

Department of Information Technology and Management
ITMD 526 – Data Warehousing

GROUP 4 - PROJECT REPORT

TEAM MEMBERS:

Name	Email ID
Amruta Lele	alele@hawk.iit.edu
Hashmitha Shantharam	hshantharam@hawk.iit.edu
Pooja Kuchani	pkuchani@hawk.iit.edu
Srinath Sunku	ssunku@hawk.iit.edu
Vishnu Pajjuri	vpajjuri1@hawk.iit.edu
Bharath Manjunath	bmanjunath@hawk.iit.edu

AIM:

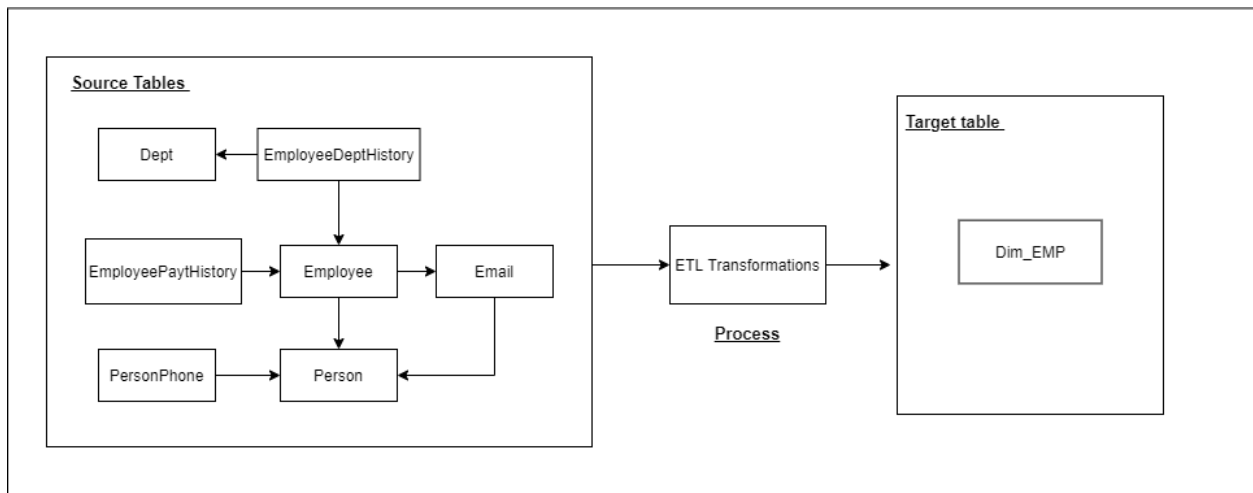
To Perform denormalization of the Adventure Works OLTP dataset to build two-dimension tables as below:

- An Employee dimension.
- A Product dimension.

Step 1:

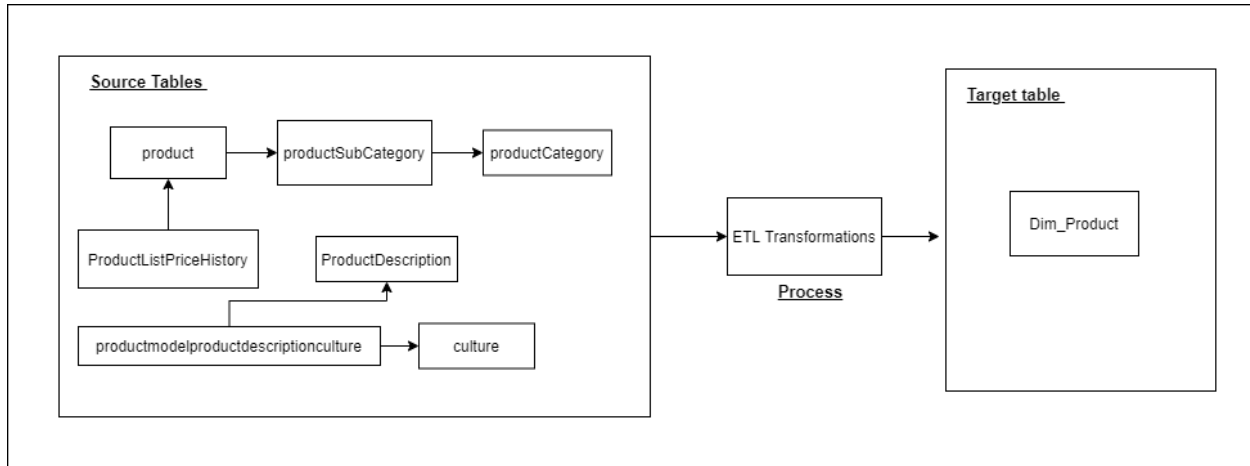
DATA FLOW DIAGRAM:

Employee Dimension:



Department of Information Technology and Management
ITMD 526 – Data Warehousing

Product Dimension:



Step 2:

SOURCE TO TARGET MAPPING:

Employee Dimension:

