Guide to PortalProdigy THTML and Page Management for Web Designers

Created November 6, 2006
Last Modified March 28, 2007

<u>Overview:</u>

This guide provides an overview of the PortalProdigy page design process. It will introduce you to THTML and explain how to use THTML to create custom page and component designs for PortalProdigy websites. This guide is intended for Web Designers and it assumes that the reader has a thorough understanding of HTML, CSS, and Java Script.

What is THTML?

THTML is an acronym for Template Hyper Text Markup Language.

THTML is an extension of the HTML markup language and is used to create template styles for PortalProdigy. Template styles control the visual design of the website. THTML provides tags for inserting content in the template styles and describing how it should be displayed. THTML gives you access to PortalProdigy's forms, fields, controls, and CSS classes.

THTML tags follow the same conventions as HTML tags. The PortalProdigy Server dynamically replaces THTML tags with HTML formatted content to deliver W3C compliant HTML to the user's browser.

Using THTML you can create new designs from scratch as well as create modified versions of the designs that are included with PortalProdigy. You can also use THTML to duplicate the design of an existing website for conversion to PortalProdigy.

The best way to learn THML is to study the template designs that are included with PortalProdigy. PortalProdigy calls its template designs "Styles". The Styles that are included (shipped) with PortalProdigy are classified as Standard Styles. When you create a style it gets classified as a Custom Style. The code for each of the Standard Styles is accessible from within PortalProdigy so you can view, study, and modify it. The code can be viewed directly from within PortalProdigy as well as downloaded to your computer. This guide also provides a number of examples to assist you in learning THTML.

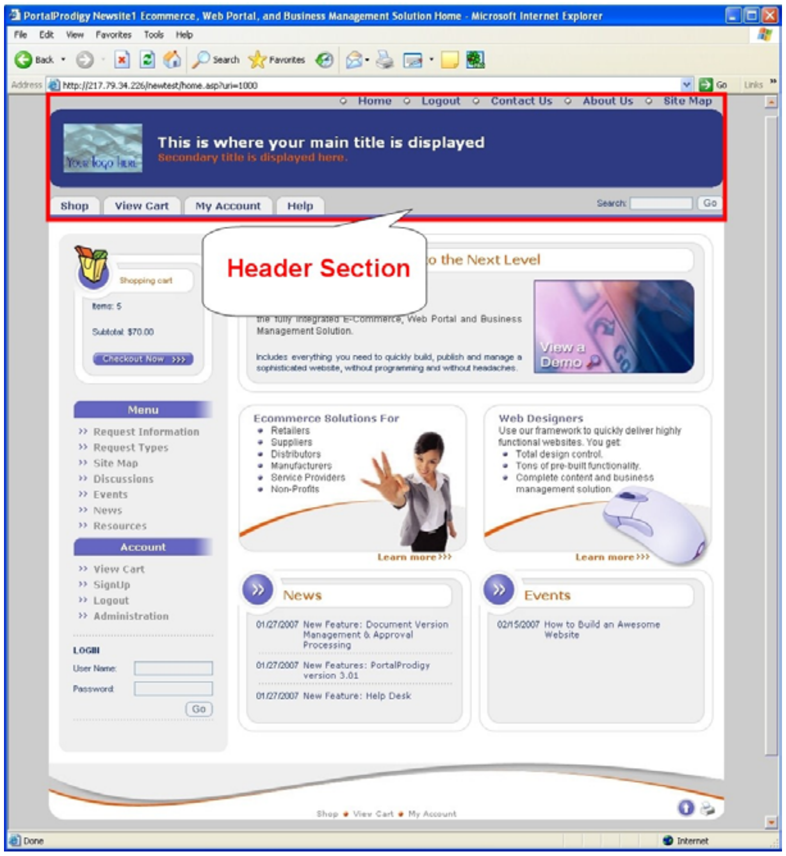
Page Construction

The basic architecture of a PortalProdigy page is as follows.

The template for a page design is called a Page Style. Page Styles define the layout and general design of a page. Page Styles provide a framework and serve as containers for dynamically inserted objects.

A Page Style is divided into sections:

- Header
- Left Side Bar
- Right Side Bar
- Main Content
- Footer


PortalProdigy Newsite1 Ecommerce, Web Portal, and Business Management Solution Home - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://217.79.34.226/newtest/home.asp?uri=1000
Go Links
Home Logout Contact Us About Us Site Map
Your logo here
This is where your main title is displayed
Secondary title is displayed here.
Shop View Cart My Account Help
Search: Go
Header Section
Shopping cart
Items: 5
Subtotal: $70.00
Checkout Now >>>
o the Next Level
View a Demo
Menu
Request Information
Request Types
Site Map
Discussions
Events
News
Resources
Account
View Cart
SignUp
Logout
Administration
Login
User Name:
Password:
Go
Learn more >>>
Learn more >>>
News
Events
Shop View Cart My Account
Done
Internet

the fully integrated E-Commerce, Web Portal and Business Management Solution.

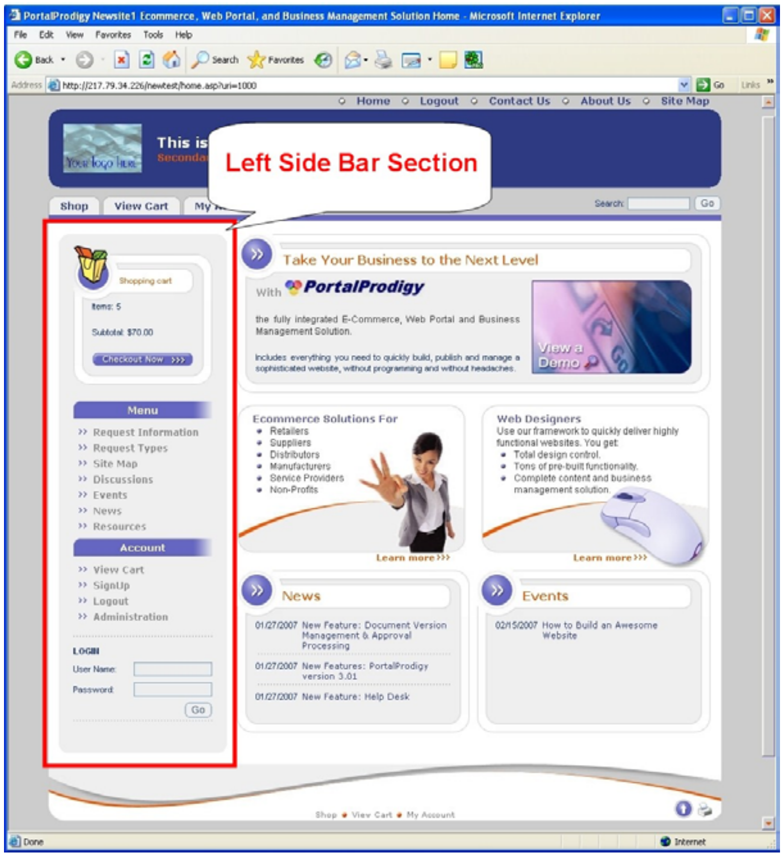
Includes everything you need to quickly build, publish and manage a sophisticated website, without programming and without headaches.

Ecommerce Solutions For
- Retailers
- Suppliers
- Distributors
- Manufacturers
- Service Providers
- Non-Profits

Web Designers
Use our framework to quickly deliver highly functional websites. You get:
- Total design control.
- Tons of pre-built functionality.
- Complete content and business management solution.

01/27/2007 New Feature: Document Version Management & Approval Processing
01/27/2007 New Features: PortalProdigy version 3.01
01/27/2007 New Feature: Help Desk

02/15/2007 How to Build an Awesome Website



Take Your Business to the Next Level

With PortalProdigy

the fully integrated E-Commerce, Web Portal and Business Management Solution.

Includes everything you need to quickly build, publish and manage a sophisticated website, without programming and without headaches.

Ecommerce Solutions For
- Retailers
- Suppliers
- Distributors
- Manufacturers
- Service Providers
- Non-Profits

Web Designers
Use our framework to quickly deliver highly functional websites. You get:
- Total design control.
- Tons of pre-built functionality.
- Complete content and business management solution.

News

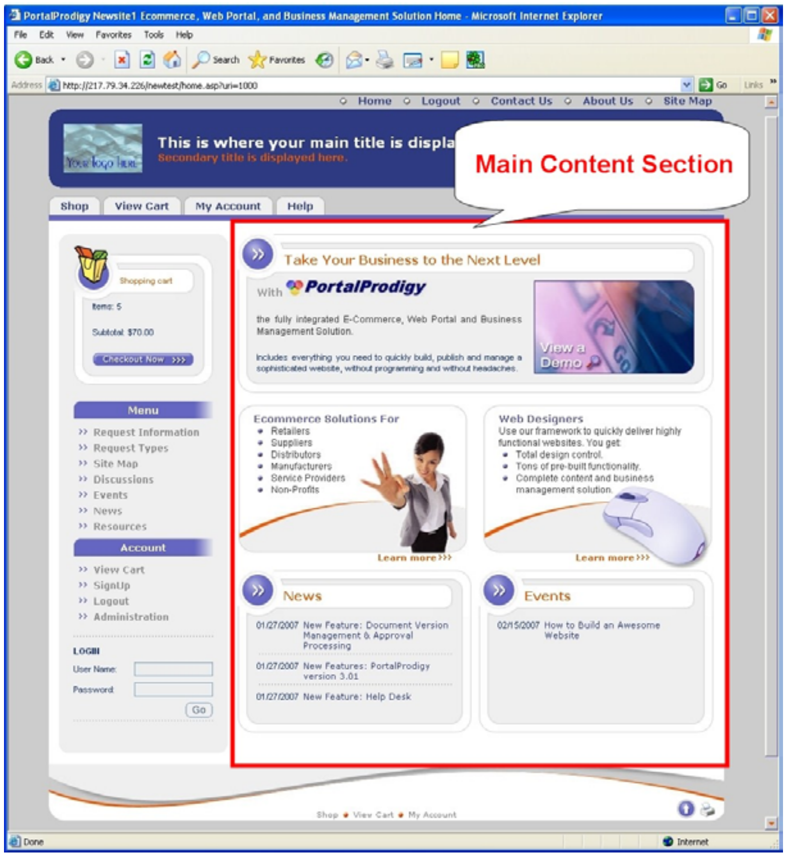
01/27/2007 New Feature: Document Version Management & Approval Processing

01/27/2007 New Features: PortalProdigy version 3.01

01/27/2007 New Feature: Help Desk

Events

02/15/2007 How to Build an Awesome Website


PortalProdigy Newsite1 Ecommerce, Web Portal, and Business Management Solution Home - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://217.79.34.226/newtest/home.asp?uri=1000
Go Links
Home Logout Contact Us About Us Site Map
Your logo here
Main Content Section
Shop View Cart My Account Help
Shopping cart
Checkout Now >>>
Menu
Request Information
Request Types
Site Map
Discussions
Events
News
Resources
Account
View Cart
SignUp
Logout
Administration
LOGIN
User Name:
Password:
Go
View a Demo
Learn more >>>
Learn more >>>
Shop View Cart My Account
Done
Internet

This is where your main title is displa

Secondary title is displayed here.

Items: 5

Subtotal: $70.00

Take Your Business to the Next Level

With PortalProdigy

the fully integrated E-Commerce, Web Portal and Business Management Solution.

Includes everything you need to quickly build, publish and manage a sophisticated website, without programming and without headaches.

Ecommerce Solutions For
- Retailers
- Suppliers
- Distributors
- Manufacturers
- Service Providers
- Non-Profits

Web Designers
Use our framework to quickly deliver highly functional websites. You get:
- Total design control.
- Tons of pre-built functionality.
- Complete content and business management solution.

