

# VteSync (synchronizations)

VteSync connectors are now extendable via `SDK::setClass` so it's easier to add new functionalities or changing the field mapping.

For example, to extend the WooCommerce connector:

```
// classes are namespaced
SDK::setClass('VteSyncLib\Connector\WooCommerce', 'WooCommerceCustom',
'modules/SDK/src/CUSTOMER/WooCommerceCustom.php');
```

The extended connector:

```
<?php

// namespace the class, so it's easier to work with it
namespace VteSyncLib\Connector;

class WooCommerceCustom extends WooCommerce {

    // extend the constructor to alter the standard models
    public function __construct($config = array(), $storage = null) {
        parent::__construct($config, $storage);

        // in this case, the class is in the same folder, so no need to include it
        $this->classes['Accounts'] = array(
            'module' => 'Accounts',
            'commonClass' => 'VteSyncLib\Model\CommonRecord',
            'class' => 'VteSyncLib\Connector\WooCommerce\Model\AccountCustom'
        );
    }

    // or it's possible to redefine any existing method
}
```

And the custom model:

```

<?php

// include parent class
require_once(__DIR__.'/Account.php');

namespace VteSyncLib\Connector\WooCommerce\Model;

class AccountCustom extends Account {

    // here I redefine the mapping, removing or adding fields
    protected static $fieldMap = array(
        // WooCommerce => CommonRecord
        'email' => 'email',
        'username' => 'name',
        // 'phone' => 'phone', // THIS IS COMMENTED

        // billing address
        'address' => 'billingstreet',
        'city' => 'billingcity',
        'postcode' => 'billingpostalcode',
        'state' => 'billingstate',
        'country' => 'billingcountry',
        'companybill' => 'companybill',
        'firstnamebill' => 'firstnamebill',
        'lastnamebill' => 'lastnamebill',
        // shipping address
        'address_shipping' => 'shippingstreet',
        'city_shipping' => 'shippingcity',
        'postcode_shipping' => 'shippingpostalcode',
        'state_shipping' => 'shippingstate',
        'country_shipping' => 'shippingcountry',
        'companyship' => 'companyship',
        'firstnameship' => 'firstnameship',
        'lastnameship' => 'lastnameship',

        //
        // 'otherfield' => 'otherfield', // THIS IS CUSTOM!
        // // you probably also need to alter the VTE models, to connect this mapping to vte's one
    );

    // this function is called to prepare the array to be sent to woocommerce

```

```
public function toRawData($mode) {  
    $raw = parent::toRawData($mode);  
      
    // when sendind data to woo, hardcode this additional field:  
    $row['somefield'] = 'somevalue';  
      
    return $raw;  
}  
  
}
```

---

Revision #1

Created 2026-03-05 08:44:25 UTC by Daniele

Updated 2026-04-20 15:30:41 UTC by Daniele