GRFT – genetic records family tree web applet

Samuel Pimentel*, Virginia Walbot and John Fernandes

Department of Biology, Stanford University, Stanford, CA, USA

Current software for storing and displaying records of genetic crosses does not provide an easy way to determine the lineage of an individual. The genetic records family tree (GRFT) applet processes records of genetic crosses and allows researchers to quickly visualize lineages using a family tree construct and to access other information from these records using any Internet browser. Users select from three display features: (1) a family tree view which displays a color-coded family tree for an individual, (2) a sequential list of crosses, and (3) a list of crosses matching user-defined search criteria. Each feature contains options to specify the number of records shown and the latter two contain an option to filter results by the owner of the cross. The family tree feature is interactive, displaying a popup box with genetic information when the user mouses over an individual and allowing the user to draw a new tree by clicking on any individual in the current tree. The applet is written in JavaScript and reads genetic records from a tab-delimited text file on the server, so it is cross-platform, can be accessed by anyone with an Internet connection, and supports almost instantaneous generation of new trees and table lists. Researchers can use the tool with their own genetic cross records for any sexually reproducing organism. No additional software is required and with only minor modifications to the script, researchers can add their own custom columns. GRFT's speed, versatility, and low overhead make it an effective and innovative visualization method for genetic records. A sample tool is available at http://stanford.edu/ walbot/grft-sample.html.

Keywords: pedigree, family tree, records visualization

INTRODUCTION

Researchers whose model organism reproduces sexually must keep track of each cross performed in order to trace lineages, determine the genetic make-up of specific individuals, and track the progeny of an individual. The most common choices are Microsoft Excel and general purpose database applications such as MS Access or FileMaker Pro because of wide availability, familiarity of use, and, in the case of Excel, an overlap with the data gathering, and analysis features. Although these general use tools allow searches to find crosses of interest, determining the lineage of an individual requires multiple lookups, patience, and often a sheet of paper to outline a clear graphical pedigree or "family tree." In these situations, a computer-generated family tree display for an individual would be extremely useful, particularly if combined with the ability to redraw the tree using any individual within the currently displayed tree. Some software packages to display family trees already exist, including Pedigraph, PediTree, and CraneFoot. However, these tools generate static pictures of family trees rather than providing an interactive interface for exploring them and generally require local installation (Van Berloo and Hutten, 2005; Mäkinen, 2006; Garbe and Da, 2008).

To address the need for an easily accessible and interactive family tree tool, we have developed a JavaScript applet named genetic records family tree (GRFT), that requires no additional software or database installations. The tool reads a tab-delimited file (easily generated from any of the general use applications mentioned above) with each cross as a separate record containing the male

and female parents. A user can then select from three options: (1) generate a color-coded family tree based on a specific individual showing multiple parental generations and any F1 crosses, (2) search for keywords to produce a list of individuals matching those terms, and (3) generate a sequential list of individuals for a certain generation. Important features of this new tool are that no commercial software is required and that it is readily customized to fit specific needs. This allows researchers to host the script on their own web servers and adapt it as necessary with minimal effort. All of the implementation steps are listed under Section "Materials and Methods."

MATERIALS AND METHODS

The genetic records data displayed by the tool are read from a tab-delimited text file created by the researcher and saved on a web server. The JavaScript makes an Ajax request to read the genetic records data from the web server when the page first loads. The data are kept in memory by the script to allow quick redrawing of new trees and lists.

The JavaScript code employs a drawing object that uses CSS to create boxes and lines for the family tree. The JavaScript graphics library providing the box and line functionality is written by David Flanagan and was originally distributed as part of his book *JavaScript: The Definitive Guide* from O'Reilly Press.

Besides the graphics library, the main JavaScript (grft.js) depends upon two other helper scripts, split.js by Steven Levithan and attributes.js (which also requires the file attributes.htc) by Paul

Sowden. Both adjust for discrepancies in JavaScript interpretation between different browsers and ensure functionality is the same on all browsers. The applet was tested on a variety of browsers including Internet Explorer (version 7 and above), Mozilla Firefox, Google Chrome, and Apple Safari.

