

Microsoft[®] Office Excel Services Technical Overview

Date of Publication:

Build:

This is a preliminary document and may be changed substantially prior to final commercial release of the software described herein.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This White Paper is for informational purposes only. **MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.**

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, email address, logo, person, place or event is intended or should be inferred.

© 2006 Microsoft Corporation. All rights reserved.

Microsoft, Excel, SharePoint are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

Abstract

Excel Services, part of Microsoft® Office SharePoint® Server 2007 (MOSS), extends the capabilities of Excel 2007 by allowing broad sharing of spreadsheets, improved manageability and security and the ability to re-use spreadsheet models through a scalable server-based calculation service and interactive web-based user interface. Excel Services provides both a web-based UI for browser-based access and a web services API for programmatic access that have robust options for controlling what data is available to users and applications. As part of Microsoft Office SharePoint Server 2007, Excel Services also takes advantage of the other capabilities of SharePoint including the enterprise content management features such as: check-in/check-out, 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 SQL Server 2005 Analysis Services to provide a rich end-user analysis tool. Excel 2007 includes newly designed Pivot Tables, a new formula-based model for accessing Analysis Services data, improved sorting and filtering capabilities and a completely redesigned visualization engine. Excel Services empowers spreadsheet authors to share spreadsheets easily and broadly that leverage this new BI functionality through the browser. Fully interactive, data bound spreadsheets including charts, tables, and pivot tables can be created as parts of a portal, dashboard, or business scorecard, without requiring any development, and presenting a single version of the truth.

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

The application server contains Excel Calculation Service which loads spreadsheets, calculates them, and provides access to external data. Either or both of these tiers can be scaled up or out, including scaling to high-performance computing clusters. Excel Services supports configurable load balancing and a number of options that can improve performance.

Table of Contents

| | |
|------------------------------------------------------------------|-------------------------------------|
| Overview of Excel Services | 5 |
| What is Excel Services? | 5 |
| Intelligent, Secure, Sharing of Spreadsheets | 8 |
| Publishing Excel Workbooks to Office Sharepoint Server | 8 |
| Managing Permissions on Published Workbooks | 9 |
| Connecting to External Data Sources..... | 10 |
| Authenticating External Data | 11 |
| Managing Workbooks in Office Sharepoint Server..... | 12 |
| Business Intelligence Reporting and Dashboards | 14 |
| Business Intelligence Features in Office Excel® 2007 | 14 |
| Using PivotTables and Cubes Easily and Effectively | 15 |
| Bringing Data Together in Dashboards..... | 17 |
| Extending Excel Web Services | 20 |
| Exploring the Excel Web Services API | 20 |
| Implementing Excel Web Access and Other Web Parts | Error! Bookmark not defined. |
| Extending Excel Service with User-Defined Functions | 21 |
| Excel Services Architecture and Deployment | 23 |
| Communications Topology | 24 |
| Scaling and Availability | 24 |
| Optimizations for Performance..... | 25 |
| Conclusion: Security, Robustness, and Extensibility | 27 |
| Resources | 28 |

Overview of Excel Services

What is Excel Services?

Excel Services, part of Microsoft Office SharePoint Server 2007, enables businesses to securely share spreadsheets across an enterprise, making spreadsheets available through a Web browser with far greater control and manageability than has previously been possible. This server-based model complements the rich Microsoft Office Excel® 2007 client, enabling the following business benefits to the enterprise:

- **Secure Sharing of Spreadsheets** – spreadsheet authors can share workbooks while also choosing to prevent others from editing the spreadsheet or exposing any proprietary business logic that underlies it.
- **Business Intelligence** – spreadsheets connected to enterprise data sources can be published to the server and used in BI dashboards and scorecards. Users can interact with the spreadsheet in the browser to explore the data and find insights.
- **Extensibility** – reusable logic contained within spreadsheets and user defined functions are available for developers to implement in server based applications.
- **Reliability and Availability:** There are significant benefits to running calculations on servers instead of client machines. Offloading to servers frees up the user's desktop and allows them to take advantage of the 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 associated with system failure.

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

Excel Services provides two primary interfaces: a web-based UI for viewing spreadsheets in a browser, and a web services interface for programmatic access. The web-interface UI, Excel Web Access, allows users to have both interactive and read-only access to spreadsheets in a web browser by means of DHTML. No ActiveX controls are used, nor is installation of the rich client necessary. The web services interface enables external server applications to directly access spreadsheet data, making use of their business logic without needing a developer to

re-engineer it in code. In both cases, only the data that is specified by the spreadsheet author is available, while formulas and other business logic that underlie that data remains protected.

