Understanding DataEase
Reports -- Query By Model

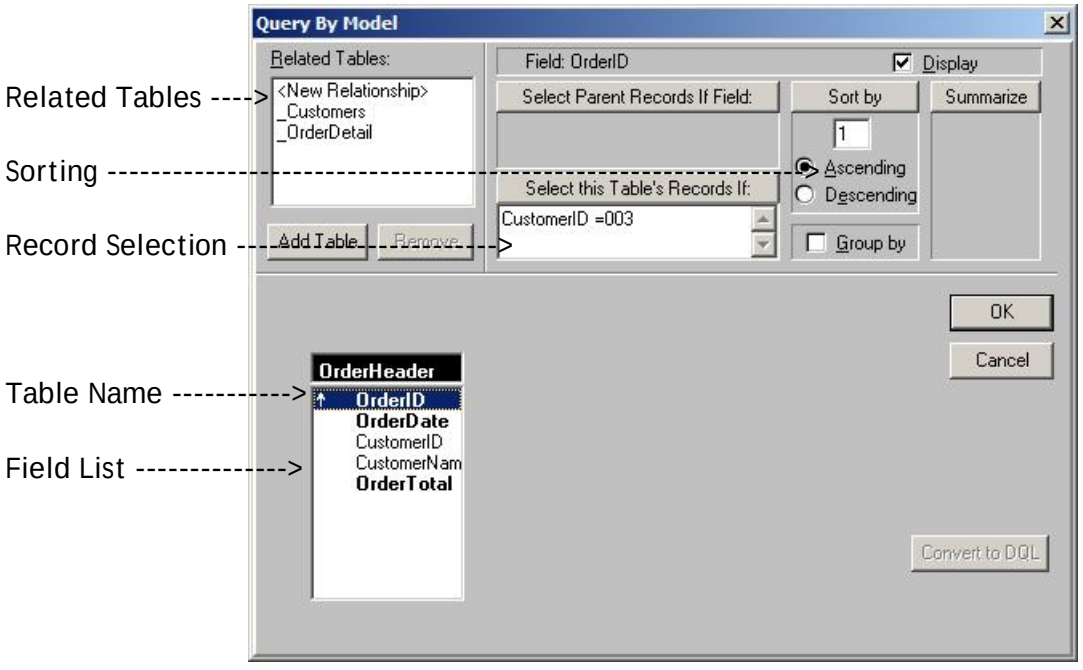
Graham Smith
PLM Consulting, Inc.
gsmith@plmconsulting.com
July 2006

Introduction

Query by Model (QBM) Reports are the most basic method for creating a report in DataEase. In fact, in the original version of this product (DataEase Express), they were the only way to create a report. The reason for this is found in an idea that is quite foreign to the normal way of thinking for people who have a background in DataEase DOS (DFD).

DataEase Forms and Reports appear to be two different types of documents but are, in reality, variations of the same thing. The only real difference between a form and a report is that a report cannot be used for data entry. This is a very important concept to understand since both forms and reports are built on top of a data model which is visualized and manipulated by means of the QBM dialog box.

Following is an example QBM dialog box from a Report:



Unfortunately, for the newcomer, the QBM dialog can be rather daunting since there is a lot going on in it. But once you start to work through it, you will find that each section is fairly easy to understand. The important thing is that this dialog represents the entire data model that the form is built on. This includes things like subforms, data filters, sorting, grouping, and summary fields (the latter are usually only used in reports). Because it does all these things, it cannot help but be somewhat complicated. Following is a brief description of each of the annotated sections:

Table Name – this is the name of the primary table in the document
Field List – the highlighted fields will appear in the document (click Display to highlight)
Sorting – specify the sorting sequence and order for fields (an arrow will appear next to sorted fields in the field list)
Record Selection – any selection criteria used to filter the records displayed
Related Tables – this is a list of related tables that can be added to the model

