

**GQL/User**  
**Version 3.0**  
**Demo Guide**



552 Princess Street  
Kingston, Ontario  
K7L 1C7  
(613) 548-4355  
**1-800-267-0665**

GQL, GQL/Admin, GQL/Design, GQL/User are trademarks of Andyne Computing Limited.  
Apple, AppleTalk, Laserwriter, Macintosh, MacTCP, and HyperCard are registered trademarks of Apple Computer, Inc. Finder and MultiFinder are trademarks of Apple Computer, Inc.  
INGRES is a trademark of Ask/Ingres Corporation.  
ORACLE and SQL\*Net are registered trademarks of Oracle Corporation.  
Wingz is a trademark and Informix is a registered trademark of Informix Software, Inc.  
VAX, Rdb, VMS, DECnet, and RMS are trademarks of Digital Equipment Corp.  
DB2, SQL/DS, AS/400, and IBM PC are registered trademarks of International Business Machines Corporation.  
UNIX is a registered trademark of AT&T.  
Tandem and NonStop are trademarks of Tandem Computer, Inc.  
Post-it is a registered trademark of 3M.  
SYBASE SQL Server is a trademark and SYBASE and DB-Library are registered trademarks of Sybase, Inc.  
Ethernet is a registered trademark of Xerox Corporation.  
AlisaTalk is a registered trademark of Alisa Systems Inc.  
SequeLink is a trademark of TechGnosis Inc.  
DBC/1012, and Teradata are registered trademarks of Teradata Corporation.  
Microsoft is a registered trademark of Microsoft Corporation.

All other trademarks are those of their respective owners.

Copyright © Andyne Computing Limited 1991.  
552 Princess St., 2nd Floor.  
Kingston, Ontario  
Canada K7L 1C7  
All rights reserved.

Printed in Canada, January 1992.

# Table of Contents

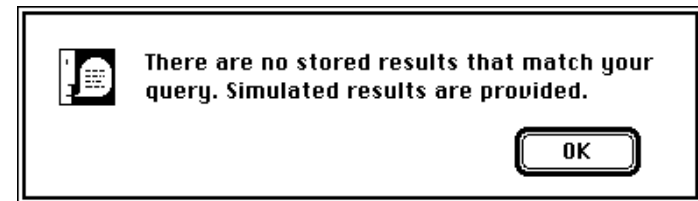
GQL/Demo Package Overview	4
Installation	5
Getting Started	6
Ad hoc Query Tool or EIS Application?	7
EIS (Executive Information System)	8
Integrated Report Module	12
Ad hoc Queries	13
The Results Window	15
Qualifications and Sorting	17
Full Menus	18
Data Values	18
Using Relationships	21
Saving Queries (Query Documents)	22
Executive Buttons	23
Qualification Prompts	26
Getting started with System 7.0	27
Publish/Subscribe (using GQL and Excel)	28
Other Features	31
Building GQL Data Models	32
GQL - The Corporate Data Access Solution	32
Appendix A - The GQL Product Family	33
Windows 3.0	34
Unix/Motif	34

# GQL/User Demo Overview

The GQL/User Demo package contains a special version of the GQL/User application that will allow you to become familiar with GQL concepts and to demonstrate how GQL can be used to access host database information. The demo version differs from the actual product in one major way - you don't need to be connected to the host computer while running the demo. In other respects, using the GQL/User Demo package is just like using GQL/User.

If you are not familiar with the GQL family of products you may wish to glance at Appendix A for a brief description of the product family.

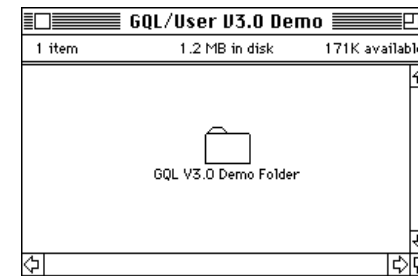
These notes will take you through a sequence of steps that introduce GQL/User features to you. Stored results are used to provide data for all of the queries described in this booklet. You may want to deviate from the scripted session. However, since the demo is not accessing a real database management system you won't get actual data back for every query you try. Whenever you generate a query that has no corresponding stored results, GQL will provide simulated data and notify you with the following dialog box.



**If you stray from programmed queries in the Demo version**

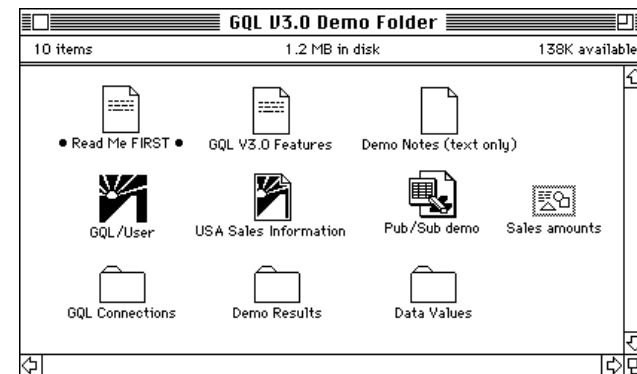
# Installation

The GQL/User Demo is distributed on a single high-density (1.4 MB) diskette, but is recommended that you run the demo from a hard disk. To install, simply drag the folder called *GQL Demo Folder* onto your hard disk.



Contents of the GQL/User Demo Disk

Please keep the contents of *GQL Demo Folder* together as the demo application requires access to the data files to present query results.



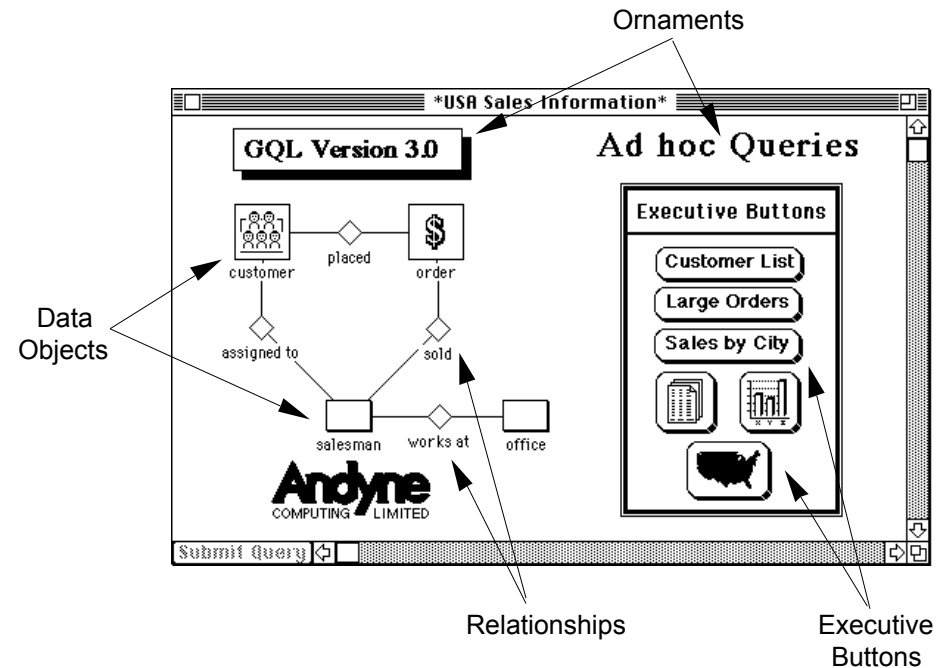
Contents of the GQL Demo Folder

**Note:** If you see the message: "This disk is unreadable: Do you want to initialize?", you are probably trying to install the demo on a system that can't read 1.4MB floppy disks. Use another system to copy the demo onto two 800KB floppy disks, or give us a call and we will send you 800KB floppy disks.

