King
Training Resources

A Picture is Worth a Thousand Words: Graphics in Forms

John King
King Training Resources
6341 South Williams Street
Littleton, CO 80121-2627 USA
www.kingtraining.com
800.252.0652 or 303.798.5727

Objectives

- Learn how Oracle Graphics may be used to “dress up” Oracle Forms applications
- Understand how Oracle Forms and Oracle Graphics interact
- Know how to pass parameter values between Oracle Forms and Oracle Graphics

Introduction

- Developer 6.0’s graphical capabilities present a resource untapped by most applications
- Effective use of Oracle Graphics charts and imported images allows development of outstanding and useful applications
- Graphic charts and forms interact using PL/SQL and triggers

Embedded Oracle Graphics

- Oracle Graphics is easy to use, powerful, and works well when executed alone or when executed in conjunction with Oracle Forms
- Graphic objects may be used in forms to graphically represent data and allow users a more-intuitive way of viewing data and/or launching queries
- Oracle Graphics is an OLE Server, so, objects created with it may be included in non-Oracle OLE Clients like Microsoft Excel and Microsoft Word
(in an MS-Windows environment)

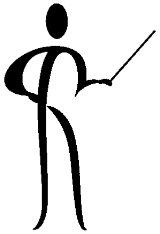
Using Oracle Graphics in Forms

- Oracle Graphics is designed to work closely with Oracle Forms and Oracle Reports, even including a Chart Wizard to aid in creation and maintenance of forms with embedded (simple) graphics
- Oracle Graphics can transfer to and receive data from Oracle Forms, allowing “drill-down” applications
- When executed by another product, Oracle Graphics starts another process, the Oracle Graphics Batch Executable — this process remains in memory for future use

Using Chart Wizard

- The basics for creating an application to work with an Oracle Graphics via the wizard is easy:
  - As a form is created, use the Chart Tool to indicate where in the visual layout a chart should go
  - Next, the Chart Wizard starts and asks for information necessary to build the chart
  - The Chart Wizard may be reentered later, so, if the chart isn’t perfect the first time it can be corrected easily later
  - Finally, the form is tested

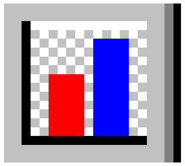
More Complex Graphics



- Some Oracle Graphics objects may be directly linked to SQL, others might not be
- “Drill down” graphics may be tied to forms using PL/SQL within a form or by having PL/SQL triggers inside the embedded graphic object
- Graphics objects may be created that both send and receive data from the form

Chart Creation

- Use the chart tool to create a chart:



- Drag and resize chart area using the layout editor
- If manually building the chart item, make it the size and shape desired

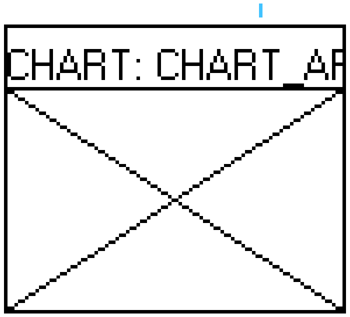
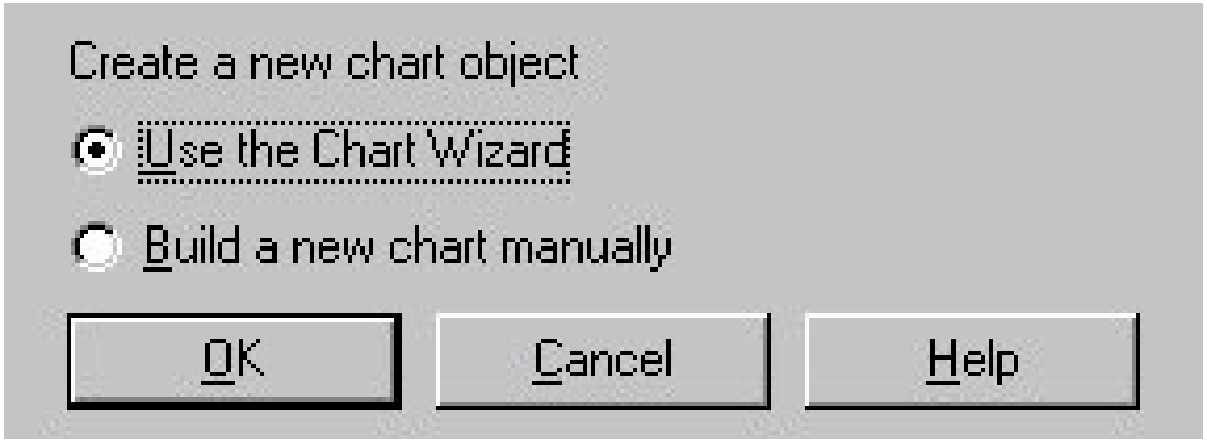


Chart Wizard Creation

- If using the Chart Wizard a series of panels begins (assumes existing master detail DEPT/EMP form)



