Excel Services Technical Overview

Microsoft Corporation

November 2006

**Applies to**: Microsoft Office SharePoint Server 2007, Microsoft Office Excel 2007

**Summary**: Learn about Excel Services and how it works with Excel 2007 to provide robust and secure shared workbook solutions. Read about Excel Services extensibility features, architecture, performance optimization, and deployment. (16 printed pages)

**Contents**

* [Introduction to Excel Services](http://msdn2.microsoft.com/en-us/library/aa972194.aspx#Office2007ExcelServicesOverview_Intro)
* [What Is Excel Services?](http://msdn2.microsoft.com/en-us/library/aa972194.aspx#Office2007ExcelServicesOverview_WhatIsExcelServices )
* [Broad and Secure Sharing of Workbooks](http://msdn2.microsoft.com/en-us/library/aa972194.aspx#Office2007ExcelServicesOverview_Sharing)
* [Managing Workbooks in Office SharePoint Server 2007](http://msdn2.microsoft.com/en-us/library/aa972194.aspx#Office2007ExcelServicesOverview_Managing)
* [Business Intelligence Reporting and Dashboards](http://msdn2.microsoft.com/en-us/library/aa972194.aspx#Office2007ExcelServicesOverview_BI)
* [Extending Excel Services](http://msdn2.microsoft.com/en-us/library/aa972194.aspx#Office2007ExcelServicesOverview_Extending)
* [Excel Services Architecture and Deployment](http://msdn2.microsoft.com/en-us/library/aa972194.aspx#Office2007ExcelServicesOverview_ArchitectureDeployment)
* [Conclusion: Security, Robustness, and Extensibility](http://msdn2.microsoft.com/en-us/library/aa972194.aspx#Office2007ExcelServicesOverview_Conclusion)
* [Additional Resources](http://msdn2.microsoft.com/en-us/library/aa972194.aspx#Office2007ExcelServicesOverview_AdditionalResources)

**Introduction to Excel Services**

Excel Services is a new server technology that is included in Microsoft Office SharePoint Server 2007. Excel Services extends the capabilities of Microsoft Office Excel 2007 by enabling the broad sharing of workbooks, improved manageability and security, and the ability to reuse workbook models through a scalable server-based calculation service and interactive Web-based user interface (UI).

Excel Services provides both a Web-based UI for browser-based access and a Web service API for programmatic access that have options for controlling what data is available to users and applications. As part of Office SharePoint Server 2007, Excel Services also takes advantage of the other capabilities of Microsoft Windows SharePoint Services, including enterprise content management features such as checking out, checking in, auditing, and versioning capabilities.

Excel is the most widely used tool for performing data analysis and reporting. As a business intelligence tool, Excel 2007 becomes more powerful by integrating with Microsoft SQL Server 2005 Analysis Services to provide a rich end-user analysis tool. Excel 2007 includes newly designed PivotTable reports, a new formula-based model for accessing Analysis Services data, improved sorting and filtering capabilities, and a completely redesigned visualization engine. Excel Services enables workbook authors to share workbooks that take advantage of this new business intelligence functionality through the browser. You can create fully interactive, data-bound workbooks that include charts, tables, and PivotTable reports as parts of a portal, dashboard, or business scorecard, without requiring any development, and that present a "single version of the truth."

The Excel Services architecture consists of a Web front end and an application-server tier. The Excel Calculation Services loads requested workbooks and performs any required calculations. Excel Web Access displays the results in HTML, and Excel Web Services provides a Web services interface to enable applications to access the workbooks.

The application server contains Excel Calculation Services, which loads workbooks, calculates them, and provides access to external data. You can scale either or both of these tiers up or out, including scaling to high-performance computing clusters. Excel Services supports configurable load balancing and several options that can improve performance.

This article examines the following:

* How Excel Services works with Excel 2007 to provide robust and more secure shared workbook solutions
* How Excel 2007 and Excel Services provide a great foundation for business intelligence solutions
* Extensibility features in Excel Services, and how developers can extend Excel Services solutions
* Architecture of Excel Services deployment, how to scale deployments and make them highly available, and performance optimizations

**What Is Excel Services?**

Excel Services enables businesses to securely share workbooks across an enterprise. You can now make workbooks available through a browser with far greater control and manageability than has previously been possible. This server-based model complements the rich Excel 2007 client, enabling the following benefits to the enterprise:

* **More secure sharing of workbooks.**   Workbook authors can share workbooks without exposing any underlying proprietary business logic, and they can prevent others from editing workbooks.
* **Business intelligence.**   Workbooks that are connected to enterprise data sources can be published to the server and used in business intelligence dashboards and scorecards. Users can interact with the workbooks in the browser to explore the data.
* **Extensibility.**   Reusable logic that is contained in workbooks and user-defined functions is available for developers to implement in server-based applications.
* **Reliability and availability.**   There are significant benefits to running calculations on servers instead of on client computers. Offloading calculations to servers frees users' computers and takes advantage of the server's additional computing power for long-running calculations. Calculating on a server provides the high reliability and availability of redundant, enterprise-class systems and dramatically reduces risks that are associated with system failure.

