



**cactusoft**

**CactuShop**  
ASP Shopping Cart  
Version 5.1

**User &  
Developer  
Manual**

Cactusoft Ltd.  
5<sup>th</sup> Floor, Tolworth Tower  
Surbiton, Surrey, KT6 7EL  
United Kingdom.

[www.cactushop.com](http://www.cactushop.com)

Copyright ©1999-2004  
Cactusoft Ltd.

## Introduction

CactuShop is an ASP shopping cart system based around established and widely adopted technologies on the Microsoft platform. The software is script-based and this, combined with a standard ODBC compliant database format, means it can be modified as much as required with commonly available development tools.

There are two versions of CactuShop – a free 'lite' version and the full licensed version. The differences are outlined further on, but this manual serves as a reference for both. Where certain features are disabled within the lite version, the software displays a notice to this effect.

CactuShop has progressed much since the original version and runs hundreds of web sites around the world. Clients range from small sole traders to large high street retailers. This documentation and the body of general ASP material online have helped numerous customers achieve the e-commerce solution that instant-CD products or Application Service Provider systems cannot offer.

Version 5 of the cart brings some exciting new features including multi-currency and multi-language support, optional UPS live shipping and recursive category structure.

Don't be intimidated by the size of this manual. It's a reference work designed to outline how to set up the product and explain how the features work. You don't need to memorize everything but a broad understanding of how the software works will help as you work to create your online store with CactuShop.

If you want to get the basic system up and running quickly before you start really tweaking it, check out the "I don't read manuals" fast set-up guide.

Happy shop building!

### **24/7 Online Help System**

The quickest way to get answers to your CactuShop problems is by searching the online knowledgebase. 80% of the problems reported by email or phone are already dealt with in the knowledgebase.

<http://www.cactushop.com/support/>

# Contents

<b>Introduction</b> .....	<b>1</b>
<b>Contents</b> .....	<b>2</b>
<b>1 Overview</b> .....	<b>14</b>
1.1 <i>What's Changed in v5</i> .....	14
1.1.1 ASP files moved to root of web .....	14
1.1.2 Images folders moved to new uploads folder .....	14
1.1.3 Multiple skins .....	15
1.1.4 Multiple back end logins .....	15
1.1.5 Naming conventions .....	15
1.1.6 Config setting changes .....	15
<b>2 CactuShop Lite – Free Trial Version</b> .....	<b>16</b>
2.1.1 Similarities/differences between the lite/full versions .....	16
2.1.2 License agreement for CactuShop Lite .....	16
2.1.3 Security issues .....	17
2.2 <i>CactuShop - Full Version</i> .....	17
<b>3 Setting Up</b> .....	<b>18</b>
3.1 <i>Base Platform Requirements</i> .....	18
3.1.1 Operating system .....	18
3.1.2 Database .....	18
3.1.3 VBscript version .....	18
3.1.4 Support for third party software.....	18
3.2 <i>Installing the Files</i> .....	19
3.2.1 Unzipping the software.....	19
3.2.2 Importing into the web server.....	19
3.2.3 Setting up CactuShop to use an ODBC datasource .....	20
3.2.4 Changes for security reasons .....	20
3.2.5 IP address security.....	20
3.3 <i>Setting Permissions</i> .....	20
3.3.1 Why do some folders need IUSR full control permissions? .....	21
3.3.2 How to set 'full control' permissions .....	21
3.3.3 Setting scripts permissions.....	21
3.4 <i>Run the Software!</i> .....	22
3.4.1 Dealing with errors.....	22
<b>4 Upgrading Your CactuShop</b> .....	<b>22</b>
4.1 <i>Why Upgrade?</i> .....	22
4.2 <i>Upgrade Policy</i> .....	22
4.3 <i>Obtaining the Latest Version</i> .....	22
4.4 <i>Upgrading Using the CactuShop Data Tool</i> .....	23
4.4.1 What it does .....	23

4.4.2	What the Data Tool can't do .....	23
4.4.3	My CactuShop is a version prior to v4.5 .....	23
4.4.4	Preparing to upgrade.....	23
<b>5</b>	<b>The CactuShop Front End .....</b>	<b>24</b>
5.1	<i>Introduction</i> .....	24
5.2	<i>Product view</i> .....	24
5.2.1	Product information.....	24
5.2.2	Version list.....	24
5.2.2.1	HTML list with hyperlinks.....	24
5.2.2.2	HTML list with buttons .....	24
5.2.2.3	HTML list with text boxes and buttons.....	24
5.2.2.4	HTML list with dropdowns and buttons.....	24
5.2.2.5	Dropdown list excluding price .....	24
5.2.2.6	Dropdown list including price.....	25
5.2.3	Auto sizing of 'large view' pop ups.....	25
5.2.4	'People who bought this product also bought' functionality .....	25
5.2.5	'Related items' functionality .....	25
5.2.6	'Try these categories' functionality .....	25
5.3	<i>Save/Recover Basket</i> .....	25
5.3.1	Basic operation .....	25
5.3.2	Used to store regular repeat orders.....	25
5.3.3	Used to store orders for another user to read .....	26
5.3.4	Issues affecting recovered baskets.....	26
5.4	<i>Wish List</i> .....	26
5.4.1	Creating and saving a wish list.....	26
5.4.2	Editing a list.....	26
5.4.3	Viewing a wish list.....	26
5.5	<i>SpeedOrder</i> .....	27
5.6	<i>Stock Tracking</i> .....	27
5.6.1	Basic operation .....	27
5.6.2	Trapping stock-related errors and issues.....	27
5.6.3	Setting up stock tracking for an item .....	27
5.7	<i>My Account</i> .....	27
5.7.1	Saved baskets.....	27
5.7.2	Order tracking and history .....	27
5.7.3	Downloads.....	28
5.7.4	Cookie-based security.....	28
5.8	<i>Affiliates Section</i> .....	28
5.8.1	Affiliate Application.....	28
5.8.2	Click-through tracking .....	28
5.8.3	Assigning a customer to an affiliate .....	28
5.8.4	Affiliate statistics .....	28
5.9	<i>The Checkout Process</i> .....	29
5.9.1	Checkout - Step 1 .....	29
5.9.2	Checkout - Step 2 .....	29
5.9.3	Checkout - Step 3 .....	29
5.9.4	Checkout - Step 4 .....	29
5.10	<i>Online Payment</i> .....	29
5.10.1	Payment through an online authorization gateway .....	29

5.10.2	Payment through CactuShop under SSL.....	30
5.10.3	Payment by PO .....	30
5.10.4	Email signup .....	30
<b>6</b>	<b>The Back-End.....</b>	<b>31</b>
6.1	<i>Logging In / Back End Password.....</i>	<i>31</i>
6.2	<i>Navigating the Back End .....</i>	<i>31</i>
6.3	<i>Listing Orders .....</i>	<i>31</i>
6.3.1	Recent orders .....	31
6.3.2	Awaiting invoice .....	31
6.3.3	Awaiting dispatch .....	31
6.3.4	Awaiting payment .....	31
6.3.5	List orders by date .....	32
6.4	<i>Stock Tracking.....</i>	<i>32</i>
6.4.1	Basic operation .....	32
6.4.2	Warnings page.....	32
6.4.3	Setting up stock tracking for an item .....	32
6.4.4	Why disable stock tracking for some items? .....	32
6.5	<i>Statistics .....</i>	<i>32</i>
6.5.1	Turning off statistical tracking .....	33
6.6	<i>Add/Modify/Delete a Category.....</i>	<i>33</i>
6.6.1	Show on site .....	33
6.6.2	Product display type .....	33
6.6.3	Sub-cat display type .....	33
6.6.4	Category name and description .....	34
6.6.5	Arbitrary order-by value and order-product dropdown .....	34
6.6.6	Category images .....	34
6.6.7	Deleting a category .....	34
6.7	<i>Add/Modify/Delete a Product.....</i>	<i>34</i>
6.7.1	Show on site .....	35
6.7.2	Featured menu .....	35
6.7.3	Version display type .....	35
6.7.4	Categories .....	35
6.7.5	Product name, strapline, description and specifications.....	35
6.7.6	Arbitrary order-by value and order-versions dropdown .....	35
6.7.7	Adding 'related products' .....	36
6.8	<i>Add/Modify/Clone/Delete a Version.....</i>	<i>36</i>
6.8.1	Cloning versions.....	36
6.8.2	Show on site .....	36
6.8.3	Code number .....	36
6.8.4	Product selection.....	37
6.8.5	Version name and description .....	37
6.8.6	Price.....	37
6.8.7	Weight .....	37
6.9	<i>Customer Database Admin .....</i>	<i>37</i>
6.9.1	Basic operation .....	37
6.9.2	Creating customer records from the back end .....	37
6.9.3	Customer record listing.....	37
6.9.4	Customer credit card details.....	38
6.9.5	Customer discount .....	38
6.9.6	Approved customers.....	38

6.10	<i>Orders</i> .....	38
6.10.1	Reviewing orders .....	38
6.10.2	Invoicing .....	38
6.10.3	Order text.....	39
6.10.4	Order progress and status .....	39
6.10.5	Turning off order progress system .....	39
6.10.6	Order summary.....	39
6.10.7	Deleting orders .....	39
6.11	<i>Affiliate Admin Section</i> .....	39
6.11.1	Basic operation .....	39
6.11.2	Technical implementation .....	39
6.11.3	Approving, setting up and modifying affiliates .....	40
6.11.4	Viewing statistics for affiliates .....	40
6.11.5	Payments to affiliates .....	40
6.12	<i>Coupons</i> .....	41
6.12.1	Generating coupons .....	41
6.12.2	Listing and finding coupons .....	41
6.13	<i>Email List Management</i> .....	41
6.13.1	Spamming .....	41
6.13.2	Send mail .....	42
6.13.3	Scope/scalability of mailing via CactuShop's scripts .....	42
6.13.4	List/find.....	42
6.13.5	New/edit address .....	42
6.13.6	Purge unauthorized addresses .....	42
6.14	<i>Searching for Products or Versions</i> .....	43
6.15	<i>Closing the Shop</i> .....	43
6.15.1	Close the shop .....	43
6.15.2	Hard-close the shop .....	43
6.16	<i>Currencies</i> .....	43
6.16.1	Basic set-up.....	43
6.16.2	Exchange rates .....	44
6.16.3	Has decimals fractions? .....	44
6.17	<i>Tax Rates</i> .....	44
6.17.1	European Union tax handling .....	44
6.17.2	US tax handling.....	45
6.18	<i>Destinations</i> .....	46
6.18.1	Charge tax by country .....	46
6.18.2	Shipping zone .....	46
6.18.3	ISO country code .....	46
6.18.4	Country name .....	46
6.18.5	The country ID .....	47
6.19	<i>Shipping Rates</i> .....	47
6.19.1	Adding extra shipping methods .....	47
6.19.2	Adding extra zones.....	47
6.20	<i>UPS Real Time Shipping</i> .....	48
6.20.1	About UPS Real Time Shipping.....	48
6.20.2	Config settings – UPS .....	49
6.20.2.1	UPS_AccessKey .....	49
6.20.2.2	UPS_Commercial .....	49
6.20.2.3	UPS_CurrencyConversion .....	49
6.20.2.4	UPS_MoreInfoURL .....	49

6.20.2.5	UPS_PackageType .....	49
6.20.2.6	UPS_Password .....	49
6.20.2.7	UPS_PickupType.....	49
6.20.2.8	UPS_ShopCountryCode .....	50
6.20.2.9	UPS_ShopPostalCode.....	50
6.20.2.10	UPS_ShopRegionCode .....	50
6.20.2.11	UPS_ShowDeliveryDays .....	50
6.20.2.12	UPS_UnitOfMeasurement .....	50
6.20.2.13	UPS_URL .....	50
6.20.2.14	UPS_UserID .....	50
6.21	<i>Languages</i> .....	50
6.21.1	Switch languages on or off .....	50
6.21.2	Back and front names .....	51
6.21.3	Skin location .....	51
6.21.4	Contact email addresses .....	51
6.22	<i>News Items</i> .....	51
6.23	<i>Mark Up Items</i> .....	51
6.24	<i>Static HTML Product Gateway Pages</i> .....	51
6.24.1	The debate over search engine indexing of database sites.....	51
6.24.2	Creating the static pages .....	52
6.24.3	Products no longer on the site .....	52
6.24.4	Folder permissions .....	52
<b>7</b>	<b>The Config System</b> .....	<b>53</b>
7.1	<i>Basic Operation</i> .....	53
7.2	<i>The License Number</i> .....	53
7.3	<i>Database Connection</i> .....	53
7.3.1	Hard-coded connection values .....	53
7.3.1.1	strDbType .....	53
7.3.1.2	strDataSourceName .....	54
7.3.1.3	numCursorType .....	54
7.3.2	Other database types .....	54
7.4	<i>The Config Table</i> .....	55
7.4.1	Overview .....	55
7.4.1.1	Basic operation .....	55
7.4.1.2	Changing config settings .....	55
7.4.2	Allowable Lengths .....	55
7.4.3	Config settings – Credit Card Details.....	56
7.4.3.1	SaveInDB .....	56
7.4.3.2	SaveInEmail .....	56
7.4.4	Config settings – Display .....	56
7.4.4.1	addtobasketdisplay .....	56
7.4.4.2	addtobasketdropdownmax.....	56
7.4.4.3	autoforwardtogateway .....	56
7.4.4.4	categorylevelsmenuback .....	56
7.4.4.5	categorylevelsmenufront .....	56
7.4.4.6	categorylevelsmenusearch.....	56
7.4.4.7	convertdescbreaks .....	56
7.4.4.8	defaultcountry .....	57
7.4.4.9	defaultproddisplay .....	57
7.4.4.10	defaultprodtypedisplay.....	57

7.4.4.11	defaultversiondisplay .....	57
7.4.4.12	newestproductslist .....	57
7.4.4.13	peoplewhobought .....	57
7.4.4.14	peoplewhoboughtjustorder .....	57
7.4.4.15	prodtypemenuasdropdown .....	57
7.4.4.16	prodtypepagesize .....	57
7.4.4.17	searchmaxrecs .....	57
7.4.4.18	searchpagesize .....	57
7.4.4.19	shortenedformatcols .....	57
7.4.4.20	shortenedformatrows .....	58
7.4.4.21	shortenedformatshowdesc .....	58
7.4.4.22	shortenedformatshowversions .....	58
7.4.4.23	showbasket .....	58
7.4.4.24	showmoredetaillink .....	58
7.4.4.25	showrrponversiondisplay .....	58
7.4.4.26	showweightonversiondisplay .....	58
7.4.4.27	specialsdisplaytype .....	58
7.4.4.28	speedrowcount .....	58
7.4.4.29	topsellingproductslist .....	58
7.4.4.30	topsellingproductsdays .....	58
7.4.4.31	truncatenewestproducts .....	59
7.4.4.32	truncateproductoncategory .....	59
7.4.4.33	truncateproductonspecials .....	59
7.4.4.34	truncatetopsellingproducts .....	59
7.4.4.35	trythesecategories .....	59
7.4.5	Config settings – Downloads .....	59
7.4.5.1	downloadlimitdays .....	59
7.4.5.2	hiddendownloads .....	59
7.4.5.3	instantdownload .....	59
7.4.6	Config settings – Email Settings .....	59
7.4.6.1	emailmethod .....	60
7.4.6.2	mailserver .....	60
7.4.6.3	newsletterfromemail .....	60
7.4.6.4	newsletterfromname .....	60
7.4.6.5	senderroremail .....	60
7.4.6.6	sendorderupdateemail .....	60
7.4.7	Config settings – Encryption .....	60
7.4.7.1	encrypt .....	60
7.4.7.2	encryptionkeyfile .....	60
7.4.8	Config settings – Ignore .....	60
7.4.8.1	lastupdated .....	60
7.4.9	Config settings – Important .....	61
7.4.9.1	cookie .....	61
7.4.9.2	shopstatus .....	61
7.4.9.3	webshopURL .....	61
7.4.9.4	weightunit .....	61
7.4.9.5	uploadsfolder .....	61
7.4.10	Config settings – Languages .....	61
7.4.10.1	defaultlanguageid .....	61
7.4.11	Config settings – Miscellaneous Settings .....	61
7.4.11.1	affiliatecommissionduedays .....	61
7.4.11.2	allowedimages .....	61
7.4.11.3	allowenquiries .....	61
7.4.11.4	basketmaxcapacity .....	62
7.4.11.5	cookiecheck .....	62
7.4.11.6	minimumcustomercode .....	62

7.4.11.7	tracking .....	62
7.4.11.8	trackingsales .....	62
7.4.11.9	usesecurearea .....	62
7.4.12	Config settings – Ordering .....	62
7.4.12.1	orderby_categories .....	62
7.4.12.2	orderby_products .....	62
7.4.12.3	orderby_versions .....	62
7.4.13	Config settings – Orders .....	62
7.4.13.1	checkinvoicedonpayment .....	62
7.4.13.2	checkreceivedonpayment .....	62
7.4.13.3	invoiceduedays .....	63
7.4.13.4	minorderval .....	63
7.4.13.5	orderhandlingcharge .....	63
7.4.13.6	ordertracking .....	63
7.4.13.7	orderhandlingcharge .....	63
7.4.13.8	ordertransactions .....	63
7.4.13.9	purgeunfinisheddays .....	63
7.4.13.10	termsandconditionscheck .....	63
7.4.14	Config settings – Payment .....	63
7.4.14.1	luhncheckON .....	63
7.4.14.2	paymentauthorizeallpo .....	63
7.4.14.3	paymentprocesscurrency .....	64
7.4.14.4	paymentsystem .....	64
7.4.15	Config settings – Payment Gateways .....	64
7.4.16	Shipping .....	65
7.4.16.1	calcshipcostbyweight .....	65
7.4.16.2	skiponeshippingmethod .....	65
7.4.16.3	shippingsystem .....	65
7.4.16.4	shiptaxband .....	65
7.4.17	Config settings – Tax .....	65
7.4.17.1	euvatcountry .....	65
7.4.17.2	euvatnumbercheck .....	65
7.4.17.3	pricesinctax .....	65
7.4.17.4	showtaxdisplay .....	65
7.4.17.5	taxshippingbilling .....	66
7.4.18	Config settings – UPS .....	66
<b>8</b>	<b>Online Payment Methods – Set Up .....</b>	<b>67</b>
8.1	<i>Setting Up WorldPay Select Junior .....</i>	<i>67</i>
8.1.1	About WorldPay .....	67
8.1.2	Config settings – WorldPay .....	68
8.1.2.1	WP_AuthCode .....	68
8.1.2.2	WP_CallbackPassword .....	68
8.1.2.3	WP_EmailCustomer .....	68
8.1.2.4	WP_EmailMerchant .....	68
8.1.2.5	WP_InstallID .....	68
8.1.2.6	WP_ProcessCurrency .....	68
8.1.2.7	WP_ReturnURL .....	68
8.1.2.8	WP_testmode .....	68
8.1.2.9	WP_WorldPayURL .....	69
8.1.3	Set up required at WorldPay side .....	69
8.1.3.1	Callback password .....	69
8.1.3.2	Use callback enabled .....	69
8.1.3.3	Use callback response .....	69
8.1.3.4	The callback URL .....	69

8.1.3.5	Further customization .....	69
8.1.4	Activation .....	69
8.2	<i>Setting Up 2CheckOut</i> .....	69
8.2.1	About 2CheckOut .....	69
8.2.2	Config settings – 2CheckOut .....	70
8.2.2.1	AUTHNET_AuthorizeNetURL .....	70
8.2.2.2	AUTHNET_CallbackURL .....	70
8.2.2.3	AUTHNET_EmailCustomer .....	70
8.2.2.4	AUTHNET_EmailMerchant.....	70
8.2.2.5	AUTHNET_LoginID.....	70
8.2.2.6	AUTHNET_ProcessCurrency.....	70
8.2.2.7	AUTHNET_ReturnURL .....	70
8.2.2.8	AUTHNET_TestMode.....	70
8.2.2.9	AUTHNET_TxnKey .....	70
8.2.3	Set up required at 2CheckOut side .....	71
8.2.3.1	Call back set up .....	71
8.2.3.2	Return URL.....	71
8.3	<i>Setting Up Authorize.Net</i> .....	71
8.3.1	About Authorize.Net .....	71
8.3.2	Setting up Authorize.Net Advanced Integration Method (AIM) .....	71
8.3.2.1	AIM_AuthorizeNetURL .....	72
8.3.2.2	AIM_EmailCustomer .....	72
8.3.2.3	AIM_EmailMerchant .....	72
8.3.2.4	AIM_LoginID .....	72
8.3.2.5	AIM_ProcessCurrency.....	72
8.3.2.6	AIM_ResponseDelimiter .....	72
8.3.2.7	AIM_TestMode .....	72
8.3.2.8	AIM_TxnKey.....	72
8.3.2.9	AIM_Type.....	72
8.3.2.10	AIM_XMLComponent .....	72
8.3.3	Setting up Authorize.Net Simple Integration Method (SIM) .....	73
8.3.4	Config Settings - Authorize.Net SIM.....	73
8.3.4.1	AUTHNET_AuthorizeNetURL .....	73
8.3.4.2	AUTHNET_CallbackURL .....	73
8.3.4.3	AUTHNET_EmailCustomer .....	73
8.3.4.4	AUTHNET_EmailMerchant.....	73
8.3.4.5	AUTHNET_LoginID.....	73
8.3.4.6	AUTHNET_ProcessCurrency.....	73
8.3.4.7	AUTHNET_ReturnURL .....	73
8.3.4.8	AUTHNET_TestMode.....	74
8.3.4.9	AUTHNET_TxnKey .....	74
8.3.5	Set up required at Authorize.Net side .....	74
8.3.5.1	Callback setup .....	74
8.3.5.2	Test mode.....	74
8.4	<i>Setting Up PayPal</i> .....	74
8.4.1	About PayPal .....	74
8.4.2	Config settings - PayPal.....	74
8.4.2.1	PP_Business .....	74
8.4.2.2	PP_Cmd .....	75
8.4.2.3	PP_EmailCustomer .....	75
8.4.2.4	PP_EmailMerchant .....	75
8.4.2.5	PP_NotifyURL .....	75
8.4.2.6	PP_ProcessCurrency.....	75
8.4.2.7	PP_TestMode.....	75

8.4.2.8	PP_ValidateCommand .....	75
8.4.2.9	PP_XMLComponent .....	76
8.4.3	Set up required at PayPal side.....	76
8.5	<i>Setting Up Protx VSP Form</i> .....	76
8.5.1	About Protx .....	76
8.5.2	Config settings – Protx .....	76
8.5.2.1	PX_CallbackURL .....	76
8.5.2.2	PX_EmailCustomer .....	76
8.5.2.3	PX_EmailMerchant .....	76
8.5.2.4	PX_OrderDescription .....	76
8.5.2.5	PX_Password .....	76
8.5.2.6	PX_ProcessCurrency.....	77
8.5.2.7	PX_SubmitURL.....	77
8.5.2.8	PX_VendorEmail .....	77
8.5.2.9	PX_VendorID.....	77
8.5.3	Protx back end.....	77
8.6	<i>Setting Up SecPay</i> .....	77
8.6.1	About SecPay.....	77
8.6.1.1	SECPAY_CallbackPassword .....	77
8.6.1.2	SECPAY_CallbackURL .....	77
8.6.1.3	SECPAY_EmailCustomer .....	77
8.6.1.4	SECPAY_EmailMerchant .....	77
8.6.1.5	SECPAY_Merchant .....	77
8.6.1.6	SECPAY_Options.....	78
8.6.1.7	SECPAY_ProcessCurrency.....	78
8.6.1.8	SECPAY_ReturnURL .....	78
8.6.1.9	SECPAY_SubmitURL.....	78
8.6.1.10	SECPAY_Template .....	78
8.7	<i>Setting Up Secure Trading</i> .....	78
8.7.1	About Secure Trading.....	78
8.7.1.1	ST_CallBackID .....	78
8.7.1.2	ST_EmailCustomer .....	78
8.7.1.3	ST_EmailMerchant .....	79
8.7.1.4	ST_Password.....	79
8.7.1.5	ST_ProcessCurrency.....	79
8.7.1.6	ST_RequiredFields .....	79
8.7.1.7	ST_SiteReference.....	79
8.7.1.8	ST_URL.....	79
8.7.2	Secure Trading back end .....	79
8.7.2.1	Configuration files.....	79
8.7.2.2	Display files .....	79
8.7.2.3	Test mode.....	80
8.8	<i>VeriSign</i> .....	80
8.8.1	About Verisign .....	80
8.8.2	Setting Up VeriSign Payflow Link.....	80
8.8.2.1	VS_EmailCustomer .....	81
8.8.2.2	VS_EmailMerchant .....	81
8.8.2.3	VS_Login.....	81
8.8.2.4	VS_Partner .....	81
8.8.2.5	VS_Password.....	81
8.8.2.6	VS_ProcessCurrency.....	81
8.8.2.7	VS_Type .....	81
8.8.2.8	VS_VeriSignURL.....	81
8.8.2.9	VS_Password.....	81

