

# Amberdms Billing System SOAP API

This document contains information about the SOAP API provided by the Amberdms Billing System available for use by third-party software components.

Copyright 2009 Amberdms Ltd

*This document may be freely distributed but modifications are not permitted.*

Version 7 April 2009

## Table of Contents

1. Introduction.....	4
2. Requirements.....	5
2.1. Amberdms Billing System.....	5
2.2. Local PHP Requirements.....	5
2.2.1. RHEL / Fedora / CentOS.....	5
2.3. HTTPS.....	5
3. Amberdms Billing System Hosted Service.....	6
3.1. Authentication.....	6
3.2. Service Usage Records.....	6
4. Getting Support.....	7
4.1. Commercial Support.....	7
4.1.1. Technical Support.....	7
4.1.2. Development/Integration Assistance.....	7
4.1.3. Not a customer?.....	7
4.2. Community Support.....	7
4.3. Customisation/Development Services.....	8
5. API Basics.....	9
6. API: authenticate.....	10
6.1. authenticate.....	10
6.1.1. login.....	10
7. API: services.....	12
7.1. services_usage.....	12
7.1.1. set_usage_record.....	12
8. API: hr.....	13
8.1. hr_staff_manage.....	13
8.1.1. get_employee_details.....	13
8.1.2. set_employee_details.....	13
8.1.3. delete_employee.....	14
9. API: customers.....	15
9.1. customers_manage.....	15
9.1.1. get_customer_id_from_code.....	15
9.1.2. get_customer_details.....	15
9.1.3. list_customer_tax.....	16
9.1.4. set_customer_details.....	17
9.1.5. set_customer_tax.....	18
9.1.6. delete_customer.....	18
10. API: vendors.....	19
10.1. vendors_manage.....	19
10.1.1. get_vendor_details.....	19
10.1.2. list_vendor_tax.....	20
10.1.3. set_vendor_details.....	20
10.1.4. set_vendor_tax.....	21
10.1.5. delete_vendor.....	22
11. API: accounts.....	23
11.1. accounts_charts_manage.....	23
11.1.1. list_chart_type.....	23
11.1.2. list_chart_menu.....	23
11.1.3. get_chart_details.....	24
11.1.4. get_chart_menu.....	24
11.1.5. set_chart_details.....	25
11.1.6. set_chart_menuoption.....	25
11.1.7. delete_chart.....	26
11.2. accounts_gl_manage.....	26
11.2.1. get_gl_details.....	26
11.2.2. get_gl_trans.....	27

---

11.2.3. prepare_gl_details.....	27
11.2.4. prepare_gl_addtrans.....	28
11.2.5. set_gl_save.....	29
11.2.6. delete_gl.....	29
11.3. accounts_taxes_manage.....	30
11.3.1. list_taxes.....	30
11.3.2. get_tax_details.....	30
11.3.3. set_tax_details.....	31
11.3.4. delete_tax.....	32
11.4. accounts_invoices_manage.....	32
11.4.1. get_invoice_id_from_code.....	32
11.4.2. get_invoice_details.....	33
11.4.3. get_invoice_items.....	34
11.4.4. get_invoice_taxes.....	35
11.4.5. get_invoice_payments.....	36
11.4.6. set_invoice_details.....	36
11.4.7. set_invoice_item_standard.....	37
11.4.8. set_invoice_item_tax.....	38
11.4.9. set_invoice_item_product.....	38
11.4.10. set_invoice_item_time.....	39
11.4.11. set_invoice_override_tax.....	40
11.4.12. set_invoice_payment.....	41
11.4.13. delete_invoice.....	41
11.4.14. delete_invoice_item.....	42
12. API: Products.....	43
12.1. products_manage.....	43
12.1.1. get_product_details.....	43
12.1.2. get_product_taxes.....	44
12.1.3. set_product_details.....	44
12.1.4. set_product_tax.....	45
12.1.5. delete_product.....	45
12.1.6. delete_product_tax.....	46
13. Sample Code.....	47
13.1. Service Collectors.....	47
13.2. Testsuite Scripts.....	47

## 1. Introduction

The Amberdms Billing System has a comprehensive SOAP API to allow access to stored data by third-party scripts and software applications.

Some examples of applications this API can be used for:

1. Uploading service usage records.
2. Importing data from old programs – for example, the SQL-Ledger migration scripts connect via SOAP.
3. Linking the billing system to another programs such as service management applications.

SOAP is a protocol for exchanging information in XML markup ontop of an application protocol such as HTTP. Amberdms chose to use SOAP rather than POST for the APIs, because of the ease of handling two-way communications.

Because SOAP runs over the HTTP protocol, no additional daemons or services need to be running on the billing system server, everything is done via the webserver.

**Please be aware – the SOAP API in the Amberdms Billing System is not fully complete – all APIs documented in this manual are stable and will not be removed, however we are working on adding additional APIs to future releases to allow better interaction with timesheets, reporting options.**

To learn more about work being performed on the SOAP APIs email [support@amberdms.com](mailto:support@amberdms.com) or join the development mailing list.

## 2. Requirements

### 2.1. Amberdms Billing System

In order to support the SOAP API, the Amberdms Billing System must be running on a server with PHP version 5. If the Amberdms Billing System is running on PHP version 4, the SOAP API will not be available.

For reasons documented in the Installation Guide, Amberdms recommend the user of the Amberdms Billing System with version 5.2.0+.

### 2.2. Local PHP Requirements

If writing SOAP applications in PHP, you will require PHP version 5 or above with the SOAP extensions installed. Please be aware that some distributions will package the SOAP extensions separately to the main PHP page.

#### 2.2.1. RHEL / FEDORA / CENTOS

On Redhat-style distributions, you can install all the required packages with:

```
$ yum install php-cli php-soap
```

### 2.3. HTTPS

SOAP calls pass confidential information such as passwords, customer records and financial records via the network. It is important to ensure all connections to SOAP are made via HTTPS.

All users of the Amberdms Billing System Hosted Service are automatically forced to connect via HTTPS.

## 3. Amberdms Billing System Hosted Service

The Amberdms Billing System Hosted Service includes the full SOAP API for all customers. The following are some notes to be aware of.

### 3.1. Authentication

All customers of the Amberdms Billing System Hosted Service have full access and support for using the SOAP API. The SOAP API for hosted customers can be accessed at the following URL:

[https://www.amberdms.com/products/billing\\_system/online/api/](https://www.amberdms.com/products/billing_system/online/api/)

When writing/configuring your SOAP applications, remember that when authenticating, your application must all three authentication variables:

1. Account – must be set to your customer ID.
2. Username
3. Password

Without the account, your SOAP applications will be denied access since the Amberdms Billing System will be unable to determine which account to connect you to.