The steps to install the GRFT applet are simple. First, in order to use the GRFT applet for his or her own data, a researcher should save the JavaScript source file (grft.js, available as File S1 in Supplementary Material in this report), the associated HTML, and CSS files (grft-sample.html and grft.css, available as Files S3 and S4 in Supplementary Material in this report), and the additional library files mentioned above (CSSdrawing.js, split.js, attributes.js, and attributes.htc – web access information is provided at the end of the section Files in Supplementary Material) to his or her web server. Second, the user must generate a data file of his or her genetic records and save it on the same web server. Third, the first line of the main script defines a variable named "filename" that gives the name of the data file to be read from the server. A researcher should redefine this variable to point to his or her own data file once both files are on the server. At this point a user should be able to access the tool by opening the HTML file in a JavaScript-compatible web browser.

The data file should be in tab-delimited text format and each row should contain values for the five built-in columns in order (name, male parent, female parent, description, and owner) followed by values for any custom columns. It should not contain column headings. If an individual is missing data for certain columns, it can generally be omitted without problems but may limit how the data can be displayed. For example, the male and female parent names need to be included and match an individual name from a prior generation in order to generate an accurate and complete family tree. The records should be organized sequentially with the oldest individuals at the beginning and the newest at the end so that the list function displays them in the correct order.
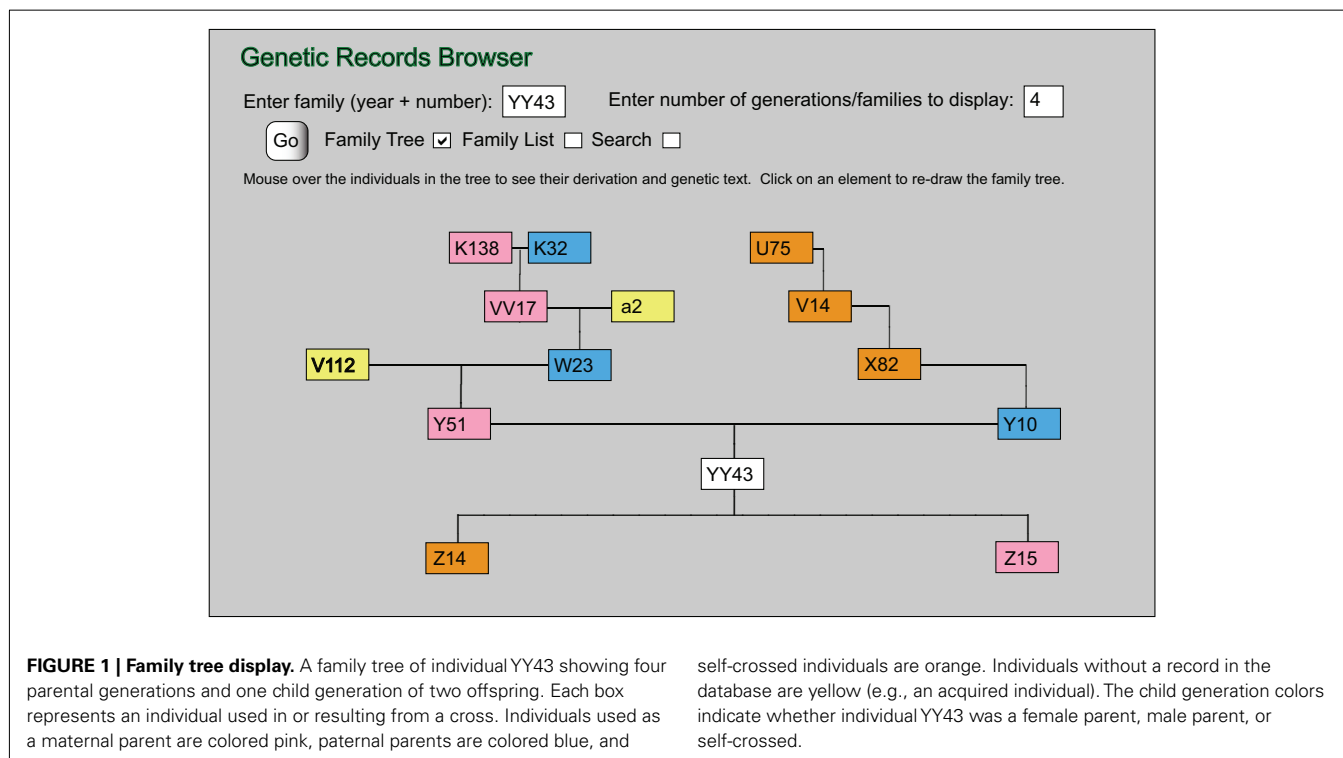
Further customization is also possible. In the method getRecordList near the end of the script all the columns in the data file (male parent, female parent, etc.) are defined and named. These built-in columns should not be removed, but a user can add additional lines to define custom columns. A comment in the script at this location gives more detailed instructions.