# Getting Started

Start the demo application by double-clicking on the document *USA Sales Information* . This will launch the application *GQL/User (demo)* and open the *USA Sales Information*.

The GQL desktop window will appear as shown in the following figure.



**The *USA Sales Information* Model**

The data model window contains data objects, relationships, executive buttons, and ornaments. The square and rectangular icons are *data objects* that represent tables, or combined tables, that hold information in the database. Data objects are connected to each other by lines that contain *relationship* (diamond shaped) icons. The relationships guide the user through the database so that information can be extracted from more than one data object. The relationships are named in a manner that helps the user to understand how the information in one data object can be combined with information from another data object. However, in some cases you may choose not to show the relationship names, or even display the relationship icons, if you feel it isn't required. In this example the *customer* data object is related to the *order* data object through the *placed* relationship. The *order* data object is related through the *sold* relationship to the *salesman* data object. This approach to the data model guides the user in getting information about the customer and the

salesman by showing the different ways that the information can be related. The user formulates queries by using the data objects and relationships.

*Executive buttons* are used to provide easy push button access to information in the corporate database. Each button can be associated with a stored query (with optional parameters), and an output (a results window, a report, another application, or a file), or can be used to activate another window. It is possible to have buttons that directly activate other applications such as Microsoft Excel to have the data charted or placed into a spreadsheet. Stored queries can be formulated by the user, or created for the user by the MIS staff.

*Ornaments* are text or graphic objects that are placed on the data model as titles, borders, comments, highlights, etc. in order to improve the layout.

## **Ad hoc Query Tool or EIS Application?**

GQL does an excellent job at both! The GQL product family includes a special tool (GQL/Admin) for customizing the interface to meet your needs. Using GQL/Admin, the graphical data model is created, attributes can be renamed to make them more meaningful for the user, customized sets of executive buttons can be created, and the overall layout is defined. The resulting GQL document can be setup as a multi-level executive buttons interface, and ad hoc query interface, or both. The data model can be hidden from users if you wish to create a push button only user environment. GQL has the unique capability of allowing you to customize the interface to suit the different needs and skill levels of different users.

The *USA Sales Information* document is has a combination of both ad hoc and EIS (Executive Interface System) styles. It is designed to give you a feel for the overall product. This demo guide starts off by taking you through an EIS style interface, shows you the integrated report module, and then moves on to ad hoc queries and executive buttons. Once you have gone through the demo you will also want to look through the menus and try some experiments to discover some of the other powerful features.in GQL.

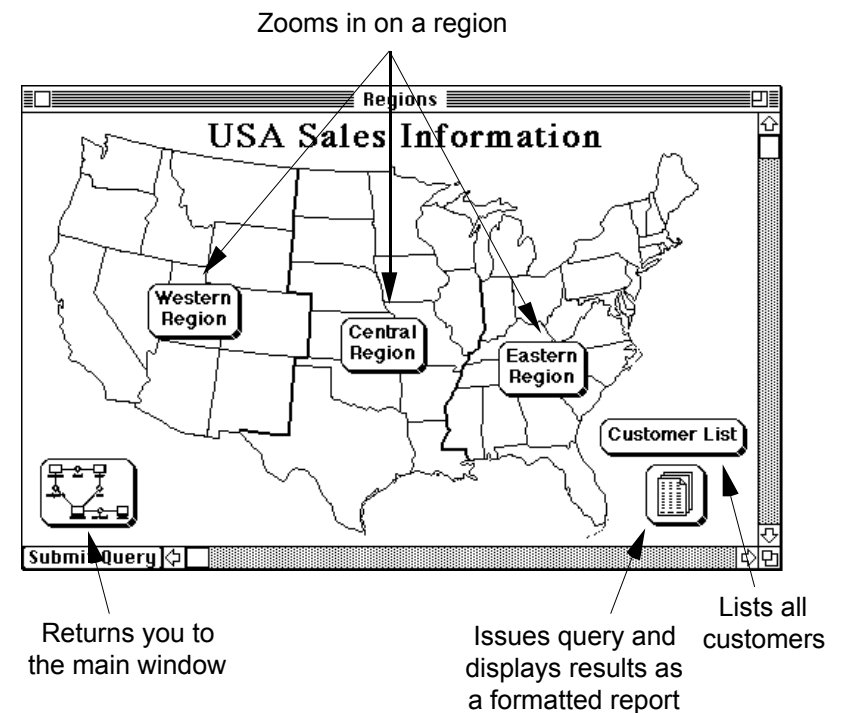
# EIS (Executive Information System)

GQL can also be customized to provide an executive interface system (EIS) style interface. This demo includes a drill-down interface that demonstrates some of these capabilities.

To begin, return to the *\*USA Sales Information\** window and press the executive button with the map of the USA on it,



the following window will appear.



**Sample EIS Interface**

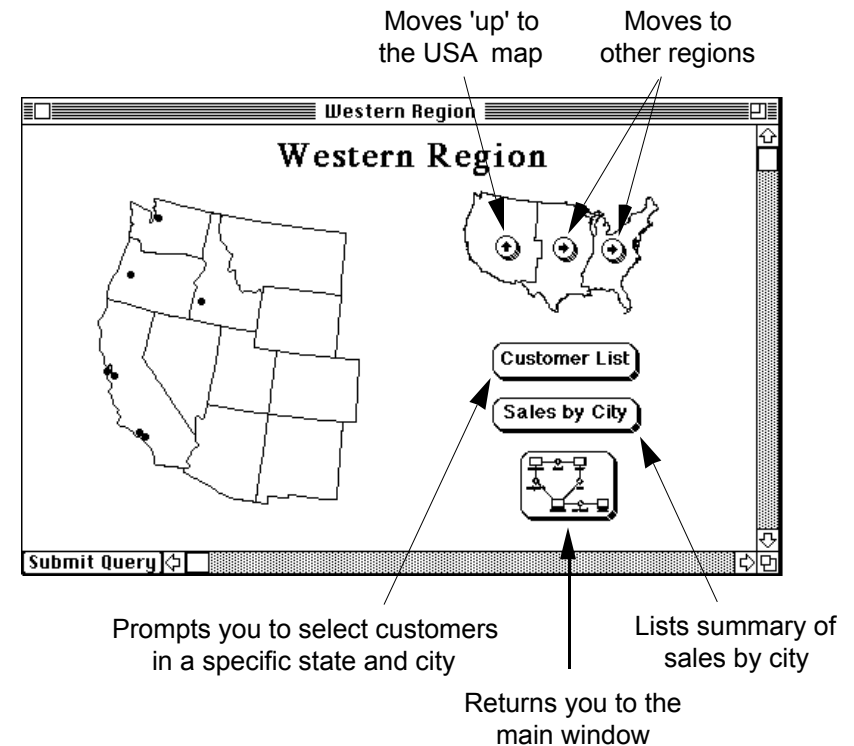
The map of the USA is divided into three sales regions. There are several executive buttons displayed that either take you to other windows, or issue stored queries. Pressing the executive button for a region will 'zoom in' on that region by moving to another window that has regional details. In this example we will look at the Western sales region.