### 3.2. Service Usage Records

Heavy use of the SOAP API for recording usage information may use up your data allocation faster than expected.

Amberdms recommend only making one upload of usage information for each day for each customer – because invoicing only runs on a daily basis, having more frequent service usage records (eg: every 5 minutes) is a waste of storage and processing.

## 4. Getting Support

### 4.1. Commercial Support

If you are a customer of the Amberdms Billing System Hosted Service or the Amberdms Billing System Subscription, Amberdms will provide free unlimited technical support for any issues you have with communicating with the SOAP API.

#### 4.1.1. TECHNICAL SUPPORT

To receive technical support, contact us using one of the methods list at:  
[http://www.amberdms.com/?page=products/billing\\_system/support\\_basic.php](http://www.amberdms.com/?page=products/billing_system/support_basic.php)

Please have your customer ID ready so that our staff may assist you effectively.

#### 4.1.2. DEVELOPMENT/INTEGRATION ASSISTANCE

Our free technical support service does not cover us supporting developers with coding issues in their own applications.

If you would like comprehensive support including phone or email assistance for developers or staff integrating the Amberdms Billing System with other applications, or for support with writing your own SOAP programs, you will need to either have a support package or be prepared to pay the hourly support rate.

If you already have a support package, you may contact us using the details available at:  
[http://www.amberdms.com/?page=products/billing\\_system/support\\_package.php](http://www.amberdms.com/?page=products/billing_system/support_package.php)

If you do not have a support package, but do have either a hosted service or subscription service with Amberdms, we can still support you but will charge for our time and add it to your monthly invoice. To make a support request, use the information at [http://www.amberdms.com/?page=products/billing\\_system/support\\_basic.php](http://www.amberdms.com/?page=products/billing_system/support_basic.php)

If you don't currently have a support package with Amberdms but would like to get one you can find out more about our offerings at [http://www.amberdms.com/?page=products/billing\\_system/getsupport.php](http://www.amberdms.com/?page=products/billing_system/getsupport.php)

#### 4.1.3. NOT A CUSTOMER?

If you are not a customer, you can try and get support from the community via our mailing lists, however we recommend that you sign up for our subscription service.

The Amberdms Billing System Subscription Service lets you run the Amberdms Billing System on your own servers, but you get the advantage of priority updates, alerts to new fixes and our commercial support services.

Full details can be found at: [http://www.amberdms.com/?page=products/billing\\_system/subscription.php](http://www.amberdms.com/?page=products/billing_system/subscription.php)

### 4.2. Community Support

If you are not a customer of Amberdms, you may be able to get assistance from other members of the community by using the Billing System mailing lists.

General Discussion List:

<http://lists.amberdms.com/mailman/listinfo/amberdms-bs>

Amberdms Billing System Developers Mailing List:

<http://lists.amberdms.com/mailman/listinfo/amberdms-bs-devel>

## **4.3. Customisation/Development Services**

If you would like to develop a SOAP application to talk to the Amberdms Billing System or integrate the Amberdms Billing System with your existing software applications, Amberdms can provide development services at a competitive price.

Talk to [sales@amberdms.com](mailto:sales@amberdms.com) about your requirements for more information on how we can help your organisation.



## 5. API Basics

The Amberdms Billing System has divided the different APIs into sections with separate WDSL definition files.

All client programs should be configured to download the WDSL files directly from the billing system rather than having local copies as this ensures client applications will be provided with the correct version document for the version of the billing system being accessed.

The documentation below defines all the available APIs, acceptable input/output and what failure messages might occur.

## 6. API: authenticate

### 6.1. authenticate

Executable	api/authenticate/authenticate.php
WSDL	api/authenticate/authenticate.wsdl
Summary	Provides authentication – required before using any other API

Before using any other SOAP calls, the client needs to authenticate against the billing system by using the authenticate API.

Authentication is handled by the same backend as the main web application. This provides two important features of note:

1. There is no distinction between SOAP and web application users – all users are configured in the same way, and their abilities are defined by the user's permissions.
2. The authenticate API will handle blacklisting in the same way as the web application.

When writing anything other than a basic script with SOAP, it is very likely the client will need to make multiple SOAP calls to different APIs.

It would be inefficient to have to authenticate for every single request, so instead the authenticate::login function will return a PHP session ID once successfully authenticated.

This session ID can then be passed via HTTP GET to any SOAP connections to gain access.

For example:

1. Call [https://www.amberdms.com/products/billing\\_system/online/api/authenticate/authenticate.php](https://www.amberdms.com/products/billing_system/online/api/authenticate/authenticate.php)
2. Get session ID string to \$PHPSESSION
3. Call [https://www.amberdms.com/products/billing\\_system/online/api/service/example.php?\\$PHPSESSION](https://www.amberdms.com/products/billing_system/online/api/service/example.php?$PHPSESSION)

This will eliminate the need to re-authenticate during the programs runtime, however it is worth noting that there is a timeout on the session.

If you have a complex application which authenticates and provides a number of different features to a user, you either need to re-authenticate on each call in your application, or have the application user timeout set to less than the timeout on the billing system, otherwise there is the risk of the session timing out between user actions.

#### 6.1.1. LOGIN

##### Input

account	string	required	ID of customer's account (Amberdms Hosted Service only)
username	string	required	Username (plaintext)
password	string	required	Password (plaintext)

##### Output

sessionid	string	Session ID/string, suitable for passing via HTTP GET
-----------	--------	------------------------------------------------------

**Failure Codes**

USER_DISABLED	Requested username/account was valid, but the user has been disabled by the administrator.
BLACKLISTED	Too many invalid authentication attempts and the client IP has been blacklisted for security reasons.
INVALID_AUTHDETAILS	Supplied username or password is invalid.
INVALID_ACCOUNT	Supplied account is invalid (Amberdms Hosted Version only)

## 7. API: services

### 7.1. services\_usage

Executable	api/services/usage.php
WSDL	api/services/usage.wsdl
Summary	Provides various calls for fetching customer usage information and also provides calls to upload usage information into the billing system.

The service usage API is very important for anyone who wants to bill for usage on service, as it is the only way to enter values into the database.

Amberdms has prepared a detailed document explaining how to work with the service usage API in more details with a focus on assisting ISPs and hosting providers to configure usage billing for their services.

*Please read through the Service Usage Guide documentation for more detailed information than the API documentation below.*

#### 7.1.1. SET\_USAGE\_RECORD

Uploads usage information into the billing system for the provided customer/service ID. This usage information is later used when the service invoices are generated.

