Implementing Multiple Fact tables in OBIEE Repository file

By

Amit Sharma

Many of time people ask me some way to create the data model consists of multiple Fact table so I though I should create a simple example for people who are new to OBIEE. The tutorial follows step by step approach to create a Data Model (RPD) with multiple fact tables.

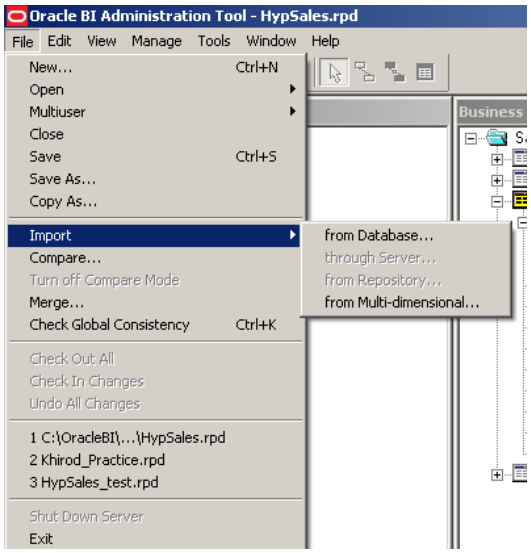
The example I'm going to use has two fact tables and three dimension tables.

Dimension Table : Period_days, Products, Stores, Region
Fact Tables : Sales_Fact, Cost_and_Prices_fact.

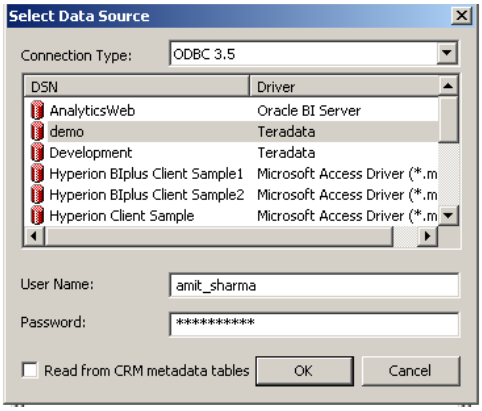
To understand the concept clearly you can download the RPD file and MS-Access DataSouce for the same from my blog..

Physical Layer :Picture Data Source with Multiple Fact Tables.

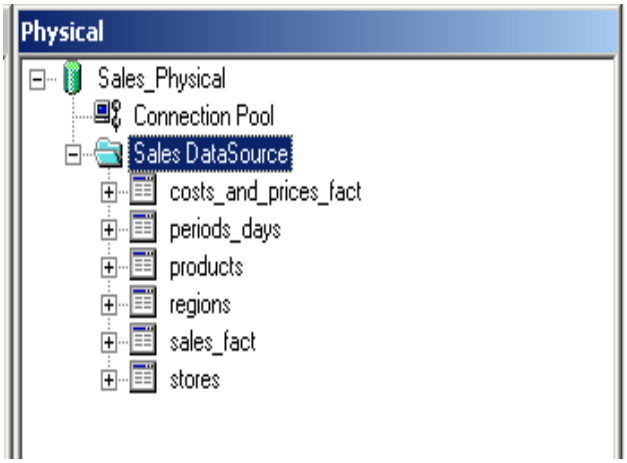
Step #1 Import the Metadata Source File->Import->Data Source



Step #2 Select the DSN and give the user credential to connet to Data Source



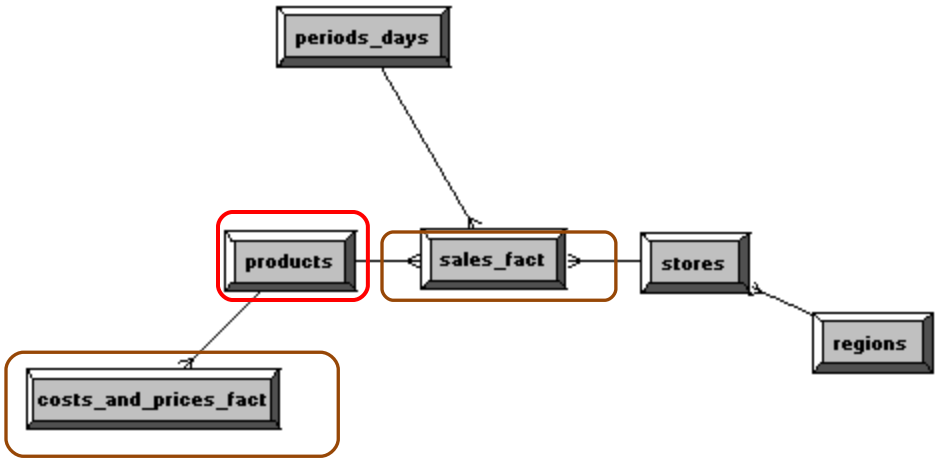
Step#3 Select only the below tables from the data source.[Delete the remaining one for this example we don't need anyother tables.]