ETL Package Name	Job.kjb <u>dim_emp.ktr</u>
Target Table	dimemp
Source Tables	department employee personsphone email address person employeedepartmenthistory employeeemployhistory
Load Frequency	Whenever a value is being updated
Target Type	SCD Type-2
Mapping Descriptions	FROM t_data a JOIN person p ON a.BusinessEntityID = p.BusinessEntityID LEFT OUTER JOIN t_dept_range b ON a.BusinessEntityID = b.BusinessEntityID AND a.seq >= b.seq_start AND a.seq < b.seq_end LEFT OUTER JOIN t_payr_range c

Department of Information Technology and Management

ITMD 526 – Data Warehousing

	ON a.BusinessEntityID = c.BusinessEntityID AND a.seq >= c.seq_start AND a.seq < c.seq_end LEFT OUTER JOIN Emailaddress em ON em.BusinessEntityID = p.BusinessEntityID LEFT OUTER JOIN personphone ph ON ph.BusinessEntityID = p.BusinessEntityID LEFT OUTER JOIN employee e ON e.BusinessEntityID = p.BusinessEntityID
Error Handling	By clearing and disabling the cache in SQL we were able to handle the Serializing-row-to-byte-array error.

Mapping Change Date	Target Table	TARGET			Data-type, Length	SOURCE					
		Target Column	Nullable	PK		Source Table	Source Column	Data-type, Length	Expression, Transformation	Default Value	Error Types and Handling
5/5/2021	dimemp	BusinessEntityID FirstName LastName MiddleName NationalIDnumber DepartmentName PayRate JobTitle HireDate BirthDate MaritalStatus SalariedFlag Gender NameStyle ModifiedDate EmailAddress PhoneNumber LoginID VacationHours SickLeaveHours EmployeeKey	Except for surrogate key all the keys can be null.	EmployeeKey	1) EmployeeKey - BIGINT AUTO_INCREMENT NOT NULL 2) BusinessEntityID - INT(11) NOT NULL 3) NameStyle, SalariedFlag - TINYINT(1) DEFAULT NULL 4) VacationHours, SickLeaveHours - SMALLINT(6) DEFAULT NULL 5) ModifiedDate, HireDate, BirthDate - DATE DEFAULT NULL 6) MaritalStatus, Gender - CHAR(1) DEFAULT NULL 7) FirstName, LastName, MiddleName, Title, LoginID, EmailAddress, DepartmentName, version - VARCHAR(50) DEFAULT NULL	department employee personphone email address person employeedepartmenthistory employee employeehistory	Person.FirstName person.LastName person.MiddleName person.namestyle employee.NationalIDnumber employee.jobtitle employee.Hiredate employee.birthdate employee.maritalstatus employee.salariedflag employee.Gender employee.loginid employee.vacationhours employee.sickleavehours employee.departmenthistory BusinessEntityID t_dept_range.DepartmentName emailAddress.emailaddress personPhone.phonenumber	BusinessEntityID int(11) FNMAELNAME.MNAME varchar(100) NationalIDnumber - Varchar(15) JobTitleVarchar(50) HireDate, birthDate - Date marital Status - char(1) SalariedFlag tinyint(1) Gender char(1) loginid Varchar(256) VacationHours smallint sick leave hours smallint	Joins used: Join, Left outer Join	No default value is used	By clearing and disabling the cache in SQL we were able to handle the Serializing-row-to-byte-array error.

Product Dimension:

ETL Package Name	Job.kjb <u>dim_prod.ktr</u>
Target Table	dim_product
Source Tables	product productsubcategory productmodel productdescriptionculture productcategory productdescription culture ProductListPriceHistory
Load Frequency	Whenever a value is being updated
Target Type	SCD Type-2
Mapping Descriptions	FROM t_price_history a JOIN productsubcategory n ON a.ProductSubCategoryID = n.ProductSubCategoryID

Department of Information Technology and Management

ITMD 526 – Data Warehousing

	LEFT OUTER JOIN productcategory z ON n.ProductCategoryID = z.ProductCategoryID LEFT OUTER JOIN culture c ON a.CultureID = c.CultureID
Error Handling	By clearing and disabling the cache in SQL we were able to handle the Serializing-row-to-byte-array error.

TARGET						SOURCE					
Mapping Change ID	Target Table	Target Column	Nullable	PK	Data-type, Length	Source Table	Source Column	Data-type, Length	Expression, Transform	Default Value	Error Types and Handling
5/5/2021	dim_product	ProductID StartDate EndDate ModifiedDate ListPrice CategoryName SubCategoryName CultureID culture_name Description version productkey	Except for surrogate key all the keys can be null.	productkey	1)ProductID - INT DEFAULT NULL 2)productkey - INT NOT NULL AUTO_INCREMENT 3) StartDate, EndDate, ModifiedDate - DATE 4) ListPrice, CategoryName, SubCategoryName, CultureID, culture_name, Description, version - VARCHAR(256) DEFAULT NULL	product productsubcategory productmodelproductdescriptionculture productcategory productdescription culture ProductListPriceHistory	Product.ProductID product.ProductSubCategoryID productCategory.ProductCategoryID product.modifiedDate ProductListPriceHistory.startdate ProductListPriceHistory.listPrice ProductCategory.CategoryName ProductSubCategory.SubCategoryName ProductDescription.Description productmodelproductdescriptionculture.CultureID	ProductID - INT ProductSubCategoryID - INT ProductCategoryID - INT modifiedDate - Timestamp startdate - DateTime enddate - DateTime ListPrice Categoryname - VARCHAR(100) SubCategoryname - VARCHAR(100) Description - VARCHAR(400) CultureId - CHAR(6) name - varchar(100)	Joins used: Join, Left out	No default values used	By clearing and disabling the cache in SQL we were able to handle the Serializing-row-to-byte-array error.