In order to better understand Excel Services it is also necessary to describe what it is not. First, and most importantly, it is NOT simply the Excel client hosted on a server. Excel Services is a new server technology designed from the ground-up to be scalable and robust. It is also not a spreadsheet-creation tool. Authoring spreadsheets requires the rich-client, Office Excel 2007. Users can, however, interact with spreadsheets through the Excel Services browser client, subject to the restrictions defined by the spreadsheet author or programmatically through the web services API. This model provides the ability on the part of the spreadsheet author to build the spreadsheet the way they want it and to restrict data consumers to only those interactions that the spreadsheet author chooses via the new browser-based access. Spreadsheet authors no longer have to use ad-hoc means of sharing spreadsheets such as e-Mail. Emailing spreadsheets often results in multiple copies of a spreadsheet, essentially creating "multiple versions of the truth" as well as exposing proprietary formulas and logic.

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

Notwithstanding the read-only behavior, spreadsheet authors can optionally mark specific cells as *parameters* using Excel 2007, enabling data consumers to perform "what if analysis" with the values in those cells – providing interactive access to the formulas and logic in the spreadsheet. For example, a spreadsheet may 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 allows the author to interactively enter a specific number of units to be sold and obtain the correct Extended Cost.

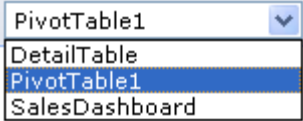
This paper will examine and explain how Excel Services works with Microsoft Excel 2007 to provide robust and secure shared spreadsheet solutions. In later sections of this paper, we

will show how Excel 2007 and Excel Services provide a great foundation for business intelligence solutions. We will also discuss extensibility features in Excel Services, and explain the methods by which developers can extend Excel Services solutions. Finally, we'll examine the architecture of Excel Services deployments, discuss how to scale out those deployments and make them highly available, as well as review performance optimizations. Before we do that, however, we should start with the first step of publishing spreadsheets to see how spreadsheet authors can easily share their spreadsheets using Excel Services, how permissions are set, how data sources are accessed, and how published workbooks are managed.

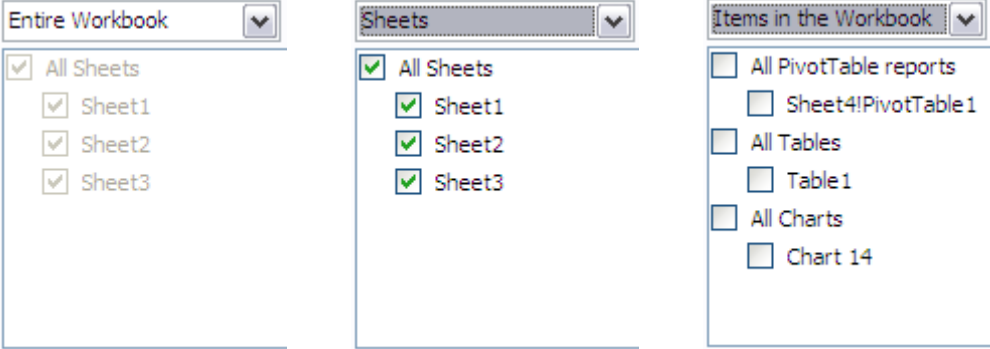
Broad, Secure, Sharing of Spreadsheets

Publishing Excel Workbooks to Office SharePoint Server

Any spreadsheet 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 spreadsheet authors can make use of Excel Services. Using the **File | Publish** menu option, a new fly-out menu called "Excel Services" enables the user to save the spreadsheet to a SharePoint (or any UNC path that is accessible by the server) server. While a full copy of the spreadsheet is copied to the server, the Excel Services Options dialogue enables the author to specify what aspects of it are accessible to users from the web browser. This is configurable at the granularity of specific named item (named items may include sheets, named cells or ranges, tables, PivotTables, or charts). Once the author has named specific items, the Items in the Workbook menu allows him or her to browse through those items, selecting whether or not each is



Users can choose dynamic named items designated by the spreadsheet author for reuse in reports and business dashboards, without requiring access to the underlying business logic or data source.



Spreadsheet authors can define viewable items at the workbook, sheet, and named-items levels.

viewable by end users.

The visibility of data under scenarios governed by Excel Services Options also applies to API calls, and not just to direct viewing by users. This mechanism constitutes a true means of control over what data is visible and to whom.

In order to make these named items readily available for users to consume, in reports or business dashboards, Excel Web Access provides a familiar interface for users to choose from among viewable items.

Since the named items remain part of a workbook that is loaded in Excel Services, they remain fully refreshable and interactive without requiring the user to be able to see the underlying business logic or data source.

Managing Permissions on Published Workbooks

What is visible, and what is not, is determined by publishing options and permissions specified by the spreadsheet author. Excel Services draws its mechanisms for authentication and access permissions from Microsoft Office SharePoint Server 2007. This allows spreadsheet authors to grant users *Reader* rights (look at content), *Contributor* rights (look at, change, and add to content), or *Administrator* rights (full control). Note that, in the context of Excel Services, the Reader right allows users to download a full version of the spreadsheet and view full details with Excel 2007. In addition to these rights, the new *Viewer* right allows authors to lock down spreadsheets for server-only viewing through a Web browser. Users with only the View Item right, cannot download a copy of the spreadsheet, 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 spreadsheet consumers is key to the complementary nature of the relationship between the Excel rich client and Excel Services. Only an in-memory snapshot of the spreadsheet is available to the users, consisting of the literal data values in the spreadsheet 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 has not been marked as viewable. In contrast, users with the Reader right can open the spreadsheet in the Excel 2007 rich client and view all of the underlying details associated with the data. Of course those with write permissions will have the ability to open spreadsheets with Excel, make changes, and save them back to a SharePoint document library.