8.8.3	Set up required at VeriSign side .....	82
8.8.3.1	Return URL Method.....	82
8.8.3.2	Return URL.....	82
8.8.3.3	Silent Post URL .....	82
8.8.3.4	Transaction Process Mode .....	82
8.8.4	Setting Up VeriSign Payflow Pro .....	82
8.8.4.1	PFP_EmailCustomer .....	82
8.8.4.2	PFP_EmailMerchant.....	82
8.8.4.3	PFP_HostAddress.....	82
8.8.4.4	PFP_HostPort .....	83
8.8.4.5	PFP_Partner .....	83
8.8.4.6	PFP_ProcessCurrency.....	83
8.8.4.7	PFP_ProxyAddress .....	83
8.8.4.8	PFP_ProxyLogin .....	83
8.8.4.9	PFP_ProxyPassword .....	83
8.8.4.10	PFP_ProxyPort.....	83
8.8.4.11	PFP_Pwd.....	83
8.8.4.12	PFP_Tender .....	83
8.8.4.13	PFP_Timeout.....	83
8.8.4.14	PFP_Type.....	83
8.8.4.15	PFP_User .....	83
8.8.4.16	PFP_Vendor .....	83
8.9	<i>Email Notification - Unencrypted</i> .....	84
8.10	<i>CactuShop Encrypted Mail ("CEM")</i> .....	84
8.10.1	Requirements .....	84
8.10.2	Installation .....	84
8.10.3	Config Settings.....	85
8.10.3.1	encrypt .....	85
8.10.3.2	encryptionkeyfile .....	85
8.11	<i>CactusPay</i> .....	85
8.11.1	CactusPay - CactuShop Configuration .....	85
8.11.1.1	CP_CactusPayURL .....	85
8.11.1.2	CP_CallbackPassword .....	85
8.11.1.3	CP_ProcessCurrency .....	85
8.11.2	CactusPay – Secure Area Configuration .....	85
8.11.2.1	emailmethod.....	86
8.11.2.2	sendorderstoaddress .....	86
8.11.2.3	emailsubjectline .....	86
8.11.2.4	mailserver.....	86
8.11.2.5	callback_password.....	86
8.11.2.6	callback1.....	86
8.11.2.7	callback2.....	86
8.11.2.8	EncryptionKey .....	86
8.11.2.9	LocalKeyFileName.....	86
8.11.3	Generating a key.....	86
8.11.4	Setting up the decryption software.....	87
8.12	<i>Secure Back End Review – Requires SSL</i> .....	87
8.13	<i>Setting Up SSL</i> .....	88
8.13.1	Obtaining a secure certificate.....	88
8.13.2	SSL issues .....	88
8.13.3	SSL access to the database.....	89
8.14	<i>Other Payment Systems</i> .....	89

<b>9</b>	<b>Language Strings System</b>	<b>90</b>
9.1	<i>Background</i>	90
9.2	<i>Multi-Language Capability</i>	90
9.3	<i>The Language String Editor</i>	90
9.4	<i>The Language Strings Table</i>	91
9.4.1	LS_ID	91
9.4.2	LS_FrontBack	91
9.4.3	LS_Name	91
9.4.4	LS_Value	91
9.4.5	LS_Description	91
9.4.6	LS_VersionAdded	91
9.5	<i>The Language Strings Functionality</i>	91
9.5.1	PageStrings	91
9.5.2	LoadStrings	92
9.5.3	WriteString and GetString functions	92
9.6	<i>Adding New Languages</i>	92
9.6.1	Overview	92
9.6.2	Add a New Skin	92
9.6.3	Activating a Second Language	92
9.6.4	Language Strings	93
9.6.5	Other Software Elements to Translate	93
9.6.5.1	Shipping Rates	93
9.6.5.2	Destinations	93
9.6.6	Selecting a Default Language	93
9.6.7	Product Data	93
9.6.8	News Stories	93
9.6.9	Mailing List	93
<b>10</b>	<b>Customizing CactuShop</b>	<b>94</b>
10.1	<i>Skins</i>	94
10.1.1	How does CactuShop create pages based on the template?	94
10.1.2	What elements make a skin?	94
10.1.3	What essential features must a 'skin' have?	94
10.1.4	The style sheet reference	95
10.1.5	How is the template used to format my pages (tech stuff)	96
10.2	<i>Customizing the Functionality of CactuShop</i>	96
<b>11</b>	<b>CactuShop Data Tool</b>	<b>97</b>
11.1	<i>Installation</i>	97
11.2	<i>Before You Start</i>	97
11.3	<i>Using the Software</i>	97
11.3.1	Using the software to upgrade your CactuShop	98
11.3.2	Upsizing (or downsizing) your database	98
11.3.3	Backing up your data	98
11.3.4	Uploading your data to a live site	99
11.3.5	Importing from spreadsheet or text file	99
11.3.5.1	Limitations of importing from Excel / text file	99
11.3.5.2	The format of the data	100
11.3.5.3	Images	102

---

11.3.5.4	Text files and the schema.INI file .....	102
11.3.5.5	Excel files.....	102
11.3.5.6	Running your import.....	102
11.3.5.7	Running your import with images .....	103
11.3.6	Importing your catalogue from another shopping cart .....	103
11.3.6.1	Limitations of importing from another software package .	103
11.3.6.2	Consideration of CactuShop versions in comparison to other products	104
11.3.6.3	Running the import.....	104
11.3.6.4	Importing from VPASP .....	105
11.3.6.5	Importing from Actinic .....	105
11.3.6.6	Importing from Comersus .....	106
11.3.7	Executing an SQL file .....	106
11.4	<i>Data Tool Disclaimer</i> .....	107
<b>12</b>	<b>Support</b> .....	<b>108</b>
12.1	<i>Scope of Tech Support</i> .....	108
12.2	<i>How to Request Support</i> .....	108
12.3	<i>FAQ</i> .....	108
12.3.1	What does the error 'must use updateable query' mean?.....	108
12.3.2	I get a 500 error. What does this mean? .....	109
12.3.3	What is the back end login and password? .....	109
12.3.4	Will the software run on Linux?.....	109
12.3.5	Will the software run a Chinese, Japanese or Korean web site?	109
12.3.6	What is your upgrade policy?.....	109
12.3.7	Can I use a visual editor? .....	109
12.3.8	How does the licensing work? .....	110
12.3.9	How much is it in US Dollars? .....	110

# 1 Overview

## 1.1 What's Changed in v5

There have been a number of changes to the structure of v5 and some of the underlying code when compared to prior versions (v4.x and earlier). Here is a brief outline of the changes and the reasons behind them for those who are familiar with earlier versions of CactuShop.

### 1.1.1 ASP files moved to root of web

In previous versions of CactuShop, the ASP files that run the site were in three folders – *ASP*, *cactushopback* and *includes*. Files in the *includes* folder were referenced from files in both the other two folders using a relative path such as `../includes/config.asp`. However, permissions settings can prevent use of the `../` part, generating a 'parent paths' error. It appears that preventing use of parent paths is IIS 6's (Windows Server 2003) default setting hence CactuShop users would need to get these settings reset by their hosting company in order to use CactuShop.

Some fixes for the above problem negate the need for permissions changes. One is to use virtual includes instead of relative ones. However, this causes problems if you wish to run CactuShop in a folder on your site rather than at the root of the web.

Instead, we have moved all the files previously in the *ASP* and *cactushopback* folders to the root of the web, keeping the include files in the same place as before. This resolves the issue because no `../` is required in the path to includes or other folders. To help distinguish back end files from front end ones, they are prefixed with an underscore ("`_`"), which should mean they list alphabetically together.

This change should also improve store speed and search engine visibility, since no redirect is required from the root of the web any longer. The home page of CactuShop now sits on the root of the web.

Some users have complained that this makes it harder to add additional security to the back end since it is no longer in its own folder. However, we have provided beefed up security by way of an optional IP address restriction (set in the *config.asp* file).

### 1.1.2 Images folders moved to new uploads folder

Previously in CactuShop, all uploaded product-related images were stored in folders such as *images\_products*, *images\_versions*, *images\_products\_large*, etc. However, some hosting companies only set one specific folder on the web with the correct permissions to allow image upload to work. In order to handle this without requiring the user to extensively recode pages, we have moved the images folders within a new folder. By default this is called *uploads*. If you need to use a similar folder with another name, you can change the name of the *uploads* folder and change the `uploadsfolder` config setting (see 7.4.9.5).

### 1.1.3 Multiple skins

To support the multi-language capabilities of CactuShop, multiple skins are required (one for each language). The names of these are no longer set in the config settings. Instead, the path to each skin is set on the *\_modify\_langs.asp* page of the back end.

### 1.1.4 Multiple back end logins

On CactuShop v4, the back end username and password were stored as config settings. However, v5 supports multiple back end users, so these are now set up from the *\_modify\_login.asp* page and stored in the *tblCactuShopLogins* table.

### 1.1.5 Naming conventions

In order to make CactuShop easier to work with, variables throughout have been renamed according to conventions that are more consistent. Each is prefixed with a three-letter code indicating the type of variable it is. These are as follows:

Prefix	Meaning
bln	boolean (true/false)
num	numeric (integer or real)
str	string
obj	object
dat	date

Values that have been directly drawn from particular database values retain that name. For example a product ID number will normally be used within a page as *P\_ID* rather than *numProductID* because the database field from which it was pulled is called *P\_ID*.

Config settings have not been renamed according to this convention as it should be clear from the context.

### 1.1.6 Config setting changes

In addition to a host of new config settings, some things that were previously config settings have been moved elsewhere. For example, *ownertoemail* and *ownerfromemail* are now language specific and hence updated from the languages back end section (see 6.21.4). The location of the skin is also set individually for each language here rather than in the config settings (as was the case for prior CactuShop versions).

## 2 CactuShop Lite – Free Trial Version

Cactusoft produces two versions of CactuShop. The lite version is a cut down version with much of the security code removed, no license for live web use, and some of the features disabled. CactuShop Lite is available as a download from the CactuShop web site for the purposes of testing the basic code on your server, populating a database with products, creating the 'skin' template design and satisfying yourself that you have the technical capability to set up and create your web store.

### 2.1.1 Similarities/differences between the lite/full versions

CactuShop Lite shares the following with the full version:

- a. database structure
- b. skin format
- c. basic back end code for adding and updating products
- d. navigation
- e. basic basket functionality
- f. coding style
- g. platform and set up requirements

However, it does **not** include the following parts:

- a. license for live web use
- b. payment scripts
- c. UPS live shipping
- d. wish list
- e. affiliate programme
- f. cookie-based basket
- g. mailing list
- h. search engine
- i. my account section
- j. saved basket functionality
- k. coupons
- l. customer percentage discounts
- m. speedordering
- n. graphical stats on sales and product page views
- o. search engine static page generator
- p. free tech support

### 2.1.2 License agreement for CactuShop Lite

Please read the license agreement that comes with CactuShop Lite carefully. In particular, you may NOT do the following with CactuShop Lite:

- a. run any of the Cactusoft copyrighted code on a live web site other than as a temporary trial development version
- b. remove copyright messages or credits from Cactusoft's code
- c. remove copyright messages or credits from the image upload and sizing scripts or any other freeware scripts; these are written by other authors – we have left their code and comments intact to give them full credit for their work, please respect their right to be credited for their efforts

### **2.1.3 Security issues**

CactuShop Lite is a trial version and as such is NOT secure. The back end has no password protection and much of the code that secures various sections of the front end is not present in the lite version. We suggest that the best way of running CactuShop Lite for test purposes is to install IIS on your own server or Windows PC (if supported) and run the lite version on that. If you do need to run CactuShop Lite on a live web server, we strongly suggest running it in a folder where the casual Google user cannot gain access to it. You should remove CactuShop Lite from your live server when you have finished with it or when you have upgraded to the full version.

## **2.2 CactuShop - Full Version**

The full licensed version of CactuShop is available from the CactuShop web site for £200 (British Pounds). The full version uses exactly the same database, skin template and folder structure as the lite version. Just delete the lite version ASP scripts in the root of the web and the includes folder, and then copy over the full version scripts in order to upgrade your lite store to a full store.

## 3 Setting Up

### 3.1 Base Platform Requirements

#### 3.1.1 Operating system

CactuShop is an Active Server Pages application and as such needs to run on a web server that incorporates an ASP scripting engine. Although Linux/Unix servers can be made to run ASP using third party software, CactuShop is neither developed nor tested on these alternative platforms and to the best of our knowledge **it will not run on Linux/Unix**. We only endorse and support running CactuShop on Microsoft's IIS web server. CactuShop will run fine on shared web space but check the platform and permission requirements with your host first if in any doubt. Use the free lite version as a test, checking that you can successfully add and delete products with images.

#### 3.1.2 Database

CactuShop ships with an Access 2000 database file. The ASP code is written to be switchable between MS Access, Microsoft SQL Server and MySQL syntax (similar but with some small differences). Access 97, MS SQL Server and MySQL versions of database are available as free downloads from our web site. CactuShop is only tested on the Windows version of MySQL although we know of users who successfully have CactuShop running with a MySQL database on Linux. However, we only support MySQL on the Windows platform.

#### 3.1.3 VBscript version

The CactuShop code is VBscript – the ASP default language. In order to update the VBscript engine, install the latest version of Internet Explorer on the server. For the image upload capability, VBscript 5.0+ is required and you will get an error trying to update products without it. This will only affect the cart if it is run on NT – Windows 2000 includes VBscript 5.0 as standard.

#### 3.1.4 Support for third party software

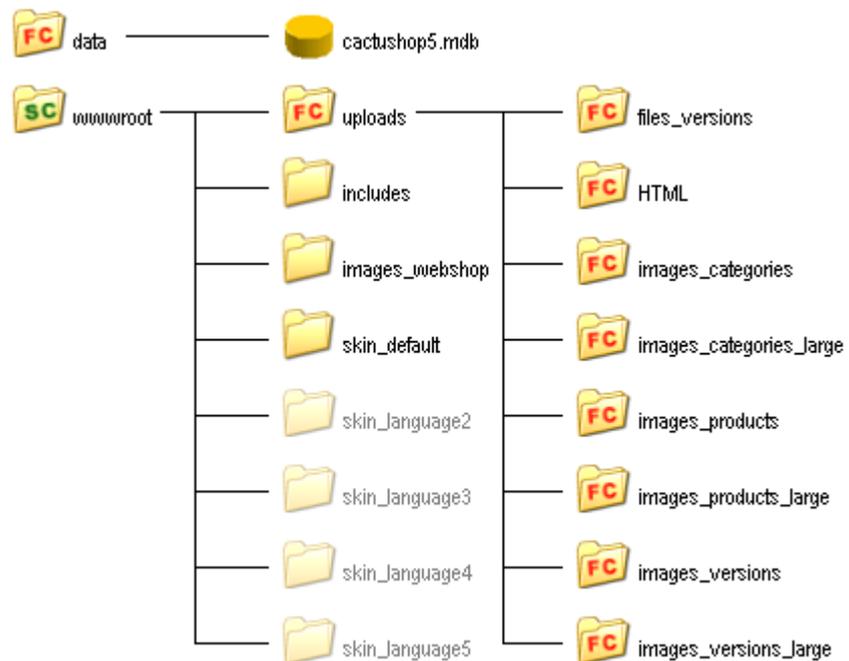
The system software set up referred to above is a prerequisite for running an ASP site. We cannot support set up and installation of Microsoft or other third party software – please consult the relevant web sites and use their sample applications to test that your server, scripting engine and RDMS (database) are running properly. Our tech support does not cover installing Microsoft or other third party software or getting ASP support to run on your site/development server.

## 3.2 Installing the Files

### 3.2.1 Unzipping the software

The product is shipped electronically as a zip file. After unzipping, you should end up with a CactuShop folder containing several other folders and files. For security, we suggest that the Access database (which by default is located in the *database* folder) should be moved to a folder that is not on the web root as shown below in figure 1 (remember to adjust the *strPathToDatabase* setting in the *includes/config.asp* file to reflect this). If using MS SQL or MySQL, the .mdb file is not required.

Figure 1. CactuShop folder layout and permissions



Key:

 run scripts

 IUSR full control permissions

 only required for multiple language sites

Note: each skin folder normally contains a folder called *templateimages* too – this has been omitted for simplicity

### 3.2.2 Importing into the web server

You can just drag and drop the whole CactuShop folder into your *wwwroot* folder, or FTP all the files within it to your web site – just as you would set up any new web.

If you set up the web by importing it into MS FrontPage, you may find that many of the back end files are not visible after importation. This is because

FrontPage sees files beginning with an underscore as 'hidden'. To rectify this, go to *tools > web settings > advanced* and ensure that 'show documents in hidden directories' is checked.

### 3.2.3 Setting up CactuShop to use an ODBC datasource

Some users prefer to connect to the database via ODBC. If so, set up the data source and change the *strDataSourceName* setting in the *includes/config.asp* file to the name of your data source, with quotes around it. For example:

```
strDataSourceName = "mydatasource"
```

### 3.2.4 Changes for security reasons

The database and its folder should both have their names changed at this point. You will need to adjust the database connection in the *config.asp* to reflect these name changes. If you do not change these default values, malicious users will easily be able to download your database and any private customer data it may hold. If your hosting provider has a 'private' or 'data' folder specifically for Access databases, we'd strongly advise using this (again, see figure 1 above). Such a folder has no read access permissions, so documents within it can't be accessed by HTTP.

Furthermore, the back end password of the main admin user account should be changed from the default ('cactushop') too.

CactuShop has code to write warnings to the screen if the default settings are not changed, so it should be difficult to inadvertently leave the default settings and put a web site live.

### 3.2.5 IP address security

For increased security of the CactuShop back end, you can limit access by IP address from within the *config.asp* file when you set up your store.

```
CONST BACKEND_IP_LOCK = "xxx.xxx.xxx.xxx"
```

You can also allow slightly broader access if you have consecutive IP addresses, for example:

```
CONST BACKEND_IP_LOCK = "xxx.xxx.xxx"
```

This feature is best for users with a fixed IP address. It will prevent anyone else from accessing the back end of the store even if they learn the username and password of a valid login account.

Do not use this feature if you have a dynamic IP address and are unaware of the range.

## 3.3 Setting Permissions

The folders and Access database file need certain security permissions set in order to function properly. Figure 1 above shows the settings required on the various folders. **Note that these permissions cannot be changed via FTP.** If your web host doesn't set these as standard, you will need either

physical access to the server, access via PC-anywhere (or some similar system) or an obliging techie at the web hosting company. The database folder should be renamed or made 'private' as mentioned in 3.2.4.

### 3.3.1 Why do some folders need IUSR full control permissions?

Folders where CactuShop needs to add, modify or delete files need IUSR full control permissions. The IUSR account is the "Internet Guest Account" – the user account that all web users use when on the web site. Without these permissions, the following problems will occur:

- uploading images or deleting images (when updating products from the back end) will give a *"permission denied"* error
- any page that writes to an Access database will give a *"must use updateable query error"*
- attempting to create gateway pages for search engines will give a *"permission denied"* error

### 3.3.2 How to set 'full control' permissions

On Windows 95 and 98, you do not need to set permissions.

On Windows NT, 2000, XP Pro and Server 2003, you need to set security permissions to allow the Internet Guest Account full rights to modify the database and create/modify files. Note (again!) that CactuShop requires IIS, and as such will not run on XP Home or Windows ME.

1. Using Windows Explorer, navigate to the folder whose permissions you wish to set. Right click it and click 'properties', then the 'security' tab\*.
2. If the Internet Guest Account (also shown as *IUSR\_MachineName* or something similar) is not present, you will need to click 'add' and then select and add this user.
3. Next, select this *IUSR* user account and tick the top box, 'full control'. Click OK and you're done

\*If you are using XP Pro and cannot see the security tab, open the 'tools' menu in any folder and click 'folder options'. Select the 'view' tab and then uncheck the bottom box, 'use simple file sharing'. Then click 'ok'. You should now see the 'security' tab in future.

### 3.3.3 Setting scripts permissions

Some folders within the web may not have 'scripts' permissions set – they may only allow reading of files. This is fine for HTML, but ASP pages are dynamic and contain active code that needs to run. To ensure that the scripts have permission to run, navigate to CactuShop in the IIS manager or FrontPage, right click the web and tick the 'allow scripts to be run' box. You cannot do this from Windows Explorer, it must be done within FrontPage or IIS.

### 3.4 Run the Software!

Type in the address of the web that you've just created into a browser and you should get the home page of the shopping cart. If you get any errors, please check the path you typed (careful!) and the points above. If you installed IIS or PWS on your own machine, the address of the server will normally be "http://localhost/", plus whatever your web is called. You could also try "http://127.0.0.1/".

#### 3.4.1 Dealing with errors

If you get any errors at any stage, please refer to the FAQ in section 12.3 and search our knowledgebase on the CactuShop web site before contacting us. Most common problems are dealt with there already.

## 4 Upgrading Your CactuShop

### 4.1 Why Upgrade?

From time to time, we bring out newer versions of the CactuShop software that include new features or redesigned pages that improve on previous offerings. There is no requirement for any customer to upgrade their existing site if they don't wish to. If the site is running fine and doing the intended job, us bringing out a new version will not change this. However, for those who do wish to take advantage of the newer features, upgrading is worth the time and effort.

### 4.2 Upgrade Policy

CactuShop upgrades are free for six months after the purchase of a CactuShop license. Beyond this you can purchase additional upgrade coverage for one year from our web site order form for £50 (GBP). Note that this is not an additional license, purely an admin fee to cover the cost of supplying upgrade code. If you have multiple stores, you may upgrade as many of them as you wish with the newest code providing they are already properly licensed CactuShops.

### 4.3 Obtaining the Latest Version

All upgrade requests are handled through our web site. Please go to:

<http://www.cactushop.com/main/upgrading.asp>

You will need your license number and the original email address you placed the order from. If this has changed, please use the contact form on our site to let us know.

Assuming your license is less than six months old or your upgrade coverage is current, the latest version will be emailed to you automatically. Please note that the zip file is over 1 MB in size, so you will need to ensure that you have no filters that will prevent such an attachment from reaching you.

## 4.4 Upgrading Using the CactuShop Data Tool

CactuShop is the only ASP shopping cart vendor that provides a free tool to make upgrading your store to the latest version quick and (relatively) simple. The Data Tool is built in Visual Basic and once installed helps you transition data from an older CactuShop to the newest format.

For more advanced information regarding the Data Tool, please see section 11.

### 4.4.1 What it does

The Data Tool lets you upsize unmodified CactuShops from v4.5, v4.6, v4.7 or v5.0 to v5.0 or v5.1. It can also transition data from MS Access to MS SQL and vice versa. Load it up and follow the on-screen instructions.

### 4.4.2 What the Data Tool can't do

If your CactuShop database has been customized and has had extra fields or tables added to it, the Data Tool will not be able to transition these customizations to the new database. These changes may also prevent the Data Tool from operating properly. Remember also that any script modifications you may have cannot be upgraded to newer versions of CactuShop automatically. Most such modifications and customizations will be lost during the upgrade process and need to be redone manually.

### 4.4.3 My CactuShop is a version prior to v4.5

If your CactuShop is older than v4.5, upgrading requires some manual steps. Please contact us for more details.

### 4.4.4 Preparing to upgrade

Before upgrading or doing any work on your database, always remember to take a back up and to keep this in a safe place.

## 5 The CactuShop Front End

### 5.1 Introduction

The front end is the part that visitors to the site see. All the pages are dynamically generated and allow the user to navigate around the site, view products, select items to add to the shopping basket and then finally, checkout and pay for the goods.

### 5.2 Product view

This is the key part of the front end – the page that displays products and their versions.

#### 5.2.1 Product information

The CactuShop code pulls out product information and checks to see whether an image exists for the product. If one is present, this is displayed. Another check is made to see if a large image has been uploaded too. If yes, the 'view' link is displayed.

#### 5.2.2 Version list

'Versions' is a critical design feature in CactuShop. Products can have any number of versions, and each version of a product can have a different spec, price and availability. Versions may typically be different sizes, colours, etc.

Versions can be listed in several ways, for more details see 6.7.3.

##### 5.2.2.1 HTML list with hyperlinks

Set `addtobasketdisplay` config setting (7.4.4.1) to 'l' (lower case 'L') and version display type at product level to 'rows'.

##### 5.2.2.2 HTML list with buttons

Set `addtobasketdisplay` config setting (7.4.4.1) to 'b' and version display type at product level to 'rows'.

##### 5.2.2.3 HTML list with text boxes and buttons

Set `addtobasketdisplay` config setting (7.4.4.1) to 't' and version display type at product level to 'rows'.

##### 5.2.2.4 HTML list with dropdowns and buttons

Set `addtobasketdisplay` config setting (7.4.4.1) to 'd' and version display type at product level to 'rows'.

##### 5.2.2.5 Dropdown list excluding price

Set version display type at product level to 'options (list just names)'. Note that in this dropdown list format, version images and descriptions are not visible. Also, if you select just names (no prices) but the prices of versions are different, the display will be incorrect – only one price will be shown. You

should therefore only select this option if all versions of the product are the same price.

#### **5.2.2.6 Dropdown list including price**

Set version display type at product level to 'options (list names and prices)'. Note that in this dropdown list format, version images and descriptions are not visible.

#### **5.2.3 Auto sizing of 'large view' pop ups**

The large view pop ups are automatically sized to fit the size of the images that occupy them.

#### **5.2.4 'People who bought this product also bought' functionality**

This displays links to other products that were also purchased by people who bought the current product. For information on the changing the number of links displayed here, see the config setting `peoplewhobought`, 7.4.4.13.

#### **5.2.5 'Related items' functionality**

CactuShop includes facilities in the back end for setting up links between products. In this way, a page containing electric torches could have links to batteries or replacement bulbs, for example. These links are one way, so in the above example, the battery page or bulbs page won't have links to electric torches unless you specifically set these up.

For information on setting up related items, see section 6.7.7.

#### **5.2.6 'Try these categories' functionality**

If a product is in multiple categories, this will provide links to the other categories that it is included in, which may therefore contain similar items.