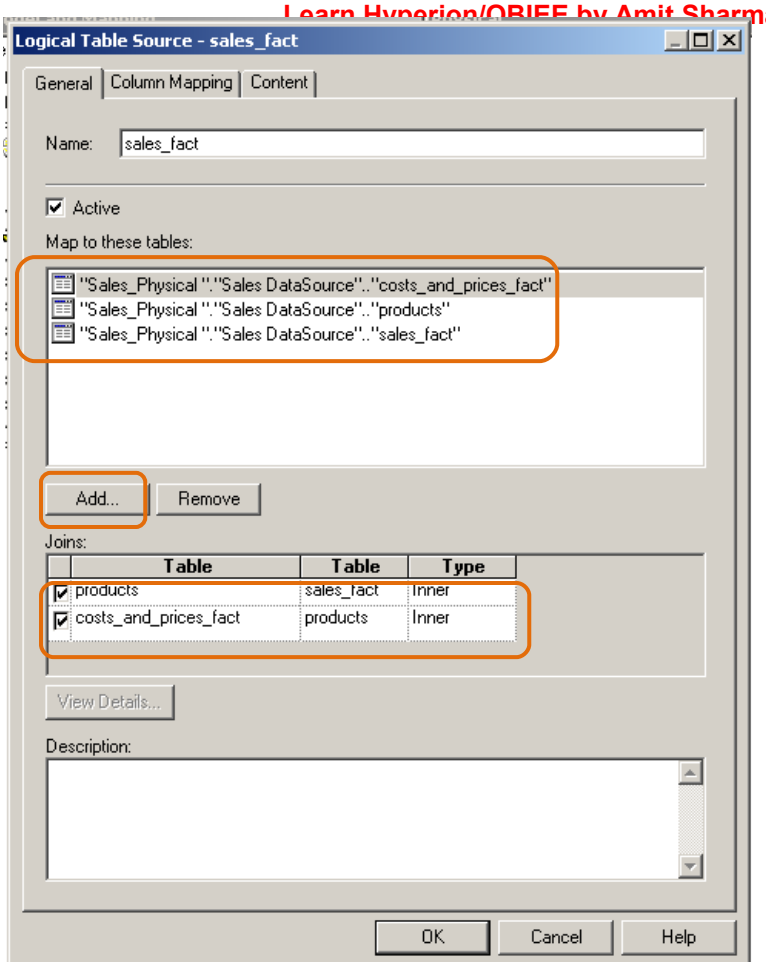
Step#4 : Now drag the Dimension and facts tables in BMM layer as given in the below diagram.

As we have two fact tables and we suppose to have a star schema. We need to combine two fact tables into one. This we'll do by adding a addition sources in Sales Fact Table. Just drag and drop the **Cost_and_Prices_Fact** over the source of **Sales_fact**.

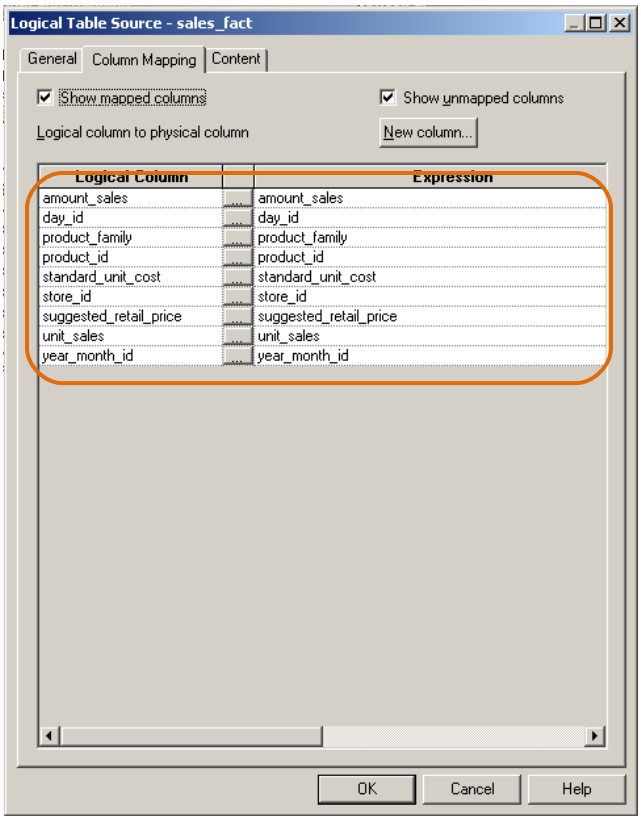
As seen in the below diagram the **Product** dimension is a confirm dimension(Bridge Dimesion)

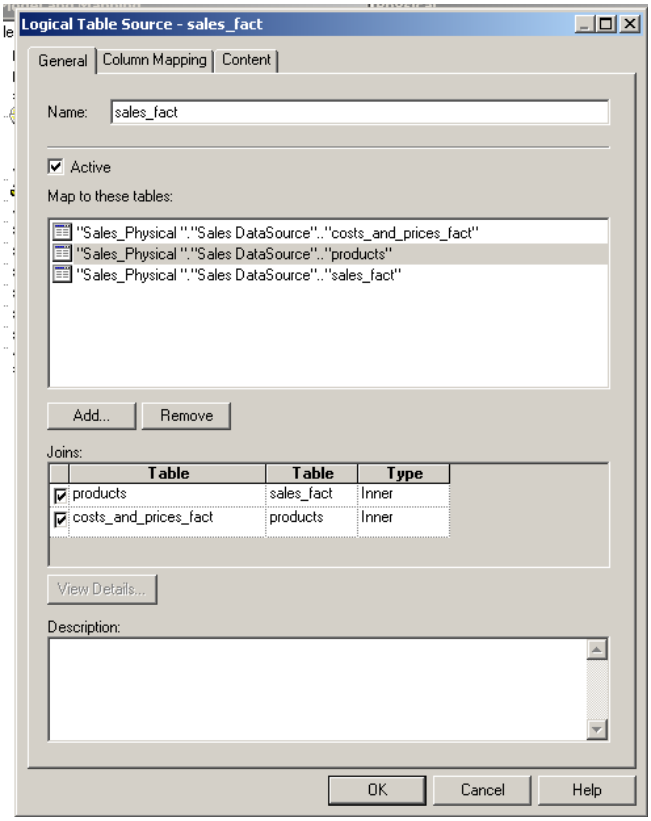


Step#5 : The next step is to add logical sources(double click on logical source) of sales_fact

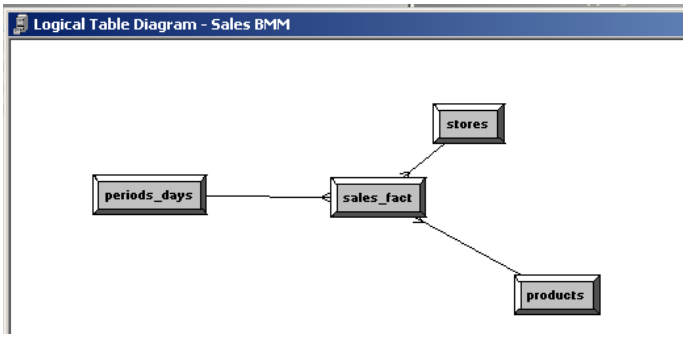


Step#6 : Make sure all columns are mapped correctly.