Step 3:

ETL PROCESS EXECUTION:

SCD Type 2:

- A Type 2 SCD retains the full history of values. When the value of a chosen attribute changes, the current record is closed.
- A new record is created with the changed data values and this new record becomes the current record.

Kettle Transformation:

Employee Transformation:

SQL Query to create the Dimension Table:

```
CREATE TABLE `dimemp` (
  `EmployeeKey` BIGINT AUTO_INCREMENT NOT NULL,
  `BusinessEntityid` INT(11) NOT NULL,
  `FirstName` VARCHAR(50) DEFAULT NULL,
  `LastName` VARCHAR(50) DEFAULT NULL,
  `MiddleName` VARCHAR(50) DEFAULT NULL,
```

Department of Information Technology and Management

ITMD 526 – Data Warehousing

```
`NameStyle` TINYINT(1) DEFAULT NULL,  
`Title` VARCHAR(50) DEFAULT NULL,  
`HireDate` DATE DEFAULT NULL,  
`BirthDate` DATE DEFAULT NULL,  
`LoginID` VARCHAR(256) DEFAULT NULL,  
`EmailAddress` VARCHAR(50) DEFAULT NULL,  
`Phone` VARCHAR(25) DEFAULT NULL,  
`MaritalStatus` CHAR(1) DEFAULT NULL,  
`SalariedFlag` TINYINT(1) DEFAULT NULL,  
`Gender` CHAR(1) DEFAULT NULL,  
`VacationHours` SMALLINT(6) DEFAULT NULL,  
`SickLeaveHours` SMALLINT(6) DEFAULT NULL,  
`DepartmentName` VARCHAR(50) DEFAULT NULL,  
`version` VARCHAR(50) DEFAULT NULL,  
`modifiedDate` DATE DEFAULT NULL,  
PRIMARY KEY (`EmployeeKey`)  
);
```

Source Input:

Source table combines multiple tables such as department, employee, personsphone, email, address, person, employeedepartmenthistory, employeepayhistory to select the columns needed to populate the employee dimension table.

Department of Information Technology and Management

ITMD 526 – Data Warehousing

Table input

Step name: Input

Connection: itmd526_final

SQL

```
WITH t_data AS (
SELECT BusinessEntityID, ModifiedDate, MAX(DepartmentName) AS DepartmentName, MAX(PayRate) AS PayRate, MAX(seq) AS seq
FROM (
SELECT ROW_NUMBER() OVER (PARTITION BY v.BusinessEntityID ORDER BY v.ModifiedDate) AS seq
, v.BusinessEntityID
, IF(v.data_type = 'DEPT', val, NULL) AS DepartmentName
, IF(v.data_type = 'PAY', val, NULL) AS PayRate
, v.ModifiedDate
FROM (
SELECT 'DEPT' AS data_type, edh.BusinessEntityID, edh.StartDate AS ModifiedDate, d.Name AS Val
FROM employeesdepartmenthistory edh
JOIN department d
ON edh.DepartmentID = d.DepartmentID
UNION ALL
SELECT 'PAY' AS data_type, eph.BusinessEntityID, eph.RateChangeDate AS ModifiedDate, eph.Rate AS Val
FROM employeeepayhistory eph
) v
GROUP BY BusinessEntityID, ModifiedDate
),
t_dept_filtered AS (
SELECT BusinessEntityID, DepartmentName, seq
FROM t_data
WHERE DepartmentName IS NOT NULL
),
t_dept_range AS (
SELECT BusinessEntityID, DepartmentName, seq AS seq_start
, IFNULL(LEAD(seq) OVER (PARTITION BY BusinessEntityID ORDER BY seq), 999999) AS seq_end
FROM t_dept_filtered
)
```

Line 23 Column 60

Enable lazy conversion ☐

Replace variables in script? ☐

Insert data from step

Execute for each row? ☐

Limit size

Help OK Preview Cancel

The selected values from source table are being sorted and mapped to its corresponding target table. The table is being truncated and new value is being inserted on every update.

Department of Information Technology and Management

ITMD 526 – Data Warehousing

Dimension lookup/update

Step name:

Update the dimension? ☒

Connection: Edit... New... Wizard...

Target schema: Browse...

Target table: Browse...