The set\_usage\_record call will always add a new row to the usage table – if there is already a row for this date another one will be added causing the totals for the date to increment.

#### Required Permissions

- services\_write\_usage

#### Input

collector	string	Name of the collector (anything is acceptable)
services_customers_id	string	ID of the services_customers record for this customer/service assignment
date	date	YYYY-MM-DD
usage1	int	Usage field #1
usage2	int	Usage field #2

#### Output

returncode	string	Return Code
------------	--------	-------------

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_SERVICES_CUSTOMERS_ID	Invalid services_customers_id record.
UNEXPECTED_DB_ERROR	Unexpected failure with the DB.

## 8. API: hr

### 8.1. hr\_staff\_manage

Executable	api/hr/staff_manage.php
WSDL	api/hr/staff_manage.wsdl
Summary	Provides various calls for viewing, adjusting and deleting employees.

#### 8.1.1. GET\_EMPLOYEE\_DETAILS

Fetches an array containing all the employee details for the provided employee ID.

##### Required Permissions

- staff\_view

##### Input

id	int	required	Employee ID
----	-----	----------	-------------

##### Output

name_staff	string	Employee Name
staff_code	string	Employee code
staff_position	string	
contact_phone	string	
contact_fax	string	
contact_email	email	Single email address
date_start	date	YYYY-MM-DD
date_end	date	YYYY-MM-DD

##### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when loading data.

#### 8.1.2. SET\_EMPLOYEE\_DETAILS

Takes the provided values and creates or updates an employee.

##### Required Permissions

- staff\_write

##### Input

id	int		Employee ID – leave blank to create a new employee
name_staff	string	required	Employee Name
staff_code	string		Employee code
staff_position	string		
contact_phone	string		
contact_fax	string		
contact_email	email		Single email address
date_start	date	required	YYYY-MM-DD
date_end	date		YYYY-MM-DD

**Output**

id	int	Employee ID
----	-----	-------------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
DUPLICATE_NAME_STAFF	An employee already exists with the name_staff value you are trying to set.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**8.1.3. DELETE\_EMPLOYEE**

Deletes an employee, provided that the employee is not locked.

**Required Permissions**

- staff\_write

**Input**

id	int	required	Employee ID
----	-----	----------	-------------

**Output**

returncode	int	
------------	-----	--

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
LOCKED	The employee can not be deleted because they have various data entries made against them. You should set the <i>date_end</i> field instead to hide the employee.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

## 9. API: customers

### 9.1. customers\_manage

Executable	api/customers/customers_manage.php
WSDL	api/customers/customers_manage.wsdl
Summary	Provides various calls for viewing, adjusting and deleting customers.

*Tip: The customers\_manage API is very similar to the vendors\_manage API and it is very easy to adjust code for one API to use the other.*

#### 9.1.1. GET\_CUSTOMER\_ID\_FROM\_CODE

Looks up the provided code\_customer value and returns the actual customer ID.

##### Required Permissions

- customers\_view

##### Input

code_customer	string	required	Customer Code/ID
---------------	--------	----------	------------------

##### Output

id	int	Customer's database ID
----	-----	------------------------

##### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid code_customer or the code_customer did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when loading data.

#### 9.1.2. GET\_CUSTOMER\_DETAILS

Fetches an array containing all the customer details for the provided ID.

##### Required Permissions

- customers\_view

##### Input

id	int	required	Customer ID
----	-----	----------	-------------

##### Output

code_customer	string	Customer Code/ID
name_customer	string	Customer Name

name_contact	string	
contact_email	email	Single email address
contact_phone	string	
contact_fax	string	
date_start	date	YYYY-MM-DD
date_end	date	YYYY-MM-DD
tax_number	string	
tax_default	int	ID of the tax used as the customer's default
tax_default_label	string	Human-readable name of the above tax
address1_street	string	
address1_city	string	
address1_state	string	
address1_country	string	
address1_zipcode	string	
address2_street	string	
address2_city	string	
address2_state	string	
address2_country	string	
address2_zipcode	string	
discount	float	Purchase discount that applies to all products.

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when loading data.

**9.1.3. LIST\_CUSTOMER\_TAX**

Displays a list of all taxes and flags which ones are enabled for the selected customer.

**Required Permissions**

- customers\_view

**Output – Multiple rows structured as:**

id	int	ID of the menu option – this is typically not used.
taxid	int	ID of the tax
name_tax	string	Human-readable name of the tax
status	string	Either “on” (aka enabled) or “off”.



### 9.1.4. SET\_CUSTOMER\_DETAILS

Takes the provided values and creates or updates a customer.

#### Required Permissions

- customers\_write

#### Input

id	int		Customer ID – leave blank to create a new customer
code_customer	string		Customer Code
name_customer	string	required	Customer Name
name_contact	string		
contact_email	email		Single email address
contact_phone	string		
contact_fax	string		
date_start	date	required	YYYY-MM-DD
date_end	date		YYYY-MM-DD
tax_number	string		
tax_default	int		ID of the tax to use as the customer's default
address1_street	string		
address1_city	string		
address1_state	string		
address1_country	string		
address1_zipcode	string		
address2_street	string		
address2_city	string		
address2_state	string		
address2_country	string		
address2_zipcode	string		
discount	float		Purchase discount that applies to all products.

#### Output

id	int	Customer ID
----	-----	-------------

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
DUPLICATE_NAME_CUSTOMER	Another customer already exists with the provided <i>name_customer</i> value.
DUPLICATE_CODE_CUSTOMER	Another customer already exists with the provided <i>code_customer</i> value.
HAS_ACTIVE_SERVICES	Customer has active services so the <i>date_end</i> value may not be set.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

### 9.1.5. SET\_CUSTOMER\_TAX

Function to enable or disable specific taxes for a customer.

#### Required Permissions

- customers\_write

#### Input

id	int	required	Customer ID
taxid	int	required	Tax ID - use function <i>list_customer_tax</i> to get all the available tax IDs
status	string	required	Either “on” or “off” to enable/disable this menu option.

#### Output

int	int	Return code
-----	-----	-------------

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

### 9.1.6. DELETE\_CUSTOMER

Deletes the customer, provided that they are not locked. A customer is “locked” if the customer has invoices or time groups assigned to them and can not be deleted for database consistency reasons.

#### Required Permissions

- customers\_write

#### Input

id	int	required	Customer ID
----	-----	----------	-------------

#### Output

returncode	int	
------------	-----	--

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
LOCKED	The customer can not be deleted because they have various data entries made against them. You should set the <i>date_end</i> field instead to hide the customer.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

## 10. API: vendors

### 10.1. vendors\_manage

Executable	api/vendors/vendors_manage.php
WSDL	api/vendors/vendors_manage.wsdl
Summary	Provides various calls for viewing, adjusting and deleting vendors.

