

# Suggested Practices

## Chapter 1: Installing SQL Server 2005 Business Intelligence Tools

### Installing SQL Server 2005 Business Intelligence Tools

- **Practice 1** Practice installing the complete SQL Server 2005 component set, then install second named instances of the Analysis Services and Reporting Services.
- **Practice 2** Install the latest SQL Server 2005 service pack on the default instance of your installation, and then remove the named instance you installed in the first practice.

## Chapter 2: Configuring SQL Server 2005 Business Intelligence Components

### Configuring SQL Server 2005 Business Intelligence Components

- **Practice 1** Practice using the command-line tools to back up and restore the encryption key on one instance. Then, install a second named instance of SSRS on the same machine and try to add the new instance to the existing instance, thereby creating a scale-out deployment.
- **Practice 2** Practice setting up SSAS server security and defining the way in which SSAS captures error and query logging.
- **Practice 3** To validate your knowledge of SSRS configuration tools, practice installing two instances of SSRS and using the command-line tools to configure them.

## Chapter 3: Migrating to SQL Server 2005 Business Intelligence Components

### Migrating to SQL Server 2005 Business Intelligence Components

- **Practice 1** Run the Analysis Services Migration Wizard against SSAS 2000 cubes and note the verification results that the Migration Wizard returns. Also, perform a full process of your new SSAS 2005 cubes through SSMS.
- **Practice 2** Run the DTS Migration Wizard against DTS packages, and then open the migrated packages in BIDS to observe the results.

## Chapter 4: Developing SSIS Packages

### Creating SSIS Packages to Import and Transform Data

- **Practice 1** Create a new SSIS project that has as a data source a development database. Create a new package within your project to move an Excel file from a network share to a folder, and then import the Excel spreadsheet data. During the import process in the data flow, convert character data types to numeric data types if necessary, and then summarize the data by using an Aggregate transformation, with a SQL Server table as the destination.
- **Practice 2** Create a data flow in an SSIS package that contains a source connection to any file. Connect the source file to a Multicast transformation, and then, for every transformation in the Data Flow toolbox, drag the component to the data flow and connect a multicast output to the transformation. Edit each transformation, and understand what each transformation performs.

## Chapter 5: Debugging and Error Handling with SSIS Packages

### Debugging and Error Handling with SSIS

- **Practice 1** Force a package you created in one of the practices to fail by setting the `ForceExecutionResult` to `Failure`. Practice identifying the error in the Execution Results window and capturing the error description in an `OnTaskFailed` event handler.
- **Practice 2** Create a new package and data flow that extracts the `Person.Address` table from the *AdventureWorks* database. Use a Data Convert transformation to try to convert the `PostalCode` column to an integer data type. Route the successful rows to a `SuccessfulConvert.txt` file, and then add an error path to send the unsuccessful conversions to an `Unsuccessful.txt` file. Add a data viewer on the error path, and then run the package.

## Chapter 6: Developing SSAS Cubes

### Creating a Cube and Dimensions

- **Practice 1** Create a new SSAS project that includes a Data Source and Data Source View that reflect a dimensionally modeled database; the *AdventureWorksDW* sample database is a good example. Create a new cube by using the Cube Wizard, selecting an appropriate fact table and a set of related dimension tables. Specify a time dimension with one or

more date properties, select the measures to include in the cube, and select the attributes and hierarchies to be created within the associated dimensions. After the new cube and dimensions are created, modify properties related to the measures and measure groups as well as to the dimensions and attributes. Process the resulting SSAS database, and then use the browser tabs within the various SSAS designers to see what effect your changes have from an end-user perspective.

- **Practice 2** Add one or more dimensions to an existing SSAS solution by using the Dimension Wizard. If possible, create at least one dimension that uses a set of related dimension tables (such as separate product, subcategory, and category tables). Use the Dimension Designer to review, modify, and browse the resulting dimensions to see how various properties change the behavior of the dimension and its attributes. Add one or more hierarchies to at least one of the dimensions, reviewing the properties related to the hierarchies and the levels. Process and browse the dimensions, using the new hierarchies to check your work.
- **Practice 3** Add one or more database dimensions to an existing cube. Ensure that the dimensions are correctly related to the measure groups within the cube by using the Dimension Usage tab in the Cube Designer. If possible, use more than one of the different types of dimension-to-measure group relationships. Consider implementing a Referenced or Many-to-Many relationship, if the underlying database model supports it.

