



# POD Translation by *pod2pdf*

---

[ajf@afco.demon.co.uk](mailto:ajf@afco.demon.co.uk)

# *README*



# Table of Contents

## README

NAME	1
Version	1
DESCRIPTION	1
CONTENTS	1
INSTALLATION	1
Terms and locations	1
Catalog Directory	1
Interchange software directory	1
Interchange User	2
Untar the extension	2
Create and copy directories and files	2
Make orders directory	2
Copy pages	2
Copy report generation file etc/trans_quickbooks	2
Copy usertags	2
Additional database fields	2
Modify catalog.cfg with additions:	3
Add quickbooks order route to order pages	3
Restart the catalog	3
Export the items	3
Test	3
DISCUSSION	3
Sales Tax	4
Shipping	4
Customer Imports	4
AUTHOR	4



## NAME

Interchange QuickBooks — QuickBooks support for transactions and items

## Version

This document describes software based on Interchange 4.5.x.

## DESCRIPTION

Interchange is a business-to-business and business-to-consumer internet ordering and cataloguing product. It has the ability to take orders via the World Wide Web, and store transaction data.

This document describes how to interface Interchange with QuickBooks, the popular small-business accounting program from Intuit.

QuickBooks has an import/export format called IIF, a mnemonic for Intuit Interchange Format. Fitting, eh?

The standard capabilities of Interchange allow production of IIF files for transaction passing. With some support from Interchange UserTags, can even import and export item listings.

## CONTENTS

The following files are used with this extension:

<code>ic_qb.pod</code>	The source for this document
<code>usertag/import_quicken_items</code>	UserTag for importing items
<code>usertag/export_quicken_items</code>	UserTag for exporting items
<code>pages/admin/quickbooks/*</code>	Menu support for Interchange UI
<code>etc/trans_quickbooks</code>	Report file used with quickbooks route
<code>qb.catalog.cfg</code>	Route for quickbooks transaction logging

## INSTALLATION

To install this extension, the basic steps are:

- \* `Untar the distribution to the extensions directory.`
- \* `Create and copy directories and files.`
- \* `Add additional database fields.`
- \* `Modify catalog.cfg with additions.`
- \* `Add "quickbooks" order route to checkout pages.`
- \* `Restart Interchange.`
- \* `Export your items from Interchange catalog (or import your existing QuickBooks items to Interchange).`
- \* `Test.`

## Terms and locations

Several terms are used in the examples.

### Catalog Directory

This is the main directory for the catalog, where `catalog.cfg` resides. It will have a NAME, the name for the catalog. (Some common Interchange demo names are `construct`, `barry`, and `simple`.)

Common locations:

```
/var/lib/interchange/NAME
$HOME/catalogs/NAME
/usr/local/interchange/catalogs/NAME
```

We will use the path `/var/lib/interchange/construct` in these examples.

### Interchange software directory

This is the main directory for your Interchange server, where the file `interchange.cfg` or `minivend.cfg` resides. Common locations:

```
/usr/local/interchange
/usr/local/minivend
/usr/lib/interchange
$HOME/ic
$HOME/mvend
```

We will use the path `/usr/lib/interchange` in these examples.

### Interchange User

The Interchange daemon runs as a user ID that cannot be root. It will require write permission on directories it must modify to do its work.

We will use the user ID `interchange` in these examples.

### Untar the extension

The assumption is that you will untar the distribution tar file to the subdirectory `extensions` in your Interchange software directory. If you stored it in the directory `/tmp`, the command would be:

```
cd /usr/lib/interchange
gzip -dc /tmp/ic_qb*.tar.gz | tar xvf -
```

This will create the directory `extensions` if it doesn't already exist.

If you place the distribution somewhere else you must adjust your paths in the example commands given below.

### Create and copy directories and files

This extension requires you to add some files to your catalog.

It is assumed you have tools and knowledge to create directories with the proper permissions. Any directories that will contain varying files like order transaction logs will require write permission for the Interchange user (daemon); pages and configuration only need have read permission.

#### Make orders directory

Create the directory `orders` in your Catalog Directory if it doesn't already exist. (It may be a symbolic link to another location.) It must have write permission on it.

```
cd /var/lib/interchange/construct
mkdir orders
```

If you are doing this as root, also do:

```
chown interchange orders
```

This directory is used to store the QuickBooks IIF files produced for orders. The files are created with the form:

```
qbYYYYMMDD.iif
```

Each day will have a file, and when a day is complete you should download the orders. (There are other schemes possible.)

