



www.ultimatesoftware.nl



Microsoft Dynamics AX
Business Intelligence

Microsoft Dynamics AX

Reporting and Business Intelligence in Microsoft Dynamics AX

White Paper

A roadmap for managing business performance with
Microsoft Dynamics AX

Date: September 2006

<http://www.microsoft.com/dynamics/ax/>



Contents

Executive summary	3
Microsoft Dynamics AX functional user roles.....	4
The Microsoft Dynamics AX Application User	4
The Microsoft Dynamics AX Application Developer	4
The Microsoft Dynamics AX Administrator	4
The Microsoft Dynamics AX product strategy for Reporting and BI	5
Reporting and BI in Microsoft Dynamics AX.....	7
Ad Hoc Reporting in Microsoft Dynamics AX.....	7
Features Available in the Product Today.....	7
Future Directions.....	17
Production Reporting in Microsoft Dynamics AX	18
Features Available in the Product Today.....	18
Future Directions.....	28
Multidimensional Reporting and Analytics in Microsoft Dynamics AX.....	29
Features Available in the Product Today.....	30
Future Directions.....	34
Business Scorecards in Microsoft Dynamics AX	35
Features Available in the Product Today.....	35
Future Directions.....	38
Financial Reporting in Microsoft Dynamics AX	39
Features Available in the Product Today.....	39
Future Directions.....	41
Conclusion	41

Executive summary

Business Performance Management features in a Business Application enable users across an organization to access and analyze data relevant to their functional roles in a timely and secure manner to enable improved business insight and better decision making. The term Business Intelligence or BI for short, is commonly used in the industry to label Products and Technologies that support implementing, integrating, and deploying Business Performance Management solutions to enable employees across an organization gain deeper business insight and make informed decisions.

The Microsoft Dynamics AX Reporting and BI Framework comprises tools and services to enable the implementation, integration, deployment, and use of business performance management features in Microsoft Dynamics AX solutions. The framework is a model driven Microsoft platform integrated solution that will continue to evolve in future releases of the product to support richer scenarios and advances in the Microsoft Reporting and BI platform technologies (Microsoft® SQL Server™ and Microsoft Office® system) upon which it is built.

The goal of this white paper is to provide a broad overview of the current features and future directions of the Microsoft Dynamics AX Reporting and BI framework, to address the following ERP Business Performance Management functional scenarios:

- Ad hoc Reporting
- Production Reporting
- Multidimensional Reporting and Analysis
- Business Scorecards
- Financial Reporting

The paper will also describe how the Microsoft Dynamics AX Reporting and BI framework caters to the requirements of the following user roles in implementing, deploying, administering, and using Business Performance Management features:

- The Microsoft Dynamics AX Application User
- The Microsoft Dynamics AX Application Developer (ISVs, VARs, Solution Integrators, IT Developers, and Microsoft Developers)
- The Microsoft Dynamics AX Administrator

Microsoft Dynamics AX functional user roles

This section provides an overview of the functional user roles in a production Microsoft Dynamics AX deployment. Each user persona has a stake in implementing, deploying, managing, and using Business Performance Management features.

The scenario sections that follow these user role overviews describe the features in the Microsoft Dynamics AX Reporting and BI framework that cater to the requirements of each of these personas relative to implementing, managing, and using Business Performance Management features.

The Microsoft Dynamics AX Application User

Microsoft Dynamics AX Application Users are business end-users with functional expertise in one or more application modules. These users interact with features enabled by the Reporting and BI framework to find the information that they need to perform their jobs and make effective decisions. These users typically aren't technical and prefer not to be involved with the implementation and administration aspects of the features that they use.

The Microsoft Dynamics AX Application Developer

Microsoft Dynamics AX Application Developers implement, extend, and/or customize Microsoft Dynamics AX solutions. These users are usually technical and have a background in software development acquired through formal education and work experience. This user role includes ISVs, VARs, Solution Integrators, IT Developers, and Microsoft Developers who implement Microsoft Dynamics AX. Advanced development tools and features that enhance their productivity when implementing Microsoft Dynamics AX solutions add the most value for these users.

The Microsoft Dynamics AX Administrator

Microsoft Dynamics AX Administrators are responsible for deploying, securing, and maintaining production deployments of Microsoft Dynamics AX. These users are usually technical and have a strong background in IT Administration, typically acquired through education and work experience. They are typically not application developers, although in many small and mid sized companies, the Microsoft Dynamics AX Application Developer and Administrator roles undertaken by a single team that consists of members with skills in both areas. Product features that support and simplify deploying, securing, monitoring, and maintaining Microsoft Dynamics AX solutions in a production environment are of primary importance to these users.

The Microsoft Dynamics AX product strategy for Reporting and BI

Microsoft is firmly committed to investing in building and enriching its BI platform technologies and products.

The primary goal of these investments is to fully realize the “BI for the masses” vision by delivering functionally comprehensive and low TCO BI platforms and applications to permeate the benefits of BI across all levels of users in an organization. Figure 1 shows this vision.



Figure 1: The Microsoft Business Intelligence Vision

The Microsoft BI Stack contains an integrated suite of platform products, technologies, and applications that can be used to implement, deploy, manage, and use rich Business Performance Management features to address the core scenarios listed in the Executive Summary section.

Figure 2 provides an overview of the platform products, technologies, and applications that form the current Microsoft BI stack. The stack consists of the following components:

- Data Platforms: Core relational ([SQL Server](#)) and Multi dimensional ([SQL Server Analysis Services](#)) database engines to facilitate the storage and retrieval of data for Reporting and Analysis.
- Data Integration Services: Platform services ([SQL Server Integration Services](#)) to support the extraction, transformation, and loading (ETL) of data from multiple

homogeneous and/or heterogeneous data sources to facilitate consolidated Reporting and Analysis.

- A Reporting Platform ([SQL Server Reporting Services](#)) that consists of services and tools to enable designing, deploying, integrating, and managing reports required to address information analysis requirements.
- Rich Ad Hoc End-User Reporting and Analysis tools (Microsoft® Office Excel® and [SQL report Builder](#)) to facilitate self-service reporting and analysis.
- A Scorecarding, Analytics, and Forecasting/Planning platform ([Microsoft® Office PerformancePoint™ Server 2007](#))
- A Portals and Collaboration platform ([Microsoft® Office SharePoint® Server 2007](#)) to enable Web portals to present, access, and share information.

Microsoft Business Intelligence

A Complete Solution

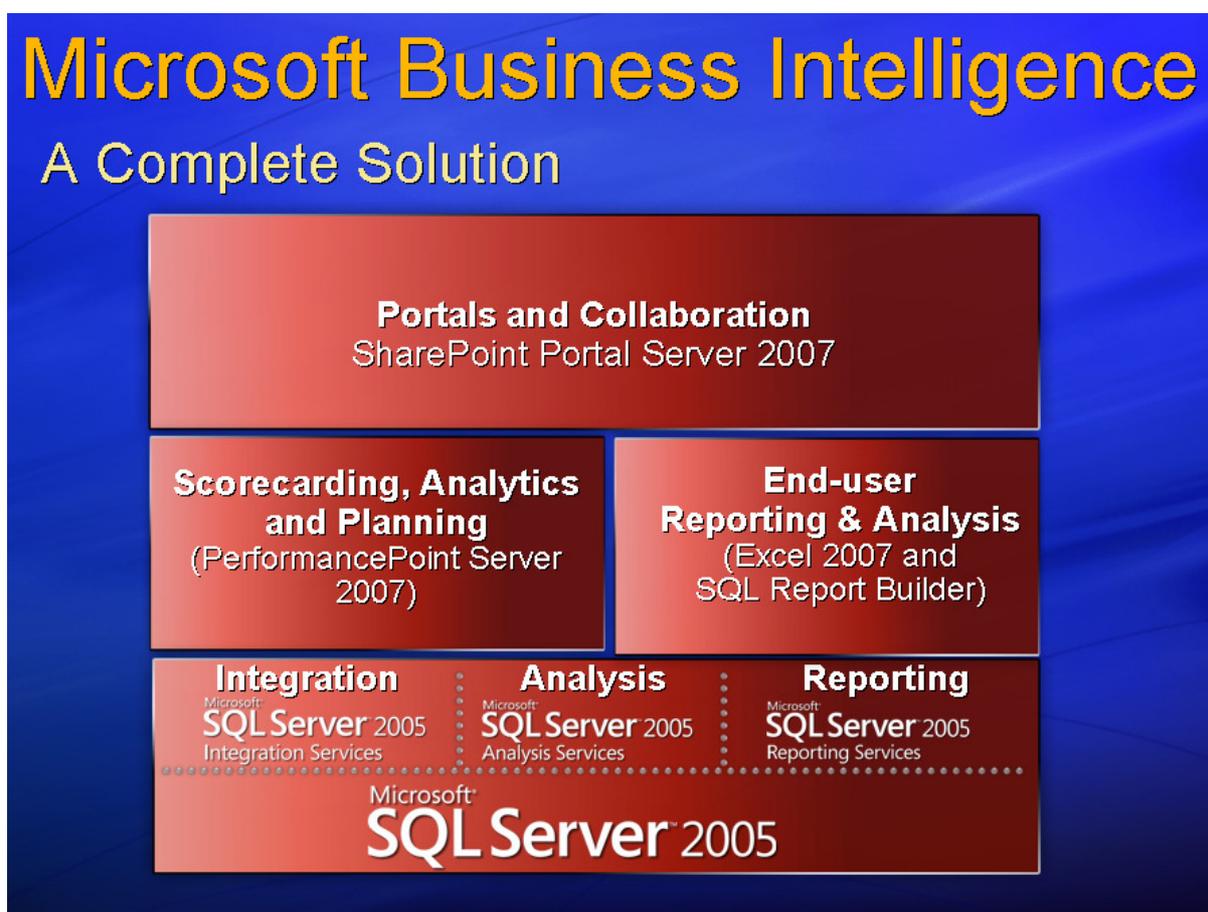


Figure 2: The Microsoft BI Stack

Microsoft is committed to enhancing the BI stack and enabling richer scenarios in future releases of the platforms and applications.

The product strategy for Reporting and BI in Microsoft Dynamics AX is to integrate tightly with, and take advantage of, the richness, capabilities, and enhancements to the Microsoft BI Stack. Many significant steps have been taken in this direction in the latest release of Microsoft Dynamics AX (Microsoft Dynamics AX 4.0), and there is more to come in the future releases. The following sections in this paper on the Business Performance Management features in Microsoft Dynamics AX will highlight the integration points in the current release and summarize related future directions.

Note: This paper will not describe the implementation architecture and details of the products and technologies in the Microsoft BI Stack. The paper will assume general reader familiarity with the Microsoft BI Stack, and will only focus on the related integration points in Microsoft Dynamics AX.

Reporting and BI in Microsoft Dynamics AX

This section describes the features available today in the Microsoft Dynamics AX Reporting and BI framework and planned future directions to address the following Business Performance Management functional scenarios that are core to current generation Business Management systems:

- Ad Hoc Reporting
- Production Reporting
- Multidimensional Reporting and Analysis
- Business Scorecards
- Financial Reporting

Ad Hoc Reporting in Microsoft Dynamics AX

Ad Hoc Reporting capabilities enable application users to create, view, and save reports that satisfy their individual information analysis requirements. Abstracted tools and services that do not require user knowledge of the underlying database schemas or experience in programming and advanced report authoring are required to enable this scenario. This scenario is also typically referred to as “self service” reporting. The primary goal for Ad Hoc Reporting is to enable application users to locate, select, filter, sort, group data, view, and format the results using predefined layout templates.

Features Available in the Product Today

Note: The features described in this section are supported in the current release (Microsoft Dynamics AX 4.0) only when the Dynamics AX database is a SQL Sever 2005 instance.

Model driven integration with SQL Server Report Builder is a new feature in Microsoft Dynamics AX 4.0 that enables secure Ad Hoc Reporting.

SQL Server Report Builder is a new Ad Hoc Reporting framework and tool introduced in SQL Server Reporting Services 2005 to enable self service end-user reporting. The following webcast provides an overview of SQL Report Builder architecture and the scenarios that it enables. We recommend that you view this webcast if you are not aware of the capabilities and scenarios enabled by SQL Report Builder:

[End-user Ad Hoc Reporting with SQL Server Reporting Services](#)

Figure 3 illustrates the integration architecture. This implementation is a prime example of the Microsoft BI stack integration focus of the future direction for Reporting and BI in Microsoft Dynamics AX

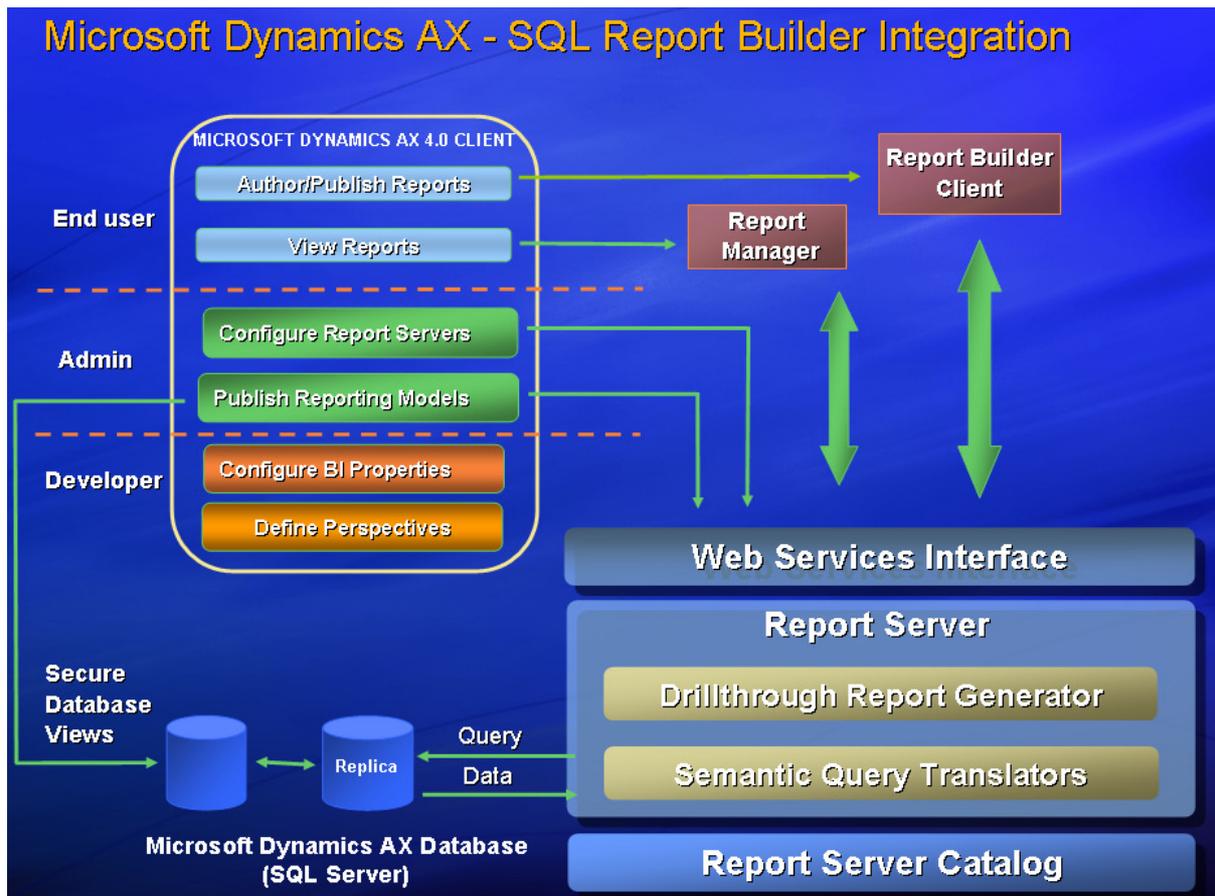


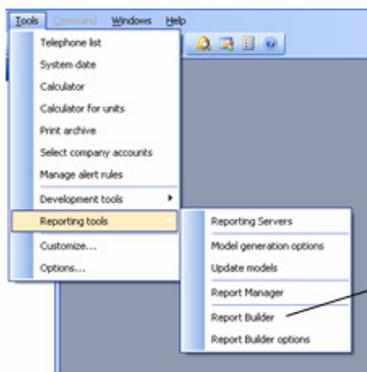
Figure 3: The Microsoft Dynamics AX – SQL Report Builder Integration Architecture

Features for the Microsoft Dynamics AX Application User

Application users can start SQL Report Builder from the Microsoft Dynamics AX client to design Ad Hoc reports and publish them to a configured SQL Report Server. They can use all features of SQL Report Builder to navigate, select, group, sort, and filter data when they design Ad Hoc Reports. Ad Hoc reports designed in SQL Report Builder can be formatted to present data in tabular, matrix (cross tab), and various chart layouts.

