
*PDG Shopping
Cart 2004*

Technical Reference Guide

PDG Software, Inc.
1751 Montreal Circle, Suite B
Tucker, Georgia 30084-6802

Copyright ©1998 - 2001 PDG Software, Inc.; All rights reserved.

PDG Software, Inc. ("PDG Software") retains all ownership rights to the software programs (referred to herein as "Software") offered by PDG Software and related documentation. Use of the Software and related documentation is governed by the license agreement accompanying the Software and applicable copyright law.

THIS DOCUMENTATION IS PROVIDED "AS-IS" WITHOUT WARRANTY OF ANY KIND. IN NO EVENT SHALL PDG SOFTWARE BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF BUSINESS, LOSS OF USE OR DATA, INTERRUPTION OF BUSINESS, OR FOR INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY KIND, ARISING FROM ANY ERROR IN THIS DOCUMENTATION.

PDG Software, Inc.
<http://www.pdgsoft.com>

August, 2003

Contents

	Contents	5
Chapter 1	Introduction	7
Chapter 2	CGI and Cookies	9
	A Note About Cookie Usage	10
Chapter 3	Country List	11
Chapter 4	HTML Template Files	17
	Repeat Tags to Display Multiple Products	18
	Text and CGI Includes	21
	Basket Template File	22
	AddItem Template File	37
	Item Template File	39
	MultiAdd Template File	51
	Checkout Template File	53
	Verify Template File	61
	ThankYou Template File	78
	DenyCredit Template File	91
	AcceptCredit Template File	93
	Search Results Template	93
	Category Template	99
	ShipCompare Template	101
	UPS Track Template File	103
	UPS Tracking Result Template File	104
	UPS Address Validation Template File	104
	Verify - Inventory Error Template	105
	Error Template File	107
	SSL Templates	108
	Empty Templates	109
	HTML Email Templates	110
	Universal Tags	110

Chapter 5 Cart Actions 115

Search Action	116
PreAdd Action	118
MultiPreAdd Action	121
Add Action	122
MultiAdd Action	124
MAdd.[#] Action	127
OptFullAdd Action	129
OAdd Action	132
Remove Action	133
RemoveAll Action	134
Display Action	134
Recalc Action	135
Return Action	136
Checkout Action	136
Verify Action	137
BuyIt Action	144
ListCategories Action	151
Cart Action Examples	153

What is the Technical Reference Guide?

This guide has been prepared for use by technical support staff, or advanced users, who may need technical information about PDG Shopping Cart 2004. The Technical Reference Guide will help you with customizing certain areas of your Shopping Cart. It should also give you a better understanding as to how PDG Shopping Cart works.

This guide is intended for users of PDG Shopping Cart 2004 who have an advanced understanding of how Web sites work, such as Web masters, site administrators and IT professionals. This Technical Reference Guide assumes that you have already read the PDG Shopping Cart User Guide.

The PDG Shopping Cart 2004 Technical Reference Guide is divided into nine chapters:

- Introduction
- CGI and Cookies
- Shopper.conf File
- Product Definition
- Inventory Configuration
- Shipping.conf File
- Country List
- HTML Template Files
- Cart Actions

At PDG we want to make sure that you get the most out of our software. That is why we've provided this and other manuals to help you better understand our products many features. Other guides you may find useful are:

- PDG Shopping Cart 2004 Installation Guide
- PDG Shopping Cart 2004 Upgrade Guide
- PDG Shopping Cart 2004 User Guide
- PDG Key Manager Guide
- PDG Permissions Guide
- Dreamweaver Component Guide
- FrontPage Component Guide
- NetObjects Component Guide



What is CGI?

The Common Gateway Interface (known as ‘CGI’) is a standard method of writing software that can receive data from a Web server. CGI programs are programs written to receive data from a Web server in order to perform additional processing on the data. This processing can be implemented to store the information, perform some operation on the data, and/or return results of the operation back to the Web server to be sent to the end user. CGI programs allow Webmasters and Web site owners to extend their Web site's functionality beyond traditional HTML.

Because CGI programs are software programs, they must be written and developed. CGI programs can be written in almost any language: C, C++, Basic, Shell script (for Unix users), Perl (for Unix users and, now, a growing segment of NT users), and other languages as well.

Typically, CGI programs are custom developed for a Web site, meeting the exact requirements dictated by the Webmaster. However, there is a growing collection of pre-made CGI programs. Pre-made CGI programs offer advantages and disadvantages over custom CGI programs.

Pre-made CGI programs may not fit every requirement needed or dictated by a Webmaster. However, pre-made CGI programs are usually tested more thoroughly and are used by more than just one Web site. Pre-made CGI programs may also be backed by a company, with technical support and/or money back guarantees.

PDG Shopping Cart is the latter. PDG Shopping Cart is a pre-made and pre-packaged CGI program.

A Note About Cookie Usage

PDG Shopping Cart uses cookies, which have been used controversially by some Web sites to obtain marketing information without the visitor's knowledge. PDG Shopping Cart only uses cookies so that a customer's shopping cart can be remembered between visits. No other information is used in the cookie for any purpose. Customers and Web site administrators may verify this with any of the popular 'cookie administration' programs currently available on the market. Under no circumstance can a cookie load or execute a virus on your system, nor can it read and send files or data through the Internet. A wealth of information on cookies can be found at Cookie Central, at <http://www.cookiecentral.com>. This site contains valuable and meaningful information on cookies, and lists all of the details of cookies, including what they are, what they do, and what they cannot do.

Also realize that cookies are used extensively on some very popular and large Web stores, including Amazon.com. By using cookies, they are better able to serve their customers and increase business and revenue.



Country List

The following table provides the complete list of countries that PDG Shopping Cart recognizes for its various operations and actions. These codes are used in all shipping & handling rules, as well as for UPS pricing of international shipments. The Index is used for the Default_Country Key found in the Shopper.conf file.

Country	Code	Index
All Countries	~AA	
Albania	~AL	1
Algeria	~DZ	2
American Samoa	~AS	3
Andorra	~AD	4
Anguilla	~AI	5
Antigua & Barbuda	~AG	6
Argentina	~AR	7
Aruba	~AW	8
Australia	~AU	9
Austria	~AT	10
Azores	~AP	11
Bahamas	~BS	12
Bahrain	~BH	13
Bangladesh	~BD	14
Barbados	~BB	15
Belgium	~BE	16
Belize	~BZ	17
Belarus	~BY	18
Benin	~BJ	19
Bermuda	~BM	20
Bolivia	~BO	21

Bonaire	~BL	22
Botswana	~BW	23
Brazil	~BR	24
British Virgin Islands	~VG	25
Brunei	~BN	26
Bulgaria	~BG	27
Burkina Faso	~BF	28
Burundi	~BI	29
Cambodia	~KH	30
Cameroon	~CM	31
Canada	~CA	32
Canary Islands	~CE	33
Cape Verde Islands	~CV	34
Cayman Islands	~KY	35
Central African Republic	~CF	36
Chad	~TD	37
Channel Islands	~NN	38
Chile	~CL	39
China, Peoples Republic of	~CN	40
Colombia	~CO	41
Congo	~CG	42
Cook Islands	~CK	43
Costa Rica	~CR	44
Croatia	~HR	45
Curacao	~CB	46
Cyprus	~CY	47
Czech Republic	~CZ	48
Denmark	~DK	49
Djibouti	~DJ	50
Dominica	~DM	51
Dominican Republic	~DO	52
Ecuador	~EC	53
Egypt	~EG	54
El Salvador	~SV	55
England	~EN	56
Equatorial Guinea	~GQ	57
Eritrea	~ER	58
Estonia	~EE	59
Ethiopia	~ET	60
Faeroe Islands	~FO	61
Federated States of Micronesia	~FM	62
Fiji	~FJ	63
Finland	~FI	64
France	~FR	65
French Guiana	~GF	66
French Polynesia	~PF	67
Gabon	~GA	68
Gambia	~GM	69
Georgia	~GE	70
Germany	~DE	71
Ghana	~GH	72
Gibraltar	~GI	73
Greece	~GR	74

Greenland	~GL	75
Grenada	~GD	76
Guadeloupe	~GP	77
Guam	~GU	78
Guatemala	~GT	79
Guinea	~GN	80
Guinea-Bissau	~GW	81
Guyana	~GY	82
Haiti	~HT	83
Holland	~HO	84
Honduras	~HN	85
Hong Kong	~HK	86
Hungary	~HU	87
Iceland	~IS	88
India	~IN	89
Indonesia	~ID	90
Israel	~IL	91
Italy	~IT	92
Ivory Coast	~CI	93
Jamaica	~JM	94
Japan	~JP	95
Jordan	~JO	96
Kazakhstan	~KZ	97
Kenya	~KE	98
Kiribati	~KI	99
Kosrae	~KO	100
Kuwait	~KW	101
Kyrgyzstan	~KG	102
Laos	~LA	103
Latvia	~LV	104
Lebanon	~LB	105
Lesotho	~LS	106
Liberia	~LR	107
Liechtenstein	~LI	108
Lithuania	~LT	109
Luxembourg	~LU	110
Macau	~MO	111
Macedonia	~MK	112
Madagascar	~MG	113
Madeira	~ME	114
Malawi	~MW	115
Malaysia	~MY	116
Maldives	~MV	117
Mali	~ML	118
Malta	~MT	119
Marshall Islands	~MH	120
Martinique	~MQ	121
Mauritania	~MR	122
Mauritius	~MU	123
Mexico	~MX	124
Moldova	~MD	125
Monaco	~MC	126
Montserrat	~MS	127

Morocco	~MA	128
Mozambique	~MZ	129
Myanmar	~MM	130
Namibia	~NA	131
Nepal	~NP	132
Netherlands	~NL	133
Netherlands Antilles	~AN	134
New Caledonia	~NC	135
New Zealand	~NZ	136
Nicaragua	~NI	137
Niger	~NE	138
Nigeria	~NG	139
Norfolk Island	~NF	140
Northern Ireland	~NB	141
Northern Mariana Islands	~MP	142
Norway	~NO	143
Oman	~OM	144
Pakistan	~PK	145
Palau	~PW	146
Panama	~PA	147
Papua New Guinea	~PG	148
Paraguay	~PY	149
Peru	~PE	150
Philippines	~PH	151
Poland	~PL	152
Ponape	~PO	153
Portugal	~PT	154
Puerto Rico	~PR	155
Qatar	~QA	156
Republic of Ireland	~IE	157
Republic of Yemen	~YE	158
Reunion	~RE	159
Romania	~RO	160
Rota	~RT	161
Russia	~RU	162
Rwanda	~RW	163
Saba	~SS	164
Saipan	~SP	165
Saudi Arabia	~SA	166
Scotland	~SF	167
Senegal	~SN	168
Seychelles	~SC	169
Sierra Leone	~SL	170
Singapore	~SG	171
Slovakia	~SK	172
Slovenia	~SI	173
Solomon Islands	~SB	174
South Africa	~ZA	175
South Korea	~KR	176
Spain	~ES	177
Sri Lanka	~LK	178
St. Barthelemy	~NT	179
St. Christopher	~SW	180

St. Croix	~SX	181
St. Eustatius	~EU	182
St. John	~UV	183
St. Kitts & Nevis	~KN	184
St. Lucia	~LC	185
St. Maarten	~MB	186
St. Martin	~TB	187
St. Thomas	~VL	188
St. Vincent & the Grenadines	~VC	189
Sudan	~SD	190
Suriname	~SR	191
Swaziland	~SZ	192
Sweden	~SE	193
Switzerland	~CH	194
Syria	~SY	195
Tahiti	~TA	196
Taiwan	~TW	197
Tajikistan	~TJ	198
Tanzania	~TZ	199
Thailand	~TH	200
Tinian	~TI	201
Togo	~TG	202
Tonga	~TO	203
Tortola	~TL	204
Trinidad & Tobago	~TT	205
Truk	~TU	206
Tunisia	~TN	207
Turkey	~TR	208
Turks & Caicos Islands	~TC	209
Tuvalu	~TV	210
Uganda	~UG	211
Ukraine	~UA	212
Union Island	~UI	213
United Arab Emirates	~AE	214
United Kingdom	~GB	215
United States	~US	216
Uruguay	~UY	217
US Virgin Islands	~VI	218
Uzbekistan	~UZ	219
Vanuatu	~VU	220
Venezuela	~VE	221
Vietnam	~VN	222
Virgin Gorda	~VR	223
Wake Island	~WK	224
Wales	~WL	225
Wallis & Futuna Islands	~WF	226
Western Samoa	~WS	227
Yap	~YA	228
Yemen	~YE	229
Zaire	~ZR	230
Zambia	~ZM	231
Zimbabwe	~ZW	232

 *End*

When customers shop within your Web store, PDG Shopping Cart displays information about products and orders using HTML templates. Although PDG Shopping Cart comes with default HTML templates for you to use, the information provided in this chapter will aid you in creating your own templates, or modifying the existing default templates.

Note: PDG tags for customizing templates will only be populated with PDG Shopping Cart data if they are called by the program (i.e, PDG tags will not display product data on a static page that was not called by a Cart link).

The HTML templates used within PDG Shopping Cart are very similar to normal HTML pages – with one exception: the HTML template files have, embedded in them, special tags which carry the functionality of the Shopping Cart. When PDG Shopping Cart encounters these tags, it replaces the tag with the corresponding information concerning the customer's Cart or order. The information that PDG Shopping Cart displays depends on the tag used.

There are at least twenty-two different HTML templates that can be used by PDG Shopping Cart:

- Basket template
- Basket Empty template
- Add Item template
- Item template
- Checkout template
- Verify template

- Alternate SSL Checkout template
- Alternate SSL Verify template
- Thank You template
- Deny Credit template (when PDG Shopping Cart is configured to interact with Internet payment services)
- Accept Credit template (when PDG Shopping Cart is configured to interact with Internet payment services)
- SSL Thank You template
- Search Results template
- Search Results Empty template
- Category template
- Multiadd template
- Error template
- Verify - Inventory Error template
- ShipCompare template (for use with USPS when comparing rates)
- UPS Track template
- UPS Tracking Result template
- UPS Address Validation template

There are literally dozens of tags that can be used within these templates – many of which are interchangeable. We will discuss which tags may be used when each template is covered in this chapter.

Note: The standard format of the PDG tags is <!--TAG-->. However, if you are using a tag in a url or input tag, it should appear as “!--TAG--”. For example, the following code would display the preadd page for the product when the thumbnail image is clicked:

```
<a href="!--SCRIPT_NAME--?preadd=action&key!--SKU--">
<!--THUMB--></a>
```

Repeat Tags to Display Multiple Products

HTML templates which display multiple products, such as the Search Results template, can incorporate the MultiAdd and OptFullAdd actions. Imagine showing your customers all of the option selections for a single product and allowing them to simply enter quantities and add them all to their Shopping Carts with a single click. PDG Shopping Cart has enabled this and other functions through the use of the following four pairs of tags:

- REPEAT Tags
- REPEAT NUMBER Tags

- OPTIONS Tags
- SEARCH ROW Tags

We will now discuss the usage of each of these sets of tags as well as what templates they may be used in. For a list of the tags that can be used between these new sets of tags, see the section for the corresponding HTML Template.

'REPEAT' Tags

<!--BEGIN_REPEAT--> and <!--END_REPEAT-->

These tags are to be used when displaying a list of different products for which you want to display the same type of information. We will call these the REPEAT tags, for convenience sake. The most common occurrence of the REPEAT tags will be in the Basket template. These tags may also appear on the MultiAdd template or between the 'SEARCH ROW' tags which appear on the Search Results template. The way these tags may be used will be explained in greater detail in each template's section in this chapter.

Instead of using the standard Basket Template tags, you can now format the output and look of the Basket template using the REPEAT tags. Any information that is called between the REPEAT tags will be called for all of the items in the customer's Shopping Cart until no products are left. Inserting an HTML table row between these tags will cause your customer's Shopping Cart to be formatted nicely in a HTML table. Once you have formatted the output of the basket, you can use the standard Basket Template tags outside of the REPEAT tags to call the order's total, discount information, etc.

For example:

```
<!--BEGIN_REPEAT-->

[product data]

<!--END_REPEAT-->

<!--SUBTOTAL-->

<!--ADD_SHIPPING_TOTAL-->
```

This syntax in the Basket template would yield the customer's current Shopping Cart (see the 'Basket HTML Template' section below for a list of the Basket Repeat tags) followed by the subtotal of all of the items, and finally, any additional shipping charges incurred by any of the items in the order. The 'product data' that will be displayed for each product in the basket comes from the Basket Repeat template tags, while these last two items are called by the standard Basket Template tags.

The REPEAT tags work in a similar fashion on the Search Results template and the Item template. For a list of tags that can be used between the REPEAT tags, see the section corresponding to the HTML template that you are working with.

'REPEAT NUMBER' Tags

<!--BEGIN_REPEAT[x]--> and <!--END_REPEAT[y]-->

These tags are to be used when you want to display several instances of the same product on a single page. These tags will generally be utilized in the Item template, and called into the Add Item template, when you want to allow a customer to add several different option combinations of the same product to their Shopping Cart with a single click.

When using the REPEAT NUMBER tags, the [x] and [y] must be replaced with non-negative integers where $x < y$. These numbers are used to tell PDG Shopping Cart how many instances of the product to display. The REPEAT NUMBER tags allow you to create a page on your Web store with a complete product display and 'Add' button for each number between x and y, inclusive.

For example:

```
<!--BEGIN_REPEAT[0]-->
```

```
[data]
```

```
<!--END_REPEAT[3]-->
```

The data that falls between the REPEAT NUMBER tags will be repeated four times (0,1,2,3). These numbers only serve as indices for PDG Shopping Cart to distinguish the product instances on the page. For each instance on the page, then, a customer can select different options and quantities and add them all to his Cart with a single click.

Please see the Item template section to see which tags may be used between the REPEAT NUMBER tags.

'OPTIONS' Tags

<!--BEGIN_OPTIONS--> and <!--END_OPTIONS-->

These tags are used for displaying the options for your Web store's products in a more streamlined manner. The OPTIONS tags can be used in either the Basket template or the Item template. Note that the OPTIONS tags are treated differently in each of these templates. For

more information on how to use the OPTIONS tags, see the sections on the respective templates.

‘SEARCH ROW’ Tags

`<!--BEGIN_SEARCH_ROW-->` and `<!--END_SEARCH_ROW-->`

These tags are used exclusively on the Search Results template. The SEARCH ROW tags are implemented for formatting the Search Results page in a number of different ways. In order to indicate how many search results will appear on each page and in each row on that page, refer to the ‘Search Settings’ section in Chapter 5 of the PDG Shopping Cart User Guide. These settings are kept in the Merchant Administrator.

The REPEAT tags (embedded in the SEARCH ROW tags) will determine the output for each product on the Search Results template. The SEARCH ROW tags themselves can be used to determine how the information is formatted on the Search Results page. See the Search Results template section in this chapter for more information.

Text and CGI Includes

You can include html, text, or other files in your PDG Cart templates, much like an ‘include’ within a server-side html page. This allows for common elements of your web store, such as a navigation bar, header, or footer, to be included throughout your web store templates. The format for this tag is

`<!--#INCLUDE [filename]-->`, where [filename] is the name of the file to be included. The file must be located in your web store’s document filesystem, and specifically not within any PDG support folder (such as PDG_Cart). You can include subfolder information within the filename, however any filename that tries to traverse up folders will be ignored. Note that this only takes a file and embeds its contents verbatim into the template. It does not execute any programs or scripts.



For example, if there is a file named 'header.html' inside the 'Shared' folder of your website, then the INCLUDE statement for your PDG Cart templates would be:

```
<!--#INCLUDE Shared/header.html-->
```

You can instruct PDG Shopping Cart to retrieve information from a webserver (your own or another) to be embedded in a template. This allows you to embed another web page that contains server-side

scripting, such as asp or php, into your templates. The format for this tag is `<!--#CGI [url]-->`, where [url] is the URL, complete or partial, to the cgi program or document to be retrieved from the webserver. The cgi program can be on a regular server, or on an SSL secured server.

Here are some valid CGI examples:

```
<!--#CGI http://www.somewhere.com/cgi/something.cgi-->
```

This retrieves the output of the cgi program 'something.cgi' from 'www.somewhere.com'.

```
<!--#CGI /cgi-bin/cgi-prog?name1=value1-->
```

This retrieves the output of the cgi program 'cgi-prog' from your web server, passing 'name1=value1' to the cgi program.

```
<!--#CGI https://www.yourstore.com/cgi/secured.cgi-->
```

This retrieves the output of the cgi program 'secured.cgi' from 'www.yourstore.com' via an SSL connection.

```
<!--#CGI /footer.asp-->
```

This retrieves the output of the asp page 'footer.asp' from your web server. The web server will execute any asp code in the page.

Note: If you execute scripts, embedding their output in a template, using the CGI tag, then please keep in mind that any HTML header information, such as setting cookies, will be ignored by PDG Shopping Cart.

Basket Template File

The Basket Template File is used when PDG Shopping Cart is called upon to display the contents of the customer's Shopping Cart to the customer. The Basket Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends it to the customer, it searches the HTML template file for special tags. When a tag is encountered, it is replaced with the appropriate information before being sent to the customer's browser.

After receiving and processing the Display action, PDG Shopping Cart uses the Basket HTML template file to display the resulting Shopping Cart.

The following are the standard Basket Template tags:

- `<!--BASKET-->`

- <!--DISPLAYCART-->
- <!--ADDITEMMSG-->
- <!--ADDITEMDATA-->
- <!--REMOVEALLMSG-->
- <!--SUBTOTAL-->
- <!--SUBTOTAL_WITH_DISCOUNT-->
- <!--WEIGHT_TOTAL-->
- <!--NUM_ITEMS-->
- <!--ORDER_SUBTOTAL_DISC-->
- <!--BONUS_PT_TOTAL-->
- <!--ADD_SHIPPING_TOTAL-->
- <!--SEC_SCRIPTNAME-->
- <!--SEC_FORCE_DATA-->
- <!--CHECKOUT_LINK-->
- <!--SECURE_CHECKOUT_LINK-->
- <!--REMOVE_ALL_LINK-->
- <!--RECALC_LINK-->
- <!--BASKET_TEMPLATE-->

PDG Shopping Cart also accommodates the REPEAT and OPTIONS tags in the Basket HTML template file. See the end of this 'Basket Template File' section for more information on how these new tags may be used in the Basket template file.

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

A sample Basket Template File is included with PDG Shopping Cart. The following is an example of the standard Basket template which displays the customer's shopping cart using the <!--BASKET--> tag. See Figure 8-2 for an example of a customized Basket template.

Qty	Description	Unit	Amount	Remove
<input type="text" value="1"/>	Thungamajig Weight: 6.50 lbs. each Bonus Points: 7.00 each size is large weight is cruiserweight	\$14.95	\$14.95	<input type="button" value="Remove"/>
		\$1.00	\$1.00	
Total Shipment Weight: 6.50 lbs. Total Bonus Points: 7.00		Subtotal:	\$15.95	
		<input type="button" value="Empty the cart"/>	<input type="button" value="Recalculate the total"/>	<input type="button" value="Shop some more"/>
		<input type="button" value="Check-out"/>		

Figure 8-1 Standard Basket Display

<!--BASKET--> Tag

This tag tells PDG Shopping Cart to replace the tag with the contents of the customer's current Shopping Cart. This output is formatted in a table. PDG Shopping Cart also embeds "calls" back to itself in the output which allow users to remove items, change the quantities of items, remove all of the items, or to proceed with purchasing the items in their Cart. These calls are in the form of buttons on the bottom of the page.

Because PDG Shopping Cart displays the contents of the customer's Shopping Cart in a table, the customer's Web browser must be capable of displaying tables.

If you have turned on "Place an input box for Global Price Code on the basket page" in the "Pricing Categories" section of the Merchant Administrator, an input box for a pricing code will be placed within this output.

The tag must have its own line in the template file, and must appear as:

```
<!--BASKET-->
```

Note: When PDG Shopping Cart encounters the <!--BASKET--> tag, it displays all of the product and order information. If this tag is used to display the customer's Shopping Cart, the use of the REPEAT tags (and subsequently the Repeat Basket template tags) will not be necessary. Likewise, if you plan on customizing your Basket template with the REPEAT tags, the <!--BASKET--> tag will not be necessary.

<!--DISPLAYCART--> Tag

This tag works essentially the same way as the <!--BASKET--> tag. See above for an explanation. For consistency's sake, it is recommended that you use either one or the other, and not both for displaying your customer's Shopping Cart. The tag must have its own line in the template file, and must appear as:

```
<!--DISPLAYCART-->
```

<!--ADDITEMMSG--> Tag

This tag tells PDG Shopping Cart to replace the tag with a special message only when items are added to the Cart. If items are not being added to the Cart, then this tag is ignored. The text to be displayed should be embedded in the tag, after the ADDITEMMSG and a colon. The tag must appear in the template file as:

<!--ADDITEMMSG [text]-->

In the above example, [text] is the message to be displayed on the Basket page only if items are being added to the Cart.

<!--ADDITEMDATA--> Tag

This tag tells PDG Shopping Cart to replace the tag with a description and quantity of what items are being added to the Cart. If items are not being added to the Cart when it is called to be displayed, then this tag is ignored. The tag must appear in the template file as:

<!--ADDITEMDATA-->

An example of the above two tags used in conjunction might be:

**<!--ADDITEMMSG:The following has been added to your Cart: -->
>
**

<!--ADDITEMDATA-->

<!--REMOVEALLMSG--> Tag

This tag tells PDG Shopping Cart to replace the tag with a special message only when the Shopping Cart is being emptied. If the Cart is not being emptied, then this tag is ignored. The tag must appear in the template file as:

<!--REMOVEALLMSG:[text]-->

In the above example, [text] is the message that is to be displayed only if the Cart is being emptied.