## Chapter 7: Extending SSAS Cubes

### Extending SSAS Cubes

- **Practice 1** Add one or more AdventureWorks fact tables to the DSV. Add measure groups to the AdventureWorks DW cubes that use the fact tables as a source. After you add a measure group, verify the measure group relationships (regular, reference, and many-to-many) to the cube dimensions. Practice creating a measure group for a distinct count measure. Create reports in the Cube Browser or Excel 2007 that request measures from two or more measure groups. If the measure groups are joined correctly to the cube dimensions, you should be able to slice the measures in the measure groups as though they belong to a single measure group.
- **Practice 2** Implement an end-user model in the AdventureWorks DW cube consisting of KPIs, actions, perspectives, and translations. Use MDX expressions to define the KPI properties. Practice creating a regular action, such as a URL action, and limit its scope to an attribute hierarchy. Implement Internet Sales and Reseller Sales perspectives to expose subsets of the cube metadata. Localize the cube to a foreign language by using dimension translations.
- **Practice 3** Implement a Reseller Sales Average Order Amount calculated member by dividing the Reseller Sales Amount measure by Reseller Sales Count. Assign the Reseller

Sales Average Order Amount to a display folder. Implement a named set that returns the top-selling products. Use SSMS to create a basic MDX query that displays all products on rows.

## Chapter 8: Defining SSAS Storage, Partitions, and Aggregations

### Defining SSAS Storage, Partitions, and Aggregations

- **Practice 1** Partition the Reseller Sales measure group. Experiment with different storage modes to understand the ways in which they affect query performance. Implement a low-latency partition that uses proactive caching.
- **Practice 2** Create a few MDX queries in SSMS. Test the queries and establish a performance baseline. Run the Aggregation Design Wizard and process the cube. Execute the same queries again and measure the query response times. If the performance results are the same, use the Aggregation Manager utility to see the aggregations that were created by the Aggregation Design Wizard. Use SQL Server Profiler to see if a given query results in an aggregation hit.
- **Practice 3** Enable the server query log. Execute the Practice 2 queries and make sure they are logged in the query log table. Run the Usage-Based Optimization Wizard to change the aggregation design. Execute the queries again and measure the query response times. Use SQL Server Profiler to see if a given query results in an aggregation hit.

## Chapter 9: Developing SSAS Data Mining Models

### Developing SSAS Data Mining Models

- **Practice 1** Use the SSIS Percentage Sampling transformation to split the data into training sets and test sets.
- **Practice 2** Create additional models in the predictive model structure you created through the practices in this chapter. Try to control the behavior of different algorithms by changing their parameters.
- **Practice 3** Look at the business practices of your company or of another company whose practices are familiar to you. Determine some ways that data mining could be used to help improve the business.

## Chapter 10: Developing SSRS Reports

### Developing SSRS Reports

- **Practice 1** Walk through the SQL Server 2005 Books Online (BOL) Reporting Services Tutorials. The “Creating a Basic Report” and the “Adding Grouping, Sorting, and Formatting to a Basic Report” tutorials put into practice some of the material covered in this chapter.
- **Practice 2** Download, install, and study SQL Server 2005 Report Packs. You can download the Report Packs from <http://www.microsoft.com/downloads/details.aspx?FamilyId=D81722CE-408C-4FB6-A429-2A7ECD62F674&displaylang=en>. Pay particular attention to the SQL Server 2005 Report Pack for Financial Reporting and the different techniques it uses to provide an attractive user experience.
- **Practice 3** Install and study the SQL Server Books Online Reporting Services Samples. These reports are located by default in the C:\Program Files\Microsoft SQL Server\90\Samples\Reporting Services folder.

## Chapter 11: Formatting and Extending SSRS Reports

### Formatting and Extending SSRS Reports

- **Practice 1:** Walk through the SQL Server 2005 Books Online (BOL) Reporting Services Tutorials. The Advanced Features Using Parameters and the Tutorials for Ad Hoc Reporting put into practice some of the material covered in this chapter.

Control data by applying parameters; assign parameter defaults and data types; bind datasets to parameters; and expose parameters to the user interface.

- **Practice 2** Download, install, and study SQL Server 2005 Report Packs. Report Packs can be downloaded from: <http://www.microsoft.com/downloads/details.aspx?FamilyId=D81722CE-408C-4FB6-A429-2A7ECD62F674&displaylang=en>. Pay particular attention to the SQL Server 2005 Report Pack for Financial Reporting and the different techniques the Report Pack offers to provide an attractive user experience.

Expose parameters to the user interface; apply formatting and style; apply dynamic formatting by using expressions; apply static formatting; and extend a report by using expressions.

- Install the Books on Line SQL Server Reporting Services Samples and study the Report Samples. By default, these reports are installed in the C:\Program Files\Microsoft SQL Server\90\Samples\Reporting Services folder.

Create a Report Model for Report Builder; Import a DSV into a Report Model project; create a DSV from a relational database source by using Model Designer; create a model from a multi-dimensional data source by using Report Manager.