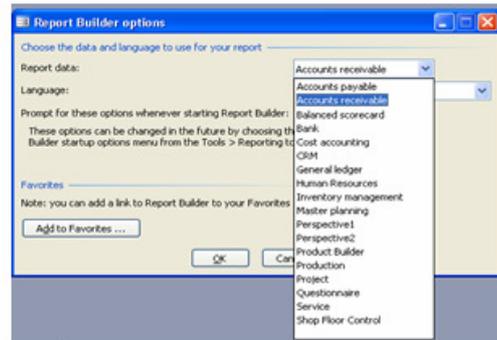
Application users can also start the SQL Report Manager from the Microsoft Dynamics AX client to browse report server catalogs and view published reports.

Figures 4 and 5 illustrate how users can start and use SQL Server Report Builder and Report Manager from the Microsoft Dynamics AX Client.

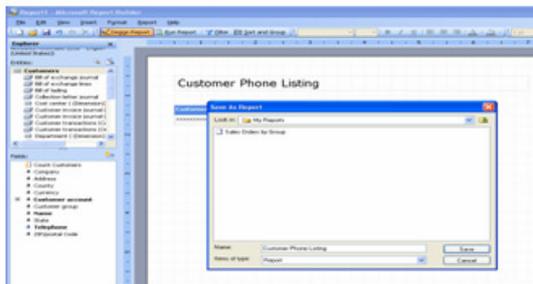


Step 1: User selects the Reporting tools menu option to launch SQL Server Report Builder

Step 2: User selects a data model (perspective)



Step 3: User designs Ad Hoc reports using SQL Report Builder



Step 4: User publishes Ad Hoc reports to a SQL Server Report Server

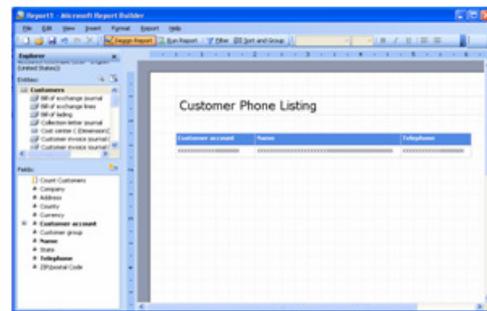


Figure 4: Starting and using SQL Report Builder from the Microsoft Dynamics AX client

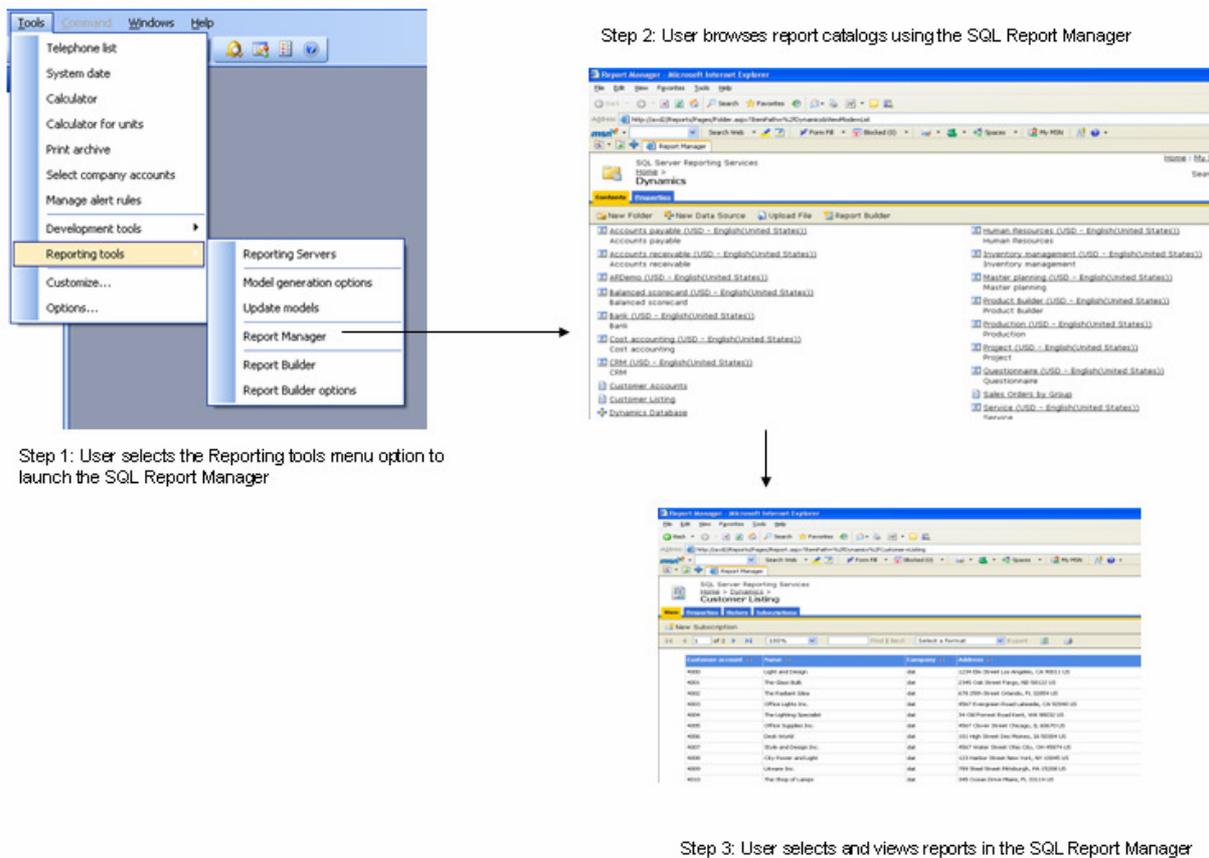


Figure 5: Starting and using SQL Report Manager from the Microsoft Dynamics AX client

Features for the Microsoft Dynamics AX Application Developer

Application developers define abstracted views of the Microsoft Dynamics AX data model to enable Ad Hoc end-user reporting. These abstracted views are materialized as SQL Report Builder Semantic Model Definition Language (SMDL) models and deployed to a SQL Report Server. Application users can open a SMDL model in SQL Report Builder and use it to author Ad Hoc reports.

Building a secure production quality SQL Report Builder model that abstracts data relationships that are defined in the Microsoft Dynamics AX metadata and reflects core Microsoft Dynamics AX application semantics is a complex developer task. The Microsoft Dynamics AX Reporting framework includes developer features that shield the developer from the intricacies of building secure production quality abstracted views of the Microsoft Dynamics AX data model to enable Ad Hoc Reporting with SQL Report Builder.

The following are the steps that a Microsoft Dynamics AX developer has to execute to define views of the Microsoft Dynamics AX data model for Ad Hoc Reporting:

Define Perspectives

A new metadata type called **Perspective** was added to the Microsoft Dynamics AX data dictionary in the Microsoft Dynamics AX 4.0 release. A Perspective is a collection of related tables that represent a subset of the Microsoft Dynamics AX data model. Instances of this type can be created to define module specific and/or role based views of the Microsoft Dynamics AX data model for Ad Hoc Reporting and Analysis. Perspectives defined in a Microsoft Dynamics AX application model are deployed as SQL Report Builder SMDL

models to a SQL Report Server. Application users can access the published SMDL models using SQL Report Builder and author Ad Hoc reports to address their information analysis requirements. All data relationships that are defined in the Microsoft Dynamics AX metadata for the Tables added to a **Perspective** are automatically reflected in the SMDL models generated by Microsoft Dynamics AX without requiring any explicit developer effort.

Best Practice Recommendation: Microsoft Dynamics AX 4.0 contains **Perspectives** for each of the application modules as part of the shipped product. Users are strongly encouraged to define more detailed role based **Perspectives** to support the Ad Hoc reporting requirements of specific user roles in their organizations.

Figure 6 illustrates the steps to define a new Perspective for Ad Hoc Reporting.

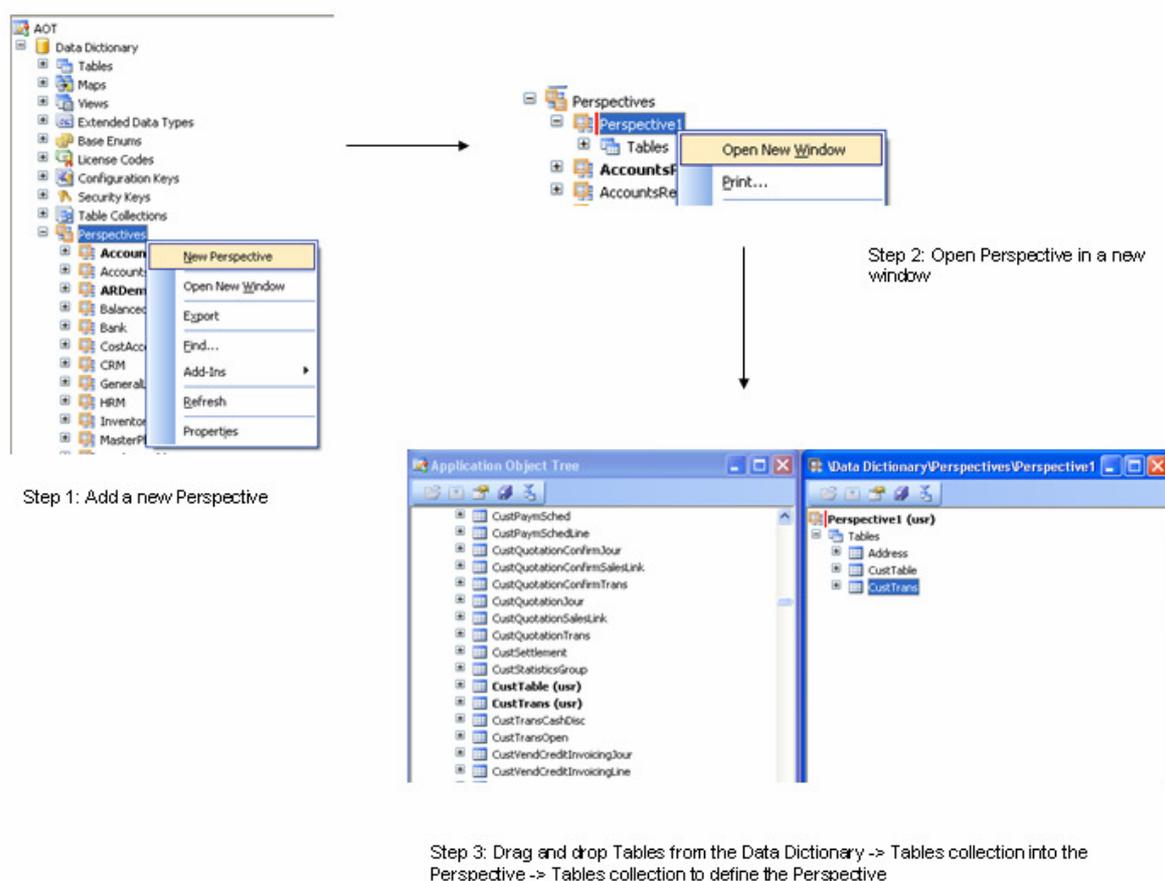


Figure 6: Defining a new Perspective

Configure BI Metadata Properties

Developers can define settings for the BI metadata properties of the Microsoft Dynamics AX data model objects (Tables, Fields, Extended Data Types, and Security keys) to configure the content, visibility, and use of data elements exposed to application users in the Ad Hoc Reporting models. The BI properties can be set to configure aspects such as the visibility and visual positioning of data elements in the SQL Report Builder Model Explorer, the generation of aggregate fields, the currency conversion semantics for monetary fields, the flattening of lookup table fields into referencing tables, and UI requirements for selecting records when users build Ad Hoc reports. The following table lists some of the most frequently used BI metadata properties. We recommend that you refer to the Microsoft Dynamics AX product documentation for a listing of all BI properties and descriptions of their usage.

Object	BI Property	Description
Table	AnalysisVisibility	Used to control the visibility of a table in Ad Hoc reporting models
	TypicalRowCount	Used to specify the approximate number of records in the table to control value selection when you perform analysis
	IsLookup	Used to specify if a table is a lookup table whose analysis fields must be flattened into the field lists of tables that reference it
Table Field	AnalysisVisibility	Used to control the visibility of a table in Ad Hoc reporting models
	AnalysisTotaling	Used to specify the aggregations to generate for the field
	CurrencyCode	Applies only to monetary fields. Specifies the currency code to use for the field when rolling up aggregates for multi currency
	CurrencyDate	Specifies the date to use to retrieve the conversion rate for translating instance values of the field
Security Key	AnalysisVisibility	Used to control the visibility of a tables associated with a security key in Ad Hoc reporting models
Extended Data Type	AnalysisDefaultSort	Specifies the default sort order to use when fields of the type are added to an Ad Hoc report

Features for the Microsoft Dynamics AX Administrator

The Microsoft Dynamics AX Reporting Framework includes a comprehensive feature set to support the deployment and administration of Ad Hoc Reporting. The Microsoft Dynamics AX Administrator is responsible for the following related tasks:

Configuring and Registering SQL Report Servers

Ad Hoc Reporting **Perspectives** are materialized as SQL Report Builder SMDL models that must be deployed to a SQL Report Server. Administrators can use the **Reporting Servers** administration form to register and configure SQL Report Server instances to which Ad Hoc Reporting models will be published to enable application user access. Figure 7 illustrates the steps to register and configure a SQL Report Server instance in Microsoft Dynamics AX.

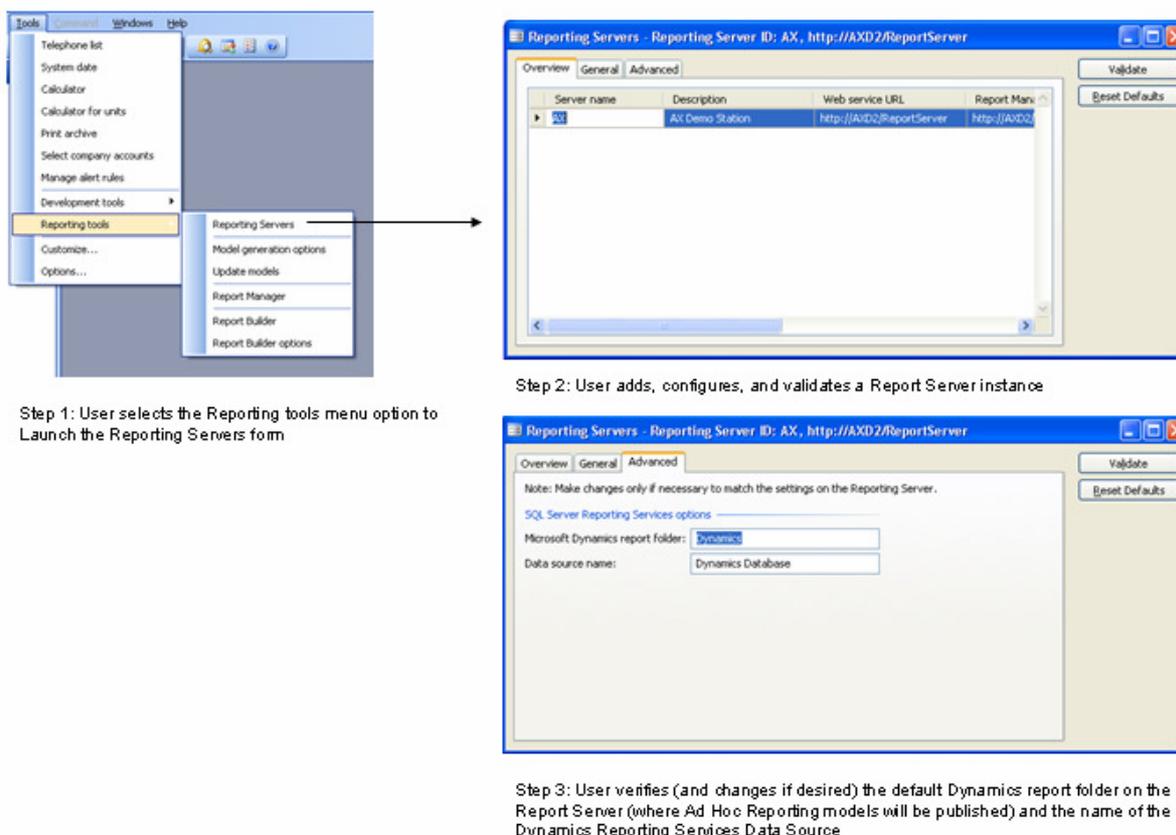


Figure 7: Configuring Report Servers

The option to set up a Reporting Server role can be selected when you install Microsoft Dynamics AX. This option, when it is selected, will require the user to specify a SQL Report Server instance that must be configured to support Ad Hoc Reporting. The installer will create a default Microsoft Dynamics report folder and report server data source on the specified SQL Report Server instance. This report server instance must be registered by the Administrator (using the Reporting Servers administration form) before Ad hoc reporting models can be published to it.

