
SOAP Interface, Shop4Sale

Version 5.5

Kodmyran AB



SOAP Interface Shop4Sale

Content

- Introduction..... 3
 - General Information..... 3
 - Consistency..... 4
 - Security..... 4
 - Multiple shops..... 4
 - Error Handling 4
- Approach to synchronization 5
- Retrieving information from Shop4Sale 6
- Modifying information in Shop4Sale..... 6
- Methods 7
 - Retrieving system information 7
 - getVersion 7
 - getModuleList..... 7
 - Retrieving, creating and modifying products 8
 - getProducts 8
 - setProduct 9
 - createProduct..... 9
 - Retrieving, creating and modifying customers 10
 - getCustomers 10
 - setCustomer 11
 - createCustomer..... 11
 - Retrieving, creating and modifying suppliers..... 12
 - getSuppliers..... 12
 - setSupplier..... 13
 - createSupplier 13
 - Retrieving, creating and modifying manufacturers..... 14
 - getManufacturers..... 14
 - setManufacturer..... 15
 - createManufacturer 15
 - Retrieving, creating and modifying purchase orders 16

getPurchaseOrders.....	16
setPurchaseOrder.....	17
createPurchaseOrder	17
Retrieving, creating and modifying orders.....	18
getOrders.....	18
setOrder.....	19
createOrder	19
Structures	20
Address.....	20
Associationtype	20
Customer	20
Description	21
Manufacturer	21
Module	21
Order	21
Orderrow	22
Price.....	22
Product	22
Productlocation	23
Productstructure	23
Purchaseorder	23
Purchaseorderrow.....	24
Supplier.....	24
Waybill.....	24
Terminology.....	25

Introduction

This document describes the SOAP interface available to users of the Shop4Sale e-commerce system. The interface allows external parties, be it a person or an automated system, to extract, modify and insert various data in the Shop4Sale database.

The SOAP interface is the preferred way for external integration as it provides an interface that is easily expandable, industry-standard and provides a layer of abstraction on top of the Shop4Sale base code.

All calls in this API are designed to allow for efficient two-way synchronization.

General Information

The types of objects that can be manipulated through the SOAP interface are listed below; additional objects will be made available over time.

- Customers
- Products
- Manufacturers
- Suppliers
- Orders
- Purchase orders

At the time of writing there are no methods available to create, read or modify any details regarding categories; nor can products be added to or removed from existing categories through the SOAP interface.

All of the above mentioned objects have two special attributes that may be of use when performing some form of external synchronization. These fields are called sync1 and sync2 and their function is the same across all objects. Consider that these two fields are shared; if multiple synchronization methods are used simultaneously they may alter the fields independently of each other. This may in turn cause confusion and misbehavior in other modules. The general recommendation is to avoid multiple synchronization methods from being used at the same time, in the case of synchronization modules within the Shop4Sale backend-code this is strictly enforced by the main classes – but for external interfaces, such as SOAP, this enforcement can no longer be made.

Field	Purpose, Description
Sync1	<p>The sync1 field is a field that most commonly is used to hold a reference to a remote object. This field is indexed uniquely in the database preventing any duplicates within its class (e.g. if this was an order number of some kind its uniqueness would be enforced among all orders).</p> <p>It typically contains a user id, product number, order number, supplier identity etc. that maps to the remote system.</p> <p>If this field has not been explicitly set it will always be set to null, but can be searched for in all query functions by requesting the sync1 field with an empty string.</p> <p>The sync1 field is a character string comprised of 40 characters.</p>

Sync2	<p>Is a field similar to sync1, but differs in that it is indexed non-uniquely. It can be used for many purposes and is searchable with the same criteria as sync1. A common use for this field is to store specific flags that are used in the synchronization with external systems.</p> <p>The sync2 field is a character string comprised of 40 characters.</p>
-------	---

Consistency

All objects within Shop4Sale are designed such that a number of getter/setter methods are called, and finally a commit operation is performed on the object to flush its contents to stable storage. The SOAP interface adheres to the same pattern; a requested operation that cannot be fully completed must be reversed in its entirety (general ACID rules apply) and either no change or everything requested to change must be updated.