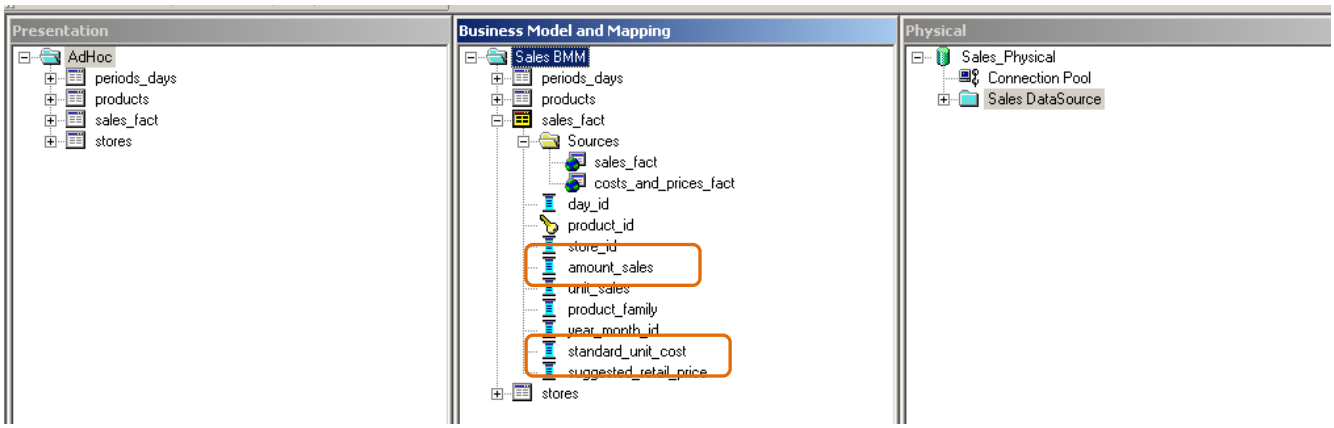




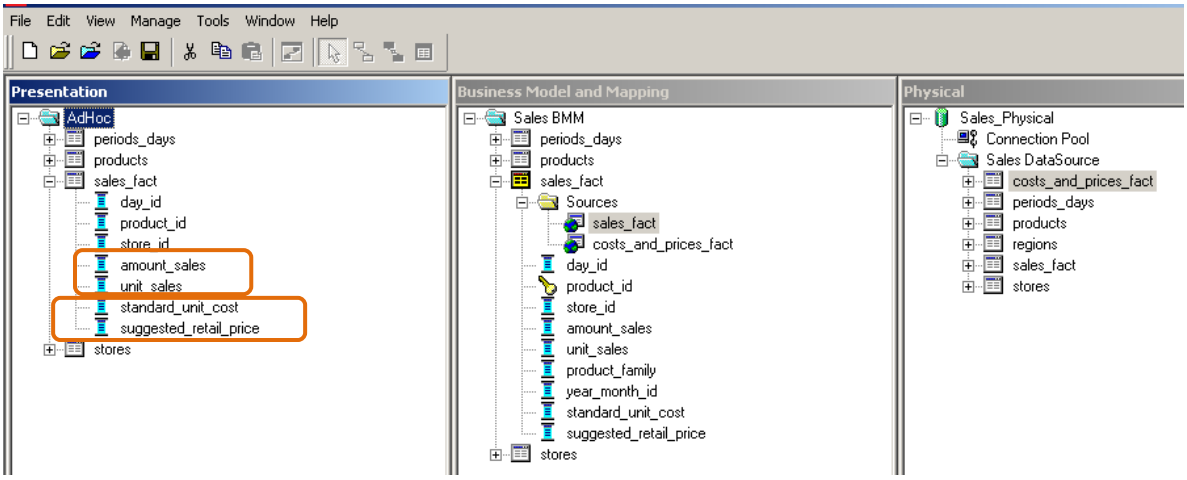
Step#7: Finally you're BMM Logical Model should look like this