Administrators can also use the SQL Report Manager and/or SQL Server Management Studio to manually create and configure Microsoft Dynamics AX report folders and Microsoft Dynamics AX data sources. This is typically done when configuring a Report Server role is omitted during application installation or when multiple Reporting Server instances must be configured and registered to support Ad Hoc Reporting in a Microsoft Dynamics AX deployment.

Best Practice Recommendation: Accessing an OLTP (Online Transaction Processing) database for Ad Hoc reporting is generally not recommended unless real time data access is a mandatory requirement. Creating replicas of transactional databases to support Ad Hoc reporting is a strongly encouraged practice to off load related query processing from production OLTP databases. The flexibility to register multiple Report Server configurations supports mixed mode deployments to address different latency requirements. Administrators can use the SQL Report Manager or SQL Management Studio to configure Microsoft Dynamics AX Report Server data sources for real time and non-real time data access. They can then register Report Server instances in Microsoft Dynamics AX (using the Reporting Servers administration form) to enable Ad Hoc Reporting to take advantage of both the real time and the non-real time data sources.

Configuring Model Generation Options

Ad Hoc reporting **Perspectives**, defined by Microsoft Dynamics AX developers, must be published as SQL Report Builder SMDL models to a registered SQL Report Server to enable Ad Hoc Reporting in Microsoft Dynamics AX. The **Model Generation Options** form can be used to configure model deployment options and deploy/update Report Builder SMDL models for Ad Hoc reporting **Perspectives**.

The Administrator can use the **Model Generation Options** form to:

- Configure the destination location for publishing Ad Hoc Reporting models. The destination location can be a registered Report Server or a file folder on disk. The file folder destination option is an advanced option that can be used to generate the Report Builder SMDL models to a folder on a disk to enable manual model customization using the SQL Report Model Designer. The models must eventually be published to a registered SQL Report Server before they can be leveraged for Ad Hoc Reporting.
- Configure Log file generation to troubleshoot model generation failures.
- Select the default exchange rate company and currency to use for multi currency rollups.
- Control the display of elements that contain no data in the SQL Report Builder Model Explorer.
- Configure the sorting of fields displayed for an Entity in the SQL Report Builder Model Explorer.
- Select the languages for which localized versions of Ad Hoc Reporting models must be published. Label translations defined for the data elements in the Microsoft Dynamics AX metadata are used to generate localized versions of the Ad Hoc Reporting models for each selected language.
- Deploy/Update Report Builder models using the settings configured for the model generation options.

Figure 8 shows the Model Generation Options form in Microsoft Dynamics AX.

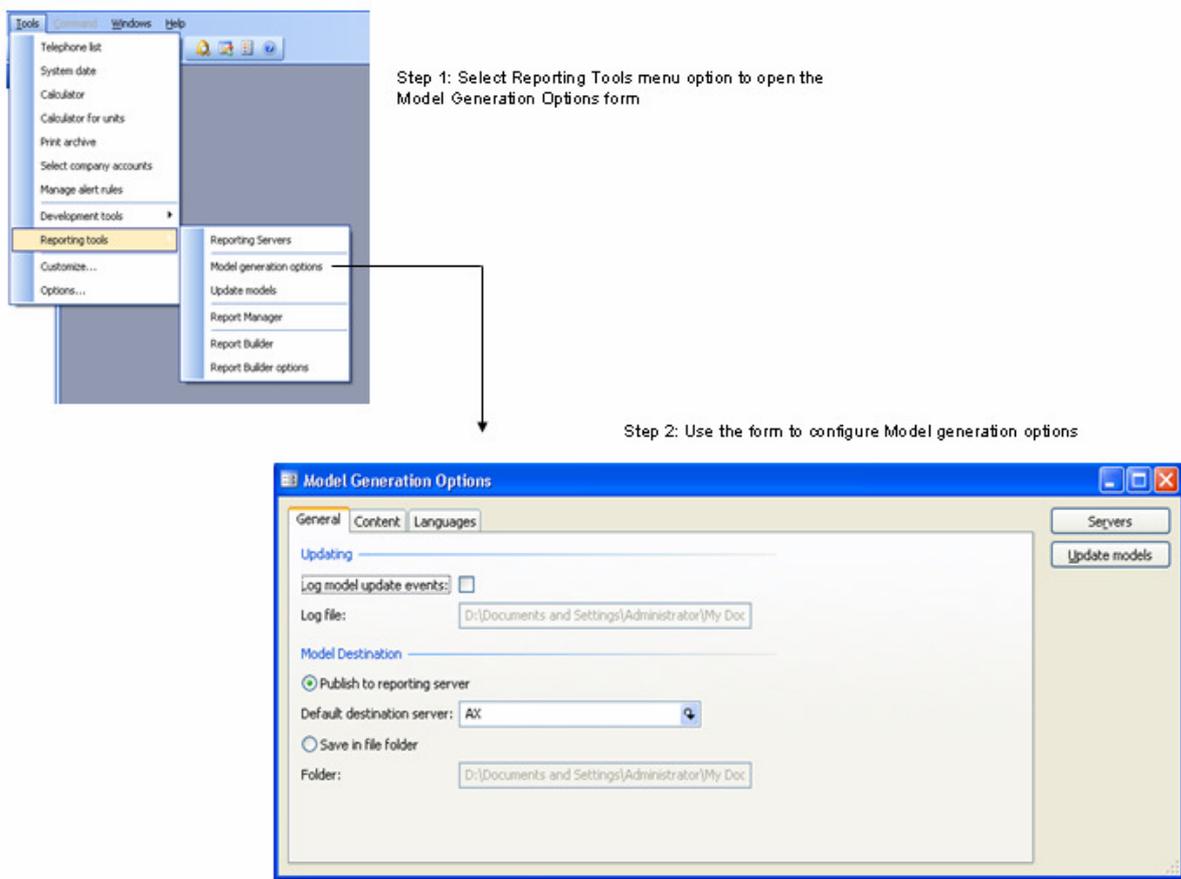


Figure 8: Configuring Model Generation Options

Updating Ad Hoc Reporting Models

The Administrator can use the **Update Models** form to publish/update SQL Report Builder SMDL Models for Ad Hoc Reporting Perspectives defined in the application metadata. The settings that are defined in the **Model Generation Options** form are used to publish/update the models. The Administrator can select to publish/update models for a single language or all languages selected in the **Model Generation Options** form. The Administrator can also select to execute the publish/update process as a Microsoft Dynamics AX batch job.

Figure 9 shows the Update Models form.

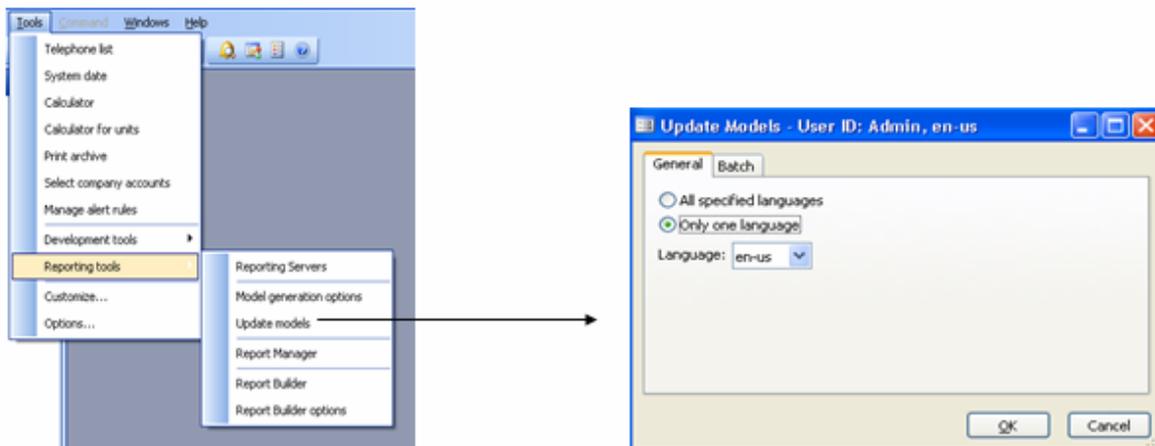


Figure 9: The Update Models form

Securing Ad Hoc Reporting

Securing Access to Reporting Tools

The Reporting Tool options that a user can access are typically determined by the user's role and persona. For example, permitting application users to access the Reporting Servers, Model Generation Options, and Update Models forms is generally not required or desired. Administrators can use the Microsoft Dynamics AX **User group permissions** form to control access to the Reporting tools. Figure 10 illustrates the **User group permissions** form being used to deny the Retail Stores Sales Representatives user group access to the Reporting Servers form.

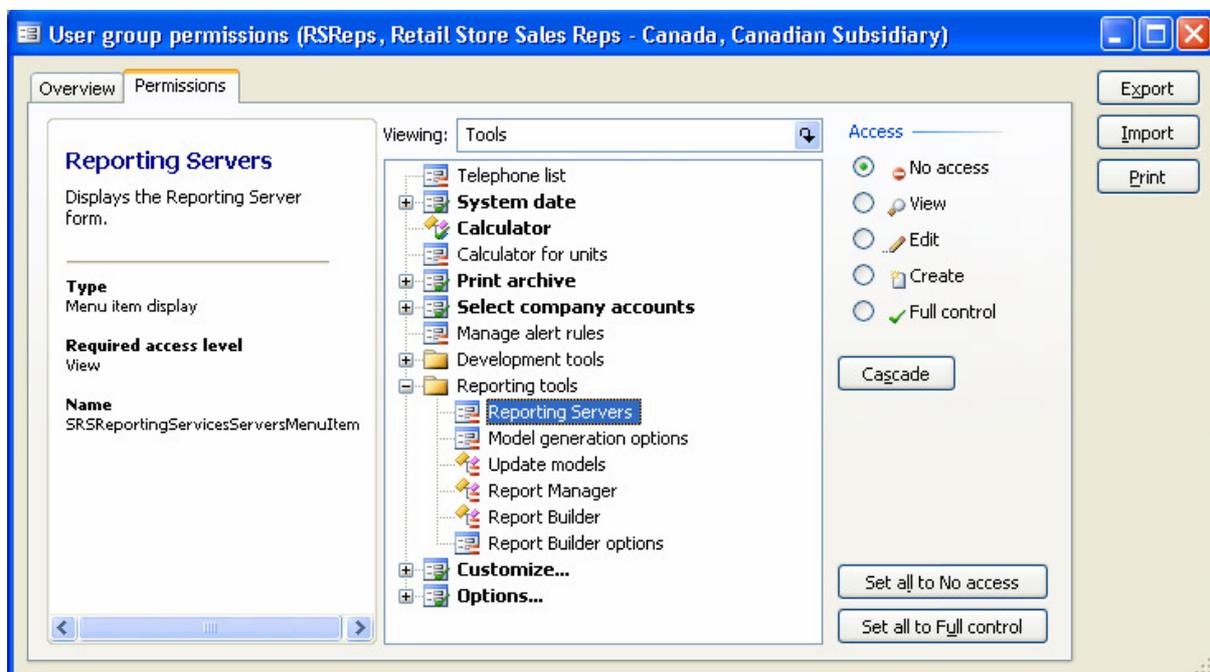


Figure 10: Using the User group permissions form to control access to Reporting tools

Data Access Security

Data Access Security for Ad Hoc Reporting is addressed by Secure Database Views deployed by the Microsoft Dynamics AX Reporting framework for tables referenced by Ad Hoc Reporting Perspectives. These views are published to the Microsoft Dynamics AX database when Ad Hoc Reporting models are deployed, and are used to resolve data access queries generated by Ad Hoc reports. The secure database view implementations encompass all semantics required to enforce field and record level security as defined in Microsoft Dynamics AX. This ensures secure data access for Ad Hoc reporting. The view definitions are also auto updated when changes are made to the Microsoft Dynamics AX security configurations.

Report Server Security

Standard SQL Report Server security features and tools can be leveraged to control access to report folders, Ad Hoc Reporting SMDL models, and published reports. The following are some common Report Server security administration tasks related to Ad Hoc Reporting:

- Deny users and/or user groups access to Ad Hoc Reporting SMDL models generated by Microsoft Dynamics AX.
- Enable the **My Reports** SQL Report Server option to provide each application user a personal report folder.
- Limit users and/or user groups access to shared report folders.

The SQL Server product documentation contains detailed conceptual and reference information and step-by-step tutorials on topics related to administering Report Server security.

Best Practice Recommendations:

1. Enable the **My Reports** SQL Report Server option to provide each application user a personal report folder where they can publish their Ad Hoc reports. Look up the SQL Server online documentation for more information about this option and how it can be enabled.
2. Limit application user access to shared report folders to prevent random publishing of Ad hoc reports and accidental deletion/modification of shared items.

Future Directions

The following are the core enhancements planned for Ad Hoc Reporting in future releases of Microsoft Dynamics AX:

- Role based Ad Hoc Reporting Perspectives and Models shipped as part of the product.
- Use of Dynamics AX Views when you define Ad Hoc Reporting Perspectives.
- Selective publishing of Ad Hoc Reporting Models.
- Integration and access of Ad Hoc reports from Application menus, My Favorites, and Forms
- Denormalized relational data marts for Ad Hoc Reporting.
- Multi dimensional Ad Hoc Reporting models that are based on SQL Server Analysis Services cubes.

Production Reporting in Microsoft Dynamics AX

Predefined reports in a business solution are known as Production reports. These reports are designed and integrated into a solution to enable access to frequently referenced data. Production reports have predefined layouts, support limited user interactivity, and present data that has been identified as being useful to multiple users.

Features Available in the Product Today

The core Microsoft Dynamics AX solution contains hundreds of production reports distributed across the various application modules. Microsoft Dynamics AX developers can design new production reports, customize existing reports, and integrate reports into a Microsoft Dynamics AX solution. Microsoft Dynamics AX application users can access these reports through application menu items and action buttons on forms to which they are connected. Microsoft Dynamics AX Administrators can configure user group permissions to control access to Production reports in Microsoft Dynamics AX.

Features for the Microsoft Dynamics AX Application Developer

Developers can use the native Microsoft Dynamics AX report writer and SQL Server Reporting Services to implement secure production reports for Microsoft Dynamics AX solutions.

The Microsoft Dynamics AX native report writer

The Microsoft Dynamics AX native report writer is the primary tool for designing and integrating secure production reports in Microsoft Dynamics AX. The definitions of production reports designed using the native report writer are stored in the application metadata and can be customized in a layered manner like all other Microsoft Dynamics AX objects. These reports can be easily linked to menu items and forms in the Microsoft Dynamics AX Windows client and Enterprise Portal.