Press the button called Western Region,



the following window will appear.



**Western Region for USA Sales Information**

A map of the western sales region is shown on the left side of the window. The small USA map in the upper right corner has executive buttons with arrows that are used to switch to other regions or return to the main USA map. The button in the lower right corner is used to switch back to the main window for ad hoc queries. The other two executive buttons issue stored queries. We will use the Customer List button to 'drill-down' to specific information about customers in San Jose, California. GQL provides an extensive prompting facility that allows you to enter qualifying information before a query is issued.

Press **Customer List** button,

**Customer List**

The following dialog will appear.

Provides a pop-up list of values

Please Enter Qualification Values:

State  
WA

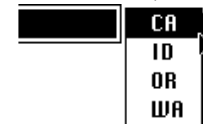
City  
Seattle

Data Values... Cancel OK

Loads a list of values from the database

#### User Prompt for Qualifying Information

This dialog allows you to enter additional information that qualifies or restricts the information that will be retrieved when you submit the query. You can enter information by typing directly into the prompt boxes, or by using data values to select from a list of known values. The values that are initially displayed in the prompt boxes are the last values that were entered. We will use two different ways to access stored values for the prompts. A button with a down arrow is located at the right of the *State* prompt. Press down and hold the mouse button while the cursor is over this button, and a pop-up list of value will appear as follows:



Choose **CA** (for California) from the list. The value for the *State* prompt will be set to CA and the value for the *City* prompt will be highlighted (in preparation for your next entry). In the next example we will actually issue a query to load a list of valid values from the database. The query that retrieves the list of values for *City* uses the *State* information you just entered to only retrieve names of cities in California that are stored in the database.

With the *City* prompt value highlighted, press the **Data Values...** button.. The following list will appear.

**Data Values**

Anaheim  
East Los Angeles  
SFrancisco  
San Jose

Data loaded: 8:39 AM

Load Data Load All

Cancel Insert

#### Data Values for Cities in California

Select *San Jose* from the list and press **Insert** (or double-click on *San Jose*). Now that you have entered the qualifications, press the **OK** button and the query will be submitted. The results will be displayed as follows.

Query Results 3

Query completed: 10:47 AM Fri, Dec 13, 1991  
Elapsed Time: 00:00

	Name	Address 1	City	State
1	Guardian Packaging Corporation	Post Office Box 589	San Jose	CA

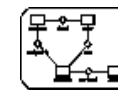
Filter...

#### Results for San Jose, California (with columns expanded)

You have just completed your first query! Using the facilities provided by GQL you were able to 'drill-down' to the specific information we wanted.

Let's take a look at reports , then we'll move on to ad hoc queries.

Return to the *\*USA Sales Information\** window by pressing the executive button that has a data model diagram on it,



or by selecting *\*USA Sales Information\** from the **Windows** menu.

# Integrated Report Module

GQL's report module features an interactive (WYSIWYG) interface that permits end users to create their own reports using up-to-the-minute database information. Frequently used reports can be stored with executive buttons.

Press the executive button with the picture of a report on it,



a stored query will be issued. After the results are returned, the following report will be displayed.

Header Section      Ornaments      Column Headings      Current Date Field

State	City	Date	Amount
ID	Boise	11/15/1984	\$2468.74
		11/15/1984	\$5397.75
		11/30/1984	\$12000.00
		<b>Total Sales</b>	\$19,866.49
IL	Bloomington	12/10/1984	\$23450.00
		<b>Total Sales</b>	\$23,450.00
IN	Indianapolis	12/23/1984	\$2345.75
		01/05/1985	\$2956.84
		06/07/1985	\$3875.44
		05/05/1984	\$15879.89
		11/20/1984	\$22350.00
		<b>Total Sales</b>	\$47,407.92
NY	Albany Englishtown Hempstead	11/25/1984	\$245.67
		12/01/1984	\$2348.34
		12/06/1984	\$120000.00
		07/07/1985	\$543.09
		01/01/1985	\$985.75
		12/14/1984	\$12450.50
		09/07/1994	\$37402.40
		10/13/1984	\$67000.00
		04/23/1984	\$86970.00
<b>Subtotal</b>			

Detail Section      Suppressed Duplicates      Subtotal

USA Sales Summary Report

# Ad hoc Queries

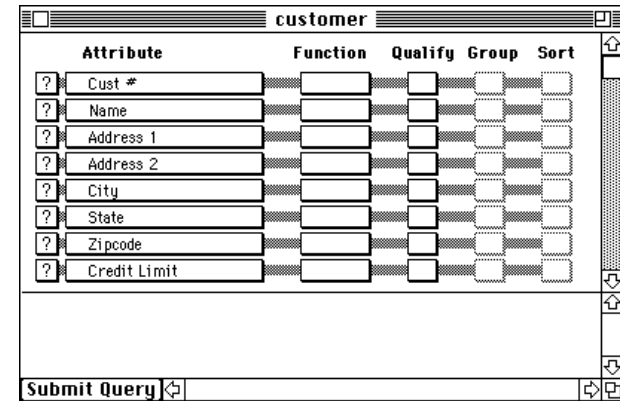
The *USA Sales Information* data model describes a database containing customers, orders and salesmen. Let's make your first ad hoc query to find out who the customers are.

If you are not already at the main ad hoc query window, select *\*USA Sales Information\** from the **Window** menu.

Double click on the *customer* icon



to open the *customer* attribute window.

A screenshot of a software window titled "customer". The window contains a table with five columns: "Attribute", "Function", "Qualify", "Group", and "Sort". There are eight rows of attributes: "Cust #", "Name", "Address 1", "Address 2", "City", "State", "Zipcode", and "Credit Limit". Each row has a small question mark icon to its left. The "Name" row is highlighted, and the "City" row is also highlighted. At the bottom of the window, there is a "Submit Query" button and a small text input field.

**The *customer* Attribute window**

You will want to list the customer names and their cities. Click on the attribute labeled *Name*. Note how *Name* is changed to bold face and a "●" character is put in front to indicate that this field will be included in the results. Click on the attribute labeled *City* to include it in the query as well.

Now your query is ready! Submit the query by pressing the **Submit Query** button at the lower left corner of the window (or by choosing **Submit Query** from the **Query** menu).

Selected Attributes

Attribute	Function	Qualify	Group	Sort
? Cust #				
? • Name				
? Address/1				
? Address/2				
? • City				
? State				
? Zipcode				
? Credit Limit				

Submit Query

Submit Button

Query ready to be submitted

GQL will create a *Query Results 1* window and as data is returned from the host it will fill the window.