*Tip: The vendors\_manage API is very similar to the customers\_manage API and it is very easy to adjust code for one API to use the other.*

#### 10.1.1. GET\_VENDOR\_DETAILS

Fetches an array containing all the vendor details for the provided ID.

##### Required Permissions

- vendors\_view

##### Input

id	int	required	Vendor ID
----	-----	----------	-----------

##### Output

code_vendor	string	Vendor Code/ID
name_vendor	string	Vendor Name
name_contact	string	
contact_email	email	Single email address
contact_phone	string	
contact_fax	string	
date_start	date	YYYY-MM-DD
date_end	date	YYYY-MM-DD
tax_number	string	
tax_default	int	ID of the tax used as the vendor's default
tax_default_label	string	Human-readable name of the above tax
address1_street	string	
address1_city	string	
address1_state	string	
address1_country	string	
address1_zipcode	string	
address2_street	string	
address2_city	string	
address2_state	string	
address2_country	string	

address2_zipcode	string	
discount	float	Purchase discount that applies to all products.

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when loading data.

**10.1.2. LIST\_VENDOR\_TAX**

Displays a list of all taxes and flags which ones are enabled for the selected vendor.

**Required Permissions**

- vendors\_view

**Output – Multiple rows structured as:**

id	int	ID of the menu option – this is typically not used.
taxid	int	ID of the tax
name_tax	string	Human-readable name of the tax
status	string	Either “on” (aka enabled) or “off”.

**10.1.3. SET\_VENDOR\_DETAILS**

Takes the provided values and creates or updates a vendor.

**Required Permissions**

- vendors\_write

**Input**

id	int		Vendor ID – leave blank to create a new vendor
code_vendor	string		Vendor Code
name_vendor	string	required	Vendor Name
name_contact	string		
contact_email	email		Single email address
contact_phone	string		
contact_fax	string		
date_start	date	required	YYYY-MM-DD
date_end	date		YYYY-MM-DD
tax_number	string		
tax_default	int		ID of the tax to use as the vendor's default
address1_street	string		

address1_city	string		
address1_state	string		
address1_country	string		
address1_zipcode	string		
address2_street	string		
address2_city	string		
address2_state	string		
address2_country	string		
address2_zipcode	string		
discount	float		Purchase discount that applies to all products.

**Output**

id	int	Vendor ID
----	-----	-----------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
DUPLICATE_NAME_VENDOR	Another vendor already exists with the provided <i>name_vendor</i> value.
DUPLICATE_CODE_VENDOR	Another vendor already exists with the provided <i>code_vendor</i> value.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**10.1.4. SET\_VENDOR\_TAX**

Function to enable or disable specific taxes for a vendor.

**Required Permissions**

- vendors\_write

**Input**

id	int	required	Vendor ID
taxid	int	required	Tax ID - use function <i>list_vendor_tax</i> to get all the available tax IDs
status	string	required	Either “on” or “off” to enable/disable this menu option.

**Output**

int	int	Return code
-----	-----	-------------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

### 10.1.5. DELETE\_VENDOR

Deletes the vendor, provided that they are not locked. A vendor is “locked” if they have AP invoices or products assigned to them.

#### Required Permissions

- vendors\_write

#### Input

id	int	required	Vendor ID
----	-----	----------	-----------

#### Output

returncode	int	
------------	-----	--

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
LOCKED	The vendor can not be deleted because they have various data entries made against them. You should set the <i>date_end</i> field instead to hide the vendor.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

## 11. API: accounts

### 11.1. accounts\_charts\_manage

Executable	api/accounts/charts_manage.php
WSDL	api/accounts/charts_manage.wsdl
Summary	Provides functions for displaying accounts/charts as well as adjusting or deleting them.

#### 11.1.1. LIST\_CHART\_TYPE

Displays a list of all available chart types (from the *account\_chart\_type* table). This output is needed in order to do name and ID lookups of the different chart types to use when creating a new chart.

##### Required Permissions

- accounts\_charts\_view

##### Output – Multiple rows structured as:

id	int	ID of the chart type
value	string	name of the chart type
total_mode	string	total mode – tells your code how to total up transactions in this account.  If set to “debit”, then perform additional on debit transactions and subtraction on credit transactions. If set to “credit”, perform the alternative.

##### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
---------------	--------------------------------------

#### 11.1.2. LIST\_CHART\_MENU

Displays a list of all available chart menu options (from the *account\_chart\_menu* table). Any chart can have multiple menu options, but before they can be set you will need to call this function to get a list of all the options available.

##### Required Permissions

- accounts\_charts\_view

##### Output – Multiple rows structured as:

id	int	ID of the menu option – this is typically not used.
value	string	Name of the menu option.
groupname	string	Field to allow you to group similar menu options together.
description	string	Information about this option – suitable for a UI

##### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
---------------	--------------------------------------

### 11.1.3. GET\_CHART\_DETAILS

Fetches all the details (excluding the menu options) for the selected chart.

#### Required Permissions

- accounts\_charts\_view

#### Input

id	int	required	Chart ID
----	-----	----------	----------

#### Output

code_chart	int	Chart code value – unlike other code values in ABS, this one can only be an integer.
description	string	Human-readable label of the chart.
chart_type	int	ID of the chart type value.
chart_type_label	string	Human-readable label of the chart type.

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when loading data.

### 11.1.4. GET\_CHART\_MENU

Displays a list of all menu options currently enabled by the selected chart. The output syntax is exactly the same as the *list\_chart\_menu* function.

#### Required Permissions

- accounts\_charts\_view

#### Input

id	int	required	Chart ID
----	-----	----------	----------

#### Output – Multiple rows structured as:

id	int	ID of the menu option – this is typically not used.
value	string	Name of the menu option.
groupname	string	Field to allow you to group similar menu options together.
description	string	Information about this option – suitable for a UI

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records



### 11.1.5. SET\_CHART\_DETAILS

Takes the provided values and creates or updates a chart.

#### Required Permissions

- accounts\_charts\_write

#### Input

id	int		Chart ID – leave blank to create a new chart.
code_chart	int	required	Chart code – must be an integer.
description	string	required	Human-readable label of the chart.
chart_type	int	required	ID of the type for this chart – use <i>list_chart_type</i> function to get a list of available types.