This server-side solution enables more secure data sharing and logic reuse across the enterprise, providing a single, centrally managed version of workbook data for a "single version of the truth." The author saves the workbook to a Windows SharePoint Services document library and grants specific users or roles access to the server-generated version of that workbook in a browser.

**User Interfaces**

Excel Services provides two primary interfaces: a Web-based UI for viewing workbooks in a browser, and a Web services interface for programmatic access. With the Web-based UI, Excel Web Access, users can have both interactive access and read-only access to workbooks in a browser by means of DHTML. No ActiveX controls are used and installation of Excel is not necessary.

The Web services interface enables external server applications to directly access workbook data, enabling them to use their business logic without needing a developer to reengineer it in code. In both cases, only the data that is specified by the workbook author is available; formulas and other business logic that underlie that data remain protected.

To better understand Excel Services, it is also necessary to describe what it is not. First, and most important, it is not just the Excel client application hosted on a server. Excel Services is a new server technology that was designed from the ground up to be scalable and robust. It is also not a workbook-creation tool. Authoring workbooks requires the client application, Excel 2007. Users can, however, interact with workbooks through Excel Web Access, subject to the restrictions defined by the workbook author, or they can interact programmatically through the Web services API. This model enables workbook authors to customize the workbook and to restrict data consumers to only those interactions that the workbook authors choose through the new browser-based access. Workbook authors no longer need to use improvised methods of sharing workbooks, such as sending them in e-mail messages. Sending workbooks in e-mail messages often results in multiple copies of a workbook, essentially creating "multiple versions of the truth," and exposing proprietary formulas and logic.

**Multiuser Scenarios**

Another important aspect of the Excel Services model is that, in a simultaneous multiuser scenario, each user has his or her own session in the server's memory. The server loads a read-only instance of the workbook in memory, and each user has a separate session representing the state. If a user applies a filter to a workbook, other users are not affected by those changes, and their interactions do not modify the original file. This behavior is identical whether the workbook is being viewed in a browser by using the Web-based UI, or being viewed programmatically through the Web services interface.

Despite the read-only behavior, workbook authors can optionally mark specific cells as parameters by using Excel 2007. This enables data consumers to perform "what-if analysis" with the values in those cells, providing interactive access to the formulas and logic in the workbook. For example, a workbook might contain three columns: "Quantity," "Unit Cost," and "Extended Cost," with "Extended Cost" calculated as the product of "Quantity" and "Unit Cost." The author could make "Quantity" a parameter controllable by the data consumer, "Unit Cost" a fixed value, and "Extended Cost" the result of a formula. This enables the data consumer to interactively enter a specific number of units to be sold and obtain the correct "Extended Cost."

**More Secure Sharing of Workbooks**

This section discusses how workbook authors can easily share their workbooks by using Excel Services, how to set permissions, how to access data sources, and how to manage workbooks that are saved to Office SharePoint Server 2007.

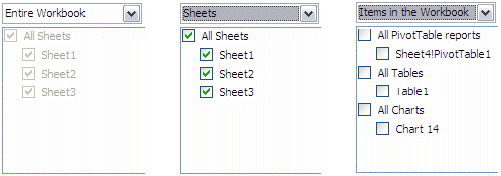
**Publishing Excel Workbooks to Office SharePoint Server**

Any workbook saved to a location that the server can access can be calculated and viewed in the browser or accessed through the Web services interface. Excel 2007 also includes a simple means by which workbook authors can use Excel Services.

The **Publish** option on the **Microsoft Office Button** has a new menu item called **Excel Services** that enables you to save the workbook to an Office SharePoint Server 2007 server (or any UNC path that the server can access). When a full copy of the workbook is copied to the server, you can use the **Excel Services Options** dialog box to specify which aspects of it can be accessed by users from the browser. You can configure it at the granularity of specific named items (named items can include sheets, named cells or ranges, tables, PivotTables, or charts).

After you name specific items, you can use the **Items in the Workbook** menu on the **Show** tab to browse through those items and choose whether each is viewable by end users, as shown in Figure 1.

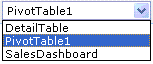
**Figure 1. Defining viewable items at the workbook, sheet, and named items levels**



The visibility of data that **Excel Services Options** governs also applies to API calls, and not just when directly viewed by users. This mechanism constitutes a true means of control over what data is visible and to whom.

