

Handbook of  
SQL-Ledger  
Version 3.2.10

as applied to  
a Religious Entity  
such as a  
Bahá'í Spiritual Assembly

“These will include a well-functioning regional office that provides the secretary with basic organizational support; a sound system of accounting that accommodates diverse channels for the flow of funds to and from clusters...” – Universal House of Justice (CBC Conference 28Dec2010)

Version 0.2 updated 01-Feb-2022

## Table of Contents

The Rationale.....	3
Initial Considerations.....	3
Preparation.....	3
First Steps (SQL-Ledger installed and running).....	5
Chart of Accounts (COA).....	5
System Defaults.....	6
User Management.....	7
User Setup.....	7
Section A: Prep for User Addition.....	7
a. Wage Structure.....	7
b. Statutory Employee Income Tax and Social Security deductions.....	7
c. Access to Menu Items, using Roles.....	9
Section B: Add/Edit User.....	9
Donors and Contributions.....	10
a. Donors (as Customers).....	11
c. Donations – Contributions (as Services).....	11
c.1 Posting a Donation.....	12
Departments and Projects.....	12
Customizations.....	13
Templates.....	13
Other Customizations.....	14
Document Management.....	14
Assets with Depreciable Value.....	15
Search with Delimiter '%'.....	16
References, Tips and Useful Links.....	16
Appendix 1: Backup, Restore and Upgrade SQL-Ledger.....	17
a. Backup, Restore Databases.....	17
b. Backup, Restore SQL-Application Files.....	17
c. Application Upgrade.....	18
Appendix 2: Notes.....	18

## The Rationale

Every Bahá'í Spiritual Assembly requires good financial management. Many Assemblies have yet to have permanent offices and even then, financial work may require acquire a number of assistants. As SQL-Ledger is professional accounting application which is browser-based and can be setup on a virtual server (VPS with an IP), such that work on the ledger can be done on any Internet-linked device, enabling users to work in or out of office.

The features and qualities that make SQL-Ledger\* a very suitable choice for many Spiritual Assemblies:

- 1) open-source application based on PostgreSQL also open-source
  - 2) with warehouse ability (note: not necessarily needed for this user-case)
  - 3) multi-user, multi-currency (note: multi-currency is not very refined and a separate database recommended)
  - 4) browser based – server-driven which can be securely setup on the Internet
  - 5) Mobile-browser usable
  - 6) Double-entry approach
  - 7) Highly Customizable
  - 8) Backend Queries
- \*Creator: **Dieter Simader** (SQL-ledger)

Based on the use of SQL-Ledger (starting with version 2.6.4) from 2006, this writer has gained good experience in creating and using this platform and well as his years of being a Spiritual Assembly treasurer. This purpose of this short manual is to share basic knowledge of the use of SQL-Ledger for any Spiritual Assembly present and future as it is noted that SQL-ledger documentation available at large is quite limited. With the assumption that the Assembly has either elected a Treasurer with some basic understanding of accounts and/or consults with person of good financial experience, this writer created two useful sections for SQL-Ledger on his website: [admin42day.com](http://admin42day.com) The first section under “Server” is to largely assist with the setting up of SQL-Ledger on a Linux server such as Debian or Ubuntu. The second section is focused on the technical use of SQL-Ledger but in a more broader business-oriented approach.

## Initial Considerations

### *Preparation*

Prior to setting up SQL-Ledger, the following needs to be considered:

1. Financial Policy and Practice manual – if it doesn't exist, an effort should be made to address this as it is an important document with any audit that follows.
2. The Chart of Accounts (COA) as required with the Assembly's present and future financial activity. As the business logic is incorporated into SQL-Ledger, lineitems/accounts for Inventory and Cost of Goods (COGS) should not be deleted but closed. Sales Tax can be set to 0%. **Note** that once a transaction has involved an account it cannot thereafter be deleted only closed. For that reason, most of the changes to the COA should be done during the setup of the accounts.

3. As SQL-Ledger was created with the business community in mind, the terms “Employee” is synonymous with “Staff Member” and/or “Volunteer” and “Youth-Year-Service” Similarly, “Customer” could to be viewed as “Donor” and/or “Contributor”.
4. If two or more currencies are dealt with, then it is recommended to have a separate dataset for each currency. Any transaction between currencies has the exchange rate in the notes with a corresponding entry in each ledger.
5. A test database: highly recommended to familiarize oneself and explore the ledger's features.