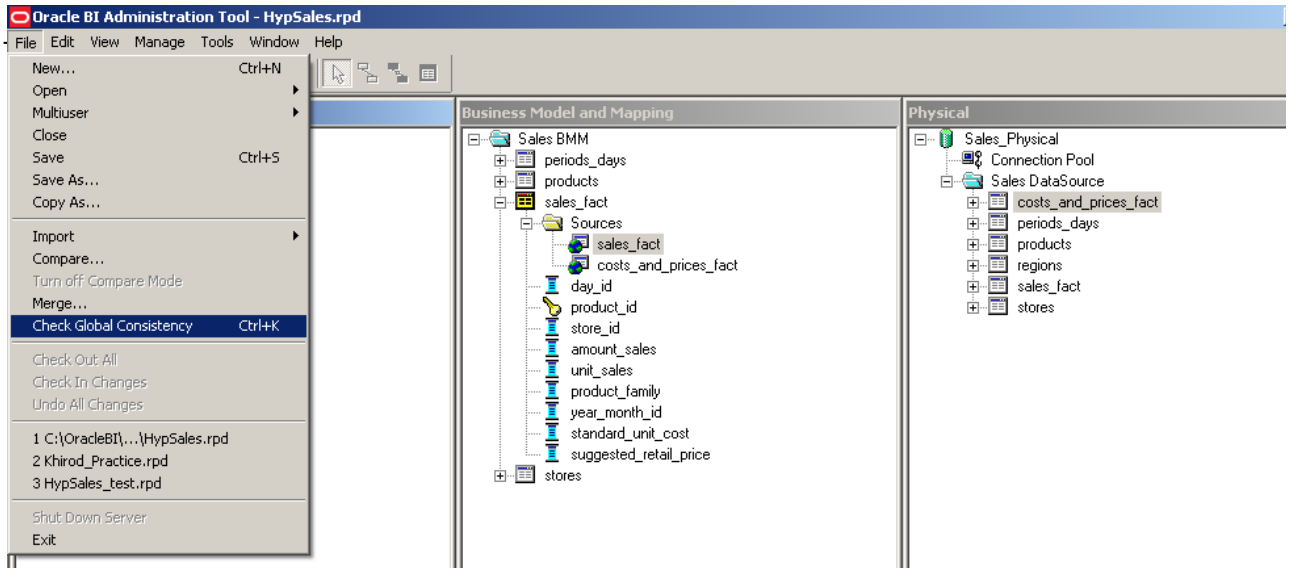
Step#8: In the below diagram is it crealy shown that the measures are coming from two different fact tables.



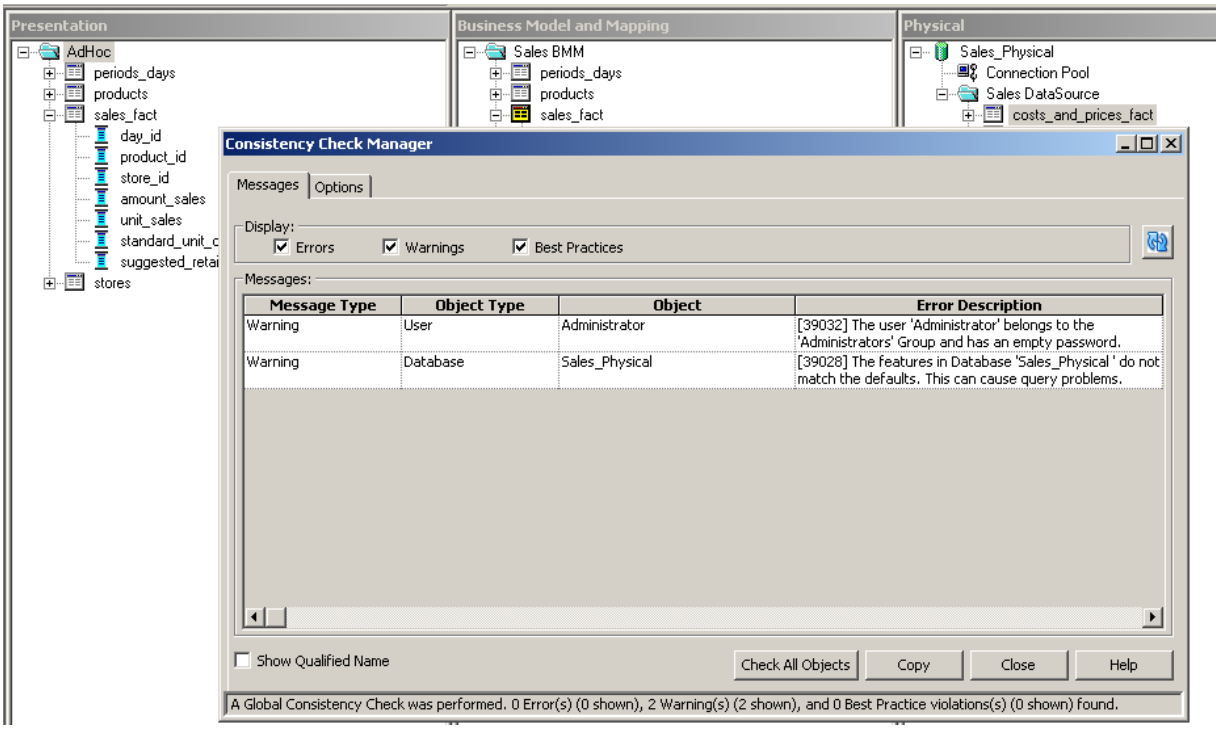
Step#9: Now simple drag the BMM Sales BMM to Presentation Layer.



Step#10 The next step is to run the "Consistency Check"



Step#11: Make sure there is no error.