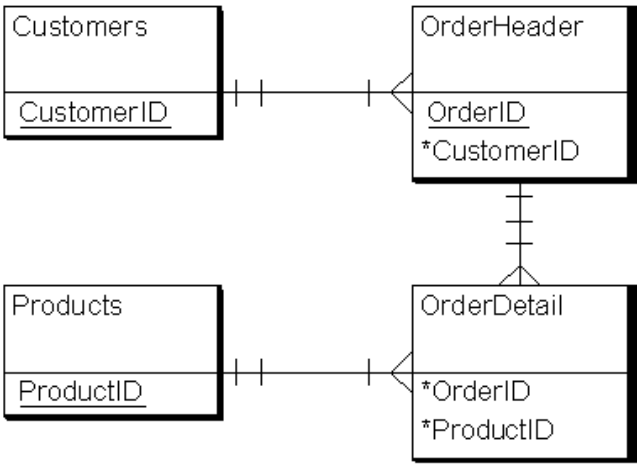
It is important to know how the data is structured when creating a multi-table query so that you can know where to begin in building the query. But it is not the purpose of this document to substitute for in-depth training in database design and reporting. Rather it is to merely introduce the user to the basic ideas found in the Query by Model tool.

The best way to learn QBM is by doing. There is nothing you can do to alter the data or in any way harm the application, so feel free to jump in and get your feet (and ankles, legs, etc) wet. Just bear in mind that the only thing you are doing is describing a different view of the data – you cannot see something that is not there already.

To understand the concepts behind a multi-table QBM, you must first understand a bit about relational databases and how data structures are modeled or diagrammed. To do this, we will turn to a classic data model, Order Entry.

There are four data tables in this model:

Customers
OrderHeader
OrderDetail
Products



Each table has a primary key field (PK) which is unique. This field is underlined in the diagram. The PK is the most important field in the table as it is the basis of all relationships.

Each table is related to one other table by way of a foreign key (FK) This field is marked with an * in the diagram.

Relationships are between tables using the primary key and foreign key fields. Relationship are designated by a line between the tables.

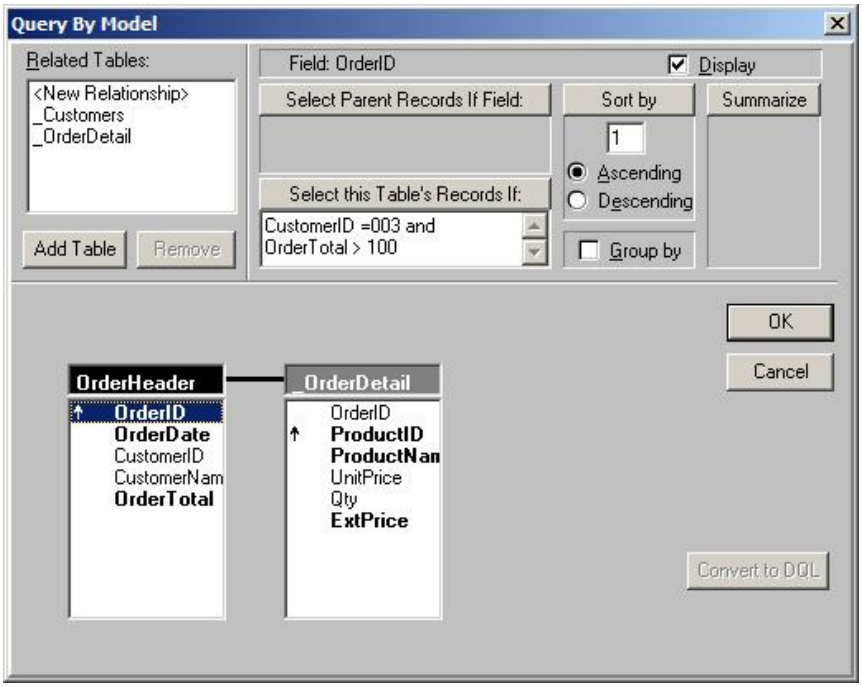
The relationships displayed are all one-to-many with the forked end of the line denoting the many side of the relationship. This means that for every record in the table with the primary key, there can be many records in the table with the foreign key.

