Getting started with
Omnis Studio

Omnis Software

February 2000

Contents

What's in this book?

This book is aimed at the first-time Omnis Studio user. It provides enough basic information and practical exercises to get you up and running in developing software applications with Omnis. We recommend you work through this book before delving into the more detailed manuals such as *Using Omnis Studio*.

Getting started with Omnis Studio comprises the following sections:

The basics

What is a database?

Those sufficiently familiar with databases may skip this sub-section. It describes how databases store information and how applications built using Omnis Studio can work with existing databases or create new ones. It explains such concepts as:

- Tables

- Records

- Fields

- Primary Keys

- Creating an Omnis database within your application

- Using other types of database with your application

What is Omnis Studio?

This gives you a general understanding of what Omnis Studio is and what you can do with it. The subsections are:

- What kind of applications can I build?

- The Omnis Studio main screen:

    1. The Welcome library

    2. The main toolbar and menus

- The Library Browser

- The Component Store

- The Property Manager
- Writing and editing code in Omnis
- The steps commonly taken in building an application in Omnis Studio:
    1. Creating a library
    2. Adding classes
    3. Creating a database or working with an existing one
    4. Creating user interfaces/windows/web forms
    5. Adding menus
    6. Adding toolbars
    7. Creating reports/queries
    8. Releasing your application

The tutorial

Follow this quick tutorial and within minutes you'll build a database application (a portfolio of cover designs for books, CDs and TV programs) with an attractive user interface. It will even include a remote form to browse data via the web!

Tech Tips

At various points throughout the tutorial and the *Let's get deeper* section, we have inserted Tech Tips, little nuggets of technical information you can use to enhance your applications.

Let's get deeper

This takes you beyond the tutorial and looks deeper into application development. Without going into too much detail, it will start you off on the following tasks and point to where you can learn more about them:

- Adding a report
- Adding a menu
- Adding a toolbar
- Modifying the Startup Task

Where to next?

This suggests some possible areas to explore in furthering your Omnis application development skills.

Glossary

This provides a glossary of Omnis Studio terms and the features available to you when building applications.

Appendix 1 - Omnis naming conventions

This is a guide to our recommended conventions for naming various types of objects when developing Omnis applications.

Appendix 2 – Shortcut keys

This appendix lists some useful shortcut keys.

The basics

What is a database?

If you already know all about databases, you can skip this sub-section.

In simple terms, a database is a collection of data.

Chances are, you will ultimately be using Omnis Studio to build applications that will manipulate and present information extracted from existing databases, or, indeed, from databases built into the applications themselves. Before you start using Omnis Studio, it is therefore important to know how databases (and good ones in particular) are constructed.

Let's take a look at the typical components of a database.

Tables

These contain data. A table might hold such information as customers' names, addresses and phone numbers. So, you might ask, is a table the same as a database?

Well, a small database (that is, one that only stores a few details for each record) might only need one table. An example of this would be a computerized personal address book, where each record consists of a name and a few lines of address details, all held in one table. But what if you then wanted to store a lot more information about each person?

A common flaw amongst badly designed databases is that they store too much information in one table. They become slower, more difficult to maintain and less user-friendly. More importantly, it becomes harder to get useful information out of them.

A well-designed database will have the information divided into meaningful groups, with each group stored in its own table. For example, a software company might keep a customer database that has one table for contact names, another for addresses, another for products used and so on. The tables for each customer record are then linked using a primary key, that is, a field common to each table, such as the customer name or identification number.

Records

A record is all the information in a table that relates to one item. For example, the address details held in the addresses table for one customer would be considered one record.

Fields

A field is the storage space for a single piece of data held in a table. This might be for a name or a phone number.

In most databases (Omnis-generated ones included), when you create a field, you specify the type of data it is to store, such as characters, numbers, dates etc. This makes data entry and manipulation easier and less prone to error.

Primary Keys

A primary key is a field that links together all the information held in a number of tables for a single item. It might be the field that holds the customer name or the identification number. In this way, you could link together the address details, contact names, phone numbers and products used for a single customer. In the picture below, the Primary Key could be the Company Name, which would be common to all the tables:
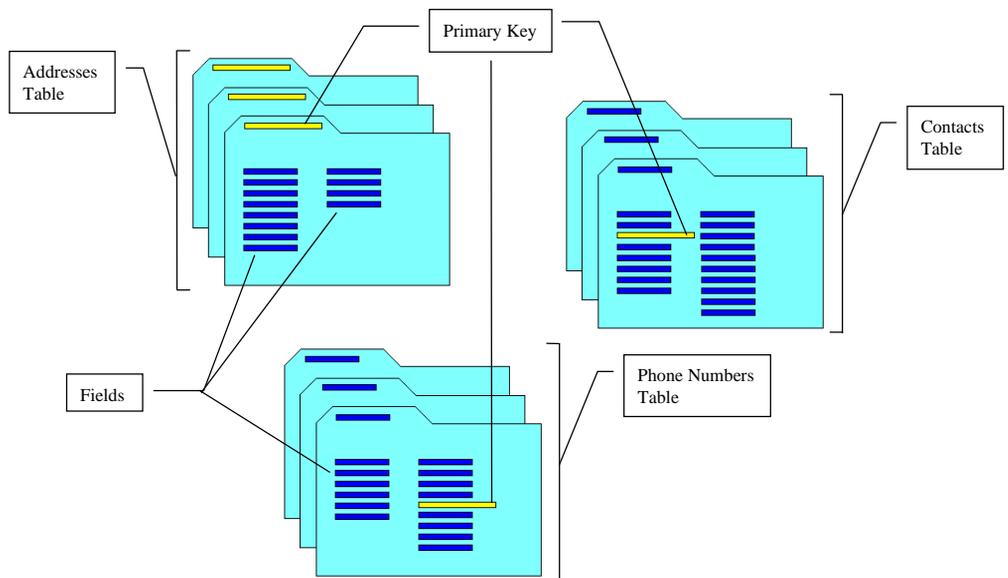
Creating an Omnis database within your application

As you will see when you do the tutorial, Omnis Studio has its own built-in database developer that lets you seamlessly create an Omnis database within your application. To access and manipulate an Omnis database, Omnis provides its own 4GL Database Manipulation Language (DML), or, if you prefer, you can use SQL (see the next sub-heading *Using other types of database with your application*). The working examples in this book use DML for simplicity and ease of use.

Using other types of database with your application

Omnis Studio also provides a tool (the SQL Browser) enabling you to build applications in which users can access and maintain proprietary databases such as Oracle, Sybase and Informix. Linking to other databases is not covered in this book; for more information, refer to the manual *Using Omnis Studio*, *Chapter 5 - Data Classes* and *Chapter 12 -Accessing Your Database* (available in PDF format on the Omnis Studio CD).

What is Omnis Studio?

Omnis Studio is a rapid application development (RAD) environment that supports component-based development and integration as well as providing access to a full range of databases and Internet services.

Well, that's the one-sentence "manual" definition, but what does that really mean to you? What makes Omnis Studio so special?

At the risk of sounding like an advertisement, Omnis Studio really does enable you to develop quality applications very quickly. It combines a simple user interface with powerful tools and versatile wizards, so you can create elegantly customized application components built of flawless code without getting bogged down in huge tracts of an esoteric programming language. And by quickly, we mean minutes instead of hours, hours instead of days, etc.

We will prove this by showing you. In the remainder of this section, we'll first give you some idea of the kind of applications you can build with Omnis Studio, then introduce you to its main interface and development tools. In our tutorial in the following section, within minutes you will build a simple database application with an attractive user interface and even a remote form to browse data via the web.

What kind of applications can I build?

With such a wide-ranging and easy-to-use product, Omnis Studio users have built sophisticated applications that handle payroll and expense tracking, human resources and administration, healthcare, electronic banking, commerce and even a professional quality virtual recording studio.

Check out the success stories at our website on www.omnis.net/partners/success/success.html.

The Omnis Studio main screen

Let's take a look at Omnis Studio. When you first open it, it looks like this:

Note that, throughout this book, the screenshots show the Windows version of Omnis Studio.

The Welcome window

In the center of the main screen, you have the Welcome window:

This gives you a number of quick-start options. For example, the New Users option opens a screen where you can open the TUTORIAL (as in the next section). The New Users screen also provides some other useful options; we particularly recommend you have a look at the Application Builder after you've worked through this book.

For now, though, click Close Welcome (or Close at the New Users screen) so you can have a look at Omnis Studio itself.
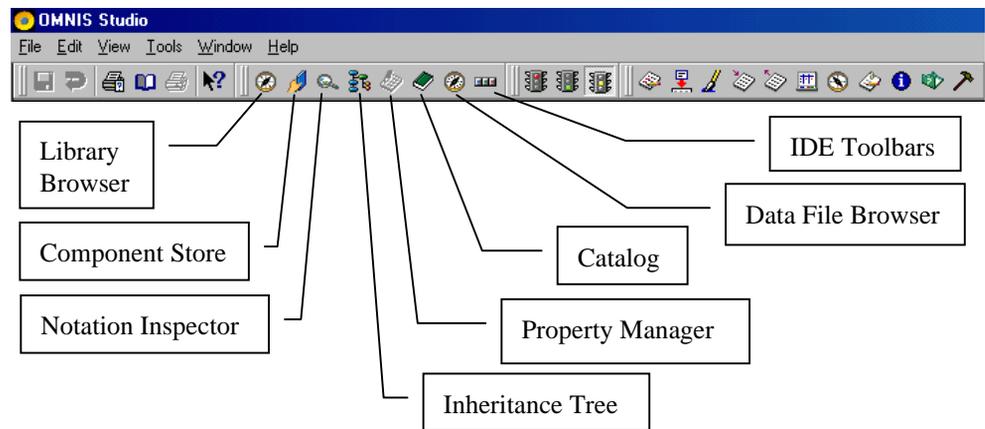
Note: Take care that you click Close Welcome to close the Welcome window and not "Click to toggle Startup State" (at the bottom), at least while you're getting to grips with Omnis Studio. The latter option will prevent the Welcome window from opening at startup. Having said that, you can open the New Users window at any time from the main screen toolbar and menu (Tools/New Users).

The main toolbar and menus

The main toolbar provides all the Omnis Studio tools you need to build an application. Most of these can also be found in the main View and Tools menus (the other buttons/options are general Windows ones).

The toolbar buttons are divided into three groups; View, Desktop and Tools. The pictures below indicate what feature each button opens (as do the ToolTips in the toolbar itself). Some of these features are described in this section; others provide more advanced development tools and so are not discussed in this book. Don't worry too much about what each button does at this stage; the buttons will become familiar to you the more you use Omnis Studio. In fact, you can build entire applications using just a few of the key ones, as you'll find when you do the tutorial.

View buttons

	Library Browser	This is where you create or open a library. In Omnis, a library holds all the components of an application.
	Component Store	This contains wizards and templates you can use to add classes and other objects to your library.
	Notation Inspector	This displays your Omnis object tree, including the contents of your libraries. This can be useful to get the correct notation/path for an object.
	Inheritance Tree	This displays the superclass/subclass structure of your libraries.
	Property Manager	This shows the properties for the currently object selected. You can edit the properties here.

	Catalog	This displays all the variables in your library and provides a convenient list of all Omnis functions and constants.
	Data File Browser	This allows you to open and manage Omnis datafiles.
	IDE Toolbars	This allows you to select which Omnis toolbars to display (see also Desktop buttons below).

Desktop buttons
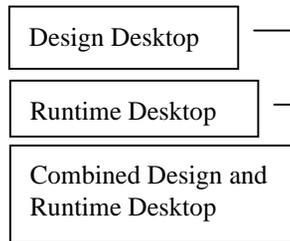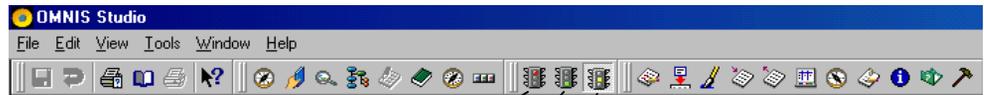
	Design Desktop	This displays the Omnis and standard Windows buttons only.
	Runtime Desktop	This displays just the buttons you have added to your application (that will appear during runtime).
	Combined Design and Runtime Desktop	This displays both types of the above buttons.

Tools buttons

Import Data

Export Data

Icon Editor

Trace Log

IDE Options

Version Control System

SQL Browser

Installs or removes the Adhoc Reports menu

New Users

Help Project Manager

Add-On tools
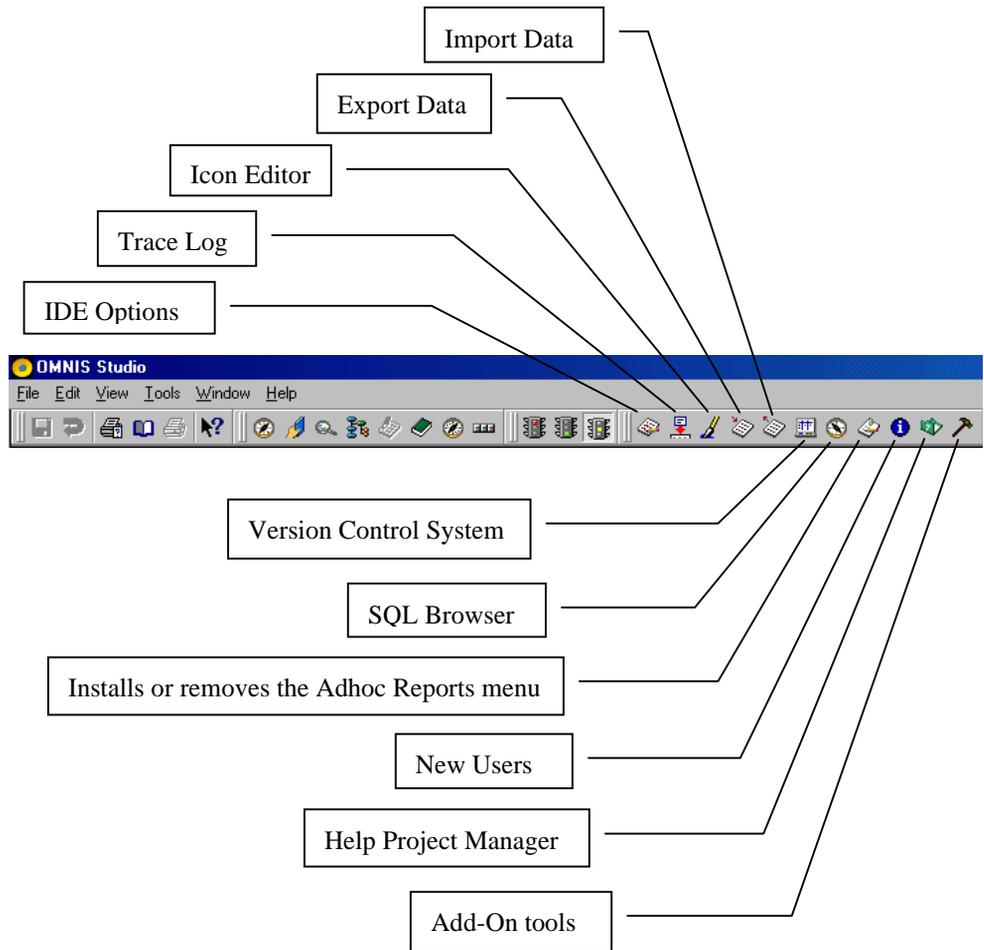
	IDE Options	This displays the Omnis preferences in the Property Manager. These preferences affect all libraries and Omnis as a whole.
	Trace Log	This opens the Omnis trace log, which lets you examine the flow of your code while it is executing.
	Icon Editor	This opens the Icon Editor, which lets you add your own icons to Omnis.
	Export Data	This lets you export data from an Omnis data file using a

number of different data formats.

|		
	Import Data	This lets you import data into a list from an existing export file or text file from another application.
	Version Control System	The Omnis Version Control System, or VCS, lets you control Omnis application development in a team environment.
	SQL Browser	This lets you access and maintain your proprietary databases such as Oracle, Sybase and Informix, and Omnis databases via Omnis SQL.
	Adhoc Reports menu	Installs (or, if already installed, removes) the Adhoc Reports menu.
	New Users	This opens the New Users window (see earlier sub-section *The Welcome window*).
	Help Project Manager	This lets you create new help projects for your application and modify existing ones.
	Add-On tools	This provides the following additional development tools:

- Method Checker

- ODBC Administration

- Port Profile Editor

- String Table Editor

- Web Client Tools

The Library Browser

As you can see, when you open Omnis Studio, it automatically opens a Library Browser for you:
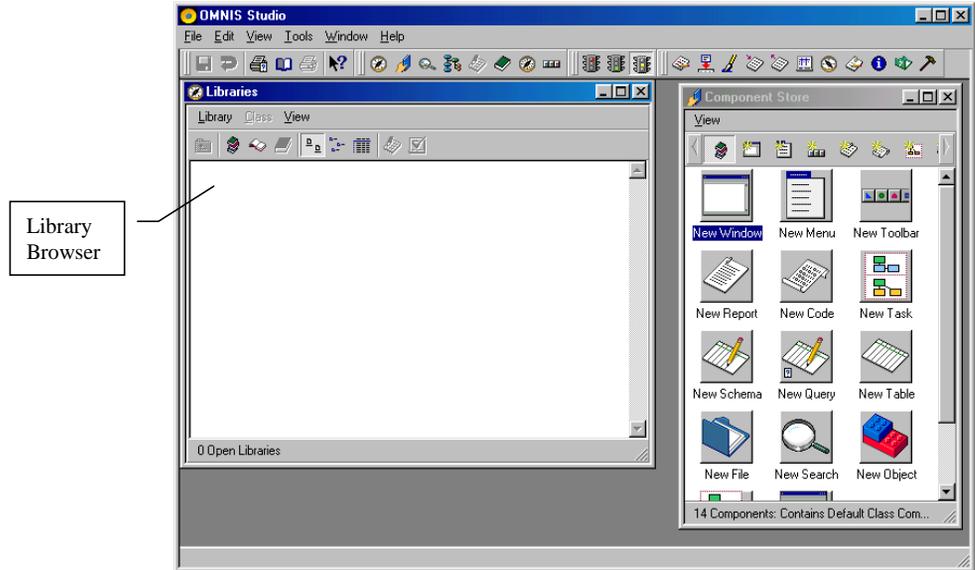
Library Browser

This is where you create or open a library. In Omnis, a library holds all the components of an application.

When you build a new application, the first thing you do is create a library, then add the application components you need into it.

If you want to work on an existing application, you open its library in the Library Browser, then open the components you want to work on.

You could also build more complex applications that comprise a number of libraries.

The Startup Task

It's worth noting here that, when you create a library, Omnis Studio automatically creates a Startup Task in it for you:
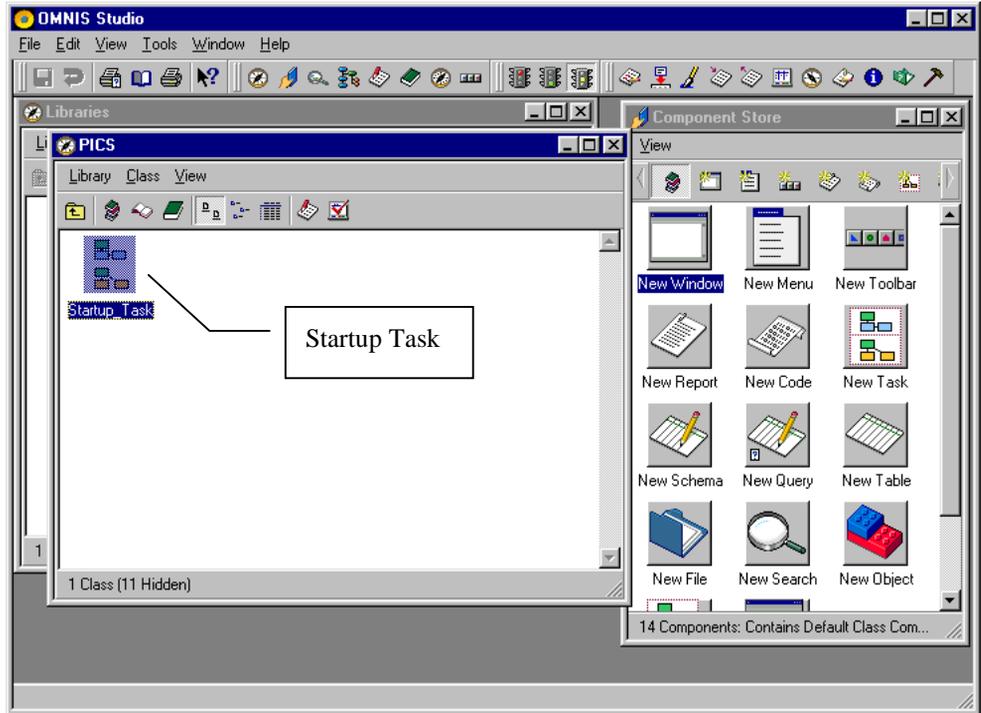
This runs the initialization code whenever you open the library, thus providing the functionality you need to run your application.

Whilst you're getting to grips with Omnis Studio and developing a simple application via the tutorial, you need not concern yourself with the Startup Task.

Later on, though, you may develop an application where you want something to happen when you open the library. For example, you might want an "About" screen or install menu to appear. To do this, you would insert additional code into the Startup Task. This is quite easy to do, as you will discover in the later section "Let's get deeper" under "Modifying the Startup Task".

