

Caddy - the Shopping Cart

Caddy is the TYPO3 shopping cart. See <http://typo3-caddy.de>. You can link your own database with products or other items to caddy. If you like a ready-to-use example, please install Caddy with one mouse click by the Quick Shop Installer.



Version: 3.0.0, 2013-12-28

Extension Key: caddy 3

Language: en

Keywords: forDevelopers, forIntegrators, caddy, cart, shopping, wt_cart, e-payment

Copyright 2013, Dirk Wildt, Die Netzmacher, <<http://wildt.at.die-netzmacher.de>>

This document is published under the Open Content License available from <http://www.opencontent.org/opl.shtml>
The content of this document is related to TYPO3 - a GNU/GPL CMS/Framework available from www.typo3.org

Table of Contents

Caddy - the Shopping Cart	1	HTML COLORS	28
Screenshots	4	HTML WIDTH	28
Frontend	4	PAGES	29
Order	4	PDF	29
Confirm	5	PDF CADDY	29
Confirmation E-Mail	5	PDF DATE	29
Attachments	6	PDF DELIVERY ORDER	30
Colour Schemes	7	PDF INVOICE	30
Backend	8	PDF ORDER	30
Plugin	8	PDF REVOCATION	31
Order Records	9	PDF TERMS	31
Installation	9	PDF TEXTCOLOR	31
Introduction	10	SYMBOLS AND SERVICE ATTRIBUTES	31
What does Caddy do?	10	TAX	32
What doesn't Caddy do?	11	TEMPLATES	32
What do you need?	11	Corporate Design	33
Target Group	11	Colours	33
Installation	12	Screenshots	33
Quick Installation	12	Select a colour scheme	33
One-Click-Installation (recommended)	12	Send your scheme to the developer!	33
Advantages	12	Icons	34
Work flow for your own needs	12	Controlling	34
Organiser Installer	13	Ready-to-use icons	35
Quick Shop Installer	13	E-Mail	35
Manually Installation (not documented)	13	HTML Template	35
Plugins	15	Attachments	36
Caddy	15	Corporate Design	36
Frontend	15	Your own draft	36
Backend	15	Properties	38
Controlling	15	Work-flow	42
Mini Caddy	16	Add, update and delete an item	42
Controlling	16	Add an item	42
TypoScript	16	Update an item	42
Sum Gross	17	Delete an item	42
Controlling	17	Calculating	42
TypoScript	17	Ordering	42
Controlling Caddy	18	Variants	44
Overview	18	TypoScript Snippet	44
Plugin / Flexform	19	Archive audit proof	45
Check it!	19	Database is read-only	45
Notes	19	Disable audit proof	45
Caddy	20	Possible reasons	45
Email	21	Extension Manager	45
Billing Address	22	Reset numbers	46
Delivery Address	23	Get all keys	46
PDF	23	Reset (remove the keys)	46
Sponsors	24	All keys	46
Help!	24	Keys of a single Caddy plugin	46
Static Templates	25	Reference	47
Syntax – Order of the Templates	25	Constant Editor	47
Main (obligated)	25	Setup	47
CSS	25	tx_caddy_pi1 – Overview	47
Localization	25	tx_caddy_pi1.conditions	47
Powermail (obligated)	25	tx_caddy_pi1.db	48
Constant Editor	26	tx_caddy_pi1.db.sql	48
MAIN	26	tx_caddy_pi1.debug	48
MAIN NUMBERFORMAT	26	tx_caddy_pi1.getpost	49
DATABASE	27	tx_caddy_pi1.marker.items	49
DEBUG	27	tx_caddy_pi1.marker.sum	50
GETPOST	27	tx_caddy_pi1.options	51

tx_caddy_pi1.pdf	52	From Powermail 1.x to Powermail 2.x	67
tx_caddy_pi1.pid	52	From wt_cart 1.x to Caddy 2.x	67
tx_caddy_pi1.pluginCheck	53	Packages for Sponsors	68
tx_caddy_pi1.settings	53	Fun – 50 EUR	68
tx_caddy_pi1.symbols	53	Logo – 150 EUR	68
tx_caddy_pi1.templates	53	Patron – 500 EUR	68
tx_caddy_pi1.tax	54	Credits	69
tx_caddy_pi1._HTMLMARKER	55	Made in Nature	69
tx_caddy_pi1._CSS_DEFAULT_STYLE	55	wt_cart Developer Team	69
tx_caddy_pi1.powermail	56	Known Bugs	70
tx_caddy_pi2	57	TCPDF-Error	70
tx_caddy_pi3 – Overview	58	Prompt	70
tx_caddy_pi3.content.item	59	Reason and Solution	70
tx_caddy_pi3.content.sum	60	More Bugs	70
tx_caddy_pi3.pid	61	Further Information	71
tx_caddy_pi3.templates	61	Forum	71
tx_caddy_pi3._HTMLMARKER.linktocaddy	62	TYPO3 forge	71
tx_caddy_pi3._HTMLMARKER.linktoshop	63	History of Caddy	71
Debugging	64	Fork of wt_cart	71
DRS – Development Reporting System	64	Caddy versus wt_cart	72
Inspect work-flow, session data and errors	64	Other extensions published by Die Netzmacher	73
Enabling by the Extension Manager	64	Alphabetical Index	74
Enabling by the Plugin / Flexform (recommended)	65	Illustration Index	75
Enabling by the Constant Editor (recommended)	65	Change Log	76
The DRS report	66		
Updates	67		

Screenshots

Frontend

Sorry, most screens are in German language.

Order

TYPO3 Caddy
Der TYPO3-Warenkorb für Deine Datenbank.

Warenkorb

Menge	Art.-Nr.	Artikel	MWSt.	Preis	Gesamt	Entfernen
3	typo3-book-de	Geheime Bilder	7%	29,99 €	89,97 €	X
# Summe Artikel (netto)					84,08 €	
+ Service- und Versandkosten (netto)					0,00 €	
+ Mehrwertsteuer					5,89 €	
zuzüglich 19% MWSt.					0,00 €	
zuzüglich 7% MWSt.					5,89 €	
# Gesamtsumme (brutto)					89,97 €	

Rechnungsadresse

Firma: Die Netzmacher
 Vorname*: Dirk
 Nachname*: Wildt
 Straße und Hausnummer: Bornthalweg 4
 Postleitzahl: 99092
 Ort: Erfurt
 Land:

Lieferadresse

Kontaktdaten

Mail*: wildt@die-netzmacher.de
 Telefon:
 Fax:

Bestellung

Bemerkung:

AGB*: ☒ Ich akzeptiere die AGB.
 Widerrufsbelehrung*: ☒ Ich habe die Widerrufsbelehrung gelesen.

Quick Shop unverbindlich testen!

Bestellsumme **89,97 €**

Illustration 1: Caddy with an item and the order form

Caddy (1) and the order form (2). The order form is delivered by powermail. Caddy controls powermail while runtime.

In the example above, caddy is connected with the database of TYPO3 Quick Shop. But Caddy is a shopping cart for every database. You can connect Caddy with your database or a database of any other TYPO3 extension.

Live

- <http://typo3-quick-shop.de/warenkorb/>

Confirm



Illustration 2: Enabled confirmation mode: The customer has to confirm his data.

If the confirmation mode of powermail is enabled, the customer has to confirm his data before sending.

Confirmation E-Mail

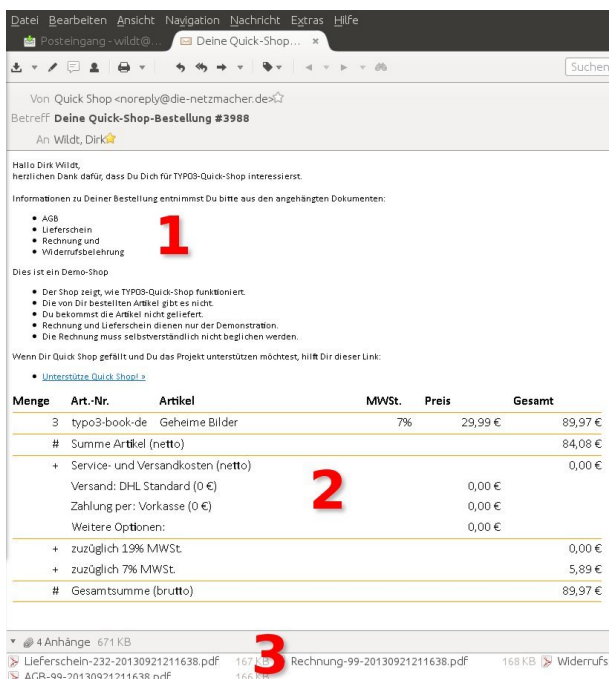


Illustration 3: Confirmation e-mail

The confirmation e-mail contains:

1. a text
2. the caddy
3. attachments

All is configurable of course.

Attachments

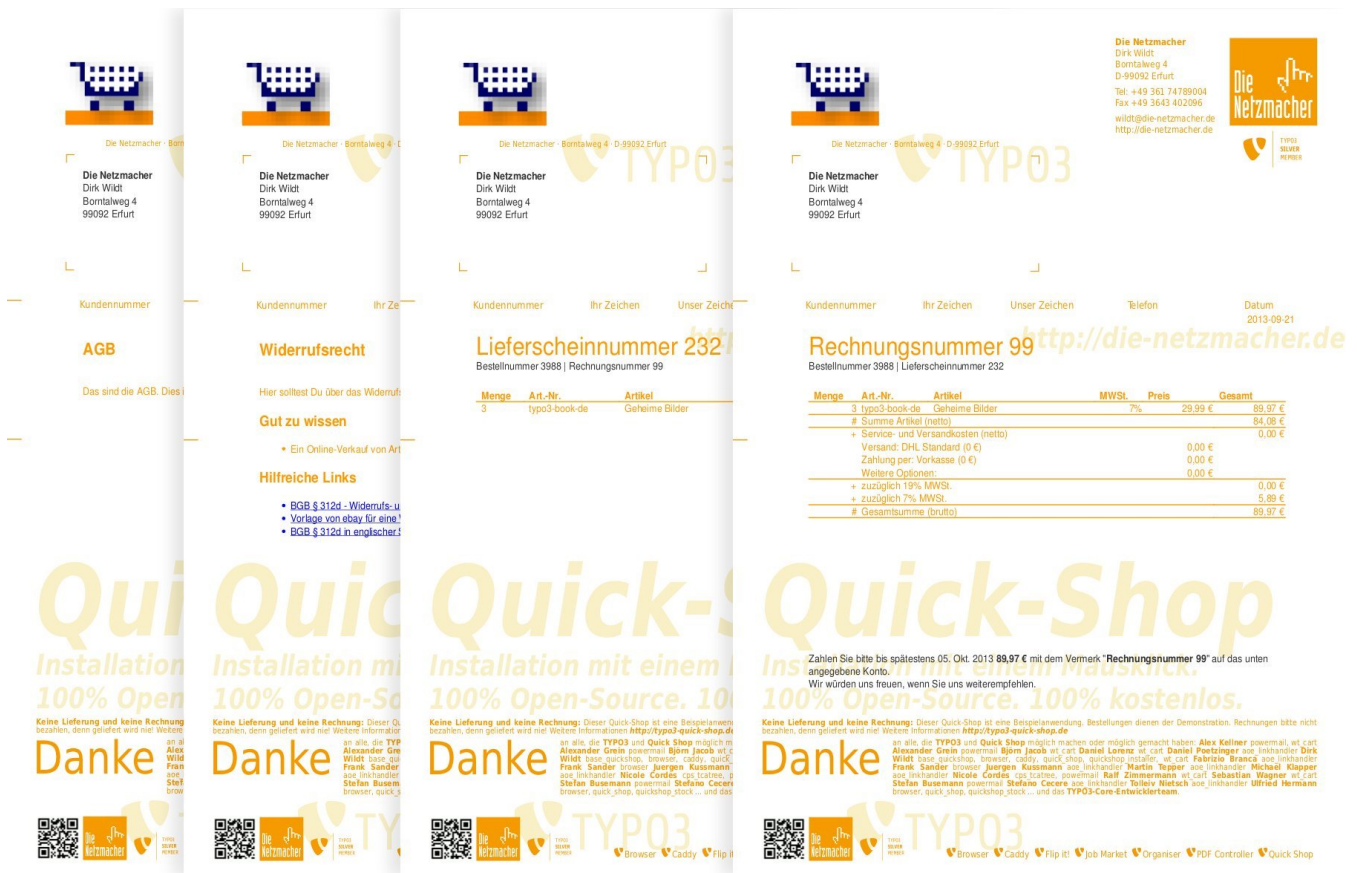


Illustration 4: The caddy e-mail attachments (here: Quick Shop)

Attachments by default are

- the invoice,
- the delivery order
- the power of revocation and
- the terms and conditions.

The corporate design of the attachments is configurable. See

- "Attachments" on page 36 below.

Customer and vendor can get different attachments. See

- "Plugin / Flexform > PDF" on page 23 below.

Colour Schemes

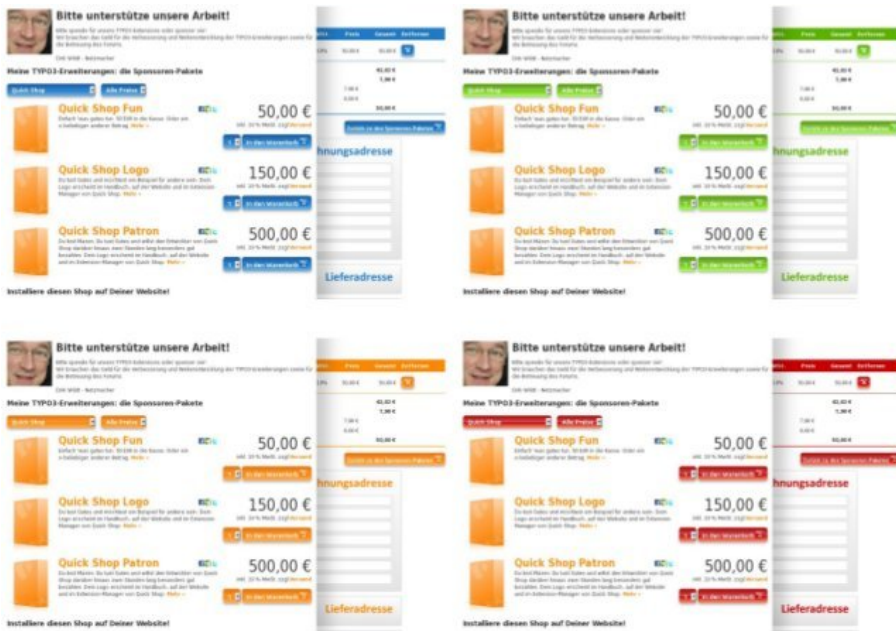


Illustration 5: Caddy colour schemes

Caddy is delivered with four colour schemes. Colours are

- blue
- green
- orange
- red

And you can configure your own scheme with a user interface. See

- "Corporate Design" on page 33 below

Backend

Plugin

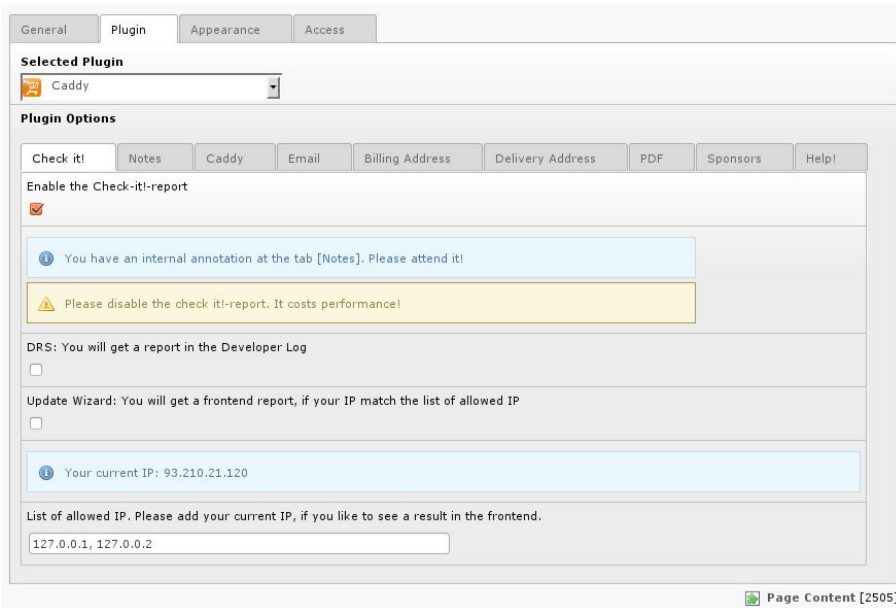


Illustration 6: The caddy plugin.

Caddy has a powerful plugin.

The plugin contains 9 tabs.

The plugin evaluate the configuration of

- the caddy plugin,
- the caddy TypeScript and
- the powermail plugin.

If there is a configuration missing or it is unproper, you will get a prompt within the first tab [Check it!]

See details at "Plugin / Flexform" on page 19 below.

Order Records

Illustration 7: A caddy order record

Caddy archives each order in a database audit proof.

An order record contains among others

- who got when the confirmation e-mail
- which attachments are sent
 - to the customer and
 - to the vendor
- the sent attachments.

Installation

Quick Shop Installation

Check extension dependencies

- ✓ autumaktemplates: "Template Auto-parser" is loaded.
- ✓ base_quickshop: "Quick Shop - Template" is loaded.
- ✓ browser: "browser - TYPO3 without PHP" is loaded.
- ✓ caddy: "Caddy - your shopping cart" is loaded.
- ✓ epa_bastree: "Record tree for TCA" is loaded.
- ✓ powermail: "Powermail" is loaded.
- ✓ quickshops: "Quick Shop" is loaded.

Start the Installation

- 1 Create a backend group
- 2 Create a page tree
- 3 Create the plugins browser, caddy and powermail
- 4 Create page content
- 5 Create TypoScript for the browser, caddy and powermail
- 6 Create configuration records for powermail
- 7 Create six example items
- 8 Create product images
- 9 Create page content
- 10 Consolidating

Start the Installation



Illustration 8: The Quick Shop Installer enables you, to install Caddy ready-to-use with one mouse click.