Excel Web Access provides a familiar interface in which you can choose from among the viewable items, making named items readily available to use in reports or business dashboards. Because the named items remain part of a workbook that is loaded in Excel Services, they remain fully refreshable and interactive without requiring you to be able to see the underlying business logic or data source.

**Figure 2. Named items designated as viewable by a workbook author**



**Managing Permissions on Published Workbooks**

What is visible, and what is not, is determined by the publishing options and permissions that the workbook author specifies. Excel Services draws its mechanisms for authentication and access permissions from Office SharePoint Server 2007. This enables workbook authors to grant users Reader rights (viewing content), Contributor rights (viewing, changing, and adding to content), or Administrator rights (full control).

In the context of, having Reader rights allows users to download a full version of the workbook and view full details using Excel 2007. In addition to these rights, new Viewer rights enable authors to lock down workbooks for server-only viewing through a browser. Users who have only View Item rights cannot download a copy of the workbook, modify it, or view any hidden formulas or other business logic that may underlie the data values.

This ability to differentiate what is visible to workbook consumers is essential to the complementary nature of the relationship between the Excel client application and. Only an in-memory snapshot of the workbook is available to the users. It consists of the literal data values in the workbook without any of the underlying business logic. This mechanism enables administrators to make data available without revealing sensitive intellectual property. These rights are also enforced if an application using the Web services interface attempts to access data that is not marked as viewable.

In contrast, users who have Reader rights can open the workbook in the Excel 2007 and view all the underlying details associated with the data. Of course, those with write permissions can open workbooks with Excel, make changes, and save them back to a SharePoint document library.

Because workbooks can be powerful and can contain critical data, it is necessary to control where they come from, and who can access them. The mechanism for controlling who can act as an author for workbooks that are available under Excel Services is managed by a system of trusted file locations, which can be designated as SharePoint locations, UNC paths, or HTTP paths. Excel Services loads workbooks only from directory paths that the administrator designates. With SharePoint rights (or simple file-system rights for UNC paths), administrators can control who can save workbooks to these trusted locations, which ensures that only authors with appropriate access can create or modify workbooks that are presented by Excel Services.

**Connecting to External Data Sources**

Controlling where workbooks come from and who has access to them is essential to protect proprietary formulas and to present "a single version of the truth." At the same time, workbooks themselves must have a flexible and robust mechanism for connecting to data, both in workbooks, and from external data sources.

Access to external data and using existing data sources in a highly configurable and secure fashion is one key to the power of Excel Services. Workbooks can connect to live data sources by means of business-friendly and user-friendly mechanisms that enable general users to take advantage of the technology without assistance, subject to rules put in place by database administrators and other resource owners. Excel Services supports getting external data from the same data sources that the Excel client application supports, such as Microsoft SQL Server Analysis Services, Microsoft SQL Server relational databases, and other database platforms that use mechanisms like ODBC and OLEDB. Data can be returned into PivotTables reports or new formulas that can access Microsoft SQL Server Analysis Services data.

|  |
| --- |
| **NoteNote:** |
| Query tables are not supported on the server in this release. |

**Data Connection Libraries**

One powerful way of governing access to external data sources by Excel Services is by using a *Data Connection Library* (DCL) list, created by a site administrator. A DCL list contains Office Data Connection (ODC) files that persist specific data connections. Business users can take advantage of DCL by using the Excel client through a directory service operated by the server. Business users identify the connections they need that are contained within a DCL by using user-friendly names and descriptions that do not require knowledge of any technical details.

The significance of a DCL is definable in terms of three core capabilities:

* **Discovery.**   The ability to make persistent connections to external data sources visible to business users. This enables connections to external data sources to be reusable. Business users can then take advantage of them from Excel 2007 without having access to the technical details that underlie those connections.
* **Management.**   The ability to update a large number of reports at the same time. Because a DCL allows changes that are made to the external data source to automatically propagate to Excel 2007 and Excel Services workbooks, those workbooks do not need to be updated manually. This creates efficiencies and avoids potential errors and liability.
* **Security.**   Microsoft Windows SharePoint Services permissions enable access control lists, which control who is allowed to change the data source definitions and who can see what data. Excel Services can also limit access to external data through the following three settings in the **Allow** **External Data** options:
  + **None.** (default setting)   No connections to external data sources are allowed.
  + **Trusted data connection libraries only.**   Only a DCL can govern connections to external data sources.
  + **Trusted data connection libraries and embedded.**   In addition to a DCL, external data connections can be established by connection strings embedded in workbooks.

By choosing one of these settings, you can control access to data. These settings can also be configured per trusted location. This enables administrators to configure the type of connectivity that is allowed to specific types of workbooks. For example, financial data might be stored in one location that is restricted to a limited set of workbook authors, whereas other locations might have fewer restrictions.