The Component Store

Once you have your library, you can start to add into it the application components you need. This is where the Component Store comes into its own. In many cases, adding a component (such as a window or a report) is simply a case of dragging the appropriate wizard from the Component Store and dropping it into your library, as in the picture below:
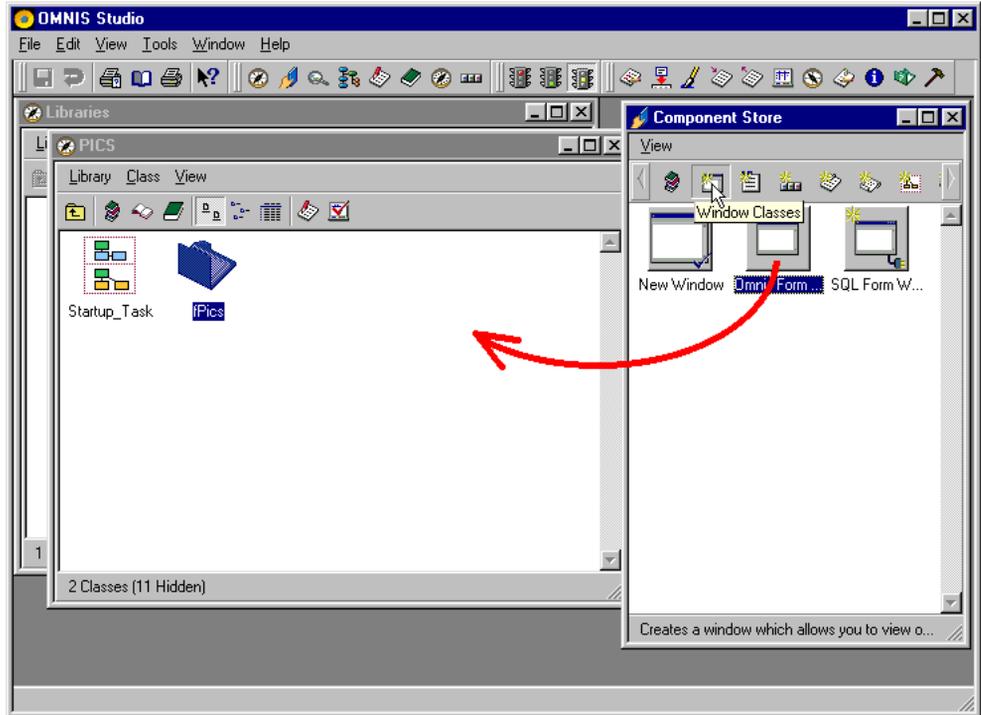
When you give it a name and click it, the wizard runs and prompts you to enter the relevant details. The Component Store also provides other components in the form of templates that you drag into your library and alter to suit your needs.

In Omnis, the application components held in a library are called Classes and may include windows, menus, table schemas, reports, web forms etc.

Take a look at this user interface window for a simple database application (you will create this in the tutorial):
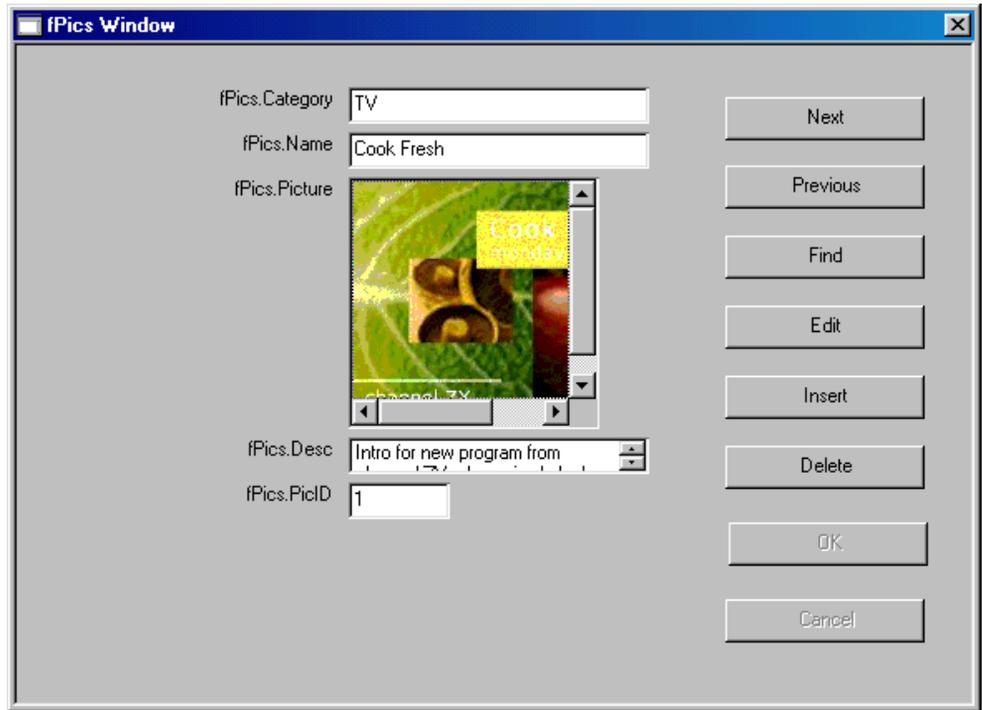
To get this, we first created a library (called PICS.LBS) then, from the Component Store, added two classes; a file class (to map records from the sample data file) and the window class itself (created in a matter of seconds using a wizard).

For the window to work, we used the Data File Browser to open the sample data file. To open and test the window in runtime mode, we just right-clicked (Ctrl-clicked) the window class object and selected the Open Window pop-up menu option.

When you do the tutorial in the next section, you will see how easy it all is.

The Property Manager

At any point in your application development, you can fine-tune any object (be it a library, class, button etc.) by editing its properties in the Property Manager. Just click the object and click the Property Manager button (or right-click/Ctrl-click the object and select the Properties pop-up menu option). This opens the Property Manager for the object. The picture below shows the Property Manager for a window class:
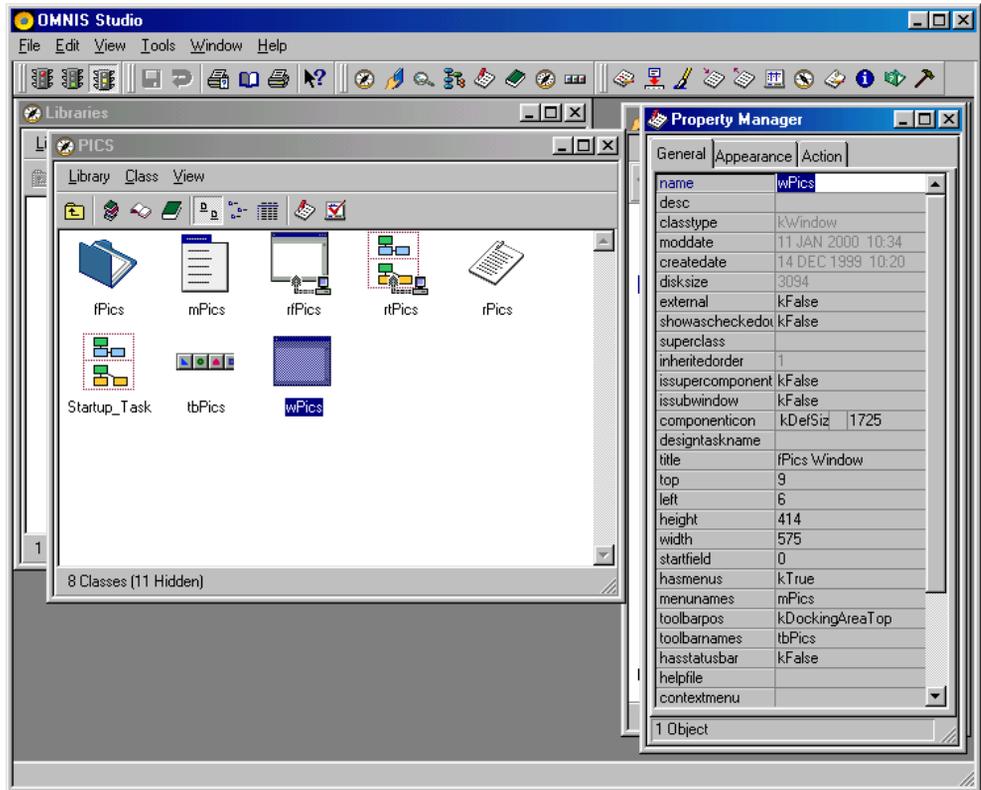

Click on another object and the Property Manager changes to show that object's properties.

Writing and editing code in Omnis

Thanks to the versatility of the Component Store's wizards and templates, and the familiar interface of the Property Manager, you can build quite complex applications without ever needing to get down to code level. But if you ever want to do any low-level fine-tuning, you will find code writing in Omnis extremely intuitive, powerful and compact, made especially easy with interactive editing tools such as automated keyword expansion and selection.

For example, to get an object (such as a toolbar button) to perform a specific task, you would double-click it to display its Methods editor and enter the required code. Here's a typical Methods editor; in this case, for a toolbar button that will print a report when clicked:
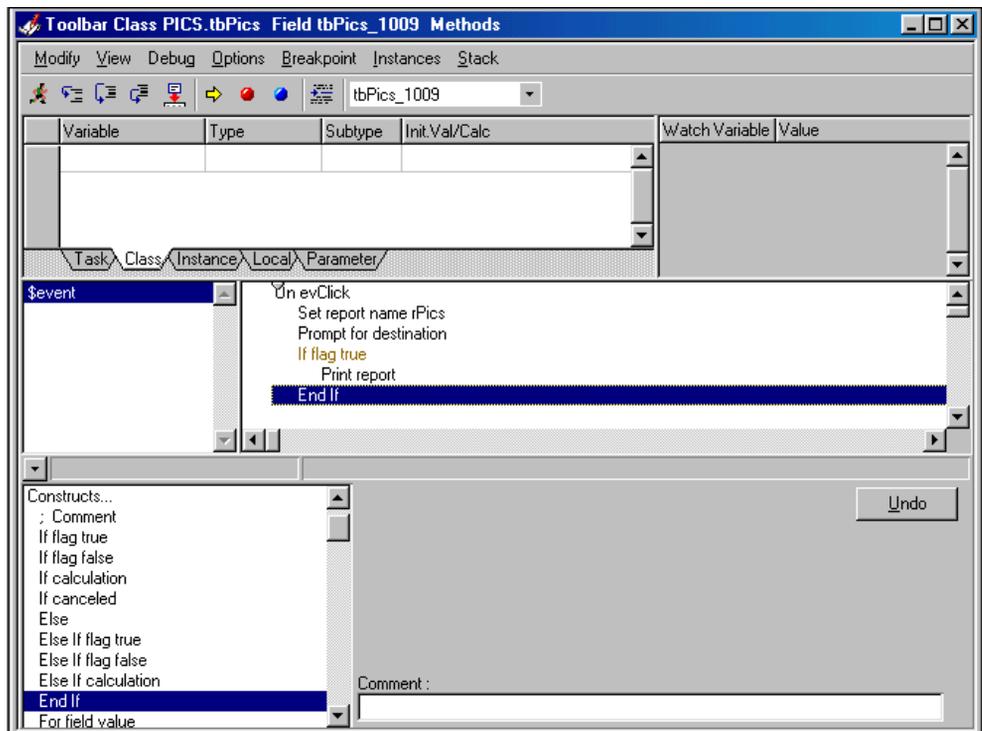
To add a new line of code in a Methods editor, you click to highlight the line into which you want to insert the code, then type it in. You don't need to worry about upper/lower case or indentation; Omnis takes care of these automatically. As you type, Omnis tries to anticipate the command you're entering and inserts it for you. Basically, you just keep typing until the correct command appears. Another way to insert code is to locate the command you want in the list at the bottom left of the Methods editor. You then just click it and Omnis inserts it in the curent line.

If you enter a command where you need to specify the name of an existing object (as in "Set report name rPics" in the picture above), Omnis recognizes this and, as soon as it has established you're entering such a command, provides a list of existing objects. You just click the one you want and it appears in the command line automatically.

You will discover just how easy writing Omnis code is when you come to step 5 in the later section "Let's get deeper".

Building an application step-by-step

So, with Omnis Studio, you can break down the development cycle of an application into the following basic steps:

Note: It is important you determine at the outset whether you are going to create (or work with) an Omnis database using DML or link to an existing one (including Oracle, Sybase and Informix) using SQL. This will affect the type of database/file classes you will need to add to your library from the Component Store.

1. Create a library

In the Library Browser, click New Library and give it a name.

2. Add the application components (classes)

Double-click your library to open it, then from the Component Store, drag the wizard or template for any component you require and drop it into your library. Give it a name and enter the relevant details.

Using this method, you can add such components (classes) as:

Windows

Menus

Toolbars

File classes or Schema/Query and Table classes, depending on whether you are creating (or working with) an Omnis database using DML or linking to an existing one (including Oracle, Sybase or Informix) using SQL

Reports

Web forms (Remote forms)

3. Fine-tune

Click any object in your application and click the Property Manager button (or right-click/Ctrl-click the object and select the Properties pop-up menu option). This opens the Property Manager for the object. Edit the properties as required.

Click on another object and the Property Manager changes to show that object's properties.

To get an object (such as a toolbar button) to perform a specific task, double-click it to display its Methods editor and enter the required code.

4. Test

Omnis Studio lets you test your application at any time during the development process.

For example, if you have created a window to display details from an existing data file, you can simply open it in runtime mode. First, use the Data File Browser to open the sample data file and make it the current one (using the Make Current option). Next, right-click (Ctrl-click) the window class object and select the Open Window pop-up menu option.

5. Release your application

The Omnis Studio CD provides a runtime version of the Omnis Studio executable or program (for each platform) that contains all the necessary elements for your end user to install and run your application. This procedure is not covered here; refer to the manual *Using Omnis Studio*, *Chapter 15 - Deploying your Application* (available in PDF format on the Omnis Studio CD).

The tutorial

If you've already worked through the tutorial and created the picture database application (you might have done this from the Welcome screen when you first opened Omnis Studio), go straight on to the next section Let's Get Deeper, where you will expand what you've already created.

This tutorial shows you how to create an Omnis application to browse a picture database, in this case, a portfolio of sample TV programs, CD, and book designs undertaken by a design agency. You could, however, use the application to store any type of picture data (such as a photo library).

The tutorial shows you how to create the file structure required in the database and how to create a simple data entry window to browse and insert new pictures. Finally, it shows you how to create a form to browse the database in a web browser. At the end of the section, we describe how you can deploy the application to the web, although you may need help from your webmaster to do this.

1. If you are running this tutorial from the Welcome window, you won't need to start Omnis Studio; otherwise, double-click on the Omnis icon on your Desktop (or in Windows Explorer) to start it. When you start Omnis Studio, the Library or *Class Browser* (on the left) and the *Component Store* (on the right) should appear. You'll see what these do later. If any Omnis applications are currently open, they will appear in the Browser.
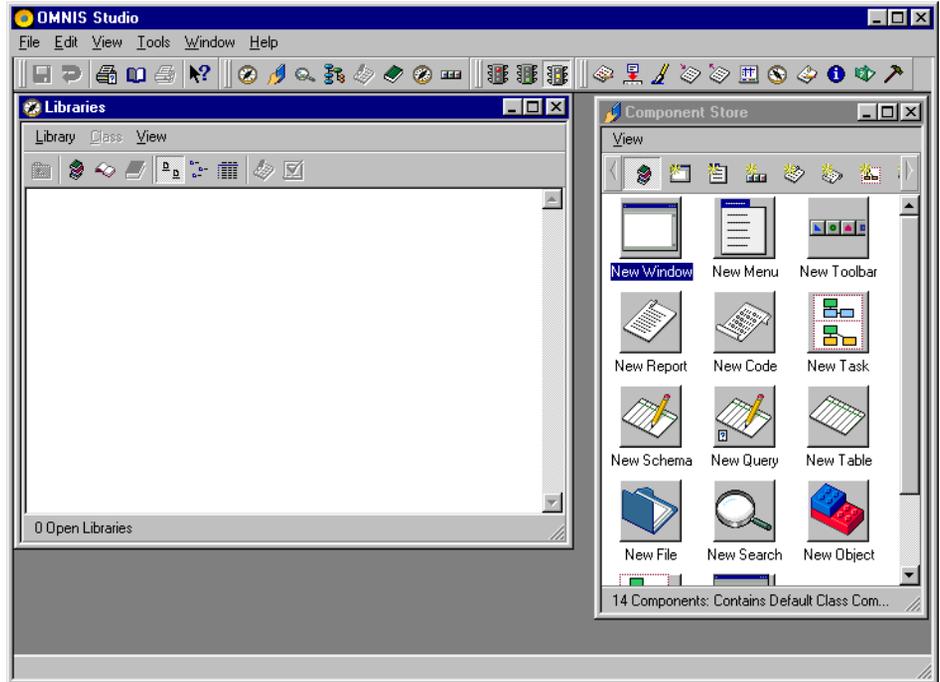
2. The starting point for your Omnis application is an *Omnis library*. A library stores all the windows, menus, reports, and other objects in your application. To create a library, press F2/Cmnd-2 or click on the Browser to bring it to the top, then select the **Library>>New** menu option on the Browser menu bar. Open the Welcome/Tutorial folder, enter the name **PICS.LBS**, including the extension .LBS, and click on OK (details of Omnis naming conventions are provided in the appendix).
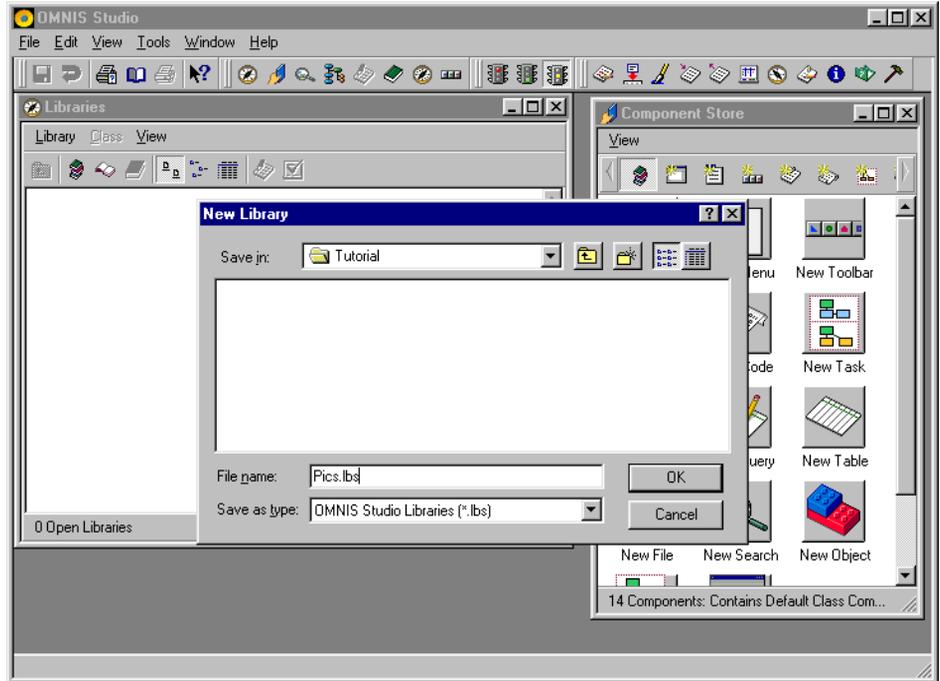
3. When you create or open a library it appears in the Browser. To view the contents of a library you double-click on the library icon. A second Browser window is opened displaying the contents of your library. At this stage, your library contains a *Startup Task* that simply initializes the library when it starts up.
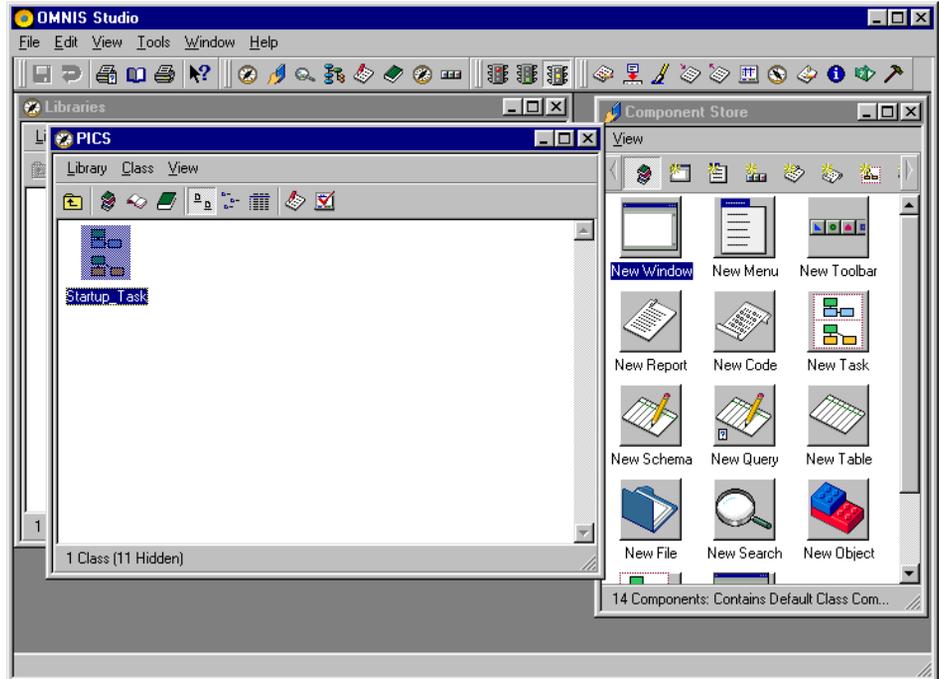
4. To create the database structure in your library you need to create an Omnis *file class*. File classes define the column names and data types for the information stored in your database. To create a file class, press F3/Cmnd-3 or click on the Component Store to bring it to the top. Drag the New File icon onto the Browser, release the mouse, name the class fPics, and press Return; fPics appears in the Browser.
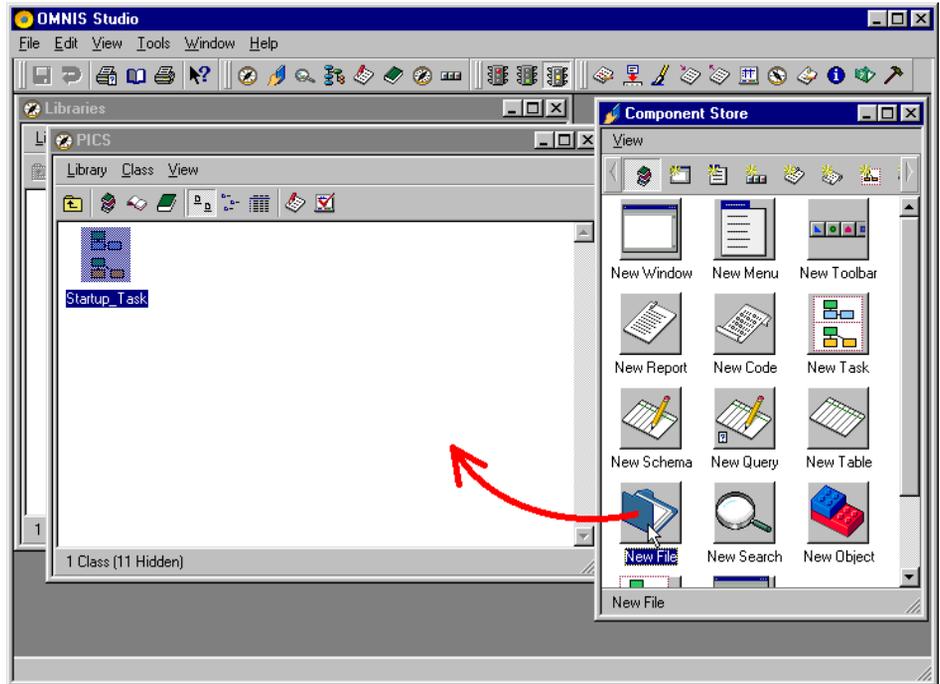
5. Double-click on the fPics file to open the file class editor. The file editor lets you add the name and *data type* of each *column* or field in your database. To create the first column or field, enter the name **Category**. The name is entered at the bottom of the editor window. Next, tab to the Field type box and choose the *Character* type. Next, tab to the Max length box and enter 50 as the maximum length. Lastly, click on the Indexed check box (near the bottom right corner of the window). Don't worry about the other check boxes and fields in the file editor, you can accept all the default settings.
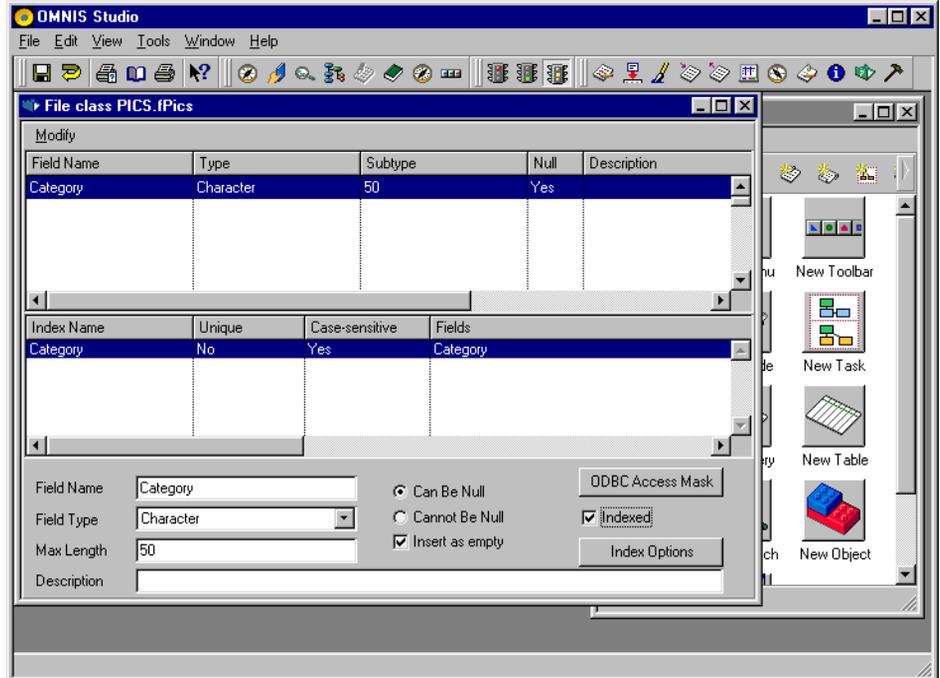
6. To create a second column or field in the fPics file class press Ctrl/Cmnd-N, enter the name **Name** in the Field name box, choose *Character* as the field type, enter 100 as the maximum length, and check the Indexed check box. To create each new column or field in the file class, press Ctrl/Cmnd-N, enter the field name, and select the appropriate field types and subtypes, as follows. The *Sequence* field called PicID will provide a unique reference for each data record.