<!--SUBTOTAL--> Tag

This tag tells PDG Shopping Cart to replace the tag with the cost of all of the products in the current Shopping Cart before sales tax, shipping, or order subtotal discounting is applied. This calculation may be used in determining any additional shipping costs or discounts if these have been specified in the Shipping Rules section of the Merchant Administrator (See Chapter 8 of the PDG Shopping Cart User Guide for more information). The tag must appear in the template file as:

<!--SUBTOTAL-->

<!--SUBTOTAL_WITH_DISCOUNT--> Tag

This tag tells PDG Shopping Cart to replace the tag with the cost of all the products in the current Shopping Cart before sales tax or shipping is applied. Unlike the <!--SUBTOTAL--> tag, this tag will reflect the subtotal of the basket including any order subtotal discounts that may be applied. If no discount applies to the order, then this tag will display the same value as the <!--SUBTOTAL--> tag. The tag must appear in the template file as:

<!--SUBTOTAL_WITH_DISCOUNT-->

<!--WEIGHT_TOTAL--> Tag

This tag tells PDG Shopping Cart to replace the tag with the total weight of the order. This is done by adding the weights of all of the quantities of each product in the order. This calculation may be used to determine shipping costs if weight tables have been specified in the 'Shipping Rules' section of the Merchant Administrator (See chapter 8 in the PDG Shopping Cart User Guide for more information). The tag must appear in the template file as:

<!--WEIGHT_TOTAL-->

<!--NUM_ITEMS--> Tag

This tag tells PDG Shopping Cart to replace the tag with the total number of items in the order. This is done by adding the quantities of all of the products in the order. This calculation may be used in determining any additional shipping costs or discounts if specified in the Shipping Rules section of the Merchant Administrator (See Chapter 8 of the PDG Shopping Cart User Guide for more information). The tag must appear in the template file as:

<!--NUM_ITEMS-->

<!--ORDER_SUBTOTAL_DISC--> Tag

This tag tells PDG Shopping Cart to replace the tag with whatever subtotal-based discount may be applied to the order. This calculation is made depending upon the discounts that you defined with the Merchant Administrator. If no discount applies to the order, then this tag is ignored. The tag must appear in the template file as:

<!--ORDER_SUBTOTAL_DISC-->

<!--BONUS_PT_TOTAL--> Tag

This tag tells PDG Shopping Cart to replace the tag with the total number of bonus or “frequent shopper” points accumulated for the order. These frequent shopper points may be used in calculating discounts or credits toward future purchases. If no bonus points are earned from the current order, then this tag is ignored. The tag must appear in the template file as:

```
<!--BONUS_PT_TOTAL-->
```

<!--ADD_SHIPPING_TOTAL--> Tag

This tag tells PDG Shopping Cart to replace the tag with any additional shipping charges that may have been applied to any of the products in the Shopping Cart. This includes additional shipping charges for all quantities of all applicable products. If no additional shipping charges are incurred from any of the products on the order, PDG Shopping Cart will ignore this tag. The tag must appear in the template file as:

```
<!--ADD_SHIPPING_TOTAL-->
```

<!--SEC_SCRIPTNAME--> Tag

This tag tells PDG Shopping Cart to replace the tag with the URL of the PDG Shopping Cart on the secure server. The tag must appear in the template file as:

```
<!--SEC_SCRIPTNAME-->
```

<!--SEC_FORCE_DATA--> Tag

This tag tells PDG Shopping Cart to replace the tag with the cookie information that needs to be passed to your secure server/directory in order to perform a secure checkout. This tag is necessary if you wish to offer a secure checkout option from your custom Basket HTML template. The tag must appear in the template file as:

```
<!--SEC_FORCE_DATA-->
```

<!--CHECKOUT_LINK--> Tag

This tag is replaced with the code for a CheckOut button. Please note that this is the complete code for displaying the button image as defined in the Merchant Administrator as well as the proper URL. The tag must appear in the template file as:

```
<!--CHECKOUT_LINK-->
```

<!--SECURE_CHECKOUT_LINK--> Tag

This tag is replaced with the code for a Secure CheckOut button. Please note that this is the complete code for displaying the button image as defined in the Merchant Administrator as well as the proper URL. The tag must appear in the template file as:

```
<!--SECURE_CHECKOUT_LINK-->
```

<!--REMOVE_ALL_LINK--> Tag

This tag is replaced with the code for a Remove All button, which will remove all items from the basket when clicked. Please note that this is the complete code for displaying the button image as defined in the Merchant Administrator as well as the proper URL. The tag must appear in the template file as:

```
<!--REMOVE_ALL_LINK-->
```

<!--RECALC_LINK--> Tag

This tag is replaced with the code for a Recalc button, which will recalculate the totals (subtotal, weight, etc.) for the items in the basket when clicked. Please note that this is the complete code for displaying the button image as defined in the Merchant Administrator as well as the proper URL. The tag must appear in the template file as:

```
<!--RECALC_LINK-->
```

<!--BASKET_TEMPLATE--> Tag

This tag is replaced with a hidden field that passes a specified template path. If you are accessing the Basket page using a template other than the template specified in your Merchant Administrator, including this tag on the template will ensure that the recalculate and empty cart actions also display the alternate template. The tag must appear in the template file as:

```
<!--BASKET_TEMPLATE-->
```

'REPEAT' Tags

In addition to the standard Basket Template tags, the Basket HTML Template file can also use the REPEAT tags. If you are planning on customizing the layout of the Basket template, you will need to employ the REPEAT tags. These consist of two tags, <!--BEGIN_REPEAT--> and <!--END_REPEAT-->, between which all of

the tags for displaying each product in the basket will be placed. Any information that falls between the REPEAT tags in the Basket HTML template will be repeated for each item in the customer's Shopping Cart, until no items are left.

These tags must have their own lines in the template file, and must appear as:

```
<!--BEGIN_REPEAT-->
```

```
[product data]
```

```
<!--END_REPEAT-->
```

The 'product data' comes from the Repeat Basket tags which are explained below. For more information on how the REPEAT tags work and how they may be used, see the beginning of this chapter.

While the tags will be explained below, here is an example of a Basket template customized using the REPEAT Basket tags:

Quantity	Description	Unit	Amount	Remove
QTY: <input type="text" value="1"/>	a0003LCW: Thingamajig		\$14.95	<input type="button" value="Remove"/>
	size: large	\$0.00		
	weight: cruiserweight	\$1.00		
	Total Price of Options:	\$1.00		
	Price of Product with Options:	\$15.95	\$15.95	
	Discount:	\$0.00	\$0.00	
	Additional Shipping:	\$0.00	\$0.00	
	Weight:	6.50	6.50	
Total Ship Weight: 6.50 lbs.		Subtotal:	\$15.95	
Total Bonus Points: 7.00		Additional Shipping:	\$0.00	
<input type="button" value="Empty the cart"/>	<input type="button" value="Recalculate the total"/>	<input type="button" value="Shop some more"/>	<input type="button" value="Check-out"/>	

Figure 8-2 Customized Basket Template Display

The tags that are allowed to be used between the REPEAT tags in the Basket HTML template will be called the Basket Repeat tags. They are as follows:

<!--SKU--> Tag

This tag is replaced by the SKU or Product Code for the selected product. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--SKU-->
```

<!--QTY--> Tag

This tag is replaced by a text field displaying the quantity of the selected product that is currently in the basket. Note that this is a form

field that can be modified by the customer from the Basket page. The tag must appear in the template file, between the REPEAT tags, as:

<!--QTY_BUTTON--> Tag

This tag is replaced by a 'recalculate' button that cues PDG Shopping Cart to re-figure the price of the order depending upon the updated quantity in the field created by the <!--QTY--> tag, as explained above. The tag must appear in the template file, between the REPEAT tags, as:

<!--QTY_BUTTON-->

<!--INDEX--> Tag

This tag is replaced with the Basket index number assigned by PDG Shopping Cart for the corresponding product. This number is generally used internally and should not be embedded in your Basket template. However if you do decide to display the index number, the tag must appear in the template file, between the REPEAT tags, as:

<!--INDEX-->

<!--BASEPRICE--> Tag

This tag is replaced by the unit price for the selected product as it was entered into your Merchant Administrator (see 'Maintaining Products' in Chapter 6 of the PDG Shopping Cart User Guide for more information). The tag must appear in the template file, between the REPEAT tags, as:

<!--BASEPRICE-->

Note: This tag may be appended with a "C", <!--BASEPRICEC-->. PDG Shopping Cart will replace this tag with the unit price converted to the secondary currency.

<!--QTYPRICE--> Tag

This tag is replaced by the total price of the selected product being ordered. This value is calculated by multiplying the base or unit price by the quantity of the product being ordered. The tag must appear in the template file, between the REPEAT tags, as:

<!--QTYPRICE-->

Note: This tag may be appended with a "C", <!--QTYPRICEC-->. PDG Shopping Cart will replace this tag with the quantity price converted to the secondary currency.

<!--BASEAMOUNT--> Tag

This tag is replaced by the unit price for the product including any selected options. If there are no charges applied to any of the selected options, this amount will reflect the same as <!--BASEPRICE-->. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--BASEAMOUNT-->
```

Note: This tag may be appended with a “C”, <!--BASEAMOUNTC-->. PDG Shopping Cart will replace this tag with the base amount converted to the secondary currency.

<!--QTYAMOUNT--> Tag

This tag is replaced by the total price for the all of the products with their selected options. This value is calculated by multiplying the product’s price including the selected options by the quantity ordered. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--QTYAMOUNT-->
```

Note: This tag may be appended with a “C”, <!--QTYAMOUNTC-->. PDG Shopping Cart will replace this tag with the quantity amount converted to the secondary currency.

<!--BASEADDONS--> Tag

This tag is replaced by the cost of any add-ons or options that may have been included with the selected product. If there are no charges that apply to any of the selected options for the given product, then this tag is ignored. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--BASEADDONS-->
```

Note: This tag may be appended with a “C”, <!--BASEADDONSC-->. PDG Shopping Cart will replace this tag with the add-on cost converted to the secondary currency.

<!--QTYADDONS--> Tag

This tag is replaced by the cost of all of the add-ons or options for the selected product. This value is figured by multiplying the base add-on price by the number of add-ons being ordered (i.e. the quantity of the product being ordered with the add-on). If there are no charges that apply to any of the selected options for the given product, then this tag is ignored. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--QTYADDONS-->
```

Note: This tag may be appended with a “C”, <---QTYADDONSC--->. PDG Shopping Cart will replace this tag with the quantity add-on price converted to the secondary currency.

<!--BASEDISCOUNT--> Tag

This tag is replaced by the “per unit” discount incurred by the selected product when a valid quantity has been ordered. The valid quantity is determined by discount tiers created in the Merchant Administrator. If there is no discount, then this tag is ignored. The tag must appear in the template file, between the REPEAT tags, as:

<!--BASEDISCOUNT-->

Note: This tag may be appended with a “C”, <---BASEDISCOUNTC--->. PDG Shopping Cart will replace this tag with the product’s base discount converted to the secondary currency.

<!--QTYDISCOUNT--> Tag

This tag is replaced by the total discount that may be incurred due to the quantity of the selected product ordered. These discounts are configured in the Product configuration section of the Merchant Administrator (See ‘Maintaining Products’ in Chapter 6 of the PDG Shopping Cart User Guide). If there is no quantity discount that applies, then this tag is ignored. The tag must appear in the template file, between the REPEAT tags, as:

<!--QTYDISCOUNT-->

Note: This tag may be appended with a “C”, <---QTYDISCOUNTC--->. PDG Shopping Cart will replace this tag with the order’s quantity discount converted to the secondary currency.

<!--KEY--> Tag

This tag is replaced by the specific SKU or product code, without any option suffixes, for the selected product. The tag must appear in the template file, between the REPEAT tags, as:

<!--KEY-->

<!--DESC--> Tag

This tag is replaced by the product’s one-line description as defined in the Product Definition in the Merchant Administrator. The tag must appear in the template file, between the REPEAT tags, as:

<!--DESC-->

<!--UNITOFMEASURE--> Tag

This tag is replaced with the unit of measurement associated with the value for the selected product's weight. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--UNITOFMEASURE-->
```

<!--BASEWEIGHT--> Tag

This tag is replaced with the numerical value associated with the above unit of measure for the selected product's weight. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--BASEWEIGHT-->
```

<!--QTYWEIGHT--> Tag

This tag is replaced by the total weight of the given product being ordered. This value is calculated by multiplying the product's base weight by the quantity of the product being ordered. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--QTYWEIGHT-->
```

<!--THUMB--> Tag

This tag is replaced with the thumbnail image for the given product. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--THUMB-->
```

<!--IMG--> Tag

This tag is replaced by the image for the given product. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--IMG-->
```

<!--IMAGE--> Tag

This tag is replaced by the image for the given product. This tag works the same as the <!--IMG--> tag explained above. It is recommended that you only use one of these conventions, not both. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--IMAGE-->
```

<!--BASE_ADD_SHIPPING--> Tag

This tag is replaced by any additional shipping charge that is incurred by the product itself. If there is no additional shipping charge associ-

ated with the product, then this tag is ignored. The tag must appear in the template file, between the REPEAT tags, as:

<!--BASE_ADD_SHIPPING-->

<!--QTY_ADD_SHIPPING--> Tag

This tag is replaced by the total shipping charges incurred from ordering this product. This value can be calculated by multiplying the additional shipping charge associated with the given product by the quantity of the product ordered. If there is no additional shipping charge associated with the product, then this tag is ignored. The tag must appear in the template file, between the REPEAT tags, as:

<!--QTY_ADD_SHIPPING-->

<!--USERDEF#--> Tag

This tag is replaced by any User Defined data that you may have entered for the given product. The # (0-9) indicates which user defined field to display. If there is no user defined data for the given field, then this tag is ignored. The tag must appear in the template file, between the REPEAT tags, as:

<!--USERDEF1-->

Note: There may be as many as ten (0-9) of these tags in your Basket HTML template file.

<!--REMOVE_BUTTON--> Tag

This tag is replaced by a "Remove" button for the selected product. When the customer clicks this button, the Shopping Cart is re-displayed with all quantities of the selected product removed from the basket. The tag must appear in the template file, between the REPEAT tags, as:

<!--REMOVE_BUTTON-->

<!--DESC_OPT--> Tag

This tag is replaced with the option description for the product in the basket. The tag must appear in the template file, between the REPEAT tags, as:

<!--DESC_OPT-->

<!--VALUE_OPT--> Tag

This tag is replaced by the option selection for the selected product in the basket. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--VALUE_OPT-->
```

<!--ORIGINAL_PRICE--> Tag

This tag is replaced by the original price of the current product, before any product discounting, in the basket. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--ORIGINAL_PRICE-->
```

<!--TEXT_ORIGINAL_PRICE--> Tag

This tag is replaced by the original price of the current product, before any product discounting, in the basket, with the label "Original Price:". The tag must appear in the template file, between the REPEAT tags, as:

```
<!--TEXT_ORIGINAL_PRICE-->
```

<!--DISCOUNT_AMOUNT--> Tag

This tag is replaced by the discounted amount of the current product in the basket. The tag must appear in the template file, between the REPEAT tags, as:

```
<!--DISCOUNT_AMOUNT-->
```

<!--TEXT_DISCOUNT_AMOUNT--> Tag

This tag is replaced by the discount amount of the current product in the basket, with the label "Discount Of:". The tag must appear in the template file, between the REPEAT tags, as:

```
<!--TEXT_DISCOUNT_AMOUNT-->
```

'OPTIONS' Tags

If you are planning on customizing the layout of the Basket template, you will need to employ the REPEAT tags. See above if you have not already done so.

In order to display options for a product using the REPEAT tags, you must also use the OPTIONS tags inside of the REPEAT tags. These consist of two tags, `<!--BEGIN_OPTIONS-->` and `<!--END_OPTIONS-->`, between which all of the tags for displaying the preselected options will be placed. Any information that falls between the OPTIONS tags in the Basket HTML template calls the specific

option information that has been selected by the customer for the corresponding product. The information that is called is repeated for all of the available options for the given product.

These tags must appear in the HTML template file as:

<!--BEGIN_OPTION-->

[option data]

<!--END_OPTION-->

The 'option data' comes from the Basket Options tags that work with the OPTIONS tags within the Basket HTML template in which they are embedded. For more information on how the OPTIONS tags work, see the beginning of this chapter.

The tags that are allowed to be used between the OPTIONS tags in the Basket HTML template are as follows:

<!--DESC--> Tag

This tag is replaced by the name of the option for the given product. The tag must appear in the template file, between the OPTIONS tags, as:

<!--DESC-->

<!--VALUE--> Tag

This tag is replaced by the customer-selected value for the corresponding option of the given product. The tag must appear in the template file, between the OPTIONS tags, as:

<!--VALUE-->

<!--PRICE--> Tag

This tag is replaced with the cost reflected by the particular option selection for the given product. If there is no additional cost for the option selection, then this tag is ignored. The tag must appear in the template file, between the OPTIONS tags, as:

<!--PRICE-->

<!--MNGRMTEXT--> Tag

If the particular product option is for custom monogramming, this tag is replaced by the text selected by the customer to be monogrammed or engraved on the selected product. If there is no monogramming, then

this tag is ignored. The tag must appear in the template file, between the OPTIONS tags, as:

```
<!--MNGRMTEXT-->
```

AddItem Template File

The AddItem Template File is used when PDG Shopping Cart needs to obtain quantity and option information from a user after receiving the PreAdd action. For more information on the PreAdd action, please refer to Chapter 9 of this Technical Reference Guide.

The AddItem Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends it to the customer, it searches the HTML Template File for special tags. When a tag is encountered, it is replaced with the appropriate information before being sent to the customer's Web browser.

The following are the standard AddItem Template tags:

- <!--ITEM-->
- <!--UNITPRICERECALCSCRIPT-->
- <!--OPTIONPRICECALCSCRIPT-->

Note: The <!--UNITPRICERECALCSCRIPT--> and <!--OPTIONPRICECALCSCRIPT--> tags may cause run-time errors when used in conjunction with customized Item templates that utilize the REPEAT, REPEAT NUMBER or OPTIONS tags. This tag only works with the standard Item template style.

The <!--ITEM--> tag adds functionality to the AddItem template by embedding the Item template in the AddItem template.

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

A sample AddItem Template File is included with PDG Shopping Cart.

<!--ITEM--> Tag

PDG Shopping Cart replaces this tag with selectable information about the product to be added. The <!--ITEM--> tag is replaced with data

that is displayed in the Item HTML template. What the Item template can display includes, but is not limited to, the following:

- [Item description]
- [Item price]
- Enter quantity: [Edit box for quantity]
- Option 1 description: [Pull-down box with selections for option 1]
- [More options...]
- [Submit button with the ADD action]

For more information, see the following section on the Item Template File. The tag must have its own line in the template file, and must appear as:

```
<!--ITEM-->
```

<!--UnitPriceRecalcScript--> Tag

This tag gets inserted between the ‘head’ (<head>,</head>) tags and tells PDG Shopping Cart to replace the tag with a dynamically created JavaScript function which enables the real-time price recalculation based on the options that the customer selects on the AddItem page. When this script is inserted, the pricing on the AddItem page will automatically reflect that of any options changed by the customer.

Note that this script updates the unit price for the options selected. The displayed price will not change if the quantity ordered field is changed.

There is also additional Javascript support functions in the AddItem and Item templates for this tag, which are statically written in these templates. For this tag to operate properly, these Javascript functions need to be in the AddItem and Item templates as well.

Note: This tag is case-sensitive.

The JavaScripts located in the default template must remain there. If you create your own template, these scripts must be present if you wish for real-time price calculations on your catalog pages.

The tag must appear in “head” section of the template file as:

```
<!--UnitPriceRecalcScript-->
```

<!--OPTIONPRICECALCSCRIPT--> Tag

This tag gets inserted between the ‘head’ (<head>,</head>) tags and tells PDG Shopping Cart to replace the tag with a dynamically created

JavaScript function which enables the real-time price recalculation based the option(s) that the customer selects on the AddItem page. When this script is inserted, the pricing on the AddItem page will automatically reflect that of any added cost incurred by options chosen by the customer. This script, unlike the UnitPriceRecalcScript tag, also inserts Javascript code to prevent invalid option combinations for the product, as defined in the Merchant Administrator.

The JavaScripts located in the default template must remain there. If you create your own template, these scripts must be present if you wish for real-time price calculations on your catalog pages.

The tag must appear in “head” section of the template file as:

```
<!--OPTIONPRICECALCSCRIPT-->
```

Item Template File

The Item Template File is used to customize the Cart's output on the AddItem template. Essentially, this is an HTML document which gets inserted into the AddItem template.

As with all other templates, before it is sent to the customer, PDG Shopping Cart searches the Item template for special tags. When a tag is encountered, it is replaced with the appropriate information before being sent to the customer's Web browser.

The following are the standard Item Template tags:

- <!--SKU-->
- <!--DESC-->
- <!--UNITOFMEASURE-->
- <!--PRICE-->
- <!--TEXT-->
- <!--PRCAT-->
- <!--INV-->
- <!--WEIGHT-->
- <!--BONUSPOINT-->
- <!--IMAGE-->
- <!--THUMB-->
- <!--ITEMOPT[#]-->
- <!--ITEMRECALCFIELD-->
- <!--ITEMRETURNURL-->
- <!--USERDEF-->

- <!--INVOPT-->
- <!--DESC_OPT-->
- <!--ORIGINAL_PRICE-->
- <!--TEXT_ORIGINAL_PRICE-->
- <!--DISCOUNT_AMOUNT-->
- <!--TEXT_DISCOUNT_AMOUNT-->
- <!--ADD_LINK-->
- <!--MIN_QTY-->
- <!--MAX_QTY-->
- <!--ITEM_QTY_DISC_TABLE-->
- <!--PROD_CAT_DESC-->
- <!--MIN_QTY_TEXT-->
- <!--MAX_QTY_TEXT-->
- <!--CROSS_SALE-->
- <!--UP_SALE-->

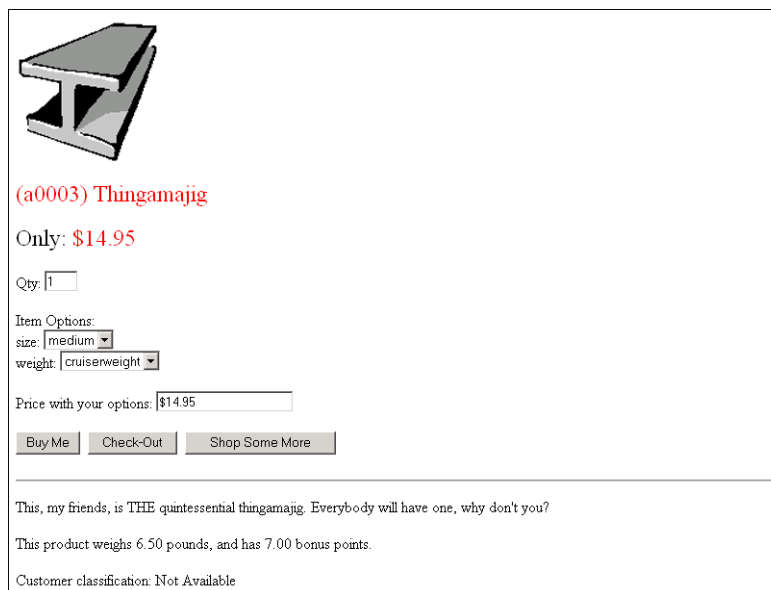
In addition to the tags listed above, several additional tags not containing brackets can appear on this page within “form” tags. These are:

- !--SKU--
- !--IMAGE--
- !--THUMB--
- !--ORIGINAL_PRICE--
- !--TEXT_ORIGINAL_PRICE--
- !--DISCOUNT_AMOUNT--
- !--TEXT_DISCOUNT_AMOUNT--
- !--ADD_LINK--

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

PDG Shopping Cart comes with a sample Item template file. Below is an example of how this template will be displayed. For information on

how to customize this display, see the 'REPEAT NUMBER Tags' section below.



The image shows a sample item template display for a product named "(a0003) Thingamajig". At the top left is a 3D perspective illustration of a grey, I-shaped metal component. Below the illustration, the product name "(a0003) Thingamajig" is displayed in red text. Underneath the name, the price "Only: \$14.95" is shown in red. A quantity input field labeled "Qty:" contains the number "1". Below the quantity field, there are two dropdown menus for "Item Options": "size:" is set to "medium" and "weight:" is set to "cruiserweight". Below these options, a text box displays "Price with your options: \$14.95". At the bottom of the form are three buttons: "Buy Me", "Check-Out", and "Shop Some More". Below the form, there is a horizontal line, followed by three lines of descriptive text: "This, my friends, is THE quintessential thingamajig. Everybody will have one, why don't you?", "This product weighs 6.50 pounds, and has 7.00 bonus points.", and "Customer classification: Not Available".

Figure 8-3 Sample Item Template Display

<!--SKU--> Tag

PDG Shopping Cart replaces this tag with the SKU or product code of the item being displayed. The tag must appear in the template file as:

```
<!--SKU-->
```

<!--DESC--> Tag

PDG Shopping Cart replaces this tag with the One-line description (name) of the item that you entered when initially setting up the product in your Merchant Administrator. The tag must appear in the template file as:

```
<!--DESC-->
```

<!--UNITOFMEASURE--> Tag

This tag is replaced with the unit of measurement associated with the weight value for the selected product. The tag must appear in the template file as:

```
<!--UNITOFMEASURE-->
```

<!--PRICE--> Tag

PDG Shopping Cart replaces this tag with the unit price of the indicated product. The tag must appear in the template file as:

```
<!--PRICE-->
```

Note: This tag may be appended with a “C”, <!--PRICEC-->. PDG Shopping Cart will replace this tag with the unit price converted to the secondary currency.

<!--TEXT--> Tag

PDG Shopping Cart replaces this tag with the product text, or detailed description, of the item being displayed. The tag must appear in the template file as:

```
<!--TEXT-->
```

<!--PRCAT--> Tag

PDG Shopping Cart replaces this tag with the name of the pricing category that is appropriate for the customer at the time the product is “called” (clicking a ‘buy’ button or ‘order’ button that calls PDG Shopping Cart into action and instructing it to display the desired product). If the shopper has no appropriate pricing category code, this tag is ignored. The tag must appear in the template file as:

```
<!--PRCAT-->
```

<!--INV--> Tag

PDG Shopping Cart replaces this tag with the current inventory for the product being displayed. If inventory is being tracked by product options for the selected product, this tag will be replaced by a link that reads ‘Display Inventory by Options.’ If there is none of the selected product in stock, the proper message regarding backorders will be displayed. If inventory is not being tracked for the selected product, this tag is ignored. The tag must appear in the template file as:

```
<!--INV-->
```

<!--WEIGHT--> Tag

PDG Shopping Cart replaces this tag with the weight value assigned to the product during its setup. The tag must appear in the template file as:

<!--WEIGHT-->

<!--BONUSPOINT--> Tag

PDG Shopping Cart replaces this tag with the bonus point (frequent shopper points) value of the item being displayed. The tag must appear in the template file as:

<!--BONUSPOINT-->

<!--IMAGE--> Tag

PDG Shopping Cart replaces this tag with the image associated with the selected item. The tag must appear in the template file as:

<!--IMAGE-->

<!--THUMB--> Tag

PDG Shopping Cart replaces this tag with the thumbnail image associated with the selected item. The tag must appear in the template file as:

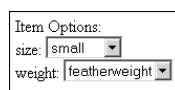
<!--THUMB-->

<!--ITEMOPT[#]--> Tag

PDG Shopping Cart replaces this tag with the product's option description and subsequent drop-down menu of option selections as referenced by that options zero index in the tag. The tag must appear in the template file as:

<!--ITEMOPT[#]-->

The item descriptions and selection boxes will be displayed in the following manner:



Item Options:
 size:
 weight:

Figure 8-4 Product Option Selection Displays on the Item Template

<!--ITEMRETURNURL--> Tag

This tag is replaced by a hidden field. It informs PDG Shopping Cart of the "reference page" for the shopper. The reference page is the page which was being viewed prior to the AddItem page. Note that any URL

entered into the “Destination of Shop Some More Button” field in the Cart Options section of the Merchant Administrator will override this reference page identifier. However, if the page was created dynamically, the user will be returned to the last statically created page.

The tag must appear in the template file as:

`<!--ITEMRETURNURL-->`

`<!--USERDEF[#]-->` Tag

This tag is replaced by any User Defined data that may be associated with the particular product. The # (0-9) indicates which user defined field to display for the selected product. If there is no user defined data for the given field, then this tag will be ignored. The tag must appear in the template file as:

`<!--USERDEF[#]-->`

`<!--INVOPT-->` Tag

This tag is replaced by the current inventory for the selected product. If inventory is being tracked by options, then this value will be displayed. If there is no available inventory in stock, the proper message regarding backorders will be displayed. The tag must appear in the template file as:

`<!--INVOPT-->`

`<!--DESC_OPT-->` Tag

This tag is replaced by the option descriptions associated with the selected product. The tag must appear in the template file as:

`<!--DESC_OPT-->`

`<!--ORIGINAL_PRICE-->` Tag

This tag is replaced by the original price of the item before any discounting. The tag must appear in the template file as:

`<!--ORIGINAL_PRICE-->`

<!--TEXT_ORIGINAL_PRICE--> Tag

This tag is replaced by the original price of them before any discounting, with the label “Original Price:”. The tag must appear in the template file as:

```
<!--TEXT_ORIGINAL_PRICE-->
```

<!--DISCOUNT_AMOUNT--> Tag

This tag is replaced by the discounted amount of the item. The tag must appear in the template file as:

```
<!--DISCOUNT_AMOUNT-->
```

<!--TEXT_DISCOUNT_AMOUNT--> Tag

This tag is replaced by the discounted amount of the item, with the label “Discount Of:”. The tag must appear in the template file as:

```
<!--TEXT_DISCOUNT_AMOUNT-->
```

<!--ADD_LINK--> Tag

This tag displays the “Add to Cart” button. The tag must appear in the template file as:

```
<!--ADD_LINK-->
```

<!--MIN_QTY--> Tag

This tag displays the quantity entered in the “Minimum number to order” field in the product details in the Merchant Administrator. The tag must appear in the template file as:

```
<!--MIN_QTY-->
```

<!--MAX_QTY--> Tag

This tag displays the quantity entered in the “Maximum number to order” field in the product details in the Merchant Administrator. The tag must appear in the template file as:

```
<!--MAX_QTY-->
```

<!--ITEM_QTY_DISC_TABLE--> Tag

This tag displays the product's specified item quantity discounts in a formatted table. This tag must appear in the template file as:

```
<!--ITEM_QTY_DISC_TABLE-->
```

<!--PROD_CAT_DESC--> Tag

This tag displays the product's category description, if you are using product categories. The tag must appear in the template file as:

```
<!--PROD_CAT_DESC-->
```

<!--MIN_QTY_TEXT--> Tag

This tag can be used to display specific text along with the value entered in the "Minimum number to order" field in the product details in the Merchant Administrator. The tag must appear in the template file as:

```
<!--MIN_QTY_TEXT [text]-->
```

In the above example, [text] would be replaced with the message to display on the template. You can use the [VALUE] field to display the minimum quantity value in this tag, such as in the following example:

```
<!--MIN_QTY_TEXT You must order at least [VALUE]-->
```

<!--MAX_QTY_TEXT--> Tag

This tag can be used to display specific text along with the value entered in the "Maximum number to order" field in the product details in the Merchant Administrator. The tag must appear in the template file as:

```
<!--MAX_QTY_TEXT [text]-->
```

In the above example, [text] would be replaced with the message to display on the template. You can use the [VALUE] field to display the maximum quantity value in this tag, such as in the following example:

```
<!--MAX_QTY_TEXT You may not order more than [VALUE]-->
```

<!--CROSS_SALE--> Tag

This tag will display links for all items listed in the Cross Sale section of the product's details in the Merchant Administrator. The tag must appear in the template files as:

```
<!--CROSS_SALE-->
```

<!--UP_SALE--> Tag

This tag will display links for all items listed in the Cross Sale section of the product's details in the Merchant Administrator. The tag must appear in the template files as:

```
<!--UP_SALE-->
```

'REPEAT NUMBER' Tags

When the Item template is embedded in the Add Item template, you may find it convenient to use the REPEAT NUMBER tags. This yields an Item page that displays multiple entries for a single product and its options. This layout allows customers to select several different option combinations for the product, enter a quantity for each, and add them all to their Shopping Cart with a single click.

The tags that may be used between the REPEAT NUMBER tags within the Item template are the same as the standard Item Template tags listed above.

Warning: Please be aware that if you use the REPEAT NUMBER tags on the Item template, any Item Template tags outside of these REPEAT NUMBER tags will cause errors in PDG Shopping Cart.

These tags must have their own lines in the template file, and must appear as:

```
<!--BEGIN_REPEAT[x]-->
```

```
[item data]
```

```
<!--END_REPEAT[y]-->
```

In the example above, x and y are integers (where $x < y$) and the 'item data' is called by the standard Item Template tags. The same types of product data will be displayed for each number between x and y , inclusively.

Note: If more than one set of REPEAT NUMBER tags are used in the Item HTML template, you must make sure that the spans (x,y) DO NOT overlap.

For more information on how the REPEAT NUMBER tags work, please see the beginning of this chapter. Figure 8-5 shows an example of an Item template that utilizes the REPEAT NUMBER tags.

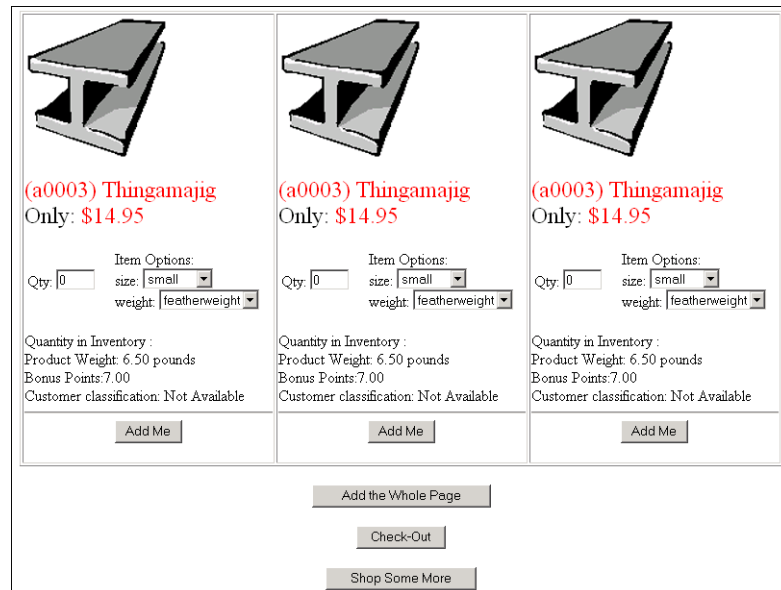


Figure 8-5 Custom Item Template Display

From this Item template, a customer can choose quantities of up to three different option combinations and add them to his Shopping Cart with a single click.

‘OPTIONS’ Tags

You have seen the OPTIONS tags used to repeat the option information for several different products in the Basket HTML template. The OPTIONS tags may also be used, with slightly different functionality, in the Item HTML template.

When used in the Item HTML template, the OPTIONS tags will repeat the product listing for all of the possible option combinations. Therefore, if a product has two options each with three selections, there would be nine product instances on the Item page, each with a unique option combination. Each listing will be accompanied by a ‘Quantity’ field and an ‘Add’ button for easy shopping by your customers.


These tags must appear in the HTML template file as:

```
<!--BEGIN_OPTIONS-->
```

[option data]

<!--END_OPTIONS-->

The 'option data' comes from the Options Item tags that work with the Item HTML template. For more information on how the OPTIONS tags work, see the beginning of this chapter. Figure 8-6 illustrates how the Item page may look for an item that has two options with three possible selections each.



(a0003) Thingamajig

Full SKU	size	weight	Inventory	Price	Quantity	
a0003SFW	small	featherweight		\$14.95	0	Add Selection to Cart
a0003MFW	medium	featherweight		\$14.95	0	Add Selection to Cart
a0003LFW	large	featherweight		\$14.95	0	Add Selection to Cart
a0003SCW	small	cruiserweight		\$15.95	0	Add Selection to Cart
a0003MCW	medium	cruiserweight		\$15.95	0	Add Selection to Cart
a0003LCW	large	cruiserweight		\$15.95	0	Add Selection to Cart
a0003SHW	small	heavyweight		\$16.45	0	Add Selection to Cart
a0003MHW	medium	heavyweight		\$16.45	0	Add Selection to Cart

Add the Whole Page

This, my friends, is THE quintessential thingamajig. Everybody will have one, why don't you?

This product weighs 6.50 pounds, and has 7.00 bonus points.

Customer classification: Not Available

Figure 8-6 Custom Item Template Display

Note: The <!--UNITPRICERECALCSCRIPT--> and <!--OPTION-PRICECALCSCRIPT--> tags may cause run-time errors when used in conjunction with customized Item templates that utilize the REPEAT, REPEAT NUMBER or OPTIONS tags. This tag only works with the standard Item template style.

The tags that are allowed to be used between the OPTIONS tags in the Item HTML template (i.e. the Options Item tags) are as follows:

<!--OPT_LINE_PRICE--> Tag

This tag is replaced by the price of the product before any additional option charges have been applied. The tag must appear in the template file, between the OPTIONS tags, as:

<!--OPT_LINE_PRICE-->

<!--OPT_LINE_ADDED_PRICE--> Tag

This tag is replaced by any added price that accompanies the given option selections. The tag must appear in the template file, between the OPTIONS tags, as:

<!--OPT_LINE_ADDED_PRICE-->

<!--OPT_LINE_SKU--> Tag

This tag is replaced by the specific product code for the product with the given option combination. This product code includes any SKU suffixes that accompany that product's option selections. The tag must appear in the template file, between the OPTIONS tags, as:

<!--OPT_LINE_SKU-->

<!--DESC_OPT--> Tag

This tag is replaced by the names or descriptions given to the options for the particular product. The tag must appear in the template file, between the OPTIONS tags, as:

<!--DESC_OPT-->

Note: This tag may be appended with a number, **<!--DESC_OPT[#]-->**. PDG Shopping Cart will replace this tag with only the option description that corresponds to that option's index number. If an option doesn't exist for that index number, the tag will be ignored.

<!--VALUE_OPT--> Tag

This tag is replaced by the particular option selection for the given product entry. The tag must appear in the template file as:

<!--VALUE_OPT-->

Note: This tag may be appended with a number, **<!--VALUE_OPT[#]-->**. PDG Shopping Cart will replace this tag with only the option selection that corresponds to that option's index number. If an option doesn't exist for that index number, the tag will be ignored.

<!--OPT_LINE_INV--> Tag

This tag is replaced by the physical inventory for the product with the given option combination. If there is no inventory, this line will indicate whether backordering is allowed on the selected item. The tag must appear in the template file, between the OPTIONS tags, as:

<!--OPT_LINE_INV-->

<!--OPT_LINE_QTY--> Tag

This tag is replaced by the 'name' portion of the HTML text input that will contain the quantity of the product to be added. This tag is important for accurately indexing the product's option combinations on the template. The tag must appear within an HTML input tag, between the OPTIONS tags, as:

```
<input name="!---OPT_LINE_QTY---" value="0">
```

<!---OPT_LINE_ADD---> Tag

This tag is replaced by the 'name' portion of the HTML submit button that is used for changing quantities. This tag is important for accurately indexing the product's option combinations on the template. The tag must appear within an HTML input tag, between the OPTIONS tags, as:

```
<input type="submit" name="!---OPT_LINE_ADD---" value="[text to display]">
```

This will create a button that will add the corresponding quantity entered in the "OPT_LINE_QTY" text field, above, to the shopper's Cart.

MultiAdd Template File

The MultiAdd Template File is used whenever the MultiPreAdd action is called. The MultiAdd template is used to allow customers to add several products to their Shopping Cart with a single click.

The MultiAdd Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends it to the customer's browser, it searches the file for special tags. When PDG Shopping Cart encounters these tags, they are replaced with the appropriate information before being sent to the customer's Web browser.

In order for PDG Shopping Cart to properly receive and process information from this template file, the MultiAdd Template File must contain a form submission back to PDG Shopping Cart with the MultiAdd action. For all the required fields for this action, please see the MultiAdd action definition in Chapter 9 of this Technical Reference Guide.

The tags used with the MultiAdd Template are the same as those that can be used with the REPEAT tags in the Item template. However, it is imperative that you use the REPEAT tags on the MultiAdd Template, as this provides the multiple listing which makes MultiAdd a special feature. For more information, see the Repeat Item template tags in the 'Item Template File' section, above.

Below are two examples of how the MultiAdd template file may be displayed. Please see the section on REPEAT Tags at the beginning of this chapter for more information on how to customize this template.

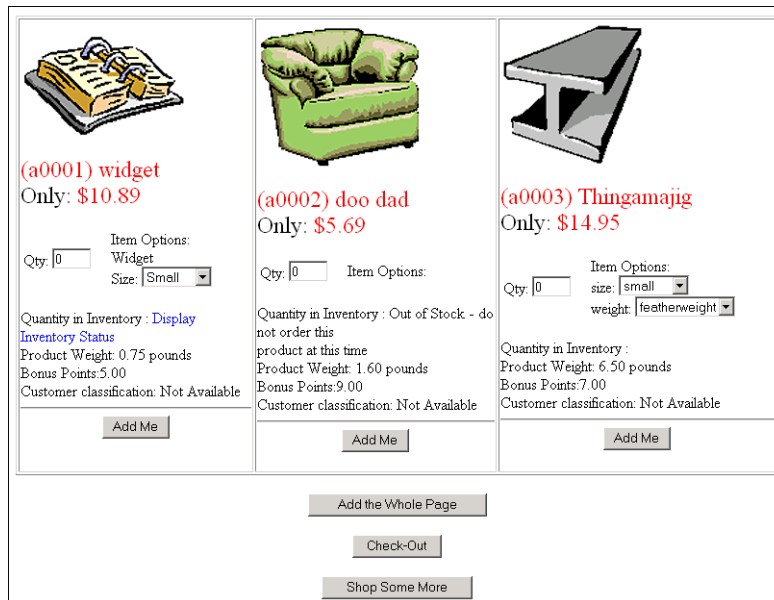


Figure 8-7 Sample MultiAdd Template Display

The example in Figure 8-8 is essentially the same, however it contains HTML 'table row' tags (<tr>, </tr>) within the REPEAT tags. Figure 8-8 also contains thumbnail images instead of the full-size images.

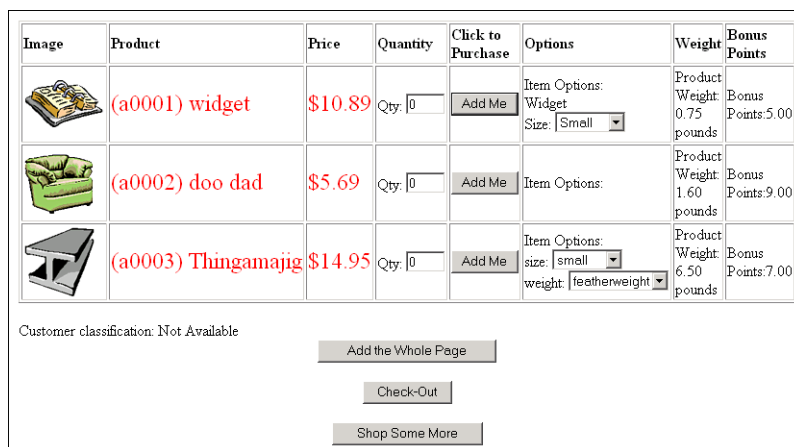


Figure 8-8 Sample MultiAdd Template Display

Note: PDG Shopping Cart 2004's "locked options" feature will not be enabled when the MultiAdd template is called from the MultiPreAdd action. Please see 'Maintaining Products' of the PDG Shopping Cart User Guide for more information on locked options.

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

Checkout Template File

The Checkout Template File is used when PDG Shopping Cart needs to obtain the customer's contact, billing, and shipping information from a user after receiving the Checkout action (usually from the Basket page). For more information on the Checkout action, please see Chapter 9 of this Technical Reference Guide.

The Checkout Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends it to the customer, it searches the HTML Template File for special tags. When PDG Shopping Cart encounters these tags, they are replaced with the appropriate information before being sent to the customer's Web browser.

In order for PDG Shopping Cart to properly receive and process information from this template file, the Checkout Template File must contain a form submission back to PDG Shopping Cart with the Verify action. For all the required fields for this callback, please see the Verify action definition in Chapter 9 of this Technical Reference Guide.

The following are the standard Checkout Template tags:

- <!--ORDER-->
- <!--SHIPPINGMETHOD-->
- <!--COUNTRIES-->
- <!--BILLCOUNTRY-->
- <!--SHIPCOUNTRY-->
- <!--DISPLAYCART-->
- <!--DISPLAYFORCECOOKIE-->
- <!--PDGCARTFORMDATA-->
- <!--SUBTOTAL-->
- <!--SUBTOTAL_WITH_DISCOUNT-->
- <!--WEIGHT_TOTAL-->
- <!--NUM_ITEMS-->
- <!--ORDER_SUBTOTAL_DISC-->
- <!--BONUS_PT_TOTAL-->
- <!--ADD_SHIPPING_TOTAL-->
- !-- SUBMIT_LINK--

- `<!--SEC_FORCE_DATA-->`

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

A sample Checkout Template File is included with PDG Shopping Cart. Below is an example of how this Checkout Template will be displayed. Please see the section on REPEAT Tags at the beginning of this chapter for more information on how to customize this template.

You are ordering:			
Qty	Description	Unit	Amount
(1)	Thingamajig Weight: 6.50 lbs. each Bonus Points: 7.00 each size is large weight is cruiserweight	\$14.95	\$14.95
Total Shipment Weight: 6.50 lbs.		Subtotal:	\$15.95
Total Bonus Points: 7.00			

Please provide us with the following information so we may process your order promptly. If any items are missing, we may not be able to process your order. Most importantly, please ensure your email address is correct, as this is how we will notify you if any problems occur while processing your order.

When you have finished, please click the 'Continue' button to continue submitting your order.

Contact Information

Please ensure this information is correct. We have no other method of notifying you in case of order problems.

Name:

Email:

Billing Information

If paying by credit card, then your billing name should match the name on the credit card and your billing address should match your credit card's billing address.

First Name:

Last Name:

Street Address line 1:

Street Address line 2:

City:

State/Province (US and Canada only):

Other (for international use only):

Postal Code:

Country:

Shipping Information

Shipping Method:

Please ship this order to the billing address.

Or ship to:

Name:

Street Address line 1:

Street Address line 2:

City:

State/Province (US and Canada only):

Other (for international use only):

Postal Code:

Country:

Figure 8-9 Sample Checkout Template Display

<!--ORDER--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with the customer's current Shopping Cart. Unlike the Shopping Cart display used in the Basket template file, this display does not allow the customer to change product quantities. While it is still formatted in a table, the total displayed for the order does not contain any additional ship-

ping charges or sales tax. These totals can be added to the Checkout template with some of the tags explained later.

The tag must have its own line in the template file, and must appear in the template file as:

```
<!--ORDER-->
```

<!--SHIPPINGMETHOD--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with a drop-down menu containing all the available shipping methods as defined in your Merchant Administrator.

This tag must appear inside the template as.

```
<!--SHIPPINGMETHOD-->
```

This field is almost always required. If you are planning to remove this field (for instance, if you have only one shipping method), you will need to replace it with an input field like the one below:

```
<input type="hidden" name="shipmethod" value="0">
```

<!--COUNTRIES--> Tag

When PDG Shopping Cart encounters this tag, it replaces it with a list of HTML “option” form inputs for all of the available countries. This tag must be inserted between HTML “select” tags in the template and must appear as:

```
<select name="country">
```

```
<!--COUNTRIES-->
```

```
</select>
```

This syntax will produce a drop-down menu which lists all of the available countries in PDG Shopping Cart.

<!--STATES--> Tag

When PDG Shopping Cart encounters this tag, it replaces it with a list of HTML “option” form inputs for all of the available states. This tag must be inserted between HTML “select” tags in the template and must appear as:

```
<select name="country">
```

```
<!--STATES-->
```

```
</select>
```

This syntax will produce a drop-down menu which lists all of the available U.S. states and Canadian provinces.

<!--BILLCOUNTRY--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with a drop-down menu containing all the countries with their corresponding numeric code for PDG Shopping Cart. The value selected by the customer may later be used for UPS real-time cost capturing, as well as determining the applicable shipping, handling, and tax rules. The name of the drop-down menu is 'billcountry' and is used specifically to generate the 'billcountry' form input.

This tag must be inside the form submission to PDG Shopping Cart with the Verify action.