### **5.3 Save/Recover Basket**

#### **5.3.1 Basic operation**

This feature lets users save their basket contents to the database so they can recover them on a future visit. To save a basket, a user needs only enter an email address and name/password for the saved record. The same information is entered in the appropriate place to recover the saved basket, instantly reinstating the products to the current basket.

#### **5.3.2 Used to store regular repeat orders**

Users can save any number of baskets, so this system is also perfect for storing several regular orders that a user may make from your web shop.

### 5.3.3 Used to store orders for another user to read

Since the basket can be recovered by any user with the email address and name/password, a CactuShop saved basket is perfect if you wish to select items and get someone else on a different machine to recover the basket and pay for the items.

### 5.3.4 Issues affecting recovered baskets

If a basket has been stored for some time, it may be that the products in the basket are no longer available, or the specification/price may have changed. In the first instance, items no longer available or items out of stock will be removed from the cart automatically. However, the user should always check recovered items through to ensure the specification and price is acceptable. CactuShop provides a link to the basket and a message explaining this so that customers will be encouraged to check through recovered basket details.

## 5.4 Wish List

The wish list lets individuals prepare a list of items on the site that they can then make available to friends and relatives. This is perfect for wedding lists, children's Christmas presents and other such situations.

### 5.4.1 Creating and saving a wish list

In order to create a wish list, simply load up the basket with all the items required and then click the 'wish list' link on the main menu. In the second section on the page, there is a link to create a new wish list. The user specifies their email address, first and last names, the private password they wish to use for editing the list and the public password they will circulate to their friends and relatives. In order to personalize the experience, they may also specify a greeting that those viewing the list will see above it.

### 5.4.2 Editing a list

Editing a list is just as easy. Click the wish list link and enter the email address and private password in the boxes on the second section of the page. You can first recover the list (which will fill your basket with all the items in the wish list) and then make any additions/changes before returning to save the list. Remember that items that have already been purchased will be shown on the list with a note to that effect.

At the point the wish list is saved, a confirmation email with the details is sent for future reference.

### 5.4.3 Viewing a wish list

All friends and relatives need to view a list is the email address and public password. They simply enter these in the top section of the wish list page and click the 'recover' button. This displays the items on the wish list so that they can click through and view these. From here, they can add items to their basket and buy them just as with normal items. At the point of purchase, these purchased items will be removed from the wish list.

A user stays logged into the wish list so simply clicking on the 'wish list' link on the main menu will bring up the list. Users can log out once they are finished (this is necessary if you wish to view another wish list for example).

## 5.5 SpeedOrder

This lets users enter the code number of versions and a desired quantity to add them to their basket. This is particularly useful when users may prepare their order from printed versions of the catalogue and don't want to search the site looking for those items to add to their cart. CactuShop will trap and display any errors with the data entered, such as an item not being found or the quantity being an illegal value.

## 5.6 Stock Tracking

### 5.6.1 Basic operation

CactuShop supports stock quantity tracking for items. The total of each item is adjusted each time any are purchased and items where the stock has reached zero will automatically display as 'out of stock', preventing further users from adding these to their order.

### 5.6.2 Trapping stock-related errors and issues

If a user tries to add more of an item to their basket than are currently in stock, or increase the quantity of an item in their basket beyond the total available, CactuShop will only add the available quantity. A message explaining why the quantity added was reduced is displayed and the relevant line in the basket is highlighted.

### 5.6.3 Setting up stock tracking for an item

Stock tracking can be enabled or disabled for items on an individual basis. For more details see 6.4.

## 5.7 My Account

### 5.7.1 Saved baskets

Logged in users can view a list of their saved baskets. They can load any of these saved baskets or delete them.

### 5.7.2 Order tracking and history

CactuShop allows the user to see their order history, including the quantities of items bought, pricing and discount (if any). A full history is retained so users can go back and check orders they made at any point in the past. If order tracking is enabled (`ordertracking` config setting, 7.4.13.6) then the customer can see the current status of the order and any comments you have added regarding the order.

### 5.7.3 Downloads

CactuShop now supports the ability to ship items as Downloads. This is an excellent way to ship items like software, images, online books, etc which can be downloaded rather than sent via post or email.

When you create the version in the backend, specify a full URL for the item (including `http://`). Once a customer has purchased an item that is shipped by download, they complete the transaction in the usual way. Once the transaction is completed, the item will be listed as an available download in the customer's "My Account" section. From here the customer can access ALL items previously purchased, assuming that item has not been removed from the store. In this way customers who have previously bought an item and then mislaid it can obtain another copy without requiring assistance from the merchant.

### 5.7.4 Cookie-based security

The user's email address and password are stored on a per-session cookie that expires with the end of the session.

## 5.8 Affiliates Section

### 5.8.1 Affiliate Application

Visitors to the site can apply to become affiliates. They enter their details and submit. An email is sent informing you that there is a new application to check in the back end.

### 5.8.2 Click-through tracking

CactuShop has an affiliate system that lets you track click-throughs to the site via affiliate links. This is dealt with in more detail in section 6.11.2. This page logs an affiliate click-through in the *tblCactuShopAffiliateLog* table and also plants a cookie on the visitor's browser that is used later on if the visitor makes an order through the site.

### 5.8.3 Assigning a customer to an affiliate

During checkout, CactuShop checks to see if there is a cookie planted with an affiliate ID, indicating that the visitor came to the site via an affiliate link. If there is such a cookie, the customer record created on the *mailorder.asp* page (or payment system alternative page), it is linked to this affiliate ID. All future orders by this customer (as well as the current one) will be credited to the affiliate responsible for them first signing up.

### 5.8.4 Affiliate statistics

Affiliates can access statistics on the hits and sales their link has generated for each month of the last 12 months.

## 5.9 The Checkout Process

The checkout process has changed a little in v5. These changes make the improved shipping calculations possible and reduce the amount of data entry for returning customers.

### 5.9.1 Checkout - Step 1

The user is given the opportunity to login if they have previously ordered. If so, the details and options they chose last time are recovered so they can skip through to payment with minimal further data entry. New users need only enter their email address. If it has been used before, they will be asked to login. A password reminder box is present so users can quickly recover their details if they've forgotten them.

### 5.9.2 Checkout - Step 2

Customers select the destination of goods (normally this would be their location), enter their postcode/zip (for use later if UPS shipping is used), select their preferred method of payment (these options are set from the `paymentsystem` config setting, 7.4.14.4) and coupon code (if they have one).

### 5.9.3 Checkout - Step 3

This page calculates shipping and provides the customer with a list of options. If using UPS shipping, this page will make a live lookup on the UPS real-time shipping web application and return a list of options with prices. If the store is set to use CactuShop built-in shipping, the page returns a list of options and prices in a similar way, although these are drawn from the rates set up in the back end of CactuShop.

### 5.9.4 Checkout - Step 4

The user is presented with the total price of their order including customer discounts (where applicable), coupon discounts (where applicable), shipping, and tax. The customer enters their details (if they are a returning customer, the details are already filled out). Please see 5.10 for more details about the ways that payments can be handled.

## 5.10 Online Payment

This part may vary depending on whether the user is paying via an online payment provider, through a secure area on the CactuShop system itself or by purchase order (cheque or offline method).

### 5.10.1 Payment through an online authorization gateway

On the *payonline.asp* page, the user will fill in customer information that will be retained on the CactuShop system. This is not done under SSL since no credit card or other sensitive financial information is stored. The user then clicks a button which transfers them to the secure payment form on the payment provider's web site. Once the payment is made, the gateway either

calls back the CactuShop system, or forwards the user back, confirming to CactuShop that the order was paid for successfully. At this point, CactuShop mails the order to you, a confirmation to the customer and the order will be set to 'paid' in the back end of CactuShop. Note that if you're using CactusPay, the process is the same as that employed by an online gateway, however, you will receive the card details encrypted rather than confirmation that the gateway authorized the payment.

#### **5.10.2 Payment through CactuShop under SSL**

If the payment area of CactuShop is set up under SSL and the Cactusoft Encrypted Mail system is employed, the user will fill out their personal details and credit card info into the CactuShop payment form.

#### **5.10.3 Payment by PO**

If you have customers who you wish to allow to pay on account, or by purchase order (with payment to be invoiced for later), CactuShop supports this too. You can decide whether only trusted existing customers can buy in this way, or whether all customers can. The payment method is similar to payment through CactuShop, but instead of a section of the form for credit card details, the user gets a text box to enter their PO (or reference) number that they wish to use for the order. This method can be used to accept orders from those that will pay by an offline method such as a cheque.

See section 7.4.14.2 for further information.

#### **5.10.4 Email signup**

The newsletter mailing list signup is incorporated into the online payment part of the site so users can opt-in to the newsletter when ordering. Our newsletter database follows best practice by sending an email with coded hyperlink in it to those signing up. This must be clicked to complete the signup process. This ensures that email addresses are verified before they can receive the newsletter. If the mail was sent to a non-existent address or to someone who did not request their address be added, the link won't be clicked and the user will not be added to the mailing list.

See section 6.13 for further information.

## 6 The Back-End

The back-end is used by the shop owner or operator to add, modify or delete products, categories and other items.

### 6.1 Logging In / Back End Password

Go to the *\_default.asp page* on the root of the web. This should bring up the login screen. The default login and password is "cactushop" for both. We recommend for obvious security reasons that you make changing this a priority before you put the site online. CactuShop will display a warning on the front end if the default password is not changed.

Changing the default username and password is easy. In the back end, find the 'Logins' menu section and click 'List / Find'. You are prevented from deleting or changing permissions on the main admin account but can change both the username and password for it. Other accounts can have any permissions set, can be deleted and can have the usernames and passwords changed.

### 6.2 Navigating the Back End

This is similar to the front end in appearance. The number of category levels displayed in the category menu is set from the `categorylevelsmenuback` config setting. If the user is logged in to an account with full permissions, three ticks display in the top left corner of the page. Users without full permissions will find certain links are greyed out and these pages cannot be accessed.

### 6.3 Listing Orders

#### 6.3.1 Recent orders

This link on the navigation menu lists the 25 most recent orders.

#### 6.3.2 Awaiting invoice

This link lists the orders that have not had the checkbox indicating that an invoice was issued set.

#### 6.3.3 Awaiting dispatch

This link lists the orders that have not yet been marked as dispatched.

#### 6.3.4 Awaiting payment

This link lists the orders that are marked as awaiting payment.

### **6.3.5 List orders by date**

Enter a date in the box in the format requested and a list of orders from that day is brought up.

## **6.4 Stock Tracking**

### **6.4.1 Basic operation**

Stock tracking stores levels of particular items in the database and flags them as 'out of stock' on the front end if the supply is exhausted. For more information about the front end aspects of stock tracking, see 5.6.

### **6.4.2 Warnings page**

This is the easiest way to see at a glance which items are out of stock and which are below their warning levels. Items in red are out of stock, those in grey are below their preset warning levels.

### **6.4.3 Setting up stock tracking for an item**

When creating or editing a version, two boxes refer to stock tracking. The first is the current level of that version in stock. This can be adjusted by changing the value and submitting the form. It is also adjusted as customers purchase stock through the site. The second box is the warning level. If this is set to zero, stock tracking is disabled for that item. But if an integer value higher than zero is set, stock tracking is enabled and the whole stock tracking system kicks into action for that product.

### **6.4.4 Why disable stock tracking for some items?**

Stock tracking requires commitment on your part. As stock is depleted, you must adjust the quantities of stock upwards when new stock arrives to ensure that products do not show as out of stock on the front end. For products that are never out of stock (core products, warranties, service agreements, etc.) stock tracking may be an inconvenience. Therefore, setting the stock tracking warning level to zero for such items is recommended. If you don't wish to use stock tracking for any items, just leave the stock warning level as zero for all items.

## **6.5 Statistics**

CactuShop provides graphical statistics for page views on products as well as sales. The back end stats pages dynamically count these records and display data by month, category and product, allowing you to view trends and compare what you are selling with where people are looking. Statistics for the various products and categories are now displayed in descending order of popularity.

### 6.5.1 Turning off statistical tracking

If you don't require page views statistics, you can turn the system off from the config settings. These stats can cause your database to grow in size and the more traffic your site gets, the faster the database will grow.

## 6.6 Add/Modify/Delete a Category

Use the green 'CAT' button next to 'modify the page' in the top right of the category page to bring up the category details for editing. To add a new category, click the 'CAT' button next to 'add a new category'.

### 6.6.1 Show on site

This checkbox determines whether this category should be live on the site or not. You can temporarily remove a category from display using this checkbox without having to delete it (and its products and versions).

### 6.6.2 Product display type

CactuShop has several ways of displaying products within a category. There is the single-column 'detailed' format or the multi-column 'summary' format. From this dropdown menu, you can set this as follows:

- **Default** – products in this category will be formatted according to the `defaultproddisplay` config setting, which will specify one of the three options below for use across the entire site.
- **Extended** – products in this category will be formatted in the detailed single-column format with versions displaying too.
- **Normal** - products in this category will be formatted in the detailed single-column format but without versions displaying.
- **Shortened Multi-Column** - products in this category will be formatted in multiple columns according to the `shortenedformatcols` and `shortenedformatrows` config settings.

See 7.4.4.20, 7.4.4.19 and 7.4.4.9 for more information.

### 6.6.3 Sub-cat display type

CactuShop has two ways of displaying sub categories within a category. There is the single-column format or the multi-column 'summary' format. From this dropdown menu, you can set this as follows:

- **Default** – sub categories within this category will be formatted according to the `defaultprodtypedisplay` config setting, which will specify one of the three options below for use across the entire site.
- **Normal** - sub categories within this category will be formatted in the detailed single-column format with links and description.
- **Shortened Multi-Column** - sub categories within in this category will be formatted in multiple columns according to the

`shortenedformatcols` and `shortenedformatrows` config settings.

- **Link only** - sub categories within this category will be listed as links with no description.

See 7.4.4.20, 7.4.4.19 and 7.4.4.10 for more information.

#### 6.6.4 Category name and description

There will be boxes for entry of category name and description. If you have multiple languages set up, there will be multiple boxes.

#### 6.6.5 Arbitrary order-by value and order-product dropdown

CactuShop v5 includes very flexible support for ordering products belonging to each category. From the dropdown, you can select to have this category's products ordered by name (alphabetically), price (low to high), default (which takes its cue from the global `orderby_products` config setting, see 7.4.12.2) or by an arbitrary value. Note that the arbitrary value this dropdown refers to is that set for each product (see 6.7.6), not that set in the arbitrary value box on this page. That field is used for ordering this category if the config setting `orderby_categories` (see 7.4.12.1) is set to 'PT\_OrderByValue'.

#### 6.6.6 Category images

Both a category image and large image may be uploaded.

#### 6.6.7 Deleting a category

Because of the database structure that enables products to feature in any number of categories, deleting a category becomes a complex operation both in terms of coding and the concept of what will actually be deleted.

In order to preserve database integrity, it is clearly logical to delete 'orphan' records – the products and versions that belong to a category you are removing. However, this can cause problems. What if you delete a category and find that you've wiped out products that also belonged to another category that you didn't wish to remove?

This is why CactuShop has two delete options for categories. Option 1 is a 'soft delete' – it will remove products (and their versions) only where they belong to the category you're deleting. Where they belong to other categories too, it will leave the products intact but just clean up the linkage with the deleted category. Option 2 is a 'hard delete' - it will remove the products whether they belong to other categories or not. In this case, it will also clean up the linkages between those other categories and the products that have been removed. It is important to understand the differences between these two category-delete routines.

### 6.7 Add/Modify/Delete a Product

When you click a category from the left hand menu, the list of products and versions within this category are displayed. The orange 'PRO' button brings up

a product for editing. The 'REL' button next to it launches a window for setting up related products (see 6.7.7). You can add a new product by clicking the

#### 6.7.1 Show on site

This product and its versions can be removed from the site temporarily without the need to delete it.

#### 6.7.2 Featured menu

If you wish for this product to appear on the home page as a 'featured product', select a number bigger than zero in this dropdown. Items on the home page will be ordered by this number, the bigger the number, the higher up the page the item will appear.

#### 6.7.3 Version display type

From CactuShop v5.1 onwards, there is a choice of how versions are displayed. The first is for this product to follow the `defaultversiondisplay` config setting, see 7.4.4.11.

#### 6.7.4 Categories

The categories in which this product appears can be selected from the multi-dropdown. To click multiple categories, hold down the *Ctrl* and click them (on a Mac, you need to hold down the 'apple' key instead). To deselect a category, hold down *Ctrl* and click it again.

#### 6.7.5 Product name, strapline, description and specifications

Because products can be in several languages, there will be a name, strapline, description and specifications field for each language you have set up. The strapline is the subtitle that appears in red (by default!) on the front end pages just under the product title.

The specifications field is designed for pasting of tabular data (specifications tables, etc.) and as such is formatted on the front end with a fixed-width font. Normally text on a page uses letters that vary in width (such as in this sentence!) but using a fixed-width font means that each character will occupy the same width, making simple text line up into columns. Also, the specifications text appears within `<pre></pre>` tags, meaning that lines won't wrap. The text that displays will only have line breaks where you add them. In other words, if you copy and paste a table from a word processor document or from Excel (for example), the values will line up nicely into columns.

Occasionally some stores decide to use this field for extra description rather than tabular data – in this case we advise removing the `<pre></pre>` tags from the *product.asp* page.

#### 6.7.6 Arbitrary order-by value and order-versions dropdown

CactuShop v5 includes very flexible support for ordering versions belonging to each product. From the dropdown, you can select to have this product's

versions ordered by name (alphabetically), price (low to high), default (which takes its cue from the global `orderby_versions` config setting) or by an arbitrary value. Note that the arbitrary value this dropdown refers to is that set for each version, not that set in the arbitrary value box on this page. That field is used for ordering this product within categories where the method of ordering was selected as the arbitrary order-by value.

### 6.7.7 Adding 'related products'

Click the *related products* link (the orange button with 'REL' on it) located on the right alongside each product title in the category page (*\_prodtype\_back.asp*). This will spring a pop-up window. Select the category of the product and then the product, then add it. You can select multiple products by holding down the *Ctrl* key and clicking the items you want (hold down the 'apple' key instead on a Mac). Finally, confirm the products and they're saved to the database. To delete relationships, click the *related products* link to spawn the pop-up window, then click the edit icon by the product you wish to remove. You can remove a relationship in the multiple dropdown menu by holding the *Ctrl* key and clicking it. Then save these changes and exit.

## 6.8 Add/Modify/Clone/Delete a Version

Click on the blue 'VER' button by a version in order to edit it. To add a new version, click on the 'VER' button to the right of the 'add a version of this product' text.

### 6.8.1 Cloning versions

Because of the way CactuShop allows you to create versions of products, it is often the case that you will have several different colour and size combinations. In order to minimize data entry for what are essentially similar items, cloning was introduced. By cloning a version, you bring up that existing version's details, but can make changes to it and submit it to create a new version.

This way, if you have a red item, you can easily clone it and change the colour to black, change the unique version code (SKU) and submit it as a new version.

### 6.8.2 Show on site

This checkbox determines whether this version is displayed on the web site or not. It is an easy way to temporarily remove a version without having to delete it.

### 6.8.3 Code number

All items generally have a unique code number (SKU) created by the manufacturer or the vendor's internal reference code. The software will error check to ensure these are unique as this is a handy way of accidentally adding an item twice or forgetting to change the code number when submitting a cloned version.

#### 6.8.4 Product selection

This dropdown lists the products that are in this category. You can reassign this version to another product if you wish.

#### 6.8.5 Version name and description

There will be a name and description for each language that the store is set to use.

#### 6.8.6 Price

This is the price in the default currency. From v5 of CactuShop onwards, the currency is set up from the *\_modify\_currency.asp* page. Prices for other currencies are converted dynamically on the front end using the exchange rates configured from that page. Note that prices entered here should be tax-inclusive values if the config setting `pricesinctx` is set to 'y'. If `pricesinctx` is set to 'n', the value entered here should be the price of the product without tax. If you enter a price as (say) 20 while CactuShop is set to `pricesinctx = 'y'`, then switch `pricesinctx` to 'n', the value 20 remains unchanged, but will now be considered an ex-tax value.

#### 6.8.7 Weight

The weight of the item is used by UPS shipping system to calculate the order weight, and then lookup the appropriate shipping cost. CactuShop's internal shipping calculations can also be based on weight by setting the `calcshipcostbyweight` config value to 'y'. The unit of weight used is set by the `weightunit` config setting.

### 6.9 Customer Database Admin

#### 6.9.1 Basic operation

Each time a customer first orders through CactuShop, a customer record is created. Because the email address is required to make an order, the shop can find if that customer has an existing customer record and use that for reordering. For more information, see 5.9.

#### 6.9.2 Creating customer records from the back end

Customer records can also be created from the back end, so existing clients can be set up and given the appropriate discount without needing to make a prior order through your store.

#### 6.9.3 Customer record listing

Customer records can be looked up using the email address (or any part of it, such as just the companyname.com part). The red 'ORD' button will show a full list of all orders that customer has made. The yellow 'CUS' button brings up this customer record for editing.

#### 6.9.4 Customer credit card details

Customer card details are stored in the database and in order logs if the `saveinDB` config setting is set to 'y' and credit card details are made through the secure form on CactuShop (rather than a payment gateway such as WorldPay). Retaining credit card details in the database is not advisable on security grounds, therefore CactuShop provides an easy way to remove these details once the card has been authorized and billed.

#### 6.9.5 Customer discount

To give a customer a % discount, enter the amount in the appropriate box near the bottom of the record.

#### 6.9.6 Approved customers

In some cases, you will want customers to be able to make orders and then pay offline. In such circumstances, the customer may be sent an invoice for payment later, or be required to send a cheque before the goods are dispatched. CactuShop calls such payments 'payment by PO' (purchase order). The payment form will not ask the customer for credit card details or route them to a payment gateway. Instead, it will just give a text field into which they may enter an arbitrary reference (perhaps a PO number or cheque number). This order can then be dealt with offline using their reference number.

Some stores may wish to allow any customers to order in this way. In that case, set the config setting `paymentauthorizeallpo` to 'y'. If you only wish for specifically approved (trusted) customers to order this way, set that config setting to 'n' and check the 'approved' box on the customer's record to approve just the individual customers you trust.

### 6.10 Orders

#### 6.10.1 Reviewing orders

CactuShop stores orders in the database and these can be reviewed online. You can view orders by type using the links on the left hand side of the page.

Unfinished orders are ones where the customer has entered their details and gone off to the payment gateway but there was no successful callback, either because their card was declined or because they decided not to proceed once reaching that stage.

You can find orders by date from the search box on the `_orders_list.asp` page.

The yellow 'CUS' button brings up the record of the customer who placed the order. The red 'ORD' button brings up the order itself.

#### 6.10.2 Invoicing

The stored information is used to produce invoices by simply clicking a link. The invoices can be issued in any of the languages supported on the front

end (and hence languages that the orders could be made in). The link representing the language that this order was actually made in is in capital letters. The invoices are dynamically generated based on an HTML template that is located in the skin folder of the respective language. This invoice appearance can be customized by editing the HTML template.

### **6.10.3 Order text**

This brings up the text of the email order that was sent to the customer. It is useful to have a copy of what customers receive stored, just in case the customer queries anything.

### **6.10.4 Order progress and status**

Four checkboxes allow you to set the current progress of the order. There is also a text field where you can add comments that are available to the customer through the 'my account' feature on the front end.

### **6.10.5 Turning off order progress system**

If you do not intend keeping order progress and status up-to-date (this obviously requires a time commitment), it can be switched off from the `ordertracking` config setting. This does not change anything on the back end, but removes the order status and progress sections from the 'my account' view. Order history is still available as before.

### **6.10.6 Order summary**

Details of the items ordered, shipping, coupon or customer discounts and the total actually paid are stored for each order.

### **6.10.7 Deleting orders**

This should be used sparingly to remove test orders that you may place. Removing customer orders can leave you without records of transactions that you might need for audit purposes in future.

## **6.11 Affiliate Admin Section**

### **6.11.1 Basic operation**

Affiliates are other web sites that are paid commission for putting visitors who go on to make orders through your web site. CactuShop tracks both click-throughs (visits) from affiliate web sites and the value of orders subsequently made by those visitors. Detailed stats for both are available and the system also calculates the commission due for any month in the last year for sales produced.

### **6.11.2 Technical implementation**

Affiliate sites can post a coded hyperlink pointing to your `aftrack.asp` page. They can also post a redirect page address to this as a querystring, in order

that customers can end up on a specific page of the site. The coded hyperlink should look something like this:

<http://www.cactushop.com/aftrack.asp?ID=2&Req=product.asp?product=7>

Where 'ID' is the affiliate ID number and 'Req' is the page to redirect to once the affiliate data has been logged.

The *aftrack.asp* script records the following information to the database if the affiliate number is a valid one:

- a. HTTP referer: the address of the page where the user clicked from
- b. Remote address: the IP of the user who clicked the link
- c. Requested page: the 'Req' value above
- d. Date and time: time stamp for the record.

Users also have a cookie planted on their machine containing the affiliate ID of the link they came in from. If they order from the web site, this ID number is read and the affiliate is credited with the sale. For more information see 5.8.3.

### **6.11.3 Approving, setting up and modifying affiliates**

Affiliate records can be created and edited in a similar way to customer records. When the listing of affiliates is displayed, it shows which have been approved. Affiliates that have just signed up should appear at the top of the listings. To review the affiliate application, click on the edit icon. The commission level that you wish to grant them can be set, and a password **MUST** be assigned to them. When the submit button is pushed, the details are updated and an email is automatically sent to the affiliate with the details of the hyperlink they must post on their site, their commission rate and their password. Note that it is the action of adding a password to an affiliate that approves them. Affiliates without a password remain unapproved.