RESULTS

The input file of genetic records data contains a unique name or identifier for each individual as well as its male and female parents. In addition, it normally includes a short description of the individual and the name of the person who conducted the cross. The tool loads the file into memory on startup and displays a message indicating the number of records read. At this point the user can access the data using any of the three display methods: family tree, list, or search.

FEATURE: FAMILY TREE DISPLAY

The family tree feature requires two parameters – the query individual (i.e., generation 0) and the number of historical generations to include. Although the screen size can be limiting, up to 4 parental generations can normally be displayed with no overlapping branches. A single row at the bottom shows the individuals for which the query individual was the male or female parent (**Figure 1**).

FIGURE 1 | Family tree display. A family tree of individual YY43 showing four parental generations and one child generation of two offspring. Each box represents an individual used in or resulting from a cross. Individuals used as a maternal parent are colored pink, paternal parents are colored blue, and self-crossed individuals are orange. Individuals without a record in the database are yellow (e.g., an acquired individual). The child generation colors indicate whether individual YY43 was a female parent, male parent, or self-crossed.
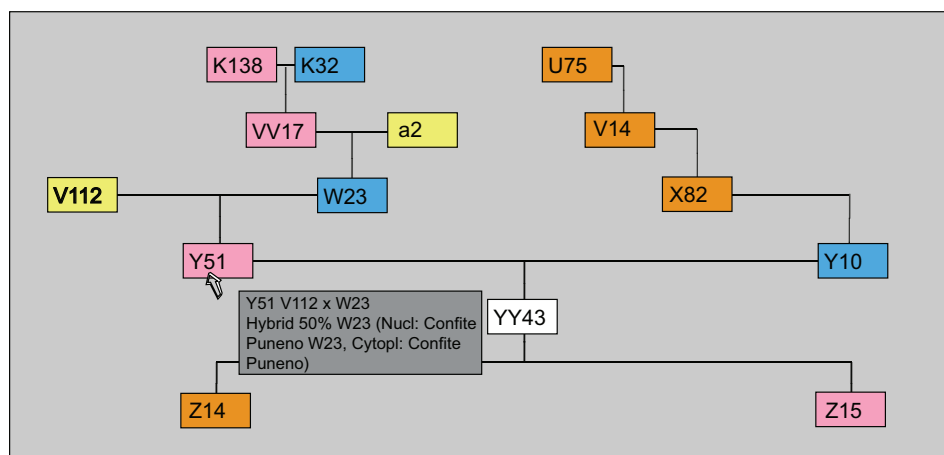
FIGURE 2 | Detail of pop-up box feature of the family tree display. A pop-up box with a gray background is shown for individual Y51, the maternal parent of YY43, is shown. Mousing over an individual generates a pop-up.

Individuals acting as female parents in the pedigree are colored pink, individuals acting as male parents are colored blue, and individuals that have been self-crossed are colored orange. The query individual is white. A yellow individual indicates a parent whose information is not in the data file (e.g., an acquired line).

For the progeny of the query individual, color reflects the parental role of the query individual in the cross. For example, a pink child indicates that the query individual acted as the female parent (see **Figure 1**).

When the user clicks on an individual in the tree, the tree is immediately redrawn with that individual as the query individual (generation 0). As shown in **Figure 2**, hovering the mouse pointer over an individual on the tree diagram will open a pop-up window containing more information on the individual, including the description if present.