News

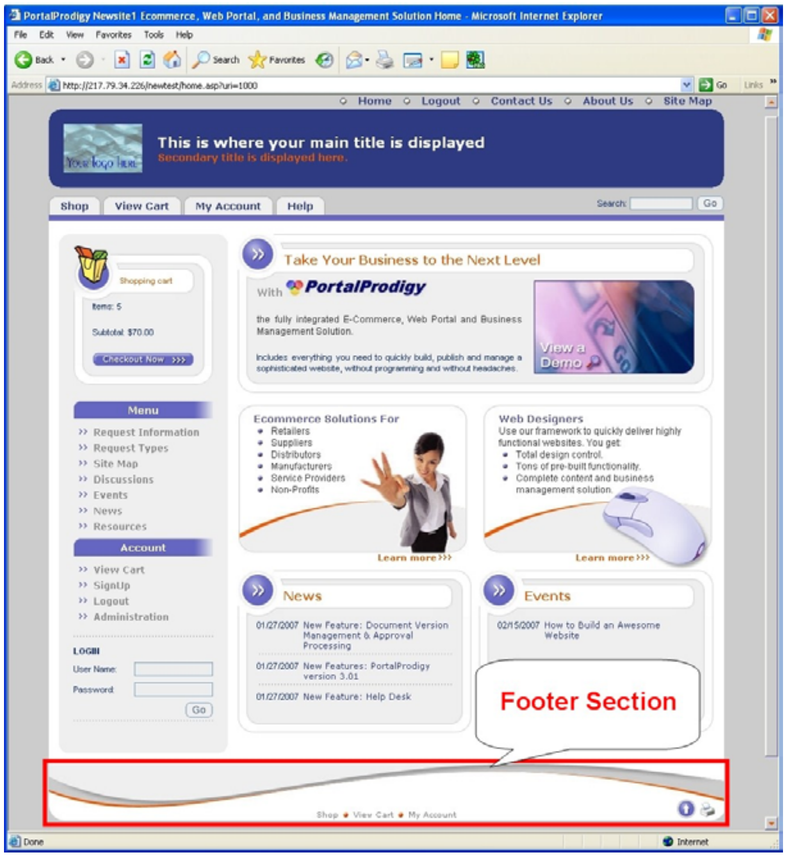
01/27/2007 New Feature: Document Version Management & Approval Processing

01/27/2007 New Features: PortalProdigy version 3.01

01/27/2007 New Feature: Help Desk

Events

02/15/2007 How to Build an Awesome Website

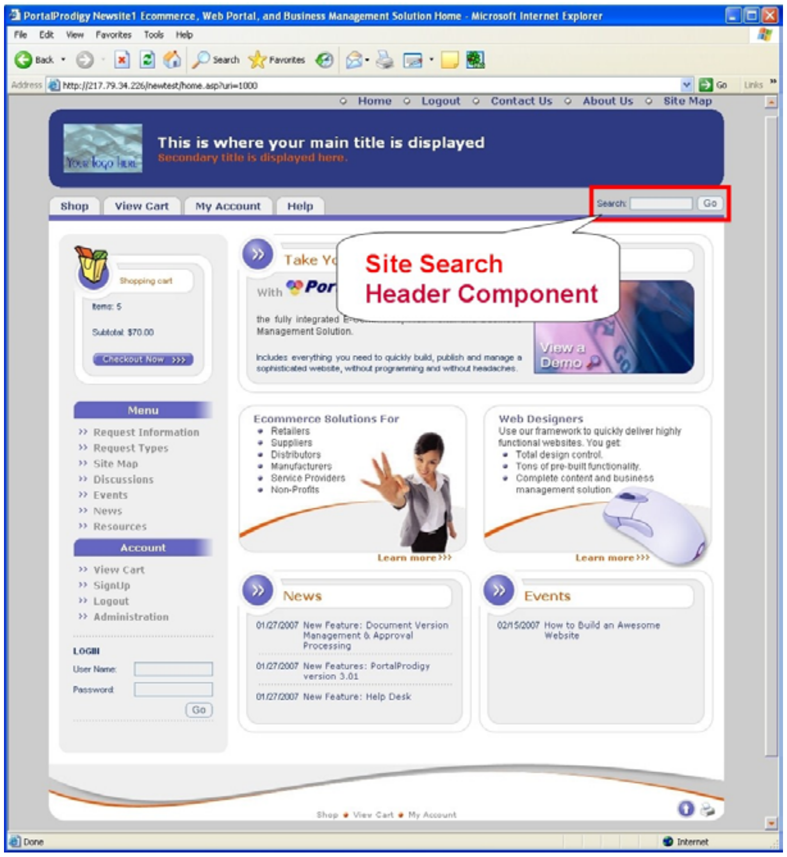


Sections serve as containers for components. Components are predefined objects that provide data and/or functionality for some specific purpose or feature.

Some examples of components that are available for inclusion in the Header Section are:

- Logo
- Organization Title
- Login form
- Site Search
- Shopping Cart Summary
- Menu Bar

The picture below highlights a Site Search Component located in menu bar of the Header Section:



This is where your main title is displayed

Secondary title is displayed here.

Items: 5

Subtotal: $70.00

the fully integrated E-

Management Solution.

Includes everything you need to quickly build, publish and manage a sophisticated website, without programming and without headaches.

Ecommerce Solutions For
- Retailers
- Suppliers
- Distributors
- Manufacturers
- Service Providers
- Non-Profits

Web Designers
Use our framework to quickly deliver highly functional websites. You get:
- Total design control.
- Tons of pre-built functionality.
- Complete content and business management solution.

News

01/27/2007 New Feature: Document Version Management & Approval Processing

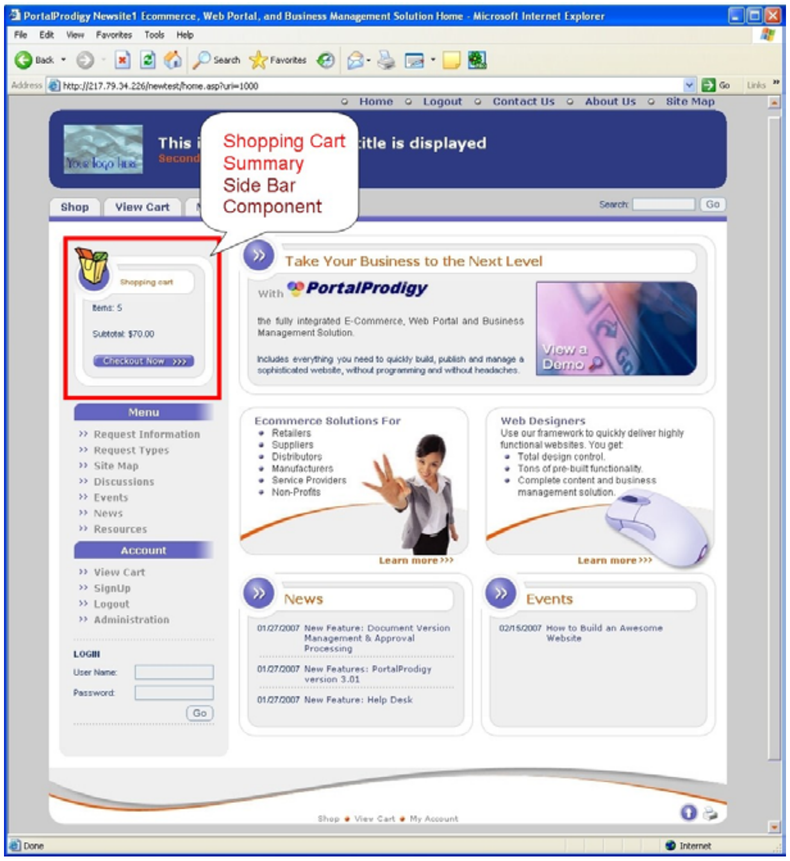
01/27/2007 New Features: PortalProdigy version 3.01

01/27/2007 New Feature: Help Desk

Events

02/15/2007 How to Build an Awesome Website

The picture below highlights a Shopping Cart Summary Component located in the Left Side Bar Section:


PortalProdigy Newsite1 Ecommerce, Web Portal, and Business Management Solution Home - Microsoft Internet Explorer
File Edit View Favorites Tools Help
http://217.79.34.226/newtest/home.asp?uri=1000
Home Logout Contact Us About Us Site Map
Shopping Cart Summary Side Bar Component
This i… title is displayed
Shop View Cart
Search: Go
Shopping cart
Items: 5
Subtotal: $70.00
Checkout Now >>>
Menu
Request Information
Request Types
Site Map
Discussions
Events
News
Resources
Account
View Cart
SignUp
Logout
Administration
Login
User Name:
Password:
Go
Take Your Business to the Next Level
With PortalProdigy
View a Demo
Learn more >>>
News
Events
Shop • View Cart • My Account

the fully integrated E-Commerce, Web Portal and Business Management Solution.

Includes everything you need to quickly build, publish and manage a sophisticated website, without programming and without headaches.

Ecommerce Solutions For
- Retailers
- Suppliers
- Distributors
- Manufacturers
- Service Providers
- Non-Profits

Web Designers
Use our framework to quickly deliver highly functional websites. You get:
- Total design control.
- Tons of pre-built functionality.
- Complete content and business management solution.

01/27/2007 New Feature: Document Version Management & Approval Processing
01/27/2007 New Features: PortalProdigy version 3.01
01/27/2007 New Feature: Help Desk

02/15/2007 How to Build an Awesome Website

THTML tags are used to define the location of each section along with its visual design. The visual design for how a section is displayed is called a Section Style. THTML tags are also used to define the visual design of components. The visual design for how a component is displayed is called a Component Style.

Components are managed by end users from within the PortalProdigy user interface. End users have the ability to select the components they want to include in each section on a page by page basis. When they select a component they select a style that defines the visual design for how the component is displayed. In addition, they can manage and select data that is displayed within a component.

Each component can have one or more styles. This will be explained in greater detail later on in this guide. What is important to understand is that you can define component styles within your page style, but the end user has ultimate control to include or exclude components and select the styles used to display them. PortalProdigy also provides the ability to create your own Components which are called Custom Unstructured Mini Browsers (MBs). Custom Unstructured MBs will be explained in greater detail later on within this guide.

When you create a style for PortalProdigy you have the option of creating a single HTML file that contains everything we have discussed thus far, including the Page Style, Section Styles and Component Styles. You also have the option to define your Section Styles and Component Styles independently of Page Styles. E.g. you may want to create a customized Component Style for the Shopping Cart Summary shown in the example above. You can create the Style for this one component and register it in PortalProdigy. You may want to create a different Header for use on Product Catalog pages. You can create the Style for just a Header and register it in PortalProdigy.

When you create Page Styles you have the option of defining Section Styles without defining the Components that will be inserted within them. You simply define where each of the Sections shall be inserted within the Page Style along with each section's general design. E.g. you might indicate where a Left Side Bar Section is to be inserted. When you define the Left Side Bar you will leave it empty. You will simply define its characteristics such as the width, border style, and independent graphical elements.

You also have the option to define the Component Styles internally within the Page Style. This second method allows you to design web pages in a manner that you are most likely accustomed to, where everything is included in a single HTML page that can be visualized as a single page using standard web design tools.

When a Page Style is registered in PortalProdigy, the PortalProdigy Page Style Processor extracts each Section Style and each Component Style from the page and separately registers each of them in the system. With the Section Styles and Component Styles extracted from the Page Style, the end user is then able to individually manage them, change them, and to re-use them in other Page Styles. Thus PortalProdigy allows end users to create their own combinations of Page Styles, Section Styles and Component Styles. This architecture gives the end user tremendous flexibility.