### **6.11.4 Viewing statistics for affiliates**

This works in a very similar way to the main stats system (section 6.5). The total amount of clicks, order value generated and commission can be looked up by month.

### **6.11.5 Payments to affiliates**

The affiliates post links to your site in order to earn commission. Consequently, you need to track payments to these affiliates as well as the commission due in order to keep track of the account. To bring up a list of affiliates, click the *List / Find / Pay* link on the back end home page (affiliate menu section). The grey arrow button allows you to bring up an affiliate's record for editing. The pink 'PAY' button brings up a list of commissions due to the affiliate.

The list includes all commissions owed to the affiliate at that time. Overdue commissions are checked by default. You can check or uncheck boxes individually, or select all as checked/unchecked using the buttons at the top of the checkbox column. When you click the 'set as paid' button, CactuShop

creates a payment record, which is listed at the bottom of the page. The commissions can be cancelled by clicking the 'dustbin' icon.

## 6.12 Coupons

Coupons are unique codes that correspond to a percentage or fixed amount discount. Coupons are entered during the checkout procedure.

### 6.12.1 Generating coupons

Clicking the *New* link in the coupons section brings up the coupon creation form. The coupon value as a percentage or as a fixed amount in default currency can be set from the first line. Next, a start and finish date for the coupon's validity can be set. If the end date is not set, it is assumed to be valid forever (which in reality is 2050). The quantity box is used to specify the number of coupons you wish to generate – many store owners will generate coupons in bulk as part of a promotion. Finally, there is the option to set coupons as reusable. By default, a coupon cannot be used twice.

Reusable coupons with a set expiry are quite useful as promotional tools – the same code can be published in a newsletter, giving those who subscribe a period to use the coupon and obtain the accompanying discount. There is not the overhead of having to send individual coupon codes to everyone.

### 6.12.2 Listing and finding coupons

The *List / Find* link brings up a list of recent coupons, grouped by the time of generation. Similar coupons therefore appear on one line with a quantity shown. The individual coupons can be viewed and edited/deleted by clicking on the *List Coupons* link. Specific coupons can be found by entering part of the code into the search box.

## 6.13 Email List Management

Having the facility to let visitors and customers sign-up to an email newsletter is a valuable way of keeping in touch with those interested in your products. Users can sign-up via the mailing list box on the site, the 'make enquiry' form or when making an order.

### 6.13.1 Spamming

Spamming is a modern-day scourge. It is the act of sending bulk unsolicited emails to users who have not consented or requested them. Spam increases the overhead on networks and irritates the recipients. The cost of spam in Europe alone has recently been estimated as billions of Euros per year. Consequently, most networks will remove users who spam and it is against the AUP (Acceptable Use Policy) of any ISP that wants to remain connected to the rest of the internet.

Concern over spam can cause problems if your legitimate mailshot is interpreted by a recipient as spam because their address was signed up maliciously by another user, or they forgot they signed up. For this reason,

ISPs are increasingly requiring 'confirmed opt-in' for all newsletters sent through their systems.

CactuShop's mailing list is such a 'confirmed opt-in' system that uses industry best-practice to ensure that addresses are never subscribed to the list without the address-owner's consent. When someone signs up, a link with a randomly generated key is sent to them in a confirmation email. This key is also recorded in the database record corresponding to this sign-up. The subscriber must click the link in the email (which passes the random key to the web site) to confirm that they wish to be added to the list. If they do not click this, they are not added to the list. Only the real owner of the email address could have received the link with randomly generated key – hence only the email address owner can approve their addition to the list.

CactuShop logs the date of sign-up and the IP address from which the subscriber made the request. This information will be required by your host as evidence if someone erroneously makes an accusation of spamming against you.

#### **6.13.2 Send mail**

You can mail all the confirmed subscribers to your newsletter from CactuShop. Since users can opt for either plain text or HTML email, you can put two different versions of the email body text into the fields provided.

#### **6.13.3 Scope/scalability of mailing via CactuShop's scripts**

Please note that mass-mailing via an ASP script is inherently less efficient than using a dedicated mailshot program. Scripts are slower for intensive repetitive tasks and most servers have a timeout value of a few minutes set. We would therefore not recommend using CactuShop's own mailing functionality for lists of more than 1000 recipients. If you reach that level, we would suggest copying addresses from CactuShop and mailing with a dedicated mailing program. This is not a limitation of CactuShop per se, it is a limitation of ASP scripts.

#### **6.13.4 List/find**

You can view the most recent signups or look for particular people with the search feature.

#### **6.13.5 New/edit address**

You can create or edit email address records from the back end. You should beware that addresses added in this fashion will not undergo the confirmation procedure. By adding addresses in this way, you open yourself up to allegations of spamming unless you are 100% sure that the recipient's relationship with you is such that they will be happy to receive your mailings.

#### **6.13.6 Purge unauthorized addresses**

If users don't authorize their sign-up within a reasonable period, the record is effectively useless because CactuShop will not send mail to users who have not verified their email address by clicking on the link in the confirmation mail.

CactuShop can clean the database of these records in order to reduce the database size.

## 6.14 Searching for Products or Versions

In a large store or any database system with numerous records, items occasionally get filed in the wrong place. If you accidentally put a product or version in the wrong category, finding it again can be a difficult task. The back end search is designed to help you find items quickly, especially if you are not sure which category they are attached to.

The search will find both products and versions where the name, description or product code (or part of them) matches the search text you have entered. This means you should be able to find items even when the product code has been incorrectly entered or you can only remember basic details about the name or description.

## 6.15 Closing the Shop

From time to time, you might be making a major update such as uploading a modified database file or changing prices across the system. During these periods, it is a good idea to close the shop. CactuShop's back end home page gives two options for closing the shop.

### 6.15.1 Close the shop

This option sets the `shopstatus` config setting to 'locked' and shows a message on the front of the store that it is closed. Note that this setting still runs skinning and other parts of code that rely on the database, so it should only be used if you want to close the store while making price changes, etc.

### 6.15.2 Hard-close the shop

This option sets the `shopstatus` config setting to 'hardlocked' and diverts front end page requests to the `closed.html` page. This should be used if the database is about to be replaced/overwritten or other updates that will prevent the front end pages skinning code from functioning are intended. See 7.4.9.2.

## 6.16 Currencies

CactuShop supports multiple currencies. These can be edited from the 'currencies' link in the back end.

### 6.16.1 Basic set-up

Your default currency should be currency number one. So US users (for example) should change currency 1 to USD.

### 6.16.2 Exchange rates

The exchange rates for the other currencies should then be entered accordingly. The default currency exchange rate is always 1 of course.

### 6.16.3 Has decimals fractions?

Some currencies such as Japanese Yen and Taiwanese Dollars have no decimal fractions. So whereas a price in Europe could be €2.35, a price in JYN or NTD is always an integer value. To accommodate this, CactuShop lets you specify if the currency has decimal fractions or not. This ensures that it rounds to the right number of decimal places (0 or 2).

## 6.17 Tax Rates

Some items might have no tax (0%), others might have a standard rate, and some items may even have a special rate. By default, CactuShop supports up to four different tax rates, but adding more is relatively simple. Setting up these tax rates is not to be confused with whether or not tax applies to certain individuals because of their location. For example, in the UK, even an item that requires tax will be tax free to someone from outside the EU. The determination of which customers pay tax (as opposed to which items require tax) is handled in section 6.18.1.

### 6.17.1 European Union tax handling

CactuShop has support for EU VAT. If you are running a consumer-oriented site, you may decide that you will assume all customers are consumers and there are no VAT-registered companies making purchases. In this case, set up for EU VAT will be as follows:

1. Set each of the EU countries (destinations) to charge tax. For more information on setting up destinations see 6.18.1.
2. Ensure the `euvatcountry` config setting is left blank. This way, customers from any EU country will be charged VAT at the appropriate rate for each product they purchase. Customers outside the EU will not be charged tax.
3. Ensure the `euvatnumbercheck` config setting is set to 'n' – since you are not taking EU VAT numbers, you don't want to check them!

If you will have customers who are VAT-registered businesses in other EU countries, you can set CactuShop up to collect EU VAT numbers where applicable, check these, record them and deduct tax from the order. Follow these steps:

1. Set each of the EU countries (destinations) to charge tax. For more information on setting up destinations see 6.18.1.
2. Set the `euvatcountry` config setting to the two-letter ISO code of the country where your store is located (see 7.4.17.1). **Note:** in some cases this is NOT the same as the two-letter country prefix used in European VAT numbers. This way, the store knows to

charge VAT to ALL customers in that country, but only to the other EU countries where no valid EU VAT number was supplied. Customers outside the EU are not charged tax.

3. Set the `euvatnumbercheck` config setting to 'y' (see 7.4.17.2).  
**Note:** this check is only a basic check on the length etc. according to EU VAT number formats for each country. There is no real-time lookup of EU VAT numbers available to check the number exists or that it belongs to the company ordering. Furthermore, a number passing the check may still be invalid. You should always call your country's VAT authorities to check out any VAT numbers supplied and ensure that they belong to the company stated on the order.

In most EU countries, it is customary to display prices inclusive of tax. If this is the case, set the `pricesinctax` config setting to 'y' (see 7.4.17.3). This also means that the value you input for the price of a version will be considered as the tax-inclusive price. So for example, if you enter the price of an item as £10, this means £10 inclusive of whatever tax that version has. If the item has 17.5% tax, then the £10 means £8.51 (price) + £1.49 VAT. However, if the item has no tax (0%) then £10 means £10 (price) and no VAT. In this way, nice looking final prices are easily entered.

If you have tax-inclusive pricing on, order pricing is totalled by calculating the price and tax per item and then multiplying these by the quantity of items to give row totals for items value and tax. The row values are then summed to give the totals (before shipping, discounts, etc.). This method of calculating tax is how a high street store in the EU would handle tax calculations – on a per item basis.

If you have tax-inclusive pricing off, order pricing is totalled by calculating the price and tax per row (i.e. items are totalled and the tax calculated based on this). The row values are then summed to give the totals (before shipping, discounts, etc.). This method of calculating tax is how popular accounts packages in the UK handle tax when creating invoices.

### 6.17.2 US tax handling

In the US, tax works on a per state basis rather than per country (as in the EU). To enable this, you need to pin down the state that a customer is in.

The CactuShop destinations table contains records for each individual US state as well as the USA as a whole. CactuShop's outside of the USA will need just the *USA* record active and the individual US states can be disabled.

If your store is located in the US, you need to do the following:

1. Disable the main *USA* destination record and activate the individual state records, e.g. *USA – Alabama*.
2. Set the state where the store is located to charge tax, other states and countries should be set not to charge tax.
3. Set up your sales tax value as one of the tax rates and link all products for which this tax rate applies to that.

In many states, all products are subject to tax, and hence all versions in your store should be linked to the tax. However, in some states, some items such as food from a grocery store may be exempt.

Note that this system will not work if you have more than 1 state to which tax must be charged. In that case, some modifications will be required.

## 6.18 Destinations

A web site is accessible to users across the globe and you want to be able to deal with those users as smoothly as you deal with customers in your own country. Clearly shipping cost and tax due will vary depending on the location of the customer.

### 6.18.1 Charge tax by country

A UK company needs to charge VAT (sales tax) to customers in the UK and Europe but those outside of the EU are not required to pay this tax. Deciding on whether residents of a particular country must pay tax is something you might want to check with your tax authorities or accountant. In some countries like the US, tax may also be different depending on the state or province the customer is in. In this case, we would advise adding US states to the dropdown menu from the US destinations table included in the database. You can then treat these as different tax jurisdictions.

### 6.18.2 Shipping zone

Shipping to different locations costs different amounts of money. CactuShop lets you place each country into a shipping zone – each zone will have a different scale of charges applied to it. By default, zone 1 is domestic (UK), zone 2 is regional (Europe) and zone 3 is worldwide. Depending on the location of your shop, you may need to adjust these settings, making your own country zone 1 and working outwards from there. To add more shipping zones, create a new set of band records in the *tblCactuShopShippingRates* and a new *zonearray* name in the config settings.

Shipping is also dependent on either the total value of the order or the weight of the order. Which of these factors is actually used is decided by the *calcshipcostbyweight* setting in the *config.asp* file. Default of "n" will result in shipping costs being calculated based on order value. Change the value to "y" for shipping cost based on weight.

### 6.18.3 ISO country code

WorldPay and other payment systems need to know the country where goods are being shipping to. Because country names can often be quoted in different ways (UK, England, Great Britain, GB, etc.), standardized ISO codes are often used. You probably won't need to change these, but if you do, you can do it here.

### 6.18.4 Country name

Some parts of the world like their coups and revolutions and seem to change their names on a regular basis. Keep up with foreign affairs by listening or

watching to the BBC World Service and help keep your CactuShop country list up-to-date. Hey, those ISO codes might even change too...

#### 6.18.5 The country ID

This is just the database ID used for that country's record. You don't need to do anything with this but it can sometimes be useful during development.

### 6.19 Shipping Rates

This section only applies if your store is set to use CactuShop's internal shipping system. To activate this, set the config setting `shippingsystem` to 'c' (see 7.4.16.3). If you wish to use UPS real time shipping calculations, see 6.20.

Shipping rates are set in bands, each representing a range of order value (or weight) within a particular regional zone and for a particular shipping method. By default, CactuShop has five shipping zones and four different shipping methods (including 'electronic shipping'). But extra ones can be created if required.

#### 6.19.1 Adding extra shipping methods

First open the `tblCactuShopShippingMethods` table. Add a new record and enter the name(s) and description(s) required. Make a note of the `SM_ID` value assigned to it.

Next, open the `tblCactuShopShippingRates` table. Copy all records where the `S_ShippingMethod` value is 1 and paste them into the foot of the table. To copy records you will need to click and hold down the mouse button on the grey margin to the left of the record ID column, and drag down, highlighting the required records as you go. After this, change the `S_ShippingMethod` value for all the new records to the `SM_ID` value you noted above. You can edit the values of these from the back end later.

#### 6.19.2 Adding extra zones

To add an extra shipping zone, copy all the records in the `tblCactuShopShippingRates` table where the `S_Zone` value is 1 and paste them into the foot of the table, then change the `S_Zone` value to 6 (or whatever the new zone is to be).

You will then need to add some new code to handle the names of new zones. First step is to create a new language string called `Config_Zone6` (for example). You would then need to add the `LS_ID` value of this to the PageStrings statement at the top of pages where it is used (for example, the `shipping.asp` page). For that page, you'd also need to add an extra array line under the others that are there:

```
aryZone(6) = GetString("Config_Zone5")
```

## 6.20 UPS Real Time Shipping

### 6.20.1 About UPS Real Time Shipping

For further information regarding UPS Real Time Shipping, see:

<http://www.ups.com/>

Download the shipping documentation UPS provide. You will need to sign up for an account on the UPS web site in order to use UPS shipping, though this is free (at time of writing).

The UPS real time Shipping integration in CactuShop allows you to get live shipping costs dynamically from the UPS web site. During the checkout process, various pieces of information (such as the weight of the items in the basket, the address of the shop and the destination address) are passed over to UPS, which then works out the various shipping methods (and corresponding prices) it can offer. These are passed back in XML format and CactuShop uses this information to format the shipping selection page, *payonline.asp* page and *mailorder.asp* page (and gateway equivalents such as *WORLDPAY\_mailorder.asp*).

The result is a display that looks very similar to when CactuShop integrated shipping is selected. The difference is that the methods and prices are those quoted by UPS in real time, specific to this one order. It will also give estimated delivery times for some of the shipping methods.

Please note that accuracy of the options and prices quoted is subject to some limitations. The price quote of UPS is only as good as the accuracy you have used when setting product weight within CactuShop. Various other factors that are not passed to UPS, such as package dimensions, can also affect the price. That said, CactuShop's UPS integration should produce accurate rates providing your products are not unduly bulky or oversized in one or more dimensions.

Furthermore, returned delivery estimates (e.g. "by 8am tomorrow") are only accurate if you place your order with UPS the moment that the customer's order is made. Clearly 'next-day' will not be next day (as far as the customer is concerned) if you only package and dispatch the order 24 hours after receiving it. The language string `ContentText_UPSGuaranteedDeliveryExpl` is displayed with delivery estimates so that the customer is made aware of this. The text of this notification can obviously be changed through the language strings editor (see 9.3 for more detail on editing language strings).

Assuming you have set up a UPS account, to switch on UPS Real Time Shipping you first need to ensure that you have Microsoft's MSXML3.0 DLLs installed on your server. Contact your host for information if you get errors relating to this. You then need to configure the UPS config settings, and, finally, change the config setting `shippingsystem` to 'u' (see 7.4.16.3).

## 6.20.2 Config settings – UPS

### 6.20.2.1 UPS\_AccessKey

Your UPS access key; this is required in order to use any UPS feed. You can request this through your UPS back end. Make sure you request a key for the XML feed and not for an HTML one.

### 6.20.2.2 UPS\_Commercial

y/n - whether all addresses you deliver to are commercial. Commercial deliveries are slightly cheaper than residential, so if you're sure all your products are going to commercial addresses, give your customers a discount by switching this on.

### 6.20.2.3 UPS\_CurrencyConversion

UPS will return the shipping cost in the currency of the shop country (see config setting `UPS_ShopCountryCode`). If this is a different currency to your default site currency (e.g. your shop is in the UK, but your default currency is Euros rather than Pounds), enter a conversion rate here. Otherwise, leave this as a conversion rate of 1.

### 6.20.2.4 UPS\_MoreInfoURL

When the UPS shipping method selection is created, a URL is displayed to find out more information on the different shipping options. The default points to a US-specific page, so you may want to alter this to make it specific to your location.

### 6.20.2.5 UPS\_PackageType

The type of package UPS will be sending (default is 02).

00	unknown
01	UPS letter
02	customer supplied package
03	tube
04	PAK
21	UPS Express Box
24	UPS 25kg box
25	UPS 10kg box

See UPS documentation for more details.

### 6.20.2.6 UPS\_Password

Your UPS account password.

### 6.20.2.7 UPS\_PickupType

How UPS will pick up your package (default is 01).

01	daily pickup
03	customer counter
06	one time pickup
07	on call air
11	suggested Retail Rates
19	letter center
25	air service center

See UPS documentation for more details.

#### 6.20.2.8 UPS\_ShopCountryCode

The country code for the shop location – i.e. where UPS will be picking up the packages from. Use GB for all of Great Britain, or the ISO code for your country (these can be found in the *tblCactuShopDestinations* table).

#### 6.20.2.9 UPS\_ShopPostalCode

The postal code/zip of the shop location.

#### 6.20.2.10 UPS\_ShopRegionCode

us/eu/c/pr/m/o - The region where the shop is located. United States, European Union, Canada, Puerto Rico, Mexico and Other, respectively.

#### 6.20.2.11 UPS\_ShowDeliveryDays

y/n - whether to show the number of days and the time the delivery is guaranteed to arrive. Note that only some inland deliveries retrieve this information.

#### 6.20.2.12 UPS\_UnitOfMeasurement

KGS/LBS – this is the unit of weight measurement that your store uses. It should match your `weightunit` config setting (see 7.4.9.4). The difference between these two is that this config setting must be either KGS or LBS (case sensitive) whereas the `weightunit` config setting is purely for display purposes and as such can be set to 'lbs', 'Pounds', 'kg', etc. CactuShop will pass the unit to UPS when requesting prices – if the unit used generates an error from UPS (e.g. your store uses Kilos but you are in the USA so the UPS system wants to see lbs) then CactuShop can handle this automatically. It will convert the weight to the alternative unit and repost it to UPS in order to get a price back.

#### 6.20.2.13 UPS\_URL

URL of the UPS XML feed – this is very unlikely to change.

#### 6.20.2.14 UPS\_UserID

Your UPS account username.

## 6.21 Languages

CactuShop supports up to 5 languages. These are switched on or off via the back end. For more information on working with languages in CactuShop, see section 9).

### 6.21.1 Switch languages on or off

You can choose which languages are featured using the checkboxes by each language. If you only have one language, it is recommended to switch the others off both front and back of the site so that you aren't presented with unnecessary fields when editing products, categories, etc.

### **6.21.2 Back and front names**

These are the names by which the language is referred to on the front and back of the site. Typically, the language will be referred to by its local name on the front of the site and by the default language version in the back end.

### **6.21.3 Skin location**

Since CactuShop supports multiple languages, each will require a skin of its own. The path to each skin is edited from here.

### **6.21.4 Contact email addresses**

In prior versions of CactuShop, the email addresses used to send orders to the store owner were stored as config settings. However, with the introduction of multiple languages, these email addresses have been moved to the language section. This ensures that you can (if you wish) set up different email addresses (and hence different support staff) to process orders from each language.

## **6.22 News Items**

You can easily add news items here for display on the front of the site. The news items are added from the back end of the site and are stored in the database - no FTPing required.

## **6.23 Mark Up Items**

CactuShop includes a tool to help you make global changes to prices on your site in bulk.

You can select a price range of the items you wish to re-price, the category they are in and the amount you wish to mark them up or down by. The amount can be a percentage or a fixed amount.

A confirmation screen listing all the versions that will be affected then displays. You can uncheck any that you don't wish to re-price before confirming.

## **6.24 Static HTML Product Gateway Pages**

### **6.24.1 The debate over search engine indexing of database sites**

All serious e-commerce web sites use dynamically generated pages created from information stored in an online database. However, those vendors selling lightweight e-commerce products often argue that static HTML pages are better because they can be indexed by search engines, whereas database content cannot. In the past, this was true. However, since most major news, e-commerce and general interest sites are now databased, search engines have made major advances in indexing dynamic content. Otherwise, they would miss the best parts of the web! But just to be on the safe side, CactuShop now includes a facility to build static versions of the product pages

automatically – so you get the vast benefits of using a database, with all the benefits of static HTML too.

#### **6.24.2 Creating the static pages**

On the gateway pages screen, just click the link. The system will build a series of static pages, one for each product. It will also generate a static default.htm document that links to all the static HTML pages. This document is linked to by a hidden image on the front of the web site, ensuring that a search engine can follow through the pages and index them.

#### **6.24.3 Products no longer on the site**

No facility is built in for removing pages of products that are no longer present. You should keep all static HTML pages where possible because each may have search engine links to it that will result in lost sessions if users click to come through to your site only to get a 404 'page not found' error.

If a user clicks through from a gateway page but the product is no longer available, they will simply get a 'product not available' message once they click on to the CactuShop product page. At least this way they still arrive successfully at your site and can search for other products.

#### **6.24.4 Folder permissions**

The HTML folder must be set to allow the Internet Guest Account to write to it (as for all image upload folders). If this permission is not set, the attempt to create static pages will fail and an error message will display. For more details see 3.3.

## 7 The Config System

This is the “junction box” of the whole system and contains most of the switches to configure the commonly used elements of CactuShop. In v3 and prior versions, the *config.asp* file was a list of constants and settings that controlled many elements of the store. Since v4, this has been largely replaced by a databased application variable system using the *tblCactuShopConfig* table to store the default settings. The database connection string still needs to be set in the *config.asp* file, but once this is in place, most other settings can be changed online from the back end.

### 7.1 Basic Operation

The config settings are stored in server memory, using ‘application variables’. These values are created by the *config.asp* file if it cannot find existing ones in memory, or if a page tells the config settings to be rebuilt from those stored in the database. Application variables are shared across all pages and users of the application. Only one instance of each is stored in memory. Although similar in nature to session variables, application variables do not time out and do not hog huge amounts of server memory when the site is busy since all users share the same single set of variables.

It is important to understand that CactuShop uses the config settings that are stored in the server’s memory; **it does not** check the database each time a page is generated. If you change the config settings values in the database using Enterprise Manager (for an SQL site) or upload a new Access database to an existing site, the site may start to error because it is still using the old copies of the config settings that it had in memory. For information in dealing with problems caused by this, see **Error! Reference source not found.**

### 7.2 The License Number

Each copy of CactuShop has a unique license number that must be present for any site using the software on the web. Each license number may only be used for one site. The part of the code that builds the license number into the page HTML may not be removed. Please see the terms and conditions if in doubt. The license number should be entered into the space provided within the *config.asp* file.

### 7.3 Database Connection

#### 7.3.1 Hard-coded connection values

The settings relating to CactuShop’s connection with the database need to be set up in the *config.asp* file. These are coded into the script rather than in the database. The settings are as follows:

##### 7.3.1.1 strDBtype

This can be “access” for MS Access, “mssql” for MS SQL Server or “mysql” for the open-source MySql database. This setting controls the syntax used in

certain SQL statements since there are subtle differences in places for each database.

#### 7.3.1.2 `strDataSourceName`

Two sample strings are supplied, for either MS Access or MS SQL Server. These will obviously need to be edited to reflect the name of your database, machine name and/or password. You can also setup your database as an ODBC (Open Database Connectivity) datasource. To do this, first open the ODBC administrator. In Win2000, this is under "programs" but in older versions of Windows it is called 32bit-ODBC and is a control panel. You need to add a system datasource, selecting the Access file or SQL Server database you wish to use and giving it a name. Scripts such as CactuShop can then link to this database by datasource name without knowing the path to the file. So your connection string will change to:

```
strDataSourceName="myodbcdatasourcename"
```

Note that we don't specify any file with ODBC connections – just the name of the data source. In theory, you can change the database type and providing you give it the same ODBC data source name as before, the scripts do not need to be changed. In practice though (as previously discussed), minor changes to the SQL statements are needed when changing from Access to SQL Server. We've built in support for this, hence all you need to change is the `strDbType` switch.

