A Web-Based Genealogy System

Yongjie Fang and Ian Stuart

Abstract

While genealogy software products have been available for years, both as shrink-wrapped packages and more recently as Web-based systems, current approaches typically require a single user to serve as an administrator for a project. We propose a Web-based system that allows multiple users to work on a genealogical project as a group, with an administrator that determines access privileges for other users of the system. Our approach features a point-and-click interface that minimizes learning time for new users. Genealogical data are stored in a backend database, and family tree displays can be quickly and easily generated for any family member within a given tree.

Introduction

Genealogy software products have been available for many years as shrink-wrapped packages and more recently as Internet-based systems. The existent family tree software on the market today, such as Family Treemaker [3] and Cumberland Family software [4] have weaknesses primarily concerning how the information is presented and how it can be updated and maintained. They provide a user with the ability to enter genealogical data and display the information as family trees, but each approach only allows contributions from a single user. This user typically becomes an “administrator” of the tree for a family, a task that can be tedious and time-consuming.

Methodology

Our approach allows multiple users to contribute to a common genealogical project via the Web, distributing the data entry and coordination usually reserved for a single person, and using a graphical point-and-click interface that minimizes the learning time for new users and consolidates their work into a single unified tree.

Once a user of the Genealogy System establishes a password-protected account on the system, he/she can create family tree projects and grant access to any of those projects to any other user of the system that he/she specifies. Although the creator of the tree is the de facto administrator of the project, he/she is not the only individual that may contribute to its development; any user granted modification access by the administrator may add or modify elements of a tree. Thus, the laborious task of research and recording family histories can be accomplished by several users working as a team, and, of course, the team members can be physically located anywhere they can access the Internet. The administrator may optionally limit a user’s access to a tree to a read-only status that allows the display of information without modification rights.

Relationships between members of a tree are established automatically when a user adds a new member to the project. The tree “grows organically” around its first family member as each relation is introduced. Parents, children and partners are placed in the tree as relatives of members previously entered into the tree by users.

* The term partner which is more general than spouse is used herein to indicate a member representing a person that has (or had) a legal, social and/or biological relationship with another member in the same tree project. The Genealogy System provides several options for union types in addition to marriage, such as civil unions, domestic partnerships and common law arrangements.
The Genealogy System also separates data representation from presentation. Family members and their relationships are stored within the same tree, but the tree may be viewed from several perspectives. Each view may also be centered around any member within that tree. This multi-dimensional capability can encourage users to maintain a single tree for extended families rather than administering multiple trees for (say) in-laws and distant cousins. In short, if a family relationship can be established between any two people, it can be stored and represented within the same tree.

Given the private nature of the genealogical information, data security is a primary consideration. All data are stored in a backend database, accessible only to those registered users that have been granted access to the information by that tree’s creator.

**System Design**

The Genealogy System has a data-centered architecture. Data for users, trees and members are maintained in a MySQL database. The Genealogy System requires six tables to store data pertaining to the System’s users, trees, and members; control user access to trees; maintain user preferences and manage sessions.

Each tree and member are assigned unique ID values at creation. Parents, children and partners are recursively referenced by their respective ID values within the Member table. Comparing to A Web-Based Genealogy Application System of 2002 [5], our system has added more features, such as cookies for security purposes and user preferences which may be tailored by project, providing custom settings even for separate users working on the same tree. There’s also a search function in our system to help the user search particular family member or family members born in a particular year range within the user’s family tree.

**Use Case Scenario**

Any individual may become a registered user of the Genealogy System by providing a unique username, an e-mail address and an access password (Fig. 1), after which he/she may login immediately (Fig. 2). The user may create a new tree or participate in an existing tree project that he/she can access.

New trees can be created simply by selecting an option to create a new tree and providing a title for the project. The creator of the new tree becomes the *Family Tree Administrator (FTA)*, with the authority to grant or revoke access rights to other users of that tree via a window that is only available to the tree project’s administrator (Fig. 3). An FTA has the option to provide modification (read/write) access or simply the ability to view the tree (view-only) to any invited user of the Genealogy System.
Figure 1: The registration window for the Genealogy System

Figure 2: The login window
Family trees develop by adding members to a tree. Members are the summary information that represents a person within a genealogical construct. When a new tree is created, the Genealogy System prompts the user to enter information (Fig. 4) for the first member of the tree, such as name, date of birth, date of death, etc. At a minimum, first name and gender must be provided for each new family member, and dates are checked for validity (days per month, leap years, invalid characters, etc.) if they are provided.