**Authenticating External Data**

The **Properties** dialog box that is associated with workbook connections in Excel enables you to control server authentication. You can choose among the following three techniques:

* **None.**   Does not take any special action in terms of authentication, but forms a connection with the provided connection string. It assumes the user name and password are stored in the connection string, or that none is required.
* **Single sign-on (SSO).**   Implies that the credentials are stored in an SSO store and that an SSO ID is provided at connection time. An SSO solution is included with Office SharePoint Server 2007, which you can replace with custom tools or third-party tools.
* **Windows authentication.**   Requires the Kerberos protocol to authenticate between computers unless the server and external data sources are on a single computer.

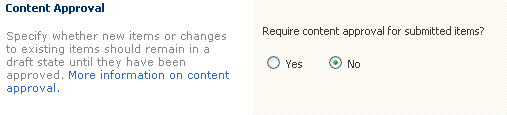
To customize the view for individual users in any of these cases, you can set the workbook to refresh the data from the server each time the workbook is opened and to fail to open if external data authentication fails. Each user is able to see only the information to which they have been granted access, to avoid inadvertently revealing inappropriate information by way of cached data.

**Managing Workbooks in Office SharePoint Server 2007**

All workbook authors must manage their workbooks regardless of how the workbooks are constructed, or what permissions are applied. Workbooks must frequently be updated, sometimes by multiple authors, and updates might also require approvals. With Excel Services, these processes are made easier with Office SharePoint Server 2007 built-in content management tools. Workbook management is improved by the robust checking in, checking out, and versioning mechanisms of Windows SharePoint Services. They allow for major and minor version numbering and security, specifically for old versions of workbooks and other documents. Additionally, Windows SharePoint Services has built-in functionality for the retention and expiration of documents, so that old versions of workbooks are automatically archived or destroyed to meet compliance requirements.

By using document approval within Windows SharePoint Services, an administrator can set up a document library so that when a workbook author saves a new version of a workbook in the library, it is not immediately available to other users to view until it is approved, as shown in Figure 3. This approval can be as simple as the administrator monitoring and changing a flag on the workbook in the document library, or it can depend on a custom workflow that sends an e-mail message to a group of approvers to ensure that the workbook meets any number of internal requirements prior to its approval.

**Figure 3. Document approval settings**



The owner of a document library can also have default settings for whether workbooks will open on the server or in the Excel client. When the document library owner wants to keep the connection information and formulas associated with data secret, he or she can prevent users from opening the workbooks within a document library, except on the server itself, to prevent private information from being viewable.

Windows SharePoint Services also enables administrators to audit document libraries. Whereas the server does not audit items within the workbooks themselves, it logs any open, create, modify, and delete events to the centralized audit trail. Windows SharePoint Services also provides several built-in reports to analyze the audit trails, and also provides tools to generate custom Excel reports.

**Business Intelligence Reporting and Dashboards**

Now that you are familiar with how to publish, set permissions on, and manage workbooks, you can learn about some of the new features in Excel 2007 and Excel Services. These features provide both workbook authors and consumers with rich controls, integrating multiple data sources and business analytics in a way that makes it easier to create, manage, and deliver business-critical information. These capabilities provide information workers with the right information at the right time to accomplish specific goals, making Excel 2007 and Excel Services a key business intelligence solution offering.

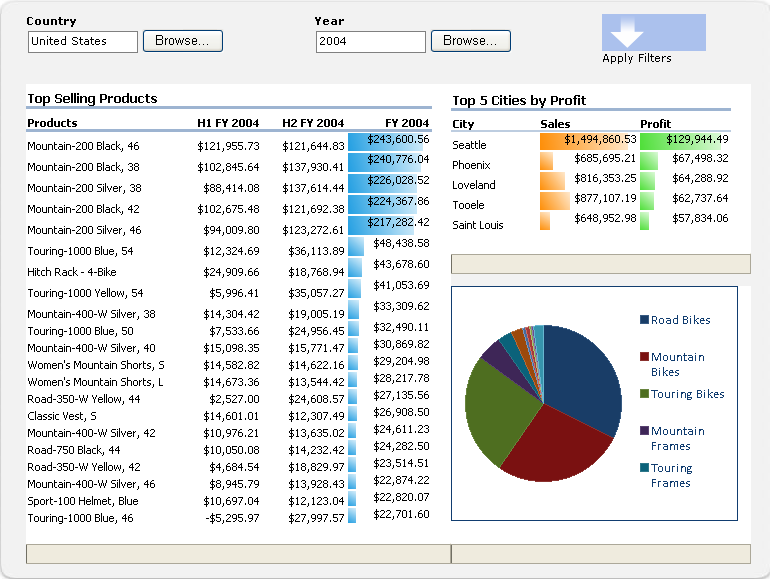
**Business Intelligence Features in Office Excel 2007**

Excel 2007 represents a substantial advance in the development of business intelligence. Of particular note are the advances in integration with Microsoft SQL Server Analysis Services. By empowering ordinary users to work more easily with cubes and to analyze data, this functionality reduces dependence on developers and database administrators and increases efficiency.