Security

All methods in the SOAP interface, for both reading and writing data, expects a username and a password. To successfully access the interface the user must have been granted the “Remote calls (read)” and “Remote calls (write)” permissions (the latter only required for operations updating or creating data).

Even though the default “admin” user works, even without granting the above permissions, its use is discouraged for this purpose. A separate user should be created with the only permissions being the ones mentioned above. This will prevent that user from logging in and accessing anything in the administration interface, should the password be lost.

The “admin” user should only be used to access the SOAP interface as part of troubleshooting or during testing with non-live data.

Multiple shops

Shop4Sale provides the ability to create several e-commerce shops that utilize the same backend-database. This enables warehouse sharing and smoother logistics.

When using the SOAP interface there are no calls that take the store ID as its parameter, future versions of the API may include a store ID field. With the current version a separate user must be created for each multi-shop and synchronization performed individually for each one for full integration. Certain calls, such as the getOrders call, will return a storied field in the response that may be enough to allow use of only a single synchronization user.

Error Handling

All method calls typically return a “status” field, when this field is set to “OK” (capitals) or “Found” the response is valid. All errors will return a SOAP-Fault exception, in most programming languages this will be translated to a native type of exception.

Approach to synchronization

There are several possible methods of performing synchronization with Shop4Sale, the procedure below serves as a reference only – other methods are possible.

1. Call the getVersion method to validate the communication link is ok, and also to retrieve the current date and time from the remote system.
2. Verify the difference in time between the local and remote systems. If this is too great either record the offset and compensate for it throughout or give up and issue an error message to the end-user.
3. Search for the given type of object, using a combination of the syncStart, syncEnd, sync1 and sync2 fields. The syncStart field is compared against the last modified timestamp (larger than or equal), the syncEnd field is also compared against the last modified timestamp (less than or equal). The most common approach would be to search the database for one of the following combinations:
 - a. Shop4Sale:syncStart >= remote:lastSuccessfulSync AND Shop4Sale:sync1=""
Looks mainly at the modified timestamp, and only for those objects that have not been assigned a remote identity yet.
 - b. Shop4Sale:syncStart >= remote:lastSuccessfulSync
Look at the timestamp only
 - c. Shop4Sale:sync1 = ""
Ignore any timestamp, just look for objects that have not been assigned a remote identity yet.

Retrieving information from Shop4Sale

When retrieving information from Shop4Sale all the conditions specified in the request are logically sent through an 'and' operation. There is no way to specify an 'or' operation instead or using intervals (with the exception of syncStart and syncEnd that provide two fields for a range, and the primary key for each object – e.g. multiple product numbers).

Optional fields that are not specified in the request are simply ignored and do not affect the final database query.

Almost all operations that retrieve data can, and most of the time will, return multiple rows of data. The returned results are always complete, with all of the optional fields filled in. All operations that can return multiple rows of data also have a limit field, this field can be used to limit the amount of returned answers. If the limit field is not specified it will default to 50 entries.

Modifying information in Shop4Sale

To update data inside the Shop4Sale database the remote end is responsible for determining whether a 'modify' or 'create' operation is required. This is an intentional design choice for ensuring that the Shop4Sale security policies can be applied properly. Separate SOAP methods exist for all 'create' operations and 'set' (modify) operations.

Unlike the case of retrieving information, modifications can only be performed one object at a time; it is not possible to send a list of objects for update. This is intentional to ensure ACID rules.

When using the various 'create' methods they will return an object identifier along with the status code, e.g. a product ID is returned when creating a new product.

Optional fields that are not included in a request are simply ignored, this means that any value that an attribute already had is preserved. E.g. not specifying a list of rows/products for an order leaves the product list intact while changing the other requested values.

Normal rules apply to the SOAP interface for jumping between statuses etc., this means that certain changes of attributes may not be performed if they violate business rules. As an example it may not be possible to resurrect an already cancelled purchase order.

Another surprise may be that changing attributes such as warehouse stock levels may trigger automated e-mails such as "Your desired product is back in stock" or "Warehouse completely out of..." – this can be adjusted in the template code if required.