Field name	Type/Length	Indexed?
Category	Character/50	Yes
Name	Character/100	Yes
Picture	Picture/No	
Desc	Character/1000	No
PicID	Sequence	Indexed automatically
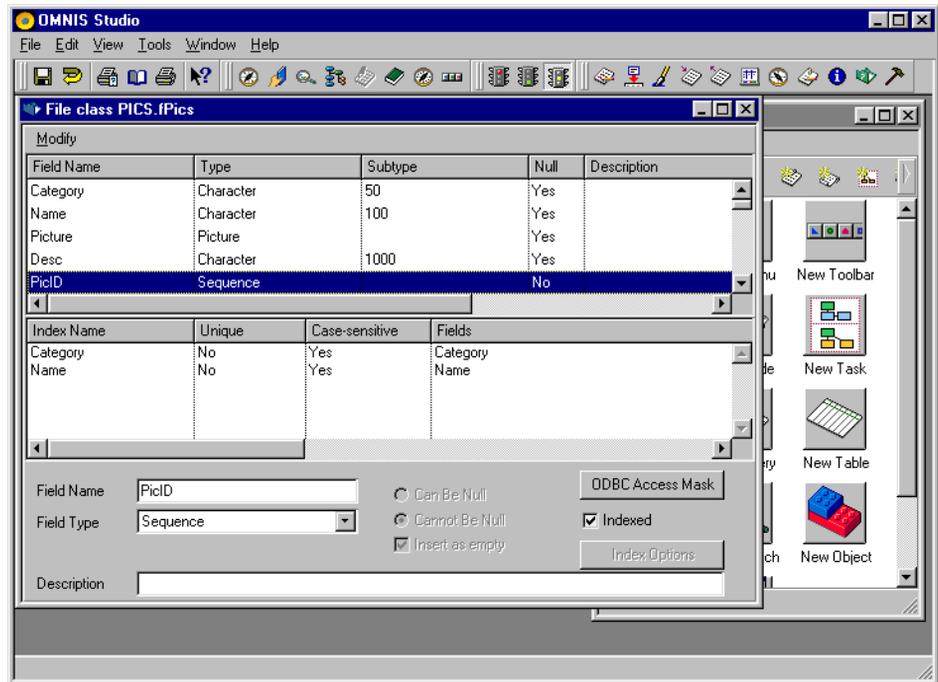
7. When you have finished modifying a class in Omnis you can simply close it to save it, or you can use the **Save** option from the *File menu* at any time. So, to save the fPics file class, close the file class editor. The browser should contain 2 classes, the Startup_Task and fPics.
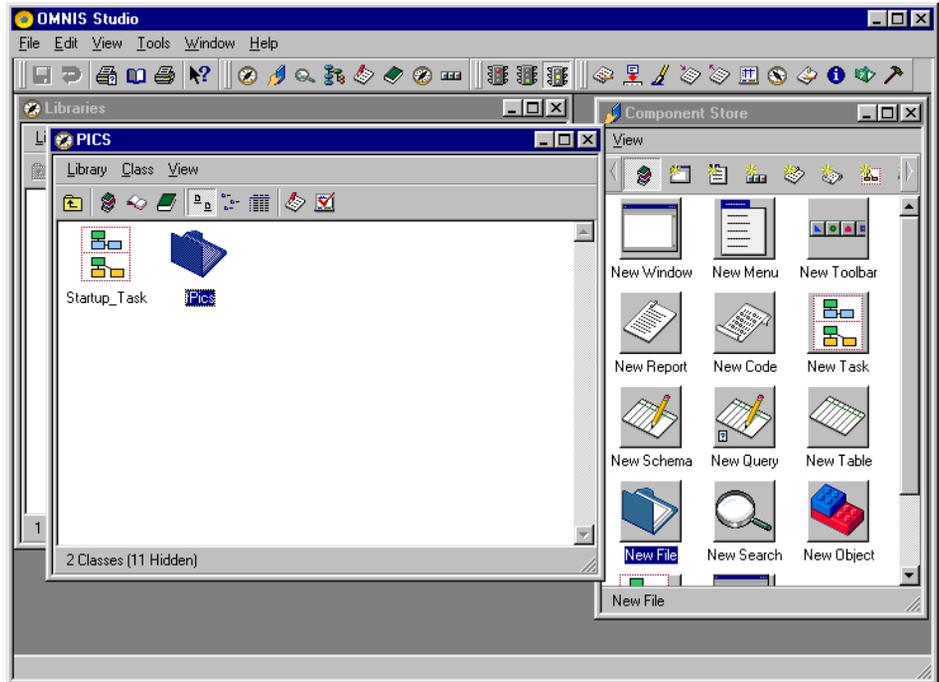
8. Having created the data file structure for your database, you can create a window or *form* to browse and insert data. Press F3/Cmnd-3 to open the Component Store or bring it to the top. Click on the Window Classes button in the top toolbar (second from left). Drag the Omnis Form *Wizard* from the Component Store and drop it on your library in the Class Browser. Name the new form **wPics** and press Return. A wizard window is displayed.
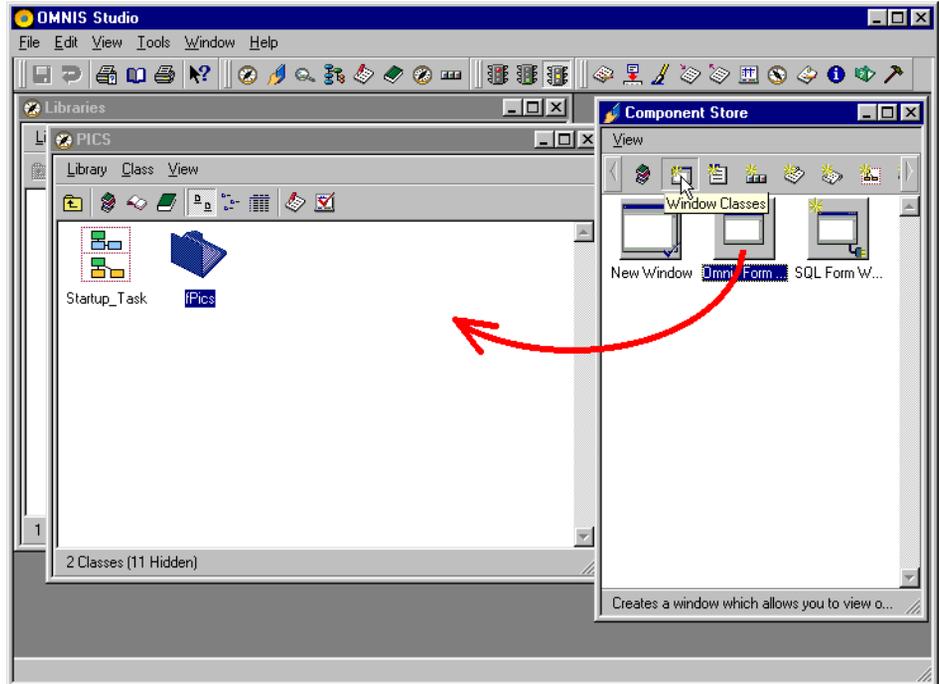
9. The form wizard lets you choose exactly what you want to include in your window or form. Keep the default setting, **One field per file field**, and click on the Next button.
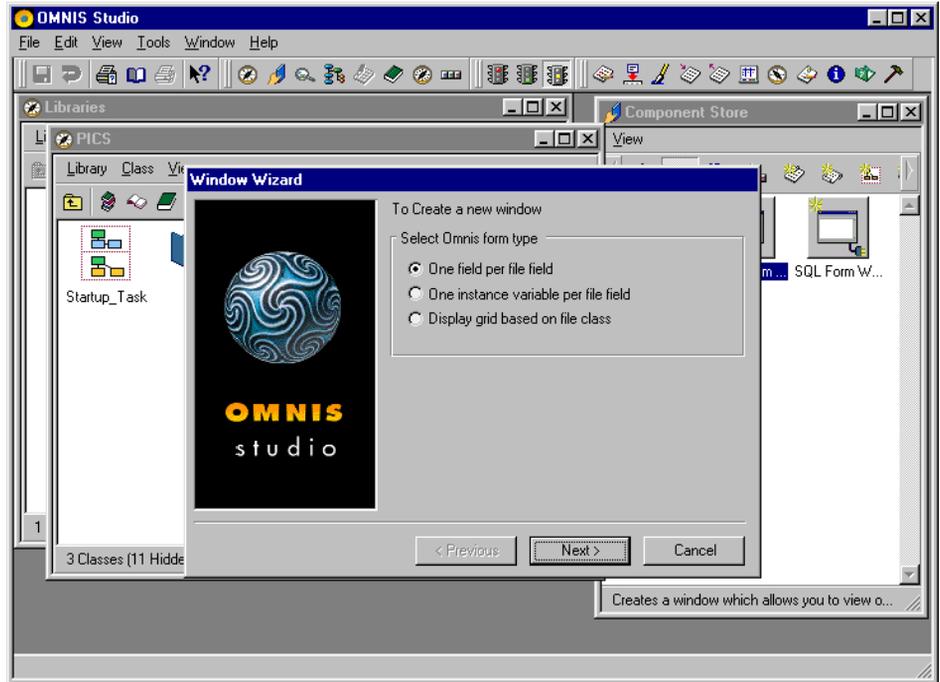
10. Expand the tree list by clicking on the + button. You should see the columns or fields you created in the file class: Name, Category, Picture, Desc, and PicID. The wizard lets you select individual fields, but in this case, you can include all the fields in the form, so check the box at the top next to the fPics file name.
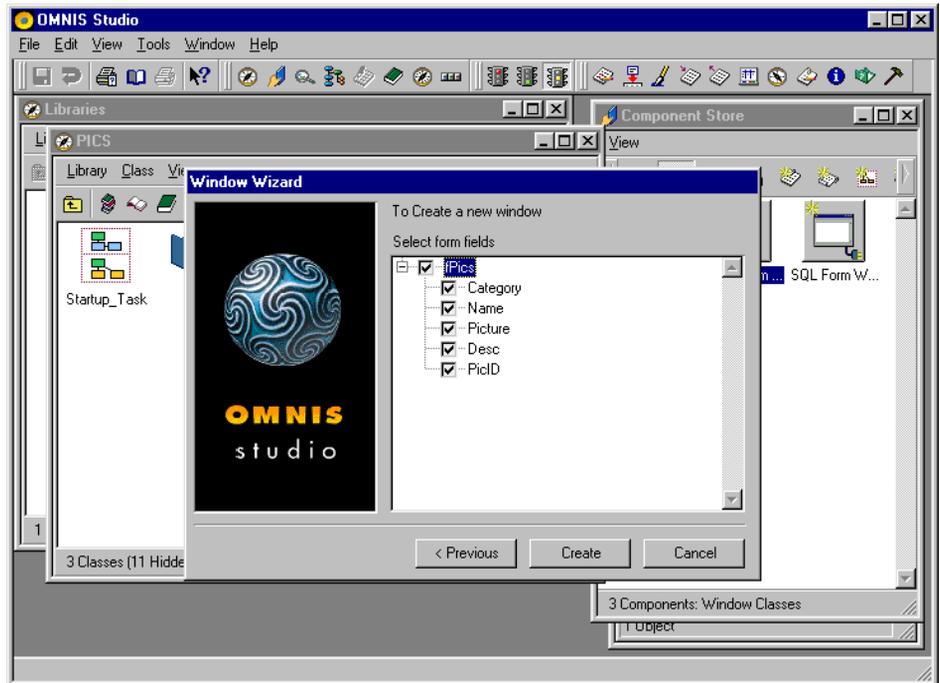
11. When the details for your form are complete, click on the Create button. Omnis creates the *form* for you automatically, based on the selections you have made in the wizard. When the wizard finishes, the form opens in design mode. You may notice the *Property Manager* is opened (on the right) showing the properties of the form, but you don't need to use it at this stage.
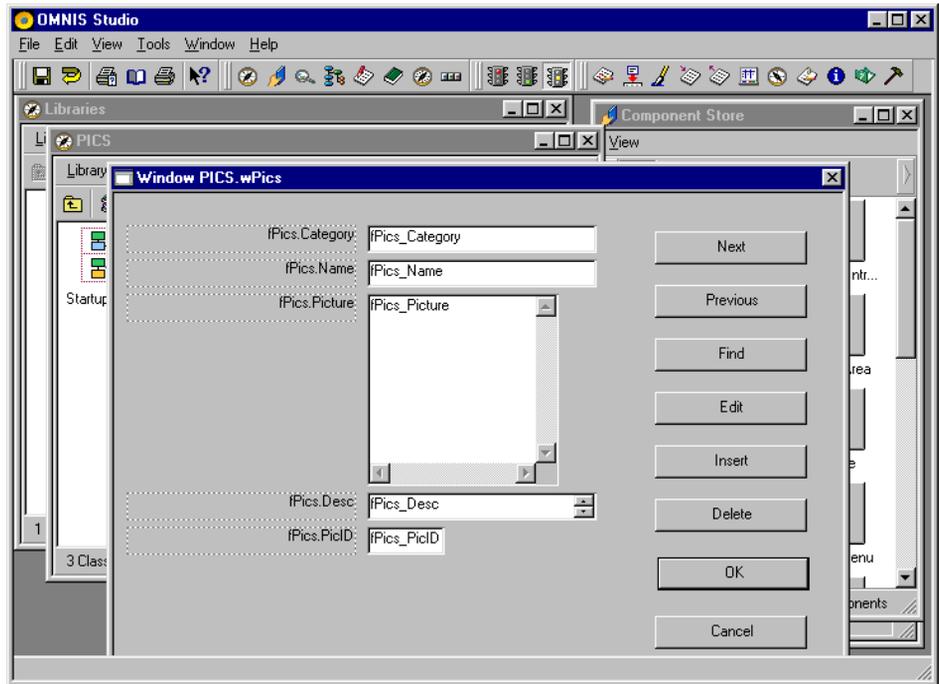
Tech Tip: You can control the order in which the user can tab from one field to the next in a data entry window.

*Omnis assigns a number to each field as you place it on your window. This number is stored in the **order** property for the field. You can see the order number for a field in the Property Manager, or you can **Show Field Numbers** for all the fields on a window using the context menu for the window. You can reorder the fields on your window, and hence change the tabbing order, by changing the order property for a field. When you change the order number for one field, the number for all the other fields on the window will change.*

12. Before you can try out the window you need to open the picture database or *data file*. Select the **Data file Browser** option from the **View** menu on the main Omnis menubar. Select the **Open** option from the **Datafile** menu in the Data File Browser. Open the welcome folder (in your Omnis program's tutorial folder) and double-click on the Omnis datafile called **PICS.DF1**. Make sure the PICS datafile is selected in the Data File Browser, then select the **Make Current** option from the **Datafile** menu; an arrow appears next to the PICS datafile icon to show it's the current one. Finally, close the Data File Browser.