Query completed: 4:15 PM Sat, Dec 14, 1991  
Elapsed Time: 00:02

	Name	City
1	Acme Manufacturing Co.	Chicago
2	Flex-O-Glass	Anderson
3	Dennison Distributing	Boise
4	Paulsen Packing	Hempstead
5	Chicago Transparent Products	Amarillo
6	Freds Markets	Hicksville
7	Guardian Packaging Corporation	San Jose
8	Gingiss Inc.	Dallas
9	Graham Industries	Holkind
10	Portion Packaging, Incorporated	Bay City

Filter...


Query Results for the first query

Congratulations, you've formulated your first ad hoc query using GQL!

## The Results Window

The data for your query is displayed in the query results window. This window presents the data in a spreadsheet-like form. Individual cells or groups of cells can be selected directly from the query results window for copying to other applications through the Macintosh clipboard facility.

Before sending data to another application, you may want to rearrange the data or manipulate it in various ways. Many operations on the local query results are supported by GQL. For example, the data in the results window can be sorted locally (without going back to the host database), ranges taken on the data, columns reordered or completely hidden, or the columns resized.

To resize a column, simply move the cursor to the dividing line between a column and its neighbor. (The cursor will look like this: )

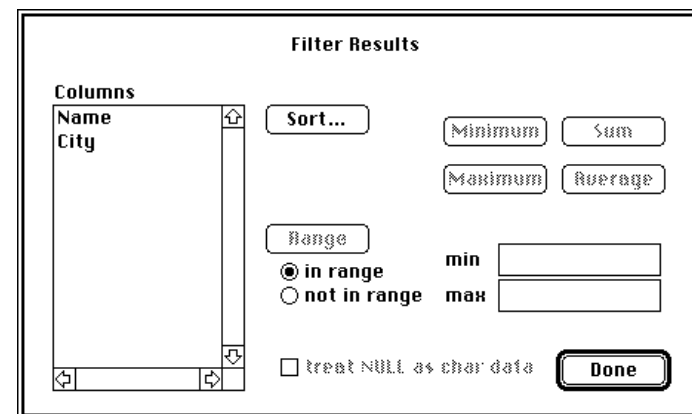
Drag the line right to widen the column, or left to narrow the column.

An alternative method to set the column width is to select a column and then choose **Column Width...** from the **Results** menu. A dialog is displayed, that allows you to enter the desired width or select an option to automatically set the width to the widest data item in the column.

Column ordering can be changed by choosing **Reorder Columns...** from the **Results** menu.

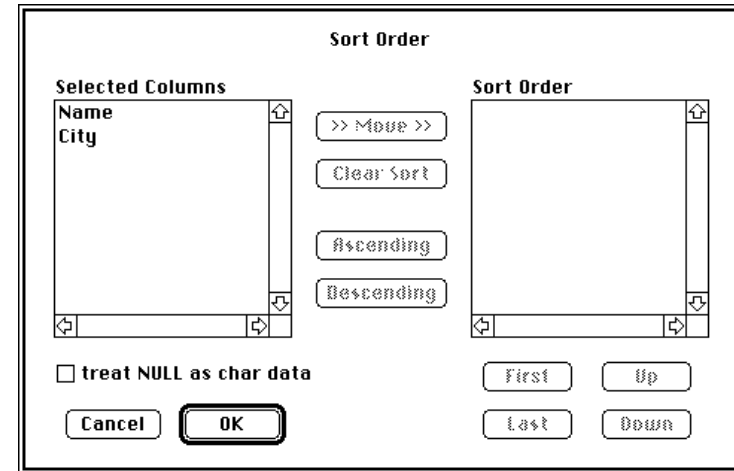
The **Filter** button located at the bottom left corner of the query results window activates the local operations for sorting, range selection and computing minimum, maximum, sum and averages on columns.

Press the **Filter** button to open the Filter dialog.



The Filter dialog

Press the **Sort...** button to open a dialog for selected the sort order.



**The Sort dialog**

Double click on *City* in the Selected Columns list (*City* will move to the Sort Order column), and then click on the *OK* button. This specified that the data should be sorted in ascending order according to *City*. The filter dialogs will disappear, and a second results window, *Query Results 1.1* will be displayed. GQL creates a second window so that you can still refer to the original results from the query. This way you can feel free to manipulate the data locally without concern of losing the original data.

The 'Query Results 1.1' window shows a table with 10 rows. The columns are 'Name' and 'City'. The data is sorted by City in ascending order. The window title is 'Query Results 1.1' and it includes a 'Filter...' button at the bottom.

	Name	City
1	Nicolo Corp	Albany
2	Chicago Transparent Products	Amarillo
3	Clear View Plastics, Inc.	Anaheim
4	Flex-O-Glass	Anderson
5	Sealtran Corporation	Asheville
6	W.R. Grace & Company	Atlantic City
7	Portion Packaging, Incorporated	Bay City
8	Hotpoint Division	Biloxi
9	Holcum Mfg.	Bloomington
10	SAT Plastics Corporation	Bloomington-Normal

**Results sorted by City**



The data in the results window can be saved to a file and subsequently read into other applications. You can even arrange to save the data with another file type so that double-clicking the document will open the desired application.

## Qualifications and Sorting

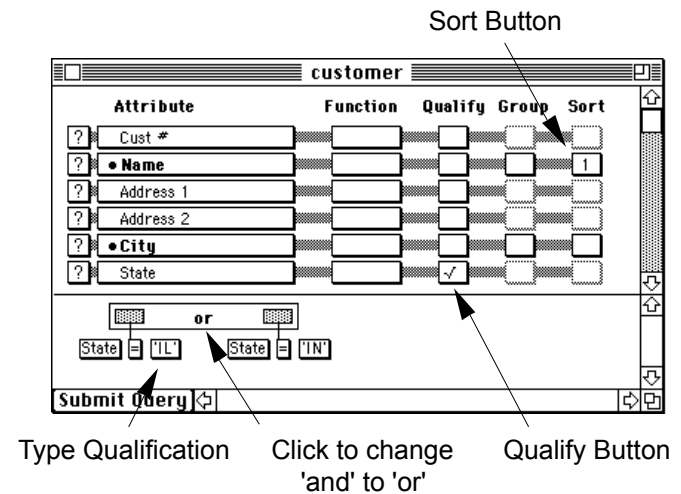
As is often the case with an ad hoc query, you may decide to change the query once you've seen the data. In this case you may decide you want to have the companies in alphabetical order, and you only want to look at customers from Illinois or Indiana.

An important feature of GQL is its support for incremental query formulation. You can go back to the query you just submitted and change it to reflect your new goals.

Let's go back to the customer attribute window. The **Window** menu lists all the GQL windows currently open. Choose *customer* from the **Window** menu.

[Note: if customer doesn't appear in the list then you may have closed the window earlier. In that case, select the window named *\*USA Sales Information\** to get back to the data model window. Double Click on the *customer* icon to reopen the customer attribute window.]

To the right of each attribute name are four buttons. The fourth column of buttons (at the far right) are the **Sort** buttons. Click the one beside *Name* to sort the results by *Name*. A numeral 1 will appear in the button to indicate that this is the primary sort field (i.e. sort the results so that the companies come out in alphabetical order).



Extra options added to query

To restrict the results to those companies in Illinois or Indiana, we will add two qualifications (restrictions) to the query.