Any update of an object through the SOAP interface will also update the changed/modified timestamp of the object, regardless of the size (number of attributes) of the change. Any object that is newly created will initially have both the created and changed/modified timestamps set to the current date and time.

Methods

Retrieving system information

Shop4Sale provides two calls to report the remote software version and associated data and the available general modules (payment, freight, sync, cart etc.).

getVersion

The getVersion call retrieves basic information on the Shop4Sale installation such as current software version, timestamp and database schema level. As it does not modify data or run any expensive operations it can also be used to verify the system is reachable and functional.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password

Response:

Field	Type	Value
baseVersion	String	The base version of the Shop4Sale code. Reported as major.minor.revision, e.g. 5.5.1
dbLevel	String	The database schema level, generally a certain base version runs with a certain dbLevel. Reported as an integer, e.g. 1330.
phpVersion	String	The PHP interpreter in use, reported as major.minor.revision, e.g. 5.3.3
currentTime	Datetime	Iso8601 formatted string representing the current date and time at the server.

getModuleList

The getModuleList call fetches a complete list of all modules visible to the administrative user logged in. In a multiple shop environment only the modules in the active shop will be returned.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
Modules	Module	A list of modules

Retrieving, creating and modifying products

To manage products Shop4Sale provides three methods, two of which are very similar.

getProducts

The getProducts call returns a list of matched products and their associated attributes.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Limit	Int., opt.	The maximum number of entries to return, default 50
Prodno	Array of Int., opt.	The system-defined product number
Orderpnr	String, opt.	The user-defined ordering part number
Productname	String, opt.	The name of the product
Manufacturerid	Int., opt.	The manufacturer ID
Manufacturerpnr	String, opt.	The manufacturers part number
Upcean	String, opt.	The UPC/EAN barcode
Weight	Int., opt.	The weight in grams
Supplierid	Int., opt.	The ID of the supplier of this product
Supplierpnr	String, opt.	The part number used by the supplier
Association	Int., opt.	The parent product this product is associated with
Freightclass	Int., opt.	A freight class, valid range is 1-10
Vatid	Int., opt.	The VAT id
Endofsale	Boolean, opt.	Status of end-of-sale flag
Active	Boolean, opt.	Status of active flag
Orderitem	Boolean, opt.	Status of orderitem flag
Onsale	Boolean, opt.	Status of onsale flag
Structural	Productstructure, opt.	A structure of products making up this structural product.
Accessories	Array of Int., opt.	A list of accessories (product ID's) associated with this product
Permanentstock	Boolean, opt.	This flag is set if this product will always be shown as in stock to end-users – regardless of real stock levels
Discountcategory	String, opt.	An uppercase single letter representing the freight class, valid range A-J.
Storied	Int, opt.	The ID of the store where to search
Sync1	String, opt.	Uniquely indexed synchronization field
Sync2	String, opt.	Index synchronization field
syncStart	Datetime, opt.	Return rows with a modified date above this field
syncEnd	Datetime, opt.	Return rows with a modified date below this field

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
Productlist	Array of product	A list of products matching the search criteria

setProduct

The setProduct call will modify an existing product. Fields not specified in the request will be ignored and left unchanged.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Product	Product, required	A product object describing the changes to make

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise

createProduct

The createProduct call is very similar to setProduct but can only be used to create new products, not change existing ones.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Product	Product, required	A product object describing the new product

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise
Prodno	Int	The newly created products ID number

Retrieving, creating and modifying customers

To manage customers Shop4Sale provides three methods, two of which are very similar.

getCustomers

The getCustomers call will search the customer tables and returns the matching rows.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Limit	Int., opt.	The maximum number of rows to return, default 50
Userid	Array of Int., opt.	A list of userid's to search for
Firstname	String, opt.	The customers first name
Lastname	String, opt.	The customer last name
Email	String, opt.	The customers e-mail address
Address	String, opt.	Address
Zipcode	String, opt.	Zipcode
City	String, opt	City
Country	String, opt	Country in ISO format, 2 digit. E.g. lowercase 'se' for Sweden.
Telephone	String, opt	The customers telephone number
Mobile	String, opt	The customers mobile phone number
Fax	String, opt	Fax number
Language	String, opt	The language spoken by this individual, 2-letter ISO, e.g. 'sv' for 'Svenska', 'fi' for 'Finnish' and 'en' for 'International English'.
Usertype	String, opt.	Determines if this is a person (returns 'user' or company (returns 'company')
Companyname	String, opt.	The name of the company
Sync1	String, opt.	Uniquely indexed synchronization field
Sync2	String, opt.	Index synchronization field
syncStart	Datetime, opt.	Return rows with a modified date above this field
syncEnd	Datetime, opt.	Return rows with a modified date below this field