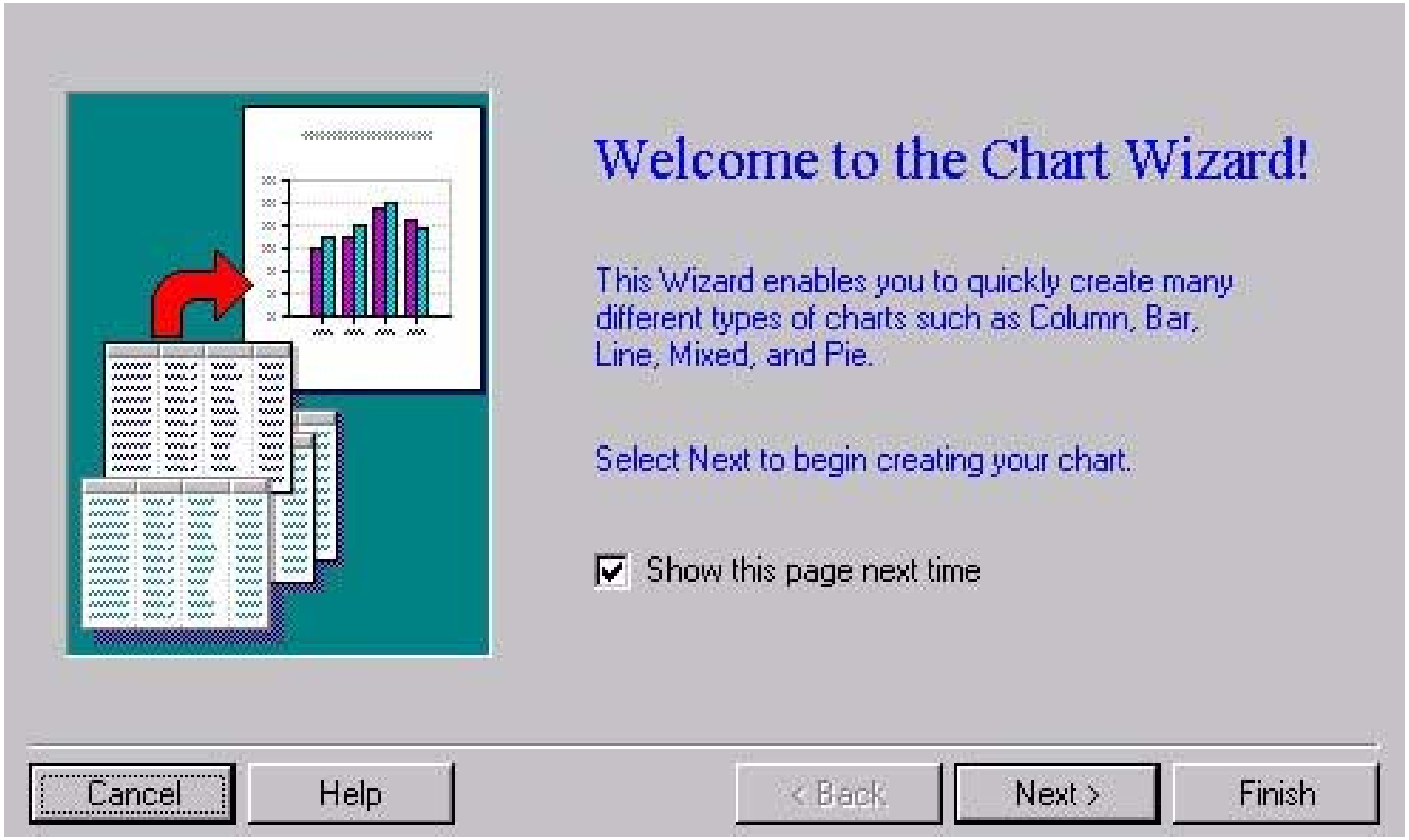
Create a new chart object

● Use the Chart Wizard

○ Build a new chart manually

Chart Wizard Welcome

- The first panel is simply a welcome to the Chart Wizard, it may be omitted if desired:



Welcome to the Chart Wizard!

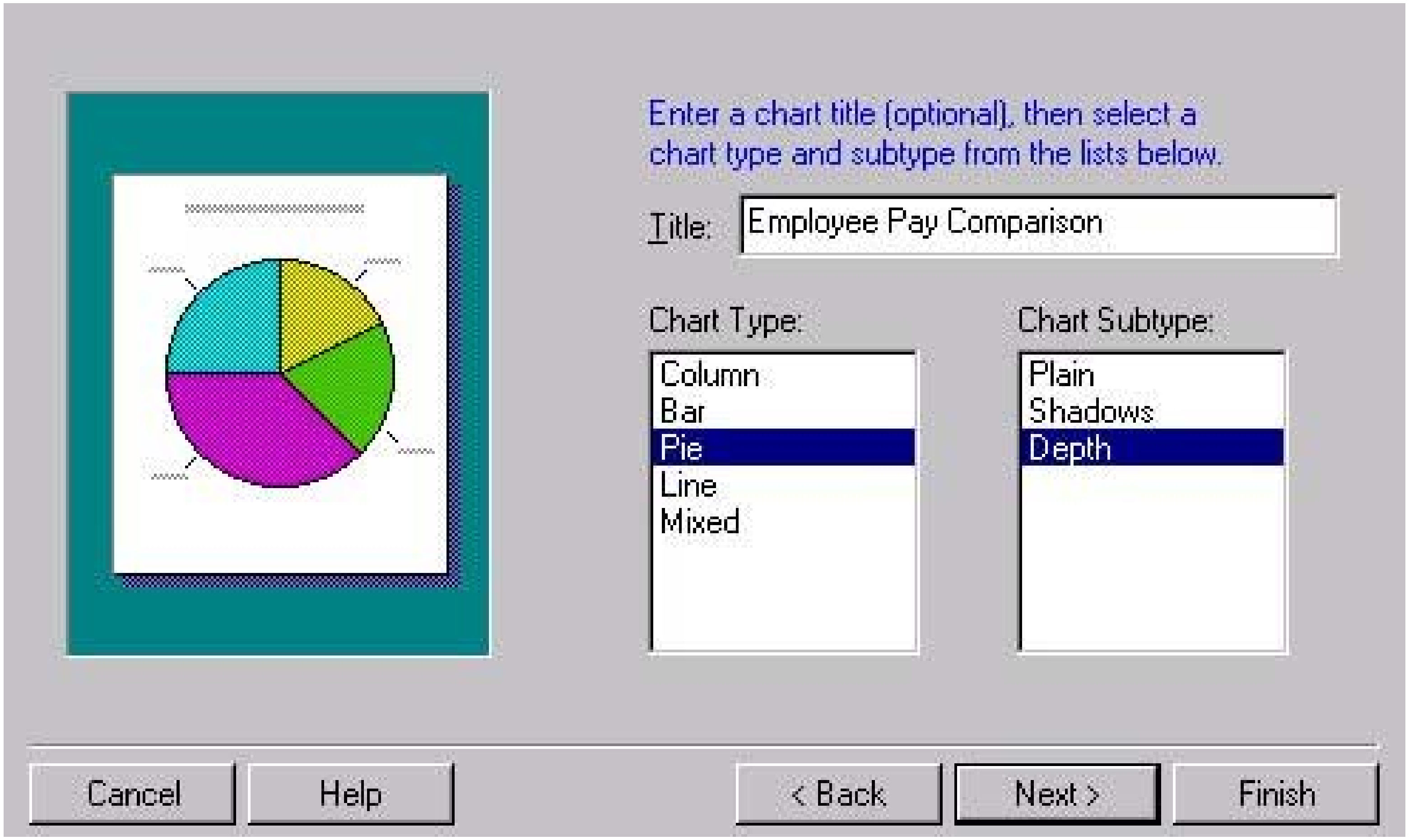
This Wizard enables you to quickly create many different types of charts such as Column, Bar, Line, Mixed, and Pie.

Select Next to begin creating your chart.

☑ Show this page next time

Choose Chart Type

- This panel allows selection of the chart format, the wizard is reentrant so experiment:



Enter a chart title (optional), then select a chart type and subtype from the lists below.