Introduction

What does Caddy do?

- Caddy is a shopping cart for TYPO3 extensions.
 - You can use Caddy for selling items, for reserving tickets or for ordering documents among others.
 - Caddy has features like
 - PDF attachments in the e-mail to the customer and the vendor like
 - delivery order,
 - invoice,
 - power of revocation and
 - terms and conditions
 - minimum and maximum limitations
 - for each item
 - for all items
 - delivery costs depending on
 - quantity
 - size
 - weight
 - variants (one dimension only)
 - a mini cart
 - e-payment
 - AJAX
 - Caddy archives each order in a database audit proof.
 - Caddy is using powermail for
 - creating order forms and confirmation forms
 - sending e-mails to the customer and the vendor
 - Caddy evaluates the self-configuration and the needed Powermail properties.
 - You can configure the colours of Caddy and Powermail by a user interface.
 - Caddy has ready-to-use templates for the colours
 - blue,
 - green,
 - orange,
 - red.
 - Caddy is delivered with a lot of icons.
 - You can test Caddy in your own TYPO3 instance with one mouse click!
 - The Organiser Installer installs a ready-to-use website with two caddies: one for online-tickets, the other for the ordering of documents.
 - The Quick Shop Installer installs a ready-to-use online shop with TYPO3 sample items.
- See "One-Click-Installation (recommended)" on page 12 below.
- Caddy has a powerful debugging:
 - The DRS – the Development Reporting System – prompts the whole work flow to the backend.
 - Another modul prompts TypoScript properties to the frontend.

What doesn't Caddy do?

Caddy doesn't support features like

- cross selling and
- vouchers

What do you need?

- Powermail (version 1.x or 2.x)
- t3_tcpdf
- Caddy is tested in TYPO3 version 4.5, 4.6 and 4.7 and 6.1

Target Group

- TYPO3 agencies
- TYPO3 integrators
- TYPO3 developers

Installation

Quick Installation

- Install Caddy
- Add a page "Caddy" to your page tree:
 - Add a TypoScript extension template.
 - Add a Caddy plugin.
 - Add a Powermail plugin.
- Configure Caddy supported by the Constant Editor.
- Configure Powermail:
 - Add the needed fields.
 - Add the needed fieldsets.
 - Configure the plugin.
- Configure Caddy by the plugin.

One-Click-Installation (recommended)

The one-click-installation is recommended¹.

Advantages

The advantages of the one-click-installations are:

- You will have a minimum of costs for the installation and the configuration:

You will need

 - the first time 30 minutes over all and
 - all next times 15 minutes over all.
- You will get a running sample with caddy and without
 - any unproper configuration and
 - any costs for debugging.
- A running sample is the best for understanding.
- You can test all features of caddy immediately. Play with Caddy!
- If something goes wrong and you can't cancel it, it doesn't matter.
 - Waste your installation.
 - Repeat the one-click-installation.

Work flow for your own needs

- The one-click-installation offers three Caddy samples:
 - for ordering documents²
 - for reserving tickets³
 - for selling items⁴
- If no one of the three samples doesn't match your needs,
 - waste Caddy,
 - look for another TYPO3 extension, which could match your needs, or

¹ The one-click-installation is obligated only, because there isn't any manual for a manual installation.

² See "Organiser Installer" on page 13 below

³ See "Organiser Installer" on page 13 below

⁴ See "Quick Shop Installer" on page 13 below

- develop your own extension.
- If one of the three samples matches your needs,
 - you can move the page with the Caddy within the page tree to your preferred position,
 - you have to adapt Caddy with the Constant Editor to your needs, and
 - you have to take care, that your order form address all needed values to the page with the Caddy plugin.

Organiser Installer

The Organiser Installer installs a ready-to-use Organiser. The Organiser is TYPO3 for the lobby and for the organisers.

The Organiser supports the publishing of documents, events, headquarters, locations, news and staff.

The Organiser contains Caddy twice:

- for reserving tickets for events
- for ordering documents

See

- Organiser <http://typo3-organiser.de/en/>
- Organiser Installer¹ <http://typo3-organiser.de/en/one-click/>

Quick Shop Installer

Quick Shop Installation

Check extension dependencies

- ✓ autoselecttemplate: "Template An-Parser" is loaded.
- ✓ base_quickshop: "Quick Shop - Template" is loaded.
- ✓ browser: "browser" - TYPO3 without PHP is loaded.
- ✓ caddy: "Caddy - your shopping cart" is loaded.
- ✓ cps_boatsree: "Boatree for TCA" is loaded.
- ✓ powermail: "Powermail" is loaded.
- ✓ quick_shop: "Quick Shop" is loaded.

Start the Installation

- 1 Create a backend group
- 1 Create a page tree
- 1 Create the plugins browser, caddy and powermail
- 1 Create page content
- 1 Create TypoScript for the browser, caddy and powermail
- 1 Create configuration records for powermail
- 1 Create six example items
- 1 Create product images
- 1 Create page content
- 1 Core updating

Start the Installation



Illustration 9: Quick Shop Installer - Installation with one mouse click

The Quick Shop Installer (1) installs a ready-to-use Quick Shop (2, 3) with one mouse click.

The Quick Shop is a small online shop. It sells TYPO3 items by default like TYPO3 clothes, cups and books. You can use Quick Shop for your own items.

The Quick Shop contains Caddy once:

- for selling items.

See

- Quick Shop <http://typo3-quick-shop.de/en/>
- Quick Shop Installer² <http://typo3-quick-shop.de/en/one-click/>

Manually Installation (not documented)

The manually installation of Caddy is complex and isn't recommended. The cause is:

- Caddy isn't a shop.
- Caddy is a shopping cart only.

Because Caddy is a shopping cart only, Caddy has interfaces for third party extensions for

- managing items (done by the Quick Shop, the Organiser or your extension) and

¹ You need the Organiser Installer from version 3.3. I purpose to publish it in September of 2013.

² You need the Quick Shop Installer from version 3.3. I purpose to publish it in September of 2013.

- interactive actions with the customer (done by Powermail)

A manually installation has a lot of possibilities for an unproper configuration.

And the manual for a manually installation has a lot of possibilities for unproper or missing statements too.

I would be happy, if you are appreciative of a missing documentation for a manually installation.

Plugins

Caddy

Frontend



Illustration 10: Plugin Caddy (frontend)

Backend

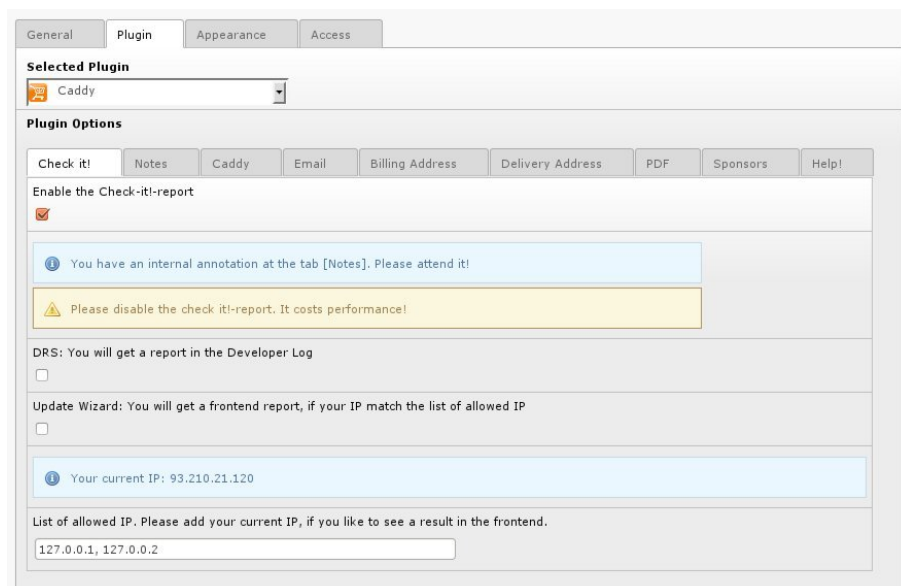


Illustration 11: Plugin Caddy (backend)

Controlling

See details at

- "Controlling Caddy" on page 18 below.

Mini Caddy

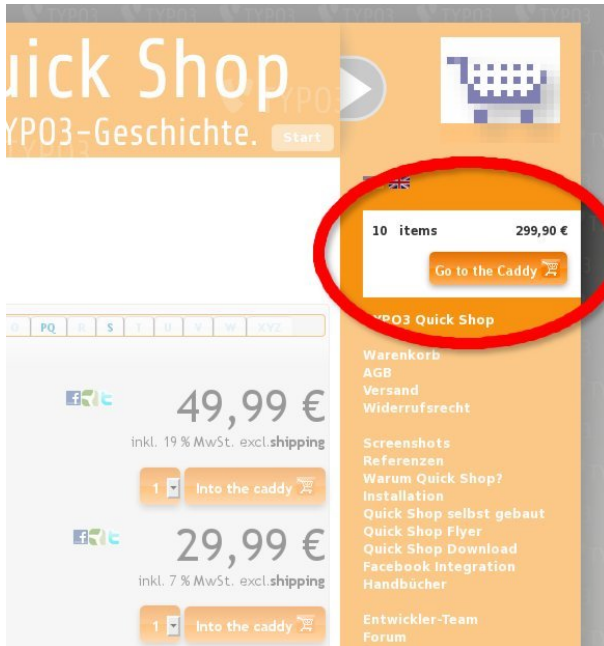


Illustration 12: Plugin Caddy Mini (frontend)

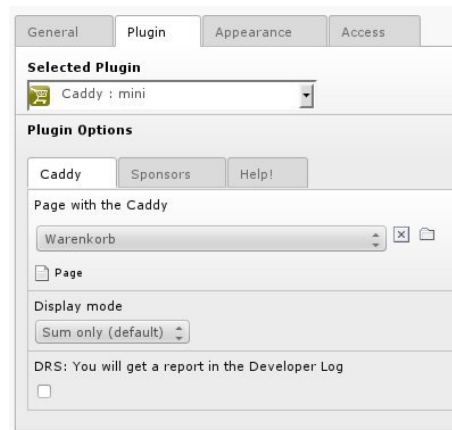


Illustration 13: Plugin Caddy Mini (backend)

Controlling

- Add the Plugin "Caddy: Mini" to your page.
- Configure the Plugin:
 - Enter the page with the Caddy plugin.
 - Display mode:
 - Items and sum
 - Sum only (default)

TypoScript

See "Setup > tx_caddy_pi2" on page 57 below.

Sum Gross

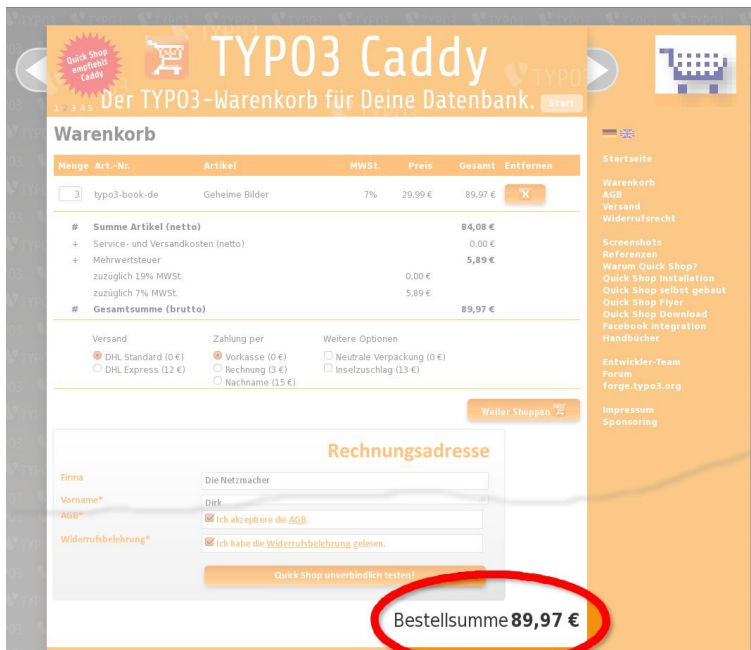


Illustration 14: Plugin Caddy Mini (frontend)

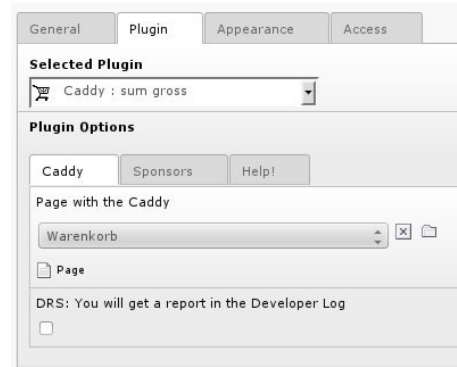


Illustration 15: Plugin Caddy Mini (backend)

Controlling

- Add the Plugin "Caddy: sum gross" to your page.
- Configure the Plugin:
 - Enter the page with the Caddy plugin.

TypoScript

See "Setup > tx_caddy_pi3 – Overview" on page 58 below.

Controlling Caddy

You are controlling Caddy by both user interfaces:

- the Caddy plugin / flexform and
- the Constant Editor.

If you have high demands, you can configure Caddy by TypoScript directly.

Overview

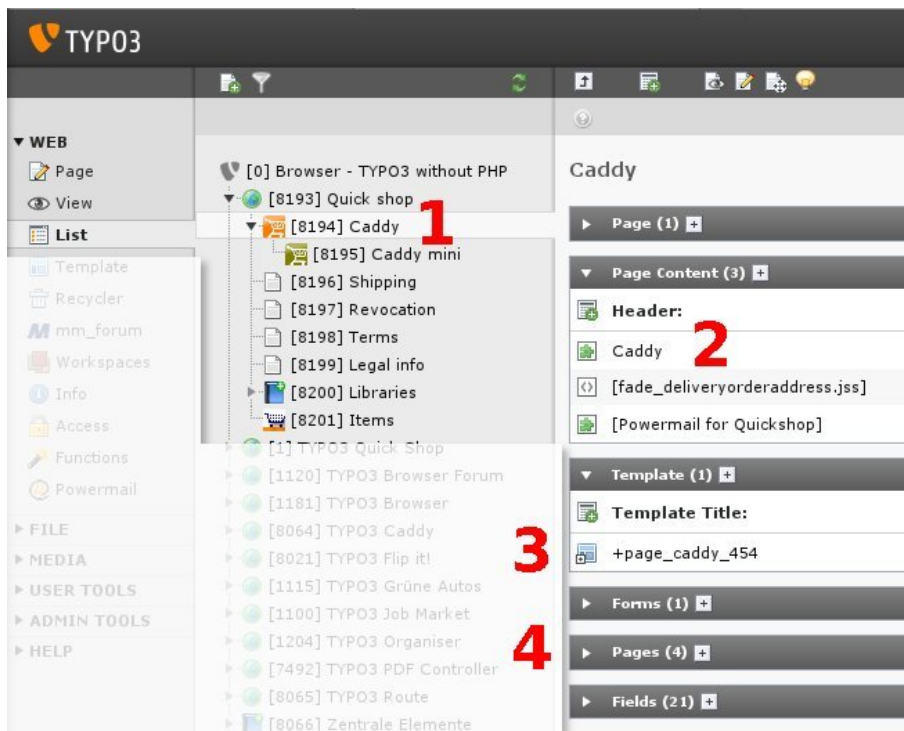


Illustration 16: Sample with Quick Shop

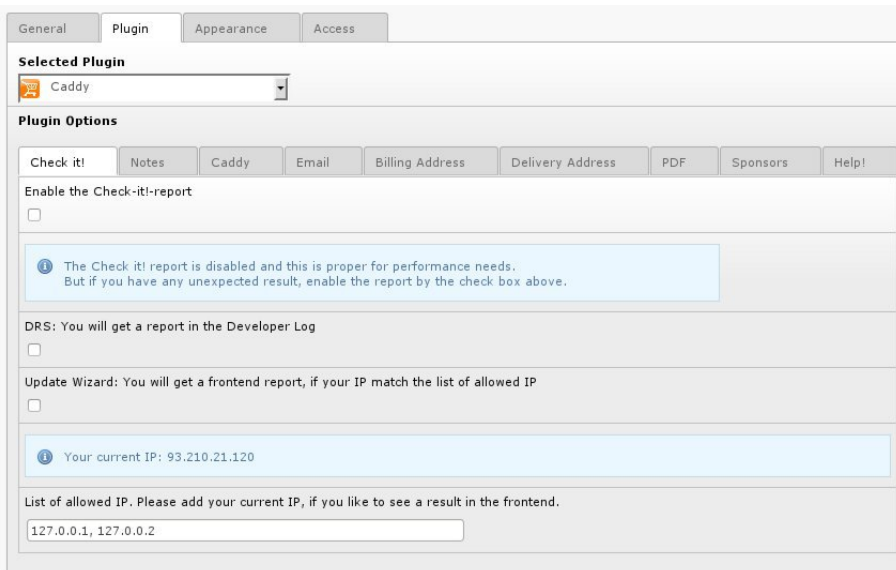
1. The page Caddy (here: [8194 Caddy]) should contain:
2. two or three content elements:
 - the Caddy plugin
 - a JavaScript snippet for slide in and slide out the fieldset delivery address (optional)
 - the Powermail plugin¹
1. a TypoScript extension template
2. and Powermail items like²
 - forms,
 - pages and
 - fields

¹ Caddy supports both versions: Powermail 1.x and 2.x

² Depends on version 1.x and 2.x

Plugin / Flexform

Check it!



The screenshot shows the 'Caddy' plugin configuration page with the 'Check it!' tab selected. The 'Selected Plugin' dropdown is set to 'Caddy'. Under 'Plugin Options', there are several checkboxes: 'Enable the Check-it!-report' (unchecked), 'DRS: You will get a report in the Developer Log' (unchecked), and 'Update Wizard: You will get a frontend report, if your IP match the list of allowed IP' (unchecked). A message box states: 'The Check it! report is disabled and this is proper for performance needs. But if you have any unexpected result, enable the report by the check box above.' Below this, a text box shows 'Your current IP: 93.210.21.120'. At the bottom, there is a section for 'List of allowed IP. Please add your current IP, if you like to see a result in the frontend.' with a text input field containing '127.0.0.1, 127.0.0.2'.