Commit size:

Enable the cache? ☐

Pre-load the cache? ☐

Cache size in rows (0 = cache all):

Keys Fields

Lookup/Update fields

#	Dimension field	Stream field to compare with	Type of dimension update
1	FirstName	FirstName	Insert
2	LastName	LastName	Insert
3	MiddleName	MiddleName	Insert
4	NationalIdnumber	NationalIdnumber	Insert
5	DepartmentName	DepartmentName	Insert
6	PayRate	PayRate	Insert

Technical key field: New name:

Creation of technical key

☐ Use table maximum + 1

☐ Use sequence

☒ Use auto increment field

Version field:

Stream Datefield:

Data range start field: End Date: Min. year:

Help

Result:

Spoon - Dim_Emp

File Edit View Action Tools Help

Connect

View Design

Search

Input

Output

Streaming

Transform

Utility

Flow

Scripting

Pentaho Server

Lookup

Joins

Data Warehouse

Validation

Statistics

Big Data

Agile

Cryptography

Job

Mapping

Bulk loading

Inline

Experimental

Deprecated

Welcome! NewyorkDataXML NewyorkDataXML2 NewyorkDataXML3 Dim_Emp Dim_prod Job

100%

Input Sort rows Dim_Emp

Execution Results

Logging Execution History Step Metrics Performance Graph Metrics Preview data

2021/05/06 10:54:45 - Spoon - Launching transformation [Dim_Emp]...

2021/05/06 10:54:45 - Spoon - Started the transformation execution.

2021/05/06 10:54:45 - Dim_Emp - Dispatching started for transformation [Dim_Emp]

2021/05/06 10:54:45 - Input.0 - Finished reading query, closing connection.

2021/05/06 10:54:45 - Input.0 - Finished processing (I=322, O=0, R=0, W=322, U=0, E=0)

2021/05/06 10:54:45 - Sort rows.0 - Finished processing (I=0, O=0, R=322, W=322, U=0, E=0)

2021/05/06 10:54:46 - Dim_Emp.0 - Finished processing (I=322, O=322, R=322, W=322, U=0, E=0)

2021/05/06 10:54:46 - Spoon - The transformation has finished!!

Activate Windows
Go to Settings to activate Windows.

Type here to search

10:54
06-05-2021

Department of Information Technology and Management

ITMD 526 – Data Warehousing

Snapshot of Table after running kettle Transformation:

SQLyog Community 64 - [itmd526/itmd526_final - root@localhost*]

Filter tabl... Data Search - search and navigate to the respective record in a single click : Reason #72 to upgrade

emp_dim* GROUP_PROJEC...Session1 (1)* prod_dim test* x History Query 1 +

PRIMARY KEY ('EmployeeKey')

```
INSERT INTO dimemp (BusinessEntityid,FirstName,LastName,EmailAddress,Phone)
VALUES ('728','Amruta','Lele','abc@works.com','34737434');

TRUNCATE TABLE dimemp;

ALTER TABLE dimemp

SELECT * FROM dimemp;
```

EmployeeKey	BusinessEntityID	FirstName	LastName	MiddleName	NameStyle	Title	HireDate	BirthDate	LoginID	EmailAddress
12	9	Gigi	Matthew	N	0	(NULL)	2200-01-01	1900-01-01	adventure-works\gigi0	gigi0@adventure-works.com
13	10	Michael	Raheem	(NULL)	0	(NULL)	2200-01-01	1900-01-01	adventure-works\michael6	michael6@adventure-works.com
14	11	Ovidiu	Crsacium	V	0	(NULL)	2200-01-01	1900-01-01	adventure-works\ovidiu0	ovidiu0@adventure-works.com
15	12	Thierry	D'Hers	B	0	(NULL)	2200-01-01	1900-01-01	adventure-works\thierry0	thierry0@adventure-works.com
16	13	Janice	Galvin	M	0	(NULL)	2200-01-01	1900-01-01	adventure-works\janice0	janice0@adventure-works.com
17	14	Michael	Sullivan	I	0	(NULL)	2200-01-01	1900-01-01	adventure-works\michael8	michael8@adventure-works.com
18	15	Sharon	Salavaria	B	0	(NULL)	2200-01-01	1900-01-01	adventure-works\sharon0	sharon0@adventure-works.com
19	16	David	Bradley	M	0	(NULL)	2002-01-20	1900-01-01	adventure-works\david0	david0@adventure-works.com
20	16	David	Bradley	M	0	(NULL)	2002-01-20	2002-01-20	adventure-works\david0	david0@adventure-works.com
21	16	David	Bradley	M	0	(NULL)	2200-01-01	2002-01-20	adventure-works\david0	david0@adventure-works.com
22	17	Kevin	Brown	F	0	(NULL)	2200-01-01	1900-01-01	adventure-works\kevin0	kevin0@adventure-works.com
23	18	John	Wood	L	0	(NULL)	2200-01-01	1900-01-01	adventure-works\john5	john5@adventure-works.com
24	19	Mary	Dempsey	A	0	(NULL)	2200-01-01	1900-01-01	adventure-works\mary2	mary2@adventure-works.com

SELECT * FROM dimemp LIMIT 0, 1000

Exec: 0.025 sec Total: 0.028 sec 323 row(s) Ln 32, Col 1 Connections: 1

SQLyog Community 64 - [itmd526/itmd526_final - root@localhost*]

Filter tabl... Data Search - search and navigate to the respective record in a single click : Reason #72 to upgrade

emp_dim* GROUP_PROJEC...Session1 (1)* prod_dim test* x History Query 1 +

PRIMARY KEY ('EmployeeKey')

```
INSERT INTO dimemp (BusinessEntityid,FirstName,LastName,EmailAddress,Phone)
VALUES ('728','Amruta','Lele','abc@works.com','34737434');

TRUNCATE TABLE dimemp;

ALTER TABLE dimemp

SELECT * FROM dimemp;
```