The native report writer and related tools have existed since the initial release of Microsoft Dynamics AX. Developers can design production reports using the native report writer in one of the following ways:

- Using the Report Wizard. The Report Wizard makes it easy to create reports using the native report writer. The wizard helps the developer select the desired report data and determine how the data should be sorted, grouped, filtered, and formatted. Report definitions generated by the wizard can be further edited and customized in the Application Object Tree (AOT) and/or by using the native What You See Is What You Get (WYSIWYG) Report Designer.
- Creating reports in the AOT. The developer can create new report objects in the AOT, define queries to configure report data sources, and generate auto designs from the data source query definitions. The report designs can be further edited and customized directly in the AOT and/or by using the native WYSIWYG Report Designer.

An existing production report can be customized by locating the Report definition in the AOT and editing it in place.

New report designs and customizations applied to existing designs can be previewed in the AOT.

This section only introduces the range of native report writer features that are used widely to implement production reports in Microsoft Dynamics AX solutions. The Microsoft Dynamics AX product documentation contains more information about how to use the native report writer and its advanced features to design and integrate production reports in Microsoft Dynamics AX.

Figures 11, 12, 13, and 14 illustrate some of the native report writer features discussed in this section.

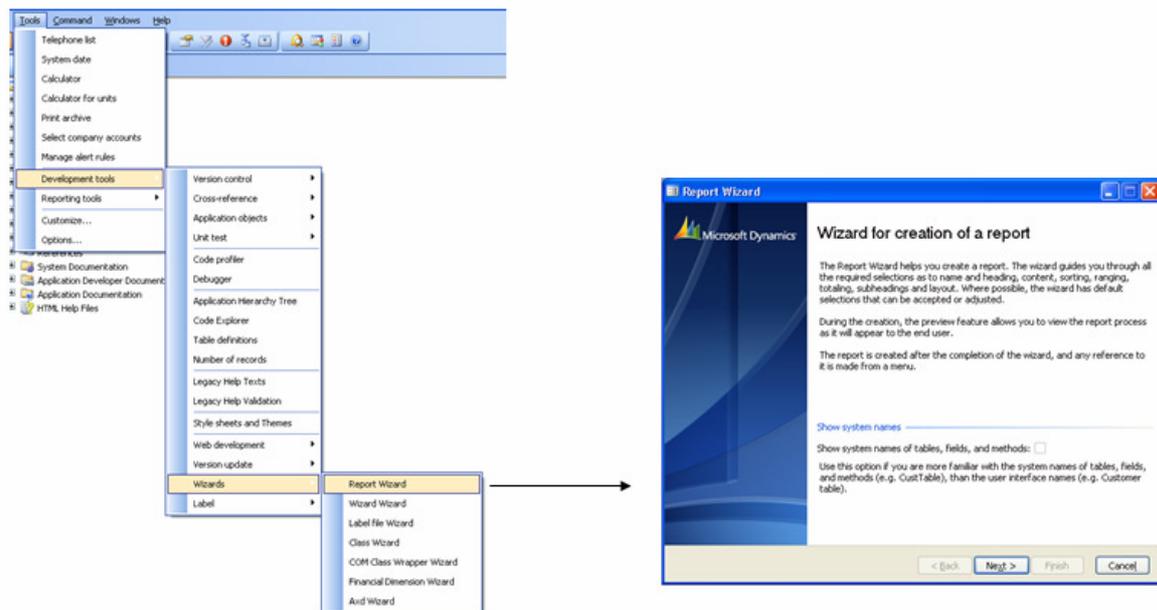


Figure 11: Starting the Report Wizard

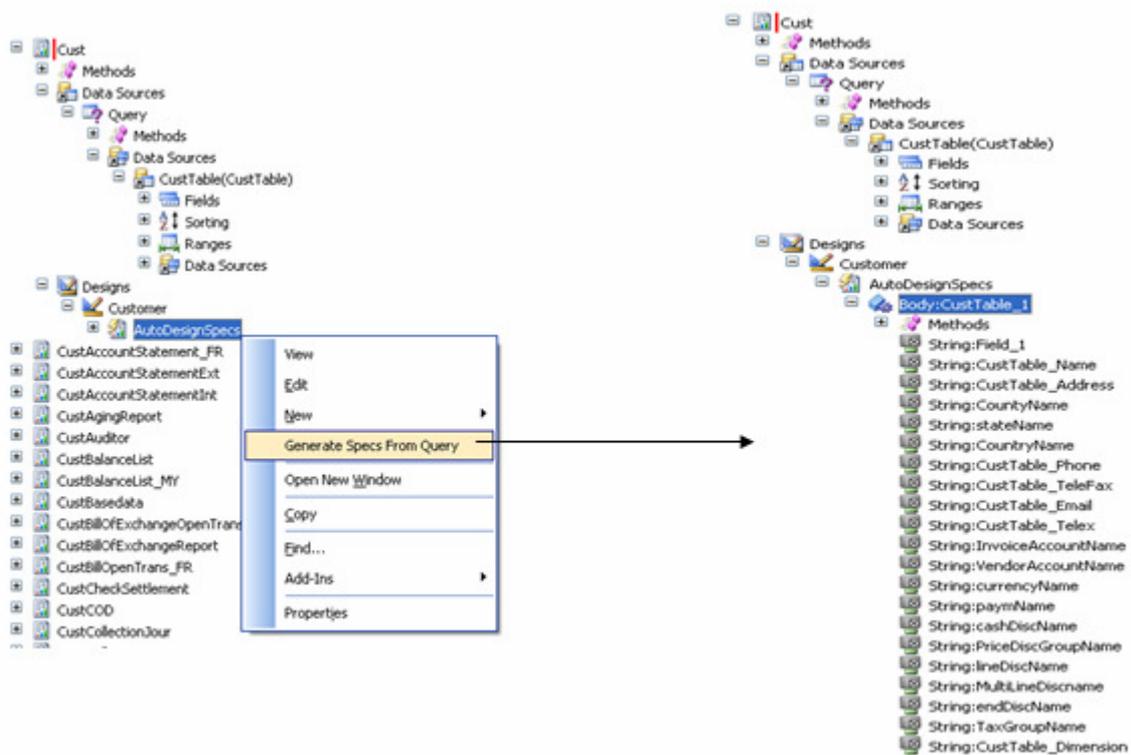


Figure 12: Generating Auto Report Designs in the AOT

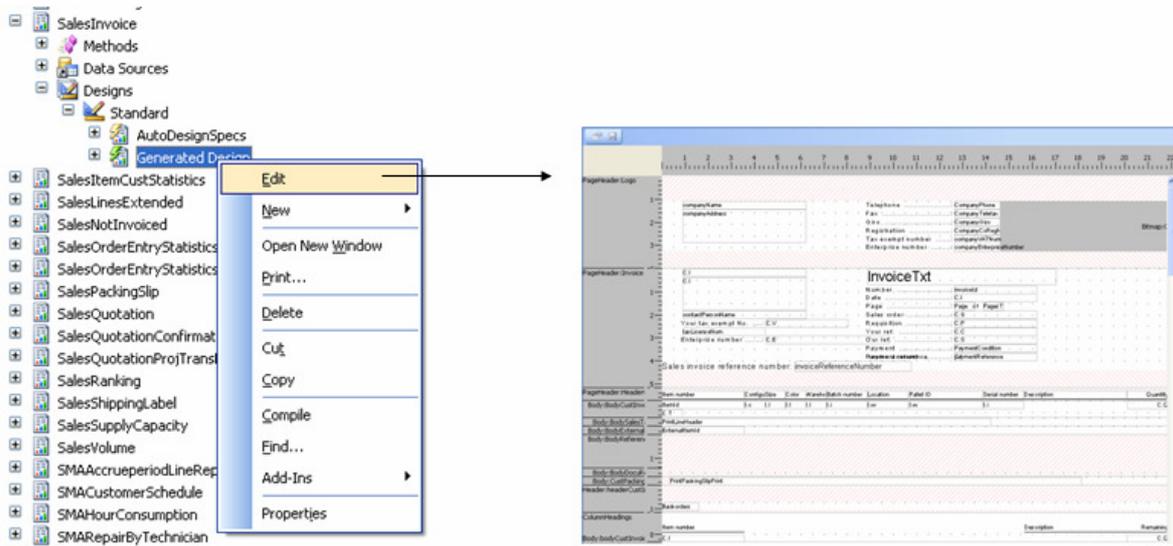


Figure 13: Accessing the native WYSIWYG Report Designer

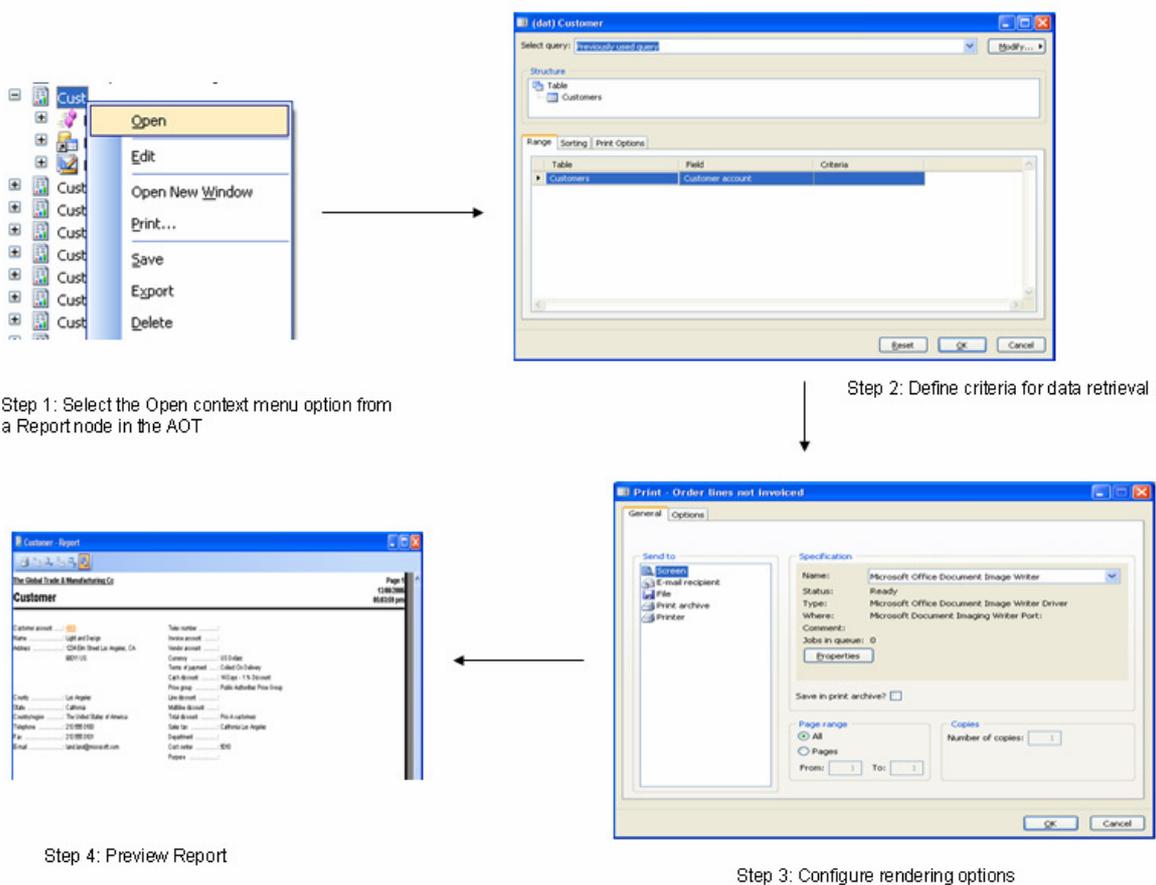


Figure 14: Previewing a Report

Using SQL Server Reporting Services

Note: The features described in this section are supported in the current release (Microsoft Dynamics AX 4.0) only when the Dynamics AX database is a SQL Server 2005 instance.

Developers can also use SQL Server Report Builder Models generated for Microsoft Dynamics AX Perspectives to design secure production reports using the SQL Report Designer and SQL Report Builder.

The SQL Report Builder can be used to design simple production reports whose data retrieval/processing requirements do not require the execution of business logic and whose visual design requirements can be satisfied by simple tabular, matrix, and chart layouts.

The SQL Report Designer is better suited for more complex production reports that require business logic execution and have free form precision layout design requirements.

Note: The SQL Report Designer and the SQL Report Builder are different tools. They are both components of the SQL Reporting Services platform, but serve different purposes. The SQL Report Builder is a stand-alone smart client application primarily targeted at business application users to enable Ad Hoc reporting. It can however also be used to design simple production reports. The SQL Report Designer is integrated in the Visual Studio .NET Report Server Project system, and is targeted at application developers to enable designing

production reports that require precision control over the report layout and programmability features.

SQL Server Reporting Services (SSRS) report definitions are not stored in the Microsoft Dynamics AX application metadata and there are no integrated developer tools in Microsoft Dynamics AX 4.0 to connect them to the Microsoft Dynamics AX Windows client or Enterprise Portal. Developers can implement custom code and use the SQL Reporting Services Report Viewer Controls and Web Parts to integrate production reports designed by using the SQL Report Designer and/or SQL Report Builder in Microsoft Dynamics AX.

Building a production report using the SQL Report Builder resembles using SQL Report Builder to build an Ad Hoc report as described in the section on [Ad Hoc Reporting in Microsoft Dynamics AX](#).

The SQL Report Designer is a tool that is integrated in the Visual Studio .NET Report Server Project system, and is targeted at application developers to enable designing production reports that require precision control over the report layout and programmability features. The Visual Studio .NET Report Server Project system includes features that enable developers to do the following:

- Define and configure report data sources to access data from any data source for which there exists a SQL Server Reporting Services data extension.
- Design reports using the SQL Report Designer.
- Publish reports and report data sources to a configured SQL Report Server.

Figure 15 illustrates the Microsoft Dynamics AX – SQL Report Designer integration options in Microsoft Dynamics AX 4.0 to build and deploy secure production reports using the SQL Report Designer.