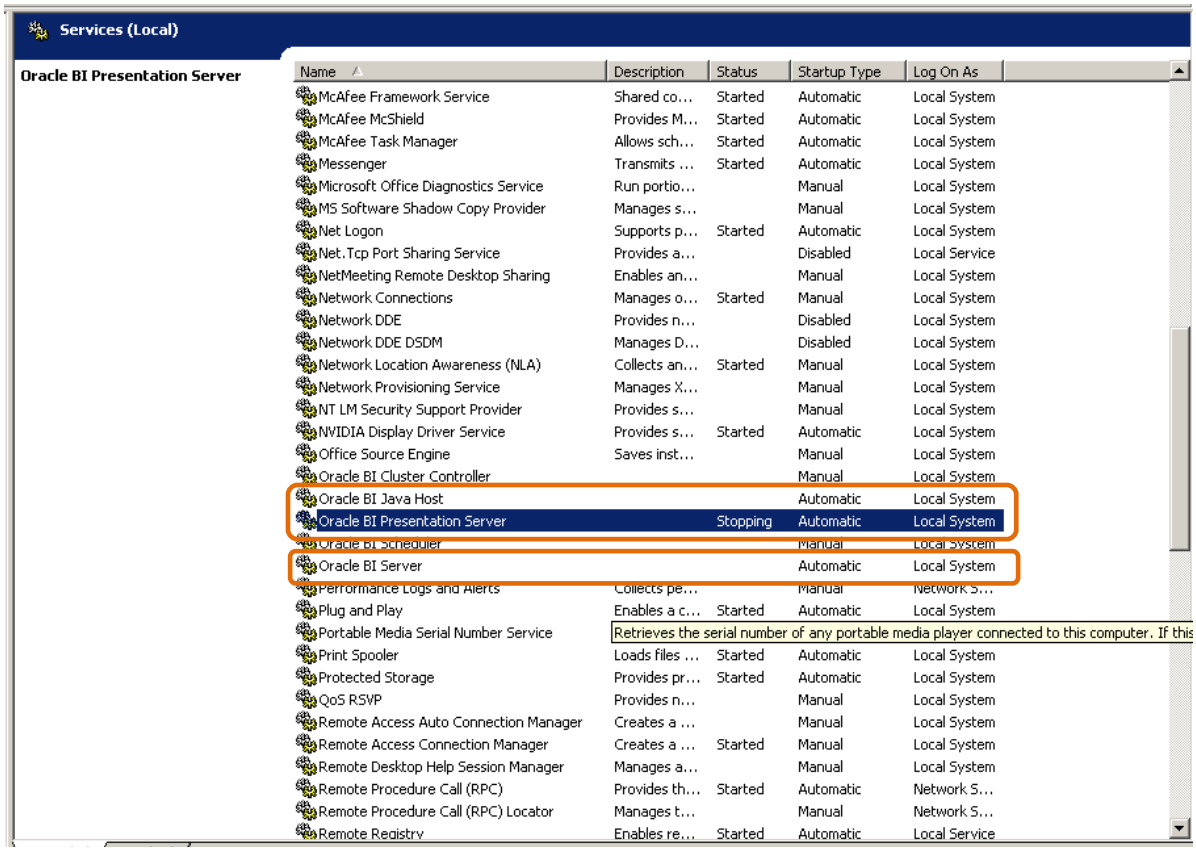
Step#12: In order to deploy the RPD file we need to make the changes in NQSConfig.INI file as given below in the diagram.

NQSConfig.INI - Notepad

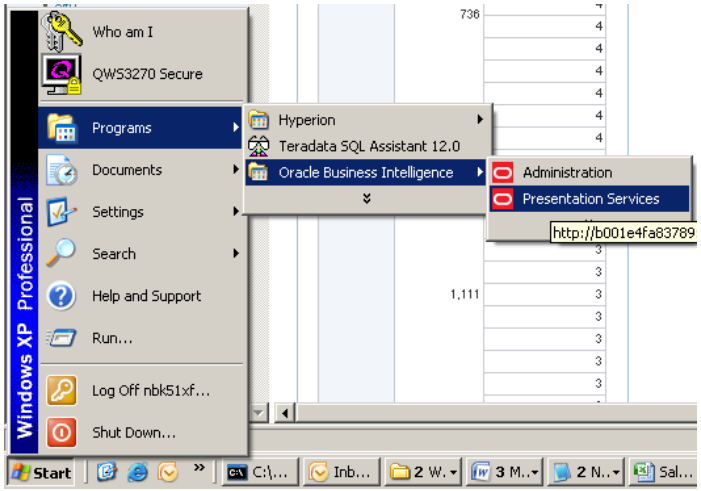
File Edit Format View Help

```
################################################################################
#
#  NQSConfig.INI
#
#  Copyright (c) 1997-2006 Oracle Corporation, All rights reserved
#
#  INI file parser rules are:
#  If values are in literals, digits or _, they can be
#  given as such. If values contain characters other than
#  literals, digits or _, values must be given in quotes.
#
################################################################################

################################################################################
#
#  Repository Section
#
#  Repositories are defined as logical repository name - file name
#  pairs.  ODBC drivers use logical repository name defined in this
#  section.
#
#  All repositories must reside in OracleBI\server\Repository
#  directory, where OracleBI is the directory in which the Oracle BI
#  Server software is installed.
#
################################################################################

[ REPOSITORY ]

Star    =       HypSales.rpd, DEFAULT;

################################################################################
#
#  Query Result Cache Section
#
################################################################################

[ CACHE ]

ENABLE  =       YES;
// A comma separated list of <directory maxSize> pair(s)
//   e.g. DATA_STORAGE_PATHS = "d:\OracleBIData\nQSCache" 500 MB;
DATA_STORAGE_PATHS      =       "C:\OracleBIData\cache" 500 MB;
MAX_ROWS_PER_CACHE_ENTRY = 100000;  // 0 is unlimited size
MAX_CACHE_ENTRY_SIZE = 1 MB;
MAX_CACHE_ENTRIES = 1000;
POPULATE_AGGREGATE_ROLLUP_HITS = NO;
USE_ADVANCED_HIT_DETECTION = NO;

MAX_SUBEXPR_SEARCH_DEPTH = 7;
```