Before entering data into SQL-Ledger,

- 1) **Basic Assumptions:** As a financial entity that has no sales (but donors and contributors as customers), there are no Goods or Warehouses to consider in developing the Chart of Accounts.
  - a) **Donors, Contributors (Customers):** For any income to the Assembly, a 'Service' can be created and to be posted where that income is allocated. To the Local Fund itself, posting will go into an income account but any Fund outside the Local Fund, specific holding (liability) accounts are to be established. All such contributions will be posted as an 'invoice'. If the contribution has earmarked components, the invoice will charge the corresponding 'Service'. The receipt number can be reflected in the payment section – usually in “Source' box for easy tracking.
  - b) **Vendors:** These are relevant as the Assembly may need supplier services and goods such as a computer with a maintenance contract.
  - c) **Departments:** For any agency of the Assembly, beginning with the Secretariat and even ending with specific committees, this SQL-Ledger facility is useful
  - d) **Projects:** For time-bounded works or events, this facility can be applied as to monitor their specific costs.
  - e) **Assets:** (eg. furniture, office equipment, etc – perhaps even fixed assets: eg. real-estate values) These can be incorporated into the ledger and depreciation applied, where appropriate, on a yearly basis.
  - f) **Source & Memo:** every transaction within the ledger provides for the entry of brief short information in these two categories. It is recommended to use the **Source** for info on the basic legal document be it a cheque number such eg CHQ0023 or a Local Purchase Order LPO0045. For **Memo**, a very concise description of one to three words.
- 2) **A Database With Users:** Initially the 'Admin' user solely establishes other users via the menu item 'HR → Add Employee' and a login name and password is given. For example, the database (called a dataset in SQL-Ledger) for the financial year of 2022-23(or BE179) is “bsa2223” and the users created can then login following a naming pattern [username]@bsa2223. **Note:** Access to menu items can remain or switch off for each user via the 'Access' function on every user's profile. Moreover for a new financial year database is named, say “bsa2324”, and fresh starter dataset (that includes user-profiles but no transactions), the users can then login updated to [username]@bsa2324. This way, the user continues to his/her username/password into the new financial year.
- 3) **Populating the Database with Structure:** This easiest route to populate the database with structure is to use the former year's starting database where the Chart of Accounts, Vendors and Employee profiles etc have all be entered but without any transactions: a starter database. This last year's initial database is then modified to suit the current year –

adding/subtracting the changes in the Chart of Accounts and other sections such as Employee profiles, Donors etc – all prior to any transactions made. Typically it is good to start a month ahead of the beginning of the new financial year (1<sup>st</sup> March) so that structure and users/permissions are reviewed, corrections implemented and by the start of the year all the accounting structure and accessibility is sound and tested. After these preparations, one would save the revised database (still empty of postings) to be as a **starter database** for next year (see section: Establishing a New Accounting Year).

- 4) **Notes concerning the Database:** Experience also show that the database for this ledger's activity grows about 50Kb per week (for a financially active Assembly), accumulating to a file that is nearly 3Mb of size by the end of the year. Given the restraints to bandwidth and to ensure quick backups, it is recommended that, as financial policy, to start a fresh starter database on a yearly basis.

### ***First Steps (SQL-Ledger installed and running)***

- 1) Create the database: Go to the SQL-Ledger administration portal ([http://\[your-server\]/sql-ledger/admin.pl](http://[your-server]/sql-ledger/admin.pl)) then click on “Pg Database Administration” then directly to “Create Database”. Accept the default dialogue by clicking on “continue”.
- 2) In the “Create Database” dialogue box, type in the database name (as an example: bsa2223) then go to the bottom of the page and click on “continue”. This will create the database (“bsa2223”) with the default settings (a VAT tax-related British style Chart of Accounts layout). One may wish to examine the choices for a layout that suits a particular country by creating other datasets with alternative choices.
- 3) Only one user to the database is added at this initial stage, where this user is the “admin” user and is the main controller of this database (eg. bsa2223). The “Administrator” dialog box is simply to state the name of the administrator, but the “Company Name” should be stated as it appears in the reports.
  - 3.1) Add password (at least 9 alphanumeric characters) , email etc. Note that there is no login username to be stated here: it is always “admin” followed by the dataset (eg. admin@ <dataset>).
  - 3.2) Go to the bottom of the page and click “Save”.
- 4) Open SQL-Ledger ([http://\[your-server\]/sql-ledger/](http://[your-server]/sql-ledger/)) and login with username 'admin' followed by the database (eg. admin@ bsa2223) and password then undertake the following:
  - 4.1) Go to “**System**” to make customizations, starting with “**Defaults**” where company information is entered. Keep Precision as “2” or chose “0” (no cents) with the corresponding rounding. The dialogue boxes, those starting with “GL Reference”, are for starting values, where automatic numbering started. See below for more detail.
  - 4.2) Taxes → create VAT 0%, we assume there are no sales taxes unless the database is being considered to be used for a bookstore (a different scenario altogether). However there will be employee taxes to be address in the HR section below.
  - 4.3) In “**Peferences**”, our recommendation for dates is to use the yyyy-mm-dd format.