Each individual in the tree is shown below in a table containing all the genetic information supplied in the data file (see **Figure 3**). The table is organized by order of increasing parental generation with the list of any F1 progeny at the end. Rows in the table can be selected and copy/pasted into a document editing or spreadsheet application.

The code in the first column, labeled "Lvl," indicates the parental generation with a "P" followed by the generation number. In each row, the columns following "Lvl" are color-coded the same as in the tree, so the color indicates the male and female parent for an individual in the next (more recent) generation. In contrast, the color of the "Lvl" column shows whether that individual is from the maternal or paternal side of the tree with respect to the query individual. For example, **Figure 3** shows three rows with a "Lvl" code of "P2." The first two such rows, for individuals V112 and W23, have the "Lvl" column colored pink to indicate they are from the maternal side of the query individual's family tree (in this case the parents of the query individual's mother). The third row, for individual X82, is colored blue in the "Lvl" column to indicate it is from the paternal side of the query individual's family tree and

the rest of the columns are colored orange to indicate this individual was used in a self-cross to generate individual Y10 (the query individual's father) in the P1 generation.

To avoid printing duplicate sections of a tree, individuals that would otherwise appear in more than one place in the tree are replaced with a code of one or more asterisks. **Figure 4** shows an example of a duplicated branch representing the parents of individual SS66. This branch is printed once on the paternal side of the tree and marked there with a single asterisk, with the subsequent duplicate branch on the maternal side replaced by the single asterisk. Each distinct duplicated branch is marked with a different number of asterisks as needed (for example the distinct lineages of individuals SS64 and QH63 in **Figure 4** are marked by sets of two and three asterisks respectively).
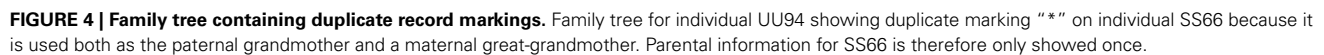
FEATURE: SEQUENTIAL LIST OF INDIVIDUALS
Selecting the "list" option will generate a list of individuals based on up to three user parameters. **Figure 5** shows a list example with the standard input file fields. The first parameter is the name of an individual in the data file and provides a starting point for the list. The second parameter is the number of individuals from that generation to include in the list. Individuals will be listed in the same order as they were recorded in the input file. The third, optional parameter allows the user to choose an "owner" (person who has conducted crosses) from a drop-down menu and filter the results so that only the crosses conducted by that individual are displayed.

FEATURE: LIST OF INDIVIDUALS BASED ON A KEYWORD SEARCH
The search feature examines all fields in the data for one or more given search terms as illustrated in **Figure 6**. It is case sensitive and will return results in which the search term is a fragment of a longer word. As with the list, the results can be filtered by owner. Multiple keywords are separated by spaces and the search can either return records with all of the keywords (using the AND option) or with any of the keywords (using the OR option).

Lvl	Individual	Female Parent	Male Parent	Description	Owner
F0	YY43	Y51	Y10	Confite Puneno 75% W23 (Nucl: W23 Confite Puneno, Cytopl: Confite Puneno)	
P1	Y51	V112	W23	Hybrid 50% W23 (Nucl: W23 Confite Puneno, Cytopl: Confite Puneno)	
P1	Y10	X82	X82	bz2 (Nucl: W23 , Cytopl: W23)	
P2	V112				
P2	W23	VV17	a2	a2-mu2/a3 some big Bz1 (Nucl: W23 WK Rob Coop, Cytopl: WK)	
P2	X82	V14	V14	bz2 R-g (Nucl: W23, Cytopl: W23)	
P3	VV17	K138	K32	a2-mu3/a2 Bz1/bz1-mu1/bz1 (Nucl: W23 WK Rob Coop, Cytopl: WK)	
P3	a2				
P3	V14	U75	U75	bz2 R-g (Nucl: W23, Cytopl: W23)	
P4	K138	JJ48	JJ47	bz1-mu1/Bz1 or bz1/Bz1 a2-mu3/A2 (Nucl: W23 WK Rob Coop, Cytopl: WK)	
P4	K32	E83	bulk	a2 Pr Bt (Nucl: Coop, Cytopl: Coop)	
P4	U75	SS66	SS66	bz2 R-g (Nucl: W23, Cytopl: W23)	
F1	Z14	YY43	YY43	Confite Puneno 75% W23 pink anther (Nucl: W23 Confite, Cytopl: Confite)	
F1	Z15	YY43	YY54	Confite Puneno 87.5% W23 (Nucl: W23 Confite, Cytopl: Confite)	