For example, users can connect to external data sources, such as Microsoft SQL Server databases, to provide real-time analysis of data. Given a set of figures in a workbook, users can easily apply visual formatting to them to illustrate how those figures relate to a specified threshold value. Color-coded data bars and heat maps can automatically provide visual cues to the user, providing instantaneous insight as to the significance of individual data points as part of a whole.

Similarly, users can apply conditional formatting such as data bars, color scales, and icon sets to provide explanatory graphical indicators that simplify analysis and decision making, as shown in Figure 4.

**Figure 4. Conditional formatting features displayed in Excel Services Web-based user interface**

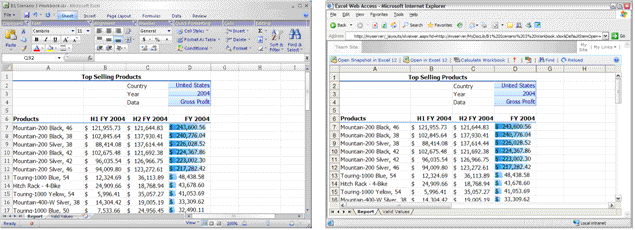


You can access these options on the menus in the new Office Ribbon UI, in the same way you would define background colors for a cell. As the data is updated from the external data source, conditional formatting changes dynamically. This functionality enables users to visualize Key Performance Indicators (KPIs) easily, and greatly enhances workbooks as a business intelligence tool.

You can incorporate the data brought in from Analysis Services (or other external data sources) in addition to workbook calculations and charts. You can also format the reports by using the rich formatting capabilities of Excel. Calculations that are performed by using Excel Services have 100 percent fidelity to those performed with Excel 2007. Visually, workbooks have the same formatting in both representations. This ensures that the need to display them in both environments does not increase the complexity of maintaining them, as shown in Figure 5. Selection and hover effects that are present in Excel 2007 are also present in the browser representation through Excel Web Access, as are sorting, filtering, outlining, drill-down, and drill-up capabilities.

Figure 5 shows two versions of a product sales workbook showing live data. The version on the left is displayed by using Excel 2007, and the version on the right is displayed by using the Excel Services Web interface.

**Figure 5. Two versions of a product sales workbook showing live data**

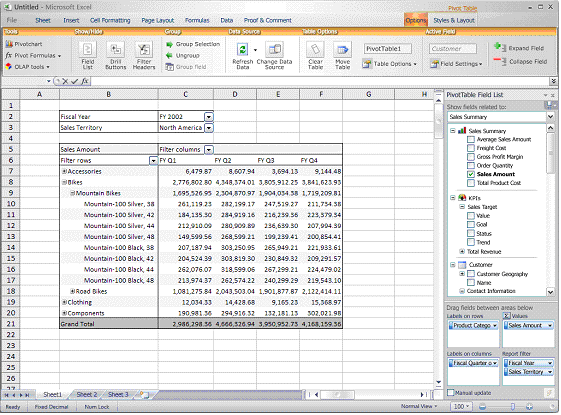


**PivotTables and Cubes**

With PivotTable reports, you can interact with Microsoft SQL Server Analysis Services data and with data from any other supported data source. Excel 2007 also supports a new class of functions that can return data directly from Microsoft SQL Server Analysis Services. Much of the functionality of Excel 2007 has been implemented in Excel Services, including the ability to expand and collapse levels, sort data in ascending or descending order, and apply filters such as the multi-select filter, top 10 filter, quick filters (for example, **Above Average** and **Below Average**), and custom filters (for example, **Less Than** and **Contains**). Workbook authors can also limit users' ability to use filters.

To create a PivotTable report, users need only to connect to a data source and select fields on and off the PivotTable by using Excel 2007. The new cube functions add to the business intelligence functionality of Excel 2007 and Excel Services: You can convert PivotTables directly into cube functions by using the new cube functions in Excel 2007. This enables general users to rapidly model data when needed, to support business decisions.

**Figure 6. Creating a PivotTable from data cubes using drag-and-drop operations**



For example, a business user could determine how to respond to changing inventory needs without needing to involve the IT organization in creating the necessary business logic. Users can write queries in individual workbook cells, pulling in data and cube functions. By using cube functions, users are not limited to pulling data from a single cube. Asymmetrical reporting enables users to pull in data from outside the cube, which greatly increases the flexibility to combine external data points and create reports.