Illustration 17: Plugin [Check it!]

Enable the Check-it!-Report

- If the report is enabled, Caddy will check your Cady configuration and your Powermail configuration.
- If there is an unproper property, you will get a prompt probably with helpful information.

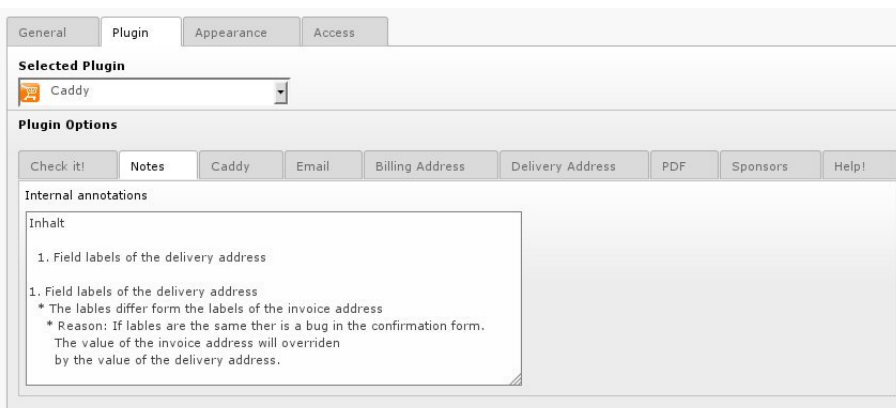
DRS: You will get a report in the Developer Log

- If the DRS is enabled, Caddy will prompt the work flow of the plugin to devlog.
The DRS supports TYPO3 integrators for a proper configuration and for bugfixing.
The DRS supports TYPO3 developers.
- You need the extension devlog, if you want to use this feature.

Update Wizard

- Without any effect in the current version.

Notes



The screenshot shows the 'Caddy' plugin configuration page with the 'Notes' tab selected. The 'Selected Plugin' dropdown is set to 'Caddy'. Under 'Plugin Options', the 'Notes' tab is active. The 'Internal annotations' section contains a text area with the following content: 'Inhalt', '1. Field labels of the delivery address', '1. Field labels of the delivery address', '* The labels differ form the labels of the invoice address', '* Reason: If labes are the same ther is a bug in the confirmation form.', 'The value of the invoice address will overriden by the value of the delivery address.'

Illustration 18: Plugin [Notes]

Notes

- For your internal notes

Caddy

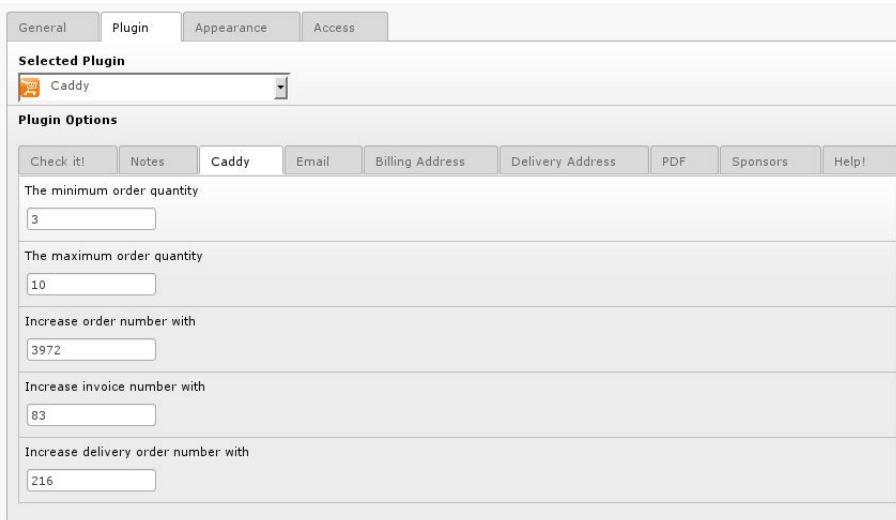


Illustration 19: Plugin [Caddy]

The minimum order quantity

- Minimum order quantity over all (for all items in the caddy)

The maximum order quantity

- Maximum order quantity over all (for all items in the caddy)

Increase order number with

- Caddy starts counting from 0. If you need a higher start number like 1.000, please adapt the starting number to your needs.

Increase invoice number with

- See above.

Increase delivery order number with

- See above.

Reset numbers

If there is a need, to reset the counter of the numbers, please refer to

- "Reset numbers" on page 46 below.

Email

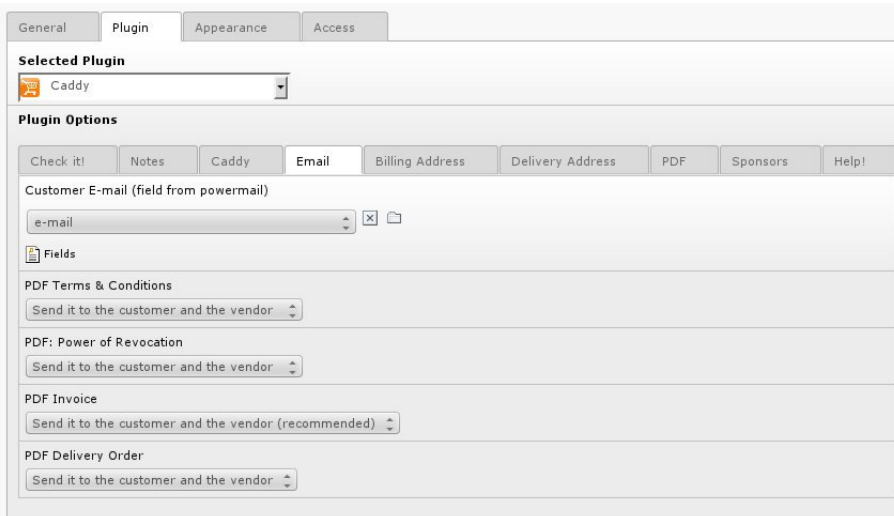


Illustration 20: Plugin [Email]

Customer E-mail (field from powermail)

- Caddy must know the email from the customer.
- You have to link this field with the corresponding field of the powermail form.

PDF Terms & Conditions

- Attach the PDF to
 - nobody,
 - the customer,
 - the vendor or
 - both

PDF: Power of Revocation

- See above

PDF Invoice

- See above

PDF Delivery Order

- See above

Design of the attachments

See

- "Plugin / Flexform > PDF" on page 23 below
- "Corporate Design > Attachments" on page 36 below

Billing Address

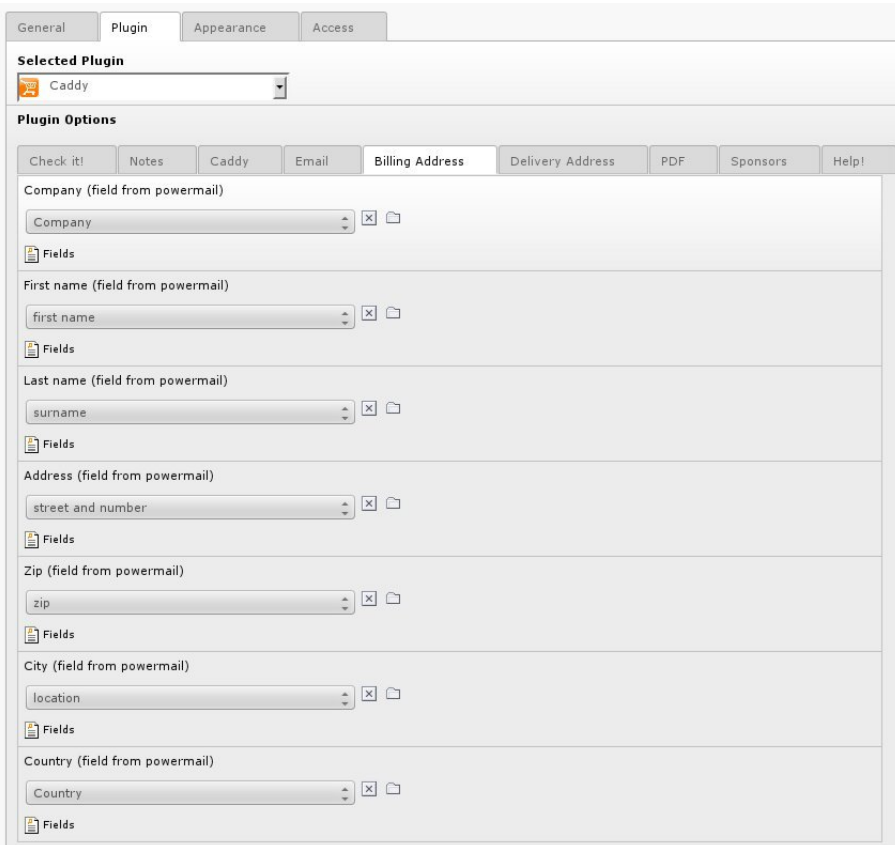


Illustration 21: Plugin [Billing Address]

This is the Powermail connector.

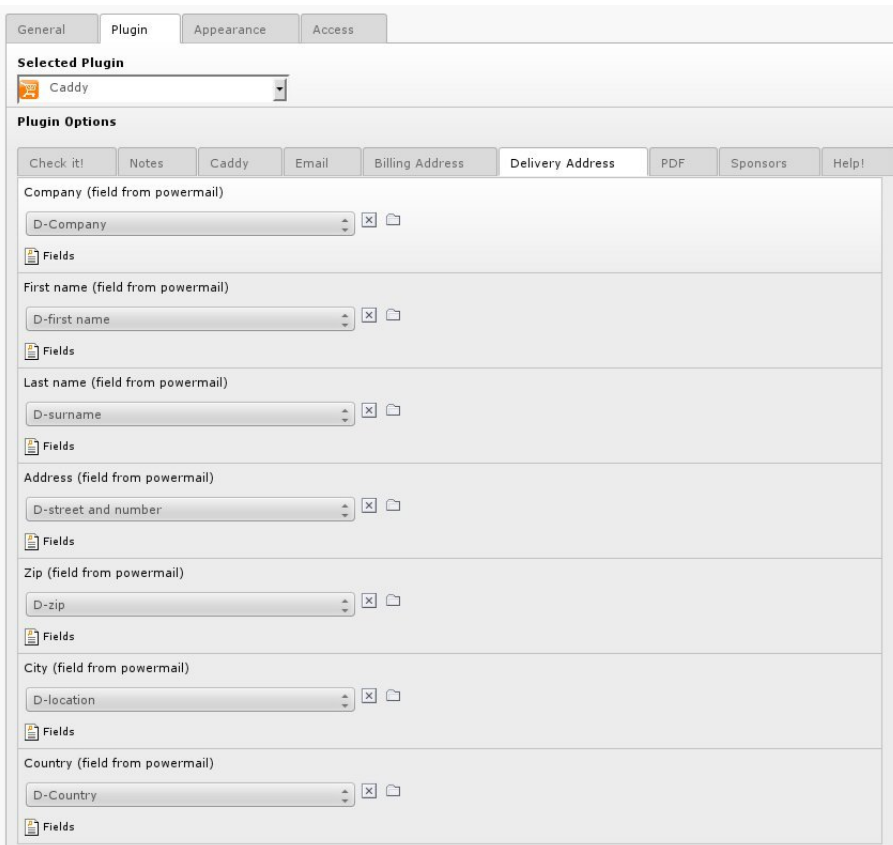
Caddy needs these values for the billing address in PDF attachments.

Each field should be linked with the corresponding field of the powermail form.

It is recommended to have all Powermail fields configured. And to hide a field or fieldset, if you don't need the field or fieldset.

It isn't recommended to leave needed Powermail fields, because Caddy will prompt a warning in the Check-it!-Report.

Delivery Address



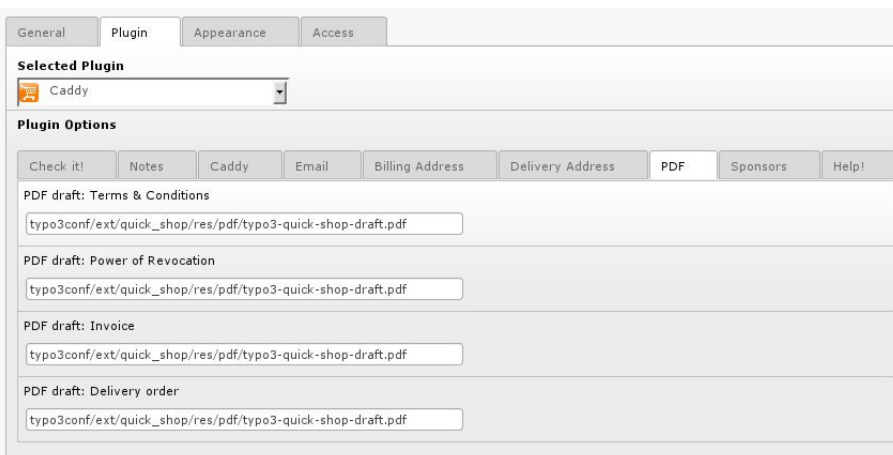
The screenshot shows the 'Delivery Address' configuration tab for the Caddy plugin. The 'Selected Plugin' is 'Caddy'. The 'Plugin Options' section includes tabs for 'Check it!', 'Notes', 'Caddy', 'Email', 'Billing Address', 'Delivery Address' (active), 'PDF', 'Sponsors', and 'Help!'. The 'Delivery Address' section contains several fields, each with a 'Fields' icon and a 'D-' prefix:

- Company (field from powermail): D-Company
- First name (field from powermail): D-first name
- Last name (field from powermail): D-surname
- Address (field from powermail): D-street and number
- Zip (field from powermail): D-zip
- City (field from powermail): D-location
- Country (field from powermail): D-Country

Illustration 22: Plugin [Delivery Address]

See "Billing Address" above.

PDF



The screenshot shows the 'PDF' configuration tab for the Caddy plugin. The 'Selected Plugin' is 'Caddy'. The 'Plugin Options' section includes tabs for 'Check it!', 'Notes', 'Caddy', 'Email', 'Billing Address', 'Delivery Address', 'PDF' (active), 'Sponsors', and 'Help!'. The 'PDF' section contains four fields, each with a 'Fields' icon and a 'D-' prefix:

- PDF draft: Terms & Conditions: typo3conf/ext/quick_shop/res/pdf/typo3-quick-shop-draft.pdf
- PDF draft: Power of Revocation: typo3conf/ext/quick_shop/res/pdf/typo3-quick-shop-draft.pdf
- PDF draft: Invoice: typo3conf/ext/quick_shop/res/pdf/typo3-quick-shop-draft.pdf
- PDF draft: Delivery order: typo3conf/ext/quick_shop/res/pdf/typo3-quick-shop-draft.pdf

Illustration 23: Plugin [Delivery Address]

Background PDF for your attachments.

Please upload a PDF with your corporate design to the fileadmin directory.

Update the values in the fields of this tab.

Further informations about attachments

See

- "Plugin / Flexform > Email" on page 21 above
- "Corporate Design > Attachments" on page 36 below

Sponsors

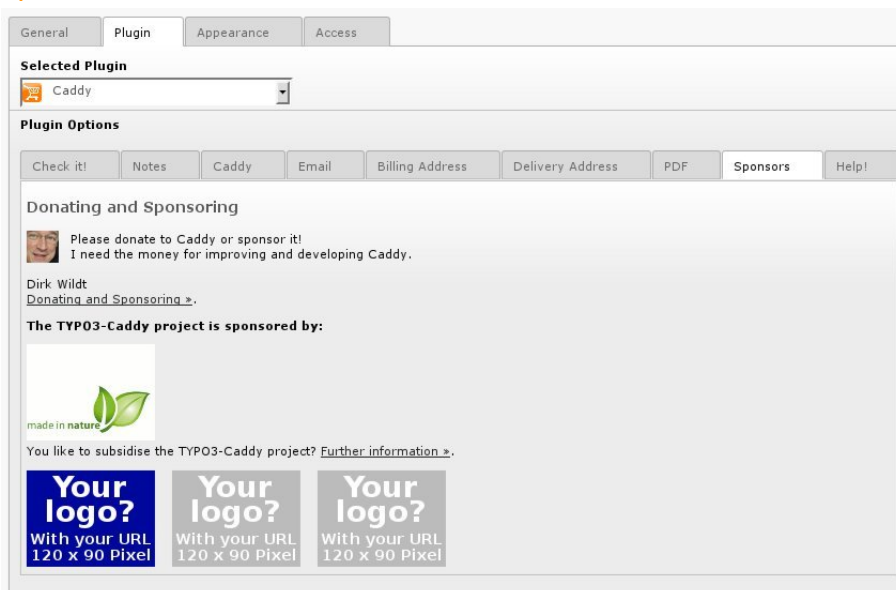


Illustration 24: Plugin [Sponsors]

List of sponsors.

Nothing to do but sponsoring. See

- "Packages for Sponsors" on page 68 below.

Help!

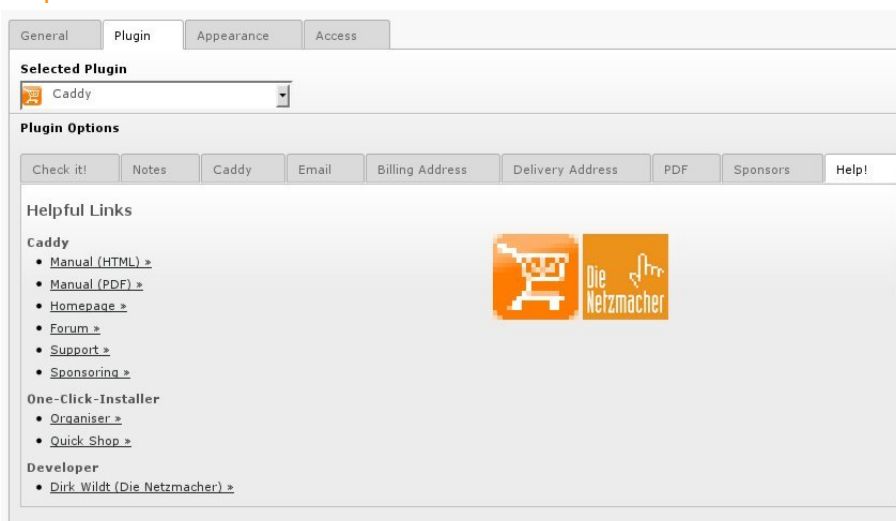


Illustration 25: Plugin [Help!]