## Chapter 12: Deploying and Configuring SSIS Packages

### Deploying SSIS Packages

- **Practice 1** Create two packages that perform some simple control flow and/or data flow logic against a test database; use the machine name in the connection rather than using (**local**). Make a copy of the test database on a second server, and practice creating a single SSIS configuration file with the test database connection that both packages share. Move the packages and the configuration to the second server and modify the configuration file to point to the second server. Without opening the packages in BIDS, run the packages on the second server, and then check to be sure that the test database on the second server responded to your control flow and data flow logic.
- **Practice 2** Practice using DTUtil in a command window to deploy a package from the file system to a remote SQL Server. After the deployment, open SSMS, connect to the SSIS service, and check to make sure that the package was successfully deployed. After verifying successful deployment, use DTUtil to delete the package on the remote server and check the results.

## Chapter 13: Deploying and Processing SSAS Objects

### Deploying and Processing SSAS Objects

- **Practice 1** Experiment with different options for partitions and roles in the Deployment Wizard to understand how they affect the management settings of a deployed database. Consider creating a new named SSAS instance on your local machine to test the Synchronize Database Wizard.
- **Practice 2** Make changes to one of the Adventure Works dimension tables. Fully process the dimension and attempt to browse a cube that uses this dimension. Note that the cube is invalidated and you need to fully process the cube before you can browse its data. Compare this with processing the dimension incrementally by using the Process Update option.
- **Practice 3** Create a data mining structure and enable drillthrough. Experiment with different processing options to see how they affect the models inside the structure and the drillthrough feature.

## Chapter 14: Deploying and Scheduling SSRS Reports

### Deploying and Scheduling SSRS Reports

- **Practice 1** Create a report, and test all the different ways it can be deployed.
- **Practice 2** Try to create an email subscription to a report.
- **Practice 3** Analyze your company's needs to see whether the company could benefit from using shared schedules and data-driven subscriptions.

## Chapter 15: Securing and Executing SSIS Packages

### Securing and Executing SSIS Packages

- **Practice 1** To validate your knowledge of using SQL Server storage and package roles, create two packages that perform some simple tasks. Import the first package into the SQL Server *msdb* database by using SSMS. Then, create a test local account on the server, give the test account login access to your SQL Server, and assign the new account the *db\_dtsoperator* role in the *msdb* database. Next, log on to the server with the new account, connect to the SSIS service in SSMS using the new user/login, and try to execute the package that already exists. Next, try to delete the package, and then attempt to import the second package that you created. Verify that you can only execute existing packages, but cannot delete packages or import new packages into SQL Server.
- **Practice 2** Check your understanding of the command-line utilities DTUtil and DTExec by using them to encrypt packages and then creating a batch file that runs the packages. For the first step, use DTUtil to encrypt the entire contents of an existing package. Then, try to open the package in BIDS to validate that the package password needs to be used. In addition, open the package .dtsx file with Notepad to observe the encrypted characters. Finally, create a batch file (.bat) and enter a single execution line by using DTExec to run the same package. Be sure to use the /Decrypt "password" command parameter so the package can be decrypted at execution time.

## Chapter 16: Administering and Securing SSAS Cubes

### Administering and Securing SSAS Cubes

- **Practice 1** Create a new database role. For the Product Category attribute hierarchy, define an allowed set that permits only the Bikes member. Connect under the new role and create a report proving that the role can see only the Bikes product category. Create a new report that uses the Product attribute of the Dim Product dimension. Notice that the database role can see only products that belong to the Bikes category.

- **Practice 2** Back up an SSAS database. Restore the database. Create a backup script and schedule it by using SQL Server Agent.
- **Practice 3** Use SQL Server Profiler to create an SSAS trace. Use SSMS to browse the cube metadata and data. Observe the events in Profiler as you perform different actions in SSMS. Save the trace as a file. Open the trace file in Profiler and replay it.

## Chapter 17: Securing and Querying SSAS Data Mining Models

### Securing and Querying SSAS Data Mining Models

- **Practice 1** Using DMX, create a mining structure that includes a nested table.
- **Practice 2** Create a report based on a DMX query to browse the content of a model you create manually. Do not forget to use the FLATTENED DMX keyword.

## Chapter 18: Managing and Securing SSRS Reports

### Managing and Securing SSRS Reports

- **Practice 1** Create different mixes of cached reports and snapshot configurations. Verify the results in the ExecutionLog, SnapshotData, and ChunkData tables in the ReportServer database and in the ExecutionCache and ChunkData tables in the ReportServer-TempDB database.
- **Practice 2** Implement a set of nested folders and reports. Create five Windows local users, and grant permissions to different objects with different roles, at both the system level and the item level. Test the access and abilities of each one.