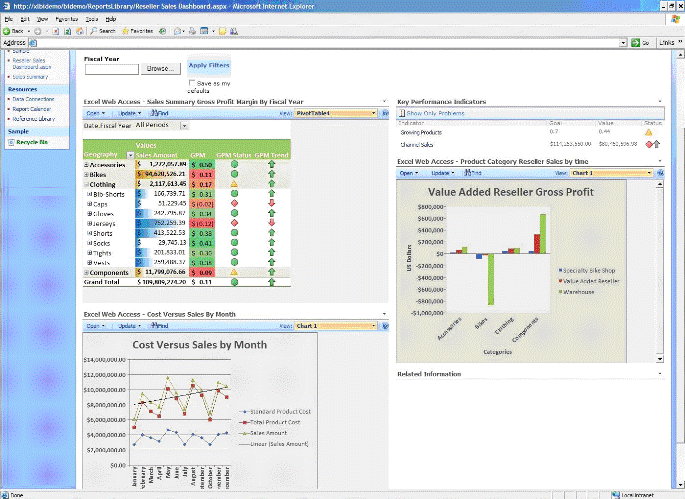
PivotTables access is available in both Excel 2007 and from Excel Services. You can use Excel 2007 and Excel Services to provide complementary functionality that you can modify according to users' needs. For example, in Excel 2007, a user has full control over the content, whereas in Excel Services, the workbook author can control the level of functionality that is available to the workbook consumer, in addition to controlling the visibility of formulas and the ability to edit data values.

**Bringing Data Together in Dashboards**

You can use the new features in Excel 2007 and Excel Services to access, manipulate, and display data in new and compelling ways. When data is ready for publication, information workers can combine their data with other data in *dashboards*. Dashboards enable visualization and reporting on large amounts of data in an easily consumable format. The ability to create business dashboards is a key business intelligence capability of Excel Services and Office SharePoint Server 2007.

Figure 7 shows a dashboard that was created by using Excel Services, KPIs, and filtered Web Parts from Office SharePoint Server 2007. It focuses on identifying trends, anticipating challenges and opportunities, and supporting business decisions.

**Figure 7. Dashboard created by using Excel Services**



You can create dashboards with just a few mouse clicks, and they require no code. Office SharePoint Server 2007 includes a set of building blocks called *Web Parts* that enable users to rapidly develop and maintain dashboards, including dynamic, interactive data capabilities such as drill-down, filter, and sort. Dashboards that are built on Office SharePoint Server 2007 take advantage of the user’s familiarity with Excel, by using PivotTables reports, Excel charts, and formulas. Dashboards also conform to the look and feel of the Office SharePoint Server 2007 environment, enabling users to automatically adhere to a corporate appearance. Dashboards can also take advantage of Office SharePoint Server 2007 Web Parts that are external to Excel, such as Key Performance Indicators, which provide highly visual representations of important business factors or of any Windows SharePoint Services or custom Web Part.

**Excel Web Access Web Part**

The Excel Web Access Web Part can be incorporated into a dashboard and customized. It provides a variety of toolbars that can be customized for individual situations:

* **Full Toolbar.**   Provides the full range of options that are available for manipulating the workbook.
* **Summary Toolbar.**   Provides a subset of the options available on the Full Toolbar. Some options, such as **Refresh**, are not available on the Summary Toolbar.
* **Navigation Only.**   Provides only the options necessary to move around in the workbook, such as switching between named objects.

Each of these toolbar options intelligently shows only the options that are relevant to the material that is displayed. Restrictions on which elements are displayed and the related functionality are based on the trusted location in which the workbook is published. For example, refreshing data might be allowed at any time, or it might be restricted because of the length and complexity of the calculations that underlie the refresh functionality.

Office SharePoint Server 2007 introduces a feature that lets users filter the Web Parts contained in a dashboard through a central interface. Filters can narrow selections based on a user-created list of options, data-defined parameters, or extracted information from a SharePoint list or profile store. Users can select information based on product name, selected dates, or other filtering needs. You can also create filters that let users choose from a predetermined set of values.

You can customize a dashboard or other report based on the identity of the person who views it. The server can capture a value, such as logon ID, and pass it to the workbook as a parameter, applying filters or other logic to display only relevant and allowed data.

**Extending Excel Services**

Excel Services was designed to be extensible and to be able to both reuse existing business models and take advantage of extension libraries within other applications. It gives developers several options for reusing business models that are contained in Excel workbooks.

Excel Services includes a Web services API that enables developers to access workbooks programmatically to set values, perform calculations, and return results. Additionally, the Excel Web Access control can be reused and embedded as a Web Part in a business intelligence dashboard or any other Office SharePoint Server 2007 application.

Historically, organizations invested heavily in building business models within workbooks and creating their own custom function libraries (for example, Excel add-ins). Excel user-defined functions are usually a collection of functions that extend the capabilities of Excel and typically include custom library functions, or the ability to connect to non-native data sources. Frequently, these represent a significant investment.