#### Output

id	int	Chart ID
----	-----	----------

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
DUPLICATE_CODE_CHART	A chart already exists with same code that you are trying to set.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

### 11.1.6. SET\_CHART\_MENUOPTION

Function to allow menu options to be added/removed for the selected chart.

#### Required Permissions

- accounts\_charts\_write

#### Input

id	int	required	Chart ID
menuvalue	int	required	Name of the menu option – use <i>list_chart_menu</i> function to get a list of available options.
status	string	required	Either “on” or “off” to enable/disable this menu option.

#### Output

id	int	Chart ID
----	-----	----------

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.

INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

### 11.1.7. DELETE\_CHART

Deletes a chart, provided that the chart is not locked. A chart is automatically locked if any transactions, invoice items or quote items have been assigned to it.

#### Required Permissions

- accounts\_charts\_write

#### Input

id	int	required	Chart ID
----	-----	----------	----------

#### Output

returncode	int	
------------	-----	--

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
LOCKED	The chart can not be deleted because transactions have been posted against it.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

## 11.2. accounts\_gl\_manage

Executable	api/accounts/gl_manage.php
WSDL	api/accounts/gl_manage.wsdl
Summary	Provides functions for viewing and adjusting GL transactions.

### 11.2.1. GET\_GL\_DETAILS

Fetches all the details (excluding the transaction rows) about the selected GL transaction.

#### Required Permissions

- accounts\_gl\_view

#### Input

id	int	required	Transaction ID
----	-----	----------	----------------

**Output**

code_gl	string	GL Transaction Code
date_trans	date	Transaction date
employeeid	int	ID of the employee who created the transaction
employeeid_label	string	Human-readable name of the employee who created the transaction
description	string	Transaction description
notes	string	Additional space for notes about the transaction.

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when loading data.

**11.2.2. GET\_GL\_TRANS**

Returns all the transaction rows belong to the selected GL transaction. All transaction will have at least two rows, with no maximum limit to the number of permitted rows.

**Required Permissions**

- accounts\_gl\_view

**Input**

id	int	required	Transaction ID
----	-----	----------	----------------

**Output – Multiple rows structured as:**

id	int	ID of the transaction row – typically unused
chartid	int	Name of the menu option.
chartid_label	string	Human-readable name of chart.
debit	money	Debit amount
credit	money	Credit amount
source	string	Source of the transaction - “cheque, cash, etc”
description	string	Information about the transaction

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records

**11.2.3. PREPARE\_GL\_DETAILS**

Because GL Transactions contain a number of transaction rows but must also be balanced, it is not possible to add or update a GL transaction in a single query. In order to perform updates to GL transactions, SOAP clients must call the

*prepare\_gl\_details* and *prepare\_gl\_addtrans* functions to define all the data desired, before saving it into the database with the *set\_gl\_save* function.

#### Required Permissions

- accounts\_gl\_write

#### Input

code_gl	string		GL Transaction Code
date_trans	date	required	Date of the transaction
employeeid	int	required	ID of the employee to add this transaction under.
description			Description of the transaction.
description_useall	string		Set to “on” and the description set above will be used for all transaction rows, instead of having unique descriptions for each row.
notes	string		Additional notes

#### Output

returncode	int	Not used
------------	-----	----------

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.

### 11.2.4. PREPARE\_GL\_ADDTRANS

This function creates new transaction rows for the GL transaction in memory, which then need to be saved with *set\_gl\_save* once all the rows have been defined.

#### Required Permissions

- accounts\_gl\_write

#### Input

chartid	int	required	Chart ID
credit	int	required	Credit amount
debit		required	Debit Amount
source	string		Source of the transaction - “cheque, cash, etc”
description	string		Information about the transaction

#### Important notes:

1. *Each transaction row can only contain either a credit OR a debit, it is not valid to set both a credit and debit amount in a single row.*
2. *The description option will only be used if the description\_useall field in the prepare\_gl\_details function is not set to “on”.*

#### Output

returncode	int	Not used.
------------	-----	-----------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
MISSING_FINANCIAL_VALUES	Missing request data – either chartid, debit or credit
BOTH_CREDIT_AND_DEBIT_SET	Both the credit and debit values are set – this is invalid, only one can be set.

**11.2.5. SET\_GL\_SAVE**

This function takes the data defined by the two functions above and then creates/updates a GL transaction. Be aware that all existing transaction rows will be replaced with the ones defined by *prepare\_gl\_addtrans*, so if you are just adding more rows, you need to define all the existing rows as well.

**Required Permissions**

- accounts\_gl\_write

**Input**

id	int	required	ID of GL transaction – leave blank to create new transaction
----	-----	----------	--------------------------------------------------------------

**Output**

returncode	int	Not used.
------------	-----	-----------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
LOCKED	Transaction is locked and can not be adjusted
DUPLICATE_CODE_GL	The requested code_gl value is already used by another transaction. Either set to something else, or leave blank to get an automatically generated value.
UNBALANCED_TRANSACTIONS	The total debits do not match the total credits.
MISSING_TRANS_DATA	No valid transaction rows supplied.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**11.2.6. DELETE\_GL**

Deletes a GL transaction, provided that it is not locked. A GL transaction may be locked due to an administrator closing/locking off past financial years or enabling the autolock feature.

**Required Permissions**

- accounts\_gl\_write

**Input**

id	int	required	GL Transaction ID
----	-----	----------	-------------------

**Output**

returncode	int	Not Used
------------	-----	----------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
LOCKED	The transaction is locked and can not be deleted.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

## 11.3. accounts\_taxes\_manage

Executable	api/accounts/taxes_manage.php
WSDL	api/accounts/taxes_manage.wsdl
Summary	Provides functions for managing different sales tax types.

### 11.3.1. LIST\_TAXES

Returns a list of all the taxes configured in the billing system.

**Required Permissions**

- accounts\_taxes\_view

**Output – Multiple rows structured as:**

id	int	Tax ID
name_tax	string	Unique name of the tax
taxrate	int	Tax rate (percentage)
chartid	string	ID of the account to file all transactions with this taxid
chartid_label	string	Human-readable name of the chart
taxnumber	string	Tax number
description	string	Description of the tax

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
---------------	--------------------------------------

### 11.3.2. GET\_TAX\_DETAILS

Fetch all the details of the selected tax.

**Required Permissions**

- accounts\_taxes\_view

**Input**

id	int	required	Tax ID
----	-----	----------	--------

**Output**

name_tax	string	Unique name of the tax
taxrate	float	Tax rate (percentage)
chartid	int	ID of the account to file all transactions with this taxid
chartid_label	string	Human-readable name of the chart
taxnumber	string	Tax number
description	string	Description of the tax

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when loading data.