#### Copy pages

You will want the Interchange UI support if you are using the UI. It provides links for importing/exporting items, downloading and viewing IIF files, and possibly other functions over time. At the UNIX command line:

```
cp -r /usr/lib/interchange/extensions/quickbooks/pages/admin/quickbooks \
/var/lib/interchange/construct/pages
```

#### Copy report generation file etc/trans\_quickbooks

This file is used to generate the IIF file(s) for transaction oimport into QuickBooks.

```
cp /usr/lib/interchange/extensions/quickbooks/etc/trans_quickbooks \
/var/lib/interchange/construct/etc
```

#### Copy usertags

If you want to use the UI item import/export, two usertags are required. The easiest thing is just to copy them to the Interchange Software Directory subdirectory `dist/lib/UI/usertag`, which is #included as a part of the UI configuration file.

```
cd /usr/lib/interchange
cp -i extensions/quickbooks/usertag/* dist/lib/UI/usertag
```

### Additional database fields

Quickbooks requires an account to debit for the split transactions it uses to track item sales. If you don't create these fields to relate to each SKU, the account "Other Income" will be used in the exports.

Add the following fields to the "inventory" table:

```
account
cogs_account
```

To add the fields in MySQL, you can issue the following queries at the mysql prompt:

```
alter table inventory add column account char(20);
alter table inventory add column cogs_account char(20);
```

Other SQL databases will have similar facilities.

If you are using Interchange DBM files, just export the inventory database, stop the Interchange server (to prevent corruption), add the fields on the first line by editing the `inventory.txt` file, then restart Interchange.

### Modify catalog.cfg with additions:

Add the entries in `qb.catalog.cfg` to `catalog.cfg` (you can use a `#include` statement if you wish).

There are some Require directives to ensure that the needed UserTag definitions are included in the catalog, as well as the Route which is used

### Add quickbooks order route to order pages

Edit the checkout pages, `pages/ord/checkout.html` (and possibly `pages/multi/checkout.html`), in your Catalog Directory. Find the hidden field referring to `mv_order_route`.

If the line looks like this:

```
<INPUT TYPE=hidden NAME=mv_order_route VALUE="log main copy_user">
```

You will want to change it to:

```
<INPUT TYPE=hidden NAME=mv_order_route VALUE="log quickbooks main copy_user">
```

If it looks like:

```
<INPUT TYPE=hidden NAME=mv_order_route
      VALUE="[either]__ORDER_ROUTES__[or]log main copy_user[/either]">
```

you will want to change the Variable/Knar entry `ORDER_ROUTES` to include the quickbooks route.

Place it after the transaction logging step, i.e.

```
Variable ORDER_ROUTES log quickbooks main copy_user
```

### Restart the catalog

This can be done by restarting the Interchange server or by clicking `Apply Changes` in the UI.

### Export the items

You can access the Quickbooks UI index by making your URL:

```
http://YOURCATALOG_URL/admin/quickbooks/index
```

It will provide options for importing and exporting items. This is necessary so QuickBooks will be able to take orders for your items.

QuickBooks uses the product "name" as an SKU, along with an integer reference number. Either you need to make your SKUs match the integer reference number, or you must ensure your product title is unique.

### Test

Place a test order on your Interchange catalog once you have finished installing. You should find a file in the `orders` directory with the name `qbYYYYMMDD.iif`. (YYYY=year, MM=month, DD=day.) Transfer this file to your QuickBooks machine and run File/Import and select that file as the source. This should import the customer and order into the system. If it doesn't work, it may be due to lack of sales tax or shipping definitions, discussed below.

## DISCUSSION

The interface provided works for the sample company data distributed with QuickBooks. There are certain requirements to make sure it works in your environment.

Also, you can change the configuration by editing the file `etc/trans_quickbooks` to suit your IIF file needs.

**Sales Tax**

QuickBooks has a taxing system whereby tax rates are defined by customer location. There is usually also a generic `Sales Tax Item`, such as contained in the sample company data. This allows Interchange to calculate the `salestax`. If that item is not present then you will need to create it.

**Shipping**

Interchange will add a generic item `Shipping` to each order that has a shipping cost. It's `MEMO` field will contain the text description of the mode. If that item is not in your QuickBooks item definitions, then you must create it.

**Customer Imports**

To generate a QuickBooks transtype of `INVOICE`, a `CUSTOMER` is required. Interchange outputs a `CUST IIF` record for each sale with the customer information. Since QuickBooks uses the customer name or company to generate the unique listing, we place the Interchange username in parentheses after the company or name.

**AUTHOR**

Mike Heins, Akopia, Inc. <heins@akopia.com>.