

Exam 70-463: Implementing a Data Warehouse with Microsoft SQL Server 2012/2014 – Skills Measured

Audience Profile

This exam is primarily intended for ETL and data warehouse developers who create Business Intelligence (BI) solutions, and whose responsibilities include data cleansing, and Extract Transform Load and data warehouse implementation.

Skills Measured

NOTE: The bullets that appear below each of the skills measured are intended to illustrate how we are assessing that skill. This list is not definitive or exhaustive.

NOTE: In most cases, exams do NOT cover preview features, and some features will only be added to an exam when they are GA (General Availability).

Design and implement a data warehouse (10–15%)

Design and implement dimensions

- design shared/conformed dimensions; determine if you need support for slowly changing dimensions; determine attributes; design hierarchies; determine whether you need star or snowflake schema; determine the granularity of relationship with fact tables; determine the need for auditing or lineage; determine keys (business transactional or your own data warehouse/surrogate keys); implement dimensions; implement data lineage of a dimension table

Design and implement fact tables

- design a data warehouse that supports many to many relationships; appropriately index a fact table; using columnstore indexes; partitioning; additive measures; semi additive measures; non additive measures; implement fact tables; determining the loading method for the fact tables; implement data lineage of a fact table; design summary aggregation tables

Extract and transform data (20–25%)

Define connection managers

- plan the configuration of connection managers; package level or project level connection manager; define a connection string; parameterization of connection strings

Design data flow

- define data sources and destinations; distinguish blocking and non-blocking transformations; use different methods to pull out changed data from data sources; determine appropriate data flow components; determine the need for supporting Slowly Changing Dimensions (SCD); determine whether to use SQL Joins or SSIS lookup or merge join transformations; batch processing versus row by row processing; determine the appropriate transform to use for a specific task; determine the need and method for identity mapping and deduplicating; fuzzy lookup, fuzzy grouping and Data Quality Services (DQS) transformation; determine the need for custom data sources, destinations, and transforms; determine what to do with erroneous rows; determine auditing needs; trusted/authoritative data sources, including warehouse metadata; extracting only changed rows

Implement data flow

- debug data flow; use the appropriate data flow components; SQL / SSIS data transformation; create SSIS packages that support slowly changing dimensions; use the lookup task in SSIS; map identities using SSIS fuzzy lookup (advanced); specify a data source and destination; use data flows; different categories of transformations; read, transform and load data; understand which transforms to use to accomplish a specific business task; data correction transformation; performance tune an SSIS dataflow; optimize Integration Services packages for speed of execution; maintain data integrity, including good data flow

Manage SSIS package execution

- schedule package execution by using SQL Server Agent; execute packages by using DTEXEC; execute packages by using SQL Server Management Studio; implement package execution; plan and design package execution strategy; use PowerShell to execute script; monitor the execution using Management Studio; use DTEXECUI; ETL restartability

Implement script tasks in SSIS

- determine if it is appropriate to use a script task; extending the capability of a control flow; perform a custom action as needed (not on every row) during a control flow

Load data (25–30%)

Design control flow

- determine control flow; determine containers and tasks needed; determine precedence constraints; design an SSIS package strategy with rollback, staging and transaction control; decide between one package or multiple packages; determine event handlers; determine variables; determine parameters on package and project level; determine connection managers and whether they are package or project level; determine the need for custom tasks; determine how much information you need to log from a package; determine the need for checkpoints; determine security needs

Implement package logic by using SSIS variables and parameters

- user variables; variable scope, data type; implement parameterization of properties using variables; using variables in precedence constraints; referring to SSIS system variables; design dynamic SSIS packages; package configurations (file or SQL tables); expressions; package and project parameters; project level connection managers; variables; implement dynamic package behavior; configure packages in SSIS for different environments, package configurations (xmlconfiguration file, SQLServer table, registry entry; parent package variables, environment variable); parameters (package and project level); project connection managers; property expressions (use expressions for connection managers)

Implement control flow

- checkpoints; debug control flow; implement the appropriate control flow task to solve a problem; data profiling; use sequence containers and loop containers; manage transactions in SSIS packages; managing parallelism; using precedence constraint to control task execution sequence; creating package templates; using the execute package task

Implement data load options

- implement a full and incremental data load strategy; plan for an incremental update of the relational Data Mart; plan for loads into indexed tables; configure appropriate bulk load options; select an appropriate load technique (SSIS Destination versus T-SQL) and load partitioned tables

Implement script components in SSIS

- create an SSIS package that handles SCD Type 2 changes without using the SCD component; work with script component in SSIS; deciding when it is appropriate to use a script component versus a built in; source, transformation, destination component; use cases: web service source and destination, getting the error message

Configure and deploy SSIS solutions (20–25%)

Troubleshoot data integration issues

- performance issues; connectivity issues; execution of a task or transformation failed; logic issues; demonstrate awareness of the new SSIS logging infrastructure; troubleshoot a failed package execution to determine the root cause of failure; troubleshoot SSIS package failure from an invalid datatype; implement break points; data viewers; profile data with different tools; batch cleanup

Install and maintain SSIS components

- software installation (IS, management tools); development box and server; install specifics for remote package execution; planning for installation (32- versus 64-bit); upgrade; provisioning the accounts; creating the catalog

Implement auditing, logging, and event handling

- audit package execution by using system variables; propagate events; use log providers; log an SSIS execution; create alerting and notification mechanisms; use Event Handlers in SSIS to track ETL events and errors; implement custom logging

Deploy SSIS solutions

- create and configure an SSIS catalog; deploy SSIS packages by using the deployment utility; deploy SSIS packages to SQL or file system locations; validate deployed packages; deploy packages on multiple servers; how to install custom components and tasks; deploy SSIS packages by using DTUTIL

Configure SSIS security settings

- SSIS catalog database roles; package protection levels; secure Integration Services packages that are deployed at the file system; secure Integration Services parameters, configuration

Build data quality solutions (15–20%)

Install and maintain data quality services

- installation prerequisites; .msi package; adding users to the DQ roles; identity analysis, including data governance

Implement master data management solutions

- install Master Data Services (MDS); implement MDS; create models, entities, hierarchies, collections, attributes; define security roles; import/export; subscriptions

Create a data quality project to clean data

- profile Online Transaction Processing (OLTP) and other source systems; data quality knowledge base management; create data quality project; use data quality client; improve data quality; identity mapping and deduplicating; handle history and data quality; manage data quality/cleansing