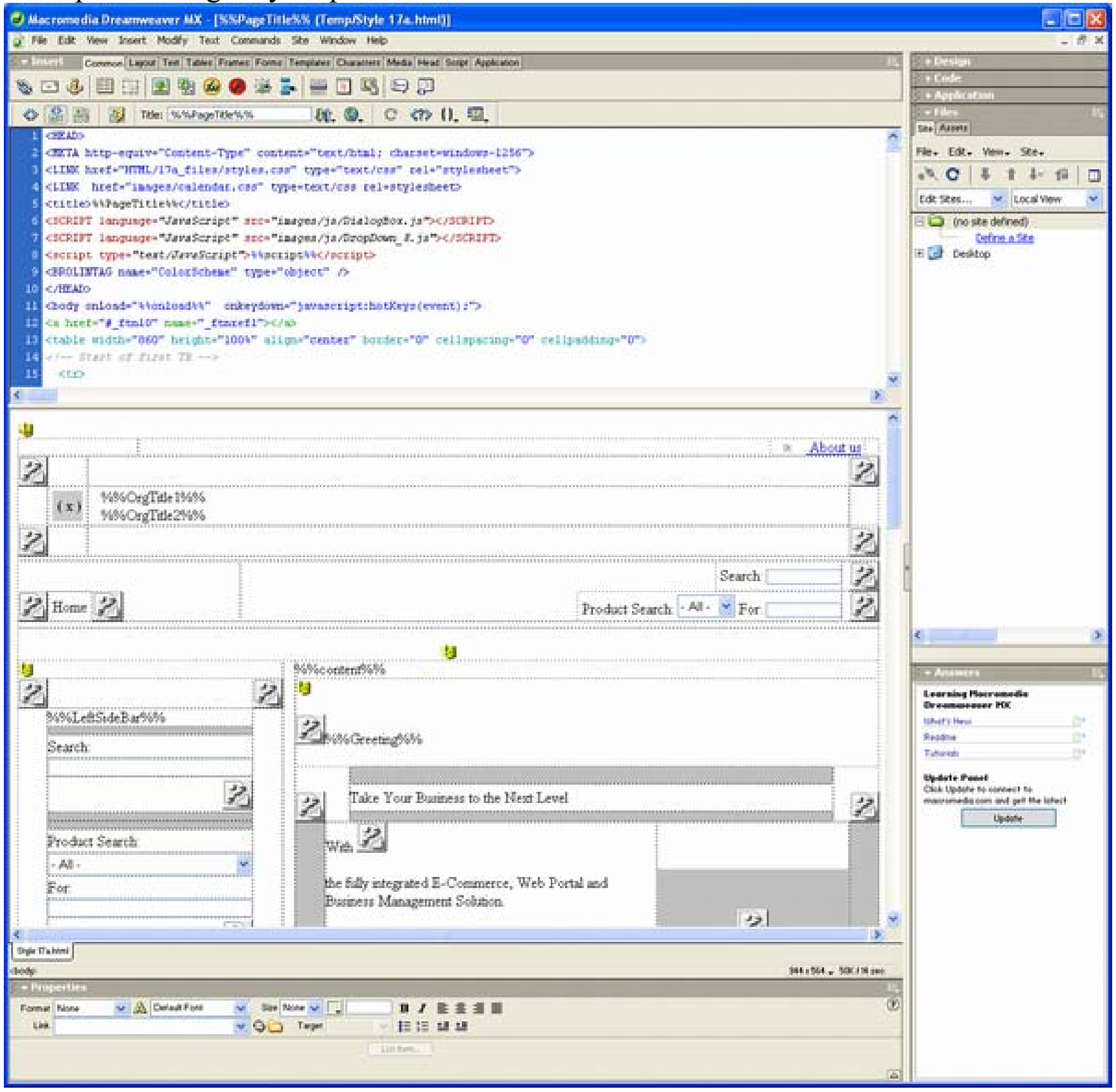
PortalProdigy also gives the end user the ability to select a unique Page Style for each Page Type. Examples of Page Types are Home Page, Product Catalog Category Page, Product Detail Page, Event Listing Page, etc. In some cases end users are given the ability to select a unique Page Style for specific records such as an event record, product record, etc. End users are also given the ability to independently select a Style for the Header Section, Main Content Section, Footer Section, and for each Component that is used within a Section.

For each page, the Page Style, Component Styles, and other parameters that define the page are referred to as "Page Settings". A tool named Page Manager is used to dynamically manage these Page Settings. Collectively the Page Settings for a single page are referred to as a "Page Settings Collection". Page Settings Collections can be saved and re-used by other pages. Each page has the option to inherit Page Settings from a Page Settings Collection along with the ability to override individual settings. Inheritance of Page Settings greatly minimizes the maintenance burden, allowing changes to cascade from one page to others.

Each Page Settings Collection allows you to define separate Component Collections for the page's Header, Left Side Bar, Main Content, Right Side Bar, and Footer. These Component Collections can be used by other Page Setting Collections. If it all seems a little overwhelming, just take a look at Page Manager and go through its options. Once you see it in action it should start to make sense. And once you get the hang of it, you will discover how truly flexible and how easy it is to use. We also explain this process in greater detail in the following sections.

You may create styles using any HTML Editor that can output W3C compliant HTML. Styles are a combination of HTML, THTML, CSS, Java Script, images, and other resources. Obviously your editor will not be able to interpret the THTML tags. The THTML tags will either be ignored by your editor or displayed as text.

Example of a Page Style opened in Dreamweaver MX:


Macromedia Dreamweaver MX - [%%PageTitle%% (Temp/Style 17a.html)]
File Edit View Insert Modify Text Commands Site Window Help
Insert Common Layout Text Tables Frames Forms Templates Characters Media Head Script Application
Title: %%PageTitle%%
Design
Code
Application
Files
Site Assets
File Edit View Site
Edit Sites... Local View
(no site defined)
Define a Site
Desktop
Answers
Learning Macromedia Dreamweaver MX
What's New
Readme
Tutorials
Update Panel
Click Update to connect to macromedia.com and get the latest
Update
Style 17a.html
Properties
Format None
Default Font
Size None
Link
Target
List Item...

```
<HEAD>
<META http-equiv="Content-Type" content="text/html; charset=windows-1256">
<LINK href="HTML/17a_files/styles.css" type="text/css" rel="stylesheet">
<LINK  href="images/calendar.css" type=text/css rel=stylesheet>
<title>%%PageTitle%%</title>
<SCRIPT language="JavaScript" src="images/js/DialogBox.js"></SCRIPT>
<SCRIPT language="JavaScript" src="images/js/DropDown_#.js"></SCRIPT>
<script type="text/JavaScript">%%script%%</script>
<BROLINTAG name="ColorScheme" type="object" />
</HEAD>
<body onload="%%onload%%"  onkeydown="javascript:hotKeys(event);">
<a href="#_ftn10" name="_ftnref1"></a>
<table width="860" height="100%" align="center" border="0" cellspacing="0" cellpadding="0">
<!-- Start of first TR -->
  <tr>
```

About us
%%OrgTitle1%%
%%OrgTitle2%%
Search:
Home
Product Search: - All - For:
%%content%%
%%LeftSideBar%%
Search:
%%Greeting%%
Take Your Business to the Next Level
Product Search:
- All -
For:
With
the fully integrated E-Commerce, Web Portal and Business Management Solution.

When creating a new template you begin the design process by creating a Page Style. The Page Style provides a framework for the page. You will divide the page into sections. The Page Style may include elements that are outside of sections. E.g. it may contain a background color that applies to the entire page. However most of the design elements will be placed within the sections. Within the Page Style you will use THTML to specify where each section belongs. You will also use THTML to define any Component Styles that you include within the Page Style.

When you register the HTML file in PortalProdigy, the PortalProdigy Style Processor will parse the file, removing each section style and component style, and independently registering them. Why does it do this? It does this so it can give the end user the ability to manage them.

The Page Style is stored as a shell within PortalProdigy. When a page is requested from the website, PortalProdigy's Page Processor will assemble the page based on Page Collection Settings. This process involves taking the Page Style (the shell) and dynamically inserting the components and data that are specified in the Page Collection Settings. To optimize performance, PortalProdigy automatically pre-assembles and caches pages.

To summarize, we have told you that Styles determine the visual design. That there are three types of Styles. There are Page Styles, Section Styles, and Component Styles. Page Styles define the layout and general design of a page. Page Styles include Sections for the header, left side bar, right side bar, main content, and footer. Sections serve as containers for components. Section Styles determine the visual design of sections. Components are objects that include data and/or functionality for specific functions and features. Component Styles determine the visual design of a component.

Styles & Page Construction in Detail:

All Template Styles that you create or modify are categorized as Custom Styles. All Template Styles that ship with PortalProdigy are categorized as Standard Styles. When you create a modified version of a Standard Style, it is stored as a Custom Style. This preserves the Standard Styles that ship with PortalProdigy so you can revert back to them. It also maintains their accessibility as examples.

Physically a Style is an HTML file. In addition to HTML code, a Style may contain CSS code, JavaScript code, and THTML code. A style may also include external resource files such as images, audio, flash/shockwave, CSS, and JavaScript. Resource files must be placed in a sub-folder of the folder where the Style's HTML file is located. The sub-folder must use the standard naming convention of the HTML file's name with "_files" appended to it. For example, the resource folder for:

c:\Page Styles\MyStyle.htm

is: **c:\Page Styles\MyStyle_files**

PortalProdigy uses a layered approach to web page design. A web page is broken down into 4 layers as follows:

Layer 1 = Page
Layer 2 = Sections
Layer 3 = Components
Layer 4 = Data Elements

This approach allows you (the web designer) design control over each page type and each component contained within a page type while at the same time providing the end user with the ability to both select and combine the components and Component Styles that you have created as well as to manage the content of each component. End users manage this process using a tool called Page Manager. As the web designer you too will use Page Manager to register and initially configure your designs. A complete description of Page Manager including tutorials can be found in the PortalProdigy Software User and Administration Guide.

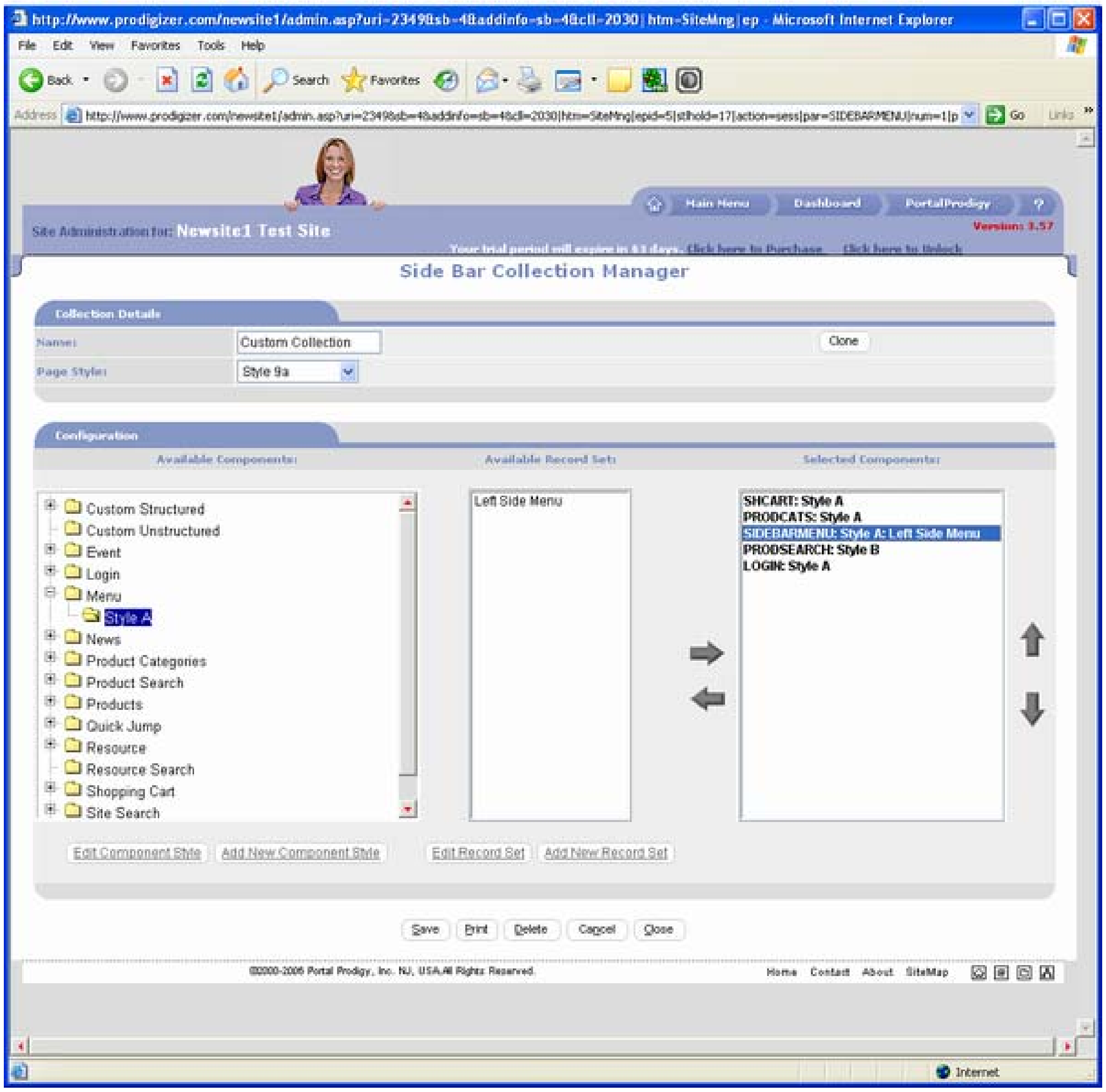
Let's begin by describing the hierarchy of layers and providing you with a definition for each layer. It is not critical for you to memorize or fully comprehend the layer hierarchy at this time. It will become clearer as you study the examples, create page designs and register them in the system.