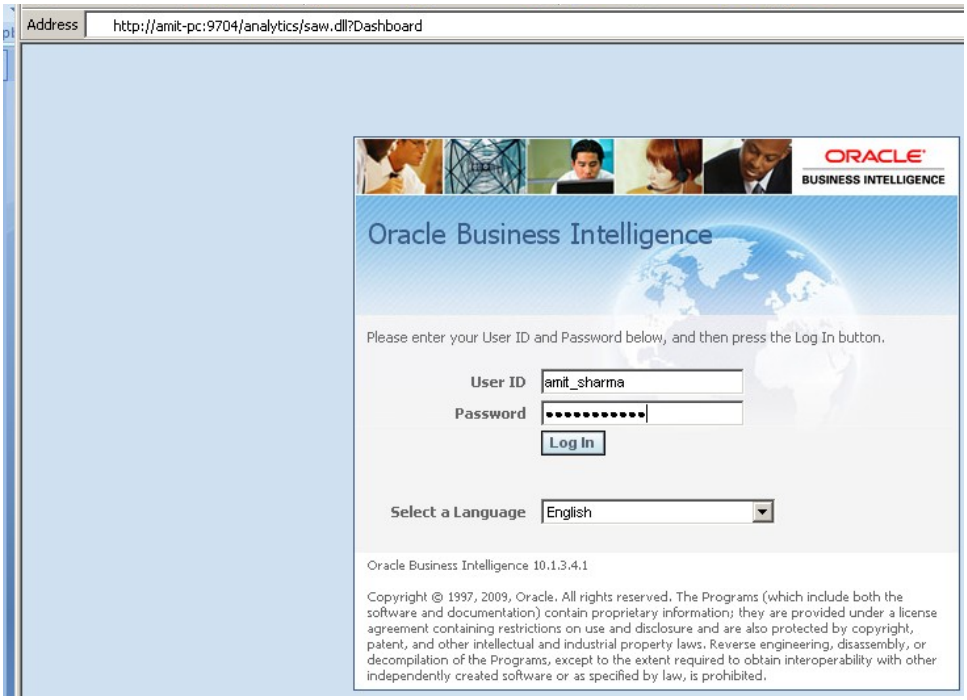
Step#13: Now restart the services.



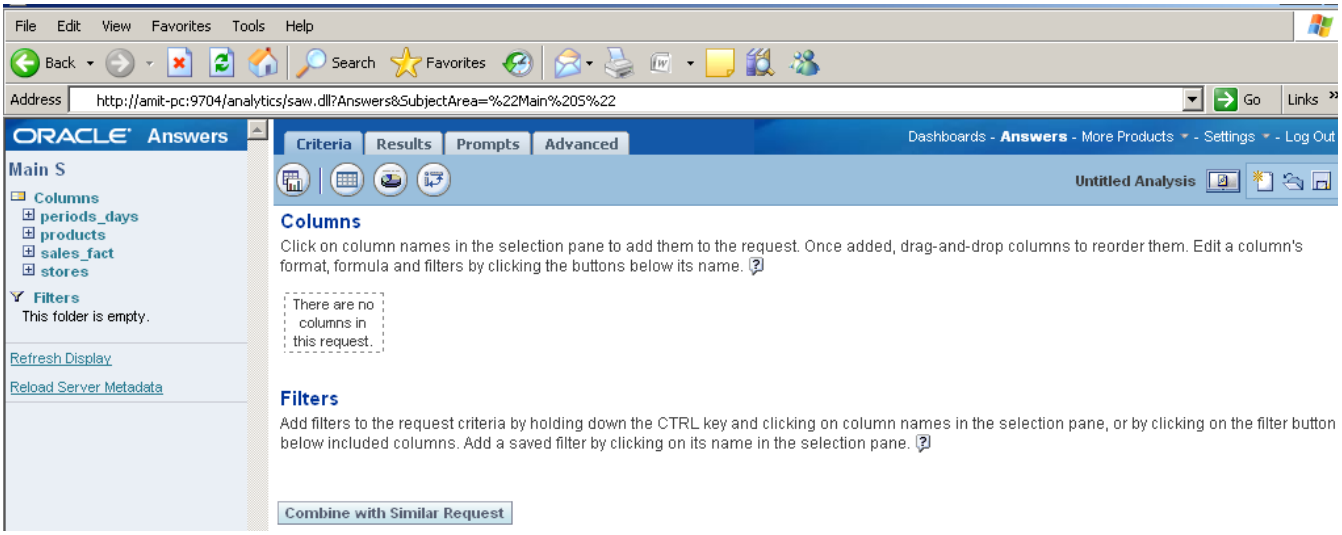
Step#14: Finally we'll run the Answer and check the query is working correct.



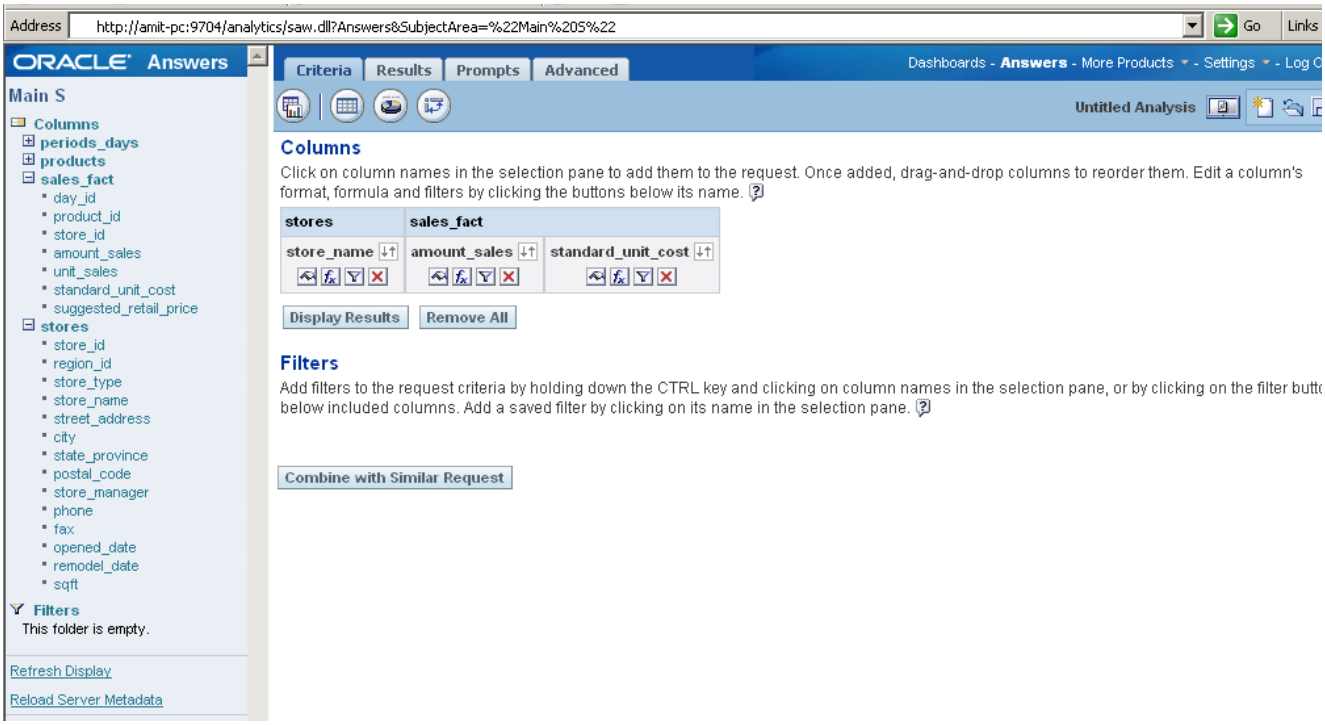
Step#15: Login to the BI Answer.