Since spreadsheets are powerful and contain critical data, it is necessary to control where they come from, as well as control access to them. The mechanism for controlling who can act as an author for spreadsheets 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 will only load spreadsheets from directory paths designated by the administrator. With SharePoint rights (or simple file-system rights for UNC paths), administrators can control who can save spreadsheets to these trusted locations which ensures that only authors with appropriate access can create or modify spreadsheets that are presented by Excel Services.

Connecting to External Data Sources

While controlling where spreadsheets come from, and who has access to them is essential to protect proprietary formulas and presenting a single version of the truth, the spreadsheets themselves must have a flexible and robust mechanism for connecting to data, both in workbooks, and from external data sources. Access to external data is one key to the power of Excel Services, making use of existing data sources in a highly configurable and secure fashion. Spreadsheets can connect to live data sources by means of business-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 Excel client supports, such as SQL Server Analysis Services, SQL Server relational databases and other database platforms using mechanisms such as ODBC and OLE-DB. Data can be returned into PivotTables™ or new formulas that can access SQL Server Analysis Services data. Query tables are not supported on the server in this release.

One powerful way of governing access to external data sources by Excel Services is by using the Data Connection Libraries (DCLs). Created by database administrators, DCLs contain .ODC (Office Data Connection) files that persist specific data connections, enabling business users to take advantage of them, using the Excel client via a directory service operated by the server. Business users identify the connections they need that are contained within DCLs using user-friendly names and descriptions that do not require knowledge of any technical details. The significance of DCLs is definable in terms of three core capabilities:

- **Discovery:** The ability to make persistent connections to external data sources visible to business users. This capability allows connections to external data sources to be reusable, allowing business users to take advantage of them from Excel 2007 without having access to the technical details that underlie those connections.
- **Management:** The ability to support updating a large number of reports at once. Since DCLs allow changes 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, creating efficiencies and avoiding potential errors and liability.

- **Security:** SharePoint 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 for External Data Options:
 - **None** (default setting): No connections to external data sources are allowed.
 - **DCL Only:** This setting allows only DCLs to govern connections to external data sources.
 - **DCLs or Embedded Connections from Workbooks:** In addition to DCLs, this setting allows external data connections to be established by connection strings embedded in workbooks.

By choosing one of these settings, it is possible to control the transfer of data as needed. These settings can also be configured per trusted location. This allows administrators to configure the type of connectivity that is allowed to specific types of spreadsheets. For example, financial data might be stored to one location that is restricted to a limited set of spreadsheet authors, while other locations may have fewer restrictions.

Authenticating External Data

The properties dialogue associated with workbook connections in Excel client allows you to control server authentication, choosing among the following three techniques:

- **None:** This technique does not take any special action in terms of authentication, but simply forms a connection with the provided connection string. It assumes the user name and password are stored in the connection string, or that none are required.
- **SSO:** This technique implies that the credentials are stored in a single sign-on store and an SSO ID is provided at connection time. MOSS provides an out-of-the-box SSO solution that can also be replaced with custom and third party tools.
- **Windows authentication:** This technique relies on Windows authentication, also requiring Kerberos to authenticate between machines unless the server and external data sources are on a single machine.

To tailor the view for individual users in any of these cases, it is possible to set the workbook to refresh the data from the server each time the workbook is opened and to make it fail to open if external data authentication fails. Each user is only able to see the information to

which they have been granted access, avoiding inadvertently revealing inappropriate information by way of cached data.

Managing Workbooks in Office SharePoint Server

No matter how workbooks are constructed, or what permissions are applied, all spreadsheet authors must manage their workbooks. Spreadsheets must frequently be updated, sometimes by multiple authors, and updates may also require approvals. With Excel Services, these processes are made easier with Microsoft Office SharePoint Server's built-in content

Content Approval

Specify whether new items or changes to existing items should remain in a draft state until they have been approved. [More information on content approval.](#)

Require content approval for submitted items?

Yes No

SharePoint document-approval settings.

management tools. Spreadsheet management is improved by the robust check-in/check-out and versioning mechanisms of SharePoint Server which allow for major and minor version numbering, as well as security specifically for old versions of spreadsheets (and other documents). Additionally, SharePoint has built-in functionality for retention and expiration of documents, so that old versions of spreadsheets are automatically archived or destroyed in order to meet compliance requirements.

Document approval within SharePoint allows an administrator to set up a document library so that when a spreadsheet author saves a new version of a spreadsheet in the library, it is not immediately available to other users to view until it is approved. This approval can be as simple as the administrator monitoring and changing a flag on the spreadsheet in the document library, or it can depend upon a custom workflow that sends e-Mails to a group of approvers in order to ensure that the spreadsheet meets any number of internal requirements prior to its approval.