Helpful links to

- the manual, the caddy forum and some other caddy websites,
- the one-click-installer and
- the developer.

Nothing to do.

Static Templates

Syntax – Order of the Templates

The labels of the static templates have order numbers.

For example: you have to include Caddy (2+) before Caddy (4+).

Main (obligated)

If you like to use Caddy, you have to include the main template. This is obligated!

- Caddy (1)

TypoScript

- "Setup" from page 47 below.

CSS

CSS templates are needed, if you like to use the CSS of the Caddy extension.¹

CSS in general:

- Caddy (2+) - CSS²

Colours

- Caddy (3+) - CSS blue
- Caddy (3+) - CSS orange
- Caddy (3+) - CSS green
- Caddy (3+) - CSS red

CSS for Powermail 2.x³

- Caddy (6+) - Powermail 2.x CSS fancy

Example

- "Colours" on page 33 below.

TypoScript

- "tx_caddy_pi1._CSS_DEFAULT_STYLE" on page 55 below.

Localization

Some formats are localized like an amount.

- 1,234.56 is the default notation of an amount.

If you like a German format like 1.234,56, please include this static template:

- Caddy (4+) - German

Powermail (obligated)

One of the both Powermail templates is obligated. The proper template depends on your Powermail version.

- Caddy (5+) - Powermail 1.x
- Caddy (5+) - Powermail 2.x Main

TypoScript

- "tx_caddy_pi1.powermail-" on page 56 below.

¹ You are controlling a lot of CSS properties by the Constant Editor.

² Colour is blue

³ There isn't any CSS for Powermail 1.x

Constant Editor

All properties have the prefix plugin.caddy.

Example: main.pid is plugin.caddy.main.pid.

MAIN

Property	Type	Description	Default
main.pid	text	Uid of storage folder: Uid of the folder with the caddy records. Leave it empty (recommended!), to store records at the same page like the caddy plugin.	
main.dueDate	options	Due date for payment. Options <ul style="list-style-type: none"> • now • +1day • +2 days • +3 days • +4 days • +5 days • +6 days • +1 week • +2 weeks • +3 weeks • +1 month 	+2 weeks
main.dueDateFormat	string	Format of the due date. Examples: <ul style="list-style-type: none"> • %Y-%m-%d • %d. %b. %Y 	%Y-%m-%d

MAIN NUMBERFORMAT

For German rules there isn't any need to use the Constant Editor. See

- "Static Templates > Localization" on page 25 above.

Property	Type	Description	Default
main.decimal	text	Decimal. This setting will be used for formatting prices.	2
main.dec_point	text	Decimal symbol: Dot or comma. This setting will be used for formatting prices.	.
main.thousands_sep	text	Thousands separator. Dot or comma. This setting will be used for formatting prices.	,
main.currencySymbol	text	Currency symbol. Example: &euro, \$	\$
main.currencySymbolBeforePrice	boolean	Show currency symbol before price.	true
main.percentSymbol	string	Percent symbol	%

DATABASE

Property	Type	Description	Default
db.table	text	Table*: *Obligate! Name of the table with the items (products). Examples: <ul style="list-style-type: none"> tx_myextension_products tx_quickshop_products tx_cars_model 	
db.title	text	Title*: *Obligate! Field with the title of the item (product).	title
db.price	text	Price*: *Obligate! Field with the gross price of the item (product).	price
db.tax	text	Tax*: *Obligate! Field with the tax category of the item (product).	tax
db.sku	text	SKU: field with the sku (unique id) of the item (product).	
db.min	text	MIN: field with the minimum order quantity of the item (product).	
db.max	text	MAX: field with the minimum order quantity of the item (product).	
db.service_attribute_1	text	Service Attribute 1: the field with the service attribute 1 of the item (product).	
db.service_attribute_2	text	Service Attribute 2: the field with the service attribute 1 of the item (product).	
db.service_attribute_3	text	Service Attribute 3: the field with the service attribute 1 of the item (product).	

DEBUG

Property	Type	Description	Default
debug.paramsAndTs	boolean	Debug params/typoscript (frontend): Get a report in the frontend with received GET-/POST-parameters and typoscript configuration	false
debug.userfunc	boolean	Debug user functions (backend): Enable the DRS for user functions	false

GETPOST

Property	Type	Description	Default
getpost.uid	text	Uid*: *Obligate! Parameter name for a unique id (integer) of your item. caddy uses this id for SQL requests. Example <ul style="list-style-type: none"> tx_browser_pi1 showUid 	
getpost.quantity	text	Quantity*: *Obligate! Parameter name for the quantity/amount field of your order form (e.g. tx_trbooks_qty). The provided parameter must be equal to the name in your HTML template.	qty
getpost.title	text	Title: Parameter name for the title field in your order form.	
getpost.price	text	Parameter name for the gross price field in your order form.	
getpost.service_attribute_1	text	Service Attribute 1: Parameter name for the service attribute field 1 of your order form (e.g. weight).	
getpost.service_attribute_1	text	Service Attribute 2: Parameter name for the service attribute field 2 of your order form (e.g. volume).	
getpost.service_attribute_1	text	Service Attribute 3: Parameter name for the service attribute field 3 of your order form (e.g. length).	
getpost.sku	text	Parameter name for your SKU (stock keeping unit).	

HTML COLORS

There isn't any need to use the Constant Editor, if you like to use these colours:

- blue, green, orange, red

Please include the static template:

- +Caddy CSS +Colour (caddy)

where colour is one of the colours from above.

See

- "Static Templates > CSS" on page 25 above.
- "Colours" on page 33 below.

Property	Type	Description	Default
html.colour.border	colour	Border colour: border colour as CSS name or HTML notation like #FFF or #123456. It is used in the HTML table, which is placed in e-mails and pdf attachments.	#207CCA
html.colour.button.text	colour	Button text colour	#FFF
html.colour.button.gradient1	colour	Button gradient colour 1	#1E5799
html.colour.button.gradient2	colour	Button gradient colour 2	#2989D8
html.colour.button.gradient3	colour	Button gradient colour 3	#207CCA
html.colour.button.gradient4	colour	Button gradient colour 4	#7DB9E8
html.colour.icon.caddy	string	Icon caddy. Default <ul style="list-style-type: none"> • EXT:caddy/files/img/caddy_080_08.png Example <ul style="list-style-type: none"> • "Corporate Design > Icons" on page 34 below 	see description
html.colour.icon.delete	string	Icon delete. Default <ul style="list-style-type: none"> • EXT:caddy/files/img/delete_080_08.png Example <ul style="list-style-type: none"> • "Corporate Design > Icons" on page 34 below 	see description
html.colour.powermail.text	colour	Powermail text	#207CCA
html.colour.table.head.background	colour	Table header background	#207CCA
html.colour.table.head.text	colour	Table header text	#FFF

HTML WIDTH

Property	Type	Description	Default
html.width	text	Width column Quantity: The width of the caddy column quantity in HTML tables for CSS. Units can be px, em or % among others.	10%
html.width	text	Width column SKU: The width of the caddy column stock keep unit in HTML tables for CSS. Units can be px, em or % among others.	15%
html.width	text	Width column item: The width of the caddy column item in HTML tables for CSS. Units can be px, em or % among others.	35%
html.width	text	Width column tax: The width of the caddy column tax in HTML tables for CSS. Units can be px, em or % among others.	10%
html.width	text	Width column net: The width of the caddy column net in HTML tables for CSS. Units can be px, em or % among others.	15%
html.width	text	Width column sum: The width of the caddy column sum in HTML tables for CSS. Units can be px, em or % among others.	15%
html.width	text	Width 3 columns: The total width of the caddy columns sku, item and tax in HTML tables for CSS. Units can be px, em or % among others.	60%
html.width	text	Width 4 columns: The total width of the caddy columns sku, item, tax and net in HTML tables for CSS. Units can be px, em or % among others.	75%

PAGES

Property	Type	Description	Default
pages.caddy	integer	Caddy*:Obligate! Uid of the page with the caddy - your shopping cart	
pages.caddymini	integer	Caddy mini:Uid of the page with the caddy mini - the cart for marginal column	
pages.shop	integer	Shop*:Obligate! Uid of the page which contains the items. Usually the root page.	
pages.revocation	integer	Power of Revocation:Uid of the page with the revocation info (German: Widerrufsbelehrung)	
pages.terms	integer	Terms & Conditions:Uid of the page with the terms & conditions (German: AGB)	

PDF

Property	Type	Description	Default
pdfGeneral.fontfamily	options	Font family: The font family of your PDF attachments Options <ul style="list-style-type: none"> Courier Helvetica Symbol Times ZapfDingbats 	Helvetica
pdfGeneral.fontStretching	integer	Font Stretching: In percent. 100: no effect. 90: smaller. 110: expanded	90

PDF CADDY

Property	Type	Description	Default
pdf.caddy.x	int+	Caddy position left: Position of the caddy (items) from the left margin in millimeters	25
pdf.caddy.y	int+	Caddy position top: Position of the caddy (items) from the top margin in millimeters	130
pdf.caddy.width	int+	Caddy width: Width of the caddy (items) in millimeters	165
pdf.caddy.fontsize	int+	Caddy font-size: Font-size of the caddy (items) in points	10

PDF DATE

Property	Type	Description	Default
pdf.date.align	options	Date alignment: L: left. C: center. R: right. empty: left for LTR or right for RTL Options <ul style="list-style-type: none"> empty L C R 	L
pdf.date.x	int+	Date position left: Position of the date from the left margin in millimeters	175
pdf.date.y	int+	Date position top: Position of the date from the top margin in millimeters	104
pdf.date.width	int+	Date width: Width of the date in millimeters	25
pdf.date.fontsize	int+	Date font-size: Font-size of the date in points	10
pdf.date.strftime	string	Date format: Format of the date. Examples: %Y-m-%d, %d. %M %Y	%Y-%m-%d

PDF DELIVERY ORDER

Property	Type	Description	Default
pdf.deliveryorder.address.x	int+	Address position left: Position of the delivery order address from the left margin in millimeters	25
pdf.deliveryorder.address.y	int+	Address position top: Position of the delivery order address from the top margin in millimeters	55
pdf.deliveryorder.address.fontsize	int+	Address font-size: Font-size of the delivery order address in points	10
pdf.deliveryorder.number.x	int+	Address position left: Position of the delivery number from the left margin in millimeters	25
pdf.deliveryorder.number.y	int+	Address position top: Position of the delivery number from the top margin in millimeters	110
pdf.deliveryorder.number.fontsize	int+	Address font-size: Font-size of the delivery number in points	24
pdf.deliveryorder.number.prefix	string	Number prefix: Prefix for the delivery number. Example : D-	
pdf.deliveryorder.termOfCredit.x	int+	Address position left: Position of the term of credit from the left margin in millimeters	25
pdf.deliveryorder.termOfCredit.y	int+	Address position top: Position of the term of credit from the top margin in millimeters	250
pdf.deliveryorder.termOfCredit.fontsize	int+	Address font-size: Font-size of the term of credit in points	10

PDF INVOICE

Property	Type	Description	Default
pdf.invoice.address.x	int+	Address position left: Position of the invoice address from the left margin in millimeters	25
pdf.invoice.address.y	int+	Address position top: Position of the invoice address from the top margin in millimeters	55
pdf.invoice.address.fontsize	int+	Address font-size: Font-size of the invoice address in points	10
pdf.invoice.number.x	int+	Address position left: Position of the invoice number from the left margin in millimeters	25
pdf.invoice.number.y	int+	Address position top: Position of the invoice number from the top margin in millimeters	110
pdf.invoice.number.fontsize	int+	Address font-size: Font-size of the invoice number in points	24
pdf.invoice.number.prefix	string	Number prefix: Prefix for the invoice number. Example : D-	
pdf.invoice.termOfCredit.x	int+	Address position left: Position of the term of credit from the left margin in millimeters	25
pdf.invoice.termOfCredit.y	int+	Address position top: Position of the term of credit from the top margin in millimeters	250
pdf.invoice.termOfCredit.fontsize	int+	Address font-size: Font-size of the term of credit in points	10

PDF ORDER

Property	Type	Description	Default
pdf.order.number.prefix	string	Number prefix: Prefix for the order number. Example : O-	

PDF REVOCATION

Property	Type	Description	Default
pdf.revocation.align	options	Revocation alignment: L: left. C: center. R: right. empty: left for LTR or right for RTL Options <ul style="list-style-type: none"> • <i>empty</i> • L • C • R 	
pdf.revocation.x	int+	Revocation position left: Position of the date from the left margin in millimeters	25
pdf.revocation.y	int+	Revocation position top: Position of the date from the top margin in millimeters	100
pdf.revocation.width	int+	Revocation width: Width of the date in millimeters	165
pdf.revocation.fontsize	int+	Revocation font-size: Font-size of the date in points	10

PDF TERMS

Property	Type	Description	Default
pdf.terms.align	options	Terms alignment: L: left. C: center. R: right. empty: left for LTR or right for RTL Options <ul style="list-style-type: none"> • <i>empty</i> • L • C • R 	
pdf.terms.x	int+	Terms position left: Position of the date from the left margin in millimeters	25
pdf.terms.y	int+	Terms position top: Position of the date from the top margin in millimeters	100
pdf.terms.width	int+	Terms width: Width of the date in millimeters	165
pdf.terms.fontsize	int+	Terms font-size: Font-size of the date in points	10

PDF TEXTCOLOR

Property	Type	Description	Default
pdf.textColour.address	colour	Address: font colour as CSS name or HTML notation like #FFF or #123456	navy
pdf.textColour.caddy	colour	Caddy: font colour as CSS name or HTML notation like #FFF or #123456	navy
pdf.textColour.date	colour	Date: font colour as CSS name or HTML notation like #FFF or #123456	navy
pdf.textColour.numberDeliveryorder	colour	Number delivery order: font colour as CSS name or HTML notation like #FFF or #123456	navy
pdf.textColour.numberInvoice	colour	Number invoice: font colour as CSS name or HTML notation like #FFF or #123456	navy
pdf.textColour.numberLine	colour	Number line: font colour as CSS name or HTML notation like #FFF or #123456	navy
pdf.textColour.numberOrder	colour	Number order: font colour as CSS name or HTML notation like #FFF or #123456	navy
pdf.textColour.revocation	colour	Revocation: font colour as CSS name or HTML notation like #FFF or #123456	navy
pdf.textColour.termOfCredit	colour	Term of credit: font colour as CSS name or HTML notation like #FFF or #123456	navy
pdf.textColour.terms	colour	Terms: font colour as CSS name or HTML notation like #FFF or #123456	navy

SYMBOLS AND SERVICE ATTRIBUTES

Property	Type	Description	Default
symbol.quantitySymbol	text	Quantity symbol: for example: pc. (English), St. (German)	pc.
symbol.service_attribute_1_symbol	text	Service Attribute 1 symbol: e.g. Weight: g, kg, lbs	
symbol.service_attribute_2_symbol	text	Service Attribute 2 symbol: e.g. Volume: l, gal	
symbol.service_attribute_3_symbol	text	Service Attribute 3 symbol: e.g. Length: m, cm, inch	

TAX

Property	Type	Description	Default
tax.reduced	text	Tax rate for reduced items (e.g. 7). Value will be displayed.	7
tax.reducedCalc	text	Tax rate for reduced items (e.g. 0.07). Used for calculation.	0.07
tax.normal	text	Tax rate for normal items (e.g. 19). Value will be displayed.	19
tax.normalCalc	text	Tax rate for normal items (e.g. 0.19). Used for calculation.	0.19

TEMPLATES

Property	Type	Description	Default
templates.e-mail	string	E-mails: Path to the HTML template file, which is used in e-mails. Default <ul style="list-style-type: none">EXT:caddy/files/templates/caddy.html	see description
templates.html.caddy	string	HTML caddy: Path to the HTML template file, which is used in HTML pages for the caddy Default <ul style="list-style-type: none">EXT:caddy/files/templates/caddy.html	see description
templates.html.caddymini	string	HTML caddy mini: Path to the HTML template file, which is used in HTML pages for the mini caddy Default <ul style="list-style-type: none">EXT:caddy/files/templates/caddy.html	see description
templates.html.caddysum	string	HTML caddy sum: Path to the HTML template file, which is used in HTML pages for the caddy sum Default <ul style="list-style-type: none">EXT:caddy/files/templates/caddy.html	see description
templates.pdf.deliveryorder	string	PDF delivery order: Path to the HTML template file, which is used in the delivery order - a PDF attachment Default <ul style="list-style-type: none">EXT:caddy/files/templates/caddy.html	see description
templates.pdf.invoice	string	PDF invoice: Path to the HTML template file, which is used in the invoice - a PDF attachment Default <ul style="list-style-type: none">EXT:caddy/files/templates/caddy.html	see description

Corporate Design

Colours

Caddy is delivered with the four colour schemes:

- blue, green, orange, red.

This schemes has an effect on

- HTML content,
- E-mail and
- PDF attachments.

