A.5 Quick Start

NeuArchitect v3.0

December 2000

6 Armstrong Road, Shelton, Connecticut 06484 USA
www.neuvis.com

Contents

1 Overview

Objectives

Welcome to the NeuVis NeuArchitect Quick Start. This workshop (based on some of the common elements of the Sample Application delivered with NeuArchitect), is designed to get you up and running quickly in the NeuArchitect Application Development Environment. This Quick Start also functions as a review for those who have had some contact with NeuArchitect, via an earlier class experience or from exposure to earlier versions of NeuArchitect. Using this Quick Start, you can expect to:

Discover NeuArchitect Components and Features

- Describe the configuration required to Run NeuArchitect
- Launch NeuArchitect
- Navigate through NeuArchitect
- Name the major components of NeuArchitect
- Relate features/benefits to critical elements of web design and construction

Create a Functional NeuArchitect Web Application

- Design elements of a Business Model using the Class Architect
- Create data connected Web Pages using Page Architect
- Construct and test an application using IIS/Personal Web Server.

Time Requirements

The amount of time required to complete the Quick Start will vary by experience.

- Experienced developer - from 1 to 1.5 hour(s)
- Less experienced developer – from 3 to 3.5 hours

Hardware/Software Requirements

The suggested hardware configuration for optimal performance includes:

- Processor - Pentium III
- Memory - 256 Megabytes
- Hard drive - 10 Gigabytes

In order to use this tutorial, the student must have the following software loaded:

- Microsoft NT Server or Workstation
- Microsoft Access or SQL Server
- Microsoft IIS or Personal Web Server
- NeuArchitect – latest version

EJB developers will need to download and install the Javasoft JDK 1.1.2

Steps for Building a Web Application with NeuArchitect

The steps for building a simple web application with NeuArchitect are outlined in the table below. The steps have been categorized according to their respective Quick Start Section in column 2.

Steps	Quick Start Section
1. Create the application's database	Section 2 - Define the Application Infrastructure
2. Create an ODBC connection to the database	
3. Configure web server (if required)	
4. Create a subdirectory for the application and images	
5. Create the application	Section 3 - The NeuArchitect Integrated Application Environment - Launch and Configuration
6. Specify Package settings	
7. Create the business objects (Components and Classes)	Section 4 – Creating the Business Object Model
8. Create tables and test data rows	
9. Create the web pages	Section 5 - WebSite Design
10. Construct and test the application	
11. Produce documentation	Section 6 – Creating NeuArchitect Documentation

Quick Start Overview

As mentioned in the Overview above, this workshop is based on a few of the common elements that can be found in the Sample Application delivered with the product. While the Sample Application provides the bandwidth of eCommerce functionality, the Quick Start explores rudimentary concepts designed to give the user a hands-on experience with NeuArchitect.

It should be noted that there are often several ways to accomplish a task in the NeuArchitect development environment. This Quick Start will, in most cases, expose you to concepts via an easily comprehendible set of guidelines. As your experience with the product broadens, you may become more comfortable with alternative techniques.

All steps in creating the Quick Start application should be followed without exception. The Quick Start takes a ground up development approach starting with business modeling and finishing with website design and functionality.

Finally, note that this Quick Start employs a skill-building methodology. The first time through a task, for instance, creating a Button, the steps are rich with instructions and supporting graphics. The next time a Button is required, just the specifications for the element will be given with reference back to the previously explained procedure for details.

The following topics provide a summary of the items that will be created in the Quick Start.

Creating the Classes for this Application

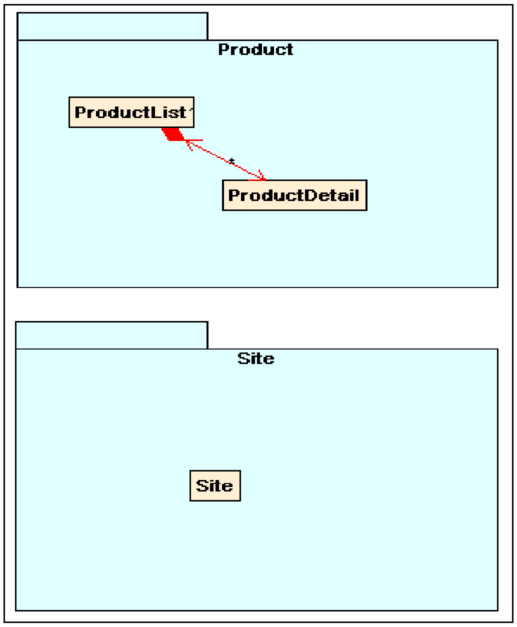
Three Classes will be modeled and created in this exercise to store data:

- Site Class – Registered User information

- ProductList Class – Catalog of Product Categories retailed by sellers
- ProductDetail Class – Detailed information about products listed in the Catalog.

Creating the Relationships for this Application

The only relationship that will be created in the Quick Start is a one-to-many, parent/child, between the ProductList and ProductDetail Classes. The diagram below, taken from Class Architect, shows the relationship between the Classes.



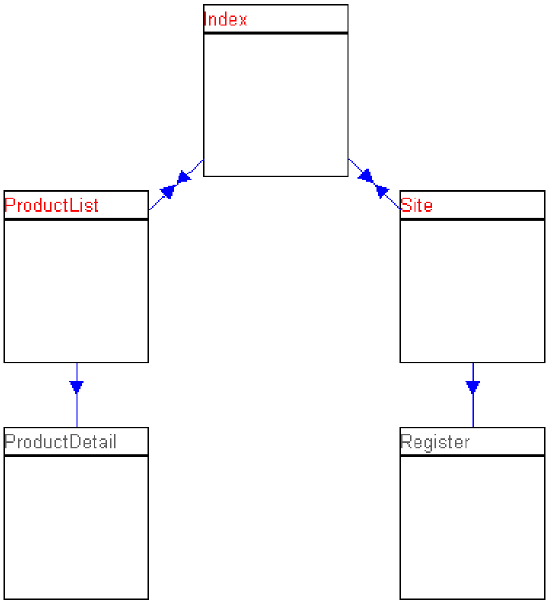
Creating the Methods for this Application

The Quick Start employs one method to Insert a row into the Site Class. You are not required to author the method as NeuArchitect will construct the Insert method automatically.

Defining the Web Pages for this Application

The Quick Start requires the developer to create 5 web pages and preview the pages in a browser: The diagram below, taken from Site Architect, shows the relationships between the pages.

- Index – Introductory page and navigational nerve center for the website. From this page the user can access Site, ProductList and ProductDetail.
- ProductList – Presents the user with a list of product categories in a table format. The user can select a product and access specific product information
- ProductDetail - Presents the user with a specific information about the product
- Site - Presents the user with a list of site participants in a table format. The site user can select an option to register from this screen
- Register – An Input screen to add participants to the Site Class



Saving Your Work

Use the same precautions that you would with any development tool and save your work often. Throughout this Quick Start guide you will see reminders to save your work like the one shown below.

Save your work.

2 Defining the Application Infrastructure

Overview

As with any application development product, successfully creating and deploying your new website depends on a correctly defined the infrastructure of supporting products and required directory folders. Accordingly, in this Section, you will:

- Create the application directory structure and populate the directories with required files.
- Create a new database - All HTTP transactions require "data persistence" — the temporary or permanent storing of client information on the server. NeuArchitect will create the RDBMS table definitions for your application.
- Create an ODBC driver data source to access the database - Because NeuArchitect constructs SQL ODBC calls to access the relational tables, create an ODBC driver data source through which your application can access the database.
- For JavaScript/ASP, Configure your Web Server to employ NeuArchitect preview features - Incorrect settings for IIS or the Personal Web Server can result in Web Pages that do not display when previewed in PageArchitect.

Create an Application Directory

Transfer to Windows Explorer and create a the subdirectory structure (shown below) where x is the drive where you will create your application. (While you can create the directory structure in NeuArchitect when saving your application for the first time, you will need to create it now, enabling you to set configurations in the steps that follow.

(`x:\Applications\Qstart`)

Creating a New Access or SQL Server Database

Transfer to Microsoft Access or SQL Server now and complete the following steps:

▼ To create a MS Access database

1. Start MS Access.
2. Select **Blank Database** and press **OK**.
3. Browse to your `QStart` project folder.
4. Key in your new database name – **Qstart**.
5. Press **Create**.
6. Close Access.

▼ To create an SQL Server database

1. Open your SQL Server Service Manager.
2. Start the SQL Server Enterprise Manager.
3. Use the Action Menu to create a new database called **Qstart**.

4. Close SQL Server Enterprise Manager.

Creating an ODBC Connection for the New Database

Transfer to the ODBC Data Source tool found in Windows Control Panel and complete the following steps:

▼ To create an ODBC connection in MS Access

1. Select the **System DSN** tab.
2. Press the **Add** button.
3. Select the Microsoft Access driver (`.mdb`).
4. Press **Finish**.
5. Specify the name (**Qstart**) and supply a description.
6. Press the **Select** button.
7. Browse for the Qstart database in the `x:\Project\Applications\Qstart` folder and double-click on it.
8. Press **OK** to complete the definition and **OK** to close the connection screen.

▼ To create an ODBC connection in SQL Server.

1. Select the **System DSN** tab.
2. Press the **Add** button.
3. Select the SQL Server driver.
4. Press **Finish**.
5. Specify the name (**Qstart**), supply a description and specify (local) Server.
6. Press **Finish** to complete the definition and **OK** to close the connection screen.

Configuring IIS/Personal Web Server

▼ To configure Personal Web Server

1. Transfer to Microsoft Personal Web Manager tool.
2. Click **Advanced Options**.
3. Click the **Add** button.
4. Click the **Browse** button and select your default application directory. (`x:\Applications`)
5. Supply an alias name – **Applications**.
6. Check the Access boxes for Read, Execute, and Scripts.
7. Click **OK**. (If prompted to create the directory, click Yes.)
8. Minimize or close the Personal Web Manager.

▼ To configure IIS

1. Start up the Internet Service Manager.

2. Using the + to expand views, drill down as follows:

   Internet Information Server | Machine ID | Default Website

3. Right Click on **Default Website** and select **New | Virtual Directory**
4. Supply an alias name – **Qstart**.
5. Supply the physical path of the directory with content `x:\applications\qstart`
6. On the next screen, check off the first three boxes **Allow Read**, **Allow Script Access**, **Allow Execution Access**.
7. Click **Finish**.
8. Still in Service Manager, use the + to open the Default Website list.
9. Right click on **Qstart** and select **Properties**.
10. Click the **Documents** tab.
11. Move `default.asp` to the top of the list.
12. Click **OK** and Close the Internet Service Manager.

3 NeuArchitect IDE: Launch and Configuration

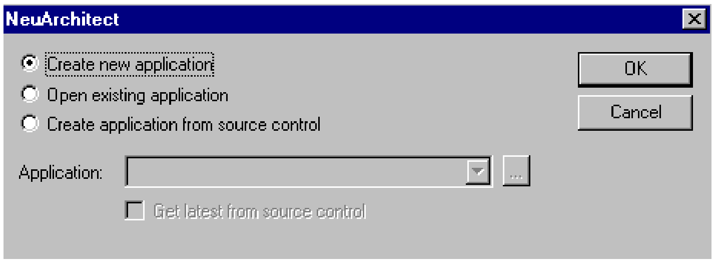
Overview

In this Section you will launch NeuArchitect and configure the setting required to develop the Quick Start application. The steps in Section 2 – Defining the Application Infrastructure, must be completed before starting this Section.

Creating the NeuArchitect Application

▼ To create a NeuArchitect application

1. Launch NeuArchitect.
2. When prompted, check **Create new application** and Click **OK**.



● Create new application
○ Open existing application
○ Create application from source control

Application:

□ Get latest from source control

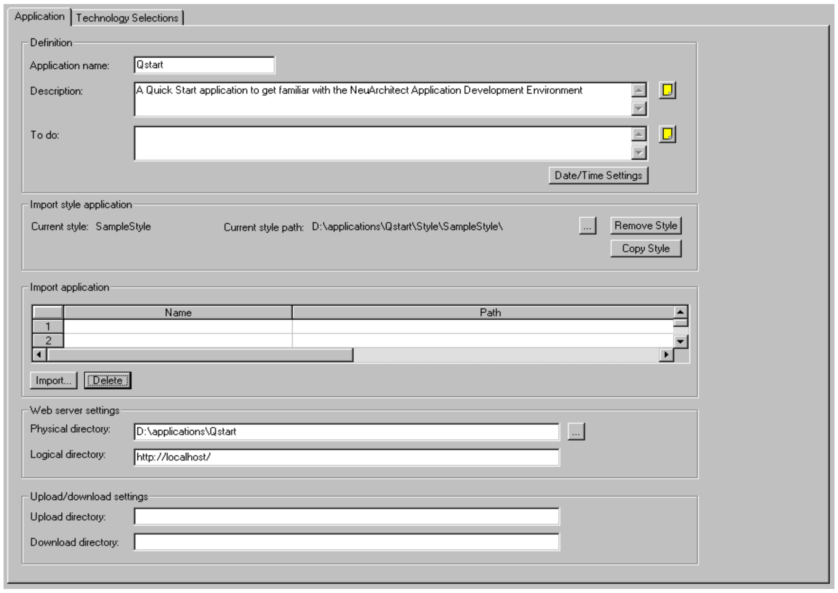
Defining Application Parameters

The Application Tab is used to set the parameters for the application. Complete the required general information about the application on the Application tab.

▼ To set the parameters for the application

1. Supply the application name – **Qstart.**
2. Optionally, enter Description information about the application.
3. Select Save from the File pulldown menu and Save the application as `Qstart.OMD` directing the save into `x:\application\Qstart` where x is the drive where you want to save the application
   - Note: NeuArchitect will create, on your behalf, a Component called 'NewComponent' containing a Global Class. The use of this Class will not be covered in this QuickStart.
4. If necessary, modify the logical directory to include your directory alias
   - If using the JavaScript/ASP: `http://localhost/Applications`
   - If using EJB/WebSphere or EJB/WebLogic, the default should already be set: `http://localhost`

When you have finished, your screen should appear similar to the illustration below (logical directory shown is for javaScript/ASP).