#### 7.3.1.3 `numCursorType`

This setting can prevent certain types of operations such as recordcounts from returning correct values. Different versions of database drivers and operating systems require a different setting here. By default the cursor is type 3, but after upsizing to SQL, you may find the cart functions erratically and procedures that require recordcounts return -1. In this case, try changing the `cursorType` to 1. The effects of wrong cursor type can be so erratic that it may not be immediately apparent what the problem is. If the cart acts weirdly, try this before tearing your hair out.

### 7.3.2 Other database types

CactuShop is only tested with MS Access 2000, MS SQL Server 7.0 & 2000 and MySQL (Windows version). Other versions of Access and SQL Server should work fine providing your server has the correct drivers installed. Older versions of MySQL may not support some SQL syntax used in CactuShop. No changes of scripts are required. If you need an Access file in 97 format this can be downloaded from our web site 'downloads' page.

If you wish to use CactuShop with Oracle or other databases, we would suggest starting by switching the `strDbType` to SQL syntax. However, we would expect that some SQL syntax changes would be required. We cannot offer tech support for databases that CactuShop has not been specifically designed to work with.

## 7.4 The Config Table

### 7.4.1 Overview

#### 7.4.1.1 Basic operation

The *tblCactuShopConfig* table contains the name of the application variable, the value to be stored in it, and an identifier 's' or 'n' to designate whether the variable is a string or a number. So although every value is stored in the database as a string, it is possible to control what type of application variable is created from these settings. Also, there is a field that lists the minor version of CactuShop in which this variable was added (starting from v5). This should make the process of upgrading stores easier, as it will be clear which new config settings need to be added.

#### 7.4.1.2 Changing config settings

To edit the config values, go to the *cactushopback/default.asp* page and click the 'Edit Config Settings' link in the centre of the page. This will bring up a list of all settings in groups. To change a setting, click the modify button, update the value, and submit it.

**Important:** the database values held are simply the default values that CactuShop will use to create the application variables. It is the application variables held in memory that the site uses to generate the pages. If you change values in the database using the CactuShop back end, these application values are automatically refreshed when you click the submit button after changing a value. Remember that if you change config settings on the database and upload this, these new settings will not take effect until the server reboots or you edit one of the values from the CactuShop back end and submit the change. Sometimes, having a stored set of old config settings in memory can cause page errors when you upload a database that has different config values. To fix this, pass this querystring to any page: *appvar=reset*. For example:

<http://www.mydomaingoeshere.com/default.asp?appvar=reset>

### 7.4.2 Allowable Lengths

In order to error check database input of strings to ensure they don't exceed the length allowed in the database, we can set values to use in error-checking. These values should be the same as the corresponding field size in the database. If you change the size of those fields within the database, you should update the allowable lengths to ensure error-checking continues to work. If overlong string input is made to a database, some older Access drivers will just truncate the data string, but SQL Server and newer Access drivers will display a database error and the query will fail. All allowable length settings are named in the following format:

[name of database field] + *\_AL*

So for example *CD\_CardholderCompany\_AL* is the allowable length of the cardholder company field.

### 7.4.3 Config settings – Credit Card Details

This excludes the config settings that apply to allowable lengths, UPS and payment gateways.

#### 7.4.3.1 SaveInDB

Determines whether full credit card details are saved in the database or not - y/n. If 'n', all but the last four digits are replaced with xxxx.

#### 7.4.3.2 SaveInEmail

Determines whether credit card details are sent in the order email - y/n. This is only advisable if you have SSL and the encryption system running on your site. It has no effect if you are using CactusPay or an online gateway.

### 7.4.4 Config settings – Display

#### 7.4.4.1 addtobasketdisplay

l/b/t/d - Specifies whether the 'add to basket' method should be a link, button, textbox or dropdown quantity menu. This config setting applies to the versions in all products where the version display type is not selected as 'As options'.

#### 7.4.4.2 addtobasketdropdownmax

If dropdown is set for the `addtobasketdisplay` config setting, this sets the maximum quantity in the dropdown. So a number of 15 will see the 'add to basket' dropdown go from 1 to 15.

#### 7.4.4.3 autoforwardtogateway

y/n - determines whether to use JavaScript to forward direct to payment gateways rather than the user having to click a button to go there. If set to 'y', a customer will be automatically forwarded through the `[gateway]_mailorder.asp` page and onto the payment gateway if no errors from the `payonline.asp` page are flagged. If set to 'n', the customer will see a page with a button that they must manually push in order to go off to the gateway.

#### 7.4.4.4 categorylevelsmenuback

Total number of hierarchy levels to show on the back end category list on the left menu. Set to 0 to show all. Too many levels might make the menu too long and complex on a large site with lots of categories.

#### 7.4.4.5 categorylevelsmenufront

Total number of hierarchy levels to show on the front end category dropdown/list in the skin. Set to 0 (zero) to show all.

#### 7.4.4.6 categorylevelsmenusearch

Total number of hierarchy levels to show on the front-end search category list. Set to 0 (zero) to show all.

#### 7.4.4.7 convertdescbreaks

y/n - whether to convert the breaks in description fields (for categories, products and versions) in to HTML break tags ("`<br>`"). Turn this off if you want full control over the HTML formatting of description fields.

**7.4.4.8 defaultcountry**

ID number (in *tblCactuShopDestinations* table) of country to pre-select by default choice on menus.

**7.4.4.9 defaultproddisplay**

e/n/s - determines the default product display type on category pages - e for extended (shows version information), n for normal and s for shortened multi-column format. See 6.6.2 for further explanation.

**7.4.4.10 defaultprodtypedisplay**

n/s/l - determines the default category display type - n for normal, s for shortened multi-column format, l for link only. See 6.6.3 for further explanation.

**7.4.4.11 defaultversiondisplay**

r/o/p - determines the default version display type – 'r' for rows (standard HTML table layout as with previous CactuShop versions), 'o' for dropdown (just titles), 'p' for dropdown with titles and prices. Note that with the latter two options, any version level images and descriptions will NOT be displayed.

**7.4.4.12 newestproductslist**

This is the number of products that display on the home page 'newest products' list.

**7.4.4.13 peoplewhobought**

Maximum number of links to display on the *product.asp* page for 'people who bought this product also bought'.

**7.4.4.14 peoplewhoboughtjustorder**

y/n – whether the 'people who bought this also bought' is calculated based on items bought in the same order (y) or to items bought by them in any order (n).

**7.4.4.15 prodtypemenuasdropdown**

y/n - determines whether the category menu is displayed as a dropdown or hyperlinks. To control the number of levels of hierarchy in these menus, see the *categorylevelsmenufront* config setting (7.4.4.5).

**7.4.4.16 prodtypepagesize**

Number of products per category page when display is in normal or extended view. For multi-column view, see config settings *shortenedformatrows* and *shortenedformatcols* (7.4.4.20 and 7.4.4.19).

**7.4.4.17 searchmaxrecs**

Maximum number of records pulled out in a search.

**7.4.4.18 searchpagesize**

Number of products per page of search results.

**7.4.4.19 shortenedformatcols**

The number of columns to use when displaying products and sub-categories on *prodtype.asp* if the category is set to 'multi-column' format.

**7.4.4.20 shortenedformatrows**

The number of rows to use when displaying products and sub-categories on *prodtype.asp* if the category is set to 'multi-column' format.

**7.4.4.21 shortenedformatshowdesc**

y/n - whether to show the product description when products are displayed in shortened, multi-column format. This should only be used if your products have short descriptions.

**7.4.4.22 shortenedformatshowversions**

y/n - whether to show the product description when products are displayed in shortened, multi-column format. This should only be used if your products have short descriptions.

**7.4.4.23 showbasket**

This setting controls how the basket behaves when an item is added to it. There are three options – 'y' means the basket will display when an item is added to it, 'n' means that no basket will display (so the item will just appear in the mini basket). The third choice is to set this value to a number – in this case a message indicating that the item has been added to the basket will display for the number of seconds specified before redirecting back to the product page. **Note:** CactuShop accomplishes the redirection using a meta-refresh HTML tag. This is placed on the page by reference to the *<head>* tag in your template. In order for this redirection to work, your template file **MUST** include a *<head>* tag.

**7.4.4.24 showmoredetailink**

l/b – determines whether the 'more details' link by each product is a link or a button.

**7.4.4.25 showrrponversiondisplay**

y/n - whether to show an 'RRP' column on the version display on the front end.

**7.4.4.26 showweightonversiondisplay**

y/n - whether to show a 'weight' column on the version display on the front end.

**7.4.4.27 specialsdisplaytype**

s/n/e - determines how the front specials are displayed - Shortened Multi-Column Format, Normal, or Extended (with Version information)

**7.4.4.28 speedrowcount**

Specifies how many rows appear on the speedorder form.

**7.4.4.29 topsellingproductslist**

The number of products that should appear on the top seller list. Zero turns the list off.

**7.4.4.30 topsellingproductsdays**

The number of days to use when calculating 'top selling items' on the home page. For example, 30 means that the list represents the top selling products over the last 30 days. Set to zero for all orders in the history of the store.

**7.4.4.31 truncatenewestproducts**

Specifies number of characters to truncate the product names on the newest products list. Zero is for no truncation (show whole name regardless of length).

**7.4.4.32 truncateproductoncategory**

Specifies number of characters to truncate the product description on the category display page to. Zero is for no truncation (show whole description regardless of length).

**7.4.4.33 truncateproductonspecials**

Specifies number of characters to truncate the product descriptions on the front end specials. Zero for no truncation (show whole description regardless of length).

**7.4.4.34 truncatetopsellingproducts**

Number of characters to truncate the product names on the 'top selling products' list. Zero for no truncation (show whole title regardless of length).

**7.4.4.35 trythesecategories**

This is the maximum number of entries to show on the 'try these categories:' section on the product display (this section shows the other categories this product exists in - won't show if there are no other categories).

**7.4.5 Config settings – Downloads****7.4.5.1 downloadlimitdays**

This is the number of days for which the download will be available to the customer to download. Beyond this, they will not be able to download the software by logging into their account. This time limit is useful if you issue software upgrades for a set period.

**7.4.5.2 hiddendownloads**

y/n - If downloads are hidden they are streamed to the page - making the location of the file hidden from the user. If you enable this, then all links to files must be relative and in directories that have IUSR read permissions. We would recommend that you upload the item through CactuShop if you are unsure of the path. If this setting is disabled (n), you must ensure that the download file permissions enable web users to download them.

**7.4.5.3 instantdownload**

y/n - determines whether or not downloadable products are instantly available when an order has been put through. If set to 'n', the order has to be set as 'dispatched' before the customer can download the product. This is useful if you have high value downloadable products or suffer a high level of fraud and want to ensure that someone is not given access to downloads until you have manually checked out the order and satisfied yourself that it is genuine.

**7.4.6 Config settings – Email Settings**

#### 7.4.6.1 emailmethod

This specifies which of the several supported mail methods, or the two test modes, is used. The four methods currently supported are: 'CDO' (Microsoft standard), 'Jmail' ([www.dimac.net](http://www.dimac.net)), 'ASPmail' ([www.serverobjects.com](http://www.serverobjects.com)) and 'Persits' ([www.persits.com](http://www.persits.com)). The two test modes are 'write', which disables mail sending and writes the email to the screen instead, and 'off' which simply disables mail sending altogether. When developing sites on a local machine or network, the test modes allow you to simulate formulating and sending mail without having to install mail components or wait for mail to be sent and received in order to check it is formatted correctly.

#### 7.4.6.2 mailserver

This is the address of the outgoing SMTP mail server. It might be an IP address or in the form of *smtp.cactusoft.com*. CDO does not require a value for this setting (since it is configured at the server), but the third party components do. Virtually all hosts using a component will have an SMTP server on their network that is set not to require authentication for requests coming from the web server IP. For this reason, CactuShop does not include code for authentication when sending via an SMTP server.

#### 7.4.6.3 newsletterfromemail

Email address the newsletter should appear to be from.

#### 7.4.6.4 newsletterfromname

Person/name the newsletter emails should appear to be from.

#### 7.4.6.5 senderroremail

y/n - whether to send an email to the store owner when an error on the site occurs.

#### 7.4.6.6 sendorderupdateemail

y/n - whether to offer the cardholder the option to receive email updates when their order is updated (this can be overridden on the modify orders page, 6.10.1).

### 7.4.7 Config settings – Encryption

#### 7.4.7.1 encrypt

Determines whether to use encryption on order emails sent to store owner - y/n. To use this, you must have the Cactusoft Encrypted Mail system installed. For more information see **Error! Reference source not found..**

#### 7.4.7.2 encryptionkeyfile

The name of the keyfile at the client end. This enables the CEM system to use the correct key to decrypt. For more information see **Error! Reference source not found..**

### 7.4.8 Config settings – Ignore

#### 7.4.8.1 lastupdated

Don't change this. It holds the time that the config settings were last rebuilt and allows CactuShop to run on multi-server environments. The value is

stored and saved by CactuShop automatically, you should not manually adjust this.

#### 7.4.9 Config settings – Important

##### 7.4.9.1 cookiename

Name of the cookie used by the cart.

##### 7.4.9.2 shopstatus

The status of the shop - can be 'open', 'locked' (the shop stays in the site skin and displays a message explaining the shop is shut) or 'hardlocked' (the shop redirects to closed.html). Use 'hardlocked' if you're going to be modifying the database. This config setting can also be set from the links on the back end home page. See 6.15 for more information.

##### 7.4.9.3 webshopURL

The name of the shop used within emails and on some pages.

##### 7.4.9.4 weightunit

Unit of weight used in pages.

##### 7.4.9.5 uploadsfolder

CactuShop requires certain folders to have IUSR full control permissions (see 3.3 for further details). All such folders are within a folder on the root of the web which by default is called 'uploads'. Some hosts only set permissions for one such folder and it may have a different name. In that case, the folders within the *uploads* folder in CactuShop should be copied to that folder on your hosts server and this `uploadsfolder` config setting set to the name of that folder.

#### 7.4.10 Config settings – Languages

##### 7.4.10.1 defaultlanguageid

The ID of the language in *tblCactuShopLanguages* that all default users use before they specify one.

#### 7.4.11 Config settings – Miscellaneous Settings

##### 7.4.11.1 affiliatecommissionduedays

Number of days from the day an order is placed when an affiliate commission is flagged as 'overdue'. Only used as indication on the back end.

##### 7.4.11.2 allowedimages

List of acceptable image file types, comma separated. If you attempt to upload files that are not of any of the types listed here, CactuShop will flag an error.

##### 7.4.11.3 allowenquiries

y/n - determines whether the enquiry system is enabled or not.

**7.4.11.4 basketmaxcapacity**

Size of array used to store basket items, can be any value from 2 - 500. Note that higher values will slow the cart down so set it only as big as it needs to be. This number represents the maximum number of different items the basket can hold.

**7.4.11.5 cookiecheck**

y/n -

**7.4.11.6 minimumcustomercode size**

Minimum size allowed for customer codes. To allow any length, set to 0. Customer codes that are too short are less secure.

**7.4.11.7 tracking**

This specifies whether statistical tracking of product views is activated. Switching this off can result in a small performance gain.

**7.4.11.8 trackingsales**

This specifies whether statistical tracking of sales is activated. Switching this off can result in a small performance gain.

**7.4.11.9 usesecurearea**

If your site has a secure certificate on the domain itself (as opposed to a separate shared secure area), set this to 'y'. For this to work, you need to be able to call any page on the site with HTTPS instead of HTTP.

**7.4.12 Config settings – Ordering**

By 'ordering' here, we are referring to the order that items appear on the page rather than the process of making an order through the store.

**7.4.12.1 orderby\_categories**

Determines which field is used to order the categories. Common choices are PT\_Name, PT\_OrderByValue.

**7.4.12.2 orderby\_products**

The default setting for ordering of products - can be overridden for each category.

**7.4.12.3 orderby\_versions**

The default setting for ordering of versions - can be overridden for each product.

**7.4.13 Config settings – Orders****7.4.13.1 checkinvoicedonpayment**

y/n - on a successful payment (a payment gateway has authorized payment), setting this to 'y' will automatically check the 'An invoice for payment has been issued' checkbox for the order.

**7.4.13.2 checkreceivedonpayment**

y/n - on a successful payment (a payment gateway has authorized payment), setting this to 'y' will automatically check the 'Payment has been received' checkbox for the order.

**7.4.13.3 invoiceduedays**

Number of days in which invoices should be paid... used for showing overdue ones in the back end.

**7.4.13.4 minorderval**

Minimum order value that the online store will accept (value before tax, in default currency). If the value is lower than this and the user attempts to checkout, a JavaScript alert box springs up.

**7.4.13.5 orderhandlingchargeband**

Determines which tax rate is applied to the order handling charge. To set or disable the order handling charge, see the `orderhandlingcharge` config setting, 7.4.13.7.

**7.4.13.6 ordertracking**

Order tracking lets users login and gain information on the progress of their order. If you don't wish to enable this feature due to the added workload of updating order details, it can be easily disabled here (the link on the template will also need to be removed manually).

**7.4.13.7 orderhandlingcharge**

An additional order-handling charge that is added on to all orders. Set to 0 to disable.

**7.4.13.8 ordertransactions**

y/n - whether to send through the order as a 'transaction'. With this switched on, if an error occurs in any of the queries during the order process then the database will be rolled back and the whole order cancelled. For example, during an order, CactuShop creates a customer record, order record and invoice rows record(s). However, in some versions of MySQL transactions are not supported.

**7.4.13.9 purgeunfinisheddays**

Determines the number of days after which an incomplete order will be purged when the 'purge order' button on the 'unfinished orders' listing is pushed.

**7.4.13.10 termsandconditionscheck**

y/n - whether to force the user to check a 'I have read and understand the terms & conditions of the site' checkbox when ordering. Link pops up site terms & conditions (set in language string `ContentText_TermsAndConditions`).

**7.4.14 Config settings – Payment****7.4.14.1 luhncheckON**

y/n - determines whether the check to see if the credit card number format is correct is on or off.

**7.4.14.2 paymentauthorizeallpo**

This setting determines whether the shop will give **all** customers the ability to pay with a purchase order number (essentially asking for you to invoice them, arrange payment by cheque, etc.). If not, only those customers who have been set to 'approved' in the back end can make orders in this way.

#### 7.4.14.3 **paymentprocesscurrency**

This value is the ISO code of the currency that orders will be passed in when made through the CactuShop built-in payment system. For example, your store may offer several currencies, but you may wish to only accept orders in GBP. In this case, set this config setting to 'GBP'. Customers will see a message explaining the conversion reasoning at the point before they submit their order. If this config value is left blank, the order will be passed in the currency that the customer has selected and is using to view the web shop.

#### 7.4.14.4 **paymentsystem**

CactuShop supports several different payment methods, which can run side-by-side. This field stores a comma-separated list of the various options that you wish to make available to your customers. Note that you will still need to configure the individual methods that you have set up accounts for – all this setting does is determine which methods you make available in the dropdown on the *shipto.asp* page. Note that the string should be lower case and contain no spaces.

### 7.4.15 **Config settings – Payment Gateways**

#### **WORLDPAY**

For information on setting up WorldPay, and the purpose of the various WorldPay config settings, please see section 8.1.

#### **2CHECKOUT**

For information on setting up 2CheckOut, and the required values for the various Authorize.Net config settings (2CheckOut implementation in CactuShop uses the Authorize.Net config settings), please see section 8.2.

#### **AUTHORIZENET**

For information on setting up Authorize.Net, and the purpose of the various Authorize.Net config settings, please see section 8.3.

#### **PAYPAL**

For information on setting up PayPal, and the purpose of the various PayPal config settings, please see section 8.4.

#### **PROTX**

For information on setting up Protx, and the purpose of the various Protx config settings, please see section 8.5.

#### **SECPAY**

For information on setting up SecPay, and the purpose of the various SecPay config settings, please see section 8.6.

#### **SECURE TRADING**

For information on setting up Secure Trading, and the purpose of the various Secure Trading config settings, please see section 8.7.

#### **VERISIGN**

For information on setting up VeriSign, and the purpose of the various VeriSign config settings, please see section 8.8.

## 7.4.16 Shipping

### 7.4.16.1 `calcshipcostbyweight`

y/n - determines whether shipping price calculated by CactuShop's internal system is based on weight or order value. If config setting `shippingsystem` isn't set to 'c' (CactuShop integrated shipping) then this config setting is irrelevant. See 7.4.16.3 for further information.

### 7.4.16.2 `skiponeshippingmethod`

y/n - If there is only one shipping method available to the customer, setting this to 'y' will bypass the shipping form on checkout.

### 7.4.16.3 `shippingsystem`

c/u - 'c' for CactuShop internal shipping. This calculates the shipping price based on the rates in the `tblCactuShopShippingRates` table and the methods in the `tblCactuShopShippingMethods` table. The shipping rates can be adjusted from the `_modify_shippingrate.asp` page. See 6.19 for more information.

### 7.4.16.4 `shiptaxband`

This is the ID value of the tax band that should be applied to shipping. Note that this should not be set to zero (since there is no tax band with the ID value of zero). Instead, '1' would link to tax band 1 which is zero percent tax (in default database).

## 7.4.17 Config settings – Tax

### 7.4.17.1 `euvatcountry`

The ISO code of the EU country where the store is based. Leave blank if you don't wish to activate EU VAT handling. For further information on EU VAT, see 6.17.1.

### 7.4.17.2 `euvatnumbercheck`

y/n - determines whether you wish to activate CactuShop's error checking of EU VAT numbers. For further information on EU VAT, see 6.17.1.

### 7.4.17.3 `pricesinctax`

y/n - determines whether the prices entered for versions include or exclude tax. Business to business (B2B) and North American web sites generally show prices excluding tax. European business to consumer (B2C) sites generally show prices including tax. The choice of whether prices entered include or exclude tax also affects the price calculations in a technical way. B2C pricing (prices ex tax) is calculated by summing each row of the basket and then applying tax to that. These values are then added up to give the total. In the case of B2C (prices inc tax), tax is calculated for each item. These are then multiplied by the quantity and then the row values added together to give the total.

### 7.4.17.4 `showtaxdisplay`

y/n - switch CactuShop between showing an ex tax and inc tax price or just an inc tax price when displaying tax inclusive prices - if `pricesinctax` (7.4.17.3) is set to 'n' then tax is always shown (as a percentage).

#### **7.4.17.5 taxshippingbilling**

s/b - determines whether decision to charge VAT is based on the shipping address or the billing address. This config setting only matters if EU VAT is activated. See the `euvatcountry` config setting (7.4.17.1).

#### **7.4.18 Config settings – UPS**

See the UPS section 6.20.2 for these.

## 8 Online Payment Methods – Set Up

This section deals with the methods supported by CactuShop for accepting online payments.

**IMPORTANT:** We would **strongly** recommend using an online gateway to accept credit card payments for a number of reasons. Firstly, they maximize security. The card details are given on the gateway's own site, so you and your site never see the credit card details. Even if your site or web host was ever compromised, credit card numbers could not be stolen. Secondly, online gateways have a number of security checks in place, reducing the likelihood of fraud against you (it is the merchant that bears the cost of fraud). Finally, online gateways are easier to set-up (no need for SSL or encryption software) and easier to use (since the transaction process is largely automated).

Many merchants do not realize that using an online authorization service does NOT mean that you have to bill cards immediately a transaction is made. Most of the systems provide the option of authorizing a card and then billing it later at a time of your choosing. This is useful when you need to only bill the customer after the goods are despatched, or if you wish to carry out other checks before accepting the order.

### 8.1 Setting Up WorldPay Select Junior

#### 8.1.1 About WorldPay

For further WorldPay information, see [www.worldpay.com](http://www.worldpay.com) and read WorldPay's documentation.

WorldPay is a major provider for online credit card transactions. By using the scripts provided, CactuShop can easily be configured to use WorldPay for payment. You will need WorldPay to set up an account for you on their system. The CactuShop software handles the entire ordering process and formats the customer and order details in a form for submission to WorldPay. The customer then inputs credit card details at WorldPay's secure site. Once the transaction has been completed, WorldPay informs your CactuShop, which updates its records to show the status of the transaction and sends order notification to you and the customer (if required).

One big advantage of using an external card payment system like WorldPay is that you do not need to implement SSL on your site since the card details are entered on WorldPay's system.

Some configuration to your account must be done at WorldPay's Web site (WorldPay will have supplied you with access details). This is referred to as WorldPay's Merchant Management System.

The WorldPay scripts are:

- *WORLDPAY\_mailorder.asp*
- *WORLDPAY\_callback.asp*

They are located on the root of the CactuShop web.

To switch CactuShop to use WorldPay, you must edit the `paymentsystem` config setting (see 7.4.14.4) to include 'worldpay' (lower case).

## 8.1.2 Config settings – WorldPay

### 8.1.2.1 WP\_AuthCode

This can be set to 'A' for a full authorization (card is authorized and billed immediately) or 'E' for a pre-authorization (the card is authorized but payment is not taken until you choose to bill the card from WorldPay's back end). Pre-authorization is used when you will need to manually check whether stock is available or the order passes extra fraud detection measures you may have in place, for example. To use pre-authorization, you will need to contact WorldPay and request that this facility is set up at their end.

### 8.1.2.2 WP\_CallbackPassword

This is an added security measure. Set a password or phrase here and then set up the corresponding phrase in the WorldPay merchant management system. WorldPay passes this back to CactuShop after the transaction has completed.

### 8.1.2.3 WP\_EmailCustomer

Whether you want CactuShop to email the customer when an order goes through. Alternatively you can get WorldPay to send the emails (see WorldPay's Merchant Management System).

### 8.1.2.4 WP\_EmailMerchant

Whether you want CactuShop to email you when an order goes through. Alternatively you can get WorldPay to send the emails (see WorldPay's Merchant Management System).