S.	G.	Vac...	SickL...	DepartmentName	version	modifiedDate	NationalIdnumber	PayRate	jobtitle	phonenumber
1	F	63	51	Research and Development	1	2003-02-17	658797903	40.8654	Research and Development Engineer	185-555-0186
1	M	16	64	Research and Development	1	2003-06-04	879342154	42.4808	Research and Development Manager	330-555-2568
0	M	7	23	Tool Design	1	2005-01-05	974026903	28.8462	Senior Tool Designer	719-555-0181
0	M	9	24	Tool Design	1	2002-01-11	480168528	25.0000	Tool Designer	168-555-0183
0	F	8	24	Tool Design	1	2005-01-23	486228782	25.0000	Tool Designer	473-555-0117
1	M	3	21	Engineering	1	2005-01-30	42487730	36.0577	Senior Design Engineer	465-555-0156
1	F	4	22	Engineering	1	2005-02-18	56920285	32.6923	Design Engineer	970-555-0138
1	M	40	40	Purchasing	1	2002-01-20	24756624	24.0000	Marketing Manager	913-555-0172
1	M	40	40	Marketing	2	2003-08-16	24756624	28.7500	Marketing Manager	913-555-0172
1	M	40	40	Marketing	3	2006-06-01	24756624	37.5000	Marketing Manager	913-555-0172
0	M	42	41	Marketing	1	2001-02-26	253022876	13.4615	Marketing Assistant	150-555-0189
0	M	48	44	Marketing	1	2005-03-10	222969461	14.4231	Marketing Specialist	496-555-0150
0	F	43	41	Marketing	1	2005-03-17	52541318	13.4615	Marketing Assistant	124-555-0114

SELECT * FROM dimemp LIMIT 0, 1000

Exec: 0.025 sec Total: 0.028 sec 323 row(s) Ln 32, Col 1 Connections: 1

Product Transformation:

SQL Query to create the Dimension Table:

CREATE TABLE `dim_product` (

Department of Information Technology and Management

ITMD 526 – Data Warehousing

```
`ProductID` INT DEFAULT NULL,  
`StartDate` DATE DEFAULT NULL,  
`EndDate` DATE DEFAULT NULL,  
`ModifiedDate` DATE DEFAULT NULL,  
`ListPrice` VARCHAR(256) DEFAULT NULL,  
`CategoryName` VARCHAR(256) DEFAULT NULL,  
`SubCategoryName` VARCHAR(256) DEFAULT NULL,  
`CultureID` VARCHAR(256) DEFAULT NULL,  
`culture_name` VARCHAR(256) DEFAULT NULL,  
`Description` VARCHAR(256) DEFAULT NULL,  
`version` VARCHAR(256) DEFAULT NULL,  
`productkey` INT NOT NULL AUTO_INCREMENT,  
PRIMARY KEY (`productkey`));
```

Source Input:

Source table combines multiple tables such as product, Productsubcategory, productmodel, productdescriptionculture, productcategory, productdescription, culture, ProductListPriceHistory to select the columns needed to populate the product dimension table.

Table input

Step name

Input

Connection

itmd526_final

Edit...

New...

Wizard...

SQL

Get SQL select statement...

```

WITH t_data AS (
SELECT BusinessEntityID, ModifiedDate, MAX(DepartmentName) AS DepartmentName, MAX(PayRate) AS PayRate, MAX(seq) AS seq
FROM
(
SELECT ROW_NUMBER() OVER (PARTITION BY v.BusinessEntityID ORDER BY v.ModifiedDate) AS seq
, v.BusinessEntityID
, IF(v.data_type = 'DEPT', val, NULL) AS DepartmentName
, IF(v.data_type = 'PAY', val, NULL) AS PayRate
, v.ModifiedDate
FROM (
SELECT 'DEPT' AS data_type, edh.BusinessEntityID, edh.StartDate AS ModifiedDate, d.`Name` AS Val
FROM employeedepartmenthistory edh
JOIN department d
ON edh.DepartmentID = d.DepartmentID
UNION ALL
SELECT 'PAY' AS data_type, eph.BusinessEntityID, eph.RateChangeDate AS ModifiedDate, eph.Rate AS Val
FROM employeeepayhistory eph
) v
) w
GROUP BY BusinessEntityID, ModifiedDate
).
t_dept_filtered AS (
SELECT BusinessEntityID, DepartmentName, seq
FROM t_data
WHERE DepartmentName IS NOT NULL
).
t_dept_range AS (
SELECT BusinessEntityID, DepartmentName, seq AS seq_start
, IFNULL(LEAD(seq) OVER (PARTITION BY BusinessEntityID ORDER BY seq), 999999) AS seq_end
FROM t_dept_filtered
)

```

Line 1 Column 0

Enable lazy conversion ☐

Replace variables in script? ☐

Insert data from step

Execute for each row? ☐

Insert size

0

[illegible]

Department of Information Technology and Management

ITMD 526 – Data Warehousing

Target output Table:

The selected values from source table are being sorted and mapped to its corresponding target table. The table is being truncated and new value is being inserted on every update.