Screenshots

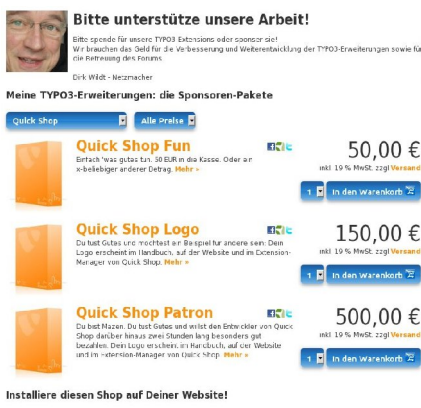


Illustration 26: Caddy blue

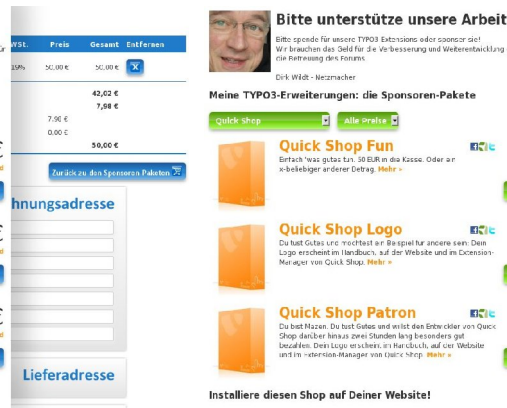


Illustration 27: Caddy green

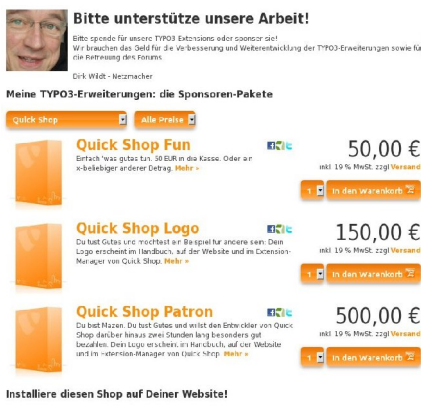


Illustration 28: Caddy orange

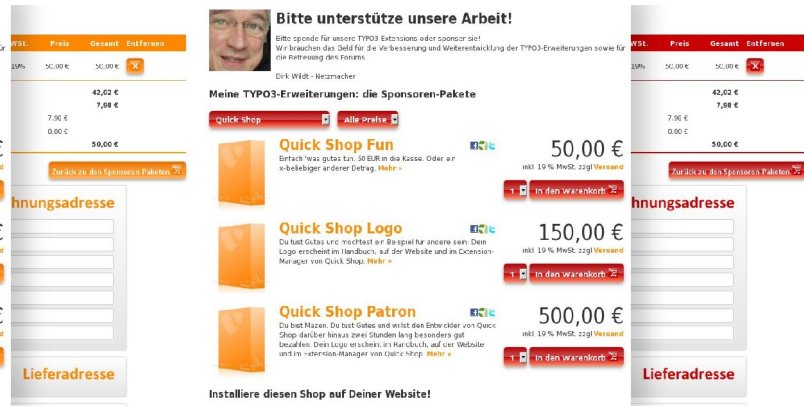


Illustration 29: Caddy red

Select a colour scheme

You select a colour scheme by including a static template for this colour. See

- "Static Templates > CSS" on page 25 above.

The static template overwrites the colour properties in the Constant Editor. If you need another colour scheme, you have to configure the colours in the Constant Editor. See

- "Constant Editor > HTML COLORS" on page 28 above.

Send your scheme to the developer!

If you are creating your own scheme, please send the values of the Constant Editor to the developer. I will include this scheme as a static template. My email: <http://wildt.at.die-netzmacher.de/>

Icons

Controlling

Preis	Gesamt	Entfernen
50,00 €	50,00 €	
	42,02 €	
	7,98 €	
7,98 €		
0,00 €		
	50,00 €	
Zurück zu den Sponsoren-Paketen 		

Illustration 30: Caddy blue

Constant Editor

Template tools

Edit constants for template

page_t3browserForum_133 (TYPO3-Browser-Forum - TYPO3 ohne PHP)

Category

CADDY - HTML COLORS (13)

Border color [plugin.caddy.html.color.border]
border color as CSS name or HTML notation like #FFF or #123456. It is used in the HTML
#207CCA

Button text color [plugin.caddy.html.color.button.text]
#FFF

Button gradient color 1 [plugin.caddy.html.color.button.gradient1]
#1E5799

Button gradient color 2 [plugin.caddy.html.color.button.gradient2]
#1E5799

Icon caddy [plugin.caddy.html.color.icon.caddy]
EXT:caddy/files/img/caddy_080_01.png

Icon delete [plugin.caddy.html.color.icon.delete]
EXT:caddy/files/img/delete_080_01.png

Powermail text [plugin.caddy.html.color.powermail.text]
#207CCA

Illustration 31: Icon path in the Constant Editor

Preis	Gesamt	Entfernen
50,00 €	50,00 €	
	42,02 €	
	7,98 €	
7,98 €		
0,00 €		
	50,00 €	
Zurück zu den Sponsoren-Paketen 		

Illustration 32: caddy blue/red

If you like to change the icons, you have to configure two properties with the Constant Editor:

1. html.colour.icon.caddy
2. html.colour.icon.delete

See "Constant Editor > HTML COLORS" on page 28 above.

We replaced blue icons with red icons in the illustration above.

Ready-to-use icons

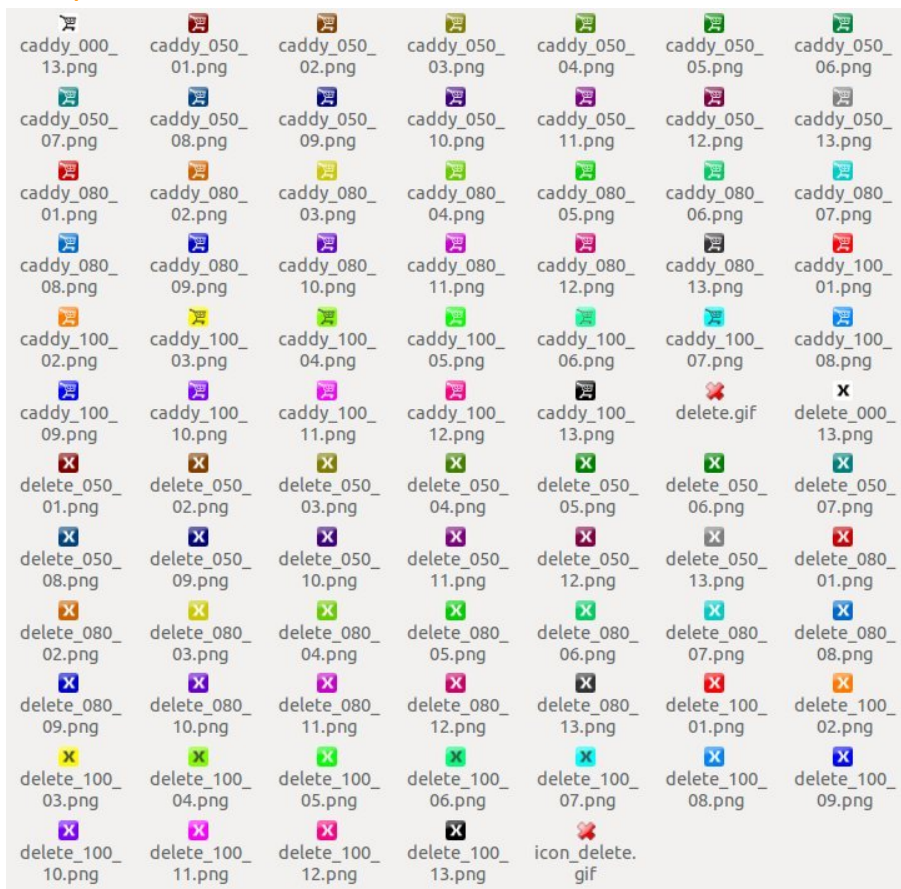


Illustration 33: Caddy icons

Caddy is delivered with two icons:

- the caddy icon
- the delete icon

Each icon is available in 13 colours.

Each colour is available in three intensities.

See the 82 icons in the illustration above.

The icons are stored at

- files/img and
- <http://forge.typo3.org/projects/extension-caddy/repository/show/trunk/files/img>

E-Mail

Coloured lines are used in e-mails.

You configure the colour by the Constant Editor. See

- "Colours" on page 33 above.
- "Constant Editor > HTML COLORS" on page 28 above.

HTML Template

- "Constant Editor > TEMPLATES" on page 32 above.
- "Setup > tx_caddy_pi1.templates" on page 53 below.

Attachments

Caddy attaches PDF files to an e-mail.

This PDF attachments are applicable:

- invoice
- delivery order
- power of revocation
- terms and conditions

You can add more attachments unlimited.

Corporate Design



Illustration 34: Your (1) draft will merged ... Illustration 35: ... with (2) Caddy content ... Illustration 36: ... to the (3) attachment.

You can use your own corporate design for each attachment:

1. Your draft (left hand) will merged with
2. the content generated by Caddy (middle) to
3. the attachment (right hand).

The illustration above displays an example for an invoice.

Your own draft

Your draft must be a PDF file. You will have best quality.

Caddy is using the German DIN A/4 paper size by default:

- width: 210 millimeter
- height: 297 millimeters

Upload

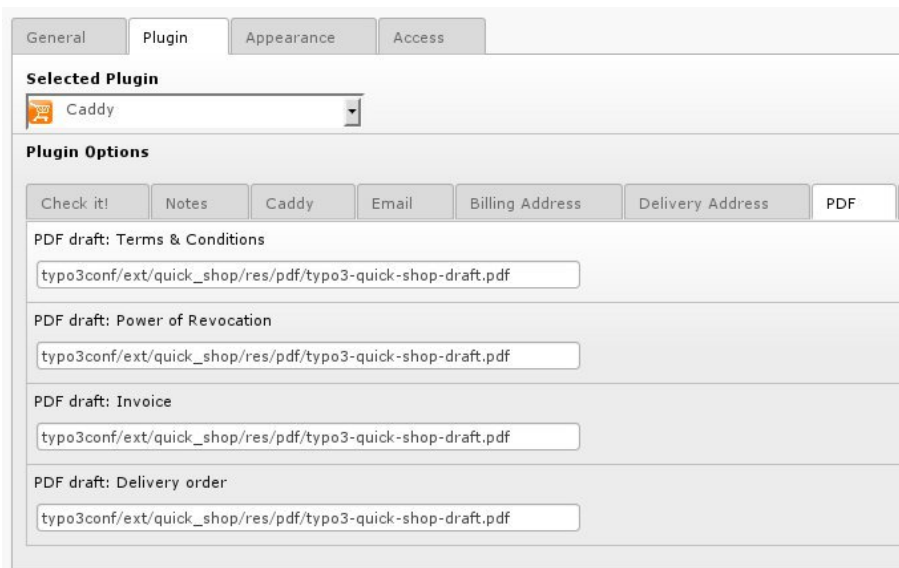
Please upload your draft:

1. Modul File Filelist
2. Page tree Select a folder
3. Edit area Upload your PDF file

In our example we uploaded my.pdf into the directory fileadmin/user_upload/quick_shop.

Configuring Paths

Please configure the Caddy paths.



The screenshot shows the 'Caddy' plugin configuration interface. At the top, there are tabs for 'General', 'Plugin', 'Appearance', and 'Access'. The 'Plugin' tab is selected. Below the tabs, there is a 'Selected Plugin' dropdown menu showing 'Caddy'. Underneath, there is a 'Plugin Options' section with several sub-tabs: 'Check it!', 'Notes', 'Caddy', 'Email', 'Billing Address', 'Delivery Address', and 'PDF'. The 'PDF' tab is selected. This tab contains four sections, each with a text input field for a PDF draft path:

- PDF draft: Terms & Conditions**: `typo3conf/ext/quick_shop/res/pdf/typo3-quick-shop-draft.pdf`
- PDF draft: Power of Revocation**: `typo3conf/ext/quick_shop/res/pdf/typo3-quick-shop-draft.pdf`
- PDF draft: Invoice**: `typo3conf/ext/quick_shop/res/pdf/typo3-quick-shop-draft.pdf`
- PDF draft: Delivery order**: `typo3conf/ext/quick_shop/res/pdf/typo3-quick-shop-draft.pdf`

Illustration 37: [PDF] tab of the Caddy plugin

1. Modul Web List
2. Page tree The page with the caddy plugin
3. Edit area Select the caddy plugin
4. tab [Plugin] > tab [PDF]
(see illustration above)
5. Enter the path of your PDF file
(here: `fileadmin/user_upload/quick_shop/my.pdf`)

Colours

You configure the colour ready-to-use by the include static templates and for individual needs by the Constant Editor. See

- "Colours" on page 33 above.
- "Constant Editor > HTML COLORS" on page 28 above.
- "Constant Editor > PDF TEXTCOLOR" on page 31 above

Properties

You can control nearby everything within a PDF attachment:

- the invoice address (1)
- the delivery address (2)
- the date (3)
- the so called number line (4)
- the caddy (5), the power of revocation and the terms and conditions
- the term of credit (6)
- the background and
- the font.

You are controlling this properties overall:

- the date,
- the font family,
- the font stretching and
- the font colour.

You are controlling this properties per attachment¹:

- invoice addresses and delivery address
- invoice number and delivery number
- terms of credit

Background

You can use your own corporate design for the background in an attachment. See

- "Corporate Design" on page 36 above.

Font Family and Font Stretching

You are controlling

- the font family and
- the font stretching

in general by the Constant editor. See

- "Constant Editor > PDF" on page 29 above,
- "Constant Editor > PDF" on page 29 above.

You are controlling it for each element by TypoScript. See

- "Setup > tx_caddy_pi1.pdf" on page 52 below.

Font Colour

You are controlling the font colour in general by the Constant editor. See

- "Constant Editor > PDF TEXTCOLOR" on page 31 above.

You are controlling it for each element by TypoScript. See

- "Setup > tx_caddy_pi1.pdf" on page 52 below.



Illustration 38: Properties of a PDF attachment (here: invoice)

¹ Attachments are: invoice, delivery order, power of revocation and terms & conditions

Invoice Address

The invoice address is used in the attachments:

- invoice
- delivery order, if there isn't any different address for the delivery
- power of revocation
- terms and condition

You are controlling the invoice address in general by the Constant editor. See

- "Constant Editor > PDF INVOICE" on page 30 above.

You are controlling it for each element by TypoScript. See

- "Setup > tx_caddy_pi1.pdf" on page 52 below.

Delivery Address

If there are different addresses for the invoice and the delivery, the delivery address is used in the attachments:

- invoice
- delivery order

You are controlling the invoice address in general by the Constant editor. See

- "Constant Editor > PDF DELIVERY ORDER" on page 30 above.

You are controlling it for each element by TypoScript. See

- "Setup > tx_caddy_pi1.pdf" on page 52 below.

Date

The date is used in the attachments:

- invoice
- delivery order
- power of revocation
- terms and condition

You are controlling the date in general by the Constant editor. See

- "Constant Editor > PDF DATE" on page 29 above.

You are controlling it for each element by TypoScript. See

- "Setup > tx_caddy_pi1.pdf" on page 52 below.

Numbers

You are controlling the numbers in general by the Constant editor:

- invoice
"Constant Editor > PDF INVOICE" on page 30 above.
- delivery order
"Constant Editor > PDF DELIVERY ORDER" on page 30 above.
- power of revocation
"Constant Editor > PDF REVOCATION" on page 31 above.
- terms and condition
"Constant Editor > PDF TERMS" on page 31 above.

You are controlling it for each element by TypoScript. See

- "Setup > tx_caddy_pi1.pdf" on page 52 below.

Caddy

The caddy is used in the attachments:

- invoice

- delivery order

The caddy format depends on the attachment.

- The caddy of an invoice displays items and calculation.
- The caddy of a delivery order displays items only.

You are controlling

- position,
- width and
- font size

in general by the Constant editor:

- "Constant Editor > PDF CADDY" on page 29 above.

Term of Credit

The term of credit is used in the attachments:

- invoice
- delivery order

You are controlling it in general by the Constant editor:

- "Constant Editor > PDF INVOICE" on page 30 above.
- "Constant Editor > PDF DELIVERY ORDER" on page 30 above.

Terms and Conditions

The terms and conditions are taken from the page Terms & Conditions. See

- "Constant Editor > PAGES" on page 29 above.

The terms and conditions are used in the attachment:

- terms and conditions

You are controlling

- position,
- width and
- font size

in general by the Constant editor:

- "Constant Editor > PDF TERMS" on page 31 above.

You can control it by TypoScript. See

- "Setup > tx_caddy_pi1.pdf" on page 52 below.

Power of Revocation

The power of revocation is taken from the page Power of Revocation. See

- "Constant Editor > PAGES" on page 29 above.

The power of revocation are used in the attachment:

- power of revocation

You are controlling

- position,
- width and
- font size

in general by the Constant editor:

- "Constant Editor > PDF REVOCATION" on page 31 above.

You can control it by TypoScript. See

- "Setup > tx_caddy_pi1.pdf" on page 52 below.

Work-flow

This section is for understanding only.

Add, update and delete an item

Add an item

A customer can add an item into the caddy.

- You have to create the form – or the link – with the GET parameters or POST parameters needed by Caddy.
- Caddy needs the uid of the item and the quantity selected by the customer.
- Caddy checks the price and the tax of the item in the table of the item.
- You have to configure the table label and some field labels by the TypoScript Constant Editor.
- Caddy stores all needed values in a session.
- You can use this data everywhere at your website i.e. with the mini caddy plugin.
- The DRS – Development Reporting System – prompts the session data to devlog.

Update an item

The customer can update the quantity of an item.

- Caddy updates the session data.

Delete an item

The customer can delete an item from the caddy.

- Caddy updates the session data.

Calculating

Caddy calculates the cost of the order.

Parts of this calculation are among others

- items x quantity x price gross
- delivery costs
- payment costs
- special costs
- sum net
- sum gross
- tax

Caddy stores all values in the session. You can use this data everywhere at your website i.e. with the mini caddy plugin.

The DRS – Development Reporting System – prompts the session data to devlog.

TypoScript

- "Setup > tx_caddy_pi1.marker.sum" on page 50 below.

Ordering

The customer can order the items of course.

The confirmation e-mail – the e-mail to the customer – and the order e-mail – the e-mail to the vendor – is generated and sent by Powermail.

Caddy controls Powermail:

- If the caddy is empty, it hides the Powermail form by CSS.

Caddy is listening on powermail parameters:

- GET parameters and POST parameters
- Some other "internal" parameters like
 - the Powermail confirmation mode and
 - the Powermail version (1.x or 2.x)

Caddy is listening on the Powermail session.

Caddy can detect – depending on the parameters above –, if the Powermail form is sent.

If the Powermail form is sent:

- Caddy updates all needed session data
- Caddy serves some methods (user functions), which are called by Powermail like
 - controlling attachments for the e-mails to the customer and the sender
 - serving data like the whole caddy and the numbers for the delivery order, the invoice and the order itself.
- Caddy cleans up the own session.