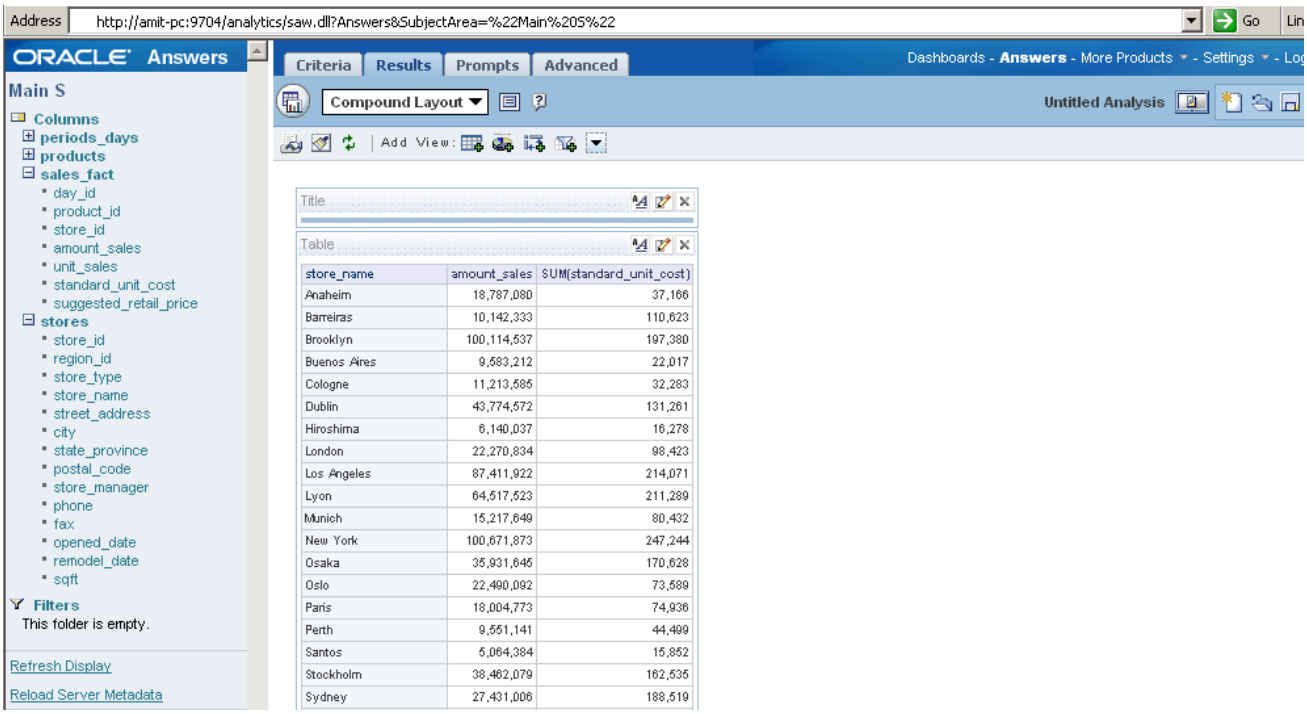
Step#16: The presentation layer may look like this.



Step#17: Drag and drop the column from the presentation layer and create query like this.