PortalProdigy organizes things as follows:

- 1. Page Types
  - 2. Page Settings Collections
    - 3a. Page Style
    - 3b. Section Styles
      - 4. Section Collections
        - 5. Components
          - 6a. Component Record Sets
          - 6b. Component Styles
    - 3c. Other Settings

For each Page Type you can have one or more Page Settings Collections. E.g. the Home Page is a Page Type, the Product Detail page is a Page Type, etc. For each Page Settings Collection you specify a Page Style and Section Styles for the Header, Footer, and Main Content section. For each Section you create a Section Collection. Each Section Collection is comprised of one or more Components. For each Component you specify a Style and Record Set.

The following is the Side Bar Collection Manager which is used to select components for inclusion in a side bar.


http://www.prodigizer.com/newsite1/admin.asp?uri=2349&sb=4&addinfo=sb=4&cll=2030|htm=SiteMng|ep - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://www.prodigizer.com/newsite1/admin.asp?uri=2349&sb=4&addinfo=sb=4&cll=2030|htm=SiteMng|epid=5|stlhold=17|action=sess|par=SIDEBARMENU|num=1|p
Go Links
Main Menu
Dashboard
PortalProdigy
?
Version: 3.57
Site Administration for: Newsite1 Test Site
Your trial period will expire in 43 days. Click here to Purchase Click here to Unlock
Side Bar Collection Manager
Collection Details
Clone
Configuration
Available Components:
Available Record Sets:
Selected Components:
Edit Component Style
Add New Component Style
Edit Record Set
Add New Record Set
Save
Print
Delete
Cancel
Close
©2000-2006 Portal Prodigy, Inc. NJ, USA.All Rights Reserved.
Home Contact About SiteMap
Internet

Name: Custom Collection

Page Style: Style 9a

Available Components:
- Custom Structured
- Custom Unstructured
- Event
- Login
- Menu
  - Style A
- News
- Product Categories
- Product Search
- Products
- Quick Jump
- Resource
- Resource Search
- Shopping Cart
- Site Search

Available Record Sets:
- Left Side Menu

Selected Components:
- SHCART: Style A
- PRODCATS: Style A
- SIDEBARMENU: Style A: Left Side Menu
- PRODSEARCH: Style B
- LOGIN: Style A

Available Components are listed as a tree structure. Clicking on an Available Component lists the Available Styles for the component. When a Component Style is highlighted, each of the Available Record Sets for the component are listed. Highlighting a record set and clicking the right arrow adds it to the Selected Components.

Definitions of each Layer:

1. Page Type = a category of web pages based on feature and function. Examples are Home Page, Product Catalog Page, Product Detail Page, Shopping Cart Page, Event Detail Page, Event Registration Page, etc. A complete list of Page Types can be viewed using the tool named Page Type Manager which is accessible from the Utilities Menu.

2. Page Settings Collection = a set of standardized options that specify the design and content of a page. Page Settings Collections are created and managed using the tool named Page Manager.

3a. Page Style = a design for a web page. At bare minimum a Page Style defines a framework that specifies which standardized sections are to be included in the page and where. A Page Style may also include the design (Style) for each of the sections and their corresponding components. You have the choice of including them in the Page Style or loading them separately. Page Styles are added and managed using a tool named Style Manager.

3b. Section Style = a design (Style) for a specific section of a web page. PortalProdigy breaks up a page into the following sections: Header, Left Side Bar, Right Side Bar, Main Content, and Footer. For each section you can specify the design (Style) that is used to format the section. Section Styles are selected using Page Manager and become part of a Page Settings Collection. Note that the style for the Left Side Bar and Right Side Bar are always incorporated into the Page Style and are not individually managed in Page Manager like the other sections (Header, Main Content, and Footer).

3c. Other Settings = additional standardized options that are part of a Page Settings Collection. They are used to manage everything from the Color Scheme to Search Engine Optimization. Other Settings are configured using Page Manager.

4. Section Collection = a set of components for a section. Section Collections are created and managed using Collection Manager. Section Collections are assigned to Sections in Page Manager.

5. Component = a predefined object that combines data and functionality for a specific feature that can be inserted in web pages. Examples are menus, search, login, news, events, advertisement, product detail, etc. Components are selected as part of the process of creating Section Collections using Collection Manager.

6a. Component Record Set = data that is dynamically inserted within a component when the component is displayed on a web page. Some Component Record Sets are automatically created based on predefined program logic that is built into PortalProdigy such as the five most recent events, the 5 most recent news items, the selected product item, etc. Other Component Record Sets require someone to manually define them such as menus, structured mini browsers, and quick registration forms. Component Record Sets that are manually defined and managed using tools such as Menu Builder, Structured Mini Browser Builder and Quick Registration Builder. Component Record Sets are selected as part of the process of selecting Components.

6b. Component Style = a design for a component. For each component you can specify the design (Style) that is used to format the component. Component Styles are selected as part of the process of selecting Components.

The Parent Page Style

Section Styles and Component Styles always have a parent Page Style. This is necessary for many reasons. Amongst them is the necessity to keep the dimensions of Section

Styles and Component Styles synchronized with a Page Style's dimensions so that everything fits correctly.

PortalProdigy automatically limits the listing of available Section Styles and Components Styles to the children of the selected Page Style (the Parent). Each of PortalProdigy's Standard Page Styles has a set of Standard Section Styles and Standard Component Styles that are designed specifically for the Page Style. PortalProdigy uses a standard naming scheme for Styles such that changing Page Styles automatically selects the Standard Section Styles and Standard Component Styles of the same name for the newly selected Page Style. I.e. they are synchronized and automatically selected so the end user does not have to be concerned with this. There are some exceptions where a Section Style or Standard Component Style selection may not be available for a Page Style. In such cases, PortalProdigy will default to different one. E.g. all Standard Page Styles have at least one Standard Header Style named "Style A", but they may not have a "Style B". If a Page Style is selected that does not have a "Style B", PortalProdigy will automatically select "Style A".

When you load Custom Section Styles and Custom Component Styles they become children of a Parent Page Style. Thus if you load a Custom Section Style or Custom Component Style for a Standard Page Style, that Standard Page Style becomes its parent. If you load a Custom Section Style or Custom Component Style for a Custom Page Style, that Custom Page Style becomes its parent.

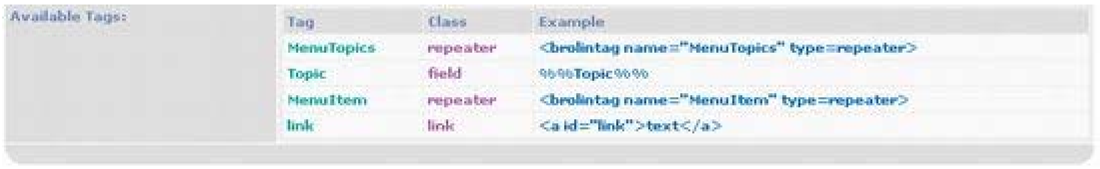
When you add a Custom Page Style, PortalProdigy allows you to inherit Section Styles and Component Styles from one of the Standard Page Styles, thus making all of its Section Styles and Component Styles available to your Custom Page Style. When you select a Standard Page Style to inherit from, you will want to select the one that best matches the dimensions and other characteristics of your Custom Page Style.

THTML Tags:

THTML tags come in several forms. There are standalone tags such as **`<BROLINTAG name="ColorScheme" type="object" />`**; there are variable tags for use within HTML tags such as **`%%PageTitle%%`** which is used as follows **`<title>%%PageTitle%%</title>`**; and attribute variable tags such as **`link`** which are used as parameter attributes within HTML tags such as **`<A class=BrolinSideMenulight id=link href="">Discussions</A></TD>`**.

In this section we will explain each form, their corresponding classes, and other options. We will also provide examples of how they are used. Note that in this guide we collectively refer to standalone tags, variable tags, and attribute variable tags, as THTML tags.

For each Page Type, Section Type, and Component Type, Style Manager lists the available tags along with an example. It also categorizes them by Class which we'll explain further on down in this section. The following are the available tags for Side Bar Menu Styles as listed in Style Manager:



Available Tags:

Tag	Class	Example
MenuTopics	repeater	<brolintag name="MenuTopics" type=repeater>
Topic	field	%%Topic%%
MenuItem	repeater	<brolintag name="MenuItem" type=repeater>
link	link	<a id="link">text</a>

The Classes for THTML tags are:

- Style
- Field
- Image
- Link
- Control
- Form
- Repeater
- Grid Repeater
- Code
- Separator
- Custom Values

Style Tags

Style tags are used to define Styles. The following tag is used to mark the beginning of a Header Style definition **<brolintag name="HeaderStyle" type="object" title="Style A">** and **</brolintag>** marks the end.

Field Tags

Field tags are used to insert values from PortalProdigy database fields. For example **%%PageTitle%%** is used as follows **<title>%%PageTitle%%</title>** to dynamically insert a Page Title.

Image Tags

Image tags are used to insert images that are stored in the PortalProdigy database. For example **%%Picture_small%%** is used as follows **<img src="%%Picture_small%%"></img>** to dynamically insert a product's small image.

Link Tags

Link tags are used to insert a dynamic link to a record or record set for a specific page type. Link tags are actually an implementation of the standard HTML Anchor tag. They contain a partial URL as the href value such as

href="main.asp?uri=1029&pi=" and an identifier such as **id="link"** or **id="HREF">** such as in **<a href="main.asp?uri=1029&pi=" id="link">text</a>**. PortalProdigy dynamically completes the URL by appending the applicable record ID to the end of the URL. For example **<a href="main.asp?uri=1029&pi=1002" id="link">text</a>** is used to open the Product record with Product ID = 1002 in the Product Detail page.

Control Tags

Control tags are used to insert controls that are mapped to predefined fields and/or actions within a form. Control tags are actually an implementation of standard HTML tags such as the Input tag and the Anchor tag. Examples are **<INPUT name=LoginName class="BrolinSearchForm" style="width:100%; height:16;">** and **<a onClick="javascript: document.frmSSrch.submit();" class="BrolinSearchFormTitle" style="cursor:pointer;">**

Form

Form tags are used to insert PortalProdigy forms for collecting data and performing a predefined action. For example **<form name="frmPrSrch" action="main.asp?uri=1065&fi=1" method=post>** is used to collect product search criteria and perform a search for products.

Repeater

Repeater tags are used to define a section of markup code that is to be repeated for each value in a record set. I.e. Repeater tags provide the ability to define patterns for the dynamic display of record sets. Repeater tags are used when the quantity of data objects is unknown at design time, such as with a menu or a listing of products that is maintained within PortalProdigy by end users. As the web designer you will simply define a pattern for how a single object such as one menu item or one product is to be displayed. PortalProdigy then applies your pattern dynamically to each object in the record set. For example **<BROLINTAG name="MenuItem" type="repeater">** is used to mark the beginning of a section of code that is to be applied to each menu item that is dynamically inserted in the menu.

Grid Repeater

Grid Repeater tags are used to create grids containing dynamically inserted content. Grid Repeater tags define a section of markup code that is to be repeated for each value in a record set that is to be dynamically inserted in the grid. For example **<BROLINTAG name=element type=gridrepeater grid_width=2 grid_height=10>** is used to mark the beginning of a section of code that is to be applied to each item that is dynamically inserted in a grid. It defines the grid as having two items per row and a maximum of 10 rows. I.e. a grid with two columns and a maximum of 10 rows. When the number of items in the record set exceeds the maximum rows, PortalProdigy has the ability to automatically insert next and previous options so the user can view the remainder of the record set.

Code

Code tags are used to dynamically insert Java Script stored in the PortalProdigy database. For example **%%script%%** inserts the script defined for the Page Collection and **%%onload%%** in **<body onload=%%onload%%>** inserts standard PortalProdigy scripting necessary in every page.

Separator

Separator tags are used to define markup code that is to be applied based on some predefined logic. Separator tags are often used within repeater tags to insert graphical elements that separate the repeated items. For example **<brolinsection name="separator">** is used to apply special formatting after each item except for the last item.

Custom Values:

Custom Values (CVs) are a type of THTML Field tag used to define variables that can be easily managed from site administration. CVs are dynamically replaced with values/objects maintained from within PortalProdigy. Custom Values are placed inside of Field tags and have the prefix of **cv#** for Style specific scope or **cvg#** for global scope. For example **%%cv#SeasonalImage%%** or **%%cvg#SeasonalImage%%**.