TypoScript

- "Setup > tx_caddy_pi1.powermail" on page 56 below.

Variants

Sorry, that variants isn't explained in detail. See below only a sample of the TYPO3 Organiser.

The TYPO3 Organiser is using the property variants for reserving tickets.

The sample is a variant of

- ticket and
- price

TypoScript Snippet

```

plugin.tx_caddy_pil {
    db {
        sql >
        sql = TEXT
        sql {
            value (
                SELECT
                    CAST('{GP:tx_org_cal|uid}' AS SIGNED) AS pseud,
                    CONCAT('<strong>', tx_org_cal.title, '</strong><br />',
                        '###GP:TX_ORG_CAL.DATETIME###<br />',
                        'Karte ', tx_org_calentrance.title, ' ', tx_org_calentrance.value, ' &euro;')
                    tx_org_calentrance.value AS title,
                    tx_org_tax.value AS gross,
                    tx_org_tax.value AS tax,
                    CAST('{GP:tx_org_cal|uid}' AS SIGNED) AS sku
                FROM `tx_org_cal`
                LEFT JOIN tx_org_cal_mm_tx_org_calentrance
                ON (
                    tx_org_cal_mm_tx_org_calentrance.uid_local = tx_org_cal.uid)
                LEFT JOIN tx_org_calentrance
                ON (
                    tx_org_cal_mm_tx_org_calentrance.uid_foreign = tx_org_calentrance.uid
                    AND tx_org_calentrance.uid = '###GP:TX_ORG_CALENTRANCE.UID###'
                    ###ENABLE_FIELDS:TX_ORG_CALENTRANCE###)
                LEFT JOIN tx_org_tax
                ON (
                    tx_org_calentrance.tx_org_tax = tx_org_tax.uid
                    ###ENABLE_FIELDS:TX_ORG_TAX###)
                WHERE tx_org_cal.uid = '###GP:TX_ORG_CAL.UID###'
                    ###ENABLE_FIELDS:TX_ORG_CAL###
            )
            insertData = 1
        }
    }
    settings {
        variant {
            10 = tx_org_calentrance.uid
        }
    }
}

```

Archive audit proof

Caddy archives each order in a database.

Database is read-only

By default this database is read-only:

- You can't change any value in a record.
- You can't delete any record.

The database stores among others:

- All attachments, which are sent to
 - the customer
 - the vendor
- The e-mail address of the customer.

Both – read-only and the disabled delete property – is meant with audit proof¹.

Be aware:

If you need the logged data for ever, you have to take care off a backup!

Disable audit proof

You can disable the audit proof mode by the extension manager.

Possible reasons

There are some needs for disabling the audit proof:

- You have tested Caddy and you want to delete the logged data.
- Your database is growing an growing and you want to delete records and files.

Extension Manager

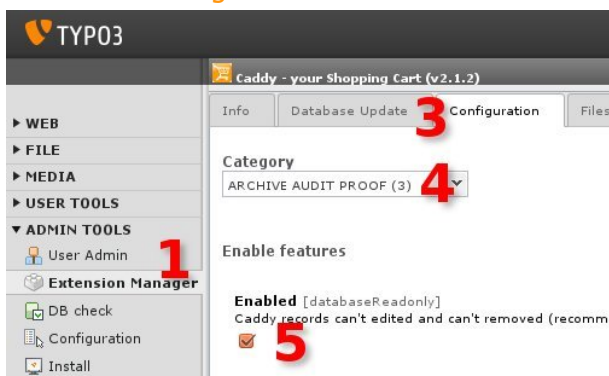


Illustration 39: Audit proof by the Extension Manager

- | | |
|------------------------|---------------------------------|
| 1. Module: Admin Tools | Extension Manager |
| 2. Edit Area | Select the Caddy extension |
| 3. | Tab [Configuration] |
| 4. | Category: [ARCHIVE AUDIT PROOF] |
| 5. | Enabled: [] |
| 6. | [Update] |

¹ German: revisionssicher

Reset numbers

Caddy is storing the numbers in the registry of TYPO3 – table sys_registry.

If you like to reset the numbers, you need a SQL command like in the examples below.

Get all keys

```
SELECT *
FROM sys_registry
WHERE entry_namespace = 'tx_caddy'
```

The result will look like this:

```
"uid","entry_namespace","entry_key","entry_value"
"561","tx_caddy","lastOrder_1279","i:1;"
"553","tx_caddy","lastOrder_15","i:4;"
"563","tx_caddy","lastOrder_7567","i:4;"
...
"626","tx_caddy","page_7843_deliveryorder","i:1;"
"627","tx_caddy","page_7843_invoice","i:1;"
"628","tx_caddy","page_7843_order","i:1;"
```

Syntax of the entry_key

The entry_key has the format:

- pageUid_KindOfNumber

where

- pageUid is
 - the uid of the page, which contains the Caddy plugin
 - format is: page_uid
- KindOfNumber is
 - deliveryorder
 - invoice
 - order

Reset (remove the keys)

All keys

```
DELETE
FROM sys_registry
WHERE entry_namespace = 'tx_caddy'
```

Keys of a single Caddy plugin

```
DELETE
FROM sys_registry
WHERE entry_namespace = 'tx_caddy'
AND entry_key LIKE 'page_7567_%'
```

where 7567 is the page uid of your Caddy plugin

Reference

It is recommended to configure Caddy by the Constant Editor and not by the TypoScript Object Browser directly.

Constant Editor

Please refer to the section "Controlling Caddy > Static Templates" on page 25 above.

Setup

It is recommended to configure Caddy by the Constant Editor and not by the TypoScript Object Browser directly.

All properties have the prefix plugin.tx_caddy_pi1.

Example: db.table is plugin.tx_caddy_pi1.db.table

tx_caddy_pi1 – Overview

Property:	Data type:	Description:
plugin.tx_caddy_pi1	array	<p>Configuration for the Caddy main plugin</p> <p>Code:</p> <pre> plugin.tx_caddy { conditions ... db ... debug ... getpost ... marker ... options ... pdf ... pid ... pluginCheck ... symbols ... templates ... tax ... _HTMLMARKER ... _CSS_DEFAULT_STYLE ... powermail ... }</pre>

tx_caddy_pi1.conditions

Property:	Data type:	Description:
conditions	array	<p>Conditions for calculating. Array contains one condition only: the minimum limit for the gross sum over all items (without any other cost like for delivery or service)</p> <p>Code:</p> <pre> plugin.tx_caddy { conditions { ... <-please inspect it with the TypoScript Object Browser } }</pre>

tx_caddy_pi1.db

Property:	Data type:	Description:
db	array	<p>Interface for your database.</p> <p>Please configure it with the Constant Editor. See</p> <ul style="list-style-type: none"> "Constant Editor > DATABASE" on page 27 above. <p>Example:</p> <pre> plugin.tx_caddy { db { table = tx_quickshop_products title = title gross = price tax = tax sku = sku min = quantity_min max = quantity_max service_attribute_1 = service_attribute_2 = service_attribute_3 = sql { ... -> See tx_caddy_pi1.db.sql below } } } </pre>

tx_caddy_pi1.db.sql

Property:	Data type:	Description:
db.sql	array	<p>Extend the database interface for special needs – usually variants.</p> <p>Example:</p> <pre> plugin.tx_caddy { db { sql { ... -> See Error: Reference source not found on page Error: Reference source not found Error: Reference source not found } } settings { ... -> See tx_caddy_pi1.settings on page 53 below } } </pre>

tx_caddy_pi1.debug

Property:	Data type:	Description:
debug	boolean	<p>Prompts a debug report with informations about TypoScript to the frontend.</p> <p>Please configure it with the Constant Editor. See</p> <ul style="list-style-type: none"> "Constant Editor > DEBUG" on page 27 above. <p>Example:</p> <pre> plugin.tx_caddy { debug = 0 } </pre>

tx_caddy_pi1.getpost

Property:	Data type:	Description:
getpost	array	<p>Configuration for the Get-/Post-parameters, which are sent by GET or POST. Each item has the COA property.</p> <p>Please configure it with the Constant Editor. See</p> <ul style="list-style-type: none"> "Constant Editor > GETPOST" on page 27 above. <p>Code:</p> <pre> plugin.tx_caddy { getpost { gross ... min ... max ... qty ... sku ... tax ... title ... uid ... volume ... weight ... } } </pre>

tx_caddy_pi1.marker.items

Property:	Data type:	Description:
marker.items	array	<p>Configuration of marker, which are used in HTML templates for an item. Example for an HTML marker:</p> <ul style="list-style-type: none"> ###SUMGROSS### <p>Please inspect the items below with the TypoScript Object Browser.</p> <p>Code:</p> <pre> plugin.tx_caddy { marker { items { delete = IMAGE // delete icon gross = COA // gross costs of the item min = TEXT // minimum quantity for ordering max = TEXT // maximum quantity for ordering net = COA // net costs of the item uid = TEXT // uid of the item qty = TEXT // quantity of the item service_attribute_1 = TEXT // service attribute 1 service_attribute_2 = TEXT // service attribute 2 service_attribute_3 = TEXT // service attribute 3 sku = TEXT // stock keeping unit sumgross = COA // gross costs sum (gross costs * qty) sumnet = COA // net costs sum (net costs * quantity) sumtax = COA // tax costs sum (tax costs * quantity) tax = TEXT // tax value of the item (0 , 1, 2) taxrate = COA // taxrate of the item title = TEXT // label of the item } } } </pre>

tx_caddy_pi1.marker.sum

Property:	Data type:	Description:
marker.sum	array	<p>Configuration of marker, which are used in HTML templates for a sum.</p> <p>Example for an HTML marker:</p> <ul style="list-style-type: none"> ###SUMSUMTAXNORMAL### <p>Please inspect the items below with the TypoScript Object Browser.</p> <p>Code:</p> <pre> plugin.tx_caddy { marker { sum { labels { optionspaymentlabel = TEXT optionsshippinglabel = TEXT optionsspecialslabels = TEXT taxrate_reduced_string = COA taxrate_normal_string = COA } rates { optionspaymentsumrate = COA optionsshippingsumrate = COA optionsspecialssumrate = COA } values { optionspaymentsumgross = COA optionspaymentsumnet = COA optionspaymentsumtaxnormal = COA optionspaymentsumtaxreduced = COA optionsshippingsumgross = COA optionsshippingsumnet = COA optionsshippingsumtaxnormal = COA optionsshippingsumtaxreduced = COA optionsspecialssumgross = COA optionsspecialssumnet = COA optionsspecialssumtaxnormal = COA optionsspecialssumtaxreduced = COA sumitemsgross = COA sumitemsnet = COA sumitemstaxnormal = COA sumitemstaxreduced = COA sumoptionsgross = COA sumoptionsnet = COA sumoptionstaxnormal = COA sumoptionstaxreduced = COA sumsumgross = COA sumsumnet = COA sumsumtaxnormal = COA sumsumtaxreduced = COA sumsumtaxsum = COA } } } } </pre>

tx_caddy_pi1.options

Property:	Data type:	Description:
options	array	<p>Configuration of the options</p> <ul style="list-style-type: none"> • payment, • shipping, • special <p>You are controlling the calculating and the output in HTML, e-mail and PDF.</p> <p>Please inspect the items below with the TypoScript Object Browser.</p> <p>Example: Sorry, there isn't any example.</p> <p>Code:</p> <pre> plugin.tx_caddy { options = // payment, shipping, special options { payment = // preset, options payment { preset = 1 // default method options = // cash in advance, invoice, cash on delivery options { ... } } shipping = // preset, options shipping { preset = 1 // default method options = // maildefault, mailexpress options { ... } } specials = // preset, options specials { divider = TEXT // divider for list of options // in e-mails und PDF attachments options = // neutralpackage, addfeeforlands options { ... } } } } </pre>

tx_caddy_pi1.pdf

Property:	Data type:	Description:
pdf	array	<p>Configuration for the PDF attachments.</p> <p>Please configure it with the Constant Editor. See</p> <ul style="list-style-type: none"> "Constant Editor > PDF" on page 29 above. <p>Please inspect the items below with the TypoScript Object Browser.</p> <p>Code:</p> <pre> plugin.tx_caddy { pdf { deliveryorder { filename = COA // label - delivery order number - date . pdf content { address = // deliveryorder, invoice caddy = // body date = // body numbers = // deliveryorder, invoice and order } } invoice { filename = COA // label - invoicenumber - date . pdf content { address = // deliveryorder, invoice caddy = // body date = // body numbers = // deliveryorder, invoice and order termOfCredit = // body } } revocation { filename = COA // label - invoicenumber - date . pdf content { address = // invoice date = // body additionaltextblocks = // revocation } } terms { filename = COA // label - invoicenumber - date . pdf content { address = // invoice date = // body additionaltextblocks = // terms } } } } </pre>

tx_caddy_pi1.pid

Property:	Data type:	Description:
pid	integer	<p>Id of the page with the CADDY plugin.</p> <p>Please configure it with the Constant Editor. See</p> <ul style="list-style-type: none"> "Constant Editor > PAGES" on page 29 above. <p>Example:</p> <pre> plugin.tx_caddy { pid = 8154 } </pre>

tx_caddy_pi1.pluginCheck

Property:	Data type:	Description:
pluginCheck	string	<p>For self evaluation only. Never touch it!</p> <p>Code:</p> <pre>plugin.tx_caddy { pluginCheck = dummy }</pre>

tx_caddy_pi1.settings

Property:	Data type:	Description:
settings	string	<p>Extend the database interface for special needs – usually variants.</p> <p>Example:</p> <pre>plugin.tx_caddy { db { sql { ... -> See tx_caddy_pi1.db.sql on page 48 above } } settings { ... -> See Error: Reference source not found on page Error: Reference source not found Error: Reference source not found } }</pre>

tx_caddy_pi1.symbols

Property:	Data type:	Description:
symbols	array	<p>Configuration of symbols.</p> <p>Please configure it with the Constant Editor. See</p> <ul style="list-style-type: none"> "Constant Editor > SYMBOLS AND SERVICE ATTRIBUTES" on page 31 above. <p>Example:</p> <pre>plugin.tx_caddy { symbols { decimal = 2 dec_point = , thousands_sep = . currencySymbol = &euro; currencySymbolBeforePrice = 0 quantitySymbol = St. service_attribute_1_symbol = service_attribute_2_symbol = service_attribute_3_symbol = } }</pre>

tx_caddy_pi1.templates

Property:	Data type:	Description:
templates	array	<p>Configuration of templating.</p> <p>Please configure it with the Constant Editor. See</p> <ul style="list-style-type: none"> "Constant Editor > TEMPLATES" on page 32 above. <p>Code:</p> <pre> plugin.tx_caddy { templates { e-mail { file = EXT:caddy/files/templates/caddy.html marker { // all, item } table { // border, cellpadding, cellspacing } } html { caddy { // pi1: file, marker, table } caddymini { // pi3: file, marker } caddysum { // pi2: file, marker } } pdf { deliveryorder { // file, marker, table } invoice { // file, marker, table } revocation { // file, marker, table } terms { // file, marker, table } } } } </pre>

tx_caddy_pi1.tax

Property:	Data type:	Description:
tax	array	<p>Configuration for tax calculating.</p> <p>Please configure it with the Constant Editor. See</p> <ul style="list-style-type: none"> "Constant Editor > TAX" on page 32 above. <p>Example:</p> <pre> plugin.tx_caddy { tax { reduced = 7 reducedCalc = 0.07 normal = 19 normalCalc = 0.19 } } </pre>

tx_caddy_pi1._HTMLMARKER

Property:	Data type:	Description:
_HTMLMARKER	array	<p>Feel free to add any marker to the templates for HTML, the e-mail and the PDF attachments.</p> <p>The property from below will replace ###_HTMLMARKER_LINTOSHOP###.</p> <p>Please inspect the code from below with the TypoScript Object Browser.</p> <p>Example:</p> <pre> plugin.tx_caddy { _HTMLMARKER { - linktoshop = COA linktoshop { // Label 10 = TEXT 10 { ... } // Icon 20 = IMAGE 20 { ... } wrap = <div class="linktoshop caddylinktoshop"> </div> } } } </pre>

tx_caddy_pi1._CSS_DEFAULT_STYLE

Property:	Data type:	Description:
_CSS_DEFAULT_STYLE	string	<p>Inline CSS. An inline CSS can configured while runtime.</p> <p>This property is configured only, if a CSS static template is included. See</p> <ul style="list-style-type: none"> "Static Templates > CSS" on page 25 above. <p>Please inspect the code from below with the TypoScript Object Browser.</p> <p>Code:</p> <pre> plugin.tx_caddy_pi1 { _CSS_DEFAULT_STYLE (form.caddy { text-align: right; } form.caddy button.submit img { position: relative; top: .1em; } ...) } </pre>

tx_caddy_pi1.powermail

Property:	Data type:	Description:
powermail	array	<p>Configuration for controlling powermail.</p> <p>This property is configured only, if a powermail static template is included. See</p> <ul style="list-style-type: none"> "Static Templates > Powermail (obligated)" on page 25 above. <p>The configuration depends on the powermail version 1.x and 2.x</p> <p>The example below corresponds with the fluid marker in the powermail form.</p> <p>Code (powermail 2.x):</p> <pre> plugin.tx_caddy { powermail { caddy = USER caddy { // The caddy content } caddydeliveryordernumber = USER caddydeliveryordernumber { // Delivery order number } caddyinvoicenummer = USER caddyinvoicenummer { // Invoice number } caddyordernumber = USER caddyordernumber { // Order number } clearcaddyssession = USER clearcaddyssession { // Clear the caddy session } } } </pre>

tx_caddy_pi2

Frontend and Plugin

- See "Plugins > Sum Gross" on page 17 above.