Note that if you want to create a database of your own, you simply need to create a new datafile in the Data File Browser and make it the current one.
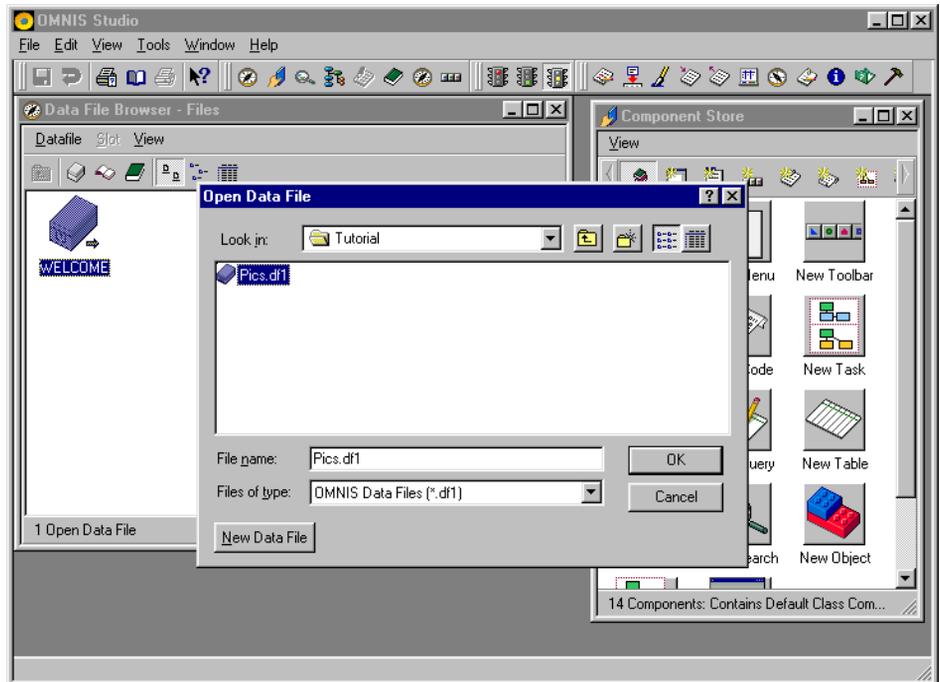
13. In Omnis it's very easy to try out a window or form. Assuming the wPics form is the top window, press Ctrl/Cmnd-T to open or "Test" your window. The window is opened on top of the design window. Click on the Next button to locate the first record in the database. Click on Next a few more times to open each record in turn. You can click on Previous to go back to the previous record. When you reach the end of the database Omnis bleeps, but you can click Next again to locate the first record.

Note that you can move fields, and change other properties of fields and the window itself at any time, and use Ctrl/Cmnd-T to toggle between design mode and runtime to try out your modifications.

Important: You must be in runtime mode to enter data and design mode to continue with the tutorial.
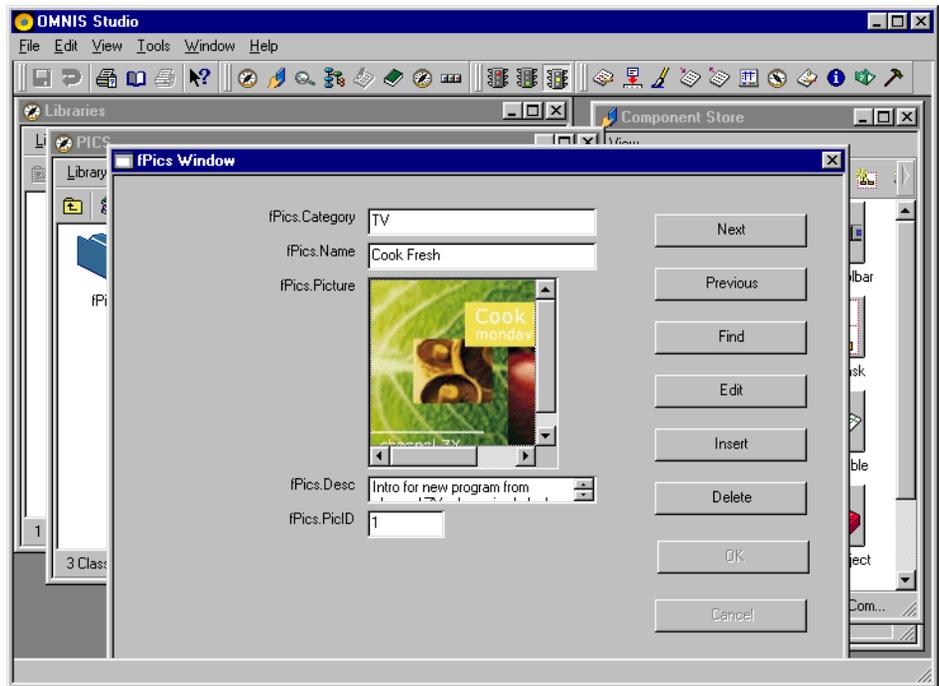
14. You can use the same window to browse and insert data into the database. With the window still open in runtime, click on the **Insert** button and enter the details below, using the tab key to go from one entry field to the next. To insert the picture, tab to the picture field, select Paste From File in the *Edit menu*, select *.jpg from the file type list, locate the **bookchaos.jpg** file in the Welcome/Tutorial folder, and click on OK. You do not need to enter anything into the PicID field, Omnis assigns a number automatically which uniquely identifies each data record. To save the complete data record, click on the **OK** button in the data entry window.

Field   Enter

Category Book

Name   Science in Chaos

Picture   bookchaos.jpg

Desc   Cover art for Bob Zurich's latest book, etc.

PicID   Leave blank: assigned automatically
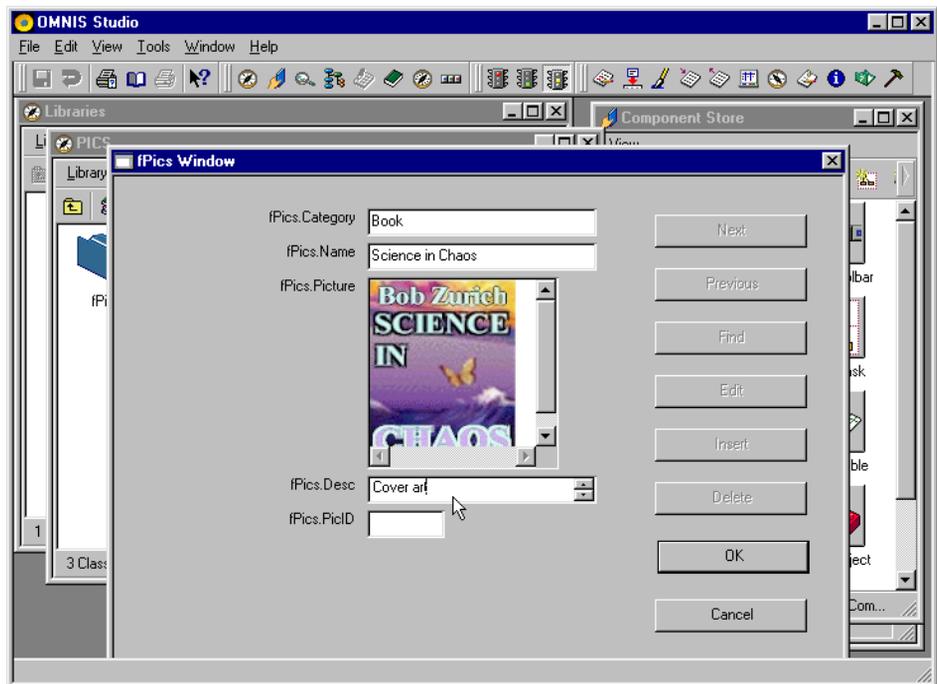
15. Having inserted some data, close the wPics window and its design window; you can also close the Property Manager if you like. Now would be a good time to save your library. To save your library, press F2/Cmnd-2 or click on the Class Browser to bring it to the top, and select **Library>>Save** from the browser menubar. At this stage you should have 3 classes, including the file and window classes you created, and the default Startup_Task.
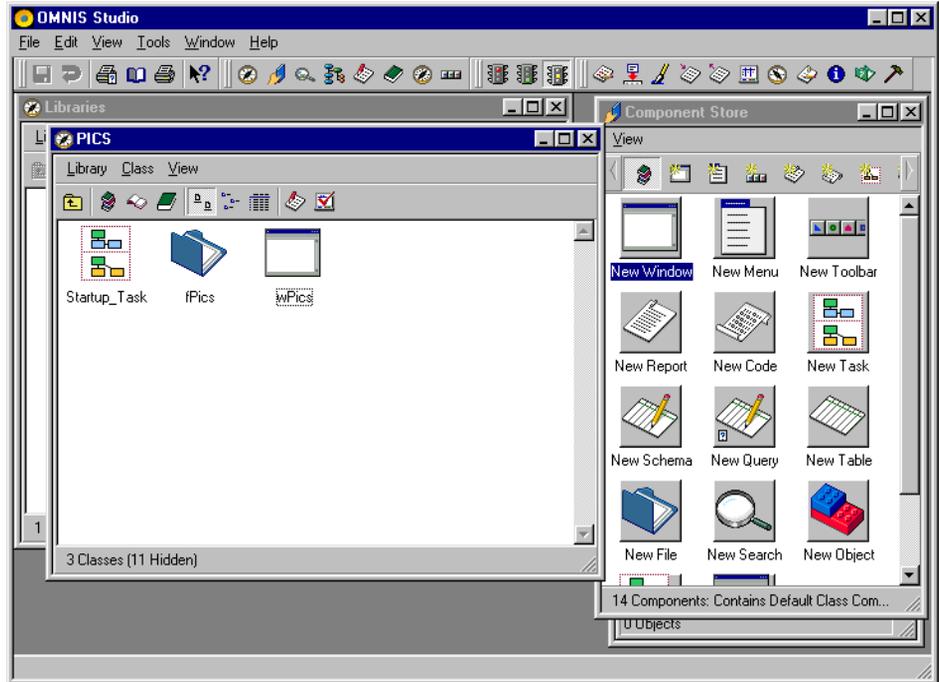
16. Next you are going to create a *remote form* or web form to browse your database in a *web browser*. You can place the same form on a website to allow anyone in the world to browse your picture database. To create the web form, press F3/Cmnd-3 or click on the *Component Store* to bring it to the top. Scroll the toolbar at the top of the Component Store and click on the Remote Form Classes button. Drag the *File Form wizard* onto your library in the Browser, enter the name **rfPics**, and hit Return. A wizard window is displayed.
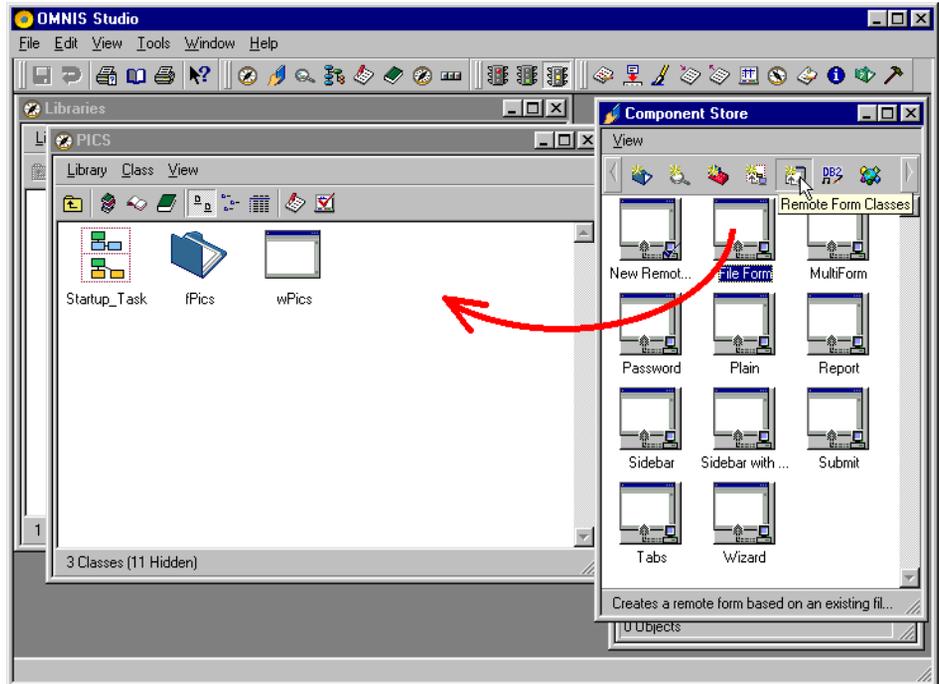
17. First you need to create a *remote task* for your form; a remote task simply handles the connection between your web browser and the server application, in this case, your picture database. Select **Create New Remote Task** and select the **Plain Remote Task** template from the Remote Task Template droplist, enter the name **rtPics** in the New Remote Task Name box (note the prefix is "rt" this time) and click on the Next button.
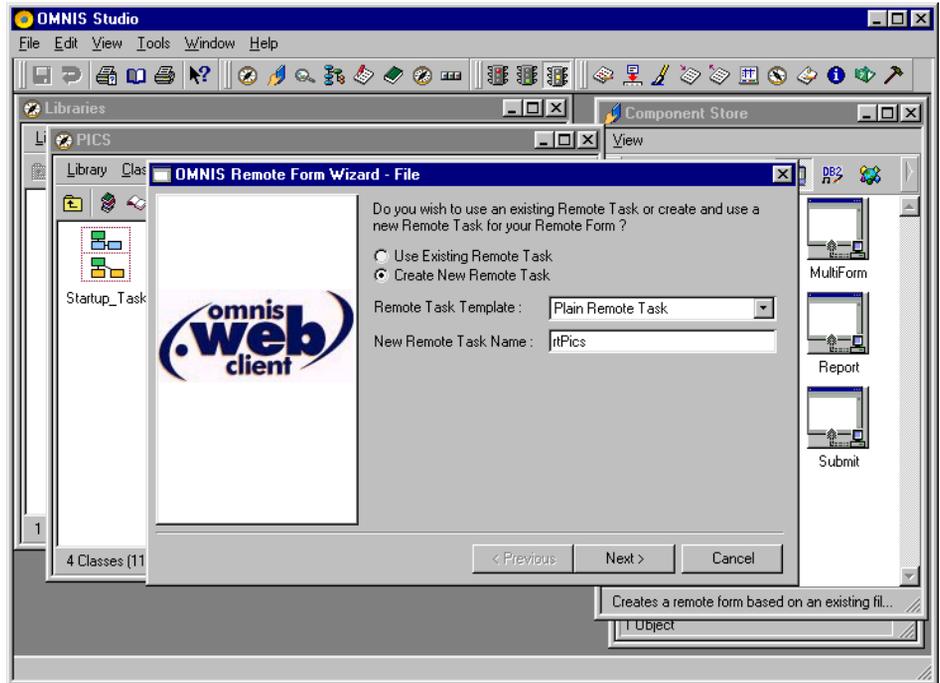
18. Expand the tree list by clicking on the + button. You should see the columns or fields you created in the file class: Category, Name, Picture, Desc, and PicID. You don't need to display the picture ID field in the web form so leave the PicID field unchecked, but check all the other fields. When the field selections are correct click on Next.
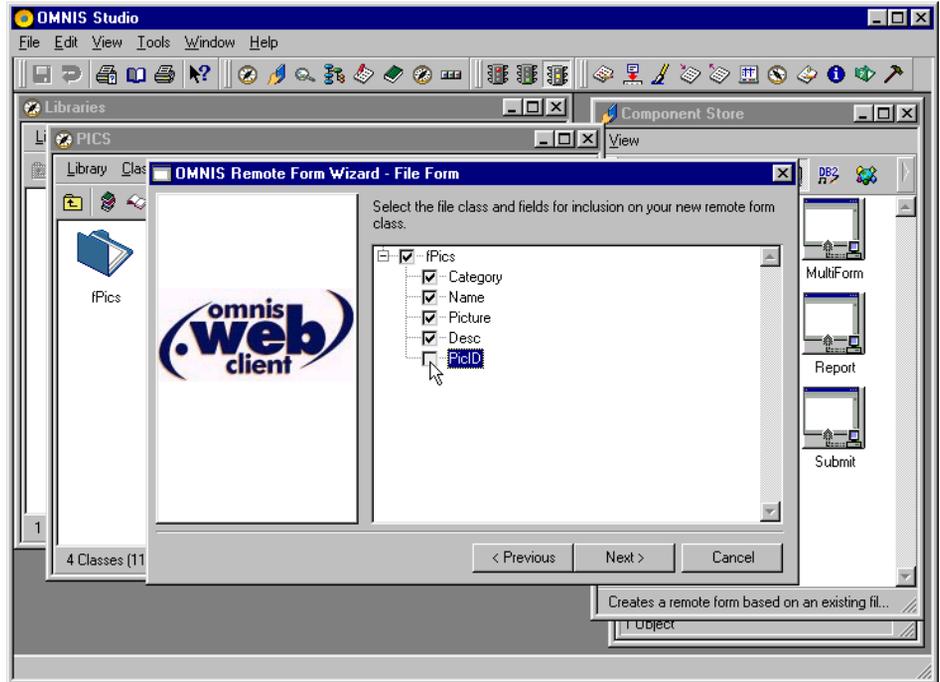
19. Next the form wizard asks you to select a field to be used as the *unique key* field. This is simply a column or field in your database that uniquely identifies each data *record*. You can use the PicID field which stores a unique number or reference for each record. Therefore check the PicID in the list and click on Next.
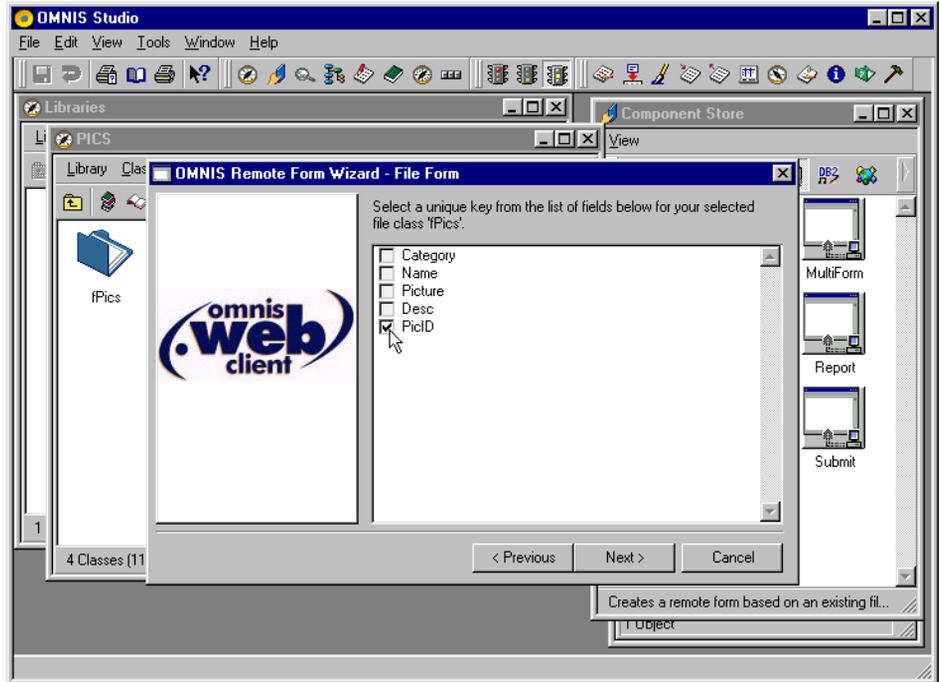
20. Next the wizard lets you choose the type of *web browser* for your form; you can select either Internet Explorer or Netscape Navigator, or create a form for both. For this tutorial, select either the Internet Explorer or Netscape option (whichever is your default browser) and click on the Create button. Omnis creates the rfPics web form for you automatically and displays it in the Omnis Browser. Double-click on rfPics to open it in design mode; it looks very similar to the data entry window you created earlier.
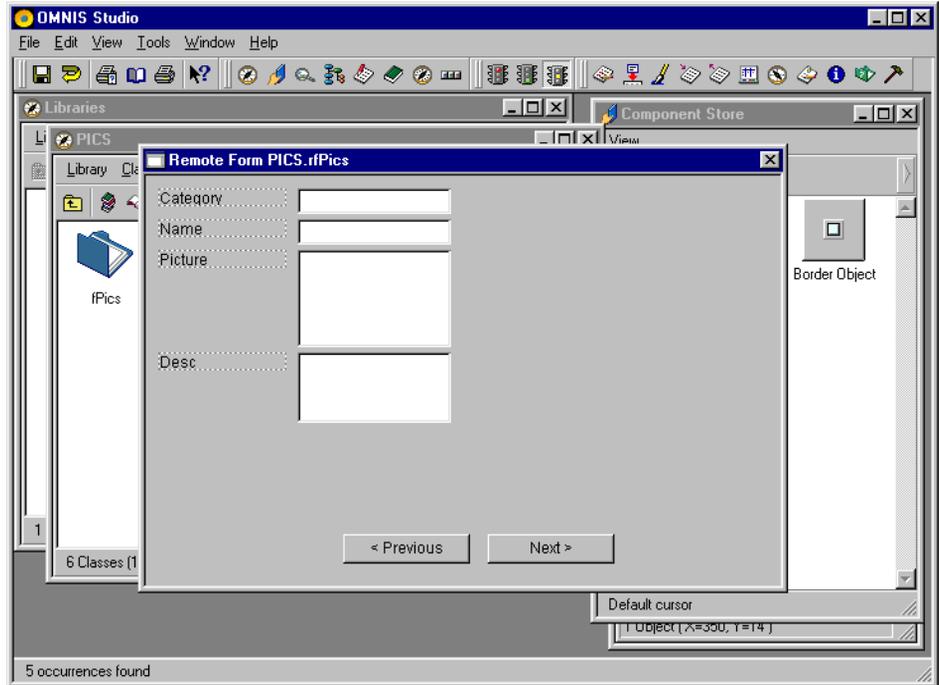
21. You can open or "test" your web form locally, before adding it to your website, by pressing Ctrl/Cmnd-T. Omnis opens your web browser automatically and displays the web form. Assuming the Picture database is still open, Omnis opens the first record, otherwise you'll need to open the PICS.DF1 data file, using the Data File Browser as before.