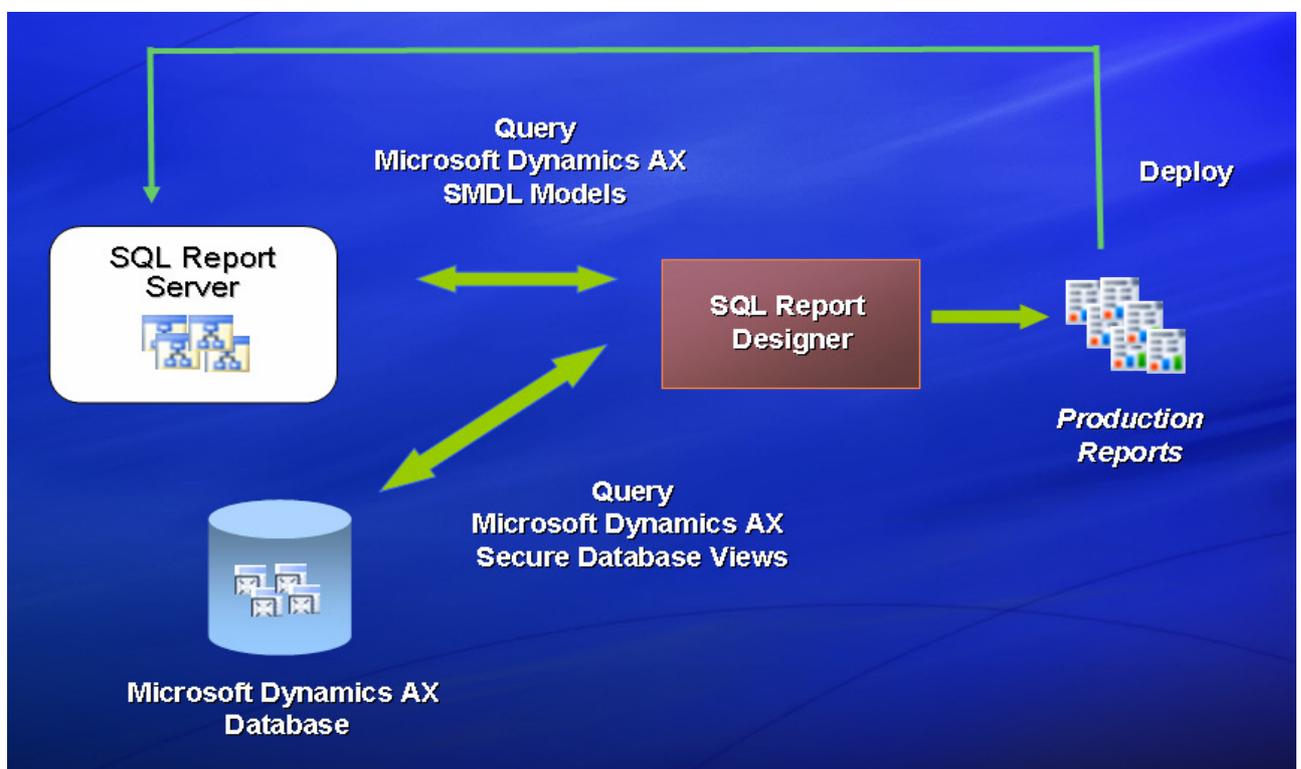


Figure 15: Microsoft Dynamics AX - SQL Report Designer Integration

Building secure production reports implies that users should only be able to view the data to which they have been granted access when they view the reports. Trying to enforce Microsoft Dynamics AX field and record level security in reports that access data directly from the Microsoft Dynamics AX database tables is a very complex task. SQL Server Reporting Services SMDL models can be referenced as report data sources when developers design reports using the SQL Report Designer. This implies that the SMDL models generated for Microsoft Dynamics AX Perspectives can also be leveraged to enable secure production report authoring using the SQL Report Designer. Reporting queries modeled against a SMDL model generated for a Microsoft Dynamics AX perspective are resolved by accessing the secure database views described in the section on Ad Hoc Reporting in Microsoft Dynamics AX. The secure database views automatically enforce field and record level security defined in Microsoft Dynamics AX to make sure that report users can only see data that they have been granted access to.

Figure 16 shows a report data source configuration to connect to the SMDL model generated for the Accounts Receivables Perspective were included with the base Microsoft Dynamics AX solution.

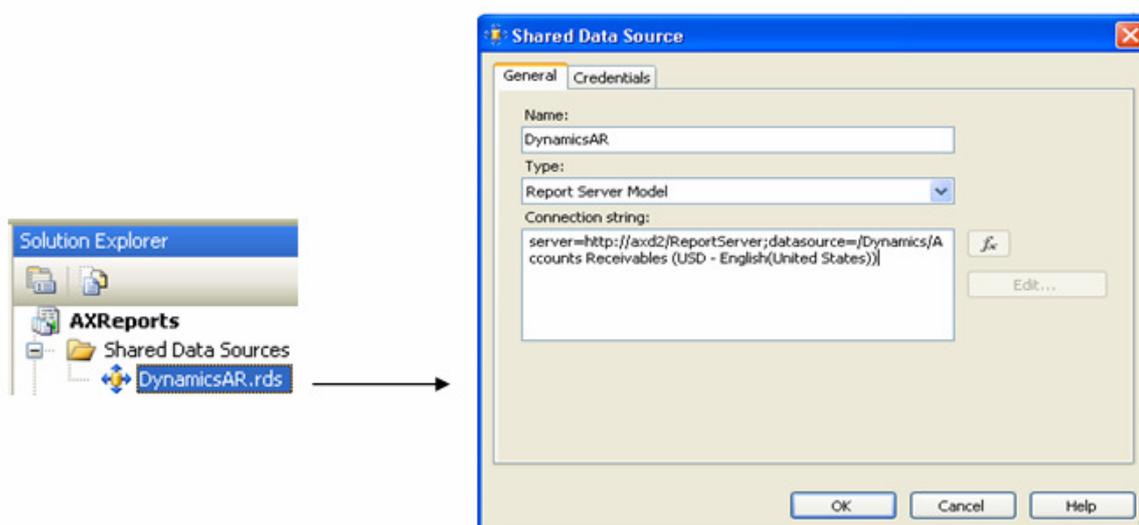


Figure 16: Configuring a Microsoft Dynamics AX SMDL Model report data source in a Visual Studio .NET Report Server project.

Developers can use the Dynamics AX SMDL model report data sources to design secure Microsoft Dynamics AX production reports in the SQL Report Designer. Figure 17 illustrates the steps to design and preview Microsoft Dynamics AX reports using the SQL Report Designer.

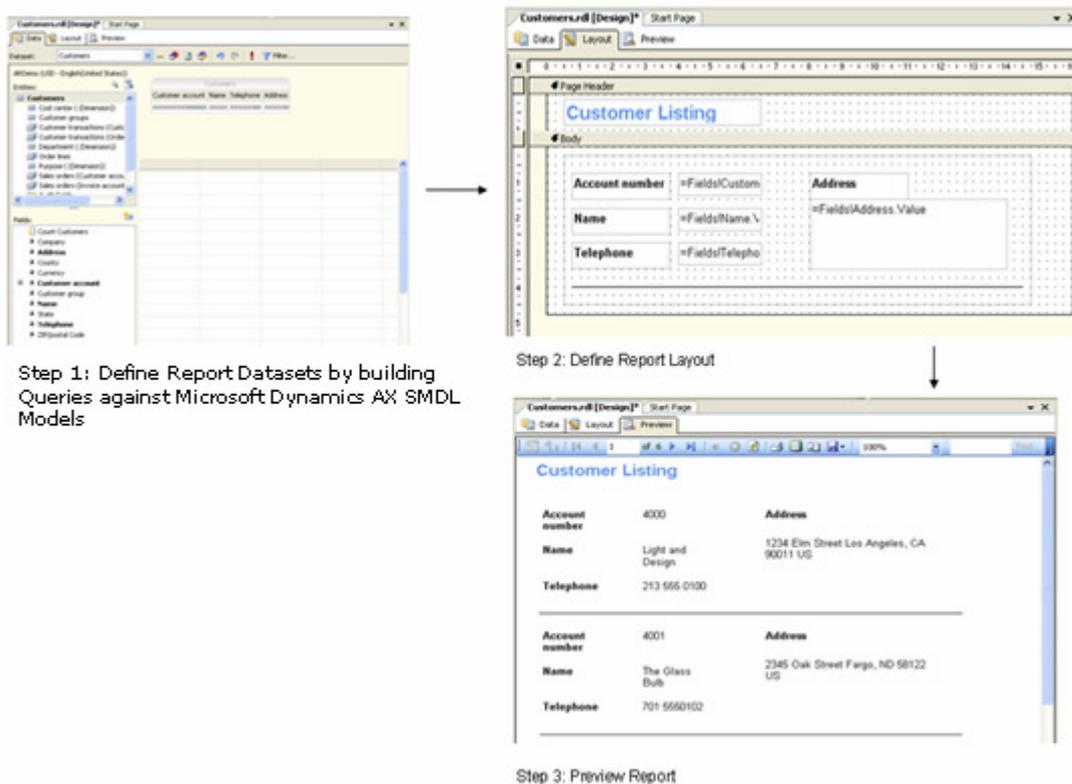


Figure 17: Designing reports in the SQL Report Designer

An alternative approach to using the SMDL models as a data source for production reporting, is to write SQL database queries that source data directly from the secure database views. This approach must be used when there is a requirement to write a more sophisticated query than the types of queries supported against SMDL models and/or when there are requirements to parameterize the database queries used to retrieve report datasets. Parameterization of database queries can be worked around by defining parameterized data region filters in the reports, but this approach will not scale for larger report datasets and can result in the retrieval of more data than required to render a report.

When writing a SQL query to use a secure database view, the developer can directly reference the view or use a table value function generated for the view. When directly referencing a secure database view, the SQL query will need to contain a WHERE predicate to specify a restriction on the WindowsID column to retrieve data that a context user has permissions to access. In a query that references multiple secure database views, one WHERE predicate will need to be added for each referenced view to ensure secure data retrieval.

A wrapper table value function (TVF) is generated for each secure database view to simplify view referencing in manually written SQL queries. The naming convention for TVFs generated for the secure database views is <TableName>USERFILTERED. For example, the TVF created for a secure database view generated for the CUSTTABLE is named CUSTTABLEUSERFILTERED. The TVF can be referenced as a regular table in a SQL query and takes a single parameter that is used to specify the windows user id of a user for whom data must be retrieved by resolving the corresponding Microsoft Dynamics AX security permissions. The following is a sample query that illustrates using the TVF generated for the secure database view created for the CUSTTABLE, to retrieve from the table that is accessible by the windows user DomainA\User1

```
select * from dbo.CUSTTABLEUSERFILTERED ('DomainA\User1')
```

Secure database views and their wrapper TVFs are by default generated or tables included in Microsoft Dynamics AX Perspectives or which Report Builder SMDL models are generated from within Microsoft Dynamics AX. The **Create all Secure Views** form in the Administration module (Administration -> Periodic -> Business Analysis -> Create all Secure Views) can be used to generate secure database views and wrapper TVFs for all tables in the Microsoft Dynamics AX database.

Additional developer effort is required to integrate Microsoft Dynamics AX SSRS reports in the Microsoft Dynamics AX Windows client and Microsoft Dynamics AX Enterprise Portal sites.

The simplest way to integrate SSRS reports in the Microsoft Dynamics AX Windows client is to execute WinAPI::shellExecute() method calls in menu item action implementations to open Internet Explorer and browse to a SQL Report Server URL that references a published report. Values for input report parameters gathered from application forms can also be supplied in the Report Server URL used to access a report. A more advanced solution would be to use a Web Browser control or the Visual Studio.Net 2005 SQL Report Viewer Windows Forms control to implement and integrate a custom Microsoft Dynamics AX Windows client SSRS Report Viewer form that can be invoked from Microsoft Dynamics AX menu items and menu item buttons.

Integrating SSRS Reports in Microsoft® Windows® SharePoint® Services (WSS) Microsoft Dynamics AX Enterprise Portal sites can be easily accomplished by using the SSRS Report Viewer Web part. The SQL Server Reporting Services product documentation contains information on installing, configuring, and using the Report Viewer Web part to integrate SSRS Reports in SharePoint sites.

Figure 18 illustrates using the SSRS Report Viewer Web part to integrate a Dynamics AX SSRS Report in a Microsoft Dynamics AX Enterprise Portal site.

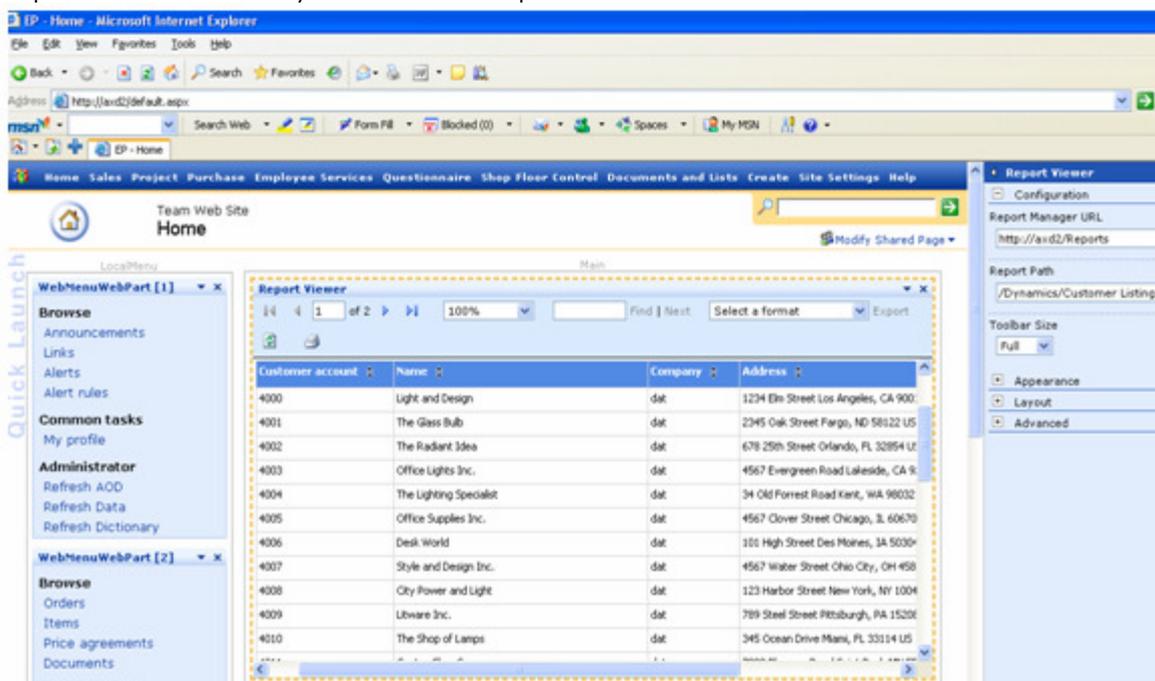


Figure 18: Using the SSRS Report Viewer Web part to integrate Microsoft Dynamics AX SSRS Reports in the Enterprise Portal

Features for the Microsoft Dynamics AX Application User

Application users can access production reports in the Windows client through application menus and action buttons on forms to which they are connected. They can specify values for input report parameters, configure rendering options, and execute reports to view and analyze data.

Figures 19 and 20 show production reports (implemented by using the native report writer and SQL Report Designer) being accessed from application menus and forms in the Microsoft Dynamics AX Windows client.

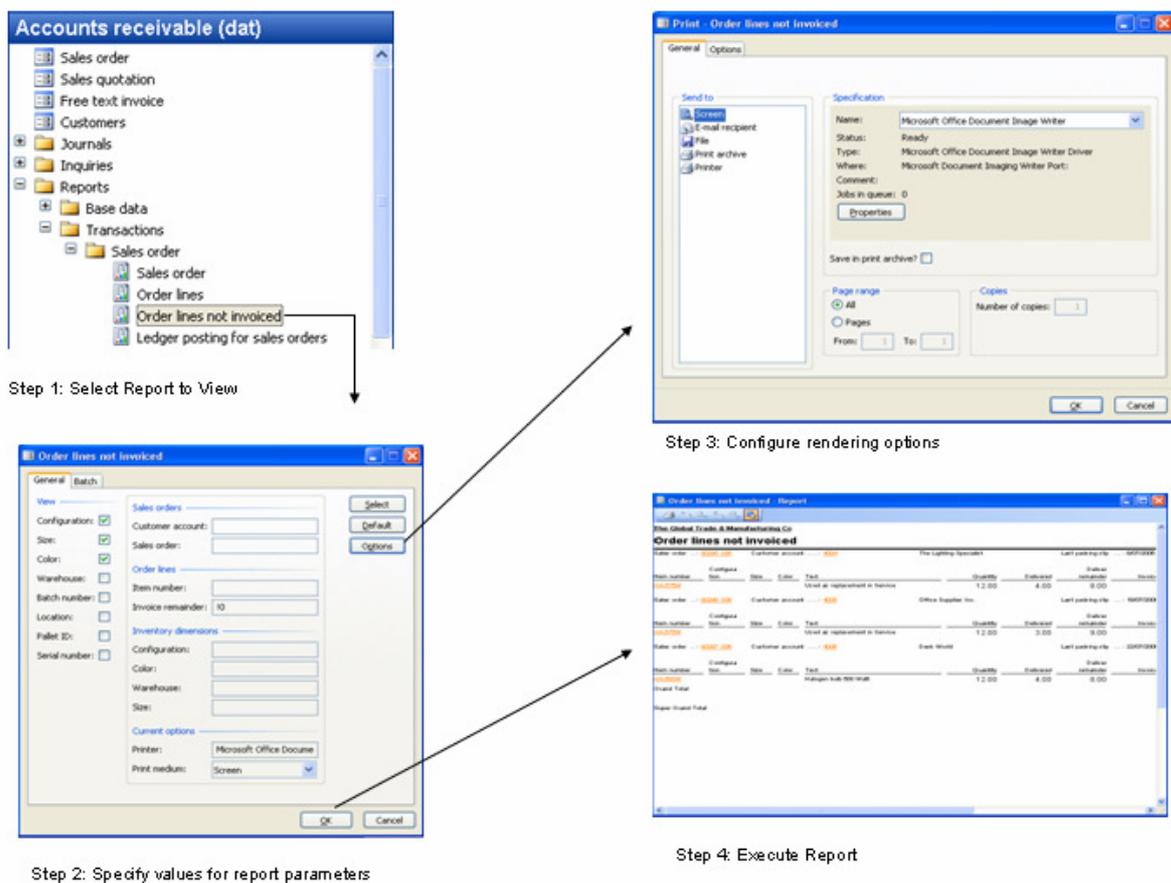


Figure 19: Accessing a Production Report in the Microsoft Dynamics AX Windows Client