The second column of buttons (in the middle) are the **Qualify** buttons. Click the one beside *State* to indicate that you want to add a new qualification. A check mark will appear in the button to indicate that there is a qualification on the field.

At the bottom of the *customer* attribute window a qualification node for *state* will be displayed and the text insertion point is active. Simply type *IL*, the state code for Illinois. (This must be upper case.)

But we wanted two conditions (Illinois or Indiana), so click the qualify button beside *state* again. Type *IN*, the state code for Indiana. (Also in upper case.)

There is one more step. Right now the qualification requires that the customer's state code be IL **and** IN, we want IL **or** IN. Simply click on the word **and** between the two qualification nodes, it will change to **or**.

Now press the **Submit Query** button to get the new results!

## Full Menus

GQL can be set to use either short menus or full menus. Short menus provide a simplified interface that meets the needs of many users, but in order to use some of the advanced features in GQL you will need to enable full menus. Before proceeding with the next section of the demo, select **Full Menus** from the **Edit Menu**.

## Data Values

We'll start a new query. First close the results window from the previous query (click on the go away box), then select **New Query** from the **Query** menu. This clears out the previous query so we can start from scratch.

Now let's get information on *Nicolo Corporation*.

Double click the *customer* data object icon to open the *customer* attribute window. Choose *Name*, *Address I*, *City*, and *State* as the attributes for the query.

If we submit the query now we'll get all the customers - not just *Nicolo Corporation*. Let's put the qualification on company name by clicking on the **Qualify** button beside the *Name* attribute. Type *Nicolo Corporation*.

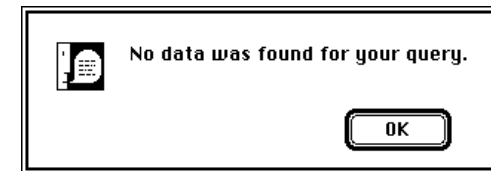
Attribute	Function	Qualify	Group	Sort
? Cust #				
? • Name		✓		
? • Address 1				
? Address 2				
? • City				
? • State				

Name = Nicolò Corporation

Submit Query

### First attempt at the qualification

Press the **Submit Query** button. The query returns with an alert saying that no data matched our qualification.



### When there are no results for your query

Sometimes you want to put a qualification on a query but you don't know the value exactly as it occurs in the database. *Nicolò Corporation* may not be the exact name as it appears in this database. We'll use the Data Values facility to get the value as it is in the database.

Close the query results window and use the Window menu to reactivate the customer attribute window. (There is an entry in the Window menu 'customer'. This is the customer attribute window.) Select the text of the qualification (leaving the quote marks out of the selection).

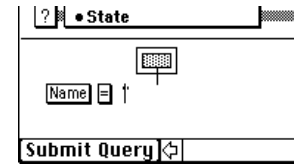
Attribute	Function	Qualify	Group	Sort
? Cust #				
? • Name				
? • Address 1				
? Address 2				
? • City				
? • State		✓		

Name = Nicolò Corporation

Submit Query

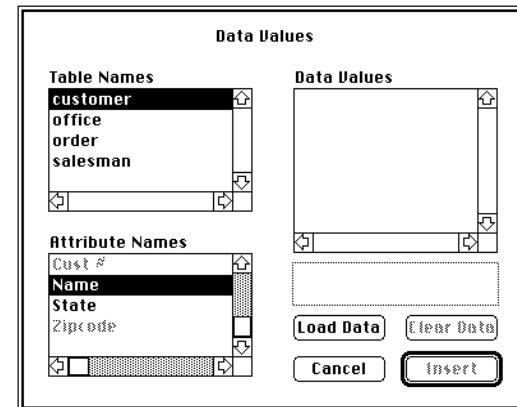
### Select the qualification to be replaced

Type the **backspace** or **delete** key to delete the qualification *Nicolo Corporation*.



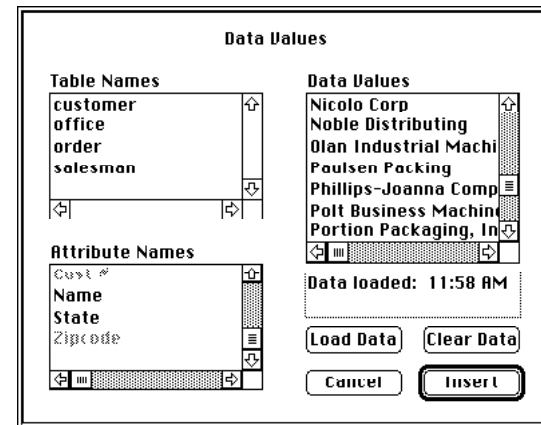
Clear the previous qualification value

Select **Data Values...** from the **Query** menu. The data values dialog will appear.



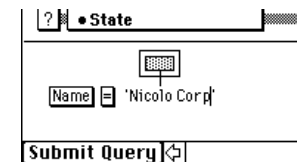
**Data Values dialog**

Click the **Load Data** button. The customer names will fill the data value list. Type *N* to scroll the list down to the *Ns*. Select the one for *Nicolo Corporation*.



Select the correct value

Click the Insert button to have the value copied into the customer name qualification. Now we see why our first attempt didn't work - the actual value in the database was *Nicolò Corp*.

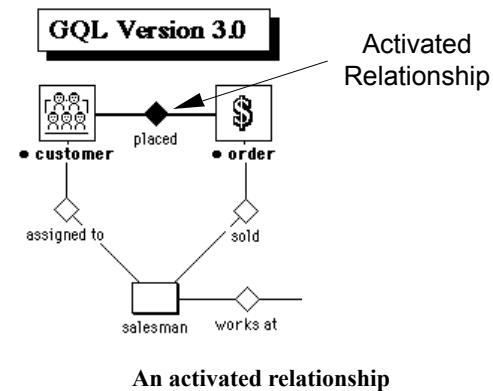


The correct qualification

Now press the **Submit Query** button, and voilà, the information on *Nicolò Corporation*! The values loaded for the company names will be cached in GQL until you quit from the application, so if you need the company name again you won't have to press the **Load Data** button.

## Using Relationships

This query now gets the information we wanted about a particular customer. Let's now find out about the orders that this customer has placed. Reactivate the *\*USA Sales Information\** window. There is a relationship between *customer* and *orders* called *placed*. Click once on the *placed* icon (the diamond). The connecting line from *customer* to *placed* and the line from *placed* to *orders* are highlighted. We have just activated the relationship between *customer* and *orders*.



Now double-click on *order*. (If you forget to activate the relationship before you try to open another data object, GQL warns you that you have forgotten the relationship. Simply **Cancel** the alert and go back to put the relationship in.) The order attribute window is open. Select *Amount* and *Date Required* (Notice that you're dealing with attribute names like Date Required rather than cryptic database names like *req\_date*.)

Now press the Submit button, and the information on *Nicolo Corporation* now includes the current orders.

## Saving Queries (Query Documents)

The query we just executed may be one that you want to be able to use again in the future. Let's save it so we can use it later.

Select **Save Query...** from the **Query** menu. Specify a name for this query (say *Nicolo*) and press the **Save** button. This will save the query as part of the *\*USA Sales Information\** document. Should you want to exchange this query with another user, or provide this query as a Data Access Manager (DAM) *query document* to another application that uses the Data Access Manager High Level Interface, you could have chosen to save the query in a separate document (the **To File** button).