The owner of a Document Library can also set defaults for whether spreadsheets will open on the server or the Excel rich client. In cases where the Document Library owner wishes 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, preventing the secret information from being viewable.

Office SharePoint Server 2007 also allows administrators to audit document libraries. While the server does not audit items within the spreadsheets themselves, it will log any open, create, modify, and delete events to the centralized audit log. SharePoint also provides several built-in reports to analyze the audit logs, as well as tools to generate custom Excel reports.

Now that we've examined how to publish spreadsheets, set permissions on them and manage them, let's look at some of the new features in Excel 2007 and Excel Services that combine to provide both spreadsheet authors and consumers with rich controls, integrating multiple data sources, and business analytics in a way which 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.

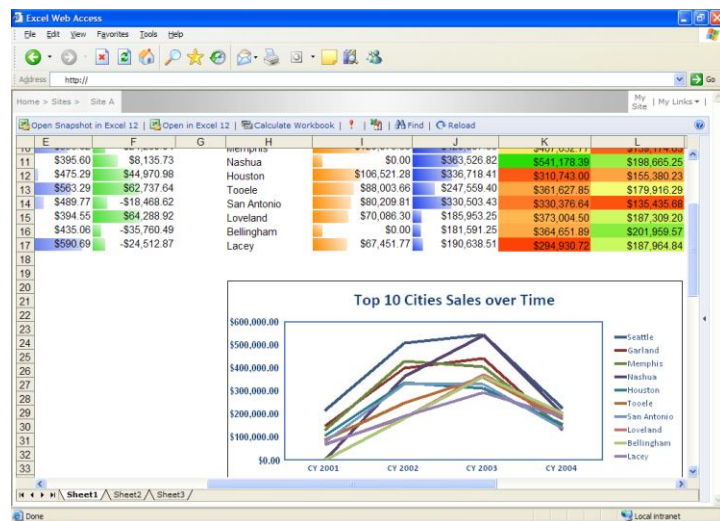
Business Intelligence Reporting and Dashboards

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 SQL Server Analysis Services. By empowering ordinary users to work more easily with cubes and to analyze data, dependence on developers and database administrators is reduced, increasing efficiency.

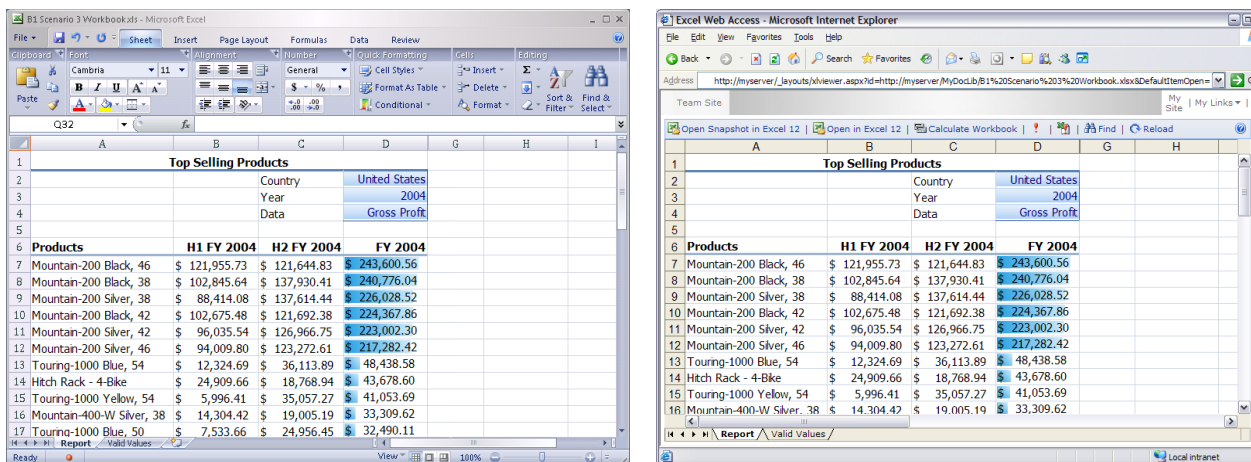
For example, users can connect to external data sources, such as SQL Server databases, in order to provide real-time analysis of data. Given a set of figures in a spreadsheet, users easily can apply visual formatting to them illustrating how those figures relate to a given threshold value. Color-coded 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 which provide explanatory graphical indicators simplifying analysis and decision making. These options are readily accessible using drop-down menus from the new Office "ribbon" user interface, in a similar fashion to how background colors are defined for a cell. Conditional formatting will change dynamically as the data is updated from the external data source. This functionality allows users to visualize Key Performance Indicators easily, and greatly enhances spreadsheets as a business intelligence tool.



New Business Intelligence features such as data bars and heat maps appear in the Excel Services web-based User Interface.