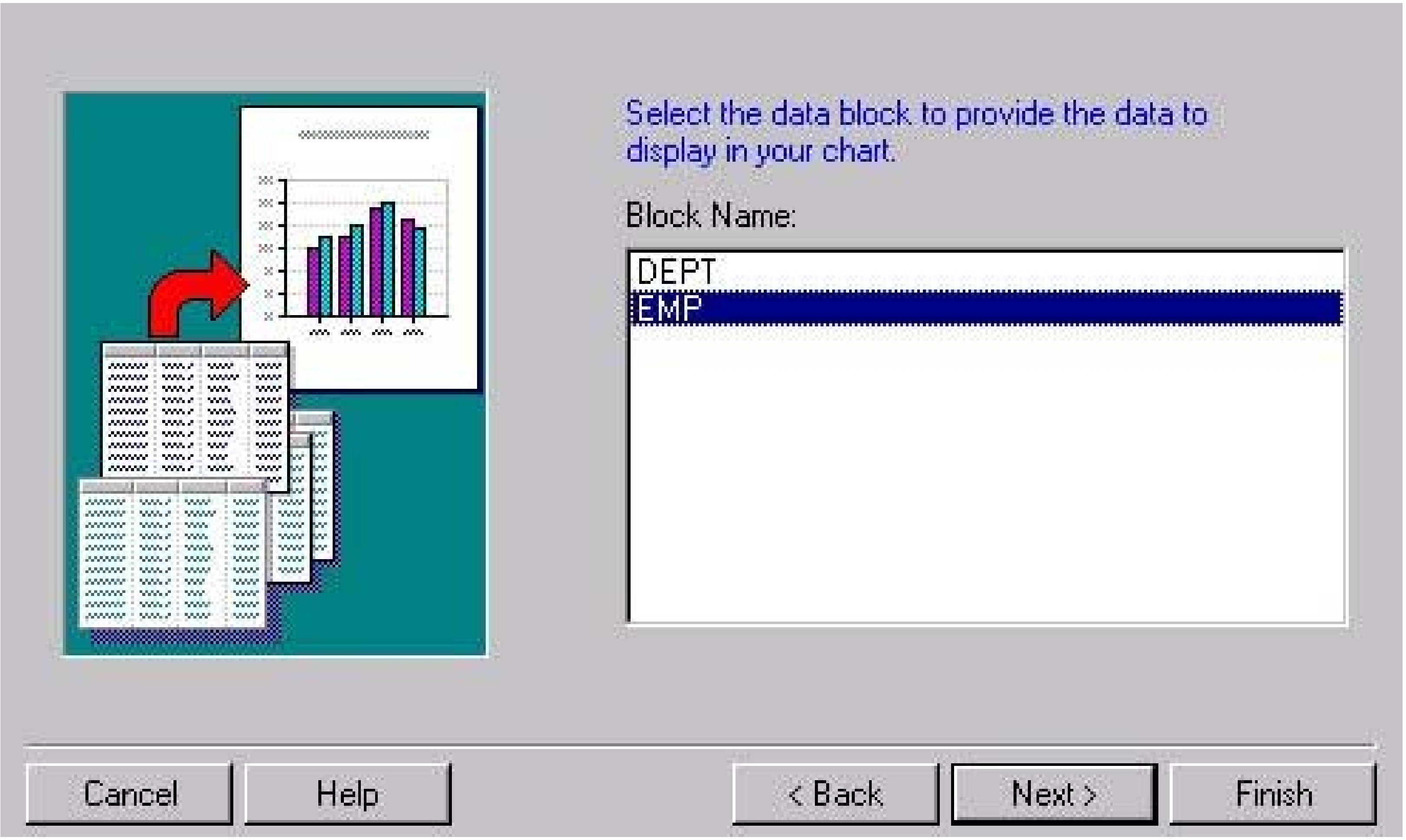
Title: Employee Pay Comparison

Chart Type: Pie

Chart Subtype: Depth

Chart Data Block

- Next, specify the block the chart's data comes from:



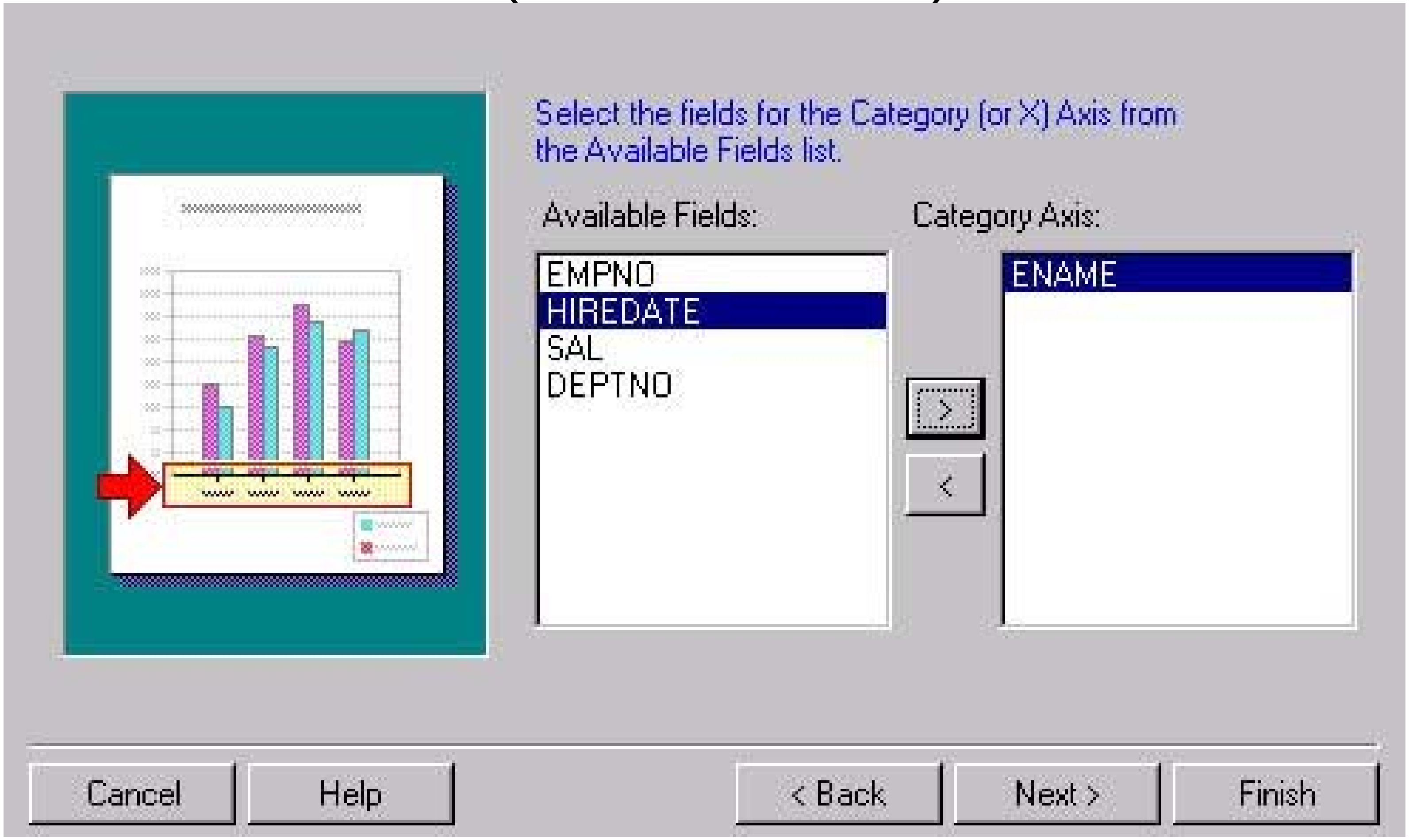
Select the data block to provide the data to display in your chart.

Block Name:

DEPT
EMP

Category Column

- Choose the column(s) that will supply the category data for the chart (chart item labels):



Select the fields for the Category (or X) Axis from the Available Fields list.

Available Fields:

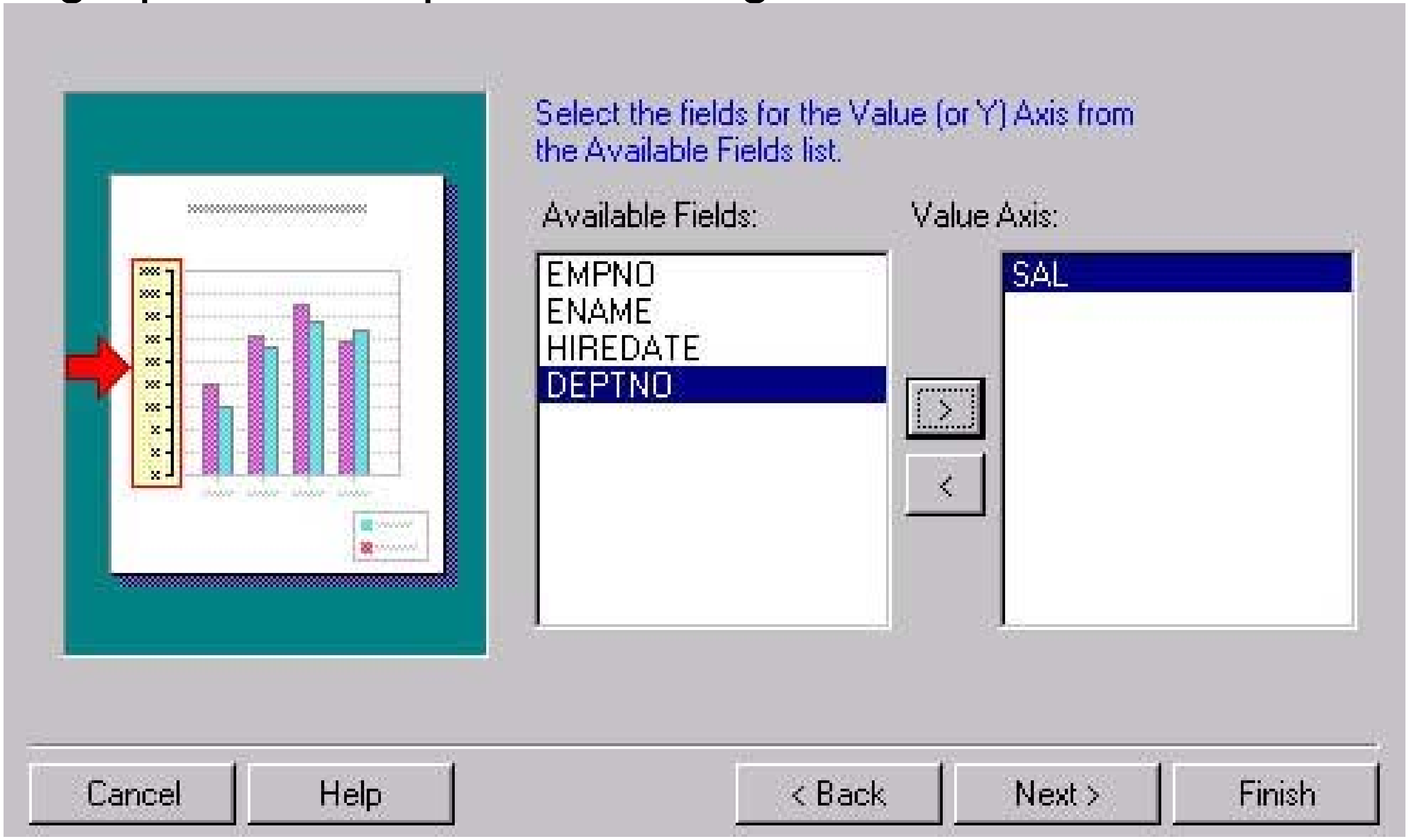
- EMPNO
- HIREDATE
- SAL
- DEPTNO

Category Axis:

- ENAME

Value Column

- Choose the column(s) to supply values for the chart graphics, multiple columns get different colors:



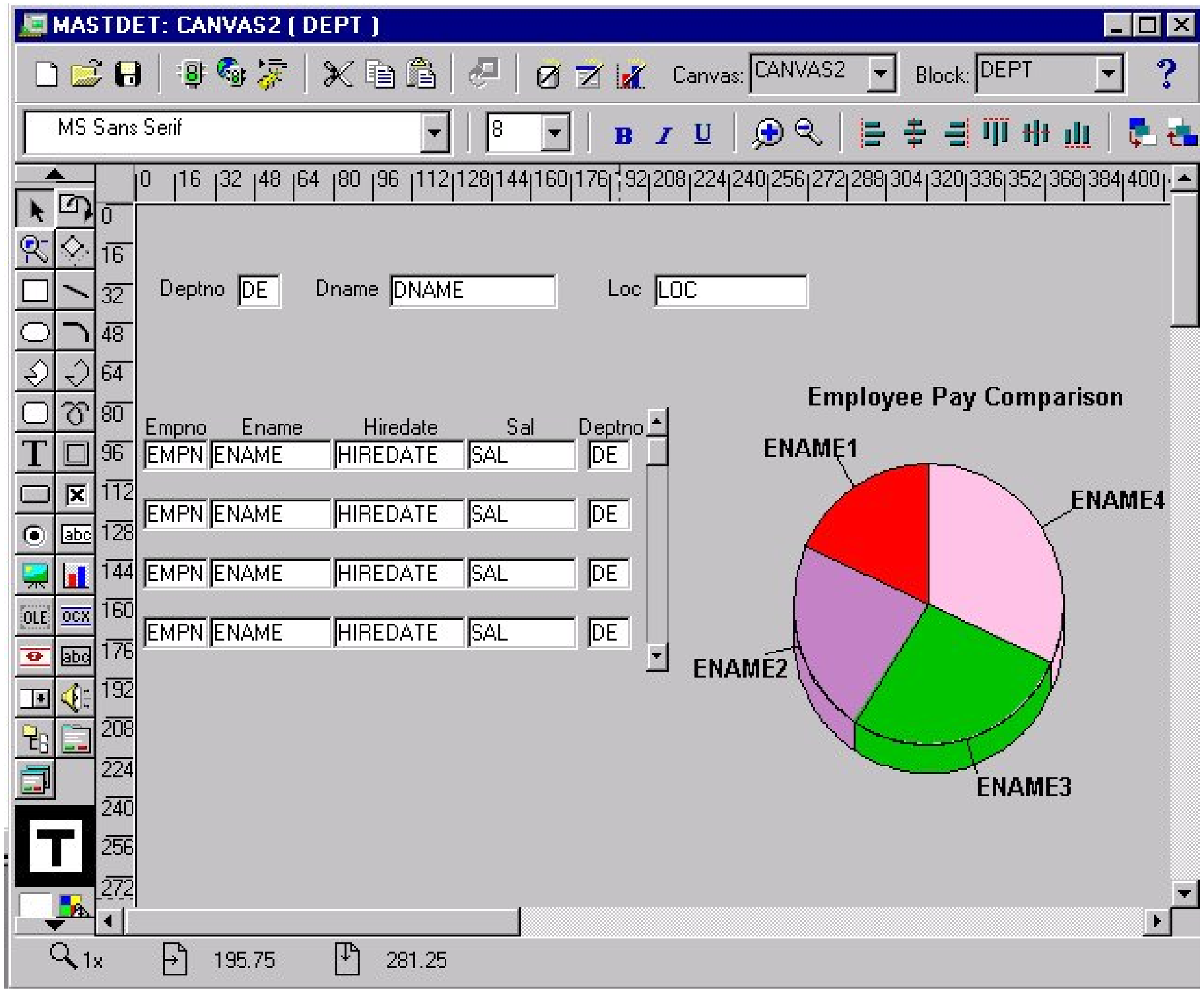
Select the fields for the Value (or Y) Axis from the Available Fields list.

