Actions, filters, triggers

Implementing deprecated filters

If you for some strange reason would like to implement filters that has been used in the old plugin, implementation should look like this. In this live example, you actually execute the deprecated filter before executing the up-to-date one. Doing this, you get the chance to inject old values on the way through the new filter.

```
$deprecatedStoreId = WordPress::applyFiltersDeprecated('set_storeid', null);
$storeId = (int)WordPress::applyFilters('setStoreId', $deprecatedStoreId);
```

Implemented and available filters

This plugin offers a bunch of internal filters and actions. Below is a list of them.

Filter name	Type	In	Return	Description
~~rbwc_admin_dynamic_content~~	~~filter~~	~~$content $current_section~~	~~$yourContent~~	~~Renders dynamic content for each section. If something specific, that is not included in the form fields, this could be used and will be appended in the~~ of each section.
rbwc_order_status_update	action	$orderId $newStatusSlug	*Silent*	Preferred as an action to fire up more features during callback status changes and manually updated statuses in the order admin. Will be used to run payment_completed() during finalized orders. If further actions is required, we or a developer can make use of this action. resurs_bank_order_status_update mentioned here, should be fired up separately in the same code area.
rbwc_get_custom_form_fields	filter	$currentArray	$yourArray	If settings are dependent of other activated options in the configuration, this filter should be used. The plugin use this internally, to enable/disable the developer-options tab. For a code example, take a look at this commit.
rbwc_get_plugin_information	filter	$content	$yourContent	Content added to this section will be shown in the information/support, as a part of the "version view". If you for example need to show openssl releas other vital data for your platform, you should add it here. **Example**: `[ 'OpenSSL version' => defined('OPENSSL_VERSION_TEXT') ? OPENSSL_VERSION_TEXT : '' ]`
rbwc_can_display_order_info_after_details	filter	boolean	boolean	rbwc_can_display_order_info_after_<WHERE> The can_display-trio is used to disable each payment info section in the admin view. The details section also has a separate action that triggers inste standard action from woocommerce. That action delivers all payment information required to recreate the "bluebox information" from v2.x.
rbwc_show_order_details	action	$orderData	**Is echo**	This action is executed right after the **rbwc_can_display_order_info_after_details**-filter. By disabling the three can_display_order_info and execut hook with a full order representation, it is possible to create your own view right after the order details in the admin. This is pushed in here, in case th requires a different view from this plugin.

In the rbwc_get_plugin_information row:

Plugin-version	0.0.1.0
Composer-version	0.0.1.0
PHP-Version	7.2.31-1+ubuntu18.04.1+deb.sury.org+1

rbwc_can_display_order_info_after_billing	filter	boolean	boolean	rbwc_can_display_order_info_after_<WHERE> The can_display-trio is used to disable each payment info section in the admin view.
rbwc_can_display_order_info_after_shipping	filter	boolean	boolean	rbwc_can_display_order_info_after_<WHERE> The can_display-trio is used to disable each payment info section in the admin view.
rbwc_get_payment_field_size	filter	$size, $fieldName	integer	If you believe you need a shorter or longer value for the default payment field lenght, that is set to 24, you can apply for that value here.
rbwc_prefer_article_number_sku	filter	boolean	boolean	If you have further requirements of how article numbers are fetched, you could return true for this, to make the plugin use Sku instead - if it exists. **This is not configurable through the admin interface as it is handled as a filter instead. If it is really necessary to use SKU, use this filter.**
rbwc_get_article_number	filter	$currentArticleNumber, WC_Product	$theArticleNumber	If you have further ways of fetching an article from a WC_Product object, this is where you do it.
rbwc_coupons_ex_tax	filter	boolean, WC_Coupon	boolean	For special occasions when the options should remain untouched but we still need to make exceptions for coupons.
rbwc_get_coupon_vat_pct	filter	integer	integer	Default value for vat percent in the coupon calculcation method is 0. Coupons are normally, at least in Sweden, already applied with vat.
rbwc_get_address_disabled	filter	boolean	boolean	Disable getAddress form in checkout (where it is implementable).
rbwc_js_loaders_checkout	filter	array	array	List of javascripts (frameworks?) that should be loaded during RCO.
rbwc_get_shipping_name	filter	string	string	Generates own article number for shipping if the default "shipping" is not preferred, or if another translation is needed.
rbwc_get_shipping_description	filter	string	string	Generates a orderline description for the shipping, if the preferred translation for "Shipping" is not preferred.
rbwc_get_fee_name	filter	string	string	Generates own article number for payment fees, if the default "fee" is not preferred, or if another translation is needed.
rbwc_get_fee_description	filter	string	string	Generates a orderline description for the fee, if the preferred translation for "Payment fee" is not preferred.
rbwc_set_store_id	filter	string	string	Set a store id for the payment request.
rbwc_trigger_callback	filter	array	array	External code to handle test callbacks.
rbwc_get_address_field_controller	filter	array	array	getTransformedAddressResponse, ability to use other fields than the defaults, generated by the internal filter.
rbwc_get_address_button_text	filter	string	string	Alternative naming or translation for the getAddress request button. Default is "Get address".
resurs_trigger_test_callback	filter	string	string	External trigger for the test callback. **2.X-DEPRECATED**
set_storeid	filter	string	string	Store id, old filter. **2.X-DEPRECATED**
resurs_getaddress_enabled	filter	boolean	boolean	Sets, in the old fashioned way, a state if getAddress should be enabled where it is normally disabled. **2.X-DEPRECATED**
rbwc_customer_synchronize	trigger		-	Javascript frontend trigger. Used to synchronize customer billing/shipping fields. Event contains. depending on version.

```
$('body').trigger(

'rbwc_customer_synchronize',
{

version:
2
    }
)
```