To put this in more straight forward terms:

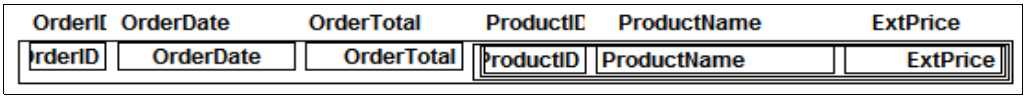
- A Customer may have only one Customer record
- A Customer may have many Orders
- An Order may have many Detail records
- Each Detail record is related to one Product record
- A Product may have only one Product record

Let's say that we wanted to create a report for Customer 003 that lists all their Orders including the Detail but only for Orders > $100.00. To do this we need data from two tables: OrderHeader and OrderDetail. We also need to know which Relationship between the two to use. In this case, the relationship we want is named _OrderDetail. In cases where there are more than on relationship between the same two tables, you will need to know which one to use

Following is an example QBM dialog box from this new report on the OrderHeader table that includes OrderDetail as a subform:

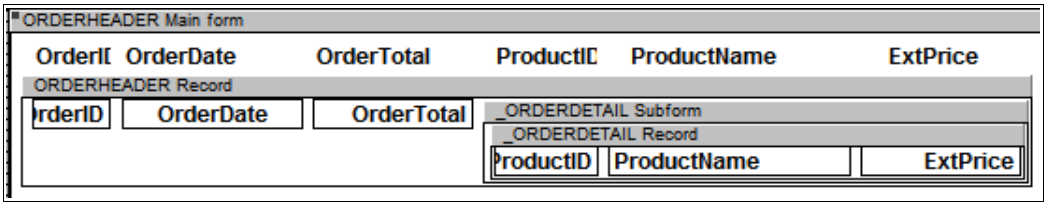


The initial layout of the report will appear something like this:



For DFD users who are used to writing DQLs, you should note that to achieve the same result, you would have had to group on OrderID. However, grouping is not needed as often in DataEase since a query with a related subform such as this creates a natural grouping.

If you look at the layout you will see that the fields are inside boxes that are often referred to as "containers". These containers correspond to the tables seen in the QBM. To make it clearer as to what the various containers represent, here is the same query with Form and Record Labels turned on:



The outermost container is the main form for OrderHeader (note: this is badly named as it is really the main table not form).

Inside that is the record container for OrderHeader. In a report, this will almost always be laid out as a single record, but it could also be setup to display two records side-by-side.

Inside the OrderHeader record container is the OrderDetail subform (subtable) and inside that is the OrderDetail record. Once again, this is usually going to be a single record layout in a report but would more likely be set to display multiple records if this were a form document.

Running the report would produce a result somewhat like this:

OrderID	OrderDate	OrderTotal	ProductID	ProductName	ExtPrice
001	12/12/2004	1,010.00	00001	Apples	300.00
			00002	Oranges	350.00
			00003	Pears	360.00
002	12/12/2004	700.00	00003	Pears	180.00
			00004	Bananas	520.00