Application name: Qstart

Description: A Quick Start application to get familiar with the NeuArchitect Application Development Environment

To do:

Current style: SampleStyle

Current style path: D:\applications\Qstart\Style\SampleStyle\

Physical directory: D:\applications\Qstart

Logical directory: http://localhost/

Upload directory:

Download directory:

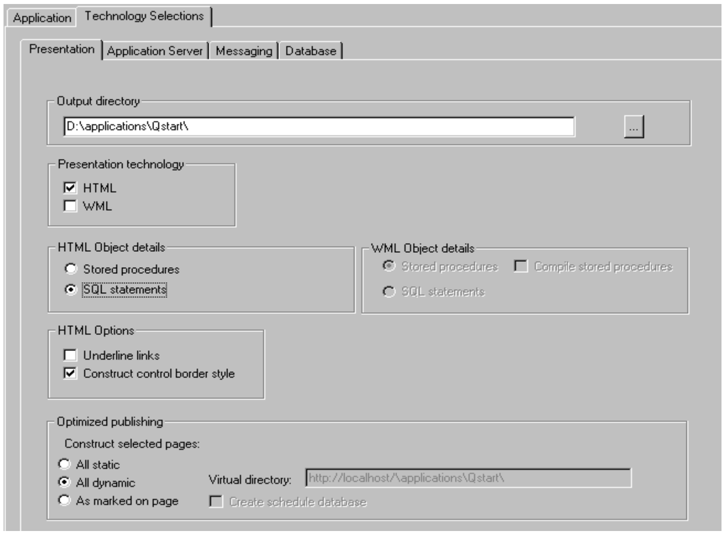
Save your work.

Defining the Application Technology

The steps in this topic are executed from the Technology Selections tab. The Technology Selections tab displays four new tabs associated with application parameters. Your first settings will be on the Presentation tab. These are default settings that are picked up when you create pages and can be can be changed for individual pages. The individual page settings are actually used during construction.

▼ Verify the settings on the Presentation tab

1. Presentation technology - **HTML**.
2. HTML Object details - **SQL statements**
3. Optimized publishing - **All dynamic**



Output directory

D:\applications\Qstart\ ...

Presentation technology

☑ HTML
□ WML

HTML Object details

○ Stored procedures
● SQL statements

WML Object details

● Stored procedures □ Compile stored procedures
○ SQL statements

HTML Options

□ Underline links
☑ Construct control border style

Optimized publishing

Construct selected pages:

○ All static
● All dynamic
○ As marked on page

Virtual directory: http://localhost/\applications\Qstart\

□ Create schedule database

▼ To make the settings on the Application Server tab, follow the steps for the appropriate technology

For Microsoft JavaScript with ASP pages

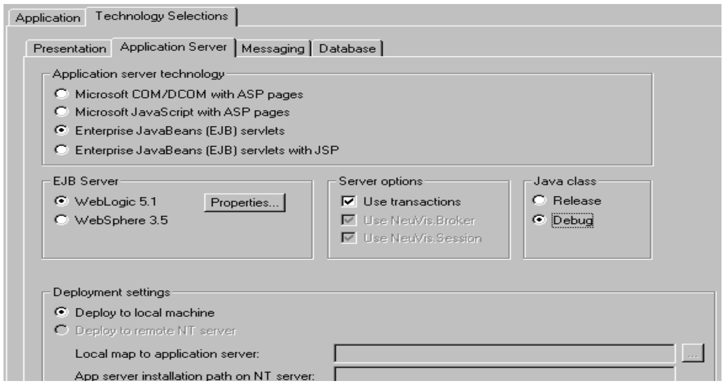
1. Check **Microsoft JavaScript with ASP pages**

For Enterprise JavaBeans (EJB) servlets with WebLogic 5.1

1. Check **Enterprise JavaBeans (EJB) servlets**
2. Check **WebLogic 5.1**
3. Check **Debug** for **Java Class.**

For Enterprise JavaBeans (EJB) servlets with WebSphere 3.5

4. Check **Enterprise JavaBeans (EJB) servlets**
5. Check **WebSphere 3.5**
6. Check **Debug** for **Java Class.**



Application server technology

○ Microsoft COM/DCOM with ASP pages
○ Microsoft JavaScript with ASP pages
● Enterprise JavaBeans (EJB) servlets
○ Enterprise JavaBeans (EJB) servlets with JSP

EJB Server

● WebLogic 5.1
○ WebSphere 3.5

Server options

☑ Use transactions
☑ Use NeuVis.Broker
☑ Use NeuVis.Session

Java class

○ Release
● Debug

Deployment settings

● Deploy to local machine
○ Deploy to remote NT server

Local map to application server:

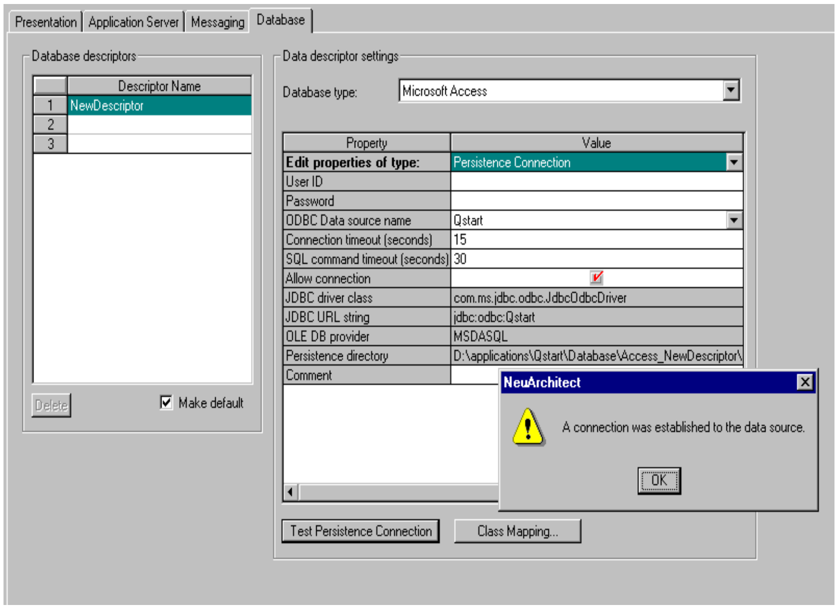
App server installation path on NT server:

▼ To make the settings on the Database tab

1. Select **Microsoft Access** or **SQL Server** as the Database type.

2. Set Edit properties of type to **Persistence Connection**

3. Click the **ODBC Data source name** dropdown.
4. Choose **Qstart** as your Data source name.
5. Click the **Test Persistence Connection** button to ensure you are connected to your database.
6. Click back to the Application tab.



	Descriptor Name
1	NewDescriptor
2	
3	

Database type: Microsoft Access

Property	Value
Edit properties of type:	Persistence Connection
User ID	
Password	
ODBC Data source name	Qstart
Connection timeout (seconds)	15
SQL command timeout (seconds)	30
Allow connection	✓
JDBC driver class	com.ms.jdbc.odbc.JdbcOdbcDriver
JDBC URL string	jdbc:odbc:Qstart
OLE DB provider	MSDASQL
Persistence directory	D:\applications\Qstart\Database\Access_NewDescriptor\
Comment	

NeuArchitect

A connection was established to the data source.

OK

Save your work.

4 Creating the Business Object Model

Overview

Defining the Business Model to NeuArchitect involves Classes, Class Relationships and Business Rules, Each NeuArchitect application has one Package that captures settings and organizes all of the business models.

A Component logically groups/contains a related set of business objects (Classes). Most real-world applications consist of at least 100 Classes. Components help manage the complexity of real-world business models by segmenting and organizing them at a higher level of abstraction. Components are "logical" elements of a NeuArchitect application that do not construct code.

A Class represents anything of relevance to the application. Each Class contains Attributes that define Class characteristics. Classes are associated to other Classes through Relationships - a link between two Classes that describes how they interact. Methods (custom business rules) and Pages (web pages) are also part of a Class, but defined elsewhere in NeuArchitect.

In the QStart application, ProductList and ProductDetail are linked by "owning" one-to-many relationships. "Owning" implies that the without the ancestor object, the information stored in the dependent, "owned”, object can be deleted when the ancestor object is deleted.

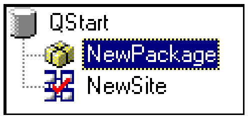
Defining the Package, Components, and Classes.

Define the Package

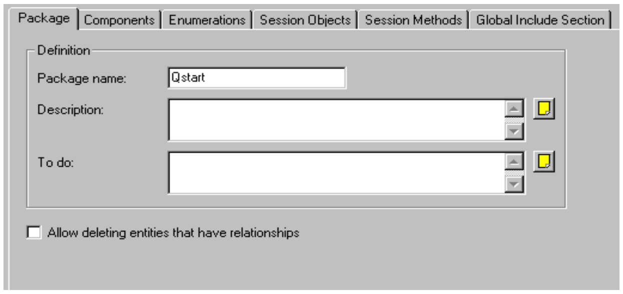
Each NeuArchitect application has one Package that captures settings and organizes all of the business models.

▼ To define the package

1. Click on the **NewPackage** icon.



2. Use the Package tab to name the Package **Qstart**.



Package name: Qstart

Description:

To do:

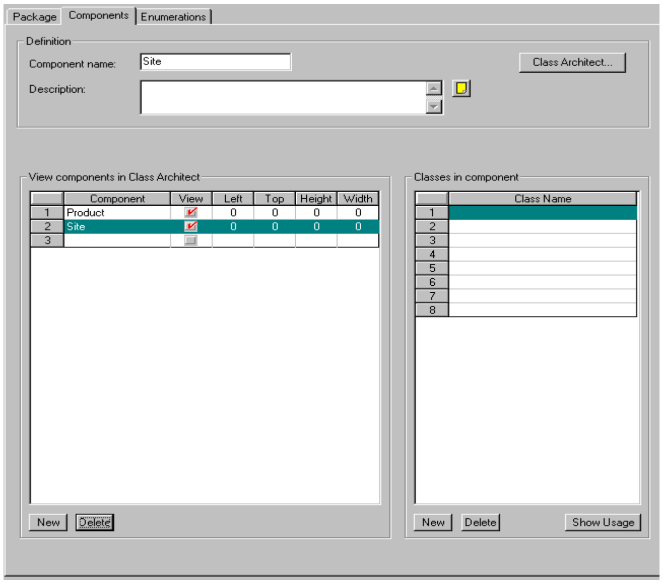
□ Allow deleting entities that have relationships

Define the Components

Components logically group/contain related sets of business objects (Classes).

▼ To define the Components

1. Click on the Components tab to create the Components for the Quick Start application.
2. Click the **New** button.
3. Type in **Product** for the Component name.
4. Click the **New** button again
5. Type in **Site** for the Component name.



Component name: Site

Description:

	Component	View	Left	Top	Height	Width
1	Product	☑	0	0	0	0
2	Site	☑	0	0	0	0
3		□				

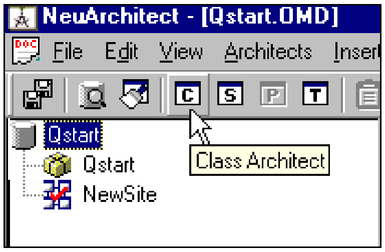
	Class Name
1	
2	
3	
4	
5	
6	
7	
8	

Define Classes

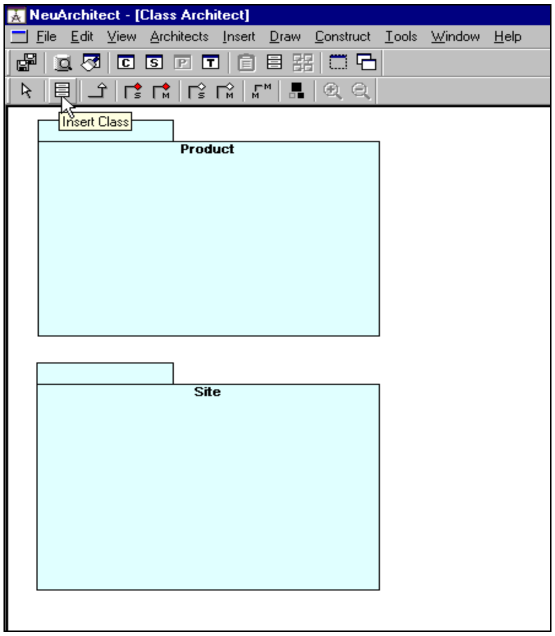
A Class represents anything of relevance to the application. Each Class contains Attributes that define Class characteristics. While there are several ways to create Classes in NeuArchitect, Class Architect will be used for the Quick Start application.

▼ To define classes

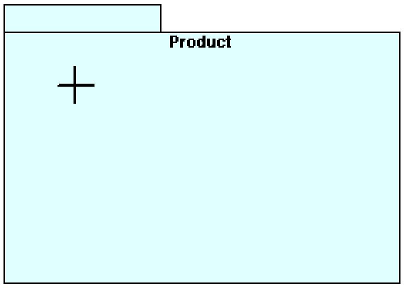
1. Click on the Class Architect icon.



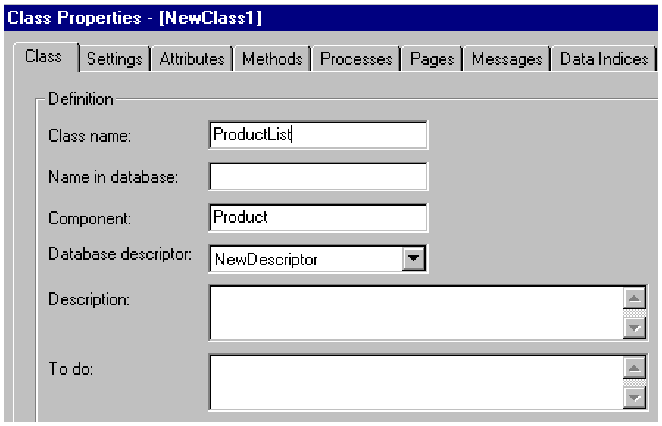
2. In the illustration below, the Components: Product and Site have been resized for a more convenient display. To resize your display, select the Component and resize using handles. To move the Component on the canvas, hold down the Ctrl key, click and drag the Component to the desired location.
3. Click on the Insert Class icon located on the ClassArchitect toolbar.



4. Move your cursor into the Product Component. You should see intersecting lines appear as you move your cursor.



5. Click your left mouse button to drop your Class into the Product Component.
6. Double-clickon the Class to access Class Properties.
7. Change the Class name to **ProductList** and click **OK**.