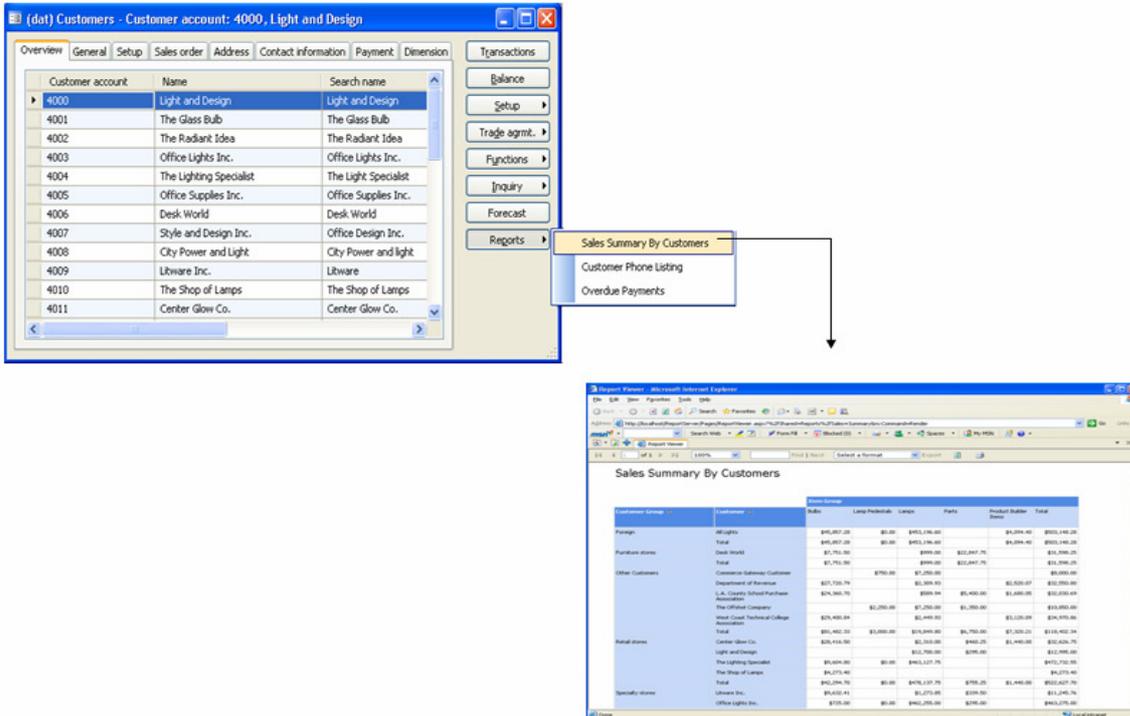


Figure 20: Accessing a SSRS Report from the Microsoft Dynamics AX Windows Client

Application users can also view production reports (reports implemented by using the native report writer and SQL Server Reporting Services) rendered on Microsoft Dynamics AX Enterprise Portal pages and can be accessed through Enterprise Portal menus. Figure 21 illustrates two SSRS chart reports rendered in a Microsoft Dynamics AX Enterprise Portal site using the SSRS Report Viewer Web part.

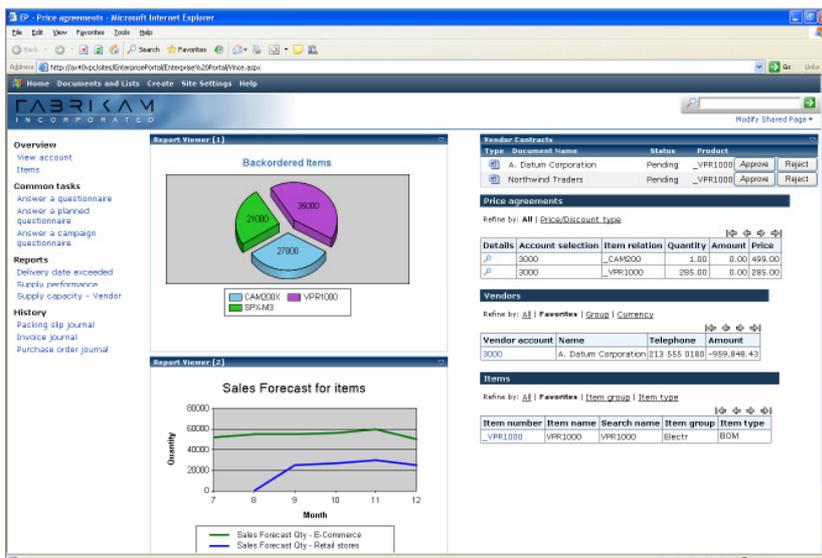


Figure 21: Viewing Microsoft Dynamics AX SSRS Reports in the Microsoft Dynamics AX Enterprise Portal.

Action links can also be configured in Microsoft Dynamics AX EP pages to enable application user customization of reports authored using the SQL Report Builder.

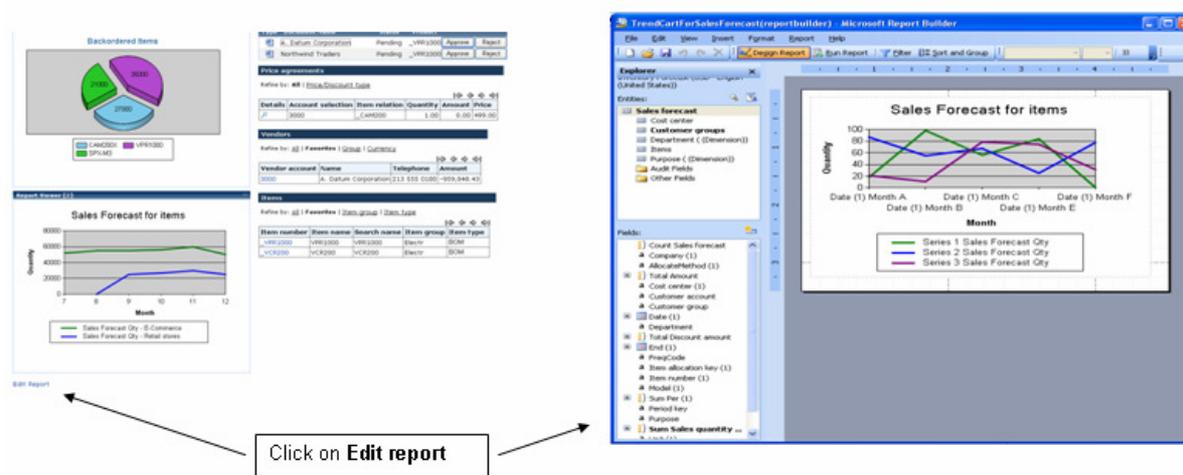


Figure 22: Action Link to enable the user to start the SQL Report Builder from the Microsoft Dynamics AX Enterprise Portal to customize a report

Features for the Microsoft Dynamics AX Administrator

Microsoft Dynamics AX Administrators can use the **User Group Permissions** form to set up user rights to production reports by denying or granting access to menu items wired to the reports.

Administrators can also use the SQL Report Manager and/or the SQL Management Studio to configure Report Server security to deny or grant access to published Microsoft Dynamics AX SSRS Reports.

Future Directions

The future direction for Production Reporting in Microsoft Dynamics AX is to enable a model driven and metadata integrated production reporting framework that integrates tightly with SQL Server Reporting Services and the SQL Report Designer.

The Microsoft Dynamics AX – Reporting Services integration framework will enable users to open and use the SQL Report Designer from the Microsoft Dynamics AX tools to design production reports. The SSRS Report definitions will be stored in the application metadata and can be customized across multiple layers like all other Microsoft Dynamics AX objects. The framework will also include many services that wrap and add value over the base SSRS platform and tools to enable the following features:

- Sourcing report data from Microsoft Dynamics AX Queries and Business logic.
- Auto Report Generation
- Report Style Templates
- Report Localization
- Upgradeable Report Customizations
- Built in tools and runtime support for integrating SSRS reports in the Microsoft Dynamics AX Windows Client and Enterprise portal.

Best Practice Recommendation:

Adopt and use the following principles of declarative report design when users build production reports:

1. Report data retrieval must be cleanly separated from report rendering.
2. Avoid executing imperative code that triggers database calls to retrieve and/or update data during report rendering.
3. Limit the use of imperative code executed during report rendering to perform in-memory transformations and/or calculations using data values in the result sets returned by report data source queries and values supplied by users for ranges/report parameters.
4. Avoid writing imperative code that interfaces directly with the report rendering engine.

These principles are widely acknowledged in the industry today as best practices for designing reports.

The Microsoft Dynamics AX native report writer does not enforce these principles and enables a great degree of freedom and flexibility in the use of imperative code. This freedom and flexibility when it isn't used optimally will result in functional, but not optimal, report designs.

Designing reports in SQL Server Reporting Services is based on the declarative design principles. Unlike as in the native report writer, users will be unable to write imperative code that interfaces directly with the report rendering engine. The SSRS report expression language can be used to define powerful declarative expressions to configure value settings for most of the report design properties. These expressions can include call outs to imperative code methods whose recommended usage is for scenarios where in-memory data value transformations and/or calculations cannot be expressed declaratively.

Adopting the principles of declarative report design when users design reports by using the native Microsoft Dynamics AX report writer will simplify the future migration of reports to the SSRS integrated production reporting framework.

Multidimensional Reporting and Analytics in Microsoft Dynamics AX

Note: The Microsoft Dynamics AX Business Analysis component is included in the Advanced Management edition of Business Ready Licensing.

Multidimensional Reporting and Analytics features enable users to efficiently aggregate and analyze business metrics across core business dimensions to gain a deeper business insight into business trends and drivers.

A highly scalable Analytics data platform and rich data visualization features that can surface rolled up aggregate views of business metrics with built in support to drill up, down, and through to related detail data are cornerstone requirements to realize the benefits of Multidimensional Reporting and Analytics.

Microsoft SQL Server 2005 and Microsoft Office offer industry leading server and client BI platforms that can be leveraged to enable compelling Multidimensional Reporting and Analytics capabilities in business solutions. The Microsoft Dynamics AX product team is committed to fully using these platform technologies to enable rich BI and Analytics scenarios in Microsoft Dynamics AX.

Features Available in the Product Today

Features for Microsoft Dynamics AX Application Users

The Application user OLAP reporting capabilities of the Microsoft Dynamics AX Business Analysis module enable integrated application UIs for performing multidimensional analysis without having to leave the familiar Microsoft Dynamics AX environment.

Predefined multidimensional analysis views are integrated into Microsoft Dynamics AX application modules such as the General Ledger and Inventory/Warehouse Management. Business users can use these views to analyze related business metrics by a variety of business dimensions. For example, a business user can compare account activity for different departments, compare inventory levels for various item colors, and compare warehouse picking efficiency by operator.

Each analysis view is displayed as a Pivot table within the Microsoft Dynamics AX client. General widespread user familiarity with the Excel Pivot Table makes it a powerful business user tool for OLAP Reporting and Analysis. Users can drag-and-drop business metrics and dimension attributes onto the Pivot Table/Chart canvas to analyze data and instantaneously view on-screen results.

		Year 2005		Year 2006		Grand Total	
Customer Group	Customer	Sales Balance	Customer Revenue	Sales Balance	Customer Revenue	Sales Balance	Customer Revenue
Foreign	4015	\$320,960.36	\$320,960.36	\$123,592.12	\$123,592.12	\$444,552.48	\$444,552.48
	Total	\$320,960.36	\$320,960.36	\$123,592.12	\$123,592.12	\$444,552.48	\$444,552.48
Furniture stores	4006		\$1,440.00	\$1,440.00	\$1,440.00	\$1,440.00	\$1,440.00
	Total		\$1,440.00	\$1,440.00	\$1,440.00	\$1,440.00	\$1,440.00
Other Customers	4017		(\$1,440.00)	(\$1,440.00)	(\$1,440.00)	(\$1,440.00)	(\$1,440.00)
	Total		(\$1,440.00)	(\$1,440.00)	(\$1,440.00)	(\$1,440.00)	(\$1,440.00)
Retail stores	4004	\$343,739.75	\$343,424.65	\$117,936.00	\$117,936.00	\$461,675.75	\$461,360.65
	Total	\$343,739.75	\$343,424.65	\$117,936.00	\$117,936.00	\$461,675.75	\$461,360.65
Specialty stores	4003	\$350,594.75	\$350,594.75	\$108,697.75	\$108,697.75	\$459,292.50	\$459,292.50
	4005	\$346,564.00	\$346,564.00	\$113,123.00	\$113,123.00	\$459,687.00	\$459,687.00
	4022		\$81,200.00	\$81,200.00	\$81,200.00	\$81,200.00	\$81,200.00
	Total	\$697,158.75	\$697,158.75	\$303,020.75	\$303,020.75	\$1,000,179.50	\$1,000,179.50
Grand Total		\$1,361,858.86	\$1,361,543.76	\$544,548.87	\$544,548.87	\$1,906,407.73	\$1,906,092.63

Figure 23: Interactive pivot tables in the Microsoft Dynamics AX Client

SQL Server Analysis Services Cubes serve as the backend data sources for Analysis Views canned into Microsoft Dynamics AX. Users can also connect to these cubes using other popular BI client tools such as Excel and Proclarity Desktop Professional that offer richer Ad Hoc reporting and analysis capabilities.

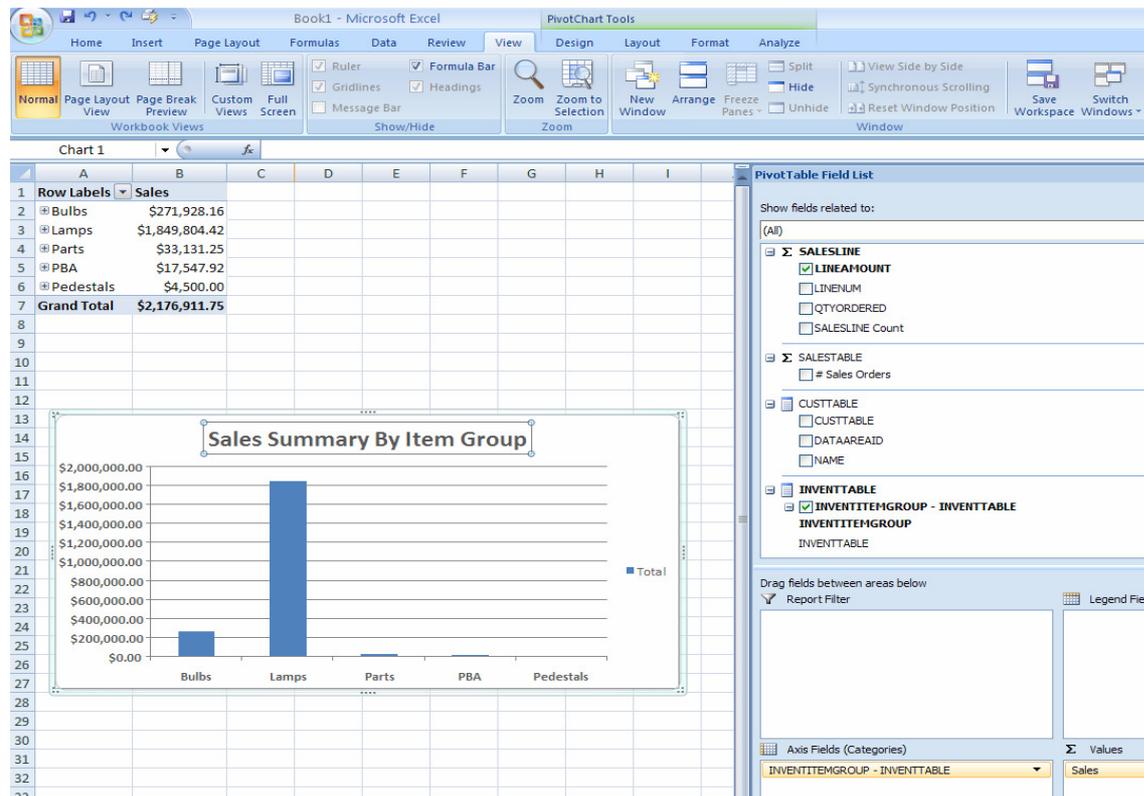


Figure 24: Exploring Microsoft Dynamics AX cubes in Microsoft® Office Excel®

Features for Microsoft Dynamics AX Application Developers

Figure 25 illustrates the high level architectural model for integrating Microsoft Dynamics AX with SQL Server Analysis Services, and the scenarios enabled by the integration.