Now the query is available for use at anytime in the future. Simply use the Load Query... operation from the Query menu to have the saved query reloaded. Of course the query can be edited or extended once it is reloaded. You'll have to save it again if you want the edited query to be stored in place of the original.

# Executive Buttons

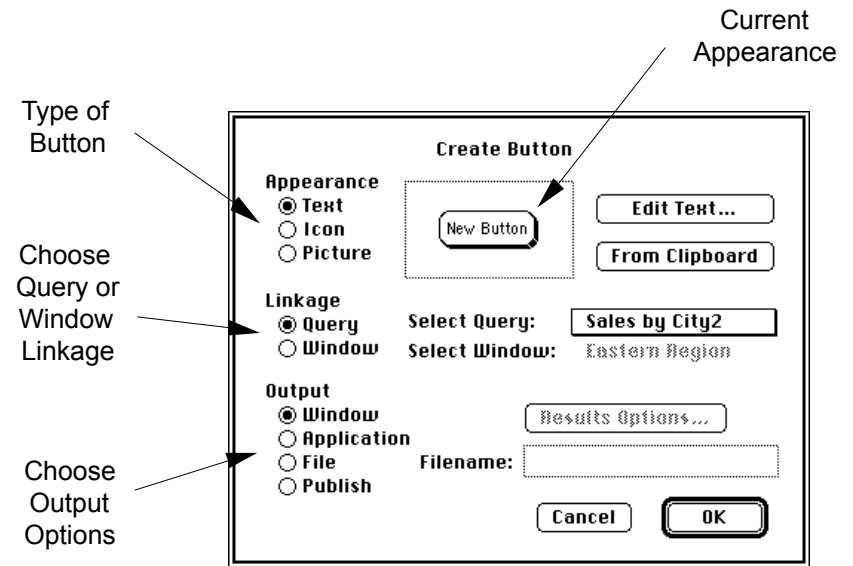
Saved queries are very useful but some queries are used so regularly that you may want even easier access to them. GQL's executive buttons provide a convenient way to organize and access your standard queries.

The *USA Sales Information* has several executive buttons already defined. Return to the *\*USA Sales Information\** window.

Press the button labeled **Customer List**. GQL will load the query associated with this button, and Submit the query. The results will be returned in a query results window.

Now let's create a button for our own query. Return to the *\*USA Sales Information\** window.

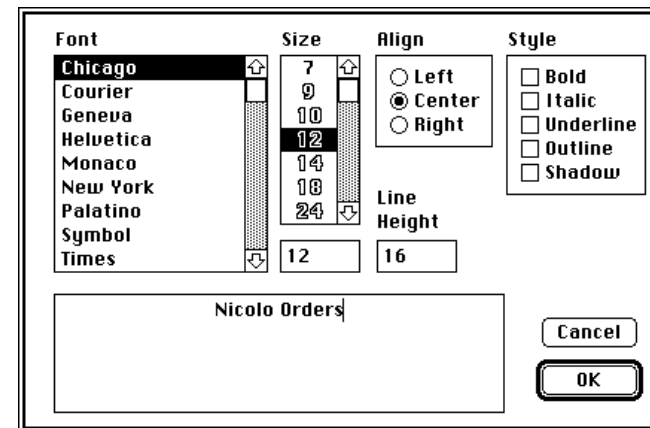
Select **Create Button** from the **Layout** menu. The create button dialog will appear.



Creating an Executive Button

There are three steps in creating an executive button: First, specify the appearance of the button, Second, specify the linkage to a query or window, and third, specify the output destination for the results.

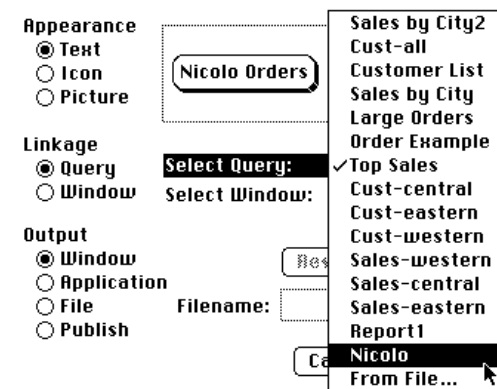
We'll create a Text button. Press the **Edit Text...** button.



Specifying the button label

Type *Nicolo Orders*, and select the font, size, alignment, and style that you want. Press **OK** to return to the button dialog.

The Linkage is set to Query by default. To select the actual query, press and hold down the mouse button while the cursor is over the box to the right of *Select Query*:. A pop-up list of the available queries is displayed. Choose the *Nicolo* query you created earlier.



Choose the query

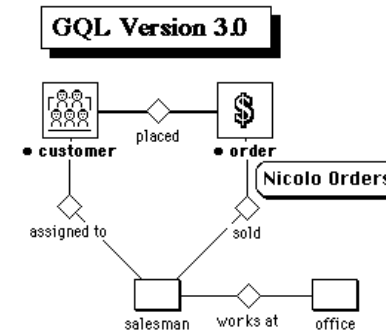
From the list of stored queries select the *Nicolo* query. Press **Link** to select this query and return to the button dialog.



Leave the Action section for now - the results will be sent directly to a GQL query results window.

Press OK to return to the *\*USA Sales Information\** window.

You will find the button you have just created selected in the window. Drag it to a convenient location, then press the mouse button when it is not over the executive button in order to deselect the button. It's ready to use! Just press it to execute the query.



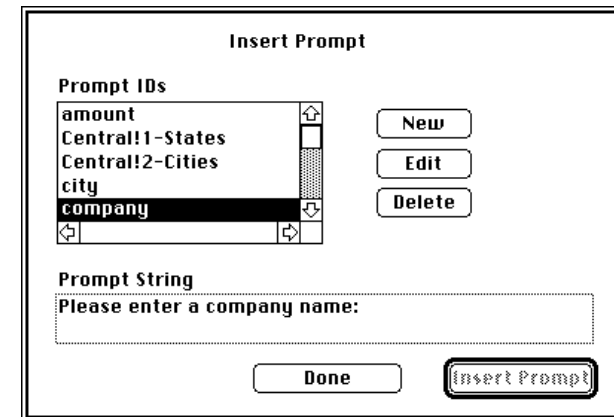
**The new executive button**

[Note: If you want to edit the button specification, or move it elsewhere in the diagram, select Edit Mode from the Layout menu. While in Edit Mode you can access the button dialog by double clicking the button, and move it by dragging it with the hand cursor.]

# Qualification Prompts

The query we created and saved with a button was useful for getting the sales to Nicolo but not as useful for other companies (although it can always be loaded, the qualification edited, and the revised query submitted). Let's generalize the query so that it will allow us to enter the company name whenever the query is selected. We will use GQL's Qualification Prompt facility.

There's no sense starting from scratch, let's load the Nicolo query. Select **Load Query...** from the **Query** menu. Select *Nicolo* from the list and press **OK**. The query is now loaded and ready for modification; the query window named *Query: Nicolo* is displayed. Return to the *\*USA Sales Information\** window. Double-click on the *customer* icon to open the *customer* attribute table. In the qualification area select the text *Nicolo Corp.* Press **backspace** or **delete** to clear the text and just leave the quote marks. Now select **Insert Prompt...** from the **Edit** menu..