**Exploring Excel Web Services**

The Excel Web Services API provides programmatic access to workbooks calculated by the server. Using this API, developers can communicate with the Office SharePoint Server 2007 application server and take advantage of the. This enables those applications to use the logic in an Excel workbook without needing a developer to recode that logic into the application. The business expert who created the logic can maintain it in Excel, the server administrator can maintain user access and security, and the developer can use the business logic in. All three roles are independent of one another.

Key functionality supported by the methods in Excel Web Services falls into the following areas:

* **Managing sessions.**   Several methods open and close server-side calculation sessions and set properties of the server session, such as the language context of the session.
* **Setting values.**   Several methods can be used to set new values in workbook cells (or ranges), definable by cell address, named range, or integer coordinates.
* **Data and workbook processing.**   Several methods can refresh data from an external data source and recalculate the results for the entire workbook or for parts of the workbook.
* **Returning results.**   Several methods can be used to get the value of a cell, a range, or the entire workbook as a byte array. Entire workbooks can be returned as an XML or binary file, or the user can request a snapshot in which all the formulas and external data references are removed and only the values, formatting, and charts are returned.

The same security model that controls a user's ability to access Excel data by using Excel Web Access also applies to the user's ability to obtain information from an application by using the Excel Web Services. Workbook data is controlled consistently whether you use Excel Web Access or Excel Web Services.

The session ID enables the application code and the server to maintain a session state, keeping the data in cache. This capability enables the server to calculate only those data points that are necessary to support programmatic interaction, such as changing an input parameter or refreshing a data connection. This mechanism represents a potentially substantial savings in execution resources versus a session-less model where the server might need to repeatedly read an entire workbook from the disk.

**Extending Excel Services with User-Defined Functions**

You can extend the calculation capabilities in Excel by using user-defined functions (UDFs). You can write these functions in Microsoft Visual Basic for Applications (VBA) or any native code language. Excel Services supports UDFs, but the interface for developers is a managed-code interface. Existing functions must be wrapped or rewritten to work on the server. You can also write managed-code UDFs and wrap them to work on the client.

Workbook authors often use custom libraries that enhance the core calculation capabilities. Many organizations use these types of Excel extensions extensively; Excel Services was designed to support the implementation of server-side UDFs. Many workbooks use non-managed-code Excel add-ins. In this case, you can create a UDF that accesses the Excel add-in through COM interop. In this way, workbook authors can continue to use their local Excel add-in and Excel Services, having the equivalent functionality on the server.

After the UDF assembly is registered on the trusted assembly list, you can use it to take full advantage of the UDF by using formulas in workbooks that are stored in trusted locations and marked to allow the use of UDFs.

Whereas Excel Services supports UDFs, it does not support macros, VBA, or code that accesses the Excel object model.

**Excel Services Architecture and Deployment**

Excel Services consists of three core components:

* Excel Calculation Services

This component is the "heart" of Excel Services. It loads the workbooks, calculates them, refreshes external data, and maintains session state for interactivity.

* Excel Web Access

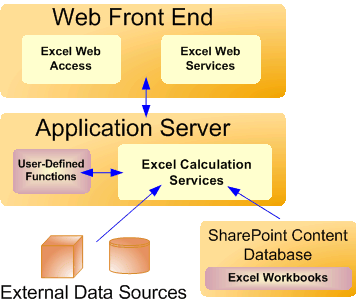
This SharePoint Web Part (a reusable component) creates HTML renderings from the results returned by Excel Calculation Services so that users can display them in a browser and interact with them. In a customization scenario, SharePoint developers can use this component along with other SharePoint Web Parts to create a wide range of Web pages.

* Excel Web Services

This component is a Web service hosted on Windows SharePoint Services that provides programmatic access to workbooks calculated by Excel Calculation Services. It enables applications to automatically update Excel workbooks or to incorporate calculations performed by Excel Services without having to re-create the underlying business logic.

These three components are divided into two groups: those associated with the Web front end and those associated with the application-server tier. The simplest configuration is for both these tiers to reside on the same physical server. To add capacity, these two tiers would more likely be located on separate computers. You can add more Web front-end or application server computers to a configuration to increase capacity and performance.

**Figure 8. Core components of Excel Web Services**



Also included in the Web front end is an Excel Calculation Services proxy that handles communication between the tiers in a multiple-server configuration. It also handles load balancing in cases where there is more than one physical application server (this component is not exposed to developers).

Office SharePoint Server 2007, and thus Excel Services, uses two Microsoft SQL Server databases at the back end: a content database and a configuration database. The content database stores all the user files, including workbooks and dashboard pages. The configuration database contains supporting information, such as access control lists and policies that govern issues such as publication details and refresh settings. The database back end can be scaled to increase performance or capacity.