Microsoft Dynamics AX – SSAS Integration

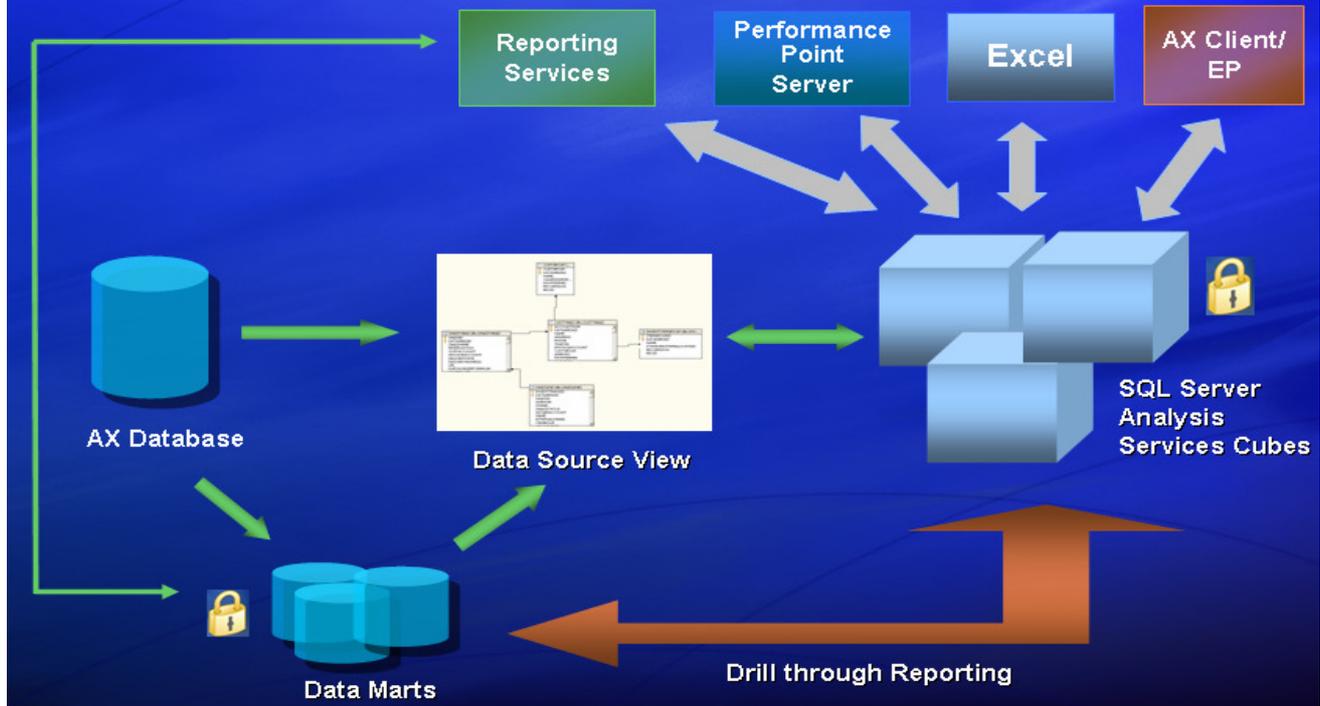


Figure 25: Integrating Microsoft Dynamics AX with SQL Server Analysis Services 2005

Application developers can use one of the following approaches to build SQL Server Analysis Services cubes for Microsoft Dynamics AX:

- Use the Microsoft Dynamics AX Cube Definition Manager to define and deploy SSAS cubes. Predefined Analysis views in Microsoft Dynamics AX solutions are connected to cubes defined and deployed using the Cube Definition Manager.

The Cube Definition Manager has been enhanced in Microsoft Dynamics AX 4.0 to support deploying cubes to SSAS 2005. Its core functionality relative to defining cubes has not been enhanced to take advantage of any of the new SSAS 2005 cube design features.

Cubes designed and deployed using the Cube Definition Manager must be customized using the SQL Server 2005 BI Development Studio to take advantage of the new features in SQL Server Analysis Services 2005.

It should also be noted that cubes deployed using the Cube Definition Manager are not secured by default and that Microsoft Dynamics AX Administrators are responsible for securing them using the SQL Management Studio.

The cube definitions are not stored in the Microsoft Dynamics AX application metadata implying that they cannot be customized in a layered manner and that there are no stored metadata relationships linking them to the Microsoft Dynamics AX relational data model.

Data Mart generation and drill down reporting are also not enabled when you use the Cube Definition Manager to define and deploy Microsoft Dynamics AX cubes.

Approximately 12 cube definitions are shipped in the Microsoft Dynamics AX Business Analysis module. These cube definitions can be imported into the Cube Definition Manager and deployed to SQL Server Analysis Services Databases. They also support the predefined Analysis Views in the base package.

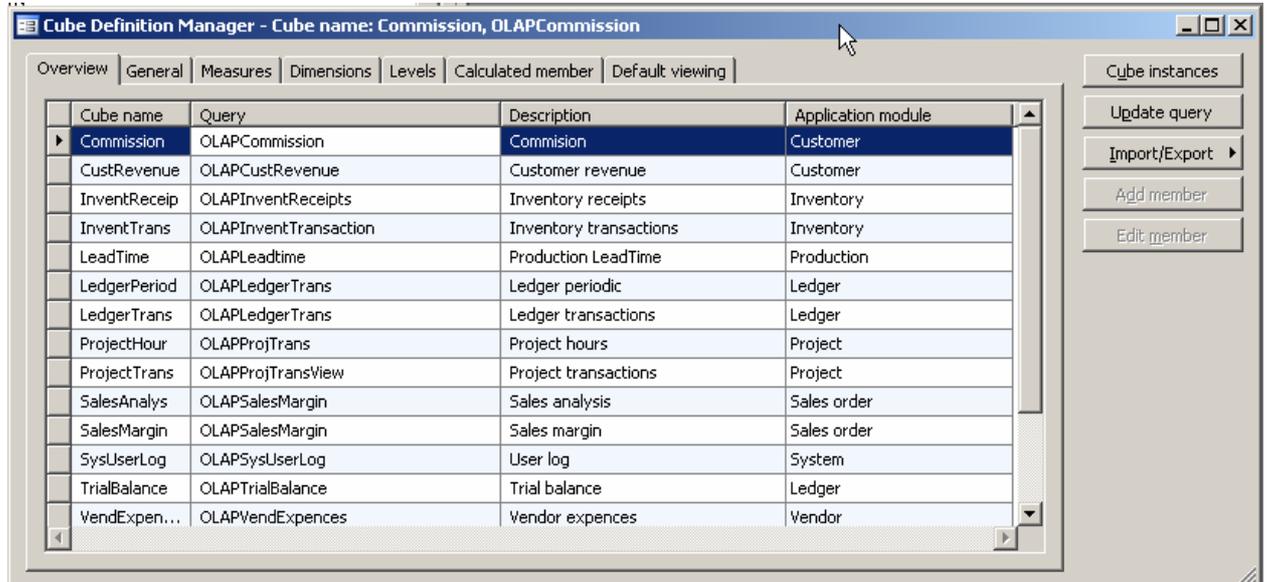


Figure 26: The Microsoft Dynamics AX Cube Definition Manager

- Use the SQL Server 2005 BI Development Studio to design and deploy Microsoft Dynamics AX Data Source Views and Cubes. This approach requires developer expertise in designing SSAS 2005 cubes. However, it is currently the only way to take advantage of the many new cube design features and build enterprise scale BI infrastructures that include data marts and warehouses, and support drill down reporting. The [Microsoft SQL Server Analysis Services](#) product home page contains links to many resources that can be used to become familiar with using the developer features to design production quality OLAP cubes.

Features for Microsoft Dynamics AX Administrators

Microsoft Dynamics AX Administrators can use the **OLAP Servers** Administration form to register SQL Server Analysis Servers to which cubes defined by using the Cube Definition Manager can be deployed.

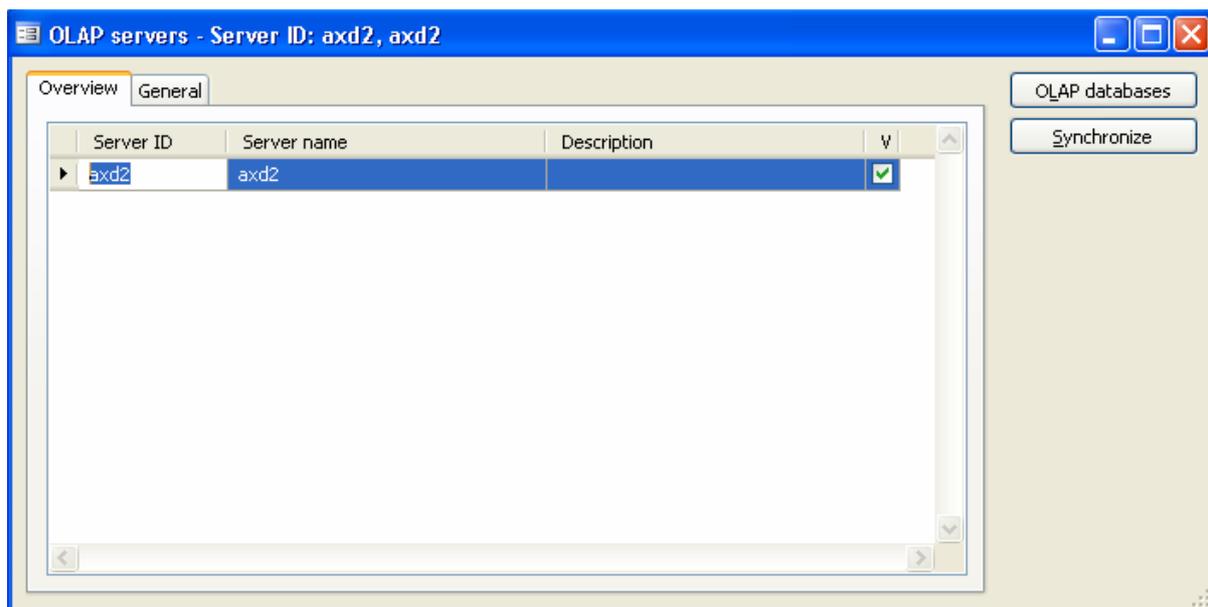


Figure 27: The OLAP Servers Administration form

Administrators are also responsible for securing the Microsoft Dynamics AX cubes deployed using the Cube Definition Manager to support Analysis Views in client and Ad Hoc Reporting using BI client tools such as Microsoft Office Excel and the Proclarity Analytics client tools. This requires knowledge of the SSAS security model and familiarity with using the SQL Server Management Studio to help secure SSAS cubes.

Future Directions

The plans for Multidimensional Reporting and Analysis include providing richer shipped content and enabling a tighter model driven integration with SSAS. The following are the core enhancements planned for this area in future releases of Microsoft Dynamics AX:

- Role-based BI content shipped with the product (Dashboards, Reports, and rich analytical Views integrated in the Windows Client and Enterprise Portal).
- Multi dimensional databases and cubes that enable the BI content shipped with the product. This can be leveraged for Ad Hoc Reporting and Analysis.
- Model driven integration with SSAS to generate localized and secure Cubes that reflect Microsoft Dynamics AX application semantics.
- Store Multi dimensional data models (Cube and Dimension definitions) in the application metadata to enable a tighter integration at the metadata level with related Microsoft Dynamics AX objects such as Tables, Views, Queries, Reports, and Forms.
- Two way tooling integration with the SQL Server BI Development Studio to enable advanced cube design and customization scenarios.
- SQL Report Builder Models based on OLAP Cubes.
- Improved Excel integrated OLAP reporting capabilities

- Generation and management of denormalized relational data marts and data warehouses to enable consolidation of data from multiple Microsoft Dynamics AX deployments, analysis of historical data, and high performance Analytical Views to Detail data drill down reporting.

Best Practice Recommendation:

Embrace SQL Server Analysis Services. OLAP cubes are the cornerstone of Business Intelligence solutions. Current and future Microsoft BI solutions work best with and prescribe the usage of OLAP databases as the data platform of choice. [Microsoft SQL Server Analysis Services](#) (SSAS) is a multidimensional database platform that consists of a variety of services and tools to support the implementation, deployment, and maintenance of highly scalable BI infrastructures and solutions. The future directions for SSAS integration in Microsoft Dynamics AX will enable a low TCO model driven integration to enable richer Analytics scenarios with minimal development and administration efforts. We strongly recommend that you start implementing multidimensional SSAS OLAP databases for your Microsoft Dynamics AX deployments as soon as possible by using the integration capabilities that are available in Microsoft Dynamics AX today and the native SSAS tools. Investing in this technology and developing related skills will definitely reap significant benefits that contribute to competitive business gains.

Business Scorecards in Microsoft Dynamics AX

KPI (Key Performance Indicator) Scorecards and Dashboard views with drill down capabilities are fast becoming popular business performance management tools to monitor the overall health of an organization and analyze business trends and drivers. This section provides an overview of the options that can be used to implement, deploy, and manage business scorecards in Microsoft Dynamics AX solutions.

Features Available in the Product Today

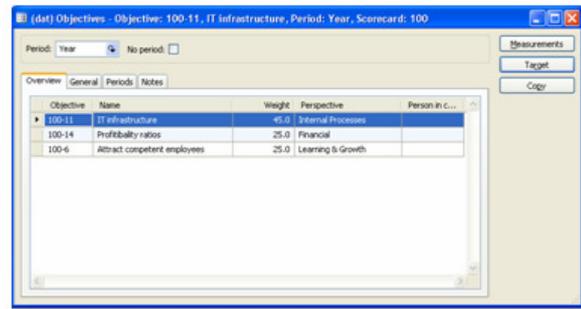
The Balanced Scorecard Module

The Microsoft Dynamics AX Balanced Scorecard module can be used to configure and view scorecards that reflect the overall health of an organization. Users can check the state of the company using customized dashboard views created by using built-in metric monitoring graphics included with the module.

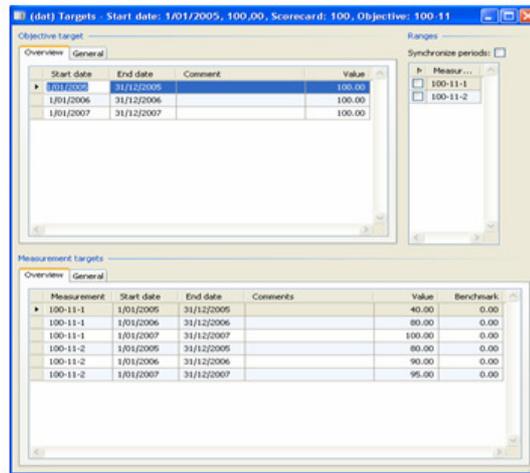
Figures 28 and 29 illustrate configuring scorecards and viewing dashboard views of scorecards in the Microsoft Dynamics AX Balanced Scorecard module.