Definition

Class name: ProductList

Name in database:

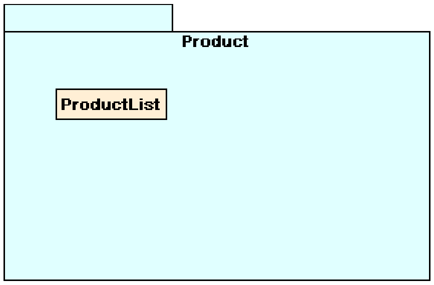
Component: Product

Database descriptor: NewDescriptor

Description:

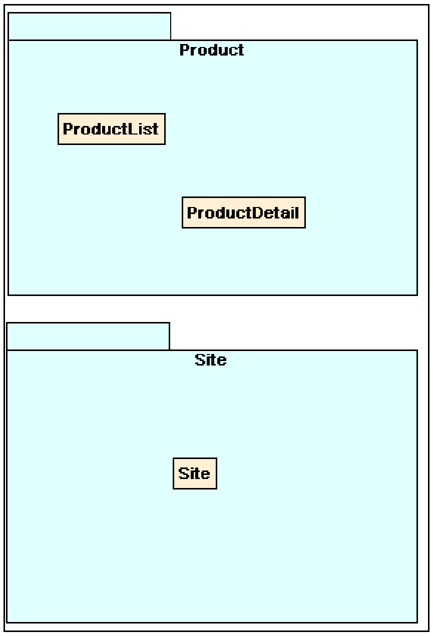
To do:

8. Your Product Component should appear as follows:



9. Repeat steps 1-6 to create the **ProductDetail** Class in the Product Component.

10. Repeat steps 1-6 to create the **Site** Class in the Site Component. Your Components for Product and Site should appear as follows:



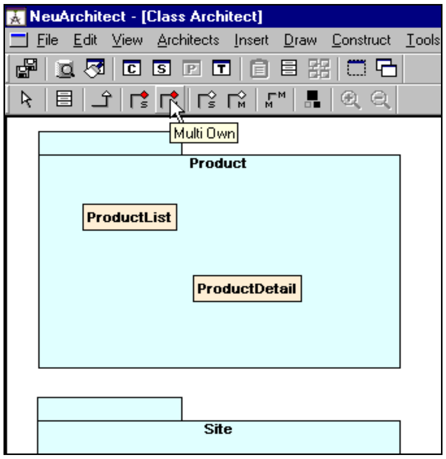
Save your work.

Define Links between the Related Classes

Classes are associated to other Classes through Relationships - a link between two Classes that describes how they interact. ProductList requires a one to many relationship with ProductDetail.

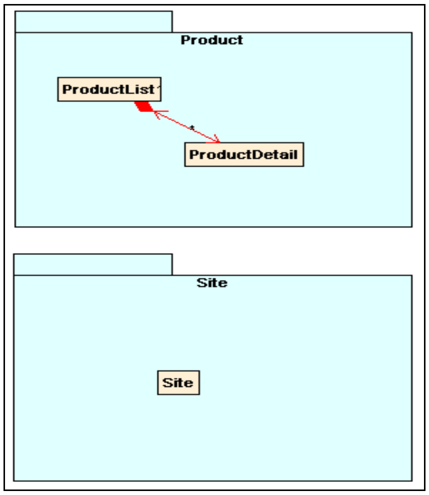
▼ While in Class architect, to create this relationship

1. In Class Architect, click on the Multi Own icon.


NeuArchitect - [Class Architect]
File Edit View Architects Insert Draw Construct Tools
Multi Own
Product
ProductList
ProductDetail
Site

2. Using the following technique, establish the link between ProductList and ProductDetail.
   - Click and hold over the **from**, ancestor Class (ProductList).
   - Drag and draw over the **to**, dependent Class (ProductDetail).
   - Release the mouse button.

Your view in Class Architect should appear similar to the following when refreshed.



Defining Class Attributes

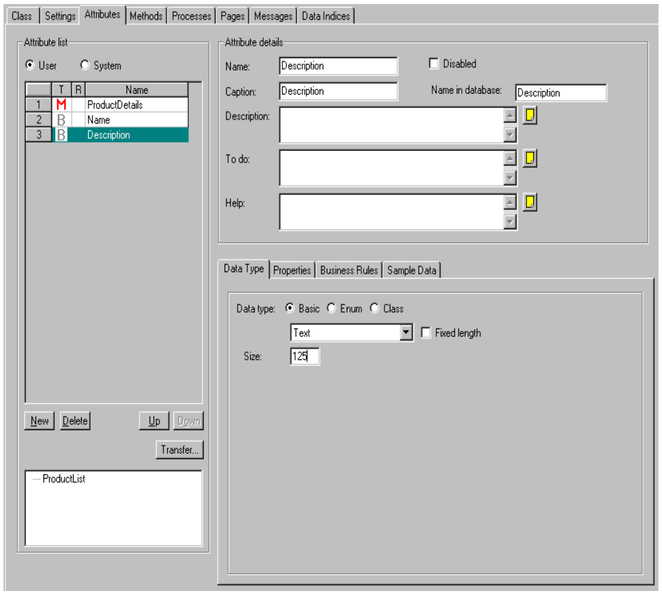
Each Class contains Attributes that define Class characteristics.

To simplify the task of creating the attributes for this application in Class Architect, you will find a process defined below. The process is followed by a series of tables under **Class Attributes for the Quick Start Application** listing Attribute names, types and field lengths where applicable. Follow the process to create all the attributes in the tables.

Define Class Attributes – Process

▼ To create the Attributes for your Classes.

1. Double-clickon the Class.
2. Select the **Attributes** tab.
3. Use the **New** button to add a new attribute to the list from the tables below.
4. Type the Attribute name in the Name field under Attribute details (Do not remove or overwrite the key fields already present in the list).



● User ○ System

	T	R	Name
1	M		ProductDetails
2	B		Name
3	B		Description

ProductList

Name: Description

□ Disabled

Caption: Description

Name in database: Description

Description:

To do:

Help:

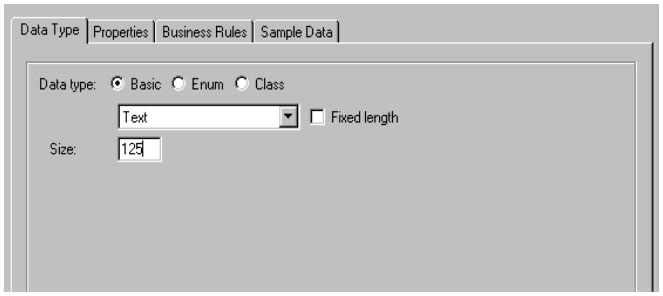
Data type: ● Basic ○ Enum ○ Class

Text

□ Fixed length

Size: 125

5. Specify the Data type using the Data type radio buttons.
6. Further define the Attribute Data type.
   - If a **Basic** Data type, specify type (text, integer, etc.) in the drop-down list and indicate size if required.

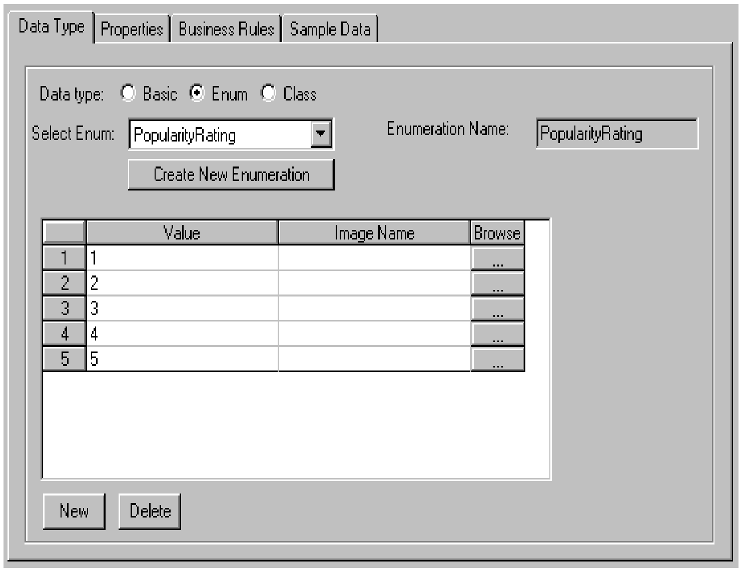


Data type: ● Basic ○ Enum ○ Class

Text □ Fixed length

Size: 125

- If an **Enum** (Enumeration) Data type, select **Create New Enumeration** and use the **New** button to supply each of the required Enum values.



Data type: ○ Basic ● Enum ○ Class

Select Enum: PopularityRating

Enumeration Name: PopularityRating

	Value	Image Name	Browse
1	1		...
2	2		...
3	3		...
4	4		...
5	5		...

7. Repeat steps 3-6 for all attributes required for a Class.
8. Click **OK** to close the window.

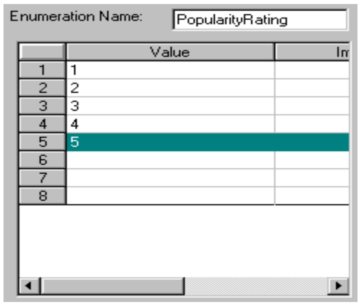
Class Attributes for Quick Start Application

Create the following Attributes for the ProductList Class:

Attribute Name	Type	Length
Name	Basic - Text	50
Description	Basic - Text	125

Create the following Attributes for the ProductDetail Class:

Attribute Name	Type	Length
Name	Basic - Text	50
Description	Basic - Text	200

Brand	Basic - Text	75
Price	Basic - Money	N/A
PopularityRating	Enum - as shown  Enumeration Name: PopularityRating Value 1 1 2 2 3 3 4 4 5 5 6 7 8	N/A

Create the following Attributes for the Site Class:

Attribute Name	Type	Length
Name	Text	50
Description	Text	100

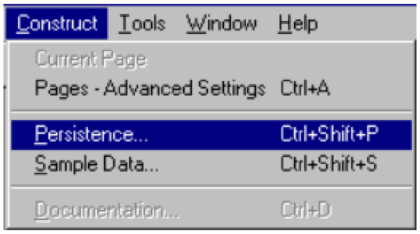
Save your work.

Constructing Persistence

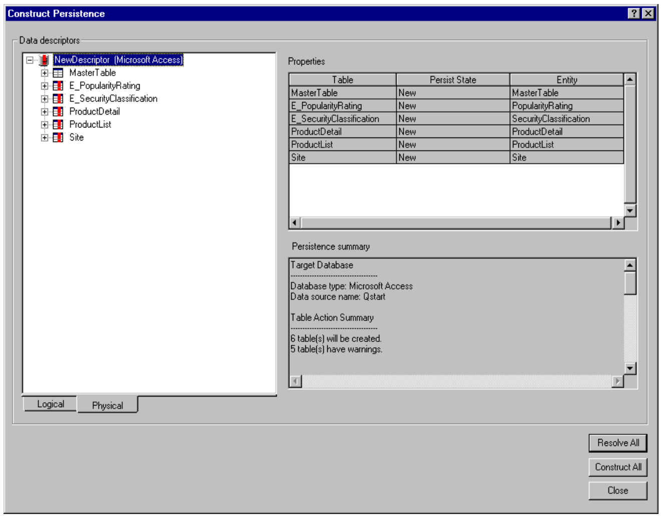
When you Construct Persistence, a persistent class (a physical table) will be created in your database for each Class you defined in NeuArchitect. If you have defined an Enum, an Enum table will also be created. An Enum table bears the Enumeration name supplied to NeuArchitect preceded by E_. In our case the table will be named E_PopularityRating.

▼ To construct persistence

1. From the top menu bar, select **Construct**.
2. Select **Persistence**.


Construct
Tools
Window
Help
Current Page
Pages - Advanced Settings Ctrl+A
Persistence... Ctrl+Shift+P
Sample Data... Ctrl+Shift+S
Documentation... Ctrl+D

3. Double-click on the root of the tree (**New descriptors**) to select all of the Classes.



NewDescriptor (Microsoft Access)
- MasterTable
- E_PopularityRating
- E_SecurityClassification
- ProductDetail
- ProductList
- Site

Properties

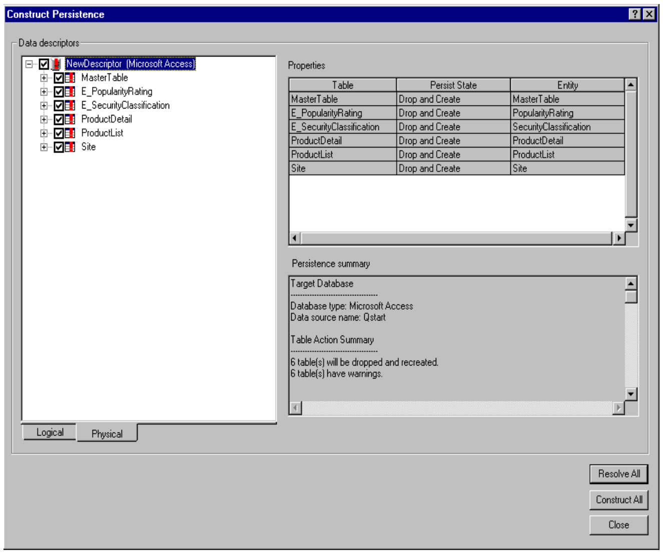
Table	Persist State	Entity
MasterTable	New	MasterTable
E_PopularityRating	New	PopularityRating
E_SecurityClassification	New	SecurityClassification
ProductDetail	New	ProductDetail
ProductList	New	ProductList
Site	New	Site

Persistence summary

```
Target Database
------------------------------
Database type: Microsoft Access
Data source name: Qstart

Table Action Summary
------------------------------
6 table(s) will be created.
5 table(s) have warnings.
```

4. Click **Resolve All**.



☑ NewDescriptor (Microsoft Access)
- ☑ MasterTable
- ☑ E_PopularityRating
- ☑ E_SecurityClassification
- ☑ ProductDetail
- ☑ ProductList
- ☑ Site

Properties

Table	Persist State	Entity
MasterTable	Drop and Create	MasterTable
E_PopularityRating	Drop and Create	PopularityRating
E_SecurityClassification	Drop and Create	SecurityClassification
ProductDetail	Drop and Create	ProductDetail
ProductList	Drop and Create	ProductList
Site	Drop and Create	Site

Persistence summary

```
Target Database
------------------------------
Database type: Microsoft Access
Data source name: Qstart

Table Action Summary
------------------------------
6 table(s) will be dropped and recreated.
6 table(s) have warnings.
```

5. Click **Construct All**, the window will close automatically when complete.
6. Optionally, you can start your database manager and view the tables that were created.

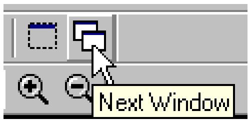
Save your work.