TypoScript

Property:	Data type:	Description:
plugin.tx_caddy_pi2	array	<p>Configuration for the Caddy Sum Gross plugin</p> <p>Code:</p> <pre> plugin.tx_caddy_pi2 { // item, marker content = content { // label, sumsumgross sum = COA sum { // label 20 = TEXT 20 { data = LLL:EXT:caddy/pi2/locallang.xml:phrasesum noTrimWrap = } // sumsumgross 30 = COA 30 { // value 10 = USER 10.userFunc = tx_caddy_userfunc->numberformat 10.userFunc { number = TEXT number.field = sumsumgross decimal = 2 dec_point = {\$plugin.caddy.main.dec_point} thousands_sep = {\$plugin.caddy.main.thousands_sep} drs = {\$plugin.caddy.debug.userfunc} } // currency 20 = TEXT 20 { value = {\$plugin.caddy.main.currencySymbol} noTrimWrap = } wrap = } } } // page id of the caddy pid = {\$plugin.caddy.pages.caddy} // e-mail, html, pdf templates = templates < plugin.tx_caddy_pi1.templates } </pre>

tx_caddy_pi3 – Overview

Frontend and Plugin

- See "Plugins > Mini Caddy" on page 16 above.

TypoScript

Property:	Data type:	Description:
plugin.tx_caddy_pi3	array	<p>Configuration for the Caddy Mini plugin</p> <p>Code:</p> <pre> plugin.tx_caddy_pi3 { // item, marker content = content { // quantity, label, gross item = COA item { ... } // quantity, label, gross sum = COA sum { ... } } // page id of the caddy pid = {\$plugin.caddy.pages.caddy} // e-mail, html, pdf templates = templates < plugin.tx_caddy_pi1.templates // linktocaddy, linktoshop _HTMLMARKER = _HTMLMARKER { ... } } </pre>

tx_caddy_pi3.content.item

Property:	Data type:	Description:
content.item	array	<p>Configuration for the items of the Caddy Mini plugin</p> <p>Code:</p> <pre> plugin.tx_caddy_pi3 { // item, marker content = content { // quantity, label, gross item = COA item { // quantity 10 = TEXT 10 { field = quantity wrap = <div class="quantity itemquantity"> </div> } // label 20 = TEXT 20 { field = label wrap = <div class="label itemlabel"> </div> } // gross 30 = COA 30 { // value 10 = USER 10.userFunc = tx_caddy_userfunc->numberformat 10.userFunc { number = TEXT number.field = gross decimal = 2 dec_point = {\$plugin.caddy.main.dec_point} thousands_sep = {\$plugin.caddy.main.thousands_sep} drs = {\$plugin.caddy.debug.userfunc} } // currency 20 = TEXT 20 { value = {\$plugin.caddy.main.currencySymbol} noTrimWrap = } wrap = <div class="gross itemgross"> </div> } wrap (<div class="item minicaddyitem"> </div> <div class="caddy_cleaner"></div>) } } } </pre>

tx_caddy_pi3.content.sum

Property:	Data type:	Description:
content.sum	array	<p>Configuration for the Sum of the Caddy Mini plugin</p> <p>Code:</p> <pre> plugin.tx_caddy_pi3 { // item, marker content = content { // quantity, label, gross sum = COA sum { // quantity 10 = TEXT 10 { field = quantity wrap = <div class="quantity sumquantity"> </div> } // label 20 = COA 20 { // in case of one item 10 = TEXT 10 { if { value = 2 isLessThan { field = quantity } } data = LLL:EXT:caddy/pi3/locallang.xml:item } // in case of more than one item 20 = TEXT 20 { if { value = 2 isLessThan { field = quantity } negate = 1 } data = LLL:EXT:caddy/pi3/locallang.xml:items } wrap = <div class="label sumlabel"> </div> } // gross 30 = COA 30 { 10 = USER 10.userFunc = tx_caddy_userfunc->numberformat 10.userFunc { number = TEXT number.field = gross decimal = 2 dec_point = {\$plugin.caddy.main.dec_point} thousands_sep = {\$plugin.caddy.main.thousands_sep} drs = {\$plugin.caddy.debug.userfunc} } 20 = TEXT 20 { value = {\$plugin.caddy.main.currencySymbol} noTrimWrap = } wrap (<div class="gross sumgross"> </div> <div class="caddy_cleaner"></div>) } wrap = <div class="sum minicaddysum"> </div> } } } </pre>

tx_caddy_pi3.pid

Property:	Data type:	Description:
pid	integer	<p>Id of the page with the Caddy plugin.</p> <p>Please configure it with the Constant Editor. See</p> <ul style="list-style-type: none"> "Constant Editor > PAGES" on page 29 above. <p>Code:</p> <pre>plugin.tx_caddy_pi3 { // page id of the caddy pid = {\$plugin.caddy.pages.caddy} }</pre>

tx_caddy_pi3.templates

Property:	Data type:	Description:
templates	array	<p>Configuration of the templates is taken from the Caddy plugin.</p> <p>Code:</p> <pre>plugin.tx_caddy_pi3 { // e-mail, html, pdf templates = templates < plugin.tx_caddy_pi1.templates }</pre>

tx_caddy_pi3._HTMLMARKER.linktocaddy

Property:	Data type:	Description:
_HTMLMARKER. linktocaddy	array	<p>Link to the page with the Caddy plugin. Link contains an image – the Caddy icon.</p> <p>Code:</p> <pre> plugin.tx_caddy_pi3 { // linktocaddy, linktoshop _HTMLMARKER = _HTMLMARKER { // label, icon. Replaces _HTMLMARKER_LINTOCADDY linktocaddy = COA linktocaddy { // Label 10 = TEXT 10 { data = LLL:EXT:caddy/pi3/locallang.xml:linktocaddy typolink { parameter = {\$plugin.caddy.pages.caddy} title { data = LLL:EXT:caddy/pi3/locallang.xml:linktocaddy } } noTrimWrap = } // Icon 20 = IMAGE 20 { file = {\$plugin.caddy.html.color.icon.caddy} altText { data = LLL:EXT:caddy/pi3/locallang.xml:caddy } titleText { data = LLL:EXT:caddy/pi3/locallang.xml:caddy } params = class="linktocaddy" imageLinkWrap = 1 imageLinkWrap { enable = 1 typolink { parameter = {\$plugin.caddy.pages.caddy} } } } } wrap = <div class="linktocaddy minicaddylinktocaddy"> </div> } } </pre>

tx_caddy_pi3._HTMLMARKER.linktoshop

Property:	Data type:	Description:
_HTMLMARKER. linktoshop	array	<p>Link to the page with the Shop plugin. Link contains an image – the Caddy icon.</p> <p>Code:</p> <pre> plugin.tx_caddy_pi3 { _HTMLMARKER { // label, icon. Replaces _HTMLMARKER_LINTOSHOP linktoshop = COA linktoshop { // Label 10 = TEXT 10 { data = LLL:EXT:caddy/pi3/locallang.xml:linktoshop typolink { parameter = {\$plugin.caddy.pages.shop} title { data = LLL:EXT:caddy/pi3/locallang.xml:linktoshop } } noTrimWrap = } // Icon 20 = IMAGE 20 { file = {\$plugin.caddy.html.color.icon.caddy} altText { data = LLL:EXT:caddy/pi3/locallang.xml:shop } titleText { data = LLL:EXT:caddy/pi3/locallang.xml:shop } params = class="linktoshop" imageLinkWrap = 1 imageLinkWrap { enable = 1 typolink { parameter = {\$plugin.caddy.pages.shop} } } } } wrap = <div class="linktoshop minicaddylinktoshop"> </div> } } </pre>

Debugging

If you have any unexpected result, you can inspect Caddy in the TYPO3 backend with the DRS – the Development Reporting System – and in the TYPO3 frontend with Firebug.

DRS – Development Reporting System

Inspect work-flow, session data and errors

The DRS – Development Reporting System – enables you

- to follow the work-flow of Caddy,
- to inspect alternate configuration possibilities,
- to inspect session data and their corresponding cObject fields,
- to inspect improper configuration and last but not least
- the DRS prompts warnings and errors

Enabling by the Extension Manager

Enable the DRS – Development Reporting System – for all caddy plugins by the Extension Manager. If you have more than one plugin on a page (like a caddy and a mini caddy), you will the report for each item. It is recommended to enable the DRS by the Plugin / Flexform (see below).

BE AWARE: This feature can't debug user functions. For a complete report, you have to enable this feature by the Constant Editor (see below).

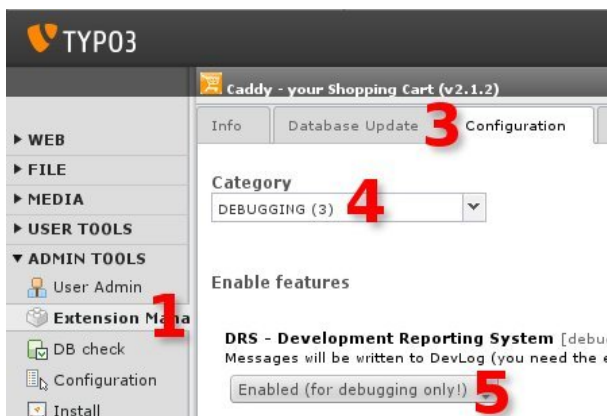


Illustration 40: Enable the DRS by the Extension Manager

1. Module: Admin Tools Extension Manager
2. Edit Area Select the Caddy extension
3. Tab [Configuration]
4. Category: [Debugging]
5. DRS – Development Reporting System: [Enabled]
6. [Update]

Enabling by the Plugin / Flexform (recommended)

Enable the DRS – Development Reporting System – for the current plugin only by the plugin / flexform.

BE AWARE: This feature can't debug user functions. For a complete report, you have to enable this feature by the Constant Editor (see below).

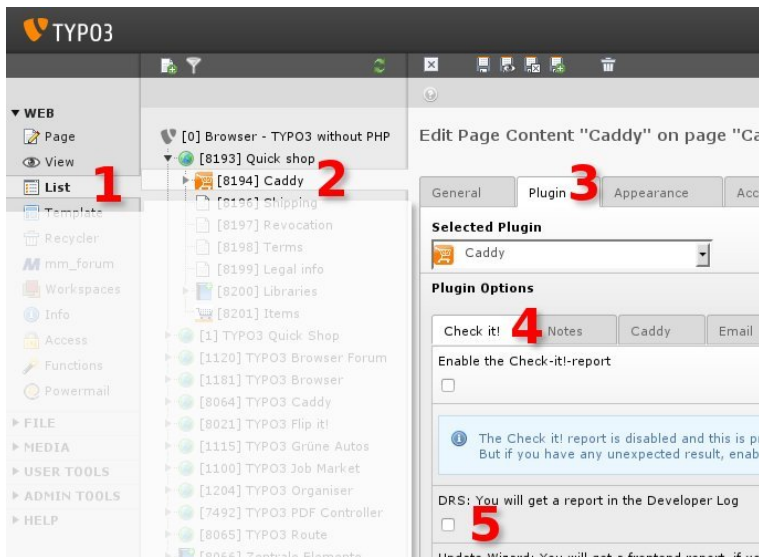


Illustration 41: Enable the DRS by the Plugin / Flexform

1. Module: Web List
2. Page tree Page with the Caddy plugin (here: [8194] Caddy)
3. Edit Area Caddy plugin: tab [Plugin]
4. Tab [Check it!]
5. DRS: You will get a report ... [X]

Enabling by the Constant Editor (recommended)

Enable the DRS – Development Reporting System – for user functions.

BE AWARE: This feature debugs user functions only. For a complete report, you have to enable the DRS by the plugin / flexform or the extension manager too.

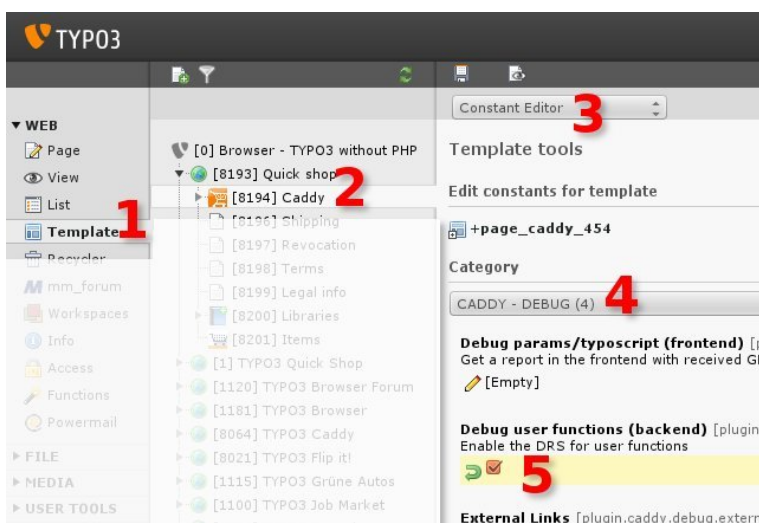


Illustration 42: Enable the DRS by the Plugin / Flexform

1. Module: Web Template
2. Page tree Page with the Caddy plugin (here: [8194] Caddy)
3. Edit Area [Constant Editor]
4. category [CADDY - DEBUG]
5. Debug user functions (backend) [X]

The DRS report

Precondition

You need the extension

- Developer Log (devlog)

Report

	Date	Severity	Extension	Message	Called from	Page	User	Extra data
WEB	24-09-13 00:40:20	i	caddy	[INFO/DRS] The DRS - Development Reporting System is enabled by the flexform (frontend mode).	class.tx_caddy_drs.php, line 180			Warenkorb
FILE	24-09-13 00:40:20	i	caddy	[INFO/DRS] "Warenkorb" (pid: 8148, uid: 2505)	class.tx_caddy_drs.php, line 185			Warenkorb
MEDIA	24-09-13 00:40:20	i	caddy	[INFO/INIT] pid is taken from main.pid and is set to 8148	class.tx_caddy_pi1.php, line 797			Warenkorb
ADMIN TOOLS	24-09-13 00:40:20	w	caddy	[WARN/INIT] No access: current IP doesn't match the list of allowed IP. Result won't prompt to the frontend.	class.tx_caddy_pi1.php, line 473			Warenkorb
User Admin	24-09-13 00:40:20	i	caddy	[INFO/POWERMAIL] Powermail version is 2.0.9 (internal 2000009)	class.tx_caddy_powermail.php, line 828			Warenkorb
Extension Manager	24-09-13 00:40:20	i	caddy	[INFO/SQL] SELECT * FROM tt_content WHERE pid = 8148 AND hidden = 0 AND deleted = 0 AND list_type = 'powermail_pi1' ORDER BY sorting LIMIT 1	class.tx_caddy_powermail.php, line 943			Warenkorb
DB check	24-09-13 00:40:20	i	caddy	[INFO/POWERMAIL] powermail.uid: "2507"	class.tx_caddy_powermail.php, line 846			Warenkorb
Configuration	24-09-13 00:40:20	i	caddy	[INFO/POWERMAIL] powermail.title: "[Powermail für Quickshop]"	class.tx_caddy_powermail.php, line 848			Warenkorb
Install	24-09-13 00:40:20	i	caddy	[INFO/POWERMAIL] powermail.confirm: "6"	class.tx_caddy_powermail.php, line 850			Warenkorb
Log	24-09-13 00:40:20	i	caddy	[INFO/INIT] tx_quickshop_products is an element of the \$TCA.	class.tx_caddy_pi1.php, line 516			Warenkorb
Reports	24-09-13 00:40:20	i	caddy	[INFO/INIT] Delivery order number from registry: 19	class.tx_caddy_pi1.php, line 588			Warenkorb
ExtDevEval	24-09-13 00:40:20	i	caddy	[INFO/INIT] Order number from registry: 19	class.tx_caddy_pi1.php, line 590			Warenkorb
Developer Log	24-09-13 00:40:20	i	caddy	[INFO/INIT] Invoice number from registry: 19	class.tx_caddy_pi1.php, line 592			Warenkorb
Scheduler	24-09-13 00:40:20	i	caddy	[INFO/INIT] Current delivery order number: 236	class.tx_caddy_pi1.php, line 594			Warenkorb
T3 jQuery	24-09-13 00:40:20	i	caddy	[INFO/INIT] Current invoice number: 103	class.tx_caddy_pi1.php, line 596			Warenkorb
UTF8 convert	24-09-13 00:40:20	i	caddy	[INFO/INIT] Current order number: 3992	class.tx_caddy_pi1.php, line 598			Warenkorb
HELP	24-09-13 00:40:20	i	caddy	[INFO/INIT] Current numbers are stored in the session.	class.tx_caddy_pi1.php, line 632			Warenkorb
	24-09-13 00:40:20	i	caddy	[INFO/VARIANTS] settings.variant is empty: Variants aren't handled.	class.tx_caddy_session.php, line 2113			Warenkorb
	24-09-13 00:40:20	i	caddy	[INFO/CALC] Case: add an item	class.tx_caddy_session.php, line 1102			Warenkorb
	24-09-13 00:40:20	i	caddy	[INFO/CALC] Current item (Geheime Bilder: 4) hasn't any maximum limit. Maximum limit won't checked.	class.tx_caddy_session.php, line 1145			Warenkorb
	24-09-13 00:40:20	i	caddy	[INFO/CALC] Current item (Geheime Bilder: 4) hasn't any minimum limit. Minimum limit won't checked.	class.tx_caddy_session.php, line 1211			Warenkorb
	24-09-13 00:40:20	i	caddy	[INFO/CALC] Quantity of current product is:	class.tx_caddy_session.php, line 1499			Warenkorb
	24-09-13 00:40:20	i	caddy	[INFO/CALC] Quantity: all #0 + add #1 = sum #1	class.tx_caddy_session.php, line 1499			Warenkorb

Illustration 43: The DRS Report

Get the report:

- Module > Admin Tools: Development Log

Updates

From Powermail 1.x to Powermail 2.x

Caddy supports both versions: Powermail 1.x and Powermail 2.x.

If you have made an update from 1.x to 2.x, please evaluate your Caddy configuration with the tab [Check it!]. See

- "Plugin / Flexform > Check it!" on page 19 above.
- "Plugin / Flexform > Billing Address" on page 22 above.
- "Plugin / Flexform > Delivery Address" on page 23 above.

From wt_cart 1.x to Caddy 2.x

- Please install Caddy with one of the both one-click-installer – Organiser or Quick Shop. See
 - "One-Click-Installation (recommended)" on page 12 above.
- Than move the caddy page to the page with the wt_cart plugin.
- Adapt the Caddy configuration to your needs.
- Remove wt_cart.