Dimension lookup/update

Connection: itmd526_final [Edit...] [New...] [Wizard...]

Target schema: itmd526_final [Browse...]

Target table: dim_product [Browse...]

Commit size: 100

Enable the cache? ☐

Pre-load the cache? ☐

Cache size in rows (0 = cache all):

Fields

Lookup/Update fields

Dimension field	Stream field to compare with	Type of dimension update
ModifiedDate	ModifiedDate	Insert
ListPrice	ListPrice	Insert
CategoryName	CategoryName	Insert
SubCategoryName	SubCategoryName	Insert
CultureID	CultureID	Insert
culture_name	culture_name	Insert

Technical key field: productkey [New name]

Creation of technical key

☐ Use table maximum + 1

☐ Use sequence

☒ Use auto increment field

Version field: version

Stream Datefield: ModifiedDate

Date range start field: StartDate [Min. year: 1900]

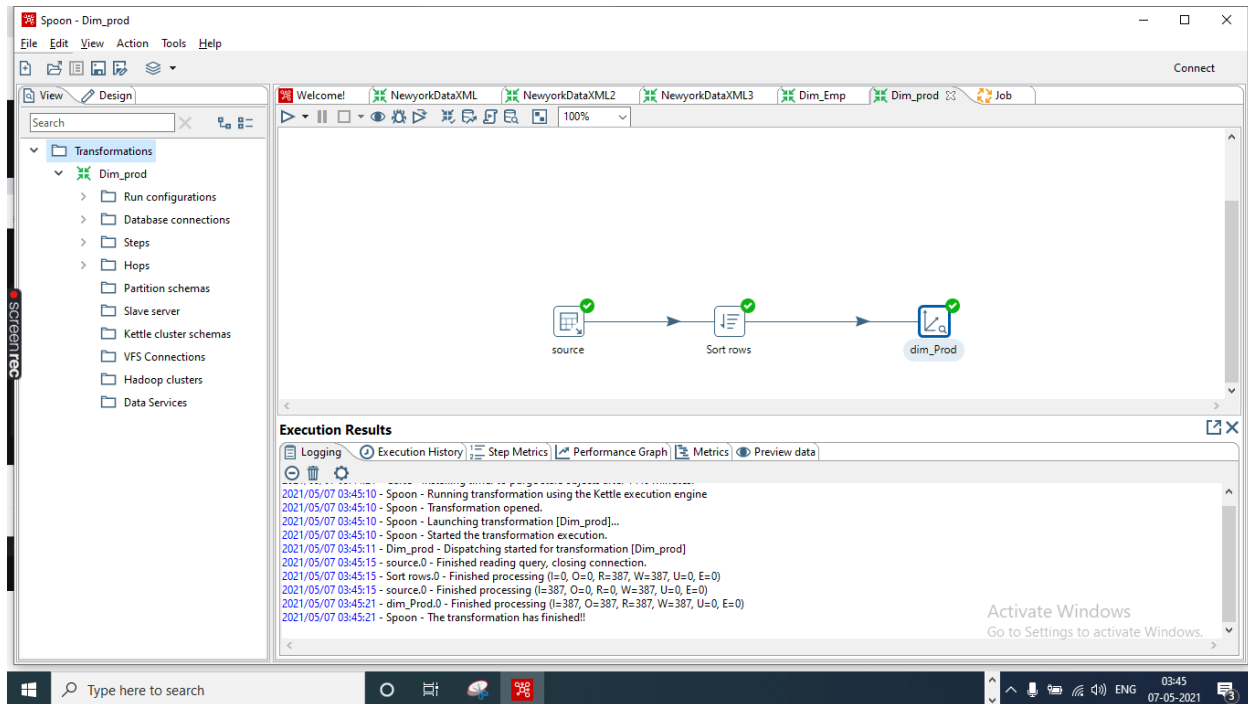
Use an alternative start date? ☐ <Select Option>