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
Customerlist	Array of customer	A list of customers matching the search criteria

setCustomer

The setCustomer calls updates the details of an existing customer.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Customer	Customer, req.	The customer details

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.

createCustomer

The createCustomer call will create a new customer. It cannot be used to update an existing customer.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Customer	Customer, req.	The customer details

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
userid	Int.	The ID of the newly created customer record

Retrieving, creating and modifying suppliers

To manage suppliers Shop4Sale provides three methods, two of which are very similar.

getSuppliers

The getSuppliers call will return a list of suppliers matching a given search criteria.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Limit	Int., opt.	The maximum number of rows to return, default 50
Supplierid	Array of Int., opt.	A list of supplier IDs
Name	String, opt.	The name of the supplier
Telephone	String, opt.	Telephone
Fax	String, opt.	Fax
Contact	String, opt.	The contacts name
Address	String, opt.	The address
Zipcode	String, opt.	Zipcode
City	String, opt.	City
Country	String, opt.	ISO formatted 2-digit country code, e.g. 'se' for Sweden
Email	String, opt.	E-mail address
Account	String, opt.	The account number with this supplier
Sync1	String, opt.	Uniquely indexed synchronization field
Sync2	String, opt.	Index synchronization field
syncStart	Datetime, opt.	Return rows with a modified date above this field
syncEnd	Datetime, opt.	Return rows with a modified date below this field

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
Supplierlist	Array of supplier	A list of suppliers matching the search criteria

setSupplier

The setSupplier call will update an existing supplier with new information.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Supplier	Supplier, req.	The changes to make to the object

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.

createSupplier

The createSupplier call creates a new supplier record as described by the request structure.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Supplier	Supplier, req.	A structure describing the supplier

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
supplierid	Int.	The ID of the newly created supplier record

Retrieving, creating and modifying manufacturers

To manage manufacturers Shop4Sale provides three methods, two of which are very similar.

getManufacturers

The getManufacturers call retrieves a list of manufacturers according to the specified search criteria. The order of the list is not guaranteed.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Limit	Int, opt.	The maximum number of entries to return, default 50
Manufacturerid	List of Int, opt.	A list of manufacturer IDs – entire set matches (e.g. 10, 20, 22 will match 10, 20 and 22)
Name	String, opt.	Name of manufacturer
Telephone	String, opt.	Telephone number
Fax	String, opt.	Fax number
Contact	String, opt.	The contact person
Address	String, opt.	Address
Zipcode	String, opt.	Zipcode
City	String, opt.	City
Country	String, opt.	A valid ISO countrycode (2-digit) in lowercase, e.g. 'se'.
Email	String, opt.	E-mail address
Sync1	String, opt.	Uniquely indexed synchronization field
Sync2	String, opt.	Index synchronization field
syncStart	Datetime, opt.	Return rows with a modified date above this field
syncEnd	Datetime, opt.	Return rows with a modified date below this field

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
Manufacturerlist	Manufacturer	A list of manufacturer objects

setManufacturer

The setManufacturer call will update an existing manufacturer.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Manufacturer	Manufacturer, req.	A structure describing the changes to make

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.

createManufacturer

The createManufacturer call will create a new manufacturer according to the supplied structure.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Manufacturer	Manufacturer, req.	A structure describing the new manufacturer

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
manufacturerid	Int.	The ID of the newly created manufacturer record

Retrieving, creating and modifying purchase orders

To manage purchase orders Shop4Sale provides three methods, two of which are very similar.

