initiated look-up operations instead of XML element focus or smart tag

recognition. Installing Office 2003 also makes the Research Task Pane available within IE. IE users can alt-click on a selection to bring up the Research Task Pane and initiate a search, as shown in FIGURE 3.

GuruNet (<http://www.gurunet.com>), which introduced a pioneering action-in-context offering in 1999, also uses the alt-click model. GuruNet works with all Windows applications and even works on the Windows desktop; you can alt-click on a menu item or text associated with a Windows control, for instance. The product also runs on versions of Windows prior to 2000 and XP. I've been an avid GuruNet user for several years and am glad that Office 2003 doesn't often interfere with it. IE, rather than GuruNet, will consume alt-click actions if you have the Research Task Pane open in IE, but Office 2003 and GuruNet are otherwise compatible.

The table in FIGURE 4 summarizes how the three approaches to smarter documents in Office 2003 relate.

	Invocation Model	User Interface	Administration
Smart tag technology	Automatic recognizers and user-selected (or automatic) actions	Smart tag indicators, icons, and action menus	User-selectable recognizers and action handlers
Smart documents	Automatic based on current XML element	Document Actions task pane	XML Expansion Pack manifests and associated resources; simplest to use via SharePoint
Research Library	User-initiated (right mouse menu look-up or alt-click)	Research task pane	User-selectable research references; can also be updated by administrators

FIGURE 4: Microsoft's smart technologies enable related tasks and activities to be performed within a document.

Smart Moves for Microsoft Customers

The new smarter document features in Office 2003 are powerful for end users and application developers. Together, they offer a much richer information worker experience, significantly extending the IntelliSense services (spelling and grammar checking) in Office in ways that will be exceptionally difficult for good-enough Office alternatives and browser-centric clients to match.

They provide very deep support for XML, including XML Schema and Web services. The new XML features in Office 2003 will give dedicated XML editors some serious competition, and also will shift market segmentation models for content and document management systems.

Strong synergy with other Microsoft offerings, especially SharePoint for collaborative applications and BizTalk for XML-based workflow/business process automation and enterprise application integration, makes the new Office even more compelling as a solution platform. And more effective use of PC resources for local execution and storage, makes applications more intuitive, responsive, and often fully available when off-line.

The smart technologies offer a powerful means for developers to take full advantage of information resources and enterprise applications, and to make the most effective use of information captured and managed in Office applications.

The new capabilities in Office 2003 are also smart and strategic for Microsoft. If successful, Office 2003 will challenge the conventional wisdom about good-enough productivity applications and will clearly differentiate information worker Office applications from browser-based applications. One final note: Microsoft InfoPath, a new application in Office 2003, has significant overlap with many of the smarter document features; I'll cover InfoPath in my next e-newsletter column.

Peter O'Kelly (email: pbokelly@acm.org; blog: <http://pbokelly.blogspot.com>) is an independent software industry analyst, consultant, and author with more than 20 years of experience in software product management, application development, and consulting.

Plan for Success

To fully exploit smarter documents within your organization:

Develop criteria for when to use smart tags, smart documents, and the Research Task Pane. All are currently most practical for intranet contexts in which you can be confident users will have Office 2003 and, ideally, SharePoint. Earlier releases of Office will still be able to open smarter documents created with Office 2003, but will ignore the smart document and Research Task Pane features.

Explore means of making enterprise applications and databases more broadly useful by incorporating them into Office 2003 smart document templates. Follow a service-oriented architecture and use XML Web services to make enterprise resources accessible to Office 2003 documents.

Optimize your information service subscriptions using smart tag technology, smart documents, and the Research Library features in Office 2003. LexisNexis and WestLaw are great examples to explore.

Reassess your content management and document management product requirements and assumptions — Office 2003 has potential to significantly alter these product categories.

Forget about Windows 98/ME/NT and Mac OS clients, if you want smarter documents; Office 2003 requires Windows 2000 with Service Pack 3 or Windows XP, and Microsoft hasn't yet announced its plans for smarter documents in Mac Office. Also, plan on Windows Server 2003, if you want to take full advantage of Office 2003, as doing so requires Windows SharePoint Services — which is available only as part of Windows Server 2003. The smart document security features in Office 2003 also require use of the .NET security model for trusted servers.

Expect to see many of the new smarter document capabilities find their way into Longhorn, the next major release of the Windows client; with Longhorn and “Office 12” (the code-name for the Longhorn-era version of Office) the boundaries between the Windows platform and Office will be much blurrier (this is speculation; Microsoft has not yet announced detailed Longhorn or Office 12 plans).