

# March 2014 Features Update

## Table of Contents

Global.....	3
Family Tree Enhancements.....	4
Census Database Enhancements.....	9
Vital Statistics Enhancements.....	10
Bug Fixes.....	12

## Illustration Index

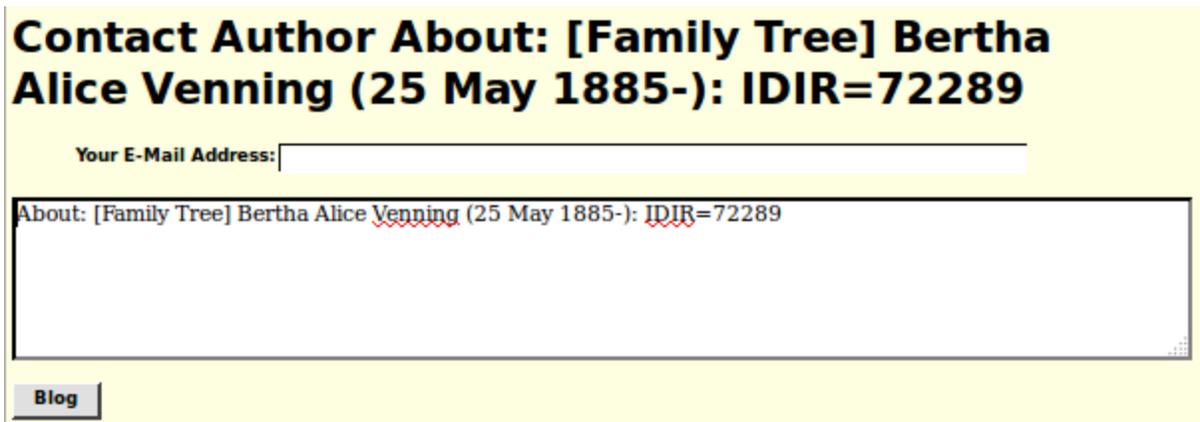
Illustration 1: Nominal Index with Birth Year Range.....	4
Illustration 2: Edit Families Form.....	6
Illustration 3: Adding a Child to the Marriage Form.....	7
Illustration 4: Multiple Matches for a Partial Location Name.....	8
Illustration 5: Marriage Registration Query.....	10
Illustration 6: Edit Marriage Registration Dialog.....	11

## Global

The restructuring of input forms to use CSS for layout rather than HTML tables is complete.

There was a serious privacy exposure because the e-mail addresses of all contributors who have permission to update a particular individual were exposed in the page footer so that the “Contact Author” link would notify all of them. This is resolved by no longer using e-mail as the notification method. Instead these notifications become “blog” entries attached to the user account record. This also resolves an issue if the visitor has not configured their browser with the required information for sending e-mail messages.

When a guest, that is a visitor who is not signed in as a contributor, clicks on the “Contact Author” link on the bottom of a page now sees a new “Contact Author” page:



The screenshot shows a web form titled "Contact Author About: [Family Tree] Bertha Alice Venning (25 May 1885-): IDIR=72289". Below the title is a text input field labeled "Your E-Mail Address:". Underneath is a large text area containing the text "About: [Family Tree] Bertha Alice Venning (25 May 1885-): IDIR=72289". At the bottom left of the form is a button labeled "Blog".

*Illustration 1: Contact Author Form*

The only difference if you are a signed on subscriber is that the e-mail address field is not present, since that information is available from your subscriber record. If the guest does not fill in the e-mail address then there is no way for the author to contact them. The user can add any additional text to the message, and then click on the “Blog” button.

Formerly when you as a subscriber signed on to the network you were just presented with the general page. However if someone has posted a message to you as one of the “authors” of the individual in question then you will see a page like:

## Welcome jcobban

At 2014-03-31 03:58:32 user 'jacobban' wrote:

About: [Family Tree] Bertha Alice Venning (25 May 1885-): IDIR=72289

Testing

Sender's e-mail address: [garbage email address](#)

---

Delete Blog

Reply

*Illustration 2: Messages Displayed During Signon*

If the message is from a subscriber then you can reply to it by filling in the text area and clicking on the “Reply” button. The next time that subscriber signs in they will see the message. Otherwise you can click on the e-mail address link to send the guest or subscriber a message by that medium. Or you can click on the “Delete Blog” button. When there are no more pending messages you will be presented with the nominal index search dialog.

### **Registration Improvements**

The registration dialog now provides more guidance to the user. Previously invalid or questionable input values would only be reported after the user had submitted the form. Now they are reported dynamically as the user enters the data. For example if there are unsupported characters in the user name or e-mail address the entered value is displayed in red as a warning. If the two password values do not match an alert is popped up. Also the password value is evaluated to determine whether it adequately protects your account.