When v1 is sending data

```
{
  version: 1
}
```

resursBankRcoDataContainer Version 1 sample data:

```json
{
        "rco_customer": {
                "paymentMethod": "INVOICE",
                "customerData": {
                        "address": {
                                "firstname": "Korv",
                                "surname": "Alexandersson",
                                "address": "Glassgatan 15",
                                "addressExtra": "",
                                "postal": "41655",
                                "city": "Göteborg",
                                "countryCode": "SE",
                                "telephone": "0701122334",
                                "email": "noreply@resurs.se"
                        },
                        "delivery": {},
                        "ssn": "198001010001",
                        "paymentMethod": "INVOICE"
                }
        },
        "rco_payment": "INVOICE"
}
```

When v2 is sending data

```
{
  version: 1
}
```

resursBankRcoDataContainer Version 2 sample data (when deliveryAddress has content it looks similarly as billingAddress):

resursBankRcoDataContainer Version 2

```
{
        "rco_customer": {
                "type": "CustomerChangeEvent",
                "ts": 1632576948828,
                "source": "@rco-app",
                "phone": "0701122334",
                "email": "noreply@resurs.se",
                "billingAddress": {
                        "firstName": "Vincent",
                        "lastName": "Alexandersson",
                        "addressRow1": "Glassgatan 15",
                        "addressRow2": null,
                        "postalCode": "41655",
                        "city": "Göteborg"
                },
                "deliveryAddress": {
                        "firstName": null,
                        "lastName": null,
                        "addressRow1": null,
                        "addressRow2": null,
                        "postalCode": null,
                        "city": null
                }
        },
        "rco_payment": {
                "type": "PSP_DEBIT_CARD",
                "ts": 1632576948828,
                "source": "@rco-app",
                "method": "PSPCARD_DEBIT",
                "id": "PSPCARD_DEBIT",
                "fee": 0
        }
}
```
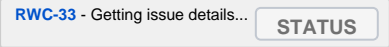
rbwc_purchase_reject	trigger	`$('body').trigger('rbwc_purchase_reject', { type: '<type>' });`	-	Then ResursCheckout framework v1 or v2 rejects a payment, either because of a failure or a denied payment (v1 only), this trigger executes and can picked up by integrated plugins to handle a rejected order.

rbwc_purchase_reject_customer_message	filter	string	string	When Resurs Bank Checkout rejects a payment due to failure/deny, this filter adds an extra note to customer in a front-end message. The default loo below:

```
$failNote = sprintf(
    __('Order was rejected by Resurs Bank with status %s.', 'trbwc'),
    $rejectType
);
```

The above message is always injected in order notices and in the default customer message. After this message has been added to the order, furthe added:

```
$failNote .= ' ' . WordPress::applyFilters(
    'purchaseRejectCustomerMessage',
    __('Please contact customer service for more information.', 'trbwc')
);
```

The above "Contact customer service"-message is added with a filter setup (which is this). If you want to change this message, you can use the men filter to update the string with your own.

rbwc_get_payment_method_icon	filter	$url, $paymentMethodData	string	When logos are active for payment methods, you can customize the look of the icon by adding this filter. The output string should be a URL to your cu logo. For each payment method. $paymentMethodData contains details about type, specificType and id that is vital for method type detection. This also deprecates **RWC-33** - Getting issue details... **STATUS** .
rbwc_get_specific_type_fields	filter	$return (array), $key	array	Customizable customer form fields for checkouts. See this section.
rbwc_get_minimum_annuity_price	filter	$integer, $customerCountry	integer	When annuity factors are handled, the minimum payment price is 150 SEK/DKK/NOK or 15 EUR for FI. If you really need to change this value to som lower, this is the filter to use.
rbwc_part_payment_string	filter	string, ['currency' => $currency, 'monthlyPrice' => $monthlyPrice, 'monthlyDuration' => $annuityDuration, 'paymentLimit' => $minimumPaymentLimit]	string	When displaying part payment information on product pages, the below string is set to be default. This filter executes an internal feature that makes it to show part payment information via a WordPress page. It also helps if you want to generate your own information without a page. The second varia filter is the the default values for duration and months. Currency is based on **get_woocommerce_currency_symbol()** and follows the "woocommerce currency ruleset". **Undocumented shortcodes** is some of the data that can be extracted from getPaymentMethods, starting with "method". For example, to get the id f current payment method in the annuity factor setup you can use "[metodId]" or "[methodDescription]". **Default text: Part pay from %s per month.**
rbwc_method_min_limit	filter	int, $paymentMethodInformation	int	If you want to limit the minimum allowed payment amount for a specific payment method, this is the one to set the data with. For example, you require payment amounts to be higher than 1000, to turn on the INVOICE, but the payment method allows payments from 500. *The minAmount of the payment method can never be exceeded.*
rbwc_method_max_limit	filter	int, $paymentMethodInformation	int	If you want to limit the maximum allowed payment amount for a specific payment method, this is the one to set the data with. For example, you require SWISH to not be higher than 10000 but the payment method allows payments up to 20000. *The maxAmount of the payment method can never be exceeded.*

part_pay ment_rea d_more_ string	filter	string	string	Text to be shown in the installment sections (Defaults to '**Read more.**', with translations in mind).
rbwc_set _curl_tim eout	filter	int	int	If you need to change the curl timeouts (for very slow connections) this gives you the value 12 by default, and this can be changed to a higher or lowe on demand. The timeouts are linked to ecom php communication with Resurs Bank.
rbwc_get _default_ country	filter	string	string	Default returns get_option('woocommerce_default_country') but can be changed on fly if this is very important for you.