CVs provide a convenient method of extending the content management capabilities of the site. E.g. you may want to include a seasonal image within the home page header. If the image file is specified as a CV within the Style, it will be registered by the system and can be managed by going to the Style Management page and selecting Custom Value Management. Files such as images and shockwave/flash plug-ins can be replaced using the browse and upload functions in Custom Value Management.

PortalProdigy supports three types of CVs: text values, files, and links. A text value can be used to insert text, a font setting, HTML attribute, etc. Text values may contain up to 255 alphanumeric characters. A file can be used to insert a jpg file, gif file, wav file, mp3 file, Shockwave plug-in (Flash) file, etc. A link can be used to insert a link label with corresponding URL. Be sure to only use letters and numbers in CV names and do not use spaces. CVs are case sensitive.

Custom Values can be specified as having global scope so they may be re-used by other Styles. The concept is similar to the programming concept of declaring a global variable. If not specified as global the CV will be specific to the Style; i.e. it will have a local scope and only be available to the style where it was declared. Global CVs are managed using Global Custom Values option listed on the Utilities Menu.

THTML Tag Usage:

Standalone THTML tags have two required attribute parameters, **Name** and **Type**. **Name** is used to specify the name of the object. As mentioned above, Style Manager lists the available Tag Names which are predefined by the system. The **Type** attribute is used to specify whether the tag defines the display of a single object or a pattern for displaying repeating objects such as menu items. Set **Type=Object** when you want to display a single object. Set **Type=Repeater** when you want to describe a pattern for the display of repeating objects. A third parameter, **Title**, is used to assign a name to the Style of an object. The **Title** attribute is only used when defining a Style for the object. The following is an example of a beginning standalone tag that is used to define a Style:

```
<brolintag name="HeaderStyle" type=object name=”Style A”>
```

Between the beginning and ending tags is where you define the object. Just like with standard HTML tags, you can nest THTML Standalone Tags. This allows you to define Component Styles within another Component Style and to insert Data Element objects. When your Style is registered using Style Manager, PortalProdigy will extract each of your nested Component Styles and independently register them. The following is an example of a Header Style definition that contains nested Menu Style and Search Component Style definitions:

```
<brolintag name="HeaderStyle" type="object" title="Style A">

        <table width="100%" height="100%" border="0" cellspacing="0"
        cellpadding="0">

        <!-- Start of first HeaderStyle TR -->
        <tr>
        <td align="left" valign="top" height="19">
        <table width="100%" border="0" cellspacing="0" cellpadding="0">
        <tr>
        <td height="19" align="left" valign="top" style="padding-left:18;">
        <span class="BrolinSmallStyle">Phone: (949) 595-8300
        &nbsp;&nbsp;&nbsp;
        Email: </span><a href="mailto:sales@brolin.net" target="_blank"
        class="BrolinDarkLink">sales@brolin.net</a></td>
        <td align="right" valign="top" height="19" style="padding-
        right:15;">

        <brolintag name="HeaderMenu1" type="object" title="Style A">

                <table width="100%" border="0" cellspacing="0"
                cellpadding="0">
                <tr>
        <td align="right" valign="top">

        <BROLINTAG name="MenuItem" type="repeater">

                <img src="HTML/17_files/bullet_04.gif"
                name="Img_%%itm_id%%" width="8" height="8"
                border="0" style="margin-right:15; margin-left:15;"><a
                href="" onMouseOut="MM_swapImgRestore()"
                onMouseOver="MM_swapImage('Img_%%itm_id%%',''
                ,'HTML/17_files/bullet_03.gif',1)"
                class="BrolinAMenuBar" id=link>
                About us</a>

        </BROLINTAG></td>
        </tr>
        </table>

</brolintag>

</td>

</tr>
</table>


</td>
</tr>
<!-- End of first HeaderStyle TR -->
<!-- Start of second HeaderStyle TR -->
<tr>
<td align="left" valign="top" width="860" height="125"
class="BrolinHeader"></td>
</tr>
<!-- End of second HeaderStyle TR -->
<!-- Start of third HeaderStyle TR -->
<tr>
<td align="left" valign="top" height="5"></td>
</tr>
<!-- End of third HeaderStyle TR -->
<!-- Start of forth HeaderStyle TR -->
<tr>
<td align="left" valign="bottom" height="28">

<table width="100%" border="0" cellspacing="0" cellpadding="0">
<tr>
<td align="left" valign="bottom">
<brolintag name="HeaderMenu2" type="object" title="Style B">

        <table border="0" cellspacing="0" cellpadding="0">
        <tr>

                <BROLINTAG name="MenuItem" type="repeater">
                <td>
                <table width="100%" border="0" cellspacing="0"
                cellpadding="0" class="BrolinBMenuBar"
                style="cursor:pointer;"
                id="Menu_%%itm_id%%" onMouseOver="if
                (document.readyState == 'complete')
                aspnm_updateCell(this.id, 'BrolinBMenuBar_hover',
                'over');"
                onmouseout="if (document.readyState == 'complete')
                aspnm_updateCell(this.id, 'BrolinBMenuBar', 'out');">
                <tr>
                <td align="left" valign="bottom" width="10"
                height="24"><img
                src="HTML/17_files/MenuBarCornerLeft.gif"></td>
                <td align="center" valign="middle" style="border-
                top:1px solid #A7A7A7; padding-left:5; padding-
                right:5;"><a id="link"
                class="BrolinBMenuBar">Home</a></td>
                <td align="right" valign="bottom" width="10"
                height="24"><img
                src="HTML/17_files/MenuBarCornerRight.gif"></td>
                </tr>
                </table>

                </td>
                <td>&nbsp;</td>

        </brolintag>
        </tr>
        </table>

</brolintag>
</td>
<td height="30" align="right" valign="middle"
class="BrolinSmallStyle">

<brolintag name="SearchForm" type="object" title="Style A">

        <form name="frmSSrch" action="main.asp?uri=1001&fi=1"
        method=post>
        <table cellpadding="0" cellspacing="0" border="0">
        <tr>
        <td align="right" valign="middle" class="BrolinSmallStyle"
        style="padding-left:5;">
        Search: <input class="BrolinSearchForm" value=""
        name="SearchFor" type="text" height="14"
        width="60"></td>
        <td align="right" valign="middle" width="40"><a
        onMouseOut="MM_swapImgRestore()" onClick="javascript:
        document.frmSSrch.submit();"
        onMouseOver="MM_swapImage('Image_%%itm_id%%','','HT
        ML/17_files/go_on.gif',1)">
        <img src="HTML/17_files/go_off.gif"
```

```
          name="Image_%%itm_id%%" width="35" height="20"
          border="0"></a></td>
          </tr>
          </table>
          </form>

     </brolintag>

     <brolintag name="ProdSearchForm" type="object" title="Style A">

          <form name="frmPrSrch" action="main.asp?uri=1065&fi=1"
          method=post>
          <table cellpadding="0" cellspacing="0" border="0">
          <tr>
          <td align="right" valign="middle" class="BrolinSmallStyle"
          style="padding-left:5;">
          Product Search:
          <select name=PrDetailCategory class="BrolinSearchForm">
          <option value="All" selected>- All -</option>
          </select>
          For:
          <input class="BrolinSearchForm" name="SearchFor" value=""
          type="text" height="14" width="60"></td>
          <td align="right" valign="middle" width="40"><a href="#"
          onMouseOut="MM_swapImgRestore()" onClick="javascript:
          document.frmPrSrch.submit();"
          onMouseOver="MM_swapImage('Image_%%itm_id%%','','HT
          ML/17_files/go_on.gif',1)">
          <img src="HTML/17_files/go_off.gif"
          name="Image_%%itm_id%%" width="35" height="20"
          border="0"></a></td>
          </tr>
          </table>
          </form>

     </brolintag>
     </td>
     </tr>
     </table>
     </td>
     </tr>
     <!-- End of forth HeaderStyle TR -->
     </table>

</brolintag>
```

Let’s breakdown the previous example.

The Header Style “Style A” begins with:

```
<brolintag name="HeaderStyle" type="object" title="Style A">
```

And ends with:

```
</brolintag>
```

Between these two tags is the definition of the Header as well as the definitions for sub-components contained with the Header. The following code is part of the Header Style definition:

```
        <table width="100%" height="100%" border="0" cellspacing="0"
        cellpadding="0">

        <!-- Start of first HeaderStyle TR -->
        <tr>
        <td align="left" valign="top" height="19">
        <table width="100%" border="0" cellspacing="0" cellpadding="0">
        <tr>
        <td height="19" align="left" valign="top" style="padding-left:18;">
        <span class="BrolinSmallStyle">Phone: (949) 595-8300
        &nbsp;&nbsp;&nbsp;
        Email: </span><a href="mailto:sales@brolin.net" target="_blank"
        class="BrolinDarkLink">sales@brolin.net</a></td>
        <td align="right" valign="top" height="19" style="padding-
        right:15;">
```

After this is a nested definition of the Header Menu Style "Style A":

```
        <brolintag name="HeaderMenu1" type="object" title="Style A">

                <table width="100%" border="0" cellspacing="0"
                cellpadding="0">
                <tr>
                <td align="right" valign="top">

                <BROLINTAG name="MenuItem" type="repeater">

                        <img src="HTML/17_files/bullet_04.gif"
                        name="Img_%%itm_id%%" width="8" height="8"
                        border="0" style="margin-right:15; margin-left:15;"><a
                        href="" onMouseOut="MM_swapImgRestore()"
                        onMouseOver="MM_swapImage('Img_%%itm_id%%',''
                        ,'HTML/17_files/bullet_03.gif',1)"
                        class="BrolinAMenuBar" id=link>
                        About us</a>

                </BROLINTAG></td>
                </tr>
                </table>

        </brolintag>
```

Nested within the definition of the Header Menu Style "Style A" is a Repeater:

```
<BROLINTAG name="MenuItem" type="repeater">
```

The Repeater is used to define how each menu item is to be displayed. The following includes the beginning Repeater Tag, the definition of how to display the repeating object, and the ending Repeater Tag:

```
<BROLINTAG name="MenuItem" type="repeater">

        <img src="HTML/17_files/bullet_04.gif"
        name="Img_%%itm_id%%" width="8" height="8"
        border="0" style="margin-right:15; margin-left:15;"><a
        href="" onMouseOut="MM_swapImgRestore()"
        onMouseOver="MM_swapImage('Img_%%itm_id%%',''
        ,'HTML/17_files/bullet_03.gif',1)"
        class="BrolinAMenuBar" id=link>
        About us</a>

</BROLINTAG></td>
```

When the page is built, PortalProdigy will use the Repeater definition to dynamically insert and format each Menu Item that is part of the Menu's Record Set. You could hard code the menu items in your code, but this would prevent end users from managing the menu items using PortalProdigy's content management capabilities. There may be some exceptions where you may want to hard code content; however, in most cases this defeats the purpose and value of having a menu driven content management system such as PortalProdigy.

Another form of the Standalone Tag is used for defining the insertion of sections within a component where some programming logic is needed. The following is an example of a Section Tag:

```
<brolinsection name="separator">
```

Section Tags are often used within Repeater Tags to insert something between the objects. Let's say you want to insert a graphical element between all menu items except for the last one. You could use a Section Tag to accomplish this. The following is an example:

```
<brolinsection name="separator">
    <tr>
    <td height="9" background="HTML/17a_files/dot.gif"
    style="background-repeat:repeat-x; background-position:center;"
    colspan="2"></td>
    </tr>
</brolinsection>
```

THTML also uses the Class and ID attributes to reference PortalProdigy components and objects within standard HTML tags. E.g. the standard HTML Anchor tag can be used as follows `<A class=BrolinAMenuBar id="cv#Avl" href=""`

target="">Donations</A> to insert a link defined as a Custom Value and to format it using a PortalProdigy CSS class. Note that the **href=""** and **target=""**. PortalProdigy will insert the attribute values dynamically based on the specified **ID**. The text **Donations** will be removed by PortalProdigy's Style Processor and replaced with the value from the record set. It is just used as a placeholder/example for visualization purposes within the Style.

Page Styles:

The minimum requirements for a Page Style are:

- Preface tags/code (described below)
- Header Style
- Left Side Bar section tag (if your design includes a Left Side Bar)
- Main Content section tag
- Right Side Bar section tag (if your design includes a Left Side Bar)
- Footer Style

You may also include definitions for Component Styles, but they are not required. The following examples are taken from the Standard Page Style 17a.