What You Want, Where You Want It

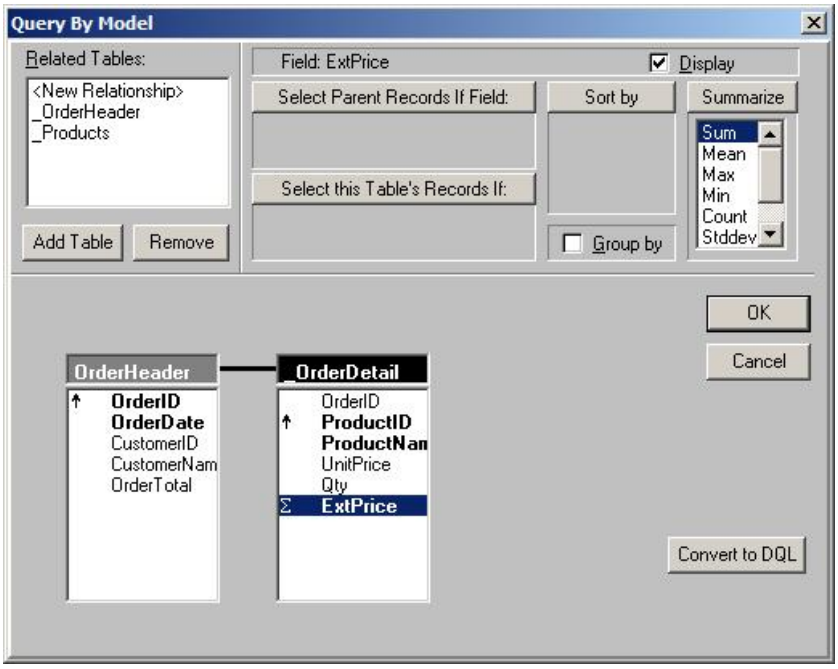
This concept, developed by Simon Irwin from Sapphire, is key to deciding which fields to choose and how to choose them when creating a QBM, particularly a multi-table QBM. Here are some simple rules:

- The fields will appear in the layout in the order in which you choose them in the QBM. For example, in the above QBM, the fields were marked for display (by double clicking them) in the same order you see them in the layout. If OrderDate had been selected before OrderID was, then that is the order they would have appeared in the layout.
- In the case of a multi-table query, the table that the field is selected in dictates which container the field will appear in. For example, in the above QBM, if OrderID had been marked in the OrderDetail item, it would have appeared in that container rather than in the OrderHeader container.
- You cannot move a field from one container to another. This goes hand in hand with the previous rule. It is particularly important in deciding in which container to create any summary fields.

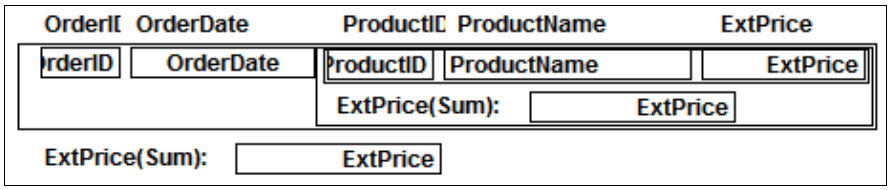
The whole idea behind this is to set up the QBM in exactly the way you want the report to layout, thus minimizing the amount of rearranging you have to do.

Summary Values

In a report that includes columns of numbers, it is common to want to do so sums as well. This can be done by selecting a field and clicking on the Summarize button in the QBM dialog then picking the summary type you wish. This will place a sigma symbol (∑) next to the field. Following is a variation of the above report that eliminates the OrderTotal field and replaces it with a sum on the ExtPrice field:



This will produce a layout that looks like this:



And when run, an output that looks like this:

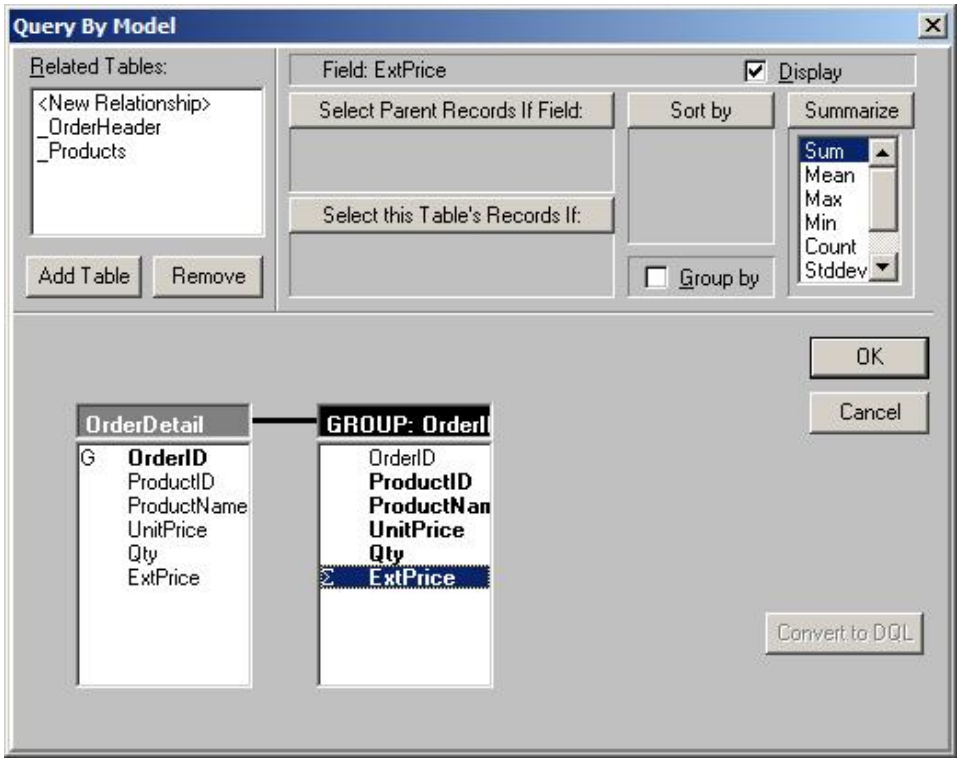
OrderID	OrderDate	ProductID	ProductName		ExtPrice
001	12/12/2004	00001	Apples		300.00
		00002	Oranges		350.00
		00003	Pears		360.00
		ExtPrice(Sum):		1,010.00	
002	12/12/2004	00003	Pears		180.00
		00004	Bananas		520.00
		ExtPrice(Sum):		700.00	
ExtPrice(Sum):		1,710.00			

Needless to say, all these reports can be cleaned up a bit to align things better, expand the display of the labels a bit where needed, and add some lines. Even though these are all cosmetic changes, it is important to remember that a report is of no value if it is difficult to understand. If this is just a quick report you need to get an answer, you can let this slide, but if others are going to be viewing it, take a few minutes to neaten things up a bit.

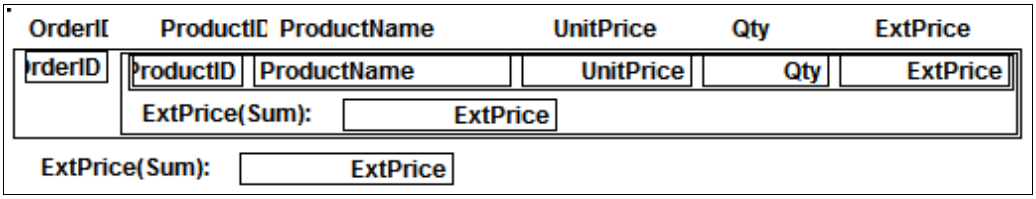
Using Groups in Reports

There are going to be times where you will want to create a report that lists records in groups. Often this is done using a single table so that is what will be demonstrated.

In the following QBM, OrderDetail has been grouped by OrderID. This will produce an outer container with the OrderID and an inner container with all the records that share that same OrderID. A sum has been added which will sum up the ExtPrice for each group and a total for the report.



The report layout will appear like this:



If you were to display form and record labels, you would see that they would be very much the same as was seen in the above queries. That is because grouping creates a kind of pseudo MainForm/Subform construct. This makes grouping fairly easy to use in Reports since you pretty much use the same techniques as you would with a regular MainForm/Subform construct.