Creating Sample Data

NeuArchitect will populate a designated number of rows with sample data for testing purposes. You will create sample data for each of the Classes in the Quick Start, ProductList, ProductDetail and Site.

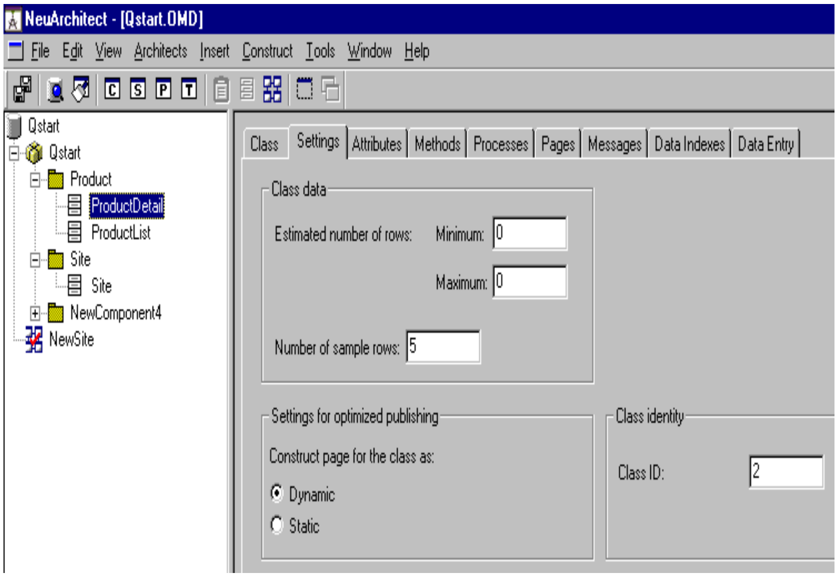
▼ To create sample data

1. If you are still in ClassArchitect, click the Next Window icon to get back to the Navigator view.



2. Expand the Components in the Application Navigator enabling you to see their respective Classes.

3. Use the **Settings** tab for each Class you select in the Application Navigator, starting with ProductList, to indicate the number of sample rows of data required for the selected Class. Follow the guidelines in the table just below the illustration for the number of sample rows for each Class.


NeuArchitect - [Qstart.OMD]
File Edit View Architects Insert Construct Tools Window Help
Qstart
Qstart
Product
ProductDetail
ProductList
Site
Site
NewComponent4
NewSite
Class Settings Attributes Methods Processes Pages Messages Data Indexes Data Entry

Class data

Estimated number of rows: Minimum: 0

Maximum: 0

Number of sample rows: 5

Settings for optimized publishing

Construct page for the class as:

● Dynamic

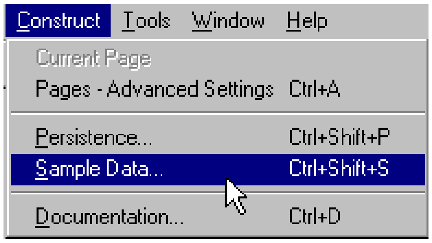
○ Static

Class identity

Class ID: 2

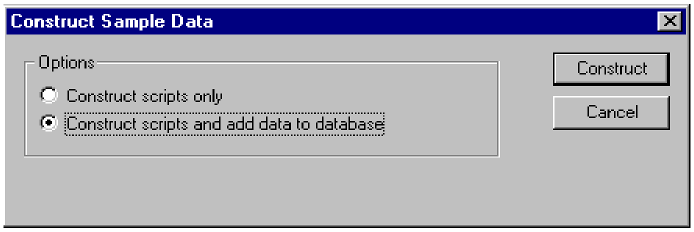
Class	# of Rows
ProductList	5
ProductDetail	10
Site	5

4. Select **Sample data** from the **Construct** pulldown menu


Construct Tools Window Help
Current Page
Pages - Advanced Settings Ctrl+A
Persistence... Ctrl+Shift+P
Sample Data... Ctrl+Shift+S
Documentation... Ctrl+D

5. Select ‘Construct scripts and add data to database’ Option

6. Click **Construct** button.



Construct Sample Data

Options

○ Construct scripts only

● Construct scripts and add data to database

Note: the sample data in the screens that follow has been replaced with more meaningful data. It is suggested that you **not** make data changes at this time.

Save your work.

5 Website Design

Overview

Page Architect, within NeuArchitect, is the editor for creating your web pages. Page Architect is:

- A technology independent web-page development facility
- Fully integrated with the business and process models
- A WYSIWYG editor for GUI development and web page preview

You will be creating, designing and testing 5 web pages. The first page will be an Index page with standard Buttons that will load the either the ProductList or Site page when clicked. The Site page will list system users with an option to branch to a Register page to add a user. Register employs a SubmitButton to insert a row into the Site Class. The ProductList page will list product categories with a Button to drill into the ProductDetail page.

Important

Note that each type of control in Page Architect has a predetermined set of properties that includes, size, shape, x/y coordinates (relative to the browser) color, font and many other characteristics that can be customized for optimum rendering. A few highlight notes about defining page controls follow.

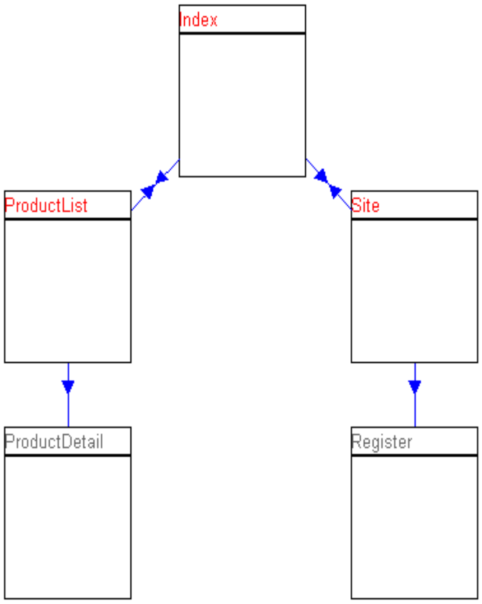
There are two forms of Buttons:

- SubmitButtons, which are used to fire off business rules
- Standard Buttons, which do not fire off methods, but are used to navigate from page to page.

An ObjectGrid control is one of several NeuArchitect graphical elements that allow you to construct dynamic web-content, based on supported data sources. ObjectGrids are very simple to create and maintain, providing an efficient and attractive means of displaying data in a browser.

Creating Pages in Site Architect

Site Architect is a tool that supports visual web design. Pages can be created and linked together using the tools provided with Site Architect. An example is shown below.

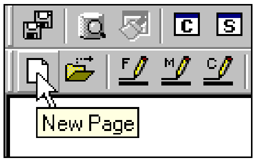


Create Pages for Quick Start

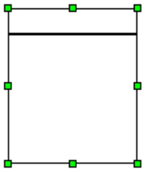
Open Site Architect by clicking the **S** icon on the toolbar. Five pages, (**Index, ProductList, ProductDetail, Site, and Register**) are needed for the Quick Start website.

▼ To create these five pages, repeat steps 1 through 3 below for each page.

1. Click the **New Page** icon and drop the page on the map.

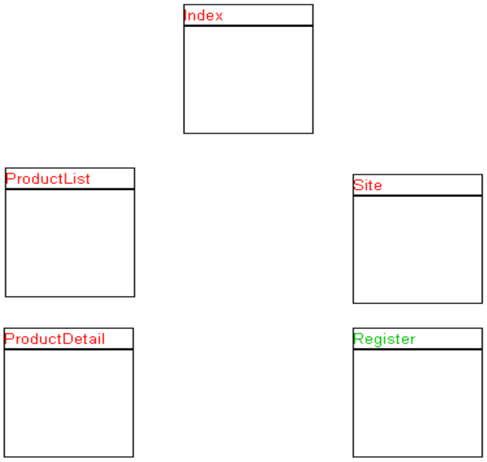


2. Click on the blank canvas area to drop the new page on the canvas.



3. Type the name of the page.

4. You may drag the pages in different positions to arrange them into a logical format. When finished, your view in Site Architect should appear similar to the following:



Set Page Properties in Site Architect

There are two important properties that must be set for the pages in the Quick Start.

- Class – the page needs to be associated with a Class.
- Data access type – sets the type of access the page will have to data (Full, Enter only, etc).

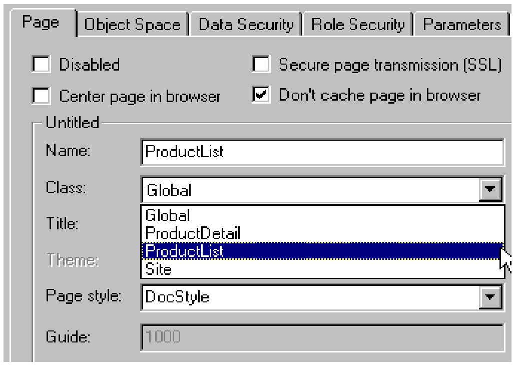
Both of these properties are set on the Page tab of the selected page's properties. Follow the procedure for setting the required page properties as defined in the table below.

Page	Class	Data access type
Index	Site	Read Only
ProductList	ProductList	Read Only
ProductDetail	ProductList*	Read Only
Site	Site	Read Only
Register	Site	Enter Only
* Setting the ProductDetail’s page Class property to ProductList will enable selective access to the children in ProductDetail from ProductList.		

▼ To set the page properties

1. Double-click on the page in Site Architect to access the **Page** properties.

2. Use the drop-down list to set the **Class**.



□ Disabled
□ Secure page transmission (SSL)
□ Center page in browser
☑ Don't cache page in browser

Untitled

Name: ProductList

Class: Global

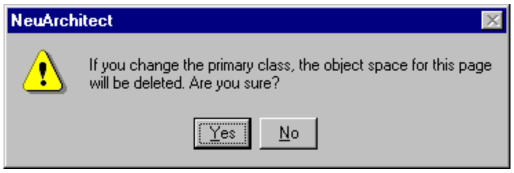
Title:

Theme:

Page style: DocStyle

Guide: 1000

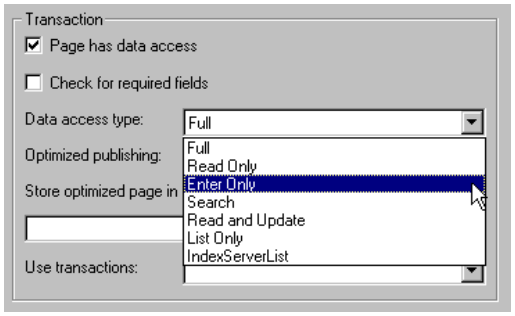
3. If you receive the following message, click **Yes**.



NeuArchitect

If you change the primary class, the object space for this page will be deleted. Are you sure?

4. Use the drop-down list to set the **Data access type**.



Transaction

☑ Page has data access

□ Check for required fields

Data access type: Full

Optimized publishing:

Store optimized page in

Use transactions:

5. Click **OK** when finished.

Save your work.

Designing Pages in Page Architect

Page Architect is used to create content for the pages constructed in Site Architect. You will be utilizing some of the tools provided for

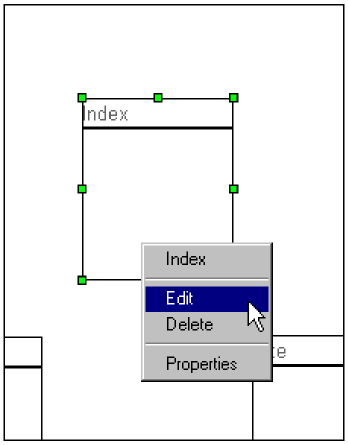
- Page Content
- Page Linking
- Navigational Controls
- Data Controls

Follow the steps below to access page Architect for the first time in a given application. Use the steps on the Index page:

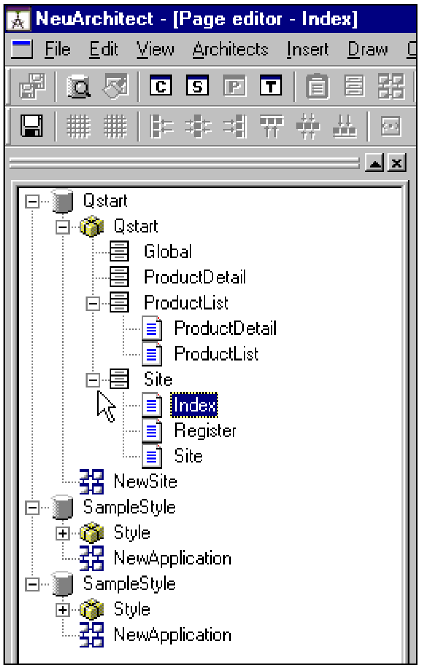
Click on the page to ensure it has the focus

Right click on the page

Select **Edit**.



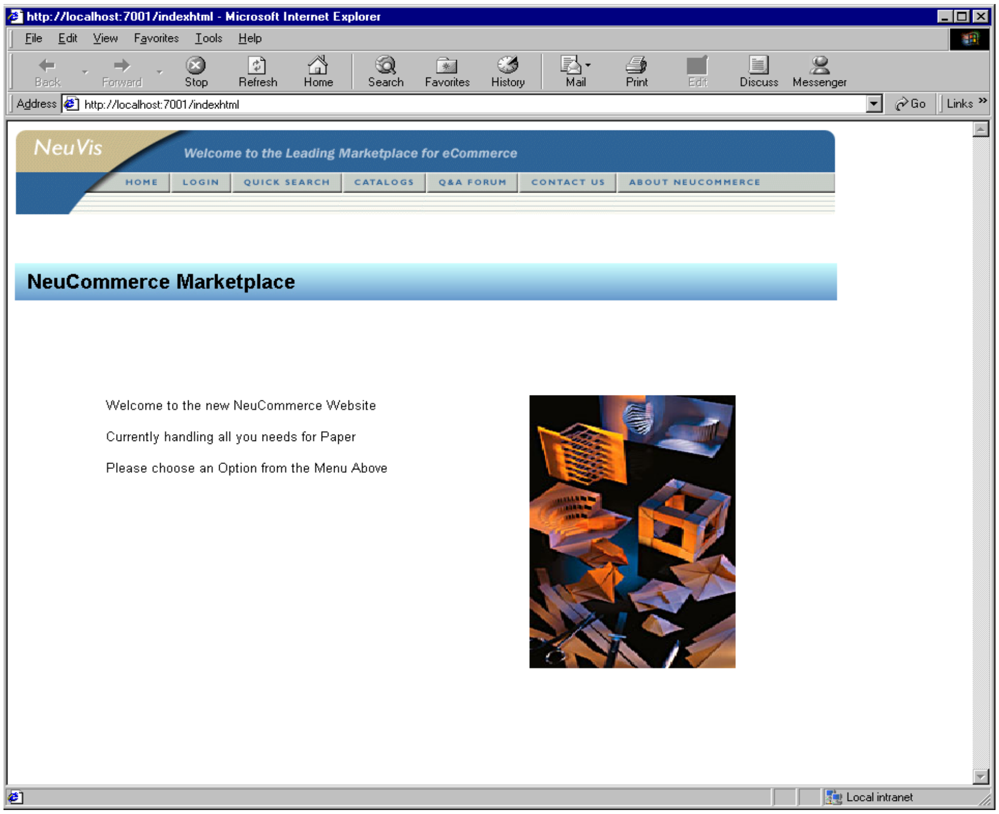
In future sessions you will be able to open Page Architect by clicking the **P** icon. This will launch page Architect with the most recent page that has been edited within your application. Once in Page Architect, expand the view in your Page Navigator window to display the pages associated with the application Classes.



