Capstone Project 2: Oracle® and Access® -- Linked Tables

THE infinite actuary.com

Learning Without Bounds

Linking tables -- procedure

- Open Access® database or create a new one
- Click on “External Data”
- Click on “ODBC Database”
- Click on “Link to the data source by creating a linked table”, then “OK”
- Select your DSN
- Select the table(s) you want
- Click “Cancel” when Access® asks for a unique record identifier

Working with linked tables

- Anything you can do with a table you create in Access®, you can do with a linked table
  - Open table
  - Filter table
  - Write queries against table, including joining to other tables
  - Rename table within Access® (does not affect table name in Oracle®)
  - Depending on the privileges granted to you in Oracle®, you may be able to add, delete, edit records – DON'T DO THAT PLEASE

- Let's try it out in Access®, then we can talk about some guidelines for working with linked tables

Cautions when using linked tables

- When you write queries using Access® SQL, they are translated into Oracle® PL/SQL – sometimes **very** poorly
- When you join Oracle® tables to Access® tables, you are making Access® evaluate the join(s) and you are forcing a lot of data to travel over the network
- Access® does a very poor job of optimizing – you can’t take advantage of the fact that the Oracle® server has much more processing power than your PC
- Access® does not support many of the functions available in Oracle® PL/SQL

Things to avoid with linked tables

- Note: these are not hard and fast rules, sometimes there is no reasonable way around them
  - Don’t join Access® and Oracle® tables where the purpose of the Access® table is to determine which rows you want from Oracle®
  - Don’t write update queries including Oracle® tables
  - Avoid using custom VBA functions in queries against Oracle® tables where the result of the function is used in a “where” or “having” clause

Best Practices

- Do:
  - Write queries against Oracle® tables (preferably pass-thru) to summarize and filter data using the big server, then use that query as the source in a second query joined to an Access® table or
  - Summarize data in a query that creates a local Access® table where you will be running multiple different queries against that table
  - Develop a feel for the tradeoff between the time it takes you to develop the most efficient queries versus the time it takes to run inefficient queries