The data brought in from Analysis Services (or other external data sources) can be incorporated with addition spreadsheet calculations and charts. The reports can be formatted using Excel's rich capabilities. Calculations performed with Excel Services have 100% fidelity to those performed with Excel 2007. Visually, spreadsheets have the same formatting in both representations ensuring that the need to display them in both environments does not increase the complexity of maintaining them. Selection and hover effects that are present in Excel 2007 are also present in the browser representation via Excel Web Access, as are



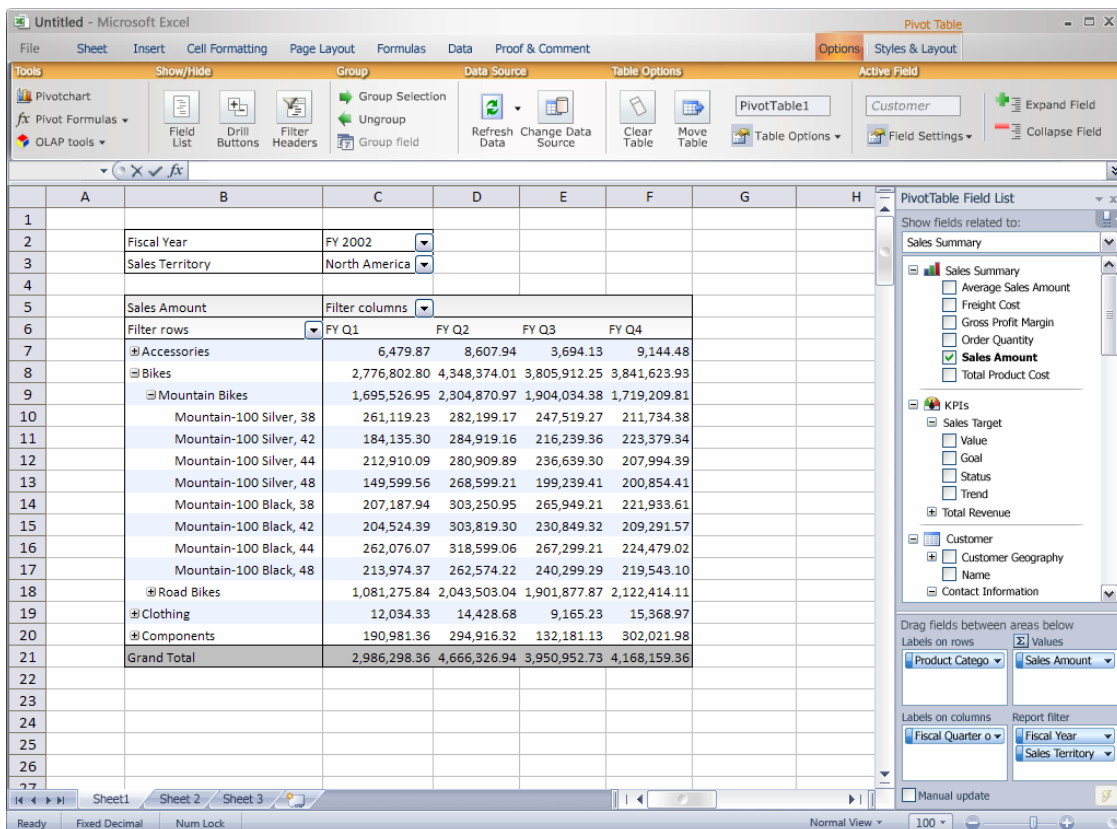
Two versions of a product sales spreadsheet, showing live data from SQL Server; the version on the left is displayed using Excel 2007, and the version on the right is displayed through the Excel Services web interface.

sorting, filtering, outlining, drill-down and drill-up capabilities.

Using PivotTables and Cubes Easily and Effectively

PivotTables™ provide the ability to interact with SQL Server Analysis Services data as well as data from any other supported data source. Excel 2007 also supports a new class of functions that can return data directly from 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 (e.g., "Above Average" and "Below Average"), and custom filters (e.g., "Less Than" and "Contains"). Spreadsheet authors can also limit users' ability to use filters.

In order to create a PivotTable, users only need to connect to a data source and simply check fields on and off the PivotTable using Excel 2007. New cube functions add to the Business Intelligence functionality of Excel 2007 and Excel Services. PivotTables can be converted directly into cube functions, enabling general users to rapidly model data on an ad-hoc basis



Excel 2007 enables PivotTables to be created from data cubes with simple drag and drop operations, with easy rearrangement of data and application of formatting.