Creating Page Content

Index Page

Our design for the Index page will be simple yet functional. When you have finished, your Index page will look similar to the example below: You may want to refer back to this page as you create the content items for look and feel.



NeuCommerce Marketplace

Welcome to the new NeuCommerce Website

Currently handling all you needs for Paper

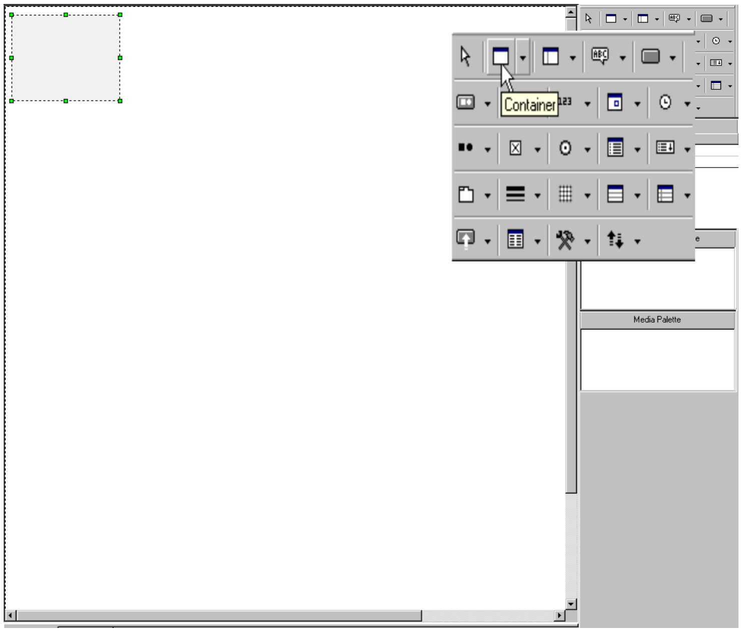
Please choose an Option from the Menu Above

Select the Index page in the Application Navigator and follow the steps below to design your Index Page in PageArchitect.

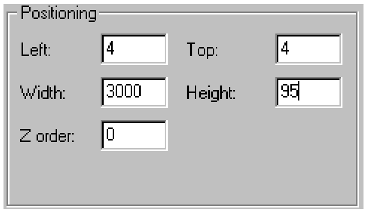
Containers

▼ To create a container for the main graphic

1. Select a Container from the Control Palette and drop it on the page approximately as shown.



2. Double-click on the container to access its properties.
3. On the Control tab, set the **Positioning Width** to **800** and the **Height** to **95**.


Positioning

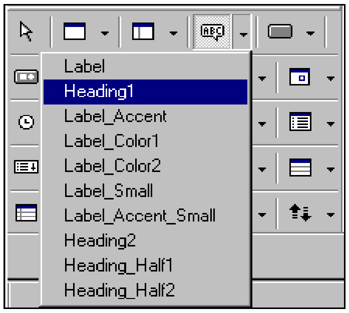
Left:	4	Top:	4
Width:	3000	Height:	95
Z order:	0		

4. Click the **Display** tab.
5. Click the dropdown for **Back image:**.
6. Select `HeaderMain_bg.gif`.
7. Click the **Center** radio button.
8. Click **OK**.

Labels

▼ To create a label

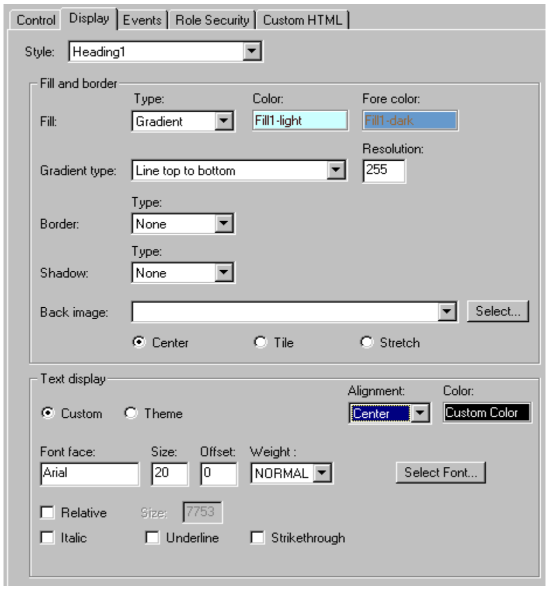
1. From the Control Palette, Select the **Label** drop-down and click the style, **Heading1**.



2. Position your cursor in the under the container you just created and click to drop the label in place.

3. Type an appropriate heading for your index page such as **NeuCommerce Marketplace**.

4. To change the font size, Double-clickon the label, select the **Display** tab, click **Custom** under **Text display**, make your font changes and click **OK**.


Control | Display | Events | Role Security | Custom HTML

Style: Heading1

Fill and border

Fill: Type: Gradient; Color: Fill1-light; Fore color: Fill1-dark

Gradient type: Line top to bottom; Resolution: 255

Border: Type: None

Shadow: Type: None

Back image: Select...

● Center ○ Tile ○ Stretch

Text display

● Custom ○ Theme; Alignment: Center; Color: Custom Color

Font face: Arial; Size: 20; Offset: 0; Weight : NORMAL; Select Font...

□ Relative Size: 7753

□ Italic □ Underline □ Strikethrough

5. Resize the label for a proportional look and feel.

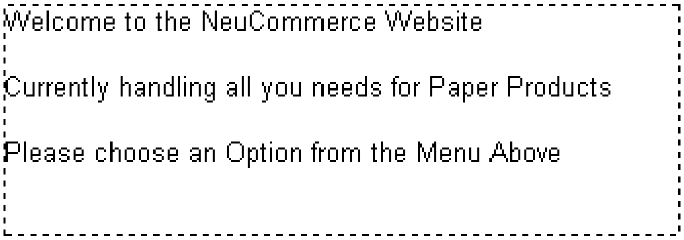
6. Create your next label using the plain Label style.

7. To display a multi line label, right click on the new label and select **Properties**.

8. Click **Multi-line** next to the Label field on the **Control** tab.

9. Click the Icon to open the Editor.

10. Type your welcome message with instructions for the user to select from the options available.



Welcome to the NeuCommerce Website

Currently handling all you needs for Paper Products

Please choose an Option from the Menu Above

11. Click OK in the Editor and OK once again on the Control Dialog.

Graphics

Adding a graphic to your Index page is optional. Any graphic you have available may be used. The process for displaying a graphic is similar to creating a label.

▼ To display a graphic

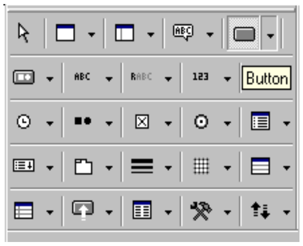
1. Drop a label on the page in the area you would like a graphic displayed.
2. Right click on the label and select **Properties**.
3. On the **Control** tab, blank out the name of the label.
4. Click on the **Display** tab.
5. Click the dropdown for **Back image:**.
6. Select `paper.gif`.
7. Click the **Stretch** button.
8. Click **OK**. (Do not be concerned with the display of the label name. It appears in design but not at run time.

Buttons

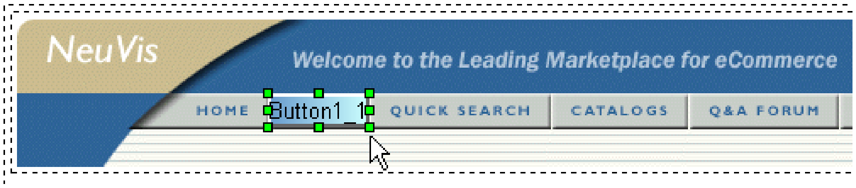
Buttons are used to drive events such as loading other pages into the client browser. Typically, a Login button would drive into validation for site access or send the user to Registration. We will use the Login button to access Registration and to display a list of Site users. We will use a Catalog button to open our ProductList page. On the Index page we will use clear buttons that will overlay the options on our main graphic.

▼ To create a button

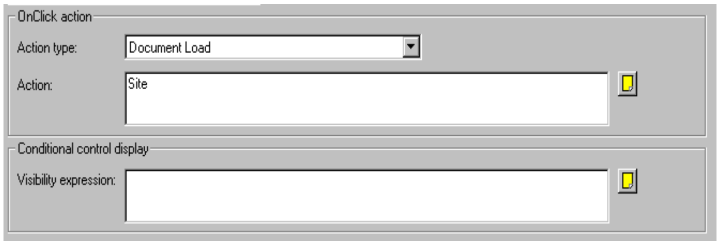
1. Select the **Button** control from the Control Palette.



2. Draw it on the page over the Login image in the upper graphic as shown below.



3. Double-clickon the button.
4. Clear the **Name** field on the **Control** tab.
5. Using the dropdown, set **OnClick** action, **Action type** to **Document Load**.
6. Type the name of the page you want to load in the action field – **Site**.



OnClick action

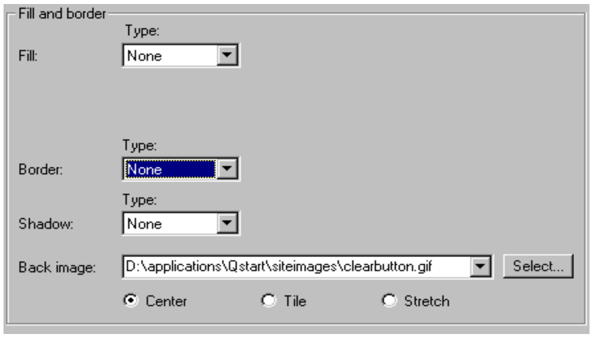
Action type: Document Load

Action: Site

Conditional control display

Visibility expression:

7. On the **Display** tab, change **Fill** and **Border** to **None**.
8. Set the **Back image** to `clearbutton.gif`.



Fill and border

Fill: Type: None

Border: Type: None

Shadow: Type: None

Back image: D:\applications\Qstart\siteimages\clearbutton.gif Select...

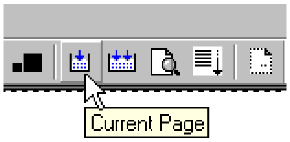
● Center ○ Tile ○ Stretch

9. Click **OK**.
10. Repeat steps 1-9 creating a similar Button over Catalogs. This should do a Document Load of ProductList.

Save your work.

Construct the Index Page

Click on the Current Page icon to construct the Index page.

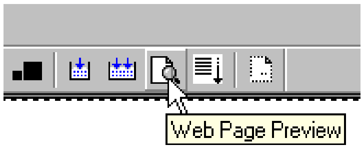


Preview the Index Page

Note: if you are using EJB/WebLogic refer to Appendix A, Starting and Stopping the WebLogic Server

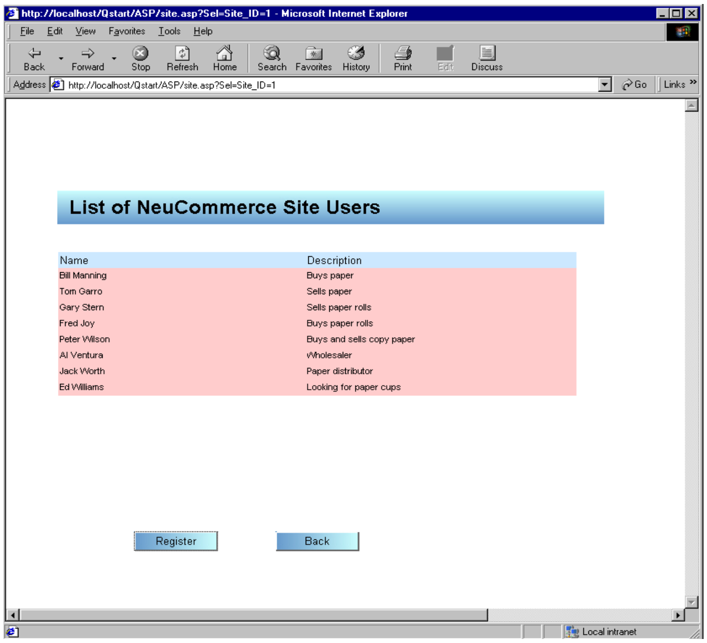
Click on the Web Page Preview icon to view the Index page in your browser.

Note: You will not be able to test your Buttons for Login or Catalog until those pages are created.



Site Page

Here you will be creating a page that will be used to list Site users. Our design for this page will be simple yet functional. When you have finished your pages it will look similar to the example below:



http://localhost/Qstart/ASP/site.asp?Sel=Site_ID=1 - Microsoft Internet Explorer

http://localhost/Qstart/ASP/site.asp?Sel=Site_ID=1

List of NeuCommerce Site Users

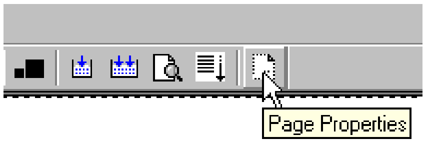
Name	Description
Bill Manning	Buys paper
Tom Garro	Sells paper
Gary Stern	Sells paper rolls
Fred Joy	Buys paper rolls
Peter Wilson	Buys and sells copy paper
Al Ventura	Wholesaler
Jack Worth	Paper distributor
Ed Williams	Looking for paper cups

Page Properties

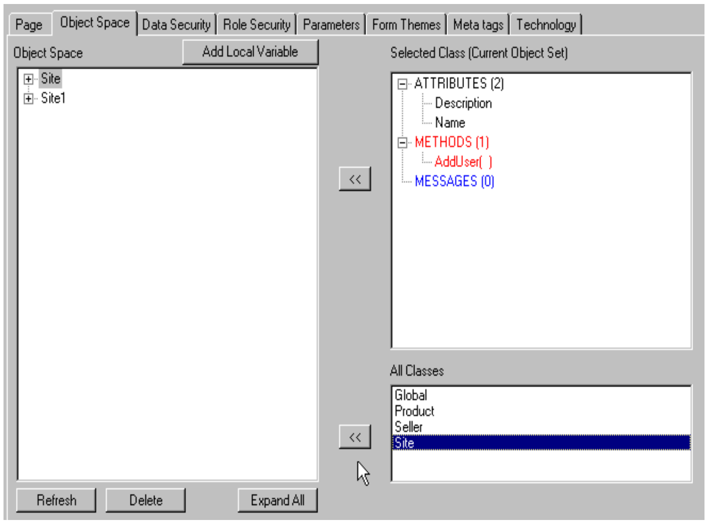
Select the Site page in the Application Navigator.