Help

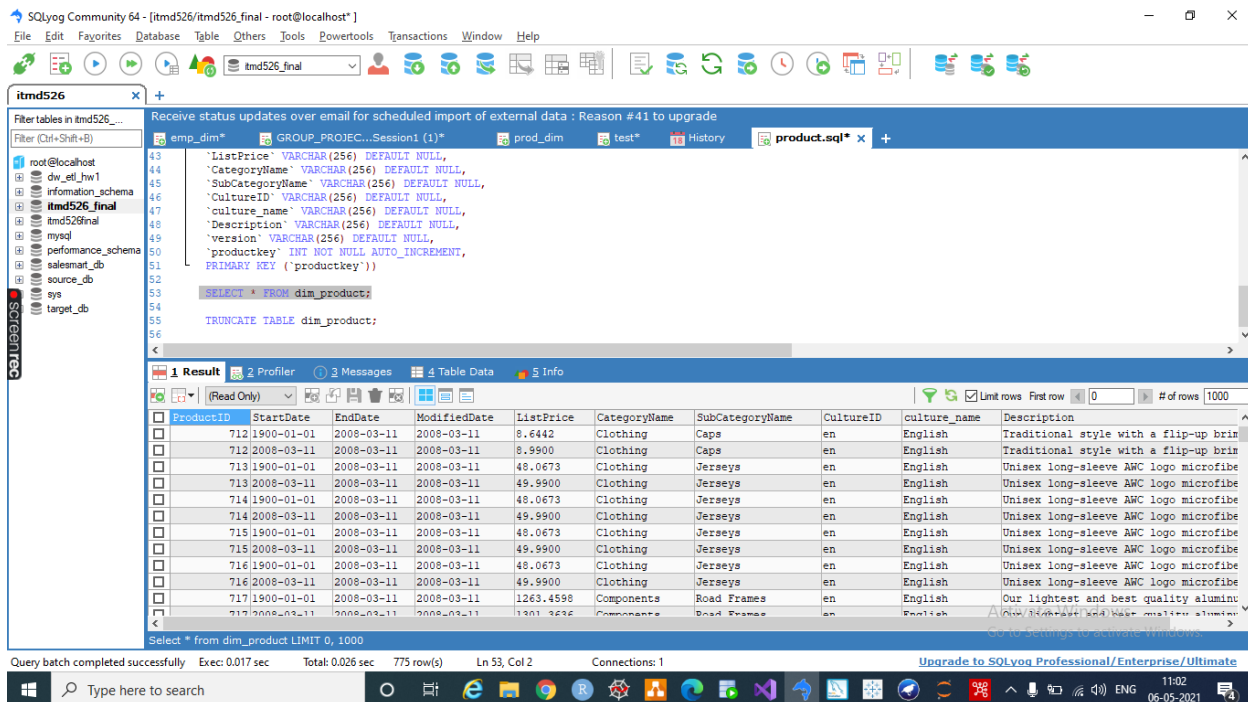
Department of Information Technology and Management

ITMD 526 – Data Warehousing

Result:



Snapshot of Table after running kettle Transformation:



Department of Information Technology and Management

ITMD 526 – Data Warehousing

SQLYog Community 64 - [itmd526/itmd526_final - root@localhost*]

Receive status updates over email for scheduled import of external data : Reason #41 to upgrade

```
emp_dim* GROUP_PROJECT...Session1 (1)* prod_dim test* History product.sql* +
43 'ListPrice' VARCHAR(256) DEFAULT NULL,
44 'CategoryName' VARCHAR(256) DEFAULT NULL,
45 'SubCategoryName' VARCHAR(256) DEFAULT NULL,
46 'CultureID' VARCHAR(256) DEFAULT NULL,
47 'culture_name' VARCHAR(256) DEFAULT NULL,
48 'Description' VARCHAR(256) DEFAULT NULL,
49 'version' VARCHAR(256) DEFAULT NULL,
50 'productkey' INT NOT NULL AUTO_INCREMENT,
51 PRIMARY KEY ('productkey'))
52
53 SELECT * FROM dim_product;
54
55 TRUNCATE TABLE dim_product;
56
```

1 Result 2 Profiler 3 Messages 4 Table Data 5 Info

CategoryName	SubCategoryName	CultureID	culture_name	Description	version	productkey
Clothing	Caps	en	English	Traditional style with a flip-up brim; one-size fits all.	1	10
Clothing	Caps	en	English	Traditional style with a flip-up brim; one-size fits all.	2	11
Clothing	Jerseys	en	English	Unisex long-sleeve AWC logo microfiber cycling jersey	1	12
Clothing	Jerseys	en	English	Unisex long-sleeve AWC logo microfiber cycling jersey	2	13
Clothing	Jerseys	en	English	Unisex long-sleeve AWC logo microfiber cycling jersey	1	14
Clothing	Jerseys	en	English	Unisex long-sleeve AWC logo microfiber cycling jersey	2	15
Clothing	Jerseys	en	English	Unisex long-sleeve AWC logo microfiber cycling jersey	1	16
Clothing	Jerseys	en	English	Unisex long-sleeve AWC logo microfiber cycling jersey	2	17
Clothing	Jerseys	en	English	Unisex long-sleeve AWC logo microfiber cycling jersey	1	18
Clothing	Jerseys	en	English	Unisex long-sleeve AWC logo microfiber cycling jersey	2	19
Components	Road Frames	en	English	Our lightest and best quality aluminum frame made from the newest a	1	20
Components	Road Frames	en	English	Our lightest and best quality aluminum frame made from the newest a	2	21