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

Accessing PivotTables is available in both Excel 2007 and from Excel Services, providing complementary functionality can be modified to meet the user's needs. In the Excel 2007 client, the user has full control over the content. In Excel Services, the spreadsheet author

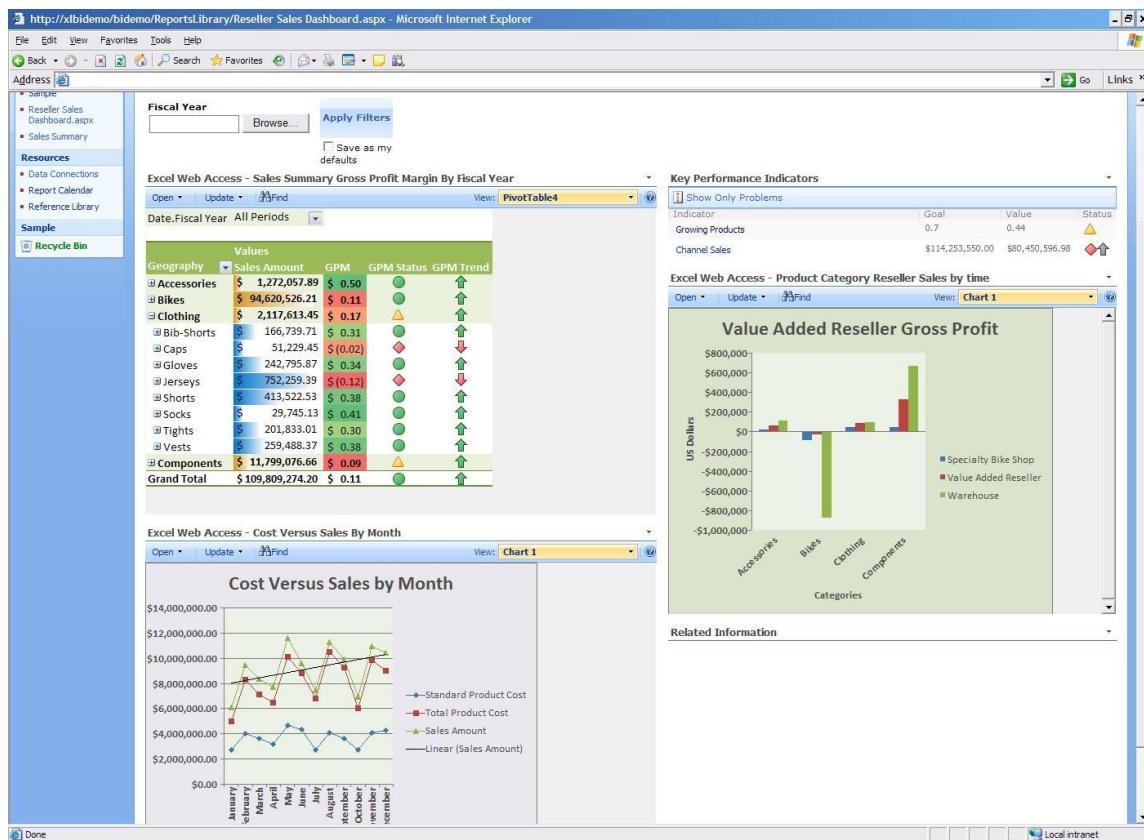
can control the level of functionality available to the spreadsheet consumer, as well as the visibility of formulas and the ability to edit data values.

Bringing Data Together in Dashboards

The new features of Excel 2007 and Excel Services allow data to be accessed, manipulated and displayed in new and compelling ways. Once data is ready for publication, information workers can combine their data with other data through the use of dashboards. Dashboards allow visualization and reporting on large amounts of data in an easily consumable format, and the ability to create business dashboards is a key Business Intelligence capability of Excel Services and MOSS.

These screens focus on identifying trends, anticipating challenges and opportunities, and supporting business decisions. Dashboards can be created easily with just a few mouse clicks and require no use of code.

SharePoint Server 2007 includes a set of building blocks called *web parts* that enable users to



Dashboard created using the Excel Services, key performance indicators, and filter web parts from SharePoint Server 2007

rapidly develop and maintain dashboards, including dynamic, interactive data capabilities such as drill-down, filter, and sort. Dashboards that are built on SharePoint Server take advantage of the user's familiarity with Excel, making use of PivotTables, Excel charts, and formulas. Dashboards also conform to the look and feel of the SharePoint environment, allowing users to automatically adhere to a corporate look and feel. Dashboards can also take advantage of SharePoint web parts external to Excel, such as Key Performance Indicators (KPIs), which provide highly visual representations of important business factors or any SharePoint or custom web part.

The Excel Web Access web part can be incorporated in a dashboard and customized. For example, it provides a variety of toolbars that can be tailored to individual situations:

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

Each of these toolbar options intelligently show only the options that are relevant to the material being 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 may be allowed at any time, or it may be restricted because of the length and complexity of the calculations that underlie the refresh functionality.

MOSS introduces a feature which allows the user to 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 extract information from a SharePoint list or profile store. This allows the user to select information based on product name, selected dates or other filtering that meets his requirement. Filters can also be created that allow the user to pick from a predetermined set of values.

It is also possible to customize a dashboard or other report based on the identity of the person viewing it. The server can capture a value, such as login ID, and pass it to the workbook as a parameter, applying filters or other logic to display only relevant (and allowed) data.

Now that we've discussed the business Intelligence features in Excel 2007 and Excel Services, and examined the powerful and easy-to-use tools built into both for displaying and modeling data, it's time to look at how to extend Excel Services solutions.