▼ To design your Site Page in PageArchitect

1. Click on the **Page Properties** icon.



2. Click on the **Object Space** tab.
3. Under **All Classes**, highlight **Site**.
4. Click the **<<** button to add Site1 to the Object Space.
5. Click **OK**.

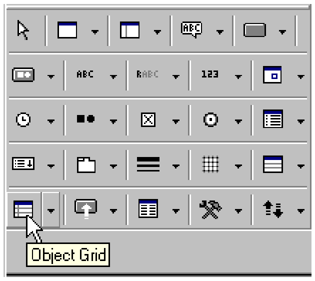


Object Grid

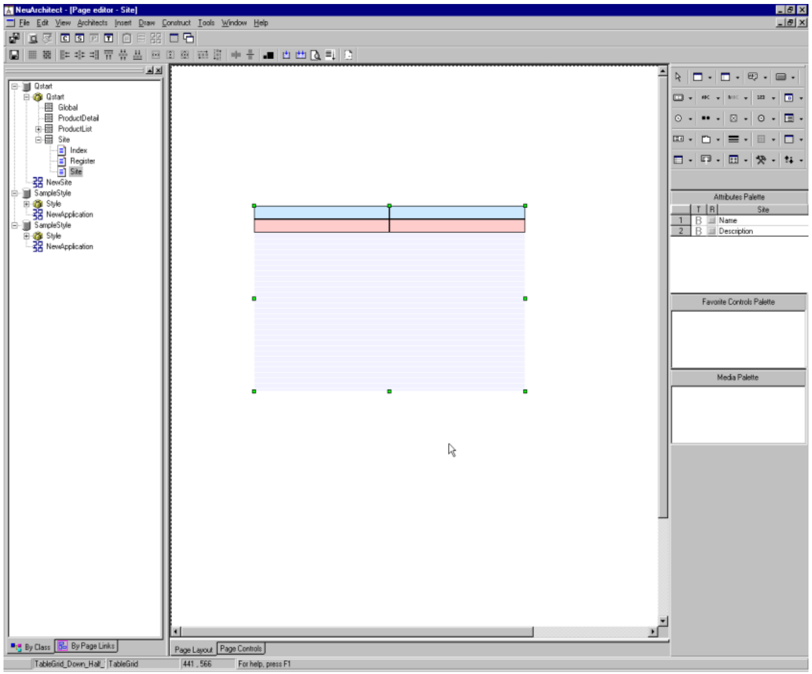
An Object Grid will be used to display site users in row format on the screen.

▼ To create an object grid

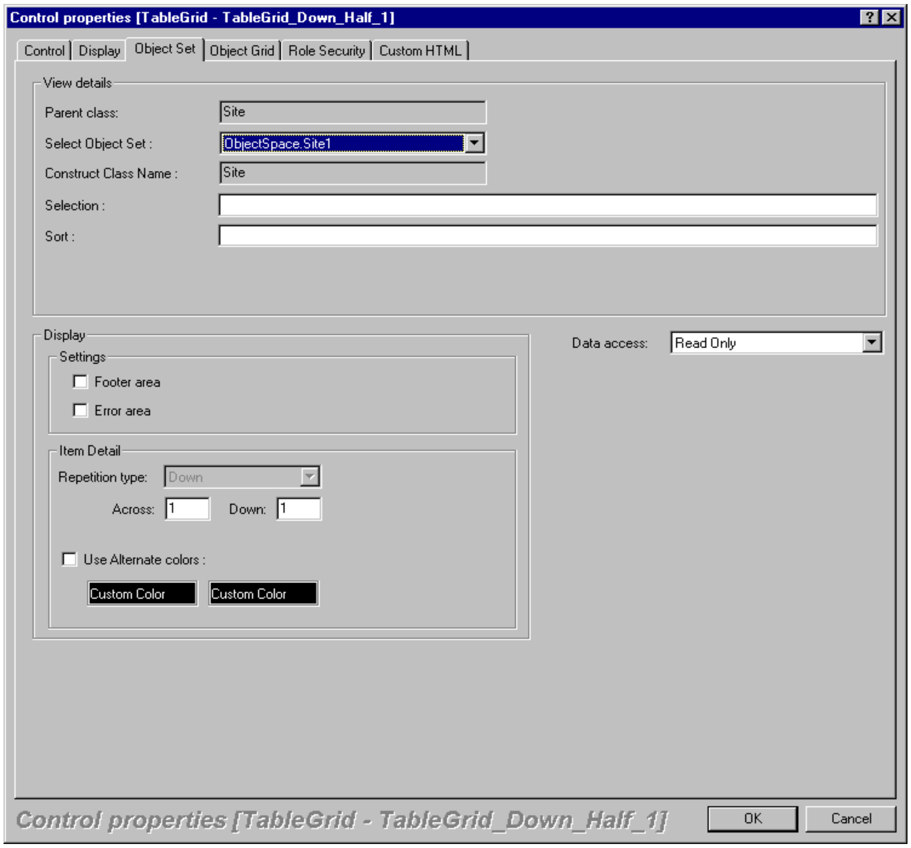
1. On the Control Palette, select **Object Grid**.



2. Drop and size it on the blank page similar to the following example.



3. Double-click on the Object Grid.
4. Click on the **Object Set** tab.
5. Be sure **Data access** is set to **Read Only**.
6. Using the drop-down list box choose **ObjectSpace.Site1** for **Select Object Set**.
7. Click OK.



Control properties [TableGrid - TableGrid_Down_Half_1]

View details

Parent class: Site

Select Object Set : ObjectSpace.Site1

Construct Class Name : Site

Selection :

Sort :

Display

Settings

□ Footer area

□ Error area

Item Detail

Repetition type: Down

Across: 1 Down: 1

□ Use Alternate colors :

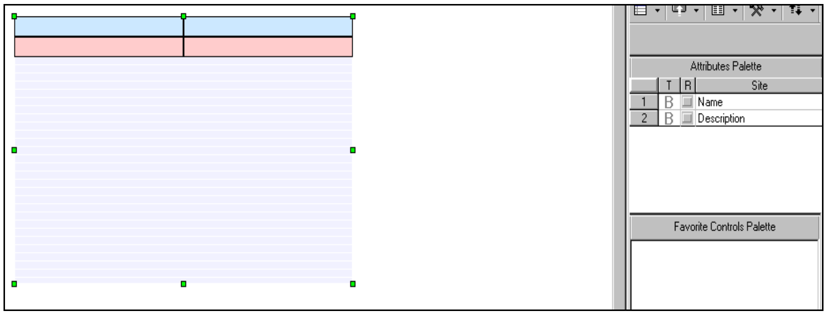
Custom Color Custom Color

Data access: Read Only

8. To refresh the screen, click your cursor somewhere in the blank area of the page to de-highlight the Object Grid.

9. Re-select the Object Grid by clicking on the bottom half of the grid.

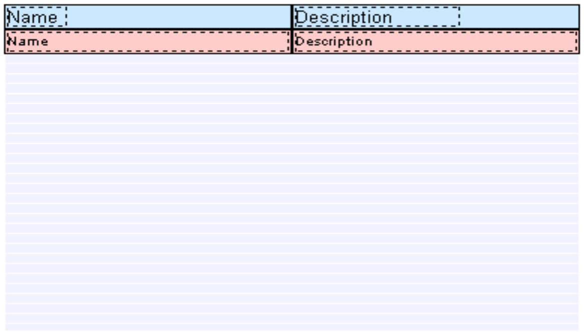
10. The Object Grid will be highlighted again with a list of Attributes displayed on the right of the screen.



Attributes Palette

	T	R	Site
1	B	□	Name
2	B	□	Description

11. With the Object Grid still highlighted, click the check boxes next to the Attributes for **Name** and **Description**, respectively. Your Object Grid should appear as follows.



Labels

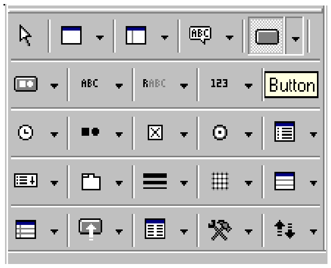
Add a Header1 label that indicates you are displaying a list of users (List of NeuCommerce Site Users)

Buttons

You will need two Buttons on this page.

▼ To create a Button

1. Select the **Button** control from the Control Palette.



2. Draw it on the page below the Object Grid.
3. Double-clickon the button.
4. Type **Register** in the **Name** field on the **Control tab**.
5. Using the dropdown, set **OnClick** action, **Action type** to **Document Load**.
6. Type the page you want to load – **Register**.
7. On the Display tab, check **Construct image**.

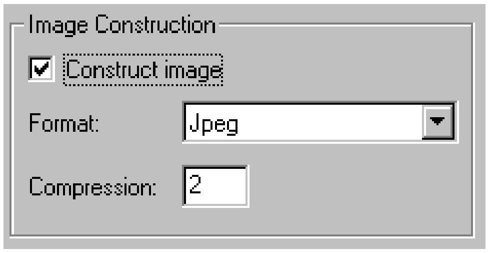


Image Construction

☑ Construct image

Format: Jpeg

Compression: 2

8. Format the other display elements (font size, centering) to taste.

9. Click **OK**.
10. Follow steps 1-9 above to create a second Button according to the following specs:
    - Button name: **Back**
    - **OnClick Action** is a **Document Load**, **Page name** is **Index**.
    - Size and center text as desired.
    - Be sure to check **Construct image**.

Save your work.

Construct the Users Page

Refer to Index Page if you need to refresh on the procedure for constructing a page.

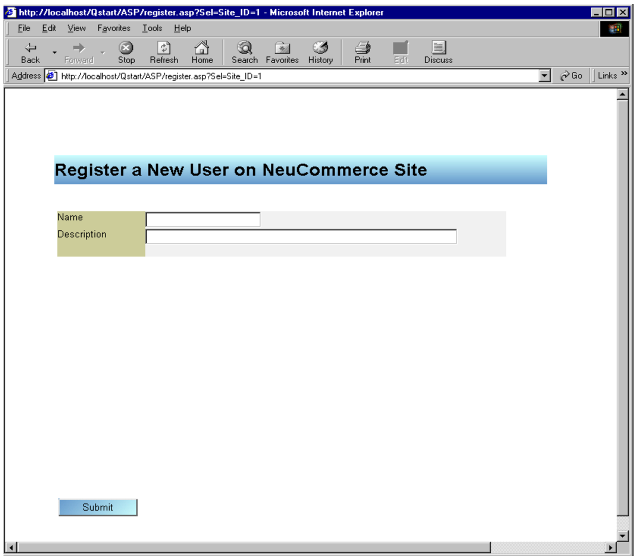
Preview the Users Page

Note: if you are using EJB/WebLogic refer to Appendix A, Starting and Stopping the WebLogic Server

Refer to Index Page if you need to refresh on the procedure for previewing a page. This page can be previewed separately. You may also want to preview the Index page again and test your Login button. The Login button should take you to this page. From there, you can test the Back button as well.

Register Page

This page will be used to add users to Site Class. Our design for this pages will be simple yet functional. When you have finished your page it will look similar to the example below:


http://localhost/Qstart/ASP/register.asp?Sel=Site_ID=1 - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Forward
Stop
Refresh
Home
Search
Favorites
History
Print
Edit
Discuss
Address
http://localhost/Qstart/ASP/register.asp?Sel=Site_ID=1
Go
Links
Register a New User on NeuCommerce Site
Name
Description
Submit

Select the Register page in the Application Navigator and follow the Steps below to design your Register Page in PageArchitect.

Page Properties

Refer to Site Page if you need to refresh on the procedure for accessing page properties.

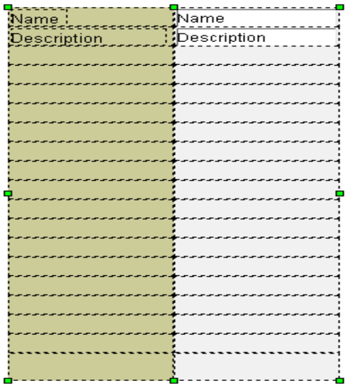
Check the Register Page Properties. The following should be set on the **Page** tab:

- Data access type: **Enter Only**
- **Post to the same form for HTTP posting** should be checked.

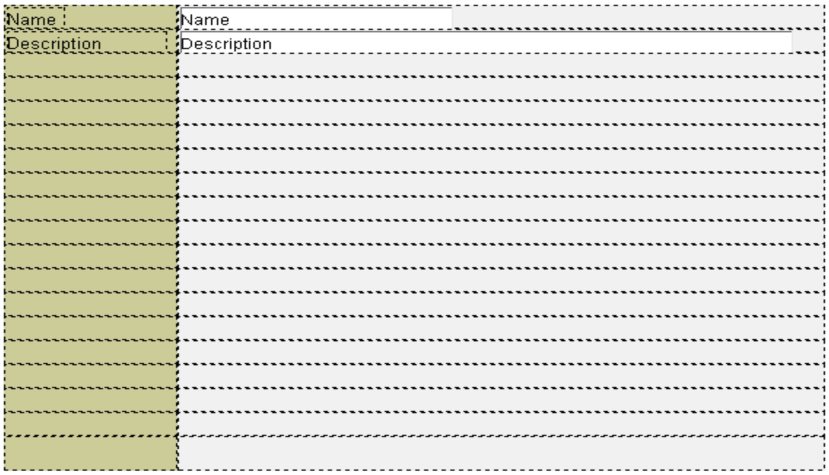
Grid

▼ To add a grid

1. From the Control Palette, select a Grid (not an Object Grid) and drop it on the blank page.
2. Click the check boxes next to the Attributes for **Name** and **Description**, respectively. Your Grid should appear as follows:



3. Using your mouse, you can resize both the label and data areas for a better fit. Select the entire Grid to resize and move columns. Click on the field to resize the individual item.