Available Fields:
- EMPNO
- ENAME
- HIREDATE
- DEPTNO

Value Axis:
- SAL

Layout Editor

- Here’s what the Layout Editor display looks like:



MASTDET: CANVAS2 (DEPT)

Canvas: CANVAS2 Block: DEPT

MS Sans Serif 8

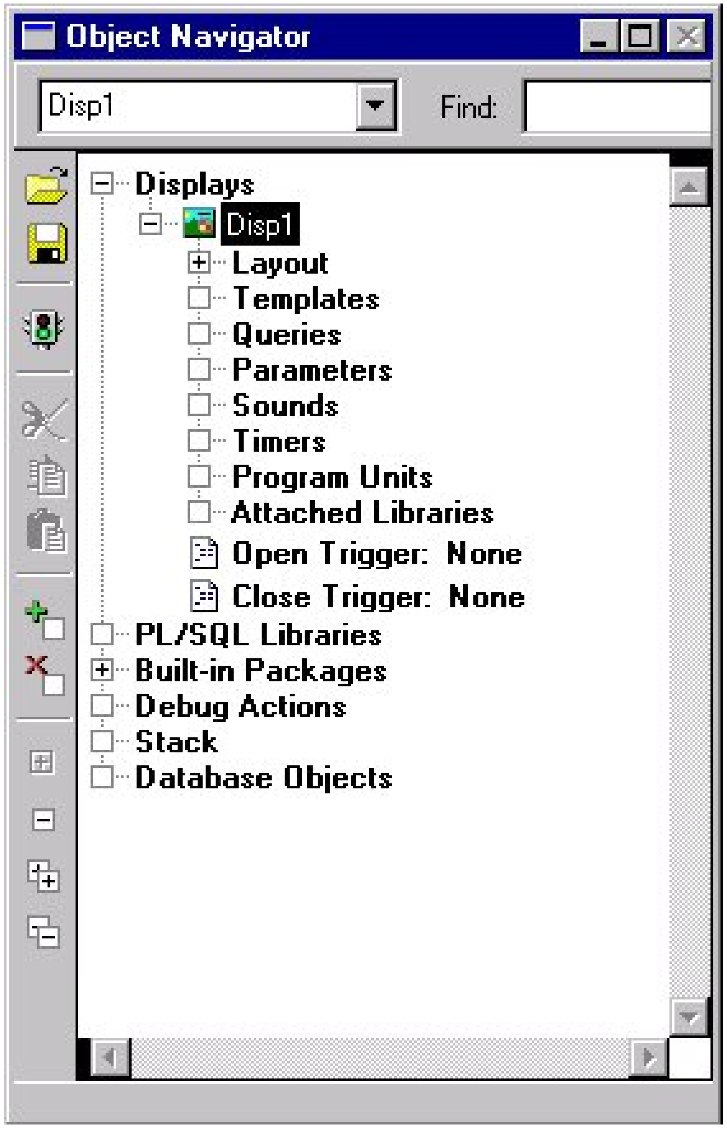
Deptno DE Dname DNAME Loc LOC

Empno Ename Hiredate Sal Deptno
EMPN ENAME HIREDATE SAL DE
EMPN ENAME HIREDATE SAL DE
EMPN ENAME HIREDATE SAL DE
EMPN ENAME HIREDATE SAL DE

1x 195.75 281.25

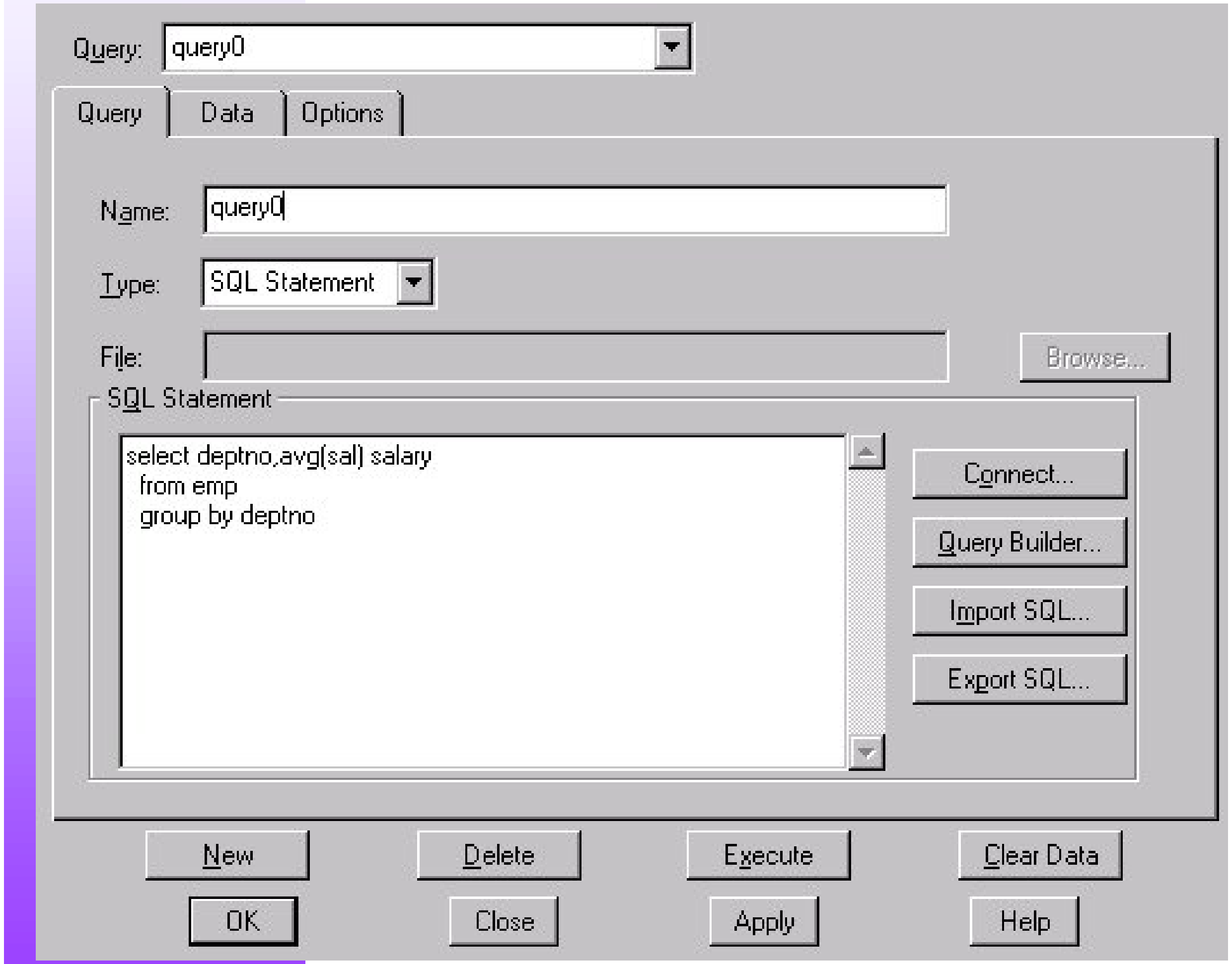
Create using Oracle Graphics

- Oracle Graphics uses the familiar Object Navigator:



- Displays
  - Disp1
    - Layout
    - Templates
    - Queries
    - Parameters
    - Sounds
    - Timers
    - Program Units
    - Attached Libraries
    - Open Trigger: None
    - Close Trigger: None
- PL/SQL Libraries
- Built-in Packages
- Debug Actions
- Stack
- Database Objects

Create a Query - Query Tab



Query: query0

Name: query0

Type: SQL Statement

File:

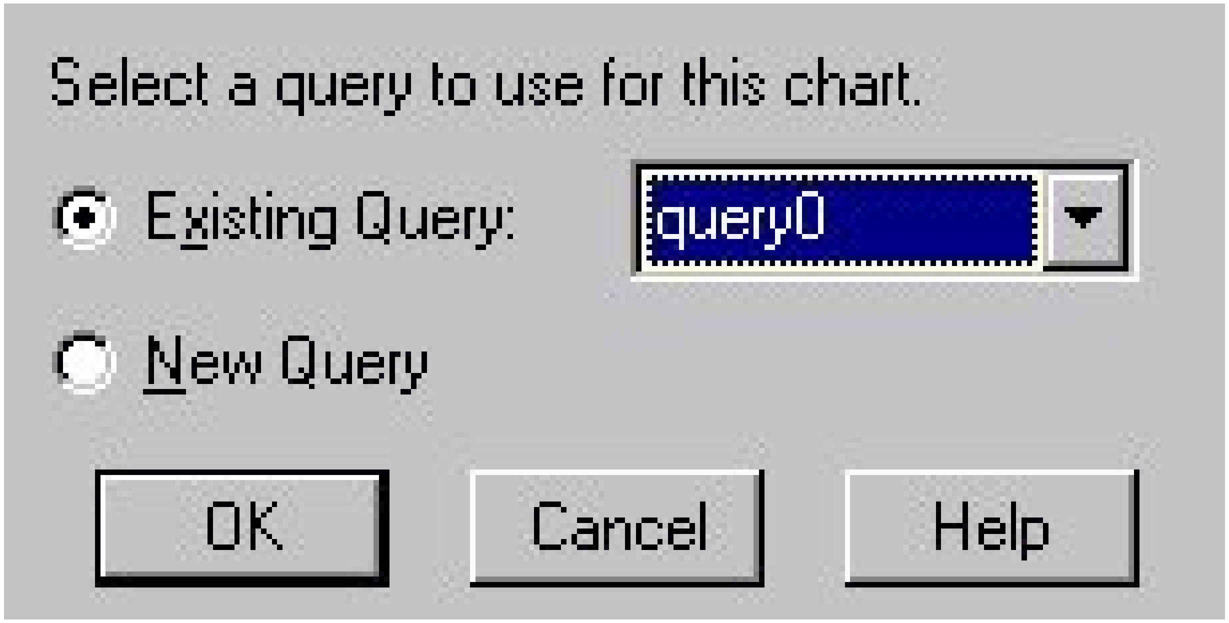
SQL Statement

```
select deptno,avg(sal) salary
  from emp
  group by deptno
```

- Create a new query, parameter values may be used in the WHERE clause or other parts of the queries
- The Data Tab shows the values in the graphic, the Options Tab controls number of rows, date format, and PL/SQL associated with the graphic

Create a Chart

- Use the Layout-Editor Chart Tool, tell the Chart Genie to use the existing query



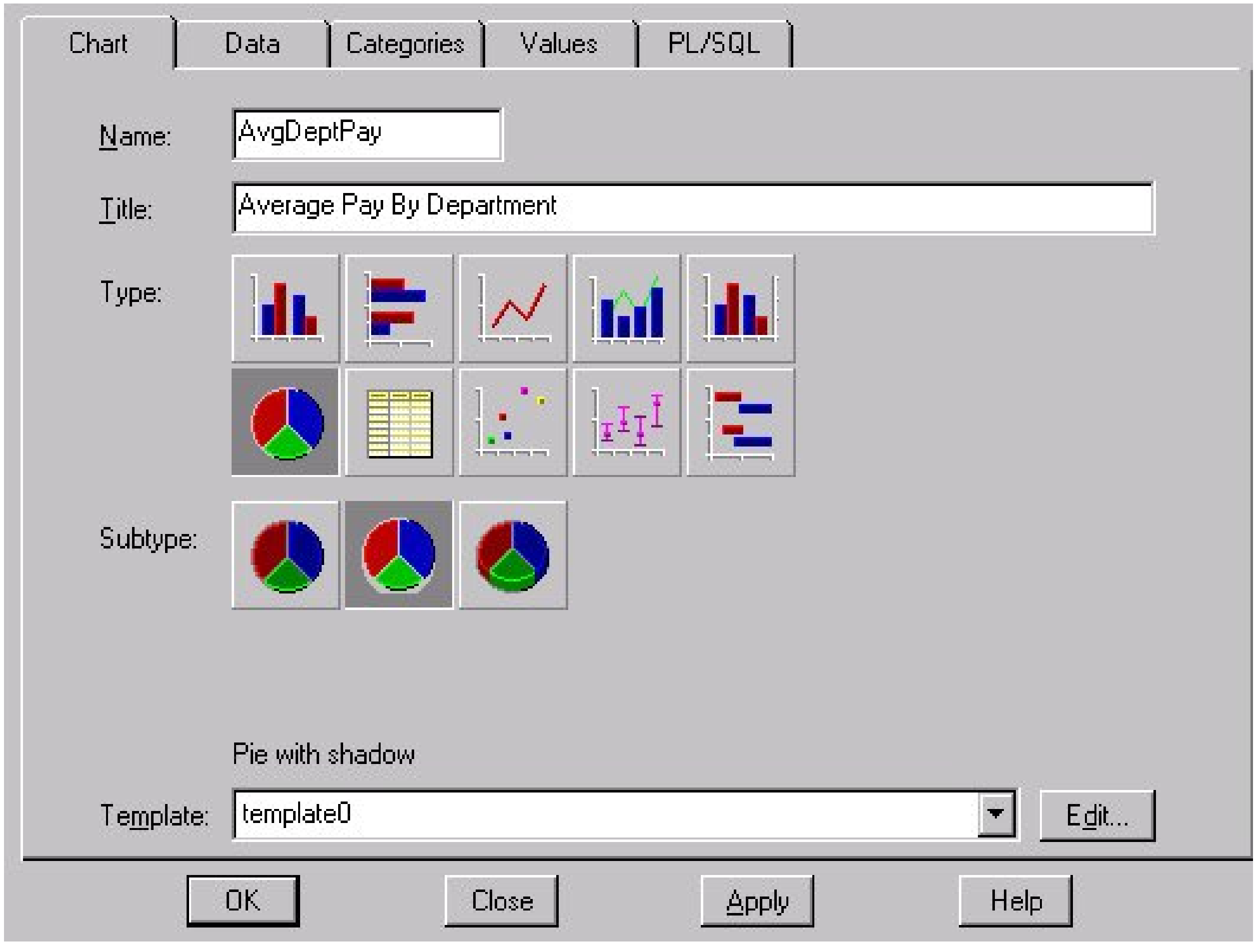
Select a query to use for this chart.

● Existing Query: query0

○ New Query

Chart Properties

- Describe the Chart Properties, chart type, data columns, categories, values, and position:



Name: AvgDeptPay

Title: Average Pay By Department

Type:

Subtype:

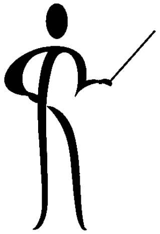
Pie with shadow

Template: template0

Forms Connection

- When creating the chart manually, the Form’s PL/SQL must be prepared to use the chart as well
- This is a four-step procedure:
  - Define the chart object on the for
  - Attach the OG (Oracle Graphics) library
  - Notify Oracle Forms that you will be using a graphics object
  - Use the object

Attaching the OG Library



- Graphics includes a library of PL/SQL program units called OG (og.pll) specifically to manipulate graphics (usually in directory: \Tools\Devdem60\Demo\Forms)
- The OG library must be attached in order to use it; use the object navigator to attach the library
- The OG library contains many built-in subprograms, some are listed on the next page
- Documentation may be found in a .pdf file name gbref60.pdf accessible from the Help menu of the Graphics Builder

Useful OG Built-Ins

◆ OG.Open	Initiate use of object
◆ OG.MouseDown	Simulate mouse click down
◆ OG.MouseUp	Simulate mouse click up
◆ OG.Interpret	Execute PL/SQL unit in chart
◆ OG.GetNumParam	Retrieve numeric parameter from chart
◆ OG.GetCharParam	Retrieve character parameter from chart