*Illustration 3: Warning of Weak Password*

The Account Management panel is similarly enhanced.

### **PHP Class Restructure**

Although the internal restructuring of the functionality of the classes used to encapsulate the functionality of the database records on the Web server does not usually result in externally visible behavior changes it should result in more consistent reliable behavior and easier enhancement. The objective is that the application functionality, presented by page specific scripts, should access the database exclusively through the class wrappers, rather than through specific database requests. One of the reasons for this is that although all database server implementations theoretically support the same Structured Query Language (SQL) there are very significant differences in the exact dialect each product implements. Also as new functionality is added, for example the ability to sort pictures by date which is introduced this month, the exact set of information in each record changes, and field names may change. For example each `LegacyPicture` record formerly contained a field named "PicDate" which contained an uninterpreted textual date associated with the picture for display purposes only. This is replaced by a pair of internally coded fields, "PicD" and "PicSD" which support the new sort capability. However the application layer continues to present the illusion that the "PicDate" field still exists. By accessing the database information only through class wrappers the application code is insulated from these considerations.

## Family Tree Enhancements

The cause of death field is now edited as part of the details on the Death event, rather than taking up an extra line in the Edit Individual page. For clarity the cause of death portion of the Death event panel is enclosed in a field-set boundary.

**Edit Death Event for Mary Augusta Lang** [? Help](#)

Date: 20 Feb 1939

Location: London, ON, CA

Notes:

Path:

Citations:		Source Name	Details (Page)	
<input type="button" value="Edit Citation"/>		Cemetery Transcription, Mt. Pleasant Cemetery, London, ON	H-8-19	<input type="button" value="Delete Citation"/>
<input type="button" value="Add Citation"/>				

Cause of Death

Cause of Death:

Citations:	Source Name	Details (Page)
<input type="button" value="Add Citation"/>		

Illustration 4: Edit Death Fact with Cause of Death

The alternate names section of the Edit Name Event panel is enclosed in a field-set boundary for clarity.

**Edit Name Fact for Mary Augusta Lang** [? Help](#)

Surname:

Given Name:

Name Prefix:

Name Suffix:

Notes: 

Font family Font size Format

Path:

**Citations:**

	Source Name	Details (Page)	
<input type="button" value="Edit Citation"/>	<input type="text" value="Marriage Register, CA, Ontario"/>	<input type="text" value="1877-06421"/>	<input type="button" value="Delete Citation"/>
<input type="button" value="Add Citation"/>			

**Alternate Names**

Alternate Name:

**Citations:** **Source Name** **Details (Page)**

New Surname:

New Given Name:

Illustration 5: Edit Name Fact with Alternate Names

The “Order Pictures byDate” button on the Edit Pictures dialog now works.

**Edit Pictures for Indiv James Alan Cobban** [? Help](#)

Caption	File Type	Date	Action	
<input type="text" value="jim as an infant"/>	<input type="text" value="Image File"/>	<input type="text" value="1951"/>	<input type="button" value="Edit Picture"/>	<input type="button" value="Delete Picture"/>
<input type="text" value="jim at 15"/>	<input type="text" value="Image File"/>	<input type="text" value="1966"/>	<input type="button" value="Edit Picture"/>	<input type="button" value="Delete Picture"/>
				<input type="button" value="Add Picture"/>

Illustration 6: Edit Pictures Dialog with Order Pictures by Date Button

The Add/Edit Parents dialog now shares almost all of the functionality and appearance of the Add/Edit Family dialog, including the ability to add children.

## Census Database Enhancements

There are only minor bug fixes to the Census pages.

The script `database/SqlCommand.php`, used to issue SQL commands against the database directly, is enhanced to support the AS keyword. For example you can now enter:

```
“SELECT Surname, COUNT(*) as num FROM tblIR GROUP BY Surname ORDER BY num  
DESC LIMIT 100”
```

to get a list of the 100 most common surnames in the family tree.

## Vital Statistics Enhancements

There are only minor changes to Vital Statistics support.

The Death Registration Update dialog has expanded the number of characters that can be recorded for birth and death dates to 32 characters, and the number of characters of occupation to 64. Also the panel now permits your browser to perform auto-completion functionality. The list of family tree individuals matching an entry now includes the names of their parents. The link to a match in the family tree is now colour coded by gender. The selection lists for sex and marital status are now initialized by the web server, rather than by code running on the browser.

In the Birth Registration Update dialog the list of family tree individuals matching an entry now includes the names of their parents. The link to a match in the family tree is now colour coded by gender. The selection list for sex is now initialized by the web server, rather than by code running on the browser. Also the panel now permits your browser to perform auto-completion functionality. If you change the father's name and the previous value was the same as the informant's name, then the informant's name is also updated.