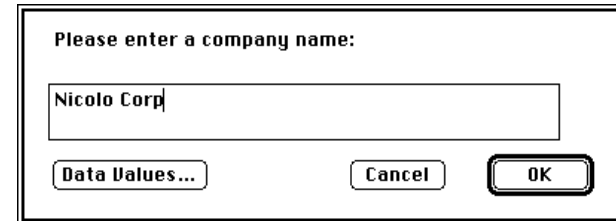
The Qualification Prompt dialog

We've already arranged to create the prompt for you so select the Prompt Id *company*. Press the **Insert Prompt** button to have the special qualification prompt inserted into the qualification for you.



The qualification with prompt token

Now test the query by pressing the Submit Query button. GQL prompts you for the company name. Obviously you could type the name if you know it, but it's easier to use the **Data Values** button to get at the values from the database. Use **Data Values** to select *Acme* . Press the **OK** button to execute the query and get *Acme*'s orders.



The prompt dialog when the query is submitted

Qualification Prompts provide a mechanism to create generalized query templates that you complete at the time the query is to be submitted. There can be more than one qualification prompt in each query. They can be used in many contexts in the query to achieve the results you want. The value last used for a qualification prompt is remembered by GQL in the event you want to reuse the value in several related queries.

## Getting Started with System 7.0

This section is provided for those who are familiar with GQL and simply want to explore the Macintosh System 7.0 Specific Features.

**Data Access Manager:** GQL supports use of the System 7.0 Data Access Manager Low Level Interface and Data Access Language extension.

**Publish/Subscribe:** GQL/User allows executive buttons to publish query results. (To edit existing buttons select **Edit Mode** from the **Layout** Menu). Once the publisher is created and the GQL document is saved, the linkage is maintained by GQL. Each time that button is pressed, the query is issued and the Edition file is rewritten with new results.

**Query Documents:** When you save a GQL query in a file GQL creates a Data Access Manager Query Document. These query docs can be used by other applications that support the Data Access Manager High Level Interface. At present the queries cannot contain qualification prompts.

**Balloon Help:** GQL/User has Balloon help for all its menus and the data objects in the data model window.

**AppleEvents:** GQL currently supports the required AppleEvents.

## Publish/Subscribe (using GQL and Excel)

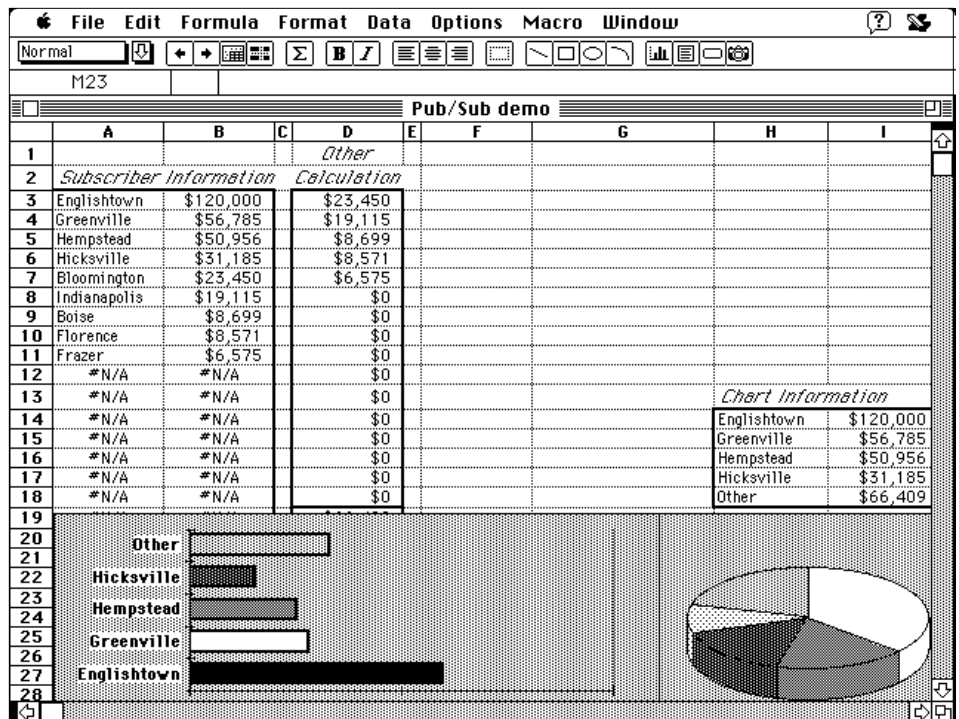
Query results are often most useful when they are exported from GQL into other Macintosh applications. GQL supports export of query results in several ways. Data can be selected in the Query Results window and copied into the clipboard. (All the data can be selected by choosing **Select All** from the **Edit** menu, or by pressing on the row count in the upper left part of the results window.) From the clipboard the data can be pasted into any application (try the CMD-Shift-V power user paste).

Query results can be saved to a file. The file can be read by most applications. To save a results file select **Save Results** from the **Results** menu.

Executive buttons can be set up to save results directly to a file, directly to another application through the clipboard, or to Publish an edition file that can be Subscribed to by other 7.0 savvy applications. Lets see how the executive button feature can be set up with a specific application such as Excel.

You must have Microsoft Excel 3.0 to run the next demonstration.

In the Finder locate the Pub/Sub Demo document. Launch Excel by double-clicking on the Excel document labeled *Pub/Sub demo*. This will open the Excel spreadsheet that has been set up for this demo.



The Pub/Sub demo screen

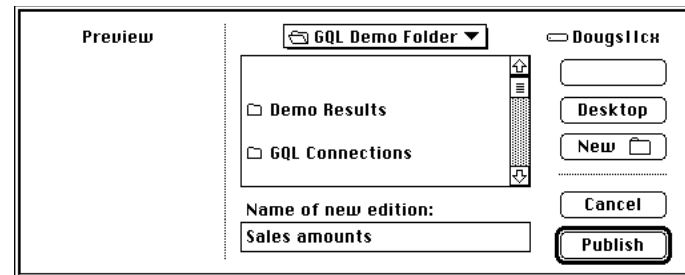
Return to the GQL application, leaving Excel running.

Press the button with the icon of the bar chart on it.



This button executes a query that retrieves selects all orders over a certain amount. The button was specified to publish its results to an Edition file named *Sales Amounts*. This is specified by the Publish operation in the Action Section of the button dialog.

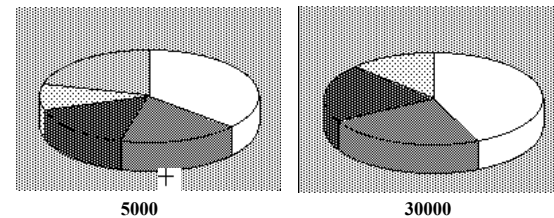
You will be prompted for the name of the edition file. Select **Publish** to OK the default name (Sales Amounts).



