Department of Computer Science and Informatics
CSID6853 – Data Warehousing – 2025
Practical Project - Part C

Instructions

- Ensure your data warehouse has been pre-created on CSI server with all the relevant fact and dimension tables.
- All the tables must be empty.
- Create a stored procedure (in your database as well), called spCreateDM, that will drop and create all the tables in the database. Upload the SQL file also onto Blackboard.
  (format: spCreateDM_studentnumber_InitialsSurname.zip).
- Upload in **one zip file,** all the files required to run your **SSIS** Visual Studio SSIS Solution onto Blackboard. Ensure in includes the **three** packages as described below. Include the store procedure as well.
  (format: SSIS_studentnumber_InitialsSurname.zip)
- Book a 50-minute timeslot on **03-Jun-2025, 04-June-2025** or **05-Jun-2025,** to demonstrate the SSIS ETL to me.

SQL Server Integration Services Packages [100 marks]

- You need to develop the necessary ETL packages to load the dimensional model (see figure 1) for the Maker Inc business.
- Use Microsoft Visual Studio SSIS for this assignment.
- Make sure you can run your package(s) multiple times.
  - In other words, cleanup / delete the data from the tables before you load the necessary data.
- Save all error files to a directory called **C:\SSISTemp\Errors**.
- **Use Project Conversions for all connections.**
- **Create all the tables (exactly as in figure 1) in your SQL Server.**
- Your SSIS solution should consist of the following packages:
  - Master
  - LoadDimension
  - LoadFact