FIGURE 3 | Color-coded table provided with the family tree display. Table of all individuals charted in the family tree for individual YY43. Row color coding matches the boxes in the family tree drawing.

FIGURE 4 | Family tree containing duplicate record markings. Family tree for individual UU94 showing duplicate marking "*" on individual SS66 because it is used both as the paternal grandmother and a maternal great-grandmother. Parental information for SS66 is therefore only showed once.

DISCUSSION

A major advantage of the GRFT tool is its speed. Even with over 15000 founding individuals in our data file (representing hundreds of thousands of crosses), the individual lists, family trees, and searches can be called up almost instantaneously after the input file is read into memory when the page first loads (a process which usually requires several seconds, depending on the connection speed). This speed is possible because the full set of records resides in memory and the graphics are implemented entirely with HTML and CSS. Combined with the feature that allows users to redraw the tree by clicking on individuals within it, GRFT's speed allows researchers to explore and interact with

Genetic Records Browser

Enter family (year + number): Z10   Enter number of generations/families to display: 12

Go   Family Tree ☐   Family List ☑   Search ☐

Lvl	Individual	Female Parent	Male Parent	Description	Owner
1	Z10	Y51	Y10	Confite Puneno 75% W23 (Nucl: W23 Confite Puneno, Cytopl: Confite Puneno)	V Walbot
2	Z11	V112	W23	Hybrid 50% W23 (Nucl: W23 Confite Puneno, Cytopl: Confite Puneno)	V Walbot
3	Z12	X82	X82	bz2 (Nucl: W23 , Cytopl: W23)	G Nan
4	Z13	X82	W23	bz2 (Nucl: W23, Cytopl: W23)	G Nan
5	Z14	VV17	a2	a2-mu2/a3 some big Bz1 (Nucl: W23 WK Rob Coop, Cytopl: WK)	V Walbot
6	Z15	V14	V14	bz2 R-g (Nucl: W23, Cytopl: W23)	V Walbot
7	Z16	YY44	YY44	W23 Mischa Ecuador 75% W23 (Nucl: W23 Mischa Ecuador, Cytopl: Mischa E.)	V Walbot
8	Z17	X81	X81	bz2	V Walbot
9	Z18	U75	U75	bz2 R-g (Nucl: W23, Cytopl: W23)	D Skibbe
10	Z19	JJ48	JJ47	bz1-mu1/Bz1 or bz1/Bz1 a2-mu3/A2 (Nucl: W23 WK Rob Coop, Cytopl: WK)	D Skibbe
11	Z20	E83	bulk	a2 Pr Bt (Nucl: Coop, Cytopl: Coop)	V Walbot
12	Z21	SS66	SS66	bz2 R-g (Nucl: W23, Cytopl: W23)	V Walbot

FIGURE 5 | Sequential list display. Table showing 12 individuals starting from individual Z10 and listed in the order they are found in the data file. Table columns are customizable.