## **Chart of Accounts (COA)**

The foundation of accounting system is its COA. Given a clear COA, the next step is to harmonize

it with one of the available SQL-Ledger's COA. The default COA for many Assemblies is a good start.

For account to appear in one of the dropdown menus, that account's profile has features to be enabled to permit its appearance. For example at the start of setting up 'HR→ Setup→ **Wages**', the account dropdown menu will not present the correct account (such as Account#5410 Wages & Salaries) because Account#5410 profile has not activated under 'AP' the checkbox 'lineitem'. Go to System→Chart of Account→ List Accounts and click on 5410 to then activate the AP lineitem. Return to '**Wages**' and now that account 5410 will appear.

As from this example and where one can not find the desired account in any dropdown menu, then examine one of the accounts that have appeared, find it in the COA to note what feature(s) has been activated in its profile. From the example, activate the same for the desired account so that it will also appear in the dropdown menu. In the same understanding, one can remove from the menu by deactivating those features in the account profile that cause it to appear.

## System Defaults

In the menu item System→ Defaults, it is suggested to activate Reporting Cash and Sort Names by Number. The precision set for 2 will give cents. One useful settings in the 'Last Number' section is to define how transaction numbers are presented. Creating a consistent structure will prove very useful in searching for items in the ledger. Numbers can start with 3-4 alphacharacters follow by 0001 as to have 4 digit auto-numbering eg. CCCNNNN. Leaving these blank will have the auto-numbering start with a single digit '1' unless these numbers are overruled by the numbers set manually (in any transaction or profile creation). This manual setting is particularly useful when creating profiles for employees, services, customer/donors and vendors are given a **few distinct alpha-numeric characters** such as CBC -for Continental Board of Counsellors. With such few memorable characters, one can easily search for a specific profile or transaction. Here below is a suggested numbering scheme. **Note:** Automatic numbering with characters, despite ending in 1, will give 2. To correct this, manually adjust the first transaction to end with 1 and return to 'System→ Defaults' and reset back to 1.

System	Last Numbers
Defaults	GL Reference Number <b>GL0001</b>
Audit Control	Sales Invoice/AR Transaction Number <b>DTN0001</b>
Audit Log	Sales Order Number
Bank Accounts	Vendor Invoice/AP Transaction Number <b>APT0001</b>
Taxes	Batch Number <b>BAT001</b>
Currencies	Voucher Number <b>VCHR001</b>
Payment Methods	Purchase Order Number
Workstations	Sales Quotation Number
Roles	RFQ Number
Warehouses	Part Number
Departments	Job/Project Number
Type of Business	Employee Number
Language	Customer Number
Mimetypes	Vendor Number
SIC	
Yearend	
<b>Maintenance</b>	
Repost Invoices	
Mapfile	
Clear Semaphores	
Lock Dataset	
Unlock Dataset	
Restore	
Monitor	
<b>Backup</b>	

# User Management

- done by the dataset 'admin' or any user with access to the 'HR' section

## User Setup

Prior to adding any employee, the wage scales with their respective Social Security (eg. NSSF) and employee income tax (PAYE) rates need to be set. Also preferable to establish 'Roles' and 'Departments' but these can be set later. When in HR→Employee →Add Employee, this is where other users of the ledger can be connected. **Only** those where one adds a “login” and “password” can that employee access the ledger and, via 'Access', priviledges granted according to menu items left active. The use of Roles makes this granting of priviledges much easier.

## Section A: Prep for User Addition

### a. Wage Structure

Establish the wage structure via HR→Employee→Setup→ Wages:

continue as to complete the full structure, having something similar to this:

b.

Description	Amount	
Wages-Level01	300,000.00	5410
Wages-Level02	600,000.00	5410
Wages-Level03	1,000,000.00	5410

### Statutory Employee Income Tax and Social Security deductions

Establish these structures via HR→Employee→Setup→ Deductions:

Admin Testdb Edit Deduction

Description \*

Employee pays x

Employer pays x

Exempt age <  >   DOB

Rate %	Amount	Above	Below
10	30,000.00		

Based on  Deduct  Withholding  %

for Social Security where Employee and Employer have both an obligation:

Admin Testdb Edit Deduction

Description \*

Employee pays x

Employer pays x

Exempt age <  >   DOB

Rate %	Amount	Above	Below
5	15,000.00		

Based on  Deduct  Withholding  %

The resulting list of deductions may have a similar table as this:

Admin Testdb Deductions / TestCompany

Description	Rate %	Amount	Above	Below	Based on	Employee	Account	Employer	Account
NSSF Employee Level01	5	15,000.00				1	2250	2	5430
NSSF Employee Level02	5	30,000.00				1	2250	2	5430
NSSF Employee Level03	5	50,000.00				1	2250	2	5430
PAYE-Employee Level 01	10	30,000.00				1	2280		
PAYE-Employee Level 02	10	90,000.00				1	2280		
PAYE-Employee Level 03	10	200,000.00				1	2280		



### c. Access to Menu Items, using Roles

Via 'System→ Defaults', one could establish roles, for example (with diminishing privileges); Supervisor, Manager, and Assistant. Such 'Roles' make it easy to quickly establish a new user particular access rights. According to menu sections are featured on the left and sub-sections on the right. Click checking on or off the sections or sub-sections switches the user access accordingly.

Roles / TestCompany			
No.	Description	Disable	
1	Supervisor	Stylesheet Preferences	↕ ↕
2	Manager	Logistics Quotations Stylesheet Preferences	↕ ↕
3	Assistant	AP Vendors HR Order Entry Logistics Quotations Goods & Services Projects & Jobs Exchange Rates Import Export System Stylesheet Preferences	↕ ↕

Logistics  
Quotations  
General Ledger  
Goods & Services  
Projects & Jobs  
Reference Documents  
Image Files  
Reports  
Recurring Transactions  
Batch  
Exchange Rates  
Import  
Export  
System  
Defaults  
Audit Control  
Audit Log  
Bank Accounts  
Taxes  
Currencies  
Payment Methods  
Workstations  
Roles  
...

Add Role

Without 'Roles', access control is then manually set for each user which needs careful consideration as the default is all sections/subsections are allowed.

In a similar fashion, one can create **departments** to improve cost analysis. Suggested departments are as follows: Secretariat, Treasury, and other agencies/committees. The creation of **projects** for a similar purpose of tracking costs are for time-bound works such as Local Centre Construction or a community service project which again will last only for a certain period.

### Section B: Add/Edit User

1. **Employee No.:** Unless set automatically in 'Defaults', this is very useful to track the employee.
2. **Name:** This must be entered to establish the Employee profile. Employees are treated essentially as 'vendors' and therefore it is beneficial to distinguish between the two by giving the employee number an initial letter 'Z' as well as for the name. If the employee is called 'John Mukasa' for example, it is recommended the employee number be something like ZJMK and name ZZ John Mukasa. If so labelled, the Vendor dropdown menu will have actual Vendor names above those of employees.
3. **Login:** [[name@bsa2223](mailto:name@bsa2223)] is suggested where the current Bahá'í year is BE179. Former years will have bsa2122 for the year 2021-2022 or Bahá'í year 178.
4. **Password:** [6 alphanumeric characters]
5. **Department:** if left blank, user has access to all departments
6. **Role:** If Roles are set in the 'Defaults'. This is useful for quickly setting access privileges.

Admin Testdb

**Edit Employee**

Employee Number EMP01

Name \* Z01-Kitimba

Address

City

State/Province

Zip/Postal Code

Country

E-mail

Role Assistant

Login kitimba

Password 1234ABC

E-mail TAN

Sales

Bank

Address

City

State/Province

Zip/Postal Code

Country

Work Phone

Work Fax

Work Mobile

Home Phone

Home Mobile

Startdate 2022-01-28

Enddate

SSN

DOB

Notes

IBAN

BIC

Member No.

Clearing No.

**Reference Documents**

Wages	Payroll Deduction	Exempt	Maximum
Wages-Level01	NSSF Employee Level01		
	PAYE-Employee Level 01		

AP 2110--Accounts Payable-Staff

Payment 1060--Checking Account

Pay Rates

Over

Pay Periods 12

With the preparations done, one can use the payroll facility effectively:

**Reports**

Search

History

**Cash**

**Vouchers**

Payable

Payment

Payments

Payment Reversal

General Ledger

**Reports**

**HR**

**Employees**

Add Employee

Reports

**Payroll**

Add Transaction

Transactions

**Setup**

Wages

Deductions

**Order Entry**

**Logistics**

**Quotations**

**General Ledger**

**Goods & Services**

**Edit Payroll**

Employee \* Z01-Kitimba

Pay Period Ending 2022-01-28

Wages-Level01 1 x 300,000.00

Department Treasury

Project

Description Sal-Dec21-Kitamba

Source xhq001

Memo Sal-Dec21-Kitamba

Paid 255,000.00

Date 2022-01-28

Payment 1060--Checking Account

AP 2110--Accounts Payable-Staff

**Pay Slip 1**

Wages-Level01	300,000.00
NSSF Employee Level01	-15,000.00
PAYE-Employee Level 01	-30,000.00
<b>Net</b>	<b>255,000.00</b>