Every Page Style should include the following preface tags/code inserted at the very top before all other code:

```
<HTML>
<HEAD>
<TITLE>%%PageTitle%%</TITLE>
<META http-equiv=Content-Type content="text/html;
charset=windows-1251">
<META content="" name=Description>
<META content="" name=keywords>
<META http-equiv=Cache-control content=no-cache>
<META content="MSHTML 6.00.3790.0" name=GENERATOR>
<LINK href="HTML/1_files/styles.css" type=text/css rel=stylesheet>
<LINK href="images/calendar.css" type=text/css rel=stylesheet>
<BROLINTAG name="ColorScheme" type="object" />
<SCRIPT language=JavaScript src="images/js/DialogBox.js"
type=text/javascript></SCRIPT>
<script type="text/JavaScript">%%script%%</script>
</HEAD>
<BODY onLoad="%%onload%%"
onkeydown=javascript:hotKeys(event);>
```

Let's breakdown the previous code:

1) Page Title tag. The Page Title value will be inserted dynamically by PortalProdigy.

```
<title>%%PageTitle%%</title>
```

2) Meta tags as follows. The Description and Keyword content attribute values will be inserted dynamically by PortalProdigy. The Charset value will be dynamically replaced only if one is entered in Page Settings; otherwise the value you placed in the style is used.

```
<META http-equiv=Content-Type content="text/html;
charset=windows-1251">

<META content="" name=Description>
<META content="" name=keywords>
<META http-equiv=Cache-control content=no-cache>
<META content="MSHTML 6.00.3790.0" name=GENERATOR>
```

3) CSS. Place your CSS before inserting the tags for PortalProdigy's standard CSS and Color Scheme as follows:

```
<LINK href="HTML/1_files/styles.css" type=text/css rel=stylesheet>
<LINK href="images/calendar.css" type=text/css rel=stylesheet>
<BROLINTAG name="ColorScheme" type="object" />
```

Even when you are not using PortalProdigy's CSS tags in the Page Style, you want to include the above lines because components that are inserted dynamically will need them. CSS Classes for Colors are managed using Color Manager. For each element's Background Color and Font Color, Page Manger displays the CSS Class name as a tool tip.

4) JavaScript. If you are using PortalProdigy's JavaScript insert as per the following example:

```
<SCRIPT language=JavaScript src="images/js/DialogBox.js"
type=text/javascript></SCRIPT>
```

5) Script tag. The script value **%%script%%** will be inserted dynamically by PortalProdigy.

```
<script type="text/JavaScript">%%script%%</script>
```

6) Onload tag. The onload value **%%onload%%** will be inserted dynamically by PortalProdigy. The **onkeydown** is to used enable PortalProdigy's hotkeys. Hotkeys allow administrators to access administrative features including the Site Administration Menu.

```
<BODY onLoad="%%onload%%"
onkeydown=javascript:hotKeys(event);>
```

You may insert inline Javascript anywhere within your Page Style as per the example below:

```
<script type="text/JavaScript">
<!--

function MM_swapImgRestore() { //v3.0
var i,x,a=document.MM_sr;
for(i=0;a&&i<a.length&&(x=a[i])&&x.oSrc;i++) x.src=x.oSrc;
}

function MM_preloadImages() { //v3.0
var d=document; if(d.images){ if(!d.MM_p) d.MM_p=new Array();
var i,j=d.MM_p.length,a=MM_preloadImages.arguments; for(i=0;
i<a.length; i++)
if (a[i].indexOf("#")!=0){ d.MM_p[j]=new Image;
d.MM_p[j++].src=a[i];}}
}

function MM_findObj(n, d) { //v4.01
var p,i,x; if(!d) d=document;
if((p=n.indexOf("?"))>0&&parent.frames.length) {
d=parent.frames[n.substring(p+1)].document; n=n.substring(0,p);}
if(!(x=d[n])&&d.all) x=d.all[n]; for (i=0;!x&&i<d.forms.length;i++)
x=d.forms[i][n];
for(i=0;!x&&d.layers&&i<d.layers.length;i++)
x=MM_findObj(n,d.layers[i].document);
if(!x && d.getElementById) x=d.getElementById(n); return x;
}

//-->
</script>
```

Every Page Style should have a Header Style and Footer Style and should indicate where the Main Content Section is to be inserted. So the next step to define a Header Style within your Page Style. The easiest method of managing the placement of the Header and other sections is to create a table with rows for the Header, Body, and Footer. If your design includes side bars you can create separate cells for each side bar and the main content section. This is how the Standard Page Styles are designed. In the following portion of the Page Style a table is defined for holding the contents of the page, including the first row where the Header will be inserted.

```
a href="#_ftn10" name="_ftnref1"></a>
<table width="860" height="100%" align="center" border="0"
cellspacing="0" cellpadding="0">
<!-- Start of first TR -->
  <tr>
    <td align="left" valign="top" height="50">
```

The next portion of the Page Style defines the Header Style. The Header Style includes Component Styles for Header Menu 1, Header Menu 2, Site Search, and Product Search.

```
<brolintag name="HeaderStyle" type="object" title="Style A">
```

```
<table width="100%" height="100%" border="0"
cellspacing="0" cellpadding="0">

<!-- Start of first HeaderStyle TR -->

        <tr>
        <td align="left" valign="top" height="19">
        <table width="100%" border="0" cellspacing="0"
        cellpadding="0">
        <tr>
        <td height="19" align="left" valign="top"
        style="padding-left:18;">&nbsp;</td>
        <td align="right" valign="top"  height="19"
        style="padding-right:15;">
```

Within the Header Style you can define Component Styles. The following section of code defines Header Menu Style A.

```
        <brolintag name="HeaderMenu1" type="object"
        title="Style A">
                <table border="0" cellspacing="0"
                cellpadding="0">
                <tr>
                <td>
```

The next section of code defines how Menu Items are to be displayed in the Header Menu Style A. It uses a Repeater tag, **type =repeater**, to define a pattern for each Menu Items that will be dynamically inserted from a menu Record Set.

```
                <BROLINTAG name="MenuItem"
                type="repeater">
                        <img src="HTML/17a_files/bullet_04.gif"
                        name="Img_%%itm_id%%" width="8"
                        height="8" border="0" style="margin-
                        right:15; margin-left:15;"><a href=""
                        onMouseOut="MM_swapImgRestore()"
                        onMouseOver="MM_swapImage('Img_%%i
                        tm_id%%','','HTML/17a_files/bullet_03.gif'
                        ,1)" class="BrolinAMenuBar" id=link>About
                        us</a>
                </BROLINTAG></td>

                </tr></table>

        </brolintag>

        </td>
        </tr>
        </table>
        </td>
        </tr>
```

```
<!-- End of first HeaderStyle TR -->

<!-- Start of second HeaderStyle TR -->
        <tr>
        <td align="left" height="50">
        <table width="100%" height="80" border="0"
        cellspacing="0" cellpadding="0" class="BrolinHeader">
        <tr>
        <td align="left" valign="bottom" width="14"
        height="14"><img
        src="HTML/17a_files/header_corner_LT.gif"></td>
        <td></td>
        <td></td>
        <td align="right" valign="bottom" width="14"
        height="14"><img
        src="HTML/17a_files/header_corner_RT.gif"></td>
        </tr>
        <tr>
        <td>&nbsp;</td>
        <td width="5" style="padding:5;">
```

In the next section of code a Global Custom Value tag is used to dynamically insert a logo image. You will need to load the Logo image file in Custom Value Management.

```
        <a href="home.asp?uri=1000"><img id=cvg#Logo1
        border="0"></a>
        </td>
        <td align="left" valign="middle" style="padding-
        left:15;">
```

In the next section of code Field tags are used to dynamically insert the Organization Titles.

```
        <span
        class="BrolinLogo">%%OrgTitle1%%</span><br>
        <span
        class="BrolinLogo2">%%OrgTitle2%%</span></td>
        <td>&nbsp;</td>
        </tr>
        <tr>
        <td align="left" valign="top" width="14"
        height="14"><img
        src="HTML/17a_files/header_corner_LB.gif"></td>
        <td></td><td></td>
        <td align="right" valign="top" width="14"
        height="14"><img
        src="HTML/17a_files/header_corner_RB.gif"></td>
        </tr>
        </table>
        </td>
        </tr>
```

```
<!-- End of second HeaderStyle TR -->

<!-- Start of third HeaderStyle TR -->

        <tr>
        <td align="left" valign="top" height="5"></td>
        </tr>

<!-- End of third HeaderStyle TR -->

<!-- Start of forth HeaderStyle TR -->

        <tr>
        <td align="left" valign="bottom" height="28">
        <table width="100%" border="0" cellspacing="0"
        cellpadding="0">
        <tr>
        <td  align="left" valign="bottom">

        <brolintag name="HeaderMenu2" type="object"
        title="Style B">
                <table border="0" cellspacing="0"
                cellpadding="0">
                <tr>

                <BROLINTAG name="MenuItem"
                type="repeater">
                        <td><table width="100%" border="0"
                        cellspacing="0" cellpadding="0"
                        class="BrolinBMenuBar"
                        style="cursor:pointer;"
                        id="Menu_%%itm_id%%"
                        onMouseOver="if (document.readyState ==
                        'complete') aspnm_updateCell(this.id,
                        'BrolinBMenuBar_hover', 'over');"
                        onmouseout="if (document.readyState ==
                        'complete') aspnm_updateCell(this.id,
                        'BrolinBMenuBar', 'out');">
                        <tr>
                        <td align="left" valign="bottom"
                        width="10" height="24"><img
                        src="HTML/17a_files/MenuBarCornerLeft.gi
                        f"></td>
                        <td align="center" valign="middle"
                        style="border-top:1px solid #A7A7A7;
                        padding-left:5; padding-right:5;"><a
                        id="link"
                        class="BrolinBMenuBar">Home</a></td>
                        td align="right" valign="bottom"
                        width="10" height="24"><img
                        src="HTML/17a_files/MenuBarCornerRight.
                        gif"></td>
                        </tr>
                        </table>
                        </td>
                        <td>&nbsp;</td>
                </brolintag>

                </tr>
                </table>
        </brolintag>

        </td>
        <td height="30" align="right" valign="middle"
        class="BrolinSmallStyle">

        <brolintag name="SearchForm" type="object"
        title="Style A">
                <table cellpadding="0" cellspacing="0"
                border="0">
                <form name="frmSSrch"
                action="main.asp?uri=1001&fi=1" method=post>
                <tr>
                <td align="right" valign="middle"
                class="BrolinSmallStyle" style="padding-
                left:5;">Search:<input class="BrolinSearchForm"
                value="" name="SearchFor" type="text"
                style="height:16; width:80;"></td>
                <td align="right" valign="middle"
                width="40"><a onClick="javascript:
                document.frmSSrch.submit();"
                style="cursor:pointer;">
                <img src="HTML/17a_files/go_off.gif"
                border="0"></a></td>
                </tr>
                </form>
                </table>
        </brolintag>

        <brolintag name="ProdSearchForm" type="object"
        title="Style A">
                <table cellpadding="0" cellspacing="0"
                border="0">
                <form name="frmPrSrch"
                action="main.asp?uri=1065&fi=1" method=post>
                <tr>
                <td align="right" valign="middle"
                class="BrolinSmallStyle" style="padding-
                left:5;">Product Search:<select
                name=PrDetailCategory
                class="BrolinSearchForm">
                <option  value="All" selected>- All -</option>
                </select>For:<input class="BrolinSearchForm"
                name="SearchFor" value="" type="text"
                style="height:16; width:80;"></td>
```

```
                <td align="right" valign="middle"
                width="40"><a onClick="javascript:
                document.frmPrSrch.submit();" style="
                cursor:pointer;">
                <img src="HTML/17a_files/go_off.gif"
                border="0"></a></td>
                </tr>
                </form>
                </table>
            </brolintag>

            </td>
            </tr>
            </table>
            </td>
            </tr>

        <!-- End of forth HeaderStyle TR -->

        </table>
        </brolintag>
        </td>
        </tr>

    <!-- End of first TR -->

    <!-- Start of  second TR -->

        <tr>
        <td align="left" valign="top" height="7"
        class="BrolinBMenuBar_hover"></td>
        </tr>

    <!-- End of  second TR -->

    <!-- Start of  third TR -->
        <tr>
        <td align="center" valign="top" bgcolor="#FFFFFF"
        style="padding-bottom:15; padding-top:15;">
```