```
<!--BILLCOUNTRY-->
```

This field can be removed, but must be replaced with its HTML equivalent. For instance, if you only offer shipping to the United States, this field would look like:

```
<input type="hidden" name="billcountry" value="461">
```

If you wanted to offer shipping to the US and Canada:

```
<select name="billcountry">
```

```
<option value="461">United States
```

```
<option value="69">Canada
```

```
</select>
```

A list of the country codes you would need to reference for this type of syntax can be obtained from PDG.

<!--SHIPCOUNTRY--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with a drop-down menu containing all the countries with their corresponding numeric codes for PDG Shopping Cart. This code may later be used for UPS real-time cost capturing, as well as determining the applicable shipping, handling, and tax rules. The name of the drop down menu is 'shipcountry' and is used to specifically generate the shipcountry form input.

This tag must be inside the form submission to PDG Shopping Cart with the Verify action. The tag must have its own line in the template file, and must appear in the template file as:

```
<!--SHIPCOUNTRY-->
```

<!--DISPLAYCART--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with the customer's current Shopping Cart. Unlike the Shopping Cart display used in the Basket Template File, this display does not allow the customer to change product quantities. While it is still formatted in a table, the total displayed for the order does not contain any additional shipping charges or sales tax.

The tag must have its own line in the template file, and must appear in the template file as:

```
<!--DISPLAYCART-->
```

Note: This tag works essentially the same as the <!--ORDER--> tag, explained above.

<!--DISPLAYFORCECOOKIE--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with the cookie that has been appended to the shopper's Web browser. Forced cookies are only needed if the customer's Cart is being directed to a different server (i.e. a secure server). This tag must appear in the template as:

```
<!--DISPLAYFORCECOOKIE-->
```

<!--PDGCARTFORMDATA--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with hidden information that is necessary for the checkout process. This tag must be located within the form submission in the Checkout HTML template file. No visible information is replaced by this tag, therefore, it does not affect the visible layout of your HTML template in any way.

This tag is required for the Checkout Template, and for PDG Shopping Cart to operate properly. This tag must appear inside the form submission to PDG Shopping Cart with the Verify action. The tag must appear in the template file as:

```
<!--PDGCARTFORMDATA-->
```

<!--SUBTOTAL--> Tag

This tag is replaced by the subtotal for the order in the customer's current Shopping Cart. This value represents just the total price of all of the products in the Cart. Sales tax, shipping, and discounts have not been considered. The tag must appear in the template file as:

```
<!--SUBTOTAL-->
```

<!--SUBTOTAL_WITH_DISCOUNT--> Tag

This tag tells PDG Shopping Cart to replace the tag with the cost of all the products in the current Shopping Cart before sales tax or shipping is applied. This tag will reflect the subtotal of the basket including any order subtotal discounts that may be applied. If no discount applies to the order, then this tag will display the same value as the <!--SUBTOTAL--> tag. The tag must appear in the template file as:

```
<!--SUBTOTAL_WITH_DISCOUNT-->
```

<!--WEIGHT_TOTAL--> Tag

This tag is replaced by the total weight of all of the products in the current Shopping Cart. This value may be used to calculate shipping charges if weight tables have been specified in the 'Shipping Rules' section of the Merchant Administrator (See chapter 8 in the PDG Shopping Cart User Guide). The tag must appear in the template file as:

```
<!--WEIGHT_TOTAL-->
```

<!--NUM_ITEMS--> Tag

This tag is replaced by the total number of items in the current Shopping Cart. This value may be used to calculate shipping charges if quantity tables have been specified in the 'Shipping Rules' section of the Merchant Administrator (See the **PDG Shopping Cart User Guide** for more information). The tag must appear in the template file as:

```
<!--NUM_ITEMS-->
```

<!--ORDER_SUBTOTAL_DISC--> Tag

This tag tells PDG Shopping Cart to replace the tag with whatever subtotal-based discount may be applied to the order. This calculation is made depending upon the discounts you defined in the Merchant Administrator (See the **PDG Shopping Cart User Guide** for more

information). If no discount applies to the order, then this tag is ignored.

The tag must appear in the template file as:

```
<!--ORDER_SUBTOTAL_DISC-->
```

<!--BONUS_PT_TOTAL--> Tag

This tag is replaced by the total number of bonus or 'frequent shopper' points awarded to the current order. If no bonus points have been earned by the products in the order, then this tag is ignored. The tag must appear in the template file as:

```
<!--BONUS_PT_TOTAL-->
```

<!--ADD_SHIPPING_TOTAL--> Tag

When PDG Shopping Cart encounters this tag, it is replaced with any additional shipping charges that may have been applied to any of the products in the current order. If no additional shipping charges are incurred from any of the products in the order, PDG Shopping Cart will ignore this tag. The tag must appear in the template file as:

```
<!--ADD_SHIPPING_TOTAL-->
```

!--SUBMIT_LINK-- Tag

PDG Shopping Cart replaces this with the submit button for the form, which will send shipping and billing information to the cart and proceed with collecting payment information. The tag must appear in the template file as:

```
!--SUBMIT_LINK--
```

<!--SEC_FORCE_DATA--> Tag

This tag tells PDG Shopping Cart to replace the tag with the cookie information that needs to be passed to your secure server/directory in order to perform a secure checkout. This tag is necessary if you wish to offer a secure checkout option from your custom Basket HTML template. The tag must appear in the template file as:

```
<!--SEC_FORCE_DATA-->
```

ShipText and BillText

You can provide blank form inputs on your Checkout template to allow customers to provide you with any special shipping or billing instructions. The field for additional billing information must be named "billtext." The field for additional shipping information must be named "shiptext." The text in these form fields will be saved by PDG Shopping Cart and can be optionally displayed on the Verify and ThankYou templates.

Basket Template Tags

In addition to the standard Checkout Template tags, the Checkout HTML Template can also take any of the standard Basket Template tags. The standard Basket Template tags allow you to customize the display of the customer's Shopping Cart on the Checkout page.

'REPEAT' and 'OPTIONS' Tags

Like the Basket template, the Checkout Template can also utilize the REPEAT and OPTIONS tags. The tags used within the REPEAT and OPTIONS tags for the Checkout HTML template are the same as those used in the Basket HTML template (i.e. the Repeat Basket tags and the Options Basket tags). For information on what tags can be used with the REPEAT and OPTIONS tags in the Checkout Template, please see the 'Basket Template File' section of this chapter.

Passthrough Fields

Any form field whose name begins with the four letters "PASS" will have its values automatically forwarded to the final ThankYou templates and emails. For example:

```
<input type="text" name="PASSPhone" size="20">
```

The example above will create a phone number field in this template for customer input. Whatever the user types in this field will automatically be forwarded to the notification e-mails, ThankYou template, and order log file.

Verify Template File

The Verify Template File is used when PDG Shopping Cart needs to verify a customer's contact, billing, and shipping information. It is also

used to collect payment information from a customer after receiving the Verify action.

The Verify Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends it to the customer's browser, it searches the file for special tags. When a tag is encountered, it is replaced with the appropriate information before being sent to the customer's Web browser.

In order for PDG Shopping Cart to properly receive and process information from this template file, the Verify Template File must contain a form submission back to PDG Shopping Cart with the BuyIt action. For all the required fields for this callback, please see the BuyIt action definition in Chapter 9 of this Technical Reference Guide.

The following are the standard Verify Template tags:

- <!--CUSTOMER-->
- <!--BILLTO-->
- <!--SHIPTO-->
- <!--PDGCARTFORMDATA-->
- <!--BILLMETHOD-->
- <!--BILLNAME-->
- <!--BILLADDRESS-->
- <!--BILLCITY-->
- <!--BILLSTATE-->
- <!--BILLZIP-->
- <!--BILL_PROV-->
- <!--BILLCOUNTRY-->
- <!--BILLEMAIL-->
- <!--BILLPHONE1-->
- <!--BILLPHONE2-->
- <!--BILLAMOUNT-->
- <!--BILLDATE-->
- <!--ORDER-->
- <!--SHIPNAME-->
- <!--SHIPADDRESS-->
- <!--SHIPCITY-->
- <!--SHIPSTATE-->
- <!--SHIPZIP-->
- <!--SHIPPROV-->
- <!--SHIPCOUNTRY-->
- <!--SUBTOTAL-->
- <!--WEIGHT_TOTAL-->
- <!--NUM_ITEMS-->

- <!--ORDER_SUBTOTAL_DISC-->
- <!--BONUS_PT_TOTAL-->
- <!--ADD_SHIPPING_TOTAL-->
- <!--TOTAL-->
- <!--TAX-->
- <!--SHIPPING_TOTAL-->
- <!--FLUX_AMOUNT-->
- <!--PRE_FLUX_TOTAL-->
- <!--SUBMIT_LINK-->
- <!--SUBTOTAL_WITH_DISCOUNT-->
- <!--SHIP_METHOD-->
- <!--BILLTEXT-->
- <!--SHIPTEXT-->

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

A sample Verify Template File is included with PDG Shopping Cart. Here is an example of how that template may be displayed. Please see the sections below on 'Basket Tags' and 'REPEAT Tags' for more information on how to customize this template.

Please verify that the following information is correct. If it is not, then use your browser's back arrow and make the necessary changes. After you have verified that the information is correct, please supply the necessary payment information to place your order.

Qty	Description	Unit	Amount
(1)	Thingamajig Weight: 6.50 lbs. each Bonus Points: 7.00 each size is large weight is cruiserweight	\$14.95	\$14.95
Total Shipment Weight: 6.50 lbs.		Subtotal:	\$15.95
		Order Discount:	\$1.00
		Shipping:	\$5.00
		Total:	\$19.95

Total Bonus Points: 7.00

Customer Information

Billing To: John Customer 1234 Meadow Ln. Springdale, ID 01010 United States	Shipping To: John Customer 1234 Meadow Ln. Springdale, ID 01010 United States Deliver via USPS Book Rate
---	--

Please Provide Payment Information

Method of payment

Credit Card Information
(Fill out only if paying by credit card on-line.)

(Credit Card) Type of Credit Card

Commercial Credit Card

(Credit Card) Credit Card Number

(Credit Card) Expiration Date

Credit Card Verification Value: Number is not embossed.
American Express is 4 digits (located on front of card),
for all other cards it is 3 digits (normally on the signature strip).

Check Information
(Fill out only if paying by check on-line.)

(Check) Bank Name

(Check) Routing #

(Check) Check #

(Check) Account #

(Check) Driver License #

(Check) Driver License State (US only)

(Check) Telephone # (no spaces)

Figure 8-10 Sample Verify Template Display

<!--CUSTOMER--> Tag

This tag is replaced by all of the customer's complete billing and shipping information as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--CUSTOMER-->
```

Note: If the <!--CUSTOMER--> tag is used, then none of the following tags relating to the customer's address need to be implemented.

<!--BILLTO--> Tag

When PDG Shopping Cart encounters this tag, the tag is replaced with the customer's billing information. This includes the name and complete billing address as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--BILLTO-->
```

Note: If the <!--BILLTO--> tag is used, then none of the following tags relating to the customer's billing address need to be implemented.

<!--SHIPTO--> Tag

When PDG Shopping Cart encounters this tag, the tag is replaced with the customer's shipping information. This includes the name and complete shipping address as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--SHIPTO-->
```

Note: If the <!--SHIPTO--> tag is used, then none of the following tags relating to the customer's shipping address need to be implemented.

<!--PDGCARTFORMDATA--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with hidden information that is necessary for the checkout process. This tag must be located within the form submission in the Verify HTML template file. No visible information is replaced by this tag, therefore, it will not affect the visible layout of your Web store in any way.

This tag is required for the Verify Template, and for PDG Shopping Cart to operate properly. This tag must appear inside the form submission to PDG Shopping Cart with the Verify action. The tag must appear in the template file as:

```
<!--PDGCARTFORMDATA-->
```

<!--BILLMETHOD--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by a pull-down select box, allowing the customer to choose a payment method. The payment methods are defined by the store merchant in the Merchant Administrator. The tag must appear in the template file as:

```
<!--BILLMETHOD-->
```

<!--BILLNAME--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the billing name as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--BILLNAME-->
```

<!--BILLADDRESS--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the billing address as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--BILLADDRESS-->
```

<!--BILLCITY--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the billing city as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--BILLCITY-->
```

<!--BILLSTATE--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the billing state as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--BILLSTATE-->
```

<!--BILLZIP--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the billing ZIP code as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--BILLZIP-->
```

<!--BILLPROV--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the billing province as supplied by the customer on the Checkout page. If no province was entered, this tag is ignored. The tag must appear in the template file as:

<!--BILLPROV-->

<!--BILLCOUNTRY--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the billing country as supplied by the customer on the Checkout page. The tag must appear in the template file as:

<!--BILLCOUNTRY-->

<!--BILLEMAIL--> Tag

This tag is replaced by the e-mail address to which billing information should be sent. This value should have been entered by the customer on the Checkout page. The tag must appear in the template file as:

<!--BILLEMAIL-->

<!--BILLPHONE1--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the primary phone number as supplied by the customer on the Checkout page. The tag must appear in the template file as:

<!--BILLPHONE1-->

<!--BILLPHONE2--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the secondary phone number as supplied by the customer on the Checkout page. The tag must appear in the template file as:

<!--BILLPHONE2-->

<!--BILLAMOUNT--> Tag

This tag is replaced with the total amount of the order that is to be billed to the customer as well as the payment method specified by the shopper on the Checkout page. This total includes tax, shipping and any additional charges or discounts that may have been applied. The tag must appear in the template file as:

<!--BILLAMOUNT-->

<!--BILLDATE--> Tag

This tag is replaced by the date on which the order is to be billed. This will generally yield the current date. The tag must appear in the template file as:

```
<!--BILLDATE-->
```

<!--ORDER--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with the customer's current Shopping Cart. This information is formatted in a table. Unlike the Shopping Cart display used in the Basket Template File, this display does not allow the customer to change product quantities. Unlike the <!--ORDER--> tag used in the Checkout Template File, this order output contains all additional shipping costs and applicable tax.

The tag must have its own line in the template file, and must appear in the template file as:

```
<!--ORDER-->
```

<!--SHIPNAME--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the name to which the order will be shipped, as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--SHIPNAME-->
```

<!--SHIPADDRESS--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the address to which the order will be shipped, as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--SHIPADDRESS-->
```

<!--SHIPCITY--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the city to which the order will be shipped, as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--SHIPCITY-->
```

<!--SHIPSTATE--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the state to which the order will be shipped, as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--SHIPSTATE-->
```

<!--SHIPZIP--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the ZIP code to which the order will be shipped, as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--SHIPZIP-->
```

<!--SHIPPROV--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the province to which the order will be shipped, as supplied by the customer on the Checkout page. If no province was entered, this tag is ignored. The tag must appear in the template file as:

```
<!--SHIPPROV-->
```

<!--SHIPCOUNTRY--> Tag

When PDG Shopping Cart encounters this tag, it is replaced by the country to which the order will be shipped, as supplied by the customer on the Checkout page. The tag must appear in the template file as:

```
<!--SHIPCOUNTRY-->
```

<!--SUBTOTAL--> Tag

This tag is replaced by the subtotal for the order. This price does not include sales tax, shipping charges or any other charges or discounts. The tag must appear in the template file as:

```
<!--SUBTOTAL-->
```

Note: This tag may be appended with a “C”, <!--SUBTOTALC-->. PDG Shopping Cart will replace this tag with the subtotal of the order converted to the secondary currency.

<!--WEIGHT_TOTAL--> Tag

This tag is replaced by the total weight of all of the products in the current Shopping Cart. This value may be used to calculate shipping charges if weight tables have been specified in the 'Shipping Rules' section of the Merchant Administrator (See Chapter 8 in the PDG Shopping Cart User Guide for more information). The tag must appear in the template file as:

```
<!--WEIGHT_TOTAL-->
```

<!--NUM_ITEMS--> Tag

This tag is replaced by the total number of items in the current Shopping Cart. This value may be used to calculate shipping charges if quantity tables have been specified the 'Shipping Rules' section of the Merchant Administrator (See chapter 8 in the PDG Shopping Cart User Guide for more information). The tag must appear in the template file as:

```
<!--NUM_ITEMS-->
```

<!--ORDER_SUBTOTAL_DISC--> Tag

This tag tells PDG Shopping Cart to replace the tag with whatever subtotal-based discount may be applied to the order. This calculation is made depending upon the discounts you defined in the Merchant Administrator (See Chapter 5 of the PDG Shopping Cart User Guide for more information). If no discount applies to the order, then this tag is ignored. The tag must appear in the template file as:

```
<!--ORDER_SUBTOTAL_DISC-->
```

Note: This tag may be appended with the letter "C", <!--ORDER_SUBTOTAL_DISCC-->. PDG Shopping Cart will replace this tag with the order's subtotal discount converted to the secondary currency.

<!--BONUS_PT_TOTAL--> Tag

This tag is replaced by the total number of bonus or 'frequent shopper' points incurred by the order. If no bonus points are earned by the order, then this tag is ignored. The tag must appear in the template file as:

```
<!--BONUS_PT_TOTAL-->
```

<!--ADD_SHIPPING_TOTAL--> Tag

This tag tells PDG Shopping Cart to replace the tag with any additional shipping charges that may have been applied to any of the products in the current Shopping Cart. If no additional shipping charges are incurred from any of the products on the order, PDG Shopping Cart will ignore this tag. The tag must appear in the template file as:

```
<!--ADD_SHIPPING_TOTAL-->
```

Note: This tag may be appended with the letter “C”, <!--ADD_SHIPPING_TOTALC-->. PDG Shopping Cart will replace this tag with the total additional shipping charges for any products in the order converted to the secondary currency.

<!--TOTAL--> Tag

This tag is replaced by the total cost of the order. This price includes all product and option costs, sales tax, shipping and additional charges placed on the order as well as any discounts that may have been applied to the order. The tag must appear in the template file as:

```
<!--TOTAL-->
```

Note: This tag may be appended with a “C”, <!--TOTALC-->. PDG Shopping Cart will replace this tag with the total cost of the order converted to the secondary currency.

<!--TAX--> Tag

This tag is replaced by the total sales tax placed on the order. If there is no sales tax to include, then this tag is ignored. The tag must appear in the template file as:

```
<!--TAX-->
```

Note: This tag may be appended with a “C”, <!--TAXC-->. PDG Shopping Cart will replace this tag with the order’s total sales tax converted to the secondary currency.

<!--SHIPPING_TOTAL--> Tag

This tag is replaced by the total shipping charges applied to the order. This includes all shipping charges attributed to the weight, subtotal, shipping method or the products themselves. The tag must appear in the template file as:

```
<!--SHIPPING_TOTAL-->
```

Note: This tag may be appended with a “C”, <!--SHIPPING_TOTALC-->. PDG Shopping Cart will replace this tag with the total shipping charges for the order converted to the secondary currency.

<!--FLUX_AMOUNT--> Tag

This tag is replaced by the amount that was added to the order to accommodate changes in currency exchange rates. If a currency conversion has not been made, then this tag is ignored. The tag must appear in the template file as:

```
<!--FLUX_AMOUNT-->
```

Note: This tag may be appended with a “C”, <!--FLUX_AMOUNTC-->. PDG Shopping Cart will replace this tag with the order’s flux amount converted to the secondary currency.

<!--PRE_FLUX_TOTAL--> Tag

This tag is replaced by the total of the order after the amount has been converted to the secondary currency, but before the “Flux Amount” is added. If a currency conversion has not been made, then this tag is ignored. The tag must appear in the template file as:

```
<!--PRE_FLUX_TOTAL-->
```

Note: This tag may be appended with a “C”, <!--PRE_FLUX_TOTALC-->. PDG Shopping Cart will replace this tag with the order’s pre-flux total converted to the secondary currency.

<!--SUBMIT_LINK--> Tag

This tag is replaced by the submit button for the form, which collects the entered payment information and processes the order. The tag must appear in the template file as:

```
<!--SUBMIT_LINK-->
```

<!--SUBTOTAL_WITH_DISCOUNT--> Tag

This tag tells PDG Shopping Cart to replace the tag with the cost of all the products in the current Shopping Cart before sales tax or shipping is applied. This tag will reflect the subtotal of the basket including any order subtotal discounts that may be applied. If no discount applies to the order, then this tag will display the same value as the <!--SUBTOTAL--> tag. The tag must appear in the template file as:

```
<!--SUBTOTAL_WITH_DISCOUNT-->
```

<!--SHIP_METHOD--> Tag

This tag is replaced by the shipping method chosen by the customer. The tag must appear in the template file as:

```
<!--SHIP_METHOD-->
```

<!--BILLTEXT--> Tag

If you provided a field in the Checkout template named “billtext,” then this tag will be replaced with the customer-supplied text from that field. The tag must appear in the template file as:

```
<!--BILLTEXT-->
```

<!--SHIPTEXT--> Tag

If you provided a field in the Checkout template named “shiptext,” then this tag will be replaced with the customer-supplied text from that field. The tag must appear in the template file as:

```
<!--SHIPTEXT-->
```

Payflow Link Tags

If you are using Payflow Link as one of several payment methods, then PDG Shopping Cart will use a redirect page to take the customer to Payflow Link if it was the chosen payment method on the Verify page. This page will be displayed by default. However, if you would like to customize the redirect page, you may create a template and enter its name in the Payflow Link settings of your Merchant Administrator. Your template should contain the following code:

```
<html>

<body onLoad="javascript:setTimeout('document.forms.pdgreddirect.submit()',4000);">

<form action = "--PAYFLOW_LINK_ACTION--" method=POST
name=pdgredirect>

<!--PAYFLOW_ORDER_DATA-->

<center>Please wait while we direct you to VeriSign's secure web
site to

complete your payment.</center>
```

```
<center>Thank you for your patience.</center>
```

```
<center><input type=submit name= value="Go To VeriSign's Pay-  
Flow
```

```
Link Payment Pages (click here if you are not redirected automati-  
cally)">
```

```
</center>
```

```
</form>
```

```
</body>
```

```
</html>
```

If Payflow Link is the only payment method that you offer to your customers, you can let your Verify page be your redirect page by using the following tags.

<!--PAYFLOW_LINK_ACTION--> Tag

This tag will be replaced with a link pointing to the Payflow Link server specified in the payment settings section of your Merchant Administrator. The tag must appear in the template as:

```
<!--PAYFLOW_LINK_ACTION-->
```

Note: If you have only one payment method (Payflow Link) and you are using the above tag, you may want to pass your payment method choice as a hidden field, since there is only one option. You may pass your single payment method as a hidden field using the following code:

```
<input type=hidden name="billmethod" value="0">
```

<!--PAYFLOW_ORDER_DATA--> Tag

This tag will be replaced with hidden form data to be sent to the Payflow Link server. It is required for the interaction between PDG Shopping Cart and Payflow Link to operate properly. The tag must appear as:

```
<!--PAYFLOW_ORDER_DATA-->
```

PayPal Tags

If you are using PayPal as one of several payment methods, then PDG Shopping Cart will use a redirect page to take the customer to PayPal if it was the chosen payment method on the Verify page. This page will be displayed by default. However, if you would like to customize the redirect page, you may create a template and enter its name in the

PayPal settings of your Merchant Administrator. Your template should contain the following code:

```
<html>

<body onLoad="javascript:setTimeout('document.forms.pdgre-
direct.submit()',4000);">

<form action = "!---PAYPAL_ACTION---" method=POST name=pdgre-
direct>

<!--PAYPAL_DATA-->

<center>Please wait while we direct you to PayPal's secure web site
to

complete your payment.</center>

<center>Thank you for your patience.</center>

<center><input type=submit name= value="Go To PayPal's payment
server (click here if you are not redirected automatically)"></center>

</form>

</body>

</html>
```

If PayPal is the only payment method that you offer to your customers, you can let your Verify page be your redirect page by using the following tags.

<!--PAYPAL_ACTION--> Tag

This tag will be replaced with a link pointing to the PayPal server specified in the payment settings section of your Merchant Administrator. The tag must appear in the template as:

```
<!--PAYPAL_ACTION-->
```

Note: If you have only one payment method (PayPal) and you are using the above tag, you may want to pass your payment method choice as a hidden field, since there is only one option. You may pass your single payment method as a hidden field using the following code:

```
<input type=hidden name="billmethod" value="0">
```

<!--PAYPAL_DATA--> Tag

This tag will be replaced with hidden form data to be sent to the PayPal server. It is required for the interaction between PDG Shopping Cart and PayPal to operate properly. The tag must appear as:

<!--PAYPAL_DATA-->

WorldPay Tags

If you are using WorldPay as one of several payment methods, then PDG Shopping Cart will use a redirect page to take the customer to WorldPay if it was the chosen payment method on the Verify page. This page will be displayed by default. However, if you would like to customize the redirect page, you may create a template and enter its name in the WorldPay settings of your Merchant Administrator. Your template should contain the following code:

```
<html>  
  
<body onLoad="javascript:setTimeout('document.forms.pdgredirect.submit()',4000);">  
  
<form action = "!--WORLDPAY_ACTION--" method=POST  
name=pdgredirect>  
  
<!--WORLDPAY_DATA-->  
  
<center>Please wait while we direct you to WorldPay's secure web  
site to  
  
complete your payment.</center>  
  
<center>Thank you for your patience.</center>  
  
<center><input type=submit name= value="Go To WorldPay's pay-  
ment server (click here if you are not redirected automatically)"></  
center>  
  
</form>  
  
</body>  
  
</html>
```

If WorldPay is the only payment method that you offer to your customers, you can let your Verify page be your redirect page by using the following tags.

<!--WORLDPAY_ACTION--> Tag

This tag will be replaced with a link pointing to the WorldPay server specified in the payment settings section of your Merchant Administrator. The tag must appear in the template as:

<!--WORLDPAY_ACTION-->

Note: If you have only one payment method (WorldPay) and you are using the above tag, you may want to pass your payment method choice as a hidden field, since there is only one option. You may pass your single payment method as a hidden field using the following code:

```
<input type=hidden name="billmethod" value="0">
```

<!--WORLDPAY_DATA--> Tag

This tag will be replaced with hidden form data to be sent to the WorldPay server. It is required for the interaction between PDG Shopping Cart and WorldPay to operate properly. The tag must appear as:

```
<!--WORLDPAY_DATA-->
```

Basket Template Tags

In addition to the standard Verify Template tags, the Verify HTML Template can also take any of the standard Basket Template tags. The use of the standard Basket Template tags allows you to display the customer's Shopping Cart on the Verify page. This allows you to customize what is displayed on your Web store's Verify page.

'REPEAT' and 'OPTIONS' Tags

Like the Basket template, the Verify Template can also utilize the REPEAT and OPTIONS tags. The tags used within the REPEAT and OPTIONS tags for the Verify HTML template are the same as those used in the Basket HTML template (i.e.: the Repeat Basket tags and the Options Basket tags). For information on what tags can be used with the REPEAT and OPTIONS tags in the Verify Template, please see the 'Basket Template File' section of this chapter.

Passthrough Fields

Any form field whose name begins with the four letters "PASS" will have its values automatically forwarded to the final ThankYou templates and notification e-mails. For example:

```
<input type="text" name="PASSPhone" size="20">
```

The example above will create a phone number field in this template for customer input. Whatever the user types in this field is automatically forwarded to the emails, ThankYou templates, and order.log file.

Note: It is not mandatory for the customer to enter any information in this field – they can leave it blank and still check out normally. This field could be made mandatory by using JavaScript, but this is outside the normal functionality of the Shopping Cart.

ThankYou Template File

The ThankYou Template File is used after the BuyIt action has been received and the order has been submitted. The ThankYou Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends the HTML Template File to the customer, it searches the file for special tags. When a tag is encountered, it is replaced with the appropriate information before being sent to the customer's Web browser.