getPurchaseOrders

The getPurchaseOrders call will retrieve a list of matching purchaseorders, based upon a given search criteria.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Limit	Int., opt.	The maximum number of entries to return, default 50
Purchaseorderid	Array of Int., opt.	A list of purchaseorderid's to match
Supplierid	Int., opt.	The supplierid to match
State	String, opt.	1 = Open 2 = Closed 3 = Cancelled
Userid	Int., opt.	The userid to match
Method	String, opt.	The type of method, only two supported methods are available at this time, 'email' or 'manual'.
Sync1	String, opt.	Uniquely indexed synchronization field
Sync2	String, opt.	Index synchronization field
syncStart	Datetime, opt.	Return rows with a modified date above this field
syncEnd	Datetime, opt.	Return rows with a modified date below this field

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
Purchaseorderlist	Purchaseorder	A list of purchaseorder objects

setPurchaseOrder

The setPurchaseOrder call updates an existing purchase order with new data.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Purchaseorder	Purchaseorder, req.	The changes to make

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.

createPurchaseOrder

The createPurchaseOrder call will create a new purchase order as described by the request data.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Purchaseorder	Purchaseorder, req.	The purchase order data

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
Purchaseorderid	Int.	The ID of the newly created purchaseorder

Retrieving, creating and modifying orders

To manage orders Shop4Sale provides three methods, two of which are very similar.

getOrders

The getOrders call retrieves a list of orders matching a search criteria.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Limit	Int., opt.	The maximum number of rows to return, default 50
Orderid	Array of Int., opt.	A list of orders to scan for
State	String, opt.	“preorder”, “order”, “packing”, “sent”, “cancelled”
Userid	Int., opt.	The user/customer connected to the order
Priority	Int., opt.	The order priority, 1-10
Paid	Boolean, opt.	Payment status (how to treat the order)
Paymentreceived	Boolean, opt.	Has payment been received (has the money arrived at our account)
Paymentref	String, opt.	The payment reference, specific to each type of payment method.
Storied	Int., opt.	The store in which the order was placed
Warehouseid	Int., opt.	The warehouse to which the order is connected
Sync1	String, opt.	Uniquely indexed synchronization field
Sync2	String, opt.	Index synchronization field
syncStart	Datetime, opt.	Return rows with a modified date above this field
syncEnd	Datetime, opt.	Return rows with a modified date below this field

Response:

Field	Type	Value
Status	String	The text “OK” if everything was alright, an error message otherwise.
Orderlist	Order	A list of order objects

setOrder

The setOrder call will update an existing order with the changes specified in the request data.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Order	Order, req.	A structure describing the changes to make

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.

createOrder

The createOrder call creates a new order as described by the request structure.

Request:

Field	Type	Value
adminUser	String, required	Username
adminPassword	String, required	Password
Order	Order, req.	A structure describing the order

Response:

Field	Type	Value
Status	String	The text "OK" if everything was alright, an error message otherwise.
orderid	Int	The newly created order ID

Structures

This section details the individual structures used by the various method calls.

Address

Field	Type	Value
Name	String	Name
Address1	String	1 st line of address
Address2	String	2 nd line of address
Zipcode	String	Zipcode
City	String	City name
Country	String	An ISO 2-digit country code, e.g. 'se' for Sweden

Associationtype

Field	Type	Value
Associationindex	Int.	The association index (1-3)
Typeid	String	The type of association (ID), maps to 'color', 'size' etc
Typename	String	The type of association, e.g. 'color', 'size' etc
Description	String	The description of this association

Customer

Field	Type	Value
Userid	Int.	The users ID
Firstname	String	First name of customer
Lastname	String	Last name of customer
Email	String	The email address
Address	String	The street address
Zipcode	String	Zipcode
City	String	City name
Country	String	An ISO 2-digit country code, e.g. 'se' for Sweden
Username	String	The username, Shop4Sale may use this field depending upon settings – but will most commonly use the email address as the username. Upon initial creation, if not explicitly set, will be set equal to the email address
Usertype	String	Type of user, either 'user' or 'company'
Companyname	String	Name of the company
Locked	Boolean	Is the account locked
Lastlogin	Datetime	The timestamp of last login, if the account was auto-created the initial creation does not count as a login
Nologins	Int.	Number of logins, if the account was auto-created the auto-creation time does not count
Telephone	String	Telephone
Fax	String	Fax
Mobile	String	Mobile
Language	String	The ISO language code, e.g. 'sv' for 'Svenska' or 'en' for 'International English'.
Password	String	The password, either in plaintext if no salt field is specified or encrypted if the salt field is specified. This field is never included in search responses, it may be used in create or set operations only. This field is ignored if the length is zero.