Label

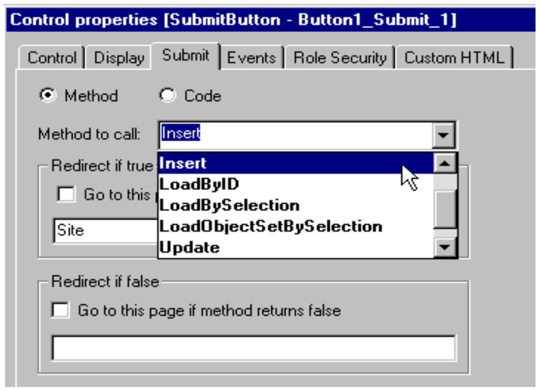
Add a Header1 label that indicates you are displaying an option to add a user to the list of users (Register as a NeuCommerce Site User).

SubmitButton

The SubmitButton will be used to Insert a row into the Site table.

▼ To add a submit button

1. From the Control Palette, select a **SubmitButton** and drop it on the page.
2. SubmitButton name - **Submit**
3. Right click on the **SubmitButton** and select **properties**.
4. Click on the **Submit** tab.
5. Click **Method to call**.
6. From the drop-down list, select **Insert**.
7. Specify the Site page in the Go to this page if method is true dialog box.


Control properties [SubmitButton - Button1_Submit_1]

● Method ○ Code

Method to call: Insert

Redirect if true

□ Go to this

Site

Redirect if false

□ Go to this page if method returns false

Save your work.

Construct the Register Page

Refer to Creating Page Content if you need to refresh on the procedure for constructing a page.

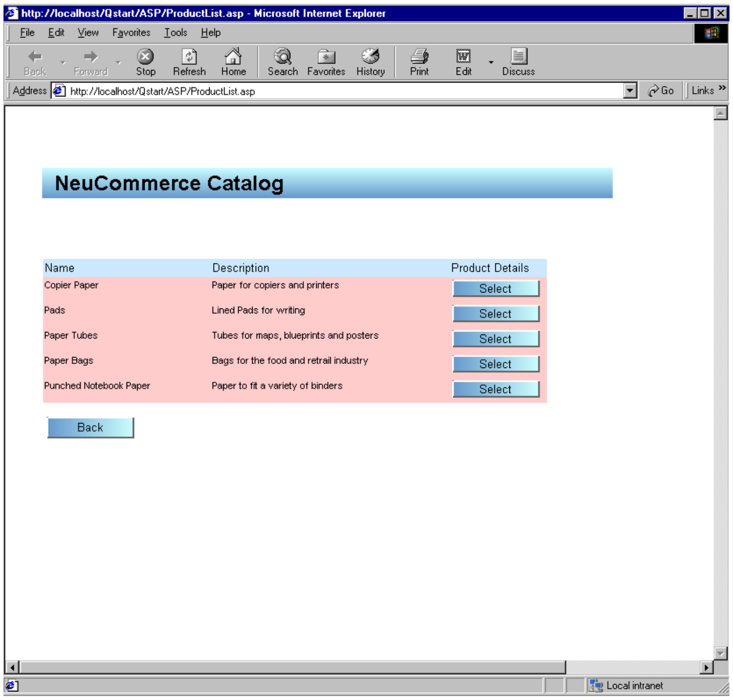
Preview the Register Page

Note: if you are using EJB/WebLogic refer to Appendix A, Starting and Stopping the WebLogic Server

Refer to Creating Page Content if you need to refresh on the procedure for previewing a page. You may also preview this page from the Index and/or Site pages.

ProductList Page

Our design for the ProductList page will be simple yet functional. When you have finished your ProductList page it will look similar to the example below:



NeuCommerce Catalog

Name	Description	Product Details
Copier Paper	Paper for copiers and printers	Select
Pads	Lined Pads for writing	Select
Paper Tubes	Tubes for maps, blueprints and posters	Select
Paper Bags	Bags for the food and retrail industry	Select
Punched Notebook Paper	Paper to fit a variety of binders	Select

Select the ProductList page in the Application Navigator and follow the Steps below to design your ProductList Page in PageArchitect.

Page Properties

▼ To set the ProductList page properties

1. On the **Page** tab, **Transaction Data access type** should be set to **Read Only**.
2. Click on the **Object Space** tab.
3. Under **All Classes**, highlight **ProductList**.
4. Click the **<<** button to add **ProductList1** to the Object Space.
5. Click **OK**.

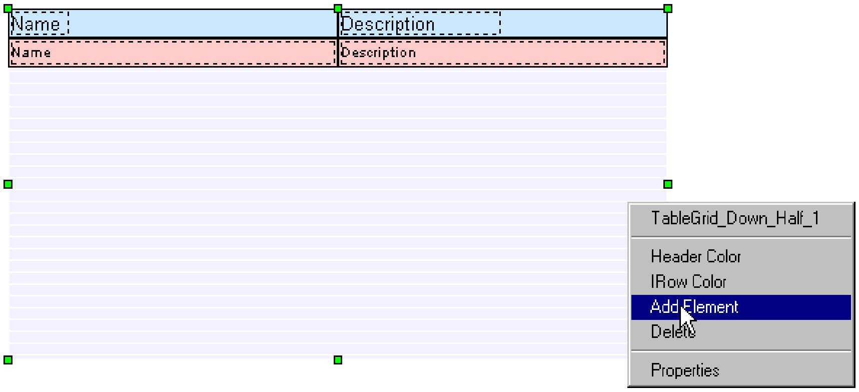
Object Grid

Refer back to the earlier Object Grid topic if you need to refresh on the procedure for creating this type of control. An Object Grid will be used to display sellers in row format on the screen.

▼ To create an object grid

1. On the Control Palette, select **Object Grid**.
2. Drop and size it on the blank page.
3. Double-clickon the Object Grid.
4. Click on the **Object Set** tab.
5. Be sure **Data access** is set to **Read Only**.
6. Using the drop-down list box choose **ObjectSpace.ProductList1** for **Select Object Set**.

7. Click **OK**.
8. Important, to refresh the screen, click your cursor somewhere in the blank area of the page to de-highlight the Object Grid.
9. Re-select the Object Grid by drawing an outbound box through and over it.
10. With the Object Grid still highlighted, click the check boxes next to the **Attributes for Name** and **Description**, respectively.
11. With the Object Grid still highlighted, right click on the grid and select **Add Element** to add an extra column to the table (will be used for a Button to access product details).



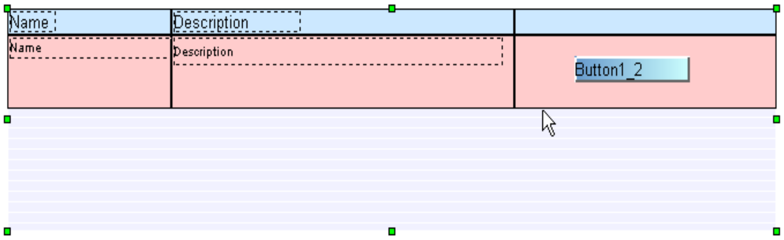
Buttons

Refer to Index Page if you need to refresh on the procedure for creating this page control. You will need two Buttons on this page.

▼ To create the first button

1. Button name – **Back**
2. **OnClick Action** is a **Document Load**, **Page name** is **Index**.
3. Size and center text as desired.
4. Be sure to check **Generate as image**.

The second button will need to be carefully placed in the Object Grid to ensure proper access to Product data. See the graphic below noting that the data row has been elongated and stretched to accommodate the button. In order to stretch the data row, the table must be actively selected.

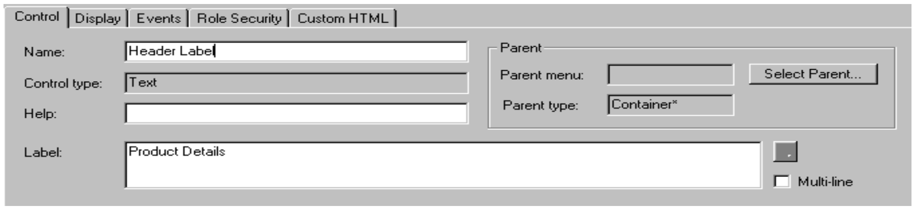


▼ To create the second button

1. Button name – **Select**
2. **OnClick Action** is a **Document Load**, **Page name** is **ProductDetail**.
3. Size and center text as desired.
4. Be sure to check **Generate as image**.
5. Position the button within the Row of the Object Grid.
6. Resize the row for a proportional look and feel.

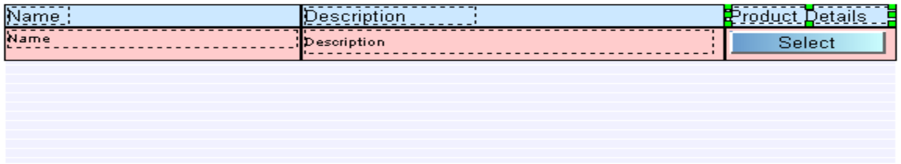
Labels

Add a standard Label over the new column in the table to indicate the column is for Product Details. Be sure that you key in your Label header in the Label field.



Name: Header Label
Control type: Text
Help:
Label: Product Details
Parent menu:
Parent type: Container*
□ Multi-line

Your finished label will appear as shown below:



Name	Description	Product Details
Name	Description	Select

Add a Header1 label that indicates you are displaying a list of Products (NeuCommerce Catalog).

Save your work.

Construct the ProductList Page

Refer to Creating Page Content if you need to refresh on the procedure for constructing a page.

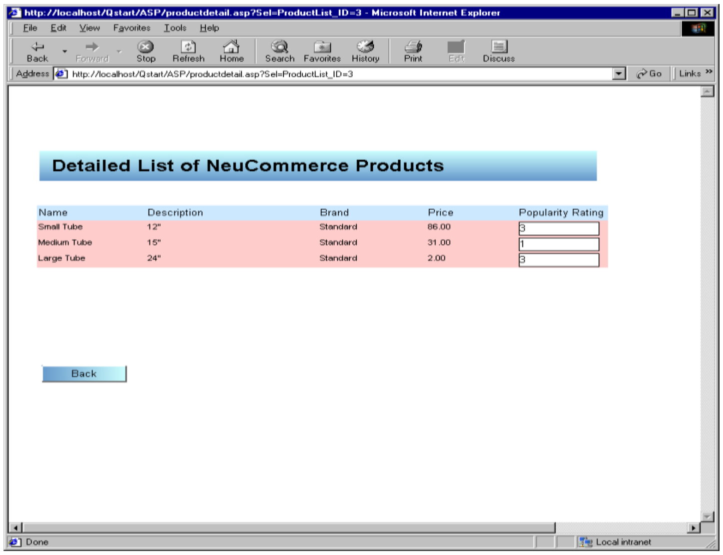
Preview the ProductList Page

Note: if you are using EJB/WebLogic refer to Appendix A, Starting and Stopping the WebLogic Server

Refer to Creating Page Content if you need to refresh on the procedure for previewing a page. You may also now preview this page from the Index page by selecting Catalog.

ProductDetail Page

Our design for the ProductDetail page will be simple yet functional. When you have finished your Products page it will look similar to the example below.



http://localhost/Qstart/ASP/productdetail.asp?Sel=ProductList_ID=3 - Microsoft Internet Explorer

Address: http://localhost/Qstart/ASP/productdetail.asp?Sel=ProductList_ID=3

Detailed List of NeuCommerce Products

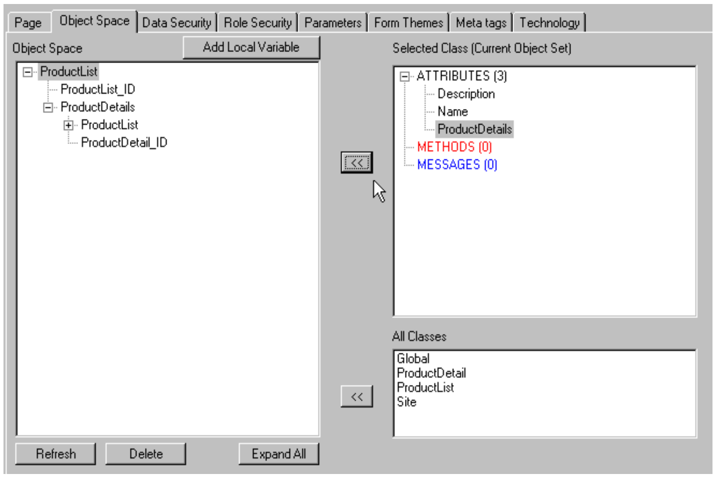
Name	Description	Brand	Price	Popularity Rating
Small Tube	12"	Standard	86.00	3
Medium Tube	15"	Standard	31.00	1
Large Tube	24"	Standard	2.00	3

Select the ProductDetail page in the Application Navigator and follow the Steps below to design your Product Detail Page in PageArchitect.

Page Properties

▼ To set the ProductDetail page properties

1. On the **Page** tab, set **Transaction Data access type** to **Read Only**.
2. Click on the **Object Space** tab.
3. Under **Selected Classes**, highlight **ProductDetails**.
4. Click the **<<** button to add **ProductDetails** to the Object Space.



Object Space

```
ProductList
    ProductList_ID
    ProductDetails
        ProductList
        ProductDetail_ID
```

Selected Class (Current Object Set)

```
ATTRIBUTES (3)
    Description
    Name
    ProductDetails
METHODS (0)
MESSAGES (0)
```

All Classes

```
Global
ProductDetail
ProductList
Site
```

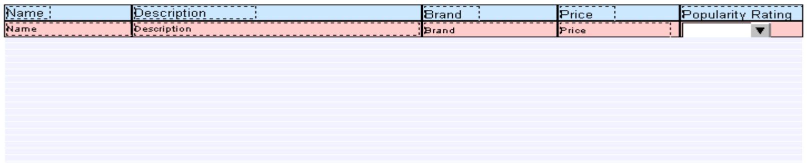
5. Click **OK**.

Object Grid

An Object Grid will be used to display products in row format on the screen.