**Reference Documents**

Pay Slip

html

Screen

On Hold  Printed

Update Preview Print Post Print and Post Post as new Print and Post as new Delete

Note that full payment does not have to be made at once. If there are balances, these can be settled later via Vouchers→Payments. **The voucher needs to be processed:** Go to Vouchers->Reports->All Batches, press continue and the pending batches will appear. Click on the batch number to open it and post it. Here one still has an opportunity to modify/delete the batch if errors are found.

## Donors and Contributions

Although the terminology here will be 'Customers', donors to the Assembly will be set-up as Customers so that their contributions will handled by the SQL-Ledger as having 'purchased' services.

**Contributions in a Foreign Currency:** It is recommended not to use the ledger's multi-currency facility but to create a parallel database (without any HR requirements) to handle a foreign currency generally, say the US dollar. Experience has shown that the varying exchange rates, can lead to inaccurate report results. Moreover both currencies are mixed in the ledger reports (unless converted in every transaction to local currency), making it difficult to audit. Where a conversion transaction is needed between the two ledgers, it is simply a General Ledger posting, with common, transaction number and note on the exchange rate, in each ledger to debt and credit according.

**a. Donors (as Customers)**

Examine the Assembly's sources of income. A generic 'Customer' labelled 'BahaiMbr' or 'BahaiLocal', for example, can cover for both individual member or collective donations. Other examples of 'Customers' may include 'ConsellorOffice' or even 'NonBahai' covering outside sources of donations directed towards community-wide projects. Some countries may require that each donation to have the tax number of the donor for tax-exemption purposes. In that case, each donor would be registered in the ledger as a customer with his/her tax information.

The screenshot shows the 'Add Customer' form. On the left is a navigation menu with options like AR, POS, Customers, Reports, AP, Vendors, Cash, Vouchers, HR, Order Entry, Logistics, Quotations, General Ledger, Goods & Services, Projects & Jobs, Reference Documents, Image Files, Reports, and Recurring Transactions. The main form area has a title bar 'Add Customer' and a 'Billi' button. Below that, there are radio buttons for 'Type' with 'Company' selected. The form contains several input fields: 'Customer Number' (BBR), 'Customer \*' (BahaiMbr), 'Address', 'City', 'State/Province', 'Zip/Postal Code', and 'Country'. At the bottom, there are checkboxes for 'VAT (0%)' and 'Tax Included', and dropdown menus for 'AR' (1200--Accounts Receivables) and 'Payment' (1060--Checking Account). There are also input fields for 'Credit Limit', 'Threshold', and 'Terms Net' (days).

**c. Donations – Contributions (as Services)**

Similarly, donations will be treated as services (eg. Local Fund, National Fund etc).

The screenshot shows the 'Edit Service' form. At the top is a title bar 'Edit Service'. Below it is a table with two columns: 'Number' and 'Description'. The first row has 'FundNatl' in the 'Number' column and 'National Fund' in the 'Description' column. Below the table, there are dropdown menus for 'Income' (2350--FundNatl) and 'Expense' (5010--Purchases). There is a 'Tax' checkbox with the value '2320--VAT (0%)'. Below that is a 'Link Accounts' section with a 'Reference Documents' button and a checkbox. At the bottom left is a 'Notes' section with a text area. On the right side, there are several fields: 'Updated' (2022-02-02), 'Sell Price' (1), 'List Price', 'Last Cost', 'Markup %', 'Unit', and 'Obsolete' checkbox.

In the 'Notes' section, any info put there appears as 'Internal Notes' with future invoicing/donations

## c.1 Posting a Donation

These are the steps:

1. Via 'AR→ Sales Invoice, select the 'Customer' (eg. BahaiMbr).
2. If not automated, enter in 'Invoice' a donation number (for tracking) such as 'Donation001'.
3. For 'Number' enter the particular Fund Name (eg. FundLocal).
4. For quantity, enter the amount of the donation, then 'update'.
5. In the 'Payments' section, enter the date.
6. For the Source: enter Cash or the Cheque number. (eg.Receipt1002 no special characters)
7. For the Memo: the Receipt number (the same in the Notes Section for better tracking) and finally the amount .
8. **Note:** Pledges could be treated as donations but with deferred payment.

