



Designing Business Intelligence Solutions with Microsoft SQL Server (467)

In association with : **Silver Microsoft Partner**



Plan business intelligence (BI) infrastructure

Plan for performance

- Optimise batch procedures: extract, transform, load (ETL) in SQL Server Integration Services (SSIS)/SQL and processing phase in Analysis Services; configure Proactive Caching within SQL Server Analysis Services (SSAS) for different scenarios; understand performance consequences of named queries in a data source view; analyse and optimise performance, including Multidimensional Expression (MDX) and Data Analysis Expression (DAX) queries; understand the difference between partitioning for load performance versus query performance in SSAS; appropriately index a fact table; optimise Analysis Services cubes in SQL Server Data Tools; create aggregations

Plan for scalability

- Change binding options for partitions; choose the appropriate Multidimensional OLAP (MOLAP), Relational OLAP (ROLAP) and Hybrid OLAP (HOLAP) storage modes

Plan and manage upgrades

- Plan change management for a BI solution

Maintain server health

- Design an automation strategy

Design BI infrastructure

Design a security strategy

- Configure security and impersonation between SQL Server service, analysis services and front end; implement Dynamic Dimension Security within a cube; configure security for an extranet environment; configure Kerberos security; design authentication mechanisms; design security tests; build secure solutions end to end; design roles for calculated measures; understand the trade-offs between regular SSAS security and dynamic security

Design a SQL partitioning strategy

- Choose the proper partitioning strategy for the data warehouse and cube, implement a parallel load to fact tables by using partition switching, use data compression

Design a high availability and disaster recovery strategy

- Design a recovery strategy, back up and restore SSAS databases, back up and restore SSRS databases, move and restore the SSIS Catalogue, design an AlwaysON solution

Design a logging and auditing strategy

- Design a new SSIS logging infrastructure (for example, information available through the catalogue views), validate data is balancing and reconciling correctly

Design a reporting solution

Design a Reporting Services dataset

- Design appropriate data query parameters, create appropriate SQL queries, create appropriate DAX queries for an application, manage data rights and security, extract data from analysis services by using MDX queries, balance query-based processing versus filter-based processing, manage data sets through the use of stored procedures

Manage Excel Services/reporting for SharePoint

- Configure data refresh schedules for PowerPivot published to SharePoint, publish BI info to SharePoint, use SharePoint to accomplish BI administrative tasks, install and configure Power View, publish PowerPivot and Power View to SharePoint

Design a data acquisition strategy

- Identify the data sources that need to be used to pull in the data, determine the changes (incremental data) in the data source (time window), identify the relationship and dependencies between the data sources, determine who can access which data, determine what data can be retained for how long (regulatory compliance, data archiving, ageing), design a data movement strategy, profile source data

Plan and manage reporting services configuration

- Choose the appropriate reporting services requirements (including native mode and SharePoint mode)

Design BI reporting solution architecture

- Linked drill-down reports, drill-through reports and sub reports; design report migration strategies; access report services API; design code-behind strategies; identify when to use Reporting Services (RS), Report Builder (RB) or Power View; design and implement context transfer when interlinking all types of reports (RS, RB, Power View, Excel); implement BI tools for reporting in SharePoint (Excel Services versus PowerView versus Reporting Services); select a subscription strategy; enable Data Alerts; design map visualisation

Design BI data models

Design the data warehouse

- Design a data model that is optimised for reporting; design and build a cube on top; design enterprise data warehouse (EDW) and OLAP cubes; choose between natural keys and surrogate keys when designing the data warehouse; use SQL Server to design, implement and maintain a data warehouse, including partitioning, slowly changing dimensions (SCD), change data capture (CDC), Index Views and column store indexes; identify design best practices; implement a many-to-many relationship in an OLAP cube; design a data mart/warehouse in reverse from an Analysis Services cube; implement incremental data load; choose between performing aggregation operations in the SSIS pipeline or the relational engine

Design a schema

- Multidimensional modelling starting from a star or snowflake schema, design relational modelling for a Data Mart

Design cube architecture

- Partition cubes and build aggregation strategies for the separate partitions; design a data model; choose the proper partitioning strategy for the data warehouse and cube; design the data file layout; identify the aggregation method for a measure in a MOLAP cube; performance tune a MOLAP cube using aggregations; design a data source view; design for cube drill-through and write back actions; choose the correct grain of data to store in a measure group;

design analysis services processing by using indexes, indexed views and order-by statements

Design fact tables

- Design a data warehouse that supports many to many dimensions with factless fact tables

Design BI semantic models

- Plan for a multidimensional cube; support a many-to-many relationship between tables; choose between multidimensional and tabular, depending on the type of data and workload

Design and create MDX calculations

- Design MDX queries, identify the structures of MDX and the common functions (tuples, sets, TopCount, SCOPE, VisualTotals and more), create calculated members in an MDX statement, identify which MDX statement would return the required result, implement a custom MDX or logical solution for a pre-prepared case task

Design an ETL solution

Design SSIS package execution

- Use the new project deployment model; pass values at execution time; share parameters between packages; plan for incremental loads versus full loads; optimise execution by using Balanced Data Distributor (BDD); choose optimal processing strategy (including Script transform, flat file incremental loads and Derived Column transform)

Plan to deploy SSIS solutions

- Deploy the package to another server with different security requirements, secure integration services packages that are deployed at the file system, demonstrate awareness of SSIS packages/projects and how they interact with environments (including recoverability), decide between performing aggregation operations in the SSIS pipeline or the relational engine, plan to automate SSIS deployment, plan the administration of the SSIS Catalogue database

Design package configurations for SSIS packages

- Avoid repeating configuration information entered in SSIS packages and use configuration files



Thank You

www.cognixia.com