Hard-Coded Path Warning

- Unless you desire to provide a hard-coded path to .pll libraries, they should be placed into a path described in the oracle.ini or registry parameters usually named:

  ORAPLSQLLOADPATH or ORACLE_PATH

- Likewise, Oracle Graphics will search for .ogd files in the path identified by the oracle.ini or registry parameters:

  GRAPHICS60_PATH or GRAPHICS60

Notify Oracle Forms of Chart’s Use

- Many applications use trigger WHEN-NEW-FORM-INSTANCE to initialize Oracle Graphics object use

```
OG.Open(‘mygraph.ogd’,’myFormItem’,TRUE);
or
OG.Open(‘mygraph.ogd’,’myFormItem’,FALSE);


     mygraph.ogd    Oracle Graphics file
     myFormItem     Name of form chart item
     TRUE/FALSE     FALSE causes object to
                    be resized to fit item
                    TRUE (the default)
                    causes object to be
                    clipped
```

Passing Mouse Activities to the Chart

- A WHEN-MOUSE-CLICK trigger on the chart item is a common way to activate an Oracle Graphics object’s code
- This allows the Oracle Forms application to pass along the mouse event to the Oracle Graphics logic so that “drill down” logic may be executed properly

```
OG.MouseDown(‘mygraph.ogd’,’myFormItem’);
```

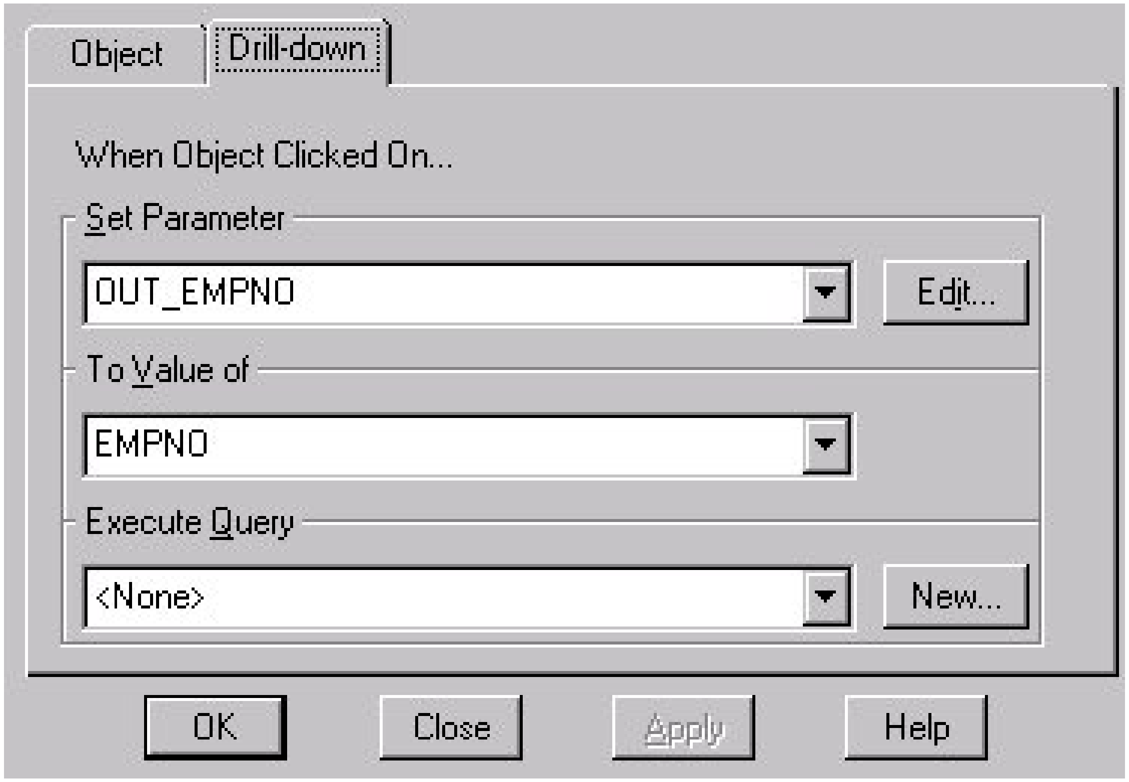
Getting Values Back from Chart

- Upon return from the graphics object, get the value(s) resulting from the graphics object’s use
- Be sure to use the correct PL/SQL function: OG.GetNumParam for numeric data or OG.GetCharParam for character data.

```
OG.GetNumParam(‘mygraph.ogd’,’myFormItem’,’numVar’);
OG.GetCharParam(‘mygraph.ogd’,’myFormItem’,’charVar’);


      mygraph.ogd   Oracle Graphics file
      myFormItem    Name of chart item on form
      numVar        Name of numeric variable
                    inside Graphics object
      charVar       Name of character variable
                    inside Graphics object
```

Chart “Drill Down”

- In the layout editor or object navigator, select the chart, then select a chart object (bar, pie slice, etc…) and double-click to see the object properties, choose the “drill down” tab



When Object Clicked On...

Set Parameter: OUT_EMPNO

To Value of: EMPNO

Execute Query: <None>

Reusing a Chart

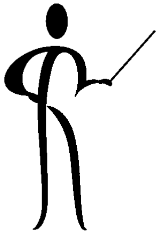
- The OG.Refresh, OG.Interpret, and OG.Close procedures are used to work with existing charts

```
OG.Refresh(‘mygraph.ogd’,’myFormItem’);
      Re-execute query and reload

OG.Intrepret(‘mygraph.ogd’,’myFormItem’,plsqlcode,
                  refresh,plist);
      Execute PL/SQL in graphic

OG.Close(‘mygraph.ogd’,’myFormItem’);
      Close chart

      (Note: it is sometimes useful to CLOSE and REOPEN
            a chart to fully refresh values)
```

Parameter Lists

- Many OG routines use parameter lists, the basics of parameter list use are:
  - Define a parameter list ID variable:
  - Create a parameter list:
  - Add parameter(s) to the list:
  - Reload a parameter (infrequently used):
  - See if a parameter list already exists:
  - Destroy a parameter list

Parameter List Code



Define a parameter list ID variable:

```
      plist   paramlist;
```

Create a parameter list:

```
      plist := create_parameter_list(‘anyname’);
```

Add parameter(s) to the list:

```
      add_parameter(plist,’param_name’,text_parameter,
                        param_value_field);
```

Reload a parameter (infrequently used):

```
      set_parameter(plist,’param_name’,text_parameter,
                        ’param_value’);
```

See if a parameter list already exists:

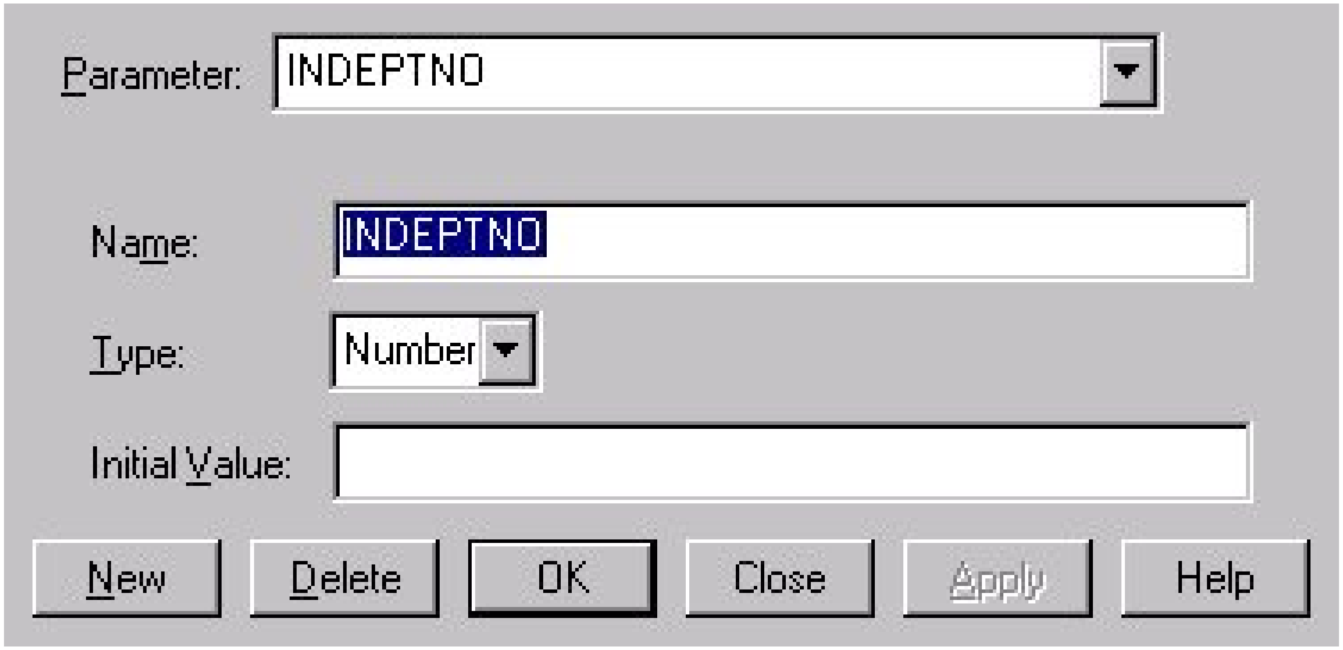
```
      plist := get_parameter_list(‘anyname’);
      if idnull (plist) …
```

Destroy a parameter list”

```
      destroy_parameter_list(plist);
```

Parameters in Graphics

- Define the parameter to the form, be careful to use the correct data type



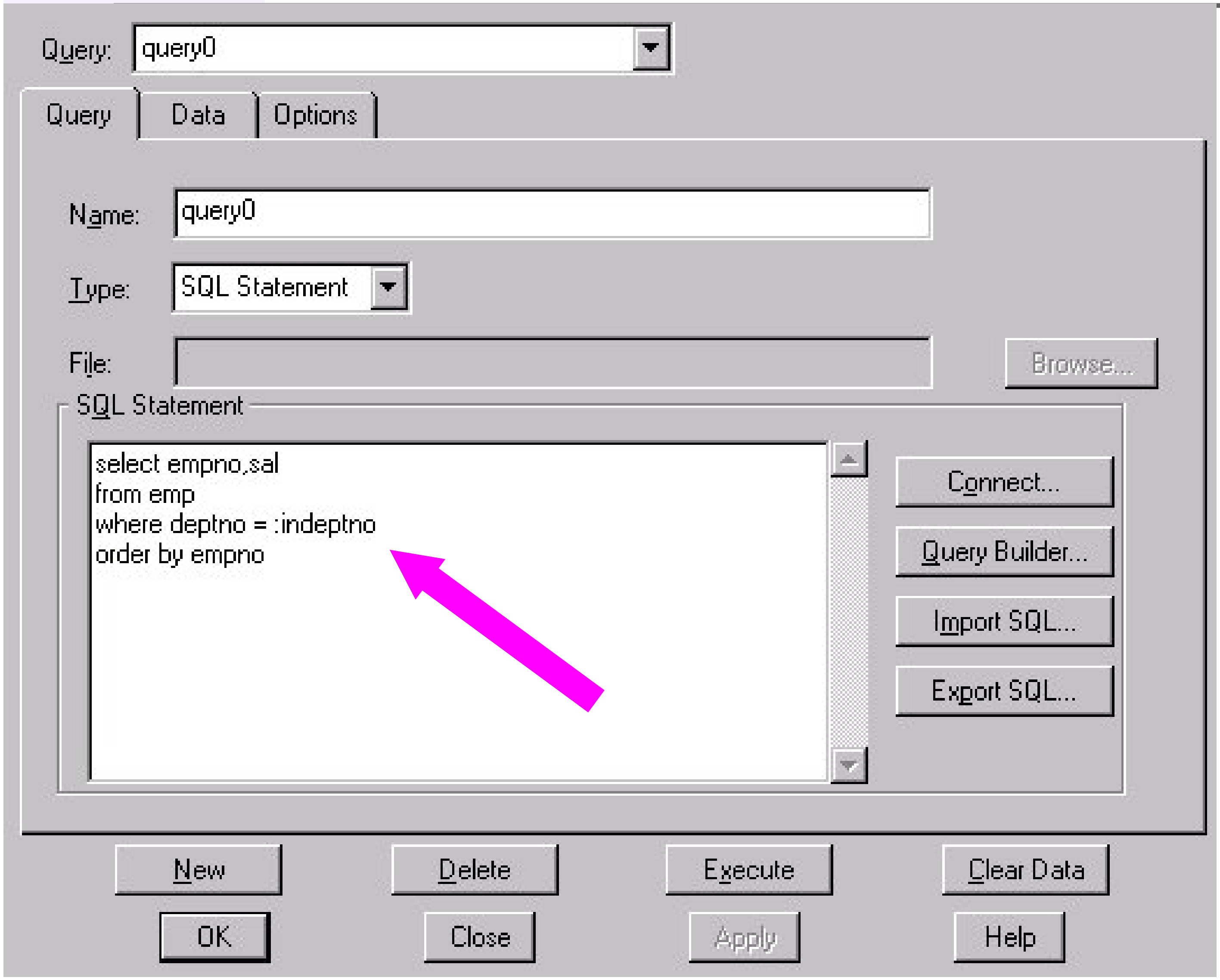
Parameter: INDEPTNO

Name: INDEPTNO

Type: Number

Initial Value:

Parameter in Graphic Query



Query: query0

Name: query0

Type: SQL Statement

File:

SQL Statement

```
select empno,sal
from emp
where deptno = :indeptno
order by empno
```

- Use the parameters in the query, or, anywhere in PL/SQL used by the graphic

Web Fonts

- Careful! Only six fonts are common to MS Windows, Motif, and Web-based forms:

Java/Web	Windows	Modif/X Windows	MacIntosh
Courier	Courier New	adobe-courier	Courier
Dialog	MS Sans Serif	b&h-lucida	Geneva
DialogInput	MS Sans Serif	b&h-lucidatypewriter	Geneva
Helvetica	Arial	adobe-helvetica	Helvetica
Symbol	WingDings	itc-zapfdingbats	Symbol
TimesRoman	Times New Roman	adobe-times	Times Roman

Conclusion

- This presentation illustrated the ease of using Oracle Graphics in making Oracle Forms applications more full-function
- Oracle Graphics is an easy to use, powerful tool allowing graphic representation of data based upon queries
- Oracle Forms triggers may be used to initiate the display of graphics and to pass values that control the actual graphic that is created
- The use of a picture (via Oracle Graphics) may well be worth a thousand words displayed individually or in summary on a form

King Training Resources

To contact the author:

John King
King Training Resources
6341 South Williams Street
Littleton, CO 80121-2627 USA
1.800.252.0652 - 1.303.798.5727
Email: john@kingtraining.com

www.kingtraining.com