The next portion of the Page Style defines a table as a container for the Main Content Section and within it defines a separate table as container for the Left Side Bar Section.

```
        <!-- Start Content + LeftSideBar Table!!! -->
            <table width="97%" height="100%" border="0"
            cellspacing="0" cellpadding="0">
            <tr>
            <td width="215" align="left" valign="top">

            <!-- Start LeftSideBar -->
                <table width="100%" height="100%"
                class="BrolinLeftSide" border="0"
                cellspacing="0" cellpadding="0">
                <tr>
                <td width="9%" align="left" valign="top"
                height="14"><img
                src="HTML/17a_files/grey_left_top.gif"></td>
                <td width="82%">&nbsp;</td>
                <td width="9%" align="right" valign="top"
                height="14"><img
                src="HTML/17a_files/grey_right_top.gif"></td>
                </tr>
                <tr>
                <td>&nbsp;</td>
                <td align="left" valign="top">
```

The following tag specifies the insertion of the Component Collection for the Left Side Bar.

```
        %%LeftSideBar%%
```

If you are going to define Side Bar Component Styles within your Page Style, insert them immediately after the **%%LeftSideBar%%** tag. See the Component Styles section of this guide for examples and additional instructions.

```
        </td>
        <td>&nbsp;</td>
        </tr>
        <tr>
        <td align="left" valign="bottom"
        height="14"><img
        src="HTML/17a_files/grey_left_bottom.gif">
        </td>
        <td>&nbsp;</td>
        <td align="right" valign="bottom"
        height="14"><img
        src="HTML/17a_files/grey_right_bottom.gif">
        </td>
        </tr>
        </table>
    <!-- End LeftSideBar -->

    </td>
    <td width="13"><img
    src="HTML/17a_files/Transperant.gif" width="13"
    height="1">
    </td>
    <td width="605" align="left" valign="top"
    style="padding-bottom:15;">
```

The following tag specifies the insertion of the Component Collection for the Main Content Section.

```
    %%content%%
```

```
                </td></tr>
                <tr>
                <td align="center" valign="bottom">
                </td>
                </tr>
```

If you are going to define Main Content Component Styles within your Page Style, insert them somewhere after the **%%LeftSideBar%%** tag. See the Component Styles section of this guide for examples and additional instructions.

```
        <!-- End Content + LeftSideBar Table!!! -->

    <!-- End of  third TR -->

        <td/>
        <tr/>

    <!-- Start of  forth TR -->

        <tr>
        <td align="center" valign="bottom" height="72"
        background="HTML/17a_files/FooterBGR.jpg"
        style="background-repeat:no-repeat;">
        <table width="100%" height="30" border="0"
        cellspacing="0" cellpadding="0">
        <tr>
        <td align="left" valign="bottom" width="60">&nbsp;</td>
        <td align="center" valign="bottom" style="padding-
        bottom:2;">&nbsp;
```

The following section of code defines a Footer Menu Style:

```
        <BROLINTAG name="FooterMenu1" type="object" title="Style
        A">
                <table width="100%" border="0" cellspacing="0"
                cellpadding="0"><tr>
                <td align="center" valign="bottom">
                <p class="BrolinTextCentered">

                <brolintag name="menuitem" type="repeater">
                        <a href="" class="BrolinFooterAMenuBar"
                        id="link">Home</a>

                        <brolinsection name="separator">
                                <img src="HTML/17a_files/bullet_02.gif"
                                style="margin-left:6; margin-right:6;">
                        </brolinsection>

                </brolintag>

                </p>
                </td>
                </tr>
                </table>
        </brolintag>

        </td>
        <td align="left" valign="bottom" width="30" style="padding-
        bottom:3;">
        <a href="#_ftnref1" name="_ftn1"
        onMouseOut="MM_swapImgRestore()"
        onMouseOver="MM_swapImage('Images_%%itm_id%%','','H
        TML/17a_files/top_page_on.gif',1)">
        <img src="HTML/17a_files/top_page_off.gif" border="0"
        title="Top of Page" name="Images_%%itm_id%%">
        </a>
        </td>
        <td align="left" valign="bottom" width="30" style="padding-
        bottom:5;"><a href=""><img src="HTML/17a_files/print.gif"
        border="0" title="Print">
        </a>
        </td>
        </tr>
        </table>
        </td>
        </tr>

<!-- End of  forth TR -->

        </table>
```

The following section of the code defines the Footer Style.

```
        <BROLINTAG name="FooterStyle" type="object" title="Style
        A">
                <p class="BrolinTextCentered">
                <span
                class="BrolinCopy">%%Copyright%%</span><br/>
                <span
                class="BrolinCopy">%%OwnerCopyright%%</span>
                <br/><br/>
                %%Owner%%
                </p>
        </brolintag>

</body>
</html>
```

This is the end of the Page Style.

Do not use the PortalProdigy HTML Editor to edit Page Styles. The reason for this is that the preface code will conflict with the Browser Page containing the editor. You may use the editor to create and edit both Section and Component Styles.

Section Styles

Every Page Style should have a Header Style and a Footer Style and should indicate where the Main Content Section is to be inserted. The Main Content Style is unique to each Page Type. Typically you will create one Page Style for use by all pages. We suggest defining your Home Page Main Content Component Styles within the one Page Style. Additional Header Styles and Footer Styles can be defined independently of the Page Style. The Main Content Section Styles for all pages other than the Home Page should be defined independently of the Page Style. Most websites will use the Standard Main Content Section Styles for most Page Types. Use Page Type Management to add Main Content Section Styles for all pages other than the Home Page. Use Home Page Manager to add Main Content Section Styles for the Home Page.

Side Bar Sections are optional. Side Bar Sections do not have separate Styles like the Header Section, Main Content, and Footer Section. The visual design of Side Bar Sections is always defined within the Page Style. This was done in order to simply things since side bars are primarily empty containers for components. Side Bar Sections are typically defined as empty cells in a table that include the Section tags **%%LeftSideBar%%** and **%%RightSideBar%%**.

When you create Header Styles, Footer Styles, and Main Content Section Styles independently of Page Styles, do not include the Preface code that was described for Page Styles.

Component Styles:

As mentioned previously, Component Styles may be created independently of Page Styles or they may be defined within a Page Style. When created independently, do not include the Preface code that was described for Page Styles. This code is always provided by the Page Style and should never be placed in Section or Component Styles.

The following is a style for a Shopping Cart Component:

```
<TABLE cellSpacing=0 cellPadding=0 width="100%" border=0>

        <TBODY>
        <TR>
        <TD height=59>

                <TABLE cellSpacing=0 cellPadding=0 width="100%"
                border=0>
                        <TBODY>
                                <TR>
                                <TD vAlign=bottom align=left width=58
                                height=59><IMG
                        src="HTML/17a_files/shopping_bag17.gif"></TD
                        >

                        <TD vAlign=bottom>

                                <TABLE height=59 cellSpacing=0
                                cellPadding=0 width="100%" border=0>
                                        <TBODY>
                                        <TR>
                                                <TD style="BACKGROUND-
                                                REPEAT: repeat-x"
                                                background=HTML/17a_files/ti
                                                tle_top.gif bgColor=#ffffff
                                                height=19></TD>
                                        </TR>
                                        <TR>
                                                <TD class=BrolinSmallStyle02
                                                vAlign=center align=left
                                                bgColor=#ffffff>Your
                                                Cart</TD>
                                        </TR>
                                        <TR>
                                                <TD style="BACKGROUND-
                                                REPEAT: repeat-x"
                                                background=HTML/17a_files/ti
                                                tle_bottom.gif
                                                height=11></TD>
                                        </TR>
                                        </TBODY>
                                </TABLE>

                        </TD>

                        <TD vAlign=bottom align=right width=49
                        height=59><IMG
                        src="HTML/17a_files/title_end02.gif"></TD>

                        </TR>

                </TBODY>

        </TABLE>
</TD>
</TR>

<TR>
<TD>
        <TABLE class=BrolinBackgroundGrey height="100%"
        cellSpacing=0 cellPadding=0 width="100%" border=0>
                <TBODY>
                        <TR>
                                <TD style="BACKGROUND-REPEAT: repeat-
                                y"
                                background=HTML/17a_files/mb_left.gif><
                                /TD>
                                <TD vAlign=top align=left>
                                        <P class=BrolinSmallStyle>Items:
                                        %%Qty%%</P>
                                        <P class=BrolinSmallStyle>Subtotal:
                                        %%Price%%</P>
                                        <P class=BrolinSmallStyle><A
                                        id=more
                                        onmouseover="MM_swapImage('Ima
                                        gesh_%%itm_id%%','','HTML/17a_fi
                                        les/shopping_bt_on.gif',1)"
                                        onmouseout=MM_swapImgRestore()
                                        href=""><IMG height=17
                                        src="HTML/17a_files/shopping_bt_o
                                        ff.gif" width=129 border=0
                                        name=Imagesh_%%itm_id%%></A
                                        ></P>
                                </TD>
                                <TD style="BACKGROUND-REPEAT: repeat-
                                y" width=24
                                background=HTML/17a_files/mb_right.gif>
                                </TD>
                        </TR>
                        <TR height=24>
                                <TD vAlign=bottom align=left width=24
                                height=24>
                                        <IMG
                                        src="HTML/17a_files/mb_left_botto
                                        m02.gif">
                                </TD>
                                <TD style="BACKGROUND-POSITION: 50%
                                bottom; BACKGROUND-REPEAT: repeat-x"
                                background=HTML/17a_files/mb_bottom.gi
                                f height=24>&nbsp;</TD>
                                <TD vAlign=bottom align=right width=24
                                height=24>
                                        <IMG
                                        src="HTML/17a_files/mb_right_bott
                                        om02.gif">
                                </TD>
                        </TR>
                </TBODY>

        </TABLE>

    </TD>
    </TR>
    </TBODY>

</TABLE>
<BR>
```

Page Management:

Page Manager

Header Collection Manager

Side-Bar Collection Manager

Main Content Collection Manager

THTML provides tags for defining patterns for the dynamic display of components as a grid. There are four different row types:

- `<brolintag name="row1" type="object">` single component
- `<brolintag name="row2" type="object">` two components
- `<brolintag name="row3" type="object">` three components
- `<brolintag name="row4" type="object">` four components

The following is the Main Content section Style A for Page Style 14:

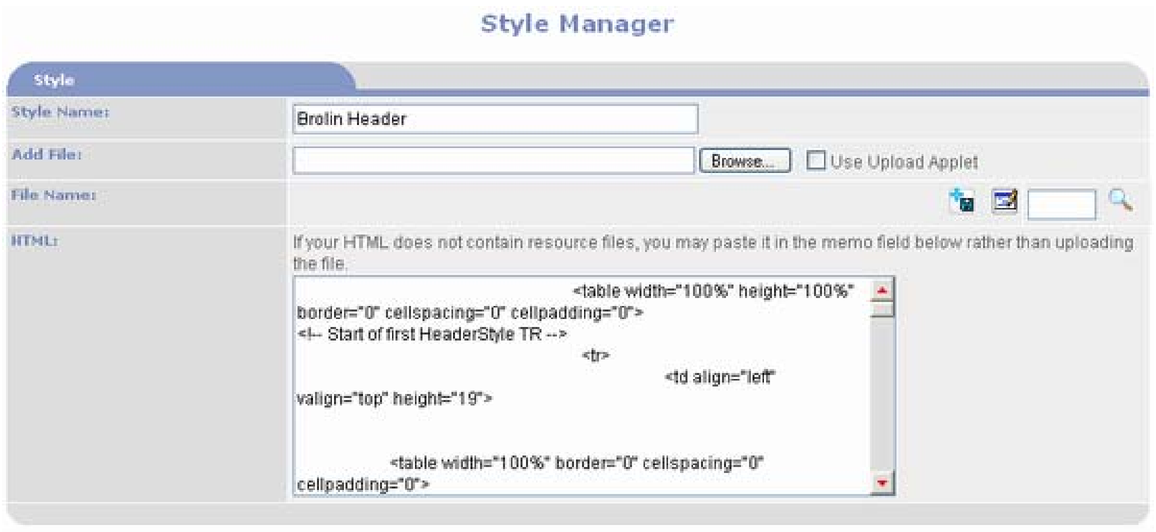
```
<table width="100%" border="0" cellspacing="0" cellpadding="0">

        <tr>

                <td align="left" valign="top">

                        %%Contents%%

                        <brolintag name="row1" type="object">

                                <table width="100%" border="0" cellspacing="0"
                                cellpadding="0">

                                        <tr>
                                                <td>%%MB%%</td>
                                        </tr>

                                </table><br/>

                        </brolintag>

                        <brolintag name="row2" type="object">

                                <table width="100%" border="0" cellspacing="0"
                                cellpadding="0" height="1">

                                        <tr>
                                                <td id="r21" valign=top style="padding-
                                                right:7;">%%MB1%%</td>
                                                <td id="r22" valign=top style="padding-
                                                left:7;">%%MB2%%</td>
```

```
                              </tr>

                         </table><br/>

                    </brolintag>

                    <brolintag name="row3" type="object">

                         <table width="100%" border="0" cellspacing="0"
                         cellpadding="0" height="1">

                              <tr><td width=33% id="r31" style="padding-
                              right:7;">%%MB1%%</td>
                                   <td width=33% id="r32" style="padding-
                                   right:7; padding-left:7;">%%MB2%%</td>
                                   <td width=33% id="r33" style="padding-
                                   left:7;">%%MB3%%</td>

                              </tr>

                         </table><br/>

                    </brolintag>

                    <brolintag name="row4" type="object">

                         <table width="100%" border="0" cellspacing="0"
                         cellpadding="0" height="1">

                              <tr>

                                   <td width=33% id="r41" style="padding-
                                   right:7;">%%MB1%%</td>
                                   <td width=33% id="r42" style="padding-
                                   right:7; padding-left:7;">%%MB2%%</td>
                                   <td width=33% id="r43" style="padding-
                                   right:7; padding-left:7;">%%MB3%%</td>
                                   <td width=33% id="r44" style="padding-
                                   left:7;">%%MB4%%</td>

                              </tr>

                         </table><br/>

                    </brolintag>

               </td>

          </tr>

</table>
```