Step 1: Define Scorecard



Step 2: Define Scorecard Objectives



Step 3: Define Targets for Objectives

Figure 28: Configuring Scorecards in the Balance Scorecard module

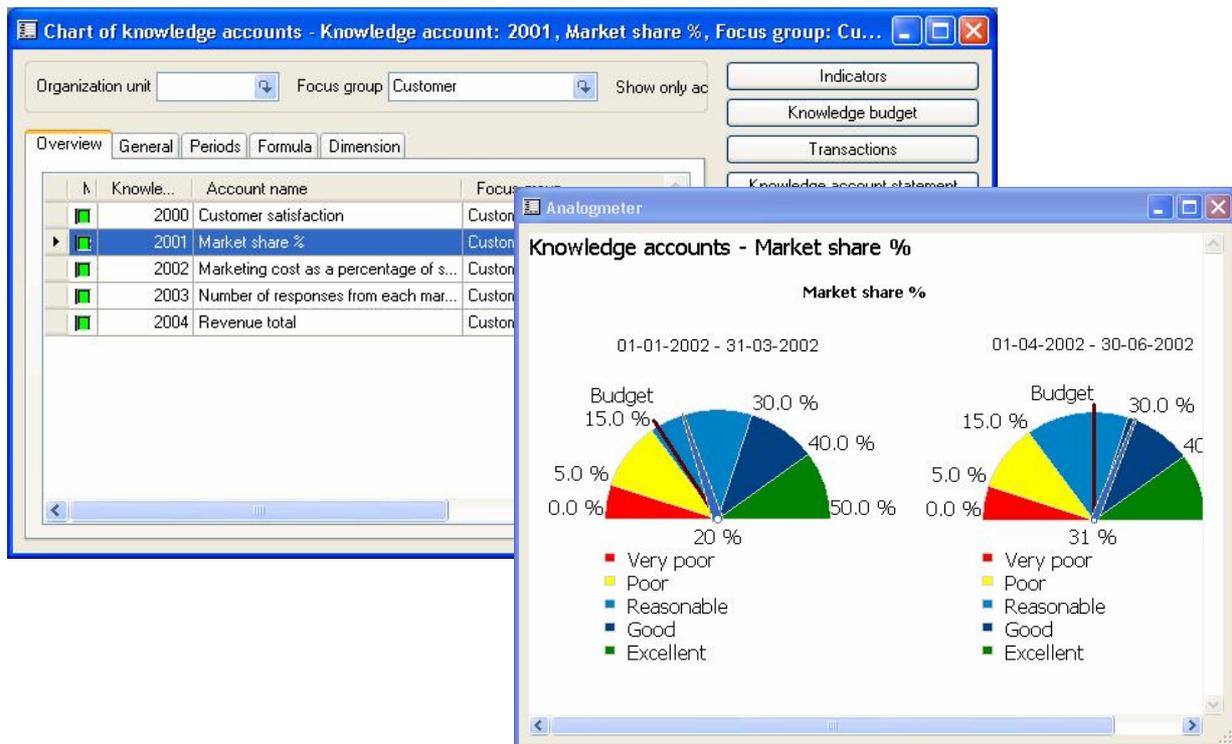


Figure 29: Dashboard view in the Balanced Scorecard module

Custom dashboard development using Microsoft Office Business Scorecard Manager 2005

[Microsoft Office Business Scorecard Manager 2005](#) (now an integral component of the recently announced [Microsoft Office Performance Point Server 2007](#)) is a comprehensive scorecarding platform that consists of services and tools to support the implementation, deployment, and maintenance of personalized KPI (Key performance indicator) dashboards that enable deep contextual insight into business health, trends and drivers.

Microsoft Dynamics AX SQL Server Analysis Services Cubes (custom built or deployed using the Microsoft Dynamics AX Cube Definition Manager) can be used as data sources for KPI scorecards implemented, deployed, and maintained using Microsoft Office Business Scorecard Manager. Supporting report views in the form of SQL Server Reporting Services reports, Excel Pivot Tables, and Excel Pivot Charts can be also be defined for KPIs (Key Performance Indicators) in an Office BSM model

Dashboard views of Microsoft Office Business Scorecard Manager KPI scorecards and supporting report views can be integrated in Microsoft Dynamics AX Enterprise Portal sites using the Office BSM dashboard viewer, the SQL Server Reporting Services Report Viewer, and the Excel Pivot Table & Chart Web parts.

Figure 30 shows a dashboard view of an Office BSM scorecard rendered in a Microsoft Dynamics AX Enterprise Portal site.

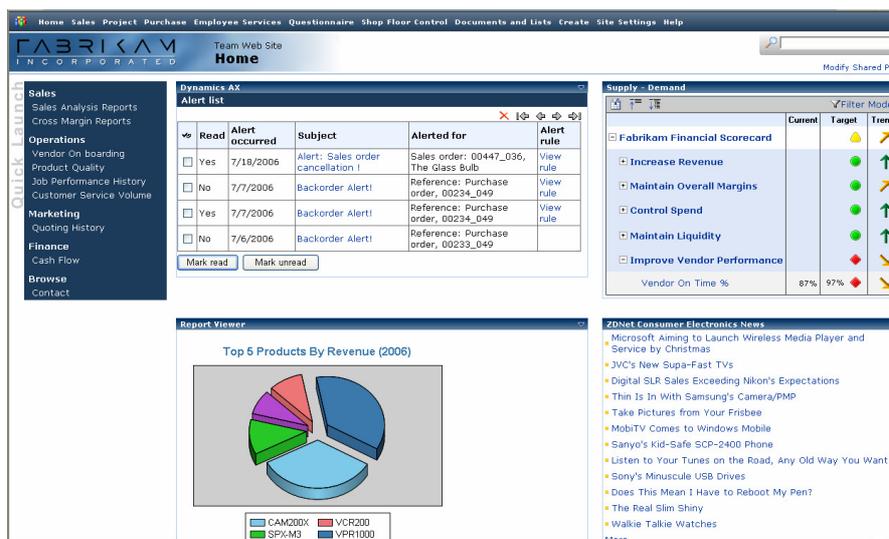


Figure 30: An Office Business Scorecard Manager 2005 dashboard rendered in a Microsoft Dynamics AX Enterprise Portal

Custom dashboard development using SQL Server Analysis Services 2005 and SQL Server Reporting Services 2005

Native support for defining and managing KPIs in cubes is a new feature in SQL Server Analysis Services 2005. The MDX query language has also been enhanced to support the retrieval of KPI values, targets, and trends.

SQL Server Reporting Services 2005 also introduces new and improved tooling support for designing reports that source data from SQL Server Analysis Services 2005 cubes.

These new native features in SSAS and SSRS can be leveraged to implement and deploy basic functional KPI dashboards and supporting report views in mid to low end Microsoft Dynamics AX deployments where the cost of deploying and maintaining an Office Business Scorecard Server can be prohibitive.

Feature rich and affordable third-party data visualization components such as those provided by [Dundas](#) can also be used to enhance the appearance and presentation of custom built dashboards.

Future Directions

The following Scorecarding and Dashboards plans are included in future releases of Microsoft Dynamics AX:

- Role based BI dashboards that consist of scorecards, reports, and rich analytical views shipped as part of the product.
- Tighter integration with the Scorecarding capabilities of [Microsoft Office PerformancePoint Server 2007](#).

Best Practice Recommendation:

Embrace SQL Server Analysis Services. OLAP cubes are the cornerstone of Business Performance Management Scorecards. Current and future Microsoft Business Performance Management and Analytics platforms/solutions work best with and prescribe the usage of OLAP databases as the data platform of choice. [Microsoft SQL Server Analysis Services](#) (SSAS) is a multidimensional database platform that consists of a variety of services and tools to support the implementation, deployment, and maintenance of highly scalable BI infrastructures and solutions. The future directions for SSAS integration in Microsoft Dynamics AX (see the section on Multidimensional Reporting and Analytics) will enable a low TCO model driven integration to enable richer Analytics scenarios with minimal development and administration efforts. However, we strongly recommend that you start implementing multidimensional SSAS OLAP databases for your Microsoft Dynamics AX deployments as soon as possible by using the integration capabilities that are available in Microsoft Dynamics AX today and the native SSAS tools. Investing in this technology and developing related skills will definitely reap significant benefits that contribute to competitive business gains.

Financial Reporting in Microsoft Dynamics AX

Financial accountants and analysts are a distinct category of users from the perspective of their requirements for reporting tools and features. They are not application developers, nor are they regular business users. They are unique users who are very proficient in the domain of financial analysis and reporting. Their primary requirements for reporting tools and features include the following:

- An abstracted view of the application data model presented as familiar financial concepts.
- Powerful dynamic report adjustment capabilities.
- Report layout templates for common financial reports.
- Powerful expression language, formula libraries, and formula engines to support complex financial calculations.

This section provides an overview of the features available today to support financial reporting in Microsoft Dynamics AX and the planned future directions.

Features Available in the Product Today

The Microsoft Dynamics AX native report writer and the SQL Report Designer can be used to implement and integrate static predefined financial reports in Microsoft Dynamics AX. Using these tools to design even the simplest of financial reports requires a deep subject matter understanding of the physical database tables and relationships that comprise the Microsoft Dynamics AX Financials data model. More significantly, neither tool is designed to address the functional requirements of a Financial Accountant/Analyst listed at the start of this section.

[Microsoft FRx](#) is a powerful and popular solution for authoring and viewing boardroom-quality, period-end, and year-end financial reports.

Using FRx, users can easily create every kind of financial report, from simple income statements and balance sheets to cash flow reports and multi-company drill down reports. The FRx user interface is based on concepts familiar to financial accountants/analysts, and using the tool does not require deep technical expertise or knowledge of the underlying Microsoft Dynamics AX database schema.

FRx is available for Microsoft Dynamics AX. It is specifically ideal for financial consolidation in companies that have many subsidiaries, especially those that might still be in the process of rolling out Microsoft Dynamics AX to their subsidiaries and are still running several different business applications.

Additionally, users who need a managed budgeting solution that works with Microsoft Dynamics AX (and many other business applications) can use FRx Forecaster.

Figure 31 illustrates the Microsoft Dynamics AX – Microsoft FRx integration. The following article contains more detailed information about the product integration:

[Microsoft FRx and Microsoft Dynamics AX working together](#)

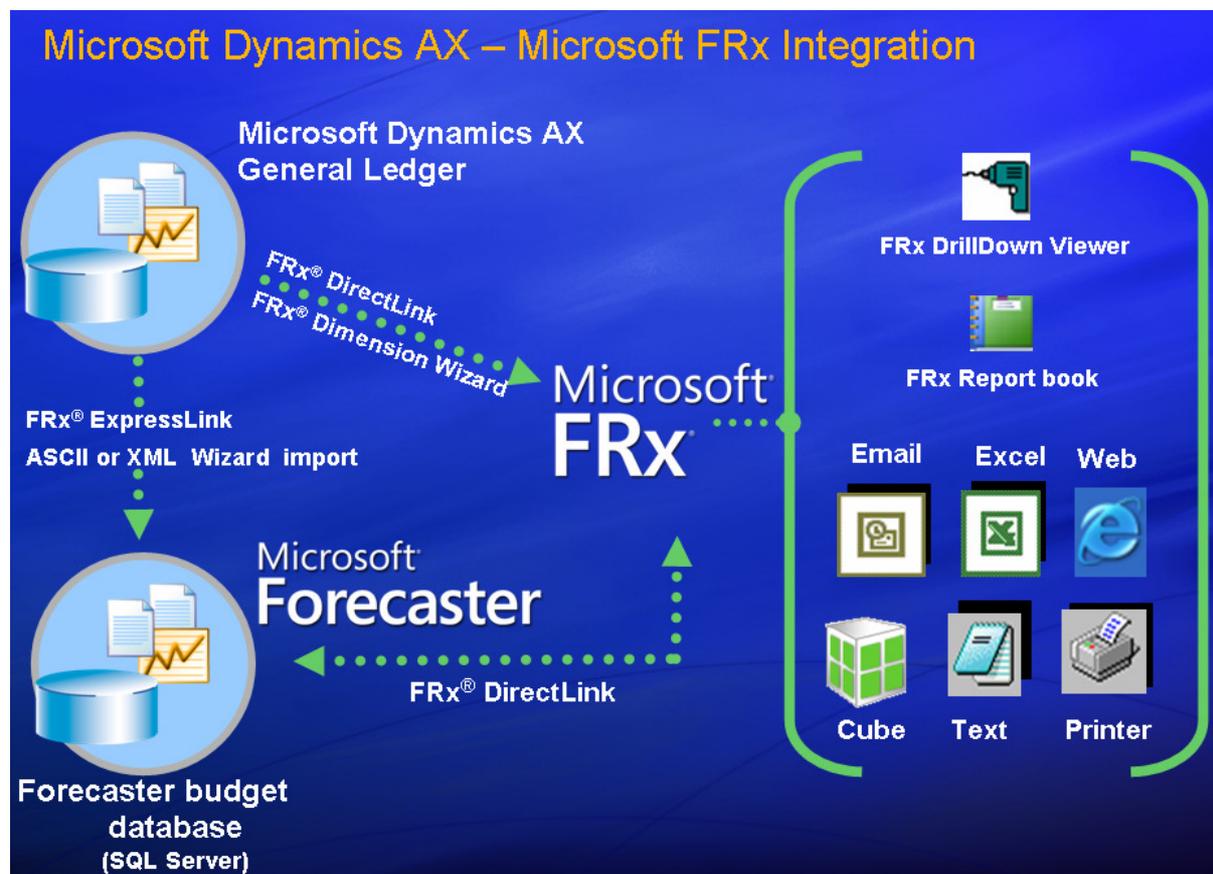


Figure 31: Microsoft Dynamics AX – Microsoft Forecaster integration

The following resources provide a more detailed overview of Microsoft FRx and Microsoft Forecaster:

[Overview of Microsoft FRx](#)
[Microsoft Forecaster](#)

Future Directions

The following Financial Reporting plans are included in future releases of Microsoft Dynamics AX:

- Continued alignment with the future directions Microsoft FRx and Microsoft Forecaster.
- Tight integration with [Microsoft Office PerformancePoint Server 2007](#) to enable rich Planning, Budgeting, and Forecasting scenarios

Conclusion

We hope you found this paper to be useful and that it provided a broad overview of the current capabilities and future directions planned for Reporting and BI in Microsoft Dynamics AX.

The Microsoft Dynamics AX Reporting and BI framework plays a significant role in enabling a wide variety of Business Performance Management scenarios in Microsoft Dynamics AX. Microsoft is fully committed to continually enhancing the framework to support and enable richer scenarios over time.

A key takeaway, relative to future directions for the framework, is the focus and commitment to integrate tightly with the Microsoft BI Platform Stack to continually take advantage of related technology advancements and enable richer platform integrated Business Performance Management solutions.

<http://www.microsoft.com/dynamics/ax/>

Microsoft Dynamics is a line of integrated, adaptable business management solutions that enables you and your people to make business decisions with greater confidence. Microsoft Dynamics works like and with familiar Microsoft software, automating and streamlining financial, customer relationship and supply chain processes in a way that helps you drive business success.

U.S. and Canada Toll Free 1-888-477-7989

Worldwide +1-701-281-6500

www.microsoft.com/dynamics

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, this document 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.

© 2006 Microsoft Corporation. All rights reserved.

Microsoft, the Microsoft Dynamics Logo, BizTalk, FRx, Microsoft Dynamics, SharePoint, Visual Basic, Visual C++, Visual SourceSafe, Visual Studio, Microsoft Office, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation, FRx Software Corporation, or Microsoft Business Solutions ApS in the United States and/or other countries. Microsoft Business Solutions ApS and FRx Software Corporation are subsidiaries of Microsoft Corporation.

Copyright © 2006 ProClarity™, A Microsoft® Subsidiary. All rights reserved. ProClarity and the ProClarity logo are trademarks and registered trademarks of ProClarity Corporation in the United States and/or other countries. All other marks mentioned herein are trademarks or service marks of their respective owners.