Extending Excel Web Services

Exploring the Excel Web Services API

Organizations historically have invested heavily in both building business models within spreadsheets and creating their own custom function libraries (e.g. Excel addins). Excel user defined functions (UDF) 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. In many cases these represent a significant investment. Excel Services was designed to be extensible and able to both reuse existing business models and to take advantage of extension libraries within other applications; and it provides developers with a number of options for reusing business models contained in Excel spreadsheets.

Excel Services includes a web services API that allow an application developer to access spreadsheets 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 SharePoint application. Each of those topics is described in more detail below.

As mentioned, the Web Services API provides programmatic access to spreadsheets calculated by the server. Using this API, developers can communicate with the SharePoint application server and take advantage of the Excel Calculation Service. This allows those applications to use the logic in an Excel spreadsheet without the need to have a developer 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 application developer can use Excel Web Services to make use of the business logic. All three roles are independent of one another.

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

- **Manage Session:** several methods open and close server-side calculation sessions and set properties of the server session, such as the language context of the session.
- **Set Values:** a number of methods can be used to set new values in workbook cells (or ranges of cells), definable by cell address, named range, or integer coordinates.
- **Data and Workbook Processing:** methods refresh data from an external data source and recalculate the results for the entire or parts of the workbook.

- **Return Results:** these methods get the value of a cell, a range of cells, 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 using Excel Web Access also applies to the user's ability to obtain information from an application using the Excel Web Services API. Spreadsheet data is controlled in a consistent manner with both avenues of access.

The Session ID mentioned above 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 in order 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 disk.

Extending Excel Service with User-Defined Functions

The Excel client allows extending the calculation capabilities with user defined functions. These functions can be written in VBA or any native code language. Excel Services also supports user defined functions, but the interface for developers is a managed code interface. Existing functions will need to be either wrapped or rewritten to work on the server. It is also possible to write managed code UDFs and wrap them to work on the client. Providing custom libraries enhances the core capabilities at the disposal of a spreadsheet author. Many organizations have made extensive use of these types of Excel extensions, and Excel Services was designed to support the implementation of server-side UDFs.

Note: *Many spreadsheets use the non-managed Excel Addins. In this case a UDF can be created which accesses the Excel Addin via COM interop. In this way, a spreadsheet author can continue to use his local Excel Addin and at the server, Excel Services will have the equivalent functionality.*

Once that assembly is registered on the Trusted Assembly List, it can be used to take full advantage of the UDF from formulas in Excel workbooks that are stored in trusted locations and marked to allow the usage of UDFs.

While Excel Services supports user defined functions, it does not support macros, VBA and code that access the Excel client object module.

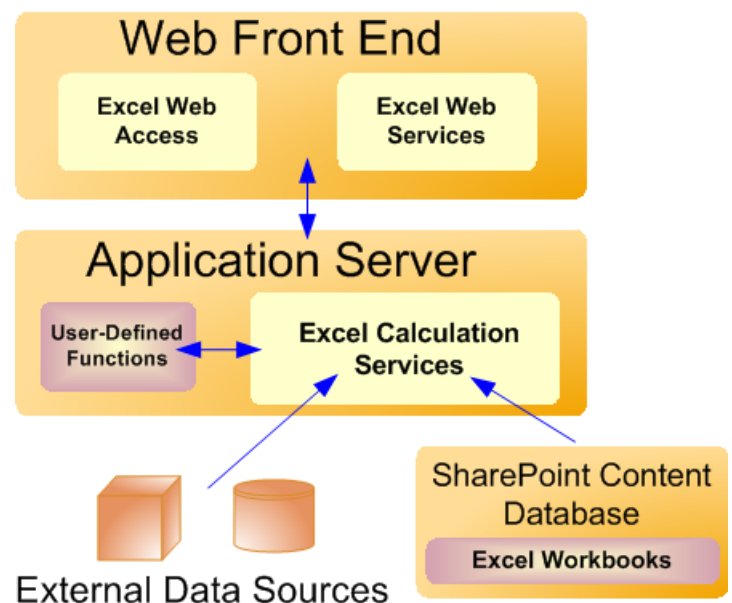
So far in this paper, we've examined what Excel Services is, discussed how to publish spreadsheets, examined how business intelligence capabilities can illuminate data, and seen the ways in which Excel services can be extended. Now we will take a closer look at the architecture of Excel Services and the deployment scenarios that are available.

Excel Services Architecture and Deployment

At a high level, Excel Services is part of Office SharePoint Server 2007 and it consists of three core components:

- **Excel Calculation Service (ECS):** This component is the 'heart' of Excel Services. It loads the spreadsheets, calculates them, refreshes external data, and maintains session state for interactivity.
- **Excel Web Access (EWA):** This SharePoint *web part* (reusable component) creates HTML renderings from the results returned by ECS 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 (EWS):** This component is a web service hosted in SharePoint that provides programmatic access to spreadsheets calculated by ECS. It allows applications to automatically update Excel spreadsheets or to incorporate calculations performed by Excel Services without having to recreate the underlying business logic.