Note that all the pictures are scaled to fit the picture field, but you can change this in a moment.
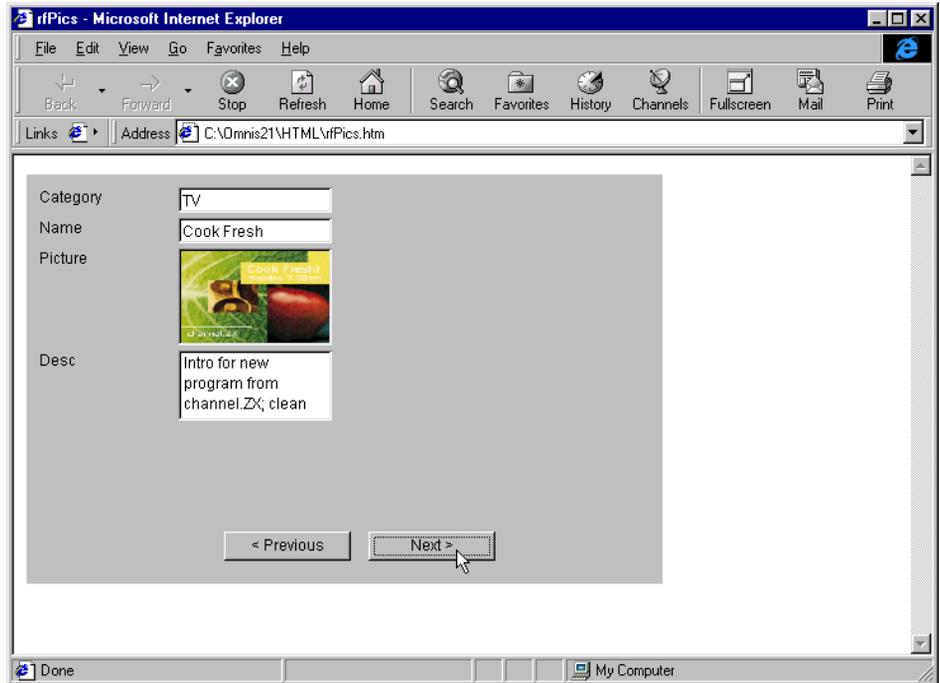
22. To cycle forwards and backwards through the database, click on the Next and Previous buttons. When you reach the end of the database Omnis goes back to the beginning. Try to find the picture(s) you entered earlier. You can switch back to Omnis and modify the form in any way, for example, you can move the fields, change the color and font type of the fields, you can change the background color of the form, and so on. The remainder of this tutorial shows you how to make a few changes to the form.
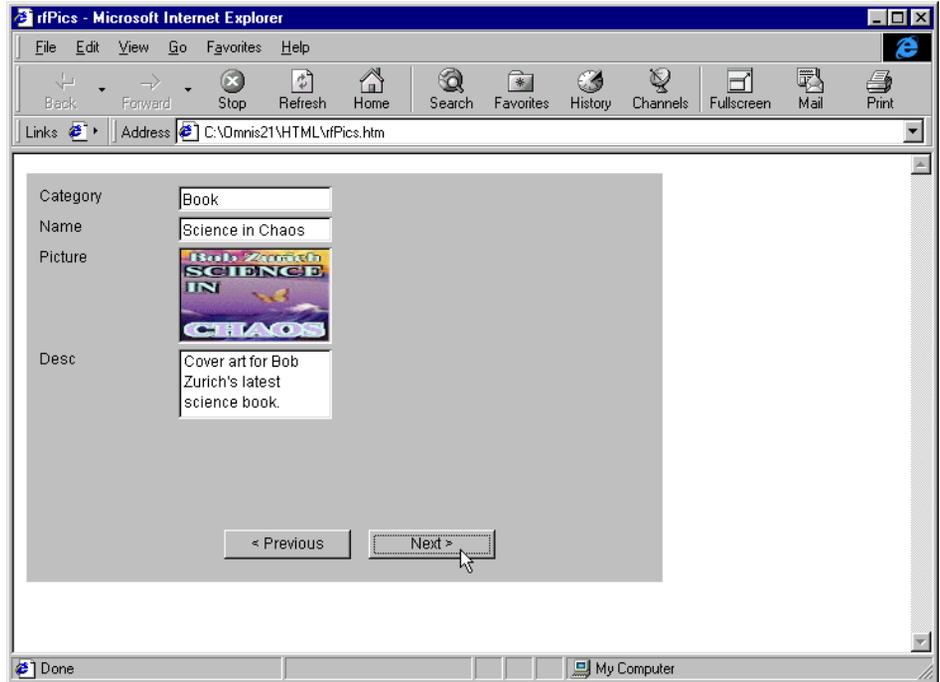
23. To modify your form, you can either switch back to Omnis or close your web browser. In the rfPics remote form, make the Desc field bigger and move it down, nearer to the Previous button. Click on the Picture field and press F6/Cmnd-6 or click on the *Property Manager* to bring it to the top. Under the General tab, click on the **height** *property* and change it to **144**: note all the pictures in the database are scaled to this height or overall size. Similarly, click on the **width** *property* and change it to **186**.
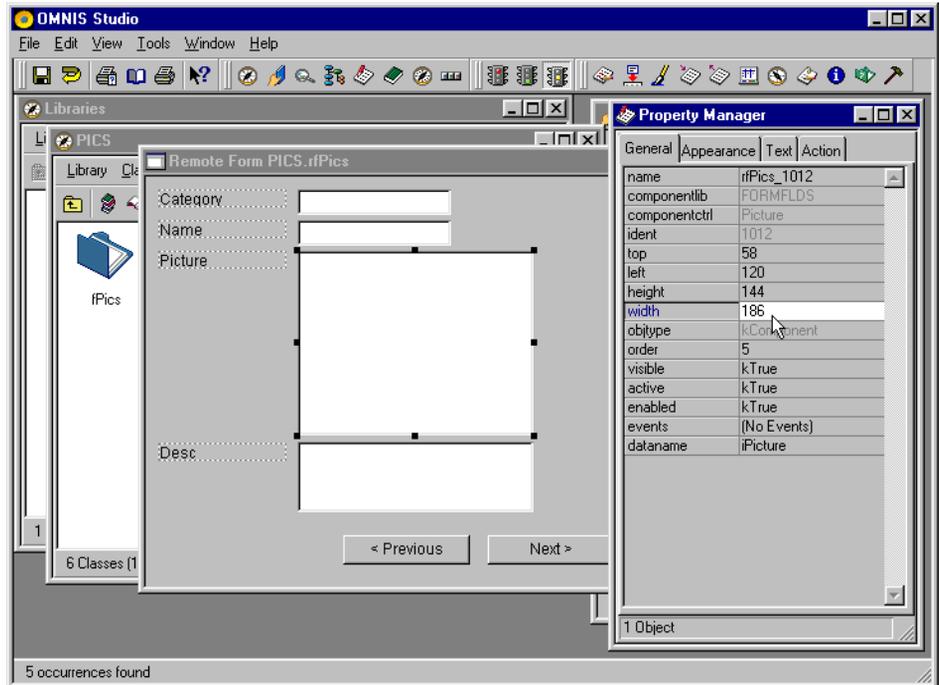
24. Next you can change some other properties of the Picture field, such as turning off the picture scaling. Assuming the Picture field is still selected, click on the Appearance tab in the *Property Manager*, and change the **noscale** property to **kTrue**. You can either double-click on such a property to toggle its value, or select a value from the dropdown list. If you want to store large pictures in your database you may want to add vertical and/or horizontal scrollbars to the picture field. To do this change the **vertscroll** and **horzscroll** properties to kTrue under the Appearance tab in the Property Manager.
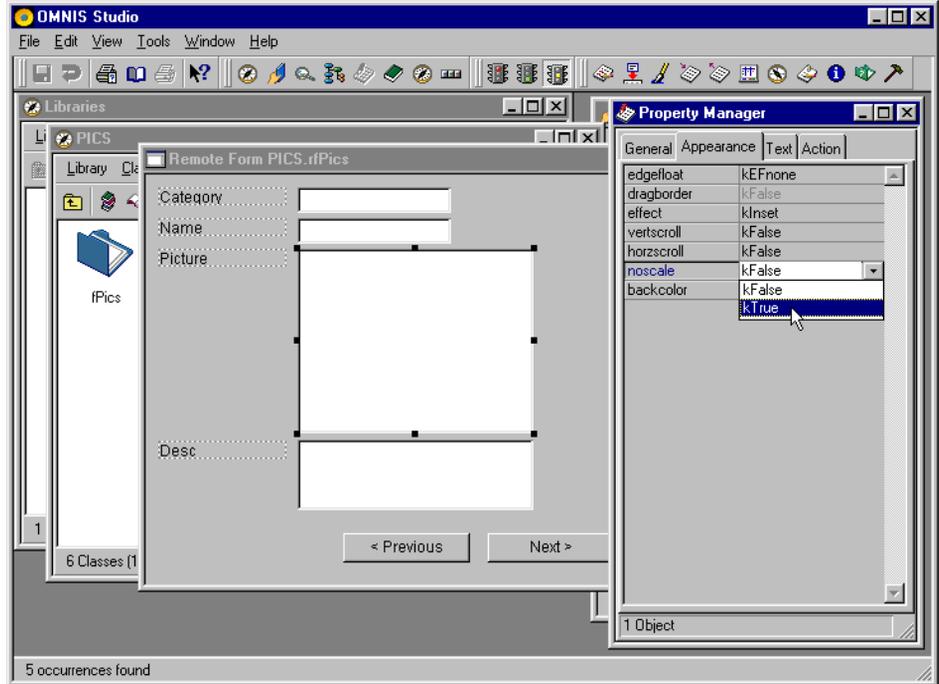
25. Next you're going to change the style and color of all 4 fields in the form. Shift-click on all 4 fields, or click and drag your mouse over the fields, press F6/Cmnd-6 or click on the Property Manager to bring it to the top, click on the Appearance tab, and change the **effect** property to **kFlat**. You may like to try out some of the other border effects or styles available in the dropdown list in the Property Manager.
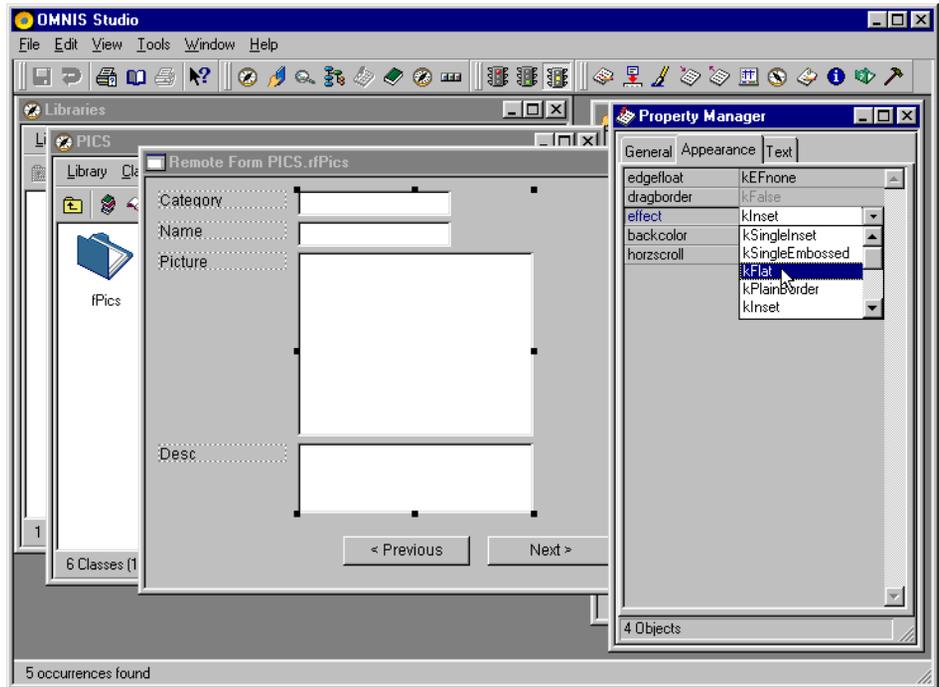
*Tech Tip: You can align or arrange a group of objects using the **Align** submenu on the object context menu. To open this menu you Right-click (Windows) or Ctrl-click (Mac) on an object or group of selected objects and select Align.*

Using the Align submenu you can align objects to top, left, right, and bottom edges. In addition you can make a group of objects the same height or width, center them horizontally or vertically, and you can evenly distribute objects horizontally or vertically. A very useful tool for tidying up your data entry windows...

26. With all 4 fields still selected, change their **backcolor** property to light brown, or any other color you want. To select a color, click on the color swatch in the Property Manager and choose a color from the palette window. Next you can click on the background of the window and set its **forecolor** property to the same color as the fields, or if you prefer a contrasting color, again using the Property Manager.
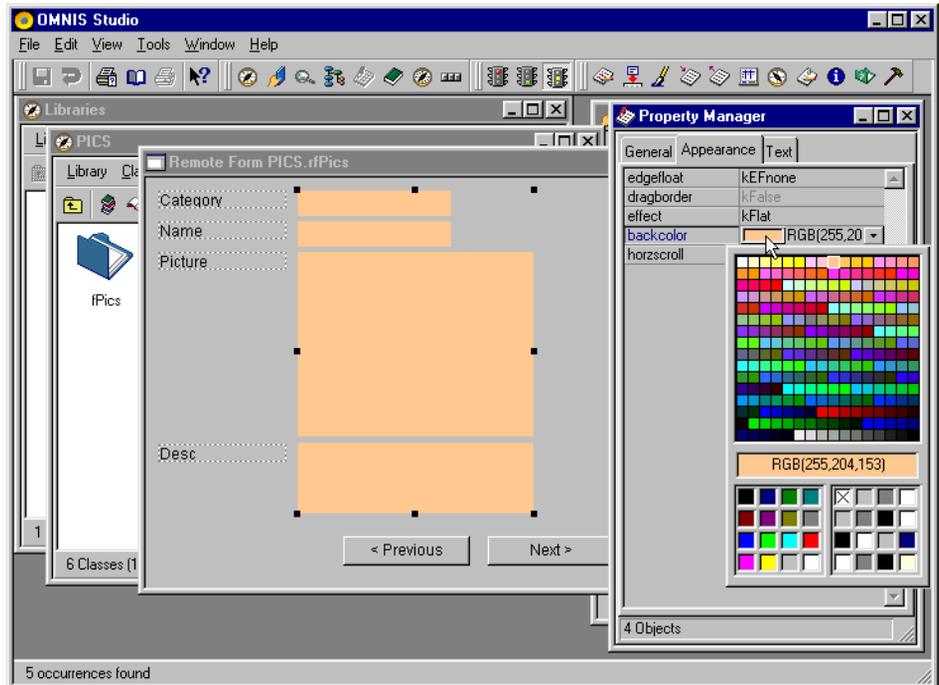
*Note: Remember that, for a window area, **forecolor** effectively means the background color. With fields, **forecolor** is the color of the field entry and **backcolor** is the background color.*

Tech Tip: You can create some very interesting window backgrounds using the Wash and Tile background external components. To use a background component, open your window in design mode, open the Component Store and click on the Background Components button on the toolbar.

*The **Tile** component takes an iconid; you can create small bitmaps (or copy some from the Windows/Mac Desktop), store them in #ICONS and assign an id. The **Wash** tile takes a start and end color. You edit properties for external comps under the Custom tab in the Property Manager.*

27. You can make as many changes as you like to the remote form, but at any stage you can press Ctrl/Cmnd-T to try out the form in your web browser. Using the changes you've made so far, your web form should look something like the one shown below (notice we have also removed the field labels).
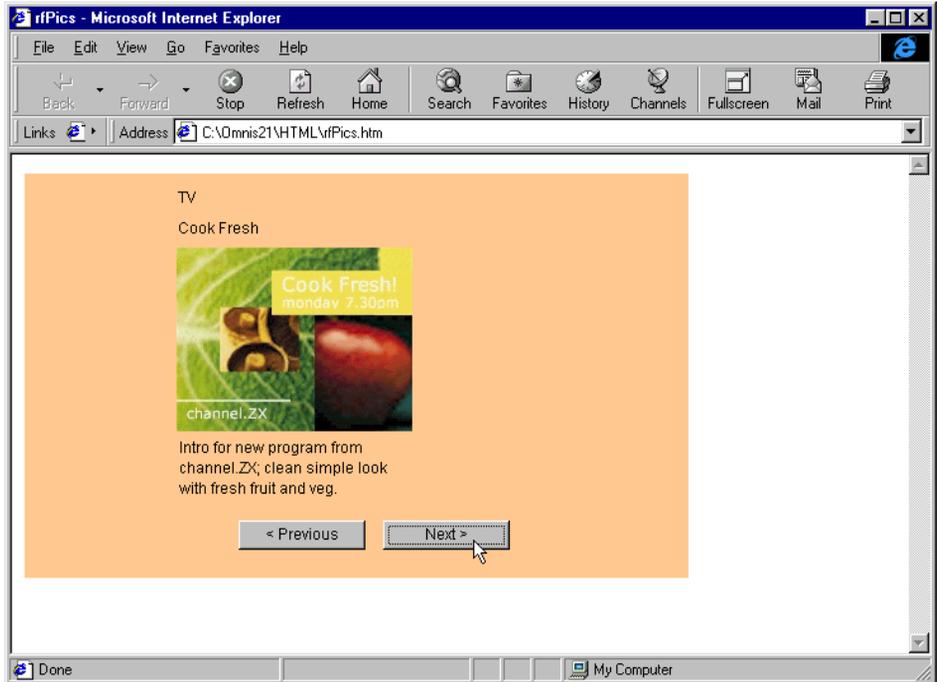
28. Other changes you could make to the form in Omnis design mode include putting the Name and Desc fields next to the Picture field, moving the Previous and Next buttons up so they are under the other fields, and resizing the form to make the whole window more compact. You can make all these changes by changing the appropriate field properties in the *Property Manager*. For example, you could change the **font, fontsize** and **fontstyle** *properties* of the Name field under the Text tab in the Property Manager; note to change the font characteristics of a field you need to set its **fieldstyle** to **none**.
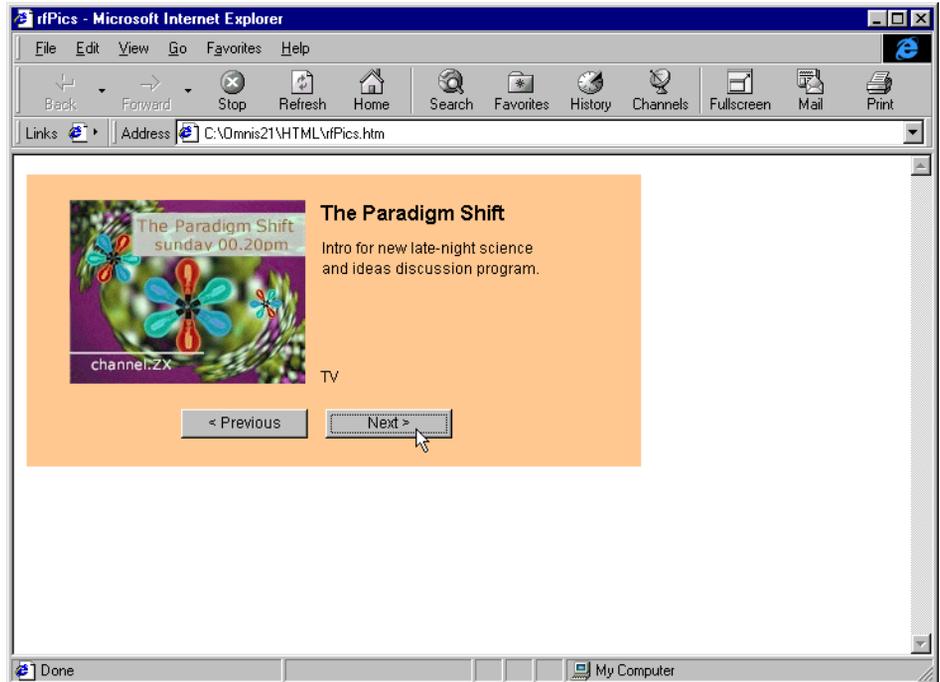
You have completed the tutorial. You may now like to read the following sub-section *Deploying your web application* using the web form and database you have just created, before moving on to the *Let's get deeper* section.

Deploying your web application

(Windows 32 and Linux platforms only)

This tutorial does not cover the process of deploying your application to the web in detail, but this section does describe what you need to do. If you want to deploy your application you may need to ask your IT department and/or webmaster to help you.

To deploy your Omnis application to the web, you need to add one or two files to your web server and set up your Omnis application on a local Win32 or Linux server. First you need to place the Omnis runtime program and your Omnis application on this server. Start Omnis and select the Option or Preferences option from the Tools menu. This option opens the general preferences for Omnis, and you need to set the serverport option to 5912 or some other port number not currently in use. Then open your Omnis application, in this case the Picture database, and keep it running.

On your web server, you need to add the OmnisAPI.DLL file to your cgi-bin folder; the file is in the Webclient/Server folder in the Omnis tree.

When you open your web form in Omnis, an HTML file is created for you, in this case it is called rfPics.htm, and is placed in the Omnis/Html folder. You need to edit this file, using a standard Html editor. The Html file contains the remote form embedded as an object and you need to edit some of its parameters: in the WebServerURL parameter add your website domain name, and in the WebServerScript parameter add "/cgi-bin/omnisapi.dll" i.e. the location of the omnisapi.dll file. Finally, you need to put the edited rfPics.htm file on your webserver.

To browse the database remotely, open your web browser and navigate to the rfPics.htm web page, depending on where you have placed it on your website. The web form should appear allowing you to browse the database.

Tech Tip: Omnis provides the tools for localizing or translating your applications. The library used to translate your libraries is located in the Local folder under the main Omnis folder. To translate a library you: Export the text and literal strings in your library that require translation, translate the text using a standard text editor or word processor, and then import the text back into your library.