### 8.1.2.5 WP\_InstallID

The Install ID given to you by WorldPay

### 8.1.2.6 WP\_ProcessCurrency

If you wish all orders to be processed in one currency as opposed to the currency that the customer was viewing the site in, enter the ISO code for the currency here, e.g. GBP. If you leave this blank, the orders will be processed in the currency that the user selected to view the site in. Note that not all currencies you set up on CactuShop may be supported by this gateway.

### 8.1.2.7 WP\_ReturnURL

This is where you want the user to return after the transaction has completed (most likely your homepage). The purchaser will be given a hyperlink to return to this location. Do not confuse this with the callback set up at WorldPay's site (which will be the full URL to *WORLDPAY\_callback.asp*).

### 8.1.2.8 WP\_testmode

If this is set to 100 whilst you are testing, then all transactions will succeed. If set to 101 all transactions will fail. See WorldPay's documentation for more information.

#### 8.1.2.9 WP\_WorldPayURL

The default value is <https://select.worldpay.com/wcc/purchase>, the URL of WorldPay. It is very unlikely this will change. Don't touch it unless WorldPay tells you to.

### 8.1.3 Set up required at WorldPay side

Within WorldPay's Merchant Management system you must set up the following (in addition to standard setup info)...

#### 8.1.3.1 Callback password

This should be same as WP\_CallbackPassword

#### 8.1.3.2 Use callback enabled

Should be activated.

#### 8.1.3.3 Use callback response

Should be activated.

#### 8.1.3.4 The callback URL

This should be the path to your *WORLDPAY\_callback.asp* page, for example:

[http://www.mydomainnamegoeshere.com/WORLDPAY\\_callback.asp](http://www.mydomainnamegoeshere.com/WORLDPAY_callback.asp)

#### 8.1.3.5 Further customization

Please review WorldPay's documentation. WorldPay gives you the facility to customize the look and feel of their pages, specify minimum transaction values and a whole lot more.

### 8.1.4 Activation

Once your shop is working in testmode then you can complete Worldpay's activation form in order to turn your account live. Please note that when prompted for the name of your payment page this will be

[http://www.mydomainnamegoeshere.com/WORLDPAY\\_mailorder.asp](http://www.mydomainnamegoeshere.com/WORLDPAY_mailorder.asp)

Assuming that your shop is located in the root of your website.

## 8.2 Setting Up 2CheckOut

### 8.2.1 About 2CheckOut

For more information see [www.2checkout.com](http://www.2checkout.com) and read their documentation.

2CheckOut provides payment processing services. The 2CheckOut system is cleverly designed to use the same form field names as Authorize.Net and as such, CactuShop can be set up to run with 2CheckOut using the Authorize.Net config settings.

To switch CactuShop to use 2CheckOut, you must edit the `paymentsystem` config setting (see 7.4.14.4) to include 'authorizenet' (lower case). Note that because 2CheckOut support uses the Authorize.Net config settings, both systems cannot be run side-by-side on CactuShop.

The scripts used for 2CheckOut are:

- *AUTHORIZENET\_mailorder.asp*
- *AUTHORIZENET\_callback.asp*

They are located on the root of the CactuShop web.

## 8.2.2 Config settings – 2CheckOut

(using Authorize.Net config settings as explained above)

### 8.2.2.1 AUTHNET\_AuthorizeNetURL

The default value for 2CheckOut is:

<https://www.2checkout.com/cgi-bin/Abuyers/purchase.2c>

### 8.2.2.2 AUTHNET\_CallbackURL

This setting is not used for 2CheckOut. Leave it blank.

### 8.2.2.3 AUTHNET\_EmailCustomer

y/n - determines whether CactuShop will send an email to the customer for orders made using this gateway.

### 8.2.2.4 AUTHNET\_EmailMerchant

y/n - determines whether CactuShop will send an email to you for orders made using this gateway.

### 8.2.2.5 AUTHNET\_LoginID

Enter your 2Checkout.com Seller ID in this config setting.

### 8.2.2.6 AUTHNET\_ProcessCurrency

If you wish all orders to be processed in one currency as opposed to the currency that the customer was viewing the site in, enter the ISO code for the currency here, e.g. USD. If you leave this blank, the orders will be processed in the currency that the user selected to view the site in. Note that not all currencies you set up on CactuShop may be supported by this gateway.

### 8.2.2.7 AUTHNET\_ReturnURL

This setting is not used for 2CheckOut. Leave it blank. This setting for 2CheckOut is set from their back end system, see 8.2.3.2 for more information.

### 8.2.2.8 AUTHNET\_TestMode

This setting is not used for 2CheckOut. Leave it blank.

### 8.2.2.9 AUTHNET\_TxnKey

This setting is not used for 2CheckOut. Leave it blank.

### 8.2.3 Set up required at 2CheckOut side

Login to your 2CheckOut account on their web site. Check the following points:

#### 8.2.3.1 Call back set up

There is a dropdown menu with "Return to a routine on your site after credit card processed?" by it. Set this menu to 'yes'. This ensures that 2CheckOut will contact CactuShop when a payment is successful.

#### 8.2.3.2 Return URL

There is a box provided for 'Return URL'. This is the call back URL for the *AUTHORIZENET\_callback.asp* page on your web site. Enter the full path, to that file, for example:

[http://www.mydomainnamegoeshere.com/AUTHORIZENET\\_callback.asp](http://www.mydomainnamegoeshere.com/AUTHORIZENET_callback.asp)

## 8.3 Setting Up Authorize.Net

### 8.3.1 About Authorize.Net

For further Authorize.Net information, see <http://www.authorize.net/>

Authorize.Net is a major provider for online credit card transactions. By using the scripts provided, CactuShop can easily be configured to use Authorize.Net for payment. You will need Authorize.Net to set up an account for you on their system. CactuShop supports two different methods of using Authorize.Net to process credit card transactions.

Authorize.Net AIM (Advanced Integration Method) works by the CactuShop payment scripts calling the Authorize.Net server and receiving a response. The customer remains on the CactuShop scripts on your site at all times – it is the server that contacts Authorize.Net. This system requires your CactuShop site to have its own SSL – a shared secure area is not sufficient.

Authorize.Net SIM (Simple Integration Method) does not require any SSL certificate on your site. Instead, customers are routed off to a secure page on Authorize.Net's own site. On a successful order, a script on your CactuShop is notified and it flags the order accordingly as completed. There is limited scope for customizing the interface of the payment form since it is on Authorize.Net's server.

For more information, please ensure you review the documentation available from Authorize.Net's web site.

### 8.3.2 Setting up Authorize.Net Advanced Integration Method (AIM)

To switch CactuShop to use Authorize.Net AIM, you must edit the `paymentsystem` config setting (see 7.4.14.4) to include 'aim' (lower case).

The Authorize.Net AIM scripts are:

- *AIM\_mailorder.asp*
- *AIM\_success.asp*

These should be located on the root of the CactuShop web.

#### **8.3.2.1 AIM\_AuthorizeNetURL**

The default value is <https://secure.authorize.net/gateway/transact.dll>. It is very unlikely this will change. Don't touch it unless Authorize.Net tells you to.

#### **8.3.2.2 AIM\_EmailCustomer**

y/n - determines whether CactuShop will send an email to the customer for orders made using this gateway.

#### **8.3.2.3 AIM\_EmailMerchant**

y/n - determines whether CactuShop will send an email to you for orders made using this gateway.

#### **8.3.2.4 AIM\_LoginID**

The login ID given to you by Authorize.Net.

#### **8.3.2.5 AIM\_ProcessCurrency**

If you wish all orders to be processed in one currency as opposed to the currency that the customer was viewing the site in, enter the ISO code for the currency here, e.g. USD. If you leave this blank, the orders will be processed in the currency that the user selected to view the site in. Note that not all currencies you set up on CactuShop may be supported by this gateway.

#### **8.3.2.6 AIM\_ResponseDelimiter**

The response from Authorize.Net is delimited with a character. By default this is a comma (,). If you change this on the Authorize.Net backend, you must make the change here too.

#### **8.3.2.7 AIM\_TestMode**

If this is set to TRUE whilst you are testing then no billing of credit cards will take place. You should also set *testmode* within the Merchant Management system just to be on the safe side. When you are in testmode the page where you enter your credit card details will clearly state this at the top.

#### **8.3.2.8 AIM\_TxnKey**

This is a transaction key – you must generate this in your Authorize.Net back end and then paste the result here.

#### **8.3.2.9 AIM\_Type**

This is the transaction type. Alternatives are AUTH\_CAPTURE (default), AUTH\_ONLY, CAPTURE\_ONLY, CREDIT, VOID, PRIOR\_AUTH\_CAPTURE. See AIM documentation for more details. There is no need to change this for normal operation.

#### **8.3.2.10 AIM\_XMLComponent**

This is the XML component used to send the data to the Authorize.Net server. Ensure your web server has it installed. If *Msxml2.ServerXMLHTTP* doesn't work, try *Msxml2.ServerXMLHTTP.3.0* or *Msxml2.ServerXMLHTTP.4.0*.

### 8.3.3 Setting up Authorize.Net Simple Integration Method (SIM)

For further Authorize.Net information, see <http://www.authorize.net/>

To switch CactuShop to use Authorize.Net, you must edit the `paymentsystem` config setting (see 7.4.14.4) to include 'authorizenet' (lower case).

The Authorize.Net scripts are:

- *AUTHORIZENET\_mailorder.asp*
- *AUTHORIZENET\_callback.asp*

These should be located on the root of the CactuShop web.

- *AUTHORIZENET\_md5.asp*
- *AUTHORIZENET\_simlib.asp*

These should be in the *includes* folder of CactuShop.

### 8.3.4 Config Settings - Authorize.Net SIM

#### 8.3.4.1 AUTHNET\_AuthorizeNetURL

The default value is <https://secure.authorize.net/gateway/transact.dll>. It is very unlikely this will change. Don't touch it unless Authorize.Net tells you to.

#### 8.3.4.2 AUTHNET\_CallbackURL

This is the full URL to *AUTHORIZENET\_callback.asp*. Remember, Authorize.Net will call this script from their server – it won't work if you set the name of a script that is not on a live web server as Authorize.Net won't have access to it!

#### 8.3.4.3 AUTHNET\_EmailCustomer

y/n - determines whether CactuShop will send an email to the customer for orders made using this gateway.

#### 8.3.4.4 AUTHNET\_EmailMerchant

y/n - determines whether CactuShop will send an email to you for orders made using this gateway.

#### 8.3.4.5 AUTHNET\_LoginID

The Login ID given to you by Authorize.Net

#### 8.3.4.6 AUTHNET\_ProcessCurrency

If you wish all orders to be processed in one currency as opposed to the currency that the customer was viewing the site in, enter the ISO code for the currency here, e.g. USD. If you leave this blank, the orders will be processed in the currency that the user selected to view the site in. Note that not all currencies you set up on CactuShop may be supported by this gateway.

#### 8.3.4.7 AUTHNET\_ReturnURL

This is where you want the user to return after the transaction is completed (most likely your homepage). The purchaser will be given a hyperlink to return to this location. Use a full URL beginning with `http`.

#### 8.3.4.8 AUTHNET\_TestMode

If this is set to TRUE while you are testing then no billing of credit cards will take place. You should also set *testmode* within the Merchant Management system just to be on the safe side. When you are in test mode, the page where you enter your credit card details will clearly state this at the top.

#### 8.3.4.9 AUTHNET\_TxnKey

This is a transaction key – you must generate this in your Authorize.Net back end and then paste the result here.

### 8.3.5 Set up required at Authorize.Net side

Some configuration to your account must be done at Authorize.Net's Web site (Authorize.Net will have supplied you with access details). This is referred to as Authorize.Net's Merchant Management System. Login at:

<https://secure.authorize.net>

and then click the settings option.

#### 8.3.5.1 Callback setup

On the manage URLs page, add the URL of your callback page (same as AUTHORIZENET\_callback.asp) and tick to make it the ADC Relay Response Default and Valid ADC or Receipt Link.

#### 8.3.5.2 Test mode

On *General Settings*, tick Test Mode. Remember to untick this when you want to go live.

## 8.4 Setting Up PayPal

### 8.4.1 About PayPal

For further PayPal information, see <http://www.paypal.com>.

To switch CactuShop to use PayPal, you must edit the `paymentsystem` config setting (see 7.4.14.4) to include 'paypal' (lower case).

The PayPal scripts are:

- *PAYPAL\_mailorder.asp*
- *PAYPAL\_callback.asp*

These should be located on the root of the CactuShop web.

### 8.4.2 Config settings - PayPal

#### 8.4.2.1 PP\_Business

Your paypal account name (the email address you use to login to your paypal account).

**8.4.2.2 PP\_Cmd**

Paypal command instruction. The default value is “\_xclick” (without the quotes). This is very unlikely to change.

**8.4.2.3 PP\_EmailCustomer**

y/n - determines whether CactuShop will send an email to the customer for orders made using this gateway.

**8.4.2.4 PP\_EmailMerchant**

y/n - determines whether CactuShop will send an email to you for orders made using this gateway.

**8.4.2.5 PP\_NotifyURL**

This is the URL of the PayPal callback on your CactuShop. It should be something like:

[http://www.mydomainnamegoeshere.com/PAYPAL\\_callback.asp](http://www.mydomainnamegoeshere.com/PAYPAL_callback.asp)

However, if you are only using your PayPal account for CactuShop transactions then you can enter the notification URL on your PayPal configuration (My Account > Profile > Instant Payment Notification Preferences > Instant Payment Notification (IPN) URL).

**PP\_PayPalURL**

This is the URL of the payment page on the PayPal site. The default is:

<https://www.paypal.com/cgi-bin/webscr>

This is unlikely to change.

**8.4.2.6 PP\_ProcessCurrency**

If you wish all orders to be processed in one currency as opposed to the currency that the customer was viewing the site in, enter the ISO code for the currency here, e.g. USD. If you leave this blank, the orders will be processed in the currency that the user selected to view the site in. Note that not all currencies you set up on CactuShop may be supported by this gateway.

**8.4.2.7 PP\_TestMode**

y/n - determines whether the system treats transactions as real ones or test ones. Most other gateways have a test mode where the gateways authorizes transactions just as it would if live, but is actually just simulating real orders. Cards are not actually billed, nor funds reserved. Since PayPal has no such facility, this CactuShop setting ensures that transactions are actually passed, but that the value of the order is changed to 1p (or Cent, Yen, etc.) at the point when it is sent to the gateway. Each test will therefore only cost a very small amount, which is better than having to test with full price orders. Remember that any confirmations from PayPal will show 1p rather than the actual order value!

**8.4.2.8 PP\_ValidateCommand**

The security validation command sent back to PayPal from the callback script. There is no reason to change this.

#### 8.4.2.9 PP\_XMLComponent

The XML component used to send data to PayPal's server. Set this according to what component you have installed on your server. Try the default value of "Microsoft.XMLHTTP" first.

### 8.4.3 Set up required at PayPal side

Make sure that you have Instant Payment Notification (IPN) turned on in your PayPal account settings (My Account > Profile > Instant Payment Notification Preferences > Instant Payment Notification (IPN)) before integrating your CactuShop with PayPal.

## 8.5 Setting Up Protx VSP Form

### 8.5.1 About Protx

For further Protx information see <http://www.protx.com>.

To switch CactuShop to use Protx, you must edit the `paymentsystem` config setting (see 7.4.14.4) to include 'protx' (lower case).

The Authorize.Net scripts are:

- *PROTX\_mailorder.asp*
- *PROTX\_callback.asp*

These should be located on the root of the CactuShop web.

### 8.5.2 Config settings – Protx

#### 8.5.2.1 PX\_CallbackURL

This is the URL of the Protx call back page on CactuShop. It should be:

[http://www.mydomainnamegoeshere.com/PROTX\\_callback.asp](http://www.mydomainnamegoeshere.com/PROTX_callback.asp)

#### 8.5.2.2 PX\_EmailCustomer

y/n - determines whether CactuShop will send an email to the customer for orders made using this gateway.

#### 8.5.2.3 PX\_EmailMerchant

y/n - determines whether CactuShop will send an email to you for orders made using this gateway.

#### 8.5.2.4 PX\_OrderDescription

Description of the transaction.

#### 8.5.2.5 PX\_Password

Your Protx account password.

#### 8.5.2.6 PX\_ProcessCurrency

If you wish all orders to be processed in one currency as opposed to the currency that the customer was viewing the site in, enter the ISO code for the currency here, e.g. GBP. If you leave this blank, the orders will be processed in the currency that the user selected to view the site in. Note that not all currencies you set up on CactuShop may be supported by this gateway.

#### 8.5.2.7 PX\_SubmitURL

The URL of the payment page on the Protx site. This is unlikely to change.

#### 8.5.2.8 PX\_VendorEmail

The email address that Protx will use to send any mails regarding the transaction to you.

#### 8.5.2.9 PX\_VendorID

Your Protx vendor ID.

### 8.5.3 Protx back end

Protx will give you account details to enable you to login and admin orders from their end.

## 8.6 Setting Up SecPay

### 8.6.1 About SecPay

For further SecPay information see <http://www.secpay.com>.

To switch CactuShop to use SecPay, you must edit the `paymentsystem` config setting (see 7.4.14.4) to include 'secpay'.

#### 8.6.1.1 SECPAY\_CallbackPassword

This is the SecPay Digest Key. On the SecPay side, you can set up a 'digest key'. This is then checked on callback. It can be omitted, but is a useful security check against spoofing.

#### 8.6.1.2 SECPAY\_CallbackURL

The location of your `SECPAY_Callback.asp` page. Don't forget it should start with 'http'!

#### 8.6.1.3 SECPAY\_EmailCustomer

y/n - determines whether CactuShop sends an email confirmation to the customer for orders through this gateway.

#### 8.6.1.4 SECPAY\_EmailMerchant

y/n - determines whether CactuShop sends an email confirmation to you for orders through this gateway.

#### 8.6.1.5 SECPAY\_Merchant

Your SecPay merchant ID.

#### 8.6.1.6 SECPAY\_Options

Comma-separated list of extra options. Use "test\_status=true", which puts you in test mode (just remove it to do live payment). You can also specify a deferred payment (deferred=full), and configure SecPay email settings. See SecPay docs for details.

#### 8.6.1.7 SECPAY\_ProcessCurrency

If you wish all orders to be processed in one currency as opposed to the currency that the customer was viewing the site in, enter the ISO code for the currency here, e.g. GBP. If you leave this blank, the orders will be processed in the currency that the user selected to view the site in. Note that not all currencies you set up on CactuShop may be supported by this gateway.

#### 8.6.1.8 SECPAY\_ReturnURL

URL of the page on your CactuShop that SecPay will return a user to after payment.

#### 8.6.1.9 SECPAY\_SubmitURL

This is the URL at SecPay that users are forwarded to to make a payment. This is very unlikely to change – don't edit it unless SecPay tell you to.

#### 8.6.1.10 SECPAY\_Template

The location of your SECPAY template that you can upload to the secpay site through the backend. This is normally:

[http://www.secpay.com/users/\[merchantname\]/\[nameoffile\].html](http://www.secpay.com/users/[merchantname]/[nameoffile].html)

Leave blank to use the SecPay default template.

## 8.7 Setting Up Secure Trading

### 8.7.1 About Secure Trading

For further Secure Trading information see <http://www.securetrading.com>.

To switch CactuShop to use Secure Trading, you must edit the `paymentsystem` config setting (see 7.4.14.4) to include 'st'.

The Secure Trading scripts are:

- *ST\_mailorder.asp*
- *ST\_callback.asp*

These should be located on the root of the CactuShop web.

#### 8.7.1.1 ST\_CallBackID

This refers to which callback ID to use from the numbered entries in `callback.txt` (see */SecureTrading Files/callback.txt*). Can leave this as 1 unless you are setting up multiple callbacks on one account.

#### 8.7.1.2 ST\_EmailCustomer

y/n - determines whether CactuShop sends an email confirmation to the customer for orders through this gateway.

### 8.7.1.3 ST\_EmailMerchant

y/n - determines whether CactuShop sends an email confirmation to you for orders through this gateway.

### 8.7.1.4 ST\_Password

This is not the password Secure Trading gave you, but an arbitrary one you choose to add security to the callback. See 8.7.2.1.

### 8.7.1.5 ST\_ProcessCurrency

If you wish all orders to be processed in one currency as opposed to the currency that the customer was viewing the site in, enter the ISO code for the currency here, e.g. GBP. If you leave this blank, the orders will be processed in the currency that the user selected to view the site in. Note that not all currencies you set up on CactuShop may be supported by this gateway.

### 8.7.1.6 ST\_RequiredFields

The fields at Secure Trading that must be filled in before the payment is accepted.

### 8.7.1.7 ST\_SiteReference

Your secure trading site reference ID.

### 8.7.1.8 ST\_URL

URL of the secure payment page at Secure Trading site - very unlikely this will change.

## 8.7.2 Secure Trading back end

Secure Trading will give you account details to enable you to login and admin things from their end. Unlike the other systems, Secure Trading requires that you upload templates for the forms and confirmation mails. While this allows for excellent customizability of the interface, it also makes setup more complex than with most other gateways.

This manual gives a brief overview of how the Secure Trading back end can be used to set up the system to work with CactuShop. For further information, please review the Secure Trading documentation.

### 8.7.2.1 Configuration files

*callback.txt* - Specifies to ST which file to callback. Modify the *url1* line to put in the URL of the ST\_callback.asp page, then *?pw=passwordhere* (where *passwordhere* is your password matching the *ST\_Password* config setting). Make sure there is a tab between the *url1* and your URL.

So the full URL might look something like:

[http://www.mydomainnamegoeshere.com/ST\\_callback.asp?pw=mychosenpassword](http://www.mydomainnamegoeshere.com/ST_callback.asp?pw=mychosenpassword)

*merchantemail.txt* - The email that gets sent to the merchant on a successful order. No need to edit this.

*failureemail.txt* - The email that gets sent to the merchant on a failed order. No need to edit this.

### 8.7.2.2 Display files

*form.html* - File displayed to show the payment form.

*failure.html* - File displayed when a failed payment occurs.

*success.html* - File displayed when a successful payment occurs. You can modify these 3 pages to change the look and feel of the pages in line with your skin design.

Various images can be uploaded for the site design. You can link to these images by referring to them using a file reference such as *\$path/image.jpg*.

#### 8.7.2.3 Test mode

Secure Trading has no manually configurable test mode. You will need to email Secure Trading to have them switch from test mode to live mode.

## 8.8 VeriSign

### 8.8.1 About Verisign

CactuShop supports two VeriSign implementations, VeriSign Payflow Link and VeriSign Payflow Pro.

Payflow Link uses a secure area on VeriSign's own site – the customer is transferred from CactuShop to the secure area to make payment and a 'callback' is made from the VeriSign server to CactuShop to let it know if the transaction was successful. This solution is cheaper than Payflow Pro since you don't need your own secure certificate and can run your site on a shared (virtual) server. Originally, this was the only VeriSign method supported by CactuShop, hence config settings for it are prefixed with 'VS' and the `paymentsystem` config setting is 'verisign'.

Payflow Pro, as the name suggests, is a heavier-weight solution. It requires SSL on your CactuShop site. It also requires that a VeriSign COM object be installed on your server. For this reason, it is generally only suitable where you have your own dedicated server, or a very obliging web host. On Payflow Pro, the customer makes the order on CactuShop's secure form (rather than one hosted by VeriSign). CactuShop uses the COM object to call VeriSign, authorize the card and get a response back. Since the 'VS' config setting prefix and `paymentsystem` name of 'verisign' was used for Payflow Link, the Payflow Pro settings are prefixed with 'PFP' and use the `paymentsystem` name of 'payflowpro'.

For more information on the differences between the two methods, please see VeriSign's web site and documentation.

### 8.8.2 Setting Up VeriSign Payflow Link

For further VeriSign Payflow Link information, see:

<http://www.verisign.com/products/payflow/link/>

To switch CactuShop to use VeriSign Payflow Link, you must edit the `paymentsystem` config setting (see 7.4.14.4) to include 'verisign'.

The VeriSign Payflow Link scripts are:

- *VERISIGN\_mailorder.asp*
- *VERISIGN\_callback.asp*

These should be located on the root of the CactuShop web.

#### **8.8.2.1 VS\_EmailCustomer**

y/n - determines whether CactuShop will send an email to the customer for orders made using this gateway.

#### **8.8.2.2 VS\_EmailMerchant**

y/n - determines whether CactuShop will send an email to you for orders made using this gateway.

#### **8.8.2.3 VS\_Login**

This is your login name for VeriSign – this information should be supplied to you by VeriSign when you set up your account.

#### **8.8.2.4 VS\_Partner**

This is your partner for VeriSign. If you sign up through the VeriSign site directly this is generally 'VeriSign'.

#### **8.8.2.5 VS\_Password**

This is a callback password (not your login password!) of your own choosing. This should match the password set in the querystring within the VeriSign 'silent post URL'. Leave it blank to disable (not recommended).

#### **8.8.2.6 VS\_ProcessCurrency**

At the time of writing, VeriSign payment processing accounts can only process one currency. For this reason, you must ensure that the ISO code of the currency you have set up your VeriSign account to process is entered in this config setting, for example 'USD'.

#### **8.8.2.7 VS\_Type**

This is VeriSign's 'type' attribute. It can be set to 'S' for sale (where transactions are authorized and billed immediately) or 'A' for authorization (where the card is authorized but not billed until you decide to accept the payment). The latter method is useful if you need to check stock levels or perform extra fraud checks before despatching the order and accepting the payment.