Salt	String	The password salt value, if this field is specified in a set/create operation the password field above must also be set and the password is treated literally. This field is never included in search responses.
Sync1	String	The current value of sync1 for this customer
Sync2	String	The current value of sync2 for this customer
Created	Datetime	The timestamp of last change
Changed	Datetime	The timestamp of creation

Description

Field	Type	Value
Languagecode	String	ISO language code, e.g. 'sv' for Swedish.
Text	String	Text content

Manufacturer

Field	Type	Value
Manufacturerid	Int.	ID of manufacturer
Name	String	Manufacturers name
Contact	String	Contact name
Telephone	String	Telephone
Fax	String	Fax
Address	String	Address
Zipcode	String	Zipcode
City	String	City
Country	String	An ISO 2-digit country code, e.g. 'se' for Sweden
Email	String	E-mail address
Weburl	String	URL to manufacturers website
Comment	String	Comment field
Sync1	String	The current value of sync1 for this manufacturer
Sync2	String	The current value of sync2 for this manufacturer
Created	Datetime	The timestamp of last change
Changed	Datetime	The timestamp of creation

Module

Field	Type	Value
Moduleid	Int.	The ID of the module
State	Boolean	The current state, enabled (true) or disabled (false)
Isdefault	Boolean	If it is the default choice in its class
Displayname	String	The name displayed in the cashiers page
Storied	Int.	The store where this module is connected
Handler	String	The name of the handler for this module
Type	String	The type of module, valid values are: 'payment', 'freight', 'sync', 'broker', 'cart', 'feed', 'discount', 'tracking', 'carrier'

Order

Field	Type	Value
Orderid	Int.	The ID of the order
Userid	Int.	The ID of the user who placed the order

State	String	The current order state, 'preorder', 'order', 'packing', 'sent', 'cancelled'
Remoteip	String	The IP address from where the order was placed
Currency		The currency, in ISO format, e.g. 'SEK' for Swedish Krona
Language	String	The language used, ISO coded, e.g. 'sv' for Swedish.
Priority	Int.	A priority value between 1 and 10.
Freightmodid	Int.	The module ID of the chosen freight method
Freightmod	String	The name of the chosen freight method
Paymentmodid	Int.	The module ID of the chosen payment method
Paymentmod	String	The name of the chosen payment method
Paymentref	String	A payment reference
Paid	Boolean	Set if order is considered paid
Coupon	Int.	The identity of a discount coupon if used for this order
Storied	Int.	The ID of the store where the purchase took place
Warehouseid	Int.	The connected warehouse ID
Orderrows	Array of orderrow	The orderrows
Waybills	Array of waybill	The various waybills
Comment	String	The comment field, visible on picklist
Note	String	The private note field, not visible on picklist
Deliveryaddress	Address	The delivery address
Sync1	String	The current value of sync1 for this order
Sync2	String	The current value of sync2 for this order
Changed	Datetime	The timestamp of last change
Created	Datetime	The timestamp of creation

Orderrow

Field	Type	Value
Rowid	Int.	The unique orderrow ID, maps to a single row in the DB
Prodno	Int.	The product ID (required for set operations)
Orderpn	String	The products ordering part number (for get only)
Quantity	Int.	The quantity ordered
Delivered	Int.	The quantity delivered (cannot be set, only read)
Price	Float	The price of the product, per piece
Discount	Float	The discount of the product, per piece
Vatrate	Float	The VAT rate

Price

Field	Type	Value
Price	Float	The price
Orgprice	Float	The original price (during non-sale)
Currencyid	Int	The ID of the currency
Currencycode	String	The currency ISO code, e.g. 'SEK' for Swedish Krona, always uppercase
Vatid	Int	The ID of the vatrate
Vatrate	Float	The vatrate expressed as a true number