Edit Sales Invoice / TestAssembly

<p>Customer * <input type="text" value="BahaiMbr"/> ?</p> <p>Customer Number <input type="text" value="BBR"/></p> <p>Address <input type="text"/></p> <p>Credit Limit <input type="text" value="0"/> Remaining <input type="text" value="0"/></p> <p>Record in <input type="text" value="1200--Accounts Receivables"/></p> <p>Currency <input type="text" value="UGX"/></p> <p>Shipping Point <input type="text"/></p> <p>Ship via <input type="text"/></p> <p>Waybill <input type="text"/></p>	<p>Invoice Number <input type="text" value="DTN0002"/></p> <p>Order Number <input type="text"/></p> <p>Invoice Date * <input type="text" value="2022-02-03"/></p> <p>Due Date <input type="text" value="2022-02-03"/></p> <p>Terms Net <input type="text"/> days</p> <p>PO Number <input type="text"/></p>
---	--

Reference Documents

?

DCN

Description

Item	Number	Description	Qty	OH	Unit	Price	%	Extended
1	FundLocal		120,000			1		120,000
2								

Notes	Internal Notes	
<input type="text" value="RCPT1003"/>	<input type="text"/>	Subtotal 120,000
		Total 120,000

Total Cost: 0 Margin: 100.0

Payments

Date	Source	Memo	Amount	Account
<input type="text" value="2022-02-03"/>	<input type="text" value="CHQ22"/>	<input type="text" value="RCPT1003"/>	<input type="text" value="120,000"/>	<input type="text" value="1060--Checking Account"/>

## Departments and Projects

Creation of Departments is useful to each track of costs for administrative branches that are established on a long-term basis. Every Assembly, for example, will have a Secretary with possibly a Secretariat as well as a Treasurer, again possibly with Treasury. Moreover there will be agencies such as committees that are established on the long-term. All these would be suited to be defined as departments in the ledger. Projects on the other hand are for cost centres of relatively short or fixed duration, such as the construction of a building or a task-force for a specific purpose and possibly funded by a particular donor. The difference how these two function is that an entire transaction can be set to department whereas a lineitem in a transaction can be set to a particular project. The setup of a department is done in System→ Departments whereas a project is set under menu item: Projects & Jobs.

Departments / TestCompany			
Description	Cost Center	Profit Center	
Secretariat	*		↕ ↕
Treasury	*		↕ ↕
Local Growth Committee	*		↕ ↕

[Add Department](#)

Add Project	
Number	LocalCtr
Description	Building Local Center
Customer	?
Startdate	2022-02-03
Enddate	
Reference Documents	
Summary Contractor Bids	<input type="checkbox"/> ?
	<input type="checkbox"/> ?

[Update](#) [Save](#)

## Customizations

### Templates

Our main interest are the templates that generate .pdf files. These are found in System->LaTeX Templates which are all based on LaTeX coding. Therefore it makes sense to use a text editor such as NotePad++ (not the MS Notepad which will not serve one well here) to edit these files. Generally, this will be the process:

1. Select and open the specific template to be edited (eg. Sales Invoice)
2. Click the “Edit” button at the bottom of the template page
3. Mouse click on the template and press Ctl-a to highlight all text and Ctl-c to copy with the purpose to save the original in case editing goes wrong.
4. Paste into a new NotePad++ file and save as [template-nameV1].tex (eg Sales Invoice.tex) then save again as [template-nameV2].tex
  - a) The resulting file will give one a highlighted text that is searchable for key words. If it fails to render, restore it with the original and try again.
  - b) It is worthwhile:
    - First review basic LaTeX terminology (via the Internet)
    - to look over the existing template for its structure and experiment with little blocks of code. After all, one can always cut and paste the original code back to the template.
    - Viewing other templates (.eg by producing their respective .pdf file) can be helpful in finding the suitable block of code to implement a change.

5. Edit the template to suit your needs.
  - a) Eg `<%businessnumber%>` will place the TIN number as entered in System->Defaults-Business Number box, `<%company%>` the Assembly's name.
  - b) `\textbf{some text now in bold}` gives: **some text now in bold**
6. Save modified template for easy restore if the restore is on a fresh setup.

## Other Customizations

Replacing SQL-Ledger logo with company logo in the templates, following these steps:

1. use a graphic application such as GIMP to create a logo (of a size in the range of ~100 x100pixels) and then saved it as logo.eps and logo.png
2. According to each dataset; put these in the directory:  
`/usr/local/sql-ledger/templates/<dataset-name>` replacing the same-named files.
3. To change SQL-Ledger logo for the ledger's welcome page: copy and rename the files; logo.eps and logo.png to sql-ledger.eps and sql-ledger.png and replace those same-named files in the directory: `/usr/local/sql-ledger/images/`

## Document Management

One can attach a document to specific transactions and profiles within SQL-Ledger. The document need to be of a type that can be seen within browser: that is .pdf file or an image file - .png, .jpg. One can associate a document to a transaction by checking on the red “?” which leads to this dialogue page:

The screenshot shows a web-based dialog box titled "Upload Document". It contains the following fields and controls:

- Description:** A text input field containing "SalesAgreement-ToyotaWish -cb.PNG".
- Folder:** A text input field containing "refer".
- Filename:** A text input field containing "cb.PNG".
- File:** A button labeled "Browse..." followed by the text "No file selected.".
- Continue:** A button at the bottom left of the dialog.