Genetic Records Browser

Start searching at year: ZI1   Enter owner to filter results: G Nan

Enter search keywords: mac1 am1 afd1   OR ☑   AND ☐

Go   Family Tree ☐   Family List ☐   Search ☑

Lvl	Individual	Female Parent	Male Parent	Description	Owner
1	ZI371	ZH625	ZH627	bz2-mu2//bz2 x afd1//+	G Nan
2	ZI372	ZH625	ZH627	bz2-mu2//bz2 x afd1//+	G Nan
3	ZI379	X82	X82	mac1//+, bz2-mu2//bz2 x mac1//+	G Nan
4	ZI383	X82	W23	RMu x afd1//+	G Nan
5	ZI384	VV17	ZH627	RMu x afd1//+ (Nucl: W23, Cytopl: not entered)	G Nan
6	ZI385	V14	V14	am1-pra//+ x am1-pral+	G Nan
7	ZI387	YY44	YY44	am1-489//+ x am1-489//+	G Nan
8	ZI388	X81	X81	bz2-mu2//bz2 x afd1//+	G Nan
9	ZI389	U75	U75	RMu x afd1//+	G Nan
10	ZI390	JJ48	JJ47	am1-489/+ or +//+ x am1-489//+	G Nan
11	ZI705	ZH642	ZH642	am1-pral//+, bz2-mu2//bz2 x am1-pral//+	G Nan
12	ZI708	SS66	SS66	am1-pra//+, bz2-mu2//bz2 x am1-pral//+	G Nan

FIGURE 6 | Search display. Table showing search results of 15 records having either "mac1" or "am1" or "afd1" keywords. Table columns are customizable.

lineage in an unprecedented way. Other software packages for pedigree visualization, which generate static images and require more complicated commands to create them, do not support this level of interactivity.

The GRFT applet also makes genetic cross information accessible from any computer. The only tool required to run the tool is an Internet browser with support for JavaScript and Ajax requests, a feature included in all modern browsers, including Internet Explorer version 7 and higher. Hence the large majority of users are not required to download or install any additional software. This also distinguishes GRFT from other pedigree generation software, which must be installed locally in order to run with full functionality (Mäkinen, 2006).

The applet can be used with any sexually reproducing organism. All that it requires is parental information provided in a tab-delimited text file, which is easily generated using Microsoft Excel or any database or scripting program. With a slight modification to the script (described later), users can include additional columns with useful information such as the inbred or hybrid line, the cytoplasmic designation of the maternal parent, or genotype information.

One limitation of the GRFT tool is its lack of support for modifying genetic records. Currently any changes to the genetic records must be entered in the original application containing the records. Afterward, the complete set of records must be re-exported to the web server. This makes updating the data file cumbersome. In future releases of the applet, however, features could be extended to allow updates from users.

A sample web tool is available at http://stanford.edu/w̄albot/grft-sample.html.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at http://www.frontiersin.org/plant_genetics_and_genomics/10.3389/fgene.2011.00014/abstract

File S1 | grft.js – main source file of the GRFT web applet.
File S2 | genetic_records_sample.txt – sample input file of tab-delimited genetic records.
File S3 | grft-sample.html – source file of the host webpage needed to run the GRFT applet.
File S4 | grft.css – style sheet controlling the format and appearance of text and images in the GRFT applet.

Additional libraries used:
CSSDrawing.js – main graphics library defining the box and line elements used to construct the family tree images. Written by David Flanagan. Available at http://www.davidflanagan.com/javascript5/display.php?n=22-6&f=22/CSSDrawing.js. Published originally in *JavaScript: The Definitive Guide*, 5th Edition, Copyright 2006 O'Reilly Media, Inc. (ISBN #0596101996).

split.js – helper script that standardizes behavior of the JavaScript split() function which is normally interpreted differently by different browsers. Written by Steven Levithan. Available at http://blog.stevenlevithan.com/archives/cross-browser-split.

attributes.js and attributes.htc – helper script designed to resolve an inconsistency in the way different browsers set style attributes. Written by Paul Sowden. Available at http://delete.me.uk/2004/09/ieproto.html (paste the sections of code written out in the blog post into files named attributes.js and attributes.htc respectively).

REFERENCES

Garbe, J. R., and Da, Y. (2008). *Pedigraph User Manual Version 2.4*. Department of Animal Science, University of Minnesota, St. Paul.

Mäkinen, V. (2006). *CraneFoot v3.2 User's Guide*. Laboratory of Computational Engineering, Helsinki University of Technology, Helsinki.

Van Berloo, R., and Hutten, R. C. B. (2005). Peditree: pedigree database analysis and visualization for breeding and science. *J. Hered.* 96, 465–468.