The following are the standard ThankYou Template tags:

- <!--ORDERINFO-->
- <!--INVOICE-->
- <!--CCAUTHNUMBER-->
- <!--TOTAL-->
- <!--SUBTOTAL-->
- <!--SOFTGOODS-->
- <!--SOFTGOOD_DATA-->
- <!--FIRSTNAME-->
- <!--LASTNAME-->
- <!--NAME-->
- <!--MIDDLEINIT-->
- <!--ADDR1-->
- <!--ADDR2-->
- <!--CITY-->
- <!--STATE-->
- <!--COUNTRY-->
- <!--ZIP-->
- <!--BRANDSFORLESS-->
- <!--INKTOMI-->
- <!--EMAIL-->
- <!--WEIGHT_TOTAL-->
- <!--NUM_ITEMS-->
- <!--ORDER_SUBTOTAL_DISC-->
- <!--BONUS_PT_TOTAL-->
- <!--ADD_SHIPPING_TOTAL-->
- <!--TAX-->
- <!--SHIPPING_TOTAL-->
- <!--FLUX_AMOUNT-->

- <!--PRE_FLUX_TOTAL-->
- <!--BILLNAME-->
- <!--BILLADDRESS-->
- <!--BILLCITY-->
- <!--BILLSTATE-->
- <!--BILLZIP-->
- <!--BILLCOUNTRY-->
- <!--BILLEMAIL-->
- <!--BILLAMOUNT-->
- <!--BILLDATE-->
- <!--BILLPHONE1-->
- <!--BILLPHONE2-->
- <!--SHIPNAME-->
- <!--SHIPADDRESS-->
- <!--SHIPCITY-->
- <!--SHIPSTATE-->
- <!--SHIPZIP-->
- <!--SHIPCOUNTRY-->
- <!--SUBTOTAL_WITH_DISCOUNT-->
- <!--SHIP_METHOD-->
- <!--BILLTEXT-->
- <!--SHIPTEXT-->

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

A sample ThankYou Template File is included with PDG Shopping Cart. Below is an example of how this template will be displayed.

Thank you for your order. It will be processed in three business days. Here is a copy of the information you have submitted. You will also receive a copy of this information via e-mail.			
Your invoice number is I00010.			
Qty	Description	Unit	Amount
1.00	{a0003LCW} Thingsamajig * Weight: 6.50 lbs. each * Bonus Points: 7.00 each * Option: size is large * Option: weight is cruiserweight	\$14.95	\$14.95
		\$1.00	\$1.00
Total Shipment Weight: 6.50 lbs.		Subtotal:	\$15.95
		Order Discount:	\$1.00
		Shipping:	\$5.00
		Total:	\$19.95
Total Bonus Points: 7.00			
Order placed at: Thu Aug 16 16:35:46 2001			
Ship/Deliver To:			
John Customer 1234 Meadow Ln. Springdale, ID 01010 United States Shipping Method: USPS Book Rate Shipping Method: jgc@emaildomain.com			
Bill To:			
John Customer 1234 Meadow Ln. Springdale, ID 01010 United States			
Payment Information: Off Line Credit Card			
Card Type: Visa			
Card Number: xxxxxxxxxxxx5167			
Card Expire: xx/xx			

Figure 8-11 Sample Thank You Template Display

<!--ORDERINFO--> Tag

When this tag is encountered in the ThankYou Template File, this tag is replaced with the complete invoice of the order submitted. The invoice will be displayed in table format. The invoice includes:

- The unique invoice number
- The complete order, with the final total considering shipping, tax and discounts
- The complete contact, billing, and shipping information.
- The payment information, with any credit card information properly censored

The tag must have its own line in the template file, and must appear in the template file as:

```
<!--ORDERINFO-->
```

Note: If the <!--ORDERINFO--> tag is used, then none of the following tags relating to the shopper's order will need to be implemented, as they will be redundant.

<!--INVOICE--> Tag

This tag is replaced with the unique invoice number for the order. The tag must appear in the template file as:

```
<!--INVOICE-->
```

<!--CCAUTHNUMBER--> Tag

This tag is replaced by the authorization number, assigned by the payment authorization service, that corresponds to the current order. This number can be referred to in order to track your authorization service's activities. The tag must appear in the template file as:

```
<!--CCAUTHNUMBER-->
```

<!--TOTAL--> Tag

This tag is replaced by the total amount for which the transaction was authorized. This represents the total amount of the order considering sales tax, shipping charges and any additional charges or discounts. The tag must appear in the template file as:

```
<!--TOTAL-->
```

<!--SUBTOTAL--> Tag

This tag is replaced by the subtotal of the order. This represents just the cost of the products ordered and does not include sales tax, shipping charges, etc. The tag must appear in the template file as:

```
<!--SUBTOTAL-->
```

<!--SOFTGOODS--> Tag

This tag is replaced by the Softgood data entered for the products being ordered. This data will generally consist of a link to a URL that contains a link to download the purchased Softgood product. If there is no Softgood data for any of the products being ordered, then this tag is ignored. The tag must appear in the template file as:

```
<!--SOFTGOODS-->
```

<!--SOFTGOOD_DATA--> Tag

If you use the <!--EXEC--> functionality, and your external program is generating data that you want to display to the customer, you may do

so with this tag. Unlike the `<!--SOFTGOODS-->` tag, this tag will only display the output of your external program, it will not display any other information included in the Softgood data section for the product being ordered. The tag must appear in the template as:

`<!--SOFTGOOD_DATA-->`

`<!--FIRSTNAME-->` Tag

This tag is replaced by the first name of the customer by whom the order was placed. The tag must appear in the template file as:

`<!--FIRSTNAME-->`

`<!--LASTNAME-->` Tag

This tag is replaced by the last name of the customer by whom the order was placed. The tag must appear in the template file as:

`<!--LASTNAME-->`

Note: If the `<!--FIRSTNAME-->` and `<!--LASTNAME-->` tags are used, then the following `<!--NAME-->` tag does not need to be implemented, as it would be redundant.

`<!--NAME-->` Tag

This tag is replaced by the full name of the customer by whom the order was placed. If this tag is used, you should not use either of the two tags above. The tag must appear in the template file as:

`<!--NAME-->`

`<!--MIDDLEINIT-->` Tag

This tag is replaced by the middle initial of the customer by whom the order was placed. The tag must appear in the template file as:

`<!--MIDDLEINIT-->`

`<!--ADDR1-->` Tag

This tag is replaced by the first line of the street address from which the customer placed the order. The tag must appear in the template file as:

`<!--ADDR1-->`

<!--ADDR2--> Tag

This tag is replaced by the second line of the address from which the customer placed the order. The tag must appear in the template file as:

```
<!--ADDR2-->
```

<!--CITY--> Tag

This tag is replaced by the city from which the customer placed the order. The tag must appear in the template file as:

```
<!--CITY-->
```

<!--STATE--> Tag

This tag is replaced by the state from which the customer placed the order. The tag must appear in the template file as:

```
<!--STATE-->
```

<!--COUNTRY--> Tag

This tag is replaced by the country from which the customer placed the order. The tag must appear in the template file as:

```
<!--COUNTRY-->
```

<!--ZIP--> Tag

This tag is replaced by the ZIP code from which the customer placed the order. The tag must appear in the template file as:

```
<!--ZIP-->
```

<!--EMAIL--> Tag

This tag is replaced by the e-mail address for the customer who placed the order. The tag must appear in the template file as:

```
<!--EMAIL-->
```

<!--BRANDSFORLESS--> Tag

This tag is replaced by the affiliate tracking information provided by Brands for Less. This is a third party affiliate tracking system, and you must first set up an account with Brands For Less before you will be able to utilize this tag. If the order contains no affiliate tracking information, then this tag is ignored. The tag must appear in the template file as:

```
<!--BRANDSFORLESS-->
```

<!--INKTOMI--> Tag

This tag is replaced by the affiliate tracking information provided by Inktomi. This is a third party affiliate tracking system, and you must first set up an account with Inktomi before you will be able to utilize this tag. If there is no affiliate tracking information to report for this order, then this tag is ignored. The tag must appear in the template file as:

```
<!--INKTOMI-->
```

<!--WEIGHT_TOTAL--> Tag

This tag is replaced by the total weight of all of the products in the order. This value may have been used to calculate shipping charges if weight tables were specified in the 'Shipping Rules' section of the Merchant Administrator (See chapter 8 in the PDG Shopping Cart User Guide for more information). The tag must appear in the template file as:

```
<!--WEIGHT_TOTAL-->
```

<!--NUM_ITEMS--> Tag

This tag is replaced by the total number of items in the order. This value may have been used to calculate shipping charges if quantity tables were specified the 'Shipping Rules' section of the Merchant Administrator (See chapter 8 in the PDG Shopping Cart User Guide for more information). The tag must appear in the template file as:

```
<!--NUM_ITEMS-->
```

<!--ORDER_SUBTOTAL_DISC--> Tag

This tag tells PDG Shopping Cart to replace the tag with whatever subtotal-based discount may have been applied to the order. This calculation is made depending upon the discounts you defined in the

Merchant Administrator (See Chapter 5 of the PDG Shopping Cart User Guide for more information). If no discount applies to the order, then this tag is ignored. The tag must appear in the template file as:

```
<!--ORDER_SUBTOTAL_DISC-->
```

Note: This tag may be appended with the letter “C”, <!--ORDER_SUBTOTAL_DISCC-->. PDG Shopping Cart will replace this tag with the order’s subtotal discount converted to the secondary currency.

<!--BONUS_PT_TOTAL--> Tag

This tag is replaced by the total number of bonus or ‘frequent shopper’ points incurred by the order. If no bonus points were earned by the order, then this tag is ignored. The tag must appear in the template file as:

```
<!--BONUS_PT_TOTAL-->
```

<!--ADD_SHIPPING_TOTAL--> Tag

This tag is replaced by the total additional shipping charges for any products in the order. These are shipping charges attributed to the products themselves. If there were no additional shipping charges incurred by any of the products in the order, then this tag is ignored. The tag must appear in the template file as:

```
<!--ADD_SHIPPING_TOTAL-->
```

Note: This tag may be appended with the letter “C”, <!--ADD_SHIPPING_TOTALC-->. PDG Shopping Cart will replace this tag with the total additional shipping charges for any products in the order converted to the secondary currency.

<!--TAX--> Tag

This tag is replaced by the sales tax applied to the order. If there was no sales tax applied to the order, then this tag is ignored. The tag must appear in the template file as:

```
<!--TAX-->
```

Note: This tag may be appended with a “C”, <!--TAXC-->. PDG Shopping Cart will replace this tag with the order’s total sales tax converted to the secondary currency.

<!--SHIPPING_TOTAL--> Tag

This tag is replaced by the total shipping charges placed on the order. This includes all shipping charges based on order's subtotal, weight, shipping method, or the products themselves. The tag must appear in the template file as:

<!--SHIPPING_TOTAL-->

Note: This tag may be appended with a "C", <!--SHIPPING_TOTALC-->. PDG Shopping Cart will replace this tag with the order's total shipping charges converted to the secondary currency.

<!--FLUX_AMOUNT--> Tag

This tag is replaced by the amount that was added to the order to accommodate changes in currency exchange rates. If a currency conversion was not made, then this tag is ignored. The tag must appear in the template file as:

<!--FLUX_AMOUNT-->

Note: This tag may be appended with a "C", <!--FLUX_AMOUNTC-->. PDG Shopping Cart will replace this tag with the order's flux amount converted to the secondary currency.

<!--PRE_FLUX_TOTAL--> Tag

This tag is replaced by the total of the order after that amount was converted to the secondary currency, but before the Flux Amount is added (see above). If a currency conversion was not made, then this tag is ignored. The tag must appear in the template file as:

<!--PRE_FLUX_TOTAL-->

Note: This tag may be appended with a "C", <!--PRE_FLUX_TOTALC-->. PDG Shopping Cart will replace this tag with the order's pre-flux total converted to the secondary currency.

<!--BILLNAME--> Tag

This tag is replaced with the name, as supplied by the customer on the Checkout page, to which the order is to be billed. The tag must appear in the template file as:

<!--BILLNAME-->

<!--BILLADDRESS--> Tag

This tag is replaced with the street address, as supplied by the customer on the Checkout page, to which the order is to be billed. The tag must appear in the template file as:

```
<!--BILLADDRESS-->
```

<!--BILLCITY--> Tag

This tag is replaced with the city, as supplied by the customer on the Checkout page, to which the order is to be billed. The tag must appear in the template file as:

```
<!--BILLCITY-->
```

<!--BILLSTATE--> Tag

This tag is replaced with the state, as supplied by the customer on the Checkout page, to which the order is to be billed. The tag must appear in the template file as:

```
<!--BILLSTATE-->
```

<!--BILLZIP--> Tag

This tag is replaced with the ZIP code, as supplied by the customer on the Checkout page, to which the order is to be billed. The tag must appear in the template file as:

```
<!--BILLZIP-->
```

<!--BILLCOUNTRY--> Tag

This tag is replaced with the country, as supplied by the customer on the Checkout page, to which the order is to be billed. The tag must appear in the template file as:

```
<!--BILLCOUNTRY-->
```

<!--BILLEMAIL--> Tag

This tag is replaced with the e-mail address, as supplied by the customer on the Checkout page, to which billing information is to be sent. The tag must appear in the template file as:

```
<!--BILLEMAIL-->
```

<!--BILLAMOUNT--> Tag

This tag is replaced with the amount that was billed to the customer for the current order. The tag must appear in the template file as:

```
<!--BILLAMOUNT-->
```

<!--BILLDATE--> Tag

This tag is replaced by the date on which the transaction was completed. This value will normally display the current date. The tag must appear in the template file as:

```
<!--BILLDATE-->
```

<!--BILLPHONE1--> Tag

This tag is replaced by the primary phone number for the person to which the order is to be billed. The tag must appear in the template file as:

```
<!--BILLPHONE1-->
```

<!--BILLPHONE2--> Tag

This tag is replaced by the secondary phone number for the person to which the order is to be billed. This value will generally be a work or cellular phone number. It should be provided by the customer on your Web store's Checkout page. The tag must appear in the template file as:

```
<!--BILLPHONE2-->
```

<!--SHIPNAME--> Tag

This tag is replaced by the name, as entered by the customer on the Checkout page, to which the order is to be shipped. The tag must appear in the template file as:

```
<!--SHIPNAME-->
```

<!--SHIPADDRESS--> Tag

This tag is replaced by the street address, as entered by the customer on the Checkout page, to which the order is to be shipped. The tag must appear in the template file as:

<!--SHIPADDRESS-->

<!--SHIPCITY--> Tag

This tag is replaced by the city, as entered by the customer on the Checkout page, to which the order is to be shipped. The tag must appear in the template file as:

<!--SHIPCITY-->

<!--SHIPSTATE--> Tag

This tag is replaced by the state, as entered by the customer on the Checkout page, to which the order is to be shipped. The tag must appear in the template file as:

<!--SHIPSTATE-->

<!--SHIPZIP--> Tag

This tag is replaced by the ZIP code, as entered by the customer on the Checkout page, to which the order is to be shipped. The tag must appear in the template file as:

<!--SHIPZIP-->

<!--SHIPCOUNTRY--> Tag

This tag is replaced by the country, as entered by the customer on the Checkout page, to which the order should be shipped. The tag must appear in the template file as:

<!--SHIPCOUNTRY-->

<!--SUBTOTAL_WITH_DISCOUNT--> Tag

This tag is replaced by the subtotal for the order in the customer's current order. This value represents just the total price of all of the products in the cart, with all discounting applied. This subtotal does not include any of the charges for any applied taxes or any shipping and handling. The tag must appear in the template file as:

<!--SUBTOTAL-->

<!--SHIP_METHOD--> Tag

This tag is replaced by the shipping method chosen by the customer. The tag must appear in the template file as:

```
<!--SHIP_METHOD-->
```

<!--BILLTEXT--> Tag

If you provided a field in the Checkout template named “billtext,” then this tag will be replaced with the customer-supplied text from that field. The tag must appear in the template file as:

```
<!--BILLTEXT-->
```

<!--SHIPTEXT--> Tag

If you provided a field in the Checkout template named “shiptext,” then this tag will be replaced with the customer-supplied text from that field. The tag must appear in the template file as:

```
<!--SHIPTEXT-->
```

Basket Template Tags

In addition to the standard ThankYou Template tags, the ThankYou HTML Template can also take any of the standard Basket Template tags. The use of the standard Basket Template tags allows you to display the customer’s Shopping Cart on the ThankYou page. This allows you to customize what is displayed on your ThankYou page.

‘REPEAT’ and ‘OPTIONS’ Tags

Like the Basket template, the ThankYou Template can also utilize the REPEAT and OPTIONS tags. The tags used within the REPEAT and OPTIONS tags for the Thank You HTML template are the same as those used in the Basket HTML template (i.e.: the ‘Repeat Basket’ tags and the ‘Options Basket’ tags). These tags allow you to customize the display of the customer’s Shopping Cart on the ThankYou page. For information on what tags can be used with the REPEAT and OPTIONS tags in the Thank You Template, please see the ‘Basket Template File’ section of this chapter.

Below is an example of a Thank You Template that has been customized using the REPEAT tags.

Thank you for your order. It will be processed in three business days. Here is a copy of the information you have submitted. You will also receive a copy of this information via e-mail.			
Quantity	Description	Unit	Amount
QTY:1	a0003MCW: Thingamajig		\$14.95
	size: medium		\$0.00
	weight: cruiserweight		\$1.00
Total Price of Options:			\$1.00
Price of Product with Options:		\$15.95	\$15.95
Discount:		\$0.00	\$0.00
Additional Shipping:		\$0.00	\$0.00
Weight: 6.50 (6.50 for 1 items)			
Total Ship Weight: 6.50 lbs.		Subtotal:	\$15.95
Total Bonus Points: 7.00		Shipping Total:	\$5.00
Additional Shipping: \$0.00		Tax Total:	\$0.00
		Pre-Flux Total:	\$19.95
		Flux Amount:	\$0.00
		Total:	\$19.95

Figure 8-12 Customized Thank You Template Display

DenyCredit Template File

The DenyCredit Template File is used when PDG Shopping Cart fails to receive authorization for a credit card purchase online. If you are not using any payment authorization services with PDG Shopping Cart, this template file will not be used. If a credit card authorization does fail, PDG Shopping Cart only displays this template file and not the ThankYou Template File.

The DenyCredit Template File is essentially an HTML Web page. However, before the template is sent to the customer's browser, PDG Shopping Cart searches it for special tags. When a tag is encountered, it is replaced with the appropriate information before being sent to the customer's Web browser.

The DenyCredit Template File can contain the following tags:

- <!--BILLINGINFO-->
- <!--REASON-->

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

A sample DenyCredit Template File is included with PDG Shopping Cart. Below is an example of how this template may be displayed.

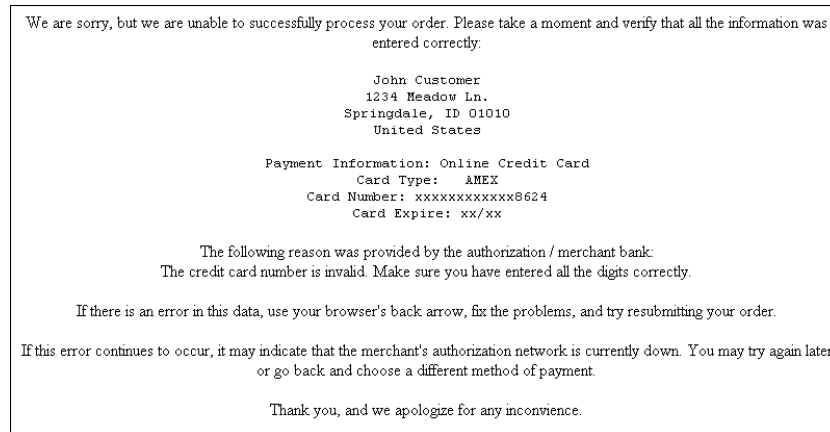


Figure 8-13 Sample Deny Credit Template Display

<!--BILLINGINFO--> Tag

When this tag is encountered in the DenyCredit Template File, the tag is replaced with the customer's billing and payment information. The billing information contains the billing name and address. The payment information is comprised of the selected payment method as well as any necessary account information (i.e.: credit card number and expiration date for credit cards; account and routing information for online checks). The customer will be able to view this information to ensure that they typed everything correctly.

The tag must appear in the template file as:

```
<!--BILLINGINFO-->
```

<!--REASON--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with the "reason" for the transaction's denial. This information comes from one of several sources including the failure of the authorization service. This tag can be removed if you do not want this information displayed. If no reason is supplied, then this tag will be ignored. If you do want the information displayed, the tag must appear in the template file as:

```
<!--REASON-->
```

AcceptCredit Template File

The AcceptCredit Template File is used after the BuyIt action has been received, but only if a credit card authorization has succeeded. The AcceptCredit Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends it to the customer, it searches the file for special tags, and replaces them with the corresponding information.

This template file is independent of the ThankYou Template File, which allows your Web store to have different order confirmation pages depending on whether the order required online credit card authorization. If you wish to have the same page displayed for all received orders including successful credit card authorizations, use only the ThankYou Template File and do not specify an AcceptCredit Template File in the Merchant Administrator (See chapter 5 of the PDG Shopping Cart User Guide for more information).

Note: The standard AcceptCredit Template tags are the same as the standard ThankYou Template tags. Please refer to the 'ThankYou HTML template' section for information on how to embed tags into the AcceptCredit HTML template file.

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

A sample AcceptCredit Template File is included with PDG Shopping Cart. Please see the Thank You Template File section for an example of how this template may be displayed.

Search Results Template

The Search Results Template File is used after the Search action has been received. This is the template that will be used to display the products which match the customer's search query, provided there were any matches. The Search Results Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends it to the customer, it searches the HTML Template File for special tags, and replaces them with the appropriate information.

The following are the standard Search Results Template tags:

- <!--SEARCHRESULTS-->
- <!--KEYWORDS-->

- <!--SEARCH_FAIL_MESSAGE-->
- <!--PRCAT-->
- <!--NEXT_SEARCH-->
- <!--PREVIOUS_SEARCH-->
- <!--PAGE_INFO-->
- <!--CAT_IMG-->
- <!--CAT_TEXT-->
- <!--CAT_DESC-->
- <!--SEC_FORCE_DATA-->
- <!--NEXT_SEARCH_URL-->
- <!--PREVIOUS_SEARCH_URL-->
- <!--RESULTS_PER_PAGE-->

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

A sample Search Results Template File is included with PDG Shopping Cart, however you may customize the look of your Search Results page by using the REPEAT tags, explained below. Below are two examples of how the standard Search Results template may be displayed (depending upon the layout chosen in the Merchant Administrator).



Figure 8-14 Sample Search Results Template Display (Thumbnail Layout)



Figure 8-15 Sample Search Results Template Display (Catalog Layout)

<!--SEARCHRESULTS--> Tag

When PDG Shopping Cart encounters this tag, it is replaced with the Thumbnail or Catalog formatted search results, depending upon which has been selected in the Merchant Administrator. See ‘Search Settings’ in Chapter 5 of the PDG Shopping Cart User Guide for more information. The tag must appear in the template file as:

```
<!--SEARCHRESULTS-->
```

Note: If you would like to customize the look of search results returned by your Web store, see the ‘REPEAT Tags’ section below. If you use the <!--SEARCHRESULTS--> tag, you should not use the REPEAT tags to customize your search results page.

<!--KEYWORDS--> Tag

This tag, when encountered, is replaced by the actual search criteria as entered by the customer. For “hard” coded searches, this tag may not be necessary and can be omitted (See Chapter 11 of the PDG Shopping Cart User Guide for more information on “hard” searches). The tag must appear in the template file as:

```
<!--KEYWORDS-->
```

<!--SEARCH_FAIL_MESSAGE--> Tag

This tag is replaced with the text message that is to be displayed if the customer's search failed. This message will be automatically generated by the Cart, depending upon the reason for the search failure. Generally, this tag will be used on the Search Results Empty HTML template (i.e.: "Search failed because there were no matches"). If the search did not fail, this tag will be ignored. The tag must appear in the template file as:

```
<!--SEARCH_FAIL_MESSAGE-->
```

<!--PRCAT--> Tag

PDG Shopping Cart replaces this tag with the name and code of the pricing category that is applicable for the customer. If the shopper has no appropriate pricing category, this tag is ignored. The tag must appear in the template file as:

```
<!--PRCAT-->
```

<!--NEXT_SEARCH--> Tag

This tag is replaced by a text link to the "next" page of search results. If there are no further pages of search results, then this tag is ignored. The tag must appear in the template file as:

```
<!--NEXT_SEARCH-->
```

<!--PREVIOUS_SEARCH--> Tag

This tag is replaced by a link to the "previous" page of search results. If the shopper is already on the first page of search results, then this tag is ignored. The tag must appear in the template file as:

```
<!--PREVIOUS_SEARCH-->
```

<!--PAGE_INFO--> Tag

This tag is replaced by the information regarding what page of the search results are being displayed. This information includes the number of the page being displayed and the total number of pages in the search results. The tag must appear in the template file as:

```
<!--PAGE_INFO-->
```

Category Template Tags

PDG Shopping Cart's Search Results template can contain certain Category template tags. These tags may be inserted in your Search Results template to provide a customer with information about the category within which the search was performed. The following standard Category template tags may be used in the Search Results template:

- <---CAT_IMG--->
- <---CAT_TEXT--->
- <---CAT_DESC--->

For information on what these tags mean, see the following section on the Category Template File.

Note: The three tags above may only be used in the Search Results template if the Search action has been called with the "Category" form input. See Chapter 9 of this Technical Reference Guide for more information on the Search action and its available inputs.

'SEARCH ROW' Tags

These tags are used to determine what data will be repeated for each 'row' on the Search Results page. You should have already defined how many search results will comprise a row using the Merchant Administrator (see Chapter 5 in the PDG Shopping Cart User Guide for more information). The repeated data may now actually comprise a row across the Search Results template depending upon how the HTML 'table' tags are formatted (see the following section for more information).

'REPEAT' Tags

The REPEAT tags can also be used on the Search Results template to customize the look of your Web store's search results page. The tags that can be used between the REPEAT tags on the Search Results template are the same as the standard Item template tags. Please see the 'Item Template File' section, above, for more information.

The REPEAT tags may only be used within the SEARCH ROW tags on the Search Result template. The placement of these tags helps to determine the format of the search results on the page. You must have first configured the Search Settings in the Merchant Administrator to display the results in "Custom Line Repeat" format. See Chapter 5 of the PDG Shopping Cart User Guide for more information on the "Search Settings" section of the Administrator.