Select * from dim_product LIMIT 0, 1000

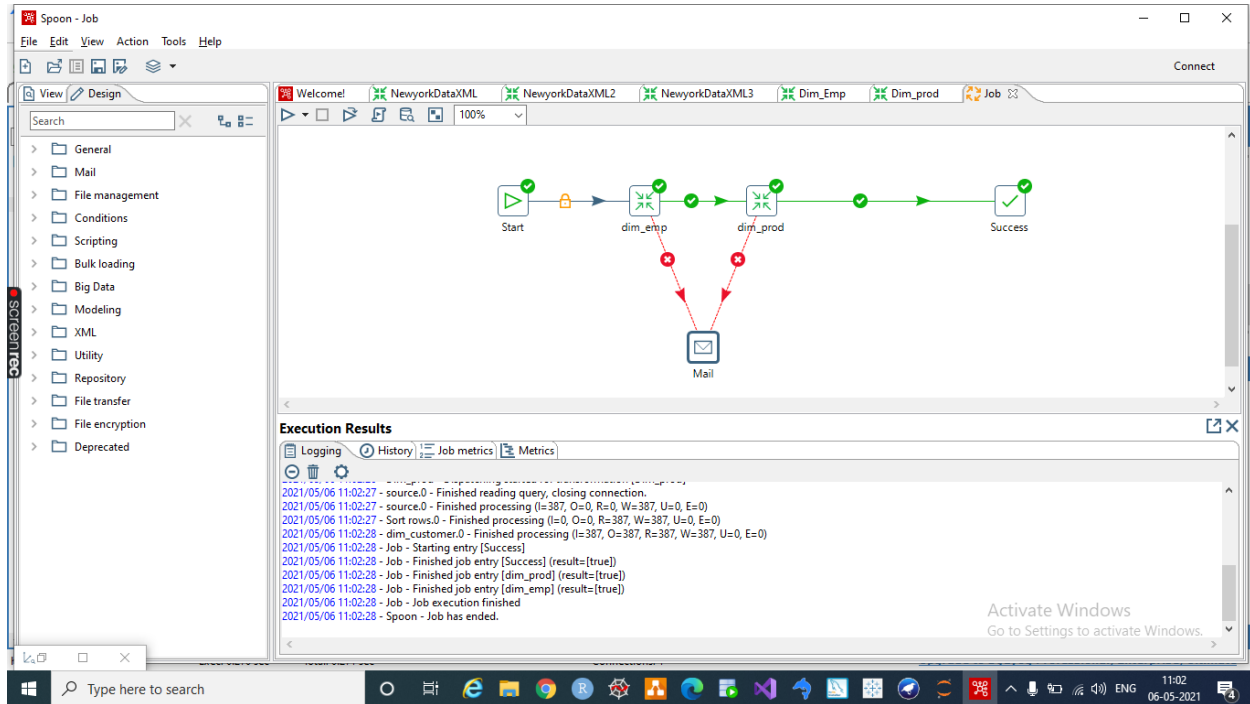
Query batch completed successfully Exec: 0.017 sec Totat: 0.026 sec 775 row(s) Ln 53, Col 2 Connections: 1 Upgrade to SQLYog Professional/Enterprise/Ulimate

Kettle Job:

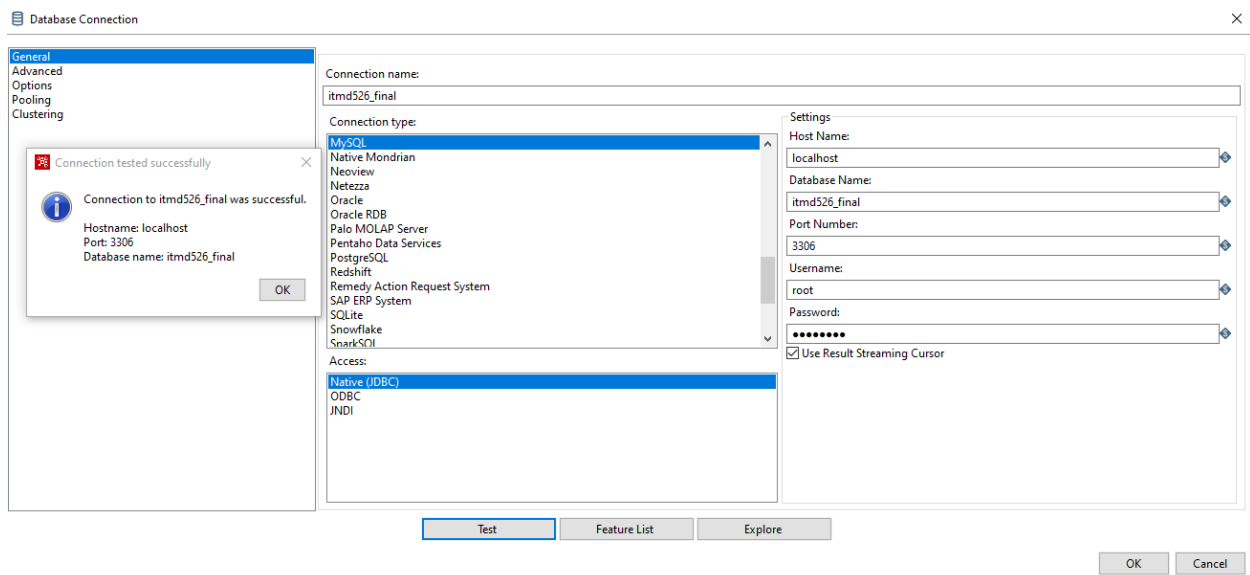
Job file runs both the employee and product transformation and provides success message when both the transformation is completed successfully else it automatically sends error message as an email to the corresponding receiver.

Department of Information Technology and Management

ITMD 526 – Data Warehousing



Database Connection: (Database Name: itmd526_final)



CONCLUSION:

Employee and Product dimension has been created and denormalized using SQL queries and SCD Type-2 is implemented using ETL Transformations.