▼ To create an object grid

1. On the Control Palette, select **Object Grid**.
2. Drop and size it on the blank page.
3. Right click on the Object Grid and select **Properties**.
4. Click on the **Object Set** tab.
5. Be sure to set **Data access** to **Read Only**.
6. Using the drop-down list box choose **ProductDetails** for **Select Object Set**.
7. Click **OK**.
8. Important, to refresh the screen, click your cursor somewhere in the blank area of the page to de-highlight the Object Grid.
9. Re-select the Object Grid by drawing an out bound box through and over it.
10. With the Object Grid still highlighted, click the check boxes next to the Product Attributes for **Name**, **Description**, **Brand**, **Price** and **PopularityRating**.
11. Resize the Grid and balance the column widths as shown below. This will require some additional manipulation of both the data fields and labels to ensure a clean fit in your browser.



Name	Description	Brand	Price	Popularity Rating
Name	Description	Brand	Price	▼

Labels

Add a header label that indicates you are displaying a list of Products (Detailed List of NeuCommerce Products).

Button

You will need one Button on this page.

▼ To create the button

1. Button name – **Back**
2. **OnClick** Action is a **Document Load**, **Page name** is **ProductList**.
3. Size and center text as desired.
4. Be sure to check **Generate as image**.

Save your work.

Construct the Products Page

Refer to Creating Page Content if you need to refresh on the procedure for constructing a page.

Preview the Products Page

Note: if you are using EJB/WebLogic refer to Appendix A, Starting and Stopping the WebLogic Server

Test this page by previewing the Index page, selecting Catalog and using the Select button to access product details. Try more than one from the Catalog (ProductList).

Testing the Quick Start

Close down NeuArchitect and test your application outside of the development environment.

▼ To test your application

1. Open your browser.
2. Supply your Quick Start URL - `http://localhost/Qstart/ASP/index.asp`.
3. Test all the functionality you have built into this website:
    a. Display Site user list via the Login button.
    b. Test registration.
    c. Display the ProductList via the Catalog button.
    d. Check ProductDetails for each of the Catalog items by using the Select button for each item.

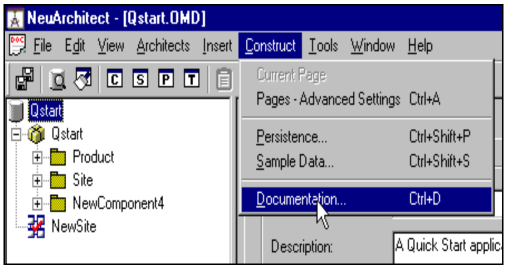
6 Creating NeuArchitect Documentation

Overview

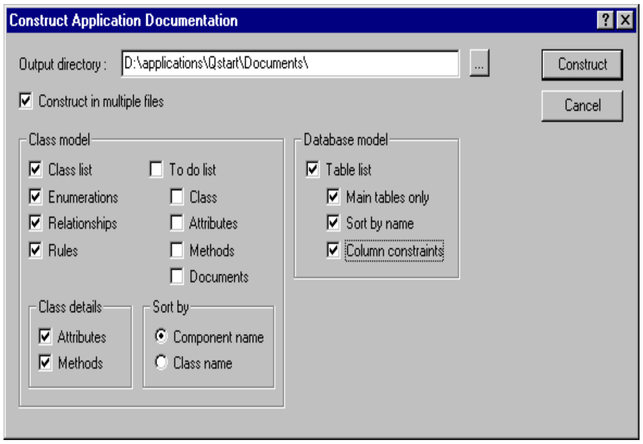
NeuArchitect can construct HTML-based system documentation, making it as painless as possible to capture the application development semantics and technical nuances necessary for application extension/enhancements, and ease-of-maintenance.

▼ To create system documentation

1. Select **Documentation** from the **Construct** menu.


NeuArchitect - [Qstart.OMD]
File Edit View Architects Insert Construct Tools Window Help
Current Page
Pages - Advanced Settings Ctrl+A
Persistence... Ctrl+Shift+P
Sample Data... Ctrl+Shift+S
Documentation... Ctrl+D
Qstart
Qstart
Product
Site
NewComponent4
NewSite
Description:
A Quick Start applic

2. Select outputs shown below and click **Construct**.



Output directory : D:\applications\Qstart\Documents\

☑ Construct in multiple files

Class model

☑ Class list
☑ Enumerations
☑ Relationships
☑ Rules
□ To do list
□ Class
□ Attributes
□ Methods
□ Documents

Class details

☑ Attributes
☑ Methods

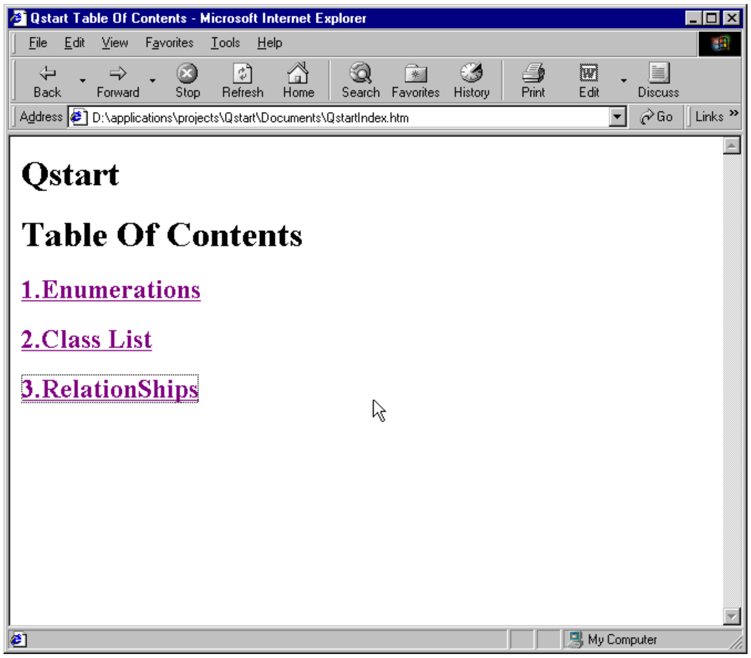
Sort by

● Component name
○ Class name

Database model

☑ Table list
☑ Main tables only
☑ Sort by name
☑ Column constraints

5. Use your browser to access the `QStartIndex` file created in the Documents subdirectory of your application directory.



Qstart Table Of Contents - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Forward
Stop
Refresh
Home
Search
Favorites
History
Print
Edit
Discuss
Address
D:\applications\projects\Qstart\Documents\QstartIndex.htm
Go
Links
My Computer

Qstart

Table Of Contents

1.Enumerations

2.Class List

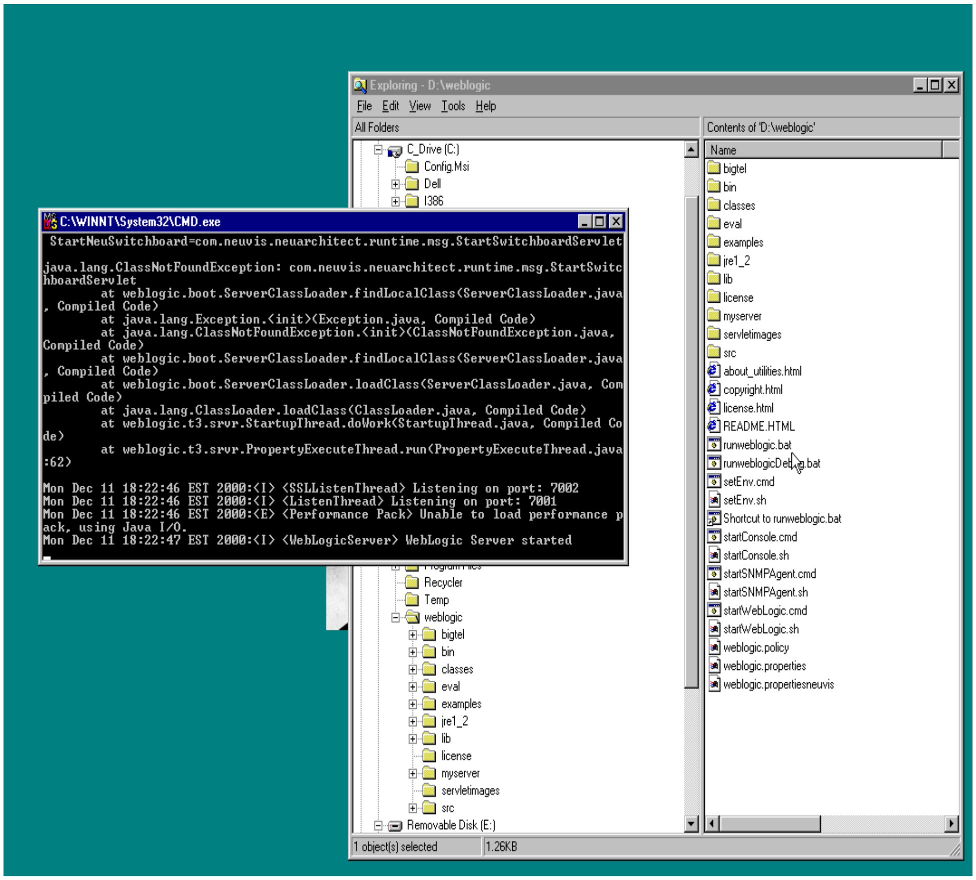
3.RelationShips

7 Appendix A Starting and Stopping the WebLogic Server

Starting the WebLogic Server

Follow the procedure below to start the WebLogic Server

1. Open your **weblogic** folder in Explorer
2. Run the file **runweblogic.bat** by double clicking on it
3. You should see the message ‘WebLogic Server Started’ in the bottom of the command window



```
 StartNeuSwitchboard=com.neuvis.neuarchitect.runtime.msg.StartSwitchboardServlet

java.lang.ClassNotFoundException: com.neuvis.neuarchitect.runtime.msg.StartSwitc
hboardServlet
        at weblogic.boot.ServerClassLoader.findLocalClass(ServerClassLoader.java
, Compiled Code)
        at java.lang.Exception.<init>(Exception.java, Compiled Code)
        at java.lang.ClassNotFoundException.<init>(ClassNotFoundException.java,
Compiled Code)
        at weblogic.boot.ServerClassLoader.findLocalClass(ServerClassLoader.java
, Compiled Code)
        at weblogic.boot.ServerClassLoader.loadClass(ServerClassLoader.java, Com
piled Code)
        at java.lang.ClassLoader.loadClass(ClassLoader.java, Compiled Code)
        at weblogic.t3.srvr.StartupThread.doWork(StartupThread.java, Compiled Co
de)
        at weblogic.t3.srvr.PropertyExecuteThread.run(PropertyExecuteThread.java
:62)

Mon Dec 11 18:22:46 EST 2000:<I> <SSLListenThread> Listening on port: 7002
Mon Dec 11 18:22:46 EST 2000:<I> <ListenThread> Listening on port: 7001
Mon Dec 11 18:22:46 EST 2000:<E> <Performance Pack> Unable to load performance p
ack, using Java I/O.
Mon Dec 11 18:22:47 EST 2000:<I> <WebLogicServer> WebLogic Server started
```

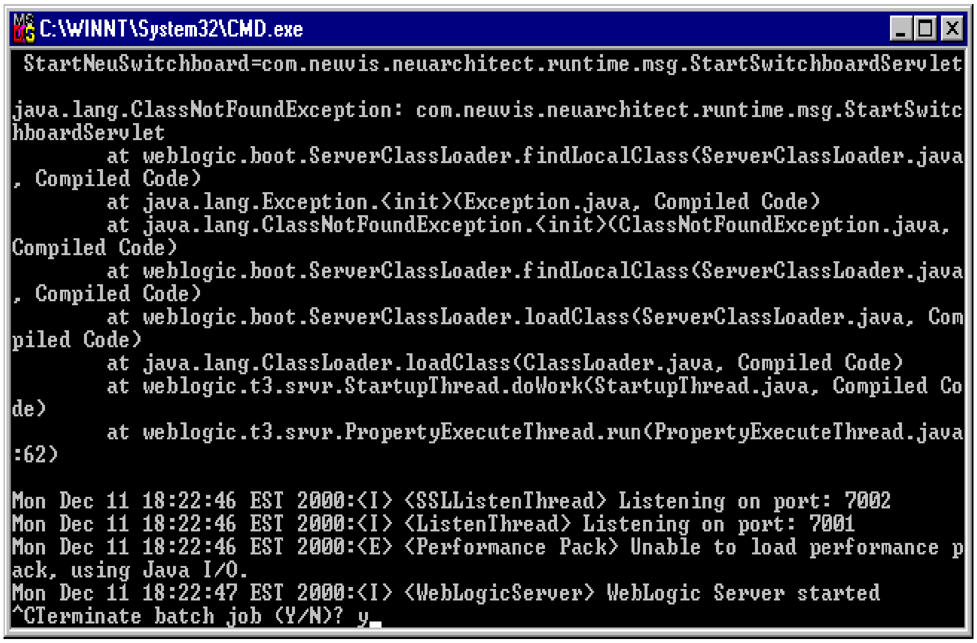
Stopping the WebLogic Server

Follow the procedure below to stop the WebLogic Server

Type **CTRL/C** in the command window

Type a **y** in answer to the question ‘Terminate batch job (Y/N)?

Press **Enter**



```
 StartNeuSwitchboard=com.neuvis.neuarchitect.runtime.msg.StartSwitchboardServlet

java.lang.ClassNotFoundException: com.neuvis.neuarchitect.runtime.msg.StartSwitc
hboardServlet
        at weblogic.boot.ServerClassLoader.findLocalClass(ServerClassLoader.java
, Compiled Code)
        at java.lang.Exception.<init>(Exception.java, Compiled Code)
        at java.lang.ClassNotFoundException.<init>(ClassNotFoundException.java,
Compiled Code)
        at weblogic.boot.ServerClassLoader.findLocalClass(ServerClassLoader.java
, Compiled Code)
        at weblogic.boot.ServerClassLoader.loadClass(ServerClassLoader.java, Com
piled Code)
        at java.lang.ClassLoader.loadClass(ClassLoader.java, Compiled Code)
        at weblogic.t3.srvr.StartupThread.doWork(StartupThread.java, Compiled Co
de)
        at weblogic.t3.srvr.PropertyExecuteThread.run(PropertyExecuteThread.java
:62)

Mon Dec 11 18:22:46 EST 2000:<I> <SSLListenThread> Listening on port: 7002
Mon Dec 11 18:22:46 EST 2000:<I> <ListenThread> Listening on port: 7001
Mon Dec 11 18:22:46 EST 2000:<E> <Performance Pack> Unable to load performance p
ack, using Java I/O.
Mon Dec 11 18:22:47 EST 2000:<I> <WebLogicServer> WebLogic Server started
^CTerminate batch job (Y/N)? y_
```