When you open the Trans.lbs library (in the Omnis/Local folder) the translation browser appears. It displays any libraries you currently have open, with the exception of the translation library itself. If the library you wish to translate is not open, use the Library>>Open option on the translation browser menu to open it.

Let's get deeper

Now that you've worked through the tutorial, you've probably grasped some of the potential of Omnis Studio and appreciate how easy it is to build applications with it. On the way, you may have glimpsed avenues you could take to extend the application you created. Although it's impractical to document every possibility here, we will look at a few key areas where you can expand it. We won't go too far in any particular direction, just far enough to show where Omnis development can lead you next. Where you do go next is, of course, up to you and the kind of applications you want to build.

If the PICS.LBS library you created in the tutorial is not currently displayed, open it in the Library Browser by clicking the Open Library button and locating the PICS.LBS file in the Open Library window – it should be in the tutorial folder (in your Omnis program's welcome folder).

Adding a report

A report is a very common and often essential feature of a database application, so Omnis Studio makes adding one particularly easy.

1. Make sure the classes in the PICS.LBS library are displayed in the browser. Then, in the Component Store, display the Report Classes (by clicking the Report Classes button in the Component Store's toolbar) and drag the Omnis Report Wizard into the PICS browser.
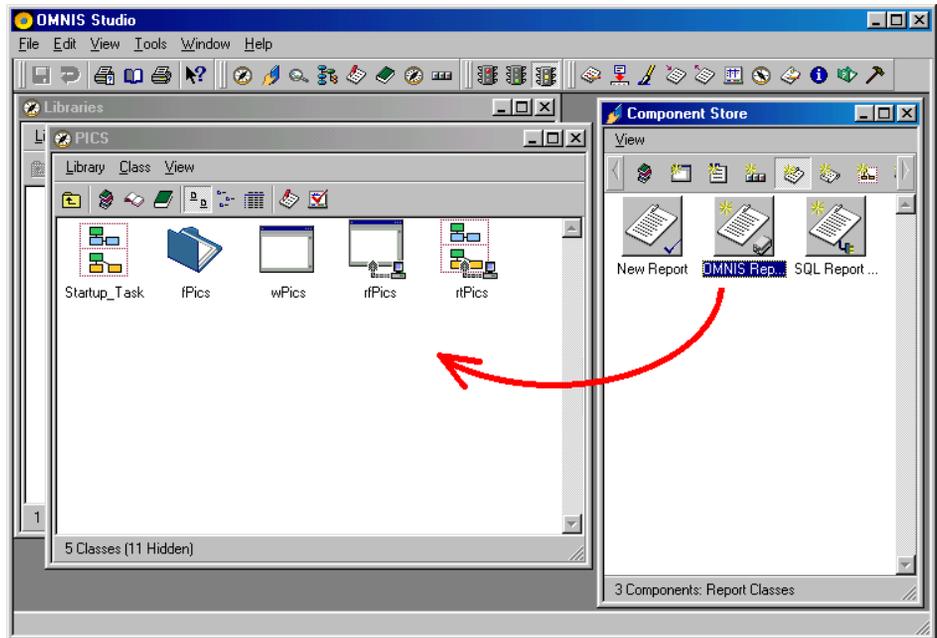
Tech Tip: You can print details of a class, a number of selected classes, or all the classes in the current library.

*To print the details about a class or classes you need to select them in the Omnis Browser and choose the **Print Class Details** option from the **File** menu; to print details about all classes deselect them all in the Browser and use the Print option. The report, which is sent to the current print destination such as the screen, contains a list of variables, methods, and properties of each class, and it describes any errors in syntax if there are any. For window and report classes the report also contains a list of objects and fields in the class; you can also open a window or report class in design mode, select an object and use the Print option to get details for the object, including its methods.*

---

2. Name the new report class "rPics" and click it to run the Omnis Report Wizard. At the wizard's "Select Report Fields" screen, click the + symbol to expand the tree list. You can now click to check the fields you want in the report. For this exercise, select all the fields. You can do this in one action by clicking to check fPics - this also checks all the fields below it automatically. When you've done that, click the Create button to create the report.
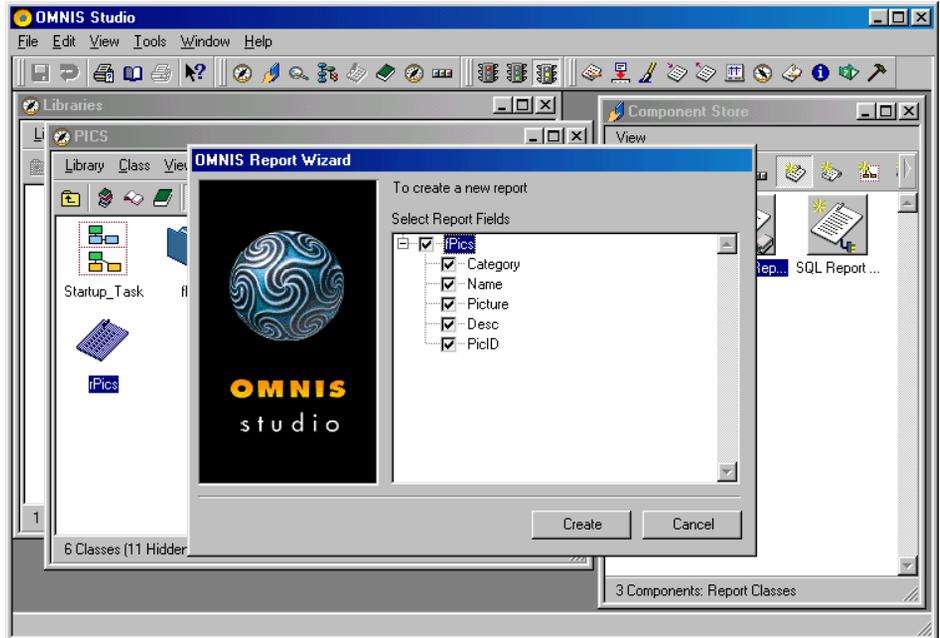
3. The wizard has now created your report and displays its layout.
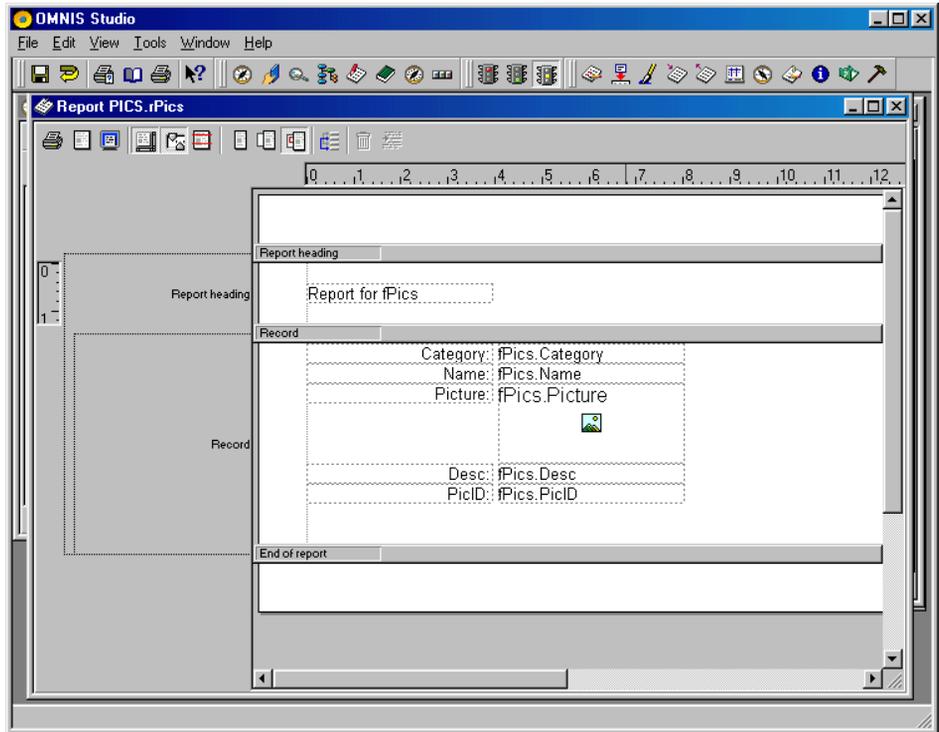
You can alter the elements of the layout using click-and-drag. For now, though, run the report to test it. First of all, you need to make sure the data file is open and current; if you have not already done so in your current Omnis Studio session, do it as follows:

*Select the **Data file Browser** option from the **View** menu on the main Omnis menubar. Select the **Open** option from the **Datafile** menu in the Data File Browser. Open the welcome folder (in your Omnis program's tutorial folder) and double-click on the Omnis datafile called **PICS.DF1**. Make sure the PICS datafile is selected in the Data File Browser, then select the **Make Current** option from the **Datafile** menu; an arrow appears next to the PICS datafile icon to show it's the current one. Finally, close the Data File Browser.*

Now click the Report window's Print To Screen button to run the report and display the results on the screen.

4. Your report is now displayed. You can scroll through it and print it as required.
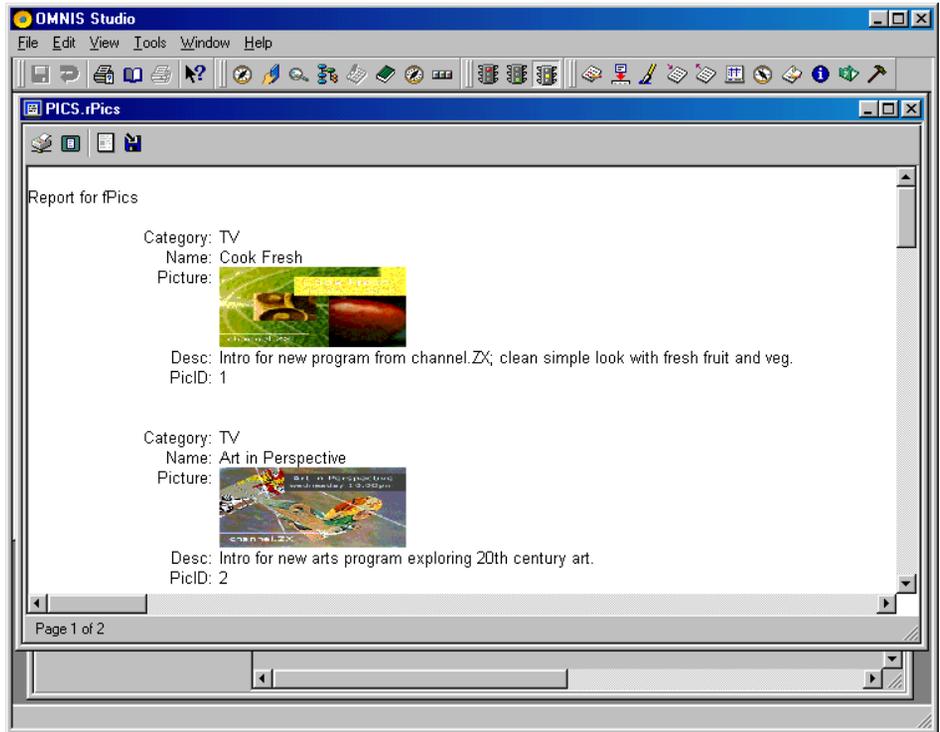
Learn more about reports

In the manual *Using Omnis Studio*, take a look at *Chapter 9 - Report Classes* (available in PDF format on the Omnis Studio CD).

Tech Tip: You can save reports to disk and send them to people via e-mail or other means. As long as the people at the other end have an Omnis Runtime or Development version, they can then print the report to any destination.

To save a report you can:

- *pick the disk destination and print the report,*

or,

- *print the report to screen or page preview and click the Save button on the toolbar.*

You can print a report from disk by:

- *double clicking the report file. On Windows and Unix the file name must have the .REP extension. On the Macintosh it must have the creator set to OO$$ and the file type to OO$Q,*

or,

- *starting your Omnis and select 'Print report from disk' from the File menu. The file must have the .REP extension in the file name for all platforms, including Macintosh, if the file type and creator have been lost during transit.*

Adding a menu

Menus enable the users of your application to perform standard operations, as well as run and navigate between the features you've built into it. For example, the database browsing window you created in the tutorial works well enough, but your users would certainly appreciate a menu in it that gave them an option to print the report.