**11.3.3. SET\_TAX\_DETAILS**

Takes the provided values and creates or updates a tax.

**Required Permissions**

- accounts\_taxes\_write

**Input**

id	int		Tax ID – leave blank to create a new tax.
name_tax	string	required	Unique name of the tax
taxrate	float	required	Tax rate (percentage)
chartid	int	required	ID of the account to file all transactions with this taxid
taxnumber	string	required	Tax number
description	string		Description of the tax
autoenable_tax_customers	string		Set this to “on” and the tax will be enabled for all customers – this should only be used when creating a new tax.
autoenable_tax_vendors	string		Set this to “on” and the tax will be enabled for all vendors – this should only be used when creating a new tax.

**Output**

id	int	Tax ID
----	-----	--------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
DUPLICATE_NAME_TAX	Another tax already exists with the name you are trying to set.
INVALID_CHARTID	The supplied chartid does not exist.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

#### 11.3.4. DELETE\_TAX

Deletes a tax provided that it isn't locked. Taxes are locked where there are transactions made using this tax.

##### Required Permissions

- accounts\_taxes\_write

##### Input

id	int	required	Tax ID
----	-----	----------	--------

##### Output

returncode	int
------------	-----

##### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
LOCKED	The tax can not be deleted because transactions have been posted against it.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

## 11.4. accounts\_invoices\_manage

Executable	api/accounts/invoices_manage.php
WSDL	api/accounts/invoices_manage.wsdl
Summary	Provides functions to view, create, update and delete invoices as well as invoice items. (both AR and AP)

#### 11.4.1. GET\_INVOICE\_ID\_FROM\_CODE

Returns the true database ID of an invoice based on the supplied code\_invoice and invoicetype values.

##### Required Permissions

- accounts\_ar\_view or accounts\_ap\_view

##### Input



code_invoice	string	required	Invoice ID/Code
invoicetype	string	required	Invoice Type - “ar” or “ap”.

**Output**

id	int	Invoice's true database ID
----	-----	----------------------------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the code_invoice is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**11.4.2. GET\_INVOICE\_DETAILS**

Fetches all the details of the selected invoice.

**Required Permissions**

- accounts\_ar\_view or accounts\_ap\_view

**Input**

id	int	required	Invoice ID
invoicetype	string	required	Invoice Type - “ar” or “ap”.

**Output**

locked	int	Locked status – (0 for unlocked, 1 for locked)
orgid	int	Either the customer or vendor ID (depending on invoice type)
orgid_label	string	Human-readable version of the above.
employeeid	int	Employee who created the invoice.
employeeid_label	string	Human-readable employee name.
dest_account	int	Destination account/chart for the invoice charges – either <i>accounts receivables</i> or <i>accounts payables</i> .
dest_account_label	string	Human-readable account/chart name
code_invoice	string	Invoice ID/Code
code_ordernumber	string	Order Number
code_ponumber	string	PO Number
date_due	date	Date invoice is due to be paid
date_trans	date	Date of invoice
date_create	date	Date invoice was added to database (non-changeable)
date_sent	date	(AR only) Date invoice was sent to customer
sentmethod	string	(AR only) Method used to send invoice to customer

amount_total	money	Total/Final amount on the invoice
amount_tax	money	Tax only amount of the invoice
amount	money	Total of line items on invoice excluding taxes.
amount_paid	money	Amount paid by customers
notes	string	Details/comments about the invoice.

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**11.4.3. GET\_INVOICE\_ITEMS**

Returns all the invoice line items (standard, product and time items) but does not return tax or payment items.

**Required Permissions**

- accounts\_ar\_view or accounts\_ap\_view

**Input**

id	int	required	Invoice ID
invoicetype	string	required	Invoice Type - “ar” or “ap”.

**Output – Multiple rows structured as:**

itemid	int	ID of item
type	string	Item type - “standard”, “product” or “time”
customid	int	<b>standard:</b> unused <b>product:</b> ID of product <b>time:</b> ID of product used for hour billing blocks
customid_label	string	Human-readable version of the above
chartid	int	ID of the account/chart for the item.
chartid_label	string	Human-readable account/chart name.
timegroupid	int	(time items only) ID of the time group used by the item.
timegroupid_label	string	(time items only) Human-readable name of the time group.
quantity	int	Number of units.
units	string	<b>product:</b> custom field <b>time:</b> always set to “hours”
amount	money	Total amount to charge for item (price * quantity)
price	money	Price per unit
description	string	Description of transaction.

taxes	LIST	Multiple rows of tax settings (see table below)
discount	float	Discount % (only exists for product or time items)

**“taxes” row data:**

The taxes row data is only produced for “standard” invoice item types and shows what taxes have been enabled or disabled for the selected invoice.

taxid	int	ID of the tax
taxid_label	string	Human-readable tax name
status	string	Either “on” or “off”

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.

**11.4.4. GET\_INVOICE\_TAXES**

Returns a list of all tax items on the invoice. Tax items are not normally added or edited directly, but are generated by the tax options assigned to the products and standard items added to the invoice.

**Required Permissions**

- accounts\_ar\_view or accounts\_ap\_view

**Input**

id	int	required	Invoice ID
invoicetype	string	required	Invoice Type - “ar” or “ap”.

**Output – Multiple rows structured as:**

itemid	int	ID of the tax item on the invoice
taxid	string	ID of the tax
taxid_label	int	Human-readable name of the tax.
amount	string	Amount of tax

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.

### 11.4.5. GET\_INVOICE\_PAYMENTS

Returns a list of all payments made against the invoice.

#### Required Permissions

- accounts\_ar\_view or accounts\_ap\_view

#### Input

id	int	required	Invoice ID
invoicetype	string	required	Invoice Type - “ar” or “ap”.

#### Output – Multiple rows structured as:

itemid	int	ID of item
date_trans	string	ID of the tax
chartid	int	ID of the account/chart the tax amount is filed to.
chartid_label	string	Human-readable name of the account/chart.
amount	int	Amount paid
source	string	Source of the payment (eg: “cheque”, “cash”, etc)
description	string	Details about the payment.

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.

### 11.4.6. SET\_INVOICE\_DETAILS

Takes the provided values and creates or updates a tax.