2 Fact Tables
10 Dimension Tables

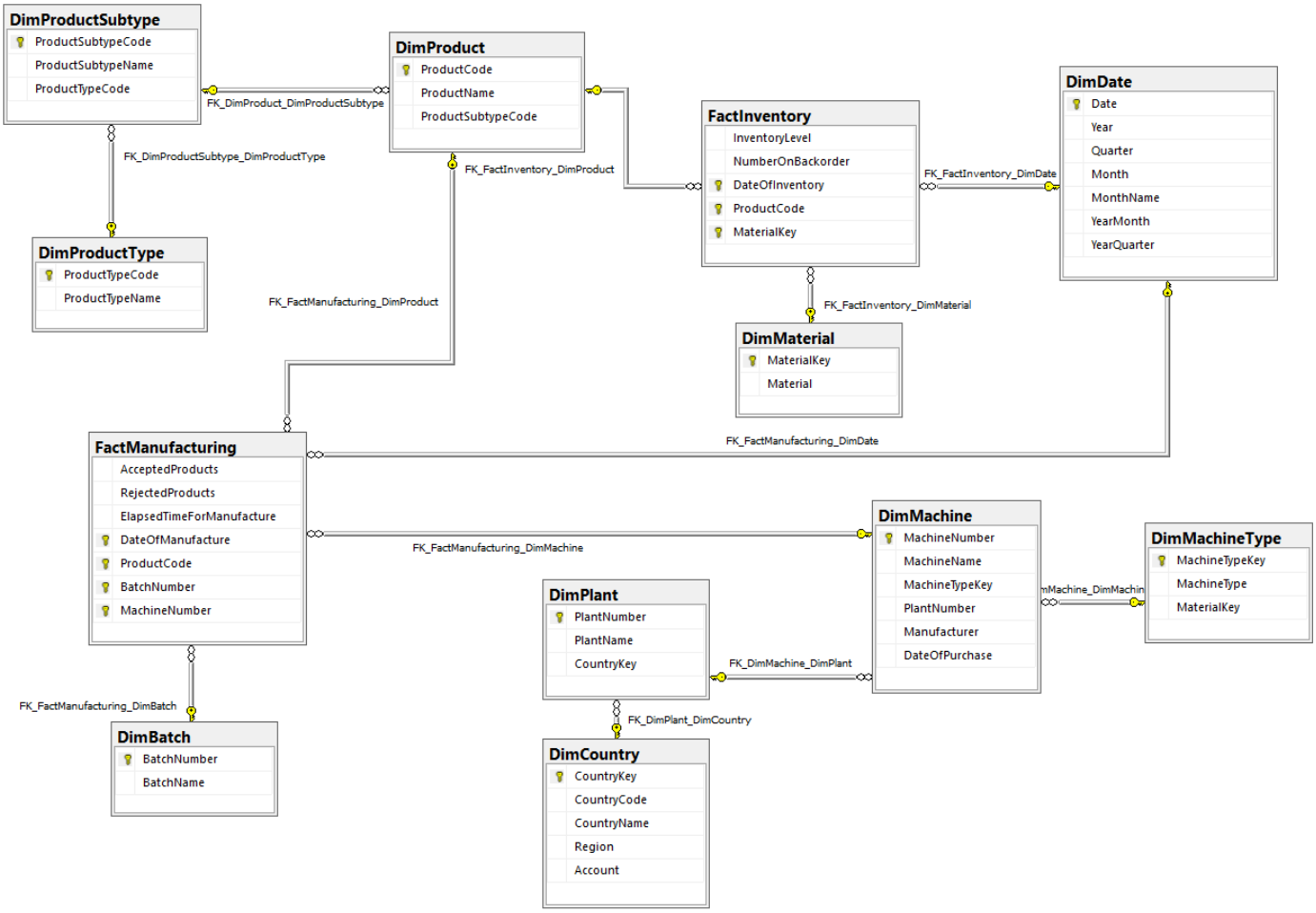
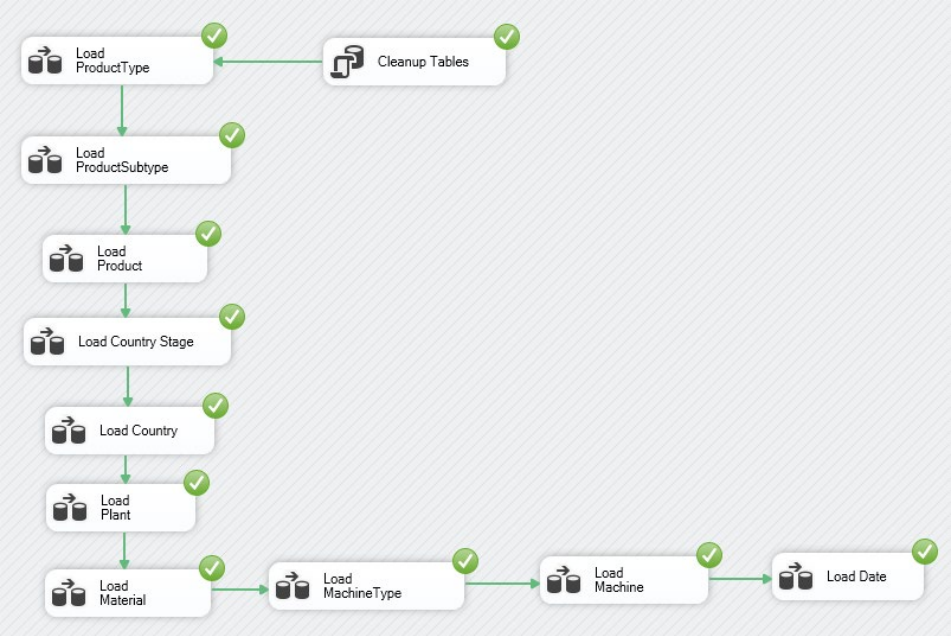


Figure 1: Dimensional Model

Package	Purpose
MasterPackage.dstx	To run all the individual packages (LoadDimensions and LoadFacts) in sequence.

LoadDimensions.dtsx	1. You must use separate Data Flows for each step (see screenshot below)



2. Cleanup the tables with SQL statements.
3. Load all the dimension tables **<u>except</u>** DimBatch in this package.
4. Use all possible data sources to load the dimension tables.
5. For the "Load Date" Data Flow, you will need to use date fields from both the BatchData.csv as well as Order Processing System to ensure you have all possible dates loaded.
6. Create the directory **C:\SSISTemp** and load BatchData.csv from it.
    a. MonthName must be the full English month name
    b. YearMonth must be in the format YYYY-mm (for example 2013-01)
    c. YearQuarter must be in the format YYYYQqq (for example 2013Q1)
7. See some sample rows for DimDate

	Date	Year	Quarter	Month	MonthName	YearMonth	YearQuarter
1	2013-01-01	2013	1	1	January	2013-01	2013Q1
2	2013-01-02	2013	1	1	January	2013-01	2013Q1
3	2013-01-03	2013	1	1	January	2013-01	2013Q1
4	2013-01-06	2013	1	1	January	2013-01	2013Q1

8. For the "Load ProductType", "Load ProductSubType" and "Load Product" Data Flows use the Accounting System as source data.