1. Make sure the classes in the PICS.LBS library are displayed in the browser. Then, in the Component Store, display the Menu Classes (by clicking the Menu Classes button in the Component Store's toolbar) and drag the Menu Wizard into the PICS browser.
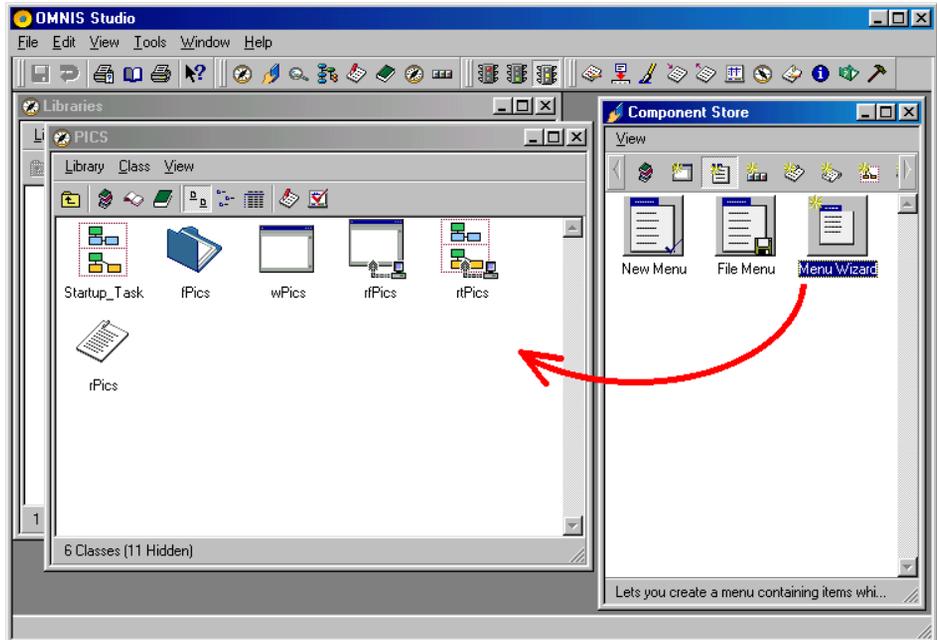
2. Name the new menu class "mPics" and click it to run the Menu Wizard. We want the menu to have an option to print the report, so, at the wizard's screen, click the + symbol to expand the Reports tree list, then drag the rPics report class over to the menu pane on the right. When you've done that, click the Create button to create the menu.
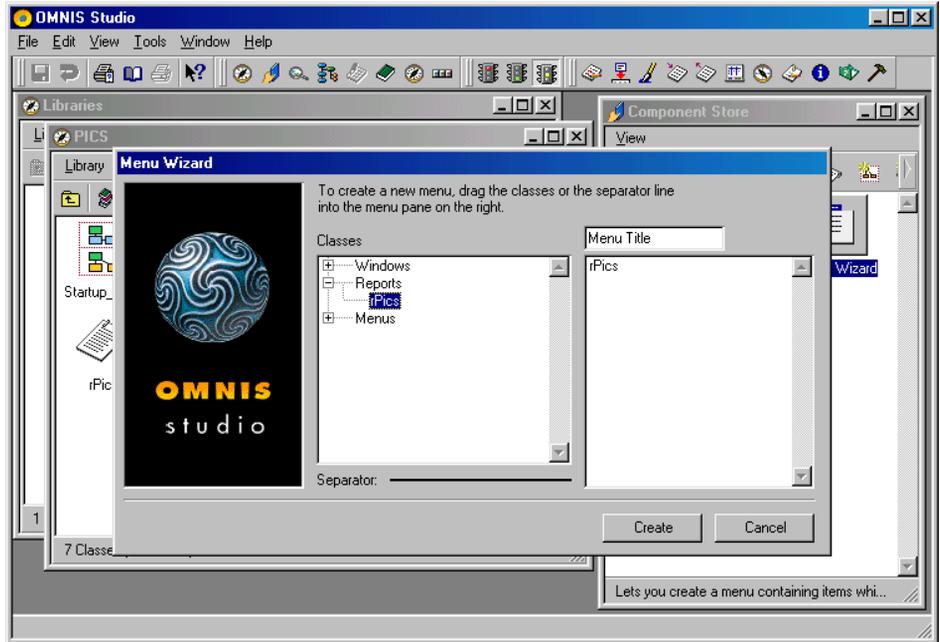
3. The details of your menu are now displayed. The menu will work perfectly well, although you can make it more user-friendly by altering the default menu title and option name.
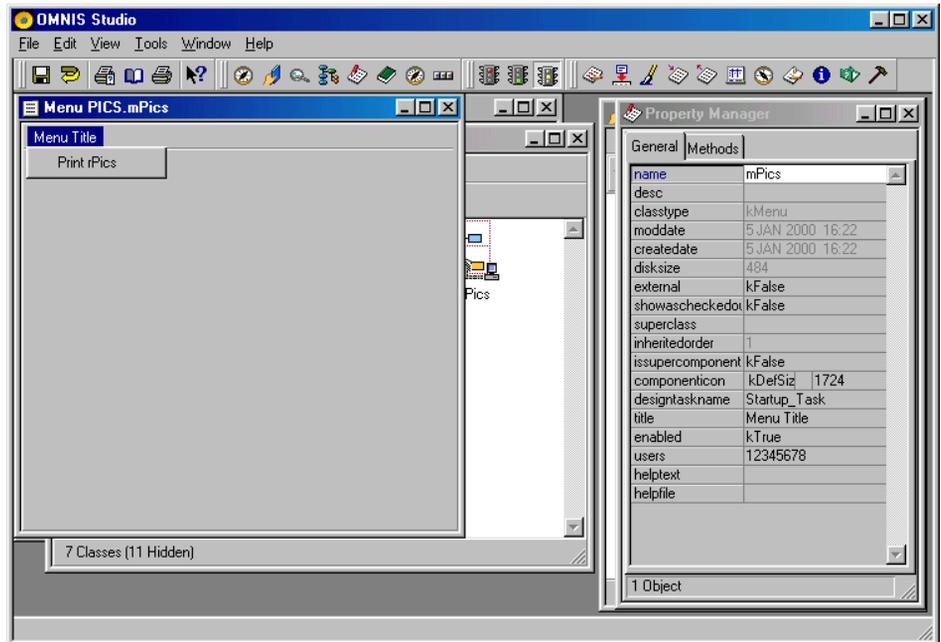
4. In the Menu PICS.mPics window, click to select the menu title (shown as "Menu Title" by default) and change it to Report. You can either type "Report" in the "title" box in the Property Manager, or click inside the menu title in the window editor itself and edit it.
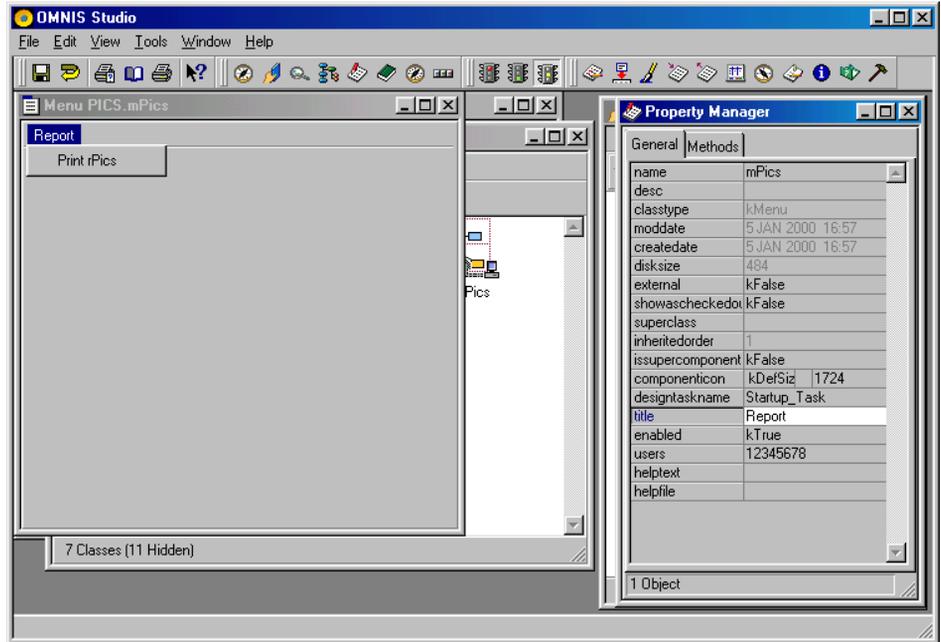
5. In the Menu PICS.mPics window, click to select the option name (shown as "Print rPics" by default) and change it to "Print the report". You can either type "Print the report" in the "text" box in the Property Manager, or click inside the option name in the window editor itself and edit it.
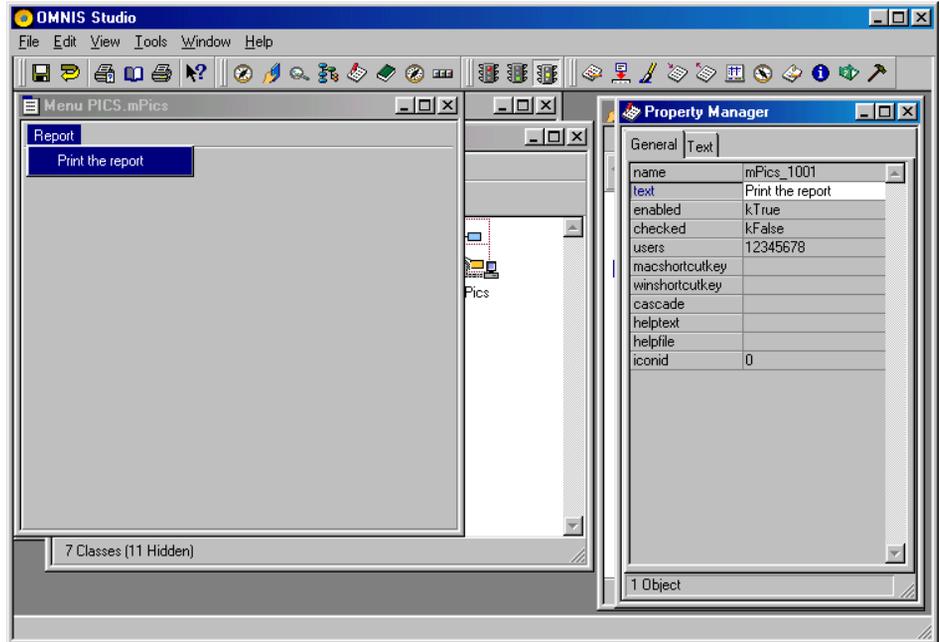
6. Another user-friendly feature of a menu option is an icon indicating what it does. In the Menu PICS.mPics window, click to select the "Print the report" option. In the Property Manager, click the drop-down arrow on the "iconid" box to open the "Select an icon" window. Use the drop-down list there to display the Reports icons, select the icon id number 1961 (a printer icon) and click the Select button.
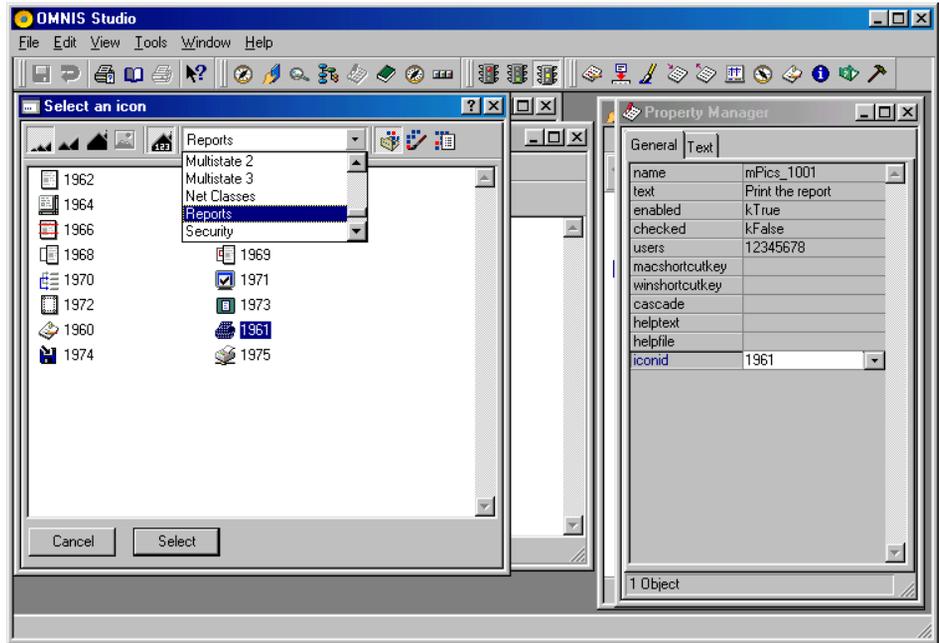
7. Now you need to alter the properties of the database browsing window so that it includes your new menu. Close the menu class editor and double-click on the wPics window class to open it. In the Property Manager, change the "hasmenus" setting to "kTrue" (by selecting it from the drop-down list) and in "menunames" select your new menu mPics. The menu "Report" then appears at the top of your window.
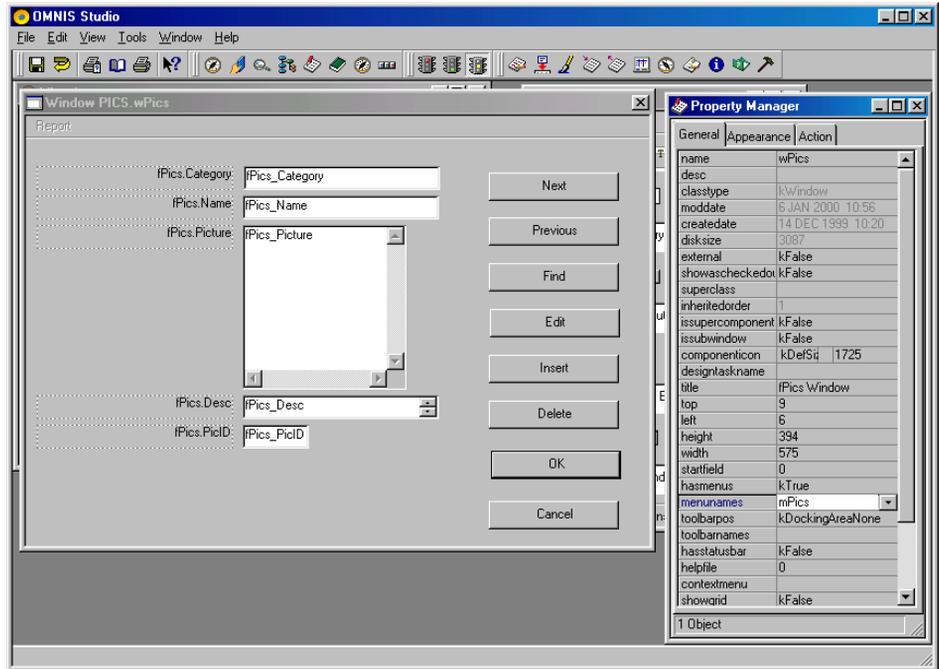
8. Now open the window in runtime mode to test the menu. Right-click (Ctrl-click) in the wPics editor area and select the Open Window pop-up menu option. Your window opens and you will see your new Report menu at the top. Click the menu and select the "Print the report" option from it.
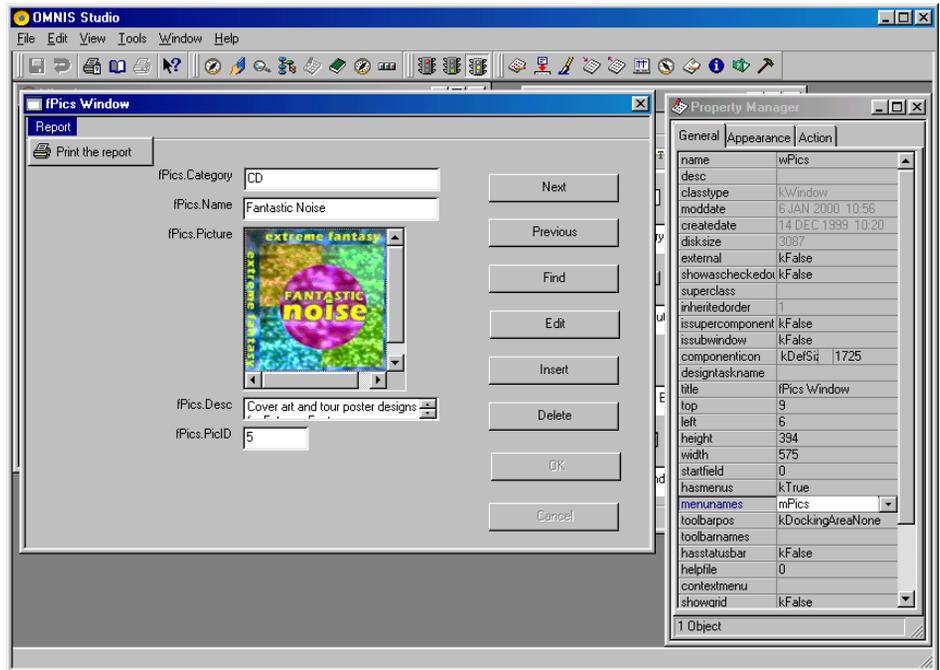
9. The Report Destination window opens from which you need to select the Screen option. When you click OK, the report runs and is displayed on screen.
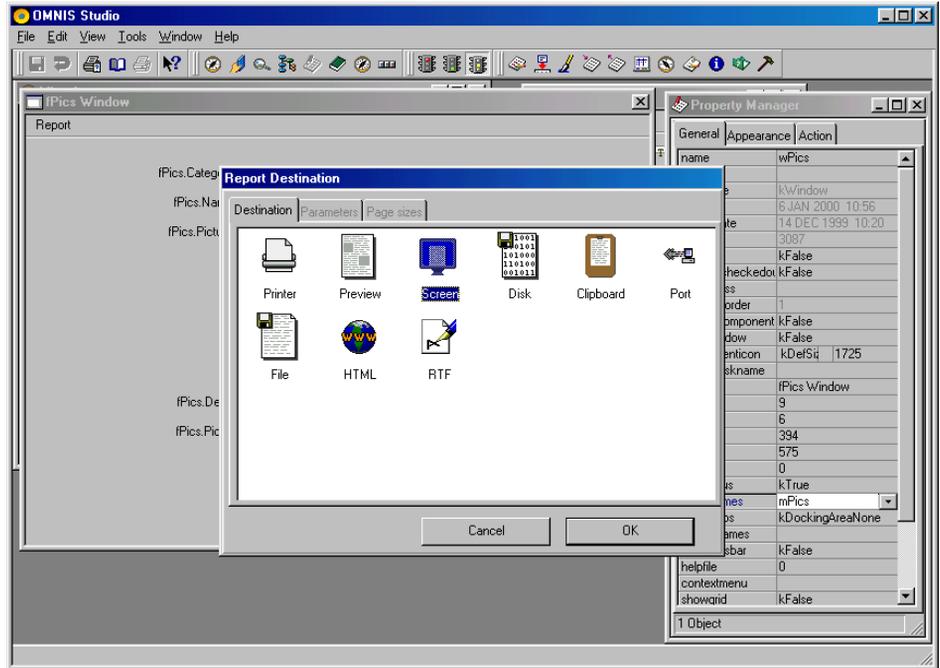
10. You can now scroll through the report and print it if required.
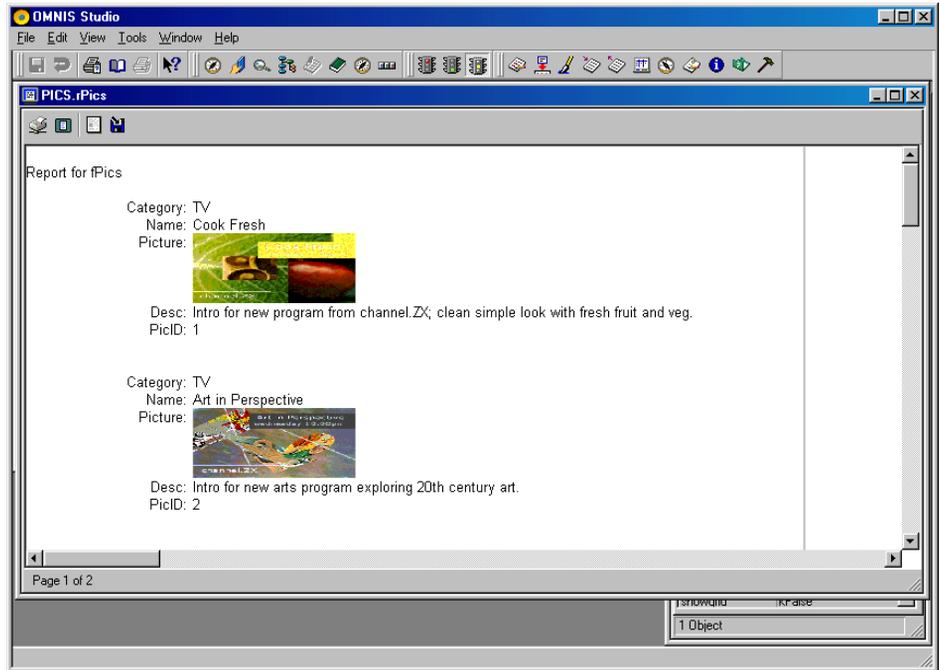
Learn more about menus

In the manual *Using Omnis Studio*, take a look at *Chapter 7 - Menu Classes* (available in PDF format on the Omnis Studio CD).

Adding a toolbar

Like menus, toolbars also provide your application's end users with access to standard operations and application features. It's common among application user interfaces to have a toolbar of buttons that duplicate the menu options. Users new to your application may initially find it easier to use the menu options, with their logical arrangement and text descriptions. Then, as they grow more familiar with the interface, they may start using the toolbar buttons for speed. With other users it can be a matter of personal preference.

Follow this exercise to add a toolbar containing a button to print the report; this will complement the menu option you created previously.

1. Make sure the classes in the PICS.LBS library are displayed in the browser. Then, in the Component Store, display the Toolbar Classes (by clicking the Toolbar Classes button in the Component Store's toolbar) and drag the New Toolbar template into the PICS browser.

2. Name the new toolbar class "tbPics" and double-click it to open it. At the moment, the toolbar has no buttons. If you display the Component Store, however, it automatically displays the Toolbar tools. You need a push button, so drag the Push Button tool to the toolbar editor (the cursor will change to indicate where you can drop it).
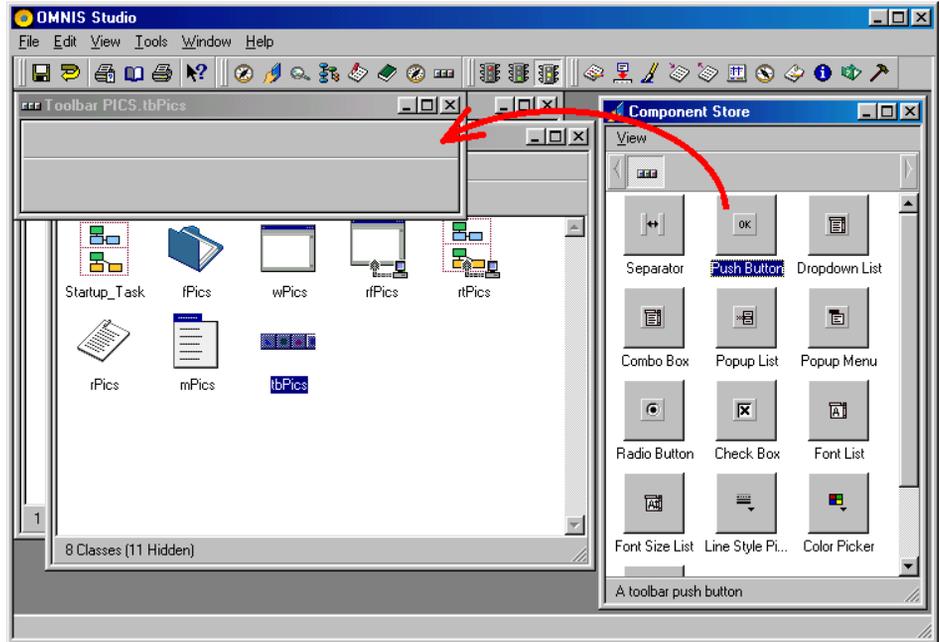
3. The button now appears in your toolbar editor. To make the button useful, you need to change its appearance (from the default "OK"), give it a helpful ToolTip and specify what is to happen when a user clicks it. First of all, change its appearance.

In the toolbar's Property Manager (which should now be displayed by default), click to display the Appearance tab, then click the "iconid" number's drop-down arrow to open the "Select an icon" window. It's good practice to use the same icon as you did for the corresponding menu option. So, use the drop-down list here to display the Reports icons (you may also need to click the toolbar button to display the 16x16 Icons, as below), select the icon id number 1961 (a printer icon) and click the Select button. The icon now appears on your button.

Click here to display 16x16 icons
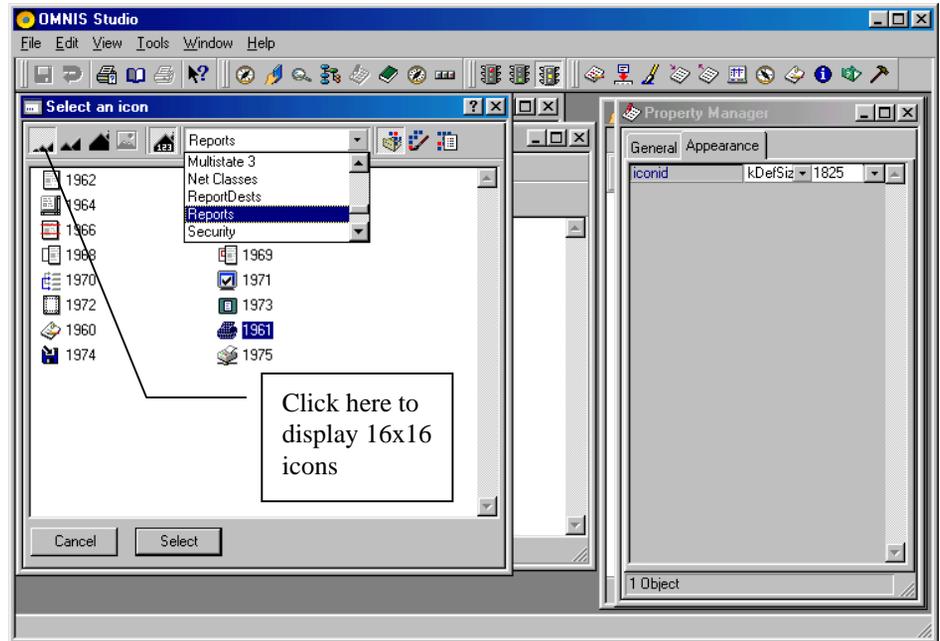
4. Another useful feature of a toolbar button is a ToolTip to tell the user what the button does. Click to display the Property Manager's General tab and type "Print the report" in the "tooltip" box.
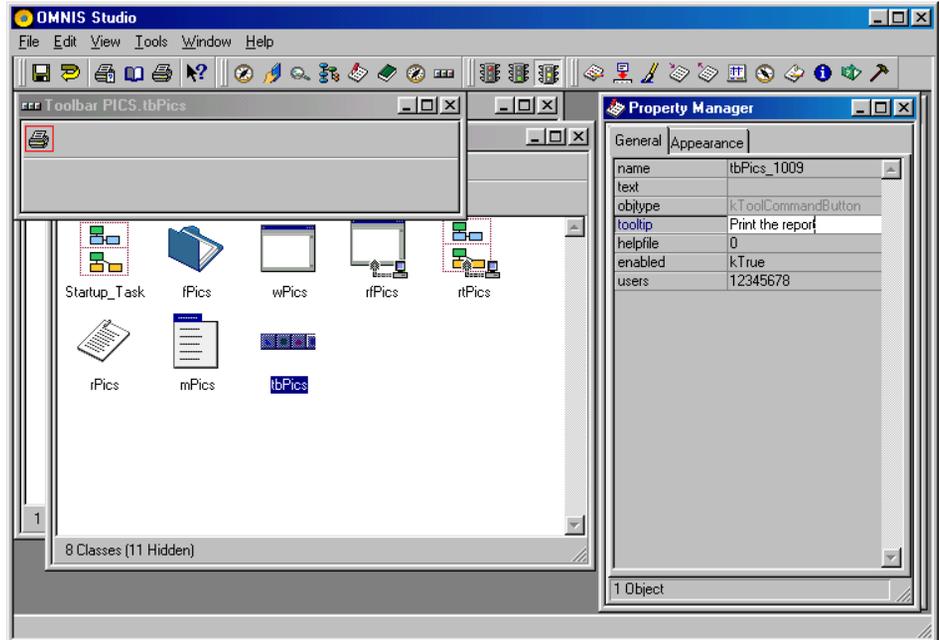
5. Now you need to specify what happens when the user clicks the button. In the toolbar editor, double-click the new button to open its Methods editor. As you can see, an "On evClick" line of code has already been entered by default. You now need to insert the other lines of code that will run and print the report. These are:

```
Set report name rPics
Prompt for destination
If flag true
    Print report
End if
```

To add a new line of code, click to highlight the line into which you want to insert the code, then type it in. Don't worry about upper/lower case or indentation; Omnis takes care of these automatically. As you type, Omnis tries to anticipate the command you're entering and inserts it for you. Keep typing until the correct command appears. When you type in the "Set report name rPics" line, you can select "rPics" from a list that appears as soon as Omnis has established you're entering the "Set report name" command. When you've entered these code lines, close the Methods editor.
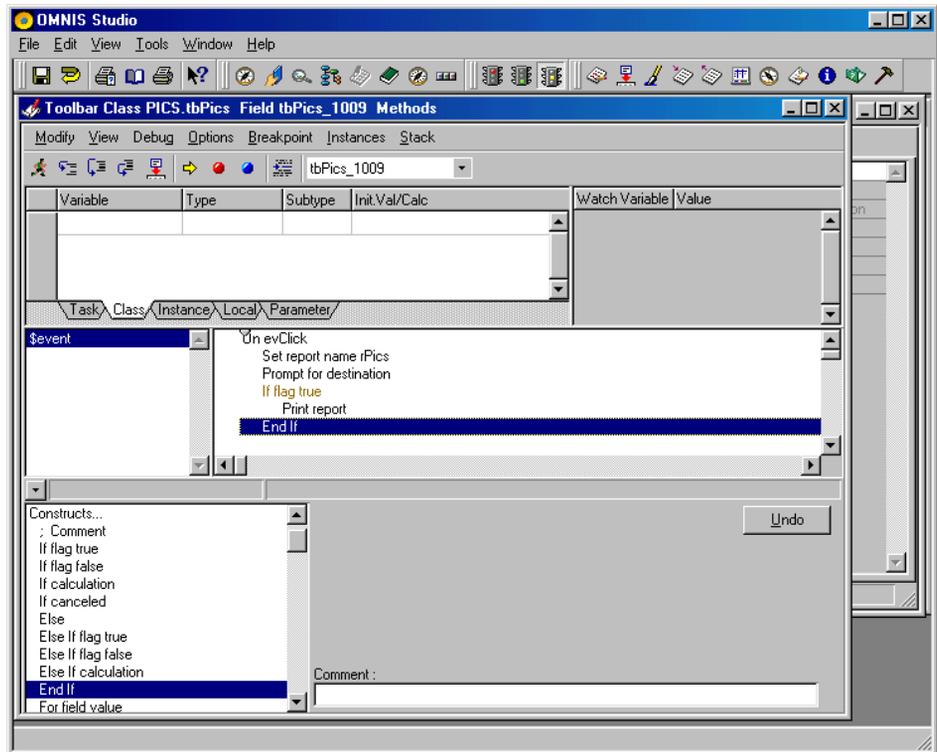
Tech Tip: In the Methods editor, you can also move to the next line using Ctrl-N/Cmnd-N.