Packages for Sponsors

Your donation and sponsoring is welcome.

Please order a package for sponsors at

- English: <http://typo3-caddy.de/en/sponsoring/>
- German: <http://typo3-caddy.de/sponsoring/>



Fun – 50 EUR



You like to treat the TYPO3 Caddy project.

You don't expect any equivalent. No logo. No service.

Throw 50 EUR or any other old sum into the cash box.

You will get a receive. You can set the amount off against tax liability.

Logo – 150 EUR



You are treating the TYPO3 Caddy project and you like to be a guide for others.

Your logo will appear in the manual, at the website, in the plugin and in the extension manager of the Caddy.

The period of validity is limited to one year.

You will get a receive. You can set the amount off against tax liability.

You can pay more than 150 EUR.

Patron – 500 EUR



You are the patron. You are treating TYPO3 Caddy project and you like to pay the developer more than best for two hours service.

Your logo will appear in the manual, at the website, in the plugin and in the extension manager of Caddy.

You can book the developer of Caddy for two hours.

The period of validity for both – the logo and the two hours – is limited to one year. If you don't use the two hours in this period, you will lost it.

You will get a receive. You can set the amount off against tax liability.

You can pay more than 500 EUR.

Credits

Made in Nature

The German TYPO3 agency Made in Nature has contracted

- PDF attachments for the delivery order and the invoice¹
- a part of the fork.

Website

- <http://made-in-nature.de/>

wt_cart Developer Team

Thanks to the wt_cart developer team for the great wt_cart extension, which was the base for Caddy:

- extco.de UG, Daniel Lorenz (Leader) – www.extco.de
- TRITUM GmbH, Björn Jacob (Co-Leader), Sebastian Wagner, Ralf Zimmermann – www.tritum.de
- in2code, Alexander Kellner – www.in2code.de

¹ It is realised with wt_cart 1.4.x in 2011 but never published :(

Known Bugs

TCPDF-Error

Prompt

TCPDF ERROR: This document (fileadmin/files/my.pdf) probably uses a compression technique which is not supported by the free parser shipped with FPDF.

Reason and Solution

Probably the version of your PDF file is the reason: Version must be lower than 1.5.

More Bugs

Please refer to

- <http://forge.typo3.org/projects/extension-caddy/issues>

Further Information

Forum

Caddy has a website and a forum. Please visit:

- <http://typo3-caddy.de/en/>
- <http://forum.typo3-caddy.de/>

TYPO3 forge

- <http://forge.typo3.org/projects/extension-caddy>

History of Caddy

Fork of wt_cart

Caddy was a fork of wt_cart version 1.4.6.

Copyright wt_cart 1.4.6: 2010-2012, wt_cart Development Team. See "wt_cart Developer Team" on page 69 above.

Why do I forked wt_cart?

A generic shopping cart for TYPO3 is a good idea.

And an improved idea is

- to fix bugs speedy,
- to develop the cart permanently, and
- to let profit the TYPO3 community of contracted features.

Caddy is the attempt, to keep the promise from above.

Caddy versus wt_cart

Having regard to plugin 1

skill	caddy	wt_cart ¹
Features		
Items from your database	yes	yes
Min/Max limit for each item	yes	yes
Min/Max limit for all items	yes	no
Delivery costs depending on <ul style="list-style-type: none"> amount size weight 	yes yes yes	yes yes yes
PDF attachments ²	yes	no
Order-/Invoice-numbers without gap	yes	no
Variants ³	yes	buggy
Interface for online payment	no	no
Configuration properties		
Plugin / Flexform ⁴ , Constant Editor	153 properties	42 properties
Database		
Orders are stored in an own database	yes	no
Archive audit proof ⁵	yes	no
Debugging and Development		
Developer versions	http://forge.typo3.org/.../caddy	https://github.com/extcode/wt_cart
GET/POST parameter	yes	yes
TypoScript	yes	yes
User functions	yes	no
Work flow	nearby 300 prompts ⁶	0 prompts
Evaluation		
by the Extension Manager	yes	no
by the plugin / flexform	yes	no
Icons		
Cart	1 icon / 40 colours	no icon
Delete	1 icon / 40 colours	1 icon / 1 colour
Installation		
Out of the box / one click installation	yes	stopped with Quick Shop installer 2.x
Manual		
Illustrations	42	8
Pages	77	30
Powermail		
Compatible with 1.x	yes	yes
Compatible with 2.x	yes	no
Independent on Powermail hooks	yes	no
Support		
Website	http://typo3-caddy.de/	http://wt-cart.com/
Forum / Support	http://forum.typo3-caddy.de/	http://snippets.wt-cart.com/

1 Version 1.4.x

2 Delivery order, invoice, terms & conditions

3 One dimension

4 Plugin 1

5 Audit proof in German: revisionssicher

6 supported by the DRS – Development Reporting System

Other extensions published by Die Netzmacher

-  +AOE Linkhandler Configurator: Configure the AOE linkhandler supported by userinterfaces. Out-of-the-box templates for cal, org, tt_news and tt_products. Don't edit page TSconfig any longer.
<http://typo3.org/extensions/repository/view/linkhandlerconf/>
-  autositemap: A smart site-map optimised for the footer. It groups menus in columns. Great menus will get two columns. Configuration is based on TypoScript HMENU.
<http://typo3.org/extensions/repository/view/autositemap/>
-  Browser – TYPO3 without PHP. Develop your TYPO3 extension 8 times faster! You need one line TypoScript for a result list with a search form, a record browser and an index browser. Images are wrapped self-acting. SEO, Search Engine Optimization, is integrated. <http://typo3.org/extensions/repository/view/browser/>
-  Caddy – the TYPO3 shopping cart. You can use it for your own needs. You need a database with products or any other items only. You can install Caddy out of the box with one mouse click – see Quick Shop below.
<http://typo3.org/extensions/repository/view/caddy/>
-  Flip it! offers lovely and smooth page flip transitions. It enables you to run over pages in PDF documents like in a real magazine. It is based on flash. Flipt it! can convert PDF documents to swf files automatically.
<http://typo3.org/extensions/repository/view/flipit/>
-  Green Cars (Grüne Autos) - Database optimized for ecological cars. It is a case study and demonstrates, how to get a complex database with the browser (see above) in three hours only.
http://typo3.org/extensions/repository/view/green_cars/
-  Jobmarket is a catalogue with job offers. Views, the a-z-browser, the page-browser, the search, social bookmarks and a lot of other stuff can configured by the Browser plugin with the mouse.
http://typo3.org/extensions/repository/view/job_market/
-  Organiser – TYPO3 for the lobby and the organisers. Handle news, events, staff, headquarters, locations, workshops and a calendar with one extension. Sell online tickets! Install the Organiser with one mouse click!
<http://typo3-organiser.de/>
-  Quick Shop – the fastest shop in the history of TYPO3. Install it with one mouse click! Quick Shop is based on the browser (see above) and powermail.
<http://typo3-quick-shop.de/>
-  PDF Controller: Easy to install. Add to your HTML page the PDF-controller-button. Link from the button to the controller. Adjust the controller by mouseclicks. The PDF Controller supports CSS 3.
<http://typo3-pdfcontroller.de/>
-  Radial Search (German: Umkreissuche) for your TYPO3 database. Your data must have a latitude and a longitude. The Browser – TYPO3 without PHP – has it integrated.
<http://typo3.org/extensions/repository/view/radialsearch/>
-  Route – Publish your routes with GoogleMaps or OpenStreetMap. Routes have points of interest (POI). You can categorise and filter both: routes and POI. Address data can geocoded automatically.
<http://typo3.org/extensions/repository/view/route/>
-  seo_dynamic_tag: Search Engine Optimization for the title tag, the canonical tag and the meta tags author, description and keywords. Configuration by a user interface.
http://typo3.org/extensions/repository/view/seo_dynamic_tag/
-  TSconfig Pages and Users by extManager (extkey: tsconf): Configure the the eight most commonly used TSconfig porperties with the mouse - like page tree uids, activated extended view, activated clipboard, ...
<http://typo3.org/extensions/repository/view/tsconf/>
-  tt_news select configuration (extkey: ttnews_selectconf) enables to select tt_news by any SQL clause. The extension adds an andWhere clause to the SQL query of the tt_news plugin.
http://typo3.org/extensions/repository/view/ttnews_selectconf/
-  Wine Catalogue provides a data base for wine with regions, wineries, styles, variety and ageing among others. It is localized. English, German and Spanish ist of the box. Wine based on the extension browser (see above).
<http://typo3.org/extensions/repository/view/wine/>

Alphabetical Index

Attachments	Debugging	27	Amount	25	CSS	25
Corporate design	Decimal	26	Marker	49	Ordering	25
Audit Proof	Due date	26	Item	49	Powermail	25
Database	GET-/POST-Parameter	27	Sum	50	Symbols	
Disabling	Icons	28	Numbers		Quantity	31
Billing	Service attribute	27	Reset	46	Service Attribute	31
Address	Storage folder	26	Sys_registry	46	Templates	
Caddy	Table width	28	Pages		Caddy	32
Invoice address	Tax	32	Caddy	29	Caddymini	32
Powermail	Templates	32	Caddymini	29	Caddysum	32
Caddy	Thousands separator	26	Revocation	29	Delivery order	32
Address	User functions	27	Shop	29	E-mail	32
Delivery	Corporate Design		Terms	29	Invoice	32
Invoice	Button	28	PDF	52	Paths	32
Audit Proof	Colour schemes	33	Address		TypoScript	
Calculating	Colours	28	Delivery	39	Decimal	53
Check-it!	Icons	28, 34	Invoice	22, 39	Deliveryorder	52
Condition	Ready-to-use	35	Attachments	36	Euro	53
Database	Upload	36	Background	38	Invoice	52
Debug	CSS		Caddy	29, 39	Labels	50
Delivery	Static Templates	25	Colour	31	Options	51
Address	Database	45, 48	Date	29, 39	Payment	51
Order number	Debugging		Delivery		Powermail	56
Evaluation	DRS	64	Address	39	Rates	50
Invoice	Extension Manager	64	Font colour	38	Revocation	52
Billing address	GET-/POST-Parameter	27	Font family	29, 38	Shipping	51
Invoice number	Plugin	65	Font stretching	38	Specials	51
Maximum order quantity ...	Report	66	Invoice	30	Sql	48, 53
Minimum order quantity	Session data	64	Address	39	Symbols	53
Numbers	User functions	27, 65	Numbers	39	Tax	54
Reset	Delivery		Order number	30	Templates	
Order number	Address	39	Paths	37	Caddy	54
Pid	Caddy	23	Power of revocation	31	Caddymini	54
Plugin	Plugin	23	Power of Revocation	40	Caddysum	54
Sql	Powermail	23	Term of credit	40	Deliveryorder	54
Static Templates	Templates	32	Terms & Conditions	40	E-mail	54
Sys_registry	DRS		TypoScript	52	Html	54
Templates	Debugging	64	Plugin		Invoice	54
Variants	E-mail		Address		Revocation	54
Work-Flow	Delivery Order	21	Delivery	23	Terms	54
Caddy Mini	Invoice	21	Invoice	22	Terms	52
Content	Plugin	21	Caddy	15, 20	Values	50
Item	Power of Revocation	21	Caddy: Mini	16	Variants	48, 53
Plugin	Templates	32	Caddy: sum gross	17	_CSS_DEFAULT_STYLE	55
Sum	Terms & Conditions	21	Debugging	65	_HTMLMARKER	55
Templates	Evaluation	19	E-mail	21	Update	67
_HTMLMARKER	Extension Manager		PDF	23	Powermail	67
Caddy Sum Gross	Debugging	64	Powermail		Wt_cart	67
Content	Get-/Post-parameter	49	Address		User functions	
Plugin	Installation		Billing	22	Debugging	65
Sum	Manual	13	Delivery	23	Variants	
Templates	One-Click	12	Invoice	22	Sql	48, 53
Colour Schemes	Organiser Installer1	13	Evaluation	19	Work-flow	
Attachments	Quick Shop Installer2	13	Static Templates	25	Caddy	42
E-mail	Invoice		TypoScript	56	Powermail	42
HTML	Address	39	Update	67	Wt_cart	
Icons	Attachments	39	Work-flow	42	Update	67
PDF	Billing address	22	Session data		Versus Caddy	72
Constant Editor	Caddy	22	Debugging	64	_CSS_DEFAULT_STYLE	55
Button	Plugin	22	Sponsoring	68	_HTMLMARKER	55
Colours	Powermail	22	Package	68	Caddy Mini	62
Currency symbol	Templates	32	Static Templates			
Database	Localization		Caddy	25		

Illustration Index

Illustration 1: Caddy with an item and the order form	4	Illustration 21: Plugin [Billing Address]	22
Illustration 2: Enabled confirmation mode: The customer has to confirm his data.	5	Illustration 22: Plugin [Delivery Address]	23
Illustration 3: Confirmation e-mail	5	Illustration 23: Plugin [Delivery Address]	23
Illustration 4: The caddy e-mail attachments (here: Quick Shop) ..	6	Illustration 24: Plugin [Sponsors]	24
Illustration 5: Caddy colour schemes	7	Illustration 25: Plugin [Help!]	24
Illustration 6: The caddy plugin.	8	Illustration 26: Caddy blue	33
Illustration 7: A caddy order record	9	Illustration 27: Caddy green	33
Illustration 8: The Quick Shop Installer enables you, to install Caddy ready-to-use with one mouse click.	9	Illustration 28: Caddy orange	33
Illustration 9: Quick Shop Installer - Installation with one mouse click	13	Illustration 29: Caddy red	33
Illustration 10: Plugin Caddy (frontend)	15	Illustration 30: Caddy blue	34
Illustration 11: Plugin Caddy (backend)	15	Illustration 31: Icon path in the Constant Editor	34
Illustration 12: Plugin Caddy Mini (frontend)	16	Illustration 32: caddy blue/red	34
Illustration 13: Plugin Caddy Mini (backend)	16	Illustration 33: Caddy icons	35
Illustration 14: Plugin Caddy Mini (frontend)	17	Illustration 34: Your (1) draft will merged	36
Illustration 15: Plugin Caddy Mini (backend)	17	Illustration 35: ... with (2) Caddy content	36
Illustration 16: Sample with Quick Shop	18	Illustration 36: ... to the (3) attachment.	36
Illustration 17: Plugin [Check it!]	19	Illustration 37: [PDF] tab of the Caddy plugin	37
Illustration 18: Plugin [Notes]	19	Illustration 38: Properties of a PDF attachment (here: invoice) ...	38
Illustration 19: Plugin [Caddy]	20	Illustration 39: Audit proof by the Extension Manager	45
Illustration 20: Plugin [Email]	21	Illustration 40: Enable the DRS by the Extension Manager	64
		Illustration 41: Enable the DRS by the Plugin / Flexform	65
		Illustration 42: Enable the DRS by the Plugin / Flexform	65
		Illustration 43: The DRS Report	66

Change Log

- 3.0.0 **Major Feature**
 - * #53360: Fit it into TYPO3 6.x
- 2.1.5 **Improvement**
 - * #i0032: Searchfields tca.php
 - * #i0031: Update the manual. Here: wt_cart
- 2.1.4 **Improvement**
 - * #52313: Try to externalise tcpdf
- 2.1.3 **Manual**
- 2.1.2 **Improvement**
 - * #i0029: Setup: minimum groos sum for items
 - * #i0028: CSS
 - * #i0027: pdf filename %H%M -> %H%M%S
 - * #52165: Update tcpdf: 5.9.149 -> 6.0.031
- 2.1.1 **Improvement**
 - * #i0026: additional template caddy_wo_options.html
 - * #i0025: DRS
- 2.1.0 **Features**
 - * #51916: Plugin for amount only
- 2.0.11 **Improvement**
 - * #i0024: db.sql**Bugfixing**
 - * #i0023: \$this->extKey -> \$_EXTKEY
- 2.0.10 **Bugfixing**
 - * #50045: rounding error
- 2.0.9 **Bugfixing**
 - * #i0019: UTF8-Bug in Powermail 2
- 2.0.8 **Features**
 - * #i0022: Unique file names
 - * #i0021: Term of Credit**Improvement**
 - * #i0020: Constant Editor Labelling**Bugfixing**
 - * #49431: misspelling nachnahme
 - * #49427: misspelling constants
 - * #49428: Localisation dependency (isn't tested in not localised context)
 - * #49430: Check TypoScript Case
- 2.0.7 **Features**
 - * #i0018: * New user functions
 - * calcDueDate
 - * calcMultiply
 - * #i0016: * Improved caddy table structur**Improvement**
 - * #i0017: * Shipping tax
 - * Moved from normal to reduced
- 2.0.6 **Improvement**
 - * #i0015: * Overwrite existing files!
- 2.0.5 **Bugfix**
 - * #i0014: * Unproper TypoScript path to HTML table properties for PDF files
 - * #i0013: * If customer doesn't get any invoice and any deliveryorder, HTML template for caddy isn't loaded. This causes a bug, while sending the confirmation e-mail. Workflow:
 - * Respect HTML template in case of all PDF files.
- 2.0.4 **Bugfix**

- * #i0012: * foreach() : Invalid argument in line 607
- * foreach() : Invalid argument in line 171

2.0.3 **Improvements**

- * #i0006: HTML mini caddy
- * #i0005: TypoScript is cleaned up

2.0.2 **Improvement**

- * #i0004: Rendering of option lists
- * #i0003: Calculation of options costs

2.0.1 **Feature**

- * #i0002: Power of Revocation

Bugfix

- * #i0001: unproper locallang.xml value in userfunc

2.0.0 **Features**

- * #45954: Database
- * #45915: Powermail Controller
- * #45863: Icons
- * #45808: PDF attachments: Bill and Shipping
- * #45967: Attachment Invoice
- * #45968: Attachment Delivery Order
- * #45969: Attachment Terms & Conditions

FORK

- * #45792: DRS - Development Reporting System
- * #45775: Fork from wt_cart 1.4.6
- * #45797: Consolidating code
- * #45783: Proper Flexforms
- * #45781: Proper extension manager
- * #45776: Move all labels from wt_cart to caddy