Footer Collection Manager

Style Manager

Style Manger is used to add and update Styles. It also used to view, download, and copy Standard Styles.   Style Manger also lists the Available Tags for the applicable Page, Section, or Component Type. For a detailed explanation of Style Manger and how to use it, see the Style Manager section of the Utilities Chapter in the PortalProdigy User and Administration Guide.

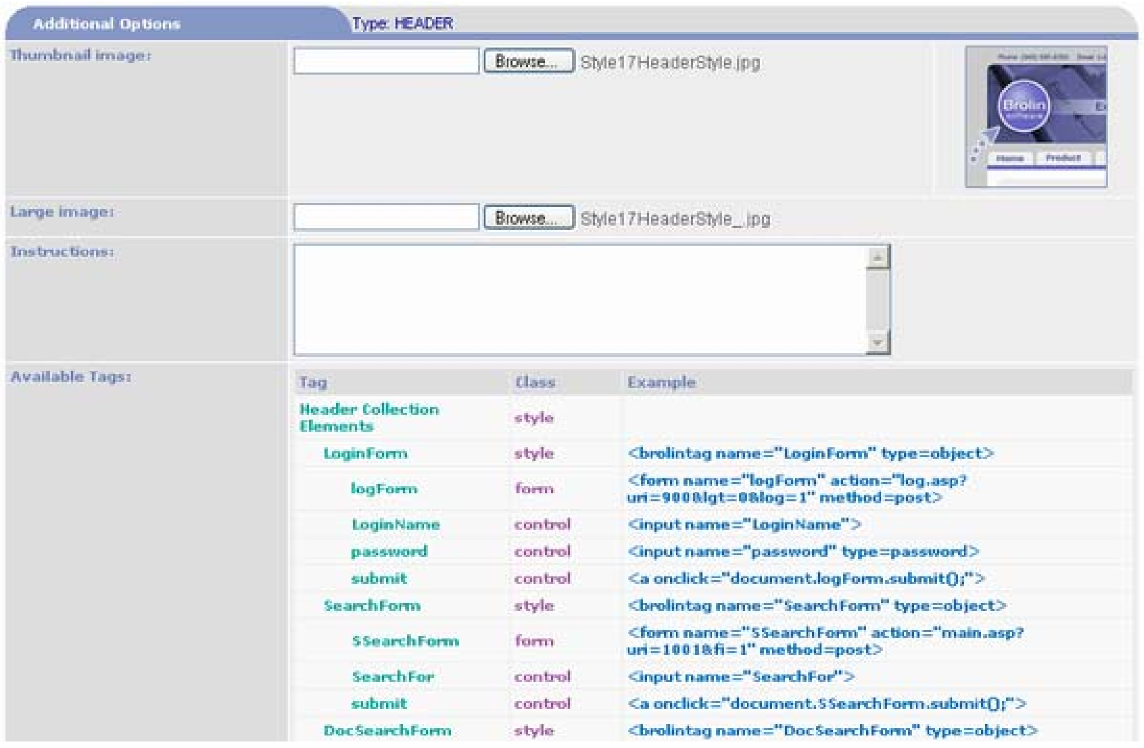


Style Manager

Style

Style Name:	Brolin Header
Add File:	Browse... ☐ Use Upload Applet
File Name:	
HTML:	If your HTML does not contain resource files, you may paste it in the memo field below rather than uploading the file.

```
                                  <table width="100%" height="100%"
border="0" cellspacing="0" cellpadding="0">
<!-- Start of first HeaderStyle TR -->
                                <tr>
                                            <td align="left"
valign="top" height="19">


                <table width="100%" border="0" cellspacing="0"
cellpadding="0">
```



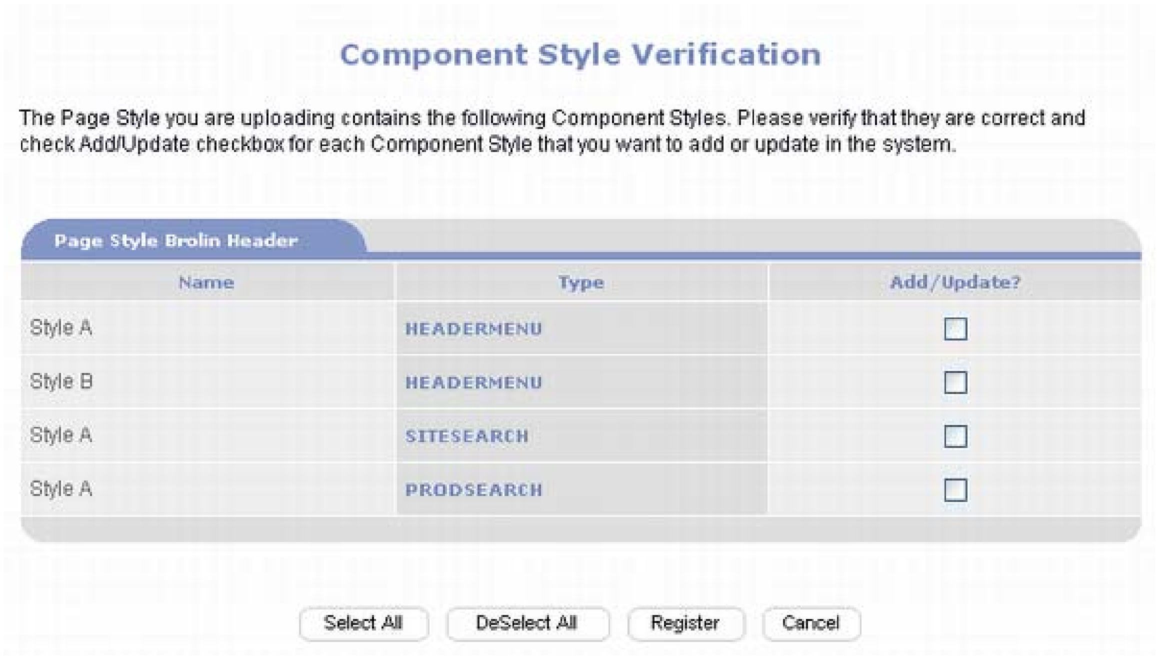
Additional Options — Type: HEADER

Thumbnail image:	Browse... Style17HeaderStyle.jpg
Large image:	Browse... Style17HeaderStyle_.jpg
Instructions:	

Available Tags:

Tag	Class	Example
Header Collection Elements	style	
LoginForm	style	<brolintag name="LoginForm" type=object>
logForm	form	<form name="logForm" action="log.asp?uri=900&lgt=0&log=1" method=post>
LoginName	control	<input name="LoginName">
password	control	<input name="password" type=password>
submit	control	<a onclick="document.logForm.submit();">
SearchForm	style	<brolintag name="SearchForm" type=object>
SSearchForm	form	<form name="SSearchForm" action="main.asp?uri=1001&fi=1" method=post>
SearchFor	control	<input name="SearchFor">
submit	control	<a onclick="document.SSearchForm.submit();">
DocSearchForm	style	<brolintag name="DocSearchForm" type=object>

After you load a Page style or a Section Style you need to run the Component Verification process. This process identifies and lists each Component Style contained within the Page or Section Style. It helps you identify problems and exclude Component Styles that you don't want updated. It is accessed from Style Manager by clicking on the Components Verification button.



Component Style Verification

The Page Style you are uploading contains the following Component Styles. Please verify that they are correct and check Add/Update checkbox for each Component Style that you want to add or update in the system.

Page Style Brolin Header

Name	Type	Add/Update?
Style A	HEADERMENU	☐
Style B	HEADERMENU	☐
Style A	SITESEARCH	☐
Style A	PRODSEARCH	☐

You need to check the checkbox for each Component Style that you want added if it is a new style, or updated if it is an existing style. If a Component Style is not listed, there is probably a problem with the tags. If you modified a Standard Style it necessary to rename and load it as a Custom Style. The Component Verification process will help you catch such oversights. For a detailed explanation of Components Style Verification and how to use it, see the Components Style Verification section of the Utilities Chapter in the PortalProdigy User and Administration Guide.

Page Type Manager

Other Useful Information:

Images

PortalProdigy includes an Image Fill Color Processor that dynamically changes the fill color of GIF images based on a Page's Color Settings. This is useful for GIFs that are used to create special affects such as rounded corners. It frees you from hard coding their colors, thus allowing the end user to dynamically control them.

To take advantage of the Image Fill Color Processor you must use PortalProdigy's pre-defined GIF images. You will see them used throughout the standard styles. They are also listed in the Image Directory which can be accessed from the Utilities menu by selecting the Images Directory option.

How does the web designer enter the path so our Parser will properly save it? Are some of these GIF images stored separately in template folders?

Page Caching:
PortalProdigy preassembles Pages and caches them in order to provide faster page delivery. Typically everything except for the main content section is preassembled and cached. If you find that a page does not include a change to one of your styles or components, go to the Page Manager for the page and click the Preview button. This will force a rebuild of the page and update the cache.

Search Engine Optimization (SEO)

Page Titles, Meta Tag Descriptions, and Meta Tag Keywords are maintained from within the PortalProdigy user interface. THTML provides tags that are used to dynamically insert these values. PortalProdigy can automatically generate Page Titles, Meta Tag Descriptions and Meta Tag Keywords for each Product Item, Event, News Item, Document, etc. These values can be manually entered for each record as well as inherited from other records. The only thing you have to do as the web designer is to include the following tags in each of your Page Styles:

```
<TITLE>%%PageTitle%%</TITLE>
<META content="" name=Description>
<META content="" name=keywords>
```

PortalProdigy also provides the ability to automatically create Landing Pages and a Landing Page Site Map. Landing Pages are pages that are optimized for search engines. Successful Landing Pages are static HTML, small in size, do not contain scripts, contain summarized information about the subject that is optimized for search engines (relative keywords are reused in viewable subject information and are contained in page title), and have URLs that contain keywords such as product names, news titles, etc. Landing pages are designed to induce the viewer to click to view additional information or to purchase a product.

PortalProdigy can create Landing Pages for the following:
- Documents
- Events
- Exchange Postings
- Memberships
- News Articles
- Product Items

PortalProdigy allows you to create custom styles for Landing Pages.

You need to provide a visible link to the Landing Page Site Map on the site’s Home Page.

Script to cache images in browser:

By default we disable caching because it prevents changes from being viewed. Use the following script to cache commonly used images that are not planned to be changed such as images included with PortalProdigy or that will be changed infrequently.

```
<script LANGUAGE="JavaScript">
        logo1Image = new Image(); logo1Image.src = "images/logo1.gif";
        img1=new Image(); img1.src ="images/bullet.gif";
        img2=new Image(); img2.src ="images/transparent.gif";
</script>
```