If you have your HTML 'table row' tags (<tr>,</tr>) outside of the REPEAT tags, then the 'Number of Results per Row' that you specified in the Merchant Administrator will hold true. PDG Shopping Cart will then create as many rows as necessary to fulfill the 'Number of Results per Page' criteria, which is also configured in the Merchant Administrator (see Chapter 5 of the PDG Shopping Cart User Guide).

You can also embed your HTML 'table row' tags (<tr>,</tr>) inside the REPEAT tags. Then, each product will comprise a single row in a standard table layout. Note that the HTML 'table row' tags override the 'Number of Search Results per Row' setting indicated in the Merchant Administrator.

If you have a heading row, you can place it within the SEARCH ROW tags. This will cause it to repeat after however many 'Number of Results per Row' you specified in the Merchant Administrator. Placing the header row outside of the SEARCH ROW tags will cause the header to display only once.

Below are two examples of how a customized Search Results template may be displayed. In Figure 8-16, the 'table row' tags are located inside the REPEAT tags. In Figure 8-17, the 'table row' tags are located outside of the REPEAT tags.

Search results for widget



Image	Product	Price	Quantity	Click to Purchase	Options	Inventory	Weight	Bonus Points
	(a0001) widget	\$10.89	Qty: <input type="text" value="0"/>	<input type="button" value="Add Me"/>	Item Options: Widget Size: <small>Small</small>	Display Inventory Status	Product Weight: 0.75 pounds	Bonus Points:5.00
	(a0002) doo dad	\$5.69	Qty: <input type="text" value="0"/>	<input type="button" value="Add Me"/>	Item Options:	Out of Stock - do not order this product at this time	Product Weight: 1.60 pounds	Bonus Points:9.00
x Image	x Product	x Price	x Quantity	x Click to Purchase	x Options	x Inventory	x Weight	x Bonus Points

Figure 8-16 Custom Search Results Template Display



Figure 8-17 Custom Search Results Template Display

Category Template

The Category Template File is used after the customer has clicked on a button to display all of your Web store's product categories. The Category Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends it to the customer's Web browser, it searches the Template File for special tags, and replaces them with the appropriate information.

The only tags that may be used on the Category HTML template file must be used within the REPEAT tags. This causes the category information to be displayed for every category in your Web store. The standard Repeat Category tags are as follows:

- <---CAT_IMG--->
- <---CAT_TEXT--->
- <---CAT_DESC--->
- <---CAT_PARENT--->
- <---CAT_CODE--->
- <---CAT_DISP_PROD--->

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

A sample Category Template File is included with PDG Shopping Cart. Below is an example of how that template may be displayed.



A.000		text: We offer all types of books	Desc: Books	parent: view products
MUSC		text: CDs, tapes, records, 8 tracks	Desc: Music	parent: view products

Figure 8-18 Sample Category Template Display

<---CAT_IMG---> Tag

This tag is replaced by a reference to the product category's corresponding image, as defined in the 'Product Category Details' section of the Merchant Administrator. The tag must appear between the REPEAT tags in the template file as:

<---CAT_IMG--->

<---CAT_TEXT--->

This tag is replaced by the name of a particular product category, as defined in the 'Product Category Details' section of the Merchant Administrator. The tag must appear between the REPEAT tags in the template file as:

<---CAT_TEXT--->

<---CAT_DESC--->

This tag is replaced by the description of the particular category as defined in the 'Product Category Details' section of the Merchant Administrator. This tag must appear between the REPEAT tags in the template file as:

<---CAT_DESC--->

<---CAT_PARENT--->

This tag is replaced by the particular category's parent category as defined in the 'Product Category Details' section of the Merchant Administrator. This tag must appear between the REPEAT tags in the template file as:

```
<---CAT_PARENT--->
```

<---CAT_CODE--->

This tag is replaced by the particular category's four character code as defined in the 'Product Category Details' section of the Merchant Administrator. This tag must appear between the REPEAT tags in the template file as:

```
<---CAT_CODE--->
```

<---CAT_DISP_PROD--->

This tag is replaced by the URL for the template that acts as the home page for the particular product category as defined in the 'Product Category Details' section of the Merchant Administrator. This tag should be used within an HTML "hypertext reference" tag (href) for creating a link to this URL. This tag must appear between the REPEAT tags in the template file. An example of this tag in the Category template would be:

```
<a href="!---CAT_DISP_PROD---">view products</a>
```

ShipCompare Template

The ShipCompare Template File is used to display shipping rates returned from UPS or USPS, when a specific PDG Cart shipping method is defined to compare rates. The ShipCompare Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends it to the customer's Web browser, it searches the Template File for special tags, and replaces them with the appropriate information.

The standard ShipCompare tags are as follows:

- <!--BEGIN_SHIP_COMPARE-->
- <!--END_SHIP_COMPARE-->
- <!--SERVICE_NAME-->

- `<!--SERVICE_COST-->`
- `<!--SERVICE_DESCRIPTION-->`
- `<!--SERVICE_ERROR-->`
- `<!--SERVICE_BUTTON-->`

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

`<!--BEGIN_SHIP_COMPARE-->` Tag

This tag denotes the beginning of a repeatable section of the Ship-Compare template. This section is then repeated, until the `END_SHIP_COMPARE` tag is reached, for each shipping service rate to be compared. The tag must appear in the template file as:

```
<!--BEGIN_SHIP_COMPARE-->
```

`<!--END_SHIP_COMPARE-->` Tag

This tag denotes the end of a repeatable section of the ShipCompare template. This section, beginning with the `BEGIN_SHIP_COMPARE` tag, is then repeated for each shipping service rate to be compared. The tag must appear in the template file as:

```
<!--END_SHIP_COMPARE-->
```

`<!--SERVICE_NAME-->` Tag

This tag is replaced with the name of the available UPS / USPS service. The tag must appear inside `BEGIN_SHIP_COMPARE` and `END_SHIP_COMPARE` tags in the template file as:

```
<!--SERVICE_NAME-->
```

`<!--SERVICE_COST-->` Tag

This tag is replaced with the cost of the available UPS / USPS service. The tag must appear inside `BEGIN_SHIP_COMPARE` and `END_SHIP_COMPARE` tags in the template file as:

```
<!--SERVICE_COST-->
```

<!--SERVICE_DESCRIPTION--> Tag

This tag is replaced with a brief description of the available UPS / USPS service. The tag must appear inside BEGIN_SHIP_COMPARE and END_SHIP_COMPARE tags in the template file as:

```
<!--SERVICE_DESCRIPTION-->
```

<!--SERVICE_ERROR--> Tag

If an error occurs while retrieving information for this service, then this tag is replaced with that error. An example might be if a service is not available because of an order's extreme weight. The tag must appear inside BEGIN_SHIP_COMPARE and END_SHIP_COMPATE tags in the template file as:

```
<!--SERVICE_ERROR-->
```

<!--SERVICE_BUTTON--> Tag

This tag is replaced with a submit button that allows the customer to choose the service and proceed with checkout. The tag must appear inside BEGIN_SHIP_COMPARE and END_SHIP_COMPATE tags in the template file as:

```
<!--SERVICE_BUTTON-->
```

UPS Track Template File

The UPS Track Template is used to allow customers to enter information and track the status of their order package. If you wish to provide UPS package tracking to your customers, you will use the UPS Track template provided with your PDG Shopping Cart. You can create a link to the tracking page using the following code:

```
<a href="http://www.mywebstore.com/cgi-bin/shopper?display=
action&template=Templates/UPSTrack.html&emptyoverride=
yes">Track Order</a>
```

Even though the UPS Track template is a static HTML page, you should use the above link to display the template through a Cart action, otherwise the country list will not be populated by PDG Shopping Cart.

The standard UPS Track template uses the following tag:

- <!--COUNTRY_LIST-->

<!--COUNTRY_LIST--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with a drop-down menu containing all the countries with their corresponding numeric code for PDG Shopping Cart. The tag must appear in the template file as:

```
<!--COUNTRY_LIST-->
```

UPS Tracking Result Template File

If you offer UPS package tracking to your customers, this template is used to display the results of the tracking inquiry. This page will display the address to which the package was shipped, the address from which the package was shipped, the details of the package, and the date and location of every action taken on the package.

The standard UPS Track Result template uses the following tag:

- <!--TRACK_RESULTS-->

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

<!--TRACK_RESULTS--> Tag

When this tag is encountered, PDG Shopping Cart replaces it with the information returned from UPS, including the address the package was shipped from, the address the package was shipped to, the details of the package, and the date and location of every action taken on the package. The tag must appear in the template file as:

```
<!--TRACK_RESULTS-->
```

UPS Address Validation Template File

If you are registered to use UPS OnLine Tools and have checked the “I want UPS® Address Validation When Customers Check Out” box in the UPS Details section of your Merchant Administrator, this template

will display if a customer enters an invalid shipping address on the Checkout page (whether or not they have chosen a UPS shipping method). This template will allow the customer to bypass the validation and use the current address, enter a new address, or choose from the closest available matches to the address entered on the Checkout page. Place the name/location of this template file here.

The standard UPS Address Validation template uses the following tags:

- `<!--UPS_CURRENT_ADDR-->`
- `<!--UPS_ADDR_SUGGEST-->`

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

`<!--UPS_CURRENT_ADDR-->` Tag

When this tag is encountered, PDG Shopping Cart will replace it with the actual address that the customer entered on the Checkout page. The tag must appear in the template file as:

```
<!--UPS_CURRENT_ADDR-->
```

`<!--UPS_ADDR_SUGGEST-->` Tag

When this tag is encountered, PDG Shopping Cart will replace it with a drop-down menu containing the closest available matches to the address entered on the Checkout page that were found as valid addresses by UPS. The tag must appear in the template file as:

```
<!--UPS_ADDR_SUGGEST-->
```

Verify - Inventory Error Template

The Verify - Inventory Error Template is used when your webstore tracks inventory on products and backordering is turned off, and not enough inventory is available to fill a customer's order during the checkout process. PDG Shopping Cart will modify the basket's contents to include only the items in inventory. Then the new basket will be displayed to the customer using this template. The customer can then continue to checkout normally, or go back and shop for other items.

Note that this template is only used if you are tracking inventory and do not allow backordering of out-of-stock products. The template can be displayed prior to the CheckOut, or between the CheckOut and Verify steps.

The Verify - Inventory Error Template File is essentially an HTML Web page. However, before PDG Shopping Cart sends it to the customer's Web browser, it searches the Template File for special tags, and replaces them with the appropriate information.

The standard Verify - Inventory Error template uses all the tags that the Verify template uses. In addition to those tags, these other tags are used:

- `<!--PRINT_BILLING-->`
- `<!--SUBMIT_ORDER_BUTTON-->`
- `<!--INV_RESET_BASKET_WARNING-->`

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

`<!--PRINT_BILLING-->` Tag

This tag is replaced with hidden fields containing any billing information that may be submitted during the checkout process. If this template is used by PDG Shopping Cart before any billing information is collected, then this tag is ignored. If this template is used by PDG Shopping Cart between the CheckOut and Verify steps, then this tag is replaced with hidden fields containing the data from the CheckOut step.

The tag must appear in the template file as:

```
<!--PRINT_BILLING-->
```

`<!--SUBMIT_ORDER_BUTTON-->` Tag

This tag is replaced with a submit button, allowing the order with the modified basket to continue to be processed. The tag must appear in the template file as:

```
<!--SUBMIT_ORDER_BUTTON-->
```

<!--INV_RESET_BASKET_WARNING--> Tag

This tag is replaced by an error message, indicating that certain products in the basket are no longer available for purchase due to inventory. The tag must appear in the template file as:

```
<!--INV_RESET_BASKET_WARNING-->
```

Error Template File

If any type of error occurs during the customer's shopping in your web store, PDG Shopping Cart uses the Error template to display the appropriate error message to the customer.

The standard Error template uses the following tags:

- <!--ERROR_MESSAGE-->
- <!--ERROR_CODE-->
- <!--SYSTEM_VERSION-->

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

<!--ERROR_MESSAGE--> Tag

When this tag is encountered, PDG Shopping Cart will replace it with an error message explaining what action has caused the problem. The tag must appear in the template file as:

```
<!--ERROR_MESSAGE-->
```

<!--ERROR_CODE--> Tag

When this tag is encountered, PDG Shopping Cart will replace it with the index value of the error that was returned. This number is useful to know if you must contact PDG Software for technical support. The tag must appear in the template file as:

```
<!--ERROR_CODE-->
```

<!--SYSTEM_VERSION--> Tag

When this tag is encountered, PDG Shopping Cart will replace it with the version, build, and patchlevel information of the Cart that is installed on your server. The tag must appear in the template file as:

```
<!--SYSTEM_VERSION-->
```

SSL Templates

The SSL (secure socket layer) Template Files work like their non-secure counterparts. They are essentially HTML Web pages, and they are also searched for special tags before PDG Shopping Cart sends them to the customer's Web browser.

The sample SSL Template Files that are included with PDG Shopping Cart are:

- SSL Checkout Template
- SSL Verify Template
- SSL Thank You Template

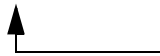
These template files should essentially be the same as their non-secure counterparts, simply located in a secure directory on you Web server.

PDG Shopping Cart is given the location of this template file from the directory and template name specified in the Templates section of the Merchant Administrator (which stores a key in the Shopper.Conf file). For more information on entering files in the Templates section, refer to Chapter 14 of the PDG Shopping Cart User Guide.

Important: If you are using custom graphics on your Web store, you must provide the complete URL in your image references on all SSL Templates. Example:

```

```



If you do not provide the complete URLs to images or maintain an 'Images' directory on the secure side of your Web server, the links to your images will be broken on your SSL Templates.

Empty Templates

There are two HTML templates that come with PDG Shopping Cart that are simply static HTML pages. These are the Basket Empty Template and the Search Results Empty Template. They are essentially static because there are no tags for PDG Shopping Cart to interpret on these templates. Actually, each template can take one tag.

Basket Empty Template

The Basket Empty Template is used in place of the Basket Template when there are no items in the customer's Cart. Instead of the tags for formatting items, you should put some text that notifies that the customer that there are no products in his Shopping Cart.

This template will be used whenever the DISPLAY action is called and the customer's Cart is empty. The Basket Empty Template will also be displayed whenever the customer clicks the 'Remove All' button to empty his Shopping Cart.

<!--INV_RESET_BASKET_WARNING--> Tag

This tag is replaced by an error message, indicating that certain products in the basket are no longer available for purchase due to inventory. The tag must appear in the template file as:

```
<!--INV_RESET_BASKET_WARNING-->
```

Search Results Empty Template

The Search Results Empty Template is displayed in place of the Search Results Template when there are no items that match the customer's search. Instead of inserting tags for formatting items, you should put some text that notifies that the customer that there are no products which fit his query. You may also use the following tag to display this message:

<!--SEARCH_FAIL_MESSAGE--> Tag

This tag is replaced with the text message that is to be displayed if the customer's search failed. This message will be automatically generated by the Cart, depending upon the reason for the search failure. The tag must appear in the template file as:

<!--SEARCH_FAIL_MESSAGE-->

The SearchResultEmpty HTML template will be used whenever the SEARCH action is called and there are no items to return.

Note: You may also want to include some search tips or links to “hard” searches on this template. See ‘Embedding PDG Shopping Cart’ in Chapter 10 of the PDG Shopping Cart User Guide for more information on “hard searches.”

HTML Email Templates

If you are using the PDG Internal Mailer, you can have PDG Shopping Cart send HTML emails to yourself and customers. With these HTML emails, you can use HTML to graphically display your order receipts sent to your customers. Remember that your customer’s email program will need to be able to display HTML to display these emails correctly.

The HTML Email templates can use any PDG Shopping Cart tags that are available for use on the Thank You template.

If you are using HTML emails, you can send attachments which you can display within the emails. All attached files must be in the Attachments folder located inside the PDG_Cart directory. Directory traversal to reach a file in a different location on the server is not allowed. To display the attached file within the email, you should code the HTML as normal, but place cid: in front of the attached filename, as in the following example:

```

```

Universal Tags

New in PDG Shopping Cart 2004 are two new template tags that can be inserted into any of your Web store’s dynamic templates. These new tags call information about the customer’s current Shopping Cart that can be displayed on PreAdd pages, the Basket page or any other page in your Web store that is generated via template.

The Universal tags that come with PDG Shopping Cart 2004 are:

- !---SCRIPTNAME---

- `!---SEC_SCRIPTNAME---`
- `<!---CURRENT_SUBTOTAL--->`
- `<!---CURRENT_ITEMS_NUM--->`
- `!---SHOPSOMEMORE_LINK---`

We will now cover these tags in detail.

!---SCRIPTNAME---

This tag tells PDG Shopping Cart to replace it with the location of your Shopping Cart's "shopper" executable. This tag can be used to call "shopper" from any one of your Web store's templates or static pages. This tag will generally fall inside of an HTML 'form' tag, but may also be used with the 'href' tag as well. Below are two examples of how this tag may be used:

```
<form method=POST action="!---SCRIPTNAME---">
<input type=submit name="display" value="View My Cart">
</form>
```

```
<a href="!---SCRIPTNAME---?display=action">View My Cart</a>
```

The first example would produce a button on your page that, when clicked, would call PDG Shopping Cart to display the customer's current shopping cart. The second example would create a text link that calls PDG Shopping Cart to display the customer's current shopping cart.

!---SEC_SCRIPTNAME---

This tag tells PDG Shopping Cart to replace it with the location of your Shopping Cart's secure "shopper" executable. This tag can be used to call your secure "shopper" from any one of your Web store's templates or static pages. This tag will generally fall inside of an HTML 'form' tag, but may also be used with the 'href' tag as well. Below are two examples of how this tag may be used:

```
<form method=POST action="!---SEC_SCRIPTNAME---">
<input type=submit name="checkout" value="Secure Checkout">
</form>
```

```
<a href="!---SEC_SCRIPTNAME---?checkout=action">Secure Check-  
out </a>
```

The first example would produce a button on your page that, when clicked, would call PDG Shopping Cart to proceed to your Web store's secure checkout page. The second example would create a text link that calls PDG Shopping Cart to proceed to your Web store's secure checkout page.

<!---CURRENT_SUBTOTAL---> Tag

This tag can be used to display the subtotal of all of the items in the customer's current Shopping Cart. This subtotal only includes the price of the products (and any applicable options) that are in the customer's Shopping Cart. Sales tax, shipping charges and product discounts are not included in this value. The tag must appear in the template as:

```
<!---CURRENT_SUBTOTAL--->
```

Note: This tag may be appended with a "C", <!--BASEPRICEC-->. PDG Shopping Cart will replace this tag with the current Shopping Cart subtotal converted to the secondary currency.

<!---CURRENT_ITEMS_NUM---> Tag

This tag can be used to display the total number of items in the customer's current Shopping Cart. This total includes all quantities of all products that are in the customer's Shopping Cart. The tag must appear in the template as:

```
<!---CURRENT_ITEMS_NUM--->
```

!---SHOPSOMEMORE_LINK--- Tag

This tag is replaced by a Shop Some More button. Please note that this tag is replaced with all the code to display the button as defined in the Merchant Administrator as well as the URL. The tag must appear in the template as:

```
!---SHOPSOMEMORE_LINK---
```



P DG Shopping Cart Actions describe the various actions that customers can perform on their Shopping Cart. These actions include adding items, removing items, performing searches and checking out.

Because PDG Shopping Cart is a CGI product, it must be called from a form within an HTML Web page. This section assumes that the installer or maintainer of the Web store has some knowledge of HTML and HTML forms.

Each action has a predefined code that is passed to PDG Shopping Cart via a form embedded in the HTML page. This section explains each action code and the required HTML form inputs that PDG Shopping Cart needs to perform the action. There is also a listing and explanation of the optional form inputs that can be used with each Cart action.

Actions are submitted to PDG Shopping Cart through the “submit” type of form input for HTML. A unique string of text identifies each PDG Shopping Cart action. This string of text must be the “name” parameter used in the “submit” input statement. Many actions also require additional information to be passed to PDG Shopping Cart through other form inputs. Here is an example of embedding an action in an HTML file:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">

```

The rest of this section defines each action and any form inputs that go with it.

Search Action

This action instructs PDG Shopping Cart to search the product database for specific keywords and return all of the products that match the query. The resulting page can then have thumbnails with PreAdd links, catalog listings with Add links or they may be custom formatted (See the ‘Search Settings’ section in Chapter 5 of the PDG Shopping Cart User Guide or the ‘Search Results template’ section of Chapter 8 of this Technical Reference Guide for more information).

Here is the syntax for using the SEARCH action:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
<input type="text" name="KEYWORDS">
<input type="submit" name="SEARCH" value="[text to appear on but-
ton]">
</form>
```

The submit input's name must be “Search.” The Keywords form input is required in order for PDG Shopping Cart to know what to look for. The Keywords form input may be a hidden field with a hard-coded value for pre-determined searches (see Chapter 10 for more information on “hard” searches).

New in PDG Shopping Cart 2004 is the ability to search fields other than products’ keywords and descriptions. Customers now have the option of searching product SKU’s or product categories. You may also allow your shoppers to search for products using a desired price range. The syntax for these inputs is essentially the same as the keywords input:

```
<input type="text" name="SKU">
<input type="text" name="CATEGORY">
<input type="text" name="PRICECODE">
<input type="text" name="PRICERANGE">
```

Note: As with the Keywords form input, any of these inputs may also be hard-coded into the template as “hidden” input types.

PDG Shopping Cart uses the SearchResult HTML Template File to generate the Web page that allows customers to view the products that matched their query. For information on this template file and others, refer to the ‘HTML Template Files’ sections in Chapter 5 of the PDG Shopping Cart User Guide and in Chapter 8 of this Technical Reference Guide.

The template that is used to display the search results may also be overridden with one of the following tags depending upon which layout you have selected for returning search results (see ‘Search Settings’ in chapter 5 for more information):

```
<input type="hidden" name="TEMPLATE" value="[location of custom template]">
```

```
<input type="hidden" name="ITEMTEMPLATE" value="[location of custom template]">
```

See below for more information on the usage of these form inputs.

Keywords form input (required)

PDG Shopping Cart uses this data field to know what to search for. This field may be a text input field, allowing users to specify their own keywords. Alternatively, this may be a hidden input field, with a hard-coded value. This hard-coding allows dynamic category pages of products to be generated. A hyperlink can be created which performs a search for a specific keyword and thus produces dynamic category pages. Customers should be informed that “and,” “or” and “not” may be used in this input field for yielding a broader or more narrow range of search results.

Note: The Keywords form input is only required for the Search action if none of the following form inputs are going to be used.

SKU form input (optional)

PDG Shopping Cart uses this data field to know what to search for. This field should be a text input field, allowing users to specify their own SKU or product code for which to search.

Category form input (optional)

PDG Shopping Cart uses this data field to know what to search for. This may be a text input field, allowing users to specify their own category. Alternatively, this field may be a drop-down menu, with hard-coded values from which the customer may choose. This hard-coding allows for dynamic searches. A hyperlink can also be created which performs a search for a specific category and thus produces dynamic category pages.

PriceRange form input (optional)

PDG Shopping Cart uses this data field to know what to search for. This field will generally be a text input field, allowing users to specify their own price range for products. Or this input may be a hidden input field, with a hard-coded value. This hard-coding can allow for dynamic pages of “bargain” products to be generated.

Note: The shopper should be informed of the syntax that may be used when searching for products within a price range (i.e.: >,<=,x-y).

Note: While the above three form inputs all indicate that they are optional, **at least one** form input is required for every Search action called from your Web store.

PriceCode form input (optional)

If the Search link is to use a specific pricing category, then this information should be included in the Pricecode form input. The value for this form input should be equal to the code of the appropriate pricing category, as defined in the Merchant Administrator. If this field is not provided with the Search action, then the unit price indicated in the product definition will be used.

This field can either be hard-coded, or set up as a text entry box, where the customer must supply the pricing category code in order to obtain that category's pricing from your Web store.

ItemTemplate form input (optional)

If you have the search results being returned in catalog page form, as described in Chapter 5 of the PDG Shopping Cart User Guide, then the sample-item template will be used. However you can override this with the ItemTemplate form input. This field must be named "ItemTemplate," and the value is the path to the new template file. This path is assumed to be relative to the 'PDG_Cart' subdirectory in your CGI directory.

Template form input (optional)

By default, the results of the Search action are returned in the SearchResult Template File. You may specify a different template file by creating a hidden input field. This field must be named "Template," and the value must be the path to the new template file. This path is assumed to be relative to the 'PDG_Cart' subdirectory in your CGI directory.

PreAdd Action

This action instructs PDG Shopping Cart to generate a Web page from the product database files allowing the customer to indicate quantity and options for a specific product. PreAdd instructs PDG Shopping Cart to automatically create an HTML page that contains a call to PDG Shopping Cart with the Add action. The PreAdd action will generate a Web page with the product's image, description, price and drop-down menus for each product option. It is from

this “Add Item” page that the customer can then select options and indicate a desired quantity. The user will then submit this data back to PDG Shopping Cart, via the Add action.

The generated HTML page is divided into two sections. The first section contains the product's image, as specified by the Image URL in the product's definition in the corresponding product database file. The second section contains the product's description, pricing, and all form inputs for selecting quantity and options.

If you do not want PDG Shopping Cart to automatically create an Add Item page for a product, then use the Add action.

Here is the syntax for using the PreAdd action:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
<input type="hidden" name="KEY" value="[product code]">
<input type="hidden" name="PRICECODE" value="[pricing category
code]">
<input type="submit" name="PREADD" value="[text to appear on
button]">
</form>
```

The submit input's name must be “PreAdd.” The Key form input is required.

PDG Shopping Cart uses the AddItem Template File to generate the Web page that allows customers to select options and quantities. This template should contain the !---ITEM--- tag in order to call the sample Item template that comes with PDG Shopping Cart. For more information on these template files, refer to the ‘HTML Template Files’ sections in Chapter 5 of the PDG Shopping Cart User Guide and in Chapter 8 of this Technical Reference Guide.

The templates that are used may also be overridden with one of the following tags depending upon which layout you will be using for viewing the product's details (see ‘HTML Template Files’ in chapter 8 of this Technical Reference Guide for more information).