Whereby one uploads the file once chosen through 'Browse'. Note that the folder 'refer' is but a freely chosen tag to ease the sorting of documents but not a folder that is found on the actual Operating System. Secondly, there is not a option to select a file that has been earlier uploaded. Lastly, one can remove a document from a transaction by simply highlighting and removing the title followed by updating and (re)posting the transaction.

The 'List Document' features not only means to delete files but provides choices that make finding files easier such as the 'Folder' tag – entering the folder name results is showing only those files in that 'folder'. Clicking on any of the column headers will result in a sorting according to that column. Clicking twice reverse the order.

It is recommended not to add files via 'List Document' but via specific transactions.

**Reference Documents**

<input type="checkbox"/>	Description	Folder	Filename	C	Attached to
<input type="checkbox"/>	SalesAgreement-ToyotaWish -cb.PNG	refer	cb.PNG		GL Transaction - ID 10092

[Add Document](#) [Delete Documents](#)

### Assets with Depreciable Value

SQL-Ledger provides useful tools to manage assets. Take for simple example the purchase of a vehicle and using the default COA lineitem#1840, one would setup a GL transaction similar to the following (note that one can attach the initial sales agreement):

**Add General Ledger Transaction**

Reference:  Date:

Department:  Currency:

Description:

Notes:

**Reference Documents**

?

Account	Debit	Credit	Source	Memo
1840--Vehicle	7,500,000		P0035	VehiclePurchase
1060--Checking Account		7,500,000	Chq099	VehiclePurchase
<input type="text" value="1060--Checking Account"/>				
	7,500,000	7,500,000		

[Update](#) [Post](#) [Schedule](#) [New Number](#)

Then SQL-Ledger provide a “Contra” lineitem where one with would deduct on a yearly basis the accumulating depreciation of the vehicle's value. Following the above and using a 20% depreciation value (value applied and the manner such deductions vary to country and local financial culture), one could see another GL transaction at the end of the year of this nature:

**Edit General Ledger Transaction**

Reference:  Date:

Department:  Currency:

Description:

Notes:

**Reference Documents**

Account	Debit	Credit	Source	Memo
1845--Accum. Amort. -Vehicle		1,500,000	EndYrNotes	AnnDep-Vehicle
3360--LastYear Carry-over Value	1,500,000		EndYrNotes	AnnDep-Vehicle
<input type="text" value="1060--Checking Account"/>				
	1,500,000	1,500,000		

[Update](#) [Post](#) [Post as new](#) [Schedule](#) [New Number](#) [Delete](#)

Using the contra account (#1845) enable the Treasurer to keep track of the original purchase price and to keep adding year after year to the contra account until the purchase value equals the accumulated depreciation value.

## Search with Delimiter '%'

Upon opening General Ledger → Report, one finds a wide range of options and specific data areas where to make searches for any kind of transaction. A powerful delimiter, the '%', is available to enhance and quickly scope a desired range. Using the word 'oil' as an example with the delimiter in the 'Memo' section, one would get these results:

- oil % (note the space within) – all memos that begin with the word 'oil' with or without the capital
- oil% – all memos that begin with the three letters 'oil'
- % oil % (spaces both sides) – all memos that have the word 'oil' within the memo
- %oil % – all memos that have words ending with the three letters 'oil'
- % oil – all memos that end with the word 'oil'

In a similar fashion, one can search with numbering eg. %78% – this will search that number pair associated with related transactions, if applied, say, in the 'Source' section.

As briefly mentioned in 'Preparation', well-thoughtout use of Notes, Description, Reference, Source and Memo sections that are consistent and well-structured will great facilitate precise searches and reports.

## References, Tips and Useful Links

- a) Warning: using the browser previous page button to return to a posted invoice to make changes will result into a duplicated invoice, if not for the purpose of creating a new invoice.
- b) Warning: **always update before posting.** Failing to do so can lead to unexpected results.



## Appendix 1: Backup, Restore and Upgrade SQL-Ledger

The Main Rule here: databases come first then the application files. Therefore in during a full **backup**; first backup databases then application files. Similarly during a full **restore**; do restore the databases first then the application files.