#### Required Permissions

- accounts\_taxes\_write

#### Input

id	int		Tax ID – leave blank to create a new tax.
invoicetype	string	required	Invoice Type - “ar” or “ap”.
locked	int		Locked status – (0 for unlocked, 1 for locked)
orgid	int		Either the customer or vendor ID (depending on invoice type)
employeeid	int		Employee who created the invoice.
dest_account	int		Destination account/chart for the invoice charges – either <i>accounts receivables</i> or <i>accounts payables</i> .
code_invoice	string		Invoice ID/Code
code_ordernumber	string		Order Number
code_ponumber	string		PO Number

date_due	date		Date invoice is due to be paid
date_create	date		Date invoice was added to database (non-changeable)
date_sent	date		(AR only) Date invoice was sent to customer
sentmethod	string		(AR only) Method used to send invoice to customer
notes	string		Details/comments about the invoice.

**Output**

id	int	Invoice ID
----	-----	------------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.
LOCKED	Invoice is locked and can not be adjusted
DUPLICATE_CODE_INVOICE	The requested <i>code_invoice</i> value is already taken by another invoice.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**11.4.7. SET\_INVOICE\_ITEM\_STANDARD**

Takes the provided values and creates or updates a standard invoice item.

**Required Permissions**

- accounts\_ar\_write or accounts\_ap\_write

**Input**

id	int	required	Invoice ID
invoicetype	string	required	Invoice Type - “ar” or “ap”.
itemid	int		Item ID – leave blank to add a new item to the invoice.
chartid	int	required	Account/Chart to bill the item too.
amount	money	required	Amount for the item.
description	string		Description field.

**Output**

itemid	int	Item ID
--------	-----	---------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.

INVALID_INPUT	Problem with supplied data – check that data matches the field type.
INVALID_ITEMID	Supplied <i>itemid</i> does not exist.
LOCKED	Invoice is locked and can not be adjusted
UNEXPECTED_PREP_ERROR	Unexpected failure when preparing item data.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

#### 11.4.8. SET\_INVOICE\_ITEM\_TAX

Used to enable/disable the specified tax for the selected standard item. Note: This feature **only** works for standard type invoice items, attempting to use it with any other item type will trigger an error.

##### Required Permissions

- accounts\_ar\_write or accounts\_ap\_write

##### Input

id	int	required	Invoice ID
invoicetype	string	required	Invoice Type - “ar” or “ap”.
itemid	int	required	Item ID
taxid	int	required	Tax ID to adjust.
status	string	required	Either “on” or “off” to enable/disable the tax.

##### Output

itemid	int	Item ID
--------	-----	---------

##### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.
INVALID_ITEMID	Supplied <i>itemid</i> does not exist.
LOCKED	Invoice is locked and can not be adjusted
UNSUPPORTED_ITEM_TYPE	Attempted to use function on a non-standard item.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

#### 11.4.9. SET\_INVOICE\_ITEM\_PRODUCT

Takes the provided values and creates or updates a product invoice item.

##### Required Permissions

- accounts\_ar\_write or accounts\_ap\_write

##### Input

id	int	required	Invoice ID
----	-----	----------	------------

invoicetype	string	required	Invoice Type - “ar” or “ap”.
itemid	int		Item ID – leave blank to add a new item to the invoice.
price	money	required	Price per unit.
quantity	int	required	Number of units.
units	string	required	Unit label/type
productid	int	required	ID of the product.
description	string		Description field.
discount	float		Discount amount

**Output**

itemid	int	Item ID
--------	-----	---------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.
INVALID_ITEMID	Supplied <i>itemid</i> does not exist.
LOCKED	Invoice is locked and can not be adjusted
UNEXPECTED_PREP_ERROR	Unexpected failure when preparing item data.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**11.4.10. SET\_INVOICE\_ITEM\_TIME**

Takes the provided values and creates or updates a time group invoice item.

**Required Permissions**

- accounts\_ar\_write only

**Input**

id	int	required	Invoice ID
invoicetype	string	required	Invoice Type - “ar” or “ap”.
itemid	int		Item ID – leave blank to add a new item to the invoice.
price	money	required	Price per hour of billable work
productid	int	required	ID of the product to bill the hours as. This product should be some product setup to collect all consulting work.
timegroupid	int	required	ID of the time group to use.
description	string		Description field.
discount	float		Discount amount

**Output**

itemid	int	Item ID
--------	-----	---------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invoice type must be “ar” only – other invoice types do not support the time item.
NOT_AVAILABLE_FOR_INVOICE_USE	See above.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.
INVALID_ITEMID	Supplied <i>itemid</i> does not exist.
INVALID_TIMEGROUPID	Supplied <i>timegroupid</i> is not available for use in this invoice or does not exist.
LOCKED	Invoice is locked and can not be adjusted
UNEXPECTED_PREP_ERROR	Unexpected failure when preparing item data.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**11.4.11. SET\_INVOICE\_OVERRIDE\_TAX**

Provides the ability to override the automatically calculated tax amount on invoices. This feature exists mainly to handle vendors who send AP invoices with incorrect rounding (eg: \$0.01 mistakes).

**Required Permissions**

- accounts\_ar\_write or accounts\_ap\_write

**Input**

id	int	required	Invoice ID
invoicetype	string	required	Invoice Type - “ar” or “ap”.
itemid	int	required	Item ID
amount	money	required	Amount of tax

**Output**

itemid	int	Item ID
--------	-----	---------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.
INVALID_ITEMID	Supplied <i>itemid</i> does not exist.
LOCKED	Invoice is locked and can not be adjusted
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.



### 11.4.12. SET\_INVOICE\_PAYMENT

Takes the provided values and creates or updates a product invoice item.

#### Required Permissions

- accounts\_ar\_write or accounts\_ap\_write

#### Input

id	int	required	Invoice ID
invoicetype	string	required	Invoice Type - “ar” or “ap”.
itemid	int		Item ID – leave blank to add a new item to the invoice.
date_trans	date	required	Data of payment
chartid	int	required	Source/Destination account for payment
amount	money	required	Amount paid.
source	string		Source of the payment (eg: “cheque”, “cash”, etc)
description	string		Details about the payment.

#### Output

itemid	int	Item ID
--------	-----	---------

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.
INVALID_ITEMID	Supplied <i>itemid</i> does not exist.
LOCKED	Invoice is locked and can not be adjusted
UNEXPECTED_PREP_ERROR	Unexpected failure when preparing item data.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

### 11.4.13. DELETE\_INVOICE

Deletes an invoice provided that it isn't locked.

#### Required Permissions

- accounts\_ar\_write or accounts\_ap\_write

#### Input

id	int	required	Invoice ID
invoicetype	string	required	Invoice Type - “ar” or “ap”.