**Edition Manager Dialog**

Since the demo disk has an edition of this name you will be prompted to replace the existing file. Press **OK**.

The Excel document subscribes to that edition and uses the information to draw two charts. Try the query with say 5000 and with 30000 as the minimum and maximum amounts. You will see the charts automatically updated by Excel.



**Charts created through the Executive Button facility**

The effect of the executive button is to draw the appropriate chart. Of course, each button can have a separate effect; the default behavior is simply to paste the data into a new spreadsheet.

## Other Features

This demo highlights the main features of GQL, but there are lots of other capabilities. You can explore some of them by experimenting with the other menu items. Some things you may discover are:

Aggregate functions (SUM, AVG etc)

Grouping

Dynamic relationships

Full/Short menus

Combining qualifications

Preferences

Color Support (including preferences)

Icon and PICT executive buttons and ornaments

Annotations

Submit query to file

Results options

Things to try...

Reports: from the results window you can generate formatted reports using commands from the Report Menu. You must have selected Full Menus (from the Edit menu) and then choose Show Report from the Report menu.

Quick Select of Attributes: with the cursor over a table object in the data model, press and hold the mouse button. A pop-up list of attributes will be displayed for easy selection.

Form View: in the results window you can double-click on any row number to switch into form view. The form view shows all columns for a single row and allows you to step through the results one row at a time.

Large Column display: in the results window, long text strings can be truncated at the column boundary. You can see the whole string by double-clicking a cell, an expanded view will be displayed.

Automatic Column Widths: there is a check box in the column width dialog that will automatically resize columns to the largest size required for the current data.

Just remember that GQL is a very versatile tool. It can be used by an experienced IS professional to create query environments for their end-users or themselves, and it can be used by end-users that want access to the data but don't have training with relational database concepts. GQL can be customized to meet the different skill levels and requirements of different users.

## **Building GQL Data Models**

You may be wondering where the data model document comes from? The data models are created by the application GQL/Admin. The task of building the data model for a user is usually done by a member of the IS staff who knows the structure of the database and who can customize the data model for their users. GQL/Admin makes it very easy to add value to the database for the end-user. By providing effective end-user access to corporate data the IS staff is freed from routine data requests and can concentrate on the more difficult tasks for which their specialized training and experience are of critical importance.

## **GQL - The Corporate Data Access Solution**

GQL is the ultimate solution for corporate database access. It enables end-users to access one of the corporation's most valuable assets - information previously locked in corporate databases.

GQL supports the goals of Information Systems professionals who are striving to provide access to the corporate database. It allows the IS staff to select the information to be presented to the users, and to take advantage of their specialized knowledge to create customized data models and EIS solutions for individuals or departments. It helps you fulfill their mandate to provide users with information, not just data.

The GQL tools provide significant productivity gains, both for end-users and for the IS staff that supports end-users. GQL delivers a true win-win solution for progressive information-oriented organizations.



# Appendix A

## The GQL Product Family

The GQL family of products includes GQL/User, GQL/Admin, GQL/Design, and GQL/Update.

**GQL/User** provides decision makers with an intuitive and graphical view of database information customized to individual, department, or corporate needs. GQL's graphical approach to data access allows users to quickly locate, retrieve, or update information critical to their personal needs. All query results are displayed in a familiar spreadsheet format and can be stored locally on a Macintosh workstation. This data can then be searched, sorted, or seamlessly integrated with other Macintosh applications, including word processors, presentation packages, and spreadsheets such as Excel or Wingz. GQL's unique Executive Button facility allows organizations to create fully customized data access environments for the widest possible range of users. Using GQL, organizations are free to design user interfaces consisting entirely of Executive Buttons, as well as those containing customized database models for satisfying ad hoc query and reporting needs. GQL supports both Windows DDE facilities and Macintosh System 7 features.

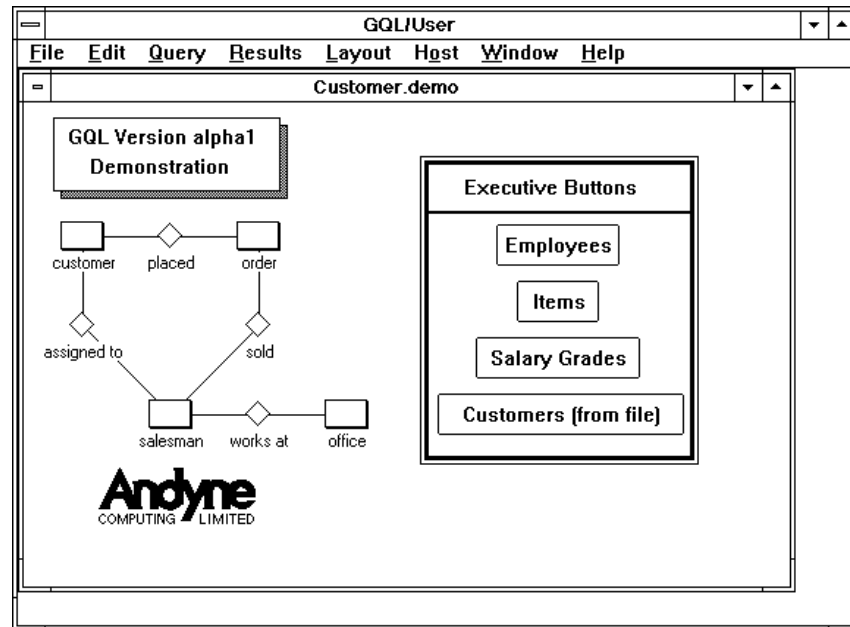
**GQL/Update** is an extended version of GQL/User that allows users with special access privileges to perform data entry and table maintenance functions.

**GQL/Admin** allows the database administrator, MIS personnel, or database application staff to develop flexible and sophisticated query environments for the widest possible range of users. The administrator begins by loading the database schema directly from the host system in order to create the data objects used in the data model. Using GQL/Admin, the administrator can rename data objects and attributes to make them more meaningful to users, as well as developing an attractive and intuitive layout of customized data models for use with GQL/User. The administrator or application developer can also store predefined queries as Executive Buttons to provide users with database applications tailored to their needs. And because GQL works transparently with popular programs such as Excel and Wingz, database application developers can use GQL/Admin to create sophisticated decision support systems that include integrated charting and data analysis capabilities.

**GQL/Design** provides database designers with a rapid prototyping tool for creating new databases and data models for use in testing and developing sophisticated query applications. Using GQL's sophisticated graphical interface, the designer first creates the data model on the Macintosh, then generates the corresponding database tables on the host computer, or stores the create commands locally as a script file. The resulting data model can then be used directly with GQL/User. GQL/Design supports the entity relationship design method and includes built-in design checks.

## Windows 3.0

GQL for Windows supports both the Dynamic Data Exchange (DDE) and Object Linking and Embedding (OLE) technologies of MS Windows 3.0 and later releases. These advance data sharing and inter-application communication facilities allows developers to use GQL to create sophisticated office automation and decision support systems for a wide range of business, academic, and technical applications.



## Unix/Motif

GQL for Motif will support X/Windows OSF Motif on a variety of Unix platforms. As with GQL for Macintosh and Windows, Unix support will enable end users to share data models, query results, and reports quickly and easily.