### a. Backup, Restore Databases

At times the Internet connection may be down or worst, a rare attack on the Internet server that leaves it non-functional. It is therefore important to backup the database on a daily basis or even hourly depending on the amount of data being inputted into the database. With a current backup file, one can quickly restore to a local server or to a refreshed Internet server.

1. **Backup:** To backup the database structure and transactions, one selects within SQL-Ledger's table of actions: **System->Backup->Save to File**. A dialogue box should pop-up requesting one to save the backup file whose title should be in the form of
  - a) **[database-name]-[version of SQL-Ledger]-[current date].sql.gz**. For example: `bsa2223-3.2.10-20220822.sql.tar.gz`. **Note:** If one chooses "Send by Email" instead of "Save to File" then these backup files just accumulate in the "dataset" directory on the server.
2. **Restore:** To restore to the server, go to System→Maintenance→Restore. One can select and restore from any 2.8.# to 3.2.# database. Usually a warning will be given if it doesn't match host dataset but it will successfully overwrite any existing database.

### b. Backup, Restore SQL-Application Files

While the database backup and restore safeguards the data, all the customization; eg the logos, the templates and even the user accounts are all found in their respective sub-directories of the SQL-Ledger directory; specifically – templates/ and users/. However, it is sometimes practical to backup the entire sql-ledger/ directory to ensure all customizations are captured as well as offering a quick full SQL-Ledger application restore. To do so, do the following:

Assumption: On a running Linux server where SQL-ledger was previously installed.

- 1) **Backup:** Using PuTTY, and as user: sammy go to the sub-directory `usr/local`

#### Step 1

```
sammy@imega: cd /usr/local/
```

#### Step 2

Compress the entire sql-ledger directory into a file: `bsa2223[SQL-Ledger-version][date].tar.gz` as follows:

```
sammy@imega: sudo tar czvf bsa2223-3.2.10-20220822.tar.gz sql-ledger/
```

#### Step 3

copy the compressed file to a home directory which a user can FTP the file to storage.

```
sammy@imega: cp bsa2223-3.2.10-20220822.tar.gz /home/sammy/
```

- 2) **Restore:** Upload the compressed file to the server. In PuTTY and as user: sammy, starting from `/home/sammy`, take the following steps:

#### Step 1

```
sammy@imega: sudo cp bsa2223-3.2.10-20220822.tar.gz /usr/local/
```

Go to the usr/local directory:

```
sammy@imega: cd /usr/local/
```

## Step 2

Decompress the file, using this command:

```
sammy@imega: sudo tar xzvf bsa2223-3.2.10-20220822.tar.gz
```

Check the directory, using the command “ls -ls” and the sql-ledger should have appeared.

## Step 3

Set permissions

```
sudo chmod -R 775 sql-ledger
```

One has completed the application restore. For a full restore, the admin's of the related databases need to restore them with their backups via 'System→ Maintenance→ Restore'.

## c. Application Upgrade

When the SQL Application has an upgrade, it requires steps done directly via the host server console. Once the steps are completed, one will check via the browser if the application has been upgraded. The steps are as follows:

- 1) **Backup First prior the Upgrade:** To ensure a restore point to fall back to, do that which is described in section a. & b.
- 2) **Logon to the host server:** Via puTTY, logon on as one of the users (eg. sammy). Do the following commands:

```
sammy@imega:$ cd /usr/local/sql-ledger
```

```
sammy@imega:/usr/local/sql-ledger$ sudo wget https://sql-ledger.com/source/setup.pl
```

```
sammy@imega:/usr/local/sql-ledger$ sudo perl setup.pl // Choose 'i' for install.
```

## Appendix 2: Notes

1. SQL-Ledger installation on a Debian/Ubuntu server: <https://admin42day.com/server/ServerLedger/>
2. SQL-Ledger Usage (more business-oriented) at website 'Admin42day.com': <https://admin42day.com/ledger/LedgerIntro/>
3. SQL-Ledger -Network, dated but still useful information: [http://www.sql-ledger-network.com/dokuwiki/doku.php?id=user\\_manual:user\\_manual](http://www.sql-ledger-network.com/dokuwiki/doku.php?id=user_manual:user_manual)
4. PostgreSQL queries: It is possible to extract customized reports direct from the database; for an example, take this SQL query (the results can be copy-pasted in a spreadsheet):

```
SELECT  
ar.invnumber, ar.transdate, c.name AS customer, ar.netamount, ar.amount - ar.netamount AS tax,  
ar.amount, ar.paid, ar.invoice  
FROM ar  
JOIN customer c ON (c.id = ar.customer_id);
```