## Bug Fixes

- The class `RecOwners` included administrators who were also owners of `Individuals` in the database twice in the e-mail list generated for `Contact Author`.
- The table `Users` was not included in the list of database tables handled by the PHP base class `Record`.
- The PHP class `Record` normally prevents non-subscribers from updating any table in the database, however the new contact author functionality requires non-subscribers to update the `Blogs` table.
- The use of e-mail to implement the contact author functionality on every page resulted in the exposure of private information about subscribers.
- The table of `Users` did not have a class wrapper.
- The class `LegacyAltName` did not provide a method for creating a new ordinary alternate name.
- The class `LegacyAltName` did not provide methods for obtaining a list of alternate names or deleting a set of alternate names.
- If there were multiple “married” name records for a spouse within a single marriage some application code crashed. The redundant record[s] are now silently removed from the database.
- The class `Blog` did not provide a complete set of methods to encapsulate the functionality employed by the application code.
- When deleting an individual the alternate name records associated with that individual were not cleaned up.
- When deleting an individual any blog records associated with that individual were not cleaned up.
- When adding or updating an individual record in the database the internal time stamps were requested to be set by the server code. For clarity they are now set in the class wrapper.
- The deprecated methods `LegacyIndiv::getNumEvents`, `LegacyIndiv::getNextEvent`, `LegacyIndiv::getNumAltNames`, and `LegacyIndiv::getNextAltName` are removed.
- When adding or updating a family record in the database the internal time stamps were requested to be set by the server code. For clarity they are now set in the class wrapper.
- Method `LegacyFamily::detachChild` now returns the removed instance of `LegacyChild`. It previously returned only the key of the wrong instance of `LegacyChild`.
- The deprecated methods `LegacyFamily::getNumChildren`, `LegacyFamily::getNextChild`, `LegacyFamily::getNumEvents`, and `LegacyFamily::getNextEvent` are removed.
- The functionality of adding a child is moved from `LegacyFamily` to the `LegacyChild` constructor.
- Bad SQL was generated if there were no family related events to delete from the database.

- New static methods LegacyFamily::getFamilies and LegacyFamily::deleteFamilies added.
- LegacyFamily now uses LegacyAltName::deleteAltNames and LegacyChild::getChildren rather than accessing the database directly.
- The class LegacyCitation now silently ignores a request to create a new citation which duplicates an existing citation to the same fact.
- The class LegacyPictureList is removed, replaced by the static method LegacyPicture::getPictures.
- The method names in class LegacyPicture are simplified by removing the pointless “Pic” that was part of each method name.
- The implementation of the date associated with a picture in class LegacyPicture is changed to match the implementation used for all other dates in the family tree, so that pictures can be sorted by date.
- The class LegacyEvent has static methods getEvents and deleteEvents added to encapsulate simple queries of this table.
- LegacyChild formerly wrote records with IDIR 0 to the database.
- The LegacyChild constructor now supports directly adding a new child to a specific family.
- A new static method LegacyChild::getChildren added to eliminate application code that searched the database directly.
- New class LegacyToDo defined to encapsulate the functionality of the database table tblTD.
- The class LegacyLocation is changed to silently treat IDLR 0 as IDLR 1.
- The constructor for Marriage vital statistics did not create the dependent MarriageParticipant records when invoked with record keys.
- The method Marriage::getGroom returned the MarriageParticipant record for the bride.
- There was an error in the creation of an instance of MarriageParticipant from an associative array of field values.
- The Signon script did an unnecessary syntax verification of the entered userid and password which resulted in the rejection of logons that would have matched the information from the original registration.
- It was possible to enter the Signon script even though the user was already signed on.
- The class LegacyPictureList is removed and replaced by the static method LegacyPicture::getPictures which returns an array of instances of LegacyPicture indexed by the numeric key IDBR.
- The class BlogList is removed and replaced by the static method Blog::getBlogs which returns an array of instance of Blog indexed by the numeric key BL\_Index.
- When merging two individuals any Blog messages associated with the second individual were lost.

- In the output for the report on individuals (reqReportIndivids.php) after the conversion to CSS layout there was not enough space for some of the field labels.
- The default gender of pre-existing children did not change based upon the text entered into the given name field, but newly added children did change.
- In CensusForm.js in one case the method addRow, the onclick method of the addRow button, was invoked directly rather than through the button, which meant that the “this” pointer was not set up properly.
- When linking a Birth or Death registration to an individual the registration number was not properly padded in the citation added to the death event of the individual.
- The keyboard combination Alt-I did not work on the Death Registration Detail panel.