Product

Field	Type	Value
Prodno	Int	The article number

Orderpn	String	The ordering part number
Product	String	The name of the product
Manufacturerid	Int.	The ID of the manufacturer
Manufacturerpn	String	The part number with the manufacturer
Upcean	String	The UPC/EAN barcode
Weight	Int.	The weight of this product, in grams
Supplierid	Int.	The supplier ID
Supplierpn	String	The part number used by the supplier
Association	Int.	The parent in an associationchain
Freightclass	Int.	Freightclass, between 1 and 10
Endofsale	Boolean	Is this product end-of-sale?
Active	Boolean	Is this product active?
Orderitem	Boolean	Is it something we need to backorder?
Onsale	Boolean	Is it onsale?
Structure	Productstructure	List of other products that make up this one
Accessories	Array of Int.	List of suitable accessories
Permanentstock	Boolean	Always display as available to customers?
Discountcategory	String	Uppercase character between A and J
Prices	Price	List of prices in various currencies
Shortdescription	Description	A short description of this product, one per language
Longdescription	Description	A longer description of this product, one per language
Specification	Description	A specification of this product, one per language
Warehouse	Productlocation	What do we have in stock
Comment	String	A product comment
Associationtypes	Association	The various associations this product is involved in
metaTitle	String	The product title (as reported by the title tag)
metaDesc	String	The meta description tag
metaKeywords	String	The meta keywords tag
metaAlttext	String	The image alternate text to use for this product
Sync1	String	The current value of sync1 for this order
Sync2	String	The current value of sync2 for this order
Changed	Datetime	The timestamp of last change
Created	Datetime	The timestamp of creation

Productlocation

Field	Type	Value
Warehouseid	Int.	The ID of the warehouse
Instock	Int.	The number of such items on the shelf
Reserved	Int.	This field is calculated, it cannot be set
Coordinates	String	The location of this product
Warning	Int.	The threshold for receiving a warning message

Productstructure

Field	Type	Value
Quantity	Int.	How many
Prodno	Int.	A product ID

Purchaseorder

Field	Type	Value
Purchaseorderid	Int.	The purchaseorder ID

Supplierid	Int.	The ID of the supplier
State	String	1=Open, 2=Closed, 3=Cancelled
Userid		The administrative userid who placed the purchaseorder
Method	String	How was it placed, 'email' or 'manual'
Purchaseorderrows	Purchaseorderrow	The individual orderrows
Comment	String	A comment
Expecteddelivery	Datetime	When is it due
Sync1	String	The current value of sync1 for this order
Sync2	String	The current value of sync2 for this order
Changed	Datetime	The timestamp of last change
Created	Datetime	The timestamp of creation

Purchaseorderrow

Field	Type	Value
Purchaseorderrowid	Int.	The purchaseorderrow ID
Prodno	Int.	The product ID
Orderpn	String	The product ordering part number
Amount	Int.	The quantity
Price	Float	The purchaseprice
Delivered	Int.	How many have been delivered from the supplier

Supplier

Field	Type	Value
Supplierid	Int.	The ID of the supplier
Name	String	Name
Telephone	String	Telephone
Fax	String	Fax
Contact	String	Contact
Address	String	Address
Zipcode	String	Zipcode
City	String	City
Country	String	ISO country code, e.g. 'se' for Sweden
Email	String	The email address
Account	String	An account number to be used when ordering from this supplier
Comment	String	A general comment field
Sync1	String	The current value of sync1 for this order
Sync2	String	The current value of sync2 for this order
Created	Datetime	The timestamp of last change
changed	Datetime	The timestamp of creation

Waybill

Field	Type	Value
Waybillid	Int.	The ID of the waybill
Carrierid	Int.	The ID of the carrier (required for set operations)
Carriername	String	The name of the carrier (read-only)
Carrierref	String	The carrier reference
Created	Datetime	The timestamp when this waybill was created

Terminology

- **Associated product:** A product available in several variations, e.g. color or size. It is referred to as associated as it makes use of the association field in the database. One product in an association chain is the mother and all variations of it have their association field pointing to the mother product.
- **Multishop:** Shop4Sale allows creation of several shops (or store fronts) that share the backend. They may or may not use the same warehouse.