#### Output

returncode	int	
------------	-----	--

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INVOICE_TYPE	Invalid invoice type – must be either “ar” or “ap”.
INVALID_INVOICE	Requested invoice does not exist – check that the ID is valid.
INVALID_INPUT	Problem with supplied data – check that data matches the field type.
LOCKED	Invoice is locked and can not be deleted.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**11.4.14. DELETE\_INVOICE\_ITEM**

Deletes a single item of any type (standard, product, time, tax or payment) from an invoice. The invoice ID and type do not need to be supplied to this function, since they can be queries from the itemid.

**Required Permissions**

- accounts\_ar\_write or accounts\_ap\_write

**Input**

itemid	int	required	ID of item
--------	-----	----------	------------

**Output**

returncode	int	
------------	-----	--

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Problem with supplied data – check that data matches the field type.
INVALID_ITEMID	Supplied <i>itemid</i> does not exist.
LOCKED	Invoice is locked and can not be adjusted
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

## 12. API: Products

### 12.1. products\_manage

Executable	api/products/products_manage.php
WSDL	api/products/products_manage.wsdl
Summary	Provides functions for managing products.

#### 12.1.1. GET\_PRODUCT\_DETAILS

Fetch all the details of the selected product.

##### Required Permissions

- products\_view

##### Input

id	int	required	Product ID
----	-----	----------	------------

##### Output

code_product	string	Product Code
name_product	string	Product name
units	string	Unit measure (5 chars)
details	string	General details/description about the product.
price_cost	money	Price product is purchased at.
price_sale	money	Price product is sold at.
date_current	date	When the quantity & pricing information was last updated.
quantity_instock	int	Number of units in stock.
quantity_vendor	int	Number of units in stock by the vendor.
vendorid	int	ID of the vendor supplying the product.
vendorid_label	string	Human-readable name of vendor supplying the product.
code_product_vendor	string	Vendor's internal code for the product.
account_sales	int	Chart/Account ID that sales are assigned to.
account_sales_label	string	Human-readable label for selected account.
discount	float	Discount % to apply to both price_cost and price_sale

##### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when loading data.

### 12.1.2. GET\_PRODUCT\_TAXES

Returns a list of all taxes attached to the selected product, including whether it is a fixed or automatic tax.

#### Required Permissions

- products\_view

#### Input

id	int	required	Product ID
----	-----	----------	------------

#### Output – Multiple rows structured as:

itemid	int	ID of the tax item assigned to the product.
taxid	string	ID of the tax
taxid_label	int	Human-readable name of the tax.
manual_option	string	If set to “on”, the tax will be fixed to the “manual_amount” provided.
manual_amount	int	Amount of tax charged.
description	string	Human-readable note specific to this particular tax & product.

#### Failure Codes

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_ID	Invalid ID/ID did not match to any records
INVALID_INPUT	Problem with supplied data – check that data matches the field type.

### 12.1.3. SET\_PRODUCT\_DETAILS

Takes the provided values and creates or updates a product.

#### Required Permissions

- products\_write

#### Input

id	int		Product ID – leave blank to create a new product.
code_product	string	required	Product Code
name_product	string	required	Product name
units	string		Unit measure (5 chars)
details	string		General details/description about the product.
price_cost	money		Price product is purchased at.
price_sale	money		Price product is sold at.
date_current	date		When the quantity & pricing information was last updated.
quantity_instock	int		Number of units in stock.
quantity_vendor	int		Number of units in stock by the vendor.
vendorid	int		ID of the vendor supplying the product.
code_product_vendor	string		Vendor's internal code for the product.

account_sales	int	required	Chart/Account ID that sales are assigned to.
discount	float		Discount % to apply to both price_cost and price_sale

**Output**

id	int	Product ID
----	-----	------------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
DUPLICATE_CODE_PRODUCT	The requested <i>code_product</i> is already in use.
DUPLICATE_NAME_PRODUCT	The requested <i>name_product</i> is already in use.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**12.1.4. SET\_PRODUCT\_TAX**

Takes the provided values and creates or updates a tax item belonging to the selected product.

**Required Permissions**

- products\_write

**Input**

id	int	required	Product ID
itemid	int		Tax item to adjust – leave blank to add a new tax to the product.
taxid	int	required	ID of the tax to use
manual_option	string		Set to “1” to make this tax a fixed amount rather than automatic.
manual_amount	money		When using the manual_option, set this to the amount of tax
description	string		Human-readable description for the tax item

**Output**

itemid	int	Tax item ID
--------	-----	-------------

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_ID	Invalid product ID
INVALID_ITEMID	Supplied <i>itemid</i> does not exist.

**12.1.5. DELETE\_PRODUCT**

Deletes the selected product.

**Required Permissions**

- products\_write

**Input**

id	int	required	Product ID
----	-----	----------	------------

**Output**

returncode	int	
------------	-----	--

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
LOCKED	The product can not be deleted because it has been used in invoicing.
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

**12.1.6. DELETE\_PRODUCT\_TAX**

Deletes a particular tax item from the selected product.

**Required Permissions**

- products\_write

**Input**

id	int	required	Product ID
itemid	int	required	Tax item ID to delete.

**Output**

returncode	int	
------------	-----	--

**Failure Codes**

ACCESS_DENIED	Invalid permissions or not logged in
INVALID_INPUT	Provided input unacceptable and caused problems with the input sanitisation.
INVALID_ID	Invalid ID/ID did not match to any records
UNEXPECTED_ACTION_ERROR	Unexpected failure when processing data.

## 13. Sample Code

A number of different sample scripts and programs have been provided which use the SOAP APIs to talk to the billing system which you can modify to use for your own applications.

Most of the same code is licensed under a permissive MIT-style software license, however it can vary, so always check the headers of any files before you use them in your own applications.

### 13.1. Service Collectors

Refer to the service usage collectors manual for information about the sample code supplied for service usage collectors.

### 13.2. Testsuite Scripts

A number of scripts are provided which are used to test the APIs during product development and testing. You can use these scripts as sample code or just to verify correct operation of the API in the event of your own code experiencing problems.

The following is a list of all the test scripts and the APIs that they test. Read the headers of the test scripts for details on the specific actions they perform.

testsuite/hr/staff_manage.php	authentication, hr_staff_manage
testsuite/customers/test_customer_details.php	authentication, customers_manage
testsuite/vendors/test_vendor_details.php	authentication, vendors_manage
testsuite/accounts/test_charts.php	authentication, accounts_charts_manage
testsuite/accounts/test_gl.php	authentication, accounts_gl_manage
testsuite/accounts/test_taxes.php	authentication, accounts_taxes_manage
testsuite/accounts/test_ar_invoice.php	authentication, account_invoices_manage
testsuite/accounts/test_ap_invoice.php	authentication, account_invoices_manage
testsuite/accounts/test_products.php	authentication, products_manage