Figure 3: The administrator’s window

Figure 4: A data entry window for a new member of a tree project.

*The elements of date fields (months, days, years) for birth, deaths, unions and divorces are entered separately to reduce data entry errors, but also because of a problem often encountered in genealogical research: dates may be unknown or incomplete. The Genealogy System allows users to provide partial information, such as a birth year without a birthday or birth month.
After successful validation, each member of a tree can be displayed within a member window containing all of the data fields for that member, and the content of those fields may be modified and saved by the user (Fig. 5). Marital and partnership information (dates, places and types of unions and divorces) may be entered if those partnerships have been defined.

Users may add additional members to a tree project by specifying the relationships (parent, child or partner) between the new member and an existing member already within the project. The System automatically defines the relational connections between members as each new member is inserted into the database. This approach prevents the creation of “orphan” members separated from the rest of a tree, and allows any tree to be represented as a multi-dimensional structure; the storage of the data are not restricted by visual limitations of a traditional family tree layout.

A user may elect to “jump” to any parent, child or partners of the member directly. (Fig. 6). There is no theoretical limit to the number of relationships to which a member may be connected.

The member window also provides the user with a menu to display ancestor and descendant trees in windows. Each tree display, the visual representation of a family tree, centers upon that member, sometimes referred as the focus or the ego in genealogical terms (Fig. 7).
Any member of the tree can be represented as the ego of its own tree. This provides the user with a variety of perspectives for the same tree. For example, the descendant tree of a member (Fig. 8) will contain a different set of related members within its tree display than the ancestor tree of one of those descendants (Fig. 9), although some intersection of the trees will exist.

4.6
Ancestral trees are conceptually simple; they represent a direct lineage of the tree display’s ego, from parents to grandparents and so on.

Descendant trees, however, use a recursive algorithm to generate a depth-first search of tree projects. Tree members are represented by PHP objects which provide methods to identify parent, children and partner objects. Each partner of the user-designated ego are identified and ordered by union date, and each child produced from each respective partnership is similarly identified and ordered by birth date. The partners of each child, in turn, are identified, and the process repeats itself until all branches of a family tree terminate at members without children. The user may optionally specify a maximum generational depth for tree displays.

A user may “climb” through a tree in three ways:

- Any branch of a tree may be navigated by clicking on a member’s name within a tree display and then displaying the tree for that member. The tree displays themselves are interactive; clicking on the name of any member within a tree produces a window for that member.
• A user may “jump” from one member to another through the member windows. The information for parents, children and partners of a member, if they exist, can each be retrieved directly from that member, without displaying a tree.

• A user may use the System’s searching feature to identify members of a tree based upon criteria he or she provides.

**Compatibility and Persistence**

All pages are scripted in PHP4 [1, 2] on a Windows NT server. The Genealogy System has been tested using Internet Explorer and Netscape Navigator.

Persistence for the web-based Genealogy System is managed by cookies that place a unique session identifier on each user’s system. No user-based or genealogical data are stored as cookies. All browser requests for user, tree and member data are authenticated by the server.

**Conclusions and Suggestions for Future Work**

Our genealogy Web application can be used anywhere one can connect to the Internet. The application successfully distributes the burden of updating and maintaining the family tree by allowing other specified family members to modify the family tree. The administrator of the family tree grants the modification right to the limited family members, and these family members make their changes remotely through Web pages. The system enables a family spread around the world to work on the family tree together.

The Genealogy System, as proposed, is only a first concept, and substantial opportunities exist to expand its functionality: the importing and exporting of genealogical data into standard formats, e-mail functions that facilitate administrative invitations, online help features and advanced searching features are just a few possibilities.

Further, we think the system would have potential commercial value. The System could be positioned as a subscription service, charging for the use of data storage and access, or as a packaged system for use on LANs and intranets. Two segments of users may be considered, and versions of a Genealogy System could be tailored to their interests. The first segment would be comprised of casual users, such as family members that need a convenient and intuitive tool to create simple trees and family histories. The second segment would consist of sophisticated researchers that want a versatile system that they can access anywhere; this group may be willing to pay for a system that is robust enough for their demanding requirements.

**References**

[2] [www.php.net](http://www.php.net)
[3] [www.genealogy.com](http://www.genealogy.com)
[4] [www.cf-software.com](http://www.cf-software.com)