These three components are divided into two groups: those associated with the web front end (WFE) 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 physical machines. Finally, it is possible to add additional web front end or application server machines to a configuration to increase capacity and performance.



Also included in the web front end is an ECS proxy that handles communication between the tiers in a multiple-server configuration, as well as load balancing in cases with more than one physical application server (this component is not exposed to developers).

MOSS and thus Excel Services uses two SQL Server databases at the back end: content databases and configuration databases. The content database actually stores all the user files including spreadsheets and dashboard pages etc., while the configuration database contains supporting information, such as Access Control Lists and policies that govern issues such as publication details and refresh settings. If need be, the database back end can be scaled to increase performance or capacity.

Communications Topology

Communications between the WFE and the application server is via HTTP. Once a user requests a spreadsheet, it is cached on the application server, and subsequent requests for the same spreadsheet will therefore typically be faster than the first request. This functionality includes caching at multiple levels so that collective performance for a group of users is improved by caching spreadsheets as well as the external data queried by the spreadsheets. All this is transparent to the end user except in terms of improving response time. Cached results are only shared among users with the same rights.

In addition to single-server and multiple-server topologies, it is also possible to support an extranet topology with Excel Services, giving external partners access to spreadsheets 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 (using the View Item right discussed earlier). It enables organizations to hide intellectual property such as formulas and external data connections, while showing the end results. For example, financial companies may have a financial model that they use while calculating derivatives. They can show the derivative price to their customers and partners without sharing the model used to generate it. Note, however, that Excel Services does not support scenarios where the WFE 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 WFE 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, either or both of these tiers can be scaled up or out independently of the other. Note that the application server can also be scaled out to a high-performance computing cluster, allowing 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 above, Excel Services provides load balancing between multiple application servers. That load balancing for the web front-end can be provided by the Network Load Balancing service available on certain Windows Server 2003 SKUs, or it could be provided by a separate load-balancing device. The load balancing for the application server is done by the Excel Services proxy component. The load balancer is designed to support the state maintained on specific application servers as well as to optimize based on cached workbooks. In either case, this capability includes load balancing separate spreadsheet calculations across multiple physical servers, as well as calculating more than one spreadsheet on an individual physical server (each requests runs on a different thread). Note, however, that the platform does not support breaking apart a single spreadsheet calculation across multiple physical servers. The design is optimized for scaling to large numbers of spreadsheets and requests, such as a scenario with many users viewing a dashboard with a number of spreadsheets on it, or programmatically running a large parametric sweep on a spreadsheet or group of spreadsheets.

Load balancing settings enable of a number of methods, including round-robin, local, or hash-based. In a configuration with multiple application servers, round-robin load balancing simply assigns jobs requested by the WFE to each application server in turn. The local (use this machine) option assumes that the WFE and application server are located on the same physical machine. Hash-based load balancing (the default setting) is optimized for utilizing the cache on each application server. Thus a request for any given workbook will usually return to the same back-end application server. Regardless of how Excel Services is deployed, there are several performance settings that administrators can use to optimize Excel Services.

Optimizations for Performance

Excel Services provides for granular control over many aspects of the topology that enable administrators to optimize the performance of the system. For example, 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 doesn't interact with a spreadsheet, administrators can conserve network resources.
- **Query timeout:** This setting is similar in nature to the session timeout, triggering a time-out if a query is not successfully completed in a given period of time.

- **Maximum workbook size:** This setting can specify a size limit in MB.
- **Maximum request duration:** This setting controls how long a request can last for data or a workbook.

Again, these settings provide for granular control over the behavior of Excel Services, but administrators may also choose to simply accept the default settings, which will 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

In conclusion, Excel Services empowers end-users to create robust solutions without developer involvement, while giving spreadsheet authors, database administrators and developers greater control over the environment. The combination of the Excel Web Access and the Excel Web Services API provide rich and versatile access to spreadsheets. At the same time the security model is the same whether the spreadsheet is accessed through Excel Web Access or programmatically via the Excel Web Service interface

Spreadsheet authors use Excel 2007 to create spreadsheets and the accompanying business logic, including connecting directly to back-end data sources for live data. Those authors, including non-technical users, can create spreadsheets, reports, and dashboards with dynamic data, 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 recreate it in application code. The physical infrastructure that underlies Excel Services is highly scalable and configurable, providing the ability to support changing business needs as they arise, as well as to support guaranteed availability for mission-critical calculations.

By creating a true enterprise solution for spreadsheets, 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.

Resources

The Future of Microsoft Office: 2007 Microsoft Office System

<http://www.microsoft.com/office/products>

Project REAL – Business Intelligence in Practice

<http://www.microsoft.com/sql/solutions/bi/projectreal.mspx>

Business Intelligence Case Studies

<http://www.microsoft.com/business/productivity/decision/intelligence/casestudies.mspx>

Business Intelligence Webcasts

<http://www.microsoft.com/events/series/sqlserverbi.mspx>

SQL 2005 Business Intelligence

<http://www.microsoft.com/sql/solutions/bi/default.mspx>