6. Now you need to alter the properties of the database browsing window so that it includes your new toolbar. Close the toolbar class editor and double-click on the wPics window class to open it. In the Property manager, change the "toolbarpos" setting to "kDockingAreaTop" (by selecting it from the drop-down list) and in "toolbarnames" select your new toolbar "tbPics". The toolbar then appears at the top of your window. As in the picture below, you may also need to alter the window size to take account of the extra space taken up by the toolbar and menu.
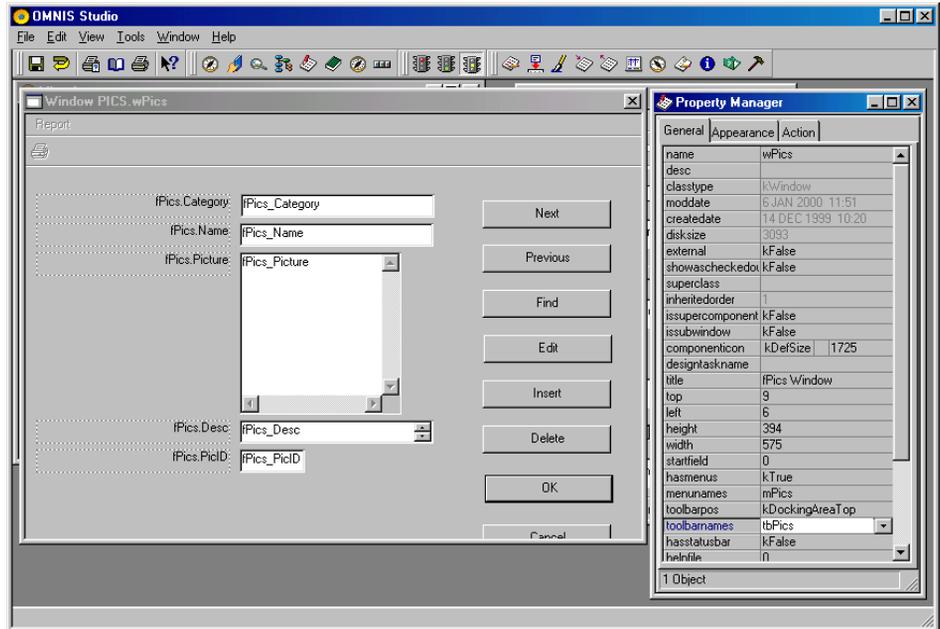
Tech Tip: You can set the global ($root) preferences or options in Omnis using the Property Manager, via the Tools menu Options (Win) or Preferences (Mac) menu item. You can set the General, Appearance, Page Setup, and Device preferences as well as view the $root methods.

7. Now open the window in runtime mode to test the toolbar button. Right-click (Ctrl-click) in the wPics editor area and select the Open Window pop-up menu option. Your window opens and you will see your new toolbar just beneath the menu. Click the "Print the report" button you added (checking that the ToolTip appears when you place the cursor over it). The Report Destination window should open as before from which you need to select the Screen option. When you click OK, the report runs and is displayed on screen.
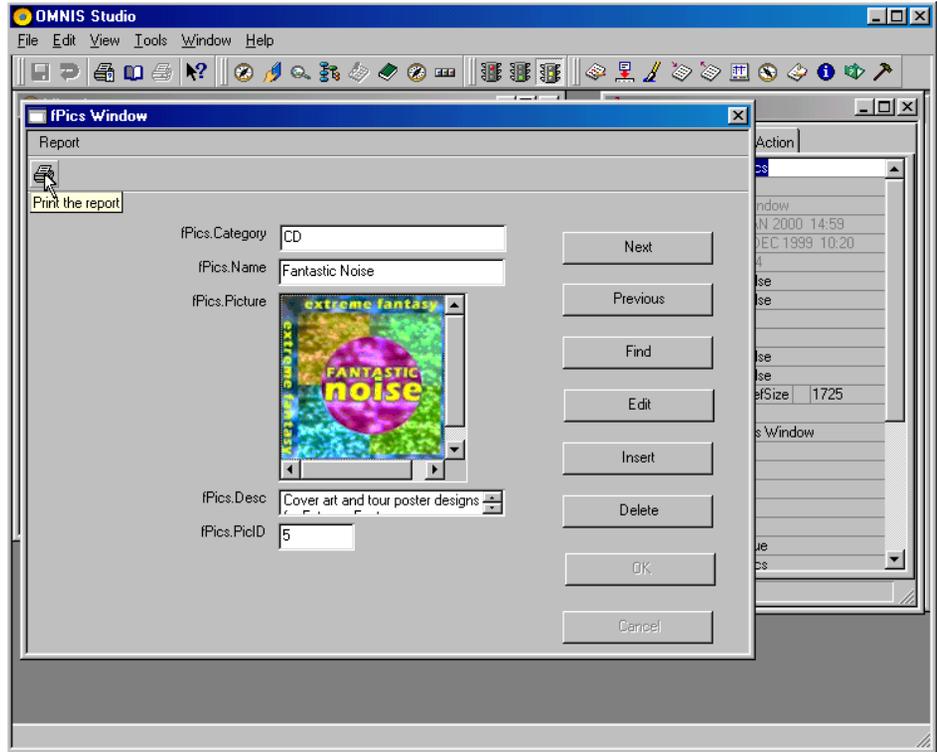
Tech Tip: You can allow an application's end user to select line styles, colors, and patterns from a toolbar using the picker controls:
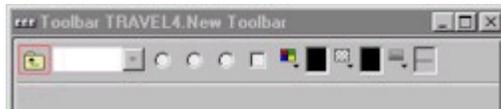
*The Line, Color, and Pattern pickers have all the general properties of a toolbar control as well as the **text** and **iconid** properties. The standard line, color, and pattern palettes appear in these controls by default. At runtime, the value selected by the user is returned as the **contents** of the control.*

Learn more about toolbars

In the manual *Using Omnis Studio*, take a look at *Chapter 8 - Toolbar Classes* (available in PDF format on the Omnis Studio CD).

Modifying the Startup Task

When you create a library, Omnis Studio automatically creates a Startup Task in it for you. This runs the initialization code whenever you open the library, thus providing the functionality you need to develop your application. If, therefore, you want something to happen when you open your library, you need to enter the appropriate code in the Startup Task.

In this exercise, you will modify your Startup task to open the database browser window automatically when you open the library.

1. Make sure the classes in the PICS.LBS library are displayed in the browser, then double-click the Startup_Task class to open its Methods editor. Using the code-entering technique you used in step 5 in the previous section *Adding a toolbar*, add the following line of code (making sure that **$construct** is selected):

   Open window instance wPics

   You will notice that, as you type the line in, you can select "wPics" from a list that appears as soon as Omnis has established you're entering the "Open window instance" command. When you've entered this code line, close the Methods editor.
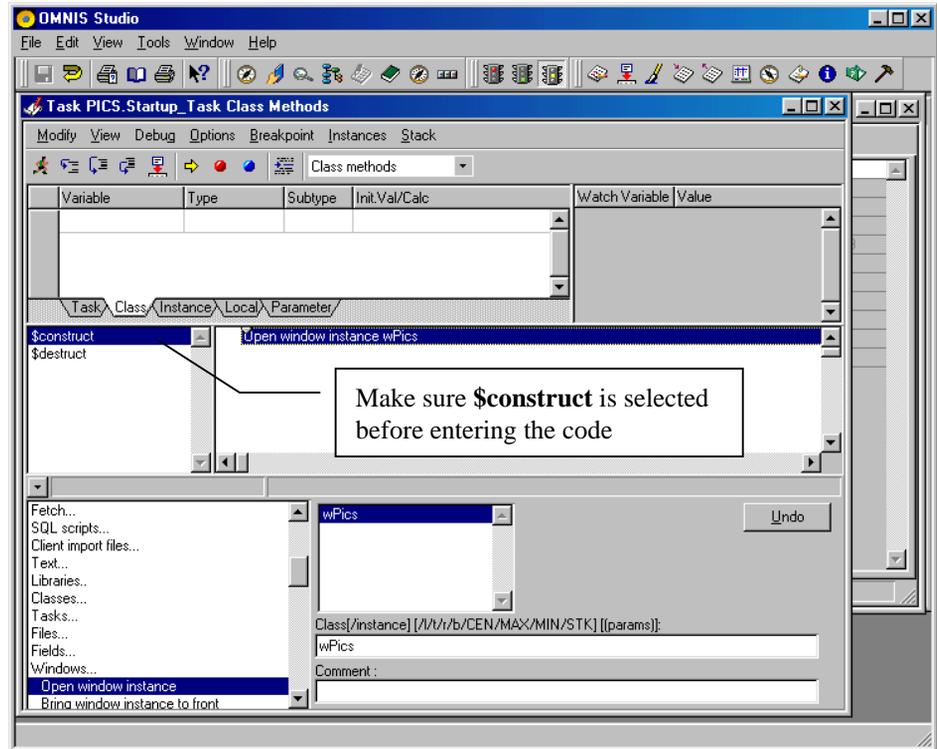
*Note: Entering this code with **$construct** selected meanse the code will run when the library (and, therefore, the application) is opened. If you wanted something to happen when the library/application closes, you would first select **$destruct** before entering the code.*

2. In the Library Browser, close the PICS library (click the Close Library button). Now open the library again (click the Open Library button and locate the PICS.LBS file in the Open Library window – it should be in your Omnis program's welcome/tutorial folder). When the library opens, the database browser window you created opens in runtime mode automatically.
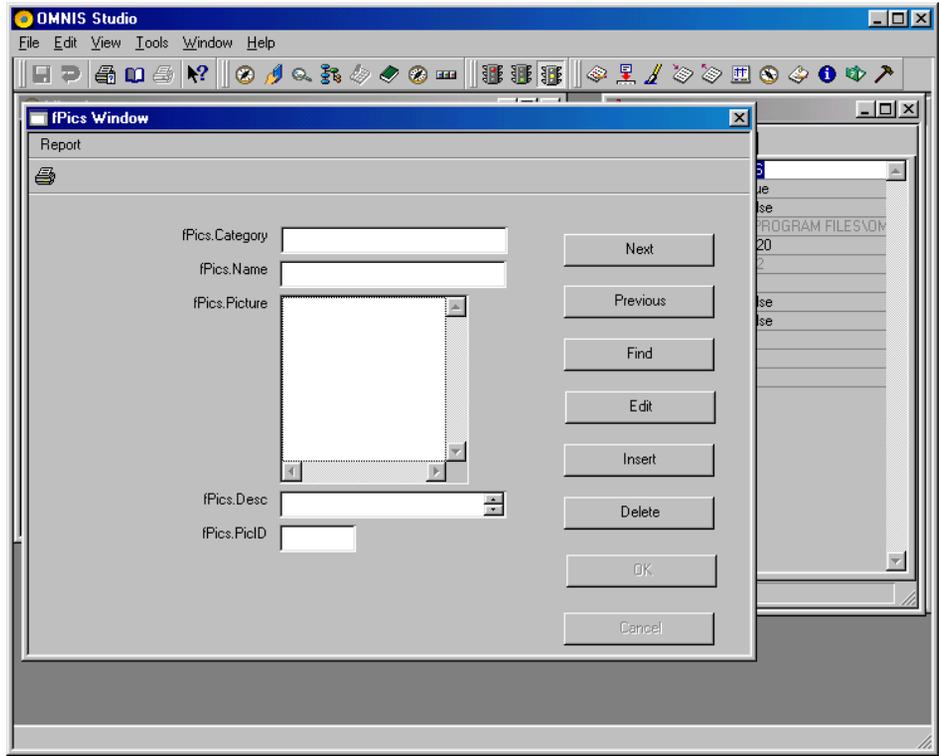
Tech Tip: You can check the methods in your library using the method checker. The method checker is available under the Tools menu on the main Omnis menu bar. It checks your code for syntax errors, unused variables, methods without code, and so on. It provides various levels of checking and reports errors in individual classes or all classes in the current library. It is particularly useful for checking libraries converted from an earlier version of Omnis.

Note that the method checker does not correct the code in your libraries! It simply reports any errors and potential problems in your code which you can then jump to and modify yourself.

Where to next?

Now that you've worked through this book, developed and expanded a simple database application, you should have a fair grasp of the capability and potential of Omnis Studio.

The next obvious step is to check out other Omnis documentation (the Omnis Studio manuals are available in PDF format on the Omnis Studio CD).

However, before you attempt to read, say, *Using Omnis Studio* cover-to-cover, you might find greater benefit exploring the following specific areas. You could even put these concepts immediately into practice and continue to build on the PICS database application you created in the tutorial and the *Let's get deeper* section.

Databases

It's important you familiarize yourself with how Omnis integrates with databases. Where you used DML to work with an Omnis database in the tutorial, you could set yourself an exercise to link your application to an existing non-Omnis database (such as Oracle, Sybase or Informix) using the SQL Browser. Refer to the manual *Using Omnis Studio*, *Chapter 5 - Data Classes* and *Chapter 12 - Accessing Your Database* for information on this.

Web forms (Remote forms)

In the tutorial, you saw how easy it was to create a form to browse the sample data on the web. If this is an area in which you're likely to be involved, it would be a good idea to follow through the procedure outlined in *The tutorial*'s subsection *Deploying your web application*. In addition, there are some downloadable Omnis web component examples on our website at www.omnis.net/webgallery.

Success stories

You can get some good development ideas from real applications built by other Omnis Studio users. Check out the success stories at our website at www.omnis.net/partners/success/success.html.

Glossary

Application a software program that provides a solution to a specific business problem.

Browser (Omnis library or class) an Omnis window that displays libraries or the contents of the current library.

Browser, Web tool for browsing web pages, such as Internet Explorer or Netscape Navigator.

Character (data type) stores character or number data, up to 10 million characters

Class a library object that contains the definition for a window, menu, form, etc.

Column (database) field definition in a database; a column has a name and data type.

Component Store an Omnis window that contains classes, templates, and wizards.

Control an object, such as a button or pick list, that lets you interact with your database.

Database a collection of data organized for easy access

Data file (database) a disk file that stores an Omnis database.

Data type defines the type of data that can be stored in a column, can be character, number, date, etc.

Edit menu used for copying and pasting objects and data.

File (class) defines the columns or fields in an Omnis database.

File menu lets you save your library, print classes and customer reports.

Form (or window) area of the screen containing fields and controls for browsing and inserting data into a database.

IS (Information Services) a department that provides computing, networking, and Internet services.

LAN (Local Area Network) a network of computers, servers, and printers within a company.

Library a disk file that stores the objects in an application such as windows, menus, and reports.

Method a piece of code contained in a class or object that performs some action when the object receives the appropriate message.

Number (data type) stores number data with many different decimal and floating point subtypes.

Object any item in a class or library, such as a window or field.

Picture (data type) stores picture data in various formats such as bitmap, jpeg, and png.

Property a characteristic of an object; the properties of an object can define its appearance or behavior for example.

Property Manager an Omnis tool that lets you view and change an object's properties.

Record a single row of data.

Remote form (or web form) an Omnis window for displaying and browsing your database or application on the web.

Remote task (class) an Omnis class that handles the connection between a client's web browser and your Omnis application.

Schema class represents a table or view on your SQL database server. It contains the name of the table or view on your server, and a list of column names and Omnis data types that map directly to the columns in your server table or view.

Sequence (data type) provides a unique number or reference to each record in your database.

Startup task (class) contains startup code that initializes a library when it is opened.

Subtype (data type) specifies the length, date, or number type of a column or field in your database.

Unique key (field) a column or field that provides a unique reference for each record in your database.

Variable the principal data container in Omnis. The scope and the kind of data it can contain depend on the variable type:

Parameter variables receive a value in a method.

Local variables are local to the method.

Instance variables are visible to the instance only, that is, all methods in the instance.

Class variables are visible within the class and all its instances.

Task variables are visible within the task, all its design classes and instances of those classes.

Hash Variables are built-in global variables. They start with the symbol "#":

> **#1-#60** are numeric variables
>
> **#L1-#L8** are list variables
>
> **#S1-#S5** are string or character variables
>
> **System** are miscellaneous values that Omnis uses

Hash variables are global, unlike any other variables, so all libraries have access to them.

Window (class) same as Form.

Wizard (template) automated set of tasks to help you build application components.

Appendix 1 - Omnis naming conventions

Although it's not essential to use a comprehensive naming system in your Omnis application development, we strongly advise that you do. This appendix outlines our recommended naming conventions, which have evolved over time through real Omnis application development experience. We find these conventions work well in all areas of Omnis development.

Library Names

A Library should have a descriptive name followed by the extension ".lbs".

Class Names

Class names should begin with one or two lower case letters followed by the descriptive name in joined capitalized words. The list below specifies the leading character(s) for each class followed by an example name. Do not use underscores '_' in the name. They are not necessary and just make it more difficult to type the names.

Class Type	Leading characters	Example
Code	c	cUtilityMethods
Menu	m	mDatabaseTools
Object	o	oDataAccess
Query	q	qInvoices
Remote Form	rf	rfMain
Remote Task	rt	rtMain
Report	r	rSalesSummary
Schema	sc	scInvoices
Search	s	sInvoices
Table	ta	taInvoices

Class Type	Leading characters	Example
Task	t	tSales
Toolbar	tb	tbDefaultNavigate
Window	w	wEditOptions

The part following the leading characters must also be consistent. We suggest the following simple rules:

- classes which edit data should start with "Edit". In case of a window class "wEdit".

- classes for navigation of data should start with "Nav". In case of a toolbar class "tbNav".

- classes whose only purpose is to be derived from should start with "Super". In case of a object class "oSuperDataAccess".

Variable Names

Variable and parameter names, with the exception of locals, should start with one or two lower case letters that identify what scope they are. The remainder of the variable name must be descriptive and contain capitalized joined words.

Variable Type	Leading characters	Example
Task variable	tv	tvMainOptions
Class variable	cv	cvMainOptions
Instance variable	iv	ivMainOptions
Local variables	-	mainOptions
Parameters	p	pMainOptions

Variables which are field references or item references should end with "Ref", such as ivMainOptionsRef.

Method and Custom Notation Names

Public methods must, of course, start with a '$' sign. The reminder of the name should be all lower case so it is consistent with the Omnis notation.

Private methods must not start with a '$' sign. The actual name should be made up off Capitalized joined words. For example, "GetInvoiceList".

Column Names

Column names should follow the local variable naming conventions.

Window/Report Object Names

The default window/report object names assigned by Omnis must always be replaced with an appropriate object name. For window objects that are data-bound, use the data name without the leading characters. For example, if the data name was "ivLastName", the object name should be "LastName". For objects that are not data-bound, use an appropriately descriptive name with capitalized joined words. For example, "MainOptionsGroup" for a group box.

Appendix 2 – Shortcut keys

This appendix lists some useful shortcut keys. There are many more available; for a full list, see the manual *Using Omnis Studio, Chapter 2 - Omnis Tools* (available in PDF format on the Omnis Studio CD).

Omnis tools

Action	Windows/ Linux	Mac
Open the Browser	F2	Cmnd-2
Open the Component Store	F3	Cmnd-3
Open the Notation Inspector	F4	Cmnd-4
Open the Inheritance Tree	F5	Cmnd-5
Open the Property Manager	F6	Cmnd-6
Open the Browser Options	F7	Cmnd-7
Open the Catalog	F9	Cmnd-9
Open Find and Replace	Ctrl-F	Cmnd-F

General

Action	Windows/ Linux	Mac
Save the current class	Ctrl+S	Cmnd-S
Open the Print Destination dialog	Shift-Ctrl-P	Shift-Cmnd-P
Print the current selected object (report, method, file or class), for example, prints complete list of properties and methods for a class	Ctrl-P	Cmnd-P

Action	Windows/ Linux	Mac
Close the top window	Ctrl-W	Cmnd-W
Instantiate a class, for example, open a window, print a report, install a menu or toolbar	Shift-select class in View menu	
Interrupt current processing, such as method execution or report printing	Ctrl-Break	Cmnd-. (period)
Exit/Quit Omnis	Alt+F4	Cmnd-Q

Index