#### **8.8.2.8 VS\_VeriSignURL**

This is the URL of the secure payment form on VeriSign's web site. The default value is <https://payflowlink.verisign.com/payflowlink.cfm>. It is very unlikely this will change. Do not alter this unless VeriSign tells you to.

#### **8.8.2.9 VS\_Password**

This is a special callback password - not your verisign login password! - that can be any made up password you choose. This should match the password set in the querystring within VeriSign 'silent post URL'. When the callback is made, the password in the querystring is checked with this password to ensure it matches. You can leave this blank to disable the security check, but this isn't recommended as it ensures that, as long as this password is unknown, users can't spoof a callback. See 8.8.3.3 for more details.

### 8.8.3 Set up required at VeriSign side

You need to configure the PayFlow Link details. Once you've logged into your management area (URL and login details will be emailed to you from VeriSign after signup), click on Account Info at the top then PayFlow Link Info on the left.

#### 8.8.3.1 Return URL Method

This should be set to LINK so the user is returned to the shop successfully.

#### 8.8.3.2 Return URL

This will provide the link at VeriSign to return the users to the store once payment has been completed. Usually just the home, e.g.

<http://www.mydomainnamegoeshere.com/shop/default.asp>

#### 8.8.3.3 Silent Post URL

The checkbox needs to be checked to enable the silent post URL (callback URL). Next to the checkbox, there's space to input the callback URL. This should be the location of the VERISIGN\_callback.asp page and then *?pw=passwordhere* (where *passwordhere* is your password matching the VS\_Password config setting). So the full URL may look like - e.g.

[http://mydomainnamegoeshere.com/shop/VERISIGN\\_callback.asp?pw=pass](http://mydomainnamegoeshere.com/shop/VERISIGN_callback.asp?pw=pass)

#### 8.8.3.4 Transaction Process Mode

Set this to TEST whilst configuring the gateway. When you're ready to go live, set this to LIVE.

The above 4 are all you need to setup to get the site working. The other options configure email sends and the design of the payment pages. Once the gateway is working correctly, play around with these options to tweak the system.

### 8.8.4 Setting Up VeriSign Payflow Pro

For further VeriSign Payflow Pro information, see:

<http://www.verisign.com/products/payflow/pro/>

The VeriSign Payflow Pro scripts are:

- *PPF\_mailorder.asp*
- *PPF\_callback.asp*

These should be located on the root of the CactuShop web.

#### 8.8.4.1 PFP\_EmailCustomer

y/n - determines whether CactuShop sends an email confirmation to customer for orders through this gateway.

#### 8.8.4.2 PFP\_EmailMerchant

y/n - determines whether CactuShop sends an email confirmation to you for orders through this gateway.

#### 8.8.4.3 PFP\_HostAddress

This is the VeriSign host address. Use `test-payflow.verisign.com` for testing, `payflow.verisign.com` for live.

**8.8.4.4 PFP\_HostPort**

This is the VeriSign host port number. Default is 443 - very unlikely this will change.

**8.8.4.5 PFP\_Partner**

Your partner for VeriSign – it will be 'VeriSign' unless you signed up through a third party.

**8.8.4.6 PFP\_ProcessCurrency**

At the time of writing, VeriSign payment processing accounts can only process single currencies. For this reason, you should ensure that the ISO code of the currency you have set up your VeriSign account to process is entered in this config setting, for example 'USD'.

**8.8.4.7 PFP\_ProxyAddress**

Proxy server address – enter a value only if your server is behind a firewall/proxy server.

**8.8.4.8 PFP\_ProxyLogin**

Proxy server login ID – enter a value only if your server is behind a firewall/proxy server.

**8.8.4.9 PFP\_ProxyPassword**

Proxy server password – enter a value only if your server is behind a firewall/proxy server.

**8.8.4.10 PFP\_ProxyPort**

Proxy server port number – enter a value only if your server is behind a firewall/proxy server.

**8.8.4.11 PFP\_Pwd**

This is the password that you designated for the VeriSign account when you set it up.

**8.8.4.12 PFP\_Tender**

VeriSign 'tender' type (method of payment). Use 'C' for credit card transactions (shouldn't need to change this).

**8.8.4.13 PFP\_Timeout**

Time-out period for PayFlow Pro transaction, in seconds. The minimum (and default) is 30. Shouldn't need to change.

**8.8.4.14 PFP\_Type**

This is VeriSign's 'type' attribute. It can be set to 'S' for sale (where transactions are authorized and billed immediately) or 'A' for authorization (where the card is authorized but not billed until you decide to accept the payment). The latter method is useful if you need to check stock levels or perform extra fraud checks before despatching the order and accepting the payment.

**8.8.4.15 PFP\_User**

This is your login username for VeriSign.

**8.8.4.16 PFP\_Vendor**

This is the vendor ID (usually the same as `PFP_User`, your login)



### 8.10.3 Config Settings

The config settings must be edited in the CactuShop backend, under Site Settings, the subsection entitled "Encryption".

#### 8.10.3.1 encrypt

Set this to *y*

#### 8.10.3.2 encryptionkeyfile

Set this to the name of your key (see 8.11.3)

**Please note:** CactuShop needs to create a temporary file when using this method. This is the encrypted file which is attached to the email to the merchant. Once the email is sent, the temporary file is deleted. The file is created in the html folder (within your specified "uploads" folder), as this should have the correct permissions. If you get a permission denied error, ensure that the IUSR\_ guest account has full permissions for this folder.

## 8.11 CactusPay

If you want to send secure email, we have a CactuShop add-on that includes script-based single & triple DES encryption. This is not a part of the standard CactuShop product - it is an add-on. For pricing, check our web site.

CactusPay is a payment gateway specifically designed for sites where the secure area is on a different machine or in a different location to the rest of CactuShop. There are thus 2 sides to setting up this payment option... configuration of CactuShop itself, and then installation/configuration of CactusPay within the secure SSL web space.

Using CactusPay, orders can be sent to you securely, and you can then print them out and process as you would a phone or fax order.

### 8.11.1 CactusPay - CactuShop Configuration

The config settings must be edited in the CactuShop backend, under Payment and Shipping Gateways

#### 8.11.1.1 CP\_CactusPayURL

Full web address of the CactusPay secure order form - probably in the form <https://www.mydomainnamegoeshere.com/orderform.asp>

#### 8.11.1.2 CP\_CallbackPassword

Password used by the secure area when calling back the order confirmation page on the main web site.

#### 8.11.1.3 CP\_ProcessCurrency

ISO code of the currency your transactions will be processed in. Leave blank to use the currency as passed from CactuShop.

### 8.11.2 CactusPay – Secure Area Configuration

Unzip the zip file containing the encryption software. Extract the contents of the folder named "SSL\_on\_Different\_Server". Install these scripts into your secure area. The path to the orderform.asp file should match the

*CP\_CactusPayURL* you have specified within the CactusPay settings of CactuShop (see 8.11.1).

You will now have to edit the *cactuspay\_config.asp* file within the secure area.

#### 8.11.2.1 emailmethod

As specified as for Cactushop itself, see 7.4.6.1 for options

#### 8.11.2.2 sendorderstoaddress

The address where orders are emailed

*mailfromaddress* – the address why emailed orders appear sent from

#### 8.11.2.3 emailsubjectline

Subject line of order emails

#### 8.11.2.4 mailserver

The IP address or domain or the mailserver you will be sending mail out via (not required for CDO). Ask the provider of your secure webspace for details

#### 8.11.2.5 callback\_password

This should match what you specified as the *CP\_CallbackPassword* in the CactuShop end of the setup (see 8.11.1)

#### 8.11.2.6 callback1

The location of the script *cactuspay\_callback1.asp* within your CactuShop.

#### 8.11.2.7 callback2

The location of the script *cactuspay\_callback1.asp* within your CactuShop.

*home* – a link to the homepage of your CactuShop (for customer to return to after order is complete)

#### 8.11.2.8 EncryptionKey

see 8.11.3 for key generation

#### 8.11.2.9 LocalKeyFileName

The name of the file storing the key on the machine of the user who will receive encrypted orders. Usually a *.cek* file extension is used.

Other settings are documented in the *cactuspay\_config* file. These should rarely require changing.

### 8.11.3 Generating a key

DES, the encryption algorithm used in CEM normally uses a 64 bit key which is reduced to 56 bits by excluding parity bits. 56 bits is effectively 56 binary 1s or 0s. Each extra bit doubles the number of possible keys so a 56 bit key would take twice as long to guess as a 55 bit key.

CEM also supports triple DES, an algorithm which repeats DES 3 times with different keys giving a key size of 168 bits – therefore taking billions of times longer to guess.

CactuShop represents DES keys in Hex code. Hex is base 16, so to count from zero to 15 in hex is 0123456789ABCDEF. 16 is therefore represented by 10.

Each character of Hex represents 4 bits or binary items. So a 56 bit key can be represented by 16 hex characters and a triple DES key by 48.

CactuShop can automatically tell the difference between a 56 and 168 bit key. Here are two example keys:

Single DES (56 bit)

```
0123456789ABCDEFxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

Triple DES (168 bit)