```
<input type="hidden" name="TEMPLATE" value="[location of custom
template]">
```

will override the standard AddItem HTML template.

```
<input type="hidden" name="ITEMTEMPLATE" value="[location of
custom template]">
```

will override the standard Item HTML template.

See below for more information on the usage of these form inputs.

Key form input (required)

PreAdd knows which product the customer wants to add from the Key form input. It is a hidden key. This code should be the same code as the product code found in the product definition in the product database file. This input should be generated automatically by the Cart when the Item template is drawn. For more information on product codes and product definitions, refer to the 'Maintaining Products' section in Chapter 6 of the PDG Shopping Cart User Guide.

Pricecode form input (optional)

If the PreAdd link is to use a specific pricing category, then this information needs to be included in the Pricecode form input. The value for this form input should be equal to the code of the appropriate pricing category, as defined in the Merchant Administrator. If this field is not provided with the PreAdd action, then the unit price specified in the product definition will be used.

This field can either be hard-coded, or setup as a text entry box, where the customer has to supply the pricing category code in order to obtain that category's pricing from your Web store.

Template form input (optional)

By default, the PreAdd action calls the standard AddItem template located in your 'PDG_Cart' subdirectory in your CGI directory. You may specify a different template file by creating a hidden input field. This field must be named "Template," and the value must be the path to the new template file. This path is assumed to be relative to the 'PDG_Cart' subdirectory.

ItemTemplate form input (optional)

The AddItem HTML template generally contains a call to the Item template in order to display the products to be added. However you can override this with the ItemTemplate form input. This field must be named "ItemTemplate," and the value must be the path to the new template file. This path is assumed to be relative to the 'PDG_Cart' subdirectory.

MultiPreAdd Action

This action instructs PDG Shopping Cart to generate a Web page from the product database file. This page will allow the customer to select options and quantities for several products at once. MultiPreAdd instructs PDG Shopping Cart to create an HTML page automatically with a call to PDG Shopping Cart with the MultiAdd action. The MultiPreAdd action will generate a Web page with multiple listings for a single product or a listing of multiple products, depending upon the form inputs that are submitted with it. These listings can be customized to include the product's image, price, weight and all available option selections for a specific product (See the 'Item Template File' section in Chapter 8 of this Technical Reference Guide for more information). The user can then select options and quantities for each entry and submit the data back to PDG Shopping Cart through the MultiAdd action.

On the generated HTML page, the data that is to be displayed for the selected product is fully customizable. See Chapter 8 of this Technical Reference Guide for more information on how to customize this template.

If you do not want PDG Shopping Cart to automatically create an option selection Web page for a product, then use the Add action.

Here is the syntax for using the MultiPreAdd action:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
  <input type="hidden" name="KEYS" value="[product code 1],[product
code 2],[product code n]">
  <input type="hidden" name="PRICECODE" value="[pricing category
code]">
  <input type="submit" name="MULTIPREADD" value="[text to appear on
button]">
</form>
```

The submit input's name must be "MultiPreAdd." The Keys form input is required.

PDG Shopping Cart uses the MultiAdd Template File to generate the Web page that allows customers to select options and quantities. For information on this template file, refer to the 'HTML Template Files' sections in Chapter 5 of the PDG Shopping Cart User Guide and in Chapter 8 of this Technical Reference Guide.

Note: PDG Shopping Cart 2004's locked options feature will not be enabled when the MultiAdd template is called from the MultiPreAdd action.

Keys form input (required)

The MultiPreAdd action knows which products the customer wants to display from the Keys form input. This is a hidden input. The value for this input should be the SKUs of the products you wish to display, each separated by a comma. These codes should be the same codes as the product codes found in the product definition in the product database file. For more information on product codes and product definitions, refer to the 'Maintaining Products' section in Chapter 6 of the PDG Shopping Cart User Guide. The input's name must be "Keys."

Pricecode form input (optional)

If the MultiPreAdd link is to use a specific Price Category, then this information needs to be included in the Pricecode form input. The value for this form input should be equal to the code of the appropriate pricing category that is defined in the Merchant Administrator. If this field is not provided with the MultiPreAdd action, then the unit price defined in the product definition will be used.

This field can either be hard-coded, or setup as a text entry box, where the customer must supply the pricing category code in order to obtain that category's pricing from your Web store. The input's name must be "Pricecode."

Template form input (optional)

By default, the MultiPreAdd action calls the standard MultiAdd template located in your 'PDG_Cart' subdirectory in your CGI directory. You may specify a different template file by creating a hidden input field. This field must be named "Template," and the value is the path to the new template file. This path is assumed to be relative to the 'PDG_Cart' subdirectory.

Add Action

This action will actually add a product to the customer's Shopping Cart. Unlike the PreAdd, this action requires quantity, price, and any available option selections passed to it via form inputs.

If you would like PDG Shopping Cart to generate an HTML page with the Add action in it and all the proper form inputs for the product as defined in your product database files, then use the PreAdd action.

Here is the syntax for using the Add action. For non-hidden form inputs, a sample input type is shown. However, it is up to the Webmaster to determine

what type of inputs to use. For pregenerated Web pages from the PreAdd action, the input types used are shown in the example.

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
<input type="hidden" name="KEY" value="[product code]">
<input type="hidden" name="REFERENCE" value="[link URL]">
<input type="text" name="QTY" value="[initial value]" size="[size of
field]">
<select name="OPT0">
<option value="[Selection 1 for Option 0]">
[more option selections...]
</select>
[more options...]
<input type="submit" name="ADD" value="[text to appear on button]">
</form>
```

Note that the options are indexed starting with the number '0.' Number '0' refers to the first option definition in the product definitions, number 1 refers to the second option definition, and so forth.

The submit input's name must be "Add." The Key and Qty form inputs are required. All other form inputs are optional.

PDG Shopping Cart will use the Basket Template File to display the contents of the customer's Shopping Cart after adding a product to the basket. For more information on this and other template files, refer to the 'HTML Template Files' sections in Chapter 5 of the PDG Shopping Cart User Guide and in Chapter 8 of this Technical Reference Guide.

Key form input (required)

The Add action knows which product the customer wants to add to his Shopping Cart from the Key form input. It is a hidden input. The value for this input should be the SKU of the product you wish to display. This code should be the same code as the product code found in the product definition in the product database file. For more information on product codes and product definitions, refer to the 'Maintaining Products' section in Chapter 6 of the PDG Shopping Cart User Guide. The input's name must be "Key."

Reference form input (optional)

This form input allows PDG Shopping Cart to create a Return action call embedded in the Web page that results from the Add action. The value of this input should be a URL. The input's name must be "Reference."

Quantity form input (required)

This form input specifies the amount of the product to add to the customer's Shopping Cart. If this value is '0' or negative, then no product will be added to the Shopping Cart. The input's name must be "Qty."

Note: "Qty" should be a text form input in order to allow the customer to change the quantity before adding the product to his Shopping Cart.

Opt[#] form inputs (optional)

The Options form inputs allow the customer to choose option selections for the product to be added to the Shopping Cart. Option values available for each selection should match the options defined in the product database files.

The Options form input's name must follow this syntax:

Opt[zero-based index of the option]

PDG Shopping Cart will generate these inputs automatically if the PreAdd action is used to generate this page. The inputs are generated from the product and option definitions in the product database files. For more information on defining product options, refer to the 'Maintaining Products' section in Chapter 6 of the PDG Shopping Cart User Guide.

MultiAdd Action

This action actually adds a wide array of products to the customer's Shopping Cart. The MultiAdd action allows shoppers to select options and quantities for several different products and add them to their Cart with a single click. Like the Add action, this action requires product codes, quantities, and any available options passed to it in form inputs.

A page with the MultiAdd action will be automatically generated whenever the MultiPreAdd action is called. However, you can also customize the Item template to contain the MultiAdd action if you are using the REPEAT or REPEAT NUMBER tags.

Here is an example of a sample Item template with REPEAT NUMBER tags followed by the HTML that it will generate:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
<!--BEGIN_REPEAT1-->
<tr>
```

```

Qty: <input type="edit" name="MQTY1" value="[initial value]"
size="[size of field]">
Item Options: <!--ITEM_OPT0-->[more options...]
<input type="submit" name="!--ADD_BUTTON--" value="Add Me">
</tr>
<!--END_REPEAT2-->
<input type="submit" name="MULTIADD" value="Add the Whole
Page">
</form>

```

Generates the following HTML on the page:

```

<form action="[location of PDG Shopping Cart executable]"
method="post">
<input type="hidden" name="MSKU1" value="[product code]">
<tr>
Qty: <input type="edit" name="MQTY1" value="[initial value]"
size="[size of field]">
Item Options: <select name="MOPT1_0">
<option value="0">[selection 0 for option 0]</option>
<option value="1">[selection 1 for option 0]</option>
<option value="2">[selection 2 for option 0]</option>
</select>
[more options...]
<input type="submit" name="MADD.1" value="[text to appear on but-
ton]">
</tr>
<tr>
<input type="hidden" name="MSKU2" value="[product code]">
Qty: <input type="edit" name="MQTY2" value="[initial value]"
size="[size of field]">
Item Options: <select name="MOPT2_0">
<option value="0">[selection 0 for option 0]</option>
<option value="1">[selection 1 for option 0]</option>
<option value="2">[selection 2 for option 0]</option>
</select>
[more options...]
<input type="submit" name="MADD.2" value="[text to appear on but-
ton]">
</tr>
<input type="submit" name="MULTIADD" value="[text to appear on but-
ton]">
</form>

```

Note: If you are displaying different products altogether using the REPEAT tags, the output will be similar to what is shown above. However, for displaying options using the REPEAT tags, you will need to utilize the OPTIONS

tags. For more information on the `OPTIONS`, `REPEAT` and `REPEAT NUMBER` tags please see 'HTML Template Files' in Chapter 8 of this Technical Reference Guide.

The index numbers on the product codes correspond to the numbers entered in the `!---BEGIN_REPEATx---` and `!---END_REPEATy---` tags. This range (x to y , inclusive) determines how many instances of the product will appear on the page. You must also note that options and option selections are indexed beginning with zero. This is so PDG Shopping Cart knows which selection goes with which option, and likewise which options and quantities go with which instance of the product on the template.

The submit input's name must be "MultiAdd." The `MSKU[#]`, `MQTY[#]` and `MOPT[#]_[#]` inputs are all required. These inputs will be automatically generated from the tags within the `REPEAT NUMBER` tags. If a product has no options, the tag that creates that input will be ignored. All other form inputs are optional.

You may also notice the `MADD.[#]` form inputs, one for each product instance on the template. These are generated by the `!---ADD_BUTTON---` tag. These are for adding a single line or instance of the product instead of the entire page. This action will be covered in its own section, following this one.

PDG Shopping Cart uses the Basket Template File to display the contents of the customer's Shopping Cart after adding the products to the basket. For information on this template file, refer to the 'HTML Template Files' sections in Chapter 5 of the PDG Shopping Cart User Guide and in Chapter 8 of this Technical Reference Guide.

MSKU[#] form input (required)

The MultiAdd action knows which product the customer wants to add from the `MSKU[#]` form input. The `[#]` is the index number that will be applied to all of the inputs which pertain to each particular instance of the product on the MultiAdd template. This is a hidden input. The value for this input should be the same code as the product code found in the product definition in the product database file. For more information on product codes and product definitions, refer to the 'Maintaining Products' section in Chapter 6 of the PDG Shopping Cart User Guide. The input's name must be "`MSKU[#]`."

MQty[#] form input (required)

The MultiAdd action knows what quantity of which product instance to add by this form input. This input should be a text field so the shopper can modify quantities on the MultiAdd page. The `[#]` corresponds to the index number on the `MSKU[#]` input for the particular product instance. The input's name must be "`MQty[#]`."

MOpt[#]_[#] form input (required)

A page containing the MultiAdd action also allows a shopper to select all of the options for several product listings. The MOpt[#]_[#] input will generally be a drop-down menu containing all of the option selections for a particular option.

When this input is passed with the MultiAdd action, PDG Shopping Cart distinguishes it from others by the index numbers. The first [#] corresponds to the index number for the product's instance on the page (same as MSKU[#] and MQty[#]). The second [#] indicates which option for the particular product is being configured. Finally, the number of the customer's option selection is passed as the input's 'value.' The input's name must be "MOpt[#]_[#]."

Reference form input (optional)

This form input allows PDG Shopping Cart to create a Return action call embedded in the Web page that results from the MultiAdd action. This should be a hidden input and the value of this input should be a URL. The input's name must be "Reference."

MAdd.[#] Action

As described in the 'MultiAdd Action' section, the MAdd.[#] action may also be used on sample-Item pages which utilize the REPEAT or REPEAT NUMBER tags. There will be a MAdd.[#] input that corresponds to each and every product listing on these sample-Item pages. Instead of adding the entire page, the MAdd.[#] action only adds the product listing that corresponds to the MSKU[#] input of the same index number. The MAdd.[#] and the MultiAdd actions may be used on the same page because they take the same form inputs. Here is an example of how the MAdd.[#] action may be used:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
<input type="hidden" name="MSKU1" value="[product code]">
<tr>
<input type="edit" name="MQTY1" value="[initial value]" size="[size of
field]">
<select name="MOPT1_0">
<option value="0">[selection 0 for option 0]</option>
[more option selections]
</select>
```

```
[more options...]  
<input type="submit" name="MADD.1" value="[text to appear on  
button]">  
...  
</form>
```

Note: If you are displaying different products altogether using the REPEAT tags, the output will be similar to what is shown above. For more information on the REPEAT and REPEAT NUMBER tags please see Chapter 8 of this Technical Reference Guide.

MSKU[#] form input (required)

The MultiAdd action “knows” which product the customer wants to add from the MSKU[#] form input. The [#] is the index number that will be applied to all of the form inputs pertaining to each particular instance of the product on the sample-Item template. MSKU[#] is a hidden input. The value for this input should be the same code as the product code found in the product definition in the product database file. For more information on product codes and product definitions, refer to the ‘Maintaining Products’ section in Chapter 6 of the PDG Shopping Cart User Guide. The input's name must be “MSKU[#].”

MQty[#] form input (required)

The MultiAdd action “knows” what quantity of which product to add by this input. This input should be a text field so that the shopper can modify quantities on the sample-Item page. The [#] corresponds to the index number on the MSKU[#] form input for each particular product instance. The input's name must be “MQty[#].”

MOpt[#]_[#] form input (required)

A page containing the MAdd.[#] action also allows a shopper to select the options for any product instance on the sample-item page. The MOpt[#]_[#] input will generally consist of a drop-down menu containing all of the option selections for a particular option.

When this input is passed with the MultiAdd action, PDG Shopping Cart distinguishes it from others by the index numbers. The first [#] corresponds to the index number for the product's instance on the page (same as MSKU[#]). The second [#] indicates which option for the particular product is being configured. Finally, the number of the customer's option selection is passed as the input's value. The input's name must be “MOpt[#]_[#].”

Reference form input (optional)

This form input allows PDG Shopping Cart to create a Return action call embedded in the Web page resulting from the MultiAdd action. This should be a hidden input and the value of this input should be a URL.

The input's name must be "Reference."

OptFullAdd Action

This action is generally invoked from the PreAdd action. When the sample-Item template that is called from a PreAdd action utilizes the OPTIONS tags, all of the possible option combinations for a single product will be displayed on that page. For instance, suppose a product has two options with three possible selections for each. With the OPTIONS tags utilized, the Item page will display nine instances of the product, each with a unique combination of option selections. The customer will still be able to indicate quantities for each product display, depending upon his preference. The OptFullAdd action will then be used to add all of the selected quantities from that page with a single click.

Note: It should be noted that an Item page with the OptFullAdd action is only able to display the option selections for a single product as entered in the Merchant Administrator.

Like the Add and MultiAdd actions, this action requires at least one quantity input. Unlike the Add and MultiAdd actions, however, the product code and option selections will be appended to the quantity input when the page is drawn. Thus, each quantity input will contain the product code and the index numbers for the corresponding option combination.

For more information on the OPTIONS tags, see Chapter 8 of this Technical Reference Guide.

The following is an example of an Item template with OPTIONS tags followed by the HTML that it will generate:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
<td>Full SKU</td><td>Quantity</td><td><!--DESC_OPT0-->
</td><td><!--DESC_OPT1--></td><td><!--OPT_LINE_PRICE--></td>
<!--BEGIN_OPTIONS-->
<tr>
```

```

<td><!--OPT_LINE_SKU--></td>
<td><input type="edit" name="!--OPT_LINE_QTY--" value="[initial
value]" size="[size of field]"></td>
<td><!--VALUE_OPT0--></td><td><!--VALUE_OPT1--></td>
<input type="submit" name="!--OPT_LINE_ADD--" value="Add Me">
</tr>
<!--END_OPTIONS-->
<input type="submit" name="OPTFULLADD" value="Add the Whole
Page">
</form>

```

Generates the following HTML code on your Web page:

```

<form action="[location of PDG Shopping Cart executable]"
method="post">
<td>Full SKU</td><td>Quantity<td><!--DESC_OPT0-->/
<td><!--DESC_OPT1--></td><td><!--OPT_LINE_PRICE--></td>
<tr>
<td>[full product code]</td>
<td><input type="edit" name="OQTY.[product code],0,0" value="[initial
value]" size="[size of field]"></td>
<td>[selection 0 for option 0]</td>
<td>[selection 0 for option 1]</td>
<input type="submit" name="OADD.[product code],0,0" value="Add
Me">
</tr>
<tr>
<td>[full product code]</td>
<td><input type="edit" name="OQTY.[product code],1,0" value="[initial
value]" size="[size of field]"></td>
<td>[selection 1 for option 0]</td>
<td>[selection 0 for option 1]</td>
<input type="submit" name="OADD.[product code],1,0" value="Add
Me">
</tr>
[more product instances]
<input type="submit" name="OPTFULLADD" value="Add the Whole
Page">
</form>

```

The number of product instances that appear on this page depends solely on the number of options and their corresponding selections. As stated above, if your product has two options with three possible selections each, then the Item page will contain nine product instances. Likewise if your product has three options, each with two possible selections, The Item page will contain eight product instances.

In the above example, each instance is formatted in a single row. The index numbers attached to the quantity inputs correspond to the selection numbers for the particular options. The submit input's name must be "OptFullAdd." At least one OQTY input is required. These OQty inputs will be automatically generated from the `!---OPT_LINE_QTY---` tag within the `OPTIONS` tags on the Item template. All other form inputs are optional.

You may also notice the `OADD.[#]` form inputs, one for each product instance on the template. These are generated by the `!---OPT_LINE_ADD---` tag. These are for adding a single line or instance of the product instead of the entire page. This action will be covered in its own section, following this one.

PDG Shopping Cart uses the Basket Template File to display the contents of the customer's Shopping Cart after adding the products to the basket. For information on this template file, refer to the 'HTML Template Files' sections in Chapter 5 of this Technical Reference Guide.

OQTY form input (required)

The OptFullAdd action "knows" which product instance the customer wants to add from the OQTY form input. This input also carries the product code and the option selection numbers that pertain to that product instance. Thus, the input's name is unique for each product instance on the template. The input's name must follow the format:

OQTY.[SKU],[zero based index of option selection],[zero based index of option selection]...

There will be as many numbers following the product code as there are available options for the selected product. As stated before, these inputs will be automatically generated from the `!---OPT_LINE_QTY---` tag within the `OPTIONS` tags on the Item template.

Pricecode form input (optional)

If the OptFullAdd link is to use a specific Price Category, then this information needs to be included in the Pricecode form input. The value for this form input should be equal to the code of the appropriate pricing category as defined in the Merchant Administrator. If this field is not provided with the OptFullAdd action, then the unit price indicated in the product definition will be used.

This field can either be hard-coded, or set up as a text entry box, where the customer must supply the pricing category code in order to obtain that category's product pricing.

Reference form input (optional)

This form input allows PDG Shopping Cart to create a Return action call embedded in the Web page that results from the OptFullAdd action. This should be a hidden input and the value of this input should be a URL. The input's name must be "Reference."

OAdd Action

As described in the OptFullAdd action section, the OAdd action may also be used on sample-item pages which utilize the OPTIONS tags. There will be an OAdd input that corresponds to each product listing on these Item pages. Instead of adding the entire page, the OAdd action only adds the indicated quantity of the corresponding product instance. The OAdd and the OptFullAdd actions may be used on the same page because they take the same form inputs. Here is an example of how the OAdd action may be used:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
<input type="edit" name="OQTY.[product code],0,0" value="[initial
value]" size="[size of field]">
<input type="submit" name="OADD.[product code],0,0" value="[text to
appear on button]">
</form>
```

OQTY form input (required)

The OptFullAdd action "knows" which product instance the customer wants to add from the OQTY form input. This input will also carry the product code and the option selection numbers that pertain to that product instance. Thus, the input's name is unique for each product instance. The input's name must follow the format:

```
OQTY.[SKU],[zero based index of option selection],[zero based index of
option selection]...
```

There will be as many numbers following the product code as there are available options for the selected product. As stated before, these inputs will be automatically generated from the !---OPT_LINE_QTY--- tag within the OPTIONS tags.

Pricecode form input (optional)

If the OAdd link is to use a specific Price Category, then this information needs to be included in the Pricecode form input. The value for this form input should be equal to the code of the appropriate pricing category as defined in the Merchant Administrator. If this field is not provided with the OAdd action, then the unit price, including all options, as specified in the product definition will be used.

This field can either be hard-coded, or set up as a text entry box, where the customer must supply the pricing category code in order to obtain that category's pricing from your Web store.

Reference form input (optional)

This form input allows PDG Shopping Cart to create a Return action call embedded in the Web page resulting from the MultiAdd action. This should be a hidden input and the value of this input should be a URL.

The input's name must be "Reference."

Remove Action

This action removes one product from a customer's Shopping Cart. In almost all cases, this action should only be called from a Web page displaying the Shopping Cart contents pregenerated by PDG Shopping Cart.

Here is the syntax of the Remove action used in an HTML document. This action will remove the first item in the customer's Shopping Cart.

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
<input type="submit" name="REMOVE0" value="[text to appear on
button]">
</form>
```

The submit input's name must follow this syntax:

Remove[zero-based index of the product to remove]

This action results in the Shopping Cart being re-displayed, with the first item removed and totals updated. PDG Shopping Cart uses the Basket Template File to display the contents of the customer's Shopping Cart after removing the specified product. Because the contents of a customer's Shopping Cart can vary, Remove actions should only be called in situations where the product is known to be in the Cart. PDG Shopping Cart will automatically generate the appropriate Remove calls when displaying the Cart's contents.

RemoveAll Action

This action allows a user to completely empty their Shopping Cart. This action may be called from PDG's pregenerated Shopping Cart display or from a static page, such as your Web store's home page.

Here is the syntax for using the RemoveAll action in an HTML page:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
  <input type="submit" name="REMOVEALL" value="[text to appear on
button]">
</form>
```

The submit input's name must be "RemoveAll."

This action results in the customer's current Shopping Cart being displayed (which is empty). PDG Shopping Cart uses the Basket Empty Template File to display the contents of the customer's Shopping Cart after removing all products. For more information on this template file, refer to the 'HTML Template Files' sections in Chapter 5 of the PDG Shopping Cart User Guide and in Chapter 8 of this Technical Reference Guide.

Display Action

This action allows a customer to display the current contents of his Shopping Cart. In addition to the content display, PDG Shopping Cart also allows the customer to modify the contents of his Cart. From the Shopping Cart display, a customer can remove a specific item, remove all of the items, recalculate the Cart's contents (assuming that the customer changed item quantities), or submit the contents for purchase.

Here is the syntax for calling the Display action from an HTML page:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
  <input type="submit" name="DISPLAY" value="[text to appear on
button]">
</form>
```

The submit input's name must be "Display."

This action results in the customer's current Shopping Cart being displayed. PDG Shopping Cart uses the Basket Template File to display the contents of

the customer's Shopping Cart. For information on this template file, refer to the 'HTML Template Files' section in Chapter 5 of the PDG Shopping Cart User Guide.

Recalc Action

This action allows a customer to recalculate his Shopping Cart's totals if they have modified any of the quantities. If the user has changed any of the quantity fields from a PDG Shopping Cart Basket page, this action instructs PDG Shopping Cart to recalculate the customer's totals and create a new Basket page with the new quantities in effect.

Here is the syntax for calling the Recalc action from an HTML page:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
<input type="text" name="QTY[zero-based index of product]"
value="[current quantity for product value]" size="[size of field]">
[other QTY[#] inputs...]
<input type="submit" name="RECALC" value="[text to appear on
button]">
</form>
```

The submit input's name must be "Recalc." At least one Qty input is required for the Recalc action.

This action results in the customer's current Shopping Cart being recalculated considering the updated quantities and re-displayed. PDG Shopping Cart uses the Basket Template File to display the contents of the customer's Shopping Cart.

Quantity form inputs (required)

These form inputs allow the customer to specify new quantities for products that are currently in their Shopping Cart.

If a quantity form input is omitted from the form for a specific product, then PDG Shopping Cart will not change the item's quantity.

The form input's name must follow this syntax:

Qty[zero-based index of the product in the Cart]

PDG Shopping Cart will generate these inputs automatically if the Display action is used to display the contents of the customer's Shopping Cart. The inputs are generated from the current contents of the customer's Shopping Cart.

Return Action

This action allows a customer to return to browsing your Web store after adding an item to his Shopping Cart. If the customer's Cart was displayed as a result of an Add or PreAdd action, this action instructs PDG Shopping Cart to reload the Web page that contained the Add or PreAdd action. However, this may be overridden in one of two ways. If you have specified a “Destination of Shop Some More Button” (see the ‘Cart Options’ section in Chapter 5 for more information) in the Merchant Administrator, then this URL will be loaded. Also, if the ‘reference’ input is used (as shown below), then its value will be the page that is loaded instead.

Here is the syntax for calling the Return action from an HTML page:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
  <input type="hidden" name="REFERENCE" value="[URL to load]">
  <input type="submit" name="RETURN" value="[text to appear on
button]">
</form>
```

The submit input's name must be “Return.” The Reference form input is optional for this action.