**Communications Topology**

The Web front end and the application server communicate through HTTP. After a user requests a workbook, it is cached on the application server. Therefore, subsequent requests for the same workbook are typically faster than the first request. This functionality includes caching at multiple levels so that collective performance for a group of users is improved through the caching of workbooks and the external data queried by the workbooks. All this is transparent to the end user except in terms of improving response time. Cached results are shared only among users who have the same rights.

In addition to single-server topologies and multiple-server topologies, you can also support an extranet topology with Excel Services, giving external partners access to workbooks without allowing access to the underlying formula, data, and databases. This capability includes the ability to show parts of the workbook and protect the rest (by using the View Item rights discussed earlier). It enables organizations to hide intellectual property such as formulas and external data connections, while showing the results. For example, financial companies might have a financial model that they use to calculate derivatives. They can show the derivative price to their customers and partners without sharing the model that is used to generate it. However, Excel Services does not support scenarios where the Web front end is in one corporate network and the application server is in another. It does support firewalls between these components, providing a potential security advantage when scaling the Web front end and application servers out to reside on separate physical devices.

**Scaling and Availability**

Because of the logical separation of the Web front end and the application server, you can scale either or both of these tiers up or out independently of the other. You can also scale the application server out to a high-performance computing cluster, enabling automatic failover for guaranteed completion of calculations. Such a configuration allows scheduling of calculation jobs at certain times or on certain nodes. The performance increase associated with adding a new server is nearly linear.

As mentioned earlier, Excel Services provides load balancing between multiple application servers. This load balancing for the Web front end can be provided by the Network Load Balancing service available on certain versions of Windows Server 2003, or it can be provided by a separate load-balancing device. The Excel Services proxy component does the load balancing for the application server. The load balancer is designed to support the state maintained on specific application servers and also to optimize based on cached workbooks. In either case, this capability includes load balancing separate workbook calculations across multiple physical servers, and also calculating more than one workbook on an individual physical server (each request runs on a different thread). However, that platform does not support breaking apart a single workbook calculation across multiple physical servers. The design is optimized for scaling to large numbers of workbooks and requests, such as a scenario with many users viewing a dashboard with several workbooks on it, or programmatically running a large parametric sweep on a workbook or group of workbooks.

Load balancing settings can specify several methods, including round-robin, local, or hash-based. In a configuration that contains multiple application servers, round-robin load balancing simply assigns jobs requested by the Web front end to each application server in turn. The local option assumes that the Web front end and the application server are located on the same computer. Hash-based load balancing (the default setting) is optimized for using the cache on each application server. Thus a request for any specified workbook usually returns to the same back-end application server. Regardless of how Excel Services is deployed, administrators can use several performance settings to optimize Excel Services.

**Performance Optimizations**

Excel Services provides granular control over many aspects of the topology that enable administrators to optimize the performance of the system. For example, the settings can control many aspects of connectivity with each data source:

* **Allow/disallow data refresh.**   By not allowing users to perform a data refresh for large calculations, administrators can decrease the load on the server.
* **Session timeout.**   By setting a period of time after which a session times out when a user does not interact with a workbook, administrators can conserve network resources.
* **Query timeout.**   This setting is similar to **Session timeout**; it triggers a time-out if a query is not successfully completed in a specific time.
* **Maximum workbook size.**   This setting can specify a size limit in megabytes (MB).
* **Maximum request duration.**   This setting controls how long a request can last for data or for a workbook.

Again, these settings provide for granular control over the behavior of Excel Services, but administrators can also accept the default settings, which provide performance that meets or exceeds business needs in many cases. In the event that usage grows faster than originally anticipated, Excel Services can be scaled out to provide better performance and high availability.

**Conclusion: Security, Robustness, and Extensibility**

Excel Services enables end users to create robust solutions without involving developers, and it gives workbook authors, database administrators, and developers greater control over the environment. The combination of the Excel Web Access and the Excel Web Services API provides rich and versatile access to workbooks. At the same time, the security model is the same whether you access the workbook through Excel Web Access or programmatically through the Excel Web Services interface.

Workbook authors use Excel 2007 to create workbooks and accompanying business logic, including connecting directly to back-end data sources for live data. Those authors, including non-technical users, can create workbooks, reports, and dashboards with dynamic data, more securely sharing that information across the enterprise and making it available to other users and applications. Developers can take advantage of business logic created by those business users without needing to re-create it in application code. The physical infrastructure that underlies Excel Services is highly scalable and configurable. This enables you to support changing business needs as they arise and provide guaranteed availability for mission-critical calculations.

The combination of Excel Services and Excel 2007 enables both technical and non-technical users to securely share, manage, and reuse data and business logic, creating a true enterprise solution for workbooks.