```
0123456789ABCDEF0123456789ABCDEF0123456789ABCDEF
```

You can see that single DES keys fill up the unused key space with "x" characters.

DES encryption is very complex and therefore slow especially when implemented in VBscript form for the ASP pages. Triple DES may take 10-15 seconds to encrypt even on a fast server. Therefore, you may prefer to use single DES – it is certainly strong enough to make cracking it for credit card numbers uneconomic (breaking 56 bit DES still takes months using large amounts of expensive computing power).

#### 8.11.4 Setting up the decryption software

Once steps 8.11.3 are complete, putting through a test order should result in an email being sent to the shop owner's email address. This will contain an attachment called something like *order342.ce2* – this is the encrypted order.

To install the decryption software, copy all the files provided to a directory on the machine you will use to receive the orders by email. In Windows Explorer go to Tools > Folder Options > File Types > New and add *ce2* as the new extension. Then associate this file type with the *.exe*.

Now ensure that you create a key of the same name you set previously, e.g. *yourkey.cek* and fill the content of this key with the hex string you formed in 8.11.3. You can do this either by running the 'key manager' from the CEM decryption software or creating the *.cek* file in notepad and saving in the same folder as your decryption software.

Now you should be able to open the attachment from your email package and see it load in decrypted form in the CEM decryption software.

## 8.12 Secure Back End Review – Requires SSL

One further secure method of viewing orders is to set CactuShop to store order text in the database and send you an email notification of new orders. The credit card details can be viewed by logging into your back end and viewing orders through SSL online.

To implement this method:

- Change the config setting under Miscellaneous `saveInDB` to 'y' (see 7.4.3.1)
- Ensure your secure certificate on the site is set up properly

- Ensure you connect to the back end with *https* as opposed to *http* (for example, [https://www.mycactushopsite.com/\\_default.asp](https://www.mycactushopsite.com/_default.asp))

Since you are now storing credit card details in the database, database security becomes more important than ever. As always, we would advise that an Access database is in a folder without read permissions that would allow a visitor to download it through a browser.

Again, we would stress that we would always recommend using a payment gateway system in preference to any system that involves you or the store handling actual credit card numbers. This way, the part where security is most critical is handled by experts with secure servers, the physical and electronic access to which is strictly controlled.

### 8.13 Setting Up SSL

In order to accept information such as card details securely through your site you will need to implement Secure Sockets Layer (SSL). SSL uses public key cryptography to encrypt a link between a browser and server, securing the information transferred across that link. Cactusoft cannot offer support with setting up SSL (you should contact your ISP); the information here is provided purely as an overview of the process and technology.

**Please note:** SSL is not required on your site if you are using a payment gateway such as WorldPay. In this case, credit card details are entered at the payment gateways site where SSL will already be installed and running.

#### 8.13.1 Obtaining a secure certificate

To run SSL, you will need a secure certificate. Contact your web host to arrange this as they will also need to install this for you. If you are testing locally, you can generate a free certificate yourself with this excellent demo on the Persits software web site:

[http://support.persits.com/encrypt/demo\\_sslcert.asp](http://support.persits.com/encrypt/demo_sslcert.asp)

These certificates work, but are not issued by a trusted authority like VeriSign or Thawte (who require paperwork and research to verify your identity). Therefore, browsers will not automatically trust them and a security warning will be flagged. They are therefore not appropriate for real applications.

#### 8.13.2 SSL issues

Running pages through SSL creates some issues for CactuShop. SSL pages can't read cookies placed by non secure-pages and you can't pass parameters via the URL ("GET"). Fortunately, CactuShop passes parameters via POST for the last few pages of the order process. You should run the cart in non-secure web space, but pass to SSL for the payment side. To get CactuShop to pass through to a secure URL for the *payonline.asp* and *mailorder.asp* pages, ensure the config setting `usesecurearea` is set to 'y' (see 7.4.11.9).

### **8.13.3 SSL access to the database**

Some ISPs will set up SSL for your domain on a separate physical machine that the one which holds your CactuShop installation. Please note that the SSL pages must have access to the database in exactly the same way as the non-secure pages. Please check with your ISP that your database will be accessible from the secure pages (i.e. the secure pages are on the same machine as the rest of the site, or the database can be accessed from the SSL enabled pages in another way). If the secure area is on a different machine and cannot gain access to the database, we have a system called CactusPay to handle this – this is also supplied with the Cactusoft Encrypted Mail system. It works in a similar way to payment gateways.

## **8.14 Other Payment Systems**

We are working on adding built-in support for other major payment methods too. These will be supplied free for CactuShop full version customers as and when they are available.

## 9 Language Strings System

### 9.1 Background

In v4.5 of CactuShop, the principle of 'language strings' was introduced. Instead of page text being hardcoded into ASP scripts, each piece of English was replaced by a reference to a string stored in the database. The required strings for each page were pulled out at the top of the page and then referred to by name throughout the pages. This system made it easier for non-English versions of the cart to be maintained, as a script for a localized version of the cart would not need to be retranslated if a bug-fixed script replaced it.

### 9.2 Multi-Language Capability

In v5 of CactuShop, the capability to support multiple languages is introduced. This is largely a development of the language string system previously used, so those who have translated previous versions of the cart should be able to use much of that translation in v5. The main change that has been made is that the single *tblCactuShopLanguageStrings* table that featured in v4.x has been superseded by 5 tables, named *tblCactuShopLanguageStrings1*, *tblCactuShopLanguageStrings2* and so on. Each of these tables contains the full set of language strings, but coded in different languages.

In order to complete the multi-language capability in CactuShop, extra fields have been added to products and other database items to take the foreign language values. Separate skins are also required for each language.

### 9.3 The Language String Editor

Language strings can be edited online from the back end of CactuShop. It is simple to search for strings and replace them. This way, you don't need to take the database down each time you want to change wording. Furthermore, the search capabilities in the online editor make it easier to find and replace strings of text than by opening up the database itself. You can use the search to find language strings by entering any of the following:

- Name of the item, or part of the name, e.g. *ContentText\_Thankyou*
- Any part of the text that is in the language string, for example, a search for 'shopping basket' will bring up all language strings that contain that phrase.

You can also add new language strings. This feature is useful if you upgrade and need to add just a handful. Unlike most tables, the language strings do not auto ID – you get to choose the ID (although it will pick the next consecutive one by default). If you add your own custom language strings to CactuShop, we'd suggest numbering them from 5000 (for example), just to avoid any future clashes with CactuShop ones if you upgrade your store.

## 9.4 The Language Strings Table

The text strings used throughout CactuShop are stored in the *tblCactuShopLanguageStrings(+n)* tables. Each record contains a number of features:

### 9.4.1 LS\_ID

The language string ID is a unique identifier used by the language string functionality to pull the language tag from the database. It is a primary key but is not set to auto-number. This makes it easier when importing new language strings in the future.

### 9.4.2 LS\_FrontBack

This is a single text digit to show whether this language tag record is used on the front (f) or back (b) end of CactuShop. It can be useful in sorting the records and allows the back end tags to be translated separately from the front-end tags if required. This is useful for users who only wish to translate the front end to their language and aren't too bothered with doing the back end since the translation is just for their customers' benefit.

### 9.4.3 LS\_Name

The name of the string is used within the ASP pages to pull the value from the scripting dictionary. The names are broadly representative of what the tag represents in English.

### 9.4.4 LS\_Value

This is the actually display text that will appear on the page. It is this text that should be translated and changed for foreign-language versions.

### 9.4.5 LS\_Description

This is a description/explanation of this particular language tag. This text is not displayed or used by CactuShop – it is purely guidance for translators and developers working with CactuShop's language strings.

### 9.4.6 LS\_VersionAdded

This is the CactuShop version number that this string was added in (starting from v5.000). This value should make it easy to see which language strings you need when you upgrade a store later.

## 9.5 The Language Strings Functionality

### 9.5.1 PageStrings

At the top of most pages, there is a constant called *PageStrings* set. This string of numbers is used by the *LoadStrings* function in the *config.asp* file to pull out the language tags required for this particular page. Each number is the *LS\_ID* value of the required string. There are also some strings that are

needed on every page (for the minibasket, etc.) specified within the language strings code of *config.asp*.

### 9.5.2 LoadStrings

This function is called from the *config.asp* file and is located in the *functions.asp* file in the *includes* folder. It formats a query to pull out the required language tags on each page by *LS\_ID* value. It creates a scripting-dictionary of all the values and their respective tag names (*LS\_Name*). This makes the pages easier to understand since language tags can be referenced by name rather than a less intuitive ID number.

### 9.5.3 WriteString and GetString functions

These functions pull the actual display text value from the scripting dictionary. *GetString* pulls the value, *WriteString* pulls the value AND writes it to the page. Therefore, areas where a string of text is built up but not written to the page (email text for example) should use *GetString*, while you can shortcut having to *response.write* the *GetString* value by using *WriteString*.

## 9.6 Adding New Languages

### 9.6.1 Overview

CactuShop ships with support for 5 languages, however only an English translation is provided. Check our website for details of any additional language we might make available. This leaves room for 4 additional language that can be added. If you wish, you can remove the English and add 5 new languages. Below is some outline advice for undertaking the addition of a new language. This assumes you are adding only one additional language, but the advice should be appropriate for those adding more additional languages.

### 9.6.2 Add a New Skin

You will probably want to create a new skin for your new language so that the menu and other items in the skin can be translated. Copy your existing skin, and paste a copy at the same level in the folder structure as the existing skin. Its useful to name it *skin\_[yourlanguage]* for example *skin\_spanish* (or *skin\_espanol*). This skin can be translated in the same way you would normally edit a skin.

### 9.6.3 Activating a Second Language

Go to list the languages in the backend and switch on language 2 (on either the front, back or both). Specify the skin as the one you created in 9.6.2 above. You can use your existing email addresses, or you might decide to create new ones for this new language (so that other staff with different language skills can handle customers in this language).

#### 9.6.4 Language Strings

You will probably find it easier to translate language strings if you use the English ones as a base. These are in the table *tblCactuShopLanguageStrings1*. Copy and paste this table as *tblCactuShopLanguageStrings2*. You can now work through each entry in this table translating as you go, either by opening it in Access/SQL Server, exporting to Excel or using CactuShop's built in LanguageString editor.

#### 9.6.5 Other Software Elements to Translate

These additional elements also need translation

##### 9.6.5.1 Shipping Rates

Each shipping rate has a name and a description in each language you have activated. Once you have turned on a new language, you should have a corresponding pair of boxes for each shipping rate to add the translated name and description.

##### 9.6.5.2 Destinations

Countries are held in the *tblCactuShopDestination* table. This has appropriate columns for translations of country names. For example, for language2 you would add the translations in the *D\_Destination2* column.

#### 9.6.6 Selecting a Default Language

You can specify which of your languages is the default by entering the number of that language in the *defaultlanguageid* config setting.

#### 9.6.7 Product Data

Once the additional language has been activated (see 9.6.3 above) each category, sub category, product and version should have appropriate fields to enter translations. If you leave out a translation then that item will not appear on the front of the site when the additional language is selected. In this way, 2 different language versions of a site may include different items for sale.

#### 9.6.8 News Stories

In the backend you can switch between editing the various different language versions of news stories that are featured on the front of the website.

#### 9.6.9 Mailing List

When users sign up for the mailing list, CactuShop notes what language version of the site they signed up on. From the mail sending system, it is possible to select which newsletter members to send any messages to, so the English version can be sent to those subscribing whilst English was selected, and other language version can be sent to appropriate Newsletter list members.

## 10 Customizing CactuShop

### 10.1 Skins

Skins are a feature uniquely implemented in CactuShop. You will want to customize the appearance of your store to some degree, even if this is simply a question of changing the header logo, etc. CactuShop's skinning system gives you the capability to do this but also so much more.

On most web sites, there are page features that occur on every page such as menus, header, footer, etc. The standard way most ASP coders accomplish this is to have include files for each of these elements. However, there are drawbacks with this way of working. Firstly, it is difficult for HTML and graphic designers to work with this format, since they're editing several pieces of HTML that then need to fit together to make the final design. Furthermore, include files still have to be positioned somewhere on a page. If the menu include file is positioned on each page to the left hand side, no amount of editing will get it on the right hand side. You'd have to go through the whole site and recode the HTML of every front-end page to move features around like this.

CactuShop improves on this with its 'skinning' system. The page is formatted dynamically based on a 'skin' file, a single HTML document (and associated stylesheet and images) that contains the basic page layout and design, plus a few custom tags.

By adding a new skin or modifying the existing one, you can totally transform the appearance of the entire front-end. Fonts, colours, alignment, arrangement of page elements – all of these can be changed by modifying the skin.

#### 10.1.1 How does CactuShop create pages based on the template?

Each front-end page references an include file called *buildpage.asp*. Code within this script reads the template file into memory. It then replaces the custom tags representing the currency menu, category menu, language menu and minibasket with HTML for each that it generates dynamically. The total HTML that results is then split into two sections around the xxxPAGECONTENTxxx tag. The first half of this HTML is written before all other page content on front-end pages, and the second half is written at the end of each page.

#### 10.1.2 What elements make a skin?

A skin is a selection of files placed in a folder that sits on the root of the web. The template HTML document itself is always called *template.htm* and the images folder is called *templateimages*. The template also contains images for the basket icons and any header images.

#### 10.1.3 What essential features must a 'skin' have?

- A style sheet file (see 10.1.4)

- *xxxPAGECONTENTxxx* tag in order for page content to appear
- the main menu must be included to give users a way to navigate
- a *<head>* tag (the basket message refresh system requires this – it MUST be lower case)
- JavaScript pop-up function (see default skin)

#### 10.1.4 The style sheet reference

In CactuShop skins, each template folder should contain another folder called *templateimages* where the skin-specific images reside. In addition, the style sheet file (*stylesheet.css*) is also located here. Some styles defined in this are purely for use within the template and as such, you may have other templates with different styles. However, all skins should include the following styles that are used by the CactuShop pages.

<code>.smallnormal</code>	small sized text
<code>.smallnormaltable</code>	small sized text on tables
<code>.smallnormaltablealt</code>	small sized text on tables, alternative colour
<code>.mednormal</code>	medium sized text
<code>.mednormaltable</code>	medium sized text on tables
<code>.medbold</code>	medium size text, bold
<code>.medboldalt</code>	medium size text, bold, alternative colour
<code>.medboldtable</code>	medium size text, bold, on tables
<code>.largebold</code>	large bold text
<code>.smallcompact</code>	very small text
<code>.smallbold</code>	small size text, bold
<code>.smallboldtable</code>	small size text, bold, on tables
<code>.smallboldtablealt</code>	small size text, bold, on tables, alternative colour
<code>.smallboldtabletop</code>	small size text, bold, on table headings
<code>.locationbar</code>	text used on the location bar
<code>.searchtexthighlight</code>	highlighted text in search results
<code>A:link</code>	general hyperlinks
<code>A:visited</code>	general hyperlinks, visited
<code>A:hover</code>	general hyperlinks, hover
<code>A.minibasket:link</code>	minibasket hyperlinks
<code>A.minibasket:visited</code>	minibasket hyperlinks, visited
<code>A.minibasket:hover</code>	minibasket hyperlinks, hover
<code>A.prodtypemenu:link</code>	category menu hyperlinks
<code>A.prodtypemenu:visited</code>	category menu hyperlinks, visited
<code>A.prodtypemenu:hover</code>	category menu hyperlinks, hover
<code>A.table:link</code>	table hyperlinks
<code>A.table:visited</code>	table hyperlinks, visited
<code>A.table:hover</code>	table hyperlinks, hover
<code>A.locationlink:link</code>	location bar hyperlinks
<code>A.locationlink:visited</code>	location bar hyperlinks, visited
<code>A.locationlink:hover</code>	location bar hyperlinks, hover
<code>table.minibasket</code>	minibasket table
<code>tr.minibasketline</code>	minibasket line colouring
<code>tr.minibasketline2</code>	minibasket alternative line colouring
<code>table.mainpage</code>	tables on pages
<code>tr.mainpagetableline</code>	line colour in tables

<code>tr.mainpagetableline2</code>	line colour in tables, alternative
<code>tr.mainpagetablespecial</code>	line colour in tables, special
<code>tr.mainpageversionhighlight</code>	line colour in tables, highlighted items
<code>td.shortformatproduct</code>	short format product and category cells
<code>hr</code>	horizontal rule
<code>pre</code>	specifications text
<code>select</code>	dropdown menus
<code>form</code>	form tag margin (spacing)
<code>.buttonstyle</code>	push buttons
<code>.textboxsmall</code>	text boxes, small

### 10.1.5 How is the template used to format my pages (tech stuff)

An include file called *buildpage.asp* features on each front end page. This reads the skin template file's HTML, builds the basket summary HTML, builds the product category listing HTML and then replaces the *xxxBASKETSUMMARYxxx*, *xxxCATEGORYMENUxxx*, *xxxCURRENCYMENUxxx* and *xxxLANGUAGEMENUxxx* tags with the appropriate HTML. Finally, the string of HTML is split around the *xxxPAGECONTENTxxx* tag, with either half being stored in a box of a 2-dimensional array called *aryPageTemplate*. Each page contains just the code that is necessary to generate the content part of the page – the *buildpage.asp* include builds the rest of the HTML and the two halves stored in the *aryPageTemplate* array are written to the page above and below the main page content.

## 10.2 Customizing the Functionality of CactuShop

Being entirely script-based, CactuShop can be modified to do almost anything. However, there are a few things to consider. Firstly, we cannot offer technical support for customer modifications, as this would be an open-ended commitment. If you choose to start modifying scripts and the database structure to suit your own

## 11 CactuShop Data Tool

The CactuShop Data Tool is a powerful software application free for CactuShop customers that can copy data from various sources to the new CactuShop v5.0 and v5.1 database format. You can import data from an older version of CactuShop (Access, MySQL or SQL Server), an Excel spreadsheet, a TXT/CSV file or from another shopping cart (VPASP, Actinic and Comersus supported).

This can be used for upgrading to the newest version, upsizing between databases, backing up live sites and importing products from other sources.

You can also execute an SQL file against a local or remote SQL Server or MySQL database, which can be used to create the structure and default data of a new CactuShop database.

### 11.1 Installation

The software can be downloaded free from the CactuShop site. To install the software, run the .MSI file in the zip. If this doesn't work, you need to download the Microsoft Installer

(<http://www.microsoft.com/msdownload/platformsdk/instmsi.htm>).

Once installed, you should have a link in your start menu. Click on this to get going.

### 11.2 Before You Start

Before you start using the Data Tool on your database, ensure you take an entire backup of your data. It is also advisable to back up all the files on your web site so that you can restore everything if there are ever any problems.

### 11.3 Using the Software

The software works in a 'wizard' style to take you through the steps of copying the data. There are two modes – Copy Data (used for upsizing or downsizing a database, backing up your data, uploading data to a live site or importing your catalogue), and Execute SQL (which is used solely for executing an SQL file against a SQL Server).

In Copy Data mode, you need to select the source database (an Access file, SQL Server or MySQL database, Excel spreadsheet or text file), and then select your destination database (Access, SQL Server or MySQL).

The source database (from where you are copying data) must match strictly the formats supported by the Data Tool. In the case of prior CactuShop versions, it must be an unmodified database. In the case of an Excel spreadsheet or text file, it must match our sample template exactly, field for field. For an Actinic, VPASP or Comersus database, it must be the most recent, unmodified version.

The destination database must be set up manually – the software cannot create the database and structure. For Access, use a copy of the original Access database that came with the software, and for SQL Server/MySQL download the latest SQL script from our web site.

In Execute SQL mode, you need to enter in the details of the SQL Server or MySQL database, and select the .SQL file. See 11.3.6 for more information.

### 11.3.1 Using the software to upgrade your CactuShop

The Data Tool can upgrade a v4.5, v4.6, v4.7 and v5.0 to either v5.0 or v5.1, on Access, SQL Server or MySQL databases. If you have a cart before v4.5 you wish to upgrade, you need to do this manually. Please contact us for further details.

Once you've selected your source and destination database, the software will attempt to connect to and recognise the version of the database. Please ensure this is correct before proceeding. Once this is confirmed, the Data Tool will proceed to go through each table in the source database. It will delete the records in the corresponding destination table and copy the new data across. With a large database, this process may take a while. The progress bars will keep you informed of how the upgrade is progressing.

If an error has occurred during the process, it will let you know the details. The most common errors are incorrect structure in one of the databases. This could be because the source database was customized previously, or because the destination database is not the latest version.

You can customize the importing process by clicking on *Customize Procedure (advanced)* button on the *Confirm* state. Here you can chose which tables to copy across. As the button suggests, this is for advanced use only, and we recommend you leave this alone unless you know what you are doing. It can be useful, however, to skip over non-vital tables that are giving errors or tables with data you don't want to copy.

### 11.3.2 Upsizing (or downsizing) your database

The Data Tool can also be move your data from one database type to another – this is useful for upsizing your Access database from to SQL Server/MySQL. You can do this while upgrading from an older database, or from the current one. For example, you can upgrade from a v4.5 Access database to a v5.1 SQL database, from a v5.1 Access database to a v5.1 MySQL database etc.

### 11.3.3 Backing up your data

If you are running a remote SQL Server or MySQL database and you don't have access to it through Enterprise Manager, you might not have a way of backing up your data easily. You can connect to the database with Access 2000, which allows you to make changes to the tables and data. However, backup tools are only included in SQL Server Enterprise Manager.

One way around this is to use the Data Tool to create Access backups of online SQL Server databases. Simply select your live database as the source database, and a local database as the destination, and the Data Tool will copy

all the information to the safety of your local machine. You can run this periodically to keep an up-to-date back up of your data.

#### 11.3.4 Uploading your data to a live site

Conversely, you may find it useful to move your local test copy of your database to the live site – e.g. when you're going live with your site for the first time, and have your shop set up locally on SQL Server. Simply reverse the process and Data Tool will do the job for you.

Please note that connecting to a remote database slows down the copying process. It is also obviously reliant on the speed of your Internet connection and live server. It may be wise to run back-ups during the night or whenever server load is lowest, as the copying is quite intensive and will put some overhead on the database. For Access databases, it'll probably be quicker just to upload it through FTP.

#### 11.3.5 Importing from spreadsheet or text file

The Data Tool can help import your catalogue information from an Excel spreadsheet or a text/CSV file. This is a rapid way to stock a new store without having to manually add product info.

However, importing does still require some preparation to get your data in the correct format for CactuShop to read from. This will normally involve some manipulation of the data in Excel.

Most packages will allow you to export the data to some form of CSV (comma-separated values) file. If you have Excel, we strongly advise you to work from that (you can load up your CSV files in Excel). It will make it much easier when it comes to moving around data.

The import feature currently only copies to these tables:

*tblCactuShopProdType*  
*tblCactuShopProducts*  
*tblCactuShopVersions*  
*tblCactuShopTaxRates*  
*tblCactuShopProductProdTypeLink*

There is currently no support for importing other tables other than these five (e.g. previous orders, customer details, etc.).

The Data Tool can also copy thumbnail and large images for categories, products and versions, renaming them and copying them to the standard uploads directory required by CactuShop. For this to work, you must have a folder called 'uploads' with the various images folders within it as the destination folder.

##### 11.3.5.1 Limitations of importing from Excel / text file

It is important to note that the limitations to importing the data;

- The format of the data is very strict, and you must manipulate your data manually to get it to the correct format. Details of the format are given in 11.2.4.2.

- Products also must have unique names. These are used as unique identifiers. If the Data Tool comes across the same product name in a different category, it will presume it is the same product and make it exist in both categories. Hence, there is no way to have different products with the same name. (Different categories, however, can have the same name. See 11.2.4.2 for more information).
- There is no support for multiple languages (all names, descriptions etc. are defaulted to be language 1).

#### 11.3.5.2 The format of the data

The format for the data is very strict: the order of the data columns must be exact. Each column in the Excel spreadsheet (or text file) represents the field, and each row represents a version, and the product/category hierarchy it lives in. The Data Tool runs through each row, adding new categories, products, and versions as it goes, and linking them up correctly.

For an example of the data, see the 'Example Data' directory. Included is an example CSV, Excel and txt file. You may wish to take a good look at this, and perhaps run the Data Tool on it to understand how the data is formulated.

The following is a list of each row, in order. Many of the fields are numbered (e.g. Category Name 1) because CactuShop supports up to five languages as standard. If your store only has one language, you only need to fill the '1' fields. Note however that each column must exist, even though many of the columns can be left blank;

- Category Name 1
- Category Description 1
- Category Image 1
- Category Large Image 1
- Category Name 2
- Category Description 2
- Category Image 2
- Category Large Image 2
- Category Name 3
- Category Description 3
- Category Image 3
- Category Large Image 3
- Category Name 4
- Category Description 4
- Category Image 4
- Category Large Image 4
- Category Name 5
- Category Description 5
- Category Image 5
- Category Large Image 5
- Product Name
- Product Description
- Product Image
- Product Large Image
- Product Hyperlink
- Product Specifications
- Product Strap line
- Version Name
- Version Description

- Version Image
- Version Large Image
- Version Code Number
- Version Price
- Version Quantity
- Version Weight
- Version RRP
- Tax Rate

The fields Category Name 1, Product Name, Version Code Number, Version Name and Version Price are required **on every single line** for the import to work. For the other fields, they need to exist on the first occurrence e.g. to add a category description, it must occur next to the first appearance of the new Category Name. Similarly, to add product details, they must appear alongside the first occurrence of the product. Further occurrences of the same category and product are ignored, so data can be omitted (i.e. you don't have to put category and product information next to every occurrence of a product and category, only the first one).

A description is required for each category in CactuShop. The Data Tool will work without them, but the CactuShop backend won't like it when you try to edit the data. However all other string fields can be left blank throughout.

Make sure all fields are valid – specifically that the price, tax, weight and RRP fields are numeric (no pound or percentage signs) and all the version quantity is an integer. If you don't want to specify these number fields, make sure you set them to 0, not blank.

As noted in the limitations, the Product Name is used as a unique identifier. You cannot have different products under the same name. There is no way the software would know the difference between different products, or just the same product in a different category, so it presumes the latter.

However, different categories can have the same name. Because a category can have only one 'parent' category, if a category exists in a different parent then the software knows that it is a different category (even if it has the same name). This allows you to have sections "Cars > Blue > " and "Vans > Blue > "; the Data Tool can identify the difference.

For the versions, the Code Number is used as the unique identifier (this is a prerequisite of CactuShop as well as the Data Tool), so make sure this field is different for every row or expect funky results. Due to this, version names can be repeated yet still be identified as different.

Five categories are included in the example above. This allows you a hierarchy of five categories: The Category 1 is the base category, Category 2 is a child, Category 3 is a child of Category 2 etc. If you don't use five categories (which will be the case in all but the biggest of databases), simply leave the fields blank – only Category Name 1 is required for each row.

It is presumed that the first row of your Excel spreadsheet, or the first line of your text file, contains the column headers. These don't have to be named anything specific, but they must exist.

### 11.3.5.3 Images

The Data Tool can also optionally rename and copy images for categories, products and versions. Both normal and large images are supported.

As with all optional fields, they can be left blank. They also only need to be present on the first occurrence of the new category, product or version.

The field should contain, if required – the directory – and the full name (including extension) of the image from the base image folder location.

For example, lets say your images are stored in a folder called 'shopimages'. Product images are stored in a subfolder called 'productimgs', and categories in 'catimages'. Category "cars" has an image called "cars.gif"; product "golf GTI" has image "golf\_thumb.gif" and large image "golf\_large.jpg". On the first occurrence of the 'Cars' category, field 'Category Image 1' should contain "catimages/cars.gif". With 'Golf GTI' the 'Product Image' field should be set to 'productimgs/golf\_thumb.gif' and 'Product Image Large' to 'productimgs/golf\_large.jpg'.

Once you have your data set correctly, you need to specifically turn on copying of images (see section 11.3.5.7).

### 11.3.5.4 Text files and the schema.INI file

There are many different ways of having a text file – comma-, character-, or tab-separated, single- or double-quotes for strings, different characters for rows etc. Thankfully, thanks to the brilliance of the ODBC Schema.ini file, the Data Tool will support just about any text file format.

The Schema.ini resides in the same directory as your .csv/.txt file, and tells the database driver how it is formed. If you have a standard (and by standard, this means the same as the defaults set in your operating system) format for your CSV file, you don't need a schema.ini file. (This is usually comma-separated with double quotes around text strings only when commas appear in the string, but regional settings may dictate otherwise.). However, for non-standard files, you can specify the details in your Schema.ini file.

It is beyond the scope of the manual to go over the format of this file. See [http://msdn.microsoft.com/library/en-us/odbc/htm/odbcjetsdk\\_98.asp](http://msdn.microsoft.com/library/en-us/odbc/htm/odbcjetsdk_98.asp) for more details, or search for "Schema.ini" on the Microsoft MSDN site.

### 11.3.5.5 Excel files

Thanks to the standardised Excel file, there should be fewer problems running the Data Tool on it. One problem that may occur is if the Excel file has a 'Data-Link' to the old table, or includes other oddities or embedded stuff. If you come across any weird problems, try copying the entire table (CTRL-A, CTRL-C) and pasting it in to a new spreadsheet file. This will get rid of anything except the raw data.

### 11.3.5.6 Running your import

Once you think you have the data in the correct form, you can get ready to run it through the Data Tool. Go through the steps outlined in 11.2.1, selecting your XLS, CSV or TXT field as the source.

For the Excel file, you can optionally enter in which worksheet the data appears on. If you leave this blank, the Data Tool will use the first.

If there are any errors in the import, the Data Tool will be sure to let you know.

#### 11.3.5.7 Running your import with images

If you are copying images, you must ensure you have it set up correctly in your source database (see 11.3.5.3). You now need to turn on image copying. On step 3 – ‘Confirm Details’, click on ‘Settings and Images’. Check the Copy Images checkbox. You now need to enter the source and destination folders for the images. The ‘Source Image Directory’ field needs to be set to the base folder of your images. In the example outlined in 11.3.5.3, this should point to the ‘shopimages’ folder. The ‘Destination’ field should point to the CactuShop Uploads folder, with all the subdirectories already created (images\_categories, images\_categories\_large etc).

Run the import as normal. When the Data Tool comes to a new category, product or version, it checks the Image and Image Large fields. If either of these aren’t blank, it’ll try and get the image. If it finds it, it will rename it to the format CactuShop needs (the ID of the record) and copy it to the correct subfolder in the selected Uploads folder.

Note that any problems on trying to get or copy this image – such as the file not existing, an incorrect or protected source or destination folder etc. – and the import is likely to fail.

#### 11.3.6 Importing your catalogue from another shopping cart

If you want to move from another shopping cart to CactuShop, then the Data Tool may be able to help you move your catalogue over so you don’t need to enter in all your data again.

CactuShop currently supports the latest VPASP, Actinic and Comersus databases. If you are using other software, look at exporting your data to a CSV file (many shopping carts have this feature built-in) and manipulating this to import to CactuShop.

If you are using VPASP, Actinic or Comersus, the Data Tool will attempt to copy your catalogue over to the latest CactuShop database.

##### 11.3.6.1 Limitations of importing from another software package

It is important to note that there are limitations on the importing from other shopping carts:

- The importer focuses on importing categories and products. Depending on the support of the other software package, it may also import images, downloadable products, weight, tax, related products etc. Other data is ignored by the data tool. See the specific sections for the database for details on what exactly is imported.
- The data that is imported will not be precise. This will be due to the differences in the software. All attempts are made to replicate the data in the closest form. However, the result may not be the exact.
- CactuShop only supports the latest version of VPASP and Actinic. If you are running an older version, you will need to upgrade this first before moving to CactuShop.

- The import routines from other software should be consider 'beta'. They have only been tested on a limited sample databases. If you encounter errors when you try to import the database, please contact us and we'll try to remedy the problem. However there is no guarantee that you will get the data in to CactuShop.

It should be generally considered that the aim of importing the data from other software packages is to cut down the ammount of data entry required to move to CactuShop. Unless you have a simple catalogue layout, it is unlikely that the data will import exactly the first time. Probably, you will then have to go in to CactuShop and tweak data – such as some prices, stock etc.

#### 11.3.6.2 Consideration of CactuShop versions in comparison to other products

It was mentioned in the limitations that the data imported would not be exact due to the differences in the products. One of the main differences is CactuShop uses product 'versions', while other software uses 'options' or 'features'.

The Data Tool will make a good attempt to convert options over to the versions sytem. A product with no options will be represented as a product with one base version. A product with one option (say colour, with choices 'blue', 'red' and 'green') will be represented as three versions. A product with two options (say colour, as above, plus 'size', with choices 'Medium' and 'Large') will be represented as six versions, one for each permutation ('Blue Medium', 'Blue Large', 'Red Medium' etc).

The Data Tool also deals with other oddities, such as 'optional' options, linked products as options etc.

Depending on your product range, it will vary how well the data is represented. If your products generally have none, one or two options to them, they will be well suited to the CactuShop 'versions' system and should import well. However, if your products have a vast choice selection then this may not translate well to the versions system.

CactuShop versions have their individual prices, weights and stock count, whereas, with software built around options, these features are usually set around the product level. This information is directly copied to each version of the product.

Note that to complete the conversion you will need to tweak the configuration settings in CactuShop, which are not set by the import. For example, the data tool may import custom ordering on categories. However, this needs to be switched on with the config setting 'orderby\_categories'. Changing various display settings will get your site to look more like your original store.

It is advised that you just try out the import and see how well it works.

#### 11.3.6.3 Running the import

To import from another software package, you follow similar steps to importing from an excel/CSV file. Set the source database to the database of an existing shopping cart (this can be an Access, SQL Server or MySQL database) and set the destination to a new CactuShop database (again this can be any database type).

Note that the source database must be from the latest version of the software and unmodified.

The software will then detect the software and proceed to import the data. If you wish to also copy the images from your old catalogue, click on 'Settings and Images'. Here you can set the source directory (where the images are currently stored in your shop – you may need to download these from the live site) and the destination directory (the 'uploads' folder of CactuShop).

#### 11.3.6.4 Importing from VPASP

The Data Tool supports VPASP v5. Previous versions are likely to fail.

The import supports:

- Categories: Name, description, image and 'live' status are moved to CactuShop. Category hierarchy will be also kept.
- Products: Name, description and description URL will be copied. The 'special offer' field will be moved to the 'strapline' field. Quantity, weight and retail price will be copied to the products versions. Small and large images will be moved.
- Product features – these will be converted in to versions. Pricing modifications will be supported. Name will be based on the options of that permutation.
- Related products.
- Tax.

The import does not support:

- Product features that require user text or price entry.
- 'Multi-select' product features.

#### 11.3.6.5 Importing from Actinic

The Data Tool supports Actinic version 6.1. It has been tested on Actinic Developer 6.1.4.0.0.0, but should work with minor updates (6.1.x) and with Actinic Business and Catalog. Other versions are likely to fail.

The import supports:

- Categories: name, description, image and 'live' status (based on the 'show on website' checkbox) are imported. The category hierarchy will be kept, as will the category order.
- Products: name and description and copied (from 'small description' and 'large description'). Images (normal and 'extended text' image) are copied to CactuShop's normal and large images. Stock, weight and 'digital download' are copied to all of the product's versions.
- Versions: These are created based on each permutation possible. The data tool supports multiple attributes, attributes through components, and multiple linked products and specific permutations. It makes a good effort at dealing with optional attributes and different pricing structures. However, import accuracy is depending on the complexity of the database. Version name is based on the name of the combined choices. Version display is set dynamically based on the number of versions and the pricing.
- Tax: Simple tax is imported in to CactuShop. If you are using only 1 (or no) tax rates then it will be correctly copied to cactushop. If your actinic products are subject to 2 or more tax rates, then these are added together and created as one tax rate. Depending on how you have exception set up this may or may not correlate correctly to CactuShop. If some of your products override the tax % and set a

hard-coded currency value for tax, this is added on to all the version prices.

The import does not support:

- Non-live or semi-deleted products, attributes, components, choices, etc.
- Stock tracking on 'additional' linked products.
- Custom fields.
- Custom pricing schedules.
- Advanced tax.

#### 11.3.6.6 Importing from Comersus

The Data Tool supports all versions Comersus v5. In comparison to the other carts the layout and simplicity of the database makes the general accuracy of importing the data much higher. The result should be a pretty close match to how your data is in Comersus.

Specifically, the import supports:

- Categories: the name, 'live' status, ordering and image are copied over. Note that a description field is required by CactuShop but isn't represented in Comersus: The import sets it to the same as the name.
- Products: name, description, small and large image, 'live' status, and 'show on front page' are copied across.
- Versions: These are created as permutations of option groups and their options. Price and weight is added up for each permutation based on the product with additions on the options. RRP, code number, delivery time, free shipping, stock quantity and stock warn level are copied for each version of the product.
- Product bundles are copied across – one version represents the bundle, and then one version for each part of the bundle.
- Related products are imported.

A few things are not supported by the data tool:

- Tax.
- Language translations (only the base language is used).
- Downloadable (electronic) products.

#### 11.3.7 Executing an SQL file

With this action, you can execute any valid .SQL against your local or remote SQL Server or MySQL database.

The main use for this is to execute the .SQL file downloaded from the CactuShop site that contains the structure and data of a CactuShop database. This is especially useful for those who don't have enterprise manager or a similar SQL Server admin tool.

Simply download the zip file from the CactuShop site and extract the .sql file. Run the Data Tool and select the Execute SQL option. You will then be prompted to enter in your database details and select the .sql file. Once you've set these options and the data connection has been checked, the Data Tool will run through the .sql file and execute the code against the SQL Server. Depending on the speed of your server and your connection, this may take a few minutes, so please be patient.

## 11.4 Data Tool Disclaimer

Please remember that the Data Tool is a free piece of software. We do offer limited support for CactuShop customers to get it working successfully, but there is no guarantee that it will work correctly. The importing from Excel and CSV files will only work under the strictest of rules, and if your data cannot be manipulated to this format, you won't get it working.

We suggest that you try it out on CactuShop Lite. The database format is the same as the full version so if you get the import working successfully, you can use the same database with the full version.

## 12 Support

### 12.1 Scope of Tech Support

Cactusoft will offer tech support to customers through documentation available from our web site, the knowledgebase and email. We make no guarantees to support the free lite version though if requests are posted through the knowledgebase they may be addressed.

We do not offer tech support on modifications to the cart that were not made by us. This would be an open-ended commitment that we could not possibly fulfil. Please understand that if you decide to (for example) add support for a payment gateway that we don't include as standard, add extra fields to the checkout procedure or create a fantastic new way to display products, we cannot and will not debug or complete the scripts for free under tech support.

### 12.2 How to Request Support

We would ask all customers to follow through the tech support channels in the following order:

- a. **Read this manual, especially the FAQ** - 80% of the problems notified to us are covered in detail already.
- b. **Search the knowledgebase** – this online resource on our web site contains questions that previous users have asked, with our answers added. A few keywords should find any issues already listed if they have been previously encountered.
- c. **Ask a question on the knowledgebase** – we give priority to questions added to the knowledge base. The answers will be entered online and will be available to future users who may come across the same issue. We do not publish the email addresses of those posting or add them to any mailing list – they are purely used for our admin staff to contact the user for more details if required.

While we do accept telephone support calls from customers, note that URLs and error messages are extremely difficult to deal with on the phone. Such information is far easier to deal with if sent to us by email.

### 12.3 FAQ

In addition to this FAQ, most commonly encountered problems are already dealt with on our searchable online knowledgebase. Please see:

<http://www.cactushop.com/support/>

#### 12.3.1 What does the error 'must use updateable query' mean?

This is the most common problem reported by those trying to set up CactuShop. Some pages work OK, even databased ones, but others seem to generate this error. Please refer to section 3.3 for details on how to set up the database with the correct permissions.

### 12.3.2 I get a 500 error. What does this mean?

Internet Explorer has a setting to make error messages look more 'friendly'. However, this is generally unhelpful, as it seems to reduce the amount of error information. To get clearer error messages, try this in IE: Tools > Internet Options, then click 'Advanced' tab. Uncheck the box for 'Show friendly HTTP error messages' and hit 'ok'. Once you have these clearer error messages, try searching for the error (or part of it) in our knowledgebase – most previously-encountered problems are there.

### 12.3.3 What is the back end login and password?

The default back end login and password on the full version are both 'cactushop'. For information on changing these values, see section 6.1.

### 12.3.4 Will the software run on Linux?

Basically, no. ASP can be run on Linux/Unix using third party software (see [stryon.com](http://stryon.com) or [chilisoft.com](http://chilisoft.com)). But CactuShop is designed, tested and supported only on Microsoft platforms and all our experience is that it will not run on Linux/Unix platforms. If you fancy proving us wrong, please test with the lite version.

### 12.3.5 Will the software run a Chinese, Japanese or Korean web site?

We advise customers to use Unicode as this supports all three languages including both simplified and traditional Chinese. Some modifications to CactuShop are required. We would advise that you test the lite version first to ensure it works correctly on your server platform. For more information, search our knowledgebase for 'unicode'.

### 12.3.6 What is your upgrade policy?

See section 4.2.

### 12.3.7 Can I use a visual editor?

The 'skins' template feature is designed to make visually designing a site far less problematic. The visual design and functional elements are separated meaning your programmers can modify code while your designers work on the appearance. The template is designed with no hidden code sections or other features that might be distorted or misread by visual editors. However, we would advise against editing the other pages in a visual editor, even newer versions of FrontPage and Dreamweaver that claim not to modify code. A good text editor such as the one with MS Visual InterDev is best. If you are stuck with Notepad, try using the excellent free 'Metapad':

<http://www.liquidninja.com/metapad/>

This is as small and quick as Notepad but has essential 'code monkey' features like line numbering and multiple undo.

### **12.3.8 How does the licensing work?**

The first copy of CactuShop you purchase is £200. Subsequent copies are £100. Even if you bought a previous version, you are still entitled to the 'subsequent license' pricing. For developers using large quantities of licenses (more than 20 per year), please contact us for discounted rates. The license number must be visible in the HTML code that the *buildpage.asp* include generates.

### **12.3.9 How much is it in US Dollars?**

Our order form online has a link to a currency converter. To find the exact rate in dollars you will be charged, contact your credit card company.