	9. For the "Load Country" Data Flow you will need to incorporate a hierarchy. a. Create the directory **C:\SSISTemp** and load ExternalAccountData.xls from it. b. ExternalAccountData.xls contains the hierarchy. c. At the lowest level there is a 'Country', which belongs to a 'Region' which belongs to an 'Account'. d. The **CountryDim** table must contain for each country, the sales region it belongs to (Region column) and the name of the Account it reports to (Account column). e. The relationship between 'Countries' and ExternalAccountData.xls is the Country's Name. f. You are free to implement this in any way you want. g. "Load Country Stage" is optional – you can incorporate all ETL in "Load Country". **10.** For the "Load Plant" Data Flow use the Accounting System and source data and load only where **LocationType="Plant Site"** 11. For the "Load Material", "Load MachineType" and "Load Machine" Data Flows use the Accounting System as source data. 12. For the "Load Material", "Load MachineType" and "Load Machine" Data Flows, load only where **AssetType = "Molding Machine"** 13. For the "Load Material", "Load MachineType" and "Load Machine" Data Flows use **aggregation** transformation when loading the DimMaterial, DimMachineType and DimMachine dimension tables. 14. Ensure that the data warehouse connection is converted to a Project Connection in the context menu.

LoadFacts.dtsx	1. You must use separate Data flows each step (see screenshot below)  2. Cleanup the tables with SQL statements. 3. Load the dimension table called DimBatch with the "Load DimBatch" Data Flow. a. Use sort and derived column to assist you. b. **Remember, you must load the DimBatch table using only the raw data from BatchData.csv.** 4. Load the fact table called FactManufacturing with the "Load Manufacturing Fact" Data Flow.

	a. Create the directory **C:\SSISTemp** and load BatchData.csv from it. b. Use BatchData.csv as the source for the fact table. c. You need to calculate the "number of products accepted", "number of products rejected", "elapsed time for manufacture" and "date of manufacture" fields. d. **Tip**: Use a DATEDIFF() function with *TimeStarted* and *TimeStopped* as parameters to calculate "elapsed time for manufacture". e. Make use of Lookups to ensure only data loads that are present in the DimBatch, DimMachine, and DimProduct tables. f. Redirect all rows to no match output (to handle rows with no matching entries). g. Make use of a Union All for all unmatched entries and write it to a delimited text file. h. Call the file **C:\SSISTemp\ManufactureErrors.txt** 5. Load the fact table called FactInventory with the "Load Inventory Fact" Data Flow. a. Use the Order Processing System as the source for the fact table. b. Make use of Lookups to ensure only data loads that are present in the DimProduct and DimMaterial tables. c. Convert the attribute Material to Unicode string. Use this as a new source column for Material in the fact table. d. Redirect rows to no match output (to handle rows with no matching entries). e. Make use of a Union All for all unmatched entries and write it to a delimited text file. f. Call the file **C:\SSISTemp\InventoryErrors.txt**

Possible Errors:
[Excel Source [2]] Error: SSIS Error Code DTS_E_CANNOTACQUIRECONNECTIONFROMCONNECTIONMANAGER. The AcquireConnection method call to the connection manager "ExternalAccountData" failed with error code 0xC0209303. There may be error messages posted before this with more information on why the AcquireConnection method call failed.

Cause:
There is a 32-bit driver and a 64-bit driver with the same name for SSIS connection managers. If both providers are visible in the environment, the correct provider will be used. If the correct provider is not visible, you may get error code 0xC0202009 at connection time.

Diagnosing The Problem:
In the properties of the SSIS project, locate the 'Run64BitRuntime' parameter.
Run64BitRunTime=False => will find and use the 32-bit provider.
Run64BitRunTime=True => will find and use the 64-bit provider.

Source:
https://www.ibm.com/support/pages/using-ssis-sql-server-integration-services-properly-configured-informix-oledb-driver-may-yield-ssis-error-code-0xc0202009

oooOooo