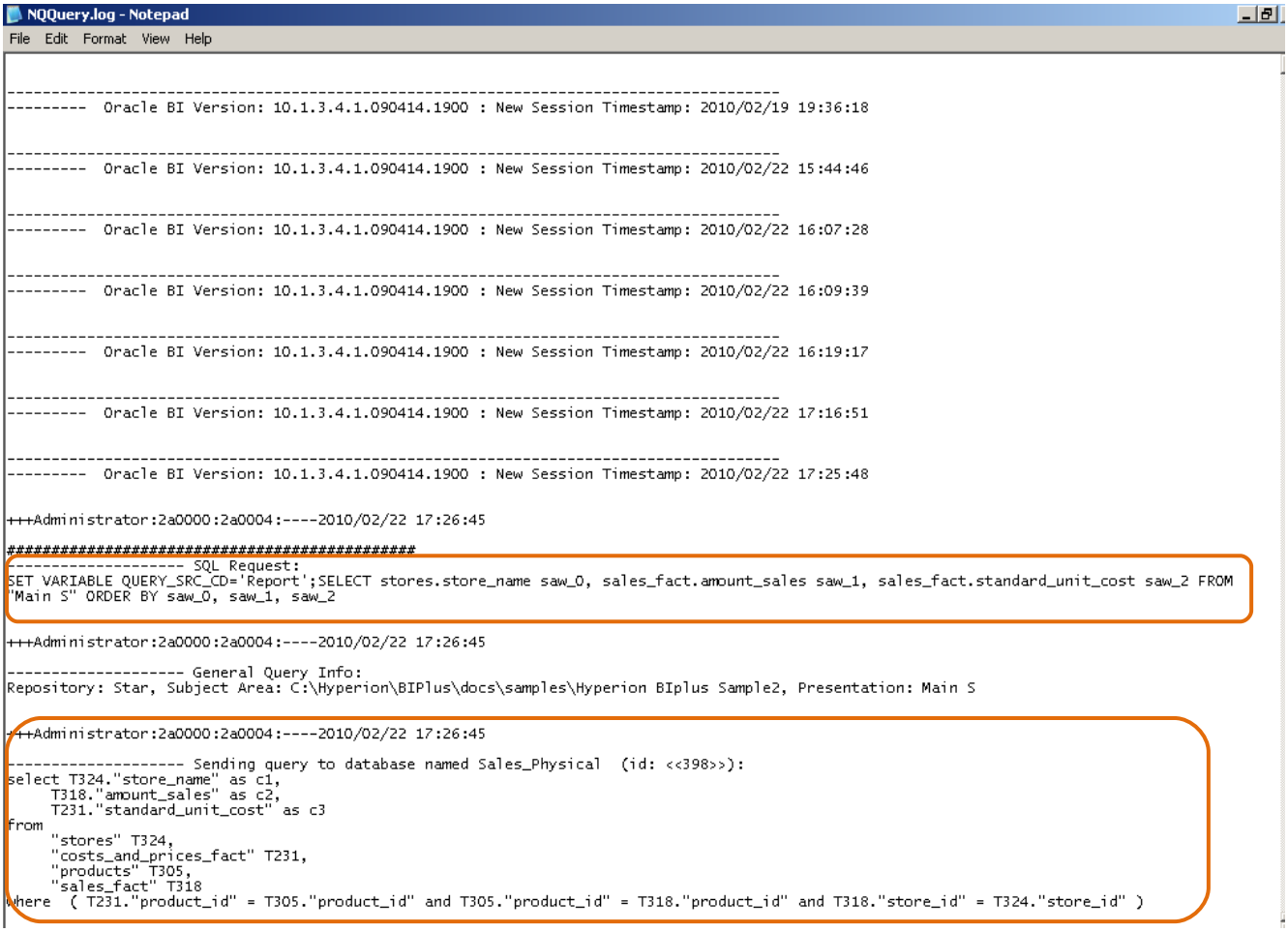


Step#18: Process the query by clicking on Results section to get the output like this.



store_name	amount_sales	SUM(standard_unit_cost)
Anaheim	18,787,080	37,166
Barreiras	10,142,333	110,623
Brooklyn	100,114,537	197,380
Buenos Aires	9,583,212	22,017
Cologne	11,213,585	32,283
Dublin	43,774,572	131,261
Hiroshima	6,140,037	16,278
London	22,270,834	98,423
Los Angeles	87,411,922	214,071
Lyon	64,517,523	211,289
Munich	15,217,649	80,432
New York	100,671,873	247,244
Osaka	35,931,645	170,628
Oslo	22,490,092	73,589
Paris	18,004,773	74,936
Perth	9,551,141	44,499
Santos	5,064,384	15,852
Stockholm	38,462,079	162,535
Sydney	27,431,006	188,519

Step#18 Now let us see the query the BI Server fires again Database.




```
--------------------------------------------------------------------------------
---------  Oracle BI Version: 10.1.3.4.1.090414.1900 : New Session Timestamp: 2010/02/19 19:36:18

--------------------------------------------------------------------------------
---------  Oracle BI Version: 10.1.3.4.1.090414.1900 : New Session Timestamp: 2010/02/22 15:44:46

--------------------------------------------------------------------------------
---------  Oracle BI Version: 10.1.3.4.1.090414.1900 : New Session Timestamp: 2010/02/22 16:07:28

--------------------------------------------------------------------------------
---------  Oracle BI Version: 10.1.3.4.1.090414.1900 : New Session Timestamp: 2010/02/22 16:09:39

--------------------------------------------------------------------------------
---------  Oracle BI Version: 10.1.3.4.1.090414.1900 : New Session Timestamp: 2010/02/22 16:19:17

--------------------------------------------------------------------------------
---------  Oracle BI Version: 10.1.3.4.1.090414.1900 : New Session Timestamp: 2010/02/22 17:16:51

--------------------------------------------------------------------------------
---------  Oracle BI Version: 10.1.3.4.1.090414.1900 : New Session Timestamp: 2010/02/22 17:25:48

+++Administrator:2a0000:2a0004:----2010/02/22 17:26:45

############################################
-------------------- SQL Request:
SET VARIABLE QUERY_SRC_CD='Report';SELECT stores.store_name saw_0, sales_fact.amount_sales saw_1, sales_fact.standard_unit_cost saw_2 FROM
"Main S" ORDER BY saw_0, saw_1, saw_2

+++Administrator:2a0000:2a0004:----2010/02/22 17:26:45

-------------------- General Query Info:
Repository: Star, Subject Area: C:\Hyperion\BIPlus\docs\samples\Hyperion BIplus Sample2, Presentation: Main S

+++Administrator:2a0000:2a0004:----2010/02/22 17:26:45

-------------------- Sending query to database named Sales_Physical  (id: <<398>>):
select T324."store_name" as c1,
     T318."amount_sales" as c2,
     T231."standard_unit_cost" as c3
from
     "stores" T324,
     "costs_and_prices_fact" T231,
     "products" T305,
     "sales_fact" T318
where  ( T231."product_id" = T305."product_id" and T305."product_id" = T318."product_id" and T318."store_id" = T324."store_id" )
```

As it is clearly visible in the log file that the Actual SQL query which fires again RDBMS consists of three tables (Two fact tables and one dimension table), however the logical query shows as if there is only one table exist(Main S Subject Area) in the above example.