PDG Shopping Cart loads the URL specified by the Reference form input when executing the Return action.

Reference form input (optional)

This form input allows PDG Shopping Cart to specify the page to which the Return action call directs your customers. The value of this input should be a complete URL. The input's name must be “reference.”

Checkout Action

When a customer is ready to proceed to checkout with all of the items in his Shopping Cart, this action initiates the purchase process. The resulting Web page of the Checkout action should request shipping and billing information from the customer. Credit card information will not be acquired from the customer at this time.

The response Web page generated from a Checkout action should contain a call to PDG Shopping Cart with the Verify action.

Here is an example of the syntax used for calling the Checkout action from an HTML page:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
<input type="submit" name="CHECKOUT" value="[text to appear on
button]">
<input type="hidden" name="GPRICECODE" value="[pricing category
code]">
</form>
```

The submit input's name must be "Checkout."

The page produced by this action will display all of the order information, as well as request the shipping and billing address information from the customer. PDG Shopping Cart uses the Checkout Template File to display the Cart and obtain the shipping and billing information. For information on this and other template files, refer to the 'HTML Template Files' sections in Chapter 5 of the PDG Shopping Cart User Guide and in Chapter 8 of this Technical Reference Guide.

GPricecode form input

This form input allows for a global price code. This means, any pricing category code that is entered for this value will be applied to all of the products in the current Shopping Cart. This field can either be hard-coded, or setup as a text entry box, where the customer must supply the pricing category code in order to obtain that category's pricing from your Web store. The input's name must be "gpricecode."

Verify Action

The Verify action receives the customer's shipping and billing information. In response to this action, PDG Shopping Cart will display the customer's current order and all entered shipping and billing information for verification by the customer. If the customer has indicated that they will purchase online with a credit card, PDG Shopping Cart will now request the credit card information from the customer. The response Web page generated from a Verify action should contain a call to PDG Shopping Cart with the BuyIt action.

There are several input fields that the Verify action needs to process an order properly. Here they are along with the syntax for calling the Verify action from an HTML page:

```

<form action="[location of PDG Shopping Cart executable]"
method="post">
  <input type="text" name="contactname">
  <input type="text" name="contactemail">
  <input type="text" name="billfirstname">
  <input type="text" name="billlastname">
  <input type="text" name="billstreet1">
  <input type="text" name="billstreet2">
  <input type="text" name="billcity">
  <input type="text" name="billstate">
  <input type="text" name="billprovince">
  <input type="text" name="billzip">
  <input type="text" name="billcountry">
  <input type="text" name="billtext">
  <input type="text" name="contactname">
  <select name="shipmethod">
  <option value="[shipping method 1]">
  [...more shipping methods]
  </select>
  <input type="checkbox" name="shiptobilladdr" value="yes">
  <input type="text" name="shipname">
  <input type="text" name="shipstreet1">
  <input type="text" name="shipstreet2">
  <input type="text" name="shipcity">
  <input type="text" name="shipstate">
  <input type="text" name="shipprovince">
  <input type="text" name="shipzip">
  <input type="text" name="shipcountry">
  <input type="submit" name="VERIFY" value="[text to appear on but-
ton]">
  <input type="text" name="pass...">
</form>

```

The following input fields are required under all circumstances: contactname, contactemail, billfirstname, billlastname, billstreet1, billcity, billstate, billzip, billcountry, shipmethod

If the chosen shipping method does not have the <NoShipInfo> tag, then the following fields are required: shipname, shipstreet1, shipcity, shipstate, shipzip, shipcountry. However these fields become optional if the “shiptobil- laddr” checkbox has been checked.

The following input fields are optional under all circumstances: billmid- dleinit, billprovince, billstreet2, shipprovince, shipstreet2

ContactName form input (required)

This is the customer's name as it will appear on the invoice and in e-mail notifications. It should contain both the first and last name of the customer. The input's name must be "contactname."

ContactEmail form input (required)

This is the customer's e-mail address. It appears on the invoice and in verification e-mails. It is also used as the e-mail address to which the automatic invoice e-mail is sent after verification of the order. The input's name must be "contactemail."

BillFirstName form input (required)

This is the first name of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this should be the first name of the name on the credit card. If the customer intends to pay by check, then this should be the first name of the person signing the check. The input's name must be "billfirstname."

BillMiddleInit form input (optional)

This optional field can hold the middle initial of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this should be the middle initial of the name on the credit card. If the customer intends to pay by check, then this should be the middle initial of the person signing the check. The input's name must be "billmiddleinit."

BillLastName form input (required)

This is the last name of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this should be the last name of the name on the credit card. If the customer intends to pay by check, then this should be the last name of the person signing the check. The input's name must be "billlastname."

BillStreet1 form input (required)

This is the first of two lines for providing the street address of the billing address. If the customer intends to pay by credit card, this should be the street address to which the credit card bill is sent. If the customer intends to pay by check, then this should be the street address of the person signing the check. The input's name must be "billstreet1."

BillStreet2 form input (optional)

This is the second of two lines for providing the street address of the billing address. If the customer intends to pay by credit card, this should be the street address to which the credit card bill is sent. If the customer intends to pay by check, then this should be the street address of the person signing the check. The customer may leave this field input blank if two lines are not needed. The input's name must be "billstreet2."

BillCity form input (required)

This is the city of the address of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this city should be the city to which the credit card bill is sent. If the customer intends to pay by check, then this should be the city of the person signing the check. The input's name must be "billcity."

BillState form input (required)

This is the two-letter state code of the address of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this should be the state to which the credit card bill is sent. If the customer intends to pay by check, then this should be the state of the person signing the check. If you are using UPS costing, then this field must be the two letter code for the state of destination. If this field is 'ZZ,' then the BillProvince field is used instead of the BillState field in the billing address. The input's name must be "billstate."

BillProvince form input (optional)

This optional field is the province of the address of the person who will be billed for the purchase. This field should only be filled in by Canadian customers. This field is only applicable if BillState is 'ZZ.' If the customer intends to pay by credit card, then this should be the province to which the credit card bill is sent. If the customer intends to pay by check, then this should be the province of the person signing the check. The input's name must be "billprovince."

BillZip form input (required)

This is the postal code of the address of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this Postal Code should be the ZIP Code to which the credit card bill is sent. If the customer intends to pay by check, then this should be the ZIP code of the person signing the check. The input's name must be "billzip."

BillCountry form input (required)

This is the country of the address of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this country should be the country to which the credit card bill is sent. If the customer intends to pay by check, then this should be the country of the person who is signing the check. If you are using UPS to obtain shipping costs, this field must be the UPS two-letter country code, preceded with a tilde ('~'). PDG Shopping Cart will convert the two-letter code back to the country's full name when producing shipping and billing addresses. The input's name must be "billcountry."

BillText form input (optional)

This form input can be used to collect any additional information that may be needed for customer billing. You have the option of labeling this field whatever you want on the Checkout page. An example of what this may be used for is the customer's county. The input's name must be "billtext."

ShipMethod form input (required)

This form input allows the customer to decide which method of shipping (from your Web store's available methods) to use to ship the products. The value passed to PDG Shopping Cart through the form should be the zero-based index of the chosen method as listed in the 'Shipping Rules' section of your Merchant Administrator.

PDG Shopping Cart will automatically generate and place this form input in the customer's HTML page through the use of a tag of the Checkout Template File. This form input may also be placed manually in the template. Refer to the 'HTML Template Files' sections in Chapter 5 of the PDG Shopping Cart User Guide, and in Chapter 8 of this Technical Reference Guide, for more information on the Checkout Template File.

Depending upon the choice of shipping method and configuration of the chosen shipping method, the rest of the fields may or may not be required. If the chosen shipping method has the tag <NoShipInfo> in the Shipping.conf file, then all other fields beginning with the letters "SHIP" are ignored and not required. For more information on the <NoShipInfo> tag and the Shipping.conf file, refer to the 'Configuring Shipping Options' section in Chapter 8 of the PDG Shopping Cart User Guide.

ShipToBillAddr form input (required)

The form input allows the customer to use the data entered for the billing address as the shipping address for the purchase. This input should be a checkbox input. If it is checked, then PDG Shopping Cart will ignore and not validate the contents of any other form input beginning with the letters SHIP.

If this checkbox is checked, then PDG Shopping Cart will use BillState to determine if sales tax is applicable.

ShipName form input (required or optional)

This is the name to which the products should be shipped. This field will be optional if the “shiptobilladdr” box is checked, or if there is no shipping information required for the selected shipping method. The input’s name must be “shipname.”

ShipStreet1 form input (required or optional)

This is the first of two lines provided for the street address to which the products will be shipped. This field will be optional if the “shiptobilladdr” box is checked, or if there is no shipping information required for the selected shipping method. The input’s name must be “shipstreet1.”

ShipStreet2 form input (optional)

This is the second of two lines provided for the street address to which the products will be shipped. The input’s name must be “shipstreet2.”

ShipCity form input (required or optional)

This is the city to which the products will be shipped. This field will be optional if the “shiptobilladdr” box is checked, or if there is no shipping information required for the selected shipping method. The input’s name must be “shipcity.”

ShipState form input (required or optional)

This is the two-letter state code to which the products will be shipped. This field's data is also used normally to determine if sales tax is applicable to the order. If you are using UPS to obtain shipping costs, this field must be the two-letter code for the state. If this field is ‘ZZ,’ then the ShipProvince field is used instead of ShipState when producing address information.

This field will be optional if the “shiptobilladdr” box is checked, or if there is no shipping information required for the selected shipping method. The input’s name must be “shipstate.”

ShipProvince form input (optional)

This field provides an optional entry for the province of the destination address. PDG Shopping Cart only uses this field if ShipState is 'ZZ.' The input's name must be "shipprovince."

ShipZIP form input (required or optional)

This is the Postal Code to which the products will be shipped. This field will be optional if the "shiptobilladdr" box is checked, or if there is no shipping information required for the selected shipping method. The input's name must be "shipzip."

ShipCountry form input (required or optional)

This is the country to which the products will be shipped. If you are using UPS to obtain real-time shipping costs, this field must be the UPS two-letter code for the country, preceded with a tilde (~). PDG Shopping Cart will convert the two-letter code back into the country's full name when producing address information. This field will be optional if the "shiptobilladdr" box is checked, or if there is no shipping information required for the selected shipping method. The input's name must be "shipcountry."

GPricecode form input (optional)

This form input allows for a global price code. This means, whatever pricing category code is entered for this value will be applied to all of the products in the current Shopping Cart. This field can either be hard-coded, or set up as a text entry box, where the customer must supply the pricing category code in order to obtain that category's product pricing. If this input only appears on the Verify page, pricing updates will not be displayed real-time. However they will appear on customer and vendor e-mail notifications.

Pass form inputs (optional) (Passthrough Variables)

Any form input whose name begins with the letters "Pass" is passed on through PDG Shopping Cart into the output without any additional processing. These form inputs allow for the collection of customer data that is needed by the merchant to process the order, but not required by PDG Shopping Cart.

BuyIt Action

The BuyIt action receives the customer's complete shipping, billing, and payment information. At this point an invoice is generated, and a copy of the invoice can be e-mailed to the customer and vendor, as well as stored in a log for later processing. In addition to the logging and e-mails, PDG Shopping Cart can authorize credit card orders by communicating with a payment authorization service. At this time, PDG Shopping Cart also decrements inventory appropriately, if inventory is being tracked.

Here is the syntax for calling the BuyIt action from an HTML page:

```

<form action="[location of PDG Shopping Cart executable]"
method="post">
  <input type="hidden" name="contactname">
  <input type="hidden" name="contactemail">
  <input type="hidden" name="billfirstname">
  <input type="hidden" name="billlastname">
  <input type="hidden" name="billstreet1">
  <input type="hidden" name="billstreet2">
  <input type="hidden" name="billcity">
  <input type="hidden" name="billstate">
  <input type="hidden" name="billprovince">
  <input type="hidden" name="billzip">
  <input type="hidden" name="billcountry">
  <input type="hidden" name="billtext">
  <input type="hidden" name="contactname">
  <input type="hidden" name="shipmethod">
  <input type="hidden" name="shipname">
  <input type="hidden" name="shipstreet1">
  <input type="hidden" name="shipstreet2">
  <input type="hidden" name="shipcity">
  <input type="hidden" name="shipstate">
  <input type="hidden" name="shipprovince">
  <input type="hidden" name="shipzip">
  <input type="hidden" name="shipcountry">
  <input type="hidden" name="pass...">
  <select name="billmethod">
    <option> Option 1
    <option> Option 2
    [other bill methods accepted...]
  </select>
  <select name="cardtype">
    <option value="[some card type]">

```

```

[other card types accepted]
</select>
<input type="text" name="cardnumber">
<input type="text" name="cardexp">
<input type="submit" name="BUYIT" value="[text to appear on but-
ton]">
</form>

```

Invoice Generation

The BuyIt action causes PDG Shopping Cart to generate an invoice for the order. When the invoice is generated, a unique invoice number is generated to identify the invoice. The invoice number is comprised of some predefined text and a unique five digit number. The text is prefixed to the five digit number, and is defined in the ‘Order Logging/Retrieval’ section of the Merchant Administrator. The five digit number is generated from the contents of the Invoice Counter File, whose location is specified in Shopper.conf. For more information on the Shopper.conf file, you will need to contact PDG.

If inventory is being tracked, then products are taken out of inventory when an invoice is created for the order.

ContactName form input (required)

This is the customer's name as it will appear on the invoice and in e-mails. It should contain both the first and last name of the customer. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be “contactname.”

ContactEmail form input (required)

This is the customer's e-mail address. It appears on the invoice and in verification e-mails. It is also used as the e-mail address to use when sending the automatic invoice e-mail after accepting the order. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be “contactemail.”

BillFirstName form input (required)

This is the first name of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this should be the first name on the credit card. If the customer intends to pay by check, then this should be the first name of the person signing the check. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart.

BillLastName form input (required)

This is the last name of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this should be the last name of the name on the credit card. If the customer intends to pay by check, then this should be the last name of the person signing the check. The input's name must be "billlastname."

BillStreet1 form input (required)

This is the first of two lines for providing the street address of the billing address. If the customer intends to pay by credit card, this should be the street address to which the credit card bill is sent. If the customer intends to pay by check, then this should be the street address of the person signing the check. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "billstreet1."

BillStreet2 form input (optional)

This is the second of two lines for providing the street address of the billing address. If the customer intends to pay by credit card, this should be the street address to which the credit card bill is sent. If the customer intends to pay by check, then this should be the street address of the person signing the check. The customer may have left this field input blank if two lines were not needed. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "billstreet2."

BillCity form input (required)

This is the city of the address of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this city should be the city to which the credit card bill is sent. If the customer intends to pay by check, then this should be the city of the person signing the check. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "billcity."

BillState form input (required)

This is the two-letter state code of the address of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this state should be the state to which the credit card bill is sent. If the customer intends to pay by check, then this should be the state of the person signing the check. If this field is 'ZZ,' then the BillProvince field is used instead of BillState when producing address information. This is a hidden field, and should have

been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "bill state."

BillProvince form input (optional)

This optional field provides the province of the address of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this should be the province to which the credit card bill is sent. If the customer intends to pay by check, then this should be the province of the person signing the check. This field is only applicable if BillState is 'ZZ.' The input's name must be "billprovince."

BillZip form input (required)

This is the ZIP Code of the address of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this ZIP Code should be the ZIP Code to which the credit card bill is sent. If the customer intends to pay by check, then this should be the ZIP code of the person signing the check. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "billzip."

BillCountry form input (required)

This is the country of the address of the person who will be billed for the purchase. If the customer intends to pay by credit card, then this country should be the country to which the credit card bill is sent. If the customer intends to pay by check, then this should be the country of the person signing the check. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "billcountry."

BillText form input (optional)

This form input may contain whatever was entered by the customer in this field on the Checkout page. An example of what this may be used for might be the customer's county of residence. The input's name must be "billtext."

ShipMethod form input (required)

This hidden form input indicates the customer's choice of shipment methods to be used to ship his order. The value of this field will be the zero-based index of the shipping method chosen as listed in the 'Shipping Rules' section of the Merchant Administrator.

Depending on the choice of shipping method and configuration of the chosen shipping method, the rest of the fields may not be required. If the chosen shipping method has the tag `<NoShipInfo>` in the `Shipping.conf` file, then PDG Shopping Cart will not place any more hidden input fields in the HTML page. The input's name must be "shipmethod."

ShipName form input (required or optional)

This is the name to which the products should be shipped. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "shipname."

ShipStreet1 form input (required or optional)

This is the first of two lines provided for the street address to which the products will be shipped. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "shipstreet1."

ShipStreet2 form input (optional)

This is the second of two lines provided for the street address to which the products will be shipped. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "shipstreet2."

ShipCity form input (required or optional)

This is the city to which the products will be shipped. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "shipcity."

ShipState form input (required or optional)

This is the two-letter state code to which the products will be shipped. This field's data may also be used to determine if sales tax is applicable to the order. If this field is 'ZZ,' then the ShipProvince field is used instead of ShipState when producing address information. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "shipstate."

ShipProvince form input (optional)

This field provides an optional entry for the province of the destination address. PDG Shopping Cart only uses this field if ShipState is 'ZZ.' The input's name must be "shipprovince."

ShipZIP form input (required or optional)

This is the ZIP Code to which the products will be shipped. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "shipzip."

ShipCountry form input (required or optional)

This is the country to which the products will be shipped. This is a hidden field, and should have been placed in the HTML page by the Checkout action of PDG Shopping Cart. The input's name must be "shipcountry."

BillMethod form input (required)

The BillMethod form input allows the customer to determine how they want to pay for the products out of the given payment methods. The Web store may list as many payment options as they would like. PDG Shopping Cart checks the value of this form input.

CardType form input (required or optional)

This input should be a drop-down menu that lists the credit card types that your Web store accepts. This field will be required if the customer has selected a credit card payment method. PDG Shopping Cart will automatically create this drop-down menu on the Verify page after the Verify action has been submitted. The input's name must be "cardtype."

CardNumber form input (required or optional)

This text input will be where the customer enters his credit card number. This field will be required if the customer has selected a credit card payment method. PDG Shopping Cart will automatically create this text field on the Verify page after the Verify action has been submitted. The input's name must be "cardnumber."

CardExp form input (required or optional)

This text input will be where the customer enters his credit card's expiration date. This field will be required if the customer has selected a credit card

payment method. PDG Shopping Cart will automatically create this text field on the Verify page after the Verify action has been submitted. The input's name must be "cardexp."

CCV2 form input (required or optional)

This text input will be where the customer enters his credit card's verification value. This field will be required if the customer has selected a credit card payment method. PDG Shopping Cart will automatically create this text field on the Verify page after the Verify action has been submitted. The input's name must be "ccv2."

BankName form input (required or optional)

This text input will be where the customer enters the bank name with which he has a checking account. This field will be required if the customer has selected to pay by online check. PDG Shopping Cart will automatically create this text field on the Verify page after the Verify action has been submitted. The input's name must be "bankname."

RouteNumber form input (required or optional)

This text input will be where the customer enters the routing number that corresponds with his bank. This field will be required if the customer has selected to pay by online check. PDG Shopping Cart will automatically create this text field on the Verify page after the Verify action has been submitted. The input's name must be "routenumber."

CheckNumber form input (required or optional)

This text input will be where the customer enters the number of the check with which he wishes to pay. This field will be required if the customer has selected to pay by online check. PDG Shopping Cart will automatically create this text field on the Verify page after the Verify action has been submitted. The input's name must be "checknumber."

AccountNumber form input (required or optional)

This text input will be where the customer enters his checking account number. This field will be required if the customer has selected to pay by online check. PDG Shopping Cart will automatically create this text field on the Verify page after the Verify action has been submitted. The input's name must be "accountnumber."

DriverLicenseNum form input (required or

optional)

This text input will be where the customer enters his valid driver license number. This field will be required if the customer has selected to pay by online check. PDG Shopping Cart will automatically create this text field on the Verify page after the Verify action has been submitted. The input's name must be "driverlicensenum."

DriverLicenseState form input (required or optional)

This text input will be where the customer enters the state for his valid driver license. This field will be required if the customer has selected to pay by online check. PDG Shopping Cart will automatically create this text field on the Verify page after the Verify action has been submitted. The input's name must be "driverlicensestate."

PhoneNumber form input (required or optional)

This text input will be where the customer enters his phone number, area code first. This field will be required if the customer has selected to pay by online check. PDG Shopping Cart will automatically create this text field on the Verify page after the Verify action has been submitted. The input's name must be "phonenumber."

Pass-- form inputs (optional) (Passthrough Variables)

Any form input whose name begins with the letters "pass" is passed on through PDG Shopping Cart into the submitted invoice without any additional processing. These form inputs allow for the collection of customer data that may be needed by the merchant to process the order, but not required by PDG Shopping Cart. If nothing is entered in the passthrough fields, then these form inputs will not be displayed on successive pages.

ListCategories Action

The ListCategories action will return the Category HTML template. The Category template is essentially a listing of all of your Web store's product cate-

gories. This page contains the category code, image, name, as well as a long description for each product category in your Web store. Each category listing also contains a link to a page that lists all of the products in the category and acts as that category's "home page."

Here is the syntax for calling the ListCategories action in an HTML page:

```
<form action="[location of PDG Shopping Cart executable]"
method="post">
  <input type="text" name="PRICECODE">
  <input type="hidden" name="TEMPLATE" value="[location of custom
template]">
  <input type="submit" name="LISTCATEGORIES" value="[text to
appear on button]">
</form>
```

The submit input's name must be "ListCategories."

Pricecode form input (optional)

If the ListCategories link is to use a specific Price Category, then this information needs to be included in the Pricecode form input. The value for this form input should be equal to the code of the appropriate pricing category that is defined in the Merchant Administrator. If this field is not provided with the ListCategories action, then the unit price defined in the product definition will be used.

This field can either be hard-coded, or set up as a text entry box, where the customer has to supply the pricing category code in order to obtain that category's unit price.

Template form input (optional)

By default, the ListCategories action calls the standard Category HTML template located in your 'PDG_Cart' subdirectory in your CGI directory. You may specify a different template file by creating a hidden input field. This field must be named "Template," and the value is the path to the new template file. This path is must be relative to the 'PDG_Cart' subdirectory.

Cart Action Examples

Below is a list of the available cart actions you can link from your web store pages:

Remember: If you are using Cart on a Windows server, these links will be to `shopper.exe`. If you are on a UNIX server, these links will be to `shopper.cgi`.

- List all products:
`http://www.mywebstore.com/cgi-bin/shopper?search=action&keywords=all`
- List all products using the template `SearchResult-custom.html`:
`http://www.mywebstore.com/cgi-bin/shopper?search=action&keywords=all&template=Templates/SearchResult-custom.html`
- List all products with the keyword shoes:
`http://www.mywebstore.com/cgi-bin/shopper?search=action&keywords=shoes`
- List all products with the keyword shoes without the keyword brown:
`http://www.mywebstore.com/cgi-bin/shopper?search=action&keywords=shoes+not+brown`
- List all products with the keyword shoes in category A000:
`http://www.mywebstore.com/cgi-bin/shopper?search=action&category=A000&keywords=shoes`
- List all categories:
`http://www.mywebstore.com/cgi-bin/shopper?listcategories=action`
- List all products in category A000:
`http://www.mywebstore.com/cgi-bin/shopper?search=action&category=A000`
- Search for sku A0005:
`http://www.mywebstore.com/cgi-bin/shopper?search=action&sku=A0005`
- Show Preadd page for product A0005:
`http://www.mywebstore.com/cgi-bin/shopper?preadd=action&key=A0005`
- Add product A0005 to cart (Cart 2004):
`http://www.mywebstore.com/cgi-bin/shopper?add=action&key=A0005`
- Add product A0005 to cart (Cart 1.x):
`http://www.mywebstore.com/cgi-bin/shopper?add=action&key=A0005&qty=1`
- Add 2 products A0005 to cart:
`http://www.mywebstore.com/cgi-bin/shopper?add=action&key=A0005&qty=2`
- Add 2 different products to cart:
`http://www.mywebstore.com/cgi-bin/commerce?multiadd=action&msku0=A0001&mqty0=1&msku1=A0005&mqty1=1`
- View Cart:
`http://www.mywebstore.com/cgi-bin/shopper?display=action`
- View Cart using a specified template (not the default template):
`http://www.mywebstore.com/cgi-bin/shopper?display=action&template=Templates/Basket-custom.html`

- Go to CheckOut page:
<http://www.mywebstore.com/cgi-bin/shopper?checkout=action>
- Show Preadd page for product A0005 using a specified item template (not the default template):
<http://www.mywebstore.com/cgi-bin/shopper?preadd=action&key=A0005&itemtemplate=Templates/Sample-Item-custom.html>
- Show Preadd page for several products at once (A0001, A0002, A0003):
<http://www.mywebstore.com/cgi-bin/shopper?multipreadd=action&keys=A0001,A0002,A0003>
- List all subcategories of a parent category A000:
<http://www.mywebstore.com/cgi-bin/shopper?listcategories=action&parent=A000>
- Display Basket template with no products, instead of BasketEmpty template if basket is empty:
<http://www.mywebstore.com/cgi-bin/shopper?display=action&emptyoverride=yes>

